Sung-Hoon Kim
Youn-sik Park
Center for Noise and Vibration Control (NOVIC)
Department of Mechanical Engineering
Korea Advanced Institute of Science and Technology
Science Town, Taejon 305-701
Korea

An Improved Finite Difference Type Numerical Method for Structural Dynamic Analysis

An improved finite difference type numerical method to solve partial differential equations for one-dimensional (1-D) structure is proposed. This numerical scheme is a kind of a single-step, second-order accurate and implicit method. The stability, consistency, and convergence are examined analytically with a second-order hyperbolic partial differential equation. Since the proposed numerical scheme automatically satisfies the natural boundary conditions and at the same time, all the partial differential terms at boundary points are directly interpretable to their physical meanings, the proposed numerical scheme has merits in computing 1-D structural dynamic motion over the existing finite difference numeric methods. Using a numerical example, the suggested method was proven to be more accurate and effective than the well-known central difference method. The only limitation of this method is that it is applicable to only 1-D structure. © *1994 John Wiley & Sons, Inc.*

INTRODUCTION

As machines become lighter and lighter and operate faster and faster, more accurate dynamic analysis is needed for a better design. Therefore, a distributed parameter model is preferred to a conventional discretized model. Using a distributed parameter model, however, the mathematical formulation of governing equations appears in the forms of partial differential equations. In general, it is nearly impossible to get analytical solutions of partial differential equations, so various methods to obtain numerical solutions have been developed. The numerical methods can be grouped largely into the finite element and finite difference methods, implicit and explicit methods, or single-step and multistep methods depending on the methods of discretizing spatial and time domains.

Among these, the finite element method FEM; (Zienkiewicz, 1977; Hughes, 1987) is the most popular approximate method for solving partial differential equations. The procedures of FEM are based upon the semidiscretization. First, the spatial domain is discretized into finite meshes, producing ordinary differential equations about time, then follows the time discretization, based on finite difference methods (FDM; Park, 1977; Dokainish and Subbaraj, 1989; Subbaraj and Dokainish, 1989). The advantages of FEM are its applicability to any general arbitrary shape structures. There are many numerical methods to

Received March 22, 1994; Accepted June 20, 1994
Shock and Vibration, Vol. 1, No. 6, pp. 569–583 (1994)

CCC 1070-9622/94/060569-15

solve partial differential equations directly. The standard Galerkin approach or several modified methods are examples. The Taylor–Galerkin method (Donea, 1984; Choe and Holsapple, 1992), which includes the Lax–Wendroff's idea and uses the Taylor series expansion and Galerkin approach, and the Petrov–Galerkin method (Morton and Parott, 1980; Bouloutas and Celia, 1991), characterized by their different sets of trial and test functions, were already developed for numerical analysis of structural dynamics and fluid dynamics problems. Also the space–time FEM (Oden, 1969; Hulbert and Hughes, 1990) involving finite mesh discretization of the time as well as the spatial domain was developed for solving elastodynamic problems.

FEM has many advantages, but it is not efficient for problems such as one-dimensional (1-D) multi-degree-of-freedom, space and time dependent coupled partial differential equations. In FEM it is difficult not only to select appropriate base functions but it also requires a large number of base functions. However, FDM (Ames, 1977; Anderson, Tannehill, and Pletcher, 1984) is superior to FEM on these kinds of problems because FDM does not require base functions and is easy to remesh. Especially in the fluid dynamics area, FDM is intensively used and many FDM algorithms have been developed. The Lax–Wendroff method, based on the Taylor series expansion, and the Crank–Nicolson method, utilizing the concept of the time centered central difference method, are examples of popular FDMs.

In this work, an improved finite difference type numerical scheme to solve partial differential equations of 1-D structure is proposed. The proposed numerical scheme uses the Newmark concept (Newmark, 1954) in discretizing the parameters in the space and time domain. So the proposed numerical scheme belongs to a single-step, second-order accurate and implicit method. The advantages of this numerical scheme are that it automatically satisfies the natural boundary conditions and all the partial differential terms can be directly interpreted to physical terms such as moment, shear force, etc. That is because the transformed governing equations are constructed with the displacement and known partial differential terms.

NUMERICAL ALGORITHM

Algorithm

A general 1-D 2p-order partial differential equation about $u(x, t)$ of a structural dynamic system (2nd-order time differential system) can be written as Eq. (1):

$$\begin{aligned} a_{2p}\frac{\partial^{2p}u(x,t)}{\partial x^{2p}} &+ a_{2p-1}\frac{\partial^{2p-1}u(x,t)}{\partial x^{2p-1}} \\ &+ a_{2p-2}\frac{\partial^{2p-2}u(x,t)}{\partial x^{2p-2}} \\ &+ \cdots + a_0 u(x,t) \\ &+ b_2\frac{\partial^2 u(x,t)}{\partial t^2} + b_1\frac{\partial u(x,t)}{\partial t} \\ &+ b_0 u(x,t) = f(x,t) \end{aligned} \tag{1}$$

where $a_{2p}, a_{2p-1}, \ldots, a_0, b_2, b_1, b_0$ are arbitrary coefficients and $f(x, t)$ is external disturbance (known for all x and t).

Equation (1) usually accompanies initial and boundary conditions and thus it is called the initial and boundary value problem. Many methods are known to approximate the partial differential terms with algebraic values, and the central difference method is most widely used in structural dynamic problems. It is well known that the central difference method transforms the partial differentials into difference equations. The method transforms the partial differentials in terms of the discretized displacements at certain spatial meshes and time steps using central difference formulae. In this article, however, an improved finite difference type numerical method is proposed. The proposed method is appropriate for numerical analysis for 1-D structural dynamic problems and more accurate and effective than the central difference method.

When solving this initial and boundary value problem, Eq. (1), the Newmark concept (Newmark, 1954) can be extensively applied to spatial domain x as well as to time domain t. As shown in Fig. 1, dividing the spatial and time domains into fine discrete meshes, and approximating the highest order partial differentials with the basis of linear or average value approximation, Eq. (1) can be transformed into a finite difference equation. For example, the highest order partial differential in spatial domain, $[\partial^{2p}u(x, t)]/\partial x^{2p}$, can be approximated by linear interpolation as shown in Fig. 1(b). Letting

$$\left.\frac{\partial^{2p}u(x,t)}{\partial x^{2p}}\right|_{x=\bar{x},t=(j+1)\,\Delta t}$$

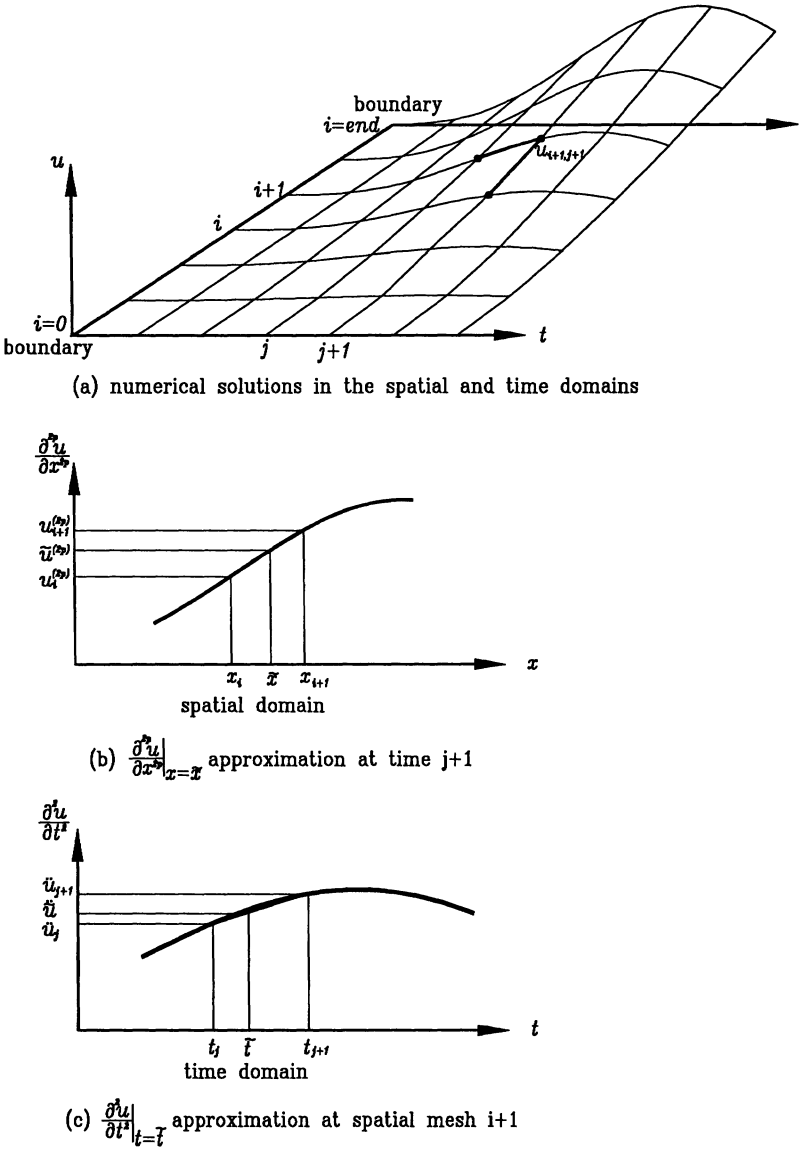


FIGURE 1 The highest order partial differential approximation.

