

Article

A Time Finite Element Method Based on the Differential Quadrature Rule and Hamilton's Variational Principle

Yufeng Xing ^{1,*}, Mingbo Qin ¹ and Jing Guo ²

¹ Institute of Solid Mechanics, Beihang University, No. 37, Xueyuan Road, Haidian District, Beijing 100191, China; qmb337912314@163.com

² Science and Technology on Reliability and Environment Engineering Laboratory, Beijing Institute of Structure and Environment Engineering, No. 1, Dahongmen Road, Fengtai District, Beijing 100076, China; guojing2662632@126.com

* Correspondence: Xingyf@buaa.edu.cn; Tel.: +86-10-8233-9964

Academic Editor: Francesco Tornabene

Received: 22 November 2016; Accepted: 23 January 2017; Published: 4 February 2017

Abstract: An accurate and efficient Differential Quadrature Time Finite Element Method (DQTFEM) was proposed in this paper to solve structural dynamic ordinary differential equations. This DQTFEM was developed based on the differential quadrature rule, the Gauss–Lobatto quadrature rule, and the Hamilton variational principle. The proposed DQTFEM has significant benefits including the high accuracy of differential quadrature method and the generality of standard finite element formulation, and it is also a highly accurate symplectic method. Theoretical studies demonstrate the DQTFEM has higher-order accuracy, adequate stability, and symplectic characteristics. Moreover, the initial conditions in DQTFEM can be readily imposed by a method similar to the standard finite element method. Numerical comparisons for accuracy and efficiency among the explicit Runge–Kutta method, the Newmark method, and the proposed DQTFEM show that the results from DQTFEM, even with a small number of sampling points, agree better with the exact solutions and validate the theoretical conclusions.

Keywords: differential quadrature rule; time finite element method; Gauss–Lobatto quadrature rule

1. Introduction

Various algorithms are available for numerical integration of Ordinary Differential Equations (ODEs). The most commonly used algorithms, derived from Finite Difference Method (FDM), are the Direct Integration Methods (DIMs), such as the Central Difference Method (CDM), the Newmark method, and the Runge–Kutta Method (RKM). Feng presented structure-preserving symplectic schemes for the Hamilton system and illustrated the fundamental cause of energy dissipation of classical DIM [1].

As is widely accepted, Finite Element Method (FEM) is superior to FDM and has advantages in space, thus it should also exhibit advantages in time in the mathematical sense. However, Zienkiewicz doubted if it is possible to use Finite Elements (FEs) in the time domain [2]. Indeed, when the involved functions are smooth enough, it is not essential to define massive time steps to integrate ODEs, and the time steps should be as many as the FEs to produce a solution of comparable accuracy. In view of the increased storage required, the use of Time Finite Elements (TFEs) to solve ODEs is questionable. However, this problem can be gradually solved due to the rapid development of computer capacity and advanced solution methods.

In many cases, conventional step-by-step algorithms like CDM may call for a very large number of time steps. This is especially true when dealing with excitation and material properties changing

rapidly in time. Comparing with the Finite Difference (FD) based method, a variation-based method, with its inherent stability and physical origin, may lower the computational effort considerably. Thus, despite some reservations [2], the extension of FE concept to the time domain is logical.

Many researchers have attempted to employ Time Finite Elements Methods (TFEMs) to discretize ODEs. Similar to FEMs in spatial domain, one can formulate TFEM based on Hamilton's variational principles [3–5], Hamilton's law of varying action [6,7], Gurtin's variational principle [8], and weighted residual methods [9,10]. Almost all of the reported works dealt with one FE interval and repeated the calculations for subsequent time intervals with new initial conditions. Hence, these TFEMs lead to a step-by-step or recurrence calculation and are conditionally stable, unlike standard FEMs, which produce simultaneous solutions of the whole range of interest. The present work introduces a high-order method, and, generally, the dynamic responses of the whole time domain can be solved in one shot rather than step by step.

The commonly used trial functions in TFEMs based on variational principles are Hermite interpolations [3,4,6,11], rarely used linear functions [12], and cubic spline functions [13–15]. The advantage of using Hermite interpolations is that the initial conditions can be imposed in the same manner as the boundary conditions in standard FEMs, and the resulting time element is C^1 continuous [16,17]. Moreover, the solution accuracy increases as the number of elements increases or the order of trial functions increases. The present TFEM is also based on a variational principle, but usually the nodes or sampling points in an element are not uniformly located, and nodal velocities are only for two end nodes rather than for the interior nodes.

As an alternative to variational principles, weighted residual approaches are also widely used to construct TFEs. The commonly used trial functions in weighted residual approaches are Lagrange interpolations [18,19], linear/quadratic polynomials [20,21], Taylor series [22,23], and Taylor formula [9,24]. Zienkiewicz proposed a TFEM based on a weighted residual approach, and gave recurrence schemes of a single step or multiple steps [2]. Remarkably, as they reported, the mainly used DIMs can also be derived using weighted residual approaches. Later on, they employed Taylor's expansion to form a unified set of single-step algorithms [9], discussed the accuracy and the stability, and applied the method of backward error analysis to obtain a few general rules to choose parameters. By separately approximating acceleration, velocity and displacement as linear, quadratic, and cubic Taylor series, Hoff and Pahl developed a single-step TFEM based on a weighted residual method, and applied it to linear [25] and nonlinear [22] structural dynamics. Numerical results showed that the method presents some advantages, especially in accuracy and overshooting behavior. Additionally, Krishna and Manjeet presented three and four steps of TFEM using the least squares method as well as Lagrange interpolations for first-order initial-value problems [18]. The drawback of the weighted residual approaches based on TFEM is the determination of the optimum weighting functions.

TFEMs were based on the assumption that state variables are continuous in the time domain. However, in the Time-Discontinuous Galerkin Method (TDGM) which was originated by Reed and Hill [26] and Lesaint and Raviart [27] based on Galerkin weighted residual approach, and in the TDGM, the unknown variables are continuous within the time element but discontinuous at time nodes. The TDGM has been employed to solve periodic differential equations and Davey–Stewartson equations [28,29]. It was first shown by Lesaint and Raviart [27] and Delfour et al. [30] that TDGM leads to A-stable, higher-order accurate TFEMs. Subsequently, Hughes and Hulbert first applied TDGM to elastodynamics and structural dynamics to develop space–time finite element methods [31–34] and TFEMs [5,35], respectively. According to their results, TDGM stability is easily proved, but the convergence had been proved only for linear elements, therefore the Time-Discontinuous Galerkin/Least Squares Method (TDGLSM) [36] was developed to solve this problem [31,32,35]. Compared with the continuous Galerkin FEM, the drawback of TDGMs is the usage of a higher number of degrees, which has been overcome by designing an improved predictor/multi-corrector algorithm [37] or a suitable adaptive strategy for iterative mesh refinement [38,39].

According to the above review, the TFEMs in existence certainly have limitations. As is well known, researchers are always seeking new numerical methods with higher accuracy, efficiency, and flexibility to promote modeling capabilities. Bellman and Casti proposed the Differential Quadrature Method (DQM) as a simply and highly efficient numerical technique for initial-value problems [40], which was regarded as a global and higher-order FD scheme [17,41–43] and could yield accurate results using a considerably smaller number of sampling points.

The general versatility of DQM has been established in a variety of physical problems. A recent survey refers to publications [44,45], and an exhaustive list of the literature on DQM up to 1996 may be found in the survey paper [46]. Although DQM’s numerical accuracy and computational efficiency has been well demonstrated, its limitations are also significant [46]. In 2009, based on the Differential Quadrature (DQ) rule, the Gauss–Lobatto quadrature rule, and the total potential energy variational principle, a DQ Finite Element Method (DQFEM) was proposed by the author and co-workers for free vibration analysis of thin plates [47], and the authors formulated explicit one- to three-dimensional (1-D to 3-D) element matrices [48]. The DQFEM uses Lagrange trial functions for C^0 and C^1 FEs and possesses high accuracy of DQM and the generality of standard FEM.

