

## Research Article

# A Modified Precise Integration Method for Long-Time Duration Dynamic Analysis

Ce Huang  and Minghui Fu 

School of Engineering, Sun Yat-Sen University, No. 135 West Xingang Road, Guangzhou, China

Correspondence should be addressed to Minghui Fu; [stsfmh@mail.sysu.edu.cn](mailto:stsfmh@mail.sysu.edu.cn)

Received 10 February 2018; Revised 24 May 2018; Accepted 30 May 2018; Published 14 June 2018

Academic Editor: Giovanni Falsone

Copyright © 2018 Ce Huang and Minghui Fu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a modified Precise Integration Method (PIM) for long-time duration dynamic analysis. The fundamental solution and loading operator matrices in the first time substep are numerically computed employing a known unconditionally stable method (referred to as original method in this paper). By using efficient recursive algorithms to evaluate these matrices in the first time-step, the same numerical results as those using the original method with small time-step are obtained. The proposed method avoids the need of matrix inversion and numerical quadrature formulae, while the particular solution obtained has the same accuracy as that of the homogeneous solution. Through setting a proper value of the time substep, satisfactory accuracy and numerical dissipation can be achieved.

## 1. Introduction

In structural dynamic analysis, direct time integration algorithms are widely used to solve the motion equation [1]. Many researchers [2–4] have focused on the improvement of numerical properties of time integration algorithms, such as numerical stability, accuracy, numerical dissipation, and computational efficiency.

An alternative way to evaluate the dynamic response is to use the Precise Integration Method (PIM-L) [5], where a numerical exponential matrix function is accurately computed to approximate the fundamental solution matrix and the applied loading is simulated by using piecewise linear functions. For linear dynamic homogeneous systems, the PIM-L can obtain precise numerical results approaching the exact solution. However, its accuracy is limited by the simulation of the applied loading in nonhomogeneous systems. Lin et al. [6] applied Fourier series expansions to simulate the applied loading and proposed the PIM-F method. Both the PIM-L and PIM-F require matrix inversion to obtain the particular solution. However, if the objective matrix is singular or nearly singular, its inverse matrix either does not exist or is hard to obtain steadily.

In order to avoid matrix inversion, Gu et al. [7] converted nonhomogeneous equations into homogeneous ones at the expense of dimensional expansion, which significantly increased storage and computational cost. Afterwards, Wang et al. [8] proposed a PIM based on Lagrange piecewise polynomials and improved the dimensional expansion approach. Another practical option to compute the particular solution without matrix inversion is to use the numerical quadrature formulae. Wang et al. [9, 10] applied Gauss quadrature, Simpson quadrature, and Romberg integration formulae to integrate the particular solution. The accuracy of particular solution using these methods depends crucially on the number of integration points used in each time-step and the size of time-step. Basically, the accuracy of the particular solution obtained by these aforementioned methods may not achieve the same accuracy as that of the homogeneous solution, especially when a large time-step is used.

In conventional PIMs, the fundamental solution matrix in the first time substep is approximated by the Taylor series expansion of the exponential matrix function. Fung [11] halved the computational effort required using step-response and impulsive-response matrices and considering the symmetry of these matrices. Later on, the Krylov subspace

method [12] and the Ritz vectors [13] were also applied to improve the computational efficiency. Gao et al. [14] utilized the special feature of the dynamic system to reduce the computational effort involved in establishing the exponential matrix function. Wang et al. [9] investigated the numerical stability and computational accuracy of the conventional PIMs in detail. Subsequently, Wang et al. [10] proposed the PIM based on Padé approximation, where the generalized Padé approximation was used to replace the Taylor series expansion to achieve unconditional stability and controllable numerical dissipation.

This paper aims to modify and improve the existing PIM for solving long-time duration dynamics problems. The numerical fundamental solution and loading operator matrices are obtained by applying an unconditionally stable time integration method and recursive formulae. Using this proposed approach, the same accuracy of particular solution as that of the homogeneous solution can be achieved even for a large time-step, which is considered as a major improvement of PIMs. Besides, the proposed method can obtain approximately the same numerical results as those using the original method with a small time-step; thus it can also be considered as an accelerator of conventional time integration methods.

## 2. The Precise Integration Methods

The motion equation of a discretized linear structural model with  $n$ -degrees-of-freedom is given by

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{F}(t), \\ \mathbf{x}(t)|_{t=0} &= \mathbf{x}_0, \quad \dot{\mathbf{x}}(t)|_{t=0} = \dot{\mathbf{x}}_0, \end{aligned} \quad (1)$$

where  $\mathbf{M}$ ,  $\mathbf{C}$ , and  $\mathbf{K}$  are the mass, damping, and stiffness matrices, respectively;  $\mathbf{F}(t)$  is the vector of applied external loading;  $\mathbf{x}(t)$ ,  $\dot{\mathbf{x}}(t)$ , and  $\ddot{\mathbf{x}}(t)$  are the displacement, velocity, and acceleration vectors of the system, respectively.

Assuming  $\mathbf{v} = \begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{x} \end{bmatrix}$ ,  $\mathbf{r}(t) = \begin{bmatrix} \mathbf{0} \\ \mathbf{F}(t) \end{bmatrix}$ , (1) can be transformed into the following first-order ordinary differential equation:

$$\dot{\mathbf{v}} = \mathbf{H}\mathbf{v} + \mathbf{r}(t), \quad (2)$$

where  $\mathbf{H} = \begin{bmatrix} \mathbf{0} & \mathbf{M}^{-1} \\ -\mathbf{K} & -\mathbf{C}\mathbf{M}^{-1} \end{bmatrix}$ . An analytical solution of (2) can be expressed as follows [15]:

$$\mathbf{v}(t) = e^{\mathbf{H}t} \mathbf{v}_0 + \int_0^t e^{\mathbf{H}(t-s)} \mathbf{r}(s) ds \quad (3)$$

Applying time discretization with a time-step  $\tau$  and assuming the analysis starts at the instant  $t_m = m\tau$ ,  $\mathbf{v}_{m+1}$  can then be represented by  $\mathbf{v}_m$ :

$$\begin{aligned} \mathbf{v}_{m+1} &= \mathbf{T}(\tau) \mathbf{v}_m + \mathbf{v}_{m+1}^* \\ \mathbf{T}(\tau) &= e^{\mathbf{H}\tau}, \\ \mathbf{v}_{m+1}^* &= \int_{t_m}^{t_{m+1}} e^{\mathbf{H}(t_{m+1}-s)} \mathbf{r}(s) ds, \end{aligned} \quad (4)$$

where  $t_m = m\tau$ ,  $t_{m+1} = (m+1)\tau$ ,  $\mathbf{v}_m = \mathbf{v}(t_m)$ , and  $\mathbf{v}_{m+1} = \mathbf{v}(t_{m+1})$ .  $\mathbf{T}(\tau)$  can be written as

$$\mathbf{T}(\tau) = e^{\mathbf{H}\tau} = \left( e^{\mathbf{H}\Delta t} \right)^{2^N} = \left( \mathbf{T}(\Delta t) \right)^{2^N}, \quad (5)$$

where  $\Delta t = \tau/2^N$ ,  $N$  is a given integer. The exponential matrix  $e^{\mathbf{H}\Delta t}$  can be approximated by using the Taylor series expansion or the generalized Padé approximation. Taking the Taylor series expansion as an example,  $\mathbf{T}(\Delta t)$  can be written as

$$\mathbf{T}(\Delta t) \approx \mathbf{I} + \tilde{\mathbf{T}}_a^{(0)}, \quad (6)$$

where

$$\tilde{\mathbf{T}}_a^{(0)} = \mathbf{H}\Delta t + \frac{(\mathbf{H}\Delta t)^2}{2!} + \frac{(\mathbf{H}\Delta t)^3}{3!} + \cdots + \frac{(\mathbf{H}\Delta t)^p}{p!}, \quad (7)$$

where  $p$  is the truncation order of the Taylor series expansion. Substituting (6) into (5) gives

$$\begin{aligned} \mathbf{T}(2\Delta t) &\approx \mathbf{I} + \tilde{\mathbf{T}}_a^{(1)} = \mathbf{I} + 2\tilde{\mathbf{T}}_a^{(0)} + \tilde{\mathbf{T}}_a^{(0)} \cdot \tilde{\mathbf{T}}_a^{(0)} \\ \mathbf{T}(2^2\Delta t) &\approx \mathbf{I} + \tilde{\mathbf{T}}_a^{(2)} = \mathbf{I} + 2\tilde{\mathbf{T}}_a^{(1)} + \tilde{\mathbf{T}}_a^{(1)} \cdot \tilde{\mathbf{T}}_a^{(1)} \\ &\vdots \\ \mathbf{T}(2^N\Delta t) &\approx \mathbf{I} + \tilde{\mathbf{T}}_a^{(N)} = \mathbf{I} + 2\tilde{\mathbf{T}}_a^{(N-1)} \\ &\quad + \tilde{\mathbf{T}}_a^{(N-1)} \cdot \tilde{\mathbf{T}}_a^{(N-1)} \end{aligned} \quad (8)$$

