AIAA Journal
Vol. 35, No. 1, January 1997

Penalized Weighted Residual Method for the Initial Value Problems

Seung Jo Kim[*] and Jin Yeon Cho[†]
Seoul National University, Seoul 151-742, Republic of Korea

To satisfy the need for network parallel computing in the whole time domain, a penalized weighted residual formulation for the second-order initial value problems is developed and its time finite element approximation is presented. To impose properly both the initial displacement and velocity, the initial velocity is imposed to the weighted residual equation as a penalized form that is constructed such that the approximated initial velocity can satisfy the initial condition in the average sense. By this procedure, the penalized weighted residual formulation makes it possible to handle the whole time domain of investigation and to use the conventional-displacement-based finite element technique without any other artificial routine. Through several numerical tests, it is confirmed that the present method gives very accurate solutions for various systems arising in the second-order initial value problems and presents promising characteristics for network parallel computation in the whole time domain of investigation.

Introduction

EARLIER, the use of finite element techniques in the time domain was suggested by Oden,[1] Fried,[2] and Argyris and Scharpf.[3] Nowadays, there are two approaches for obtaining finite element solutions in the time domain. One is Hamilton's principle, and the other is the weighted residual method. Baily[4] obtained solutions of dynamic problems using Hamilton's law of varying action, which is a weak form of Hamilton's principle, and revived interest in time finite element approximation. Bailey's work[4] was followed by that of many others.[5–7] Simkins[5] extended Hamilton's law of varying action to unconstrained variational statements. Riff and Baruch[6] applied the finite element technique using Hermite interpolation to Hamilton's law of varying action and noted the usefulness of managing the whole time domain of investigation. Hodges and Bless[7] adopted the mixed version of Hamilton's law of varying action and applied it to optimal control problems. On the other hand, Zienkiewicz and Taylor[8] unified the existing direct time integration methods by using a weighted residual approach and proposed a unified set of recurrence formulas in which weighting functions play a role as coefficients. By changing the weighting functions, the unified set of recurrence formulas could express the existing direct time integration methods. Recently, Hulbert and Hughes[9] and Johnson[10] used the discontinuous Galerkin finite element method and discretized the whole space–time domain via space–time slabs to solve the second-order hyperbolic equations.

While many theories on time finite element approximation were being developed, parallel computation techniques made startling progress in the field of elliptic boundary value problems. However, most theories previously developed on time finite element are inadequate for the full utilization of existing parallel computing power because, in many theories,[7,8] the time domain was confined to only one time step for obtaining solutions and a step-by-step sequential solving procedure was adopted, such as the direct time integration methods [i.e., the central difference method, the Newmark-β method (trapezoidal rule), the Wilson-θ method, the Houbolt method, etc.]. Thus development of time finite element methods that can fully use the existing parallel computation scheme of the boundary value problems is necessary.

Another issue, which all the previous theories skipped, must be considered. That is how to impose the initial velocity in the variational statements. This arises from the condition of the initial value problem that both displacement and velocity are imposed at the initial time, contrary to the fact that in the boundary value problem a displacement boundary condition cannot be assigned to the same boundary point that has a flux boundary condition. Thus a variational statement of the initial value problem needs more consideration of the initial velocity, which makes it different from the variational formulations of elliptic boundary value problems. However, previous theories for the initial value problems did not consider treatments of the initial velocity in the variational stage. As a result, the task of imposing the initial velocity is postponed to the stage of approximation. For example, in the particle dynamic problem, Hamilton's law of varying action[4] is as follows:

$$\int_{t_0}^{t_f} (-m\dot{u}\delta\dot{u} + c\dot{u}\delta u + ku\delta u - f\delta u)\,\mathrm{d}t + m\dot{u}\delta u\Big|_{t_f} - m\dot{u}\delta u\Big|_{t_0} = 0 \qquad (1)$$

where m, c, and k are the mass, damping, and stiffness, respectively; u, $\dot{u}$, and $\ddot{u}$ are the displacement, velocity, and acceleration, respectively; δ denotes variation; t_0 and t_f denote the initial and final times, respectively; and f is the external force. The initial velocity should be imposed on the third term of the left-hand side of Eq. (1). Because the initial displacement $u(t_0)$ is specified, the variation of the initial displacement $\delta u(t_0)$ must be zero for kinematically admissible variations. Therefore the initial velocity cannot be considered appropriately in $m\dot{u}\delta u|_{t_0} = 0$. Thus the theories using Hamilton's law of varying action[6,7] must consider tentatively $\delta u(t_0)$ as free variation independent of $u(t_0)$ and impose $u(t_0)$, $\dot{u}(t_0)$ at the approximation stage. However, it seems to be an inconsistent treatment. It may mingle the variational stage with the approximation stage. If the conventional weighted residual method is used, an artificial starting routine, which can consider the effect of initial velocity, is needed for multistep recurrence algorithms.[8]

In our weighted residual method with penalized initial velocity, the effect of initial velocity on the solution is permeated into the penalized weighted residual formulation, and as a result, there is no artificial routine for initial velocity in our method. Moreover, it can be easily applied to conventional-displacement-based finite element formulations that are usually adopted in solving the elliptic boundary value problems. Using our time finite element method, discretization of the whole time domain of investigation is performed for future applications of network parallel computing, as in the conventional discretization in elliptic boundary value problems.

Received June 3, 1996; revision received Sept. 21, 1996; accepted for publication Sept. 23, 1996; also published in *AIAA Journal on Disc*, Volume 2, Number 1.

[*] Professor, Department of Aerospace Engineering, San 56-1, Shillim-Dong, Kwanak-Ku. Member AIAA.