be $\tilde{u}^{(2p)}$, $\tilde{u}^{(2p)}$ can be written as Eq. (2a):

$$\left.\frac{\partial^{2p}u(x,\,t)}{\partial x^{2p}}\right|_{x=\tilde{x},t=(j+1)\,\Delta t} = \tilde{u}^{(2p)} = u_{i,j+1}^{(2p)} + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{\Delta x}\,\tilde{x} \quad \text{for } 0 \le \tilde{x} \le \Delta x \tag{2a}$$

where $\Delta x = x_{i+1} - x_i$, superscript $(2p)$ means the $(2p)$th-order partial differentiation with respect to spatial domain x and subscripts i and j mean the ith spatial mesh and the jth time step, respectively. Integrating Eq. (2a) with respect to $\tilde{x}$ yields one order lower differentiation, $\tilde{u}^{(2p-1)}$, as Eq. (2b):

$$\tilde{u}^{(2p-1)} = u_{i,j+1}^{(2p-1)} + u_{i,j+1}^{(2p)}\tilde{x} + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{\Delta x}\,\frac{\tilde{x}^2}{2!} \tag{2b}$$

where $\tilde{u}^{(2p-1)} = u_i^{(2p-1)}$ at $\tilde{x} = 0$. Continuous $(2p - 1)$ times integration in series with respect to $\tilde{x}$ results

$$\tilde{u}^{(2p-2)} = u_{i,j+1}^{(2p-2)} + u_{i,j+1}^{(2p-1)}\tilde{x} + u_{i,j+1}^{(2p)}\frac{\tilde{x}^2}{2!} + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{\Delta x}\frac{\tilde{x}^3}{3!} \tag{2c}$$

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$\tilde{u}^{(1)} = u_{i,j+1}^{(1)} + u_{i,j+1}^{(2)}\tilde{x} + u_{i,j+1}^{(3)}\frac{\tilde{x}^2}{2!} + \cdots + u_{i,j+1}^{(2p)}\frac{\tilde{x}^{2p-1}}{(2p-1)!} + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{\Delta x}\frac{\tilde{x}^{2p}}{(2p)!} \tag{2d}$$

$$\tilde{u} = u_{i,j+1} + u_{i,j+1}^{(1)}\tilde{x} + u_{i,j+1}^{(2)}\frac{\tilde{x}^2}{2!} + u_{i,j+1}^{(3)}\frac{\tilde{x}^3}{3!} + \cdots + u_{i,j+1}^{(2p)}\frac{\tilde{x}^{2p}}{(2p)!} + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{\Delta x}\frac{\tilde{x}^{2p+1}}{(2p+1)!}. \tag{2e}$$

In Eq. (2e), replacing $\tilde{x}$ with Δx, then $\tilde{u}$ becomes the value at the $(i + 1)$th spatial mesh and the $(j + 1)$th time step, $u_{i+1,j+1}$, as Eq. (2f):

$$\begin{aligned} u_{i+1,j+1} = {} & u_{i,j+1} + u_{i,j+1}^{(1)}\Delta x + u_{i,j+1}^{(2)}\frac{\Delta x^2}{2!} \\ & + u_{i,j+1}^{(3)}\frac{\Delta x^3}{2!} + \cdots + u_{i,j+1}^{(2p)}\frac{\Delta x^{2p}}{(2p)!} \\ & + \frac{u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}}{(2p+1)!}\Delta x^{2p}. \end{aligned} \tag{2f}$$

Rearranging Eq. (2f), $u_{i+1,j+1}^{(2p)}$, which is shown in the numerator of the last term at the right-hand side of Eq. (2f), can be expressed as Eq. (3):

$$\begin{aligned} u_{i+1,j+1}^{(2p)} = {} & \frac{(2p+1)!}{\Delta x^{2p}}(u_{i+1,j+1} - u_{i,j+1}) \\ & - \frac{(2p+1)!}{\Delta x^{2p-1}}u_{i,j+1}^{(1)} - \frac{(2p+1)!}{2!\,\Delta x^{2p-2}}u_{i,j+1}^{(2)} \\ & - \frac{(2p+1)!}{3!\,\Delta x^{2p-3}}u_{i,j+1}^{(3)} \\ & - \cdots - \frac{(2p+1)!}{(2p-1)!\,\Delta x}u_{i,j+1}^{(2p-1)} \\ & - \left\{\frac{(2p+1)!}{(2p)!} - 1\right\}u_{i,j+1}^{(2p)}. \end{aligned} \tag{3}$$

In Eq. (3), $u_{i,j+1}$ and all its partial differentials, $u_{i,j+1}^{(1)}, u_{i,j+1}^{(2)}, \ldots, u_{i,j+1}^{(2p)}$, are known values and $u_{i+1,j+1}$ is unknown. Equation (3) can be written in a simple form by defining the coefficients, $c_{m,n}$ ($m = 1, 2, \ldots, 2p$ and $n = 0, 1, \ldots, 2p$), as follows:

$$\begin{aligned} u_{i+1,j+1}^{(2p)} \equiv {} & c_{2p,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{2p,1}u_{i,j+1}^{(1)} \\ & - c_{2p,2}u_{i,j+1}^{(2)} - \cdots - c_{2p,2p-1}u_{i,j+1}^{(2p-1} - c_{2p,2p}u_{i,j+1}^{(2p)} \end{aligned} \tag{4}$$

where

$$c_{2p,0} \equiv \frac{(2p+1)!}{\Delta x^{2p}}, \quad c_{2p,1} \equiv \frac{(2p+1)!}{\Delta x^{2p-1}},$$

$$c_{2p,2} \equiv \frac{(2p+1)!}{2!\,\Delta x^{2p-2}}, \quad \ldots,$$

$$c_{2p,2p-1} \equiv \frac{(2p+1)!}{(2p-1)!\,\Delta x}, \quad c_{2p,2p} \equiv \frac{(2p+1)!}{(2p)!} - 1.$$

$u_{i+1,j+1}^{(2p-1)}$ can be also obtained from Eq. (2b) by replacing $\tilde{x}$ with Δx as Eq. (5):

$$\begin{aligned} u_{i+1,j+1}^{(2p-1)} = {} & u_{i,j+1}^{(2p-1)} + u_{i,j+1}^{(2p)}\Delta x \\ & + \frac{\Delta x}{2}(u_{i+1,j+1}^{(2p)} - u_{i,j+1}^{(2p)}). \end{aligned} \tag{5}$$

Plugging the value of $u_{i+1,j+1}^{(2p)}$ from Eq. (3) into Eq. (5) and rearranging, results in

$$\begin{aligned} u_{i+1,j+1}^{(2p-1)} = {} & \frac{(2p+1)!}{2\,\Delta x^{2p-1}}(u_{i+1,j+1} - u_{i,j+1}) \\ & - \frac{(2p+1)!}{2\,\Delta x^{2p-2}}u_{i,j+1}^{(1)} - \frac{(2p+1)!}{2\cdot 2!\,\Delta x^{2p-3}}u_{i,j+1}^{(2)} \\ & - \frac{(2p+1)!}{2\cdot 3!\,\Delta x^{2p-4}}u_{i,j+1}^{(3)} \\ & - \cdots - \left(\frac{2p\cdot(2p+1)}{2} - 1\right)u_{i,j+1}^{(2p-1)}) \\ & - \frac{\Delta x}{2}(2p-1)u_{i,j+1}^{(2p)}. \end{aligned} \tag{6}$$

Equation (6) can be written in a simple form as follows:

$$\begin{aligned} u_{i+1,j+1}^{(2p-1)} \equiv {} & c_{2p-1,0}(u_{i+1,j+1} - u_{i,j+1}) \\ & - c_{2p-1,1}u_{i,j+1}^{(1)} - c_{2p-1,2}u_{i,j+1}^{(2)} \\ & - \cdots - c_{2p-1,2p-1}u_{i,j+1}^{(2p-1} \\ & - c_{2p-1,2p}u_{i,j+1}^{(2p)} \end{aligned} \tag{7}$$

where

$$c_{2p-1,0} \equiv \frac{(2p+1)!}{2\,\Delta x^{2p-1}}, \quad c_{2p-1,1} \equiv \frac{(2p+1)!}{2\,\Delta x^{2p-2}},$$

$$c_{2p-1,2} \equiv \frac{(2p+1)!}{2 \cdot 2!\,\Delta x^{2p-3}}, \quad \ldots,$$

$$c_{2p-1,2p-1} \equiv \frac{2p \cdot (2p+1)}{2} - 1,$$

$$c_{2p-1,2p} \equiv \frac{\Delta x}{2}(2p-1).$$

Repeating this procedure continuously, all the partial differentials at the $(i+1)$th spatial mesh and the $(j+1)$th time step, $u^{(1)}_{i+1,j+1}$, $u^{(2)}_{i+1,j+1}$, . . . , $u^{(2p)}_{i+1,j+1}$, can be expressed with the displacement and differentials at the one previous spatial mesh, $u_{i,j+1}$, $u^{(1)}_{i,j+1}$, . . . , $u^{(2p-1)}_{i,j+1}$, $u^{(2p)}_{i,j+1}$, and $u_{i+1,j+1}$, as:

$$\begin{aligned}
u^{(2p)}_{i+1,j+1} &= c_{2p,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{2p,1}u^{(1)}_{i,j+1} - c_{2p,2}u^{(2)}_{i,j+1} - \cdots - c_{2p,2p-1}u^{(2p-1)}_{i,j+1} - c_{2p,2p}u^{(2p)}_{i,j+1} \\
u^{(2p-1)}_{i+1,j+1} &= c_{2p-1,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{2p-1,1}u^{(1)}_{i,j+1} - c_{2p-1,2}u^{(2)}_{i,j+1} - \cdots - c_{2p-1,2p-1}u^{(2p-1)}_{i,j+1} - c_{2p-1,2p}u^{(2p)}_{i,j+1} \\
u^{(2p-2)}_{i+1,j+1} &= c_{2p-2,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{2p-2,1}u^{(1)}_{i,j+1} - c_{2p-2,2}u^{(2)}_{i,j+1} - \cdots - c_{2p-2,2p-1}u^{(2p-1)}_{i,j+1} - c_{2p-2,2p}u^{(2p)}_{i,j+1} \\
&\vdots \\
u^{(2)}_{i+1,j+1} &= c_{2,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{2,1}u^{(1)}_{i,j+1} - c_{2,2}u^{(2)}_{i,j+1} - \cdots - c_{2,2p-1}u^{(2p-1)}_{i,j+1} - c_{2,2p}u^{(2p)}_{i,j+1} \\
u^{(1)}_{i+1,j+1} &= c_{1,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{1,1}u^{(1)}_{i,j+1} - c_{1,2}u^{(2)}_{i,j+1} - \cdots - c_{1,2p-1}u^{(2p-1)}_{i,j+1} - c_{1,2p}u^{(2p)}_{i,j+1}.
\end{aligned} \tag{8}$$

