Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **8**, No. 1, pp. 128-144

DOI: 10.4208/aamm.2014.m794
February 2016

Stability Analysis and Order Improvement for Time Domain Differential Quadrature Method

Fangzong Wang*, Xiaobing Liao and Xiong Xie

College of Electrical Engineering & New Energy, China Three Gorges University, Yichang 443002, China

Received 12 October 2014; Accepted (in revised version) 31 December 2014

Abstract. The differential quadrature method has been widely used in scientific and engineering computation. However, for the basic characteristics of time domain differential quadrature method, such as numerical stability and calculation accuracy or order, it is still lack of systematic analysis conclusions. In this paper, according to the principle of differential quadrature method, it has been derived and proved that the weighting coefficients matrix of differential quadrature method meets the important **V**-transformation feature. Through the equivalence of the differential quadrature method and the implicit Runge-Kutta method, it has been proved that the differential quadrature method is A-stable and s-stage s-order method. On this basis, in order to further improve the accuracy of the time domain differential quadrature method, a class of improved differential quadrature method of s-stage $2s$-order has been proposed by using undetermined coefficients method and Padé approximations. The numerical results show that the proposed differential quadrature method is more precise than the traditional differential quadrature method.

AMS subject classifications: 37M05, 65L05, 65L06, 65L20

Key words: Differential quadrature method, numerical stability, order, **V**-transformation, Runge-Kutta method, Padé approximations.

---

1 Introduction

The differential quadrature method (DQM) was first proposed by Bellman and his associates in the early 1970s [1,2], which is used for solving ordinary and partial differential equations. As an analogous extension of the quadrature for integrals, it can be essentially expressed as the values of the derivatives at each grid point as weighted linear sums approximately of the function values at all grid points within the domain under consideration.

*Corresponding author.
Email: fzwang@ctgu.edu.cn (F. Z. Wang)

The differential quadrature method is conceptually simple and the implementation is straightforward. It has been recognized that the differential quadrature method has the capability of producing highly accurate solutions with minimal computational effort [3, 4] when the method is applied to problems with globally smooth solutions. So far, the differential quadrature method has been widely applied to boundary-value problems in many areas of engineering and science, such as transport process [5], structural mechanics [6–8], calculation of transmission line transient response [9, 10], etc. [11] made the first attempt to apply the differential quadrature method for time domain discretization. Subsequently, the differential quadrature method has been extended to solve initial value problems in the time domain, such as, time-dependent diffusion problems [12], transient elastodynamic problems [13] and dynamic problems [14–16]. A comprehensive review of the chronological development of the differential quadrature method can be found in [4, 11]. Though the differential quadrature method has been successfully applied in so many fields, for the basic characteristics of the method, such as numerical stability and calculation accuracy or order, not much work about it has been done in this area for the differential quadrature method. According to Fung [17], using Lagrange interpolation functions as test functions, the time domain differential quadrature has been shown to be equivalent to the recast implicit Runge-Kutta method [18–20], besides, some low-order algorithms were discussed in detail. However, the method used by Fung is not the traditional sense of differential quadrature method, but involved post-processing (i.e., numerical solution at the end of grid points adopts polynomial extrapolation).

In this paper, using general polynomial as test functions [21], the weighting coefficients matrix of differential quadrature method is proved to satisfy $\mathbf{V}$-transformation [19, 22]. The equivalent implicit Runge-Kutta method is constructed through the differential quadrature method. Hence, making use of Butcher fundamental order theorem and the method of linear stability analysis [18–20], the basic characteristics of differential quadrature method can be systematically analysed. Unfortunately, the differential quadrature method is only the method of s-stage s-order and A-stable. Consequently, the differential quadrature method can't yield higher accurate solutions to the boundary-value problems with fewer computational efforts. Based on above deduction, the method of undetermined coefficients is used to make the stability function of the equivalent Runge-Kutta method become the diagonal Padé approximations to the exponential function [19, 20]. Therefore, a class of improved differential quadrature method of s-stage $2s$-order has been derived.

The manuscript is arranged as follows. In Section 2, the weighting coefficients matrix of traditional differential quadrature method using general polynomial as test functions is briefly discussed. In Section 3, the equivalent relationship between differential quadrature method and Runge-Kutta method is deduced. In Section 4, the stability and order characteristics of differential quadrature method are studied. A class of improved differential quadrature method of s-stage $2s$-order and A-stable is proposed in Section 5. In Section 6, the transient response of a double-degree-of-freedom system is computed, which is given to verify the computational accuracy with the defined three grid points.

Conclusions are then given in Section 7.

2 Traditional differential quadrature method

Suppose function $f(x)$ is sufficiently smooth in the whole interval, there are $(s+1)$ grid points with coordinates as c_i, $i\in(0,s)$. The first order derivative $f^{(1)}(c_i)$ at each sampling grid point c_i, $i\in(1,s)$, is approximated by a linear sum of all the function values in the whole domain, that is

$$f^{(1)}(c_i)=\sum_{j=0}^{s} g_{ij} f(c_j), \quad i\in(1,s), \tag{2.1}$$

where $f(c_i)$ represent function values at a grid point c_i, g_{ij} is the weighting coefficients.

In order to compute the weighting coefficients g_{ij} in Eq. (2.1), the test functions can be chosen as

$$r_k(x)=x^k, \quad (k=0,1,\cdots s). \tag{2.2}$$

Substituting Eq. (2.2) into Eq. (2.1) gives

$$k=0, \quad 0=\sum_{j=0}^{s} g_{ij}, \quad i\in(1,s), \tag{2.3a}$$

$$\sum_{j=0}^{s} g_{ij} c_j^k = k\cdot c_i^{k-1}, \quad i,k\in(1,s), \tag{2.3b}$$

Eq. (2.3a) can be expanded into matrix form as

$$\begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1s} \\ g_{21} & g_{22} & \cdots & g_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ g_{s1} & g_{s2} & \cdots & g_{ss} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} g_{10} \\ g_{20} \\ \vdots \\ g_{s0} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{2.4}$$

