Report No.
UCB/SEMM- 2015/01

Structural Engineering
Mechanics and Materials

Lyapunov Stability and Accuracy of Direct Integration Algorithms in Nonlinear Dynamic Problems and Considering the Strictly Positive Real Lemma

By

Xiao Liang

Khalid M. Mosalam

April 2015

Department of Civil and Environmental Engineering
University of California, Berkeley

LYAPUNOV STABILITY AND ACCURACY OF DIRECT INTEGRATION ALGORITHMS IN NONLINEAR DYNAMIC PROBLEMS AND CONSIDERING THE STRICTLY POSITIVE REAL LEMMA

Xiao Liang[1],
Khalid M. Mosalam[2]

Abstract

In structural dynamics, direct integration algorithms are commonly used to solve the temporally discretized differential equations of motion. Numerous research efforts focused on the stability and accuracy of different integration algorithms for linear elastic structures. However, investigations of those properties applied to nonlinear structures are relatively limited. Systematic Lyapunov stability and accuracy analyses of several direct integration algorithms for nonlinear structural dynamics are presented in this report. These integration algorithms include the implicit methods of the Newmark family of integration algorithm and the TRBDF2 algorithm, as well as the explicit methods of Operator-Splitting (OS) algorithms. Two versions of the OS algorithms, using initial and tangent stiffness formulations, are investigated. The latter one is shown to possess similar stability properties to the implicit Newmark integration. Some arguments of stability regarding these direct integration algorithms from past studies are found to be incomplete. An approach that enables performing the stability analysis numerically is proposed. It transforms the stability analysis to a problem of convex optimization. In addition, another systematic approach is proposed to investigate the Lyapunov stability of explicit algorithms by means of the strictly positive real lemma. In this approach, the stability analysis is equivalent to investigating the strictly positive realness of the transfer function for the formulated system. Subsequently, Nyquist plot is used to determine the range of the system stiffness where the explicit algorithm is stable in the sense of Lyapunov. This approach is finally considered for the commonly used explicit Newmark integration algorithm applied to single degree of freedom systems with softening or stiffening behavior. Besides the stability, accuracy of the above-mentioned five integration algorithms is investigated using a geometrically nonlinear test problem, where acceptable amounts of period elongation and amplitude decay were evident.

Keywords: Accuracy, Convex Optimization, Direct Integration Algorithm, Explicit, Implicit, Lyapunov Stability, Nonlinear, Nyquist Plot, Strictly Positive Real Lemma, Structural Dynamics.

[1]PhD Candidate, 517 Davis Hall, Univ. of Calif., Berkeley, CA 94720-1710, E-mail: benliangxiao@berkeley.edu.
[2]Professor, 733 Davis Hall, Univ. of Calif., Berkeley, CA 94720-1710, E-mail: mosalam@berkeley.edu.

Introduction

The structural response under dynamic loading is governed by the differential equations of motion. In structural dynamics, direct integration algorithms are commonly used to solve these equations of motion after they are temporally discretized. The used integration algorithms for solving a structural dynamics problem are categorized into either implicit or explicit. An integration algorithm is explicit when the responses of the next time step depend on the responses of previous and current time steps only. On the other hand, implicit algorithms require iterations because the responses of the next time step depend on the responses of previous, current and also next time steps. Numerous implicit and explicit direct integration methods have been developed, including the Newmark family of algorithms (Newmark 1959), the TRBDF2 algorithm (Bathe 2007), and the Operator-Splitting (OS) algorithms (Hughes et al. 1979).

Most of the research efforts for determining the robustness of direct integration algorithms focused on the stability and accuracy for linear structures, e.g. Bathe and Wilson 1973. Studies related to the stability and accuracy of direct integration algorithms applied to nonlinear dynamic analysis are relatively limited. For example, Hughes (1976) investigated the stability of the Newmark algorithm with constant acceleration in nonlinear dynamics. Chen and Ricles (2008) explored the stability of several direct integration algorithms by utilizing discrete control theory.

In this report, two general classes of nonlinear response of structural systems, namely stiffening and softening responses, are considered. The idealized backbone curves (force-displacement relationship) of these two systems are illustrated in Fig. 1. Systematic Lyapunov stability and accuracy analyses of several implicit and explicit direct integration algorithms for these two nonlinear structural systems are presented. Some arguments of stability regarding these direct integration algorithms from past studies are found to be incomplete and these findings are discussed in this report. The report also investigates the OS algorithm that uses tangent stiffness in the formulation, which has not been previously studied. It is shown that this algorithm possesses similar stability properties to those of the implicit Newmark integration. An approach that enables performing the stability analysis numerically is proposed. It transforms this analysis to a problem of convex optimization. It is shown that the proposed approach is generally applicable to direct integration algorithms for various nonlinear behaviors.

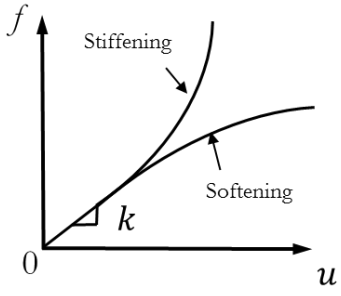


Fig. 1. Definition of Stiffening and softening systems.

In addition, another systematic approach is proposed to investigate the Lyapunov stability of explicit algorithms for the same structural systems in Fig. 1 by means of the strictly positive real lemma (Cains 1989). In this approach, a generic explicit algorithm is formulated for a nonlinear structural system that has a structural response governed by a nonlinear function of the

restoring force. Based on this formulation, the stability analysis is transformed to be equivalent to pursuing the strictly positive realness of the transfer function for the formulated system. Using this proposed approach, a sufficient condition in terms of the range of the stiffness in this study, where the explicit algorithm is stable in the sense of Lyapunov, can be obtained using Nyquist plot. Specifically, the maximum and minimum stiffness values for stable (in the sense of Lyapunov) stiffening and softening systems, respectively, are determined. Finally, the stability analysis of the explicit Newmark algorithm, as a commonly used method in structural dynamics, using the proposed approach is applied to SDOF nonlinear stiffening or softening systems. Besides the stability, the accuracy of the integration methods is examined using a geometrically nonlinear problem, which has a closed-form exact solution.

Mathematical Preliminaries

In this section, definitions, notations and the strictly positive real lemma are introduced. Here, $(\bullet)^T$ and $(\bullet)^*$ denote transpose and complex conjugate transpose, respectively. Denote

$$\mathbf{G}(z) \sim \left[\begin{array}{c|c} \overline{\mathbf{A}} & \overline{\mathbf{B}} \\ \hline \overline{\mathbf{C}} & \overline{\mathbf{D}} \end{array}\right] \tag{1}$$

as a state-space realization (Cains 1989) of a transfer function matrix $\mathbf{G}(z)$ expressed as follows:

$$\mathbf{G}(z) = \overline{\mathbf{D}} + \overline{\mathbf{C}}\left(\mathbf{I}z - \overline{\mathbf{A}}\right)^{-1}\overline{\mathbf{B}} \tag{2}$$

where $z = e^{j\theta}$ is a complex variable with $j = \sqrt{-1}$ and $\theta \in [0,\, 2\pi]$, $\overline{\mathbf{A}}$, $\overline{\mathbf{B}}$, $\overline{\mathbf{C}}$ and $\overline{\mathbf{D}}$ are real constant matrices and $\mathbf{I}$ is the identity matrix with proper dimensions.

A square transfer function matrix $\mathbf{G}(z)$ is called strictly positive real (Kapila and Haddad 1996) if: (i) $\mathbf{G}(z)$ is asymptotically stable, which is stronger than Lyapunov stability as it guarantees convergence to a specific value as "time" approaches infinity, and (ii) $\mathbf{G}(e^{j\theta}) + \mathbf{G}^*(e^{j\theta})$ is positive definite $\forall\, \theta \in [0,\, 2\pi]$. Condition (i) is equivalent to the condition that the spectral radius of $\overline{\mathbf{A}}$ must be less than 1, i.e. $\rho(\overline{\mathbf{A}}) < 1$. The *discrete-time strictly positive real lemma* (Cains 1989) is stated as follows:

$$\mathbf{G}(z) \overset{\min}{\sim} \left[\begin{array}{c|c} \overline{\mathbf{A}} & \overline{\mathbf{B}} \\ \hline \overline{\mathbf{C}} & \overline{\mathbf{D}} \end{array}\right] \tag{3}$$

The notation $\overset{\min}{\sim}$ denotes a minimal realization. The minimally realized transfer function matrix is strictly positive real if and only if there exist matrices $\overline{\mathbf{M}}$, $\overline{\mathbf{L}}$ and $\overline{\mathbf{W}}$ with $\overline{\mathbf{M}}$ positive definite such that the following conditions are satisfied (Cains 1989):

$$\overline{\mathbf{M}} = \overline{\mathbf{A}}^T \overline{\mathbf{M}}\, \overline{\mathbf{A}} + \overline{\mathbf{L}}^T \overline{\mathbf{L}} \tag{4a}$$

$$\mathbf{0} = \overline{\mathbf{B}}^T \overline{\mathbf{M}}\, \overline{\mathbf{A}} - \overline{\mathbf{C}} + \overline{\mathbf{W}}^T \overline{\mathbf{L}} \tag{4b}$$

$$\mathbf{0} = \overline{\mathbf{D}} + \overline{\mathbf{D}}^T - \overline{\mathbf{B}}^T \overline{\mathbf{M}}\, \overline{\mathbf{B}} - \overline{\mathbf{W}}^T \overline{\mathbf{W}} \tag{4c}$$

The minimal realization is guaranteed by the observability, i.e. how well internal states of a system can be inferred by knowledge of its external outputs, of $\left(\overline{\mathbf{A}}, \overline{\mathbf{C}}\right)$ and the controllability, e.g. stabilization of unstable systems by feedback, of $\left(\overline{\mathbf{A}}, \overline{\mathbf{B}}\right)$. In that regard the row ranks of the following matrices should be n.

$$\mathbf{Z}_1 = \begin{bmatrix} \overline{\mathbf{C}} \\ \overline{\mathbf{C}}\,\overline{\mathbf{A}} \\ \overline{\mathbf{C}}\,\overline{\mathbf{A}}^2 \\ \vdots \\ \overline{\mathbf{C}}\,\overline{\mathbf{A}}^{n-1} \end{bmatrix} \tag{5a}$$

$$\mathbf{Z}_2 = [\overline{\mathbf{B}} \quad \overline{\mathbf{A}}\overline{\mathbf{B}} \quad \overline{\mathbf{A}}^2\overline{\mathbf{B}} \quad \cdots \quad \overline{\mathbf{A}}^{n-1}\overline{\mathbf{B}}] \tag{5b}$$

Accordingly, $\text{rank}\left(\mathbf{Z}_1\right) = \text{rank}\left(\mathbf{Z}_2\right) = n$, where n is the dimension of the square matrix $\overline{\mathbf{A}}$.

