ENGINEERING COMPUTATIONS, VOL. 9, 669–679 (1992)

AN EFFICIENT ALGORITHM FOR THE COMPUTATION OF STABILITY POINTS OF DYNAMICAL SYSTEMS UNDER STEP LOAD

P. WRIGGERS

Institut für Mechanik, TH Darmstadt, Hochschulstr. 1, D-6100 Darmstadt, Germany

AND

C. CARSTENSEN

Institut für Angewandte Mathematik, Universität Hannover, Welfengarten 1, D-3000 Hannover 1, Germany

INTRODUCTION

Many engineering structures exhibit loss of stability under static and dynamic loading. Due to the significance of these phenomena in engineering design this topic has attracted considerable attention during the last decades. In recent years much effort has been made to devise algorithms within finite element analysis to investigate the static stability behaviour of structures. With these methods stable and unstable paths can be traced[1–3], and limit or bifurcation points can be computed efficiently[4,5]. The associated arc-length or branch-switching procedures are today standard tools in existing finite element codes.

Analogous procedures for non-linear dynamic stability problems have not been developed to this extent. This has different reasons. First, the underlying theory is not that well suited for numerical applications and the numerical computation of large scale problems is very time consuming. Thus many stability problems which exhibit dynamic effects like buckling or snap through of shells are treated as static cases. However, as has been pointed out by different authors this may restrict the space of solutions since isolated but stable post buckling states may exist which cannot be reached by static continuation procedures[6,7].

If, on the other hand, a dynamic analysis of a structure which may bifurcate is performed we need algorithms to detect instability. Contrary to the static case where the determinant of the tangent stiffness matrix indicates loss of stability we have to differentiate between several cases in dynamic problems.

Within the autonomous case we can use the so-called Ljapunov's first approximation where still the tangent stiffness matrix determines the stability behaviour. In the early 1950s researchers started to investigate these load cases[8]. Several algorithms have been developed for this class of dynamic stability. One approach is associated with the work of Budiansky and Roth[9] who observed that in the presence of loss of stability the system undergoes large deflection. Thus they formulated the criterion of finite change of deformations within the dynamic response which defines the critical load. Within finite element discretizations methods like accompanying eigenvalue analysis have been used[3,10–12]. In general, if the load magnitude is smaller than the critical load the system oscillates around static equilibrium. After the load magnitude is increased

0264–4401/92/060669–11\$2.00

Received September 1991

beyond the critical load the structure can exhibit large oscillations or even a divergent type of motion.

It is interesting that efficient algorithms are lacking for the autonomous case which are able to detect a singularity of the tangent operator. This is mainly due to the fact that in standard finite element analysis the integration of the equations of motions is performed via implicit schemes like Newmark or so-called α-methods[13], which augment the tangent stiffness by the mass and damping matrices. Thus this *effective* stiffness matrix is factored and not the tangent matrix itself which means that for a computation of the determinant of the tangent matrix we need a refactorization in the solution process. This is extremely expensive and time consuming. To circumvent this disadvantage we suggest in this paper an accompanying eigenvalue investigation based on an iterative scheme.

The paper is outlined as follows. First, we will recall the equations of the underlying non-linear finite element formulation. After this we will state a standard implicit time-stepping procedure for non-linear problems and develop an efficient algorithm for the accompanying eigenvalue investigation based on the coordinate overrelaxation method for computing the minimum of the Rayleigh quotient[14]. Finally, some examples conclude the paper and show the applicability of the theory and the efficiency of the derived algorithm.

NON-LINEAR FINITE ELEMENT FORMULATION

In this section we summarize standard notation in non-linear finite element analysis needed for our subsequent developments.

