



The application of special matrix product to differential quadrature solution of geometrically nonlinear bending of orthotropic rectangular plates

W. Chen^{a,*}, C. Shu^b, W. He^c, T. Zhong^a

^aDepartment of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, 200030, People's Republic of China

^bDepartment of Mechanical and Production Engineering, National University of Singapore, 10 Kent Ridge Crescent, 119260, Singapore

^cDepartment of Electrical Engineering, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu, 212013, People's Republic of China

Received 22 January 1997; accepted 17 November 1998

Abstract

The Hadamard and SJT product of matrices are two types of special matrix product. The latter was first defined by Chen. In this study, they are applied to the differential quadrature (DQ) solution of geometrically nonlinear bending of isotropic and orthotropic rectangular plates. By using the Hadamard product, the nonlinear formulations are greatly simplified, while the SJT product approach minimizes the effort to evaluate the Jacobian derivative matrix in the Newton–Raphson method for solving the resultant nonlinear formulations. In addition, the coupled nonlinear formulations for the present problems can easily be decoupled by means of the Hadamard and SJT product. Therefore, the size of the simultaneous nonlinear algebraic equations is reduced by two-thirds and the computing effort and storage requirements are greatly alleviated. Two recent approaches applying the multiple boundary conditions are employed in the present DQ nonlinear computations. The solution accuracies are significantly improved in comparison to the previously given by Bert et al. The numerical results and detailed solution procedures are provided to demonstrate the superb efficiency, accuracy and simplicity of the new approaches in applying DQ method for nonlinear computations. © 1999 Elsevier Science Ltd. All rights reserved.

1. Introduction

The differential quadrature (DQ) method was introduced by Bellman and his associates [1,2]. Since then, the method has been applied successfully to a broad range of problems [2–16]. A striking merit of the

method is the high efficiency in computing nonlinear problems [2,3,5,7–11,14–16]. Compared with the standard numerical techniques such as the finite element and finite difference methods, the DQ method produces solution of reasonable accuracy with relatively small computational effort. This method also does not require the trial functions satisfying boundary conditions to be sought as in the Rayleigh–Ritz and Galerkin methods [12]. Therefore, it is easily used for practical purposes. It has been found that the DQ method is closely related to the collocation (or pseudo-spectral) methods [12,17,18]. The principal advantages of the DQ method over the latter are its

* Corresponding author. Current address: Department of Mechanical Systems Engineering, Shinsu University, Nagano 380, Japan. Tel.: +81-26-226-4101 ext. 2318; Fax: +81-26-223-1924.

E-mail address: chenw@homer.shinsu-u.ac.jp (W. Chen)

simplicity and ease in using grid spacing without restriction [8,18,19].

The geometrically nonlinear behavior of thin plates is usually described by the von Karman equations and has become a benchmark problem for testing numerical solutions to nonlinear partial differential equations. Bert et al. [9] employed the DQ method to solve the static von Karman equations in analyzing geometrically nonlinear bending of isotropic and orthotropic rectangular plates. The work shows that the DQ method is an efficient numerical technique in solving complex nonlinear problems by comparing with the finite element and finite difference methods. However, the application is not very successful in plate with all edges simply supported. The solution procedure is also much more complex in comparison to the solution of linear problems. Nonlinear problems currently are considered to be in the territory of high performance workstations and supercomputers. With the development of personal computers, it has become possible to develop a computationally inexpensive approach for the nonlinear computations via PC. The DQ method should be the choice in this respect. However, in solving complex multidimensional nonlinear problems, for example, von Karman equations of plates, computational effort and storage requirements are still rather high even in the DQ method. Therefore, if the DQ method attempts to undertake more nonlinear computations via PC, its computing cost obviously needs to be reduced. Recently, some remarkable advances have been achieved in simplifying application and improving efficiency and accuracy of the DQ method. In this study, values of some new techniques in DQ computing are verified through the solution of the nonlinear geometrical bending of plates. In particular, the work validates the applicability, simplicity and high efficiency of special product approach in the DQ nonlinear computations. It should be pointed out that, due to a great reduction in the computing effort and storage requirements by using the new techniques, all results presented here were accomplished on a 386 personal computer with only 4 MB memory. In contrast, Bert et al. [9] used an IBM framework computer for the same task. In the following, innovations in this work are briefly introduced.

First, the Hadamard product of matrices and the DQ matrix approximate formulas are used to express the formulation of nonlinear partial differential operator in an explicit and easily programmable matrix form. The DQ analog formulas in matrix form [20] can be viewed as a simple and compact version of the traditional polynomial approximation of multidimensional problems [4]. By using these formulas, the formulation effort can be simplified. The often-used ordinary matrix product rises from the concept of linear transformation and is extended to handle the non-

linear problems. However, since nonlinear problems are actually different from linear ones, the ordinary matrix product seems not to undertake the task of nonlinear analysis and computations very well. The Hadamard product is a kind of special matrix product and not well known to the numerical computation community. It was found that the Hadamard product provides an explicit, compact and convenient approach to formulate the nonlinear differential operators in the DQ method [15] as well as the other numerical techniques [16]. Second, the SJT product was first introduced by Chen and Zhong [15] and Chen [16] as an efficient and simple algorithm to compute the exact Jacobian derivative matrix in the Newton–Raphson solution of the nonlinear formulations in the Hadamard product form. The SJT product is also a kind of special matrix product. It is emphasized that the SJT product approach may require minimal computing effort in all possible approaches for the same task. More importantly, the Hadamard and SJT product provides a decoupling technique in solving the coupled nonlinear DQ analog equations of the von Karman equations. By comparing with Bert et al. [9], the resulting size of the nonlinear simultaneous algebraic equations is reduced by two-thirds. Therefore, the computational effort and storage requirements are significantly alleviated. Third, a recent approach applying multiple boundary conditions in the DQ analysis of high order boundary value problems, proposed by Wang and Bert [21], is employed for the solution of the geometrically nonlinear simply supported plates. It was verified that in the linear cases, the accuracies of the DQ solutions using Wang and Bert's approach were evidently improved in comparison to using the conventional so-called δ approach [12,21,22]. Wang and Bert's approach is also effective for the other boundary conditions except for the clamped–clamped (C–C) boundary condition [21,22] and free edge of plate. Chen and Yu [23] proposed another different approach to improve accuracy and eliminate instability caused by δ effect in the conventional δ approach. Chen and Yu's approach is applicable for problems with any boundary constraint. In this study, the approach is applied to analyze the plate of clamped edges. An improvement on this approach is also presented to avoid extra four equations of corner points in the resulting formulation and reduce the size of the resultant algebraic systems. It is believed that these two new approaches applying multiple boundary conditions are first used in the DQ solution of nonlinear boundary value problems. To avoid confusion, the three approaches are denoted as the DQ δ approach [6,7,9,10], DQWB approach by Wang and Bert [21], and DQCY approach by Chen and Yu [23]. Fourth, the Chebyshev grid points are used in the present numerical study. As is expected, the DQ method using

such grid points has faster rate of convergence than using the equally spaced grid points in [9]. Finally, some conclusions are drawn based on the results reported herein.