Integration Algorithms

The discretized equation of motion of a single degree of freedom (SDOF) system under an external dynamic force excitation is expressed as follows:

$$m\ddot{u}_{i+1} + c\dot{u}_{i+1} + f_{i+1} = p_{i+1} \tag{6}$$

where m and c are the mass and viscous damping, and $\ddot{u}_{i+1}$, $\dot{u}_{i+1}$, f_{i+1}, and p_{i+1} are the acceleration, velocity, restoring force, and external force at the time step $i+1$, respectively. The restoring force, $f(u)$, is generally defined as a function of displacement, u.

Usually, single-step direct integration algorithms are defined by the following difference equations:

$$u_{i+1} = u_i + \eta_0(\Delta t)\dot{u}_i + \eta_1(\Delta t)^2\ddot{u}_i + \eta_2(\Delta t)^2\ddot{u}_{i+1} + \eta_3 \tag{7}$$

$$\dot{u}_{i+1} = \eta_4\dot{u}_i + \eta_5(\Delta t)\ddot{u}_i + \eta_6(\Delta t)\ddot{u}_{i+1} + \eta_7 \tag{8}$$

In general, Eqs. (6)-(8) require an iterative solution, which forms the basis of implicit algorithms. On the other hand, these algorithms become explicit when $\eta_2 = 0$. Coefficients of the Newmark integration family (Newmark 1959) and the explicit OS algorithms (Hughes et al. 1979) are summarized in Table 1 where $\eta = m/\left(\beta(\Delta t)^2\right) + c\gamma/(\beta(\Delta t)) + k_{i+1}$ and k_{i+1} is defined below.

Table 1. Coefficients for the Newmark and the OS Integration Algorithms

	Newmark	OS		Newmark	OS
η_0	1	$1-c/(\eta(\Delta t))$	η_4	1	$1-c\gamma/(\beta\eta(\Delta t))$
η_1	$(1-2\beta)/2$	$(1-2\beta)/2-c(1-\gamma)/(\eta(\Delta t))$	η_5	$1-\gamma$	$1-\gamma-c\gamma(1-\gamma)/(\beta\eta(\Delta t))$
η_2	β	0	η_6	γ	0
η_3	0	$(p_{i+1}-\tilde{f}_{i+1})/\eta$	η_7	0	$\gamma(p_{i+1}-\tilde{f}_{i+1})/(\beta\eta(\Delta t))$

For the Newmark family of integration algorithms, the parameters β and γ define the variation of accelerations over a time step, Δt. The values of these parameters considered in this report are either $(\beta,\gamma)=(1/4,1/2)$, which represent constant average acceleration over Δt, or $(\beta,\gamma)=(0,1/2)$, which transform the integration to the explicit Newmark algorithm. On the other hand, for the OS algorithms, $\tilde{f}_{i+1}$ is the restoring force corresponding to the predicted displacement $\tilde{u}_{i+1}$ that is represented in an explicit form as follows:

$$\tilde{u}_{i+1}=u_i+(\Delta t)\dot{u}_i+\frac{(\Delta t)^2}{2}[(1-2\beta)\ddot{u}_i] \tag{9}$$

Two versions of the OS algorithms are considered: (1) $k_{i+1}=k_I$ for the $OS_{initial}$ algorithm with initial stiffness k_I; (2) $k_{i+1}=\tilde{k}_{T_{i+1}}$ for the $OS_{tangent}$ algorithm with tangent stiffness $\tilde{k}_{T_{i+1}}$ evaluated at $\tilde{u}_{i+1}$ determined from Eq. (9). It is to be noted that the $OS_{initial}$ algorithm is commonly used for hybrid simulations (Nakashima et al. 1990; Combescure and Pegon 1997) because of the difficulties in obtaining the tangent stiffness matrices. On the other hand, analytical simulations do not have such a limitation that prevents the use of the $OS_{tangent}$ algorithm. However, $OS_{tangent}$ algorithm was not previously investigated in the literature since the OS algorithms were rarely used in pure analytical simulations. Liang et al. (2014) investigated the suitability of the $OS_{tangent}$ algorithm in nonlinear dynamic analyses of multi-degree of freedom (MDOF) bridge systems.

In addition to the previously discussed implicit and explicit Newmark algorithms and the $OS_{initial}$ and $OS_{tangent}$ algorithms, this study also considers the TRBDF2 algorithm (Bank et al. 1985; Bathe and Baig 2005; Bathe 2007) as an example of a composite multi-step integration algorithm. The TRBDF2 algorithm interchanges the use of the implicit Newmark algorithm with constant average acceleration and the use of the three point Euler backward scheme presented in Eqs. (10) and (11) in consecutive integration time steps.

$$\dot{u}_{i+1}=\frac{1}{2}(u_{i+1}-4u_i+3u_{i-1})/\Delta t \tag{10}$$

$$\ddot{u}_{i+1}=\frac{1}{2}(\dot{u}_{i+1}-4\dot{u}_i+3\dot{u}_{i-1})/\Delta t \tag{11}$$

In Bathe and Baig (2005) and Bathe (2007), every two consecutive steps in the above formulation are considered as two sub-steps of one step, i.e. integration time steps $i-1$ and i become integration time steps i and, $i+0.5$. Therefore, the integration time step in Eqs. (7), (8), (10) and (11) is $\Delta t/2$ instead of Δt. Bathe and Baig (2005) and Bathe (2007) used the TRBDF2 algorithm in structural dynamics to conserve energy and momentum at large deformations (not necessarily involving material nonlinearity) where the implicit Newmark algorithm may fail to do so and become unstable. Herein, it is considered because of its superior stability due to numerical damping introduced by the Euler backward scheme.

Lyapunov Stability Analysis

For each direct integration algorithm, the relationship between the kinematic quantities at time steps $i+1$ and i can be established as follows:

$$\boldsymbol{x}_{i+1} = \mathbf{A}_i \boldsymbol{x}_i + \mathbf{L}_i \tag{12}$$

where $\boldsymbol{x}_i = \left[(\Delta t)^2 \ddot{u}_i \;\; (\Delta t)\dot{u}_i \;\; u_i\right]^T$, $\mathbf{A}_i$ and $\mathbf{L}_i$ are the approximation operator and the loading vector at the time step i, respectively. The loading vector, $\mathbf{L}$, is generally independent of the vector of kinematic quantities, $\boldsymbol{x}$. Eq. (12) can be further extended as follows:

$$\boldsymbol{x}_{i+1} = \left(\prod_{j=i}^{1} \mathbf{A}_j\right)\boldsymbol{x}_1 + \sum_{l=1}^{i-1}\left(\left(\prod_{k=i}^{l+1}\mathbf{A}_k\right)\mathbf{L}_l\right) + \mathbf{L}_i \tag{13}$$

where $\prod_{j=i}^{1}\mathbf{A}_j = \mathbf{A}_i\mathbf{A}_{i-1}\cdots\mathbf{A}_2\mathbf{A}_1$. In order to investigate the stability of the system in Eq. (12), a Lyapunov artificial energy function candidate v_{i+1} (Franklin et al. 2015) at the time step $i+1$, can be chosen as follows:

$$v_{i+1} = \boldsymbol{x}_{i+1}^T \mathbf{M} \boldsymbol{x}_{i+1} \tag{14}$$

where $\mathbf{M}$ is positive definite, i.e. $\mathbf{M} = \mathbf{M}^T \succ \mathbf{0}$ and $\mathbf{0}$ is the null matrix of the same dimension as $\mathbf{M}$. The system in Eq. (12) is stable if the Lyapunov function in Eq. (14) is bounded for $i \to \infty$. Substituting Eq. (13) into Eq. (14) with some manipulations leads to the following:

$$\begin{aligned} v_{i+1} = {} & \boldsymbol{x}_1^T\left(\prod_{k=i}^{1}\mathbf{A}_k^T\right)\mathbf{M}\left(\prod_{j=i}^{1}\mathbf{A}_j\right)\boldsymbol{x}_1 + \sum_{l=1}^{i-1}\sum_{m=1}^{i-1}\left(\mathbf{L}_l^T\left(\prod_{k=i}^{l+1}\mathbf{A}_k^T\right)\mathbf{M}\left(\prod_{n=i}^{m+1}\mathbf{A}_n\right)\mathbf{L}_m\right) + \mathbf{L}_i^T\mathbf{M}\mathbf{L}_i + \\ & 2\boldsymbol{x}_1^T\left(\prod_{k=i}^{1}\mathbf{A}_k^T\right)\mathbf{M}\mathbf{L}_i + 2\left(\sum_{l=1}^{i-1}\mathbf{L}_l^T\left(\prod_{k=i}^{l+1}\mathbf{A}_k^T\right)\right)\mathbf{M}\mathbf{L}_i + \\ & 2\boldsymbol{x}_1^T\left(\prod_{k=i}^{1}\mathbf{A}_k^T\right)\mathbf{M}\left(\sum_{m=1}^{i-1}\left(\prod_{n=i}^{m+1}\mathbf{A}_n\right)\mathbf{L}_m\right) \end{aligned} \tag{15}$$

