Journal of King Saud University – Science 31 (2019) 209–214

Contents lists available at ScienceDirect

Journal of King Saud University – Science

journal homepage: www.sciencedirect.com

Original article

Numerical solution of thin plates problem via differential quadrature method using G-spline

Osama H. Mohammed *, Mustafa A. Saeed

Department of Mathematics and Computer Applications, College of Science, Al-Nahrain University, Baghdad, Iraq

ARTICLE INFO

Article history:
Received 3 December 2017
Accepted 2 April 2018
Available online 5 April 2018

Keywords:
Differential quadrature method
Hermite-Birkhoff problem (HB-problem)
G-spline interpolation
Chebyshev Gauss Lobatto points

ABSTRACT

In this article the numerical solution of thin plates problem is introduced by using the differential quadrature method together with Chebyshev Gauss Lobatto sampling points for modeling the vibration of a square thin plate.

The explicit formula of the weighting coefficients for approximation of derivatives is utilized with the aid of the G-spline interpolation function.

A numerical example is presented and the results that have been obtained are compared with the existing methods in order to illustrate the validity and accuracy of the proposed approach.

1. Introduction

Bellman and Casti was proposed a numerical method which is so called differential quadrature (DQ) for evaluating the derivatives of sufficiently smooth function, (Bellman and Casti, 1971). Their basic idea came from the well-known approach Gauss Quadrature for calculating the integral numerically.

Evaluating the derivatives of different orders of a sufficiently smooth function can be considered as an extension which is give rise to DQ (Bellman and Casti, 1971; Jalaal et al., 2011), where the derivatives of a smooth function are approximated with weighting sum of function values at a group of so called sampling points or nodes (Zong and Zhang, 2009).

Bellman and his co-authors presented two methods for calculating the weighting coefficients which is the key procedure in the DQ applications (Shu, 2000).

Differential Quadrature (DQ) aroused many authors and because of that, its applications rapidly developed, (Quan and Chang, 1989; Shu and Richards, 1992; Shu and Xue, 1997; Shu and Wu, 2007; Korkmaz and Dag, 2008; Jiwari et al., 2012; Pekmen and Tezer-Sezgin, 2012; Ragb et al., 2014; Jiwari, 2015; Eftekhari, 2015; Shamani et al., 2015; Ghasemi et al., 2016; Mittal and Dahiyah, 2016, 2017; Ghasemi, 2017; Thoudam, 2017; Shamani and Aghdam, 2017a,b; Shamani and Aghdam, 2018).

A comprehensive review of the differential quadrature method has been given by Bert and Malik (1996). This paper employs function approximation theory using G-spline interpolation to formulate DQ.

Nearly 71 years ago, I.J. Schoenberg (1968) introduced the subject of "spline function" since that time splines may be considered as an important tool in different branches of mathematics such as approximation theory, numerical analysis, numerical treatment of ordinary, integral, partial differential equations, statistics, etc. There are several types of splines appeared in literature given by Deboor (1978), Powell (1981) and Stephen (2002).

Among these types of spline the so called G-spline interpolation which is necessary to the work of this paper, was initially presented by Schoenberg (1968). Schoenberg used the term "G-spline" instead of generalized splines because the natural spline term "generalized spline" describes an extension of another type of spline.

The G-spline is used to interpolate the HB-data (problem), the data in this problem are the values of the function and its derivatives but without Hermite's condition that the only consecutives be used at each node. Further, Schoenberg (1968) define G-spline as smooth piecewise polynomials, where the smoothness is governed by the incidence matrix, and then he proved that G-splines, satisfies the "minimum norm property", which is used for the optimality of the G-spline function, that is defined mathematically by the following inequality:

* Corresponding author.
E-mail address: uhm@sc.nahrainuniv.edu.iq (O.H. Mohammed).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksus.2018.04.001
1018-3647/

$$\int_I [f^{(m)}(x)]^2 dx > \int_I [S_{(m)}(x)]^2 dx, \quad (1)$$

where the function S is called a G-spline and it's a polynomial spline of degree 2 m-1 over the interval I.

2. The G-spline interpolation function:

I.J. Schoenberg (1968) proposed a tool in order to specify the HB-problem or the interpolatory condition:

$$f^{(j)}(x_i) = y_i^{(j)} \text{for } (i,j) \in e \quad (2)$$

where e is a certain set of ordered pairs, and called a G-spline interpolation. It is convenient in this section to discuss the HB-problem, before we give the tractable formal definition of the natural G-spline interpolation as follows: Consider the node points

$x_1 < x_2 < \ldots < x_K$and let α be the maximum order of the derivatives to be specified at the nodes. Define an incidence matrix E, by:

$$E = [a_{ij}], i = 1, 2\ldots, k; j = 0, 1\ldots, \alpha, \quad (3)$$

where:

$$a_{ij} = \begin{cases} 1, & (i,j) \in e, \\ 0, & (i,j) \notin e, \end{cases}$$

and $e = \{(i,j) : i = 1, 2\ldots, K; j = 0, 1, \ldots \alpha\}$.

