Journal of Sound and Vibration (1998) **213**(3), 409–428

MIXED TIME FINITE ELEMENTS FOR VIBRATION RESPONSE ANALYSIS

T. C. Fung, S. C. Fan

School of Civil and Structural Engineering, Nanyang Technological University, Singapore

AND

G. Sheng

Data Storage Institute, 10 Kent Ridge Crescent, Singapore

(*Received 3 June 1996, and in final form 27 November 1997*)

This paper presents a mixed time finite element method for vibration response analysis. The underlying variational formulation is developed from the principle of virtual work. Conventional spatial discretization techniques are adopted. Time discretization is implemented by using mixed time finite elements. The proposed formulation not only forms a unified variational basis for spatial and temporal discretization; it also derives many other robust time step integration algorithms. Unconditionally stable and high order accurate algorithms with variable numerical dissipation can be constructed systematically. Moreover, the proposed formulations can be used to solve linear as well as non-linear problems. The accuracy and stability of the derived algorithms are presented.

1. INTRODUCTION

Recently there has been a lot of interest in the development of time finite elements. There are several attributes in this approach. First, it presents a unified scheme for space–time domain problems. Second, the mature approaches of the conventional spatial finite elements can be readily extended to space–time finite elements. Third, the approach could be applied to the energy directly as well as the governing differential equations. The general framework could cover temporal boundary value problems (e.g. periodic problems, optimal control problems etc.) as well as initial value problems. It turns out to be a robust tool for dynamics.

Early work on variational space–time finite element method can be found in [1–3]. The work was based on the direct numerical application of Hamilton's principle. Bailey [4, 5] proposed a direct numerical solution algorithm using the more general Hamilton's law of varying action. The work was then extended by many others. A comprehensive review can be found in [6].

Though the concept of finite elements in the time domain was proposed in the late sixties, for a long time it has not appeared to have significant advantages over other numerical integration schemes. The general perception of the time domain finite elements was therefore one where numerical divergence and instability were frequently encountered.

Recently, Borri *et al*. [7, 8] re-treated the tailing term and introduced a modified energy functional to the Hamilton's law. Apart from the inherent merits of the finite element method, time marching algorithms based on the hybrid or mixed time finite element formulation of Hamilton's law were shown to be competitive with the conventional time

0022–460X/98/230409 + 20 $25.00/0/sv971464

marching schemes. On the other hand, it is also noted that the time-discontinuous Galerkin method is a powerful time finite element method [9, 10]. The method has been extended to tackle many other dynamic problems [11, 12].

1.1. OUTLINE OF THIS PAPER

In this paper, a mixed time finite element method is presented for vibration response analysis. The underlying variational formulation is developed from the principle of virtual work. Conventional spatial discretization techniques are adopted. Time discretization is implemented by the mixed time finite elements. The formulation yields a set of unconditionally stable high-order-accurate algorithms with variable algorithmic damping. Many existing time finite elements are re-derived as special cases. The accuracy and stability of the derived algorithms are analysed. Several illustrative examples are given.

2. VARIATIONAL FORMULATION

For a general linear or nonlinear structural dynamic system, the principle of virtual displacement can be written as

$$\int_{\Omega} \delta\varepsilon^T\sigma \, \mathrm{d}\Omega - \delta \boldsymbol{W}_{\mathrm{ext}} + \int_{\Omega} \mu\delta\boldsymbol{q}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega + \int_{\Omega} \rho\delta\boldsymbol{q}^T\ddot{\boldsymbol{q}} \, \mathrm{d}\Omega = 0, \tag{1}$$

where $\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}$ denote the displacement, velocity and acceleration of a structure point respectively; $\boldsymbol{q} = (\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3)$, $\boldsymbol{q}_i = \boldsymbol{q}_i\,(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{t})$, $(i = 1, 2, 3)$; Ω, ρ, μ represent volume, mass density and viscosity density of an undeformed body respectively; $\boldsymbol{W}_{ext}$ is the work done by external loads; the δ symbol denotes virtual quantities; σ is the second Piola–Kirchhoff stress vector; ε is the Green–Lagrangian strain vector and can be defined as

$$\varepsilon_{ij} = \varepsilon_{ij}^{L} + \varepsilon_{ij}^{NL} = \tfrac{1}{2}\,(\boldsymbol{q}_{i,j} + \boldsymbol{q}_{j,i}) + \tfrac{1}{2}\,\boldsymbol{q}_{k,i}\,\boldsymbol{q}_{k,j}, \tag{2}$$

where i, j, k can each take integer values of 1, 2, 3 (representing the $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ coordinate axes respectively), the commas denote partial derivatives with respect to the initial configuration.

Integrating equation (1) over a time interval from $\boldsymbol{t}_0$ and $\boldsymbol{t}_1$ and applying integration by parts of the last term,

$$\int_{t_0}^{t_1}\int_{\Omega} \delta\varepsilon^T\sigma \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t} - \int_{t_0}^{t_1} \delta\boldsymbol{W}_{ext} \, \mathrm{d}\boldsymbol{t} + \int_{t_0}^{t_1}\int_{\Omega} \mu\delta\boldsymbol{q}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t} - \int_{t_0}^{t_1}\int_{\Omega} \rho\delta\dot{\boldsymbol{q}}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t}$$

$$+ \left[\int_{\Omega} \rho\delta\boldsymbol{q}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega\right]_{t_0}^{t_1} = 0, \tag{3}$$

which can be re-written as

$$-\delta\int_{t_0}^{t_1}\int_{\Omega} \boldsymbol{V}(\boldsymbol{q}) \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t} + \int_{t_0}^{t_1} \delta\boldsymbol{W}_{\mathrm{ext}} \, \mathrm{d}\boldsymbol{t} - \int_{t_0}^{t_1}\int_{\Omega} \mu\delta\boldsymbol{q}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t} + \delta\int_{t_0}^{t_1}\int_{\Omega} \boldsymbol{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \, \mathrm{d}\Omega \, \mathrm{d}\boldsymbol{t}$$

$$-\left[\int_{\Omega} \rho\delta\boldsymbol{q}^T\dot{\boldsymbol{q}} \, \mathrm{d}\Omega\right]_{t_0}^{t_1} = 0, \tag{4}$$

where $\boldsymbol{V}$ and $\boldsymbol{T}$ denote the stored stain energy density and the kinetic energy density respectively.

If $\delta\boldsymbol{q}(\boldsymbol{t}_0) = \delta\boldsymbol{q}(\boldsymbol{t}_1) = 0$ and $\mu = 0$, equation (4) is equivalent to Hamilton's principle for continuous media (e.g. [13]). Hamilton's principle can be modified to a multi-field form via Lagrangian multiplier method. In these mixed formulations, $\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{p}, \sigma, \epsilon$ ($\boldsymbol{p} = \rho\dot{\boldsymbol{q}}$) can be chosen as independent variables. These mixed formulations have been applied to tackle eigenvalue problems. However, it is found to be more efficient and flexible to tackle initial value problems by Hamilton's law using a two-field ($\boldsymbol{q}$ and $\boldsymbol{p}$) mixed form.

In order to cast the Hamilton's law into the Hamiltonian framework, the usual canonical transformation can be used (e.g. [13]). Let

$$\boldsymbol{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{V}(\boldsymbol{q}) - \boldsymbol{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{p}\dot{\boldsymbol{q}} - \boldsymbol{H}(\boldsymbol{q}, \boldsymbol{p}), \tag{5}$$

where $\boldsymbol{L}$ is the Lagrange density function; $\boldsymbol{p} = \rho\dot{\boldsymbol{q}}$ is the momentum density; $\boldsymbol{H}(\boldsymbol{q}, \boldsymbol{p}) = \boldsymbol{p}^2/(2\rho) + \boldsymbol{V}(\boldsymbol{q})$ is the Hamiltonian density. Substituting equation (5) into equation (4) yields the following canonical form (cf. [14])

$$\delta \int_{t_0}^{t_1} \int_{\Omega} \left(\boldsymbol{p}^T\dot{\boldsymbol{q}} - \frac{\boldsymbol{p}^2}{2\rho} - \boldsymbol{V}(\boldsymbol{q}) \right) \mathrm{d}\Omega \mathrm{d}\boldsymbol{t} + \int_{t_0}^{t_1} \delta \boldsymbol{W}_{ext}\, \mathrm{d}\boldsymbol{t} - \int_{t_0}^{t_1} \int_{\Omega} \left(\frac{\mu}{\rho} \right) \delta\boldsymbol{q}^T\boldsymbol{p}\, \mathrm{d}\Omega\, \mathrm{d}\boldsymbol{t}$$

$$- \left[\int_{\Omega} \delta\boldsymbol{q}^T\boldsymbol{p}\, \mathrm{d}\Omega \right]_{t_0}^{t_1} = 0, \tag{6}$$

where $\boldsymbol{q}, \boldsymbol{p}$ are independent variables. Equation (6) can also be derived from equation (4) directly by using the Lagrange multiplier method.

3. MIXED TIME FINITE ELEMENT FORMULATIONS

There are two main approaches to the implementation of space–time finite elements to solve structural dynamic problems. The first one is to discretize the spatial domain first. The original partial differential equations are transformed into a system of ordinary differential equations in time. The resultant system is then discretized by means of time finite elements. It is a decoupled method.

The second method is to discretize the spatial and temporal variables simultaneously. Usually, the interpolation functions are in a tensor product form based on the Lagrangian functions. This results in a regular space–time finite elements. Irregular space–time finite elements are in general not required for conventional structural dynamic problems. However, irregular space–time finite elements are useful in some special cases, such as impulsive loading problems [15], adaptive mesh refinement in contact dynamic problems [16], deployment problems as well as robotics problems [14].

Even though it is believed that decoupling the space and time domains may result in severe limitations to simulate the actual physical process accurately [17], if tensor product-type shape functions or rectangular elements are used, the second approach can be shown to be equivalent to the first approach [18]. In this paper, the space and time domains are decoupled. The formulation, however, can be extended to distorted elements in the space–time domain. This work will be reported in subsequent papers.

