Structural Engineering and Mechanics, Vol. 43, No. 1 (2012) 1-13
DOI: http://dx.doi.org/10.12989/sem.2012.43.1.001

A fast precise integration method for structural dynamics problems

Q. Gao[1a], F. Wu[1b], H.W. Zhang*[1], W.X. Zhong[1c], W.P. Howson[2d] and F.W. Williams[2e]

[1]*State Key Laboratory of Structural Analysis for Industrial Equipment, Department of Engineering Mechanics, Faculty of Vehicle Engineering and Mechanics, Dalian University of Technology, Dalian 116023, P.R. China*
[2]*Cardiff School of Engineering, Cardiff University, Cardiff CF24 3AA, UK*

(Received September 18, 2011, Revised January 4, 2012, Accepted May 16, 2012)

Abstract. A fast precise integration method (FPIM) is proposed for solving structural dynamics problems. It is based on the original precise integration method (PIM) that utilizes the sparse nature of the system matrices and especially the physical features found in structural dynamics problems. A physical interpretation of the matrix exponential is given, which leads to an efficient algorithm for both its evaluation and subsequently the solution of large-scale structural dynamics problems. The proposed algorithm is accurate, efficient and requires less computer storage than previous techniques.

Keywords: structural dynamics system; sparse matrix; precise integration method; matrix exponential; fast algorithm

1. Introduction

The dynamic responses of structural problems are usually analyzed by means of numerical integration methods (Bathe and Wilson 1976). In order to achieve stable and accurate solutions with reasonable computational effort, a careful choice of time step is required, based on the natural periods of the structures and the variation of the external loading. For linear structural dynamics problems, the most popular and effective way of computing the dynamic responses are the Newmark (1959) and Runge-Kutta (Hairer *et al.* 1993, Hairer and Wanner 1996, Hairer *et al.* 2006) methods. Both approaches require the use of small time steps to achieve high precision or good stability. Alternative ways of establishing the dynamic response are to use the precise integration method (PIM) (Zhong and Williams 1994, Zhang *et al.* 2001, Zhong 2004) or the scaling and squaring method (SSM) (Moler and Loan 1978, 2003). Both of these methods rely on the accurate computation of the matrix exponential, but PIM only computes the incremental part, with the result

*Corresponding author, Professor, E-mail: zhanghw@dlut.edu.cn
[a]E-mail: qgao@dlut.edu.cn
[b]E-mail: wufengchn2@gmail.com
[c]E-mail: zwoffice@dlut.edu.cn
[d]E-mail: Howson@Cardiff.ac.uk
[e]E-mail: WilliamsFW@cf.ac.uk

that stability and accuracy are improved. In general, PIM is very accurate and allows the use of larger time steps when computing the dynamic response of structural systems.

Lin *et al.* (1995) extended PIM to handle more general loading forms. Gu *et al.* (2001) avoided the inversion of the system matrix by transforming the governing equations of the original PIM into a homogeneous form, at the expense of expanding the dimension of the problem. Wang and Au (2007) avoided the inversion by instead using Crout decomposition.

PIM has been applied to: random response analysis of structural problems (Lin *et al.* 1997, Leung 2001); to solving time-dependent partial differential equations (Zhong *et al.* 1996); to computing the Floquet transition matrix (Cai *et al.* 2001); to solving dynamic elastic-plastic hardening and softening problems (Zhang *et al.* 2003); to solving nonlinear transient heat conduction problems (Chen *et al.* 2001); to solving nonlinear partial differential equations (Mei *et al.* 2005); to calculating the electromagnetic transients in single-phase non-uniform transmission line (Jia *et al.* 2008) and; to solving singularly perturbed boundary-value problems (Fu *et al.* 2010).

Some modified PIMs have been proposed to reduce the computational effort, as follows. Shen *et al.* (1995) improved the efficiency of PIM by using a parallel computation method. Fung (1997) presented a modified PIM which can halve the computational effort by using step-response and impulsive-response matrices and by considering the symmetry of the matrices. The numerical stability and the computation accuracy of PIM were discussed in detail by Wang and Au (2006). The Krylov subspace method has been applied to modify the original PIM in order to improve its computational efficiency (Fung 2006). A modified precise integration methods based on Lagrange piecewise interpolation polynomials is proposed to transform the non-homogeneous initial system into the homogeneous dynamic system without dimensional expansion. This method avoids the matrix inversion operation and is a general homogenized high-precision direct integration scheme (Wang and Au 2009). Finally, a precise integration method using the Ritz vectors and a modified Krylov precise integration method are presented to improve the computational efficiency (Wang 2011). However, for large-scale systems, computing the matrix exponential can still be time consuming and so there is scope for improving the efficiency of PIM, as follows.

The theory developed herein seeks to reduce the computational effort involved in establishing the matrix exponential of large structural systems. It is based on PIM and utilizes both the special nature of the system and the fact that the speed of energy propagation within the system is finite. Importantly, physical interpretation of the matrix exponential leads to the conclusion that, contrary to mathematical expectation, it has the form of a sparse matrix. In turn, this leads to an efficient and accurate algorithm for computing the dynamic response of large structural systems.