Let $y_i^{(j)}$ be prescribed real numbers for each $(i,j) \in e$, then the HB-problem is to find $f(x) \in C^{\alpha}$,such that:

$$f^{(j)}(x_i) = y_i^{(j)} \text{ for } (i,j) \in e. \quad (4)$$

The matrix E will likewise describes the set of Eq. (4) if we define the set e by:

$$e = \{(i,j) | a_{ij} = 1\}. \quad (5)$$

Then the integer $n = \sum_{i,j} a_{ij}$, is the number of interpolatory conditions required to constitute the system (4).

The G-spline interpolant of order $m$to f can be given in terms of the fundamental G-spline functions $L_{ij}(x)$, which is described in detail in Schoenberg (1968) by:

$$S_m(x) = \sum_{(i,j)\in e} L_{ij}(x) y_i^{(j)}, \quad (6)$$

where

$$L_{ij}^{(s)}(x_r) = \begin{cases} 0, \text{if } (r,s) \neq (i,j). \\ 1, \text{if } (r,s) = (i,j). \end{cases}$$

2.1. Approximation of linear functional with the sense of G-spline formula

Let I = [a, b] be a finite interval containing the node points $x_1, x_2, \ldots x_k$ and let us consider a linear functional:

$\pounds f : C^{\alpha}[a,b] \longrightarrow R$ of the form:

$$\pounds f = \sum_{j=0}^{\alpha} \int_a a_j(x) f^{(j)}(x) dx + \sum_{j=0}^{\alpha} \sum_{i=1}^{n_j} b_{ji} f^{(j)}(x_{ji}), \quad (7)$$

where $a_j(x)$are piecewise continuous functions in I, $x_{ji} \in I$ and b_{ji} are real constants. we can approximate the functional in Eq. (7) using the formula:

$$\pounds f = \sum_{(i,j)\in e} \beta_{ij} f^{(j)}(x_i) + Rf, \quad (8)$$

where Rf represents the remainder see Schoenberg (1968). Therefore, in order to find the approximation of $\pounds f$ given by (8), which is best in some sense, we propose to determine the real's β_{ij}.

Schoenberg (1968) states two procedures to determine β_{ij} one of them is Sard procedure, which can be summarized by the following theorem:

2.1.1. Theorem (Schoenberg, 1968)

If $\alpha < m < n$ and the HB-problem (4) is m-poised, then Sard's best approximation (8) to $\pounds f$ of order m is obtained by operating with £ on both sides of the G-spline interpolation formula (6) of order m.

In other words, the coefficients β_{ij} are given by:

$\beta_{ij} = \pounds L_{ij}(x)$, where L_{ij} are the fundamental functions of (6).

Details can be found in Schoenberg (1968) for the generation of these fundamentals G-splines.

3. The G-spline interpolation-based differential quadrature method

Suppose that the function $f(x)$ is sufficiently smooth on the interval $[x_1, x_N]$, and consider an m-poised HB- problem:

$f^{(j)}(x_i) = y_i^{(j)}, (i,j) \in e$, on the N distinct nodes:

$x_1, x_2 \ldots, x_N$.

Based on differential Quadrature, we have

$$\left.\frac{df}{dx}\right|_{x=x_k} = \sum_{(i,j)\in e} a_{ki}^{(j)} f_i^{(j)}, \quad k = 1, 2, \ldots N, \quad (9)$$

$$\left.\frac{d^2f}{dx^2}\right|_{x=x_k} = \sum_{(i,j)\in e} b_{ki}^{(j)} f_i^{(j)}, \quad k = 1, 2, \ldots N, \quad (10)$$

where $a_{ki}^{(j)}$ and $b_{ki}^{(j)}$ are the weighting coefficients of the first and second order derivatives.

3.1. Computation of the weighting coefficients for the first and second order derivatives using G-spline interpolation formula

To find the weights $a_{ki}^{(j)}$ and $b_{ki}^{(j)}$,we need to consider an m-poised HB-problem to approximate the function. Our purpose is to construct a polynomial of x, which is of the form

$$f(x) \simeq \sum_{(i,j)\in e} L_{ij}(x) f_i^{(j)} \quad (11)$$

Then the derivatives of order one and two at any grid points may be approximated as:

$$\left.\frac{df}{dx}\right|_{x=x_k} = \sum_{(i,j)\in e} \left.\frac{dL_{ij}(x)}{dx}\right|_{x=x_k} f_i^{(j)}, \quad (12)$$

and

$$\left.\frac{d^2f}{dx^2}\right|_{x=x_k} = \sum_{(i,j)\in e} \left.\frac{d^2L_{ij}(x)}{dx^2}\right|_{x=x_k} f_i^{(j)}, \quad (13)$$

with $a_{ki}^{(j)}$ are the coefficients for the first order derivative, obtained by the following formula

$$a_{ki}^{(j)} = \left.\frac{dL_{ij}(x)}{dx}\right|_{x=x_k}, \quad (14)$$

and $b_{ki}^{(j)}$ are the coefficients of the second order derivative, given by

$$b_{ki}^{(j)} = \frac{d^2L_{ij}(x)}{dx^2}|_{x=x_k}. \quad (15)$$

In the same manner, we may obtain formulae for higher order derivatives by using the higher order weighting coefficients, which are expressed as $e_{ki}^{(j,m)}$ to avoid confusion. They are characterized by the following recurrence formulation.