There are also two schemes to implement the time finite elements for initial value problems. One is to couple all the time finite element equations together and to solve the assembled equations for all time finite elements simultaneously. The procedure is similar to solving boundary value problems. The other scheme is to solve the equations of one element or one layer of elements at a time in a sequential manner. The procedure is similar to a time-marching scheme. The results at the final time become the initial value of the next time level. The drawback of the first approach is that it would involve too many simultaneous variables. For problems with very sparse matrices, efficient computational algorithms are available [14, 19, 20]. Besides, by treating the Lagrange multipliers as a natural by-product of the mixed method, a coupled mesh can be solved a layer at a time [21]. However, it is found that the time-marching schemes are still very commonly used in practice.

Assume equation (6) is implemented on a space–time finite element mesh organized into space-time "slabs" $\boldsymbol{S_n} = \Omega \times \boldsymbol{I_n}$, where Ω is the underlying spatial domain and $\boldsymbol{I_n} = [t_{n-1}, t_n]$ is a time interval corresponding to a partition of the time domain $\mathbf{I} = [0, T]$: $0 = t_0 < t_1 < \cdots < t_N = T$. Complying with the conventional finite element discretization technique, the variables and variations are expressed as

$$\boldsymbol{q} = \boldsymbol{N}\hat{\boldsymbol{q}} = \boldsymbol{N\Phi u}, \qquad \dot{\boldsymbol{q}} = \boldsymbol{N}\dot{\boldsymbol{\Phi}}\boldsymbol{u}, \qquad \boldsymbol{p} = \boldsymbol{N}\hat{\boldsymbol{p}} = \boldsymbol{N\Phi v},$$
$$\delta\boldsymbol{q} = \boldsymbol{N}\delta\hat{\boldsymbol{q}} = \boldsymbol{N}\tilde{\boldsymbol{\Phi}}\delta\boldsymbol{u}, \qquad \delta\boldsymbol{p} = \boldsymbol{N}\delta\hat{\boldsymbol{p}} = \boldsymbol{N}\tilde{\boldsymbol{\Phi}}\delta\boldsymbol{v}, \tag{7}$$

where the nodal vectors are

$$\hat{\boldsymbol{q}}^T = \{\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_{N_s}\}, \qquad \hat{\boldsymbol{p}}^T = \{\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_{N_s}\}, \qquad \boldsymbol{u}^T = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{N_s}\},$$
$$\boldsymbol{v}^T = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{N_s}\},$$
$$\boldsymbol{u}_i^T = \{u_{i0}, u_{i1}, \ldots, u_{ik}\}, \qquad \boldsymbol{v}_i^T = \{v_{i0}, v_{il}, \ldots, v_{ik}\}, \qquad \delta\boldsymbol{u}^T = \{\delta\boldsymbol{u}_1, \delta\boldsymbol{u}_2, \ldots, \delta\boldsymbol{u}_{N_s}\},$$
$$\delta\boldsymbol{v}^T = \{\delta\boldsymbol{v}_1, \delta\boldsymbol{v}_2, \ldots, \delta\boldsymbol{v}_{N_s}\}, \qquad \delta\boldsymbol{u}_i^T = \{\delta u_{i0}, \delta u_{i1}, \ldots, \delta u_{il}\},$$
$$\delta\boldsymbol{v}_i^T = \{\delta v_{i0}, \delta v_{i1}, \ldots, \delta v_{il}\}; \tag{8}$$

$\boldsymbol{N}$ is the shape function matrix in the spatial dimensions with nodes $\hat{p}_i$, $\hat{q}_i$ $(i = 1, 2, \ldots, N_s)$; Φ is the shape function matrix in the time dimension with nodes u_{ij}, v_{ij} $(i = 1, 2, \ldots, N_s$; $j = 0, 1, \ldots, k)$; $\tilde{\Phi}$ is the shape function matrix in the time dimension with nodes δu_{ij}, δv_{ij} $(i = 1, 2, \ldots, N_s$; $j = 0, 1, \ldots, l)$. It is noted that in the present formulation both $\boldsymbol{p}$ and $\boldsymbol{q}$ are treated as independent variables. Hence the same shape functions can be used for $\boldsymbol{p}$ and $\boldsymbol{q}$ to maintain symmetry.

Different choices of $\tilde{\Phi}$ and Φ could yield different algorithms, ranging from non-dissipative to dissipative. This will be presented in the next few sections. In the following, non-linear problems are discussed.

3.1. NONLINEAR PROBLEMS

Since

$$\delta\varepsilon^T = \delta\boldsymbol{q}^T\left(\frac{\partial\varepsilon}{\partial\boldsymbol{q}}\right)^T, \qquad \delta\boldsymbol{W}_{ext} = \delta\boldsymbol{q}^T\boldsymbol{L}, \qquad \boldsymbol{L} = \boldsymbol{N}\Phi\boldsymbol{F}, \tag{9}$$

where $\boldsymbol{L}$ is the load which can be discretized via $\boldsymbol{N}$ and Φ; $\boldsymbol{F}$ is the corresponding nodal load vector. Substituting equations (7)–(9) into equation (6) and considering the time interval $\boldsymbol{I_n}$, one has

$$\int_{t_{n-1}}^{t_n}\int_{\Omega}\left\{\delta\boldsymbol{v}^T\tilde{\Phi}^T\boldsymbol{N}^T\boldsymbol{N}\dot{\Phi}\boldsymbol{u} + \delta\boldsymbol{u}^T\dot{\tilde{\Phi}}^T\boldsymbol{N}^T\boldsymbol{N}\Phi\boldsymbol{v} - \delta\boldsymbol{q}^T\left(\frac{\partial\varepsilon}{\partial\boldsymbol{q}}\right)^T\sigma(\varepsilon) + \delta\boldsymbol{u}^T\tilde{\Phi}^T\boldsymbol{N}^T\boldsymbol{N}\Phi\boldsymbol{F}\right.$$

$$-\frac{\mu}{\rho}\,\delta\boldsymbol{u}^T\tilde{\boldsymbol{\Phi}}^T\boldsymbol{N}^T\boldsymbol{N}\boldsymbol{\Phi}\boldsymbol{v}-\frac{1}{\rho}\,\delta\boldsymbol{v}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{N}^T\boldsymbol{N}\boldsymbol{\Phi}\boldsymbol{v}\Big\}\,\mathrm{d}\Omega\,\mathrm{d}\boldsymbol{t}+\left[\int_{\Omega}\delta\boldsymbol{u}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{N}^T\boldsymbol{N}\boldsymbol{\Phi}\boldsymbol{v}\,\mathrm{d}\Omega\right]_{t_{n-1}}^{t_n}=0 \tag{10}$$

where the non-linearity of materials and geometry enter through

$$\boldsymbol{R}=\int_{t_{n-1}}^{t_n}\int_{\Omega}\delta\boldsymbol{q}^T\left(\frac{\partial\varepsilon}{\partial\boldsymbol{q}}\right)^T\sigma\,\mathrm{d}\Omega\,\mathrm{d}\boldsymbol{t}. \tag{11}$$

For convenience, re-cast equation (11) as

$$\boldsymbol{R}=\int_{t_{n-1}}^{t_n}\delta\hat{\boldsymbol{q}}^T\hat{\boldsymbol{R}}(\hat{\boldsymbol{q}})\,\mathrm{d}\boldsymbol{t}, \tag{12}$$

which is a nonlinear function $\hat{\boldsymbol{q}}$. For nonlinear "rate-type" viscoelastic materials, $\boldsymbol{R}$ may also be a function of momentum (velocity).

Because $\hat{\boldsymbol{R}}$ is a nonlinear function of the current displacements, the conventional equilibrium iteration procedure should be used to solve the nonlinear equations. For this purpose two approaches are usually used: the tangent stiffness method and the pseudo-load method [22, 23]. The tangent stiffness method is considered to be more stable and is widely used. The pseudo-load method has the advantage that the effective stiffness matrix is formulated only once for constant time step and can be used repeatedly for the entire analysis. This makes the method extremely efficient in computation. In the pseudo-load method, $\hat{\boldsymbol{R}}$ can be divided into linear and nonlinear parts as

$$\hat{\boldsymbol{R}}(\hat{\boldsymbol{q}})=\boldsymbol{K}\hat{\boldsymbol{q}}-\bar{\boldsymbol{R}}(\hat{\boldsymbol{q}}), \tag{13}$$

where $\boldsymbol{K}$ can be the linear part of the stiffness or the average stiffness over the range of expected displacement. $\bar{\boldsymbol{R}}(\hat{\boldsymbol{q}})$ denotes the pseudo-load, which will be predicted and then corrected in successive iterations so as to maintain dynamic equilibrium. Let

$$\boldsymbol{D}=\int_{\Omega}\boldsymbol{N}^T\boldsymbol{N}\,\mathrm{d}\Omega,\qquad \bar{\boldsymbol{M}}=\int_{\Omega}\frac{1}{\rho}\boldsymbol{N}^T\boldsymbol{N}\,\mathrm{d}\Omega,\qquad \bar{\boldsymbol{C}}=\int_{\Omega}\frac{\mu}{\rho}\boldsymbol{N}^T\boldsymbol{N}\,\mathrm{d}\Omega,\qquad \bar{\boldsymbol{R}}(\hat{\boldsymbol{q}})=\boldsymbol{\Phi}\bar{\boldsymbol{F}}. \tag{14}$$

Substituting equations (12)–(14) into equation (10), one has

$$\begin{bmatrix}\displaystyle\int_{t_{n-1}}^{t_n}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{D}\dot{\boldsymbol{\Phi}}\,\mathrm{d}\boldsymbol{t} & \displaystyle-\int_{t_{n-1}}^{t_n}\tilde{\boldsymbol{\Phi}}^T\bar{\boldsymbol{M}}\boldsymbol{\Phi}\,\mathrm{d}\boldsymbol{t}\\ \displaystyle-\int_{t_{n-1}}^{t_n}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{K}\boldsymbol{\Phi}\,\mathrm{d}\boldsymbol{t} & \displaystyle\int_{t_{n-1}}^{t_n}(\dot{\tilde{\boldsymbol{\Phi}}}^T\boldsymbol{D}\boldsymbol{\Phi}-\tilde{\boldsymbol{\Phi}}^T\bar{\boldsymbol{C}}\boldsymbol{\Phi})\,\mathrm{d}\boldsymbol{t}\end{bmatrix}\begin{Bmatrix}\boldsymbol{u}\\ \mathbf{v}\end{Bmatrix}$$

$$=\begin{Bmatrix}0\\ \displaystyle\int_{t_{n-1}}^{t_n}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{\Phi}\bar{\boldsymbol{F}}\,\mathrm{d}\boldsymbol{t}\end{Bmatrix}+\begin{Bmatrix}0\\ \displaystyle\int_{t_{n-1}}^{t_n}\tilde{\boldsymbol{\Phi}}^T\boldsymbol{D}\boldsymbol{\Phi}\boldsymbol{F}\,\mathrm{d}\boldsymbol{t}\end{Bmatrix}+\begin{Bmatrix}0\\ \tilde{\boldsymbol{\Phi}}^T\boldsymbol{D}\boldsymbol{\Phi}\boldsymbol{v}\end{Bmatrix}\Bigg|_{t_{n-1}}^{t_n}. \tag{15}$$

Equation (15) is the governing equation for time marching and can be solved iteratively.