Standard finite element procedures yield in case of dynamics a non-linear system of ordinary differential equations:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{R}(\mathbf{u}) - \mathbf{P}(t) = \mathbf{0} \qquad \mathbf{u}, \dot{\mathbf{u}}, \ddot{\mathbf{u}} \in \mathbb{R}^n \tag{1}$$

with the initial conditions

$$\mathbf{u}(0) = \mathbf{u}_0 \qquad \dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0$$

Here the body B under consideration is discretized by n_e finite elements Ω_e which leads to the approximation $B^h = \cup_{e=1}^{n_e} \Omega_e$. $\mathbf{u}$ denotes the deformed configuration, $\dot{\mathbf{u}}$ the velocity and $\ddot{\mathbf{u}}$ the acceleration of the body. The vectors $\mathbf{R}, \mathbf{P}$ and the matrix $\mathbf{M}$ are given by:

$$\begin{aligned}
\mathbf{R}(\mathbf{u}) &= \bigcup_{e=1}^{n_e} \int_{\Omega_e} \mathbf{B}^t(\mathbf{u}_e)\sigma \, \mathrm{d}\Omega \\
\mathbf{P}(t) &= \bigcup_{e=1}^{n_e} \int_{\Omega_e} \mathbf{N}^t \rho \bar{\mathbf{b}}(t) \, \mathrm{d}\Omega + \int_{\partial\Omega_e} \mathbf{N}^t \bar{\mathbf{t}}(t) \, \mathrm{d}(\partial\Omega) \\
\mathbf{M} &= \bigcup_{e=1}^{n_e} \int_{\Omega_e} \rho \mathbf{N}^t \mathbf{N} \, \mathrm{d}\Omega
\end{aligned} \tag{2}$$

where $\mathbf{N}$ contains the shape functions and $\mathbf{B}$ their derivatives according to the theory used. $\mathbf{R}$ denotes the so-called stress divergence term which may in general include damping and $\mathbf{P}(t)$ is the load pattern which in general depends on the time t. ρ is the density, $\bar{b}(t)$ are the body forces and $\bar{\mathbf{t}}(t)$ are the tractions. Due to geometrical or material non-linearities the system of equations (1) is non-linear in $\mathbf{u}$.

The degree of non-linearity in (1) depends on the theory used to describe the problem at hand. When for example a three-dimensional theory of non-linear elasticity is considered which bases on the St. Venant material with Green strains then **R** is a cubic polynominal in **u**. This no longer holds for different constitutive models like Ogden's strain energy function for rubberlike materials which can also be described by (1). In shell theories for large rotations **u** depends on trigonometric functions. Finally the incorporation of elastoplastic materials leads, depending on the plasticity model, to different types of non-linearity. However, once the problem is formulated and discretized the approach advocated in this paper can be applied independently from the underlying non-linear model.

M is the consistent mass matrix which may also be approximated by lumping techniques to arrive at a computationally more efficient diagonal form. For a discussion of different approaches, see Ref 13. Throughout this paper we will assume that damping is given by the so-called Rayleigh damping which yields the following structure of the damping matrix **C**

$$\mathbf{C} = d_1\mathbf{M} + d_2\mathbf{K} \qquad d_\alpha \in \mathbb{R} \tag{3}$$

where d_α are constants, the mass matrix **M** is defined in $(2)_3$ and **K** denotes the tangent stiffness linearized at the state $\mathbf{u}_0$: $\mathbf{K} = D\mathbf{R}(\mathbf{u})|_{u=u_0}$.

DYNAMIC STABILITY UNDER STEP LOAD

In this section we formulate the well known stability results for the autonomous loading case which will be the basis for the subsequent chapters.

