Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 657494, 9 pages
http://dx.doi.org/10.1155/2015/657494

Research Article

Characteristics of the Differential Quadrature Method and Its Improvement

Wang Fangzong, Liao Xiaobing, and Xie Xiong

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

Correspondence should be addressed to Wang Fangzong; fzwang@ctgu.edu.cn

Received 8 September 2014; Revised 23 December 2014; Accepted 25 December 2014

Academic Editor: Ming-Hung Hsu

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 have been proposed by using undetermined coefficients method and Padé approximations. The numerical results show that the improved differential quadrature method is more precise than the traditional differential quadrature method.

1. Introduction

The differential quadrature method (DQM) was first proposed by Bellman and his associates in the early 1970s [1, 2], which is usually 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.

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 structural mechanics [5–8], transport process [9], dynamic systems [10–12], and calculation of transmission line transient response [13, 14]. A comprehensive review of the chronological development of the differential quadrature method can be found in [4]. Although 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 them has been done in this area for the differential quadrature method. According to Fung [15], using Lagrange interpolation functions as test functions, the differential quadrature in time domain was shown to be equivalent to the recast implicit Runge-Kutta method [16–18]; 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 postprocessing (i.e., numerical solution at the end of grid points adopts polynomial extrapolation).

In this paper, using general polynomial as test functions [19], the weighting coefficients matrix of differential quadrature method is proved to satisfy **V**-transformation [17, 20]. 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 [17, 18], the basic characteristics of the differential quadrature method can be systematically

analysed. Unfortunately, the differential quadrature method is only a method of s-stage s-order and A-stable. Consequently, the differential quadrature method cannot 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 [17, 18]. Therefore, a class of improved differential quadrature method of s-stage $2s$-order is derived.

The paper 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 the differential quadrature method and the Runge-Kutta methods is deduced. In Section 4, the stability and accuracy characteristics of the 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 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,

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

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

In order to compute the weighting coefficients g_{ij} in (1), the test functions can be chosen as

$$r_k(x) = x^k, \quad (k = 0, 1, \ldots, s). \tag{2}$$

Substituting (2) into (1) gives

$$k = 0, \quad 0 = \sum_{j=0}^{s} g_{ij}, \ i \in (1, s), \tag{3}$$

$$\sum_{j=0}^{s} g_{ij} c_j^k = k \cdot c_i^{k-1}, \quad i, k \in (1, s). \tag{4}$$

Equation (3) can be expanded into matrix form as

$$\begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1s} \\ g_{21} & g_{22} & \cdots & g_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ g_{s1} & g_{s2} & \cdots & g_{ss} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} g_{10} \\ g_{20} \\ \vdots \\ g_{s0} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{5}$$

Let

$$\mathbf{G}_0 = \begin{pmatrix} g_{10} \\ g_{20} \\ \vdots \\ g_{s0} \end{pmatrix}, \qquad \mathbf{G} = \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1s} \\ g_{21} & g_{22} & \cdots & g_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ g_{s1} & g_{s2} & \cdots & g_{ss} \end{pmatrix}, \qquad \mathbf{e} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}. \tag{6}$$

Using (6), (5) can be simplified to

$$\mathbf{G}_0 \equiv -\mathbf{G}\mathbf{e}. \tag{7}$$

From $k = 1, 2, \ldots, s$, (4) can be expanded as

$$\begin{aligned} g_{i0}c_0 + g_{i1}c_1 + g_{i2}c_2 + \cdots + g_{is}c_s &= 1 \\ g_{i0}c_0^2 + g_{i1}c_1^2 + g_{i2}c_2^2 + \cdots + g_{is}c_s^2 &= 2c_i \\ &\vdots \\ g_{i0}c_0^s + g_{i1}c_1^s + g_{i2}c_2^s + \cdots + g_{is}c_s^s &= sc_i^{s-1}. \end{aligned} \tag{8}$$

Since initial grid point c_0 is usually defined as 0, (8) reduces to

$$\begin{aligned} g_{i1}c_1 + g_{i2}c_2 + \cdots + g_{is}c_s &= 1 \\ g_{i1}c_1^2 + g_{i2}c_2^2 + \cdots + g_{is}c_s^2 &= 2c_i \\ &\vdots \\ g_{i1}c_1^s + g_{i2}c_2^s + \cdots + g_{is}c_s^s &= sc_i^{s-1}. \end{aligned} \tag{9}$$

From $i = 1, 2, \ldots, s$, (9) can be also expanded into matrix form as

$$\begin{aligned} \mathbf{G}\begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & & c_s^s \end{pmatrix} &= \begin{pmatrix} 1 & 2c_1 & \cdots & sc_1^{s-1} \\ 1 & 2c_2 & \cdots & sc_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 2c_s & \cdots & sc_s^{s-1} \end{pmatrix} \\ &= \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix} \\ &\quad \cdot \begin{pmatrix} 1 & & & \\ & 2 & & \\ & & \ddots & \\ & & & s \end{pmatrix}. \end{aligned} \tag{10}$$