$$e_{ki}^{(j,m)} = \frac{d^m L_{ij}(x)}{dx^m}\bigg|_{x=x_k} \quad (i,j) \in e, \quad k = 1,2,\ldots N-1. \tag{16}$$

Here we assume that $a_{ki}^{(j)} = e_{ki}^{(j,1)}$ and $b_{ki}^{(j)} = e_{ki}^{(j,2)}$

4. Analysis of differential quadrature of thin plates

In this section, the implementation of the G-spline interpolation-based DQM will be illustrated for thin plates problem.

4.1. The controlling equations and boundary conditions

The non-dimensional controlling equations for the deflection, free vibration and buckling for a plate may be written as Shu (2000):

Plate deflection

$$\frac{\partial^4 W}{\partial x^4} + 2\lambda^2 \frac{\partial^4 W}{\partial X^2 \partial Y^2} + \lambda^4 \frac{\partial^4 W}{\partial Y^4} = \frac{a^4 q(X,Y)}{D}, \tag{17}$$

transverse vibration of thin, isotropic plates

$$\frac{\partial^4 W}{\partial x^4} + 2\lambda^2 \frac{\partial^4 W}{\partial X^2 \partial Y^2} + \lambda^4 \frac{\partial^4 W}{\partial Y^4} = \Omega^2 W, \tag{18}$$

buckling of a plate under uniaxial compression

$$\frac{\partial^4 W}{\partial x^4} + 2\lambda^2 \frac{\partial^4 W}{\partial X^2 \partial Y^2} + \lambda^4 \frac{\partial^4 W}{\partial Y^4} = \frac{N_x a^2}{D} \frac{\partial^2 W}{\partial X^2}, \tag{19}$$

where W is the dimensionless mode shape function, $q(X,Y)$ is the external distributed load, Ω is the dimensionless frequency, $X = x/a, Y = y/b$are dimensionless coordinates, a and b are the lengths of the plate edges,$\lambda = a/b$ is the aspect ratio, and N_x is the uniaxial load. Furthermore, $\Omega = \omega a^2\sqrt{\rho/D}$, where ω is the dimensionless circular frequency, $D = Eh^3/[12(1-\upsilon^2)]$ is the flexural rigidity, E, υ, ρ and h are Young's modulus, Poisson's ratio, the density of the plate material, and the plate thickness, respectively. It should be mentioned that the above equations do not cover all the cases. For example, for free vibration of the anisotropic plates, Eq. (17) has to be changed to include more terms. Eq. (18) can be modified to consider buckling under different compressions.

There are three basic boundary conditions, for free vibration analysis, these boundary conditions are:

Simply supported edge (SS)

$$W|_{X=0} = 0, \quad \frac{\partial^2 W}{\partial X^2}\bigg|_{X=1} = 0, \tag{20}$$

$$W|_{Y=0} = 0, \quad \frac{\partial^2 W}{\partial Y^2}\bigg|_{Y=1} = 0, \tag{21}$$

Clamped edge (C)

$$W|_{X=0} = 0, \quad \frac{\partial W}{\partial X}\bigg|_{X=1} = 0, \tag{22}$$

$$W|_{Y=0} = 0, \quad \frac{\partial W}{\partial Y}\bigg|_{Y=1} = 0, \tag{23}$$

Free edge (F)

$$\frac{\partial^2 W}{\partial X^2} + \upsilon\lambda^2 \frac{\partial^2 W}{\partial Y^2}\bigg|_{X=0} = 0, \quad \frac{\partial^3 W}{\partial X^3} + (2-\upsilon)\lambda^2 \frac{\partial^3 W}{\partial X \partial Y^2}\bigg|_{X=1} = 0, \tag{24}$$

$$\lambda^2 \frac{\partial^2 W}{\partial Y^2} + \upsilon \frac{\partial^2 W}{\partial X^2}\bigg|_{Y=0} = 0, \quad \lambda^2 \frac{\partial^3 W}{\partial Y^3} + (2-\upsilon) \frac{\partial^3 W}{\partial X^2 \partial Y}\bigg|_{Y=1} = 0, \tag{25}$$

and

$$\frac{\partial^2 W}{\partial X \partial Y} \tag{26}$$

at the corner of two adjacent free edges.