The loading vector, $\mathbf{L}$, is generally a function of external force, p. Therefore, it is bounded. Therefore, based on Eq. (15), the boundedness of the Lyapunov function v_{i+1} for $i \to \infty$ leads to the boundedness of $\prod_{j=i}^{1} \mathbf{A}_j$ for $i \to \infty$. For linear behavior of structures, the approximation operator, $\mathbf{A}$, remains constant, thus $\prod_{j=i}^{1} \mathbf{A}_j$ becomes $\mathbf{A}^i$ that can be decomposed as follows:

$$\mathbf{A}^i = \mathbf{V}\mathbf{D}^i\mathbf{V}^{-1} \tag{16}$$

where $\mathbf{D}$ and $\mathbf{V}$ are matrices of eigenvalues and eigenvectors of $\mathbf{A}$, respectively. The boundedness of $\mathbf{A}^i$ for $i \to \infty$ leads to the well-known stability criterion for linear systems, namely the spectral radius of the approximation operator $\rho(\mathbf{A})$ must be less than or equal to 1.0.

For nonlinear structures, $\prod_{j=i}^{1} \mathbf{A}_j$ is more involved due to the continuous variation of the approximation operator $\mathbf{A}_i$. Therefore, the stability of a nonlinear system cannot be solely determined using the spectral radius of its approximation operator $\mathbf{A}_i$. However, the investigation of the eigen properties of $\mathbf{A}_i$ is still necessary in nonlinear problems. For small values of Δt, the increment of restoring force can be approximated (Chopra 2006) as follows:

$$f_{i+1}^* - f_i = k_{T_{i+1}}^* \left(u_{i+1}^* - u_i\right) \tag{17}$$

where $\left(f_{i+1}^*, k_{T_{i+1}}^*, u_{i+1}^*\right) = \left(f_{i+1}, k_{T_{i+1}}, u_{i+1}\right)$ for the Newmark family of algorithms and $\left(f_{i+1}^*, k_{T_{i+1}}^*, u_{i+1}^*\right) = \left(\tilde{f}_{i+1}, \tilde{k}_{T_{i+1}}, \tilde{u}_{i+1}\right)$ for the OS algorithms. It is to be noted that $k_{T_{i+1}}$ is the tangent stiffness at the time step $i+1$, and other parameters are defined before. The tangent stiffness is generally a function of the displacement, thus Eq. (12) represents a nonlinear system of equations. With the approximation in Eq. (17), the approximation operator $\mathbf{A}_i$ for the Newmark and the OS algorithms is derived as follows:

$$\mathbf{A}_i = \begin{bmatrix} \dfrac{1 - \frac{1-2\beta}{2}\omega_0^2(\Delta t)^2 - 2\zeta\omega_n(1-\gamma)(\Delta t)}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} & -\dfrac{\omega_0^2(\Delta t)^2}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} & 0 \\ \dfrac{\gamma\left(1 - \frac{1-2\beta}{2}\omega_0^2(\Delta t)^2 - 2\zeta\omega_n(1-\gamma)(\Delta t)\right)}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} + (1-\gamma) & 1 - \dfrac{\gamma\omega_0^2(\Delta t)^2}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} & 0 \\ \dfrac{\beta\left(1 - \frac{1-2\beta}{2}\omega_0^2(\Delta t)^2 - 2\zeta\omega_n(1-\gamma)(\Delta t)\right)}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} + \dfrac{1-2\beta}{2} & 1 - \dfrac{\beta\omega_0^2(\Delta t)^2}{1 + 2\zeta\omega_n\gamma(\Delta t) + \beta\omega_1^2(\Delta t)^2} & 1 \end{bmatrix} \tag{18}$$

where $\zeta = c/(2m\omega_n)$, $\omega_n^2 = k_I/m$. Coefficients ω_0 and ω_1 for the Newmark integration family and the OS algorithms are listed in Table 2.

Table 2. Coefficients of approximation operators for the Newmark and the OS Integration Algorithms

	Newmark	$OS_{initial}$	$OS_{tangent}$
ω_0	$\omega_{T_{i+1}}$	$\tilde{\omega}_{T_{i+1}}$	$\tilde{\omega}_{T_{i+1}}$
ω_1	$\omega_{T_{i+1}}$	ω_n	$\tilde{\omega}_{T_{i+1}}$

where $\omega_{T_{i+1}}^2 = k_{T_{i+1}}/m, \tilde{\omega}_{T_{i+1}}^2 = \tilde{k}_{T_{i+1}}/m$. It is obvious that one of eigenvalues of $\mathbf{A}_i$ in Eq. (18) is 1. For the Newmark and OS algorithms with $(\beta,\gamma)=(1/4,1/2)$, the other two eigenvalues are obtained as follows:

$$\lambda_{1,2} = \frac{1-\frac{3}{8}\omega_0^2(\Delta t)^2+\frac{1}{8}\omega_1^2(\Delta t)^2 \pm \Delta t\sqrt{\omega_n^2\zeta^2+\frac{1}{4}\omega_n\zeta\Delta t\left(\omega_1^2-\omega_0^2\right)+(\Delta t)^2\left(9\omega_0^4-10\omega_0^2\omega_1^2+\omega_1^4\right)-\omega_0^2}}{1+\omega_n\zeta\Delta t+\frac{1}{4}\omega_1^2(\Delta t)^2} \tag{19}$$

On the other hand, for the explicit Newmark algorithm, i.e. $(\beta,\gamma)=(0,1/2)$,

$$\lambda_{1,2} = \frac{1-\frac{1}{2}\omega_{T_{i+1}}^2(\Delta t)^2 \pm \Delta t\sqrt{\omega_n^2\zeta^2+\frac{1}{4}\omega_{T_{i+1}}^4(\Delta t)^2-\omega_{T_{i+1}}^2}}{1+\omega_n\zeta\Delta t} \tag{20}$$

The conditions for $\rho(\mathbf{A}_i)\le 1$ are summarized in Table 3 for the case of zero viscous damping ($\zeta=0$), which is the most critical case for the stability analysis of direct integration algorithms. In Table 3, $T_{T_{i+1}} = 2\pi/\omega_{T_{i+1}} = 2\pi\sqrt{m/k_{T_{i+1}}}$ and $(\beta,\gamma)=(1/4,1/2)$ are used for implicit Newmark, $OS_{initial}$ and $OS_{tangent}$ and thereafter in this report.

Table 3. Conditions for $\rho(\mathbf{A}_i)\le 1$

Integration Algorithms	Limits
Implicit Newmark	$k_{T_{i+1}} \ge 0$
Explicit Newmark	$\Delta t/T_{T_{i+1}} \le 1/\pi$
$OS_{initial}$	$0 \le \tilde{k}_{T_{i+1}} \le k_I$
$OS_{tangent}$	$\tilde{k}_{T_{i+1}} \ge 0$

The conditions in Table 3 are not stability criteria of the listed direct integration algorithms used in nonlinear systems. Some past studies, however, determined the stability of direct integration algorithms based solely on the spectral radius. Combescure and Pegon (1997) claimed that the $OS_{initial}$ algorithm is unconditionally stable as long as the tangent stiffness is

smaller than or equal to the initial stiffness, i.e. they directly applied the stability criterion for linear structures to nonlinear ones. Chen and Ricles (2008) demonstrated that the Newmark method with constant average acceleration and explicit Newmak method have the stability limits listed in Table 3. They investigated such stability using discrete control theory. However, the method presented in their paper is only applicable to linear time-invariant systems (Franklin et al. 2015). This is equivalent to investigating the conditions of the spectral radius for the approximation operator $\mathbf{A}_i$ and that is the reason why the results they obtained are the same as those expressed by Eqs. (19) and (20). Accordingly, these published arguments of stability, i.e. those by Combescure and Pegon (1997) and Chen and Ricles (2008), are incomplete as they are not, in general, applicable to nonlinear problems. Another noteworthy observation is that the approximation operator of the explicit $\text{OS}_{\text{tangent}}$ algorithm is the same as that of the implicit Newmark algorithm with $\omega_{T_{i+1}}$ replaced by $\widetilde{\omega}_{T_{i+1}}$, refer to Table 2. This indicates that they possess similar stability properties, as indicated in Table 3.