2. PIM for structural dynamics problems

Suppose that the stiffness, damping and mass matrices of the structural system are **K**, **C** and **M**, respectively, then the motion equation can be written as

$$\mathbf{M}\ddot{\mathbf{y}} + \mathbf{C}\dot{\mathbf{y}} + \mathbf{K}\mathbf{y} = \mathbf{f}(t) \tag{1}$$

in which $\mathbf{f}(t)$ is the external force vector. Eq. (1) can then be written in state space form as

$$\dot{\mathbf{v}} = \mathbf{H}\mathbf{v} + \begin{Bmatrix} \mathbf{0} \\ \mathbf{f} \end{Bmatrix} \tag{2}$$

where

$$\mathbf{v} = \begin{Bmatrix} \mathbf{y} \\ \mathbf{p} \end{Bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{0} & \mathbf{M}^{-1} \\ -\mathbf{K} & -\mathbf{C}\mathbf{M}^{-1} \end{bmatrix} \tag{3}$$

and $\mathbf{p} = \mathbf{M}\dot{\mathbf{y}}$ is the momentum.

The Newmark (Newmark 1959) and Runge-Kutta (Hairer *et al.* 1993, Hairer and Wanner 1996, Hairer *et al.* 2006) methods are the most popular ways to integrate Eq. (1) or (2). However, they both require small time steps to achieve high precision or good stability. Alternative methods are PIM (Zhong and Williams 1994, Zhang *et al.* 2001, Zhong 2004) and SSM (Moler and Loan 1978, 2003), which both require accurate computation of the matrix exponential. However, for large-scale dynamics systems, matrix exponentiation is very time consuming. In the remainder of this section, the ideas underpinning the original development of PIM are reviewed, while in Section 3 an efficient version of PIM is proposed for performing fast matrix exponentiation.

For numerical integration, the time domain is divided into a series of time intervals with equal time step η, i.e.

$$t_0 = 0, \quad t_1 = \eta, \quad \ldots, \quad t_k = k\eta, \quad t_{k+1} = (k+1)\eta, \ldots \tag{4}$$

If the state vector in time t_k is represented as

$$\mathbf{v}_k = \mathbf{v}(t_k) \tag{5}$$

The solution of Eq. (2) can be given by the following equation

$$\mathbf{v}_{k+1} = \mathbf{T}\mathbf{v}_k + \int_0^{\eta} \exp[\mathbf{H}(\eta - \xi)] \begin{Bmatrix} \mathbf{0} \\ \mathbf{f}(t_k + \xi) \end{Bmatrix} d\xi \tag{6}$$

in which

$$\mathbf{T} = \exp(\mathbf{H}\eta) \tag{7}$$

where $\mathbf{T}$ defines the exponential of matrix $\mathbf{H}\eta$. From Eq. (6), it is clear that $\mathbf{T}$ and the time step integration need to be computed efficiently and accurately. In the latter case, the integration can often be achieved analytically. For example, if the external force is a polynomial in the time step, i.e., $\mathbf{f}(t) = \mathbf{f}_0 + \mathbf{f}_1 t + \ldots + \mathbf{f}_m t^m$, Eq. (6) can be written as

$$\mathbf{v}_{k+1} = \mathbf{T}\mathbf{v}_k + \Psi_0\mathbf{f}_0 + \Psi_1\mathbf{f}_1 + \ldots + \Psi_m\mathbf{f}_m \tag{8}$$

in which

$$\Psi_k = \begin{bmatrix} \Psi_k^{12} \\ \Psi_k^{22} \end{bmatrix}, \quad (k = 0, 1, \ldots, m)$$

$$\Psi_0^{12} = \mathbf{K}^{-1}(\mathbf{I} - \mathbf{C}\mathbf{T}_{12} - \mathbf{T}_{22}), \quad \Psi_0^{22} = \mathbf{M}\mathbf{T}_{12}$$

$$\Psi_k^{12} = \mathbf{K}^{-1}[\eta^k\mathbf{I} - k(\mathbf{C}\Psi_{k-1}^{12} + \Psi_{k-1}^{22})], \quad \Psi_k^{22} = k\mathbf{M}\Psi_{k-1}^{12}, \quad (k = 1, 2, \ldots, m) \tag{9}$$

where matrices $\mathbf{T}_{12}$ and $\mathbf{T}_{22}$ are the sub-matrices of $\mathbf{T}$, i.e.

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{T}_{21} & \mathbf{T}_{22} \end{bmatrix} \tag{10}$$

If the external force is harmonic with time, i.e., $\mathbf{f}(t) = \mathbf{r}\exp(i\omega t)$, Eq. (6) can be written as

$$\mathbf{v}_{k+1} = \mathbf{T}\mathbf{v}_k + \exp(i\omega t_k)(i\omega\mathbf{I} - \mathbf{H})^{-1}\left[\begin{Bmatrix} \mathbf{0} \\ \exp(i\omega\eta)\mathbf{I} \end{Bmatrix} - \begin{Bmatrix} \mathbf{T}_{12} \\ \mathbf{T}_{22} \end{Bmatrix}\right]\mathbf{r} \tag{11}$$

in which $i = \sqrt{-1}$.