[†] Research Assistant, Department of Aerospace Engineering, San 56-1, Shillim-Dong, Kwanak-Ku.

Penalized Weighted Residual Method

A penalized weighted residual equation is derived from the dynamic equation of motion. In linear structural dynamics, the equations of motion for the discretized system are given with the initial condition (I.C.) in matrix form as

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} = \boldsymbol{f} \quad \text{with I.C.} \quad \boldsymbol{u}(t_0) = \boldsymbol{u}_o \quad \text{and} \quad \dot{\boldsymbol{u}}(t_0) = \boldsymbol{v}_o \tag{2}$$

where $\boldsymbol{M}$, $\boldsymbol{C}$, and $\boldsymbol{K}$ are the mass, damping, and stiffness matrices, respectively; $\boldsymbol{u}$, $\dot{\boldsymbol{u}}$, and $\ddot{\boldsymbol{u}}$ are the displacement, velocity, and acceleration vectors, respectively; and $\boldsymbol{f}$ is the external force vector. To satisfy the equilibrium condition in the sense of distribution, the dynamic equation of motion is weighted by kinematically admissible variation $\delta\boldsymbol{u}$ and integrated over the time domain. It can be written as

$$\int_{t_0}^{t_f} \delta\boldsymbol{u}^T(\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} - \boldsymbol{f})\,\mathrm{d}t = 0$$

$$\text{for all} \quad \delta\boldsymbol{u} \quad \text{such that} \quad \delta\boldsymbol{u}(t_0) = \boldsymbol{0} \tag{3}$$

To solve the difficult issue arising from the fact that both displacement and velocity must be imposed at the initial time in the initial value problems, the initial velocity is imposed to the weighted residual formulation by a penalized form. The penalty term for initial velocity is constructed such that the approximated initial velocity can satisfy the initial condition in the average sense. It assumes the form

$$\frac{1}{\varepsilon}\int_{t_0}^{t_f} \delta\dot{\boldsymbol{u}}^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o]\,\mathrm{d}t \tag{4}$$

where ε denotes the penalty parameter and has an inverse dimension of mass and $\boldsymbol{I}$ is the identity matrix. Because $\delta\boldsymbol{u}(t_0) = 0$, this form can be reduced to $\delta\boldsymbol{u}(t_f)^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o]/\varepsilon$ by integration. This penalty term is added to the left-hand side of Eq. (3). Through this procedure, the dynamic equation of motion in the sense of the average that has a penalized initial velocity is written as follows.

For all $\delta\boldsymbol{u}$ such that $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$

$$\begin{aligned} 0 &= \int_{t_0}^{t_f} \delta\boldsymbol{u}^T(\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} - \boldsymbol{f})\,\mathrm{d}t \\ &\quad + \frac{1}{\varepsilon}\int_{t_0}^{t_f} \delta\dot{\boldsymbol{u}}^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o]\,\mathrm{d}t \\ &= \int_{t_0}^{t_f} \delta\boldsymbol{u}^T(\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} - \boldsymbol{f})\,\mathrm{d}t \\ &\quad + \frac{1}{\varepsilon}\delta\boldsymbol{u}(t_f)^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o] \end{aligned} \tag{5}$$

Because Eq. (5) holds for all $\delta\boldsymbol{u}$ such that $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$, it must hold for all $\delta\boldsymbol{u}$ such that $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$ and $\delta\boldsymbol{u}(t_f) = \boldsymbol{0}$. Thus we can obtain the differential equation of motion. Similarly, because $\delta\boldsymbol{u}(t_f)$ is arbitrary, $\dot{\boldsymbol{u}}(t_0)$ is equal to $\boldsymbol{v}_o$. As a result, we can recover the original dynamic equation of motion (2). Conversely, if the original equation of motion holds, then the penalized weighted residual equation (5) holds automatically.

The penalized weighted residual equation can be rewritten in a weak form via integration by parts, which is very similar to the conventional finite element method in elliptic boundary value problems. The weak form is as follows.

For all $\delta\boldsymbol{u}$ such that $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$

$$\begin{aligned} 0 &= \int_{t_0}^{t_f} (-\delta\dot{\boldsymbol{u}}^T\boldsymbol{M}\dot{\boldsymbol{u}} + \delta\boldsymbol{u}^T\boldsymbol{C}\dot{\boldsymbol{u}} + \delta\boldsymbol{u}^T\boldsymbol{K}\boldsymbol{u} - \delta\boldsymbol{u}^T\boldsymbol{f})\,\mathrm{d}t \\ &\quad + \delta\boldsymbol{u}^T\boldsymbol{M}\dot{\boldsymbol{u}}\big|_{t_f} - \delta\boldsymbol{u}^T\boldsymbol{M}\dot{\boldsymbol{u}}\big|_{t_0} + \frac{1}{\varepsilon}\delta\boldsymbol{u}(t_f)^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o] \end{aligned} \tag{6}$$

Because $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$, it can be rewritten in the following form.

For all $\delta\boldsymbol{u}$ such that $\delta\boldsymbol{u}(t_0) = \boldsymbol{0}$

$$\begin{aligned} 0 &= \int_{t_0}^{t_f} (-\delta\dot{\boldsymbol{u}}^T\boldsymbol{M}\dot{\boldsymbol{u}} + \delta\boldsymbol{u}^T\boldsymbol{C}\dot{\boldsymbol{u}} + \delta\boldsymbol{u}^T\boldsymbol{K}\boldsymbol{u} - \delta\boldsymbol{u}^T\boldsymbol{f})\,\mathrm{d}t \\ &\quad + \delta\boldsymbol{u}(t_f)^T\boldsymbol{M}\dot{\boldsymbol{u}}(t_f) + \frac{1}{\varepsilon}\delta\boldsymbol{u}(t_f)^T\boldsymbol{I}[\dot{\boldsymbol{u}}(t_0) - \boldsymbol{v}_o] \end{aligned} \tag{7}$$

This weak form reflects the initial velocity considered not in the approximation stage but in the variational stage. Moreover, the weak form can be applied naturally to the displacement-based finite element formulation and does not need any artificial routine. In the next section, a finite element discretization procedure is shown.