In this context, similar to DQFEM for spatial domain [47,48], Hamilton’s variational principle together with the DQ rule and the Gauss–Lobatto quadrature rule are used to develop a TFEM in the present work, where the initial conditions can be readily imposed as in standard FEM. This novel method is called the DQ Time Finite Element Method (DQTFEM). The novelties of the present work compared with the existing TFEMs can be briefly summarized as follows:

- (1) The general form of Hamilton’s variational principle, the DQ rule, and the Gauss–Lobatto quadrature rule are used to formulate a TFEM for structural dynamic analysis.
- (2) The Gauss–Lobatto quadrature points are used as time sampling points or nodes of the time element, which results in more accurate results with the same number of nodes.
- (3) The nodal velocities are only involved at two end nodes of the time element, implying that the number of Degrees of Freedom (DOFs) is less than that of the existing TFEMs based on the variational principle and Hermite interpolations.
- (4) Lagrange interpolations are employed instead of Hermite interpolations in the construction of the C^1 time element, and the stability and accuracy of DQTFEM are studied in detail.

The outline of the present paper is as follows. In Section 2, the formulation of DQTFEM is presented for structural dynamic ODEs. In Section 3, the stability and accuracy are studied. Obtained solutions are compared with analytical solutions, the results of four-order explicit RKM, and the Newmark Average acceleration Method (NMA) in Section 4. Finally, conclusions are drawn in Section 5.

2. Basic Formulations

For structural dynamic systems with damping and subjected to external loads, the ODE in a discrete form can be derived by using the general form of Hamilton’s variational principle [49]:

$$\int_{t_1}^{t_2} \delta(T - V)dt + \int_{t_1}^{t_2} \delta W dt = 0 \tag{1}$$

$$T = \frac{1}{2} \dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}}, \quad V = \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} \tag{2}$$

$$\delta W = \dot{\mathbf{u}}^T \mathbf{C} \delta \mathbf{u} + \mathbf{f}^T \delta \mathbf{u}, \tag{3}$$

where T , V , and δW stand for the kinetic energy, strain energy, and virtual work of damping forces and external forces, respectively; $\mathbf{M} = [\mathbf{M}_{ij}]_{n \times n}$, $\mathbf{K} = [\mathbf{K}_{ij}]_{n \times n}$, and $\mathbf{C} = [\mathbf{C}_{ij}]_{n \times n}$ are the mass, stiffness, and damping symmetrical matrices, respectively; n is the number of spatial DOFs of the system; $\mathbf{u}(t)$

and $f(t)$ are the displacement vector and the prescribed load vector corresponding to spatial nodes, respectively; and a time derivative is denoted by a dot over the letter.

$$\mathbf{u}^T = [u_1 \quad u_2 \quad u_3 \quad \cdots \quad u_n], \mathbf{f}^T = [f_1 \quad f_2 \quad f_3 \quad \cdots \quad f_n] \tag{4}$$

Using the variational principle, Equation (1) yields the ODE of discretized system as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f}. \tag{5}$$

For the clarity of derivations, the time evolutions of a spatial node “ i ” are written as

$$\begin{aligned} \mathbf{u}_i^T &= [u_{i1} \quad u_{i2} \quad u_{i3} \quad \cdots \quad u_{i(N-1)} \quad u_{iN}] \\ \mathbf{f}_i^T &= [f_{i1} \quad f_{i2} \quad \cdots \quad f_{iN}] \end{aligned} \tag{6}$$

for the time sampling points 1 to N , and N is the number of time sampling points. Note that u_{ij} and f_{ij} are the values of u_i and f_i at time sampling point t_j . For a Multiple DOF dynamic system (Equation (5)), each spatial DOF is the function of time; its derivative with respect to time can be given through DQ rule as

$$\dot{\mathbf{u}}_i = \mathbf{A}\mathbf{u}_i, \tag{7}$$

where $\mathbf{A} = [A_{jk}]_{N \times N}$ is the explicit weighting coefficients matrix [46]. It is noteworthy that the weighting coefficients are independent of variables, and dependent only on coordinates of time sampling points. Fung [50] solved Equation (5) by using DQM. However, Equation (1) is discretized using the DQ and Gauss–Lobatto quadrature rules in this work.

For the development of the present TFEM, all terms of Equation (1) should be dealt with by the DQ and Gauss–Lobatto quadrature rules, that is, $\dot{\mathbf{u}}$ is expressed in terms of \mathbf{u} based on the DQ rule and the integrals are fulfilled by using the Gauss–Lobatto quadrature rule; then we have

$$\int_{t_1}^{t_2} \delta T dt = \int_{t_1}^{t_2} \dot{\mathbf{u}}^T \mathbf{M} \delta \mathbf{u} dt = \bar{\mathbf{U}}^T \bar{\mathbf{M}} \delta \bar{\mathbf{U}} \tag{8}$$

$$\int_{t_1}^{t_2} \delta V dt = \int_{t_1}^{t_2} \mathbf{u}^T \mathbf{K} \delta \mathbf{u} dt = \bar{\mathbf{U}}^T \bar{\mathbf{K}} \delta \bar{\mathbf{U}} \tag{9}$$

$$\begin{aligned} \int_{t_1}^{t_2} \delta W dt &= \int_{t_1}^{t_2} (\dot{\mathbf{u}}^T \mathbf{C} \delta \mathbf{u} + \mathbf{f}^T \delta \mathbf{u}) dt \\ &= (\bar{\mathbf{U}}^T \bar{\mathbf{C}} + \bar{\mathbf{F}}^T \bar{\mathbf{E}}) \delta \bar{\mathbf{U}} \end{aligned} \tag{10}$$

where

$$\bar{\mathbf{U}} = \text{cs}(\mathbf{U}), \quad \bar{\mathbf{F}} = \text{cs}(\mathbf{F}) \tag{11}$$

$$\mathbf{U}^T = [u_1 \quad u_2 \quad \cdots \quad u_n], \quad \mathbf{F}^T = [f_1 \quad f_2 \quad \cdots \quad f_n] \tag{12}$$

$$\bar{\mathbf{M}} = (\mathbf{A}^T \mathbf{C}_G \mathbf{A}) \otimes \mathbf{M}, \quad \bar{\mathbf{K}} = \mathbf{C}_G \otimes \mathbf{K}, \quad \bar{\mathbf{C}} = (\mathbf{A}^T \mathbf{C}_G) \otimes \mathbf{C} \tag{13}$$

$$\mathbf{C}_G = \text{diag}(\mathbf{C}_j), \quad \bar{\mathbf{E}} = \mathbf{C}_G \otimes \mathbf{E}, \tag{14}$$

where ‘cs’ indicates column expansion of a matrix, ‘ \otimes ’ is the Kronecker product, \mathbf{E} is a $n \times n$ identity matrix, and \mathbf{C}_j is the coefficient of Gauss–Lobatto quadrature. Diagonal matrix $\mathbf{C}_G = \text{diag}(\mathbf{C}_j)$. Appendix A presents the derivations of Equations (8) to (10). Equation (13) shows that the expanded mass matrix $\bar{\mathbf{M}}$ and stiffness $\bar{\mathbf{K}}$ are symmetrical, but $\bar{\mathbf{C}}$ is not, even though \mathbf{C} is symmetrical.

Substituting Equations (8) to (10) into Equation (1), and noting that $\delta \bar{\mathbf{U}}$ is arbitrary, we have the dynamic equilibrium equation in algebraic form as

$$\bar{\mathbf{U}}^T [\bar{\mathbf{K}} - \bar{\mathbf{M}} - \bar{\mathbf{C}}] = \bar{\mathbf{F}}^T \bar{\mathbf{E}}, \tag{15}$$

or in a compact and standard form as

$$G\bar{U} = Q, \tag{16}$$

where

$$G = \bar{K} - \bar{M} - (\bar{C})^T, \quad Q = \bar{E}F. \tag{17}$$

Equation (16) can also be written in a different form as

$$KUC_G - MUA^T C_G A - CUC_G A = FC_G \tag{18}$$

which can be reduced to a Sylvester equation whose solutions can be found by the method of Bartels and Stewart [51] if we neglect damping. Then it has no row or column expansion of matrix in Equation (18), or it has no additional storage needed except for A and the diagonal matrix C_G when compared with a step-by-step method.