If the structural system is semi-discretized by the conventional finite element method, and $\hat{\boldsymbol{p}} = \boldsymbol{M}\dot{\boldsymbol{q}}$, it can be verified that equation (15) can be written as

$$\begin{bmatrix} -\displaystyle\int_{t_{n-1}}^{t_n} \dot{\tilde{\Phi}}^T \boldsymbol{M} \Phi \, \mathrm{d}\boldsymbol{t} & -\displaystyle\int_{t_{n-1}}^{t_n} \tilde{\Phi}^T \boldsymbol{M} \Phi \, \mathrm{d}\boldsymbol{t} \\ -\displaystyle\int_{t_{n-1}}^{t_n} \tilde{\Phi}^T \boldsymbol{K} \Phi \, \mathrm{d}\boldsymbol{t} & \displaystyle\int_{t_{n-1}}^{t_n} (\dot{\tilde{\Phi}}^T \boldsymbol{M} \Phi - \tilde{\Phi}^T \overline{\boldsymbol{C}} \Phi) \, \mathrm{d}\boldsymbol{t} \end{bmatrix} \begin{Bmatrix} \boldsymbol{u} \\ \mathbf{v} \end{Bmatrix}$$

$$= \begin{Bmatrix} 0 \\ \displaystyle\int_{t_{n-1}}^{t_n} \tilde{\Phi}^T \Phi \overline{\boldsymbol{F}} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \displaystyle\int_{t_{n-1}}^{t_n} \tilde{\Phi}^T \boldsymbol{D} \Phi \boldsymbol{F} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \tilde{\Phi}^T \boldsymbol{D} \Phi \dot{\boldsymbol{u}} \end{Bmatrix} \Bigg|_{t_{n-1}}^{t_n}, \tag{16}$$

where $\boldsymbol{M}$, $\boldsymbol{K}$, $\boldsymbol{C}$ are the usual mass, stiffness and damping matrices respectively. Hence the discretization in the space domain of the proposed formulation is consistent with the conventional semi-discretized formulations.

4. DEVELOPMENT OF TIME FINITE ELEMENTS

Based on equation (15), three kinds of time finite elements are developed.

4.1. NON-DISSIPATIVE TIME FINITE ELEMENTS

For convenience, consider a single-degree-of-freedom system. The results are known to be valid for the original multi-degree-of-freedom system. For the unforced and undamped case, equation (16) yields the following equations

$$\begin{bmatrix} \displaystyle\int_{t_{n-1}}^{t_n} \tilde{\Phi} \dot{\Phi}^T \, \mathrm{d}\boldsymbol{t} & -\displaystyle\int_{t_{n-1}}^{t_n} m^{-1} \tilde{\Phi} \Phi^T \, \mathrm{d}\boldsymbol{t} \\ -\displaystyle\int_{t_{n-1}}^{t_n} k \tilde{\Phi} \Phi^T \, \mathrm{d}\boldsymbol{t} & \displaystyle\int_{t_{n-1}}^{t_n} \dot{\Phi} \Phi^T \, \mathrm{d}\boldsymbol{t} \end{bmatrix} \begin{Bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{u}_0 \\ \boldsymbol{v}_0 \end{Bmatrix}, \tag{17}$$

where $\boldsymbol{u}_0^T = \{0, \ldots, 0\}$, $\boldsymbol{v}_0^T = \{-v_0, 0, \ldots, v_k\}$. If Φ and $\tilde{\Phi}$ are the kth and $(k-1)$th order Lagrange polynomials respectively, it can be shown that equation (17) yields a set of unconditionally stable, non-dissipative algorithms [24]. Some simple cases are given as below:

Second-order-accurate time finite element (*ND*2)

If the test function is constant $\tilde{\Phi}^h = \{1\}$ and the trial functions are linear $\Phi^h = \{1 - \tau, \tau\}$, where $\tau = (t - t_{n-1})/(t_n - t_{n-1}) \in [0, 1]$, the derived algorithm is second-order-accurate.

Fourth-order-accurate time finite element (*ND*4)

If the test functions are linear $\tilde{\Phi}^h = \{1 - \tau, \tau\}$ and the trial functions are parabolic $\Phi^h = \{(1 - 2\tau)(1 - \tau), 4\tau(1 - \tau), \tau(2\tau - 1)\}$, the derived algorithm is fourth-order-accurate.

Sixth-order-accurate time finite element ($ND6$)

If the test functions are parabolic $\tilde{\Phi}^h = \{(1-2\tau)(1-\tau), 4\tau(1-\tau), \tau(2\tau-1)\}$ and the trial functions are cubic $\Phi^h = \{9(1/3-\tau)(2/3\tau)(1-\tau)/2,\ 27\tau(1-\tau)(2/3-\tau),\ 27\tau(1/3-\tau)(1-\tau)/2,\ 9\tau(1/3-\tau)(2/3-\tau)/2\}$, the derived algorithm is sixth-order-accurate.

It is noted that the above algorithms are equivalent to the non-dissipative algorithms given by Borri *et al*. [7, 8], in which either reduced integration of bi-discontinuous form is used. In the actual implementation, the original space–time formulation equation (15) can be used via the present time interpolation, without using equations (16) and (17).

4.2. DISSIPATION-CONTROLLABLE TIME FINITE ELEMENTS

Integrating equation (6) by parts, one has

$$\int_{t_{n+1}}^{t_n}\int_{\Omega}\left(\boldsymbol{p}^T\delta\dot{\boldsymbol{q}} - \delta\dot{\boldsymbol{p}}^T\boldsymbol{q} - \frac{\boldsymbol{p}\delta\boldsymbol{p}}{\rho} - \delta\boldsymbol{V}(\boldsymbol{q})\right)\mathrm{d}\Omega\,\mathrm{d}\boldsymbol{t} + \int_{t_{n-1}}^{t_n}\delta\boldsymbol{W}_{ext}\,\mathrm{d}\boldsymbol{t}$$
$$- \int_{t_{n-1}}^{t_n}\int_{\Omega}\left(\frac{\mu}{\rho}\right)\delta\boldsymbol{q}^T\boldsymbol{p}\,\mathrm{d}\Omega\,\mathrm{d}\boldsymbol{t} + \left[\int_{\Omega}(\delta\boldsymbol{p}^T\boldsymbol{q} - \delta\boldsymbol{q}^T\boldsymbol{p})\,\mathrm{d}\Omega\right]_{t_{n-1}}^{t_n} = 0. \quad (18)$$

It is noted that in equation (18) the variables $\boldsymbol{p}$ and $\boldsymbol{q}$ have no continuity requirements ($\mathbf{C}^{-1}$ continuity); whereas the variations $\delta\boldsymbol{p}$ and $\delta\boldsymbol{q}$ have to be continuous at the boundaries and piece-wise differentiable in the time domain ($\boldsymbol{C}^0$ continuity). Assume that Φ admits discontinuous piece-wise ($\boldsymbol{C}^{-1}$ continuity) Lagrange polynomials which interpolate k nodes while $\tilde{\Phi}$ admits continuous ($\boldsymbol{C}^0$ continuity) Lagrange polynomials which interpolate $k-1$ nodes. Equation (18) becomes:

$$\begin{bmatrix} \int_{t_{n-1}}^{t_n}\dot{\tilde{\Phi}}\Phi^T\,\mathrm{d}\boldsymbol{t} & \int_{t_{n-1}}^{t_n} m^{-1}\tilde{\Phi}\Phi^T\,\mathrm{d}\boldsymbol{t} \\ -\int_{t_{n-1}}^{t_n} k\tilde{\Phi}\Phi^T\,\mathrm{d}\boldsymbol{t} & \int_{t_{n-1}}^{t_n} m^{-1}\dot{\tilde{\Phi}}\Phi^T\,\mathrm{d}\boldsymbol{t} \end{bmatrix}\begin{Bmatrix}\boldsymbol{u}\\ \boldsymbol{v}\end{Bmatrix} = \begin{Bmatrix}\boldsymbol{u}_0\\ \boldsymbol{v}_0\end{Bmatrix}, \quad (19)$$

where $\boldsymbol{u}_0^T = \{-u_0, 0, \ldots, u_k\}$, $\boldsymbol{v}_0^T = \{-v_0, 0, \ldots, v_k\}$. By choosing the interpolation functions properly, it can be shown that equation (19) could yield unconditionally stable, dissipative algorithms [24].

First/second-order-accurate time finite element ($DC12$)

Assume that the test function is constant $\tilde{\Phi}^h = \{1\}$ and the trial functions are $\{1, f_1(t)\}$ where $f_1(\tau) = 1 - \tau/\sigma$ if $0 \leqslant \tau \leqslant \sigma$ and 0 if $1 \geqslant \tau > \sigma$. It can be seen that the trial functions vary linearly from $\tau = 0$ to $\tau = \sigma$ and then maintain a constant value when $1 \geqslant \tau > \sigma$. In other words, the trial functions are continuous and not differentiable only at $\tau = \sigma$. It can be shown that the dissipative algorithms with first-order-accurate to second-order-accurate can be obtained from equation (19). Various numerical dissipation can be obtained by adjusting σ.