Finite Element Discretization in Time Domain

In this section, we observe whether the proposed weighted residual method with penalized initial velocity is compatible with finite element approximation in the time domain; that is, the time domain can be treated by the finite element method just like the space domain. To apply the finite element techniques to this penalized weighted residual method, the time domain of investigation $\Omega = \{t \in R \mid t_0 \le t \le t_f = t_n\}$ is divided into subdomains $\Omega_i = \{t \in R \mid t_{i-1} \le t \le t_i\}$ $(i = 1, 2, \ldots, n)$, like the space domain. Each subdomain Ω_i is the domain of a finite element. We assume that displacement and virtual displacement are linear combinations of shape functions with time as the independent variable. These assume the following forms:

$$\boldsymbol{u} = \phi_i(t)\boldsymbol{u}_i \tag{8}$$

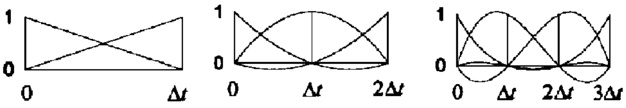
$$\delta\boldsymbol{u} = \phi_i(t)\delta\boldsymbol{u}_i \tag{9}$$

where $\boldsymbol{u}_i$ and $\delta\boldsymbol{u}_i$ stand for nodal sets of $\boldsymbol{u}$ and $\delta\boldsymbol{u}$ at a time i, respectively, and the summation convention is used. In our time finite element approximation, one can adopt a shape function as linear, quadratic, or cubic, etc. Because the second time derivative of the displacement vector should have meaning for recovery of acceleration, a linear shape function is not recommended for the second-order initial value problem. However, if one does not consider the recovery of acceleration, one can use the linear shape function in the second-order initial value problem. Various elements in the time domain are shown in Fig. 1, where time step Δt is the time length between nodal points. In linear, quadratic, and cubic elements, element sizes are Δt, $2\Delta t$, and $3\Delta t$, respectively. These approximations for displacement and virtual displacement are substituted directly into the penalized weighted residual equation (7) so as to obtain discretized algebraic equations. The resulting system of algebraic equations has the following form.

For all $\delta\boldsymbol{u}_i$ such that $\delta\boldsymbol{u}_o = \boldsymbol{0}$

$$\begin{aligned} 0 &= \delta\boldsymbol{u}_i^T\left\{\int_{t_0}^{t_f} (-\dot{\phi}_i\dot{\phi}_j\boldsymbol{M} + \phi_i\dot{\phi}_j\boldsymbol{C} + \phi_i\phi_j\boldsymbol{K})\,\mathrm{d}t\right\}\boldsymbol{u}_j \\ &\quad + \delta\boldsymbol{u}_i^T\left\{\phi_i(t_f)\dot{\phi}_j(t_f)\boldsymbol{M} + \frac{1}{\varepsilon}\phi_i(t_f)\dot{\phi}_j(t_0)\boldsymbol{I}\right\}\boldsymbol{u}_j \\ &\quad - \delta\boldsymbol{u}_i^T\left\{\int_{t_0}^{t_f} \phi_i\,\boldsymbol{f}\,\mathrm{d}t\right\} - \delta\boldsymbol{u}_i^T\left\{\frac{1}{\varepsilon}\phi_i(t_f)\boldsymbol{I}\boldsymbol{v}_o\right\} \end{aligned} \tag{10}$$

The first and third terms of Eq. (10) are usually encountered in the finite element discretization of elliptic boundary value problems. The second and fourth terms in Eq. (10) consist of a correlation between the initial velocity and the approximated initial velocity.



Linear element Quadratic element Cubic element

Fig. 1 Various time elements in the time domain.

Because Eq. (10) holds for every $\delta\boldsymbol{u}_i$ such that $\delta\boldsymbol{u}_o = \boldsymbol{0}$, it is rewritten as follows through reordering:

$$\begin{aligned}&\left\{\int_{t_0}^{t_f}(-\dot{\phi}_i\dot{\phi}_j\boldsymbol{M}+\phi_i\dot{\phi}_j\boldsymbol{C}+\phi_i\phi_j\boldsymbol{K})\,\mathrm{d}t\right\}\boldsymbol{u}_j\\&+\left\{\phi_i(t_f)\dot{\phi}_j(t_f)\boldsymbol{M}+\frac{1}{\varepsilon}\phi_i(t_f)\dot{\phi}_j(t_0)\boldsymbol{I}\right\}\boldsymbol{u}_j=\left\{\int_{t_0}^{t_f}\phi_i\,\boldsymbol{f}\,\mathrm{d}t\right\}\\&+\left\{\frac{1}{\varepsilon}\phi_i(t_f)\boldsymbol{I}\boldsymbol{v_o}-\frac{1}{\varepsilon}\phi_i(t_f)\dot{\phi}_o(t_0)\boldsymbol{I}\boldsymbol{u}_o-\phi_i(t_f)\dot{\phi}_o(t_f)\boldsymbol{M}\boldsymbol{u_o}\right\}\\&-\left\{\int_{t_0}^{t_f}(-\dot{\phi}_i\dot{\phi}_o\boldsymbol{M}+\phi_i\dot{\phi}_o\boldsymbol{C}+\phi_i\phi_o\boldsymbol{K})\,\mathrm{d}t\right\}\boldsymbol{u_o}\end{aligned}\tag{11}$$