2. Approaches applying boundary conditions and matrix approximate formulas in the DQ method

The differential quadrature method approximates the partial derivative of a function at a given discrete point as a weighted linear sum of function values at all of the discrete points along the respective variable direction in the entire domain of the variable. The m th order derivative of a single function $f(x)$ at a given discrete grid point i can be approximated by the DQ method with N discrete grid points [2–6], namely,

$$\frac{\partial^m f}{\partial x^m} \Big|_{x=x_i} = \sum_{j=1}^N w_{ij}^{(m)} f_j \quad i = 1, 2, \dots, N \tag{1}$$

where $f_j = f(x_j)$, $w_{ij}^{(m)}$ are the corresponding DQ weighting coefficients and can be determined by requiring that Eq. (1) be exact for all polynomials less than or equal to $N - 1$. The explicit formulas [11,17] were developed to conveniently obtain the accurate DQ weighting coefficients. For the weighting coefficients of the first order derivatives, we have

$$w_{ij}^{(1)} = \frac{1}{x_j - x_i} \prod_{\substack{k=1 \\ k \neq i \\ k \neq j}}^N \frac{x_i - x_k}{x_j - x_k}, \quad i = 1, 2, \dots, N \quad \text{and} \tag{2a}$$

$$j = 1, 2, \dots, N$$

$$w_{ii}^{(1)} = \sum_{k \neq i}^N \frac{1}{x_i - x_k}, \quad i = 1, 2, \dots, N \tag{2b}$$

The weighting coefficients for high order derivatives can be generated by recursion formulas [17]:

$$w_{ij}^{(m+1)} = m \left(w_{ij}^{(1)} w_{ii}^{(m)} - \frac{w_{ij}^{(m)}}{x_i - x_j} \right), \quad i \neq j \tag{3a}$$

$$w_{ii}^{(m+1)} = - \sum_{j \neq i}^N w_{ij}^{(m+1)} \tag{3b}$$

where the superscript (m) and $(m + 1)$ denote the order of the derivative.

In the present study, besides the uniform points, the zeros of Chebyshev polynomials of the first kind are considered, namely,

$$r_i = \cos \frac{(2i - 1)\pi}{2N}, \quad -1 < r_i < 1, \quad i = 1, 2, \dots, N \tag{4}$$

where r_i is the i th root of the N order Chebyshev polynomials. The zeros do not included the end points of the normalized domains ($0 \leq x \leq 1$ or $-1 \leq x \leq 1$), while in the present case, the end points ($x = 0, 1$ or $x = \pm 1$) are needed to force boundary conditions. Therefore, an algebraic transformation is required, namely,

$$x_i = \frac{r_i - 1}{r_N - r_1}, \quad 0 \leq x_i \leq 1 \quad i = 1, 2, \dots, N \tag{5a}$$

$$y_j = \frac{r_j - 1}{r_M - r_1}, \quad 0 \leq y_j \leq 1 \quad j = 1, 2, \dots, M \tag{5b}$$

The above formulas produce the coordinates of sampling points. With such grid points, a very simple formula can be obtained for computing the DQ weighting coefficients of the first order derivatives.

$$w_{ij}^{(1)} = \frac{(-1)^{(i-j)}(r_N - r_1)}{r_i - r_j} \sqrt{\frac{(1 - r_j^2)}{(1 - r_i^2)}}, \quad i \neq j \tag{6a}$$

$$w_{ii}^{(1)} = \frac{1}{2} \frac{r_i(r_N - r_1)}{1 - r_i^2}, \quad i = 1, 2, \dots, N \tag{6b}$$

The weighting coefficients for high order derivatives can be easily computed by using recurrence formulas (3a) and (3b). For more details about the DQ method see Refs. [2,4,6,11,17]. It should be especially noted that, in order to differentiate the variants from the DQ weighting coefficients more clearly, the notations for the DQ weighting coefficients of the first, second, third and fourth-order derivatives are represented as A_{ij} , B_{ij} , C_{ij} and D_{ij} , respectively, in all subsequent sections. \bar{A} , \bar{B} , \bar{C} and \bar{D} represent the corresponding modified coefficient matrices with built-in boundary conditions by the DQWB or DQCY approach.

2.1. Three approaches applying multiple boundary conditions

The governing equations in the structural mechanics usually involve the fourth-order derivatives, and the two conditions need be satisfied at each boundary. Therefore, the problems are more complex in comparison to the differential systems of no more than second order such as the Poisson equation and convection–diffusion equation, which do not involve more than one boundary condition at each boundary end of physical domain. Some careful considerations are necessary on how to properly implement the double boundary conditions at each edge. The problem is actually the DQ solution of high-order boundary value

problems with multiple boundary conditions. In what follows we discuss three existing approaches for such problems.

The earliest DQ δ approach, proposed by Bert et al. [6] and widely used in literature [7,9,10], enforces the geometry boundary condition at boundary points and derivative boundary conditions at the δ points, which have very small distance ($\delta \cong 10^{-5}$ in dimensionless value [22]) away from the boundary. Thus, the approach cannot satisfy derivative boundary conditions exactly at boundary points and the accuracies of the solutions are affected. The solution accuracy depends on the proper choice of δ . If the value of δ is small enough, the approach produces good results in some situations such as clamped condition, however, it failed to work well in the other situations such as simply supported and free edges [10,21,22]. On the other hand, too small δ will deteriorate the computations. δ is usually determined by trial and error for different cases [22], which is a rather tedious work. Arbitrariness in the choice of the δ value may introduce the unexpected oscillation behavior of the solutions. In addition, the number of grid points in the DQ δ approach cannot be large due to the ill-conditioned matrices caused by the δ grid spacing.

To overcome the drawbacks in the DQ δ approach, some new techniques were presented in the application of the multiple boundary conditions. Wang and Bert [21] developed a DQWB approach by incorporating the boundary conditions into the DQ weighting coefficient matrices in advance, and then the weighting coefficients with built-in boundary conditions were employed to analogize the governing equations of problems of interests. The essence of the approach is that the boundary conditions are applied while formulating the weighting coefficient matrices for inner grid points. The technique resulted in an obvious improvement in the DQ solution of beams and plates with free and SS boundary conditions [12,21,22]. However, the technique is not applicable for problems with the C–C boundary condition as well as cross derivative and corner boundary conditions such as completely free plates. Thus, a combination of the DQWB and DQ δ approaches was used to handle the problems with both the C–C boundary condition and the other boundary conditions in [22]. The modification of the weighting coefficient matrices also causes some loss of use flexibility and greatly increases computational effort, requiring some matrix product operations of $O(N^4)$ scalar multiplications. In this study, the DQWB approach is used to handle the problems with simple supports.

Chen and Yu [23] proposed a different DQCY approach to cure the deficiencies of the conventional DQ δ approach. The fact was noted that the rank of the DQ weighting coefficient matrix for the i th

order derivative is $N-i$, where N is the number of grid points. Naturally, the rank of the DQ coefficient matrix for the fourth-order derivative is $N-4$. Therefore, the DQ analog equations of the fourth-order governing equations at boundary points and the points immediate adjacent to the boundary need be replaced by the analog equations of the boundary equations. The DQCY approach imposes all boundary conditions exactly at boundary points. Therefore, the solution accuracy of the DQCY approach is improved evidently in comparison to the DQ δ approach. The approach is conceptually simple and effective for the C–C boundary condition as well as any other boundary condition. Since the DQCY eliminates the δ effect in the DQ δ , the stability of the solution is improved and larger number of grid points can be used. However, it has found that the DQCY approach is less efficient than the DQWB approach whenever the DQWB approach is applicable. Therefore, in this paper the DQCY approach is applied to handle the problems with the C–C boundary condition.