For the two common cases above, computing $\mathbf{T}$ efficiently and accurately is clearly the key issue when computing dynamic responses. This is most likely to be achieved using either PIM (Zhong and Williams 1994, Zhang *et al.* 2001, Zhong 2004) or SSM (Moler and Loan 1978, 2003). Based on the addition theorem, SSM gives a simple and effective procedure for computing $\mathbf{T}$, but it can be inaccurate in certain situations. On the other hand, PIM, which is also based on the addition theorem, further uses the idea of only computing the incremental part of $\mathbf{T}$ in order to improve the computational precision.

The implementation of PIM for computing the exponential of matrix $\mathbf{H}$ and time step η can be given as follows (Zhong and Williams 1994, Zhang *et al.* 2001, Zhong 2004). Let

$$\mathbf{H}' = \mathbf{H}\frac{\eta}{2^N} \tag{12}$$

in which N is a integer. If N is big enough, the norm of $\mathbf{H}'$ is very small, and the exponential of matrix $\mathbf{H}'$ can be computed approximately by the Taylor series of order q, i.e.

$$\exp(\mathbf{H}') \approx \Phi = \mathbf{I} + \mathbf{H}' + \frac{(\mathbf{H}')^2}{2!} + \ldots + \frac{(\mathbf{H}')^q}{q!} \tag{13}$$

Φ is now divided into two parts

$$\Phi = \mathbf{I} + \mathbf{R}, \quad \mathbf{R} = \mathbf{H}' + \frac{(\mathbf{H}')^2}{2!} + \ldots + \frac{(\mathbf{H}')^q}{q!} \tag{14}$$

and the addition theorem is applied to the incremental part $\mathbf{R}$ to give

$$\mathbf{R} = \mathbf{R}^2 + 2\mathbf{R} \tag{15}$$

The exponential of matrix $\mathbf{H}\eta$ can now be obtained by performing Eq. (15) N times and finally adding the unit matrix to $\mathbf{R}$.

PIM is an accurate algorithm that is easy to program. However, the computational effort of PIM is $O(n^3)$, which can therefore be very time consuming for large systems. In the next section, the special form of the matrix exponential in dynamics problems is utilized to yield a fast precise integration method (FPIM) for its computation.

3. FPIM for large-scale dynamics systems

For large-scale structural dynamics systems, it is well known that the mass, stiffness and damping matrices, and hence matrix **H**, are sparse. However, the exponential of a general sparse matrix is not sparse mathematically. Therefore, the direct application of PIM cannot use the sparse property of matrix **H**. In this section, by recognizing the physical nature of the structural dynamics system, it is demonstrated that the matrix exponential is in fact physically sparse. By using this property, an FPIM algorithm is proposed to compute the matrix exponential efficiently.

If there are no external forces, Eq. (6) can be written in block form as

$$\begin{aligned} \mathbf{y}_{k+1} &= \mathbf{T}_{11}\mathbf{y}_k + \mathbf{T}_{12}\mathbf{p}_k \\ \mathbf{p}_{k+1} &= \mathbf{T}_{21}\mathbf{y}_k + \mathbf{T}_{22}\mathbf{p}_k \end{aligned} \tag{16}$$

Eq. (16) shows the physical meaning of the matrix exponential, i.e., if the initial displacement of the j-th DOF is one, the initial displacement of the remaining DOFs are zero and the initial momentums of all DOFs are zero. Furthermore the displacement and momentum responses of the i-th DOF at $t = \eta$ are $\mathbf{T}_{11}(i,j)$ and $\mathbf{T}_{21}(i,j)$, respectively. In similar vein, if the initial displacements of all DOFs are zero, the initial momentum of the j-th DOF is one and the initial momentums of other DOFs are zero. The displacement and momentum responses of the i-th DOF at $t = \eta$ are then $\mathbf{T}_{12}(i,j)$ and $\mathbf{T}_{22}(i,j)$, respectively.

It is well known that the propagation speed of the energy in a one-dimensional rod is $\sqrt{E/\rho}$ and so is finite. The propagation speed of the energy in a structural dynamics system is similarly finite, with the result that within a small time step, the excitation of a specific DOF can only affect the adjoining DOFs of the system. Thus from the physical meaning of the matrix exponential above, $\mathbf{T}_{11}, \mathbf{T}_{12}, \mathbf{T}_{21}$ and $\mathbf{T}_{22}$ must be sparse matrices and can be stored conveniently as such in coordinate list (COO) format.

For a given matrix **H** and time step η, the procedure of the original PIM for computing the matrix exponential can be given as follows. Initially values are allocated to two integers, N and q, using a procedure described in detail by Moler and Loan (1978, 2003), Zhang *et al.* (2001). These values ensure that the relative error of $\exp(\mathbf{H}\eta)$ is less than a predefined tolerance. Next we let $\mathbf{H}' = \mathbf{H}\frac{\eta}{2^N}$ and compute **R** according to Eq. (14). Eq. (15) is then processed N times and the final step is to add the unit matrix to **R**, which yields the required matrix exponential.