Consider the limiting case when $\sigma \to 0$. The trial functions would be constant within the time interval but the nodal value may have a different value, i.e. $q(0)$ and $q(0^+)$ may be different. Equation (19) yields an algorithm equivalent to the first-order-accurate asymptotic annihilating algorithm [11, 12, 25]. When $\sigma = 1$, equation (19) leads to a second-order-accurate non-dissipative algorithm [26–28]. Other dissipative algorithms can be obtained by varying σ.

It may seem that the "first/second-order-accurate elements" are not appropriate because the trial functions are not differentiable at $\tau = \sigma$. However, they are admissible since the derivatives of the trial functions are not required in the basic formulation in equation (19).

Third/fourth-order-accurate time finite element (*DC*34)

Assume that the test functions are linear $\tilde{\Phi}^h = \{1 - \tau, \tau\}$ and the trial functions are given by $\{1, f_1(\tau), f_2(\tau)\}$ where $f_1(\tau)$ is the function defined previously and $f_2(\tau) = 0$ if $0 \leqslant \tau \leqslant \sigma$ and $(\tau - \sigma)/(1 - \sigma)$ if $1 \geqslant \tau > \sigma$. It can be seen that the trial functions vary linearly from 0 to σ and σ to 1, with a possible kink at $\tau = \sigma$. In other words, the trial functions are continuous and not differentiable only at $\tau = \sigma$. It can be shown that dissipative algorithms with third and fourth-order-accurate can be obtained from equation (19).

Consider the limiting case when $\sigma \to 0$. Equation (19) yields an algorithm equivalent to the third-order-accurate asymptotic annihilating algorithm [11, 12, 25]. When $\sigma = 0{\cdot}5$, equation (19) leads to a fourth-order-accurate non-dissipative algorithm [26–28].

4.3. EXTRAPOLATED TIME FINITE ELEMENTS

The extrapolated time finite elements can be obtained by using equation (7) and assuming $\Phi = \tilde{\Phi}$. Equation (15) yields

$$\begin{bmatrix} \int_{t_{n-1}}^{t_n} \Phi^{\mathrm{T}} \boldsymbol{D} \dot{\Phi} \, \mathrm{d}\boldsymbol{t} & \int_{t_{n-1}}^{t_n} \Phi^{T} \overline{\boldsymbol{M}} \Phi \, \mathrm{d}\boldsymbol{t} \\ -\int_{t_{n-1}}^{t_n} \Phi^{T} \mathbf{K} \Phi \, \mathrm{d}\boldsymbol{t} & \int_{t_{n-1}}^{t_n} (\dot{\Phi}^{T} \boldsymbol{D} \Phi - \Phi^{T} \overline{\boldsymbol{C}} \Phi) \, \mathrm{d}\boldsymbol{t} \end{bmatrix} \begin{Bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{Bmatrix}$$

$$= \begin{Bmatrix} 0 \\ \int_{t_{n-1}}^{t_n} \Phi^{T} \Phi \overline{\boldsymbol{F}} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \int_{t_{n-1}}^{t_n} \Phi^{T} \boldsymbol{D} \Phi \boldsymbol{F} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \Phi^{T} \boldsymbol{D} \Phi \boldsymbol{v} \end{Bmatrix} \Bigg|_{t_{n-1}}^{t_n}. \tag{20}$$

If the structural system is spatially semi-discretized according to the conventional finite element method and $\hat{\boldsymbol{p}} = \boldsymbol{M} \dot{\hat{\boldsymbol{q}}}$, equation (20) leads to

$$\begin{bmatrix} -\int_{t_{n-1}}^{t_n} \dot{\Phi}^{T} \boldsymbol{M} \Phi \, \mathrm{d}\boldsymbol{t} & -\int_{t_{n-1}}^{t_n} \Phi^{T} \boldsymbol{M} \Phi \, \mathrm{d}\boldsymbol{t} \\ -\int_{t_{n-1}}^{t_n} \Phi^{T} \boldsymbol{K} \Phi \, \mathrm{d}\boldsymbol{t} & \int_{t_{n-1}}^{t_n} (\dot{\Phi}^{T} \boldsymbol{M} \Phi - \Phi^{T} \boldsymbol{C} \Phi) \, \mathrm{d}\boldsymbol{t} \end{bmatrix} \begin{Bmatrix} \boldsymbol{u} \\ \dot{\boldsymbol{u}} \end{Bmatrix}$$

$$= \begin{Bmatrix} 0 \\ \int_{t_{n-1}}^{t_n} \Phi^{T} \Phi \overline{\boldsymbol{F}} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ -\int_{t_{n-1}}^{t_n} \Phi^{T} \boldsymbol{D} \Phi \boldsymbol{F} \, \mathrm{d}\boldsymbol{t} \end{Bmatrix} + \begin{Bmatrix} 0 \\ \Phi^{T} \boldsymbol{D} \Phi \dot{\boldsymbol{u}} \end{Bmatrix} \Bigg|_{t_{n-1}}^{t_n} \tag{21}$$

where $\mathbf{M}$, $\mathbf{K}$, $\mathbf{C}$ are the conventional mass, stiffness, damping matrices respectively. Both equations (20) and (21) can be extrapolated in the time domain. For simplicity, assume that both the test and trial functions are linear $\Phi = \tilde{\Phi} = \{1-\tau, \tau\}$. Based on equation (20) or (21) with time step Δt, the initial value of the (n-1)th slab $\boldsymbol{S}_{n-1}$, ${}^{n-1}_{0}\boldsymbol{u}^T = \{\boldsymbol{u}_{10}, \boldsymbol{u}_{20}, \ldots, \boldsymbol{u}_{N_s0}\}$, ${}^{n-1}_{0}\boldsymbol{v}^T = \{\boldsymbol{v}_{10}, \boldsymbol{v}_{20}, \ldots, \boldsymbol{v}_{N_s0}\}$ can be used to obtain the initial value of the nth slab $\boldsymbol{S}_n$. In using the extrapolated time finite elements, the initial value of the nth slab $\boldsymbol{S}_n$ can be obtained by evaulating the responses at $\beta_i \Delta t$ from the $(n-1)$th slab $\boldsymbol{S}_{n-1}$ and summing them up using various weighting factors α_i. The extrapolation parameters α_i and β_i are chosen so that the resultant algorithms are higher order accurate and stable. The extrapolation parameters are given as follows [24, 29].

Second-order-accurate parameters (EX2)

$$\alpha_0 = -1/3, \qquad \beta_0 = 0, \qquad \alpha_1 = 4/3, \qquad \beta_1 = 3/4 \tag{22}$$

Third-order-accurate parameters (EX3)

$$\alpha_0 = -\frac{1}{6}\frac{4\beta_2^2 - 6\beta_2 + 3}{\beta_2(2\beta_2 - 1)} \qquad \beta_0 = 0 \tag{23a}$$

$$\alpha_1 = \frac{1}{3}\frac{64\beta_2^3 - 144\beta_2^2 + 108\beta_2 - 27}{(2\beta_2 - 1)(8\beta_2^2 - 12\beta_2 + 3)} \qquad \beta_1 = \frac{3}{2}\frac{2\beta_2 - 1}{4\beta_2 - 3} \tag{23b}$$

$$\alpha_2 = -\frac{3}{2\beta_2(8\beta_2^2 - 12\beta_2 + 3)} \qquad \beta_2 > 3/4 \tag{23c}$$

Fourth-order-accurate parameters (EX4)

$$\alpha_0 = -\frac{1}{12}\frac{64\beta_2^2\beta_3^2 - (96\beta_2\beta_3 + 36)(\beta_2 + \beta_3) + 48(\beta_2^2 + \beta_3^2) + 108\beta_2\beta_3 + 9}{\beta_2\beta_3(16\beta_2\beta_3 - 8\beta_2 - 8\beta_3 + 3)} \tag{24a}$$

$$\alpha_1 = \frac{64}{3}\frac{(8\beta_2\beta_3 - 6\beta_2 - 6\beta_3 + 3)^4}{(16\beta_2\beta_3 - 8\beta_2 - 8\beta_3 + 3)G(\beta_2, \beta_3)G(\beta_3, \beta_2)} \tag{24b}$$

$$\alpha_2 = -\frac{3}{4}\frac{(16\beta_3^2 - 12\beta_3 + 3)}{\beta_2(\beta_3 - \beta_2)G(\beta_3, \beta_2)} \tag{24c}$$

$$\alpha_3 = -\frac{3}{4}\frac{(16\beta_2^2 - 12\beta_2 + 3)}{\beta_3(\beta_2 - \beta_3)G(\beta_2, \beta_3)} \tag{24d}$$

$$\beta_0 = 0 \tag{24e}$$

$$\beta_1 = \frac{3}{8}\frac{16\beta_2\beta_3 - 8\beta_2 - 8\beta_3 + 3}{8\beta_2\beta_3 - 6\beta_2 - 6\beta_3 + 3} \tag{24f}$$

where $G(x, y) = -96xy + 64y^2x - 48y^2 + 48y + 24x - 9$. The parameters β_2 and β_3 can be chosen from Figure 1.

The extrapolation algorithm can be summarized as follows [29]: let

$$\boldsymbol{X}_{n-1} = \begin{Bmatrix} {}^{n-1}_{0}\boldsymbol{u} \\ {}^{n-1}_{0}\boldsymbol{v} \end{Bmatrix}, \qquad \boldsymbol{X}_n = \begin{Bmatrix} {}^{n}_{0}\boldsymbol{u} \\ {}^{n}_{0}\boldsymbol{v} \end{Bmatrix}.$$

$\boldsymbol{X}_n$ is the result to be obtained from $\boldsymbol{X}_{n-1}$ via equation (21) with Δt. Suppose $\boldsymbol{X}_{ni}$ is the result obtained from $\boldsymbol{X}_{n-1}$ via equation (21) with $\Delta \boldsymbol{t}_i = \beta_i \Delta \boldsymbol{t}(i = 0, 1, \ldots, s-1)$. Then $\boldsymbol{X}_n$ is given by

$$X_n = \sum_{i=0}^{s-1} \alpha_i X_{ni}. \tag{25}$$

5. STABILITY AND ACCURACY CHARACTERISTICS

Equations (15) and (16) are recursive. The characteristics of the algorithms can be assessed by studying the numerical amplification matrices. In general, the stability of mixed finite elements in space and time cannot be proved except for very special cases [30]. In the following, only the stability and accuracy characteristics for linear problems are considered.