4.2. Numerical discretization of the problem

The domain of computation for a rectangular plate is $0 \leqslant X \leqslant 1, 0 \leqslant Y \leqslant 1$ and for a numerical calculation, the mesh generation will be given as:

$$X_i = \frac{1}{2}\left[1 - \cos\left(\frac{i-1}{N-1}\pi\right)\right], \quad i = 1,2,\ldots,N, \tag{27}$$

$$Y_r = \frac{1}{2}\left[1 - \cos\left(\frac{r-1}{M-1}\pi\right)\right], \quad r = 1,2,\ldots,M, \tag{28}$$

where N and M refers to the number of the grid points in the X and Y directions respectively. Eqs. (17)–(19) can be discretizing by using G-spline-based DQ weighting coefficients. Let $e_{k,i}^{(j,n)}$ be the DQ weighting coefficients of the $n-th$ order derivative in the X direction, and $e_{h,r}^{-(s,m)}$ be the DQ weighting coefficients of the m-th order derivative in the Y direction. Using the DQM, Eq. (17) may be discretized as

$$\begin{aligned}&\sum\nolimits_{(i,j)\in e_1} e_{k,i}^{(j,4)} W_{i,s}^{(j,s)} + 2\lambda^2 \sum_{(i,j)\in e_1}\sum_{(r,s)\in e_2} e_{k,i}^{(j,2)} e_{h,r}^{-(s,2)} W_{i,r}^{(j,s)} \\ &\quad + \lambda^4 \sum_{(r,s)\in e_2} e_{h,r}^{-(s,4)} W_{i,r}^{(j,s)} \\ &\quad = \frac{a^4 q_{i,r}}{D}, (i,j) \in e_1, (r,s) \in e_2, \\ &k = 1,2,\ldots,N, h = 1,2,\ldots,M;\end{aligned} \tag{29}$$

and Eq. (18)is discretized as

$$\begin{aligned}&\sum_{(i,j)\in e_1} e_{k,i}^{(j,4)} W_{i,r}^{(j,s)} + 2\lambda^2 \sum_{(i,j)\in e_1}\sum_{(r,s)\in e_2} e_{k,i}^{(j,2)} e_{h,r}^{-(s,2)} W_{i,r}^{(j,s)} \\ &\quad + \lambda^4 \sum_{(r,s)\in e_2} e_{h,r}^{-(s,4)} W_{i,r}^{(j,s)} \\ &\quad = \Omega^2 W_{i,r}^{(j,s)}, (i,j) \in e_1, (r,s) \in e_2, \\ &k = 1,2,\ldots,N, h = 1,2,\ldots,M;\end{aligned} \tag{30}$$

Hence, Eq. (19) is discretized as

$$\begin{aligned}&\sum_{(i,j)\in e_i} e_{k,i}^{(j,4)} W_{i,r}^{(j,s)} + 2\lambda^2 \sum_{(i,j)\in e_1}\sum_{(r,s)\in e_2} e_{k,i}^{(j,2)} e_{h,r}^{-(s,2)} W_{i,r}^{(j,s)} \\ &\quad + \lambda^4 \sum_{(r,s)\in e_2} e_{h,r}^{-(s,4)} W_{i,r}^{(j,s)} \\ &\quad = \frac{a^2 N_x}{D} \sum_{(i,j)\in e_1} e_{k,i}^{(j,2)} W_{i,r}^{(j,s)}, (i,j) \in e_1, (r,s) \in e_2, \\ &k = 1,2,\ldots,N, h = 1,2,\ldots,M,\end{aligned} \tag{31}$$

where $W_{i,r}^{(j,s)} = \frac{\partial^{j+s} W}{\partial^j X \partial^s Y}|_{x_i,y_r}$ is $(j+s)^{th}$ derivative at the net point (X_i, Y_r)

Eqs. (29)–(31) can be put in matrix form and the solution of these matrix forms can be calculated using standard solvers such as QR-algorithm.

4.3. Direct substitution of boundary conditions into discrete controlling equation

The derivatives that appeared in the boundary conditions must also discretize by the DQM and as follows:

$$W^{(j,s)}_{1,r} = 0, W^{(j,s)}_{N,r}, W^{(j,s)}_{i,1} = 0, W^{(i,s)}_{i,M} = 0 \quad (32)$$

$$\sum_{(i,j)\in e_1} e^{(j,n_0)}_{1,i} W^{(j,s)}_{i,r} = 0 \ at \ X = 0 \quad (33)$$

$$\sum_{(i,j)\in e_1} e^{(j,n_1)}_{N,i} W^{(j,s)}_{i,r} = 0 \ at \ X = 1 \quad (34)$$

$$\sum_{(r,s)\in e_2} e^{-(s,m_0)}_{1,r} W^{(j,s)}_{i,r} = 0 \ at \ Y = 0 \quad (35)$$

$$\sum_{(r,s)\in e_2} e^{-(s,m_1)}_{M,r} W^{(j,s)}_{i,r} = 0 \ at \ Y = 1 \quad (36)$$

where n_0, n_1, m_0 and m_1 are possessed as either 1 or 2, where 1 is used for the clamped edge condition and 2 is used for the simply supported edge condition.n_0, n_1, m_0 and m_1 correspond to the edges of $X = 0, X = 1, Y = 0, Y = 1$, respectively.

Obviously, Eq. (32) can be easily substituted into Eq. (30). However, Eqs. (33)–(36) cannot be directly substituted into Eq. (30).