Likewise, the time differentials at the $(j+1)$th time step and the $(i+1)$th spatial mesh can be expressed as follows:

$$\begin{aligned}
\ddot{u}_{i+1,j+1} &= \frac{6}{\Delta t^2}(u_{i+1,j+1} - u_{i+1,j}) \\
&\quad - \frac{6}{\Delta t}\dot{u}_{i+1,j} - 2\ddot{u}_{i+1,j} \\
&\equiv d_{2,0}(u_{i+1,j+1} - u_{i+1,j}) \\
&\quad - d_{2,1}\dot{u}_{i+1,j} - d_{2,2}\ddot{u}_{i+1,j}
\end{aligned} \tag{9a}$$

$$\begin{aligned}
\dot{u}_{i+1,j+1} &= \frac{3}{\Delta t}(u_{i+1,j+1} - u_{i+1,j}) \\
&\quad - 2\dot{u}_{i+1j} - \frac{\Delta t}{2}\ddot{u}_{i+1,j} \\
&\equiv d_{1,0}(u_{i+1,j+1} - u_{i+1,j}) \\
&\quad - d_{1,1}\dot{u}_{i+1,j} - d_{1,2}\ddot{u}_{i+1,j}
\end{aligned} \tag{9b}$$

where

$$d_{1,0} \equiv \frac{3}{\Delta t}, \quad d_{1,1} \equiv 2, \quad d_{1,2} \equiv \frac{\Delta t}{2},$$

$$d_{2,0} \equiv \frac{6}{\Delta t^2}, \quad d_{2,1} \equiv \frac{6}{\Delta t}, \quad d_{2,2} \equiv 2$$

and dot $(\cdot)$ denotes the differentiation with respect to time.

In the above differential approximation process, the linear approximation is used. But the average value approximation can also be used. When the average value approximation method is chosen for differential approximation, the transformation of all the differentials can be processed by almost similar manners.

Substituting Eqs. (8) and (9a,b) into Eq. (1), and solving for $u_{i+1,j+1}$ results in:

$$\begin{aligned}
&(a_{2p}c_{2p,0} + a_{2p-1}c_{2p-1,0} + \cdots + a_1c_{1,0} + a_0 \\
&+ b_2d_{2,0} + b_1d_{1,0} + b_0)u_{i+1,j+1} \\
&= (a_{2p}c_{2p,0} + a_{2p-1}c_{2p-1,0} + \cdots + a_1c_{1,0})u_{i,j+1} \\
&+ (a_{2p}c_{2p,1} + a_{2p-1}c_{2p-1,1} + \cdots + a_1c_{1,1})u^{(1)}_{i,j+1} \\
&+ (a_{2p}c_{2p,2} + a_{2p-1}c_{2p-1,2} + \cdots + a_1c_{1,2})u^{(2)}_{i,j+1} \\
&+ \cdots + (a_{2p}c_{2p,2p} + a_{2p-1}c_{2p-1,2p} \\
&+ \cdots + a_1c_{1,2p})u^{(2p)}_{i,j+1} + (b_2d_{2,0} + b_1d_{1,0})u_{i+1,j} \\
&+ (b_2d_{2,1} + b_1d_{1,1})\dot{u}_{i+1,j} + (b_2d_{2,2} + b_1d_{1,2})\ddot{u}_{i+1,j} \\
&+ f_{i+1,j+1}.
\end{aligned} \tag{10}$$

Equation (10) can be written in a vector form as Eq. (11):

$$\begin{aligned} u_{i+1,j+1} &= [A_0 \quad A_1 \quad A_2 \quad \cdots \quad A_{2p}] \\ &\quad \{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{(2p)}\}^T_{i,j+1} \\ &\quad + [B_0 \quad B_1 \quad B_2]\{u \quad \dot{u} \quad \ddot{u}\}^T_{i+1,j} \\ &\quad + F_{i+1,j+1} \\ &= \mathbf{A}\{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{(2p)}\}^T_{i,j+1} \\ &\quad + \mathbf{B}\{u \quad \dot{u} \quad \ddot{u}\}^T_{i,j+1} + F_{i+1,j+1} \end{aligned} \tag{11}$$

where $\mathbf{A} = [A_0 \quad A_1 \quad A_2 \quad \cdots \quad A_{2p}]$ and $\mathbf{B} = [B_0 \quad B_1 \quad B_2]$

$$\begin{aligned} &a_{2p}c_{2p,0} + a_{2p-1}c_{2p-1,0} + \cdots + a_1c_{1,0} \\ &\qquad + a_0 + b_2d_{2,0} + b_1d_{1,0} + b_0 = D \\ &A_0 = (a_{2p}c_{2p,0} + a_{2p-1}c_{2p-1,0} + \cdots + a_1c_{1,0})/D \\ &A_1 = (a_{2p}c_{2p,1} + a_{2p-1}c_{2p-1,1} + \cdots + a_1c_{1,1})/D \\ &\vdots \qquad\qquad \vdots \\ &A_{2p} = \\ &\qquad (a_{2p}c_{2p,2p} + a_{2p-1}c_{2p-1,2p} + \cdots + a_1c_{1,2p})/D \\ &\qquad B_0 = (b_2d_{2,0} + b_1d_{1,0})/D, \\ &\qquad B_1 = (b_2d_{2,1} + b_1d_{1,1})/D, \\ &\qquad B_2 = (b_2d_{2,2} + b_1d_{1,2})/D \\ &\qquad F_{i+1,j+1} = f_{i+1,j+1}/D. \end{aligned}$$

As shown in Eq. (11), the transformed equation is in an implicit form. Figure 2 summarizes all the above explained procedures to transform partial differential equations into finite difference equations.

In order to calculate $u_{i+1,j+1}$ [the value at the $(i + 1)$th spatial mesh and $(j + 1)$th time step] in Eq. (11), we must know all the values at the right-hand side of Eq. (11). The matrices **A** and **B** in Eq. (11) are composed of coefficients that are already defined in the process of differential approximation. The last term, $F_{i+1,j+1}$, is the external disturbance and is assumed to be known at all the time steps and spatial meshes. Noticing that the spatial and time differential values, $\{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{(2p)}\}_{i,j+1}$ and $\{u \quad \dot{u} \quad \ddot{u}\}_{i+1,j}$, in Eq. (11) are the values at the one previous spatial mesh and time step, Eq. (11) can be iteratively calculated if all the initial and boundary conditions are given. However, examining the Eq. (11), it has $2p + 1$ unknowns but a $2p$-order partial differential equation has only p boundary conditions at one boundary point, for example p boundary conditions at $i = 0$. The remain $p + 1$ unknowns should be determined before to solve Eq. (11) iteratively. In this work, the $p + 1$ unknowns are estimated from the known p boundary conditions at the other end boundary point, i = end and the original governing equation, Eq. (1).

Substituting end-1 into i in Eq. (11) results

$$\begin{aligned} u_{\text{end},j+1} &= [A_0 \quad A_1 \quad A_2 \quad \cdots \quad A_{2p}] \\ &\quad \{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{(2p)}\}^T_{\text{end}-1,j+1} \\ &\quad + \mathbf{B}\{u \quad \dot{u} \quad \ddot{u}\}^T_{\text{end},j} + F_{\text{end},j+1}. \end{aligned} \tag{12}$$

In Eq. (12), the second and the third term are already known because the third term is external disturbance and assumed to be known at all the space and time steps, and the second term is the time differential values at the one previous time step and already known.

As explained in Eq. (8), the partial differentials of any order at the $(i + 1)$th spatial mesh can be expressed with the differentials at the ith spatial mesh. Using this relationship, the partial differentials at the (end-1)th spatial mesh can be expressed with the differentials at the (end-2)th mesh like Eq. (13):

$$\begin{aligned} u^{(2p)}_{\text{end}-1,j+1} &= c_{2p,0}(u_{\text{end}-1,j+1} - u_{\text{end}-2,j+1}) - c_{2p,1}u^{(1)}_{\text{end}-2,j+1} - c_{2p,2}u^{(2)}_{\text{end}-2,j+1} - \cdots - c_{2p,2p}u^{(2p)}_{\text{end}-2,j+1} \\ u^{(2p-1)}_{\text{end}-1,j+1} &= c_{2p-1,0}(u_{\text{end}-1,j+1} - u_{\text{end}-2,j+1}) - c_{2p-1,1}u^{(1)}_{\text{end}-2,j+1} - c_{2p-1,2}u^{(2)}_{\text{end}-2,j+1} - \cdots - c_{2p-1,2p}u^{(2p)}_{\text{end}-2,j+1} \\ &\vdots \\ u^{(2)}_{\text{end}-1,j+1} &= c_{2,0}(u_{\text{end}-1,j+1} - u_{\text{end}-2,j+1}) - c_{2,1}u^{(1)}_{\text{end}-2,j+1} - c_{2,2}u^{(2)}_{\text{end}-2,j+1} - \cdots - c_{2,2p}u^{(2p)}_{\text{end}-2,j+1} \\ u^{(1)}_{\text{end}-1,j+1} &= c_{1,0}(u_{\text{end}-1,j+1} - u_{\text{end}-2,j+1}) - c_{1,1}u^{(1)}_{\text{end}-2,j+1} - c_{1,2}u^{(2)}_{\text{end}-2,j+1} - \cdots - c_{1,2p}u^{(2p)}_{\text{end}-2,j+1}. \end{aligned} \tag{13}$$