Furthermore, since a multi-degree-of-freedom system can be decomposed into a set of single-degree-of-freedom systems, it has been rigorously established that the entire coupled system reduces to the consideration of the individual modal equation (e.g. [31]). Thus the analysis can be done on a single-degree-of-freedom system but the conclusions are valid for the original multi-degree-of-freedom system.

The characteristics of an algorithm can be measured by accuracy and convergence.

5.1. ACCURACY

Numerical approximations in the time domain normally result in period error and amplitude error. It is a common practice to separate the errors of the numerical solution into amplitude error and phase error. The former is specified as dissipation and the later is specified as dispersion. The measures of numerical dissipation and dispersion are normally in terms of algorithmic damping ratio and relative period error respectively. Based on a scalar model, the error information can be obtained by comparing the numerical solutions to the analytical solutions.

5.2. CONVERGENCE

The convergence of a numerical formulation requires consistency and stability. Consistency can be determined from the truncation error. An algorithm is consistent if the

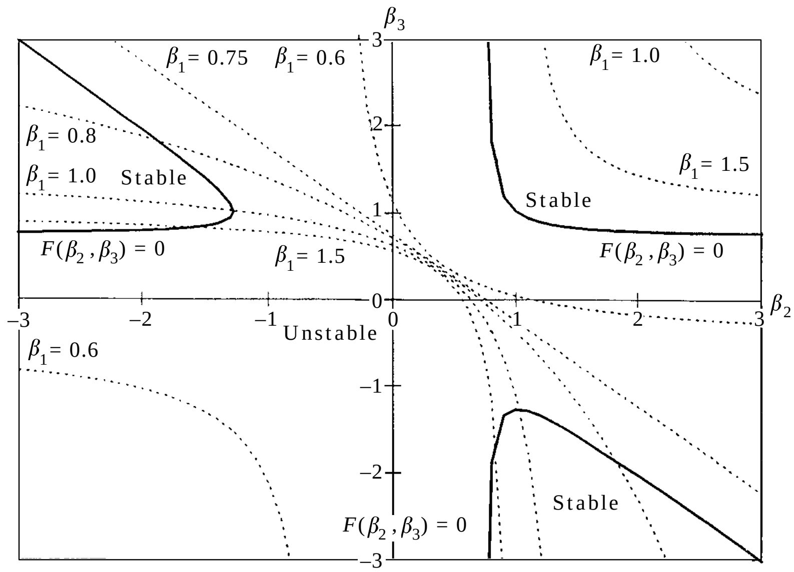

Figure 1. Stable and unstable regions for the fourth order accurate formulation.

order of accuracy is greater than zero. The stability can be determined from the magnitude of the eigenvalues of the numerical amplification matrix. The stability condition is that the modulus of all the eigenvalues should be less than or equal to unity. Moreover, an algorithm with no time step restriction imposed by stability is called unconditionally stable.

Based on the above observation, it is more convenient and sufficient for the purpose of investigating the characteristics of the proposed time finite elements to consider a single-degree-of-freedom system with mass m and spring stiffness k only. The Hamiltonian of the system can be written as $\boldsymbol{H} = p^2/(2m) + kq^2/2$, where q and $p = m\dot{q}$ are the displacement and momentum of the system respectively. The resultant recursive relation are

$$\boldsymbol{X}_n = \boldsymbol{A}\boldsymbol{X}_{n-1} \tag{26}$$

where

$$\boldsymbol{X}_{n-1} = \begin{Bmatrix} {}^{n-1}_{\;\;\,0}\boldsymbol{u} \\ {}^{n-1}_{\;\;\,0}\boldsymbol{v} \end{Bmatrix}, \qquad \boldsymbol{X}_n = \begin{Bmatrix} {}^{n}_{0}\boldsymbol{u} \\ {}^{n}_{0}\boldsymbol{v} \end{Bmatrix},$$

$\boldsymbol{A}$ is the numerical amplification matrix. The corresponding numerical amplification matrices for various algorithms are:

$ND2$

$$\boldsymbol{A}_{ND2} = \begin{bmatrix} \dfrac{4-(\omega\Delta t)^2}{4+(\omega\Delta t)^2} & \dfrac{4\omega\Delta t}{m\omega(4+(\omega\Delta t)^2)} \\ \dfrac{-4m\omega(\omega\Delta t)}{4+(\omega\Delta t)^2} & \dfrac{4-(\omega\Delta t)^2}{4+(\omega\Delta t)^2} \end{bmatrix} \tag{27}$$

$ND4$

$$\boldsymbol{A}_{ND4} = \begin{bmatrix} \dfrac{144-60(\omega\Delta t)^2+(\omega\Delta t)^4}{144+12(\omega\Delta t)^2+(\omega\Delta t)^4} & \dfrac{144\omega\Delta t-12(\omega\Delta t)^3}{m\omega(144+12(\omega\Delta t)^2+(\omega\Delta t)^4)} \\ \dfrac{m\omega(12(\omega\Delta t)^3-144\omega\Delta t)}{144+12(\omega\Delta t)^2+(\omega\Delta t)^4} & \dfrac{144-60(\omega\Delta t)^2+(\omega\Delta t)^4}{144+12(\omega\Delta t)^2+(\omega\Delta t)^4} \end{bmatrix}, \tag{28}$$

$ND6$

$$\boldsymbol{A}_{ND6} = \begin{bmatrix} \dfrac{14\,400-6480(\omega\Delta t)^2+264(\omega\Delta t)^4-(\omega\Delta t)^6}{14\,400+720(\omega\Delta t)^2+24(\omega\Delta t)^4+(\omega\Delta t)^6} & \dfrac{14\,400\omega\Delta t-1680(\omega\Delta t)^3+264(\omega\Delta t)^5}{m\omega(14\,400+720(\omega\Delta t)^2+24(\omega\Delta t)^4+(\omega\Delta t)^6)} \\ \dfrac{-m\omega(14\,400(\omega\Delta t)-1680(\omega\Delta t)^3+264(\omega\Delta t)^5)}{14\,400+720(\omega\Delta t)^2+24(\omega\Delta t)^4+(\omega\Delta t)^6} & \dfrac{14\,400-6480(\omega\Delta t)^2+264(\omega\Delta t)^4-(\omega\Delta t)^6}{14\,400+720(\omega\Delta t)^2+24(\omega\Delta t)^4+(\omega\Delta t)^6} \end{bmatrix} \tag{29}$$

$DC12$

$$\boldsymbol{A}_{DC12} = \begin{bmatrix} \dfrac{4-(2-\sigma)\sigma(\omega\Delta t)^2}{4+(2-\sigma)^2(\omega\Delta t)^2} & \dfrac{4(\omega\Delta t)}{m\omega[4+(2-\sigma)^2(\omega\Delta t)^2]} \\ \dfrac{-4m\omega(\omega\Delta t)}{4+(2-\sigma)^2(\omega\Delta t)^2} & \dfrac{4-(2-\sigma)\sigma(\omega\Delta t)^2}{4+(2-\sigma)^2(\omega\Delta t)^2} \end{bmatrix}, \tag{30}$$

*DC*34

$$\boldsymbol{A}_{DC34} = \begin{bmatrix} \dfrac{36 + (4\sigma^2 - 4\sigma - 14)\,(\omega\Delta t)^2 + (1-\sigma)\sigma(\omega\Delta t)^4}{4 + (2-\sigma)^2(\omega\Delta t)^2} & \dfrac{[36 + (4\sigma^2 - 4\sigma - 2)\,(\omega\Delta t)^2]\,(\omega\Delta t)}{m\omega[4 + (2-\sigma)^2(\omega\Delta t)^2]} \\ \dfrac{-[36 + (4\sigma^2 - 4\sigma - 2)\rho^2]m\omega(\omega\Delta t)}{4 + (2-\sigma)^2(\omega\Delta t)^2} & \dfrac{36 + (4\sigma^2 - 4\sigma - 14)\,(\omega\Delta t)^2 + (1-\sigma)\sigma(\omega\Delta t)^4}{4 + (2-\sigma)^2(\omega\Delta t)^2} \end{bmatrix}. \tag{31}$$

EXs ($s = 2, 3, 4$)

$$\boldsymbol{A}_{EXs}\,(\omega\boldsymbol{\Delta t}) = \sum_{i=0}^{s-1} \alpha_i\, \mathbf{A}_1\,(\beta_i\,\omega\boldsymbol{\Delta t}) \tag{32}$$

where α_i and β_i are given by equations (22)–(24) with

$$\boldsymbol{A}_1 = \frac{1}{9 + 4\omega^2\Delta t^2}\begin{bmatrix} 9 - 2\omega^2\Delta t^2 & 9\Delta t \\ -9\omega^2\Delta t & 9 - 2\omega^2\Delta t^2 \end{bmatrix}.$$

On the other hand, it can be verified that the analytical solution of the system is

$$\boldsymbol{X}(\boldsymbol{t}_n) = \boldsymbol{A}_a\,\boldsymbol{X}(\boldsymbol{t}_{n-1}), \tag{33}$$

where

$$\boldsymbol{A}_a = \begin{bmatrix} \cos\omega\Delta t & \dfrac{1}{m\omega}\sin\omega\Delta t \\ -m\omega\sin\omega\Delta t & \cos\omega\Delta t \end{bmatrix}, \tag{34}$$

$\boldsymbol{X} = \begin{Bmatrix} q \\ p \end{Bmatrix}$ and $\omega = \sqrt{k/m}$ is the natural frequency of the system.

Spectral radius

The spectral radiusof $\boldsymbol{A}$ is defined as the largest magnitude of the eigenvalues of $\boldsymbol{A}$, i.e.

$$\bar{\rho}(\boldsymbol{A}) = \text{Max}\,|\lambda_i|; \qquad \text{i} = 1, 2. \tag{35}$$

The integration algorithm is unconditionally stable if and only if $\bar{\rho}(\boldsymbol{A}) \leqslant 1$ for any time step size. It can be verified that all the spectral radii of the numerical amplification matrices are less than or equal to unity.

