A Matrix-array Form for the Multidimensional Discrete Poisson Equation and its Solvability Criterion

WANG Tong, GE Yaojun, CAO Shuyang
(State Key Laboratory for Disaster Reduction in Civil Engineering, Tongji University, ShangHai 200092)

Abstract: The multidimensional discrete Poisson equation (MDPE) frequently encountered in science and engineering can be expressed, in many cases, as a brief matrix-array equation firstly defined in this paper. This new-style equation consists of a series of small matrices and can be transformed using the Kronecker sum into a familiar system of linear algebraic equations, AX=b. Then it is proved that the eigenvalues and corresponding eigenvectors of A can be obtained directly from those of these small matrices consisting in that matrix-array equation. Based on this connection, a solvability criterion for the matrix-array equation is proposed. Finally, an application of this criterion is carried out, and an inspiration from the above connection are presented and analyzed.

Keywords: Matrix-array equation; Multidimensional discrete Poisson equation; Solvability criterion; Kronecker sum; Eigenvalue; Eigenvector

0 Introduction

The motivation of this paper comes from an attempt of modeling the two-dimensional (2D) lid-driven cavity flow by applying the SIMPLE-GDQ method proposed by Shu et al.[1-4] to solve the viscous incompressible Navier-Stokes equations in primitive variable form. The SIMPLE-GDQ method is a combination of the SIMPLE algorithm and the GDQ method. So it is still an iterative method comprising several steps one out of which is the computation of pressure correction. The pressure correction is essentially governed by a Poisson equation. In the SIMPLE-GDQ method, the pressure correction equation is discretized on a non-staggered grid using the GDQ method, i.e.

$$\sum_{k=1}^{N} W_{i,k}^{(2)} p'_{k,j} + \sum_{k=1}^{M} \overline{W}_{j,k}^{(2)} p'_{i,k} = S_{i,j}^{*} \big/ \Delta t \,, \tag{0.1}$$

where $W_{i,k}^{(2)} = \sum_{k1=2}^{N-1} w_{i,k1}^{(1)} w_{k1,j}^{(1)}$, $\overline{W}_{j,k}^{(2)} = \sum_{k1=2}^{N-1} \overline{w}_{j,k1}^{(1)} \overline{w}_{k1,k}^{(1)}$. Equation (0.1) comes from the paper of Shu et al. [4]. p' is the pressure correction to be calculated. N and M are the grid numbers in the x- and y-direction, respectively. $S_{i,j}^{*} / \Delta t$ can be considered as a source term that is known. $w_{i,k1}^{(1)}$ and $\overline{w}_{i,k1}^{(1)}$ are weighting coefficients of the first-order derivatives with respect to x and y, respectively. Details can be found in [4].

In fact, Equation (0.1) has considered the Neumann boundary conditions for the pressure correction p', i.e.

$$\frac{\partial p'}{\partial x} = 0 \quad \text{for} \quad i = 1, N \,, \tag{0.2}$$

$$\frac{\partial p'}{\partial y} = 0 \quad \text{for} \quad j = 1, M \,. \tag{0.3}$$

So the remaining Dirichlet boundary conditions for p', i.e.

$$p' = 0 \quad \text{for} \quad i = 1, N \text{ and } j = 1, M \,, \tag{0.4}$$

Foundations: the National Natural Science Foundation of China (No. 50978202); the NSFC-JST Cooperative Research Project (No. 51021140005)

Brief author introduction: WANG Tong, (1981-), Male, Ph.D. Candidate, Wind resistance and CFD. E-mail: wangtong.tju@gmail.com

can be applied directly. Then, Equation (0.1) can be solved by direct or iterative techniques such as LU decomposition or SOR [1] with considering these Dirichlet boundary conditions of (0.4). However, wrong or divergent solutions of pressure correction are obtained in our applications. Trial-and-error learning indicates that considering two boundary conditions (Dirichlet and Neumann) at each boundary for p' will lead to an ill-conditioned or singular coefficient matrix for the discrete pressure correction equation. From the theory of partial differential equations we can also get that it is over-specified to consider two boundary conditions at each boundary for p', because the pressure correction equation is just two-order. This issue will be detailedly analyzed in Section 4. It is the above problem that inspires us to study the solvability of the multidimensional discrete Poisson equation (MDPE).

In the next section, we will present formulations for the MDPE defined in a regular domain. In Section 2, a definition on the multiplication of a square matrix and an array is given to transform the MDPE into a matrix-array equation including a series of small matrices. Section 3 presents and proves the connection between the eigenvalues and corresponding eigenvectors of the coefficient matrix of the MDPE and those of the series of small matrices consisting in the matrix-array form of the MDPE, and then a criterion for the matrix-array equation is given based on this connection. In Section 4, this criterion is used to explain the problem of Equation (0.1), and an inspiration from the above connection is also discussed. Finally, some conclusions are presented in Section 5.