where $1 \leq i, j \leq n$ and the right-hand side shows the global force vector terms. By matrix notation, it is denoted by global dynamic matrices $\boldsymbol{A}$, global displacement $\boldsymbol{U}$, and global force vectors $\boldsymbol{F}$ such as shown in the following:

$$[\boldsymbol{A}_n+\boldsymbol{A}_p]\boldsymbol{U}=\boldsymbol{F}_n+\boldsymbol{F}_p+\boldsymbol{F}_u\tag{12}$$

where subindex n denotes the naturally derived terms and subindex p denotes the terms that are related to the penalty term. Subindex u implies the effect of the given initial displacement. The resulting system of algebraic equations include all of the information for the whole time domain of investigation.

Numerical Test via Several Shape Functions

There are several crucial facts (e.g., period elongation, unconditional stability, etc.) in numerical time integration methods that determine the usefulness of the numerical time integration method. Among these crucial facts, period elongation may be the most serious one because it is closely related to the preciseness of the numerical solution. This relation is clearly shown in Fig. 2. To observe these characteristics of the penalized weighted residual method, some important examples are solved by the present time finite element approximation. In these examples, several shape functions with time as the independent variable are used. All examples use 0.0001 as the penalty parameter except example 1 and the simulations shown later in Fig. 4.

Example 1: Free-Fall Motion of Particle

As an elementary case, the free-fall motion of a particle is solved by the present method. The free-fall problem is described as follows:

$$m\ddot{u}=-mg\quad\text{with I.C.}\quad u(0)=u_o\quad\text{and}\quad\dot{u}(0)=v_o\tag{13}$$

where g denotes gravitational acceleration. The exact solution of the free-fall problem has the following form:

$$u(t)=-\tfrac{1}{2}gt^2+v_ot+u_o\tag{14}$$

The whole time domain of investigation is discretized into the only one quadratic element. Algebraic equations for the free-fall problem are written in matrix form as follows:

$$[\boldsymbol{A}_n+\boldsymbol{A}_p]\boldsymbol{U}=\boldsymbol{F}_n+\boldsymbol{F}_p+\boldsymbol{F}_u\tag{15}$$

By solving the matrix equation (15), we obtain the nodal displacement vector. It is reduced to

$$\boldsymbol{U}=\begin{Bmatrix}u_1\\u_2\end{Bmatrix}=\begin{Bmatrix}\dfrac{-gt_f^2}{8}+\dfrac{v_ot_f}{2}+u_o\\[2mm]\dfrac{-gt_f^2}{2}+v_ot_f+u_o\end{Bmatrix}\tag{16}$$

The solution is independent of the penalty parameter. The displacement over the time domain can be constructed from the quadratic shape functions:

$$u(t)=u_o\phi_o(t)+u_1\phi_1(t)+u_2\phi_2(t)=-\tfrac{1}{2}gt^2+v_ot+u_o\tag{17}$$

As a result, we obtain the exact displacement, velocity, and acceleration. Through this simple problem, we can observe the potential contained in the present method.

Example 2: Free Oscillating System

Free oscillating motion of a spring–mass system is simulated by the present method via linear, quadratic, and cubic shape functions. This fundamental system is important in observing the characteristics of the numerical time integration method. In particular, the period elongation can be clearly observed in the simulation of a free oscillating system. The equation of motion is described by

$$m\ddot{u}+ku=0\quad\text{with I.C.}\quad u(0)=u_o\quad\text{and}\quad\dot{u}(0)=v_o\tag{18}$$

The exact solution of the system would be in the form of

$$u(t)=u_o\cos(\sqrt{k/m}\,t)+v_o\sqrt{m/k}\sin(\sqrt{k/m}\,t)\tag{19}$$

In discretization of the time domain, a smaller size is used in the first element (at the initial time) than in other elements. In the present method, because the initial velocity is imposed in the first element and operates as a slope constraint, the time domain of the first element is fully influenced by the slope constraint. Therefore if the size of the first element is large compared with the variations of the solution, the first element may not follow the changes of the solution because of the slope constraint. Thus it is reasonable to reduce the size of the first element for capturing the rapid change at the initial time and obtaining the accurate solution. As a result, for a smaller size of the first element we get a more accurate amplitude. We call this discretization smaller first element discretization (SFD). The comparison of SFD and non-SFD, which does not use the SFD scheme, is shown in Fig. 3, where $m = k = 1$, $u(0) = 0$, and $\dot{u}(0) = 1$. The simulations in Fig. 3 are performed by linear, quadratic, and cubic shape functions, respectively. In the first-order

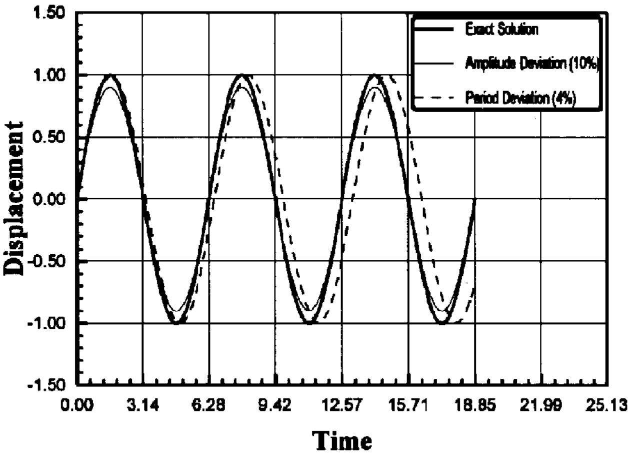


Fig. 2 Errors from amplitude deviation (10%) and period deviation (4%).