Dissipation and dispersion

The errors can be expressed in terms of dissipation and dispersion, or amplitude error and phase error respectively. From equations (27)–(32), the eigenvalue of the numerical amplification matrices can be written as

$$\lambda_{1,2} = \exp[\bar{\omega}\Delta t(-\bar{\xi} \pm i)], \tag{36}$$

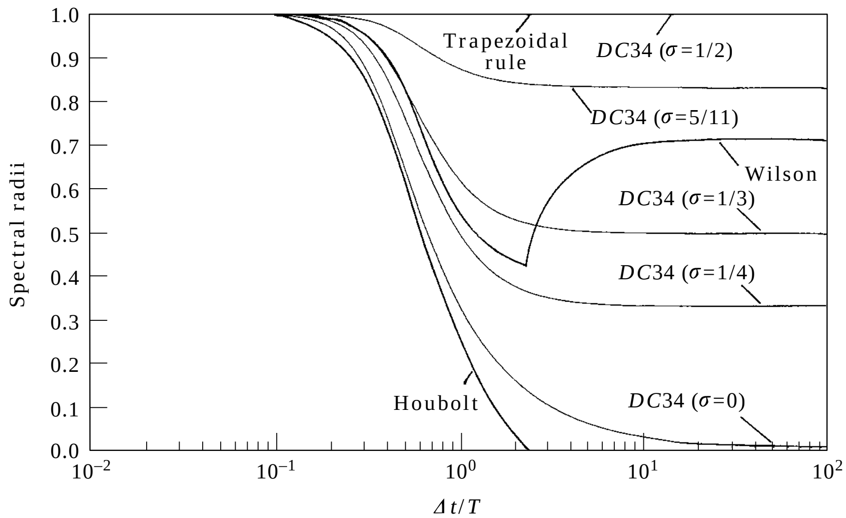

Figure 2. Spectral radii of *DC* algorithms.

where $\bar{\xi}$ and $\bar{\omega}$ are the algorithmic damping ratio and algorithmic frequency respectively. The algorithmic damping ratio is regarded as a measure of the numerical dissipation. The relative periodic error is defined as,

$$T = \frac{\omega}{\bar{\omega}} - 1, \tag{37}$$

which is regarded as the measure of the numerical dissipation.

The spectral radii of the dissipative time finite elements (*DC*) and the extrapolated time finite elements (*EX*) are plotted in Figures 2 and 3 respectively. It is noted that the non-dissipative time finite elements (*ND*) are not discussed separately since they can be treated as special cases of *DC* algorithms. Besides, $DC12(\sigma = 1)$ (or *ND*2) is found to be

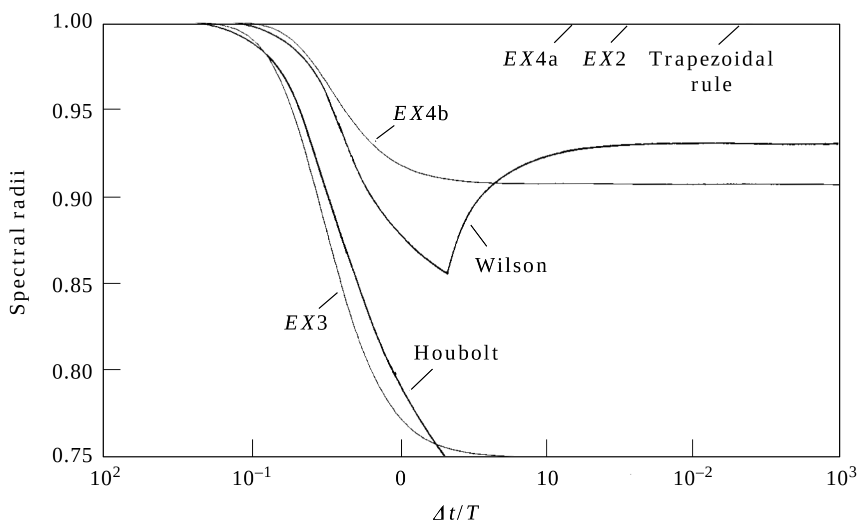

Figure 3. Spectral radii of *EX* algorithms. (*EX*3: $\beta_2 = 3/2$, *EX*4a: $\beta_2 = 0{\cdot}898883$, $\beta_3 = 1{\cdot}2$, *EX*4b: $\beta_2 = 0{\cdot}95$, $\beta_3 = 1{\cdot}2$).

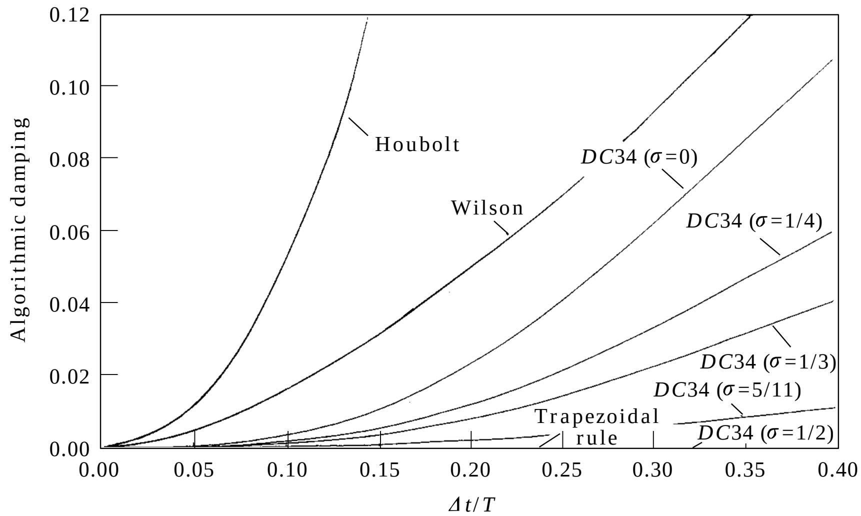

Figure 4. Algorithmic damping ratios of *DC* algorithms.

identical to *EX*2. For comparison, the spectral radii of some conventional algorithms including the Newmark method (trapezoidal rule), Wilson-θ method and Houbolt method are also plotted in Figures 2 and 3.

The algorithmic damping ratios of *DC* and *EX* algorithms are plotted in Figures 4 and 5 respectively. The algorithmic damping ratios of some conventional algorithms are also plotted in Figures 4 and 5. The period errors of *DC* and *EX* algorithms are plotted in Figures 6 and 7 respectively. The period errors of some conventional algorithms are also plotted in Figures 6 and 7. Logarithms of errors are plotted in Figure 8. It can be verified that the logarithms of errors are consistent with the absolute errors. From the above figures, it can be seen that the proposed algorithms are competitive with the conventional algorithms.

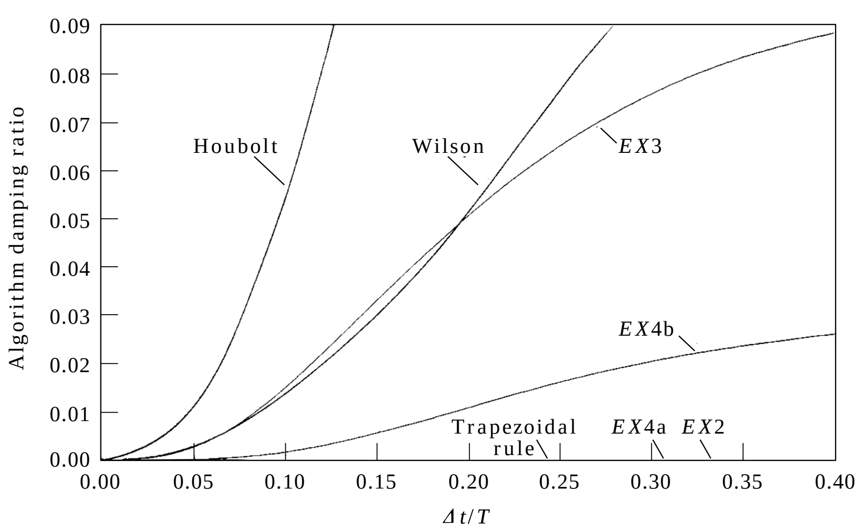

Figure 5. Algorithmic damping ratios of *EX* algorithms (*EX*3: $\beta_2 = 3/2$, *EX*4a: $\beta_2 = 0{\cdot}898883$, $\beta_3 = 1{\cdot}2$, *EX*4b: $\beta_2 = 0{:}95$, $\beta_3 = 1.2$).

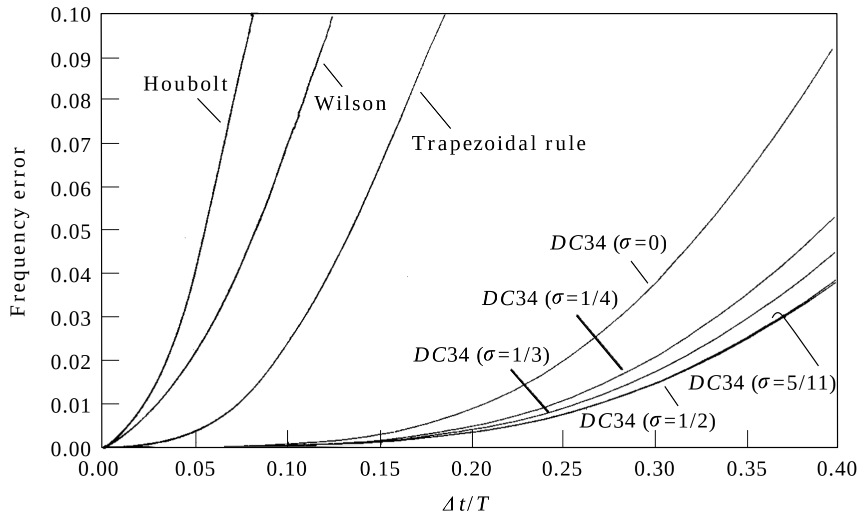

Figure 6. Period errors of DC algorithms.

6. NUMERICAL EXAMPLES

6.1. EXAMPLE 1

Consider a simply supported beam subjected to a harmonic excitation at the mid-span as shown in Figure 9. Assume the elastic modulus $EI = 10^6$, density per unit length $\rho A = 420$, length $L = 3$ and the excitation $P = 10^6 \sin(\omega_1 t/2)$, where ω_1 is the fundamental natural frequency of the beam. The analytical solution can be found in [32].