$\tilde{\mathbf{T}}_a^{(j)}$  ( $j = 0, 1, 2, \dots, N$ ) can be evaluated by applying the following recursive formula:

$$\tilde{\mathbf{T}}_a^{(j+1)} = 2\tilde{\mathbf{T}}_a^{(j)} + \tilde{\mathbf{T}}_a^{(j)} \cdot \tilde{\mathbf{T}}_a^{(j)} \quad (j = 0, 1, 2, \dots, N-1) \quad (9)$$

## 3. A Modified PIM for Long-Time Duration Dynamic Analysis

**3.1. The Numerical Fundamental Solution Matrix.** In the standard finite-element semidiscrete systems, only a few low-frequency modes can represent the original continuous system. The quality of time integration algorithms depends crucially on their dissipative property. It has been found that the PIM is conditionally stable [9]. In order to satisfy the stability conditions, the time substep has to be smaller than the highest modal period of the particular discretized structure. In this case, the dissipative property would be eliminated. Wang et al. [10] proposed the PIM based on Padé approximations to achieve unconditional stability and controllable dissipation, in which the Taylor series expansion of the matrix exponential function is replaced with the generalized Padé approximations. In this work, the fundamental solution matrix is approximated using numerical Green's function matrix to achieve desirable numerical properties.

An analytical expression of  $\mathbf{x}(m\tau + \Delta t)$ ,  $\dot{\mathbf{x}}(m\tau + \Delta t)$  can be written in terms of  $\mathbf{x}(m\tau)$ ,  $\dot{\mathbf{x}}(m\tau)$  [16–19]:

$$\begin{aligned}\mathbf{x}(m\tau + \Delta t) &= \left( \mathbf{G}(\Delta t) \mathbf{C} + \dot{\mathbf{G}}(\Delta t) \mathbf{M} \right) \mathbf{x}(m\tau) \\ &\quad + \mathbf{G}(\Delta t) \mathbf{M} \dot{\mathbf{x}}(m\tau) \\ \dot{\mathbf{x}}(m\tau + \Delta t) &= \left( \dot{\mathbf{G}}(\Delta t) \mathbf{C} + \ddot{\mathbf{G}}(\Delta t) \mathbf{M} \right) \mathbf{x}(m\tau) \\ &\quad + \dot{\mathbf{G}}(\Delta t) \mathbf{M} \dot{\mathbf{x}}(m\tau),\end{aligned}\quad (10)$$

where  $\mathbf{G}(\Delta t)$ ,  $\dot{\mathbf{G}}(\Delta t)$  represent Green's function matrix of the model and its derivative, respectively. Thus, the analytical expression of  $\mathbf{T}(\Delta t)$  is given as follows:

$$\mathbf{T}(\Delta t) = \begin{bmatrix} \mathbf{G}(\Delta t) \mathbf{C} + \dot{\mathbf{G}}(\Delta t) \mathbf{M} & \mathbf{G}(\Delta t) \\ \dot{\mathbf{G}}(\Delta t) \mathbf{C} + \ddot{\mathbf{G}}(\Delta t) \mathbf{M} & \dot{\mathbf{G}}(\Delta t) \end{bmatrix} \quad (11)$$

$\mathbf{G}(\Delta t)$  and  $\dot{\mathbf{G}}(\Delta t)$  can be calculated numerically by solving the following equation:

$$\begin{aligned}\mathbf{M}\ddot{\mathbf{G}}(t) + \mathbf{C}\dot{\mathbf{G}}(t) + \mathbf{K}\mathbf{G}(t) &= 0 \\ \mathbf{G}(0) &= \mathbf{0}, \\ \dot{\mathbf{G}}(0) &= \mathbf{M}^{-1}\end{aligned}\quad (12)$$

In order to satisfy the unconditional stability, a known unconditionally stable time integration method (referred to as original method in the following discussion) with time-step  $\Delta t$  is used to solve (12). By selecting a proper original method and setting the value of time substep, one can obtain satisfactory accuracy, numerical dissipation, and other numerical properties.

Substituting  $\mathbf{G}(\Delta t)$  into the following equation,  $\mathbf{T}_a^{(0)}$  can be obtained:

$$\mathbf{T}_a^{(0)} = \mathbf{T}(\Delta t) - \mathbf{I} \quad (13)$$

Substituting  $\mathbf{T}_a^{(0)}$  into (8) and (9),  $\mathbf{T}(\tau)$  can be evaluated. On the other hand, there is another method to evaluate  $\mathbf{T}(\Delta t)$ , which does not need the inverse of the mass matrix. The  $j$  column of  $\mathbf{T}(\Delta t)$  can be obtained through solving the following equation:

$$\begin{aligned}\dot{\mathbf{v}} &= \mathbf{H}\mathbf{v}, \\ \mathbf{v}(0) &= \mathbf{1}_j,\end{aligned}\quad (14)$$

where  $1_{ij}$  is the  $i$  entry of the unity vector  $\mathbf{1}_j$ .  $1_{ij}$  is zero if  $i \neq j$  and 1 if  $i = j$  ( $1_{ij} = \delta_{ij}$ ,  $\delta_{ij}$  being the Kronecker delta).

It can be seen that the method incorporates the PIM with Green's function matrix based method [16–19]. For this method, the sparsity of the fundamental matrix would be weakened as the number of time substeps increases. As a result, this method requires more storage usage compared to conventional time integration methods. However, much larger time-step can be used, which causes less computational cost. Users can find a balance between storage usage and computation cost by setting proper  $N$ . The improvement of the storage efficiency is feasible, which needs further investigation.

**3.2. The Loading Operator Matrix.** For homogeneous systems, the PIMs can give a precise numerical solution. However, for nonhomogeneous dynamic systems, the accuracy of numerical results depends essentially on the accuracy of the particular solution. The accuracy of the PIM-L and PIM-F methods is mainly limited by the error in the matrix inversion and the applied loading simulation. For the PIM proposed by Wang et al. [10], the main error was caused by the process of the numerical quadrature formulae. In order to obtain a high accuracy, a large number of integration points were used, which required extra computational effort. When a larger time-step is adopted, the particular solution obtained may not attain the same accuracy as that of the homogeneous solution. In this study, a novel method is proposed to evaluate the particular solution, which avoids the matrix inversion. It can retain the same accuracy of the particular solution as that of the homogeneous solution even though the time-step is very large.

In practical engineering problems, abounding loadings can be expressed in a periodical form, which can be approximated by using the Fourier series expansion

$$\{\mathbf{F}(t)\} \approx \{\mathbf{B}_0\} + \sum_{i=1}^l \{\mathbf{A}_i \sin(i\omega t) + \{\mathbf{B}_i\} \cos(i\omega t)\}, \quad (15)$$

where  $l$  is the truncation order of the Fourier series expansion. According to Euler's formula, the trigonometric functions in (15) can be transformed as an exponential function.

$$e^{i\omega t} = \cos \omega t + i \sin \omega t \quad (i^2 = -1) \quad (16)$$

An exponential function loading can be expressed as follows in the time interval  $[t_m, t_{m+1}]$ ,

$$\mathbf{r}(t) = \tilde{\mathbf{r}} e^{bt}, \quad (17)$$

where  $\tilde{\mathbf{r}}$  is a constant vector and  $b$  is a real or complex number. Substituting (17) into (4), the following equation can be obtained:

$$\mathbf{v}^*_{m+1} = \int_{t_m}^{t_{m+1}} e^{\mathbf{H}(t_{m+1}-t)} \tilde{\mathbf{r}} e^{bt} dt \quad (18)$$