1 The Multidimensional Discrete Poisson Equation

For simplicity, consider a d-dimensional ($d\geq 2$) Poisson equation defined in a regular domain or an irregular domain that can be transformed into a regular one, denoted by $\mathbf{\Omega}\subset\mathbf{R}^d$. This Poisson equation can be presented as [5-6]

$$\Delta u(\mathbf{x})=\sum_{i=1}^{d}\frac{\partial^2 u}{\partial x_i^2}=f(\mathbf{x}) \qquad \text{in } \mathbf{\Omega}, \tag{1.1}$$

where $\mathbf{x}=(x_1, x_2, \ldots, x_d)$ are the coordinate variables; $f(\mathbf{x})$ is the source function known in $\mathbf{\Omega}$, and clearly, if $f(\mathbf{x})=0$, Equation (1.1) becomes a d-dimensional Laplace equation; $u(\mathbf{x})$ is the unknown function in $\mathbf{\Omega}$ and it satisfies

$$au(\mathbf{x})+b\frac{\partial u(\mathbf{x})}{\partial \mathbf{n}}=g(\mathbf{x}) \qquad \text{on } \partial\mathbf{\Omega}, \tag{1.2}$$

in which $\partial\mathbf{\Omega}$ is the boundary of $\mathbf{\Omega}$; $\partial u(\mathbf{x})/\partial\mathbf{n}$ is the directional derivative in the direction normal to the boundary $\partial\mathbf{\Omega}$; $g(\mathbf{x})$ is given on $\partial\mathbf{\Omega}$; a and b are two constants, though variable coefficients are also possible. If b=0, then a Dirichlet problem is obtained. Alternatively, a=0 results in a Neumann problem. A third possibility is $a\neq 0$ with $b\neq 0$, corresponding to a mixed boundary problem. It has been proved that the Poisson equation has a unique gradient of the solution for the above three kinds of boundary value problems in the literatures [6-7].

Many physical situations such as gravitation, heat transfer, hydrodynamics, electromagnetism, acoustics, and so on, can be modeled by the combination of Equations (1.1)-(1.2) [8]. The problem presented in Section 0 is just an example in hydrodynamics. However, it is difficult or even impossible to get an analytical solution for it in most cases, especially for high-dimensional situations. Then a numerical solution will be an alternative. The finite difference (FD) method of low order is the most widely-used for the numerical solution of the Poisson equation [9], while the high-order technique, typically as the differential quadrature (DQ) method, is also familiar [1-4, 10]. For convenience, only the Dirichlet problem is considered subsequently to draw out the MDPE, and it is similar for the other two cases.

Let n_k denote the grid number in the kth variable direction of x_k (k=1,2,…,d), and let x_{k,i_k} represent the i_kth discrete coordinate in that direction (i_k=1,2,…,n_k and k=1,2,…,d). Using either the FD method or the DQ method to spatially discretize Equation (1.1) and Equation (1.2) with b=0 arrives at the MDPE and corresponding boundary conditions for the Dirichlet problem, respectively, i.e.

$$\sum_{l=1}^{n_1} c_{i_1,l}^{(1)} u_{l,i_2,\cdots,i_d} + \sum_{l=1}^{n_2} c_{i_2,l}^{(2)} u_{i_1,l,\cdots,i_d} + \cdots + \sum_{l=1}^{n_d} c_{i_d,l}^{(d)} u_{i_1,i_2,\cdots,l} = f_{i_1,i_2,\cdots,i_d}, \tag{1.3}$$

$$\begin{array}{llllll} au_{1,i_2,\cdots,i_d} = g_{1,i_2,\cdots,i_d} & \text{and} & au_{n_1,i_2,\cdots,i_d} = g_{n_1,i_2,\cdots,i_d} & \text{for} & x_1, \\ au_{i_1,1,\cdots,i_d} = g_{i_1,1,\cdots,i_d} & \text{and} & au_{i_1,n_2,\cdots,i_d} = g_{i_1,n_2,\cdots,i_d} & \text{for} & x_2, \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ au_{i_1,i_2,\cdots,1} = g_{i_1,i_2,\cdots,1} & \text{and} & au_{i_1,i_2,\cdots,n_d} = g_{i_1,i_2,\cdots,n_d} & \text{for} & x_d, \end{array} \tag{1.4}$$

for i_k=1,2,…,n_k and k=1,2,…,d, where $c_{i_k,l}^{(k)}$ is the discrete (or weighting) coefficient for the variable x_k, and $\varphi_{i_1,i_2,\cdots,i_d} = \varphi(x_{1,i_1}, x_{2,i_2}, \cdots, x_{d,i_d})$ in which φ=u, f, g.