The initial value problem:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{R}(\mathbf{u}) = \mathbf{P}(t) \qquad \mathbf{u}(0) = \mathbf{u}_0 \qquad \dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0 \tag{4}$$

with $\mathbf{M}, \mathbf{C} \in \mathbb{R}^{n \times n}$, **M** positive definite, and the C^2 maps $\mathbf{R}: \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{P}: \mathbb{R}^n \to \mathbb{R}^n$, can be transformed via

$$\mathbf{x} = \begin{Bmatrix} \mathbf{u} \\ \dot{\mathbf{u}} \end{Bmatrix} \in \mathbb{R}^{2n}$$

into the initial value problem $\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}(t))$

$$\dot{\mathbf{x}} = \begin{Bmatrix} \dot{\mathbf{u}} \\ -\mathbf{M}^{-1}[\mathbf{R}(\mathbf{u}) + \mathbf{C}\dot{\mathbf{u}} - \mathbf{P}(t)] \end{Bmatrix} \qquad \mathbf{x}(0) = \begin{Bmatrix} \mathbf{u}_0 \\ \dot{\mathbf{u}}_0 \end{Bmatrix} \tag{5}$$

Problems (4) and (5) are equivalent.

Let $\mathbf{x}: (-a, +a) \to \mathbb{R}^{2n}$ be the solution of (5), $a > 0$, and let $\mathbf{f}: \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ be defined by the right hand side of (5). Then for all $t \in (-a, +a)$ there holds

$$D_2\mathbf{f}(t, \mathbf{x}(t)) = \begin{bmatrix} \mathbf{D} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}(t) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \tag{6}$$

where $\mathbf{K}(t) = D_2\mathbf{R}(\mathbf{x}(t))$ is the Jacobian of **R** at $\mathbf{x}(t)$, i.e. the tangent stiffness matrix. Note that the eigenvalues of $D_2\mathbf{f}$ are just the eigenvalues of the λ-matrix:

$$[\lambda^2\mathbf{M} + \lambda\mathbf{C} + \mathbf{K}] \tag{7}$$

If we assume that the damping matrix is given by $\mathbf{C} = d_1\mathbf{M}$ for a fixed positive number d_1. Then any eigenvalue λ of (7) can be computed from:

$$\lambda = -\frac{d_1}{2} \pm \sqrt{\frac{d_1^2}{4} + \mu} \tag{8}$$

with an eigenvalue μ of $[\mu\mathbf{M} + \mathbf{K}]$. Little calculation shows that λ is an eigenvalue of (5) with a related eigenvector $\{\varphi, \eta\}^t \in \mathbb{R}^{2n}$ if λ is an eigenvalue of (7) with a related eigenvector $\varphi, \eta = \lambda\varphi$.

The well known theorem of Ljapunov stability for autonomous systems can now be formulated as:

Let the solution $\mathbf{u}$ of (5) exists on $[0, a]$ and let $t_0 \in (0, a)$. In addition let $\mathbf{K} = D\mathbf{R}(\mathbf{u})$ be symmetric and $\mathbf{C} = d_1\mathbf{M}$ with $\mathbf{M}$ positive definite and $d_1 > 0$. If $\mathbf{K}(t_0) = D\mathbf{R}(\mathbf{u}(t_0))$ is positive definite then the solution $\mathbf{u}$ is stable in t_0. If $\mathbf{K}\ (t_0)$ has a negative eigenvalue then $\mathbf{u}$ is unstable in t_0.

This theorem can be used to investigate the dynamic stability behaviour of (4) just by looking at the determinant of the eigenvalues of $\mathbf{K}(t)$.

REMARK: By defining the notion of current stability[15], we can also use the static criterion in non-autonomous cases as a measure for the sensitivity of a dynamical system against imperfections. One can then show that if $\mathbf{K}(t_0) = D\mathbf{R}(\mathbf{u}(t_0))$ is positive definite then the solution $\mathbf{u}$ is c-stable in t_0. If $\mathbf{K}(t_0)$ has some negative eigenvalue then $\mathbf{u}$ is c-unstable in t_0 and thus sensitive against imperfections. Whereas classical asymptotic stability is important (if it can be proved) we stress insensitivity as an indication of a safe engineering construction. Conversely, sensitivity characterizes the possibility that small perturbations of the current situation can increase and thus can lead to a change in the asymptotic behaviour. Consequently, sensitive situations should be considered with great attention in both the engineering design and numerical simulations.

In the next section we develop an algorithm which detects the needed zero or negative eigenvalues of $\mathbf{K}(t)$.