Assuming  $s = t_{m+1} - t$ , (18) becomes

$$\mathbf{v}^*_{m+1} = \int_0^\tau e^{\mathbf{H}s} e^{b(t_{m+1}-s)} ds \tilde{\mathbf{r}} = e^{bt_{m+1}} \tilde{\mathbf{S}} \tilde{\mathbf{r}} \quad (19)$$

$$\tilde{\mathbf{S}} = \int_0^\tau e^{\mathbf{H}s} ds = \sum_{j=0}^{2N-1} \int_{j\Delta t}^{(j+1)\Delta t} e^{\mathbf{H}s} e^{-bs} ds, \quad (20)$$

where  $\tilde{\mathbf{S}} \tilde{\mathbf{r}}$  is the loading operator matrix. It can be seen that  $\tilde{\mathbf{S}} \tilde{\mathbf{r}}$  is not associated with the integration interval  $[t_m, t_{m+1}]$ , thus it is calculated only once during the whole step-by-step integration procedure. As (19) is the analytical expression of the particular solution, once  $\tilde{\mathbf{S}} \tilde{\mathbf{r}}$  is properly computed, the particular solution would become extremely accurate. In order to ensure the same accuracy of the particular solution as that of the homogeneous solution,  $e^{\mathbf{H}\Delta t}$  in (20) can be

approximated by  $\mathbf{T}(\Delta t)$ , which is generated in Section 3.1.  $\tilde{\mathbf{S}}\tilde{\mathbf{r}}$  can be written as

$$\tilde{\mathbf{S}}\tilde{\mathbf{r}} \approx \sum_{j=0}^{2^N-1} \mathbf{T}(\Delta t)^{j-1} e^{-b(j-1)\Delta t} \int_0^{\Delta t} e^{\mathbf{H}s} e^{-bs} ds \tilde{\mathbf{r}} \quad (21)$$

It can be seen that  $\tilde{\mathbf{S}}\tilde{\mathbf{r}}$  is the sum of  $2^N$  terms. Let  $\tilde{\mathbf{S}}^{(j)}\tilde{\mathbf{r}}$  ( $j = 0, 1, 2, \dots, N$ ) denote the sum of the first  $2^j$  terms of (21). The following recursive formula can be used to evaluate  $\tilde{\mathbf{S}}^{(N)}\tilde{\mathbf{r}}$  from  $\tilde{\mathbf{S}}^{(0)}\tilde{\mathbf{r}}$ :

$$\begin{aligned} \tilde{\mathbf{S}}^{(j+1)}\tilde{\mathbf{r}} &= \mathbf{Z}^{2^j} \left( \tilde{\mathbf{S}}^{(j)}\tilde{\mathbf{r}} \right) \\ \mathbf{Z}^{2^j} &= e^{-2^j b \Delta t} \left( \mathbf{I} + \mathbf{T}_a^{(j)} \right), \end{aligned} \quad (22)$$

where  $\mathbf{Z} = \mathbf{T}(\Delta t)e^{-b\Delta t} \cdot \mathbf{T}_a^{(j)}$  ( $j = 0, 1, 2, \dots, N$ ) is obtained in Section 3.1. The matrices  $\mathbf{T}_a^{(j)}$  and  $\tilde{\mathbf{S}}^{(j)}\tilde{\mathbf{r}}$  are computed using the following looping algorithm:

$$\begin{aligned} &\text{for } j = 1 : N \\ &\quad \mathbf{T}_a = 2 * \mathbf{T}_a + \mathbf{T}_a * \mathbf{T}_a \\ &\quad \mathbf{S}\mathbf{r} = \mathbf{S}\mathbf{r} + e^{(-2^j b \Delta t)} * (\mathbf{I} + \mathbf{T}_a) * \mathbf{S}\mathbf{r} \\ &\text{end} \end{aligned} \quad (23)$$

For conventional PIMs, the algorithm (23) can also be used to acquire  $\tilde{\mathbf{S}}^{(N)}\tilde{\mathbf{r}}$ . Thus the particular solution would possess the same accuracy as the homogeneous solution. In this work,  $\tilde{\mathbf{S}}^{(0)}\tilde{\mathbf{r}}$  is obtained numerically by solving the following equation using the original method with step-size  $\Delta t$ :

$$\dot{\mathbf{v}} = \mathbf{H}\mathbf{v} + \tilde{\mathbf{r}}e^{bt}, \quad (24)$$

$$\mathbf{v}(t_m) = \mathbf{0}$$

$$\mathbf{v}(t_m + \Delta t) = e^{b\Delta t} \tilde{\mathbf{S}}^{(0)}\tilde{\mathbf{r}} \quad (25)$$

During the recursive process(22), only matrix-vector multiplications are required. Therefore, it can be summarized that only  $N$  times of matrix-matrix multiplication are needed to obtain the numerical fundamental solution and loading operator matrices (as the computational effort of matrix-vector multiplication is negligible compared to that of matrix-matrix multiplication, only matrix-matrix multiplication is considered).

**3.3. Properties Analysis.** As the recurrence relations ((9) and (22)) are analytical expressions, the accuracy, stability, dissipation, and other numerical properties of the proposed method are determined only by the original method and the size of time substep. In this section, the stability of the proposed method is investigated. For the analysis of other properties, one can refer to [16] for detailed information.

Considering the following undamped unforced single degree of freedom (SDOF) problem:

$$\ddot{x} + \omega^2 x = 0, \quad (26)$$

where  $\omega$  is the natural frequency of the model and the natural period is  $T_n = 2\pi/\omega$ . Assuming the time-step is  $\tau$ , time substep is  $\Delta t$  ( $\Delta t = \tau/2^N$ ). The following recursive relationship can be obtained using the original method:

$$\begin{bmatrix} x\left(t_m + \frac{\tau}{2^N}\right) \\ \dot{x}\left(t_m + \frac{\tau}{2^N}\right) \end{bmatrix} = \mathbf{A}\left(\frac{\tau}{2^N}\right) \begin{bmatrix} x(t_m) \\ \dot{x}(t_m) \end{bmatrix}, \quad (27)$$

where  $\mathbf{A}(\tau/2^N)$  stands for the amplification matrix of the original method. According to (5), the amplification matrix of the proposed method is obtained as  $\mathbf{A}(\tau) = \mathbf{A}(\tau/2^N)^{2^N}$ . The stability condition is  $\rho(\mathbf{A}) \leq 1$ , where  $\rho(\mathbf{A}) = \max(\lambda_{1,2})$  is the spectral radius of  $\mathbf{A}$  and  $\lambda_{1,2}$  are the eigenvalues of  $\mathbf{A}$ .

In order to investigate the stability of the proposed method, the Newmark ( $\alpha = 3/10$ ,  $\delta = 11/20$ ), the Bathe [2] and the fourth-order Runge-Kutta (RK-4) methods are selected as the original method, respectively.

Figure 1 shows the spectral radius of the proposed method, where MPIM-Newmark, MPIM-Bathe, and MPIM-RK denote the results obtained using the Newmark, Bathe, and RK-4 method as the original method, respectively. For all curves the value of spectral radius is 1 until  $\tau/T_n$  is equal to a certain value and thereafter it rapidly diminishes. As  $N$  increases, this value increases. The MPIM-Newmark has lower limit values than the Newmark method. Compared to RK-4, the stability condition of MPIM-RK4 is relaxed.

## 4. Numerical Examples

In this section, two numerical examples are investigated to evaluate the validity of the proposed method. The proposed method solution is compared with those obtained by the SPIM ( $q = 2, m = 8, \mu = 0.8$ ), the ExGA-RK method [16], and the original method with a small time-step. The Newmark ( $\alpha = 3/10$ ,  $\delta = 11/20$ ), the Bathe [2], and the fourth-order Runge-Kutta (RK-4) methods are selected as the original method, respectively. The RK-4 method is selected just for the sake of the comparison with the ExGA-RK method. An unconditionally stable method can be used when solving particular dynamic problems.

In Figures 2–11, MPIM-RK, MPIM-Newmark, and MPIM-Bathe denote the solution obtained by the proposed modified PIM using the RK-4, Newmark, and the Bathe method as the original method, respectively.

**4.1. Two-DOF Model.** The first example considers the following two-DOF system:

$$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} 6 & -2 \\ -2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \quad (28)$$

The natural periods of the system are  $T_1 = 4.443$ ,  $T_2 = 2.810$  (some values are rounded). In the following cases, the solution obtained by the mode superposition method is referred to as the reference solution. In Case 1, a constant loading is used, while the applied loading is a sine function in Case 2.