Let

$$\mathbf{G}_0=\begin{pmatrix} g_{10} \\ g_{20} \\ \vdots \\ g_{s0} \end{pmatrix}, \quad \mathbf{G}=\begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1s} \\ g_{21} & g_{22} & \cdots & g_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ g_{s1} & g_{s2} & \cdots & g_{ss} \end{pmatrix}, \quad \mathbf{e}=\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}. \tag{2.5}$$

Using Eq. (2.5), Eq. (2.4) can be rewritten as

$$\mathbf{G}_0 \equiv -\mathbf{G}\mathbf{e}. \tag{2.6}$$

From $k=1,2,\cdots,s$, Eq. (2.3b) can be expanded as

$$\begin{cases} g_{i0}c_0+g_{i1}c_1+g_{i2}c_2+\cdots+g_{is}c_s=1, \\ g_{i0}c_0^2+g_{i1}c_1^2+g_{i2}c_2^2+\cdots+g_{is}c_s^2=2c_i, \\ \vdots \\ g_{i0}c_0^s+g_{i1}c_1^s+g_{i2}c_2^s+\cdots+g_{is}c_s^s=sc_i^{s-1}. \end{cases} \tag{2.7}$$

Since initial grid point c_0 is usually defined as 0, Eq. (2.7) reduces to

$$\begin{cases} g_{i1}c_1+g_{i2}c_2+\cdots+g_{is}c_s=1, \\ g_{i1}c_1^2+g_{i2}c_2^2+\cdots+g_{is}c_s^2=2c_i, \\ \vdots \\ g_{i1}c_1^s+g_{i2}c_2^s+\cdots+g_{is}c_s^s=sc_i^{s-1}. \end{cases} \tag{2.8}$$

From $i=1,2,\cdots s$, Eq. (2.8) can be also expanded into matrix form as

$$\begin{aligned} \mathbf{G}\begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & & c_s^s \end{pmatrix} &= \begin{pmatrix} 1 & 2c_1 & \cdots & sc_1^{s-1} \\ 1 & 2c_2 & \cdots & sc_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 2c_s & \cdots & sc_s^{s-1} \end{pmatrix} \\ &= \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix}\begin{pmatrix} 1 & & & \\ & 2 & & \\ & & \ddots & \\ & & & s \end{pmatrix}. \end{aligned} \tag{2.9}$$

Vandermonde matrix $\mathbf{V}$ is defined as follows

$$\mathbf{V}=\begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix}. \tag{2.10}$$

Making use of Eq. (2.10), Eq. (2.9) can be expressed as

$$\begin{aligned} \mathbf{G}^{-1}\mathbf{V} &= \begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & \cdots & c_s^s \end{pmatrix}\begin{pmatrix} 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{pmatrix} \\ &= \begin{pmatrix} 1 & c_1 & c_1^2 & \cdots & c_1^s \\ 1 & c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & c_s^2 & \cdots & c_s^s \end{pmatrix}\begin{pmatrix} 0 & 0 & \cdots & 0 \\ 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{pmatrix}. \end{aligned} \tag{2.11}$$

Finally, it can be inferred that

$$\mathbf{V}^{-1}\mathbf{G}^{-1}\mathbf{V}=\mathbf{A}_s, \tag{2.12}$$

where $\mathbf{A}_s$ is defined as

$$\mathbf{A}_s=\begin{pmatrix} 0 & 0 & 0 & \cdots & \alpha_1 \\ 1 & 0 & 0 & \cdots & \alpha_2 \\ 0 & \dfrac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dfrac{1}{s-1} & \alpha_s \end{pmatrix}, \tag{2.13}$$

with

$$\boldsymbol{\alpha}_s=[\alpha_1,\alpha_2,\cdots,\alpha_s]^{\mathrm{T}}=\frac{1}{s}\mathbf{V}^{-1}\mathbf{c}^s, \quad \mathbf{c}^s=[c_1^s,c_2^s,\cdots,c_s^s]^{\mathrm{T}}. \tag{2.14}$$

Eq. (2.12), i.e., $\mathbf{G}=\mathbf{V}\,\mathbf{A}_s^{-1}\mathbf{V}^{-1}$, is called the implicit expression of the weighting coefficients matrix, and is also called $\mathbf{V}$-transformation.

When the grid points have been selected, the weighting coefficients matrices $\mathbf{G}$ and $\mathbf{G}_0$ are easy to calculate with the above formula. Obviously, the weighting coefficients of the differential quadrature method depend on test functions and the distribution of grid points, but are independent of some specific problems. There are four typical grid points distribution: Legendre grid, Chebyshev grid, Chebyshev-Gauss-Lobatto grid and Uniform grid [17, 21]. Legendre grid is a special case in traditional quadrature method. This paper will focus on the latter three kinds of commonly used grid points, which are defined as follows:

1) Chebyshev grid points

$$c_0=0, \quad c_k=\frac{1}{2}\Big(1-\cos\Big(\frac{2k-1}{2s-2}\pi\Big)\Big), \quad k\in(1,s-1), \quad c_s=1.$$

2) Chebyshev-Gauss-Chebysgev grid points

$$c_k=\frac{1}{2}\Big(1-\cos\Big(\frac{k}{s}\pi\Big)\Big), \quad k\in(0,s).$$

3) Uniform grid points

$$c_k=\frac{k}{s}, \quad k\in(0,s).$$

3 The equivalence of differential quadrature method and Runge-Kutta method

In order to analyse the numerical stability and order of differential quadrature method, the time domain differential quadrature method can be transformed into equivalent implicit Runge-Kutta method. Consider the following ordinary differential equations

$$\begin{cases} \dfrac{d\mathbf{x}}{dt}=\mathbf{f}(t,\mathbf{x}), \quad 0<t\le T, \\ \mathbf{x}(t=0)=\mathbf{x}_0. \end{cases} \tag{3.1}$$