Substituting Equation (1.4) into (1.3) yields

$$\sum_{l=2}^{n_1-1} c_{i_1,l}^{(1)} u_{l,i_2,\cdots,i_d} + \sum_{l=2}^{n_2-1} c_{i_2,l}^{(2)} u_{i_1,l,\cdots,i_d} + \cdots + \sum_{l=2}^{n_d-1} c_{i_d,l}^{(d)} u_{i_1,i_2,\cdots,l} = f'_{i_1,i_2,\cdots,i_d}, \tag{1.5}$$

for i_k=2,3,…,n_k-1 and k=1,2,…,d, where

$$f'_{i_1,i_2,\cdots,i_d} = f_{i_1,i_2,\cdots,i_d} - \left(c_{i_1,1}^{(1)} g_{1,i_2,\cdots,i_d} + c_{i_1,n_1}^{(1)} g_{n_1,i_2,\cdots,i_d} + \cdots + c_{i_d,1}^{(d)} g_{i_1,i_2,\cdots,1} + c_{i_d,n_d}^{(d)} g_{i_1,i_2,\cdots,n_d} \right) \Big/ a.$$

For simplicity, let N_k denote n_k-2 for k=1,2,…,d, and then (1.5) can be changed into

$$\sum_{l=1}^{N_1} E_{i_1,l}^{(1)} U_{l,i_2,\cdots,i_d} + \sum_{l=1}^{N_2} E_{i_2,l}^{(2)} U_{i_1,l,\cdots,i_d} + \cdots + \sum_{l=1}^{N_d} E_{i_d,l}^{(d)} U_{i_1,i_2,\cdots,l} = F_{i_1,i_2,\cdots,i_d}, \tag{1.6}$$

for i_k=1,2,…,N_k and k=1,2,…,d, in which $U_{i_1,i_2,\cdots,i_d} = u_{i_1+1,i_2+1,\cdots,i_d+1}$, $F_{i_1,i_2,\cdots,i_d} = f'_{i_1+1,i_2+1,\cdots,i_d+1}$, $E_{i_k,l}^{(k)} = c_{i_k+1,l+1}^{(k)}$.

So far, the multidimensional Poisson equation (1.1) has be discretized into (1.6), a system of N algebraic equations with N unknowns (N=N_1×N_2×…×N_d), and it can be presented briefly in the familiar form as

$$\mathbf{AX} = \mathbf{b}, \tag{1.7}$$

where **X** is the unknown column vector of $U_{i_1,i_2,\cdots,i_d}$, **b** is the known column vector of $F_{i_1,i_2,\cdots,i_d}$, and **A** is the coefficient matrix made up of $E_{i_k,l}^{(k)}$.

In practice, the coefficient matrix **A** of (1.7) may have many different but equivalent forms, one of which can be changed into another by matrix elementary transformation. And different forms have different characteristics. Besides, the size of **A** and corresponding computational complexity usually increase at an unexpected speed with the number of dimensions, an effect also coming from the curse of dimensionality [11]. So it is commonly not easy to obtain its characteristics such as the eigenvalues from which we can know the solvability of (1.6) or (1.7). In the following, we will transform Equation (1.6) into a certain brief form which is helpful to get its characteristics.

2 A Matrix-array Form for the MDPE

To facilitate the expression, a definition on the multiplication of a square matrix and a multidimensional array will be given following the definition of matrix multiplication.

Definition 2.1 *Let* **B** *be an* $M\times M$ *matrix, and let* $\mathbf{C}^{(n)}$ *be an n-dimensional* ($n\geq 1$) *array of size* $N_1\times N_2\times\ldots\times N_n$. *Suppose* **B** *is just valid to the k*th (k=1,2,…,n) *dimension of* $\mathbf{C}^{(n)}$, *and in such a case, M must be equal to* N_k. *The product of* **B** *and* $\mathbf{C}^{(n)}$ *is still an n-dimensional array of the same size as* $\mathbf{C}^{(n)}$, *denoted by* $\mathbf{D}^{(n)}$. *With* **B** *on the left and* $\mathbf{C}^{(n)}$ *on the right, the defining formulation for the matrix-array multiplication is given as follows*,

$$\mathbf{B}\mathbf{C}_k^{(n)}=\mathbf{D}^{(n)}, \tag{2.1}$$

in which the subscript k of $\mathbf{C}^{(n)}$ is used to indicate that **B** is just valid to the kth dimension of $\mathbf{C}^{(n)}$, and the element of $\mathbf{D}^{(n)}$ is calculated by

$$D_{i_1,i_2,\cdots,i_n}=\sum_{l=1}^{N_k}B_{i_k,l}C_{i_1,i_2,\cdots,i_{k-1},l,i_{k+1},\cdots,i_n}, \tag{2.2}$$

for i_j=1,2,…,N_j *and* j=1,2,…,n, *where* $B_{i_k,l}$, $C_{i_1,i_2,\cdots,i_n}$ *and* $D_{i_1,i_2,\cdots,i_n}$ *are the elements of* **B**, $\mathbf{C}^{(n)}$ *and* $\mathbf{D}^{(n)}$, *respectively*.