After deriving Equation (15), a DQTFEM has been formulated, which is different from classical TFEMs where Hermite interpolations are used and displacement and velocity are involved for all nodes. From the above derivation, one can see that, in DQTFEM, displacements are the only nodal parameter at all time sampling points since Lagrange interpolations are used; refer to Equations (11) and (12). In order to impose initial conditions and satisfy C^1 -compatible conditions in the present method, the nodal parameter vector \bar{U} should be modified to include velocities for all spatial DOFs at the first time sampling point. To achieve this, an accurate and convenient method is to modify \bar{U} as

$$\tilde{U}^T = \left[\begin{array}{cccccccc} u_{11} & \cdots & u_{n1}, & u_{12} & \cdots & u_{n2}, & \cdots & , u_{1(N-1)} & \cdots & u_{n(N-1)}, & \dot{u}_{11} & \cdots & \dot{u}_{n1} \end{array} \right], \tag{19}$$

which is formulated by substituting the displacements u_{1N}, \cdots, u_{nN} of the N th time sampling point by the velocities $\dot{u}_{11}, \cdots, \dot{u}_{n1}$ of the first time sampling point. The relation between \tilde{U} and \bar{U} can be readily obtained by DQ rule as

$$\tilde{U} = H\bar{U} \tag{20}$$

$$H = T \otimes E \tag{21}$$

where T is formed by substituting the first row of A for the N th row of an unit matrix, that is

$$T = \left[\begin{array}{cccccccc} 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ A_{11} & A_{12} & A_{13} & \cdots & A_{1(N-2)} & A_{1(N-1)} & A_{1N} & \end{array} \right]. \tag{22}$$

From Equations (16) and (20) one can obtain

$$\tilde{G}\tilde{U} = Q \tag{23}$$

$$\tilde{G} = GH^{-1}. \tag{24}$$

It should be noted that the solution \tilde{U} of Equation (23) does not include displacements and velocities of the N th sampling point, referred to in Equation (19), but these can be easily obtained through Equation (20) and the DQ rule, respectively. Rearranging the terms of Equation (23), it is rewritten as

$$\left[\begin{array}{cc} \tilde{G}_{ii} & \tilde{G}_{id} \\ \tilde{G}_{di} & \tilde{G}_{dd} \end{array} \right] \left[\begin{array}{c} \tilde{U}_i \\ \tilde{U}_d \end{array} \right] = \left[\begin{array}{c} Q_i \\ Q_d \end{array} \right], \tag{25}$$

where subscript ‘i’ indicates the initial conditions, and ‘d’ indicates unknowns. From Equation (25) we have

$$\tilde{\mathbf{U}}_d = \tilde{\mathbf{G}}_{dd}^{-1}(\mathbf{Q}_d - \tilde{\mathbf{G}}_{di}\tilde{\mathbf{U}}_i), \tag{26}$$

where $\tilde{\mathbf{U}}_i$ is formulated by initial displacements and velocities, thus the initial conditions are dealt with in a general and convenient way.

It is worth noting that here $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$, corresponding to all time sampling points, are solved concurrently by using the exact initial conditions; this is different from classical step-by-step DIMs, in which the results of the present time step are the initial conditions of the next time step. That is, only the initial conditions of the first time step are exact; all others are approximate. However, DQTFEM can also serve as a step-by-step method; in this situation, a time element is a time step in which there are N time sampling points. Therefore, the initial conditions except for the first time step are also approximate, but the approximate initial conditions are highly accurate since the present method is a high-order method.

Remarks:

- (1) Usually for conservative systems, the actual path will be the one for which the integral of Lagrangian function ($T - V$) is an extremum, and for the non-conservative system considered here, the general form of Hamilton’s variational principle is used and the integral still exists; however, there exists no functional that must be an extremum [49].
- (2) The method of imposing initial conditions in the present method is standard and similar to the one used in the FEM; see Equation (25).
- (3) DQTFEM is a new TFEM, since Gauss–Lobatto quadrature and DQ rules are used for the first time in the construction of a TFEM. The high accuracy and efficiency due to the employment of Gauss–Lobatto quadrature and DQ rules are validated in the following numerical comparisons; see Section 4.

3. Stability Analysis

Stability analysis is significant for a new TFEM, and for this analysis a spectral approach is generally employed because a linear dynamic system can be decoupled into a set of single DOF systems based on the orthogonality of mode functions or mode vectors. Therefore, in the stability analysis of the present method, consider a single DOF system as

$$\ddot{u} + (\omega h)^2 u = 0, \tag{27}$$

where ω denotes the angular frequency. Here, the two dots over u denote the second derivative with respect to time τ , not to t , where $\tau = t/h$, and h and t stand for time step size and the real time, respectively. For Equation (27), the displacement vector \mathbf{u} in Equation (6) becomes

$$\mathbf{u}^T = [u_1 \quad u_2 \quad u_3 \quad \dots \quad u_N], \tag{28}$$

where u_j ($j = 1, 2, \dots, N$) denotes the displacements at time sampling points, the subscript ‘i’ in Equation (6) denotes the spatial DOF number and is omitted here since $i = 1$, and

$$\mathbf{G} = \mathbf{A}^T \mathbf{C}_G \mathbf{A} - (\omega h)^2 \mathbf{C}_G, \mathbf{U} = \mathbf{u}, \mathbf{Q} = \mathbf{0} \tag{29}$$

The nodal parameter vector in Equation (19) has the form of

$$\tilde{\mathbf{U}}^T = \left[u_1 \quad u_2 \quad u_3 \quad \dots \quad u_{N-2} \quad u_{N-1} \quad \dot{u}_1 \right]. \tag{30}$$

According to Equation (26), one can obtain

$$\tilde{\mathbf{U}}_d = \begin{bmatrix} u_2 \\ u_3 \\ \vdots \\ u_{N-1} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{\alpha} & \boldsymbol{\beta} \end{bmatrix} \begin{bmatrix} u_0 \\ \dot{u}_0 \end{bmatrix}, \tag{31}$$

where column vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are extracted from $\tilde{\mathbf{G}}$, and u_0 and \dot{u}_0 are the initial displacement and velocity, respectively. Moreover, using DQ rule yields

$$\begin{aligned} \dot{u}_1 = \dot{u}_0 &= A_{11}u_0 + \begin{bmatrix} A_{12} & \cdots & A_{1(N-1)} \end{bmatrix} \tilde{\mathbf{U}}_d + A_{1N}u_N \\ \dot{u}_N &= A_{N1}u_0 + \begin{bmatrix} A_{N2} & \cdots & A_{N(N-1)} \end{bmatrix} \tilde{\mathbf{U}}_d + A_{NN}u_N \end{aligned} \tag{32}$$

From Equations (31) and (32), we have

$$\begin{bmatrix} u_N \\ \dot{u}_N \end{bmatrix} = \mathbf{J} \begin{bmatrix} u_0 \\ \dot{u}_0 \end{bmatrix}, \tag{33}$$

which is the recurrence relation between (u_N, \dot{u}_N) and (u_0, \dot{u}_0) , and the elements of the Jacobi matrix \mathbf{J} are

$$\begin{cases} J_{11} = A_{1N}^{-1} \left(\begin{bmatrix} A_{12} & \cdots & A_{1(N-1)} \end{bmatrix} \boldsymbol{\alpha} - A_{11} \right) \\ J_{12} = A_{1N}^{-1} \left(\begin{bmatrix} A_{12} & \cdots & A_{1(N-1)} \end{bmatrix} \boldsymbol{\beta} + 1 \right) \\ J_{21} = A_{N1} - \begin{bmatrix} A_{N2} & \cdots & A_{N(N-1)} \end{bmatrix} \boldsymbol{\alpha} + A_{NN}J_{11} \\ J_{22} = A_{NN}J_{12} - \begin{bmatrix} A_{N2} & \cdots & A_{N(N-1)} \end{bmatrix} \boldsymbol{\beta} \end{cases} \tag{34}$$