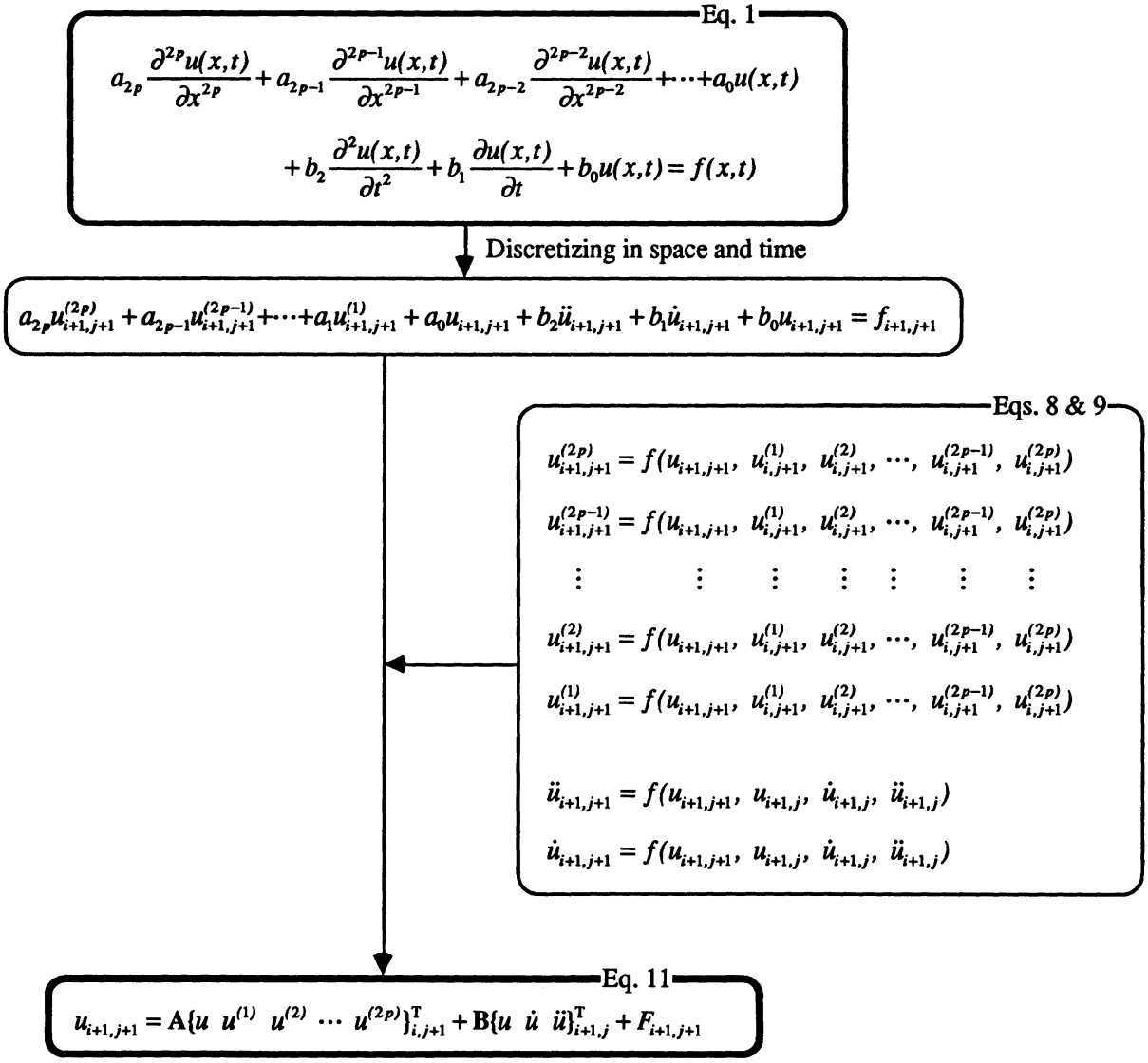


FIGURE 2 Discretizing procedures for a partial differential equation.

Substituting Eq. (13) into Eq. (12) gives

$$
\begin{aligned}
u_{\mathrm{end},j+1} = {} & A_0 u_{\mathrm{end}-1,j+1} \\
& + A_1\ [c_{1,0}(u_{\mathrm{end}-1,j+1} - u_{\mathrm{end}-2,j+1}) - c_{1,1}u^{(1)}_{\mathrm{end}-2,j+1} - c_{1,2}u^{(2)}_{\mathrm{end}-2,j+1} - \cdots - c_{1,2p}u^{(2p)}_{\mathrm{end}-2,j+1}] \\
& + A_2\ [c_{2,0}(u_{\mathrm{end}-1,j+1} - u_{\mathrm{end}-2,j+1}) - c_{2,1}u^{(1)}_{\mathrm{end}-2,j+1} - c_{2,2}u^{(2)}_{\mathrm{end}-2,j+1} - \cdots - c_{2,2p}u^{(2p)}_{\mathrm{end}-2,j+1}] \\
& \ \ \vdots \\
& + A_{2p}[c_{2p,0}(u_{\mathrm{end}-1,j+1} - u_{\mathrm{end}-2,j+1}) - c_{2p,1}u^{(1)}_{\mathrm{end}-2,j+1} - c_{2p,2}u^{(2)}_{\mathrm{end}-2,j+1} - \cdots - c_{2p,2p}u^{(2p)}_{\mathrm{end}-2,j+1}] \\
& + \mathbf{B}\{u\ \ \dot{u}\ \ \ddot{u}\}^{T}_{\mathrm{end},j} + F_{\mathrm{end},j+1}.
\end{aligned}
\tag{14}
$$

Likewise, substituting end-2 into i in Eq. (11) gives

$$
u_{\mathrm{end}-1,j+1} = \mathbf{A}\{u\ \ u^{(1)}\ \ u^{(2)}\ \ \cdots\ \ u^{(2p)}\}^{T}_{\mathrm{end}-2,j+1} + \mathbf{B}\{u\ \ \dot{u}\ \ \ddot{u}\}^{T}_{\mathrm{end}-1,j} + F_{\mathrm{end}-1,j+1}. \tag{15}
$$

and substituting Eq. (15) into Eq. (14) gives

$$
\begin{aligned}
u_{\text{end},j+1} = {} & [(A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})A_0 - (A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})]\, u_{\text{end}-2,j+1} \\
& + [(A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})A_1 - (A_1c_{1,1} + A_2c_{2,1} + \cdots + A_{2p}c_{2p,1})]\, u^{(1)}_{\text{end}-2,j+1} \\
& + [(A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})A_2 - (A_1c_{1,2} + A_2c_{2,2} + \cdots + A_{2p}c_{2p,2})]\, u^{(2)}_{\text{end}-2,j+1} \\
& \quad \vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\
& + [(A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})A_{2p} - (A_1c_{1,2p} + A_2c_{2,2p} + \cdots + A_{2p}c_{2p,2p})]\, u^{(2p)}_{\text{end}-2,j+1} \\
& + \mathbf{B}\{u \quad \dot{u} \quad \ddot{u}\}^T_{\text{end},j} + (A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})\mathbf{B}\{u \quad \dot{u} \quad \ddot{u}\}^T_{\text{end}-1,j} \\
& + F_{\text{end},j+1} + (A_0 + A_1c_{1,0} + A_2c_{2,0} + \cdots + A_{2p}c_{2p,0})F_{\text{end}-1,j+1}. \qquad (16)
\end{aligned}
$$

Equation (16) says that $u_{\text{end},j+1}$ can be written in terms of the displacement and the differentials at the (end-2)th spatial mesh, $\{u, u^{(1)}, \ldots, u^{(2p-1)}, u^{(2p)}\}_{\text{end}-2,j+1}$. It is worthwhile to note that the form of Eq. (16) is just the same as that of Eq. (12), because the last four terms at the right-hand side of Eq. (16) are already known values. The third and fourth terms are determined at the jth time step and the last two terms are known external disturbances. Repeating these procedures continuously, $u_{\text{end},j+1}$ can be expressed in terms of the differentials at the other boundary point, $\{u, u^{(1)}, \ldots, u^{(2p-1)}, u^{(2p)}\}_{0,j+1}$, as Eq. (17):

$$
\begin{aligned}
u_{\text{end},j+1} = {} & [\tilde{A}_0 \quad \tilde{A}_1 \quad \tilde{A}_2 \quad \cdots \quad \tilde{A}_{2p}] \\
& \{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{2p)}\}^T_{0,j+1} \qquad (17) \\
& + \text{known time differentials and external disturbances}
\end{aligned}
$$

where $\tilde{A}_0, \tilde{A}_1, \tilde{A}_2, \ldots, \tilde{A}_{2p}$ are constants resulting from the operations of all the coefficients defined in the process of differential approximation. Equation (17) relates the states at two different boundary points, i = end and $i = 0$ as shown in Fig. 3.