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

$$\mathbf{V} = \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix}. \tag{11}$$

Making use of (11), (10) can be expressed as

$$\mathbf{G}^{-1}\mathbf{V} = \begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & \cdots & c_s^s \end{pmatrix} \begin{pmatrix} 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{pmatrix}$$
$$= \left(\begin{array}{c|ccc} 1 & c_1 & c_1^2 & \cdots & c_1^s \\ 1 & c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & c_s^2 & \cdots & c_s^s \end{array}\right) \left(\begin{array}{cccc} 0 & 0 & \cdots & 0 \\ \hline 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{array}\right). \tag{12}$$

Finally, it can be inferred that

$$\mathbf{V}^{-1}\mathbf{G}^{-1}\mathbf{V} = \mathbf{A}_s, \tag{13}$$

where $\mathbf{A}_s$ is

$$\mathbf{A}_s = \begin{pmatrix} 0 & 0 & 0 & \cdots & \alpha_1 \\ 1 & 0 & 0 & \cdots & \alpha_2 \\ 0 & \frac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s-1} & \alpha_s \end{pmatrix}, \tag{14}$$

with

$$\boldsymbol{\alpha}_s = [\alpha_1, \alpha_2, \ldots, \alpha_s]^{\mathrm{T}} = \frac{1}{s}\mathbf{V}^{-1}\mathbf{c}^s,$$
$$\mathbf{c}^s = [c_1^s, c_2^s, \ldots, c_s^s]^{\mathrm{T}}. \tag{15}$$

Equation (13), that is, $\mathbf{G} = \mathbf{V}\mathbf{A}_s^{-1}\mathbf{V}^{-1}$, is called the implicit expression of the weighting coefficients matrix of the differential quadrature method 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 the test functions and distribution of grid points but are independent of some specific problems. There are four typical grid points' distributions: Legendre grid points, Chebyshev grid points, Chebyshev-Gauss-Lobatto grid points, and Uniform grid points (also called equally spaced grid points) [18]. This paper will focus on the latter three kinds of commonly used grid points, which are defined as follows:

(1) Chebyshev grid points:

$$c_k = \frac{1}{2}\left(1 - \cos\left(\frac{2k-1}{2s-2}\pi\right)\right), \quad k \in (1, s-1),$$
$$c_0 = 0, \qquad c_s = 1; \tag{16}$$

(2) Chebyshev-Gauss-Lobatto grid points:

$$c_k = \frac{1}{2}\left(1 - \cos\left(\frac{k}{s}\pi\right)\right), \quad k \in (0, s); \tag{17}$$

(3) Uniform grid points:

$$c_k = \frac{k}{s}, \quad k \in (0, s). \tag{18}$$

3. The Equivalence of Differential Quadrature Method and Runge-Kutta Method

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

$$\frac{d\mathbf{x}}{dt} = f(t, \mathbf{x}), \quad 0 < t \le \mathrm{T}$$
$$\mathbf{x}(t = 0) = \mathbf{x}_0. \tag{19}$$

In the following, t_n, t_{n+1} represent, respectively, the beginning and the end points at each step. $h = t_{n+1} - t_n$ will be used to denote the step size. The time interval $[t_n, t_{n+1}]$ will be normalized. That is, $c = (t - t_n)/h$, $t \in [t_n, t_{n+1}]$, $c \in [0, 1]$. At the same time, (19) can be made in the standard normalized form

$$\frac{d}{dc}\mathbf{x} = h\mathbf{f}(t_n + ch, \tilde{\mathbf{x}}), \quad \tilde{\mathbf{x}} = \mathbf{x}(t_n + ch); \tag{20}$$

then, using s-stage differential quadrature method to solve (20) yields

$$\mathbf{G}\begin{pmatrix} \tilde{\mathbf{x}}_1 \\ \vdots \\ \tilde{\mathbf{x}}_s \end{pmatrix} + \mathbf{G}_0\mathbf{x}_n = h\begin{pmatrix} f(t_n + c_1 h, \tilde{\mathbf{x}}_1) \\ \vdots \\ f(t_n + c_s h, \tilde{\mathbf{x}}_s) \end{pmatrix}, \tag{21}$$

where $\tilde{\mathbf{x}}_i = \mathbf{x}(t_n + c_i h)$, $i \in (1, s)$. Since $\mathbf{G}_0 \equiv -\mathbf{G}\mathbf{e}$, (21) reduces to

$$\begin{pmatrix} \tilde{\mathbf{x}}_1 \\ \vdots \\ \tilde{\mathbf{x}}_s \end{pmatrix} = \mathbf{x}_n \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} + h\mathbf{G}^{-1}\begin{pmatrix} \mathbf{f}(t_n + c_1 h, \tilde{\mathbf{x}}_1) \\ \vdots \\ \mathbf{f}(t_n + c_s h, \tilde{\mathbf{x}}_s) \end{pmatrix}. \tag{22}$$