In the following, t_n, t_{n+1} represent respectively the beginning and the end points at each step. $h=t_{n+1}-t_n$ denotes the step size. The time interval $[t_n,t_{n+1}]$ will be normalized. i.e., $c=(t-t_n)/h$, $t\in[t_n,t_{n+1}]$. At the same time, Eq. (3.1) can be made in the standard normalized form as

$$\frac{d}{dc}\mathbf{x}=h\,\mathbf{f}(t_n+ch,\widetilde{\mathbf{x}}),\quad \widetilde{\mathbf{x}}=\mathbf{x}(t_n+ch), \tag{3.2}$$

then, using s-stage differential quadrature method to solve Eq. (3.2) yeilds

$$\mathbf{G}\begin{pmatrix}\widetilde{\mathbf{x}}_1\\ \vdots\\ \widetilde{\mathbf{x}}_s\end{pmatrix}+\mathbf{G}_0\,\mathbf{x}_n=h\begin{pmatrix}\mathbf{f}(t_n+c_1h,\widetilde{\mathbf{x}}_1)\\ \vdots\\ \mathbf{f}(t_n+c_sh,\widetilde{\mathbf{x}}_s)\end{pmatrix}, \tag{3.3}$$

where $\widetilde{\mathbf{x}}_i=\mathbf{x}(t_n+c_ih)$, $i\in(1,s)$. Since $\mathbf{G}_0\equiv\mathbf{G}\,\mathbf{e}$, Eq. (3.3) reduces to

$$\begin{pmatrix}\widetilde{\mathbf{x}}_1\\ \vdots\\ \widetilde{\mathbf{x}}_s\end{pmatrix}=\mathbf{x}_n\begin{pmatrix}1\\ \vdots\\ 1\end{pmatrix}+h\,\mathbf{G}^{-1}\begin{pmatrix}\mathbf{f}(t_n+c_1h,\widetilde{\mathbf{x}}_1)\\ \vdots\\ \mathbf{f}(t_n+c_sh,\widetilde{\mathbf{x}}_s)\end{pmatrix}. \tag{3.4}$$

Let

$$\mathbf{G}^{-1}=\mathbf{A}=[a_{ij}],\quad i,j\in(1,s). \tag{3.5}$$

Clearly, making use of Eqs. (2.12) and (3.5) leads to

$$\mathbf{A}=\mathbf{V}\mathbf{A}_s\mathbf{V}^{-1}. \tag{3.6}$$

Therefore, the weigthing coefficients matrix $\mathbf{A}$ also satisfies $\mathbf{V}$-transformation. It can be inferred from Eq. (3.4)

$$\widetilde{\mathbf{x}}_i=\mathbf{x}_n+h\sum_{j=1}^{s}a_{ij}\,\mathbf{f}(t_n+c_jh,\widetilde{\mathbf{x}}_j),\quad i\in(1,s). \tag{3.7}$$

Since $c_s=1$, $t_n+c_sh=t_n+h=t_{n+1}$, therefore, $\mathbf{x}_i(i=s)$ is the approximate solution at the end of the step. Then, $\widetilde{\mathbf{x}}_s$ can be rewritten as the following form

$$\widetilde{\mathbf{x}}_s=\mathbf{x}_{n+1}=\mathbf{x}_n+h\sum_{j=1}^{s}a_{sj}\mathbf{f}(t_n+c_jh,\widetilde{\mathbf{x}}_j)=\mathbf{x}_n+h\sum_{j=1}^{s}b_j\mathbf{f}(t_n+c_jh,\widetilde{\mathbf{x}}_j), \tag{3.8}$$

where $b_j=a_{sj}$, $j\in(1,s)$. It can be seen that Eqs. (3.7) and (3.8) are the standard forms for an s-stage Runge-Kutta method. Since, the equivalent Runge-Kutta method is a reducible method [22]. In fact, the traditional differential quadrature method generally doesn't involve post-processing, so the Runge-Kutta method converted from traditional differential quadrature method will naturally become a reducible method. The Runge-Kutta method can be conveniently summarized in the Butcher tableau as

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}, \tag{3.9}$$

where $\mathbf{b}^{\mathrm{T}}=(b_1,b_2,\cdots,b_s)=(a_{s1},a_{s2},\cdots,a_{ss})$.

4 Stability and order analysis of the differential quadrature method

The stablity and accuracy characteristics of the newly resultant Runge-Kutta method will be investigated next. From Eqs. (2.12) and (3.5), it can be inferred that

$$\begin{aligned} &\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{ss} \end{pmatrix} \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix} \\ =&\begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & & c_s^s \end{pmatrix} \begin{pmatrix} 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{pmatrix}. \end{aligned} \tag{4.1}$$

Eq. (4.1) reduces to

$$\sum_{j=1}^{s} a_{ij}c_j^{k-1}=\frac{c_i^k}{k}, \quad i\in(1,s), \quad k\in(1,s). \tag{4.2}$$

On the other hand, since $b_j=a_{sj}$, $j\in(1,s)$ and $c_s=1$, from Eq. (4.1), it can be seen that

$$\mathbf{b}^{\mathrm{T}}\mathbf{V}=\Big[1,\frac{1}{2},\cdots,\frac{1}{s}\Big]. \tag{4.3}$$

Similarly, Eq. (4.3) reduces to

$$\sum_{i=1}^{s} b_i c_i^{k-1}=\frac{1}{k}, \quad k\in(1,s). \tag{4.4}$$

Obviously, from Eqs. (4.2) and (4.4), it has been shown that the equivalent Runge-Kutta method at least satisfies simplifying assumptions $C(s)$ and $B(s)$. Furthermore, it can be verified that the equivalent Runge-Kutta method satisfies simplifying assumptions $D(0)$.

From Theorem 5.1 on pp. 71 in [19], it can be concluded that the implicit Runge-Kutta method or the corresponding differential quadrature method is s-stage s-order.