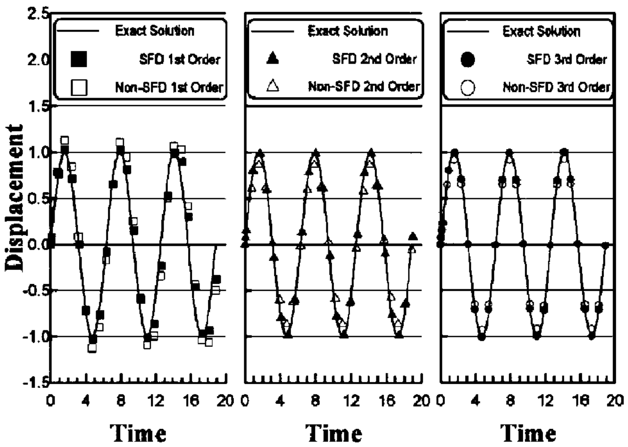


Fig. 3 Proper discretization (SFD), which can predict accurate amplitude vs improper discretization (non-SFD).

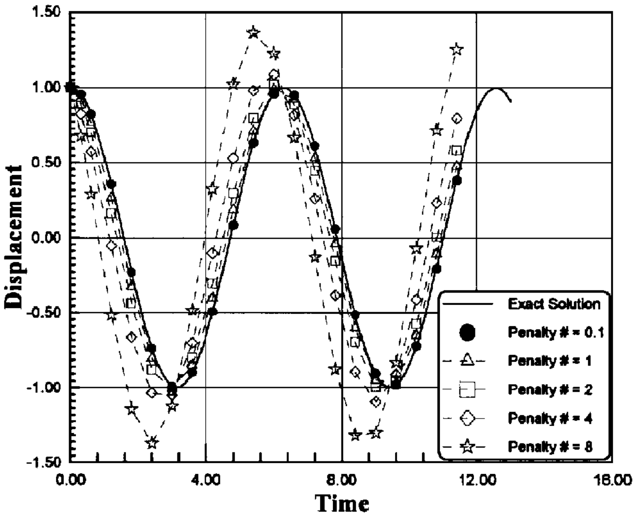


Fig. 4 Effect of penalty parameter on the global solution behavior.

SFD of Fig. 3, 25 linear elements, where the first element size is $\pi/40$ and others are $\pi/4$, are used. In the first-order non-SFD of Fig. 3, 24 linear elements, where element sizes are $\pi/4$, are used. In the second-order SFD of Fig. 3, 13 quadratic elements, where the first element sizes is $\pi/20$ and other element sizes are $\pi/2$, are used. In the second-order non-SFD of Fig. 3, 12 quadratic elements, where element sizes are $\pi/2$, are used. In the third-order SFD of Fig. 3, 9 cubic elements, where the first and second element sizes are $3\pi/40$ and $27\pi/40$ each, are used. The other element sizes are $3\pi/4$. In the third-order non-SFD of Fig. 3, 8 cubic elements, where element sizes are $3\pi/4$, are used. Though coarse meshes are used except for the first elements, the SFD shows no amplitude deviation. This shows that the amplitude is obtained precisely through SFD. Hereafter, SFD is used for all numerical simulations.

In Fig. 4, the effect of the penalty parameter on the global solution behavior is observed. With the conditions of $m = k = 1$, $u(0) = 1$, and $\dot{u}(0) = 0$, 10 quadratic elements, where the first element size is 0.6 and the other element sizes are 1.2, are used. From Fig. 4, the simulated solution converges globally to the exact solution as the penalty parameter goes to zero.

To observe the period elongation, the predicted results for $m = k = 1$, $u(0) = 1$, and $\dot{u}(0) = 0$ via linear, quadratic, and cubic shape functions are shown in Figs. 5a, 5b, and 5c, respectively. In the simulations of Figs. 5a, 5b, and 5c, 34, 18, and 12 elements, where the first element sizes are $\pi/30$, $2\pi/30$, and $\pi/10$, respectively, are used. The other element sizes are $\pi/3$, $2\pi/3$, and $\pi(9\pi/10$ for the second element), respectively. The predicted results are compared with the exact solution and the solution obtained from the trapezoidal rule (Newmark-β method) with time step $\pi/3$. The predicted results for all cases have less period elongation than the trapezoidal rule (Newmark-β method). Also, higher-order shape functions give more accurate results than lower-order shape functions.