Using the direct approach given in Shu (2000), Eqs. (33) and (34) can be coupled to give two solutions $W^{(j_1,s)}_{2,r}$ and $W^{(j_2,s)}_{N-1,r}$, where j_1 and j_2 represent the minimum partial order derivatives of $W$with respect to X at X_2 and X_{N-1} respectively, which are located at the grid points shown by the symbol ∘ in Fig. 1

$$W^{(j_1,s)}_{2,r} = \frac{1}{AXN} \sum_{(i,j)\in e^*_1} AXK1.W^{(j,s)}_{i,r}, \quad (37)$$

$$w^{(j_2,s)}_{N-1,r} = \frac{1}{AXN} \sum_{(i,j)\in e^*_1} AXKN.W^{(j,s)}_{i,r}, \quad (38)$$

for $r = 3, 4, \ldots, M-2, e^*_1 = e_1/\{(2, j_1), (N-1, j_2)\}$where

$$AXN = e^{(j,n_1)}_{N,2} e^{(j,n_0)}_{1,N-1} - e^{(j,n_0)}_{1,2} e^{(j,n_1)}_{N,N-1}$$

$$AXK1 = e^{(j,n_0)}_{1,i} e^{(j,n_1)}_{N,N-1} e^{(j,n_0)}_{1,N-1} e^{(j,n_1)}_{N,i}$$

$$AXKN = e^{(j,n_0)}_{1,2} e^{(j,n_1)}_{N,i} - e^{(j,n_0)}_{1,i} e^{(j,n_1)}_{N,2}$$

For $(i,j) \in e^*_1$

In a similar way, Eqs. (35) and (36) can be coupled to give the solutions $W^{(j,s_1)}_{i,2}$ and $W^{(j,s_2)}_{i,M-1}$ where s_1 and s_2 represent the minimum partial order derivatives of W with respect to Y at Y_2 and Y_{M-1} respectively, which are located at the net points shown by the symbol □ in Fig. 1,

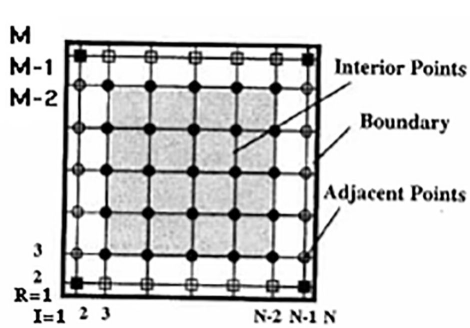

Fig. 1. Description of the points for a rectangular plate.

$$W^{(j,s_1)}_{i,2} = \frac{1}{AYM} \sum_{(r,s)\in e^*_2} AYK1.W^{(r,s)}_{i,r}, \quad (39)$$

$$W^{(j,s_2)}_{i,M-1} = \frac{1}{AYM} \sum_{(r,s)\in e^*_2} AYKM.W^{(r,s)}_{i,r}, \quad (40)$$

for $i = 3, 4, \ldots, N-2, e^*_2 = e_2 \setminus \{(2, s_1), (M-1, s_2)\}$where

$$AYN = e^{-(s,m_1)}_{M,2} e^{-(s,m_0)}_{1,M-1} - e^{-(s,m_0)}_{1,2} e^{-(s,m_1)}_{M,M-1}$$

$$AYK1 = e^{-(s,m_0)}_{1,r} e^{-(s,m_1)}_{M,M-1} - e^{-(s,m_0)}_{1,M-1} e^{-(s,m_1)}_{M,r}$$

$$AYKM = e^{-(s,m_0)}_{1,2} e^{-(s,m_1)}_{M,r} - e^{-(s,m_0)}_{1,r} e^{-(s,m_1)}_{M,2}$$

For $(r,s) \in e^*_2$

For the points near the four corners shown by the symbol ■ in Fig. 1, the four Eqs. (33)–(36) have to be coupled in order to give the following four solutions:

$$W^{(j_1,s_1)}_{2,2} = \frac{1}{AXN} \frac{1}{AYM} \sum_{(i,j)\in e_1} \sum_{(r,s)\in e_2} AXK1.AYK1.W^{(j,s)}_{i,r}, \quad (41)$$

$$W^{(j_2,s_1)}_{N-1,2} = \frac{1}{AXN} \frac{1}{AYM} \sum_{(i,j)\in e_1} \sum_{(r,s)\in e_2} AXKN.AYK1.W^{(j,s)}_{i,r}, \quad (42)$$

$$W^{(j_1,s_2)}_{2,M-1} = \frac{1}{AXN} \frac{1}{AYM} \sum_{(i,j)\in e_1} \sum_{(r,s)\in e_2} AXK1.AYKM.W^{(j,s)}_{i,r}, \quad (43)$$

$$W^{(j_2,s_2)}_{N-1,M-1} = \frac{1}{AXN} \frac{1}{AYM} \sum_{(i,j)\in e_1} \sum_{(r,s)\in e_2} AXKN.AYKM.W^{(j,s)}_{i,r}, \quad (44)$$