Different from the integration algorithms above, the TRBDF2 is a multi-step algorithm with numerical damping introduced by the Euler backward scheme. Its approximation operator in Eq. (22) is obtained for the case of zero viscous damping ($\zeta = 0$) by similar linearization approximation for the tangent stiffness as before and given as follows:

$$k_{T_{i+1}} = \left(f_{i+1} - f_{i+0.5}\right)/\left(u_{i+1} - u_{i+0.5}\right) = \left(f_{i+0.5} - f_i\right)/\left(u_{i+0.5} - u_i\right) \tag{21}$$

$$\mathbf{A}_i = \frac{1}{B}\begin{bmatrix} 9 - \frac{47}{16}\omega_{T_{i+1}}^2(\Delta t)^2 & \frac{5}{16}\omega_{T_{i+1}}^4(\Delta t)^4 - 9\omega_{T_{i+1}}^2(\Delta t)^2 & 0 \\ 9 - \frac{5}{16}\omega_{T_{i+1}}^2(\Delta t)^2 & 9 - \frac{47}{16}\omega_{T_{i+1}}^2(\Delta t)^2 & 0 \\ \frac{9}{2} + \frac{1}{16}\omega_{T_{i+1}}^2(\Delta t)^2 & 9 - \frac{5}{16}\omega_{T_{i+1}}^2(\Delta t)^2 & \left(\omega_{T_{i+1}}^2(\Delta t)^2 + 9\right)\left(\frac{1}{16}\omega_{T_{i+1}}^2(\Delta t)^2 + 1\right) \end{bmatrix} \tag{22}$$

where $B = \left(\omega_{T_{i+1}}^2(\Delta t)^2 + 9\right)\left(\frac{1}{16}\omega_{T_{i+1}}^2(\Delta t)^2 + 1\right)$. Thus, besides the one obvious eigenvalue of 1, the other two are as follows:

$$\lambda_{1.2} = \frac{9 - \frac{47}{16}\omega_{T_{i+1}}^2(\Delta t)^2 \pm \left(9 - \frac{5}{16}\omega_{T_{i+1}}^2(\Delta t)^2\right)\sqrt{-\omega_{T_{i+1}}^2(\Delta t)^2}}{\frac{1}{16}\omega_{T_{i+1}}^4(\Delta t)^4 + \frac{25}{16}\omega_{T_{i+1}}^2(\Delta t)^2 + 9} \tag{23}$$

It can be shown that for $k_{T_{i+1}} \geq 0$, magnitudes of the eigenvalues in Eq. (23) are always less than 1 because of introduced numerical damping. It is to be noted that for the case of zero viscous damping ($\zeta = 0$), eigenvalues in Eqs. (19), (20) and (23) are imaginary, which implies vibration.

As previously discussed, a system is stable if its $\prod_{j=i}^{1}\mathbf{A}_j$ is bounded for $i \to \infty$. This is equivalent to investigating the system in Eq. (12) with the loading vector $\mathbf{L} = \mathbf{0}$, i.e.

$$\boldsymbol{x}_{i+1} = \mathbf{A}_i \boldsymbol{x}_i \tag{24}$$

$\mathbf{A}_i$ can be rewritten non-dimensionally, e.g. in the implicit Newmark algorithm:

$$\mathbf{A}_i = \begin{bmatrix} \dfrac{1-2\pi\zeta\mu-\pi^2\delta_{i+1}^2\mu^2}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & -\dfrac{\omega_0^2(\Delta t)^2}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & 0 \\ \dfrac{1}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & \dfrac{1+2\pi\zeta\mu-\pi^2\delta_{i+1}^2\mu^2}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & 0 \\ \dfrac{\frac{1}{2}}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & \dfrac{1+2\pi\zeta\mu}{1+2\pi\zeta\mu+\pi^2\delta_{i+1}^2\mu^2} & 1 \end{bmatrix} \tag{25}$$

where $\delta_{i+1} = \omega_{T_{i+1}}/\omega_n, \mu = \Delta t/T_n, T_n = 2\pi/\omega_n = 2\pi\sqrt{m/k_I}$. Therefore, $\mathbf{A}_i$ is a function of δ_{i+1}. Similar to Eq. (14), the Lyapunov function candidate v_{i+1} at the time step $i+1$ can be selected as follows:

$$v_{i+1} = \boldsymbol{x}_{i+1}^T \mathbf{M}_{i+1} \boldsymbol{x}_{i+1} \tag{26}$$

where the positive definite matrix $\mathbf{M}_{i+1} = \mathbf{M}_{i+1}^T$ is a function of δ_{i+1}. A sufficient condition for the system and thus the direct integration algorithm to be stable is as follows:

$$\begin{aligned} \Delta v_{i+1} &= v_{i+1} - r_t v_i \\ &= \boldsymbol{x}_{i+1}^T \mathbf{M}_{i+1} \boldsymbol{x}_{i+1} - r_t \boldsymbol{x}_i^T \mathbf{M}_i \boldsymbol{x}_i \\ &= \boldsymbol{x}_i^T \left(\mathbf{A}_i^T \mathbf{M}_{i+1} \mathbf{A}_i - r_t \mathbf{M}_i \right) \boldsymbol{x}_i \\ &= \boldsymbol{x}_i^T \mathbf{P}_{i+1} \boldsymbol{x}_i \le 0 \end{aligned} \tag{27}$$

where $r_t \in (0,1]$ controls the rate of convergence, i.e. the smaller the r_t , the faster the convergence. Eq. (27) leads to the negative semi-definiteness of $\mathbf{P}_{i+1}$, i.e. $\mathbf{P}_{i+1} \prec= \mathbf{0}$.

Numerical Stability Analysis

In this section, a numerical approach is presented to enable investigating the stability discussed in the previous section. This approach is based on transforming the stability analysis to a problem of convex optimization, which is applicable to direct integration algorithms applied to nonlinear problems. For a direct integration algorithm, $\mathbf{M}_{i+1}$ can be expressed as:

$$\mathbf{M}_{i+1} = \sum_{j=1}^{N} \alpha_j \mathbf{\Phi}(\delta_{i+1})_j \tag{28}$$

where α_j and $\mathbf{\Phi}(\delta_{i+1})_j$ are the *j*–th constant coefficient and base function, respectively, and N is the total number of base functions. One example set of base functions is given in the Appendix.

With the range of δ_i and δ_{i+1} given, e.g. $\delta_i, \delta_{i+1} \in [a,b]$, points can be sampled within this range, e.g. sampling $n+1$ points in $[a,b]$ with interval $\Delta\delta = (b-a)/n$. This yields $(n+1)^2$ possible pairs of (δ_i, δ_{i+1}). Accordingly, the stability analysis becomes a problem of convex optimization that seeks the determination of the coefficients α_j by minimizing their norm for the selected base functions $\mathbf{\Phi}(\delta_{i+1})_j$ where $j:1 \to N$, subjected to the following conditions on the $(n+1)^2$ possible pairs of (δ_i, δ_{i+1}):

$$\begin{aligned} &\delta_i, \delta_{i+1} \in [a,b], \quad \Delta\delta = (b-a)/n \\ &\mathbf{A}_i^T \mathbf{M}_{i+1} \mathbf{A}_i - r_t \mathbf{M}_i = \mathbf{A}_i(\delta_{i+1})^T \left(\sum_{j=1}^{N} \alpha_j \mathbf{\Phi}(\delta_{i+1})_j \right) \mathbf{A}_i(\delta_{i+1}) - r_t \sum_{j=1}^{N} \alpha_j \mathbf{\Phi}(\delta_i)_j \preceq \mathbf{0} \\ &\mathbf{M}_i = \sum_{j=1}^{N} \alpha_j \mathbf{\Phi}(\delta_i)_j \succ \mathbf{0}, \quad \mathbf{M}_{i+1} = \sum_{j=1}^{N} \alpha_j \mathbf{\Phi}(\delta_{i+1})_j \succ \mathbf{0} \end{aligned} \tag{29}$$

Optionally, with knowledge about the variation of δ_{i+1}, the range of $|\delta_{i+1} - \delta_i|$ can be specified, e.g. $|\delta_{i+1} - \delta_i| < \varepsilon$, where ε is a constant that is not necessarily small. This leads to fewer possible pairs of (δ_i, δ_{i+1}) that require less computational effort.

The problem of convex optimization can be solved numerically by CVX, a package for specifying and solving convex programs (CVX Research Inc. 2011). An example for the implicit Newmark algorithm is conducted based on the following conditions:

$$\begin{aligned} &\zeta = 0.05 \quad \mu = 0.05/(2\pi) \\ &a = 0.9 \quad b = 1.0 \quad n = 20 \\ &\Delta\delta = 0.005 \quad \varepsilon = 0.05 \quad r_t = 1.0 \end{aligned} \tag{30}$$

Same set of base functions as in the Appendix is used. The coefficients α_j, $j:1 \to 12$, obtained by minimizing the 2-norm of α, i.e. $\min\left(\sum_{j=1}^{12} |\alpha_j|^2 \right)^{1/2}$, are as follows:

$$\begin{aligned} &\alpha_1 = 1.90 \times 10^{-8}, \quad \alpha_2 = 2.46 \times 10^{-9}, \quad \alpha_3 = 1.70 \times 10^{-10}, \quad \alpha_4 = -2.25 \times 10^{-9}, \\ &\alpha_5 = -2.70 \times 10^{-10}, \quad \alpha_6 = -4.60 \times 10^{-10}, \quad \alpha_7 = 1.76 \times 10^{-8}, \quad \alpha_8 = 1.05 \times 10^{-9}, \\ &\alpha_9 = 6.00 \times 10^{-11}, \quad \alpha_{10} = -3.35 \times 10^{-9}, \quad \alpha_{11} = 4.30 \times 10^{-10}, \quad \alpha_{12} = -2.00 \times 10^{-10} \end{aligned} \tag{31}$$

The set of α_j in Eq. (31) from many determined sets has the minimum 2-norm explaining the listed small values of α_j. The existence of such set of α_j implies that the implicit Newmark algorithm is stable for the conditions in Eq. (30). Accordingly, if more points are sampled, e.g. 41 points for $\Delta\delta = 0.0025$, then more computational effort is required accompanied with more accurate stability analysis.

The proposed approach in this section can be applied to investigating the stability of other direct integration algorithms, including the other four methods considered in this report. Also, various nonlinear problems, including stiffening ($\delta_{i+1} > 1$) and softening ($\delta_{i+1} < 1$) behaviors in Fig. 1, can be taken into account. Accordingly, the proposed approach is generally applicable to direct integration algorithms as long as they can be expressed as given by Eq. (24).