According to the relationship between ωh and spectral radius $\rho(\mathbf{J})$, see Figure 1, the first stable intervals for different N can be obtained as given in Table 1. From Table 1 and Figure 1 we can make the following observations:

- (1) $\rho(\mathbf{J}) \geq 1$ for any N , and when N becomes larger, the first stable interval also becomes larger. Although there are several stable intervals with $\rho(\mathbf{J}) = 1$ except for $N = 3$, the first stable interval is significant and practical; therefore, only the first intervals are listed in Table 1 whose elements are obtained by using the criterion $(\rho - 1) < 10^{-6}$.
- (2) $\rho(\mathbf{J}) < 1$ in the TFEM by Hulbert [35], but $\rho(\mathbf{J}) = 1$ in DQTFEM for all stable intervals, hence DQTFEM is more accurate than that of Hulbert [35] since DQTFEM is a high-order method that has a smaller phase error than lower order methods. Note that the stability of CDM is $\omega h \leq 2$, and $\omega h \leq 2\sqrt{2}$ for four-order explicit RKM, and the stability of the present method is $\omega h \leq 2\sqrt{2}$.
- (3) DQTFEM is highly symplectic. If the Jacobi matrix \mathbf{J} of a method satisfies Equation (35) as follows

$$\begin{cases} J_1 = J_{11}J_{22}^T - J_{12}J_{21}^T = I & , & J_2 = J_{11}^TJ_{22} - J_{21}^TJ_{12} = I \\ J_3 = J_{11}^TJ_{21} - J_{21}^TJ_{11} = \mathbf{0} & , & J_4 = J_{12}^TJ_{22} - J_{22}^TJ_{12} = \mathbf{0} \\ J_5 = J_{11}J_{12}^T - J_{12}J_{11}^T = \mathbf{0} & , & J_6 = J_{21}J_{22}^T - J_{22}J_{21}^T = \mathbf{0} \end{cases} \tag{35}$$

then the method is symplectic. For the present single DOF system, all variables in Equation (35) are numbers and Table 2 shows that the precision of Equation (35) (especially $J_3 \sim J_6$) is satisfactory, which indicates that DQTFEM is a symplectic method. In fact, we have numerically verified that DQTFEM is symplectic even if it is not stable. For a stable symplectic method, the magnitudes of all eigenvalues of \mathbf{J} are equal to 1, therefore all eigenvalues of \mathbf{J} of the present method are equal to 1 when it is stable.

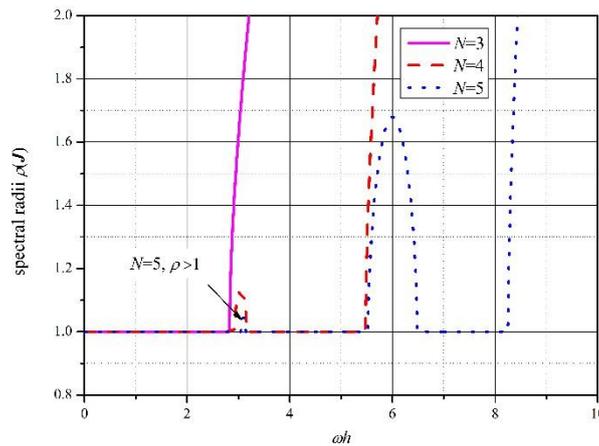


Figure 1. Spectral radii for different N.

Table 1. The first stable intervals with $\rho(J) = 1$ for different N.

N	3	4	5	10	15	20	25	30	36
$0 < \omega h \leq$	2.828	2.927	3.055	9.404	15.71	25.11	34.55	37.69	50.25

Table 2. Precision of $J_1 \sim J_6$, $|J|$ and $\rho(J)$.

$(N, \max(\omega h))$	$\text{abs}(J_1 - 1)$	$\text{abs}(J_2 - 1)$	$J_i, i = 3, 4, 5, 6$	$\text{abs}(J - 1)$	$\text{abs}(\rho(J) - 1)$
(3, 2.828)	2.459×10^{-7}	2.459×10^{-7}	0	2.812×10^{-7}	1.486×10^{-9}
(4, 2.927)	6.758×10^{-8}	6.758×10^{-8}	0	7.002×10^{-8}	4.289×10^{-9}
(5, 3.055)	2.817×10^{-8}	2.817×10^{-8}	0	3.018×10^{-8}	1.087×10^{-9}
(10, 9.404)	3.648×10^{-10}	3.648×10^{-10}	0	3.821×10^{-10}	1.821×10^{-12}
(15, 15.70)	1.085×10^{-10}	1.085×10^{-10}	0	1.663×10^{-10}	7.223×10^{-12}
(20, 25.11)	3.345×10^{-10}	3.345×10^{-10}	0	3.598×10^{-10}	4.538×10^{-11}
(25, 34.55)	1.784×10^{-8}	1.784×10^{-8}	0	1.999×10^{-8}	6.046×10^{-10}
(30, 37.69)	4.586×10^{-6}	4.586×10^{-6}	0	4.939×10^{-6}	7.919×10^{-8}
(35, 50.25)	3.882×10^{-6}	3.882×10^{-6}	0	4.051×10^{-6}	6.124×10^{-8}

Note: $\max(\omega h)$ indicates the upper limit of ωh for the first stable interval in Table 1.

4. Numerical Results and Discussion

This section aims at demonstrating the high accuracy and efficiency of DQTFEM by comparing them with RKM and NMA; note that NMA is equivalent to the Euler mid-point symplectic method. To evaluate the relative computational accuracy of the present method, the relative error for all methods is defined as

$$\text{RelErr} = \frac{|\text{Exact} - \text{Approximate}|}{\|\text{Exact}\|_\infty}, \tag{36}$$

where ‘Exact’ denotes the exact solution at each time sampling point, the infinity norm of which denotes the maximum absolute value of a vector, and the elements of this vector are the exact values at all time sampling points; ‘Approximate’ is the result of DQTFEM at each time sampling point.

4.1. Free Vibration of a Two-DOF System

Consider the following two-DOF system, its governing equilibrium equation is

$$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{bmatrix} + \begin{bmatrix} 6 & -2 \\ -2 & 4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{37}$$

whose angular frequencies $\omega_1 = \sqrt{2}$ and $\omega_2 = \sqrt{5}$. The given initial conditions are $u_0^T = [1 \ 1]$ and $\dot{u}_0^T = [0 \ 0]$.

4.1.1. Accuracy Comparison

Let time domain be $[0, 19]$, which is taken as an element with $N = 33$ non-uniform points in DQTFEM, and 32 uniform time steps are used for both RKM and NMA. The first stable interval for $N = 33$ is $[0, 43.97]$, hence the stability is satisfied since $19\omega_2 = 42.49 < 43.97$, and the stability $2\sqrt{2}$ of RKM is also satisfied by $h = 19/32 = 0.59375$ since $\omega_2 h = 1.3277 < 2\sqrt{2}$. Additionally, $N = 33$ and $h = 0.59375$ implies that the three methods have the same number of interpolation points in the domain $[0, 19]$. The exact solutions used for comparisons are obtained by the mode superposition method.

It follows from Figures 2 and 3 that DQTFEM is much more accurate than RKM and NMA, RelErr (DQTFEM) is about 10^{-6} , while RelErr (RKM) and RelErr (NMA) are 10^{-2} and 10^{-1} , respectively; and it is noticeable from Figure 3 that the errors of RKM and NMA accumulate over time.

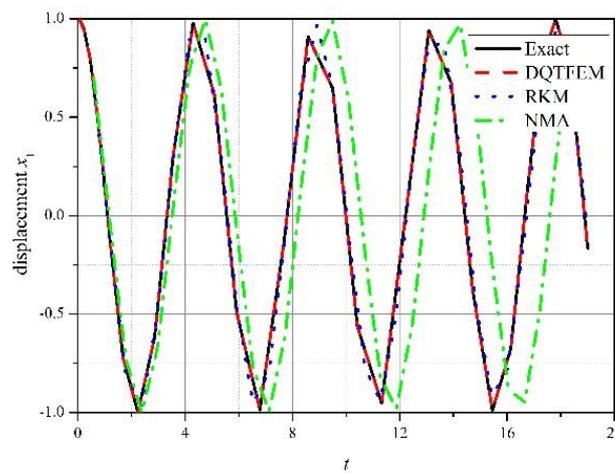


Figure 2. Comparisons of displacement x_1 ($N = 33, h = 0.59375$).