The reason for writing $W^{(j_1,s_1)}_{2,2}$, $W^{(j_2,s_1)}_{N-1,2}$, $W^{(j_1,s_2)}_{2,M-1}$ and $W^{(j_2,s_2)}_{N-1,M-1}$ in this form is to make them also as in the Eqs. (37)–(40) in terms of the points that we will get later form the Eq. (45). with Eqs. (32), (37)–(43) and (44), the boundary conditions are all directly substituted into Eq. (30). Hence, the final eigenvalue system of Eq. (30) may be given as:

$$\sum_{(i,j)\in e_1} C_1 W^{(j,s)}_{i,r} + 2\lambda^2 \sum_{(i,j)\in e_1} \sum (r,s) \in e_2 C_2 W^{(j,s)}_{i,r} + \lambda^4 \sum_{(r,s)\in e_2} C_3 W^{(j,s)}_{i,r} = \Omega^2 W^{(j,s)}_{i,r}, \quad (45)$$

$(i,j) \in e_1, (r,s) \in e_2$where

$$C_1 = e^{(j,4)}_{k,i} - \frac{e^{(j,4}_{k,2} AXK1 + e^{(j,4)}_{k,N-1} AXKN}{AXN},$$

$$C_2 = e^{(j,2)}_{k,i} e^{-(s,2)}_{h,r} - \frac{\left(AXK1.e^{(j,2)}_{k,2} + AXKN.e^{(j,2)}_{k,N-1}\right)}{AXN} e^{-(s,2)}_{h,r} - \frac{\left(AYK1.e^{(s,2)}_{h,2} + AYKM.e^{-(s,2)}_{h,M-1}\right)}{AYM} e^{(j,2)}_{k,i} + \frac{\left(AXK1.AYK1 e^{(j,2)}_{k,2}.e^{-(s,2)}_{h,2} + AXKN.AYK1.e^{(s,2)}_{k,N-1} e^{-(s,2))}_{h,2}\right)}{AXN.AYM} + \frac{\left(AXK1.AYKM.e^{(j,2)}_{k,2}.e^{-(s,2)}_{h,M-1} + AXKN.AYKM.e^{(j,2)}_{k,N-1} e^{-(s,2)}_{h,M-1}\right)}{AXN.AYM},$$

$$C_3 = e^{-(s,4)}_{h,r} - \frac{e^{-(s,4)}_{h,2} AYK1 + e^{-(s,4)}_{h,M-1}.AYKM}{AYM}$$

Eq. (45) gives a system of $(N-4)\times(M-4)$ algebraic equations with $(N-4)\times(M-4)$ unknowns.

5. The free vibration analysis of square plates via G-spline based differential quadrature method

In this section the free vibration analysis of square plates as given in Eq. (30) with $\lambda = 1$ will be solved numerically using the proposed approach given in section four. Two different sets of HB- problems have been considered in order to find the solution of such problems as follows:

Case1:

To construct the approximate solution via G-spline-based differential quadrature method an m-poised HB-problem must be chosen.

In this case we shall take a 5-poised HB- problem given for X and Y respectively by the sets

$$e_1 = \{(1,0),(2,0),(3,0),(4,0),(5,0),(6,0)\},$$

$$e_2 = \{(1,0),(2,0),(3,0),(4,0),(5,0),(6,0)\},$$

with the node points given by Eqs. (27) and (28) taking $N = 6$ and $M = 6$. First,apply the G-spline interpolation-based DQM for the simply supported simply supported simply supported simply supported (SS-SS-SS-SS) boundary conditions, and secondly for the clamped clamped clamped clamped (C-C-C-C) boundary conditions. The natural low of frequency of the (SS-SS-SS-SS) and (C-C-C-C) boundary conditions will be given in Table 1.

The fundamental G-spline functions $L_{10}(x), L_{20}(x), L_{30}(x), L_{40}, L_{50}$ and $L_{60}(x)$ are given in Appendix A.

Case 2:

In this case we shall consider another 5-poised HB sets for X and Y respectively given by:

$$e_1 = \{(1,0),(2,0),(3,0),(4,1),(5,1),(6,0)\},$$

$$e_2 = \{(1,0),(2,0),(3,0),(4,1),(5,1),(6,0)\},$$

with the node points given by Eqs. (27) and (28) with $N = 6$ and $M = 6$, first, apply the G-spline interpolation-based DQM for the (SS-SS-SS-SS) boundary conditions and (C-C-C-C) boundary conditions. The natural low frequency for (SS-SS-SS-SS) and (C-C-C-C) boundary conditions will be given in Table 2.

The fundamental G-spline functions $L_{10}(x), L_{20}(x), L_{30}(x), L_{41}, L_{51}$ and $L_{60}(x)$ are given in Appendix B.

Table 1
Comparison of natural low frequency (Ω) of a square plate using G-spline interpolation-based differential quadrature with the approximate solution given by Shu (2000) using case1.

Boundary conditions	Ω (DQM), M = N = 6	Ω (Shu, 2000), M = N = 6
SS-SS-SS-SS	19.0665	19.0970
C-C-C-C	36.4037	36.4441

Table 2
Comparison of natural low frequency (Ω) of a square plate using G-spline interpolation-based differential quadrature with the approximate solution given by Shu (2000)] using case 2.