Despite the fact that matrix **H** is sparse, matrix **R** when computed from Eq. (14) will become a dense matrix. Moreover, following each evaluation of Eq. (15), **R** becomes more dense. However, if we observe matrix **R** carefully, it can be seen that many elements of **R** are very close to zero. However, matrix **R** corresponds to the exponential of a matrix evaluated at small time steps and so according to the physical meaning of the matrix exponential, it must be a sparse matrix. The small elements of matrix **R** must therefore be caused by numerical errors, and hence should be zeros. Therefore, in the direct application of PIM, many elements that should be zero are involved in the matrix multiplication, with the resulting waste in computational effort.

The arguments above necessitate a procedure for transforming the dense matrix **R** to sparse form, which in turn leads to an algorithm (FPIM) for computing the matrix exponential efficiently.

The detailed procedure for transforming **R** is as follows. The physical meaning of **R** enables it to be divided into the four sub-matrices $\mathbf{R}_{11}$, $\mathbf{R}_{12}$, $\mathbf{R}_{21}$ and $\mathbf{R}_{22}$. Now calculate α_{11}, the maximum absolute value of all elements comprising $\mathbf{R}_{11}$, and set a tolerance, typically $\varepsilon = 10^{-25}$. Then, if the

absolute value of any element in $\mathbf{R}_{11}$ is less than $\varepsilon \times \alpha_{11}$, set the element to zero. By using this procedure, $\mathbf{R}_{11}$ is transformed into a sparse matrix and so can be stored in sparse form. The matrices $\mathbf{R}_{12}$, $\mathbf{R}_{21}$ and $\mathbf{R}_{22}$ can then be transformed in similar fashion.

Therefore, for given matrix $\mathbf{H}$ and time step η, the following FPIM can be used to compute the matrix exponential.

The algorithm of FPIM:

1. Store the matrix $\mathbf{H}$ in sparse format;
2. Determine two integers N and q (Moler and Loan 1978, 2003, Zhang *et al.* 2001);
3. Let $\mathbf{H}' = \mathbf{H}\dfrac{\eta}{2^N}$;
4. $\mathbf{R} = \mathbf{H}' + \dfrac{(\mathbf{H}')^2}{2!} + \ldots + \dfrac{(\mathbf{H}')^q}{q!}$;
5. Transform matrix $\mathbf{R}$ to sparse format;
6. Perform the following steps

   For iter = 1 : N

   R = R * (R + 2 * I);

   Transform matrix $\mathbf{R}$ to sparse format;

   End
7. Adding matrix $\mathbf{R}$ to the unit matrix gives the exponential of matrix $\mathbf{H}\eta$.

Comparing the procedure of FPIM and PIM, it can be see that the only difference between them is the addition of a procedure for transforming matrix $\mathbf{R}$ to sparse form. However, this simple addition can improve significantly the computational efficiency, as shown by the comparisons given in the numerical examples.

4. Matrices related to external forces

In addition to calculating the matrix exponential, matrices related to the external forces must often be computed. For example, if the external forces are polynomial, the matrix Ψ_k defined by Eq. (9) should be computed. The method is shown below and is easily adapted for other types of external force.

As in the case of the matrix exponential, the physical meaning of the matrix Ψ_k is given first. From Eq. (9), the $2n \times n$ matrix Ψ_k can be written in block form as

$$\Psi_k = \begin{Bmatrix} \Psi_k^{11} \\ \Psi_k^{21} \end{Bmatrix} \tag{17}$$

According to Eq. (8), if the initial displacements and momentums are all zero and the external force $t^k (k = 0, 1, \ldots)$ is applied at the j-th DOF, after one time step, the displacement and momentum responses of the i-th DOF are $\Psi_k^{11}(i,j)$ and $\Psi_k^{21}(i,j)$, respectively.

Adopting a similar approach to that used when considering the matrix exponential and noting that

the propagation speed of energy is finite, matrices Ψ_k^{11} and Ψ_k^{21} are also sparse matrices. In fact, for a linear system, the propagation speed of the energy is only related to the structure itself and is not related to the initial conditions or external forces. Therefore, the structure of the matrix related to external force is the same as the matrix exponential and can be computed using equivalent procedures.

5. Numerical examples

Example 1: Consider the mass-spring-damper system shown in Fig. 1. The system consists of 2001 masses and 2002 springs, and the mass and the stiffness are $m = 1$(kg) and $k = 10$(N/m), respectively. The mass matrix **M** and the stiffness matrix **K** can be formed easily, and the damping matrix is taken as $\mathbf{C} = 0.05\mathbf{K}$. The initial displacement of node 1001 is one; the initial displacement of all other nodes is zero; the initial momentum of all nodes is zero and there are no external forces. The interval of integration is from 0 to 1000(s).