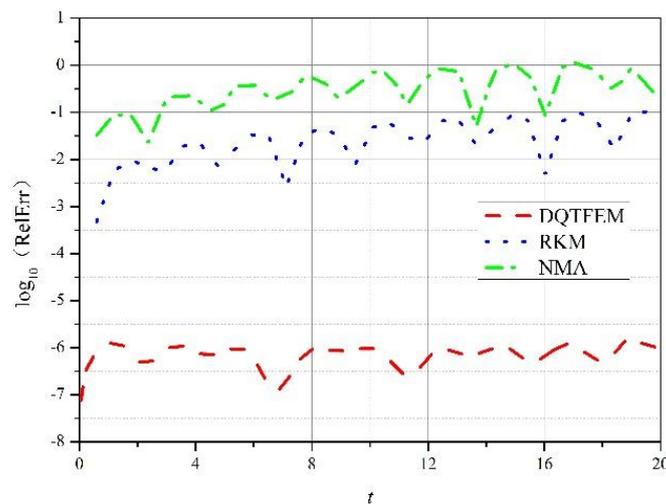


Figure 3. RelErr comparisons ($N = 33, h = 0.59375$).

4.1.2. Efficiency Comparison

Computational cost is also a crucial issue for a method. Assume the time domain is $[0, 2 \times 10^3]$ in this case. In order to achieve the same accuracy of $\text{RelErr} = 10^{-6}$, $h = 5 \times 10^{-2}$ and $h = 1 \times 10^{-3}$

must be used for RKM and NMA, respectively; that is, RKM needs 4×10^5 time steps and NMA needs 2×10^6 time steps. However, we can achieve the same accuracy by using 1588 elements and each element of size $h = 1.26 \approx 2.828/\omega_2$ has only three Gauss–Lobatto–Legendre sampling points ($N = 3$ is the least number of time sampling points in an element; see Table 2). If $N > 3$, the accuracy of DQTFEM is more than 10^{-6} . For example, if $N = 20$ in an element of size $h = 11 \approx 25.11/\omega_2$, the accuracy will be 10^{-9} . In other words, the lowest accuracy of DQTFEM is 10^{-6} . This is the reason why the data corresponding to $\text{RelErr} = 10^{-2}$ are not provided in Table 3, which indicates that DQTFEM is much more efficient than the other two methods, especially if high accuracy is required. Another observation is that RKM is more efficient than NMA.

NMA is a symplectic method of second order, in which the amplitude can be rigorously preserved for linear systems, but its phase error is larger than that of RKM. It is well known that dynamic responses must be represented by amplitude and phase together; therefore, RKM is more efficient than NMA for achieving the same accuracy. However, the weakness of RKM is its numerical dissipation and the weakness of NMA is its larger phase error. For long-time kinematics simulations in which only amplitude is taken into account, NMA is a better choice.

Table 3. CPU time comparison.

Method	RelErr = 10^{-6}			RelErr = 10^{-2}		
	h	Time Steps	CPU (s)	h	Time Steps	CPU (s)
DQTFEM	1.26	1588	0.4932			
RKM	5×10^{-2}	4×10^5	4.743	0.32	6.25×10^3	0.7843
NMA	1×10^{-3}	2×10^6	249.4	4.5×10^{-2}	4.445×10^4	1.184

4.1.3. Damping Vibration

For further validation of the present method, Rayleigh damping given in Equation (38) is added to the system as Equation (37).

$$C = \begin{bmatrix} 0.3 & -0.1 \\ -0.1 & 0.2 \end{bmatrix} \tag{38}$$

$N = 33$ and $h = 0.59375$ are also used here. The displacement x_1 and RelErr are given in Figures 4 and 5, respectively, from which one can see that DQTFEM is also the most accurate: its RelErr reaches 10^{-6} . Therefore, the present method is also applicable to the analysis of damped problems.

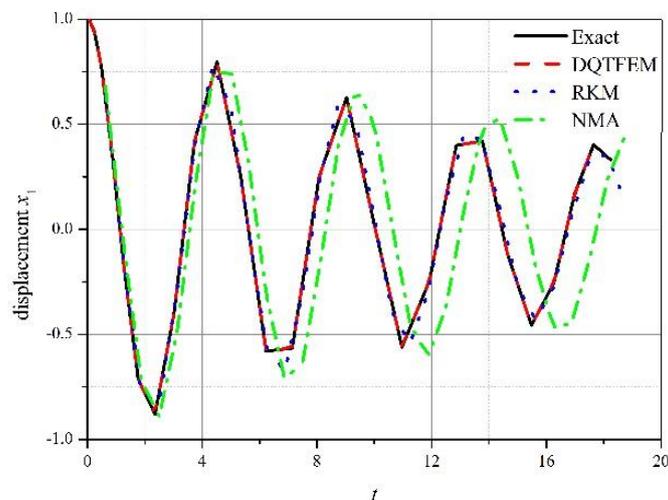


Figure 4. Displacement comparison for damping case ($N = 33, h = 0.59375$).

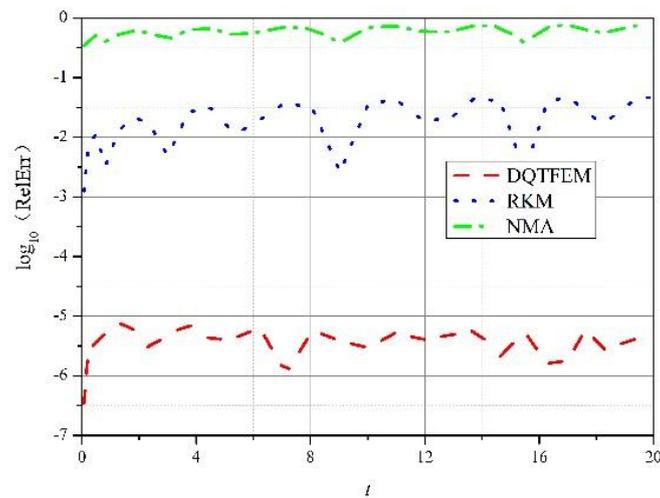


Figure 5. RelErr for damping case ($N = 33, h = 0.59375$).

4.2. Free Vibration of a Simply Supported Euler Beam

In order to further explore the accuracy and efficiency of DQTFEM, the free vibration of a simply supported Euler beam of length $L = 3$ m is studied in this section; flexural rigidity $EI = 1 \times 10^6$ N/m and mass density per unit length $\rho A = 420$ kg/m. The beam is meshed into 16 uniform cubic elements. All initial velocities and displacements are equal to zero except the initial velocity at mid-span is 1 m/s; apparently the problem is analogous to the impact problem.

Consider time domain $[0, 3.6 \times 10^{-3}]$ s, which is meshed into 10 uniform time elements, that is $h = 3.6 \times 10^{-4}$ s, and 20 Gauss–Lobatto–Legendre sampling points are involved in each element. Note that, if the highest frequency is $\omega_{33} = 6.967 \times 10^4$ rad/s, then $\omega_{33}h = 25.08$ falls into the first stable interval $0 < \omega h \leq 25.11$ for $N = 20$; see Table 1.

Figure 6 shows an excellent agreement of DQTFEM with the exact method. For reaching the same order of accuracy, NMA needs 2.110×10^5 time steps and much more CPU time than DQTFEM; see Table 4. Figure 7 presents the RelErr comparisons between DQTFEM and NMA; it follows that the error of NMA accumulates while the error of DQTFEM does not.