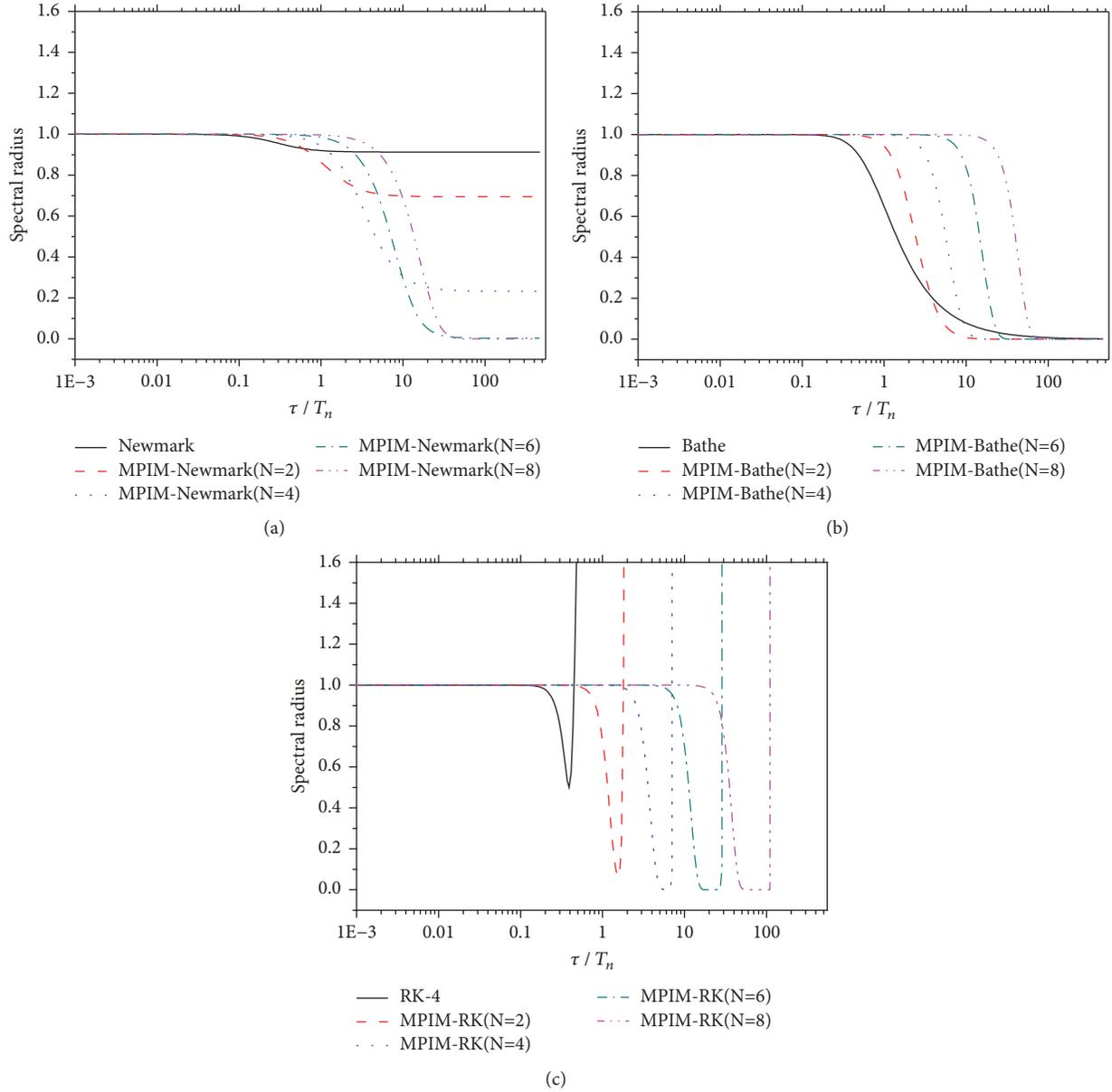


FIGURE 1: Spectral radius of the proposed method. (a)MPIM-Newmark, (b)MPIM-Bathe, and (c)MPIM-RK4.

*Case 1.* The initial conditions are set as  $x_1(0) = 1$ ,  $x_2(0) = 2$ ,  $\dot{x}_1(0) = 0$ , and  $\dot{x}_2(0) = 0$  with  $f_1 = 100$ ,  $f_2 = 0$ . In this case, the time-step is selected as  $\tau = 0.5236$ , the recursive parameter  $N = 6$ , and the time substep  $\Delta t = 0.0082$ .

Figures 2 and 3 show the simulation results obtained using various methods. The error is defined as follows:

$$\text{error} = |\hat{x}_i(t) - x_i(t)| \quad (i = 1, 2), \quad (29)$$

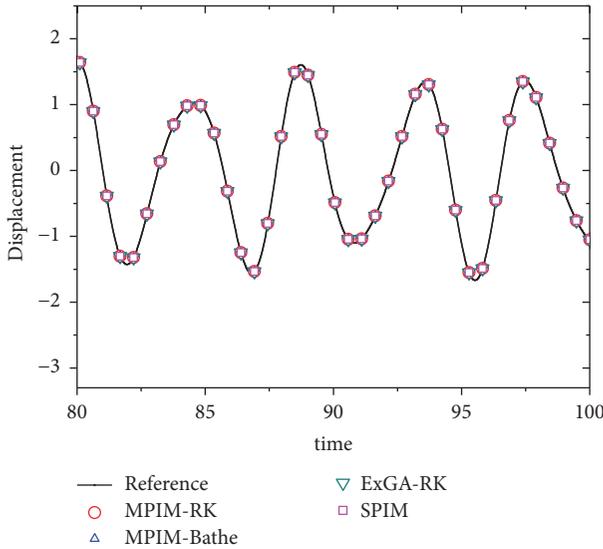
where  $\hat{x}_i(t)$  is the response obtained using numerical methods and  $x_i(t)$  is the reference solution. The ExGA-RK method employs the RK-4 to calculate the fundamental solution matrix and Simpson's 1/3 rule to obtain the particular solution. The SPIM applies the generalized Padé approximation ( $q = 2$ ,  $\mu = 0.8$ ) to calculate the fundamental solution

matrix and the repeated Simpson integration formulae (using 8 integration points) to obtain the particular solution. In this case, the applied loading is a constant vector and the integrand of the particular solution remains unchanged. As a result, all methods perform very well. The error follows the sequence of MPIM - Bathe > SPIM > ExGA-RK=MPIM-RK.

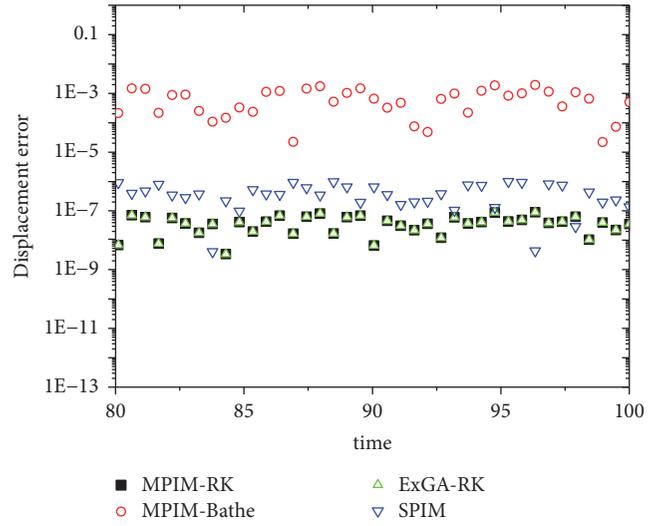
The accuracy of numerical fundamental solution matrices obtained by different methods is also investigated by fixing the time-step  $\tau = 0.262$  and choosing different recursive parameters  $N$ . The error of  $\tilde{\mathbf{T}}(\tau)$  is defined as follows:

$$\text{error of } \tilde{\mathbf{T}} = \|\tilde{\mathbf{T}}(\tau) - \mathbf{T}(\tau)\|_2, \quad (30)$$

where  $\tilde{\mathbf{T}}(\tau)$  is the numerical fundamental solution matrices obtained using different methods and  $\mathbf{T}(\tau)$  is the reference

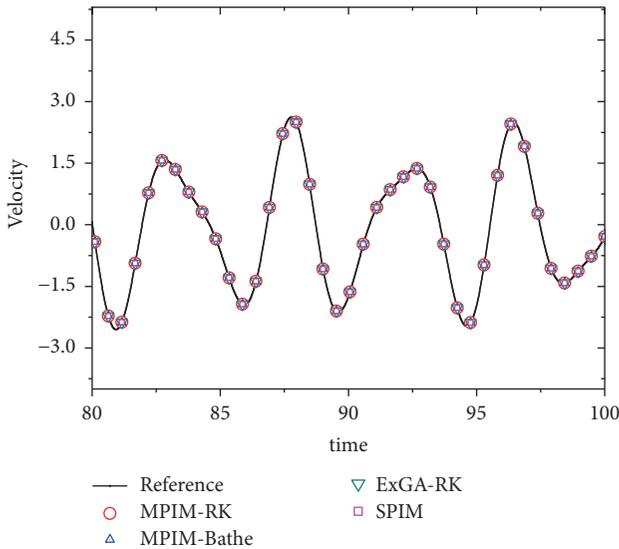


(a)

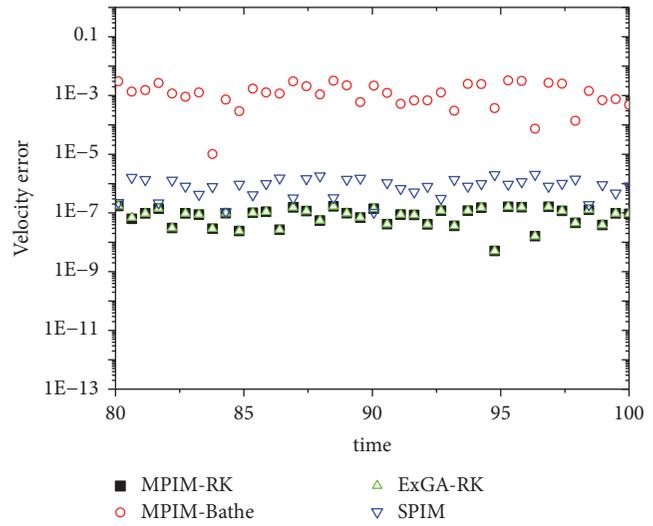


(b)

FIGURE 2: Displacement of DOF1 in Case 1 using different methods: (a) displacement; (b) displacement error.