The dynamic responses are computed using four methods: the proposed method (FPIM), PIM, the variable step Runge-Kutta method (R-K) (Hairer *et al.* 1993, Hairer and Wanner 1996, Hairer *et al.* 2006) (ODE45 in MATLAB) and the Newmark method (Newmark 1959). A 1(s) time step is used for both FPIM and PIM, three different time steps of 0.1(s), 0.01(s) and 0.001(s) are used for the Newmark method. 10^{-13} is used for both the absolute and relative tolerances in ODE45. The numerical results computed using ODE45 are used as the reference solution. The relative errors of displacement and momentum are defined, respectively, as

$$e_d = \frac{\|\mathbf{y} - \mathbf{y}_{RK}\|_2}{\|\mathbf{y}_{RK}\|_2}, \quad e_m = \frac{\|\mathbf{p} - \mathbf{p}_{RK}\|_2}{\|\mathbf{p}_{RK}\|_2} \tag{18}$$

in which $\mathbf{y}_{RK}$ and $\mathbf{p}_{RK}$ are the displacements and momentums of all DOFs at a given time computed using ODE45 and **y** and **p** are the displacements and momentums of all DOFs at a given time computed using FPIM, PIM or the Newmark method.

For this example, the two values of N and q for FPIM are 12 and 8, respectively. The displacements and momentums of all nodes at 1000(s) computed using FPIM are given in Fig. 2. The relative errors of displacement and momentum are given in Fig. 3, in which the solids lines with circle and "x" are the relative errors of FPIM and PIM, respectively, and the solid lines with right triangle, left triangle and down triangle are the relative errors of the Newmark method with time step 0.1(s), 0.01(s) and 0.001(s), respectively. Fig. 3 shows that both FPIM and PIM are very accurate, with the precision of FPIM being very close to that of PIM, which shows the validity of the proposed method. Fig. 3 also shows that the Newmark method with three smaller time steps does not achieve the precision of FPIM.

The CPU times for FPIM, R-K, PIM and the Newmark method are given in Table 1. It is clear

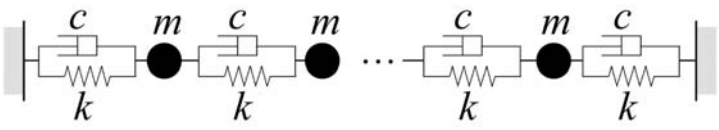

Fig. 1 A mass-spring-damper system

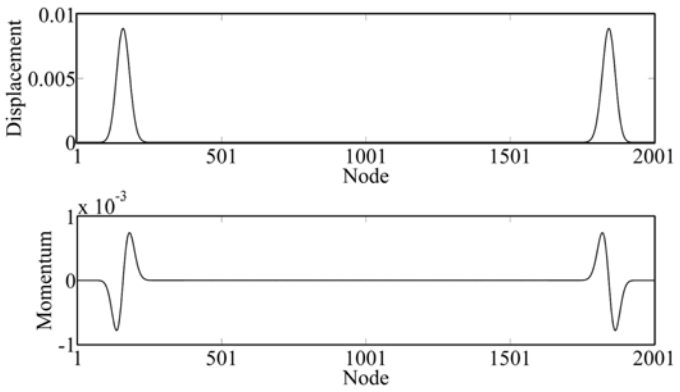

Fig. 2 Results from the proposed method for the displacement and momentum responses at $t = 1000$(s)

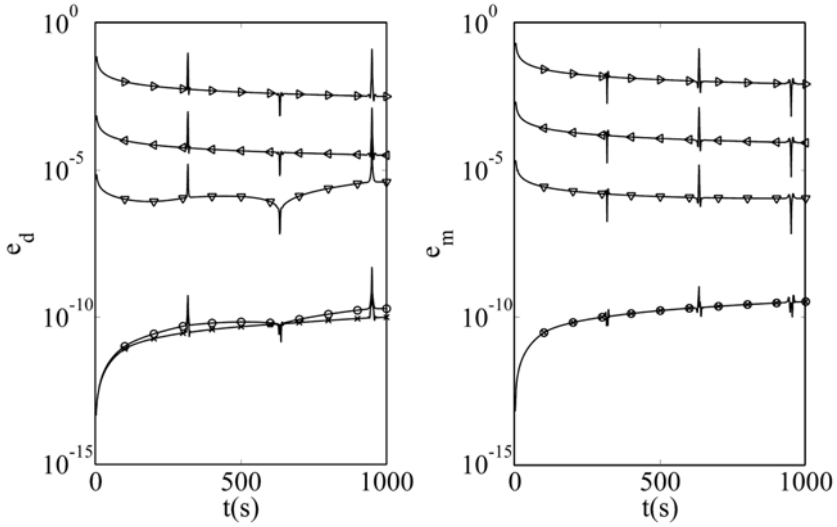

Fig. 3 The relative errors of the results for the FPIM (circle), PIM (x-mark) and Newmark (right triangle for time step 0.1, left triangle for time step 0.01 and down triangle for time step 0.001) methods for Example 1

Table 1 Comparison of CPU times for Example 1

	FPIM	R-K	PIM	Newmark		
				0.1	0.01	0.001
CPU time (s)	2.6	63.1	153.3	3.8	35.8	353.1

that FPIM is significantly more efficient than the original PIM, R-K and Newmark methods.