The stability function of the corresponding differential quadrature method, i.e., $R(z)$, is given by the formula

$$R(z)=1+z\mathbf{b}^{\mathrm{T}}(\mathbf{I}-z\mathbf{A})^{-1}\mathbf{e}=\frac{\det(\mathbf{I}+z(\mathbf{e}\mathbf{b}^{\mathrm{T}}-\mathbf{A}))}{\det(\mathbf{I}-z\mathbf{A})}, \tag{4.5}$$

where, as usual, $\mathbf{I}$ is the identity matrix of dimension s. Due to grid points's asymmetric distribution, the equivalent implicit Runge-Kutta method is not a symmetrical method. As a result, there is an unique adjoint method (also called reflected method) [18, 20], which is defined as

$$\begin{array}{c|c} \mathbf{c}^* & \mathbf{A}^* \\ \hline & (\mathbf{b}^*)^{\mathrm{T}} \end{array},$$

satisfying

$$\begin{cases} \mathbf{c}^*=\mathbf{e}-\mathbf{P}\mathbf{c}, \\ \mathbf{P}\mathbf{A}^*\mathbf{P}=\mathbf{e}\mathbf{b}^{\mathrm{T}}-\mathbf{A}, \\ \mathbf{b}^*=\mathbf{P}\mathbf{b}, \end{cases} \tag{4.6}$$

where $\mathbf{P}$ is

$$\mathbf{P}=\begin{pmatrix} & & & 1 \\ & & 1 & \\ & \iddots & & \\ 1 & & & \end{pmatrix}\in\mathbf{R}^{s\times s}. \tag{4.7}$$

Futhermore, from Theorem 343B on pp. 221 in [20], if the original method satisfies the simplifying assumptions $C(s)$ and $B(s)$, the adjoint method also satisfies the same simplifying assumptions. Hence, the adjoint method enjoys $\mathbf{V}$-transformation

$$\mathbf{A}^*=\mathbf{V}^*\mathbf{A}_s^*(\mathbf{V}^*)^{-1},\quad \mathbf{V}^*=\begin{pmatrix} 1 & c_1^* & \cdots & (c_1^*)^{s-1} \\ 1 & c_2^* & \cdots & (c_2^*)^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s^* & \cdots & (c_s^*)^{s-1} \end{pmatrix}, \tag{4.8}$$

where $\mathbf{A}_s^*$ is also defined as

$$\mathbf{A}_s^*=\begin{pmatrix} 0 & 0 & 0 & \cdots & \beta_1 \\ 1 & 0 & 0 & \cdots & \beta_2 \\ 0 & \frac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s-1} & \beta_s \end{pmatrix}, \tag{4.9}$$

with

$$\boldsymbol{\beta}_s=[\beta_1,\beta_2,\cdots,\beta_s]^{\mathrm{T}}=\frac{1}{s}(\mathbf{V}^*)^{-1}(\mathbf{c}^*)^s,\quad (\mathbf{c}^*)^s=[(c_1^*)^s,(c_2^*)^s,\cdots,(c_s^*)^s]^{\mathrm{T}}. \tag{4.10}$$

So Eq. (4.5) can be reduced to

$$\begin{aligned}R(z)&=\frac{\det(\mathbf{I}+z(\mathbf{eb}^{\mathrm{T}}-\mathbf{A}))}{\det(\mathbf{I}-z\mathbf{A})}=\frac{\det(\mathbf{I}+z(\mathbf{PA}^*\mathbf{P}^{-1}))}{\det(\mathbf{I}-z(\mathbf{VA}_s\mathbf{V}^{-1}))}=\frac{\det(\mathbf{I}+z\mathbf{A}^*)}{\det(\mathbf{I}-z\mathbf{A}_s)}\\&=\frac{\det(\mathbf{I}+z(\mathbf{V}^*\mathbf{A}_s(\mathbf{V}^*)^{-1}))}{\det(\mathbf{I}-z\mathbf{A}_s)}=\frac{\det(\mathbf{I}+z\mathbf{A}_s^*)}{\det(\mathbf{I}-z\mathbf{A}_s)}.\end{aligned} \tag{4.11}$$

Because both $\mathbf{A}_s$ and $\mathbf{A}_s^*$ are a class of special matrices, Eq. (4.11) can be evaluated as

$$R(z)=\frac{\det(\mathbf{I}+z\mathbf{A}_s^*)}{\det(\mathbf{I}-z\mathbf{A}_s)}=\frac{1-\sum\limits_{k=s}^{1}\beta_k\frac{(k-1)!}{(s-1)!}(-z)^{s-k+1}}{1-\sum\limits_{k=s}^{1}\alpha_k\frac{(k-1)!}{(s-1)!}z^{s-k+1}}. \tag{4.12}$$

Let

$$\prod_{i=1}^{s}(x-c_i)=\sum_{k=0}^{s}\gamma_k x^k,\quad \prod_{i=1}^{s}(x-c_i^*)=\sum_{k=0}^{s}\gamma_k^* x^k. \tag{4.13}$$

It can be inferred from Eqs. (2.14) and (4.10)

$$\alpha_k=-\gamma_{k-1}/s,\quad \beta_k=-\gamma_{k-1}^*/s,\quad k\in(1,s). \tag{4.14}$$

For example, s=3,

$$\begin{cases}\alpha_1=(c_1c_2c_3)/3,\\ \alpha_2=-(c_1c_2+c_1c_3+c_2c_3)/3,\\ \alpha_3=(c_1+c_2+c_3)/3,\end{cases}\qquad \begin{cases}\beta_1=(c_1^*c_2^*c_3^*)/3,\\ \beta_2=-(c_1^*c_2^*+c_1^*c_3^*+c_2^*c_3^*)/3,\\ \beta_3=(c_1^*+c_2^*+c_3^*)/3.\end{cases} \tag{4.15}$$

Since $c_s=1$, $c_1^*=1-c_s=0$, and $c_k^*=c_{k-1}$, $k\in(2,s)$, therefore

1. When s is even, $a_{2i-1}<0$, $a_{2i}>0$, $i\in(1,s/2)$; except for $\beta_1=0$, the elements of $\boldsymbol{\beta}_s$ also have similar properties.
2. When s is odd, $a_{2i-1}>0$, $a_{2i}<0$, $i\in(1,s/2)$, $\alpha_s<0$; except for $\beta_1=0$, the elements of $\boldsymbol{\beta}_s$ also have similar properties.
3. $\alpha_k^2-\beta_k^2>0$, $k\in(1,s)$.

Hence, It has been shown that the stability function of equivalent Runge-Kutta method is A-acceptability of p-order ($p\geq s-1$) rational approximation to exponential function [23]. Therefore, the corresponding differential quadrature method is A-stable.