(a)



(b)

FIGURE 3: Velocity of DOF1 in Case 1 using different methods: (a) velocity; (b) velocity error.

fundamental solution matrix. Figure 4 depicts the error versus the recursive parameters  $N$ . It can be seen that the error decreases with the increasing of  $N$ . With the same  $N$ , the error follows the sequence of MPIM- Newmark >MPIM-Bathe >SPIM>MPIM-RK.

*Case 2.* The initial conditions are set as  $x_1(0) = 1, x_2(0) = 2, \dot{x}_1(0) = 0,$  and  $\dot{x}_2(0) = 0$  and the load  $f_1 = 100 \sin(3t), f_2 = 0$ . In this case, the time substep is set as  $\Delta t = 0.0082$ . The recursive parameter is chosen as  $N = 4, 6, 8$ . Thus  $\tau = 0.131, 0.524, 2.094$  and  $\tau/T_p = 1/16, 1/4, 1$ , where  $T_p = 2.094$  is the period of the applied loading.

Figures 5–10 show the simulation results using different  $N$ . It can be seen that the errors of the ExGA-RK, SPIM solution increase as the time-step increases, while the solutions obtained by the proposed methods (MPIM-Bathe, MPIM-RK) is always the same as the reference solution. For instance, for  $N = 6, \tau = T_p/2$ , the loading is zero at the start and end points of each time-step; thus it would be considered as zero within each time-step. Although the SPIM uses 8 integration points to calculate the particular, the numerical results have large errors when  $N = 6, 8$ .

In this case, the numerical convergence property of the proposed method is also investigated by calculating the time

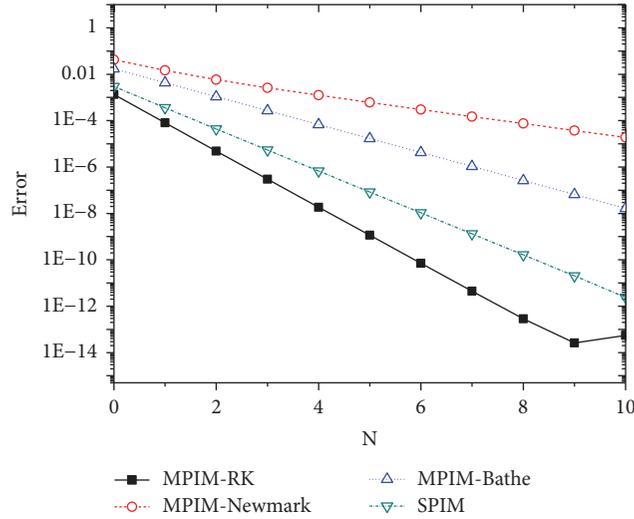


FIGURE 4: Error of numerical fundamental solution matrix versus N.

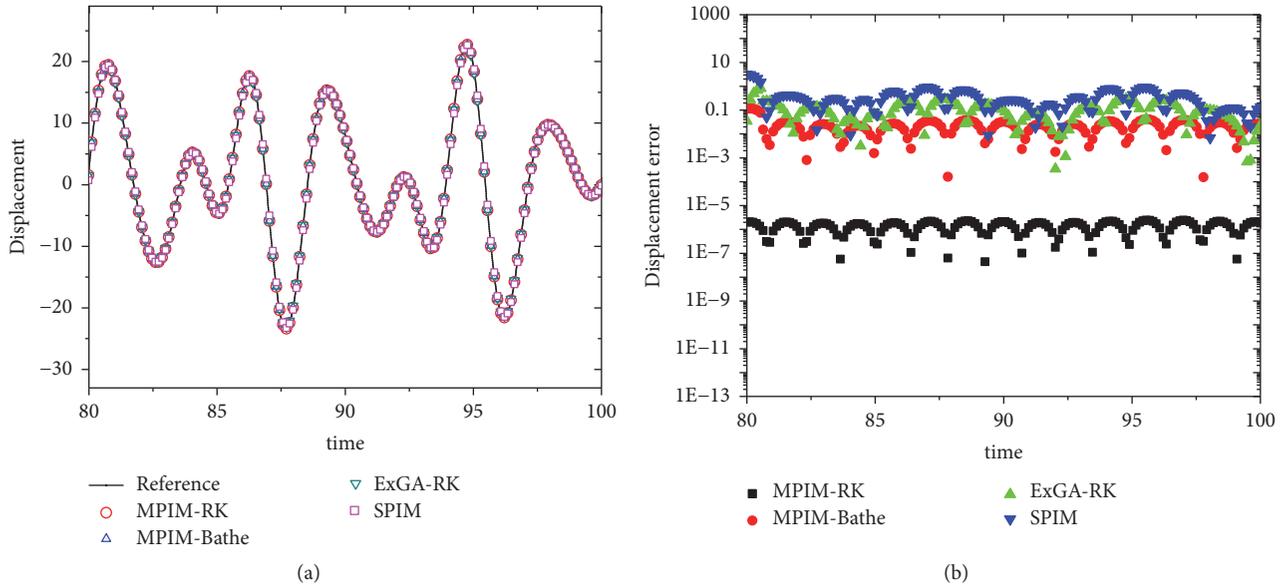


FIGURE 5: Displacement of DOF2 in Case 2 (N=4): (a) displacement; (b) displacement error.

order of convergence rate. The time order of convergence rate is calculated by applying the following formula:

$$s = \frac{\log(\text{maxerror}_1) - \log(\text{maxerror}_2)}{\log(\tau_1) - \log(\tau_2)}, \quad (31)$$

where  $\tau_1, \tau_2$  are different time-steps, and  $\text{maxerror}_1, \text{maxerror}_2$  are the maximum error of corresponding numerical result. Figure 11 shows the log of the  $\text{maxerror}$  versus the log of time-step for displacement and velocity of node 2. It can be seen that the slopes of SPIM-RK and SPIM-Bathe curves remain unchanged as  $N$  increases. The slopes of SPIM-RK and SPIM-Bathe curves are about 3.99 and 1.97, respectively.

4.2. *Regular Diagonally Positioned Pyramid.* The second example considers the regular diagonally positioned pyramid shown in Figure 12, which is composed of aluminium tubes. The section dimensions and the material properties are as follows: the outer diameter  $150\text{mm}$ , the wall thickness  $11\text{mm}$ , the density  $\rho = 2.699 \times 10^3 \text{kg/m}^3$ , the Poisson ratio  $\nu = 0.32$ , and Young's modulus  $E = 6.895 \times 10^4 \text{MPa}$ . The top level of the pyramid is fixed along four sides, while node 1 is subjected to a vertical load  $F = 5 \times 10^3 \sin(1000t) \text{kN}$ . By using spatial discretization (spatial bar element), a structural model with 543 degrees of freedom is obtained. A consistent mass matrix is used in the analysis and the material is assumed to be linearly elastic. The damping matrix is assumed to be  $C = a_0K + b_0M$  (Rayleigh damping), where the coefficients