Example 3: Damped Oscillating System

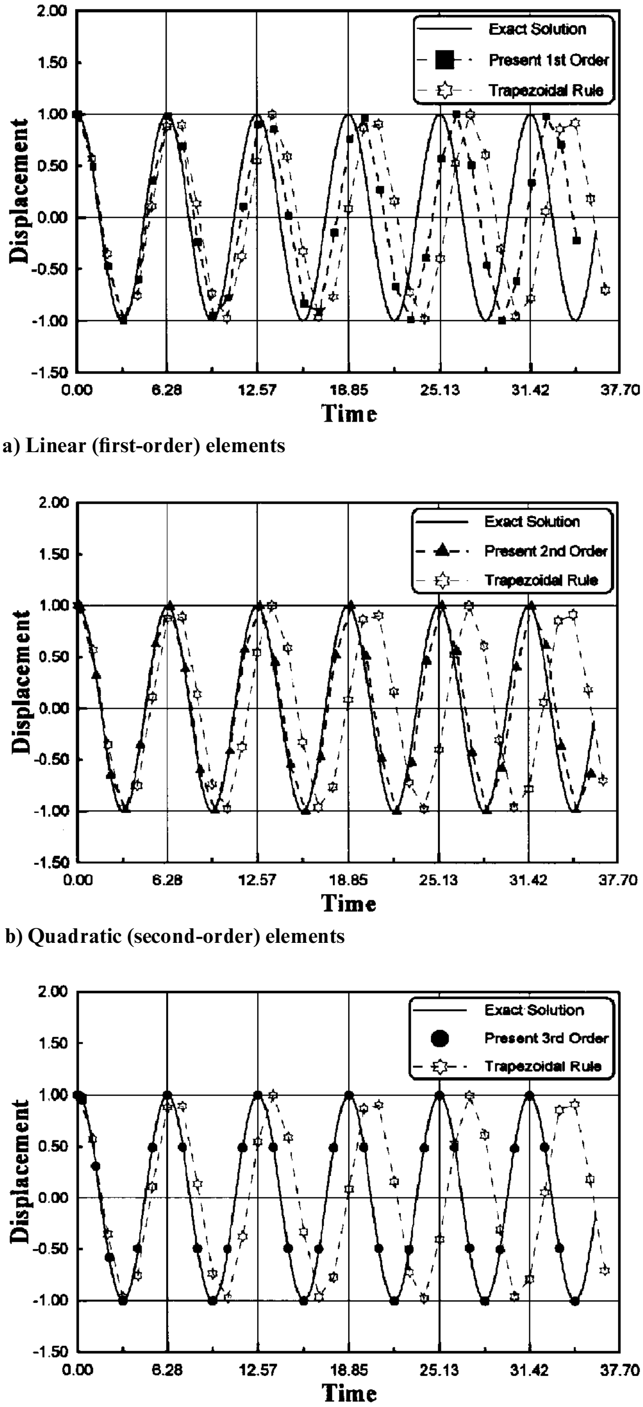
A damped oscillating system was simulated to observe the damping characteristics of the present method. The equation of motion is as follows:

$$m\ddot{u}+c\dot{u}+ku = 0 \quad \text{with I.C.} \quad u(0) = u_o \quad \text{and} \quad \dot{u}(0) = v_o \qquad (20)$$

When $u(0)$ is equal to zero, the exact solution reduces to

$$u(t) = (v_o/\omega_d)\exp(\_\zeta\omega_n t)\sin(\omega_d t)$$

$$\text{where} \quad \omega_n = \sqrt{k/m}, \quad \zeta = c/2m\omega_n, \quad \omega_d = \omega_n\sqrt{1 \_ \zeta^2} \qquad (21)$$



a) Linear (first-order) elements

b) Quadratic (second-order) elements

c) Cubic (third-order) elements

Fig. 5 Period elongations of the present method compared with simulated result by the trapezoidal rule for the free-oscillating case.

The simulation for case $m = k = 1$, $c = 0.3$, $u(0) = 0$, and $\dot{u}(0) = 1$ is performed by a quadratic shape function. In the numerical calculation, 22 quadratic elements, where the first element size is 0.2 and the other element sizes are 2, are used. The result is shown in Fig. 6. The present simulation shows less amplitude deviation and less period elongation than the result from the trapezoidal rule with time step 1. From this example, it can be confirmed that the present method predicts the damped motion of the dynamic system accurately.

Example 4: Forced Vibrating System

Forced vibrational motion with no damping is predicted in the fourth example. The equation of motion assumes the form

$$m\ddot{u} + ku = f \quad \text{with I.C.} \quad u(0) = u_o \quad \text{and} \quad \dot{u}(0) = v_o \qquad (22)$$

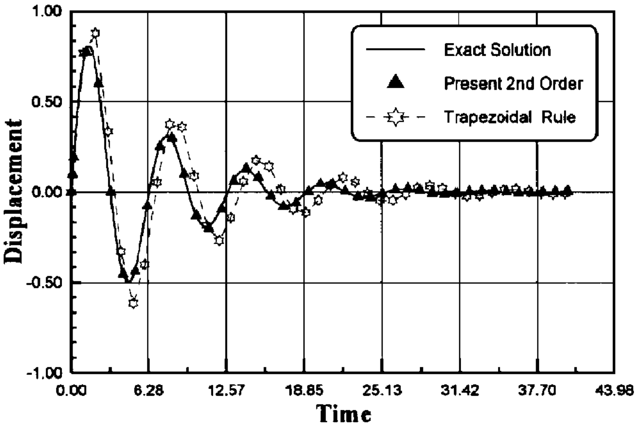


Fig. 6 Predicted result by the present method compared with simulated result by the trapezoidal rule for the damped oscillating case using quadratic (second-order) elements.

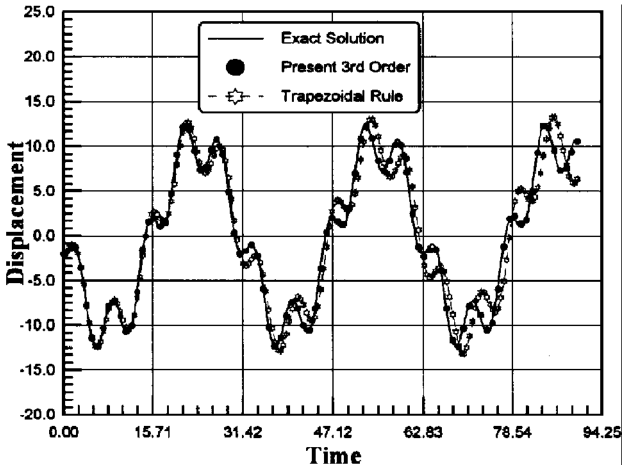


Fig. 7 Predicted result by the present method compared with simulated result by the trapezoidal rule for the forced vibrating case using cubic (third-order) elements.