ALGORITHMS FOR NON-LINEAR DYNAMIC STABILITY

In this section the iterative procedure for the solution of non-linear problems in dynamics and the algorithm for the computation of the eigenvalue of the tangent stiffness matrix $\mathbf{K}_T$ are described.

Implicit integration of the equations of motion

As has been pointed out, (1) denotes a system of non-linear ordinary differential in time t. The methods for the solution of (1) are standard. For many applications implicit integration schemes like the ones known as the Newmark family are advantageous. The application of these schemes to (1) leads to:

$$\mathbf{M}\mathbf{a}_{n+1} + \mathbf{C}\mathbf{v}_{n+1} + \mathbf{R}(\mathbf{u}_{n+1}) = \mathbf{P}_{n+1}$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t\mathbf{v}_n + \frac{\Delta t^2}{2}[(1-2\beta)\mathbf{a}_n + 2\beta\mathbf{a}_{n+1}]$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \Delta t[(1-\gamma)\mathbf{a}_n + \gamma\mathbf{a}_{n+1}] \tag{9}$$

where the subscript n means a known quantity at time t_n and the subscript $n+1$ denotes an unknown quantity at t_{n+1}. In (9) we have defined the acceleration by $\mathbf{a} = \ddot{\mathbf{u}}$ and the velocity by $\mathbf{v} = \dot{\mathbf{u}}$. β and γ are parameters which determine the accuracy and stability characteristics of the algorithm. It can be seen from $(9)_2$ and $(9)_3$ that a choice of $\beta = 0.25$ and $\gamma = 0.5$ yields the trapezoidal rule.

The elimination of $\mathbf{a}_{n+1}$ and $\mathbf{v}_{n+1}$ from $(9)_1$ by $(9)_2$ and $(9)_3$ yields a non-linear equation in $\mathbf{u}_{n+1}$

$$(c_1\mathbf{M} + c_2\mathbf{C})\mathbf{u}_{n+1} + \mathbf{R}(\mathbf{u}_{n+1}) = \mathbf{P}_{n+1} + \mathbf{M}\hat{\mathbf{a}}_n + \mathbf{C}\hat{\mathbf{v}}_n \tag{10}$$

with

$$\hat{\mathbf{a}}_n = c_1\mathbf{u}_n + c_3\mathbf{v}_n + c_5\mathbf{a}_n$$

$$\hat{\mathbf{v}}_n = c_2\mathbf{u}_n + c_4\mathbf{v}_n + c_6\mathbf{a}_n \tag{11}$$

and

$$c_1 = \frac{1}{\beta \Delta t^2}, \quad c_2 = \frac{\gamma}{\beta \Delta t}, \quad c_3 = \frac{1}{\beta \Delta t}$$
$$c_4 = \frac{\gamma}{\beta} - 1, \quad c_5 = \frac{1}{2\beta} - 1, \quad c_6 = \Delta t\left(\frac{\gamma}{2\beta} - 1\right) \tag{12}$$

Here $\Delta t = t_{n+1} - t_n$ is the time increment used in the algorithm.

If we now assume Rayleigh damping, see (3), with $d_2 = 0$ the damping matrix reduces to:

$$\mathbf{C} = d_1 \mathbf{M} \tag{13}$$

Thus we obtain

$$(c_1 + d_1 c_2)\mathbf{M}\mathbf{u}_{n+1} + \mathbf{R}(\mathbf{u}_{n+1}) = \mathbf{P}_{n+1} + \mathbf{M}\tilde{\mathbf{a}}_n \tag{14}$$

with

$$\tilde{\mathbf{a}}_n = \hat{\mathbf{a}}_n + d_1 \hat{\mathbf{v}}_n \tag{15}$$

To solve the non-linear equation (14) we will apply Newton's method. For this purpose we have to linearize (14) with respect to the unknown configuration $\mathbf{u}_{n+1}$ which yields the following algorithm within a time step Δt