The $U_{i_1,i_2,\cdots,i_d}$ in Equation (1.6) can be considered as the element of a d-dimensional array, denoted by $\mathbf{U}^{(d)}$. Similarly, the $F_{i_1,i_2,\cdots,i_d}$ in Equation (1.6) can be considered as the element of another d-dimensional array, denoted by $\mathbf{F}^{(d)}$. Specially, when d=2, $\mathbf{U}^{(d)}$ and $\mathbf{F}^{(d)}$ become two matrices, **U** and **F**, respectively. With the help of (2.1), (1.6) can be transformed directly into the following matrix-array equation,

$$\mathbf{E}_1\mathbf{U}_1^{(d)}+\mathbf{E}_2\mathbf{U}_2^{(d)}+\cdots+\mathbf{E}_d\mathbf{U}_d^{(d)}=\mathbf{F}^{(d)}, \tag{2.3}$$

where $\mathbf{E}_k$ is the matrix of $E_{i_k,l}^{(k)}$ and i_k,l=1,2,…,N_k for k=1,2,…,d.

In (2.3), $\mathbf{E}_k$ (k=1,2,…,d) is only valid to the kth dimension of $\mathbf{U}^{(d)}$, so the subscript of $\mathbf{U}^{(d)}$ can be omitted without confusion, then we obtain

$$\mathbf{E}_1\mathbf{U}^{(d)}+\mathbf{E}_2\mathbf{U}^{(d)}+\cdots+\mathbf{E}_d\mathbf{U}^{(d)}=\mathbf{F}^{(d)}. \tag{2.4}$$

Essentially, Equation (2.4) is equivalent to Equation (1.7), and consequentially Equation (2.4) can be transformed into the form of Equation (1.7). While before that, to make brief statements, we will give another two definitions and a lemma that will be helpful subsequently. This lemma can be found in [12].

Definition 2.2 *Suppose* **B** *and* **C** *are two square matrices*, and *then a function p is defined as*

$$p(\mathbf{B},\mathbf{C})=\mathbf{B}\otimes\mathbf{I}_{\mathrm{C}}+\mathbf{I}_{\mathrm{B}}\otimes\mathbf{C}, \tag{2.5}$$

where $\mathbf{I}_{\mathrm{B}}$ *and* $\mathbf{I}_{\mathrm{C}}$ *are identity matrices of the same sizes as* **B** *and* **C**, respectively, *and* "⊗" *denotes the Kronecker product* [12].

Definition 2.3 *Suppose* $\mathbf{J}^{(n)}$ *is an n-dimensional array of size* $N_1\times N_2\times\ldots\times N_n$, *then it can be transformed into a column vector* vec($\mathbf{J}^{(n)}$) *following a certain rule that the change rate of the mark number* i_k *for the array members increases with k. It means that* i_{k+1} *changes faster than* i_k. *For instance*, vec($\mathbf{J}^{(3)}$) *can be expanded into*

$$\{J_{1,1,1},\cdots,J_{1,1,N_3},\cdots\cdots,J_{1,N_2,1},\cdots,J_{1,N_2,N_3},J_{2,1,1},\cdots,J_{2,1,N_3},\cdots\cdots,J_{N_1,N_2,1},\cdots,J_{N_1,N_2,N_3}\}^{\mathrm{T}}.$$

Lemma 2.1 *Suppose* **B** *and* **C** *are square matrices of sizes* $m\times m$ *and* $n\times n$, *respectively*, *and* **X**

and $\mathbf{D}$ *are matrices of the same size* $m\times n$, *then the matrix equation* $\mathbf{BX}+\mathbf{XC}=\mathbf{D}$ *can be transformed into an equivalent Kronecker form as* $p(\mathbf{B},\mathbf{C}^{\mathrm{T}})\mathrm{vec}(\mathbf{X})=\mathrm{vec}(\mathbf{D})$, *where* $\mathbf{C}^{\mathrm{T}}$ *is the transpose of* $\mathbf{C}$.

From [12], we can also get that the function p is actually used to calculate the Kronecker sum of two square matrices. With the help of Definitions 2.2 and 2.3, we propose a theorem to state the transformation from Equation (2.4) to Equation (1.7).

Theorem 2.1 *The matrix-array equation like Equation* (2.4) *can be transformed into a system of linear algebraic equations like Equation* (1.7) *by*

$$p(p(\cdots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\cdots,\mathbf{E}_{d-1}),\mathbf{E}_d)\mathrm{vec}(\mathbf{U}^{(d)})=\mathrm{vec}(\mathbf{F}^{(d)})\,. \tag{2.6}$$

Proof Step 1: for d=2, Equation (2.4) becomes

$$\mathbf{E}_1\mathbf{U}^{(2)}+\mathbf{E}_2\mathbf{U}^{(2)}=\mathbf{F}^{(2)}\,. \tag{2.7}$$

In Equation (2.7), $\mathbf{U}^{(2)}$ and $\mathbf{F}^{(2)}$ are in fact both matrices. So $\mathbf{U}^{(2)}$ can be denoted by $\mathbf{U}$, and $\mathbf{F}^{(2)}$ denoted by $\mathbf{F}$. Then Equation (2.7) can be rewritten in the form of the famous matrix equation "$\mathbf{AX}+\mathbf{XB}=\mathbf{C}$", as

$$\mathbf{E}_1\mathbf{U}+\mathbf{U}\mathbf{E}_2^{\mathrm{T}}=\mathbf{F}\,, \tag{2.8}$$

where $\mathbf{E}_2^{\mathrm{T}}$ is the transpose of $\mathbf{E}_2$.