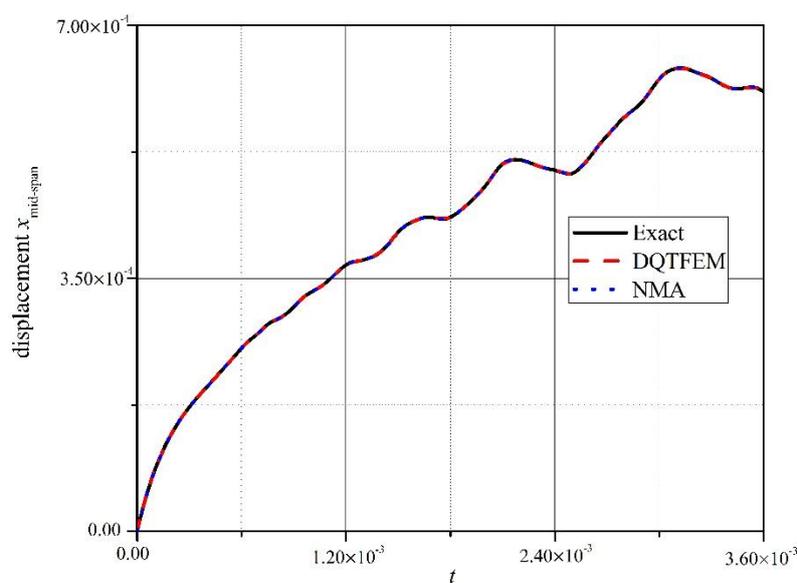


Figure 6. Displacement comparison of beam.

Table 4. CPU time comparison between DQTFEM and NMA (RelErr = 10^{-7}).

Method	h (s)	Time Steps	CPU (s)
DQTFEM	3.6×10^{-4}	10	0.9376
NMA	1.706×10^{-8}	2.110×10^5	294.8

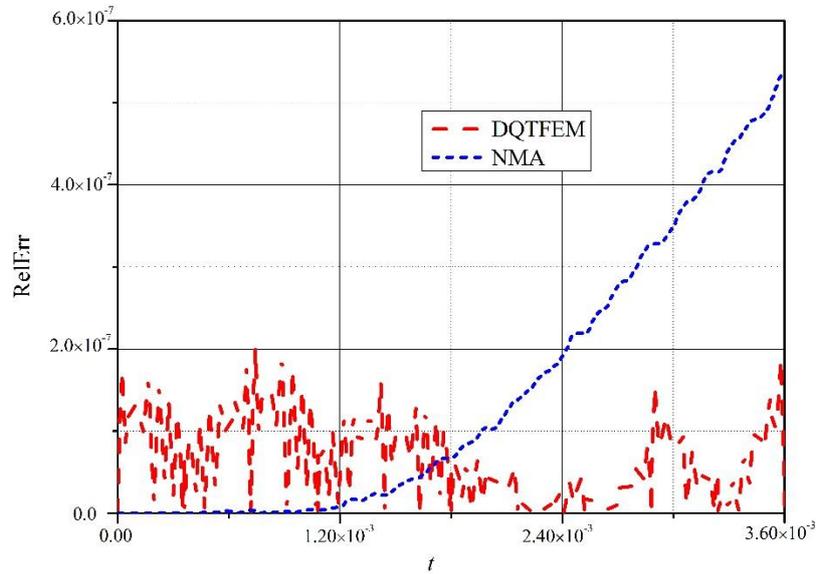


Figure 7. RelErr comparison between DQTFEM and NMA for $x_{\text{mid-span}}$.

For validating the effectiveness of DQTFEM for long-term simulation, the computational time changes from 3.6×10^{-3} s to three times the maximum period $T_{\text{max}} = 2\pi/\omega_1 = 0.1174$ s, where $\omega_1 = 53.51$ rad/s, the number of elements is $3T_{\text{max}}/h \approx 978$. Figure 8 gives a comparison of displacements by DQTFEM with the exact solution, which shows that DQTFEM is also highly accurate for this case and the needed CPU time is only 75.53 s. Nevertheless, the NMA results are not presented since NMA requires a prohibitively long CPU time.

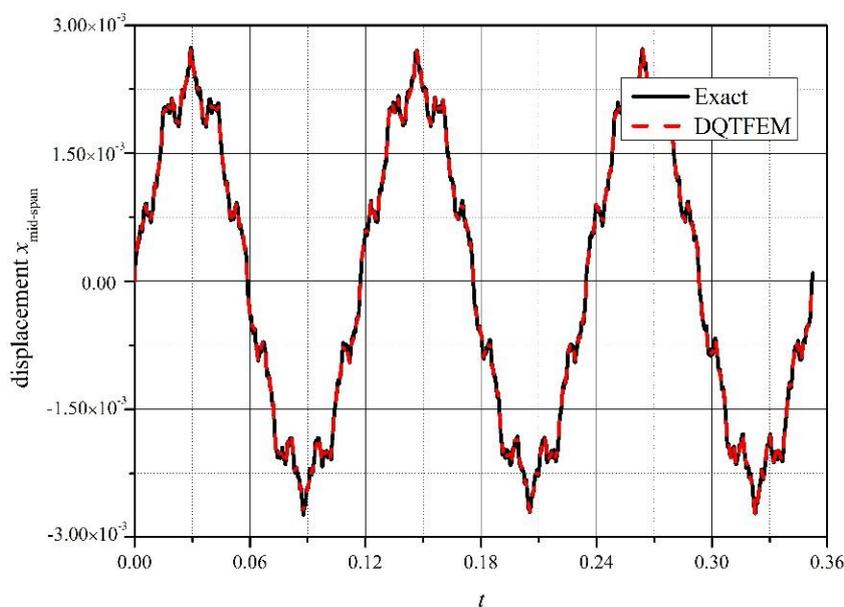


Figure 8. Displacement comparison for long-term simulation.

5. Conclusions

A highly accurate and highly efficient DQTFEM was developed in the present study by using the DQ rule, the Gauss–Lobatto integration rule, and the general form of Hamilton’s variational principle for solving structural dynamic problems involving damping and external forces.

DQTFEM incorporates the high accuracy and efficiency of DQM into the development of high-order time elements, in which initial conditions can be introduced in the same simple and general form as in standard FEM. Furthermore, DQTFEM can serve as a step-by-step approach; one time element of DQTFEM can be seen as one time step of the conventional step-by-step method. Moreover, it has been shown that DQTFEM is a highly accurate symplectic method.

Extensive numerical comparisons validated the present method, whose solutions can be seen as benchmarks to validate lower-order DIMs and TFEMs. It is expected that DQTFEM will be practical in the future.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (11172028, 11372021, 11672019 and 37686003) and the Research Fund for the Doctoral Program of Higher Education of China (20131102110039).

Author Contributions: Yufeng Xing conceived of this study and finalized the manuscript. Mingbo Qin derived some of the equations, performed the calculations, and prepared the draft of the manuscript. Jing Guo also prepared part of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