$$[(c_1 + d_1 c_2)\mathbf{M} + \mathbf{K}_T]\Delta\mathbf{u}^{i+1} = -[\mathbf{R}(\mathbf{u}^i) - \mathbf{P}_{n+1} - \mathbf{M}[(c_1 + d_1 c_2)\mathbf{u}^i + \tilde{\mathbf{a}}_n]]$$
$$\mathbf{u}^{i+1} = \mathbf{u}^i + \Delta\mathbf{u}^{i+1} \tag{16}$$

where i is the iteration index and $\mathbf{K}_T = D\mathbf{R}(\mathbf{u}^i)$.

As can be seen from $(16)_1$ the effective tangent matrix $\mathbf{K}_{\text{eff}} = (c_1 + d_1 c_2)\mathbf{M} + \mathbf{K}_T$ has to be factorized during each Newton iteration. In case of theorem III the standard approach is to monitor the determinant of the tangent matrix $\mathbf{K}_T$. This requires a factorization not only of $\mathbf{K}_{\text{eff}}$ but also of $\mathbf{K}_T$ which is double the effort of solving (16) within one iteration step. To avoid this, we develop an algorithm based of a coordinate overrelaxation method which is described in the next section.

Current eigenvalue analysis by coordinate overrelaxation (COR)

In this section we consider the general eigenvalue problem $(\mathbf{A} - \lambda\mathbf{B})\varphi = \mathbf{0}$ for $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$ symmetric and $\mathbf{B}$ positive definite where $\mathbf{A} = \mathbf{K}$ is the current tangential stiffness matrix and $\mathbf{B} = \mathbf{M}$ is the mass matrix. Since we want to detect a possible singularity of $\mathbf{K}$ we are led to the computation of the smallest eigenvalue λ_1. Clearly this can be tackled by several algorithms like factorization or inverse iteration for instance. But this increases the computational effort for the current eigenvalue analysis by the same amount which is needed for the implicit integration of the equations of motions itself.

Our basic observation in the applications section is that the eigenvector φ depends only little on time. Therefore we suggest in this paper the application of the coordinate overrelaxation (COR).

The COR has been pointed out by Schwarz[14]. It is based on searching the smallest stationary value of the Rayleigh quotient:

$$\lambda_1 = \min_{\varphi \neq 0} R(\varphi) \qquad R(\varphi) = \frac{\varphi^t \mathbf{A} \varphi}{\varphi^t \mathbf{B} \varphi}$$

Given an approximation φ of an eigenvector related to λ_1 we chose a direction $\eta \neq 0$ and minimize R over the two-dimensional space generated by φ and η. Let $\varphi' = \phi\varphi + \gamma\eta$ be the minimizer, $\lambda' = R(\varphi')$, then the stationary conditions $\partial R(\varphi')/\partial\phi = 0 = \partial R(\varphi')/\partial\gamma$ lead to the two-dimensional eigenvalue problem for λ'

$$\begin{bmatrix} \varphi^t\mathbf{A}\varphi - \lambda'\varphi^t\mathbf{B}\varphi & \varphi^t\mathbf{A}\eta - \lambda'\varphi^t\mathbf{B}\eta \\ \eta^t\mathbf{A}\varphi - \lambda'\eta^t\mathbf{B}\varphi & \eta^t\mathbf{A}\eta - \lambda'\eta^t\mathbf{B}\eta \end{bmatrix} \begin{Bmatrix} \phi \\ \gamma \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

which is easily solved in each step such that $\lambda' \leqslant R(\eta)$. Therefore φ' is regarded as a better approximation of the eigenvector related to λ_1 than φ. In one iteration cycle we chose φ as the unit vectors $e_1, \ldots, e_n$ immediately after each other.