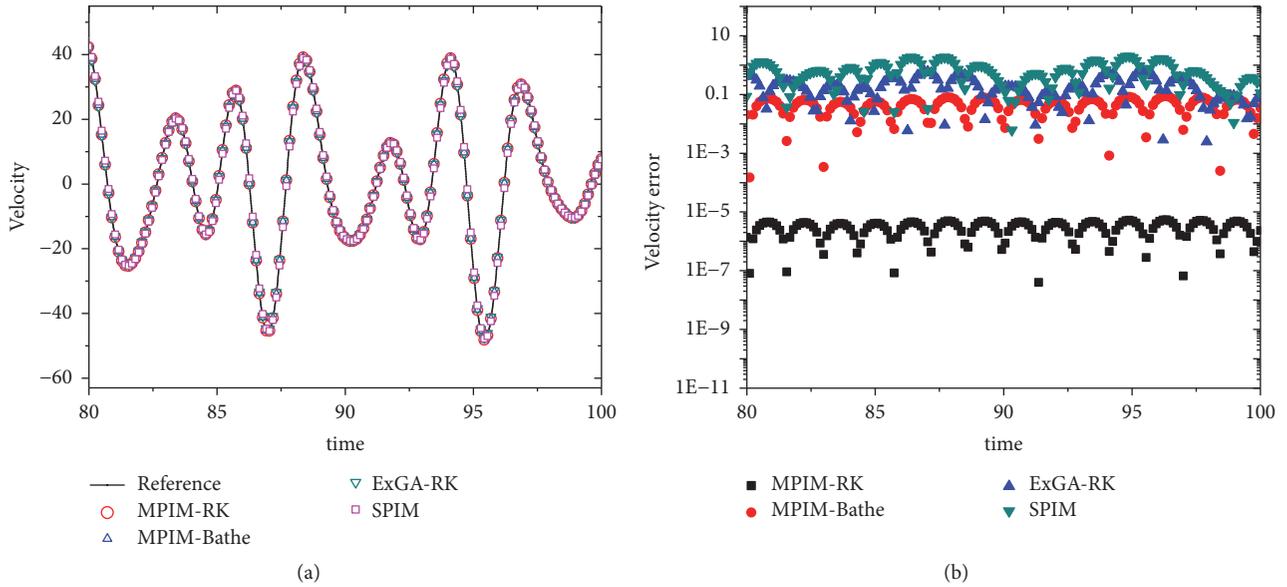


FIGURE 6: Velocity of DOF2 in Case 2 (N=4): (a) velocity; (b) velocity error.

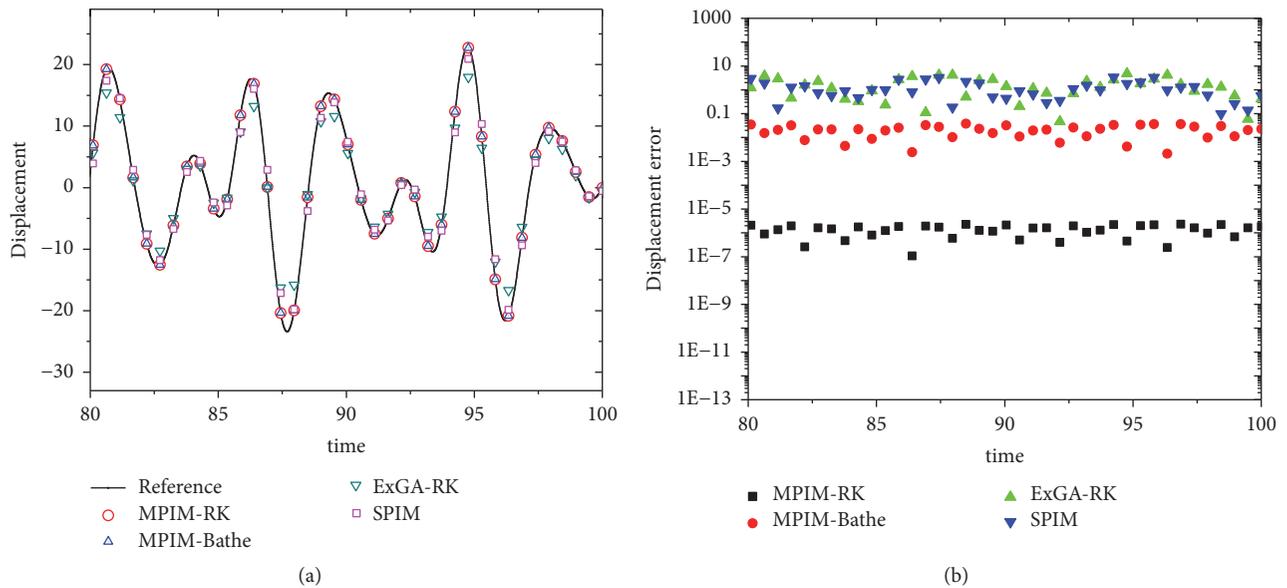


FIGURE 7: Displacement of DOF2 in Case 2 (N=6): (a) displacement; (b) displacement error.

$a_0 = 0.001$  and  $b_0 = 0.45$  [3]. The maximum natural period of the model is  $T_{\max} = 0.0565s$ .

The time-step is set as  $\tau = 0.0029s$ , the recursive parameter  $N = 5$ , and the time substep  $\Delta t = 9.062 \times 10^{-5}s$ . Thus  $\tau/T_{\max} = 0.0513$ ,  $\tau/T_p = 0.462$ , where  $T_p = 6.283 \times 10^{-3}s$  is the period of the loading. The solution obtained by the Bathe method with time-step  $9.062 \times 10^{-5}s$  is set as the reference solution. The model was solved for over 10 seconds. Table 1 shows the CPU time occupied by different methods (using a desktop PC with Inter(R) Core(TM) i5-5200U CPU @ 2.2 GHz, Ram@16G). The time-step for the RK-4 and Bathe method is  $\Delta t = 9.062 \times 10^{-5}s$ . It

can be seen that the proposed method requires much less CPU time than the original method with the small time-step.

Figure 13 shows the simulation results of the vertical displacement of node 1 and node 2, respectively. Similar conclusions can be obtained for other nodes. It demonstrates that the MPIM-Bathe and MPIM-RK solution exactly follows the reference solution. As the time-step is set as nearly half the period of the load, larger error occurs in the ExGA-RK and SPIM solutions.

From the foregoing discussion, it can be concluded that the accuracy of the proposed method is determined by the

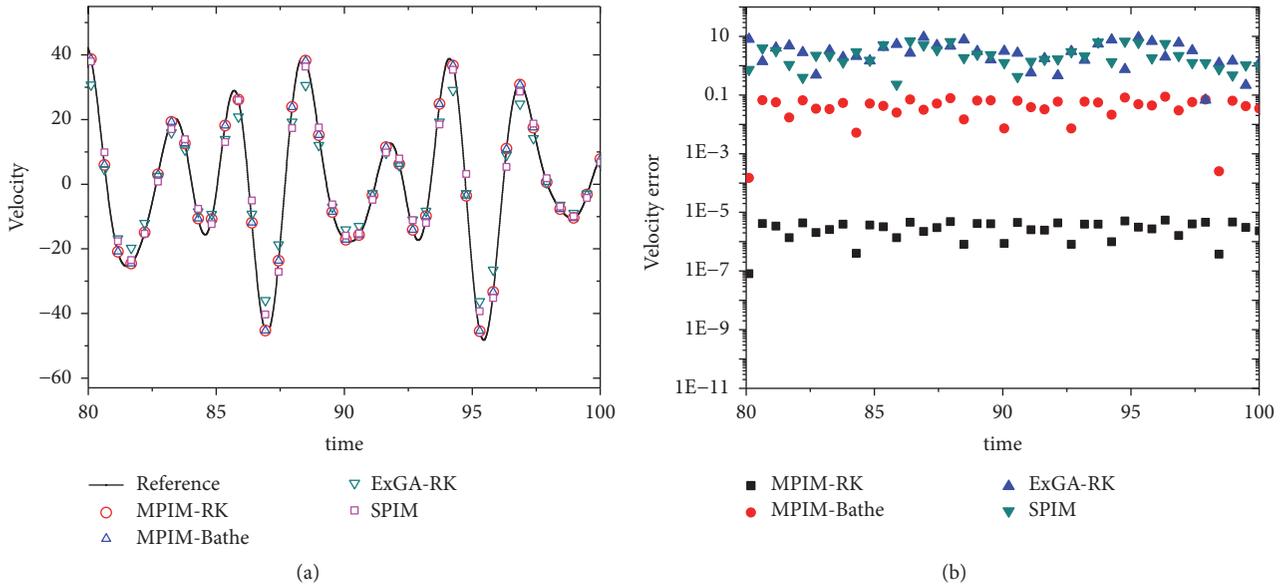


FIGURE 8: Velocity of DOF2 in Case 2 (N=6): (a) velocity; (b) velocity error.