In the following, the three-stage differential quadrature method using Uniform grid points will be given as an example. When s=3, c_1, c_2 and c_3 are given by 1/3, 2/3, 1. It can be worked out that matrices $\mathbf{G}_0$ and $\mathbf{G}$ are given by

$$\mathbf{G}_0=\begin{pmatrix} -1 \\ \dfrac{1}{2} \\ -1 \end{pmatrix},\quad \mathbf{G}=\begin{pmatrix} -\dfrac{3}{2} & 3 & -\dfrac{1}{2} \\ -3 & \dfrac{3}{2} & 1 \\ \dfrac{9}{2} & -9 & \dfrac{11}{2} \end{pmatrix}, \tag{4.16}$$

and the Butcher tableau of equivalent Runge-Kutta method would be

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} \frac{1}{3} & \frac{23}{36} & -\frac{4}{9} & \frac{5}{36} \\ \frac{2}{3} & \frac{7}{9} & -\frac{2}{9} & \frac{1}{9} \\ 1 & \frac{3}{4} & 0 & \frac{1}{4} \\ \hline & \frac{3}{4} & 0 & \frac{1}{4} \end{array}. \tag{4.17}$$

$\mathbf{V}$-transformation of matrix is

$$\mathbf{A}=\begin{pmatrix} 1 & \dfrac{1}{3} & \dfrac{1}{9} \\ 1 & \dfrac{2}{3} & \dfrac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}\begin{pmatrix} 0 & 0 & \dfrac{2}{27} \\ 1 & 0 & -\dfrac{11}{27} \\ 0 & \dfrac{1}{2} & \dfrac{2}{3} \end{pmatrix}\begin{pmatrix} 1 & \dfrac{1}{3} & \dfrac{1}{9} \\ 1 & \dfrac{2}{3} & \dfrac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}^{-1}, \tag{4.18}$$

the stability function of equivalent Runge-Kutta method is

$$R(z)=\frac{1+\frac{1}{3}z+\frac{1}{27}z^2}{1-\frac{2}{3}z+\frac{11}{54}z^2-\frac{1}{27}z^3}, \tag{4.19}$$

and the Butcher tableau of the adjoint method would be

$$\begin{array}{c|c} \mathbf{c}^* & \mathbf{A}^* \\ \hline & (\mathbf{b}^*)^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{3} & -\frac{1}{36} & \frac{2}{9} & \frac{5}{36} \\ \frac{2}{3} & \frac{1}{9} & \frac{4}{9} & \frac{1}{4} \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}. \tag{4.20}$$

$\mathbf{V}$-transformation of matrix $\mathbf{A}_s^*$ is

$$\mathbf{A}^*=\begin{pmatrix} 1 & 0 & 0 \\ 1 & \dfrac{1}{3} & \dfrac{1}{9} \\ 1 & \dfrac{2}{3} & \dfrac{4}{9} \end{pmatrix}\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & -\dfrac{2}{27} \\ 0 & \dfrac{1}{2} & \dfrac{1}{3} \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 1 & \dfrac{1}{3} & \dfrac{1}{9} \\ 1 & \dfrac{2}{3} & \dfrac{4}{9} \end{pmatrix}^{-1}. \tag{4.21}$$

5 Improved differential quadrature method

Based on the above deduction, the traditional differential quadrature method is the method of s-stage s-order. Compared with multi-stage and high-order Runge-Kutta method, for example, Gauss method (s-stage $2s$-order method), it has the disadvantage of lower precision. From Eq. (4.11), it can be seen that the stability function of equivalent Runge-Kutta method will be determined by $\mathbf{A}_s$ and $\mathbf{A}_s^*$. Suppose $\mathbf{A}_s=\mathbf{A}_s^*=\mathbf{A}_s$, without changing b and $\mathbf{V}$, a new Runge-Kutta method

$$\begin{array}{c|c} \mathbf{c} & \widetilde{\mathbf{A}} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array}$$

is redefined as

$$\widetilde{\mathbf{A}}=(\widetilde{\mathbf{G}})^{-1}=\mathbf{V}\begin{pmatrix} 0 & 0 & 0 & \cdots & \gamma_1 \\ 1 & 0 & 0 & \cdots & \gamma_2 \\ 0 & \frac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s-1} & \gamma_s \end{pmatrix}\mathbf{V}^{-1}=\mathbf{V}\widetilde{\mathbf{A}}_s\mathbf{V}^{-1}. \tag{5.1}$$

Then, the stability function of new Runge-Kutta method becomes

$$R(z)=\frac{\det(\mathbf{I}+z\widetilde{\mathbf{A}}_s)}{\det(\mathbf{I}-z\widetilde{\mathbf{A}}_s)}. \tag{5.2}$$

From Eq. (5.1), it can be seen that the last column elements of $\widetilde{A}_s$ determine the stability function. To improve the order of new Runge-Kutta method, undetermined coefficients can be selected so that the stability function of new Runge-Kutta method becomes the diagonal Padé approximations to the exponential function (defining as $e^z|_s^s$):

$$R(z)=\frac{\det(\mathbf{I}+z\widetilde{\mathbf{A}}_s)}{\det(\mathbf{I}-z\widetilde{\mathbf{A}}_s)}=e^z|_s^s. \tag{5.3}$$

By comparing the coefficients on both sides of Eq. (5.3), undetermined coefficients γ can be conveniently obtained as

$$\begin{aligned} s&=2, &\quad \gamma_2&=[-1/12,1/2],\\ s&=3, &\quad \gamma_3&=[1/60,-1/5,1/2],\\ s&=4, &\quad \gamma_4&=[-1/280,1/14,-9/28,1/2],\cdots. \end{aligned}$$

After getting $\gamma_s=(\gamma_1\cdots\gamma_s)^{\mathrm{T}}$, so that coefficients matrix $\widetilde{\mathbf{A}}$ or $\widetilde{\mathbf{G}}$ can also be easily computed through using Eq. (5.1). Therefore, a class of new Runge-Kutta method of s-stage

$2s$-order has been successfully constructed. In other words, a class of improved differential quadrature method of s-stage $2s$-order has been derived. Besides, the adjoint method of new Runge-Kutta method is also s-stage 2s-order.

If the traditional differential quadrature method using Legendre grid points is converted into Runge-Kutta method, this method is Gauss method [14, 17]. It well known that Gauss method is a symplectic method of s-stage $2s$-order and its stability function is also diagonal Pad approximations to the exponential function. Hence, the traditional differential quadrature method using Legendre grid points is a special case so that its order is impossible to be improved by the method presented in this paper. This is why traditional differential quadrature method using Legendre grid points is not discussed in this paper. Symplectic Gauss method is more suitable for integrations over long time duration (especially for Hamiltonian systems) [24–26].