Hence we obtain the following algorithm:

Algorithm COR

Input: $\varphi \neq 0$, $\omega \in (1, 2)$, $\varepsilon > 0$.
Start: Compute $\zeta = \mathbf{A}\varphi, \xi = \mathbf{B}\varphi, \alpha = \varphi^t\zeta, \beta = \varphi^t\xi, \mathbf{R} = \alpha/\beta$
Repeat:
For $j = 1, \ldots, n$ calculate R' and γ_j from

$$\begin{bmatrix} \alpha - R'\beta & \zeta_j - R'\xi_j \\ \zeta_j - R'\xi_j & A_{jj} - R'B_{jj} \end{bmatrix} \begin{Bmatrix} 1 \\ \gamma \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

and let

$$\gamma_j = \omega\gamma$$
$$\varphi_j = \varphi_j + \gamma_j$$
$$\alpha = \alpha + 2\gamma_j\zeta_j + \gamma_j^2 A_{jj}$$
$$\beta = \beta + 2\gamma_j\xi_j + \gamma_j^2 B_{jj}$$
$$R = \alpha/\beta$$
$$\zeta = \zeta + \gamma_j(A_{j1}, \ldots, A_{jn})^t$$
$$\xi = \xi + \gamma_j(B_{j1}, \ldots, B_{jn})^t$$

Until: $\|(\gamma_1, \ldots, \gamma_n)^t\|_\infty < \varepsilon$

BOX 1 Coordinate-overrelaxation algorithm

Here ω is the overrelaxation parameter which is introduced to accelerate the convergence. The choice of ω is discussed numerically in Schwarz[14]. In a later section we have used $\omega = 1.4$ and $\varepsilon = 10^{-3}$.

REMARKS

(i) During the implicit time stepping procedure we apply as a starting vector φ in COR at time t_{n+1} the converged eigenvector from time step t_n. At t_0, i.e. the beginning of the implicit integration for the initial values, we suggest a single computation of the eigenvector φ by inverse iteration or a subspace algorithm with higher accuracy because in our experiences COR seems an efficient algorithm for lower accuracy and becomes too laborious for higher accuracy, cf. Schwarz[14].

(ii) Since the eigenvector φ undergoes minor changes by passing the problem from t_n to t_{n+1} we need only a few (less than 5) iteration cycles (i.e. the fore-loop in COR). The essential computational effort of such an iteration cycle is one vector multiplication with A and B. Therefore, besides the calculation of the starting vector at t_0, the computational effort is reduced from $O(n^3)$ (for factorization) to $O(n^2)$ (for COR) in each time step.

(iii) If the second eigenvalue λ_2 is equal or close to λ_1 the convergence of COR may fail. This can be avoided by a successive application of COR for higher stationary values, cf. Schwarz[14].

General algorithm

In this section we state the general algorithm which we apply to detect singular points of the tangent operator within a non-linear dynamic simulation. In general we observe the response shown in *Figure 1* where at a certain singular point the solution branches in the unperturbed and the perturbed one. However, this must not be the case. The dynamical response of the non-linear system can also be insensitive with respect to perturbations. Then the solution curve will return to the unperturbed one. Thus to test the sensitivity at a singularity at time t_n we perturb the solution $\mathbf{u}_n$ by the eigenvector φ_n as follows: $\mathbf{u}_\varepsilon = \mathbf{u}_n + \varepsilon\varphi_n$.

The application of the Newmark method and the COR technique leads to the following global procedure:

General algorithm

- Set starting vectors: $\mathbf{u}_0$, $\mathbf{v}_0$, $\mathbf{a}_0$
- Compute starting eigenvector: φ_0 from $(\mu\mathbf{M} + \mathbf{K}_0)\varphi_0 = \mathbf{0}$
- Loop over all time steps $n = 1, \ldots, n_{\text{end}}$.
  - — Solve non-linear system (14) by Newton's method
  - — Compute with COR eigenvalue and −vector φ_n of $[\mu\mathbf{M} + \mathbf{K}_n]$
  - — In case of a negative eigenvalue perturb solution via $\mathbf{u}_\varepsilon = \mathbf{u}_n + \varepsilon\varphi_n$
  - — Update vectors: $\mathbf{u}_n$, $\mathbf{v}_n$, $\mathbf{a}_n$, φ_n
- Loop end