Example 2: The dynamic response of a plane stress problem is now considered. The structure consists of two different materials with the same density $\rho = 8 \times 10^3 (\text{kg/m}^3)$ and Poisson's ratio $\mu = 0.2$, but different Young's modulus $E_1 = 3 \times 10^{11} (\text{N/m})$ and $E_2 = 2 \times 10^{11} (\text{N/m})$. The

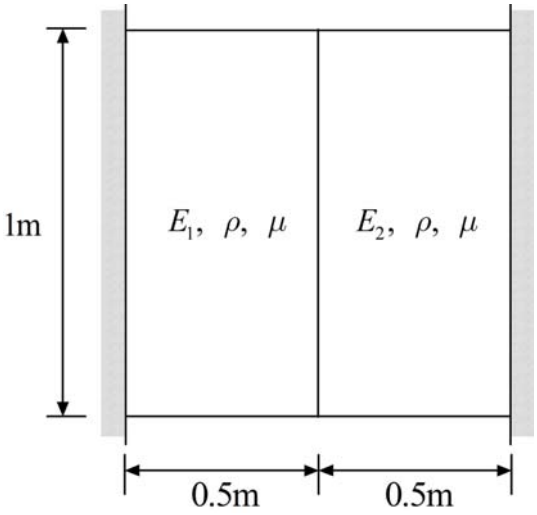

Fig. 4 The plane stress problem

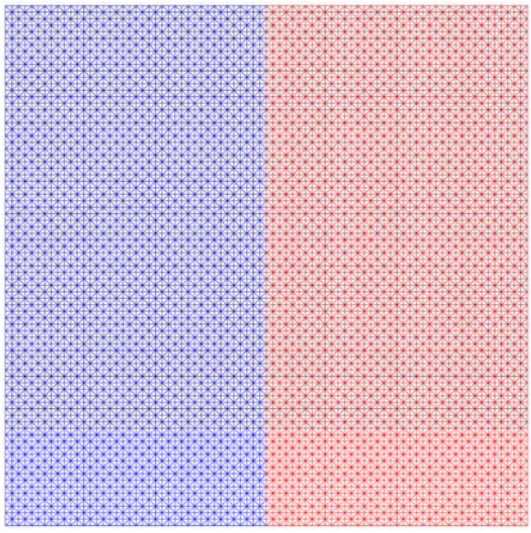

Fig. 5 The FEM mesh

structure is clamped at the left and right ends, see Fig. 4.

The structure is discretized using FEM and the mesh is shown in Fig. 5. The whole structure has 12800 elements, 12960 DOFs and a lumped mass matrix is used. The initial displacements and momentums of all DOFs are zero, and constant horizontal and vertical external forces of 10^5(N) are applied at all nodes.

The interval of integration is from 0 to 5×10^{-3}(s), and the dynamic responses are computed using FPIM, R-K and the Newmark method, respectively. Time steps of 10^{-6}(s) and 10^{-7}(s) are used for FPIM and the Newmark method, respectively, while 10^{-13} is used for both the absolute and relative tolerances in the R-K method (ODE45).

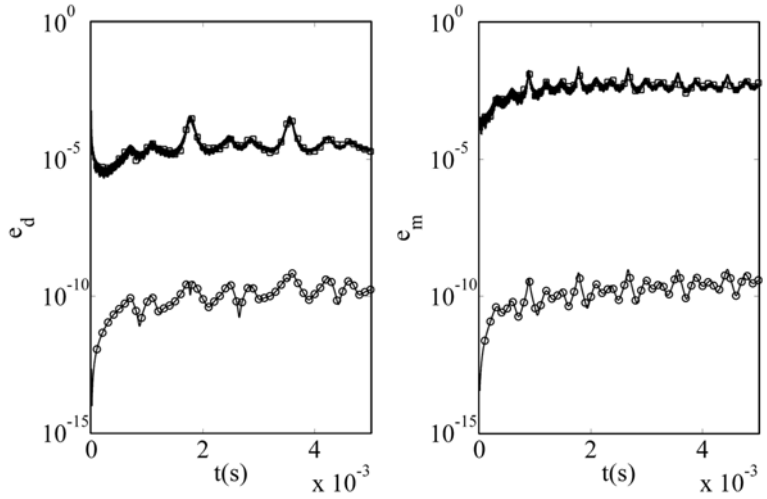

Fig. 6 The relative errors of the results for the FPIM (circle) and Newmark (square) methods for Example 2

Table 2 Comparison of CPU times for Example 2

	FPIM	R-K	Newmark
CPU time (s)	955.9	4975.8	4253.8

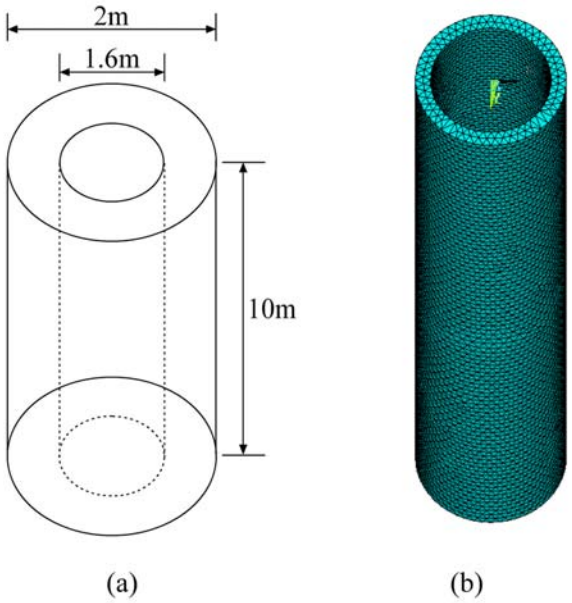

Fig. 7 The three dimensional hollow cylinder and FEM mesh

For this example, the two values of N and q for FPIM are 28 and 9, respectively. The relative errors of the displacement and momentum computed using Eq. (18) are given in Fig. 6, in which the solid lines with circle and square denote FPIM and the Newmark method, respectively. The

figure shows that FPIM is again very accurate and that the Newmark method, even with a ten times smaller time step, does not achieve the precision of FPIM.