Explicit Integration Algorithms

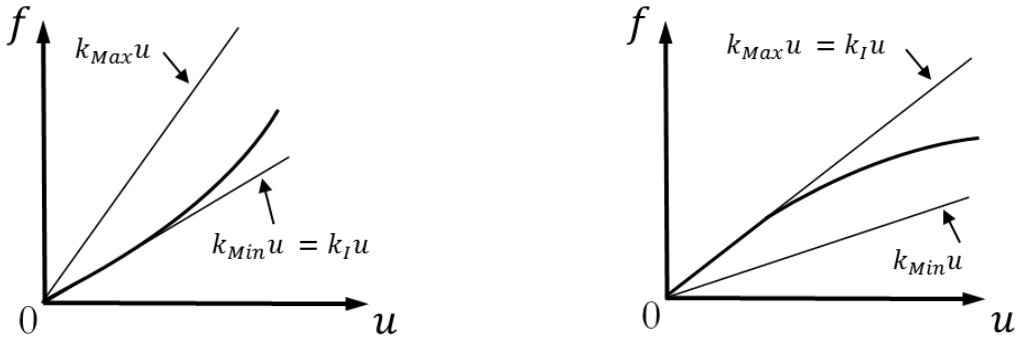
Another systematic approach is proposed in the subsequent sections to investigate the Lyapunov stability of explicit algorithms by means of the strictly positive real lemma. Single-step explicit direct integration algorithms considered in this approach are defined by the following difference equations:

$$u_{i+1} = u_i + \kappa_0 (\Delta t)\dot{u}_i + \kappa_1 (\Delta t)^2 \ddot{u}_i \tag{32a}$$

$$\dot{u}_{i+1} = \dot{u}_i + \kappa_2 (\Delta t)\ddot{u}_i + \kappa_3 (\Delta t)\ddot{u}_{i+1} \tag{32b}$$

For example, $\boldsymbol{\kappa} = [\kappa_0 \quad \kappa_1 \quad \kappa_2 \quad \kappa_3] = [1 \quad 1/2 \quad 1/2 \quad 1/2]$ leads to the explicit Newmark algorithm (Newmark 1959). The restoring force, $f(u)$, is here defined as a function of displacement, u, and restricted to the following range (to be determined in this approach according to the proposed Lyapunov stability analysis):

$$k_{Min} u^2 \le f(u) u \le k_{Max} u^2 \tag{33}$$



a) Stiffening system

b) Softening system

Fig. 2. Schematic illustrations of two structural systems with sector-bounded memoryless restoring forces.

Fig. 2 shows the schematic illustrations for stiffening (Fig. 2a) and softening (Fig. 2b) systems with memoryless restoring forces bounded in the sector between $k_{Min}u$ and $k_{Max}u$. As mentioned before, the maximum, k_{Max} and minimum, k_{Min}, stiffness values for stable (in the sense of Lyapunov) stiffening and softening systems, respectively, are to be determined.

Nonlinear Structural Systems

Stiffening Systems

For the explicit direct integration algorithm formulated by Eqs. (6) and (32), the relationship between the kinematic quantities at time steps $i+1$ and i can be established as follows:

$$\boldsymbol{x}_{i+1} = \mathbf{A}\boldsymbol{x}_i - \mathbf{B}_1 f_{i+1} + \mathbf{D}p_{i+1} \tag{34}$$

where $\boldsymbol{x}_i = \left[(\Delta t)^2 \ddot{u}_i \quad (\Delta t)\dot{u}_i \quad u_i\right]^T$, $\mathbf{A}$, $\mathbf{B}_1$ and $\mathbf{D}$ are as follows:

$$\mathbf{A} = \begin{bmatrix} -c\kappa_2/E & -c/E & 0 \\ \kappa_2 - c\kappa_2\kappa_3/E & 1 - c\kappa_3/E & 0 \\ \kappa_1 & \kappa_0 & 1 \end{bmatrix} \tag{35a}$$

$$\mathbf{B}_1 = \mathbf{D} = \left[(\Delta t)/E \quad (\Delta t)\kappa_3/E \quad 0\right]^T \tag{35b}$$

where $E = m/(\Delta t) + c\kappa_3$. The external force, p, is generally independent of the kinematic quantities, $\boldsymbol{x}$, and does not affect the Lyapunov stability of the direct integration algorithms as discussed before. Therefore, p_{i+1} is set to zero in Eq. (34) and thereafter in this report.

It is obvious that one of the eigenvalues of $\mathbf{A}$ is 1 that fails to satisfy the first condition (i) of the strictly positive realness of the transfer function. Therefore, Eq. (34) (after setting $p_{i+1} = 0$) is further manipulated as follows:

$$\begin{aligned} \boldsymbol{x}_{i+1} &= \mathbf{A}\boldsymbol{x}_i - k_{Min}\mathbf{B}_1\mathbf{C}\boldsymbol{x}_i + k_{Min}\mathbf{B}_1\mathbf{C}\boldsymbol{x}_i - \mathbf{B}_1 f_{i+1} \\ &= (\mathbf{A} - k_{Min}\mathbf{B}_1\mathbf{C})\boldsymbol{x}_i - \mathbf{B}_1(f_{i+1} - k_{Min}\mathbf{C}\boldsymbol{x}_i) \\ &= (\mathbf{A} - k_{Min}\mathbf{B}_1\mathbf{C})\boldsymbol{x}_i - \mathbf{B}_1(f_{i+1} - k_{Min}u_{i+1}) \\ &= \mathbf{A}_{e1}\boldsymbol{x}_i - \mathbf{B}_1 f_{e1} \end{aligned} \tag{36}$$

where

$$\mathbf{C} = \left[\kappa_1 \quad \kappa_0 \quad 1\right] \tag{37a}$$

$$u_{i+1} = \mathbf{C}\boldsymbol{x}_i \tag{37b}$$

$$\mathbf{A}_{e1} = \mathbf{A} - k_{Min}\mathbf{B}_1\mathbf{C} \tag{37c}$$

$$f_{e1} = f_{i+1} - k_{Min}u_{i+1} \tag{37d}$$

Softening Systems

For softening systems, similar to Eq. (34), with $\mathbf{B}_2 = -\mathbf{B}_1$ and also setting $p_{i+1} = 0$, as mentioned above:

$$\begin{aligned} \boldsymbol{x}_{i+1} &= \mathbf{A}\boldsymbol{x}_i + \mathbf{B}_2 f_{i+1} \\ &= \mathbf{A}\boldsymbol{x}_i + k_{Max}\mathbf{B}_2\mathbf{C}\boldsymbol{x}_i - k_{Max}\mathbf{B}_2\mathbf{C}\boldsymbol{x}_i + \mathbf{B}_2 f_{i+1} \\ &= \left(\mathbf{A} + k_{Max}\mathbf{B}_2\mathbf{C}\right)\boldsymbol{x}_i - \mathbf{B}_2\left(k_{Max}\mathbf{C}\boldsymbol{x}_i - f_{i+1}\right) \\ &= \left(\mathbf{A} + k_{Max}\mathbf{B}_2\mathbf{C}\right)\boldsymbol{x}_i - \mathbf{B}_2\left(k_{Max}u_{i+1} - f_{i+1}\right) \\ &= \mathbf{A}_{e2}\boldsymbol{x}_i - \mathbf{B}_2 f_{e2} \end{aligned} \tag{38}$$

where

$$\mathbf{A}_{e2} = \mathbf{A} + k_{Max}\mathbf{B}_2\mathbf{C} \tag{39a}$$

$$f_{e2} = k_{Max}u_{i+1} - f_{i+1} \tag{39b}$$

Accordingly, both stiffening and softening systems can be expressed in Eq. (40) with coefficients $\mathbf{A}_e, \mathbf{B}_e$ and f_e summarized in Table 4.

$$\boldsymbol{x}_{i+1} = \mathbf{A}_e\boldsymbol{x}_i - \mathbf{B}_e f_e \tag{40}$$

Table 4. Coefficients for stiffening and softening systems

	Stiffening Systems	Softening Systems
$\mathbf{A}_e$	$\mathbf{A}_{e1} = \mathbf{A} - k_{Min}\mathbf{B}_1\mathbf{C}$	$\mathbf{A}_{e2} = \mathbf{A} + k_{Max}\mathbf{B}_2\mathbf{C}$
$\mathbf{B}_e$	$\mathbf{B}_1$	$\mathbf{B}_2$
f_e	$f_{e1} = f_{i+1} - k_{Min}\mathbf{C}\boldsymbol{x}_i$	$f_{e2} = k_{Max}\mathbf{C}\boldsymbol{x}_i - f_{i+1}$

Lyapunov Stability Analysis of Explicit Algorithms

Based on Eqs. (33), (37d) and (39b), f_e, expressed as a function of u_{i+1}, has the following range

$$0 \le f_e\left(u_{i+1}\right)u_{i+1} \le k\,u_{i+1}^2 \tag{41}$$

where $k = k_{Max} - k_{Min}$. In order to investigate the stability of the system in Eq. (40), similar to Eqs. (14) and (26), a Lyapunov artificial energy function candidate v_{i+1} (Franklin et al. 2015) at the time step $i+1$ can be chosen as follows:

$$v_{i+1} = \boldsymbol{x}_{i+1}^T\mathbf{M}\boldsymbol{x}_{i+1} \tag{42}$$

where $\mathbf{M}$ is positive definite, i.e. $\mathbf{M}=\mathbf{M}^T \succ \mathbf{0}$ and $\mathbf{0}$ is the null matrix of the same dimensions as $\mathbf{M}$. A sufficient condition for the system, and thus the explicit direct integration algorithm, to be stable is as follows:

$$\begin{aligned}\Delta v_{i+1} &= v_{i+1}-v_i = \boldsymbol{x}_{i+1}^T\mathbf{M}\boldsymbol{x}_{i+1}-\boldsymbol{x}_i^T\mathbf{M}\boldsymbol{x}_i \\ &= \left(\mathbf{A}_e\boldsymbol{x}_i-\mathbf{B}_e f_e\right)^T\mathbf{M}\left(\mathbf{A}_e\boldsymbol{x}_i-\mathbf{B}_e f_e\right)-\boldsymbol{x}_i^T\mathbf{M}\boldsymbol{x} \\ &= \boldsymbol{x}_i^T\left(\mathbf{A}_e^T\mathbf{M}\mathbf{A}_e-\mathbf{M}\right)\boldsymbol{x}_i-2f_e^T\mathbf{B}_e^T\mathbf{M}\mathbf{A}_e\boldsymbol{x}_i+f_e^T\mathbf{B}_e^T\mathbf{M}\mathbf{B}_e f_e \le 0\end{aligned} \tag{43}$$