Let

$$\mathbf{G}^{-1} = \mathbf{A} = [a_{ij}], \quad i, j \in (1, s). \tag{23}$$

Clearly, making use of (13) and (23) leads to

$$\mathbf{A} = \mathbf{V}\mathbf{A}_s\mathbf{V}^{-1}. \tag{24}$$

Therefore, the weighting coefficients matrix $\mathbf{A}$ also satisfies $\mathbf{V}$-transformation. It can be inferred from (22) that

$$\tilde{\mathbf{x}}_i = \mathbf{x}_n + h\sum_{j=1}^{s} a_{ij}\mathbf{f}(t_n + c_j h, \tilde{\mathbf{x}}_j), \quad i \in (1, s). \tag{25}$$

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

$$\begin{aligned}\tilde{\mathbf{x}}_s = \mathbf{x}_{n+1} &= \mathbf{x}_n + h\sum_{j=1}^{s} a_{sj}\mathbf{f}\left(t_n + c_j h, \tilde{\mathbf{x}}_j\right)\\ &= \mathbf{x}_n + h\sum_{j=1}^{s} b_j\mathbf{f}\left(t_n + c_j h, \tilde{\mathbf{x}}_j\right),\end{aligned} \tag{26}$$

where $b_j = a_{sj}$, $j \in (1, s)$. It can be seen that (25) and (26) are the standard forms for an s-stage Runge-Kutta method. Since $\tilde{\mathbf{x}}_s = \mathbf{x}_{n+1}$, the equivalent Runge-Kutta method is a reducible method [20]. In fact, the traditional differential quadrature method generally does not involve postprocessing, 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 [18] as

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}, \tag{27}$$

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

4. Analysis of the Basic Characteristics of the Differential Quadrature Method

The stability and accuracy characteristics of the equivalent Runge-Kutta method will be investigated next. From (13) and (23), it can be inferred that

$$\begin{aligned}&\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{ss} \end{pmatrix}\begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ 1 & c_2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix}\\ &= \begin{pmatrix} c_1 & c_1^2 & \cdots & c_1^s \\ c_2 & c_2^2 & \cdots & c_2^s \\ \vdots & \vdots & \ddots & \vdots \\ c_s & c_s^2 & & c_s^s \end{pmatrix}\begin{pmatrix} 1 & & & \\ & \frac{1}{2} & & \\ & & \ddots & \\ & & & \frac{1}{s} \end{pmatrix}.\end{aligned} \tag{28}$$

Equation (28) reduces to

$$\sum_{j=1}^{s} a_{ij}c_j^{k-1} = \frac{c_i^k}{k}, \quad i \in (1, s), \ k \in (1, s). \tag{29}$$

On the other hand, since $b_j = a_{sj}$, $j \in (1, s)$ and $c_s = 1$, from (28), it can be inferred that

$$\mathbf{b}^{\mathrm{T}}\mathbf{V} = \left[1, \frac{1}{2}, \ldots, \frac{1}{s}\right]. \tag{30}$$

Similarly, (30) reduces to

$$\sum_{i=1}^{s} b_i c_i^{k-1} = \frac{1}{k}, \quad k \in (1, s). \tag{31}$$

Obviously, it has been shown that the equivalent Runge-Kutta method at least satisfies simplifying assumptions $C(s)$ and $B(s)$ from (29) and (31). Furthermore, it can be verified that the equivalent Runge-Kutta method only satisfies simplifying assumptions $D(0)$. From Theorem 5.1 on page 71 in [17], it can be concluded that the implicit Runge-Kutta method or the corresponding differential quadrature method is s-stage s-order.

The stability function $R(z)$ of the equivalent Runge-Kutta method or the corresponding differential quadrature method is given by the formula [16–18]

$$R(z) = \frac{\det\left(\mathbf{I} + z\left(\mathbf{e}\mathbf{b}^{\mathrm{T}} - \mathbf{A}\right)\right)}{\det(\mathbf{I} - z\mathbf{A})}, \tag{32}$$

where, as usual, $\mathbf{I}$ is the identity matrix of dimension s. Due to grid points' asymmetric distribution, the equivalent implicit Runge-Kutta method is not a symmetric method. As a result, there is a unique adjoint method (also called reflected method) [16, 18], which is defined as $\begin{array}{c|c} \mathbf{c}^* & \mathbf{A}^* \\ \hline & (\mathbf{b}^*)^{\mathrm{T}} \end{array}$, satisfying

$$\begin{aligned}\mathbf{c}^* &= \mathbf{e} - \mathbf{P}\mathbf{c},\\ \mathbf{P}\mathbf{A}^*\mathbf{P} &= \mathbf{e}\mathbf{b}^{\mathrm{T}} - \mathbf{A},\\ \mathbf{b}^* &= \mathbf{P}\mathbf{b},\end{aligned} \tag{33}$$

where $\mathbf{P}$ is

$$\mathbf{P} = \begin{pmatrix} & & & 1 \\ & & 1 & \\ & \iddots & & \\ 1 & & & \end{pmatrix} \in \mathbf{R}^{s\times s}. \tag{34}$$