Substituting end-1 into i in Eq. (8), the nth or-

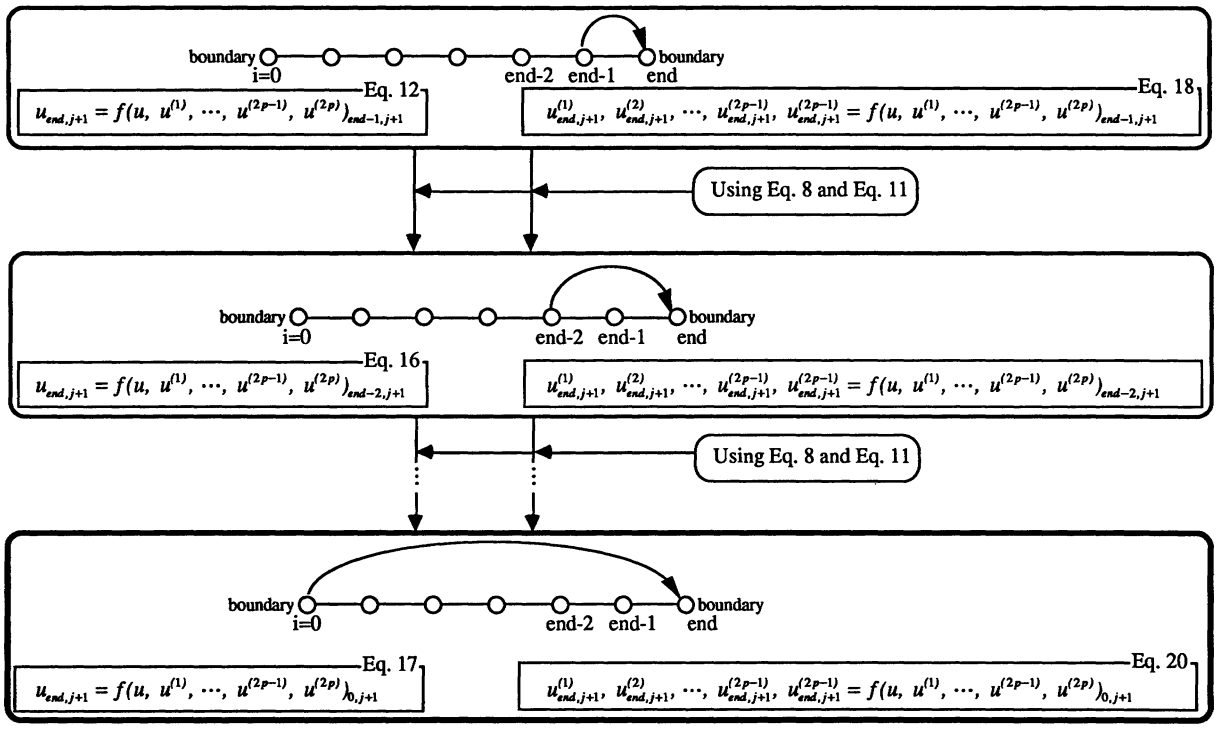


FIGURE 3 Derivation of relationship between the states at the two boundary points $i = 0$ and j = end.

der differential at i = end can be written as Eq. (18):

$$\begin{aligned} u^{(n)}_{\text{end},j+1} = {} & c_{n,0}(u_{\text{end},j+1} - u_{\text{end}-1,j+1}) \\ & - c_{n,1}u^{(1)}_{\text{end}-1,j+1} - c_{n,2}u^{(2)}_{\text{end}-1,j+1} \\ & - \cdots - c_{n,2p}u^{(2p)}_{\text{end}-1,j+1} \end{aligned} \tag{18}$$

where $1 \le n \le 2p$. Using Eq. (12), Eq. (18) can be written as Eq. (19):

$$\begin{aligned} u^{(n)}_{\text{end},j+1} = {} & (c_{n,0}A_0 - c_{n,0})u_{\text{end}-1,j+1} \\ & + (c_{n,0}A_1 - c_{n,1})u^{(1)}_{\text{end}-1,j+1} \\ & + (c_{n,0}A_2 - c_{n,2})u^{(2)}_{\text{end}-1,j+1} \\ & + \cdots + (c_{n,0}A_{2p} - c_{n,2p})u^{(2p)}_{\text{end}-1,j+1} \\ & + c_{n,0}\mathbf{B}\{u \quad \dot{u} \quad \ddot{u}\}^T_{\text{end},j} + c_{n,0}F_{\text{end},j+1}. \end{aligned} \tag{19}$$

Equation (19) says that the nth order differential at i = end point can be expressed with the displacement and their differentials at i = (end-1) point, because the last two terms are all known values. Continuously repeating the above procedures results in Eq. (20), which relates the differentials at one end point with those at the other end point.

$$\begin{aligned} u^{(n)}_{\text{end},j+1} = {} & [\hat{A}_0 \quad \hat{A}_1 \quad \hat{A}_2 \quad \cdots \quad \hat{A}_{2p}] \\ & \{u \quad u^{(1)} \quad u^{(2)} \quad \cdots \quad u^{2p)}\}^T_{0,j+1} \\ & + \text{known time differentials and} \\ & \quad \text{external disturbances} \end{aligned} \tag{20}$$

where $1 \le n \le 2p$ and $\hat{A}_0, \hat{A}_1, \hat{A}_2, \ldots, \hat{A}_{2p}$ are the constants obtained from the operation of coefficients defined in the process of differential approximation.

Equations (17) and (20) show the relationship between the states at the two boundary points: $2p + 1$ states at each boundary point, $\{u, u^{(1)}, u^{(2)}, \ldots, u^{(2p-1)}, u^{(2p)}\}_{\text{end},j+1}$ and $\{u, u^{(1)}, \ldots, u^{(2p-1)}, u^{(2p)}\}_{0,j+1}$. Figure 3 summarizes the procedures to project the states at one boundary point (i = 0) to those at the other boundary point (i = end). Among the total $4p + 2$ states at boundaries, the given boundary conditions are only $2p$ (p boundary conditions at each boundary), and the number of derived relationships in Eqs. (17) and (20) is $2p + 1$. So one more equation is needed to calculate $u_{i+1,j+1}$ from Eq. (11).

The last algebraic relationship between the states at the boundary point i = 0 can be derived from the governing equation, Eq. (1). Evaluating Eq. (1) at the spatial mesh i = 0 and time mesh (j + 1)th step gives

$$\begin{aligned} & a_{2p}u^{(2p)}_{0,j+1} + a_{2p-1}u^{(2p-1)}_{0,j+1} + \cdots + a_1u^{(1)}_{0,j+1} \\ & \quad + a_0u_{0,j+1} + b_2\ddot{u}_{0,j+1} + b_1\dot{u}_{0,j+1} \\ & \quad + b_0u_{0,j+1} = f_{0,j+1}. \end{aligned} \tag{21}$$

Substituting Eqs. (9a,b) into $\ddot{u}_{0,j+1}$, $\dot{u}_{0,j+1}$ and rearranging yields:

$$\begin{aligned} & f_{0,j+1} + (b_2d_{2,0} + b_1d_{1,0})u_{0,j} \\ & \quad + (b_2d_{2,1} + b_1d_{1,0})\dot{u}_{0,j} + (b_2d_{2,2} + b_1d_{1,2})\ddot{u}_{0,j} \\ & = a_{2p}u^{(2p)}_{0,j+1} + a_{2p-1}u^{(2p-1)}_{0,j+1} + \cdots + a_1u^{(1)}_{0,j+1} \\ & \quad + (a_0 + b_2d_{2,0} + b_1d_{1,0} + b_0)u_{0,j+1}. \end{aligned} \tag{22}$$

In Eq. (22), all the left-hand side are known values. Therefore, Eqs. (17), (20), and (22) provides $2p + 2$ equations and the known $2p$ boundary conditions at two boundary points are sufficient to get $4p + 2$ states at two boundary points. Further, those calculated $4p + 2$ boundary states can be used to calculate the displacement at an arbitrary time and spatial mesh, $u_{i+1,j+1}$, from Eq. (11).

Figure 4 explains the procedures to calculate $u_{i+1,j+1}$ when both initial and boundary conditions are given. At the starting step, i = 0 and j = 0, the initial conditions will give the values of $u_{1,0}$, $\dot{u}_{1,0}$, and $\ddot{u}_{1,0}$ and the boundary conditions and Eqs. (17), (20), and (22) will generate the values of $u_{0,1}, u^{(1)}_{0,1}, \ldots, u^{(2p-1)}_{0,1}, u^{(2p)}_{0,1}$. Then plugging in the obtained values $u_{1,0}$, $\dot{u}_{1,0}$, $\ddot{u}_{1,0}$, $u_{0,1}$, $u^{(1)}_{0,1}, \ldots, u^{(2p-1)}_{0,1}$, and $u^{(2p)}_{0,1}$ into Eq. (11), $u_{1,1}$ can be calculated. After calculating $u_{1,1}$, all the partial differentials of $u_{1,1}, u^{(1)}_{1,1}, \ldots, u^{(2p-1)}_{1,1}, u^{(2p)}_{1,1}$, and $\dot{u}_{1,1}$, $\ddot{u}_{1,1}$ can be obtained from Eqs. (8) and (9a,b). Then $u_{2,1}$ (i = 1 and j = 0) can be computed using the differentials of $u_{1,1}$ and the initial conditions, $u_{2,0}$, $\dot{u}_{2,0}$, and $\ddot{u}_{2,0}$. Continuing these procedures up to i = (end-1), then the entire motion at the first time step, $u_{i+1,1}$, $i = 0, 1, \ldots,$ end $-$ 1, is determined. Next, increasing the time step from j = 0 to j = 1, $u_{1,2}$ can be computed using the boundary conditions ($u_{0,2}, u^{(1)}_{0,2}, u^{(2)}_{0,2}, \ldots, u^{(2p)}_{0,2}$) and the states at the previous time step ($u_{1,1}, \dot{u}_{1,1}, \ddot{u}_{1,1}$). Continuing these procedures gives the entire motion at the second time step, $u_{i+1,2}$, $i = 0, 1, \ldots,$ end $-$ 1. Therefore, generally the motion at the (i + 1)th spatial mesh and (j + 1)th time step, $u_{i+1,j+1}$, can be obtained.