The CPU times for FPIM, R-K and the Newmark method are given in Table 2, which shows that the proposed method is much more efficient than the R-K and Newmark methods.

Example 3: The three dimensional dynamic response of the hollow cylinder shown in Fig. 7(a) is now considered. The density, Poisson's ratio and Young's modulus are given by

$$\rho = 2\times 10^{3}(\mathrm{kg/m^{3}}),\ \ \mu = 0.2,\ \ E = 2\times 10^{10}(\mathrm{N/m})$$

This structure is discretized using FEM and the mesh is shown in Fig. 7(b). All DOFs corresponding to the nodes at the base are fixed. The whole structure has 34068 DOFs and a lumped mass matrix is used. The initial displacements and momentums of all DOFs are zero and external forces of $10^{6}[\cos(20t)+\sin(20t)](\mathrm{N})$ are applied at all nodes in the three directions.

The interval of integration is from 0 to 0.04(s), and the dynamic responses are computed once more using FPIM, R-K and the Newmark method, respectively. As before, time steps of 10^{-6}(s) and 10^{-7}(s) are used for FPIM and the Newmark method, respectively, while 10^{-13} is used for both the absolute and relative tolerances in the R-K method (ODE45).

For this example, the two values of N and q for FPIM are 23 and 8, respectively. The relative errors of displacement and momentum computed using Eq. (18) are given in Fig. 8, in which the solid lines with circle and triangle denote FPIM and the Newmark method, respectively. The figure also clearly demonstrates the effect of variable external forces and the unrivalled precision of FPIM,

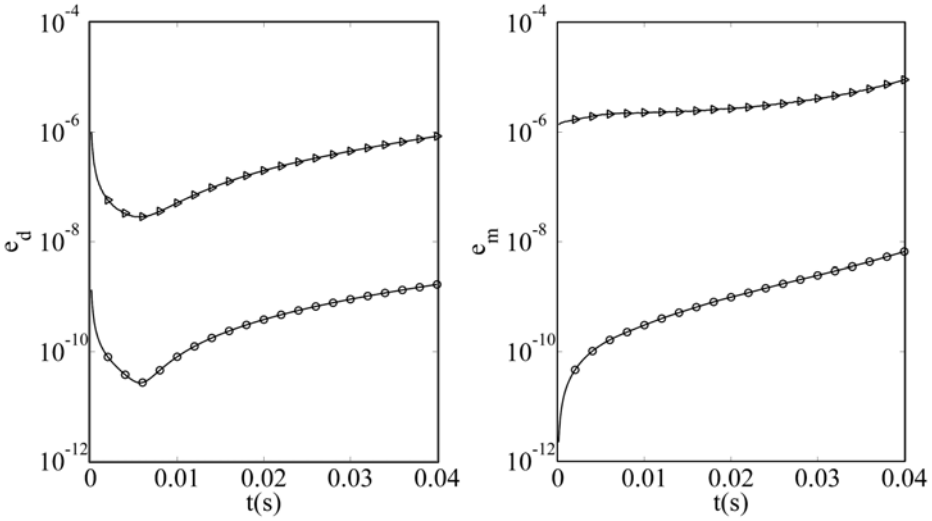

Fig. 8 The relative errors of the results for the FPIM (circle) and Newmark (triangle) methods for Example 3

Table 3 Comparison of CPU times for Example 3

	FPIM	R-K	Newmark
CPU time (s)	12470.3	146360.4	82863.7

which is more accurate than the Newmark method, even when a smaller time step is used.

The CPU times for FPIM, R-K and the Newmark method are given in Table 3, which again shows that FPIM is more efficient than both the R-K and Newmark methods.

6. Conclusions

A fast precise integration method (FPIM) for computing the responses in structural dynamics problems has been proposed. By using the energy propagation feature of the system, it is shown that there are many zero elements in the resulting matrix exponential. This leads to an efficient and accurate algorithm for both its evaluation and subsequently the solution of large scale structural dynamics problems. Importantly, the method avoids computing and sorting many zero elements, with the result that it is computationally efficient in terms of both time and storage. The numerical examples illustrate that the proposed method is fast and accurate.

Acknowledgements

The authors are grateful for the support of: the Natural Science Foundation of China (No 10902020, No 10721062); 863 program (No 2009AA044501); 973 program (No 2011CB711105); the Scientific Research and Development Project of the Ministry of Railways (2010T001-C); the Doctoral Research Fund of Liaoning (No 20081091); Key Laboratory Fund of Liaoning (No 2009S018) and; the Cardiff Advanced Chinese Engineering centre.

References

Bathe, K.J. and Wilson, E.L. (1976), *Numerical Methods in Finite Element Analysis*, Prentice-Hall, New Jersey.