An improvement is also made here for the DQCY approach. In this application, the first and last two rows of the original weighting coefficient matrices are removed, and then the four boundary equations are applied to modify these coefficient matrices into $(N-4) \times (N-4)$ matrix of full rank. The modified weighting coefficient matrices can be used to formulate the differential equations of interest directly. Since the boundary conditions are incorporated into the DQ weighting coefficient matrices in advance, extra analog equations at four corner points of plate need no longer be considered in the resulting formulation. The size of the resultant algebraic equations is also reduced. For example, consider the C–C boundary:

$$w = 0, \quad \text{at } x = 0, 1 \quad (7a)$$

$$\frac{\partial w}{\partial x} = 0, \quad \text{at } x = 0, 1 \quad (7b)$$

The corresponding DQ approximate equations are given by

$$w_1 = 0, \quad w_N = 0 \quad (8a)$$

$$\sum_{j=1}^N A_{1j} w_j = 0, \quad \sum_{j=1}^N A_{Nj} w_j = 0 \quad (8b)$$

where w_j is the displacement at the j th grid point. It is noted that all boundary conditions are exactly satisfied at boundary points. By using Eqs. (8a) and (8b), the desired displacements at the second and $(N-1)$ th grid points can be expressed in terms of the unknown displacement values at interior points, namely,

$$w_2 = -\frac{1}{A_{12}} \sum_{j=3}^{N-1} A_{1j} w_j \quad (9a)$$

$$w_{N-1} = -\frac{1}{A_{N,N-1}} \sum_{j=2}^{N-2} A_{Nj} w_j \quad (9b)$$

Substituting Eqs. (8a) and (9a), (9b) into the DQ approximate formulas for the first, second, third and fourth derivatives, respectively, we have

$$\begin{aligned} \frac{d\bar{w}}{dx} &= \bar{A}\bar{w}, & \frac{d^2\bar{w}}{dx^2} &= \bar{B}\bar{w}, & \frac{d^3\bar{w}}{dx^3} &= \bar{C}\bar{w}, \\ \frac{d^4\bar{w}}{dx^4} &= \bar{D}\bar{w} \end{aligned} \quad (10)$$

where $\bar{\cdot} = \{w_3, w_4, \dots, w_{N-2}\}$. \bar{A} , \bar{B} , \bar{C} and \bar{D} and are $(N-4) \times (N-4)$ modified coefficient matrices different from those of $(N-2) \times (N-2)$ dimension in the DQWB approach. For the other boundary conditions, the modified coefficient matrices can be obtained in a similar way.

2.2. Matrix approximate formulas in the DQ method

One of the present authors [20] proposed the following DQ formulation in matrix form for the partial derivatives of the function $\psi(x, y)$ in two-dimensional domain:

$$\begin{aligned} \frac{\partial\psi}{\partial x} &= \bar{A}_x\psi, & \frac{\partial\psi}{\partial y} &= \psi\bar{A}_y^T, & \frac{\partial^2\psi}{\partial x\partial y} &= \bar{A}_x\psi\bar{A}_y^T, \\ \frac{\partial^2\psi}{\partial x^2} &= \bar{B}_x\psi, & \frac{\partial^2\psi}{\partial y^2} &= \psi\bar{B}_y^T, & \frac{\partial^2\psi}{\partial x\partial y} &= \bar{B}_x\psi\bar{B}_y^T, \\ \frac{\partial^2\psi}{\partial x^2} &= \bar{D}_x\psi, & \frac{\partial^2\psi}{\partial y^2} &= \psi\bar{D}_y^T \end{aligned} \quad (11)$$

where the unknown ψ is a rectangular unknown matrix rather than a vector as in [4,6,8,11–14,21,22]. \bar{A} , \bar{B} and \bar{D} with subscripts x and y stand for the DQ weighting coefficient matrices for the first, second and fourth-order partial derivatives along x and y directions, respectively. The superscript T means the transpose of the matrices. The above analog equations are compact matrix version of the traditional polynomial approximate formulas given by Civan and Sliepcevic [4]. In fact, both are equivalent.

3. Hadamard and SJT product

It is not an easy task to handle the problems involving nonlinearity. Chen and Zhong [15] first found that the Hadamard product of matrices was a simple

and effective technique to formulate nonlinear differential equations in the DQ method. The SJT product was also introduced there to efficiently compute the Jacobian derivative matrix in the Newton–Raphson method for the solution of nonlinear formulation in the Hadamard product form.

Definition 3.1. Let matrices $A = [a_{ij}]$ and $B = [b_{ij}] \in C^{N \times M}$, the Hadamard product of matrices is defined as $A \circ B = [a_{ij}b_{ij}] \in C^{N \times M}$, where $C^{N \times M}$ denotes the set of $N \times M$ real matrices.

Definition 3.2. If matrix $A = [a_{ij}] \in C^{N \times M}$, then $A^{\circ q} = [a_{ij}^q] \in C^{N \times M}$ is defined as the Hadamard power of matrix A , where q is a real number. In particular, if $a_{ij} \neq 0$, $A^{\circ(-1)} = [1/a_{ij}] \in C^{N \times M}$ is defined as the Hadamard inverse of matrix A . $A^{\circ 0} = 11$ is defined as the Hadamard unit matrix in which all elements are equal to unity.

Definition 3.3. If matrix $A = [a_{ij}] \in C^{N \times M}$, then the Hadamard matrix function $f^\circ(A)$ is defined as $f^\circ(A) = [f(a_{ij})] \in C^{N \times M}$.

Considering quadratic nonlinear differential operator $W_{,x}W_{,y}$, the DQ formulation can be expressed in Hadamard product form as

$$W_{,x}W_{,y} = (\bar{A}_x\bar{W}) \circ (\bar{A}_y\bar{W}) \quad (12)$$

where \bar{W} is the unknown vector. W with the comma subscript $,x$ and $,y$ denotes vector of partial derivative of function W along x - and y -directions. \bar{A}_x and \bar{A}_y are the DQ weighting coefficient matrices, modified by the boundary conditions, for the first order derivative along the x - and y -directions, respectively. The above equation exposes an essential idea in the latter analysis. For linear and nonlinear operators with varying parameters $c(x,y)$, the DQ analog can be obtained in the same way. For example,

$$c(x,y)U_{,x} = \{c(x_j,y_j)\} \circ (\bar{A}_x\bar{U}) \quad (13a)$$

and

$$c(x,y)U_{,x}U_{,y} = \{c(x_j,y_j)\} \circ (\bar{A}_x\bar{U}) \circ (\bar{A}_y\bar{U}). \quad (13b)$$

Theorem 3.1. If A , B and $C \in C^{N \times M}$, then

$$1. A \circ B = B \circ A \quad (14a)$$

$$2. k(A \circ B) = (kA) \circ B, \text{ where } k \text{ is a scalar} \quad (14b)$$

$$3. (A + B) \circ C = A \circ C + B \circ C \quad (14c)$$

The Newton–Raphson method may be one of the most important techniques to compute nonlinear algebraic equations. One of the major time-consuming calculations in the method is to evaluate the Jacobian derivative matrix. In order to simplify the computation of the Jacobian matrix, the postmultiplying SJT product of matrix and vector was introduced by Chen and Zhong [15].

Definition 3.4. If matrix $A = [a_{ij}] \in C^{N \times M}$, vector $\vec{v} = \{v_i\} \in C^{N \times 1}$, then $A \diamond \vec{v} = [a_{ij}v_i] \in C^{N \times M}$ is defined as the postmultiplying SJT product of matrix A and vector \vec{v} , where \diamond represents the SJT product.