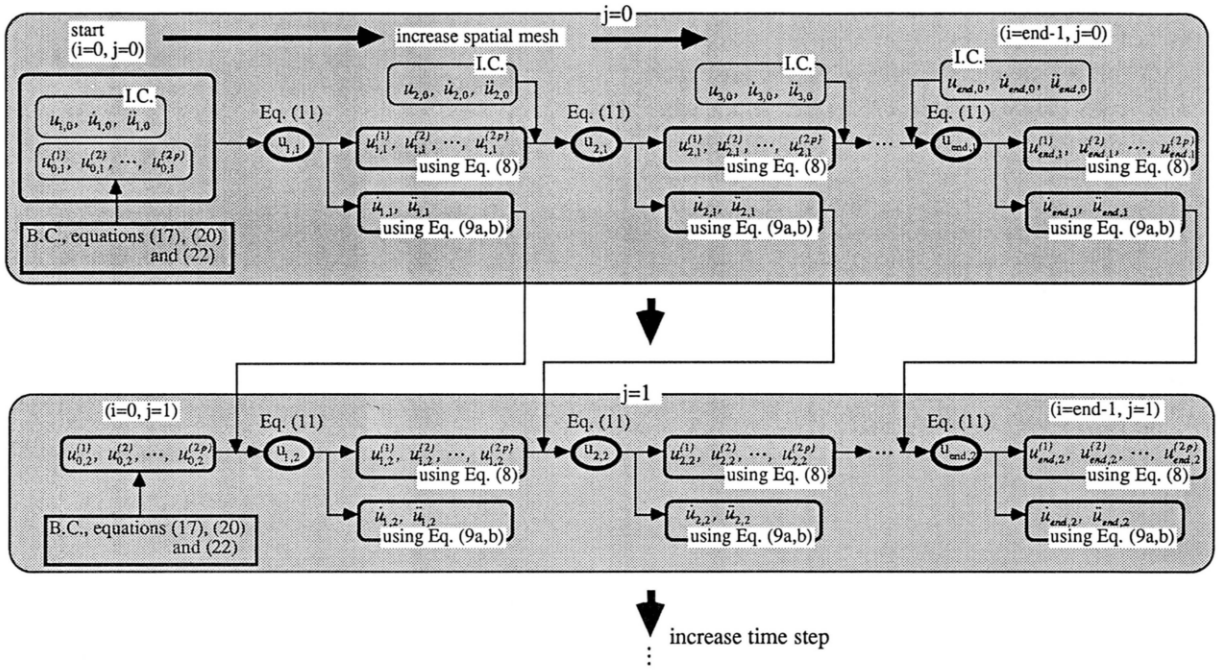


FIGURE 4 The calculation procedures for $u_{i+1,j+1}$.

Stability, Consistency, and Convergence

For stability analysis of the proposed method, a second-order hyperbolic equation $u'' = \ddot{u}$ is chosen because this equation is the simplest form in structural dynamic problems. When applying this proposed algorithm to the hyperbolic equation, the equation can be transformed into such a finite difference equation as follows:

$$u_{i+1,j+1} = \frac{1}{c_{2,0} - d_{2,0}} [c_{2,0} \quad c_{2,1} \quad c_{2,2}] \begin{Bmatrix} u \\ u' \\ u'' \end{Bmatrix}_{i,j+1} - \frac{1}{c_{2,0} - d_{2,0}} [d_{2,0} \quad d_{2,1} \quad d_{2,2}] \begin{Bmatrix} u \\ \dot{u} \\ \ddot{u} \end{Bmatrix}_{i+1,j} \tag{23}$$

when u'' and $\ddot{u}$ are linearly approximated, the used coefficients will become

$$[c_{2,0} \quad c_{2,1} \quad c_{2,2}] = \left[\frac{6}{\Delta x^2} \quad \frac{6}{\Delta x} \quad 2\right],$$

$$[d_{2,0} \quad d_{2,1} \quad d_{2,2}] = \left[\frac{6}{\Delta t^2} \quad \frac{6}{\Delta t} \quad 2\right]$$

and when those differentiations are approximated by averaged values, the coefficients will be

$$[c_{2,0} \quad c_{2,1} \quad c_{2,2}] = \left[\frac{4}{\Delta x^2} \quad \frac{4}{\Delta x} \quad 1\right],$$

$$[d_{2,0} \quad d_{2,1} \quad d_{2,2}] = \left[\frac{4}{\Delta t^2} \quad \frac{4}{\Delta t} \quad 1\right].$$

For the Fourier or von Neumann stability analysis (Ames, 1977; Anderson et al., 1984), Eq. (23) should be written in an alternative finite difference form. Evaluating the hyperbolic equation $u'' = \ddot{u}$ at spatial meshes x_{i-1}, x_i, x_{i+1} and at time steps t_{j-1}, t_j, t_{j+1}, and rearranging them using Eqs. (8) and (9a,b), the alternative finite difference form of Eq. (23) can be obtained after very long and complex but very straightforward operations:

$$\begin{aligned} &(\nu^2\beta_t - \beta_x) \\ &\quad (u_{i+1,j+1} - u_{i-1,j+1} + u_{i+1,j-1} - u_{i-1,j-1}) \\ &\quad - \{2(\nu^2\beta_t - \beta_x) + 1\}(u_{i,j+1} + u_{i,j-1}) \\ &\quad + \{\nu^2 - 2(\nu^2\beta_t - \beta_x)\}(u_{i+1,j} + u_{i-1,j}) \\ &\quad + \{4(\nu^2\beta_t - \beta_x) - 2(\nu^2 - 1)\}u_{i,j} = 0 \end{aligned} \tag{24}$$

where $\nu^2 = \Delta t^2/\Delta x^2$; β_x and β_t = coefficients determined by the used approximation method for the highest order partial differential in the spatial and time domains, respectively (β_x and $\beta_t = \frac{1}{6}$ when u'' or $\ddot{u}$ is linearly approximated; β_x and $\beta_t = \frac{1}{4}$ when u'' or $\ddot{u}$ is approximated by averaged values). It is inevitable that some error, ε, is included in the computed solution, $u_{i,j}$, due to round-off errors. Then the computed solution, $u_{i,j}$, is composed of the exact solution, $(u_{i,j})_{\text{exact}}$, and the inevitable error, ε. Recognizing that $(u_{i,j})_{\text{exact}}$ should satisfy the above Eq. (24), the error, ε, is the only term to check the equality of Eq. (24). In stability analysis, it is common to check the boundness of Eq. (24) when the Eq. (24) is perturbed with some error $\varepsilon(x, t)$.

Series expansion of $\varepsilon(x, t)$ gives

$$\varepsilon(x, t) = \sum_i e^{\alpha t} e^{\sqrt{-1}\omega i \Delta x} \tag{25}$$

where ω is real but α can be a complex value. Because Eq. (24) is linear, the superposition law holds true and it is sufficient to investigate the boundness of Eq. (24) with just any one term of series expansion of error $\varepsilon(x, t)$. Choosing a single term of $\varepsilon(x, t)$, $\varepsilon_{i,j} = e^{\alpha j \Delta t} e^{\sqrt{-1}\omega i \Delta x}$, and plugging it into $u_{i,j}$ in Eq. (24) results

$$e^{\alpha \Delta t} + e^{-\alpha \Delta t} = 2\left(1 - \frac{2\nu^2 \sin^2 \dfrac{\omega \Delta x}{2}}{1 + 4(\nu^2\beta_t - \beta_x)\sin^2 \dfrac{\omega \Delta x}{2}}\right). \tag{26}$$

In order to avoid the divergence of error as j goes to an infinite, it is necessary that $|e^{\alpha \Delta t}| \le 1$ for all real values of ω. Equation (26) is the second-order polynomial equation about $e^{\alpha \Delta t}$ and the value of $e^{\alpha \Delta t}$ can be easily obtained by solving Eq. (26). Noticing that the multiplication of the two roots of Eq. (26) is always 1 means that one root of Eq. (26) is always larger and this makes the equation unstable. In order to be stable, the discriminant of Eq. (26) should be nonpositive.

$$\left(1 - \frac{2\nu^2 \sin^2 \dfrac{\omega \Delta x}{2}}{1 + 4(\nu^2\beta_t - \beta_x)\sin^2 \dfrac{\omega \Delta x}{2}}\right)^2 - 1 \le 0. \tag{27}$$

When the highest order partial differential is approximated by averaged value, the stability follows as unconditionally stable for all real values ω. However, in the case that the highest order partial differential is linearly approximated in the time domain, the stability follows for all real values ω if

$$\nu^2 \le \frac{3}{\sin^2 \dfrac{\omega \Delta x}{2}} - 2,$$

and this is always true when $\nu^2 \le 1$. The amplification factors and the relative phase errors for the proposed method are plotted in Fig. 5.

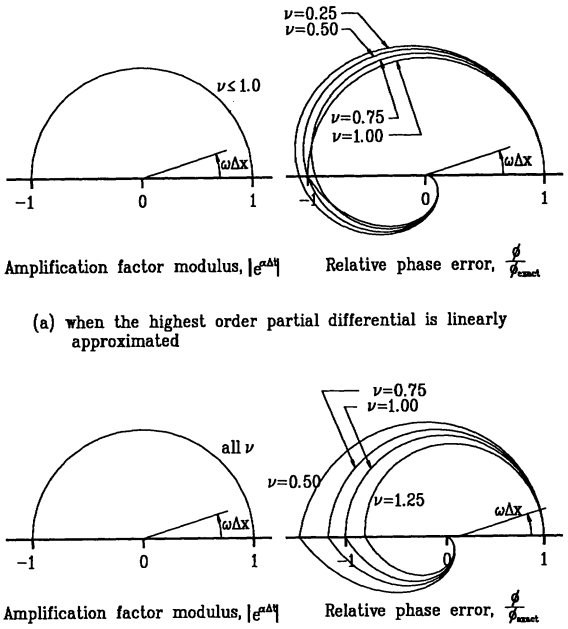


FIGURE 5 Amplification factors and relative phase errors for the proposed method.

Also substituting the exact solution obtained by the Taylor series expansion for functions of one and two variables into Eq. (24) gives

$$\left(\frac{\partial^2 u}{\partial x^2}\bigg|_{i,j} - \frac{\partial^2 u}{\partial t^2}\bigg|_{i,j}\right) + \left[\frac{\Delta x^2}{12}\frac{\partial^4 u}{\partial x^4}\bigg|_{i,j} - \frac{\Delta t^2}{12}\frac{\partial^2 u}{\partial t^4}\bigg|_{i,j} - (\Delta x^2\beta_x - \Delta t^2\beta_t)\frac{\partial^4 u}{\partial x^2\,\partial t^2}\bigg|_{i,j}\right] + O(\Delta x^4, \Delta t^4) = 0. \tag{28}$$

For Eq. (28), in this case, the order of the truncation error is $O[\Delta x^2, \Delta t^2]$. The truncation error vanishes as the mesh is refined; thus it can be

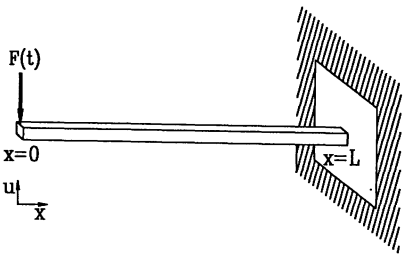


FIGURE 6 Test example: cantilever beam.

said that the proposed numerical algorithm is consistent and second-order accurate.