Note that $f_e^T = f_e$. Multiplying Eq. (41) by f_e/u_{i+1}, $|u_{i+1}|>0$, which is always positive, and rearranging, one obtains:

$$f_e\left(f_e-ku_{i+1}\right)= f_e\left(f_e-k\mathbf{C}\boldsymbol{x}_i\right)\le 0 \tag{44}$$

Eq. (43) becomes

$$\begin{aligned}\Delta v_{i+1}\le\Delta\bar{v}_{i+1} &= \boldsymbol{x}_i^T\left(\mathbf{A}_e^T\mathbf{M}\mathbf{A}_e-\mathbf{M}\right)\boldsymbol{x}_i-2f_e^T\mathbf{B}_e^T\mathbf{M}\mathbf{A}_e\boldsymbol{x}_i+f_e^T\mathbf{B}_e^T\mathbf{M}\mathbf{B}_e f_e-2f_e\left(f_e-k\mathbf{C}\boldsymbol{x}_i\right) \\ &= \boldsymbol{x}_i^T\left(\mathbf{A}_e^T\mathbf{M}\mathbf{A}_e-\mathbf{M}\right)\boldsymbol{x}_i+f_e^T\left(\mathbf{B}_e^T\mathbf{M}\mathbf{B}_e-2\right)f_e-2f_e^T\left(\mathbf{B}_e^T\mathbf{M}\mathbf{A}_e-k\mathbf{C}\right)\boldsymbol{x}_i\end{aligned} \tag{45}$$

where $\Delta\bar{v}_{i+1}$ can be further transformed as follows:

$$\begin{aligned}\Delta\bar{v}_{i+1} &= \Delta\bar{v}_{i+1}+\boldsymbol{x}_i^T\mathbf{L}^T\mathbf{L}\boldsymbol{x}_i-\boldsymbol{x}_i^T\mathbf{L}^T\mathbf{L}\boldsymbol{x}_i \\ &= \boldsymbol{x}_i^T\left(\mathbf{A}_e^T\mathbf{M}\mathbf{A}_e-\mathbf{M}+\mathbf{L}^T\mathbf{L}\right)\boldsymbol{x}_i- \\ &\quad \left[f_e^T\left(2-\mathbf{B}_e^T\mathbf{M}\mathbf{B}_e\right)f_e-2f_e^T\left(k\mathbf{C}-\mathbf{B}_e^T\mathbf{M}\mathbf{A}_e\right)\boldsymbol{x}_i+\boldsymbol{x}_i^T\mathbf{L}^T\mathbf{L}\boldsymbol{x}_i\right]\end{aligned} \tag{46}$$

where $\mathbf{L}$ is a 3×3 matrix. A sufficient condition for $\Delta\bar{v}_{i+1}\le 0$ and thus $\Delta v_{i+1}\le 0$ is as follows:

$$\mathbf{M}=\mathbf{A}_e^T\mathbf{M}\mathbf{A}_e+\mathbf{L}^T\mathbf{L} \tag{47a}$$

$$\mathbf{0}=\mathbf{B}_e^T\mathbf{M}\mathbf{A}_e-k\mathbf{C}+\mathbf{W}^T\mathbf{L} \tag{47b}$$

$$0=1+1^T-\mathbf{B}_e^T\mathbf{M}\mathbf{B}_e-\mathbf{W}^T\mathbf{W} \tag{47c}$$

where $\mathbf{W}$ is a 3×1 vector. With Eqs. (47), Eq. (45) becomes

$$\begin{aligned}\Delta v_{i+1}\le\Delta\bar{v}_{i+1} &= \boldsymbol{x}_i^T\mathbf{0}\boldsymbol{x}_i-\left(f_e^T\mathbf{W}^T\mathbf{W}f_e-2f_e^T\mathbf{W}^T\mathbf{L}\boldsymbol{x}_i+\boldsymbol{x}_i^T\mathbf{L}^T\mathbf{L}\boldsymbol{x}_i\right) \\ &= -\left(\mathbf{W}f_e-\mathbf{L}\boldsymbol{x}_i\right)^T\left(\mathbf{W}f_e-\mathbf{L}\boldsymbol{x}_i\right)\le 0\end{aligned} \tag{48}$$

Therefore, the Lyapunov stability of the explicit integration algorithm depends solely on the existence of $\mathbf{M}$, $\mathbf{L}$ and $\mathbf{W}$ such that Eqs. (47) are satisfied. Recall the strictly positive real lemma presented before, i.e. Eqs. (4), the comparison between Eqs. (4) and (47) gives

$$\overline{\mathbf{A}} = \mathbf{A}_e, \quad \overline{\mathbf{B}} = \mathbf{B}_e, \quad \overline{\mathbf{C}} = k\mathbf{C}, \quad \overline{\mathbf{D}} = 1, \quad \overline{\mathbf{M}} = \mathbf{M}, \quad \overline{\mathbf{L}} = \mathbf{L}, \quad \overline{\mathbf{W}} = \mathbf{W} \tag{49}$$

Accordingly, the stability analysis reduces to seeking k such that the transfer function $G(z)$ in Eq. (50) is strictly positive real.

$$G(z) = 1 + k\mathbf{C}(\mathbf{I}z - \mathbf{A}_e)^{-1}\mathbf{B}_e \tag{50}$$

To be strictly positive real, the first condition (i) requires that $\mathbf{A}_e$ must be asymptotically stable. It is noted that $\mathbf{A}_e$ can be expressed non-dimensionally for the SDOF governed by Eqs. (6), (32a) and (32b) as follows:

$$\mathbf{A}_e = \begin{bmatrix} -[\kappa_2 + (\pi\mu/\zeta)\kappa_1]/F & -[1 + (\pi\mu/\zeta)\kappa_0]/F & -(\pi\mu/\zeta)/F \\ \kappa_2 - [\kappa_2 + (\pi\mu/\zeta)\kappa_1]\kappa_3/F & 1 - [1 + (\pi\mu/\zeta)\kappa_0]\eta_3/F & -(\pi\mu/\zeta)\kappa_3/F \\ \kappa_1 & \kappa_0 & 1 \end{bmatrix} \tag{51}$$

where

$$F = 1/(4\pi\mu\zeta) + \kappa_3 \tag{52}$$

The second condition (ii) for this case where the dimension of $G(z)$ in Eq. (50) is 1 becomes equivalent to the following:

$$\mathrm{Re}[G(z)] > 0 \tag{53}$$

which leads to

$$\mathrm{Re}[H(z)] > -1/k \tag{54}$$

where

$$H(z) = \mathbf{C}(\mathbf{I}z - \mathbf{A}_e)^{-1}\mathbf{B}_e \tag{55}$$

The Nyquist plot (Franklin et al. 2015) can be used to plot $H(e^{j\theta})\ \forall\ \theta \in [0, 2\pi]$. From this plot, the minimum value of $\mathrm{Re}[H(z)]$ that is corresponding to the $-1/k$ can be obtained.

It is noteworthy that if $f(u)$ in Eq. (33) is strictly within the following range:

$$k_{Min}u^2 < f(u)u < k_{Max}u^2 \tag{56}$$

Eqs. (41), (44) and (48) become

$$0 < f_e u_{i+1} < k u_{i+1}^2 \tag{57a}$$

$$f_e\left(f_e - k u_{i+1}\right) = f_e\left(f_e - k\mathbf{C}\boldsymbol{x}_i\right) < 0 \tag{57b}$$

$$\Delta v_{i+1} < \Delta \bar{v}_{i+1} \le 0 \tag{57c}$$

Therefore, the explicit direct integration algorithm is asymptotically stable in this case, i.e. $f(u)u \in \left(k_{Min}u^2, k_{Max}u^2\right)$, refer to Fig. 2.

Numerical Example for the Stability Analysis

In this section, the explicit Newmark algorithm, i.e. $\boldsymbol{\kappa} = \begin{bmatrix} 1 & 1/2 & 1/2 & 1/2 \end{bmatrix}$, is used to demonstrate the approach proposed in the previous sections based on the following numerical conditions:

$$m = 1, \quad \zeta = 0.05, \quad \mu = 0.01, \quad k_I = 1 \tag{58}$$

For convenience, all units in this section are omitted. Based on Eq. (58), $\mathbf{A}_e$, $\mathbf{C}$ and $\rho(\mathbf{A}_e)$ are as follows:

$$\mathbf{A}_e = \begin{bmatrix} -0.0051 & -0.0102 & -0.0039 \\ 0.4975 & 0.9949 & -0.0020 \\ 0.5000 & 1.0000 & 1.0000 \end{bmatrix} \tag{59a}$$

$$\mathbf{C} = \begin{bmatrix} 0.5000 & 1.0000 & 1.0000 \end{bmatrix} \tag{59b}$$

$$\rho(\mathbf{A}_e) = 0.9969 \tag{59c}$$

$\rho(\mathbf{A}_e) < 1$ implies that $\mathbf{A}_e$ is asymptotically stable and thus the first condition of the strictly positive realness of $G(z)$ in Eq. (50) is satisfied.