In what follows we give the operation rules applying the SJT product to compute the Jacobian matrices, where $\partial/\partial \vec{U}$ denotes the Jacobian derivative matrix operator with respect to vector \vec{U} .

1. For $c(x,y)U_{,x} = \{c(x_j,y_j)\} \circ (\vec{A}_x \vec{U})$, we have

$$\frac{\partial}{\partial \vec{U}} \left\{ \{c(x_j,y_j)\} \circ (\vec{A}_x \vec{U}) \right\} = \vec{A}_x \diamond \{c(x_j,y_j)\} \quad (15a)$$

2. For $(U_{,x})^q = (\vec{A}_x \vec{U})^{oq}$, where q is a real number, we have

$$\frac{\partial}{\partial \vec{U}} \left\{ (\vec{A}_x \vec{U})^{oq} \right\} = q \vec{A}_x \diamond (\vec{A}_x \vec{U})^{o(q-1)} \quad (15b)$$

3. For $U_{,x}U_{,y} = (\vec{A}_x \vec{U}) \circ (\vec{A}_y \vec{U})$, we have

$$\begin{aligned} \frac{\partial}{\partial \vec{U}} \left\{ (\vec{A}_x \vec{U}) \circ (\vec{A}_y \vec{U}) \right\} \\ = \vec{A}_x \diamond (\vec{A}_y \vec{U}) + \vec{A}_y \diamond (\vec{A}_x \vec{U}) \end{aligned} \quad (15c)$$

4. For $\sin U_{,x} = \sin \circ (\vec{A}_x \vec{U})$, we have

$$\frac{\partial}{\partial \vec{U}} \left\{ \sin \circ (\vec{A}_x \vec{U}) \right\} = \vec{A}_x \diamond \cos \circ (\vec{A}_x \vec{U}) \quad (15d)$$

5. For $\exp(U_{,x}) = \exp \circ (\vec{A}_x \vec{U})$, we have

$$\frac{\partial}{\partial \vec{U}} \left\{ \exp \circ (\vec{A}_x \vec{U}) \right\} = \vec{A}_x \diamond \exp \circ (\vec{A}_x \vec{U}) \quad (15e)$$

The above computing formulas give the analytic solution of the Jacobian derivative matrix for the analog equations considered. The computational effort for one SJT product is only n^2 scalar multiplications, which may be the smallest computational cost in all possible approaches. The premultiplying SJT product was also introduced to compute the Jacobian matrix of the nonlinear formulation such as $\partial \vec{W}^m / \partial x = \vec{A}_x \vec{W}^{om}$ ($m \neq 1$)

[15,16]. Such cases are seldom encountered in structure analysis. Therefore, it is not presented here for the sake of brevity.

The finite difference method is a simple technique to obtain the approximate solution of the Jacobian matrix in practical engineering and requires $O(n^2)$ scalar multiplications. Both the SJT product approach and the finite difference method are essentially comparable in computing effort. However, the approximate Jacobian matrix yielded by the finite difference method affects the accuracy and convergence rate of the Newton–Raphson method. In contrast, the SJT product produces the analytic solution of the Jacobian matrix.

4. Applications

The von Karman equations governing a thin, homogeneous, orthotropic rectangular plate subject to a uniformly distributed transverse load are given by [9]

$$\begin{aligned} E_1 u_{,xx} + \mu G_{12} u_{,yy} + C v_{,xy} \\ = -w_{,x} (E_1 w_{,xx} + \mu G_{12} w_{,yy}) - C w_{,y} w_{,xy} \end{aligned} \quad (16a)$$

$$\begin{aligned} E_2 v_{,yy} + \mu G_{12} v_{,xx} + C u_{,xy} \\ = -w_{,y} (E_2 w_{,yy} + \mu G_{12} w_{,xx}) - C w_{,x} w_{,xy} \end{aligned} \quad (16b)$$

$$\begin{aligned} D_1 w_{,xxxx} + 2D_3 w_{,xxyy} + D_2 w_{,yyyy} \\ = q + \frac{h}{\mu} \left[\left(u_{,x} + \frac{1}{2} w_{,x}^2 \right) (E_1 w_{,xx} + \nu_{12} E_2 w_{,yy}) \right. \\ \left. + \left(v_{,y} + \frac{1}{2} w_{,y}^2 \right) (E_2 w_{,yy} + \nu_{21} E_1 w_{,xx}) \right. \\ \left. + 2\mu G_{12} w_{,xy} (u_{,y} + v_{,x} + w_{,x} w_{,y}) \right] \end{aligned} \quad (16c)$$

in terms of three displacement components, where ν_{12} and ν_{21} are Poisson's ratio, E_1 and E_2 are the Young's moduli, C is the shear modulus, D_1 , D_2 and D_4 are the principal bending and twisting rigidities, $\mu = 1 - \nu_{12}\nu_{21}$, u , v and w are the desired inplane and transverse displacements, and a , b and h are width, length and thickness of plate, respectively.

Applying the Hadamard product and the new DQ matrix approximate formulas (11), the analog equations for this case are

$$\begin{aligned}
 E_1 \bar{B}_x \hat{u} + \mu G_{12} \hat{u} \bar{B}_y^T + C \bar{A}_x \hat{v} \bar{A}_y^T \\
 = -(\bar{A}_x \hat{w}) \circ (E_1 \bar{B}_x \hat{w} + \mu G_{12} \hat{w} \bar{B}_y^T) \\
 - C(\hat{w} \bar{A}_y^T) \circ (\bar{A}_x \hat{w} \bar{A}_y^T)
 \end{aligned} \tag{17a}$$

$$\begin{aligned}
 E_2 \hat{v} \bar{B}_y^T + \mu G_{12} \bar{B}_x \hat{v} + C \bar{A}_x \hat{u} \bar{A}_y^T \\
 = -(\hat{w} \bar{A}_y^T) \circ (E_2 \hat{w} \bar{B}_y^T + \mu G_{12} \bar{B}_x \hat{w}) \\
 - C(\bar{A}_x \hat{w}) \circ (\bar{A}_x \hat{w} \bar{A}_y^T)
 \end{aligned} \tag{17b}$$

$$\begin{aligned}
 D_1 \bar{D}_x \hat{w} + 2D_3 \bar{B}_x \hat{w} \bar{B}_y^T + D_2 \hat{w} \bar{D}_y^T \\
 = q + \frac{h}{\mu} \left[\bar{A}_x \hat{u} + \frac{1}{2} (\bar{A}_x \hat{w})^{\circ 2} \right] \circ [E_1 \bar{B}_x \hat{w} \\
 + v_{12} E_2 \hat{w} \bar{B}_y^T] + \left[\hat{v} \bar{A}_y^T + \frac{1}{2} (\hat{w} \bar{A}_y^T)^{\circ 2} \right] \\
 \circ [E_2 \hat{w} \bar{B}_y^T + v_{21} E_1 \bar{B}_x \hat{w}] + 2\mu G_{12} (\bar{A}_x \hat{w} \bar{A}_y^T) \\
 \circ \left[\hat{u} \bar{A}_y^T + \bar{A}_x \hat{v} + (\bar{A}_x \hat{w}) \circ (\hat{w} \bar{A}_y^T) \right]
 \end{aligned} \tag{17c}$$

where \bar{A} , \bar{B} and \bar{D} with subscript x and y denote the modified weighting coefficient matrix along x and y directions, respectively. The orders of these matrices are $N - 2$ for the DQWB approach and $N - 4$ for the DQCY approach, where N is the number of grid points. Note that the boundary conditions have been applied in the DQWB and DQCY approach and thus are no longer considered. It is also noted that \bar{A}_x and \bar{A}_y here are different form those in Section 3. The latter are the stacked coefficient matrix, the corresponding \bar{W} and \bar{U} are the desired vectors, while the former are in one-dimensional sense and \hat{u} , \hat{v} and \hat{w} in the above formulations are rectangular matrices. It is also pointed out that \bar{A}_x , \bar{B}_x , \bar{A}_y and \bar{B}_y for different u , v and w are the same, respectively, in the cases of clamped or simply supported edges.