In the present study, the second-, third- and fourth-order-accurate extrapolation formulations are used to calculate the responses. Some conventional second-order-accurate algorithms are also used for comparison. The conventional Hermitian shape functions are used to discretize the beam spatially. The linear interpolation shape functions are used for temporal discretization, i.e.

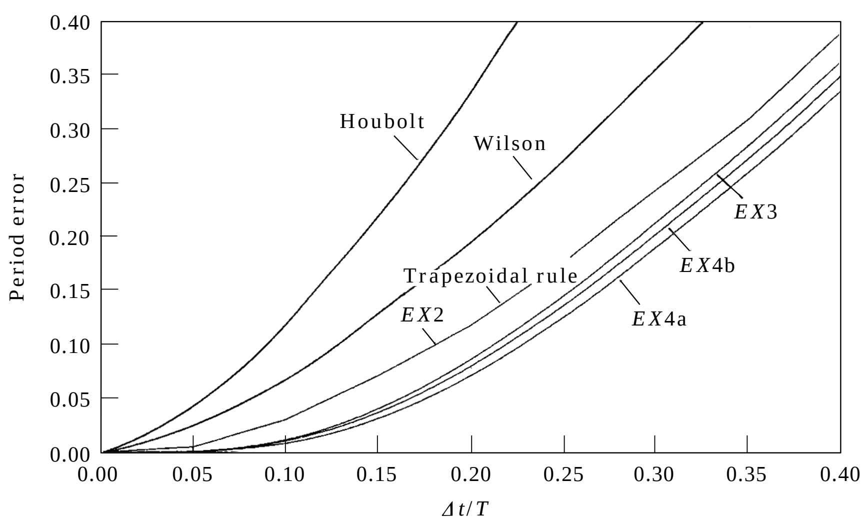

Figure 7. Period errors of EX algorithms ($EX3$: $\beta = 3/2$, $EX4$a: $\beta_2 = 0{\cdot}898883$, $\beta_3 = 1{\cdot}24$b: $\beta_2 = 0{\cdot}95$, $\beta_3 = 1{\cdot}2$).

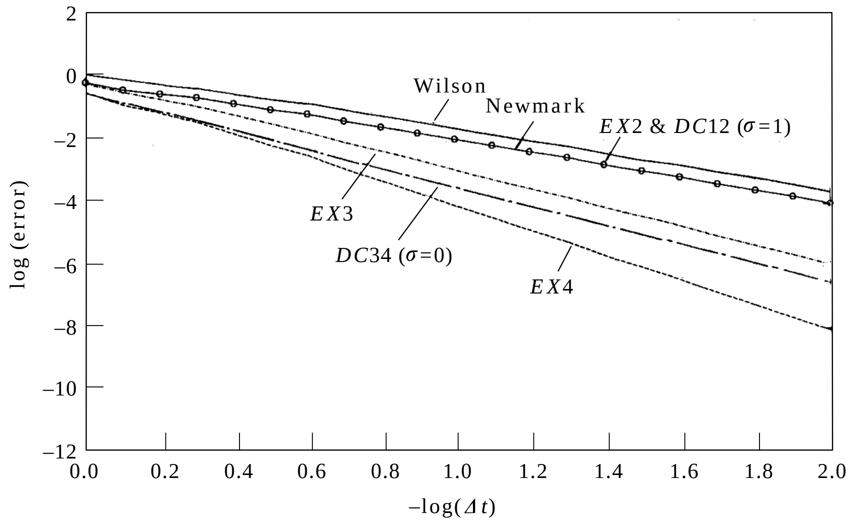

Figure 8. Convergent rate of DC and EX algorithms ($EX3$: $\beta_2 = 3/2$, $EX4$: $\beta_2 = 0{\cdot}898883$, $\beta_3 = 1{\cdot}2$).

$$\boldsymbol{N}(x) = [(2x^3 - 3lx^2 + l^3)/l^3, \quad (x^3 - 2lx^2 + l^2x)/l^2, \quad -(2x^3 - 3lx^2)/l^3, \quad (x^3 - lx^2)/l^2],$$

and

$$\Phi(t) = [(t_{n-1} - t)/\Delta t, \qquad (t - t_{n-1})/\Delta t]. \tag{38}$$

Based on equation (20), it can be shown that the related element matrices are

$$\int_{t_{n-1}}^{t_n} \Phi^T \mathbf{D} \dot{\Phi} \, \mathrm{d}t = \frac{1}{840} \begin{bmatrix} -156l & -22l & -54 & 13l & 156l & 22l & 54 & -13l \\ -22l & -4l^2 & -13l & 3l^2 & 22l & 4l^2 & 13l & -3l^2 \\ -54 & -13l & -156 & 22l & 54 & 13l & 156 & -22l \\ 13l & 3l^2 & 22l & -4l^2 & -13l & -3l^2 & -22l & 4l^2 \\ -156l & -22l & -54 & 13l & 156l & 22l & 54 & -13l \\ -22l & -4l^2 & -13l & 3l^2 & 22l & 4l^2 & 13l & -3l^2 \\ -54 & -13l & -156 & 22l & 54 & 13l & 156 & -22l \\ 13l & 3l^2 & 22l & -4l^2 & -13l & -3l^2 & -22l & 4l^2 \end{bmatrix}, \tag{39a}$$

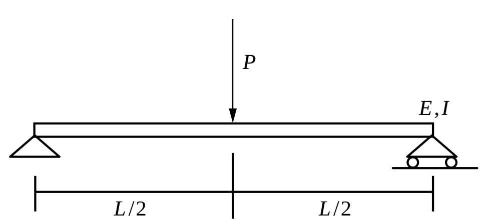

Figure 9. Simply supported beam.

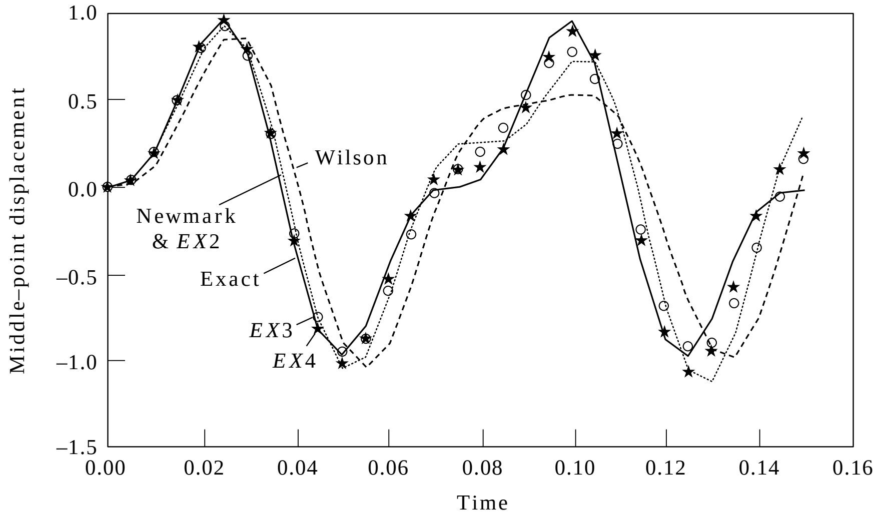

Figure 10. Time history for Example 1 ($EX3$: $\beta_2 = 3/2$, $EX4$: $\beta_2 = 0{\cdot}898883$, $\beta_3 = 1{\cdot}2$).

$$\int_{t_{n-1}}^{t_n} \Phi^T \bar{\boldsymbol{M}} \Phi \, \mathrm{d}t = \frac{-\Delta t l}{2520 \rho A}$$

$$\times \begin{bmatrix} 312l & 44l & 108 & -26l & 156l & 22l & 54 & -13l \\ 44l & 8l^2 & 26l & -6l^2 & 22l & 4l^2 & 13l & -3l^2 \\ 108 & 26l & 312 & -44l & 54 & 13l & 156 & -22l \\ -26l & -6l^2 & -44l & 8l^2 & -13l & -3l^2 & -22l & 4l^2 \\ 156l & 22l & 54 & -13l & 312l & 44l & 108 & -26l \\ 22l & 4l^2 & 13l & -3l & 312l & 44l & 108 & -6l^2 \\ 54 & 13l & 156 & -22l & 108 & 26l & 312 & -44l \\ -13l & -3l^2 & -22l & 4l^2 & -26l & -6l^2 & -44l & 8l^2 \end{bmatrix} \tag{39b}$$

$$\int_{t_{n-1}}^{t_n} \Phi^T \boldsymbol{K} \Phi \, \mathrm{d}t = \frac{\Delta t EI}{3l^3}$$

$$\times \begin{bmatrix} 24 & 12l & -24 & 12l & 12 & 6l & -12 & 6l \\ 12l & 8l^2 & -12l & 4l^2 & 6l & 4l^2 & -6l & 2l^2 \\ -24 & -12l & 24 & -12l & -12 & -6l & 12 & -6l \\ 12l & 4l^2 & -12l & 8l^2 & 6l & 2l^2 & -6l & 4l^2 \\ 12 & 6l & -12 & 6l & 24 & 12l & -24 & 12l \\ 6l & 4l^2 & -6l & 2l^2 & 12l & 8l^2 & -12l & 4l^2 \\ -12 & -6l & 12 & -6l & -24 & -12l & 24 & -12l \\ 6l & 2l^2 & -6l & 4l^2 & 12l & 4l^2 & -12l & 8l^2 \end{bmatrix} \tag{39c}$$

16 beam elements for spatial discretization and 30 time elements with $\Delta t = 0{\cdot}005$ are used. The results obtained from the extrapolation algorithms are shown in Figure 10. It can be

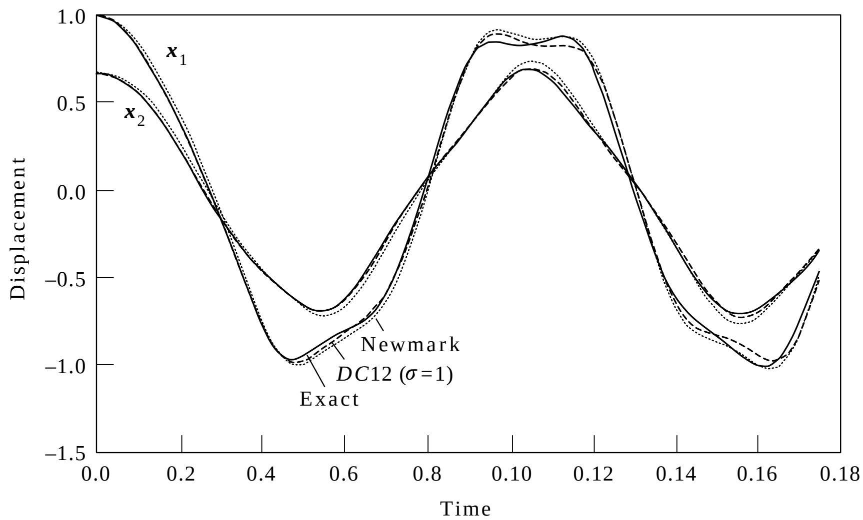

Figure 11. Time history for Example 2.

seen that the results of the present formulations are competitive with those of the conventional algorithms.