Boundary conditions	Ω (DQM), M = N = 6	Ω (Shu, 2000), M = N = 6
SS-SS-SS-SS	19.1797	19.0970
C-C-C-C	36.9222	36.4441

6. Conclusions

It is clear that the G-spline-based differential quadrature can be considered as a generalization to the usual differential quadrature method. Also, from Tables 1 and 2 one can conclude that G-spline based differential quadrature gave accurate results, although a small number of node points have been introduced.

Acknowledgements

Authors are very grateful to the reviewers for carefully reading the paper and for their comments and suggestions which have improved the paper.

Appendix A

$L_{10} = 1 - 16.713x + 78.399x^2 - 143.193x^3 + 93.468x^4 - 55.423(x-0)_+^9 + 110.86(x-0.0955)_+^9 - 110.985(x-0.3456)_+^9 + 111.195(x-0.6546)_+^9 - 111.41(x-0.9046)_+^9 + 55.763(x-1)_+^9$

$L_{20} = 20.379x - 127.056x^2 + 257.962x^3 - 177.142x^4 + 177.142(x-0)_+^9 - 221.75(x-0.0955)_+^9 + 221.999(x-0.3456)_+^9 - 222.419(x-0.6546)_+^9 + 222.85(x-0.9046)_+^9 - 111.541(x-1)_+^9$

$L_{30} = -5.24x + 72.038x^2 - 192.538x^3 + 151.627x^4 - 110.985(x-0)_+^9 + 221.999(x-0.0955)_+^9 - 222.249(x-0.3456)_+^9 + 222.669(x-0.6546)_+^9 - 223.101(x-0.9046)_+^9 + 111.666(x-1)_+^9$

$L_{40} = 2.519x - 37.745x^2 + 129.453x^3 - 120.173x^4 + 111.195(x-0)_+^9 - 222.419(x-0.0955)_+^9 + 222.669(x-0.3456)_+^9 - 223.091(x-0.6546)_+^9 + 223.523(x-0.9046)_+^9 - 111.877(x-1)_+^9$

$L_{50} = -1.681x + 25.66x^2 - 92.759x^3 + 94.766x^4 - 111.41(x-0)_+^9 + 222.85(x-0.0955)_+^9 - 223.101(x-0.3456)_+^9 + 223.523(x-0.6546)_+^9 - 223.956(x-0.09046)_+^9 + 112.094(x-1)_+^9$

$L_{60} = 0.737x - 11.273x^2 + 41.076x^3 - 42.545x^4 + 55.763(x-0)_+^9 - 111.541(x-0.0955)_+^9 + 111.666(x-0.3456)_+^9 - 111.877(x-0.6546)_+^9 + 112.094(x-0.9046)_+^9 - 56.105(x-1)_+^9$-

where $(x-\xi)_+^r = \begin{cases} (x-\xi)^r, & \text{if } x \geqslant \xi. \\ 0, & \text{if } x < \xi. \end{cases}$

Appendix B

$L_{10} = 1 - 17.597x + 90.409x^2 - 174.268x^3 + 116.155x^4 - 42.812(x-0)_+^9 + 68.302(x-0.0955)_+^9 - 25.613(x-0.3458)_+^9 - 26.239(x-0.6546)_+^8 + 6.394(x-0.9046)_+^8 + 0.122(x-1)_+^9$

$L_{20} = 22.056x - 150.009x^2 + 318.05x^3 - 221.722x^4 + 86.24(x-0)_+^9 - 137.588(x-0.0955)_+^9 + 51.595(x-0.3458)_+^9 + 52.856(x-0.6546)_+^8 - 12.881(x-0.9046)_+^8 - 2.221(x-1)_+^9$

$L_{30} = -7.198x + 99.57x^2 - 271.589x^3 + 216.346x^4 - 101.248(x-0)_+^9 + 161.532(x-0.0955)_+^9 - 60.573(x-0.3458)_+^9 - 62.054(x-0.6546)_+^8 + 15.123(x-0.9046)_+^8 + 0.29(x-1)_+^9$

$L_{41} = -0.752x + 11.049x^2 - 36.161x^3 + 32.963x^4 - 19.359(x-0)_+^9 + 30.886(x-0.0955)_+^9 - 11.582(x-0.3458)_+^9 - 11.865(x-0.6546)_+^8 + 2.892(x-0.9046)_+^8 + 0.055(x-1)_+^9$

$L_{51} = -0.606x + 8.835x^2 - 28.158x^3 + 24.221x^4 - 11.705(x-0)_+^9 + 18.674(x-0.0955)_+^9 - 7.002(x-0.3458)_+^9 - 7.174(x-0.6546)_+^8 + 1.748(x-0.9045)_+^8 + 0.033(x-1)_+^9$

$L_{60} = 2.739x - 39.97x^2 + 127.807x^3 - 110.779x^4 + 57.82(x-0)_+^9 - 92.246(x-0.0955)_+^9 + 34.591(x-0.3458)_+^9 + 35.437(x-0.6546)_+^8 - 8.636(x-0.9046)_+^8 - 0.165(x-1)_+^9$

References

Bellman, R., Casti, J., 1971. Differential quadrature and longterm integration. J. Math. Anal. Appl. 34, 235–238.

Bert, C.W., Malik, M., 1996. Differential quadrature method in computational mechanics: a review. Appl. Mech. Rev. 49, 1–28.