Take the same as above, the improved differential quadrature method using Uniform grid points will be given as an example. It can be worked out that the new matrices $\widetilde{\mathbf{G}}_0$ and $\widetilde{\mathbf{G}}$ are given by

$$\widetilde{\mathbf{G}}_0=\begin{pmatrix} \frac{4}{3} \\ \frac{4}{3} \\ -12 \end{pmatrix}, \quad \widetilde{\mathbf{G}}=\begin{pmatrix} -\frac{17}{2} & 10 & -\frac{17}{6} \\ -\frac{11}{2} & 4 & \frac{1}{6} \\ \frac{75}{2} & -42 & \frac{33}{2} \end{pmatrix}, \tag{5.4}$$

and the Butcher tableau of new Runge-Kutta method would be

$$\begin{array}{c|c} \mathbf{c} & \widetilde{\mathbf{A}} \\ \hline & \mathbf{b}^T \end{array} = \begin{array}{c|ccc} \frac{1}{3} & \frac{73}{120} & -\frac{23}{60} & \frac{13}{120} \\ \frac{2}{3} & \frac{97}{120} & -\frac{17}{60} & \frac{17}{120} \\ 1 & \frac{27}{40} & \frac{3}{20} & \frac{7}{40} \\ \hline & \frac{3}{4} & 0 & \frac{1}{4} \end{array}. \tag{5.5}$$

$\mathbf{V}$-transformation of matrix $\widetilde{A}$ is

$$\widetilde{\mathbf{A}}=\begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}\begin{pmatrix} 0 & 0 & \frac{1}{60} \\ 1 & 0 & -\frac{1}{5} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}\begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}^{-1}. \tag{5.6}$$

Then, Butcher tableau of the adjoint method would be

$$\begin{array}{c|c} \mathbf{c} & \widetilde{\mathbf{A}}^* \\ \hline & (\mathbf{b}^*)^T \end{array} = \begin{array}{c|ccc} 0 & \frac{3}{40} & -\frac{3}{20} & \frac{3}{40} \\ \frac{1}{3} & \frac{13}{120} & \frac{17}{60} & -\frac{7}{120} \\ \frac{2}{3} & \frac{17}{120} & \frac{23}{60} & \frac{17}{120} \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}. \tag{5.7}$$

$\mathbf{V}$-transformation of matrix $\widetilde{\mathbf{A}}^*$ is

$$\widetilde{\mathbf{A}}^* = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix} \begin{pmatrix} 0 & 0 & \frac{1}{60} \\ 1 & 0 & -\frac{1}{5} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix}^{-1}. \tag{5.8}$$

6 Numerical examples

Consider a two-degree-of-freedom system governed by

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{Bmatrix} + \begin{pmatrix} 6 & -2 \\ -2 & 4 \end{pmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 10 \end{Bmatrix}. \tag{6.1}$$

With initial condition

$$\begin{cases} x_1(0)=0, \\ x_2(0)=0, \end{cases} \qquad \begin{cases} \dot{x}_1(0)=0, \\ \dot{x}_2(0)=0, \end{cases} \tag{6.2}$$

the analytical solution of the problem is

$$\begin{cases} x_1 = 1 - \dfrac{5}{3}\cos(\sqrt{2}t) + \dfrac{2}{3}\cos(\sqrt{5}t), \\ x_2 = 3 - \dfrac{5}{3}\cos(\sqrt{2}t) - \dfrac{4}{3}\cos(\sqrt{5}t). \end{cases} \tag{6.3}$$

The differential quadrature method can be used to solve Eq. (6.1) and the solution procedure can look over [11]. The following three Figures show the displacement error trajectories comparison between traditional differential quadrature method and improved differential quadrature method with the same step size $h=0.5$s. In these figures, the analytical solution at each step is used for comparison. From Figs. 1-3, it can be seen that improved differential quadrature method is two orders of magnitude higher than traditional differential quadrature method. The error of improved differential quadrature method range between 10^{-5} and 10^{-4}, even with a large step size $h=0.5$s.

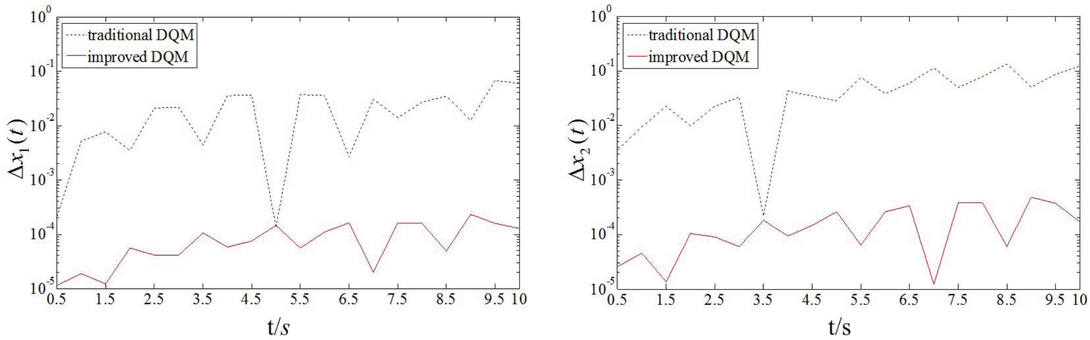

Figure 1: Error trajectories comparison of different DQM using Chebyshev grid points ($h=0.5$s, $s=3$).