6.2. EXAMPLE 2

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

$$8\ddot{x}_1 + 1200(x_1 - x_2) + 8000(x_1 - x_2)^3 = (1000/3)\cos(6{\cdot}009t)$$
$$24\ddot{x}_2 - 1200(x_1 - x_2) - 8000(x_1 - x_2)^3 + 1800x_2 = 0 \tag{40}$$

with initial conditions $x_1(0) = 1$, $\dot{x}_1(0) = 0$, $x_2(0) = 2/3$, $\dot{x}_2(0) = 0$. The results of $DC12$ algorithm with $\Delta t = 0{\cdot}025$ are plotted in Figure 11. The curve coincides with the one obtained using the trapezoidal rule with $\Delta t = 0{\cdot}025$. It can be seen that they agree well

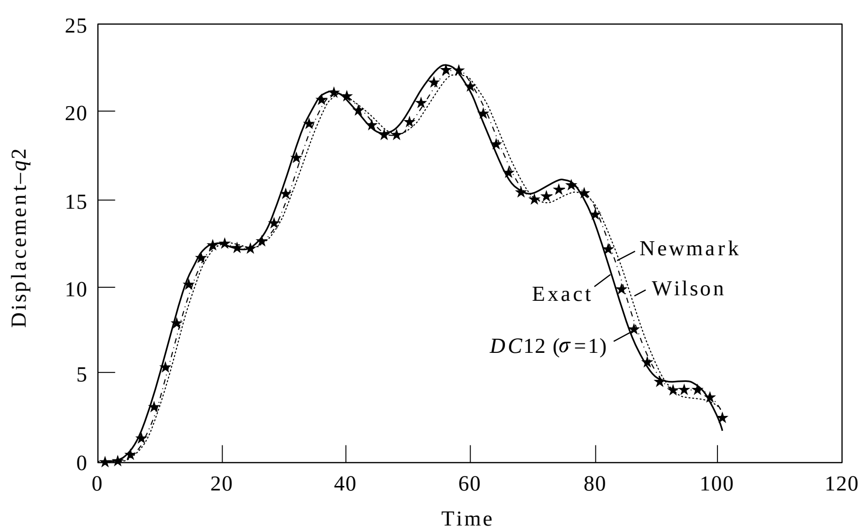

Figure 12. Time history for Example 3.

with the near-exact solution, which is obtained by using a very small time step via the trapezoidal rule.

6.3. EXAMPLE 3

Consider the beam in Example 1 now subjected to a moving load. The load has a constant magnitude of $P = 10^6$ with moving velocity $v = 10$. The analytical solution can be found in [33].

In the present study, $DC12$ algorithms with $\sigma = 1$ ($ND2$) is used for response calculation. Some conventional algorithms are also used for comparison. 10 spatial finite elements and 30 time finite elements with $\Delta t = 0{\cdot}01$ are used, and the results are shown in Figure 12. It can be seen that the results of the present formulation are competitive with those of the conventional algorithms.

7. CONCLUSIONS

In this paper, a mixed time finite element formulation for dynamic response analysis is presented. The formulation, implementation and the associated stability and accuracy characteristics are reported. The proposed formulation can be applied to linear and non-linear problems. It provides a unifying approach for space–time discretization in which the spatial discretization is consistent with the conventional finite element scheme whereas new temporal discretization and solution schemes are developed. The derived algorithms can also be used as time-marching algorithms. The present algorithms have the following salient features: they can acquire higher order accuracy with variable numerical dissipation and the higher order members of the family can be constructed systemically. The unification and versatility of the algorithms make the method attractive.

REFERENCES

1. I. Fried 1969 *AIAA Journal* **7**, 1170–1173. Finite-element analysis of time-dependent phenomena.
2. J. H. Argyris and D. W. Scharpf 1969 *Nuclear Engineering and Design* **10**, 456–464. Finite elements in time and space.
3. J. T. Oden 1969 *International Journal for Numerical Methods in Engineering* **1**, 247–259. A general theory of finite element. 2. Application.
4. C. D. Bailey 1975 *AIAA Journal* **13**, 1154–1157. Application of Hamilton's Law of Varying Action.
5. C. D. Bailey 1976 *Computer Methods in Applied Mechanics and Engineering* **7**, 235–247. The method of Ritz applied to the equation of Hamilton.
6. D. A. Peters and A. P. Izadpanah 1988 *Computational Mechanics* **3,** 73–88. hp-version finite elements for the space-time domain.
7. M. Borri, G. L. Ghiringhelli, M. Lanz, P. Mantegazza and T. Merlini 1985 *Computers & Structures* **20**, 495–508. Dynamic response of mechanical systems by a weak Hamilton formulation.
8. M. Borri 1986 *Computers and Mathematics with Applications* **12a**, 149–160. Helicopter rotor dynamics by finite element time approximation.
9. P. Lesaint and P. A. Raviart 1974 in C. de Boor (Ed.) *Mathematical Aspect of Finite Elements in Partial Differential Equations* New York: Academic Press. pp. 89–123. On a finite element method for solving the neutron transport equation.
10. C. Johnson, U. Navert and J. Pitkaranta 1984 *Computer Methods in Appled Mechanics and Engineering* **45**, 285–312. Finite element methds for linear hyperbolic problems.
11. C. Johnson 1993 *Computer Methods in Applied Mechanics and Engineering* **107**, 117–129. Discontinuous Galerkin finite element methods for second order hyperbolic problems.

12. G. M. HULBERT and J. R. HUGHES 1990. *Computer Methods in Applied Mechanics and Engineering* **84**, 327–348. Space–time finite element methods for second-order hyperbolic equations.
13. H. GOLDSTEIN 1980 *Classical Mechanics*. Addison-Wesley.
14. D. H. HODGES and R. R. BMLESS 1991. *Journal Guidance*, *Control & Dynamics* **14**, 148–156. Weak Hamiltonian finite element method for optimal control problems.
15. L. J. HOU and D. A. PETERS 1994 *Journal of Sound and Vibration* **173**, 611–632. Application of triangular space–time finite elements to problems of wave propagation.
16. C. I. BAJER, R. BOGACZ and C. BONTHOUX 1991 *Computers & Structures* **39**, 415–423. Adaptive space–time elements in the dynamic elastic-viscoplastic problem.
17. B. C. BELI and K. SURANA 1994 *International Journal for Numerical Methods in Engineering* **37**, 3545–3569. A space–time coupled p-version least-squares finite element formulation for unsteady fluid dynamics problems.
18. J. M. SANZ-SERNA 1983 *International Journal of Numerical Engineering Applications* **19**, 623–624. On finite elements simultaneously in space and time.
19. W. ZHOU, D. H. HODGES, A. R. ATIGLAN and A. ÖZBECK 1992 AIAA Paper No. 92-2379, *Proceedings of the* 33*rd Structures*, *Structural Dynamics*, *and Materials Conference*, Dallas, Texas, pp. 1747–1757. A mixed space–time finite element formulation for linear dynamic response of bars.
20. A. R. ATILGAN, D. H. HODGES, M.A. ÖZBECK and W. ZHAO 1996 *Journal of Sound and Vibration* **192**, 731–739. Space–time mixed finite elements for rods.
21. B. ZHAO and D. A. PETERS 1995 *ASME* 15*th Biennial Conference on Mechanical Vibration and Noise*, Boston, pp. 17–20. Space–time hp-version finite elements applied to wave propagation with moving loads.
22. J. A. STRICKLIN and W. E. HAISLER 1977 *Computers & Structures* **7**, 125–136. Formulation and solution procedures for non-linear structural analysis.
23. D. M. TRUJILLO 1982 *Journal of Applied Mechanics*, *ASME* **49**, 203–205. Stability analysis of an extrapolated force correction method for non-linear structural dynamics.
24. G. SHENG 1996 *Ph.D. Thesis*, *Nanyang Technological University*. Mixed time–space finite element formulations for dynamic problems.
25. M. HULBERT 1994 *Computer Methods in Applied Mechanics and Engineering* **113**, 1–9. A unified set of single-step asymptotic annihilation algorithms for structural dynamics.
26. M. BORRI and C. BOTTASSO 1993 *Computational Mechanics* **13**, 133–142. A general framework for interpreting time finite element formulations.
27. D. H. HODGES and L. J. HOU 1991 *Journal of Sound and Vibration* **145**, 169–178. Shape functions for mixed p-version finite elements in the time domain.
28. D. AHARONI and P. BAR-YOSEPH 1992 *Computational Mechanics* **9**, 359–374. Mixed finite element formulations in the time domain for solution of dynamics.
29. T. C. FUNG, S. C. FAN and G. SHENG 1996 *Computational Mechanics* **17**, 398–405. Extrapolated time finite elements.
30. R. K. KAPANIA and S. PARK 1996 *Computational Mechanics* **17**, 306–317. Nonlinear transient response and its sensitivity using finite elements in time.
31. T. J. R. HUGHES 1987 *The Finite Element Method*: *Linear Static and Dynamic Finite Element Analysis*. Englewood Cliffs, N.J.: Prentice–Hall.
32. M. PAZ 1991 *Structural Dynamics Theory and Computation*. Van Nostrand Reinhold.
33. G. B. WARBURTON 1976 *The Dynamical Behaviour of Structures*. Oxford: Pergamon Press.