Deboor, C., 1978. A Practical Guide to Splines. Springer-Verlag, New York Inc.

Eftekhari, S.A., 2015. A note on mathematical treatment of the Dirac-delta function in the differential quadrature bending and forced vibration analysis of beams and rectangular plates subjected to concentrated loads. Appl. Math. Model. 39 (20), 6223–6242.

Ghasemi, S.E., Hatami, M., Hatami, J., Sahebi, S.A.R., Ganji, D.D., 2016. An efficient approach to study the pulsatile blood flow in femoral and coronary arteries by differential Quadrature Method. Phys. A 443, 406–414.

Ghasemi, M., 2017. High order approximations using spline-based differential quadrature method: implementation to the multi-dimensional PDEs. Appl. Math. Model. https://doi.org/10.1016/j.apm.2017.01.052.

Jalaal, M., Soleimani, S., Domairry, G., Ghasemi, E., Bararnia, H., Barari, F., Mohammadi, A., 2011. Numerical simulation of electric field in complex geometries for different electrode arrangements using meshless local MQ-DQ method. J. Electrostat., 1–8 URL:www.elsevier.com/locate/elstat.

Jiwari, R., Pandit, S., Mittal, R.C., 2012. Numerical simulation of two-dimensional sine-Gordon solitons by differential quadrature method. Comput. Phys. Commun. 183, 600–616.

Jiwari, R., 2015. Lagrange interpolation and modified cubic B-spline differential quadrature methods for solving hyperbolic partial differential equations with Dirichelet and Neumann boundary conditions. Comput. Phys. Commun. https://doi.org/10.1016/j.cpc.2015.03.021.

Korkmaz, A., Dag, I., 2008. A differential quadrature algorithm for simulations of nonlinear Schrödinger equation. Comput. Math. Appl. 56, 2222–2234. URL: www.elsevier.com/locate/camwa.

Mittal, R.C., Dahiyah, S., 2016. A study of quintic B-spline based differential quadrature method for a class of semi-linear Fisher-Kolmogorov equations. Alexandria Eng. J. 55, 2893–2899.

Mittal, R.C., Dahiyah, S., 2017. Numerical simulation of three-dimensional telegraphic equation using cubic B-spline differential quadrature method. Appl. Math. Comput. 313, 442–452.

Pekmen, B., Tezer-Sezgin, M., 2012. Differential quadrature solution of nonlinear Klein-Gordon and sine-Gordon equations. Comput. Phys. Commun. 183 (8), 1702–1713.

Powell, M.J., 1981. Approximation Theory and Methods. Cambridge University Press.

Quan, J.R., Chang C.T., 1989. New insights in solving distributed system equations by the quadrature methods. II. Numerical Experiment, 13,9. pp. 1017–1024.

Ragb, O., Matbuly, M.S., Nassar, M., 2014. Analysis of composite plates using moving least squares differential quadrature method. Appl. Math. Comput. 238, 225–236.

Shu, C., 2000. Differential Quadrature and Its Application in Engineering. Springer-Verlag, London Limited.

Shu, C., Richards, B.E., 1992. Application of generalized differential quadrature to solve two dimensional incompressible Navier Stokes equations. Int. J. Numer. Methods Fluids 15, 791–798.

Shu, C., Xue, H., 1997. Explicit computation of weighting coefficients in the harmonic differential quadrature. J. Sound Vib. 204 (3), 549–555.

Shu, C., Wu, Y.L., 2007. Integrated radial basis functions-based differential quadrature method and its performance. Int. J. Numer. Methods Fluids 53, 969–984.

Shamani, S., Aghdam, M.M., Bahrami, M., 2015. On the free vibration characteristics of postbuckled third-order shear deformable FGM nanobeams including surface effects. Compos. Struct. 121, 377–385.

Shamani, S., Aghdam, M.M., 2017a. Nonlinear vibrations of pre- and post-buckled lipid supramolecular micro/nano-tubules via nonlocal strain gradient elasticity theory. J. Biomech. 65, 49–60.

Shamani, S., Aghdam, M.M., 2017b. Size-dependent nonlinear bending of micro/nano-beams made of nanoporous biomaterials including a refined truncated cube cell. Phys. Lett. A 381, 3818–3830.

Shamani, S., Aghdam, M.M., 2018. Nonlocal strain gradient shell model for axial buckling and postbuckling analysis of magneto-electro-elastic composite nanoshells. Compos. Part B: Eng., 258–274

Schoenberg, I.J., 1968. On the Ahlberg-Nilson extension of spline interpolation: the G-spline and their optimal properties. J. Math. Anal. Appl. 21, 207–231.

Stephen, W., 2002. An introduction to the mathematics and construction of splines. Addix software consultancy limited. Veraion 1.6.

Thoudam, R., 2017. Numerical solutions of coupled Klein Gordon Zakharov equations by quintic B-spline differential quadrature method. Appl. Math. Comput. 307, 50–61.

Zong, Z., Zhang, Y., 2009. Advanced Differential Quadrature Methods. Taylor & Francis Group LLC.