Furthermore, from Theorem 343B on page 221 in [18], if the original method satisfies the simplifying assumptions $C(s)$ and $B(s)$, then the adjoint method also satisfies the same simplifying assumptions. Hence, the adjoint method enjoys $\mathbf{V}$-transformation:

$$\mathbf{A}^* = \mathbf{V}^*\mathbf{A}_s^*\left(\mathbf{V}^*\right)^{-1}, \qquad \mathbf{V}^* = \begin{pmatrix} 1 & c_1^* & \cdots & (c_1^*)^{s-1} \\ 1 & c_2^* & \cdots & (c_2^*)^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s^* & \cdots & (c_s^*)^{s-1} \end{pmatrix}, \tag{35}$$

where $\mathbf{A}_s^*$ is

$$\mathbf{A}_s^* = \begin{pmatrix} 0 & 0 & 0 & \cdots & \beta_1 \\ 1 & 0 & 0 & \cdots & \beta_2 \\ 0 & \frac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s-1} & \beta_s \end{pmatrix}, \tag{36}$$

with

$$\boldsymbol{\beta}_s = [\beta_1, \beta_2, \ldots, \beta_s]^{\mathrm{T}} = \frac{1}{s}\left(\mathbf{V}^*\right)^{-1}\left(\mathbf{c}^*\right)^s,$$
$$\left(\mathbf{c}^*\right)^s = \left[\left(c_1^*\right)^s, \left(c_2^*\right)^s, \ldots, \left(c_s^*\right)^s\right]^{\mathrm{T}}. \tag{37}$$

Equation (32) can be reduced to

$$\begin{aligned} R(z) &= \frac{\det\left(\mathbf{I} + z\left(\mathbf{P}\mathbf{A}^*\mathbf{P}^{-1}\right)\right)}{\det\left(\mathbf{I} - z\left(\mathbf{V}\mathbf{A}_s\mathbf{V}^{-1}\right)\right)} = \frac{\det\left(\mathbf{I} + z\mathbf{A}^*\right)}{\det\left(\mathbf{I} - z\mathbf{A}_s\right)} \\ &= \frac{\det\left(\mathbf{I} + z\left(\mathbf{V}^*\mathbf{A}_s\left(\mathbf{V}^*\right)^{-1}\right)\right)}{\det\left(\mathbf{I} - z\mathbf{A}_s\right)} = \frac{\det\left(\mathbf{I} + z\mathbf{A}_s^*\right)}{\det\left(\mathbf{I} - z\mathbf{A}_s\right)}. \end{aligned} \tag{38}$$

Because $\mathbf{A}_s$ and $\mathbf{A}_s^*$ are a class of special matrices, (38) can be evaluated as

$$\begin{aligned} R(z) &= \frac{\det\left(\mathbf{I} + z\mathbf{A}_s^*\right)}{\det\left(\mathbf{I} - z\mathbf{A}_s\right)} \\ &= \frac{1 - \sum_{k=s}^{1} \beta_k \left((k-1)!/(s-1)!\right)(-z)^{s-k+1}}{1 - \sum_{k=s}^{1} \alpha_k \left((k-1)!/(s-1)!\right) z^{s-k+1}}, \end{aligned} \tag{39}$$

since $c_s = 1$, resulting in $c_1^* = 0$, it implies $\beta_1 = 0$. It can be verified that the stability function of equivalent Runge-Kutta method is A-acceptability of p-order ($p \geq s-1$) rational approximation to exponential function [21]. 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

$$\mathbf{G}_0 = \begin{pmatrix} -1 \\ \frac{1}{2} \\ -1 \end{pmatrix}, \qquad \mathbf{G} = \begin{pmatrix} -\frac{3}{2} & 3 & -\frac{1}{2} \\ -3 & \frac{3}{2} & 1 \\ \frac{9}{2} & -9 & \frac{11}{2} \end{pmatrix}, \tag{40}$$

and the Butcher tableau of equivalent Runge-Kutta method is

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} \frac{1}{3} & \frac{23}{36} & -\frac{4}{9} & \frac{5}{36} \\ \frac{2}{3} & \frac{7}{9} & -\frac{2}{9} & \frac{1}{9} \\ 1 & \frac{3}{4} & 0 & \frac{1}{4} \\ \hline & \frac{3}{4} & 0 & \frac{1}{4} \end{array}. \tag{41}$$

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

$$\mathbf{A} = \begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & \frac{2}{27} \\ 1 & 0 & -\frac{11}{27} \\ 0 & \frac{1}{2} & \frac{2}{3} \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}^{-1}, \tag{42}$$

the stability function of the equivalent Runge-Kutta method is

$$R(z) = \frac{1 + (1/3)z + (1/27)z^2}{1 - (2/3)z + (11/54)z^2 - (1/27)z^3}, \tag{43}$$

and the Butcher tableau of the adjoint method is

$$\begin{array}{c|c} \mathbf{c}^* & \mathbf{A}^* \\ \hline & \left(\mathbf{b}^*\right)^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{3} & -\frac{1}{36} & \frac{2}{9} & \frac{5}{36} \\ \frac{2}{3} & \frac{1}{9} & \frac{4}{9} & \frac{1}{4} \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}. \tag{44}$$