The variables are nondimensionalized as $X \equiv x/a$, $Y \equiv y/b$, $U \equiv \hat{u}/a$, $V \equiv \hat{v}/b$ and $W \equiv \hat{w}/h$. Furthermore, by using the Kronecker product of matrices, we have

$$\begin{aligned}
 H_1 \bar{U} + H_2 \bar{V} = -(H_7 \bar{W}) \circ (H_1 \bar{W}) \\
 - (H_8 \bar{W}) \circ (H_2 \bar{W})
 \end{aligned} \tag{18a}$$

$$\begin{aligned}
 H_2 \bar{U} + H_3 \bar{V} = -(H_8 \bar{W}) \circ (H_3 \bar{W}) \\
 - (H_7 \bar{W}) \circ (H_2 \bar{W})
 \end{aligned} \tag{18b}$$

$$\begin{aligned}
 H_4 \bar{W} = \frac{qa^4}{D_1 h} + \frac{a^4}{\mu D_1 h} \left[\frac{a^2}{h^2} \left[H_7 \bar{U} + \frac{1}{2} (H_7 \bar{W})^{\circ 2} \right] \right. \\
 \left. \circ [H_5 \bar{W}] + \frac{b^2}{h^2} \left[H_8 \bar{V} + \frac{1}{2} (H_8 \bar{W})^{\circ 2} \right] \right. \\
 \left. \circ [H_6 \bar{W}] + \frac{2\mu G_{12}}{C} (H_2 \bar{W}) \circ [H_8 \bar{U} + H_7 \bar{V} \right. \\
 \left. + (H_7 \bar{W}) \circ (H_8 \bar{W}) \right]
 \end{aligned} \tag{18c}$$

where $H_1, H_2, H_3, H_4, H_5, H_6, H_7$ and H_8 are defined in Appendix A. \bar{W} , \bar{U} and \bar{V} are vectors generated by stacking the rows of the corresponding rectangular matrix W , U and V into one long vector. By using the new matrix approximate formulas and Hadamard product, the present nonlinear formulations are very easily accomplished. The matrix form here is also simpler and more explicit than the conventional algebraic polynomial form given by Bert et al. [9].

Eqs. (18a) and (18b) can be also restated as

$$H_1 \bar{U} + H_2 \bar{V} = -L_1(\bar{W}) \tag{19a}$$

$$H_2 \bar{U} + H_3 \bar{V} = -L_2(\bar{W}) \tag{19b}$$

where

$$L_1(\bar{W}) = (H_7 \bar{W}) \circ (H_1 \bar{W}) + (H_8 \bar{W}) \circ (H_2 \bar{W}) \tag{20a}$$

$$L_2(\bar{W}) = (H_8 \bar{W}) \circ (H_3 \bar{W}) + (H_7 \bar{W}) \circ (H_2 \bar{W}) \tag{20b}$$

The unknown vector \bar{U} and \bar{V} can be expressed in terms of \bar{W} by

$$\bar{U} = H_9^{-1} H_3^{-1} L_2(\bar{W}) - H_9^{-1} H_2^{-1} L_1(\bar{W}) \tag{21a}$$

and

$$\bar{V} = H_{10}^{-1} H_2^{-1} L_2(\bar{W}) - H_{10}^{-1} H_1^{-1} L_1(\bar{W}) \tag{21b}$$

where $H_9 = H_2^{-1} H_1 - H_3^{-1} H_2$ and $H_{10} = H_1^{-1} H_2 - H_2^{-1} H_3$. By applying the SJT product for the evaluation of the Jacobian matrix, we have

$$\frac{\partial \bar{U}}{\partial \bar{W}} = H_9^{-1} H_3^{-1} \frac{\partial L_2(\bar{W})}{\partial \bar{W}} - H_9^{-1} H_2^{-1} \frac{\partial L_1(\bar{W})}{\partial \bar{W}} \tag{22a}$$

and

$$\frac{\partial \bar{V}}{\partial \bar{W}} = H_{10}^{-1} H_2^{-1} \frac{\partial L_2(\bar{W})}{\partial \bar{W}} - H_{10}^{-1} H_1^{-1} \frac{\partial L_1(\bar{W})}{\partial \bar{W}} \tag{22b}$$

where

$$\begin{aligned} \frac{\partial L_1(\bar{W})}{\partial \bar{W}} &= H_7 \diamond (H_1 \bar{W}) + H_1 \diamond (H_7 \bar{W}) \\ &+ H_8 \diamond (H_2 \bar{W}) + H_2 \diamond (H_8 \bar{W}) \end{aligned} \quad (23a)$$

and

$$\begin{aligned} \frac{\partial L_2(\bar{W})}{\partial \bar{W}} &= H_8 \diamond (H_3 \bar{W}) + H_3 \diamond (H_8 \bar{W}) \\ &+ H_7 \diamond (H_2 \bar{W}) + H_2 \diamond (H_7 \bar{W}) \end{aligned} \quad (23b)$$

$\partial \bar{U} / \partial \bar{W}$ and $\partial \bar{V} / \partial \bar{W}$ are relative Jacobian derivative matrices of dependent variable vector \bar{U} and \bar{V} with respect to \bar{W} . Applying formulas (21a) and (21b), and (22a) and (22b), the coupled formulations (18a)–(18c) are decoupled. The size of the resulting nonlinear simultaneous algebraic equations is reduced from $3(N-2) \times 3(N-2)$ to $(N-2) \times (N-2)$ for the DQWB approach or from $3(N-4) \times 3(N-4)$ to $(N-4) \times (N-4)$ for the DQCY approach. It is known that each iteration step in the Newton–Raphson method has to solve a set of linear simultaneous algebraic equations, which requires an order of n^3 scalar multiplications. n is the size of the solved equations. For example, the Gauss elimination method requires $n^3/3$ scalar multiplications, while the Gauss–Jordan method requires $n^3/2$ scalar multiplications. Therefore, the computational effort and storage requirements here are only about one twenty-seventh and one-ninth, respectively, as much as in Ref. [9]. \bar{W} is a basic variable vector in the present computations. Eq. (18c) is chosen as the basic iteration equation, namely,

$$\begin{aligned} \varphi\{\bar{W}\} &= H_4 \bar{W} - \frac{a^4}{\mu D_1 h} \left[\frac{a^2}{h^2} \left[H_7 \bar{U} + \frac{1}{2} (H_7 \bar{W})^{\circ 2} \right] \right. \\ &\quad \circ [H_5 \bar{W}] + \frac{b^2}{h^2} \left[H_8 \bar{V} + \frac{1}{2} (H_8 \bar{W})^{\circ 2} \right] \\ &\quad \circ [H_6 \bar{W}] + \frac{2\mu G_{12}}{C} (H_2 \bar{W}) \circ [H_8 \bar{U} \\ &\quad \left. + H_7 \bar{V} + (H_7 \bar{W}) \circ (H_8 \bar{W}) \right] - \frac{qa^4}{D_1 h} = 0 \end{aligned} \quad (24)$$