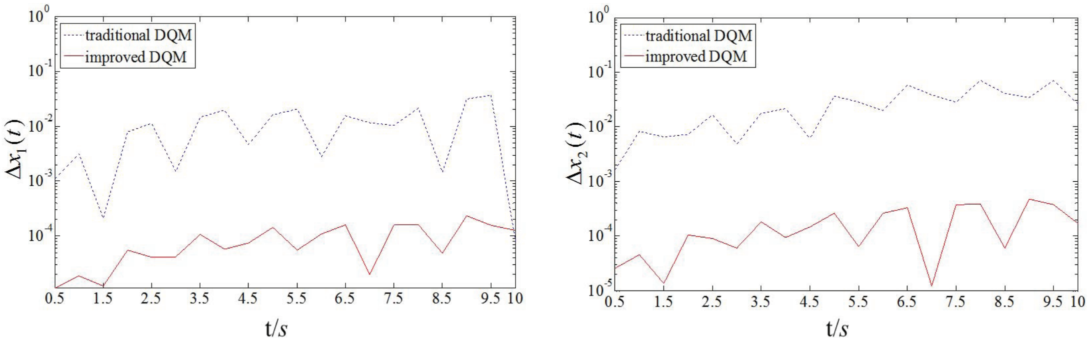

Figure 2: Error trajectories comparison of different DQM using Chebyshev-Gauss-Lobatto grid points ($h=0.5$s, $s=3$).

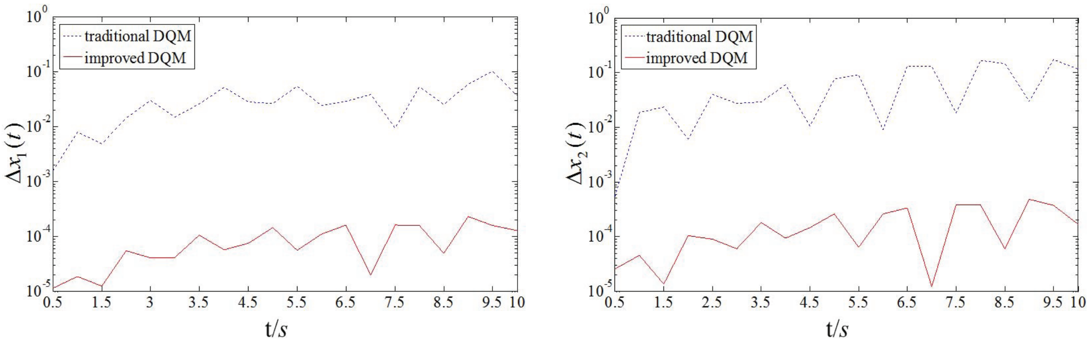

Figure 3: Error trajectories comparison of different DQM using Uniform grid points ($h=0.5$s, $s=3$).

7 Conclusions

In this paper, the stability and order of the time domain differential quadrature method are systematically studied in detail and a new class of differential quadrature method of s-stage $2s$-order is proposed. From the above analysis and derivation, the following conclusions can be made:

1. Using general polynomial as test functions, the weighting coefficients matrix of differential quadrature method satisfies **V**-transformation. **V**-transformation plays an extremely important role in the analysis of basic characteristics of differential quadrature method and its improvement.
2. The traditional differential quadrature method can be transformed into equivalent Runge-Kutta method of A-stable and s-stage s-order. Therefore, compared with the single-stage and low-order numerical integral methods, even with a small number of grid points, the traditional differential quadrature method can also gives more accurate solutions. This is the main mechanism that the differential quadrature method has been successfully applied to scientific and engineering computation.
3. Finally, by making the stability function of equivalent Runge-Kutta method become

the diagonal Padé approximations to the exponential function, a class of improved differential quadrature method of s-stage $2s$-order and A-stable is proposed. Hence, the improved differential quadrature method can be extended to multi-degree-of-freedom time domain dynamic systems, which can produce higher accurate solutions at lower computational cost.

4. With regard to the traditional differential quadrature method, using Legendre grid points is the method of s-stage $2s$-order and A-stable. So that it is better to use Legendre grid points for traditional differential quadrature method. The improved differential quadrature method using other three types grid points has the same order and stability. Therefore, there are four kinds of differential quadrature method of s-stage $2s$-order that can be applied.

Appendix

$s=2$ (**c** for 3 types of grid points is the same, $c_1=1/2$, $c_2=1$)

$$\widetilde{\mathbf{G}}_0=\begin{pmatrix}0\\6\end{pmatrix},\quad \widetilde{\mathbf{G}}=\begin{pmatrix}-2&2\\-14&8\end{pmatrix},\quad \begin{array}{c|c}\mathbf{c}&\widetilde{\mathbf{A}}\\\hline &\mathbf{b}^T\end{array}=\begin{array}{c|cc}\frac{1}{2}&\frac{2}{3}&-\frac{1}{6}\\1&\frac{7}{6}&-\frac{1}{6}\\\hline &1&0\end{array}.$$

$s=3$ (Chebyshev grid points: $c_1=1/2-\sqrt{2}/4$, $c_2=1/2+\sqrt{2}/4$, $c_3=1$)

$$\widetilde{\mathbf{G}}_0=\begin{pmatrix}-\frac{9}{2}\\-\frac{9}{2}\\-12\end{pmatrix},\quad \widetilde{\mathbf{G}}=\begin{pmatrix}-4+\frac{11\sqrt{2}}{2}&4-\frac{3\sqrt{2}}{2}&\frac{9}{2}-4\sqrt{2}\\4+\frac{3\sqrt{2}}{2}&-4-\frac{11\sqrt{2}}{2}&\frac{9}{2}+4\sqrt{2}\\-4-15\sqrt{2}&-4+15\sqrt{2}&20\end{pmatrix},$$

$$\begin{array}{c|c}\mathbf{c}&\widetilde{\mathbf{A}}\\\hline &\mathbf{b}^T\end{array}=\begin{array}{c|ccc}\frac{1}{2}-\frac{\sqrt{2}}{4}&\frac{29}{60}-\frac{53\sqrt{2}}{240}&\frac{11}{60}-\frac{43\sqrt{2}}{240}&-\frac{1}{6}+\frac{3\sqrt{2}}{20}\\\frac{1}{2}+\frac{\sqrt{2}}{4}&\frac{11}{60}+\frac{43\sqrt{2}}{240}&\frac{29}{60}+\frac{53\sqrt{2}}{240}&-\frac{1}{6}-\frac{3\sqrt{2}}{20}\\1&\frac{11}{15}-\frac{7\sqrt{2}}{30}&\frac{11}{15}+\frac{7\sqrt{2}}{30}&-\frac{7}{15}\\\hline &\frac{2}{3}-\frac{\sqrt{2}}{6}&\frac{2}{3}+\frac{\sqrt{2}}{6}&-\frac{1}{3}\end{array}.$$