Stiffening Systems

For stiffening systems, $\mathbf{B}_e = \mathbf{B}_1$ is as follows:

$$\mathbf{B}_e = \begin{bmatrix} 0.0039 & 0.0020 & 0 \end{bmatrix}^T \tag{60}$$

The row ranks of the following matrices:

$$\mathbf{Z}_1 = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A}_e \\ \mathbf{C}\mathbf{A}_e^2 \end{bmatrix} \tag{61a}$$

$$\mathbf{Z}_2 = [\mathbf{B}_e \quad \mathbf{A}_e\mathbf{B}_e \quad \mathbf{A}_e^2\mathbf{B}_e] \tag{61b}$$

are equal to 3, i.e. $\text{rank}(\mathbf{Z}_1) = \text{rank}(\mathbf{Z}_2) = 3$. Therefore, $(\mathbf{A}_e, \mathbf{C})$ is observable and $(\mathbf{A}_e, \mathbf{B}_e)$ is controllable, and thus it is a minimal realization.

The Nyquist plot of $H(z)$ in eq. (55) corresponding to $\mathbf{A}_e$, $\mathbf{C}$ and $\mathbf{B}_e$ in Eqs. (59) and (60) is shown in Fig. 3, where $\min\{\text{Re}[H(z)]\} = -4.7642$ is obtained. Based on Eq. (54), one obtains:

$$-1/k < \min\{\text{Re}[H(z)]\} \tag{62a}$$

$$k < -1/\min\{\text{Re}[H(z)]\} = 0.2099 \tag{62b}$$

Accordingly, for stiffening systems, the explicit Newmark algorithm is stable in the sense of Lyapunov in the range that $f/u \in [k_I, k_I + k) = [1, 1.2099), |u| > 0$ for the numerical conditions in Eqs. (58).

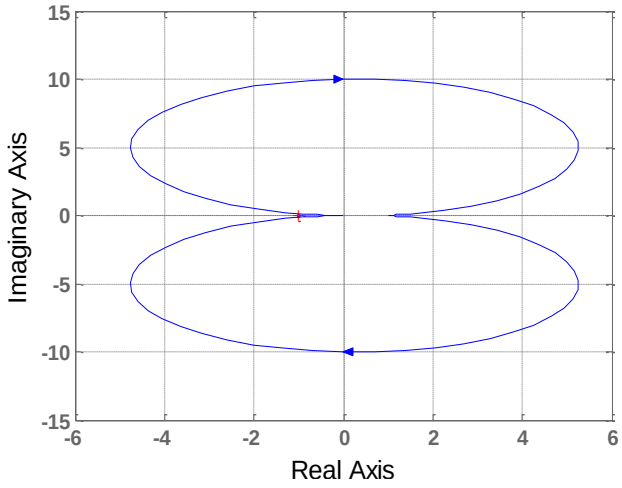


Fig. 3. Nyquist plot of $H(z)$ for a stiffening system.

Softening Systems

For softening systems, $\mathbf{B}_e = \mathbf{B}_2$ is as follows:

$$\mathbf{B}_e = -[0.0039 \quad 0.0020 \quad 0]^T \tag{63}$$

Fig. 4 shows the Nyquist plot of $H(z)$ in Eq. (55) corresponding to $\mathbf{A}_e$, $\mathbf{C}$ and $\mathbf{B}_e$ in Eqs. (59) and (63). Similar to Eqs. (62), with $\min\{\text{Re}[H(z)]\} = -5.2586$ obtained from Fig. 4, one obtains:

$$k < -1/\min\{\text{Re}[H(z)]\} = 0.1902 \tag{64}$$

Therefore, for softening systems, the explicit Newmark algorithm is stable in the sense of Lyapunov in the range that $f/u \in (k_I - k, k] = (0.8098, 1], |u| > 0$ for the numerical conditions in Eqs. (58).

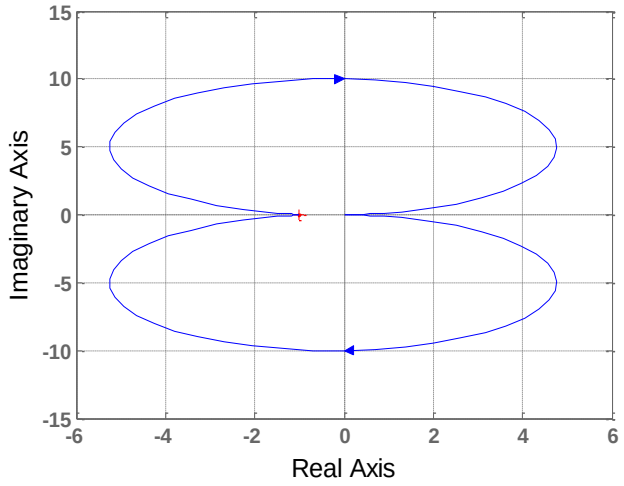


Fig. 4. Nyquist plot of $H(z)$ for a softening system.

The proposed approach can be applied to investigate the stability of other explicit direct integration algorithms. It is also noted that neither k_{Min} nor k_{Max} is necessarily equal to k_I for the stiffening or the softening systems, respectively. These stiffness values can be any other values of stiffness along the loading path. Therefore, various nonlinear systems, besides strictly stiffening and softening systems, can be treated using the proposed stability analysis approach. Accordingly, this proposed approach is generally applicable to explicit direct integration algorithms for various nonlinear systems.

Accuracy Analysis

The accuracy of the numerical integration algorithms depends on several factors, e.g. the loading, the time-step size, and the physical parameters of the system. In order to develop an understanding of this accuracy, a nonlinear test problem with an available closed-form exact solution is analyzed in this section.

Consider a simple pendulum (Fig. 5) of length l, forming a time-dependent angle $\theta(t)$ with the vertical axis and undergoing time-dependent angular acceleration $\ddot{\theta}(t)$. The governing equation, initial conditions, exact solution, and period of vibration are summarized in Table 5 where g is the gravitational acceleration, $n = K(r) - \bar{\kappa}_0 t$, $\bar{\kappa}_0 = \sqrt{g/l}$, $K(r)$ is the complete elliptical integral of the first kind, and $sn(n;r)$ is the Jacobi elliptic function (Abramowitz and Stegun 1972).

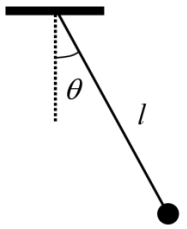


Fig. 5. Schematic of the nonlinear pendulum in a general deformed state.

Table 5. Nonlinear pendulum

Governing equation	$\ddot{\theta} + (g/l)\sin\theta = 0$
Initial conditions	$\theta(0) = \theta_0, \dot{\theta}(0) = 0$
Exact solution (Beléndez et al. 2007)	$\theta(t) = 2\arcsin\{\sin(\theta_0/2)sn(n;r)\}, r = \sin^2(\theta_0/2)$
Period	$T = 4K(r)/\bar{\kappa}_0$

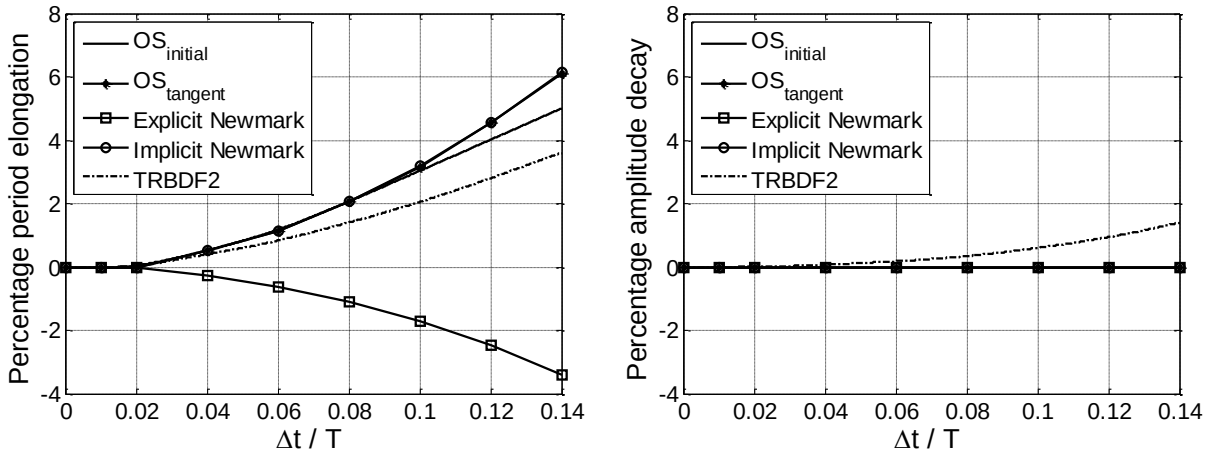


Fig. 6. Period elongation and amplitude decay for the pendulum problem with $\theta_0 = 0.10\pi$.