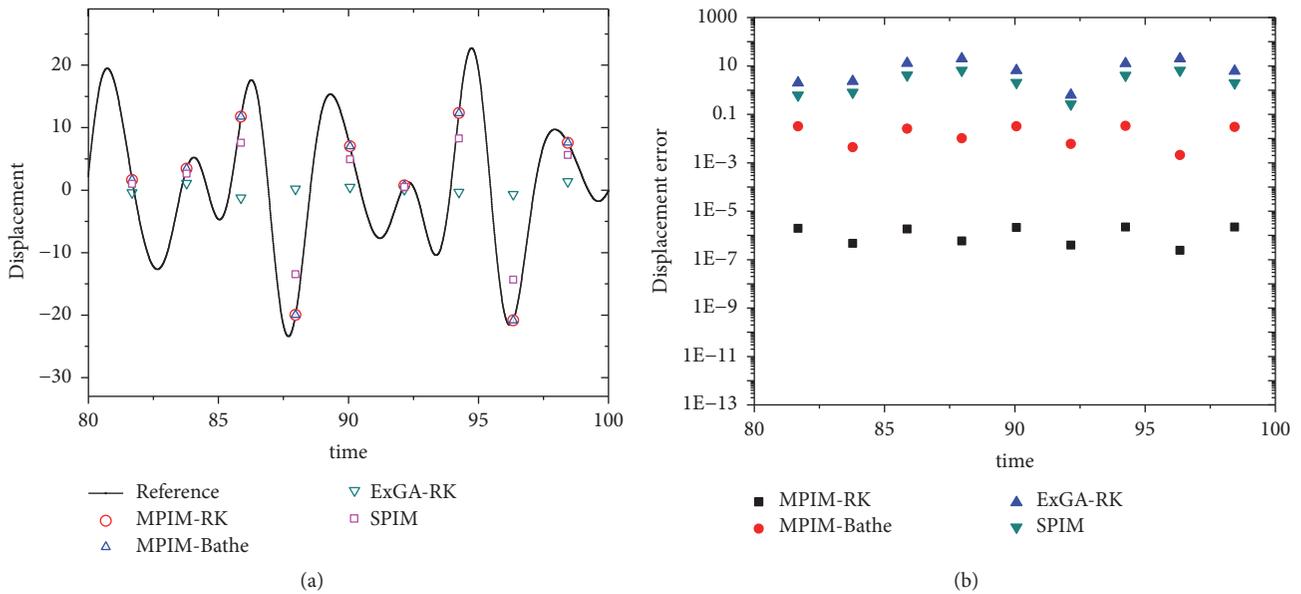


FIGURE 9: Displacement of DOF2 in Case 2 (N=8): (a) displacement; (b) displacement error.

size of time substep and the properties of the original method. Even with a large time-step, the particular solution has the same accuracy as the homogeneous solution. On the premise of obtaining almost the same numerical result, the proposed method can greatly improve the calculative efficiency of the original method.

### 5. Discussion

In recent years, many integration methods have been applied to calculate Green's function matrix, resulting in different versions of time marching method, such as the Newmark family methods [17], the generalized- $\alpha$  method [18], the

Runge-Kutta, and the central difference schemes [16]. Using a substepping procedure, these aforementioned approaches can obtain accurate homogeneous solution. However, the technologies used to evaluate the particular solution are basically based on some approximations that are only suitable for a small time-step. For instance, in [16] the external loading vector is assumed to be linear functions in each time-step. In [19] the integrand of the particular solution is assumed to be a linear or parabolic function within each time-step. When a larger time-step is considered, these approximations are oversimplified and results in unexpected errors. For instance, the time-step is set as an integral multiple of the period of the periodical loading; the loading would be considered

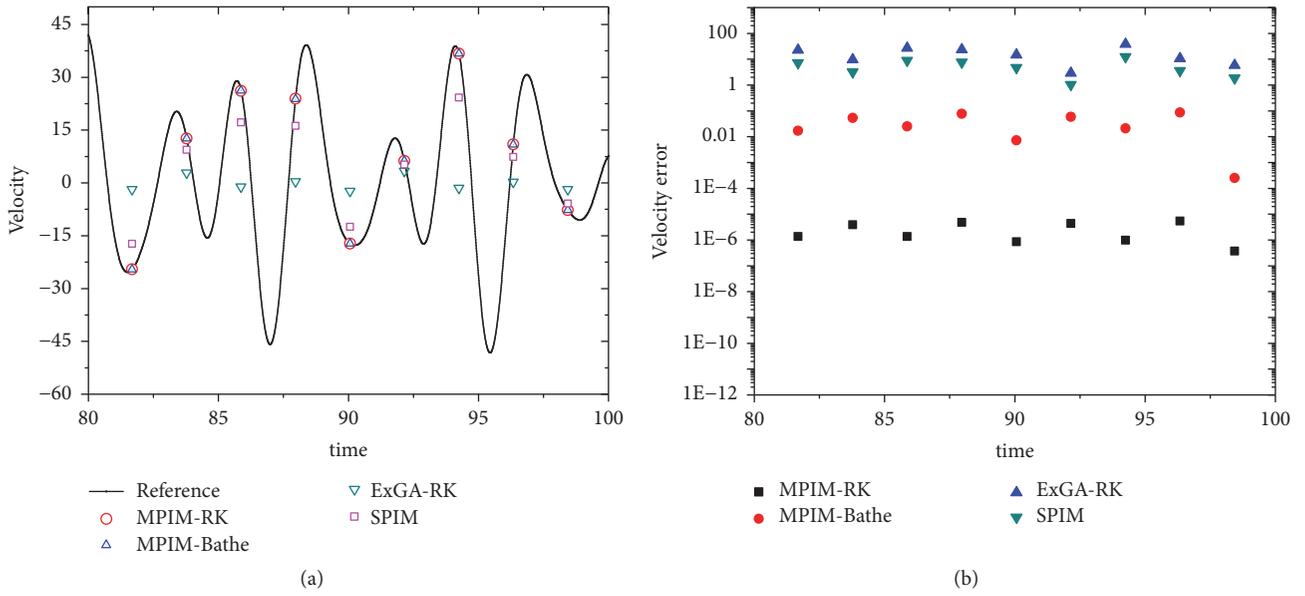


FIGURE 10: Velocity of DOF2 in Case 2 (N=8): (a) velocity; (b) velocity error.

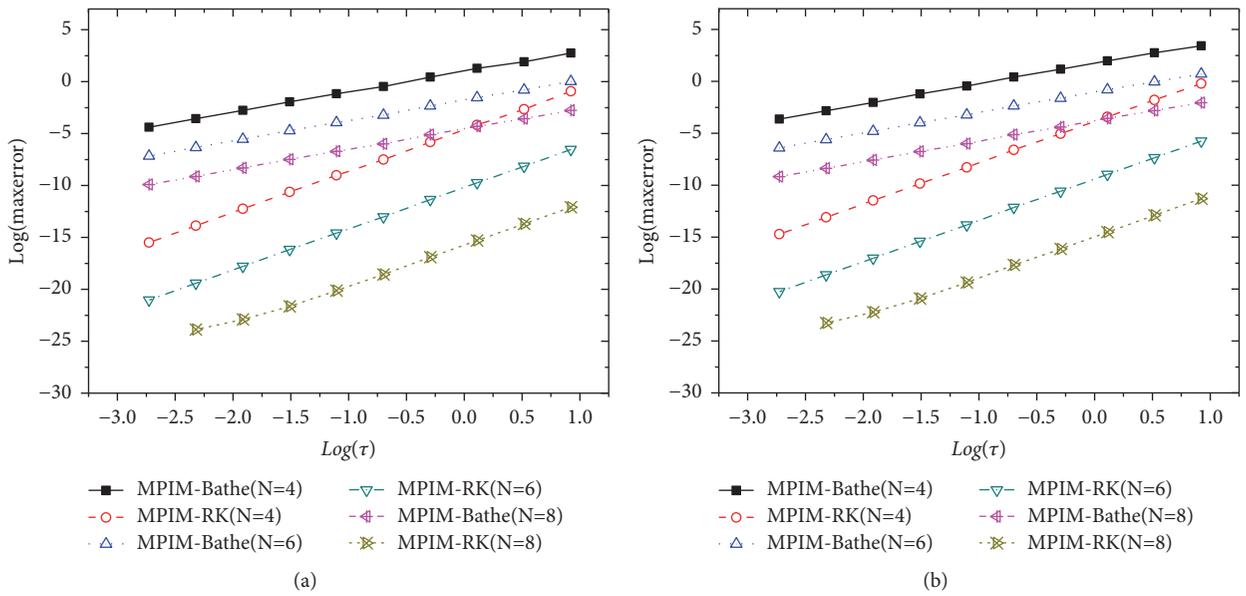


FIGURE 11: Log of maxerror versus log of time-step: (a) displacement of DOF2; (b) velocity of DOF2.

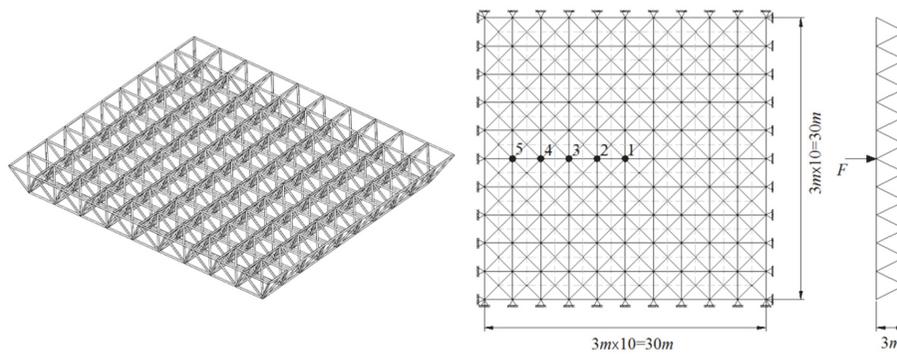


FIGURE 12: Regular diagonally positioned pyramid.