$s=3$ (Chebyshev-Guass-Lobatto grid points: $c_1=1/4$, $c_2=3/4$, $c_3=1$)

$$\widetilde{\mathbf{G}}_0=\begin{pmatrix}-\frac{3}{4}\\-\frac{3}{4}\\-12\end{pmatrix},\quad \widetilde{\mathbf{G}}=\begin{pmatrix}-\frac{11}{6}&\frac{9}{2}&-\frac{23}{12}\\\frac{5}{6}&-\frac{7}{2}&\frac{41}{12}\\\frac{74}{3}&-30&\frac{52}{3}\end{pmatrix},$$

$$\begin{array}{c|c}\mathbf{c}&\widetilde{\mathbf{A}}\\\hline &\mathbf{b}^T\end{array}=\begin{array}{c|ccc}\frac{1}{4}&\frac{251}{720}&-\frac{41}{240}&-\frac{13}{180}\\\frac{3}{4}&\frac{419}{720}&\frac{31}{240}&\frac{7}{180}\\1&\frac{23}{45}&\frac{7}{15}&\frac{1}{45}\\\hline &\frac{5}{9}&\frac{1}{3}&\frac{1}{9}\end{array}.$$

Acknowledgments

The authors gratefully acknowledge the support from the National Natural Science Foundation of China (NSFC) through its grant 51377098.

References

[1] R. BELLMAN AND J. CASTI, *Differential quadrature and long term integration,* J. Math. Anal. Appl., 34(2) (1971), pp. 235–238.

[2] R. BELLMAN, B. G. KASHEF AND J. CASTI, *Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations,* J. Comput. Phys., 10 (1972), pp. 40–52.

[3] M. MALIK AND F. CIVAN, *Comparative study of differential quadrature and cubature methods visàvis some conventional techniques in context of convection-diffusion-reaction problems,* Chem. Eng. Sci., 50(3) (1995), pp. 531–547.

[4] C. W. BERT AND M. MALIK, *Differential quadrature method in computational mechanics: a review,* Appl. Mech. Rev., 49(1) (1996), pp. 1–28.

[5] F. CIVAN AND C. M. SLIEPCEVICH, *Application of differential quadrature to transport process,* J. Math. Anal. Appl., 93 (1983), pp. 206–221.

[6] S. K. JANG, C. W. BERT AND A. G. STRIZ, *Application of differential quadrature to static analysis of structural components,* Int. J. Numer. Methods Eng., 28(3) (1989), pp. 561–577.

[7] C. W. BERT, X. WANG AND A. G. STRIZ, *Differential quadrature for static and free vibration analyses of anisotropic plates*, Int. J. Solids Structures, 30(13) (1993), pp. 1737–1744.

[8] H. DU, M. K. LIM AND R. M. LIN, *Application of generalized differential quadrature method to structural problems,* Int. J. Numer. Methods Eng., 37(11) (1994), pp. 1881–1896.

[9] Q. W. XU, Z. F. LI AND J. WANG, *Modeling of transmission lines by the differential quadrature method*, IEEE Microwave Guided Wave Letters, 9(4) (1999), pp. 145–147.

[10] Q. W. XU, *Equivalent-circuit interconnects modeling based on the fifth-order differential quadrature methods,* IEEE Transactions Very Large Scale Integration (VLSI) Systems, 11(3) (2003), pp. 1068–1079.

[11] W. CHEN, Differential Quadrature Method and Its Applications in Engineering-Applying Special Matrix Product in Nonlinear Computation (in English), PhD thesis, Shanghai Jiao Tong University, 1997.

[12] M. TANAKA AND W. CHEN, *Coupling dual reciprocity BEM and differential quadrature method for time-dependent diffusion problems,* Appl. Math. Model., 25(3) (2001), pp. 257–268.

[13] M. TANAKA AND W. CHEN, *Dual reciprocity BEM applied to transient elastodynamic problems with differential quadrature method in time,* Computer Methods Appl. Mech. Eng., 190(18-19) (2001), pp. 2331–2347.

[14] T. C. FUNG, *Solving initial value problems by differential quadrature method, Part 1: first order equations,* Int. J. Numer. Methods Eng., 50(6) (2001), pp. 1411–1427.

[15] T. C. FUNG, *Solving initial value problems by differential quadrature method, Part 2: second and higher order equations,* Int. J. Numer. Methods Eng., 50(6) (2001), pp. 1429–1454.

[16] T. C. FUNG, *Stability and accuracy of differential quadrature method in solving dynamic problems,* Computer Methods Appl. Mech. Eng., 191 (2002), pp. 1311–1331.

[17] T. C. FUNG, *On the equivalence of the time domain differential quadrature method and the dissipative Runge-Kutta collocation method,* Int. J. Numer. Methods Eng., 53(2) (2002), pp. 409–431.

[18] E. HAIRER, S. P. NØRSETT AND G. WANNER, Solving Ordinary Differential Equations I: Second Revised Edition, Springer, Berlin, 1992.

[19] E. HAIRER AND G. WANNER, Solving Ordinary Differential Equations II: Second Revised Edition, Springer, Berlin, 1996.

[20] J. C. BUTCHER, Numerical Methods for Ordinary Differential Equations, Second Edition, Wiley, New York, 2008.

[21] C. SHU, Differential Quadrature and Its Application in Engineering, Springer, Berlin, 2000.

[22] F. Z. WANG, Numerical Method for Transient Stability Computation of Large-Scale Power System, Science Press, Beijing, 2013.

[23] W. Q. WANG AND S. F. LI, *The necessary and sufficient conditions of A-acceptability of high-order rational approximations to the function* $\exp(z)$, Natural Science Journal of Xiangtan University, 22(1) (2000), pp. 4–7.

[24] K. FENG, *Difference schemes for Hamiltonian formalism and symplectic geometry,* J. Comput. Math., 4(3) (1986), pp. 279–289.

[25] K. FENG AND M. Z. QIN, *Hamiltonian algorithms for Hamiltonian dynamical systems,* Progress Natural Sci., 1(2) (1991), pp. 105–116.

[26] K. FENG AND M. Z. QIN, Symplectic Geometric Algorithm for Hamiltonian Systems, Zhejiang Science & Technology Press, Hangzhou, 2003.