When the applied force is $A\sin(\Omega t)$, the exact solution is

$$u(t) = u_o \cos\left(\sqrt{\frac{k}{m}}\,t\right) + \sqrt{\frac{m}{k}}\left(v_o - \frac{A\Omega}{k\_m\Omega^2}\right)\sin\left(\sqrt{\frac{k}{m}}\,t\right) + \frac{A}{k\_m\Omega^2}\sin(\Omega t) \qquad (23)$$

The case of $m = k = 1$, $f = \_10\sin(0.2t)$, $u(0) = \_2$, and $\dot{u}(0) = 0.2$ is simulated by 32 cubic elements. The first and second element sizes are 0.3 and 2.7, respectively. The other element sizes are 3. The predicted results are compared with the exact solution and the result of the trapezoidal rule, of which the time step Δt is 0.5. The comparisons are shown in Fig. 7. The comparison with the exact solution and the predicted one shows excellent agreement. In the trapezoidal rule, a smaller time step is used than in the present method; nevertheless, the result of the trapezoidal rule shows considerable deviation from the exact solution. By comparison, it shows the better accuracy of the present method.

Example 5: Wave Equation Discretized in Space Domain

In our final example, the wave equation is analyzed to verify the applicability to network parallel computing of the initial and boundary value problem in the space–time domain. The wave equation is as follows:

$$\alpha^2\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial t^2} = 0$$

$$\text{B.C.;} \quad u(0,t) = \bar{u}^o(t) \qquad u(l,t) = \bar{u}^l(t) \qquad (24)$$

$$\text{I.C.;} \quad u(x,0) = u_o(x) \qquad \dot{u}(x,0) = v_o(x)$$

where α is the wave propagation speed. B.C. denotes the boundary condition. In this paper, discretizations of the space domain and the time domain are carried out sequentially for the sake of focusing on the discretization of the time domain. It is exactly the same as the prismatic discretization of the space–time domain along the time axis. This sequential discretization is made up of two stages. The first stage is the conventional semidiscretization. The second stage is the time domain discretization via the present time finite element technique. As a result of the first stage, we obtain the following equation discretized in the space domain:

$$\boldsymbol{M\ddot{u}} + \boldsymbol{Ku} = \boldsymbol{f} \quad \text{with I.C.} \quad \boldsymbol{u}(0) = \boldsymbol{u_o} \quad \text{and} \quad \boldsymbol{\dot{u}}(0) = \boldsymbol{v_o} \qquad (25)$$

At the second stage, the present time finite element method is applied to the discretized wave equation (25). Via the aforementioned discretization procedure, the following matrix equation is obtained:

$$[\boldsymbol{A}_n + \boldsymbol{A}_p]\boldsymbol{U} = \boldsymbol{F}_n + \boldsymbol{F}_p + \boldsymbol{F}_u \qquad (26)$$

Matrix equation (26) has all the information for the solutions in the whole space–time domain. Because all the information is represented by only one matrix equation, it can be solved simultaneously by the existing parallel computing algorithms, such as the preconditioned conjugate gradient method, etc. In parallel computing, the whole space–time domain is decomposed into subdomains and each subdomain is allotted to one computer processor. By this procedure, network parallel computing is performed. The network parallel computing in the whole space–time domain is postponed for future work. For the case of $\alpha = 1$ with free boundaries, nonzero initial displacement, and zero initial velocity, numerical simulation is performed in Fig. 8 by one processor, In numerical simulation, 20 linear elements with element size 1 are used for the space domain. For the time domain, 51 linear elements, where the first time element size is 0.01 and other element sizes are 1, are used. This discretization is the same for 20 × 51 = 1020 linear rectangular elements in the space–time domain. The cross points of grid lines denote the nodal points in the space–time domain. It is interesting that the simulated result shows no period elongation. This can be explained as follows. Because the stiffness of the system is overestimated by the space domain discretization and the period is elongated by the time domain discretization, the errors from discretizations can be canceled out in appropriate discretization. From this example, it can be known that the present method conforms well to the initial and boundary value problems, such as wave equations, elastodynamics, etc., and can be used with a parallel solver for network parallel computing.

For the problems that have too immense degree of freedom (DOF) with respect to computer memory, an alternative conceptual numerical procedure is presented, where $\alpha = \frac{1}{2}$ with free boundaries, nonzero initial displacement, and zero initial velocity. In this numerical test, the whole space–time domain of investigation is divided into several space–time slabs and each space–time slab is divided into finite elements. The whole space domain is 133, and the whole time domain of investigation is 168. Space–time slabs and space–time elements are shown in Fig. 9. In the discretization of a slab, fine meshes are used for domains that are much affected

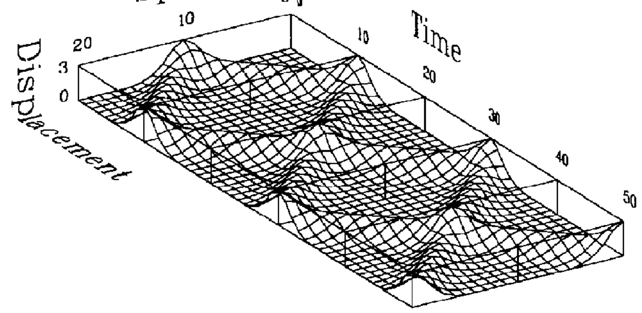


Fig. 8 Numerical simulation oriented to parallel computing for wave equation by the present method using linear elements in the space–time domain.