Following Lemma 2.1, Equation (2.8) can be transformed into the following Kronecker form,

$$p(\mathbf{E}_1,\mathbf{E}_2)\mathrm{vec}(\mathbf{U})=\mathrm{vec}(\mathbf{F})\,. \tag{2.9}$$

So Equation (2.6) is true for d=2.

Step 2: assume Equation (2.6) is true for $d=n$ ($n\geq 2$), i.e.

$$\mathbf{E}_1\mathbf{U}^{(n)}+\mathbf{E}_2\mathbf{U}^{(n)}+\cdots+\mathbf{E}_n\mathbf{U}^{(n)}=\mathbf{F}^{(n)} \tag{2.10}$$

can be transformed into

$$p(p(\cdots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\cdots,\mathbf{E}_{n-1}),\mathbf{E}_n)\mathrm{vec}(\mathbf{U}^{(n)})=\mathrm{vec}(\mathbf{F}^{(n)})\,. \tag{2.11}$$

Then for $d=n+1$, Equation (2.4) becomes

$$\mathbf{E}_1\mathbf{U}^{(n+1)}+\mathbf{E}_2\mathbf{U}^{(n+1)}+\cdots+\mathbf{E}_n\mathbf{U}^{(n+1)}+\mathbf{E}_{n+1}\mathbf{U}^{(n+1)}=\mathbf{F}^{(n+1)}\,. \tag{2.12}$$

For any given $i_{n+1}=l$ ($l=1,2,\ldots,N_{n+1}$), the elements $U_{i_1,i_2,\cdots,i_n,l}$ ($i_k=1,2,\ldots,N_k$ for $k=1,2,\ldots,n$) of $\mathbf{U}^{(n+1)}$ can be considered as an array of size $N_1\times N_2\times\ldots\times N_n$, denoted by $\mathbf{U}^{(n+1)_l}$. From (2.10) and (2.11), the following expression

$$\mathbf{E}_1\mathbf{U}^{(n+1)_l}+\mathbf{E}_2\mathbf{U}^{(n+1)_l}+\cdots+\mathbf{E}_n\mathbf{U}^{(n+1)_l}$$

can be transformed into

$$p(p(\cdots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\cdots,\mathbf{E}_{n-1}),\mathbf{E}_n)\mathrm{vec}(\mathbf{U}^{(n+1)_l})\,.$$

Accordingly, we can get N_{n+1} vectors, i.e. $\mathrm{vec}(\mathbf{U}^{(n+1)_l})$ for $l=1,2,\ldots,N_{n+1}$. Then these vectors can be pilled up to form a matrix denoted by $\mathbf{U}'$, i.e.

$$\mathbf{U}'=\left[\mathrm{vec}(\mathbf{U}^{(n+1)_1}),\mathrm{vec}(\mathbf{U}^{(n+1)_2}),\cdots,\mathrm{vec}(\mathbf{U}^{(n+1)_{N_{n+1}}})\right]. \tag{2.13}$$

Following Definition 2.3, we have

$$\mathrm{vec}(\mathbf{U}')=\mathrm{vec}(\mathbf{U}^{(n+1)})\,. \tag{2.14}$$

Similar to (2.13), we can obtain $\mathbf{F}'$ from $\mathbf{F}^{(n+1)}$, and also

$$\mathrm{vec}(\mathbf{F}') = \mathrm{vec}(\mathbf{F}^{(n+1)})\,. \tag{2.15}$$

So, the first n terms of the left hand side of (2.12) have the following transformation,

$$\mathbf{E}_1\mathbf{U}^{(n+1)} + \mathbf{E}_2\mathbf{U}^{(n+1)} + \cdots + \mathbf{E}_n\mathbf{U}^{(n+1)} \Leftrightarrow p(p(\cdots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\cdots,\mathbf{E}_{n-1}),\mathbf{E}_n)\mathbf{U}'\,, \tag{2.16}$$

and the $(n+1)$th term of the left hand side of (2.12) has the following transformation,

$$\mathbf{E}_{n+1}\mathbf{U}^{(n+1)} \Leftrightarrow \mathbf{U}'\mathbf{E}_{n+1}^{\mathrm{T}}\,, \tag{2.17}$$

where $[\mathbf{E}]_{n+1}^{\mathrm{T}}$ is the transpose of $[\mathbf{E}]_{n+1}$.