The Jacobian derivative matrix for the above iteration equation is given by

$$\begin{aligned} \frac{\partial \varphi\{\bar{W}\}}{\partial \bar{W}} &= H_4 - \frac{a^4}{\mu D_1 h} \left[\frac{a^2}{h^2} \left[H_7 \frac{\partial \bar{U}}{\partial \bar{W}} \right. \right. \\ &\quad \left. \left. + H_7 \diamond (H_7 \bar{W}) \right] \diamond [H_5 \bar{W}] + \frac{a^2}{h^2} H_5 \diamond \left[H_7 \bar{U} \right. \right. \\ &\quad \left. \left. + \frac{1}{2} (H_7 \bar{W})^{\circ 2} \right] + \frac{b^2}{h^2} \left[H_8 \frac{\partial \bar{V}}{\partial \bar{W}} + H_8 \diamond (H_8 \bar{W}) \right] \right. \\ &\quad \left. \diamond [H_6 \bar{W}] + \frac{b^2}{h^2} H_6 \diamond \left[H_8 \bar{U} + \frac{1}{2} (H_8 \bar{W})^{\circ 2} \right] \right. \\ &\quad \left. + \frac{2\mu G_{12}}{C} H_2 \diamond [H_8 \bar{U} + H_7 \bar{V} + (H_7 \bar{W}) \circ (H_8 \bar{W})] \right. \\ &\quad \left. + \frac{2\mu G_{12}}{C} \left[H_8 \frac{\partial \bar{U}}{\partial \bar{W}} + H_7 \frac{\partial \bar{V}}{\partial \bar{W}} + H_7 \diamond (H_8 \bar{W}) \right. \right. \\ &\quad \left. \left. + H_8 \diamond (H_7 \bar{W}) \right] \diamond (H_2 \bar{W}) \right] \end{aligned} \quad (25)$$

It is noted that the SJT product approach here yields the analytical solution of the Jacobian matrix quite simply and efficiently. The Newton–Raphson iteration equation for this case is

$$\bar{W}^{(k+1)} = \bar{W}^{(k)} - \left(\frac{\partial \varphi(\bar{W}^{(k)})}{\partial \bar{W}} \right)^{-1} \varphi(\bar{W}^{(k)}) \quad (26)$$

The DQ solutions for the corresponding linear isotropic and orthotropic plates are chosen as the initial guess of the iteration procedures. It is noted that the Newton–Raphson method has rather a big convergence domain in the present computations. Even if the resulting nonlinear results are even eight times larger than the initial linear solutions, the Newton–Raphson method still converges. Moreover, the solution converges very rapidly, and the iterative times vary from 1 to 10 for various loadings when the convergence criterion is the maximum residual of equation (24) no more than 10^{-5} . In contrast, the IMSL subroutine NEONE used in Ref. [9] computed the Jacobian matrix approximately by a finite difference technique. Therefore, as was shown in Fig. 8 in Ref. [9], the accuracy and converging rate of the DQ solutions are affected, especially for simply supported plates.

Refs. [8,12,17] pointed out that the accuracy and stability of the DQ method can be improved significantly if the Chebyshev grid spacing is used. In the following the DQWB and DQCY solutions are obtained by using Chebyshev grid spacing 7×7 for a simply supported plate and 9×9 for a clamped plate, respectively, unless where specified. To avoid the effects of round-off errors on the accuracy of the solution, double-precision arithmetic is used in all the results

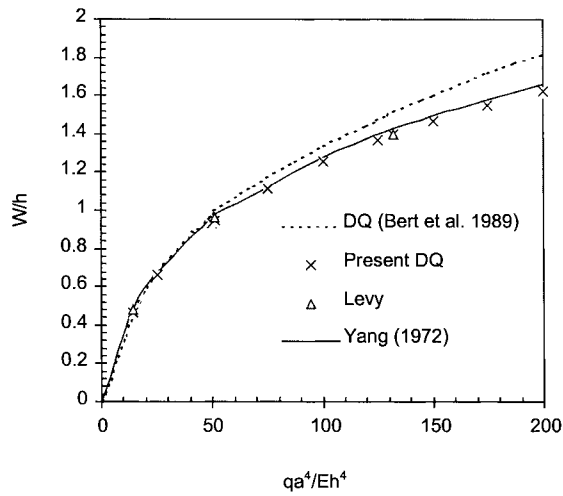


Fig. 1. Central deflections for a simply supported square isotropic plate.

presented in this paper. Ref. [9] has pointed out the high efficiency and ease of use in the DQ method in comparison to other numerical techniques such as the finite element, finite difference, perturbation, Galerkin and Rayleigh–Ritz, etc., while this paper places its emphasis on the simplification of the use and further improvement of the accuracy and efficiency in the DQ method. Therefore, the comparisons with the other numerical techniques are not discussed here.

The same simply supported and clamped isotropic square plates subject to a uniformly distributed loading as in example 1 of Ref. [9] are recalculated by the present DQ method. The solutions by both the DQCY

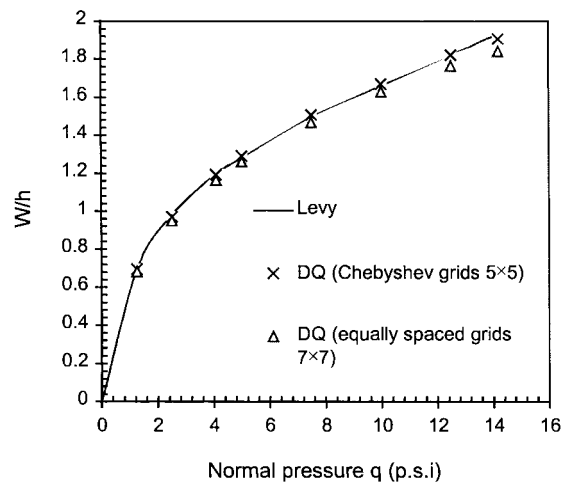


Fig. 2. Comparison of the DQ accuracies of central deflections for a square simply supported plate using the Chebyshev 5×5 and the equally spaced 7×7 grids.

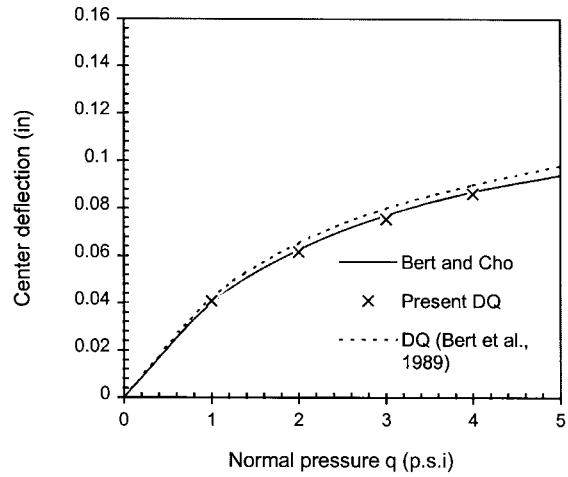


Fig. 3. Central deflections for a clamped square orthotropic plate.

and conventional $DQ\delta$ approaches are nearly in agreement and very accurate for the clamped plate and, thus, the DQCY solutions are not presented here for the sake of brevity. Nevertheless, it should be emphasized that the $DQ\delta$ approach cannot use a larger number of grid points as in the DQCY approach due to instability caused by the δ effect. The results for simply supported plate are shown in Fig. 1 and compared with the exact [24,25] and conventional DQ [9] solutions. The present DQWB results show remarkable agreement with those of Yang [24] and Levy [25]. It is also noted that the $DQ\delta$ approach using 7×7 grid points by Bert et al. [9] gave obviously better results in the clamped cases than in the simply supported cases as shown in Figs. 1 and 2 of Ref. [9]. This is because the $DQ\delta$ approach is not very suitable for the cases of supported edges. As is expected, the DQWB approach gives more accurate results than the $DQ\delta$ approach for simply supported plate. Therefore, the former is a competitive alternative to the latter for the nonlinear cases of simple supports.