The derivation procedure of Equations (8)–(10) is as follows:

$$\begin{aligned}
 \int_{t_1}^{t_2} \delta T dt &= \int_{t_1}^{t_2} \dot{\mathbf{u}}^T \mathbf{M} \delta \dot{\mathbf{u}} dt = \sum_{j=1}^N C_j (\mathbf{D}_j)^T \mathbf{M} \delta \mathbf{D}_j \\
 &= \begin{bmatrix} (\mathbf{D}_1)^T & (\mathbf{D}_2)^T & \dots & (\mathbf{D}_N)^T \end{bmatrix} \begin{bmatrix} C_1 \mathbf{M} & & & \\ & C_2 \mathbf{M} & & \\ & & C_3 \mathbf{M} & \\ & & & C_4 \mathbf{M} \end{bmatrix} \delta \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_N \end{bmatrix} \tag{A1} \\
 &= \bar{\mathbf{U}}^T (\mathbf{A}^T \otimes \mathbf{E}) (\mathbf{C}_G \otimes \mathbf{M}) (\mathbf{A} \otimes \mathbf{E}) \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G \mathbf{A}) \otimes (\mathbf{E} \mathbf{M} \mathbf{E}) \} \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G \mathbf{A}) \otimes \mathbf{M} \} \delta \bar{\mathbf{U}}
 \end{aligned}$$

$$\begin{aligned}
 \int_{t_1}^{t_2} \delta T dt &= \int_{t_1}^{t_2} \dot{\mathbf{u}}^T \mathbf{M} \delta \dot{\mathbf{u}} dt = \sum_{j=1}^N C_j (\mathbf{D}_j)^T \mathbf{M} \delta \mathbf{D}_j \\
 &= \begin{bmatrix} (\mathbf{D}_1)^T & (\mathbf{D}_2)^T & \dots & (\mathbf{D}_N)^T \end{bmatrix} \begin{bmatrix} C_1 \mathbf{M} & & & \\ & C_2 \mathbf{M} & & \\ & & C_3 \mathbf{M} & \\ & & & C_4 \mathbf{M} \end{bmatrix} \delta \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_N \end{bmatrix} \tag{A2} \\
 &= \bar{\mathbf{U}}^T (\mathbf{A}^T \otimes \mathbf{E}) (\mathbf{C}_G \otimes \mathbf{M}) (\mathbf{A} \otimes \mathbf{E}) \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G \mathbf{A}) \otimes (\mathbf{E} \mathbf{M} \mathbf{E}) \} \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G \mathbf{A}) \otimes \mathbf{M} \} \delta \bar{\mathbf{U}}
 \end{aligned}$$

$$\begin{aligned}
 \int_{t_1}^{t_2} \delta V dt &= \int_{t_1}^{t_2} \mathbf{u}^T \mathbf{K} \delta \mathbf{u} dt = \sum_{j=1}^N C_j (\mathbf{U}_j)^T \mathbf{K} \delta (\mathbf{U}_j) \\
 &= \begin{bmatrix} (\mathbf{U}_1)^T & (\mathbf{U}_2)^T & \cdots & (\mathbf{U}_N)^T \end{bmatrix} \begin{bmatrix} C_1 \mathbf{K} & & & \\ & C_2 \mathbf{K} & & \\ & & C_3 \mathbf{K} & \\ & & & C_4 \mathbf{K} \end{bmatrix} \delta \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_N \end{bmatrix} \quad (A3) \\
 &= \bar{\mathbf{U}}^T (\mathbf{C}_G \otimes \mathbf{K}) \delta \bar{\mathbf{U}}
 \end{aligned}$$

$$\begin{aligned}
 \int_{t_1}^{t_2} \delta W dt &= \int_{t_1}^{t_2} (\dot{\mathbf{u}}^T \mathbf{C} \delta \mathbf{u} + \mathbf{f}^T \delta \mathbf{u}) dt \\
 &= \sum_{j=1}^N C_j \{ (\mathbf{D}_j)^T \mathbf{C} + (\mathbf{F}_j)^T \} \delta (\mathbf{U}_j) \\
 &= \begin{bmatrix} (\mathbf{D}_1)^T & (\mathbf{D}_2)^T & \cdots & (\mathbf{D}_N)^T \end{bmatrix} \begin{bmatrix} C_1 \mathbf{C} & & & \\ & C_2 \mathbf{C} & & \\ & & C_3 \mathbf{C} & \\ & & & C_4 \mathbf{C} \end{bmatrix} \delta \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_N \end{bmatrix} + \\
 &\quad \begin{bmatrix} C_1 (\mathbf{F}_1)^T & C_2 (\mathbf{F}_2)^T & \cdots & C_N (\mathbf{F}_N)^T \end{bmatrix} \delta \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_N \end{bmatrix}, \quad (A4) \\
 &= \bar{\mathbf{U}}^T (\mathbf{A}^T \otimes \mathbf{E}) (\mathbf{C}_G \otimes \mathbf{C}) \delta \bar{\mathbf{U}} + (\text{cs}(\mathbf{F}))^T (\mathbf{C}_G \otimes \mathbf{E}) \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G) \otimes (\mathbf{E} \mathbf{C}) \} \delta \bar{\mathbf{U}} + (\text{cs}(\mathbf{F}))^T (\mathbf{C}_G \otimes \mathbf{E}) \delta \bar{\mathbf{U}} \\
 &= \bar{\mathbf{U}}^T \{ (\mathbf{A}^T \mathbf{C}_G) \otimes \mathbf{C} \} \delta \bar{\mathbf{U}} + (\text{cs}(\mathbf{F}))^T (\mathbf{C}_G \otimes \mathbf{E}) \delta \bar{\mathbf{U}}
 \end{aligned}$$

where

$$(\mathbf{D}_j)^T = \begin{bmatrix} \sum_{k=1}^N A_{jk} u_{1k} & \sum_{k=1}^N A_{jk} u_{2k} & \cdots & \sum_{k=1}^N A_{jk} u_{nk} \end{bmatrix} \quad (j = 1, 2, \dots, N) \quad (A5)$$

$$(\mathbf{U}_j)^T = \begin{bmatrix} u_{1j} & u_{2j} & \cdots & u_{nj} \end{bmatrix} \quad (A6)$$

$$(\mathbf{F}_j)^T = \begin{bmatrix} f_{1j} & f_{2j} & \cdots & f_{nj} \end{bmatrix} \quad (A7)$$

References

1. Feng, K. Difference schemes for hamiltonian formalism and symplectic geometry. *J. Comput. Math.* **1986**, *4*, 279–289.
2. Zienkiewicz, O.C. A new look at the newmark, houbolt and other time stepping formulas. A weighted residual approach. *Earthq. Eng. Struct. Dyn.* **1977**, *5*, 413–419. [[CrossRef](#)]
3. Argyris, J.H.; Scharpf, D.W. Finite elements in time and space. *Nucl. Eng. Des.* **1969**, *10*, 456–464. [[CrossRef](#)]
4. Fried, I. Finite element analysis of time dependent phenomena. *AIAA J.* **1969**, *7*, 1170–1173. [[CrossRef](#)]
5. Tang, W.S.; Sun, Y.J. Time finite element methods: A unified framework for numerical discretizations of odes. *Appl. Math. Comput.* **2012**, *219*, 2158–2180. [[CrossRef](#)]
6. Simkins, T.E. Finite elements for initial value problems in dynamics. *AIAA J.* **1981**, *19*, 1357–1362. [[CrossRef](#)]
7. Brugnano, L.; Iavernaro, F.; Trigiante, D. A simple framework for the derivation and analysis of effective one-step methods for odes. *Appl. Math. Comput.* **2012**, *218*, 8475–8486. [[CrossRef](#)]
8. Wilson, E.L.; Nickell, R.E. Application of the finite element method to heat conduction analysis. *Nucl. Eng. Des.* **1966**, *4*, 276–287. [[CrossRef](#)]
9. Zienkiewicz, O.C.; Wood, W.L.; Hine, N.W.; Taylor, R.L. A unified set of single step algorithms. Part 1: General formulation and applications. *Int. J. Numer. Methods Eng.* **1984**, *20*, 1529–1553. [[CrossRef](#)]
10. Richter, T.; Wick, T. Variational localizations of the dual weighted residual estimator. *J. Comput. Appl. Math.* **2015**, *279*, 192–209. [[CrossRef](#)]