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

$$\mathbf{A}^* = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & -\frac{2}{27} \\ 0 & \frac{1}{2} & \frac{1}{3} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix}^{-1}. \tag{45}$$

5. Improved Differential Quadrature Method

Based on the above deduction, the traditional differential quadrature method is a method of s-stage s-order. Compared with the multistage high-order Runge-Kutta method, for example, Gauss method (s-stage $2s$-order method), it has the disadvantage of lower precision. As it is well-known that if the stability function of a numerical method is diagonal Padé approximations to the exponential function, then this method is the method of A-stable and s-stage $2s$-order [22]. Inspired by this idea, the stability function of new Runge-Kutta method or new differential quadrature method have been converted into diagonal Padé approximation to the exponential function by using undetermined coefficients method. From (38), it can be seen that the stability function of the equivalent Runge-Kutta method will be determined by $\mathbf{A}_s$ and $\mathbf{A}_s^*$. Suppose $\mathbf{A}_s = \mathbf{A}_s^* = \widetilde{\mathbf{A}}_s$, without changing $\mathbf{b}$ 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

$$\widetilde{\mathbf{A}} = \left(\widetilde{\mathbf{G}}\right)^{-1} = \mathbf{V}\begin{pmatrix} 0 & 0 & 0 & \cdots & \gamma_1 \\ 1 & 0 & 0 & \cdots & \gamma_2 \\ 0 & \frac{1}{2} & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{s-1} & \gamma_s \end{pmatrix}\mathbf{V}^{-1} = \mathbf{V}\widetilde{\mathbf{A}}_s\mathbf{V}^{-1}. \tag{46}$$

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

$$\begin{aligned} R(z) &= \frac{\det\left(\mathbf{I} + z\widetilde{\mathbf{A}}_s\right)}{\det\left(\mathbf{I} - z\widetilde{\mathbf{A}}_s\right)} \\ &= \frac{1 - \sum_{k=s}^{1} \gamma_k \left((k-1)!/(s-1)!\right)(-z)^{s-k+1}}{1 - \sum_{k=s}^{1} \gamma_k \left((k-1)!/(s-1)!\right) z^{s-k+1}}. \end{aligned} \tag{47}$$

From (46) and (47), it can be inferred that the last column elements in $\widetilde{\mathbf{A}}_s$ determine the stability function. To improve the order of new Runge-Kutta method, undetermined coefficients $\boldsymbol{\gamma}$ can be selected so that the stability function of new Runge-Kutta method becomes the diagonal Padé approximations to the exponential function (defined by $e^z|_s^s$):

$$R(z) = \frac{1 - \sum_{k=s}^{1} \gamma_k \left((k-1)!/(s-1)!\right)(-z)^{s-k+1}}{1 - \sum_{k=s}^{1} \gamma_k \left((k-1)!/(s-1)!\right) z^{s-k+1}} = e^z|_s^s. \tag{48}$$

By comparing the coefficients on both sides of (48), undetermined coefficients $\boldsymbol{\gamma}$ can be conveniently obtained as

$$\begin{aligned} s = 2, \quad & \boldsymbol{\gamma}_2 = \left[-\frac{1}{12}, \frac{1}{2}\right], \\ s = 3, \quad & \boldsymbol{\gamma}_3 = \left[\frac{1}{60}, -\frac{1}{5}, \frac{1}{2}\right], \\ s = 4, \quad & \boldsymbol{\gamma}_4 = \left[-\frac{1}{280}, \frac{1}{14}, -\frac{9}{28}, \frac{1}{2}\right], \\ & \vdots \end{aligned} \tag{49}$$

After getting $\boldsymbol{\gamma}_s = (\gamma_1, \ldots, \gamma_s)^{\mathrm{T}}$, coefficients matrix $\widetilde{\mathbf{A}}$ or $\widetilde{\mathbf{G}}$ can also be easily computed through using (46). 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.

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 $\mathbf{G}_0$ and $\mathbf{G}$ are given by

$$\widetilde{\mathbf{G}}_0 = \begin{pmatrix} \frac{4}{3} \\ \frac{4}{3} \\ -12 \end{pmatrix}, \qquad \widetilde{\mathbf{G}} = \begin{pmatrix} -\frac{17}{2} & 10 & -\frac{17}{6} \\ -\frac{11}{2} & 4 & \frac{1}{6} \\ \frac{75}{2} & -42 & \frac{33}{2} \end{pmatrix}, \tag{50}$$

and the Butcher tableau of new Runge-Kutta method is

$$\begin{array}{c|c} \mathbf{c} & \widetilde{\mathbf{A}} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} \frac{1}{3} & \frac{73}{120} & -\frac{23}{60} & \frac{13}{120} \\ \frac{2}{3} & \frac{97}{120} & -\frac{17}{60} & \frac{17}{120} \\ 1 & \frac{27}{40} & \frac{3}{20} & \frac{7}{40} \\ \hline & \frac{3}{4} & 0 & \frac{1}{4} \end{array}. \tag{51}$$