Because the stability and consistency are verified, the convergence of the proposed algorithm can be guaranteed by the Lax's equivalence theorem (Anderson et al., 1984).

NUMERICAL EXAMPLE

In order to verify the validity of the proposed scheme, a cantilever beam as shown in Fig. 6 was chosen to numerically calculate an undamped forced response. The test results were compared with the responses obtained from well proven FDM.

The partial differential equation describing the motion of a cantilever beam in Fig. 6 is

$$EIu'''' + \rho A\ddot{u} = 0 \tag{29}$$

where E is Young's modulus (2×10^{11}); I is moment of area ($0.003^4/12$ m^4); ρ is material density (7800 kg/m^3), A is square cross-section area (0.003^2 m^2); and L is length (1 m). When a vertical harmonic excitation force is applied at the free end ($x = 0$), the initial and boundary conditions (IC, BC) are as follows:

$$\begin{aligned} &\text{IC: } u(x, 0) = \dot{u}(x, 0) = 0 \\ &\text{BC: } u''(0, t) = 0, \quad u'''(0, t) = \frac{\sin(2\pi \times 20t)}{EI}, \\ &\qquad u(L, t) = u'(L, t) = 0. \end{aligned} \tag{30}$$

The exact solution (Weaver, Timoshenko, and Young, 1990) of the test problem is

$$u(0, t) = \frac{4L^3}{EI} \sum_{l=1}^{\infty} \frac{1}{(k_l L)^4 (1 - (40\pi)^2/\Omega_l^2)} \left[\sin(40\pi t) - \frac{40\pi}{\Omega_l} \sin(\Omega_l t)\right] \tag{31}$$

where $\Omega_l = \sqrt{EI/\rho A}\, k_l^2$, ($l = 1, 2, 3, \ldots$), and k_l = solutions of a frequency equation, $\cos(k_l L)\cosh(k_l L) = -1$. Using Eq. (8), the highest order partial differential with respect to x, $u''''(x, t)$, is written as:

$$\begin{aligned} u''''_{i+1,j+1} = {} & c_{4,0}(u_{i+1,j+1} - u_{i,j+1}) - c_{4,1}u'_{i,j+1} \\ & - c_{4,2}u''_{i,j+1} - c_{4,3}u'''_{i,j+1} \\ & - c_{4,4}u''''_{i,j+1}. \end{aligned} \tag{32}$$

Also, from Eq. (9a), the highest order partial differential with respect to time, $\ddot{u}(x, t)$, is written as:

$$\begin{aligned} \ddot{u}_{i+1,j+1} = {} & d_{2,0}(u_{i+1,j+1} - u_{i+1,j}) \\ & - d_{2,1}\dot{u}_{i+1,j} - d_{2,2}\ddot{u}_{i+1,j}. \end{aligned} \tag{33}$$

In this case, the coefficients, c and d, in Eqs. (32) and (33) are defined as follows when those are approximated by average value:

$$\begin{aligned} &[c_{4,0} \quad c_{4,1} \quad c_{4,2} \quad c_{4,3} \quad c_{4,4}] \\ &\quad = \left[\frac{48}{\Delta x^4} \quad \frac{48}{\Delta x^3} \quad \frac{24}{\Delta x^2} \quad \frac{8}{\Delta x} \quad 1\right] \\ &[d_{2,0} \quad d_{2,1} \quad d_{2,2}] \\ &\quad = \left[\frac{4}{\Delta t^2} \quad \frac{4}{\Delta t} \quad 1\right]. \end{aligned} \tag{34}$$

Therefore, substituting Eqs. (32) and (33) into Eq. (29), the transformed difference equation can be written as follows:

$$\begin{aligned} u_{i+1,j+1} = {} & \frac{EI}{EIc_{4,0} + \rho A d_{2,0}} \\ & [c_{4,0} \quad c_{4,1} \quad c_{4,2} \quad c_{4,3} \quad c_{4,4}] \\ & \{u \quad u' \quad u'' \quad u''' \quad u''''\}^T_{i,j+1} \\ & + \frac{\rho A}{EIc_{4,0} + \rho A d_{2,0}} \\ & [d_{2,0} \quad d_{2,1} \quad d_{2,2}]\{u \quad \dot{u} \quad \ddot{u}\}^T_{i+1,j}. \end{aligned} \tag{35}$$

In order to solve $u_{1,j+1}$, all $\{u \quad u' \quad u'' \quad u''' \quad u''''\}_{0,j+1}$ and $\{u \quad \dot{u} \quad \ddot{u}\}_{1,j}$ should be known. However, only $u''_{0,j+1}$ and $u'''_{0,j+1}$ are known boundary values listed in Eq. (30), and $u_{0,j+1}$, $u'_{0,j+1}$, and $u''''_{0,j+1}$ are unknowns to be computed using the other end boundary values at $i = L$. The boundary values at $i = L$ are $u_{L,j+1} = 0$ and $u'_{L,j+1} = 0$ and these boundary conditions should be related

to the states at $i = 0$. The relationships obtained from Eqs. (12)–(17) and from Eqs. (18)–(20) are

$$\begin{aligned} u_{L,j+1} &= 0 \\ &= [\tilde{A}_0 \quad \tilde{A}_1 \quad \tilde{A}_2 \quad \tilde{A}_3 \quad \tilde{A}_4] \\ &\quad \{u \quad u' \quad u'' \quad u''' \quad u''''\}^T_{0,j+1} \\ &\quad + \text{known values at all time domain} \end{aligned} \tag{36}$$

$$\begin{aligned} u'_{L,j+1} &= 0 \\ &= [\hat{A}_0 \quad \hat{A}_1 \quad \hat{A}_2 \quad \hat{A}_3 \quad \hat{A}_4] \\ &\quad \{u \quad u' \quad u'' \quad u''' \quad u''''\}^T_{0,j+1} \\ &\quad + \text{known values at all time domain.} \end{aligned} \tag{37}$$

Equation (29), the governing equation, satisfies the discretized equation as:

$$EIu''''_{0,j+1} + \rho A\ddot{u}_{0,j+1} = 0. \tag{38}$$

Using Eq. (33), the above equation can be rewritten as follows:

$$\rho A(d_{2,0}u_{0,j} + d_{2,1}\dot{u}_{0,j} + d_{2,2}\ddot{u}_{0,j}) = EIu''''_{0,j+1} + \rho A d_{2,0}u_{0,j+1} \tag{39}$$

where all terms on the left-hand side are known values because those are the states at one previous time step j. The three unknown boundary states at $i = 0$ ($u_{0,j+1}$, $u'_{0,j+1}$, and $u''''_{0,j+1}$) can be calculated using the (Eqs. (36), (37), and (39). The procedures to calculate the unknown boundary states are explained in detail in Fig. 7. After that, the motion at an arbitrary space and time step, $u_{i+1,j+1}$ can be obtained by continuous calculations while increasing the space and time step.

In order to compare the results from this proposed algorithm, the same problem was also tested with the well known central difference method. The same equation was discretized by explicit and implicit central difference methods. Equation (40) is the equation discretized by the explicit central difference method and Eq. (41) is that by the time-centered implicit central difference method (Crank–Nicolson method). Equation (42) lists the initial and boundary conditions.

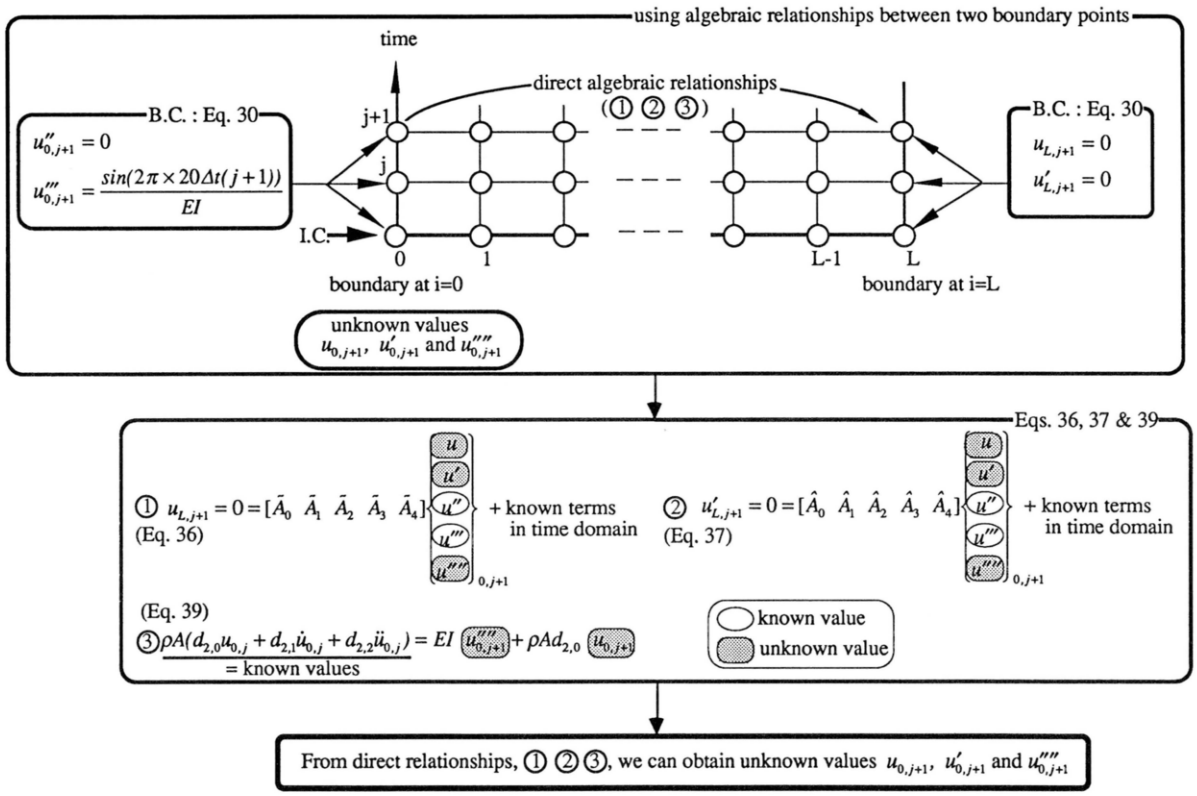


FIGURE 7 Calculating procedures for unknown boundary values at $i = 0$.