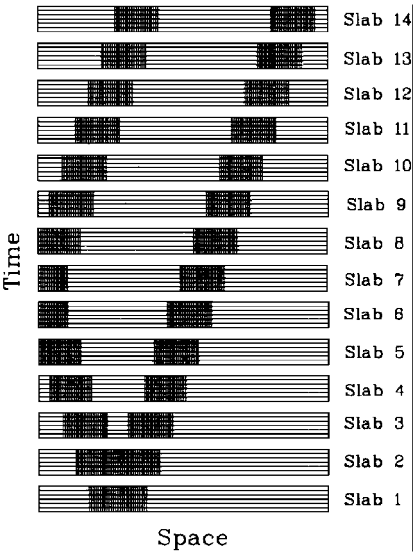


Fig. 9 Elements and slabs in the space–time domain for a conceptual numerical test that is oriented to immense DOF problems.

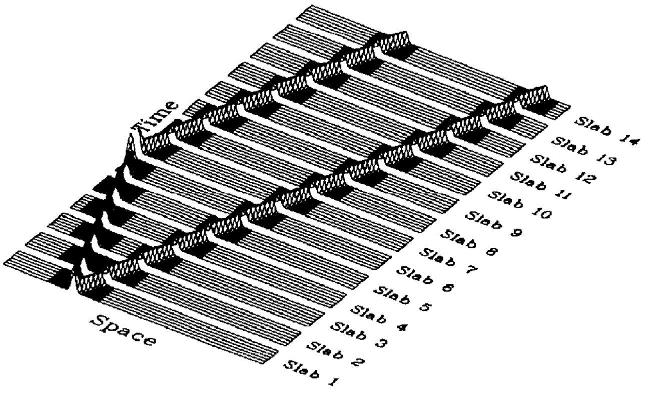


Fig. 10 Conceptual numerical test oriented to immense DOF problems by the present method using linear elements in the space–time domain.

by waves and coarse meshes are used for domains that are little affected by waves. By this procedure, fewer finite elements can be used. The solution procedure is as follows. At the outset, the first slab is solved simultaneously with the given initial conditions. Second, the obtained final displacements and velocities in the first slab are used for the initial conditions of the second slab and the second slab is solved. In this stage, only the final displacements and velocities of the first slab, which lie in nodal points of the second slab, are used. This routine is performed recursively until the last slab. In solving each slab, a parallel computation technique can be used. Note that a smaller-sized time element should be used at the final time of each slab to obtain a more accurate final velocity vector, which will be used for the initial velocity vector of the next slab. A simulated result is presented in Fig. 10, which shows good results. This conceptual numerical test shows the possibility that network parallel computing will be fully realized in the space–time domain by way of the present work.

Conclusions

In this work, a penalized weighted residual formulation for the second-order initial value problems is developed to conform well to an existing parallel computing scheme, and its time finite element approximation is performed. To impose properly both the initial displacement and the velocity, the initial velocity is imposed to this weighted residual equation as a penalized form. The penalty term for the initial velocity is constructed such that the approximated initial velocity can satisfy the initial velocity in the average sense. From the numerical test of several examples, it can be known that this penalty term imposes successfully the initial velocity. By this penalized weighted residual formulation, it is possible that the whole time domain of investigation can be managed at a time. Moreover, the conventional-displacement-based finite element technique can be easily used in the time domain without any other artificial routine. As a simple example, the free-fall problem is solved by only one quadratic element in the time domain and the exact solution is obtained over the whole time domain of investigation. In numerical tests, some important examples are solved via time finite element approximation of the penalized weighted residual method using linear, quadratic, and cubic elements. Through these examples, it is confirmed that the present method gives more accurate solutions than any other numerical integration methods for various systems, including damped and forced cases. From the example of wave propagation, it is determined that the present time finite element approximation of the penalized weighted residual method can be applied to the initial and boundary value problems, and it is concluded that the present time finite element method can be easily implemented in parallel computations of the whole time domain of investigation for the second-order initial and boundary value problems.

References

[1] Oden, J. T., “A General Theory of Finite Elements. Part 2. Applications,” *International Journal for Numerical Methods in Engineering*, Vol. 1, No. 3, 1969, pp. 247–259.

[2] Fried, I., “Finite Element Analysis of Time Dependent Phenomena,” *AIAA Journal*, Vol. 7, No. 6, 1969, pp. 1170–1173.

[3] Argyris, J. H., and Scharpf, D. W., “Finite Elements in Time and Space,” *Journal of the Royal Aeronautical Society*, Vol. 73, Dec. 1969, pp. 1041–1044.

[4] Bailey, C. D., “Application of Hamilton’s Law of Varying Action,” *AIAA Journal*, Vol. 13, No. 9, 1975, pp. 1154–1157.

[5] Simkins, T. E., “Unconstrained Variational Statements for Initial and Boundary Value Problems,” *AIAA Journal*, Vol. 16, No. 6, 1978, pp. 559–563.

[6] Riff, R., and Baruch, M., “Time Finite Element Discretization of Hamilton’s Law of Varying Action,” *AIAA Journal*, Vol. 22, No. 9, 1984, pp. 1310–1318.

[7] Hodges, D. W., and Bless, R. R., “Weak Hamiltonian Finite Element Method for Optimal Control Problems,” *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 1, 1991, pp. 148–156.

[8] Zienkiewicz, O. C., and Taylor, R. L., *The Finite Element Method*, 4th ed., McGraw–Hill, New York, 1991, Chap. 10.

[9] Hulbert, G. M., and Hughes, T. J. R., “Space-Time Finite Element Methods for Second-Order Hyperbolic Equations,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 84, No. 3, 1990, pp. 327–348.

[10] Johnson, C., “Discontinuous Galerkin Finite Element Methods for Second Order Hyperbolic Problems,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 107, Nos. 1–2, 1993, pp. 117–129.