From (2.16) and (2.17), Equation (2.12) can be transformed into

$$\mathbf{A}\big|_{d=n}\mathbf{U}' + \mathbf{U}'\mathbf{E}_{n+1}^{\mathrm{T}} = \mathbf{F}'\,. \tag{2.18}$$

where $\mathbf{A}|_{d=n}=p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{n-1}),\mathbf{E}_n)$. Clearly, Equation (2.18) has the same pattern as that of Equation (2.8). Following Lemma 2.1 and Equations (2.14)-(2.15), Equation (2.18) can be transformed into

$$p(\mathbf{A}\big|_{d=n},\mathbf{E}_{n+1})\mathrm{vec}(\mathbf{U}^{(n+1)}) = \mathrm{vec}(\mathbf{F}^{(n+1)})\,, \tag{2.19}$$

where $p(\mathbf{A}|_{d=n},\mathbf{E}_{n+1})= p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_n),\mathbf{E}_{n+1})$.

So Equation (2.6) is still true for $d=n+1$, and therefore it holds for any d ($d\geq 2$).

3 A Solvability Criterion for the MDPE

It has been demonstrated in the preceding section that the matrix-array Equation (2.6) can be transformed into Equation (1.7). It means that for d-dimensional situations the coefficient matrix $\mathbf{A}$ of Equation (1.7) can be represented by d small matrices $\mathbf{E}_l$ (l=1,2,…,d) consisting in Equation (2.6). And then it is natural for us to think about the relationship between the eigenvalues and corresponding eigenvectors of $\mathbf{A}$ and those of $\mathbf{E}_l$ (l=1,2,…,d). First of all, we present a lemma that will be used subsequently, and this lemma can also be found in [12].

Lemma 3.1 *If α is one of the eigenvalues of a square matrix* $\mathbf{B}$ *and* $\mathbf{g}$ *is a corresponding eigenvector of* $\mathbf{B}$, *and if β is one of the eigenvalues of another square matrix* $\mathbf{C}$ *and* $\mathbf{h}$ *is a corresponding eigenvector of* $\mathbf{C}$, *then $\alpha+\beta$ is one of the eigenvalues of* $p(\mathbf{B},\mathbf{C})$ *and* $\mathbf{g}\otimes\mathbf{h}$ *is a corresponding eigenvector of* $p(\mathbf{B},\mathbf{C})$.

We propose another theorem to relate the eigenvalues and corresponding eigenvectors of $\mathbf{E}_l$ (l=1,2,…,d) to those of $\mathbf{A}$.

Theorem 3.1 *Suppose* $\mathbf{E}_1$, $\mathbf{E}_2$, …, $\mathbf{E}_d$ *are a series of square matrices of sizes* $N_1\times N_1$, $N_2\times N_2$, …, $N_d\times N_d$, *respectively, and* $\mathbf{A}=p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{d-1}),\mathbf{E}_d)$. *If $\alpha^{(l)}$ is one of the eigenvalues of* $\mathbf{E}_l$ (l=1,2,…,d) *and* $\mathbf{g}^{(l)}$ *is a corresponding eigenvector of the corresponding* $\mathbf{E}_l$, *then* $\alpha^{(1)}+\alpha^{(2)}+\ldots+\alpha^{(d)}$ *is one of the eigenvalues of* $\mathbf{A}$ *and* $\mathbf{g}^{(1)}\otimes\mathbf{g}^{(2)}\otimes\ldots\otimes\mathbf{g}^{(d)}$ *is a corresponding eigenvector of* $\mathbf{A}$.

Proof Step 1: for d=2, suppose $\mathbf{A}|_{d=2}=p(\mathbf{E}_1,\mathbf{E}_2)$. From Lemma 3.1, we know $\alpha^{(1)}+\alpha^{(2)}$ is one of the eigenvalues of $\mathbf{A}|_{d=2}$ and $\mathbf{g}^{(1)}\otimes\mathbf{g}^{(2)}$ is a corresponding eigenvector of $\mathbf{A}|_{d=2}$. So Theorem 3.1 is true for d=2.

Step 2: assume Theorem 3.1 is true for $d=n$ ($n\geq 2$), i.e. $\alpha^{(1)}+\alpha^{(2)}+\ldots+\alpha^{(n)}$, denoted by γ, is one of the eigenvalues of $p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{n-1}),\mathbf{E}_n)$, denoted by $\mathbf{A}|_{d=n}$, and $\mathbf{g}^{(1)}\otimes\mathbf{g}^{(2)}\otimes\ldots\otimes\mathbf{g}^{(n)}$, denoted by $\mathbf{q}$, is a corresponding eigenvector of $\mathbf{A}|_{d=n}$.