BOX 2 General algorithm for non-linear structures

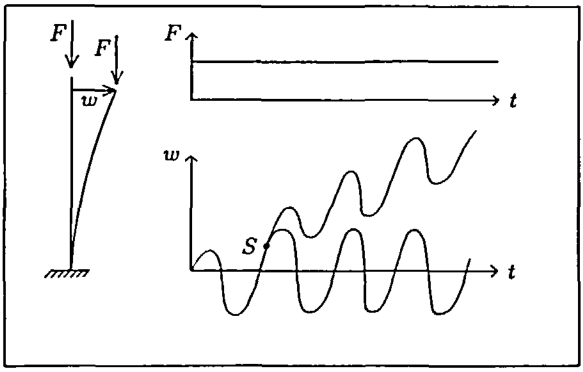

Figure 1 Global response of a non-linear structure

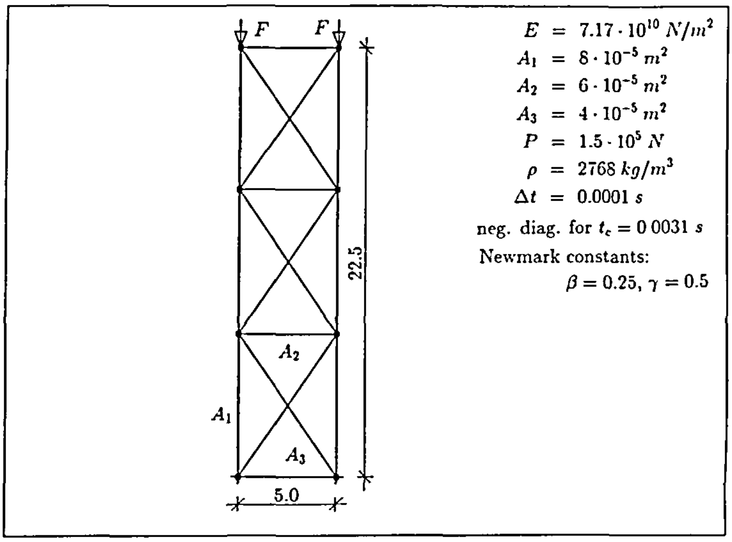

Figure 2 System and data of the two-dimensional non-linear truss

NUMERICAL EXAMPLES

In this section we present a selected sample of numerical simulations of non-linear stability processes and apply the COR method. All algorithms and elements have been implemented in FEAP, see Zienkiewicz and Taylor[16].

Dynamic stability of a non-linear truss structure

Our first example was also considered in Kleiber *et al.*[10]. The truss structure depicted in *Figure 2* is subjected to a vertical load which leads to buckling of the structure in the static case. All data are shown in *Figure 2.*

When instead of a static load a step load of the magnitude shown in *Figure 2* is applied we observe a negative eigenvalue of $\mathbf{K}_T$ during the motion of the structure at time $t = 0.0031$ sec. The location of the negative eigenvalue is marked by a dot in the curve in *Figure 3* showing the displacement *versus* time. This is, as disucssed in the previous sections, an indicator for loss of stability and leads in this example in the large to an unstable behaviour (*Figure 3*) when the displacement is perturbed by the eigenvector at the singular point of $\mathbf{K}_T$. The phase portraits of the perturbed and unperturbed solution in *Figure 4* indicate furthermore the totally different behaviour of both responses.