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

$$\widetilde{\mathbf{A}} = \begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & \frac{1}{60} \\ 1 & 0 & -\frac{1}{5} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \\ 1 & 1 & 1 \end{pmatrix}^{-1} \tag{52}$$

and the Butcher tableau of the adjoint method is

$$\begin{array}{c|c} \mathbf{c} & \widetilde{\mathbf{A}}^* \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} = \begin{array}{c|ccc} 0 & \frac{3}{40} & -\frac{3}{20} & \frac{3}{40} \\ \frac{1}{3} & \frac{13}{120} & \frac{17}{60} & -\frac{7}{120} \\ \frac{2}{3} & \frac{17}{120} & \frac{23}{60} & \frac{17}{120} \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}. \tag{53}$$

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

$$\widetilde{\mathbf{A}}^* = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix} \begin{pmatrix} 0 & 0 & \frac{1}{60} \\ 1 & 0 & -\frac{1}{5} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{3} & \frac{1}{9} \\ 1 & \frac{2}{3} & \frac{4}{9} \end{pmatrix}^{-1}. \tag{54}$$

6. Numerical Examples

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

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{Bmatrix} + \begin{pmatrix} 6 & -2 \\ -2 & 4 \end{pmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 10 \end{Bmatrix}, \tag{55}$$

with initial condition

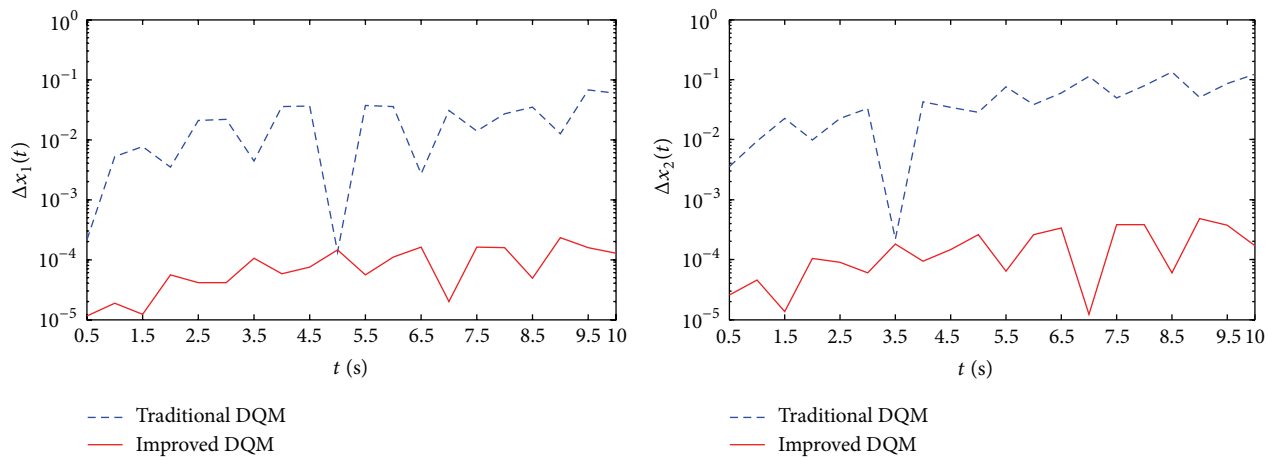
$$\begin{aligned} x_1 &= 0, \qquad \dot{x}_1 = 0 \\ x_2 &= 0, \qquad \dot{x}_2 = 0. \end{aligned} \tag{56}$$

The exact solution of this problem is

$$\begin{aligned} x_1 &= 1 - \frac{5}{3}\cos\left(\sqrt{2}t\right) + \frac{2}{3}\cos\left(\sqrt{5}t\right), \\ x_2 &= 3 - \frac{5}{3}\cos\left(\sqrt{2}t\right) - \frac{4}{3}\cos\left(\sqrt{5}t\right). \end{aligned} \tag{57}$$