Then let's study the situation of d=n+1. Suppose $\mathbf{A}|_{d=n+1}=p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_n),\mathbf{E}_{n+1})$. Following Equation (2.19), we know that

$$\mathbf{A}|_{d=n+1}=p(\mathbf{A}|_{d=n},\mathbf{E}_{n+1}). \tag{3.1}$$

From Lemma 3.1, if γ is one of the eigenvalues of $\mathbf{A}|_{d=n}$ and $\mathbf{q}$ is a corresponding eigenvector of $\mathbf{A}|_{d=n}$, and if $\alpha^{(n+1)}$ is one of the eigenvalues of $\mathbf{E}_{n+1}$ and $\mathbf{g}^{(n+1)}$ is a corresponding eigenvector of the corresponding $\mathbf{E}_{n+1}$, then

$$\gamma+\alpha^{(n+1)}=\alpha^{(1)}+\alpha^{(2)}+\cdots+\alpha^{(n)}+\alpha^{(n+1)} \tag{3.2}$$

is one of the eigenvalues of $\mathbf{A}|_{d=n+1}$ and

$$\mathbf{q}\otimes\mathbf{g}^{(n+1)}=\mathbf{g}^{(1)}\otimes\mathbf{g}^{(2)}\otimes\cdots\otimes\mathbf{g}^{(n)}\otimes\mathbf{g}^{(n+1)} \tag{3.3}$$

is a corresponding eigenvector of $\mathbf{A}|_{d=n+1}$.

So, Theorem 3.1 is still true for d=n+1, and therefore it holds for any d ($d\geq 2$).

Following Theorem 3.1, we can give a solvability criterion for the matrix-array Equation (2.4), i.e.

$$\alpha^{(1)}+\alpha^{(2)}+\cdots+\alpha^{(d)}\neq 0, \tag{3.4}$$

where $\alpha^{(l)}$ is one of the eigenvalues of $\mathbf{E}_l$ (l=1,2,…,d).

4 Applications and Discussions

In the preceding sections, a definition on the multiplication of a square matrix and an array is presented to transform the MDPE defined in a regular domain into a matrix-array equation, and then a solvability criterion for this matrix-array equation is proposed. However, this criterion is not just valid for the Equation (2.4) from the MDPE, while it can be used for any problem that can be represented by Equation (2.4). The MDPE is just used to educe Equation (2.4). From Section 1, we know that the Laplace equation is a special Poisson equation, so this criterion is undoubtedly available for the discrete Laplace equation as long as it can be transformed into the matrix-array form of Equation (2.4).

Now we use this criterion to study the problem of Equation (0.1). We know that Equation (0.1) has considered the Neumann conditions, i.e. Equations (0.2)-(0.3). Substituting Equation (0.4), the Dirichlet conditions, into (0.1) yields

$$\sum_{k=2}^{N-1}W_{i,k}^{(2)}p'_{k,j}+\sum_{k=2}^{M-1}\overline{W}_{j,k}^{(2)}p'_{i,k}=S_{i,j}^{*}/\Delta t, \tag{4.1}$$

for i=2,3,…,N-1 and j=2,3,…,M-1. Then Equation (4.1) can be written in the form of $\mathbf{AX}+\mathbf{XB}^{\mathrm{T}}=\mathbf{C}$. Here, $\mathbf{A}$ is a (N-2)×(N-2) matrix of $W_{i,k}^{(2)}$,and $\mathbf{B}$ is a (M-2)×(M-2) matrix of $\overline{W}_{j,k}^{(2)}$. $\mathbf{X}$ and $\mathbf{C}$ are (N-2)×(M-2) matrices of $p'_{i,k}$ and $S_{i,j}^{*}/\Delta t$, respectively. $\mathbf{B}^{\mathrm{T}}$ is the transpose of $\mathbf{B}$. We found that if N(or M) is odd and no matter the N(or M) grid points are equal or not, then one eigenvalue of $\mathbf{A}$(or $\mathbf{B}$) is zero or nearly zero. So, if N and M are both odd, then one eigenvalue of the coefficient matrix $p(\mathbf{A},\mathbf{B})$ of Equation (4.1) will be zero or nearly zero. It means that $p(\mathbf{A},\mathbf{B})$ is singular or ill-conditioned. In such a case, Equation (4.1) is unsolvable, or its solvability is very bad. And even if convergent results can be obtained, they are usually wrong. However, it should be noted here that Shu et al. [1-4] obtained good results using odd N and odd M, though we do not know how they got that. In our applications, we obtained satisfied results just considering the Dirichlet boundary conditions for p'. We think there is no need to consider the Neumann boundary conditions, whereas we will not continue to deepen this problem in the present paper.

Besides, the Theorem 3.1 also gives us an inspiration that if a large matrix **A** can be transformed into the form of $p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{n-1}),\mathbf{E}_n)$, i.e. the matrix **A** can be represented by n small square matrices $\mathbf{E}_l$ (l=1,2,…,n), then we can obtain the eigenvalues and corresponding eigenvectors of **A** directly from those of $\mathbf{E}_l$ (l=1,2,…,n). This would be an extraordinarily high-efficient method, because first the size of $\mathbf{E}_l$ is far smaller than that of **A**, and therefore it can save much more computational effort to obtain one eigenvalue and corresponding eigenvector for $\mathbf{E}_l$ than that for **A**. Second, there are only n small matrices, $\mathbf{E}_l$ (l=1,2,…,n), to be solved with M ($M=N_1+N_2+\ldots+N_n$) eigenvalues and corresponding eigenvectors obtained from which N ($N=N_1\times N_2\times\ldots\times N_n$) eigenvalues and corresponding eigenvectors for **A** can be determined. Third, these small matrices are independent with each other, so it will be convenient to solve them independently using the parallel algorithm. Finally, the needed storage space for $\mathbf{E}_l$ is much lesser than that for **A**, which also enhances its efficiency. From the above analysis, we can conclude that this proposed method is not only timesaving but also space-saving.