11. Ma, L.M.; He, Y.A. A single-step time element method for dynamic response by gurtin variational principle. *Eng. Mech.* **1995**, *12*, 24–29.
12. Zhong, W.; Yao, Z. Time domain fem and symplectic conservation. *J. Mech. Strength* **2005**, *27*, 178–183.
13. Ding, X.C.; Chen, Q.W. Subinterval method for direct calculation of response of dynamic system by using hamilton's principle. *Earthq. Eng. Eng. Vib.* **1987**, *7*, 9–19.
14. Tung, M.M.; Defez, E.; Sastre, J. Numerical solutions of second-order matrix models using cubic-matrix splines. *Comput. Math. Appl.* **2008**, *56*, 2561–2571. [[CrossRef](#)]
15. Bellour, A.; Sbibih, D.; Zidna, A. Two cubic spline methods for solving fredholm integral equations. *Appl. Math. Comput.* **2016**, *276*, 1–12. [[CrossRef](#)]
16. Givoli, D.; Hagstrom, T.; Patlashenko, I. Finite element formulation with high-order absorbing boundary conditions for time-dependent waves. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 3666–3691. [[CrossRef](#)]
17. Jiang, Y.J.; Ma, J.T. High-order finite element methods for time-fractional partial differential equations. *J. Comput. Appl. Math.* **2011**, *235*, 3285–3291. [[CrossRef](#)]
18. Krishna, M.S.; Manjeet, S.K. Three and four step least square finite element schemes in the time domain. *Commun. Numer. Methods Eng.* **1996**, *12*, 425–432.
19. Jiwari, R. Lagrange interpolation and modified cubic b-spline differential quadrature methods for solving hyperbolic partial differential equations with dirichlet and neumann boundary conditions. *Comput. Phys. Commun.* **2015**, *193*, 55–66. [[CrossRef](#)]
20. Kawahara, M.; Hasegawa, K. Periodic galerkin finite element method of tidal flow. *Int. J. Numer. Methods Eng.* **1978**, *12*, 115–128. [[CrossRef](#)]
21. Jia, Z.; Cheng, X.; Zhao, M. A new method for roots of monic quaternionic quadratic polynomial. *Comput. Math. Appl.* **2009**, *9*, 1852–1859. [[CrossRef](#)]
22. Hoff, C.; Hughes, T.J.R.; Hulbert, G.; Pahl, P.J. Extended comparison of the hilber-hughes-taylor α -method and the θ 1-method. *Comput. Methods Appl. Mech. Eng.* **1989**, *76*, 87–94. [[CrossRef](#)]
23. Chang, J.C. Combining genetic algorithm and taylor series expansion approach for doa estimation in space-time cdma systems. *Appl. Soft Comput. J.* **2015**, *28*, 208–217. [[CrossRef](#)]
24. Khan, I.R.; Ohba, R.; Hozumi, N. Mathematical proof of closed form expressions for finite difference approximations based on taylor series. *J. Comput. Appl. Math.* **2003**, *150*, 303–310. [[CrossRef](#)]
25. Hoff, C.; Pahl, P.J. Practical performance of the θ 1-method and comparison with other dissipative algorithms in structural dynamics. *Comput. Methods Appl. Mech. Eng.* **1988**, *67*, 87–111. [[CrossRef](#)]
26. Reed, W.H.; Hill, T.R. *Triangular Mesh Methods for the Neutron Transport Equation*; Los Alamos Scientific Laboratory: Los Alamos, NM, USA, 1973.
27. Lasaint, P.; Raviart, P.A. On a finite element method for solving the neutron transport equation. In *Mathematical Aspects of Finite Elements in Partial Differential Equations, Proceedings of a Symposium Conducted by the Mathematics Research Center, the University of Wisconsin-Madison, Madison, WI, USA, 1–3 April, 1974*; Elsevier: Amsterdam, The Netherlands, 1974; Volume 56, pp. 89–124.
28. Gudla, P.K.; Ganguli, R. Discontinuous galerkin finite element in time for solving periodic differential equations. *Comput. Methods Appl. Mech. Eng.* **2006**, *196*, 682–697. [[CrossRef](#)]
29. Gao, Y.L.; Mei, L.Q.; Li, R. A time-splitting galerkin finite element method for the davey-stewartson equations. *Comput. Phys. Commun.* **2015**, *197*, 35–43. [[CrossRef](#)]
30. Delfour, M.; Hager, W.; Trochu, F. Discontinuous galerkin methods for ordinary differential equations. *Math. Comput.* **1981**, *36*, 455–474. [[CrossRef](#)]
31. Hughes, T.J.R.; Hulbert, G.M. Space-time finite element methods for elastodynamics: Formulations and error estimates. *Comput. Methods Appl. Mech. Eng.* **1988**, *66*, 339–364. [[CrossRef](#)]
32. Hulbert, G.M.; Hughes, T.J.R. Space-time finite element methods for second-order hyperbolic equations. *Comput. Methods Appl. Mech. Eng.* **1990**, *84*, 327–349. [[CrossRef](#)]
33. Chessa, J.; Belytschko, T. A local space-time discontinuous finite element method. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 1325–1344. [[CrossRef](#)]
34. Dyniewicz, B. Space-time finite element approach to general description of a moving inertial load. *Finite Elem. Anal. Des.* **2012**, *62*, 8–18. [[CrossRef](#)]
35. Hulbert, G.M. Time finite element methods for structural dynamics. *Int. J. Numer. Methods Eng.* **1992**, *33*, 307–331. [[CrossRef](#)]

36. Nam, J.; Behr, M.; Pasquali, M. Space-time least-squares finite element method for convection-reaction system with transformed variables. *Comput. Methods Appl. Mech. Eng.* **2011**, *200*, 2562–2577. [[CrossRef](#)] [[PubMed](#)]
37. Chien, C.C.; Wu, T.Y. An improved predictor/multi-corrector algorithm for a time-discontinuous galerkin finite element method in structural dynamics. *Comput. Mech.* **2000**, *25*, 430–438. [[CrossRef](#)]
38. Aubry, D.; Lucas, D.; Tie, B. Adaptive strategy for transient/coupled problems applications to thermoelasticity and elastodynamics. *Comput. Methods Appl. Mech. Eng.* **1999**, *1*, 41–51. [[CrossRef](#)]
39. Larsson, F.; Hansbo, P.; Runesson, K. Space-time finite elements and an adaptive strategy for the coupled thermoelasticity problem. *Int. J. Numer. Methods Eng.* **2003**, *56*, 261–294. [[CrossRef](#)]
40. Bellman, R.; Casti, J. Differential quadrature and long-term integration. *J. Math. Anal. Appl.* **1971**, *34*, 235–239. [[CrossRef](#)]
41. Shu, C. *Differential Quadrature and Its Application in Engineering*; Springer: London, UK, 2000.
42. Eftekhari, S.A.; Khani, M. A coupled finite element-differential quadrature element method and its accuracy for moving load problem. *Appl. Math. Model.* **2010**, *34*, 228–238. [[CrossRef](#)]
43. Eftekhari, S.A.; Jafari, A.A. Mixed finite element and differential quadrature method for free and forced vibration and buckling analysis of rectangular plates. *Appl. Math. Mech.* **2012**, *33*, 81–98. [[CrossRef](#)]
44. Wang, X.; Yuan, Z. Harmonic differential quadrature analysis of soft-core sandwich panels under locally distributed loads. *Appl. Sci.* **2016**, *6*, 361. [[CrossRef](#)]
45. Tornabene, F.; Fantuzzi, N.; Ubertini, F.; Viola, E. Strong formulation finite element method based on differential quadrature: A survey. *Appl. Mech. Rev.* **2015**, *1*, 145–175. [[CrossRef](#)]
46. Bert, C.W.; Malik, M. Differential quadrature method in computational mechanics: A review. *Appl. Mech. Rev.* **1996**, *49*, 1–29. [[CrossRef](#)]
47. Xing, Y.; Liu, B. High-accuracy differential quadrature finite element method and its application to free vibrations of thin plate with curvilinear domain. *Int. J. Numer. Methods Eng.* **2009**, *80*, 1718–1743. [[CrossRef](#)]
48. Xing, Y.F.; Liu, B.; Liu, G. A differential quadrature finite element method. *Int. J. Appl. Mech.* **2012**, *2*, 207–227. [[CrossRef](#)]
49. Reddy, J.N. *Energy and Variational Methods in Applied Mechanics*; John Wiley & Sons: New York, NY, USA, 1984.
50. Fung, T.C. Solving initial value problems by differential quadrature method-part 1: First-order equations. *Int. J. Numer. Methods Eng.* **2001**, *50*, 1411–1428. [[CrossRef](#)]
51. Bartels, R.H.; Stewart, G.W. Solution of the matrix equation $ax + xb = c$. *Commun. ACM* **1972**, *15*, 820–826. [[CrossRef](#)]



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).