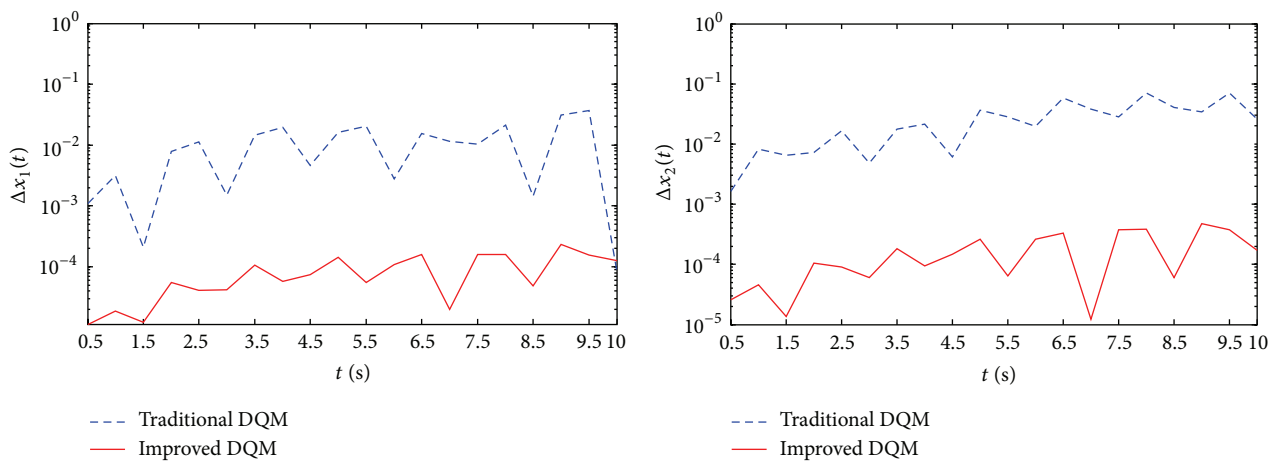
The differential quadrature method can be used to find the numerical solutions to transient response directly. The detailed calculation steps of solving second-order differential equations in [11] can be seen. Figures 1, 2, and 3 show the displacement error trajectories comparison of traditional differential quadrature method and improved differential quadrature method with the same step size $h = 0.5s$. In these Figures, the exact solution at each step is used for comparison. From Figures 1–3, it is evident that improved differential quadrature method is two orders of magnitude higher than traditional differential quadrature method. The error of improved differential quadrature method ranges between 10^{-5} and 10^{-4}, even with a large step size $h = 0.5s$.

In order to compare the calculation precision of the traditional differential quadrature method and the improved differential quadrature method better, dynamic equations (55) have been computed for different time steps. Table 1 shows the computational results at $t = 60$ s. Comparing with



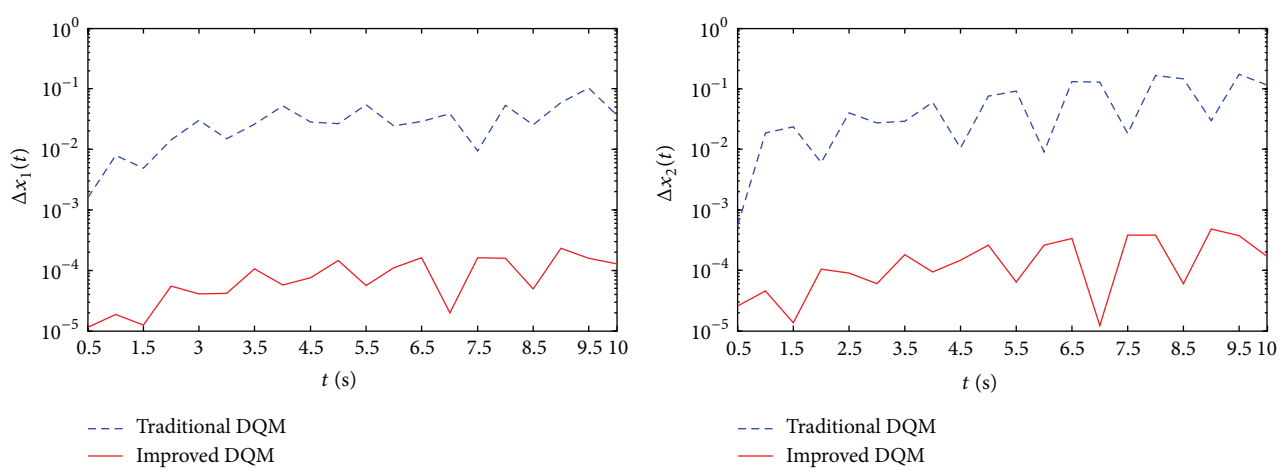
(a) Error trajectories of $x_1(t)$ (b) Error trajectories of $x_2(t)$

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



(a) Error trajectories of $x_1(t)$ (b) Error trajectories of $x_2(t)$

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



(a) Error trajectories of $x_1(t)$ (b) Error trajectories of $x_2(t)$

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

Table 1: Computational results of the displacement ($t = 60$ s).

$t = 60$ s		$x_1(t)$	$x_2(t)$
Exact solution		2.264386552	5.469005157
$s = 5$, $h = 1s$	Traditional DQM-Chebyshev	2.253975383	5.491424542
	Improved DQM-Chebyshev	2.264406270	5.468965760
	Traditional DQM-Chebyshev-Gauss-Lobatto	2.276613635	5.439926064
	Improved DQM-Chebyshev-Gauss-Lobatto	2.264406270	5.468965760
	Traditional DQM-Uniform	2.342513616	5.278336836
	Improved DQM-Uniform	2.264406270	5.468965760
$s = 10$, $h = 2s$	Traditional DQM-Chebyshev	2.264383332	5.469011768
	Improved DQM-Chebyshev	2.264386552	5.469005157
	Traditional DQM-Chebyshev-Gauss-Lobatto	2.264399479	5.468978613
	Improved DQM-Chebyshev-Gauss-Lobatto	2.264386552	5.469005157
	Traditional DQM-Uniform	2.265087814	5.467565205
	Improved DQM-Uniform	2.264386561	5.469005173

the exact solution, the numerical results using traditional differential quadrature method are the same until the first decimal place when $s = 5$, $h = 1s$, while the numerical results using improved differential quadrature method are the same until the third decimal place. When $s = 10$, $h = 2s$, the numerical results using traditional differential quadrature method are the same until the fifth decimal place, while the numerical results using improved differential quadrature method are the same until the seventh decimal place (especially using Chebyshev grid points and Chebyshev-Gauss-Lobatto grid points, the numerical results are almost the exact solution).