In addition, we compute an isotropic simply supported square plate under a uniformly distributed transverse load. The parameters of this case are $a = 16''$, $h = 0.1''$, $E = 30 \times 10^6$ and $\nu = 0.316$. Fig. 2 depicts the results obtained by the DQWB approach using 5×5 Chebyshev grids and 7×7 equally spaced grids. All solutions agree very well with those given by Levy [26]. As is expected, the DQWB method using the 5×5 Chebyshev grids yields more accurate results than using 7×7 equally spaced grids.

The center deflections of the clamped square plate ($a = 100$, $h = 1.0$, $E = 2.1 \times 10^6$, $\nu = 0.316$, $q = 3.0$) and the simply supported square plate ($a = 100$, $h = 1.0$, $E = 2.1 \times 10^6$, $\nu = 0.25$, $q = 1.0$) subject to a

Table 1
The center deflections of the clamped and simply supported square plates

Methods	Analytical [27]	FEM [28]	FEM [27]	Present
Simply supported	0.940	1.028 (9.3%)	0.942 (0.3%)	0.944 (0.4%)
Clamped	1.151	1.316 (14.3%)	1.170 (1.6%)	1.123 (2.4%)

uniformly distributed pressure are computed, respectively, by the DQWB and DQCY methods. The DQ results as well as the analytical and FEM solutions are listed in Table 1. The present DQ solutions show excellent agreement with the analytical [27] and FEM solutions [27,28]. However, the computational effort in the present DQ method is much less than in the analytical method and FEM. The DQ method is demonstrated again to be highly computationally efficient for nonlinear structural analysis.

The numerical examples on the orthotropic rectangular plate provided by Bert et al. [9] are recalculated by the present DQ methods. The specific parameters are $E_1 = 18.7 \times 10^6$ psi; $E_2 = 1.3 \times 10^6$ psi; $G_{12} = 0.6 \times 10^6$ psi; $\nu_{12} = 0.3$; $h = 0.0624$ in; $a = 9.4$ in; $b = 7.75$ in. The center deflections in the clamped and simply supported cases are displayed in Figs. 3 and 4, respectively. For the cases of clamped edges, the results by the DQCY approach are very close to those by Bert and Cho. It is noted that the DQCY approach using 15×15 grid spacing is still stable and give accurate results, but computational effort also increases exponentially, while the $DQ\delta$ approach cannot use so many grid points. The computational stability problem may be essential in some cases when many grid points are required. Therefore, the DQCY approach may have better promise in practical engineering than the

$DQ\delta$ approach. In this case, the DQCY approach produces more accurate results by using more grid points (e.g. 9×9 grid spacing).

For the case of simple supports, it is noted that the results given by Bert et al. [9] using the $DQ\delta$ approach with 7×7 equally spaced grids are obviously larger than those by Bert and Cho. In contrast, the present DQWB approach appears to give results that are much closer to those by Bert and Cho as shown in Fig. 4. Bert and Cho's solution values here are taken from graphs 9 and 10 in respect to Ref. [9] with appropriate scaling factors. Another example on orthotropic plates discussed in Ref. [9] is also recalculated by the present DQ method and the same conclusions are obtained.

Based on the centrosymmetric structures of the DQ weighting coefficient matrices, the reduction technique in the DQ method was proposed in [16,17,20] and extended to the nonlinear problems in [16]. For geometrically nonlinear bending of the isotropic and orthotropic rectangular plates with symmetric boundary conditions, the computational effort and storage requirements can be further reduced by 75% and 50% using such reduction techniques, respectively.

5. Concluding remarks

The DQ approach in using some new techniques is applied to analyze geometrically nonlinear bending of isotropic and orthotropic plates with simply supported and clamped edges. The new matrix approximate formulas offer a compact and convenient DQ procedure to approximate partial derivatives. The DQWB approach is proved to be a successful technique for geometrically nonlinear plate with SS–SS boundary conditions. It is conceivable that the DQWB approach is also highly efficient for the other boundary conditions whenever applicable, as in the linear problems. The DQCY approach is improved and shown to be a stable and accurate technique for handling the cases with the C–C boundary conditions. Apparently, the results obtained by these two approaches are more accurate than those by the traditional $DQ\delta$ approach. Although only simply supported and clamped boundary conditions are involved in the present study, it is straightforward that the same procedures can be easily employed for handling problems with the other boundary conditions.

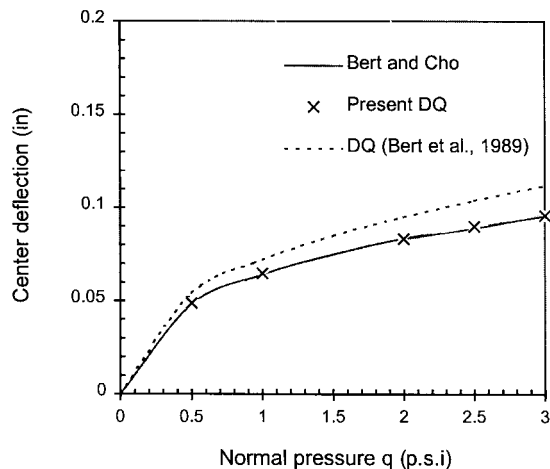


Fig. 4. Central deflections for a simply supported square orthotropic plate.

The publications in which the DQ method was applied to deal with nonlinear problems are still few due to much more complex programming, storage requirements, formulation and computing effort in comparison to linear problems. The Hadamard and SJT product approach may provide a simple and efficient technique to greatly reduce these impediments. The detailed solution procedures for the geometrically nonlinear plate cases are provided here to show the simplicity and efficiency of the Hadamard and SJT product approach. It is worth stressing that the Hadamard and SJT products as well as the relative decoupling technique are applicable for the finite difference, pseudo-spectral and collocation methods. Also, although only the application of Hadamard and SJT product within one single interval is presented in this paper, the extension of this procedure to solutions of problems with complex geometries with the coordinate mappings and multidomain techniques should be straightforward. The application of the present DQ solution procedure to the nonlinear analysis of more complex plates with varying thickness, Poisson’s ratio and Young’s modulus is currently the subject of further investigation.

Appendix A

The desired unknowns in rectangular matrix form as in equations (17a)–(17c) can be converted into the conventional vector form by using the Kronecker product [29].