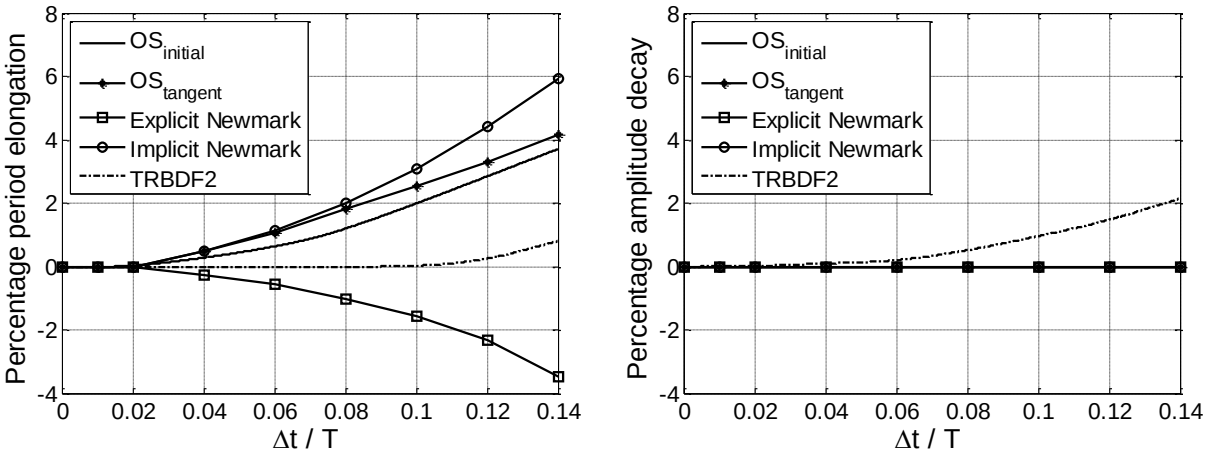


Fig. 7. Period elongation and amplitude decay for the pendulum problem with $\theta_0 = 0.50\pi$.

Figs. 6 and 7 present the period elongation and the amplitude decay of the investigated integration algorithms for $\theta_0 = 0.10\pi$ and $\theta_0 = 0.50\pi$, respectively. The period is shortened using explicit Newmark (Chopra 2006), and elongated by the other algorithms. It is observed that $OS_{tangent}$ and implicit Newmark present similar period elongations. The TRBDF2 has the smallest

period change while it is about twice computationally expensive compared to the other algorithms. Considering roughly the same computational efforts, e.g. $\Delta t/T = 0.08$ for TRBDF2 and $\Delta t/T = 0.04$ for the others, the accuracy becomes comparable. Moreover, the accuracy of all algorithms is indifferent for the integration time steps required for accuracy, i.e. $\Delta t/T < 0.01$ (Bathe 2006). All algorithms do not result in any significant amplitude decay except TRBDF2, which presents some amplitude decay due to introduced numerical damping. Up to $\Delta t/T = 0.1$, period elongation (< ±3%) and amplitude decay (< 1%) are acceptable.

Summary and Conclusions

Systematic Lyapunov stability and accuracy analyses of five direct integration algorithms (implicit and explicit Newmark, $OS_{initial}$, $OS_{tangent}$, and TRBDF2) for two types of nonlinear SDOF structural systems (stiffening and non-degrading softening), have been presented. An integration algorithm is stable if its Lyapunov artificial energy function is bounded. The general condition that the boundedness of $\prod_{j=i}^{1} \mathbf{A}_j$ for $i \to \infty$ is derived from the boundedness of the Lyapunov function. For linear structures, the stability criterion is that the spectral radius of the approximation operator is less than or equal to 1.0, which is applied to nonlinear structures by some researchers. It should be emphasized that the stability limit for linear structures, however, does not automatically hold for nonlinear structures. Therefore, some well-known stability limits of direct integration algorithms, e.g. OS algorithm with initial stiffness ($OS_{initial}$), are incomplete. An approach is proposed to perform the stability analysis numerically. This approach transforms the stability analysis to the solution of a convex optimization problem. The proposed approach is generally applicable to direct integration algorithms for nonlinear problems.

In addition, another systematic approach to investigate the Lyapunov stability of explicit direct integration algorithms has been presented. The generic explicit algorithm is formulated for a nonlinear system represented by a general nonlinear restoring force in terms of the displacement, u. In this approach, the restoring force is a memoryless nonlinear function bounded in the sector between $k_{Min}u$ and $k_{Max}u$ where k_{Min} and k_{Max} are lower and upper bounds on the SDOF stiffness. Based on this formulation, the proposed approach transforms the stability analysis to investigating the strictly positive realness of the transfer function for the formulated system. Nyquist plot is used to obtain a sufficient condition, i.e. the range of the restoring force, that the explicit algorithm is stable in the sense of Lyapunov. Furthermore, the explicit algorithm is asymptotically stable if the restoring force is strictly within the range given by Eq. (56). The proposed approach is demonstrated by a numerical example that investigates the Lyapunov stability of the explicit Newmark algorithm and this illustrated by a numerical example of a SDOF. In conclusion, the proposed approach is shown to be applicable for investigating the stability of the explicit direct integration algorithms used to determine the dynamical response of a variety of nonlinear structural systems.

A geometrically nonlinear pendulum problem with a closed-form exact solution is used to investigate the accuracy of the considered five integration algorithms. The period is shortened by explicit Newmark and elongated by the other algorithms. The $OS_{tangent}$ and implicit Newmark present similar period elongations. The more computationally expensive TRBDF2 has the

smallest period change. All algorithms present no amplitude decay except the TRBDF2 undergoes some amplitude decay due to the introduced numerical damping. Finally, observed period elongation (< ±3%) and amplitude decay (< 1%) values are practically acceptable.

Acknowledgements

The research was supported by Caltrans (Contract # 65A0454) for the project "Guidelines for nonlinear seismic analysis of ordinary bridges." The authors thank Caltrans for this support. The authors recognize Ms. Minghui Zheng for her valuable suggestions and advices.

References

Abramowitz, M. and Stegun, I. A. (1972). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Dover Publications, New York.

Bank, R. E., Coughran, W. M., Fichter, W., Grosse, E.H., Rose, D.J. and Smith, R.K. (1985). "Transient simulations of silicon devices and circuits" *IEEE TransCAD*, 4, 436–451.

Bathe, K. J. (2006). *Finite Element Procedures*, Prentice Hall, Englewood Cliffs, N.J.

Bathe, K. J. (2007). "Conserving Energy and Momentum in Nonlinear Dynamics: A Simple Implicit Time Integration Scheme" *Comput. Struct.*, 85, 437–445.

Bathe, K. J. and Baig, M. M. I. (2005). "On a composite implicit time integration procedure for nonlinear dynamics" *Comput. Struct.*, 83, 2513–2524.

Bathe, K. J. and Wilson, E. L. (1973). "Stability and accuracy analysis of direct integration methods," *Earthquake Eng. Struct. Dyn.*, 1, 283–291.

Beléndez, A., Pascual, C., Méndez, D. I., Beléndez, T. and Neipp, C. (2007). "Exact solution for the nonlinear pendulum" *Revista Brasileira de Ensino de Física*, 29(4), 645–648.

Cains, P. E. *Linear Stochastic Systems*, New York, Wiley, 1989.

Chen, C. and Ricles, J. M. (2008). "Stability analysis of direct integration algorithms applied to nonlinear structural dynamics." *J. Eng. Mech*., 134(9), 703–711.

Chopra, A. K. (2006). *Dynamics of Structures: Theory and Applications to Earthquake Engineering*, Pearson Prentice Hall, 3rd Edition, Upper Saddle River, N.J.

Combescure, D. and Pegon, P. (1997). "α-Operator splitting time integration technique for pseudodynamic testing error propagation analysis," *Soil Dyn. Earthquake Eng.*, 16, 427–443.

CVX Research, Inc. (2011). *CVX: Matlab software for disciplined convex programming*, version 2.0. http://cvxr.com/cvx

Franklin, G. F., Powell J. D. and Emami-Naeini A. (2015). *Feedback Control of Dynamic Systems*, 7th Edition, Pearson Higher Education Inc., Upper Saddle River, N.J.

Hughes, T. J. R. (1976). "Stability, convergence and growth and decay of energy of the average acceleration method in nonlinear structural dynamics," *Comput. Struct.*, 6, 313–324.

Hughes, T. J. R., Pister, K. S. and Taylor, R. L. (1979). "Implicit-Explicit Finite Elements in Nonlinear Transient Analysis," *Comput. Methods in Appl. Mech. Eng*., 17/18, 159–182.

Kapila, V. and Haddad, W. M. (1996). "A Multivariable Extension of the Tsypkin Criterion Using a Lyapunov-Function Approach," *IEEE T. Automat. Contr.*, 41(1), 149–152.

Liang, X., Günay, S. and Mosalam, K. M. (2014). "Integrators for Nonlinear Response History Analysis: Revisited," *Proc., Istanbul Bridge Conf.*, Istanbul, Turkey.

Nakashima, M., Kaminosono, T., Ishida, M., Ando, K. (1990). "Integration technique for substructure pseudodynamic test," *Proc., of the 4th U.S. National Conf. on Earthquake Eng.*, Palm Springs, CA, 12, 515–524.

Newmark, N. M. (1959). "A method of computation for structural dynamics," *ASCE J. Eng. Mech. Div.*, 85(3), 67–94.

Appendix: Base Functions Used for Numerical Stability Analysis

$$\mathbf{\Phi}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{\Phi}_4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{\Phi}_5 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_7 = \begin{bmatrix} \delta_{i+1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_8 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \delta_{i+1} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{\Phi}_9 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \delta_{i+1} \end{bmatrix}, \quad \mathbf{\Phi}_{10} = \begin{bmatrix} 0 & \delta_{i+1} & 0 \\ \delta_{i+1} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_{11} = \begin{bmatrix} 0 & 0 & \delta_{i+1} \\ 0 & 0 & 0 \\ \delta_{i+1} & 0 & 0 \end{bmatrix}, \quad \mathbf{\Phi}_{12} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \delta_{i+1} \\ 0 & \delta_{i+1} & 0 \end{bmatrix}$$