7. Conclusion

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

(1) Based on general polynomial as test functions, the weighting coefficients matrix of the differential quadrature method satisfies the $\mathbf{V}$-transformation. It 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 converted into equivalent Runge-Kutta method of A-stable and s-stage s-order. Therefore, compared with the commonly used single-stage low-order numerical integral methods, even with a small number of grid points, the traditional differential quadrature method can also give more accurate solutions. This is the main mechanism that differential quadrature method has been successfully applied in many fields.

(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. Therefore, 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.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

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," *Journal of Mathematical Analysis and Applications*, vol. 34, no. 2, pp. 235–238, 1971.

[2] R. Bellman, B. G. Kashef, and J. Casti, "Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations," *Journal of Computational Physics*, vol. 10, pp. 40–52, 1972.

[3] M. Malik and F. Civan, "A comparative study of differential quadrature and cubature methods *vis-à-vis* some conventional techniques in context of convection-diffusion-reaction problems," *Chemical Engineering Science*, vol. 50, no. 3, pp. 531–547, 1995.

[4] C. W. Bert and M. Malik, "Differential quadrature method in computational mechanics: a review," *Applied Mechanics Reviews*, vol. 49, no. 1, pp. 1–28, 1996.

[5] S. K. Jang, C. W. Bert, and A. G. Striz, "Application of differential quadrature to static analysis of structural components," *International Journal for Numerical Methods in Engineering*, vol. 28, no. 3, pp. 561–577, 1989.

[6] C. W. Bert, W. Xinwei, and A. G. Striz, "Differential quadrature for static and free vibration analyses of anisotropic plates," *International Journal of Solids and Structures*, vol. 30, no. 13, pp. 1737–1744, 1993.

[7] F. Civan, "Solving multivariable mathematical models by the quadrature and cubature methods," *Numerical Methods for Partial Differential Equations*, vol. 10, no. 5, pp. 545–567, 1994.

[8] H. Du, M. K. Lim, and R. M. Lin, "Application of generalized differential quadrature method to structural problems," *International Journal for Numerical Methods in Engineering*, vol. 37, no. 11, pp. 1881–1896, 1994.

[9] F. Civan and C. M. Sliepcevich, "Application of differential quadrature to transport processes," *Journal of Mathematical Analysis and Applications*, vol. 93, no. 1, pp. 206–221, 1983.

[10] T. C. Fung, "Solving initial value problems by differential quadrature method. I. First-order equations," *International Journal for Numerical Methods in Engineering*, vol. 50, no. 6, pp. 1411–1427, 2001.

[11] T. C. Fung, "Solving initial value problems by differential quadrature method, part 2: second and higher order equations," *International Journal for Numerical Methods in Engineering*, vol. 50, no. 6, pp. 1429–1454, 2001.

[12] T. C. Fung, "Stability and accuracy of differential quadrature method in solving dynamic problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 13-14, pp. 1311–1331, 2002.

[13] Q.-W. Xu, Z.-F. Li, J. Wang, and J.-F. Mao, "Modeling of transmission lines by the differential quadrature method," *IEEE Microwave and Wireless Components Letters*, vol. 9, no. 4, pp. 145–147, 1999.

[14] Q. Xu and P. Mazumder, "Equivalent-circuit interconnect modeling based on the fifth-order differential quadrature methods," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 6, pp. 1068–1079, 2003.

[15] T. C. Fung, "On the equivalence of the time domain differential quadrature method and the dissipative Runge-Kutta collocation method," *International Journal for Numerical Methods in Engineering*, vol. 53, no. 2, pp. 409–431, 2002.

[16] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I*, Springer, Berlin, Germany, 2nd edition, 1992.

[17] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II*, vol. 14 of *Springer Series in Computational Mathematics*, Springer, Berlin, Germany, 2nd edition, 1996.

[18] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2008.

[19] C. Shu, *Differential Quadrature and Its Application in Engineering*, Springer, Berlin, Germany, 2000.

[20] F. Z. Wang, *Numerical Method for Transient Stability Computation of Large-Scale Power System*, Science Press, Beijing, China, 2013.

[21] W. Q. Wang and S. F. Li, "Necessary and sufficient conditions for A-acceptability of high-order rational approximations to the function *exp(z)*," *Natural Science Journal of Xiangtan University*, vol. 22, no. 1, pp. 4–7, 2000.

[22] B. L. Ehle, "On Padé approximations to the exponential function and A-stable methods for the numerical solution of initial value problems," Research Report CSRR 2010, Department AACS, University of Waterloo, Waterloo, Canada, 1969.