Cai, Z.Q., Gu, Y.X. and Zhong, W.X. (2001), "A new approach of computing Floquet transition matrix", *Comput. Struct.*, **79**(6), 631-635.

Chen, B.S., Gu, Y.X., Guan, Z.Q. and Zhang, H.W. (2001), "Nonlinear transient heat conduction analysis with precise time integration method", *Numer. Heat Tr. B-Fund.*, **40**(4), 325-341.

Fu, M.H., Cheung, M.C. and Sheshenin, S.V. (2010), "Precise integration method for solving singular perturbation problems", *Appl. Math. Mech.*, **31**(11), 1463-1472.

Fung, T.C. (1997), "A precise time-step integration method by step-response and impulsive-response matrices for dynamic problems", *Int. J. Numer. Meth. Eng.*, **40**(24), 4501-4527.

Fung, T.C. and Chen, Z.L. (2006), "Krylov precise time-step integration method", *Int. J. Numer. Meth. Eng.*, **68**(11), 1115-1136.

Gu, Y.X., Chen, B.S., Zhang, H.W. and Guan, Z.Q. (2001), "Precise time-integration method with dimensional expanding for structural dynamic equations", *AIAA J.*, **39**(12), 2394-2399.

Hairer, E., Lubich, C. and Wanner, G. (2006), *Geometric Numerical Integration: Structure-preserving Algorithm for Ordinary Differential Equations*, 2nd Edition, Springer, New York.

Hairer, E., Norsett, S.P. and Wanner, G. (1993), *Solving Ordinary Differential Equations I-nonstiff Problems,* 2nd Edition, Springer, Berlin.

Hairer, E. and Wanner, G. (1996), *Solving Ordinary Differential Equations II-stiff and Differential- Algebraic Problems*, 2nd Edition, Springer, Berlin.

Jia, L., Shi, W. and Guo, J. (2008), "Arbitrary-difference precise-integration method for the computation of electromagnetic transients in single-phase nonuniform transmission line", *IEEE Trans. Power Del.*, **23**(3),

1488-1494.
Leung, A.Y.T. (2001), "Fast matrix exponent for deterministic or random excitations", *Int. J. Numer. Meth. Eng.*, **50**(2), 377-394.
Lin, J.H., Shen, W.P. and Williams, F.W. (1995), "A high precision direct integration scheme for structures subjected to transient dynamic loading", *Comput. Struct.*, **56**(1), 113-120.
Lin, J.H., Shen, W.P. and Williams, F.W. (1997), "Accurate high-speed computation of non-stationary random structural response", *Eng. Struct.*, **19**(7), 586-593.
Mei, S.L., Lu, Q.S., Zhang, S.W. and Jin, L. (2005), "Adaptive interval wavelet precise integration method for partial differential equations", *Appl. Math. Mech.*, **26**(3), 364-371.
Moler, C. and Loan, C.V. (1978), "Nineteen dubious ways to compute the exponential of a matrix", *SIAM Rev.,* **20**(4), 801-836.
Moler, C. and Loan, C.V. (2003), "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later", *SIAM Rev.*, **45**(1), 1-46.
Newmark, N.M. (1959), "A method of computation for structural dynamics", *J. Eng. Mech.-ASCE*, **85**(3), 67-94.
Shen, W.P., Lin, J.H. and Williams, F.W. (1995), "Parallel computing for the high precision direct integration method", *Comput. Meth. Appl. M.*, **126**(3-4), 315-331.
Wang, M.F. and Au, F.T.K. (2007), "Precise integration method without inverse matrix calculation for structural dynamic equations", *Earthq. Eng. Vib.*, **6**(1), 57-64.
Wang, M.F. and Au, F.T.K. (2006), "Assessment and improvement of precise time step integration method", *Comput. Struct.*, **84**(12), 779-786.
Wang, M.F. and Au F.T.K. (2009), "Precise integration methods based on Lagrange piecewise interpolation polynomials", *Int. J. Numer. Meth. Eng.*, **77**(7), 998-1014.
Wang, M.F. (2011), "Reduced-order precise integration methods for structural dynamic equations", *Int. J. Numer. Meth. Biomed. Eng.*, **27**(10), 1569-1582.
Zhang, H.W., Chen, B.S. and Gu, Y.X. (2001), "An adaptive algorithm of precise integration for transient analysis", *ACTA Mech. Solida Sin.,* **14**(3), 215-224.
Zhang, H.W., Zhang, X.W. and Chen, J.S. (2003), "A new algorithm for numerical solution of dynamic elastic–plastic hardening and softening problems", *Comput. Struct.*, **81**(17), 1739-1749.
Zhong, W.X. (2004), "On precise integration method", *J. Comput. Appl. Math.*, **163**(1), 59-78.
Zhong, W.X. and Williams, F.W. (1994), "A precise time step integration method", *P. I. Mech. Eng. C-J. Mec.*, **208**(6), 427-430.
Zhong, W.X., Zhu, J. and Zhong, X.X. (1996), "On a new time integration method for solving time dependent partial differential equations", *Comput. Meth. Appl. M.*, **130**(1-2), 163-178.