5 Conclusion

In this paper we propose a matrix-array form for the MDPE defined in a regular domain and present a solvability criterion for it. First, a definition on the multiplication of a square matrix and an array is given to change the MDPE into a brief matrix-array equation including a series of small matrices, $\mathbf{E}_l$ (l=1,2,…,n). And then it is proved that this matrix-array equation can be transformed into a system of linear algebraic equation, $\mathbf{AX}=\mathbf{b}$, and $\mathbf{A}=p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{n-1}),\mathbf{E}_n)$. Third, we propose a theorem to relate the eigenvalues and corresponding eigenvectors of these small matrices to those of **A**. Based on this theorem we give a solvability criterion for the matrix-array equation or the corresponding MDPE. Then this criterion is used to analyze the problem presented in Section 0. We found that it is not proper to consider two boundary conditions (Dirichlet and Neumann) at each boundary for the two-order pressure correction equation, because this will lead to a singular or ill-conditioned coefficient matrix. Finally, an inspiration from the Theorem 3.1 is presented, that is: if a large matrix **A** can be transformed into $p(p(\ldots p(p(\mathbf{E}_1,\mathbf{E}_2),\mathbf{E}_3),\ldots,\mathbf{E}_{n-1}),\mathbf{E}_n)$, then the eigenvalues and corresponding eigenvectors of **A** can be obtained directly from those of $\mathbf{E}_l$ (l=1,2,…,n). Analysis indicates that this method is not only timesaving but also space-saving.

References

[1] Shu C. Differential Quadrature and its Applications in Engineering[M]. London: Springer, 2000.

[2] Shu C, Wee K H A. Numerical simulation of natural convection in a square cavity by SIMPLE-generalized differential quadrature method[J]. Computers & Fluids, 2002, 31(2): 209-226.

[3] Shu C, Wang L, Chew Y T. Numerical computation of three-dimensional incompressible Navier-Stokes equations in primitive variable form by DQ method[J]. International Journal for Numerical Methods in Fluids, 2003, 43: 345-368.

[4] Shu C, Wang L, Chew Y T. Comparative Studies of Three Approaches for GDQ Computation of Incompressible Navier-Stokes Equations in Primitive Variable Form[J]. International Journal of Computational Fluid Dynamics, 2004, 18(5): 401-412.

[5] Elman H C, Silvester D J, Wathen A J. Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics[M]. New York: Oxford University Press, 2005.

[6] Polyanin A D. Handbook of Linear Partial Differential Equations for Engineers and Scientists[M]. Boca Raton: Chapman & Hall/CRC, 2002.

[7] Jackson J D. Classical Electrodynamics[M]. New York: John Wiley & Sons, 1998.

[8] Ockendon J R, Howison S D, Lacey A A, Movchan A B. Applied Partial Differential Equations[M]. Oxford: Oxford University Press, 2003.

[9] Hoffman J D. Numerical Methods for Engineers and Scientists[M]. New York: McGraw-Hill, 1992.

[10] Bert C W, Malik M. Differential quadrature method in computational mechanics: A review[J]. Applied Mechanics Reviews, 1996, 49(1): 1-28.

[11] Wilson A. Solving Poisson's Equation in High Dimensions by a Hybrid Monte-Carlo Finite Difference Method[D]. British Columbia: Simon Fraser University, 2004.

[12] Horn R A, Johnson C R. Topics in Matrix Theory[M]. Cambridge: Cambridge University Press, 1994.

多维离散 Poisson 方程的矩阵－数组形式及其可解性的判定

王通，葛耀君，曹曙阳

（同济大学土木工程防灾国家重点实验室，上海 200092）

摘要：本文首次定义了方阵与多维数组的一种乘法，进而将定义于规则区域内的多维离散 Poisson 方程转变成一种包含一系列小矩阵的矩阵－数组方程的形式。由数学归纳法证明，该矩阵－数组方程可以利用 Kronecker 和转变成常见的线性代数方程组的形式，即 AX=b。提出并证明一个定理：矩阵 A 的特征值及相应的特征向量可以直接通过包含于矩阵－数组方程中的那些小矩阵的特征值及相应的特征向量计算得到。根据这一定理，给出了矩阵－数组方程可解性的一种判定准则。最后将这一判定准则应用于一个实际问题，并深入讨论由上述定理得到的一个启示。

关键词：应用数学；矩阵－数组方程；多维离散泊松方程；可解性；Kroneker 积；特征值；特征向量

中图分类号：O29