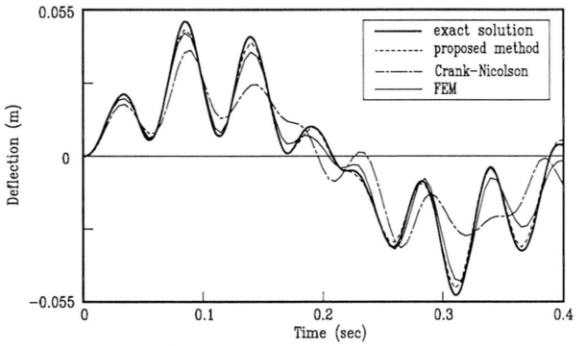


FIGURE 8 Comparison of the exact and numerical solutions.

$$u_{i,j+1} = -[v^2 \quad -4v^2 \quad (6v^2 - 2) \quad -4v^2 \quad v^2] \{u_{i-2} \quad u_{i-1} \quad u_i \quad u_{i+1} \quad u_{i+2}\}_j^T - u_{i,j-1} \tag{40}$$

where $v = \sqrt{(EI/\rho A)} \cdot \Delta t/\Delta x^2$,

$$[v^2 \quad -4v^2 \quad (6v^2 + 2) \quad -4v^2 \quad v^2] \{u_{i-2} \quad u_{i-1} \quad u_i \quad u_{i+1} \quad u_{i+2}\}_{j+1}^T = 4u_{i,j} - [v^2 \quad -4v^2 \quad (6v^2 + 2) \quad -4v^2 \quad v^2] \{u_{i-2} \quad u_{i-1} \quad u_i \quad u_{i+1} \quad u_{i+2}\}_{j-1}^T \tag{41}$$

$$\text{IC: } u_{i,0} = 0 \quad \text{and} \quad u_{i,-1} = u_{i,1}$$

$$\text{BC: } u_{-1,j} = 2u_{0,j} - u_{1,j} \quad \text{and}$$

$$u_{-2,j} = 2u_{-1,j} - 2u_{1,j} + u_{2,j} - \frac{2\Delta x^3}{EI} \sin(2\pi \times 20\Delta t \times j)$$

$$u_{L,j} = 0 \quad \text{and} \quad u_{L-1,j} = u_{L+1,j}. \tag{42}$$

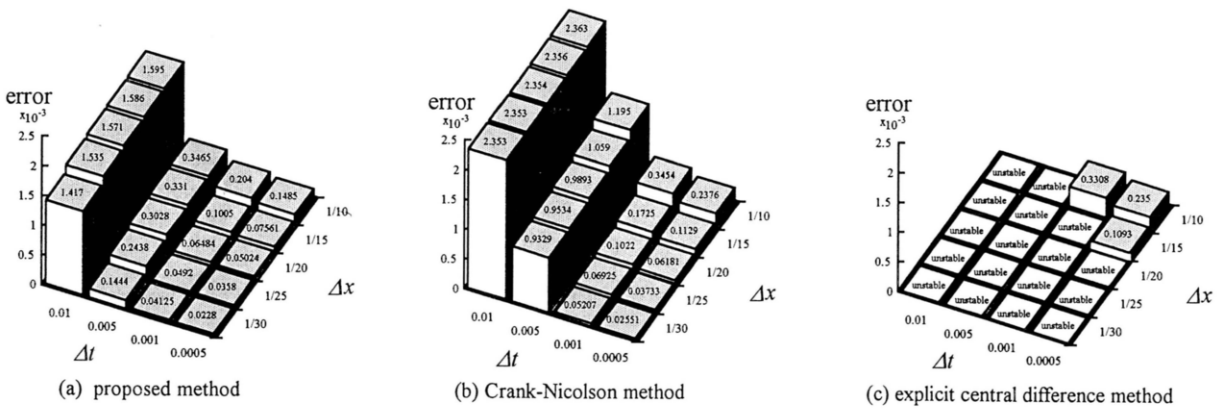
The calculated responses at the free end ($x = 0$) are plotted in Fig. 8. In order to compare the results with that from central difference methods and FEM, one of the worst computational conditions, 10 spatial mesh and $\Delta t = 5 \times 10^{-3}$ s, is chosen and simulated. In this case, v (defined as $\sqrt{(EI/\rho A)} \cdot \Delta t/\Delta x^2$) becomes 2.193, and the explicit central difference method cannot give a good result due to the inherent instability. It is known that the explicit central difference method is conditionally stable when $v \leq 1/2$ in this problem. It is observed in Fig. 8 that the solution from the proposed numerical method agrees very well with the exact value obtained from Eq. (31). When defining the amount of error as Eq. (43), the errors of each numerical method was compared as varying space and time step, Δx and Δt.

$$\text{error} = \frac{1}{N} \sqrt{\sum_{j=0}^{(N\Delta t)} [u_{0,j}|_{\text{exact sol.}} - u_{0,j}|_{\text{numerical sol.}}]^2} \tag{43}$$

where $N = \tau/\Delta t$, τ is the first natural period. Figure 9 illustrates the amount of error of each numerical method. Excluding the case of the explicit central difference, the proposed algorithm is more accurate to the exact solution than the Crank–Nicolson method for all the cases of the space and time step. The explicit central difference method showed instabilities for most cases of Δx and Δt and could not compare its error with that of other methods.

CONCLUSION

An improved finite difference type numerical method to solve the partial differential equations



(a) proposed method (b) Crank-Nicolson method (c) explicit central difference method

FIGURE 9 Comparison of the error of each numerical method as varying Δx and Δt.

was proposed in this study. This numerical algorithm belongs to a single-step, second-order accurate and implicit method. The stability, consistency, and convergence were examined analytically with the second-order hyperbolic type partial differential equation. The proposed numerical method can directly satisfy the natural boundary conditions. Thus the interpretation of the physical meanings of partial differential terms is simple, because the transformed governing equation is constructed with the displacement and known partial differential terms. Therefore, the proposed numerical method is more appropriate for 1-D structural dynamic numerical analysis than existing finite difference methods and is also easily applicable to high order partial differential equations. Both the coding and remeshing are simple. From the numerical example, it was found that the proposed method is more accurate and effective than well known central difference methods to calculate 1-D structural dynamic motion.

REFERENCES

Ames, W. F., 1977, *Numerical Methods for Partial Differential Equations,* 2nd ed., Academic Press, New York.

Anderson, P. D., Tannehill, J. C., and Pletcher, R. H., 1984, *Computational Fluid Mechanics and Heat Transfer,* McGraw–Hill, New York.

Bouloutas, E. T., and Celia, M. A., 1991, "An Improved Cubic Petrov–Galerkin Method for Simulation of Advection–Diffusion Processes in Rectangularly Decomposable Domains," *Computer Methods in Applied Mechanics and Engineering,* Vol. 92, pp. 289–308.

Choe, K. Y., and Holsapple, K. A., 1992, "The Discontinuous Finite Element Method with the Taylor–Galerkin Approach for Nonlinear Hyperbolic Conservation Laws," *Computer Methods in Applied Mechanics and Engineering,* Vol. 95, pp. 141–167.

Dokainish, M. A., and Subbaraj, K., 1989, "A Survey of Direct Time-Integration Methods in Computational Structural Dynamics-I. Explicit Methods," *Computers & Structures,* Vol. 32, pp. 1371–1386.

Donea, J., 1984, "A Taylor–Galerkin Method for Convective Transport Problem," *International Journal for Numerical Methods in Engineering,* Vol. 20, pp. 101–119.

Hulbert, G. M., and Hughes, T. J. R., 1990, "Space-Time Finite Element Methods for Second-Order Hyperbolic Equations," *Computer Methods in Applied Mechanics and Engineering,* Vol. 84, pp. 327–348.

Hughes, T. J. R., 1987, *The Finite Element Method,* Prentice–Hall, Englewood Cliffs, NJ.

Morton, K. W., and Parrott, A. K., 1980, "Generalized Galerkin Methods for First-Order Hyperbolic Equations," *Journal of Computational Physics,* Vol. 36, pp. 249–270.

Newmark, N. M., 1954, "A Method of Computation for Structural Dynamics," *Journal of the Engineering Mechanics Division,* ASCE, pp. 67–94.

Oden, J. T., 1969, "A General Theory of Finite Elements II. Applications," *International Journal for Numerical Methods in Engineering,* Vol. 1, pp. 247–259.

Park, K. C., 1977, "Practical Aspects of Numerical Time Integration," *Computers & Structures,* Vol. 7, pp. 343–353.

Subbaraj, K., and Dokainish, M. A., 1989, "A Survey of Direct Time-Integration Methods in Computational Structural Dynamics-II. Implicit Methods," *Computers & Structures,* Vol. 32, pp. 1387–1401.

Weaver, W., Timoshenko, S. P., and Young, D. H., 1990, *Vibration Problems in Engineering,* 5th ed., John Wiley & Sons, Singapore.

Zienkiewicz, O. C., 1977, *The Finite Element Method,* 3rd ed., McGraw–Hill, London.