Lemma 1. If $A \in C^{p \times m}$, $B \in C^{n \times q}$ and the unknown $X \in C^{m \times n}$, then

$$\text{vec}(AXB) = (A \otimes B^T)\text{vec}(X) \tag{A1}$$

where $\text{vec}(\)$ is the vector-function of a rectangular matrix formed by stacking the rows of matrix into one long vector, and \otimes denotes the Kronecker product of matrices. Note that $\text{vec}(\)$ here is to stack rows rather than columns as in Ref. [29] and, thus, the corresponding operation rules are also slightly different, but both are in fact equivalent. In this paper, we define $\text{vec}(AXB) = AX\bar{B}$ and $\text{vec}(X) = \bar{X}$ to simplify presentation.

Corollary.

1. $A\bar{X} = (A \otimes I_n)\bar{X}$
2. $\bar{X}B = (I_m \otimes B^T)\bar{X}$
3. $A\bar{X} + \bar{X}B = (A \otimes I_n + I_m \otimes B^T)\bar{X}$ (A2)

where I_n and I_m are the unit matrix. According to the above formulas, Eqs. (17a)–(17c) can be converted into Eqs. (18a)–(18c), and the resulting coefficient matrices in Eqs. (18a)–(18c) are given by

$$H_1 = E_1(\bar{B}_x \otimes I_y) + \mu G_{12} \left(\frac{a}{b}\right)^2 (I_x \otimes \bar{B}_y) \tag{A3}$$

$$H_2 = C(\bar{A}_x \otimes \bar{A}_y) \tag{A4}$$

$$H_3 = E_2(I_x \otimes \bar{B}_y) + \mu G_{12} \left(\frac{b}{a}\right)^2 (\bar{B}_x \otimes I_y) \tag{A5}$$

$$H_4 = \bar{D}_x \otimes I_y + \frac{D_2}{D_1} \left(\frac{a}{b}\right)^2 (\bar{B}_x \otimes \bar{B}_y) + \frac{D_3}{D_1} \left(\frac{a}{b}\right)^4 \times (I_x \otimes \bar{D}_y) \tag{A6}$$

$$H_5 = E_1 \left(\frac{h}{a}\right)^2 (\bar{B}_x \otimes I_y) + \nu_{12} E_2 \left(\frac{h}{b}\right)^2 (I_x \otimes \bar{B}_y) \tag{A7}$$

$$H_6 = E_2 \left(\frac{h}{b}\right)^2 (I_x \otimes \bar{B}_y) + \nu_{21} E_1 \left(\frac{h}{a}\right)^2 (\bar{B}_x \otimes I_y) \tag{A8}$$

$$H_7 = \frac{h^2}{a^2} (\bar{A}_x \otimes I_y) \tag{A9}$$

$$H_8 = \frac{h^2}{b^2} (I_x \otimes \bar{A}_y) \tag{A10}$$

References

- [1] Bellman RE, Casti J. Differential quadrature and long-term integration. *J Math Anal Applic* 1971;34:235–8.
- [2] Bellman RE, Kashef BG, Casti J. Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations. *J Comput Phys* 1972;10:40–52.
- [3] Mingle JO. The method of differential quadrature for transient nonlinear diffusion. *J Math Anal Applic* 1977;60:559–69.
- [4] Civan F, Sliepcevich CM. Differential quadrature for multidimensional problems. *J Math Anal Applic* 1984;101:423–43.
- [5] Naadimuthu G, Bellman RE, Wang KM, Lee ES. Differential quadrature and partial differential equations: some numerical results. *J Math Anal Applic* 1984;98:220–35.
- [6] Bert CW, Jang SK, Striz AG. Two new methods for ana-

- lyzing free vibration of structure components. *AIAA J* 1988;26:612–8.
- [7] Striz AG, Jang SK, Bert CW. Nonlinear bending analysis of thin circular plates by differential quadrature. *Thin-Walled Struct* 1988;6:51–62.
- [8] Quan JR, Chang CT. New insights in solving distributed system equations by the quadrature methods—2: numerical experiments. *Comput Chem Engng* 1989;13:1017–24.
- [9] Bert CW, Striz AG, Jang SK. Nonlinear bending analysis of orthotropic rectangular plates by the method of differential quadrature. *Comput Mech* 1989;5:217–26.
- [10] Feng Y, Bert CW. Application of the quadrature method to flexural vibration analysis of a geometrically nonlinear beam. *Nonlin Dyn* 1992;3:13–8.
- [11] Shu C, Richards BE. Parallel simulation of incompressible viscous flows by generalized differential quadrature. *Comput Systems Engng* 1992;3:271–81.
- [12] Bert CW, Wang X, Striz AG. Differential quadrature for static and free vibrational analysis of anisotropic plates. *Int J Solids Struct* 1993;30:1737–44.
- [13] Laura PAA, Rossi RE. The method of differential quadrature and its application to the approximate solution of ocean engineering problems. *Ocean Engng* 1994;21:57–66.
- [14] Malik M, Civan F. A comparative study of differential quadrature and cubature methods vis-à-vis some conventional techniques in context of convection–diffusion–reaction problems. *Chem Engng Sci* 1994;50:531–47.
- [15] Chen W, Zhong T. The study on nonlinear computations of the DQ and DC methods. *Numer Meth Partial Diff Eqns* 1997;13:57–75.
- [16] Chen W. Differential quadrature method and its applications in engineering. Ph.D. dissertation, Shanghai Jiao Tong University, China, 1996.
- [17] Quan JR, Chang CT. New insights in solving distributed system equations by the quadrature methods—1. *Comput Chem Engng* 1989;13:779–88.
- [18] Mansell G, Merryfield W, Shizgal B, Weinerl U. A comparison of differential quadrature methods for the solution of partial differential equations. *Comput Meth Appl Mech Engng* 1993;104:295–316.
- [19] Burka MK. Solution of stiff ordinary differential equations by decomposition and orthogonal collocation. *AIChE J* 1982;28:11–20.
- [20] Chen W, Wang X, Yong Y. Reducing the computational effort of the differential quadrature method. *Numer Meth Partial Diff Eqns* 1996;12:565–77.
- [21] Wang X, Bert CW. A new approach in applying differential quadrature to static and free vibrational analyses of beams and plates. *J Sound Vibr* 1993;162:566–72.
- [22] Wang X, Bert CW, Striz AG. Differential quadrature analysis of deflection, buckling and free vibration of beams and rectangular plates. *Comput Struct* 1993;48:473–9.
- [23] Chen W, Yu Y. Differential quadrature method for high order boundary value problems. In: Kwak BM, Tanaka M, editors. *Proceedings of 1st Pan-Pacific Conference on Computers in Engineering*. Amsterdam: Elsevier Science, 1993. p. 163–8.
- [24] Yang TY. Finite displacement plate flexure by the use of matrix incremental approach. *Int J Numer Meth Engng* 1972;4:415–32.
- [25] Levy S. Square plate with clamped edges under normal pressure producing large deflections. NACA report no. 740, 1942.
- [26] Levy S. Bending of rectangular plates with large deflections. NACA report no. 737, 1942. p. 139–157.
- [27] Zhu G, Wang H. Quasi-conforming penalty FEM for large deflection of composite laminated plate. *Acta Mater Compos Sinica* 1989;6:39–47 (in Chinese).
- [28] Bazeley GP, Cheung YK, Irens BM, Zienkiewicz OC. Triangular elements in bending-conforming and non-conforming solutions. In: Przemieniecki JS, et al., editors. *Proceedings of Conference on Matrix Methods in Structural Mechanics*. OH, Wright-Patterson Air Force Base: Air Force Institute of Technology, 1965. p. 547–76.
- [29] Lancaster P, Timenetsky M. *The theory of matrices with applications*, 2nd ed. Orlando, FL: Academic Press, 1985.