TABLE 1: CPU time (s) occupied by different methods.

	MPIM-RK4	RK4	MPIM-Bathe	Bathe
N=5	13.101	373.608	9.969	149.257
N=10	8.961		5.534	

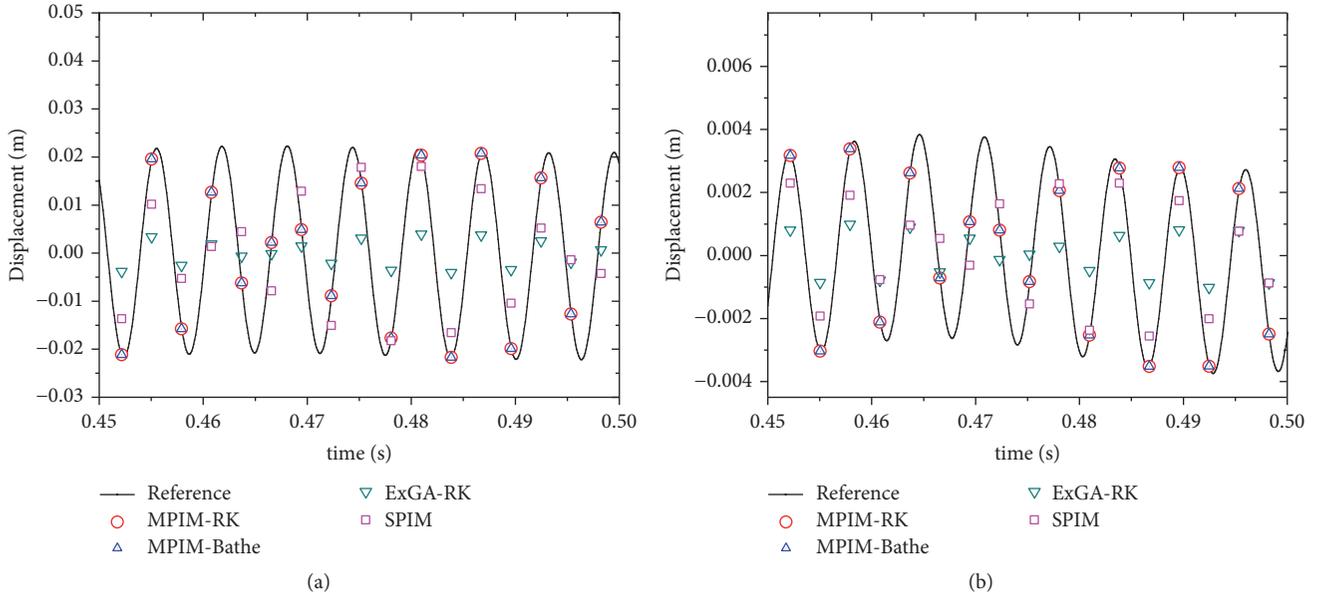


FIGURE 13: Simulation results for the vertical displacement: (a) node 1; (b)node 2.

to be unchanged within each integral interval. As a result, the accuracy of the particular solution obtained would be inadequate compared to that of the homogeneous solution.

It is widely known that the computational effort required is inversely proportional to the size of time-step in the structural dynamic analysis. Compared to these aforementioned approaches, the proposed method can use a large time-step and can preserve the same accuracy of particular solution as that of the homogeneous solution; thus it is more suitable for the long-time duration dynamic analysis.

### 6. Conclusions

An improved version of PIM is presented in this paper. This approach replaces the Taylor series expansion or the generalized Padé approximation of the matrix exponential function in the conventional PIMs with numerical fundamental solution matrix and employs loading operator matrix to obtain particular solutions. Efficient recurrence expressions are developed to evaluate the loading operator matrices in the first time-step. The highlights of the approach can be summarized as follows,

- (1) The proposed method avoids matrix inversion operations and numerical quadrature formulae; thus it expands the scope of application. Meanwhile, the accuracy of particular solution obtained can be maintained the same as that of the homogeneous solution even for a large time-step.

- (2) As the intermediate variables required for evaluation of the loading operator matrix have already been generated in calculating the fundamental solution matrix, the proposed method improves the computational efficiency of conventional PIMs.
- (3) By selecting an appropriate original method and time substep, the satisfactory numerical properties, such as accuracy, numerical dissipation, and unconditional stability, can be obtained.
- (4) Although the proposed method uses a large time-step, it can obtain almost the same numerical results as the original method using a small time-step. Thus, the proposed method is actually an accelerator of conventional time integration methods.

### Data Availability

No data were used to support this study.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant nos. 11672338 and 11202247).

## References

- [1] K. J. Bathe, *Finite Element Procedures*, Prentice Hall, 1996.
- [2] K.-J. Bathe and M. M. I. Baig, "On a composite implicit time integration procedure for nonlinear dynamics," *Computers & Structures*, vol. 83, no. 31-32, pp. 2513–2524, 2005.
- [3] S. Shojaee, S. Rostami, and A. Abbasi, "An unconditionally stable implicit time integration algorithm: Modified quartic B-spline method," *Computers & Structures*, vol. 153, pp. 98–111, 2015.
- [4] C. Huang and M. Fu, "A composite collocation method with low-period elongation for structural dynamics problems," *Computers & Structures*, vol. 195, pp. 74–84, 2018.
- [5] W. X. Zhong and F. W. Williams, "Precise time step integration method," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 208, no. 6, pp. 427–430, 1994.
- [6] J. H. Lin, W. P. Shen, and F. W. Williams, "A high precision direct integration scheme for structures subjected to transient dynamic loading," *Computers & Structures*, vol. 56, no. 1, pp. 113–120, 1995.
- [7] Y. X. Gu, B. S. Chen, H. W. Zhang, and Z. Guan, "Precise time-integration method with dimensional expanding for structural dynamic equations," *AIAA Journal*, vol. 39, no. 12, pp. 2394–2399, 2001.
- [8] M.-F. Wang and F. T. Au, "Precise integration methods based on Lagrange piecewise interpolation polynomials," *International Journal for Numerical Methods in Engineering*, vol. 77, no. 7, pp. 998–1014, 2009.
- [9] M. Wang and F. T. K. Au, "Assessment and improvement of precise time step integration method," *Computers structures*, vol. 84, no. 12, pp. 779–786, 2006.
- [10] F. M. Wang and F. T. K. Au, "On the precise integration methods based on Pad $\ominus$  approximations," *Computers & Structures*, vol. 87, no. 5, pp. 380–390, 2009.
- [11] T. C. Fung, "A precise time-step integration method by step-response and impulsive-response matrices for dynamic problems," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 24, pp. 4501–4527, 1997.
- [12] T. C. Fung and Z. L. Chen, "Krylov precise time-step integration method," *International Journal for Numerical Methods in Engineering*, vol. 68, no. 11, pp. 1115–1136, 2006.
- [13] M.-F. Wang, "Reduced-order precise integration methods for structural dynamic equations," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 27, no. 10, pp. 1569–1582, 2011.
- [14] Q. Gao, F. Wu, H. W. Zhang, W. X. Zhong, W. P. Howson, and F. W. Williams, "A fast precise integration method for structural dynamics problems," *Structural Engineering and Mechanics*, vol. 43, no. 1, pp. 1–13, 2012.
- [15] P. Hsieh and Y. Sibuya, *Basic Theory of Ordinary Differential Equations*, Springer Science & Business Media, 1999.
- [16] W. J. Mansur, F. S. Loureiro, J. Soares, and C. Dors, "Explicit time-domain approaches based on numerical Green's functions computed by finite differences-The ExGA family," *Journal of Computational Physics*, vol. 227, no. 1, pp. 851–870, 2007.
- [17] D. Soares and W. J. Mansur, "A time domain FEM approach based on implicit Green's functions for non-linear dynamic analysis," *International Journal for Numerical Methods in Engineering*, vol. 62, no. 5, pp. 664–681, 2005.
- [18] D. Soares, "A new family of time marching procedures based on Greens function matrices," *Computers & structures*, vol. 89, no. 1, pp. 266–276, 2011.
- [19] F. N. Correa, B. P. Jacob, and W. J. Mansur, "Formulation of an efficient hybrid time-frequency domain solution procedure for linear structural dynamic problems," *Computers & structures*, vol. 88, pp. 331–346, 2010.