Dynamic stability of a frame structure

The frame structure considered in *Figure 5* is subjected to a constant load of infinite duration at its top. The relevant data of the mesh consisting of 247 elements and 140 nodes are given in *Figure 5.*

When a dynamic analysis is performed with a time step of $\Delta t = 0.001$ the COR method indicates a loss of stability at time $t = 0.072$. However, the solution does not change. It is basically a motion in vertical direction. If we now perturb the displacement field at the time associated with the loss of stability we are led to a different time history plot of the displacement u, see *Figure 6*, which depicts a totally different behaviour. The COR method used on the average 1

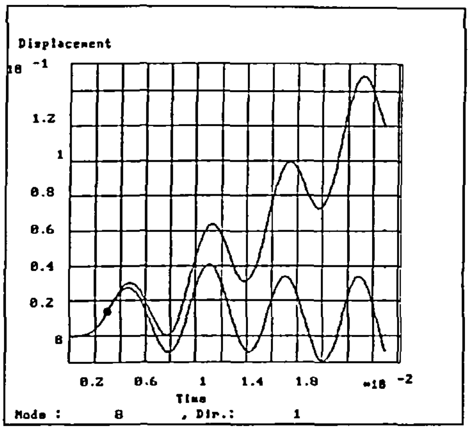

Figure 3 Displacement *versus* time for non-linear truss

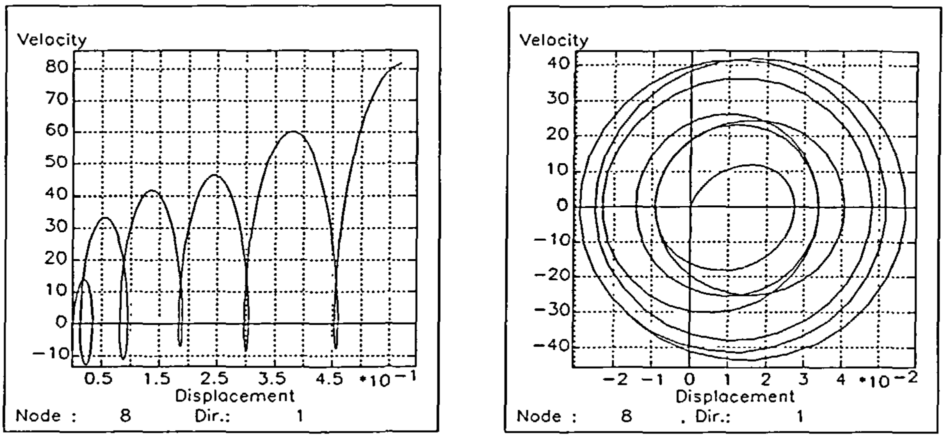

Figure 4 Phase portrait of the perturbed and unperturbed solutions

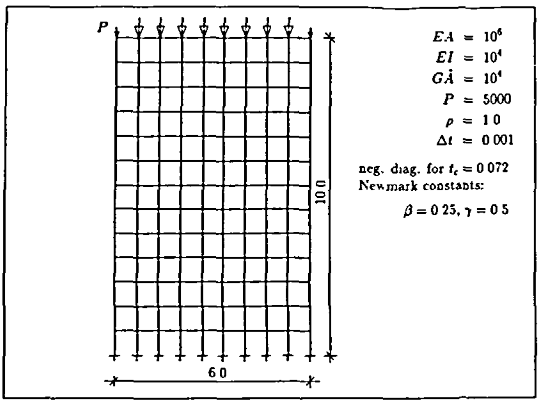

Figure 5 System and data of the frame structure

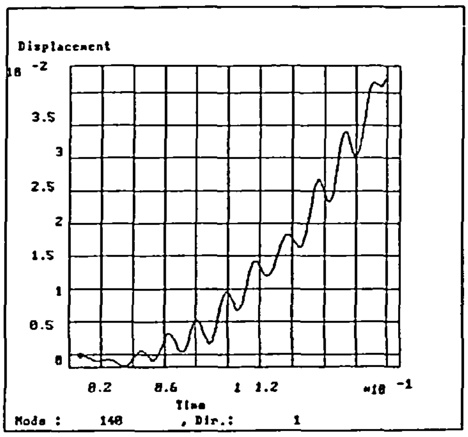

Figure 6 Time history plot of the displacement *u*

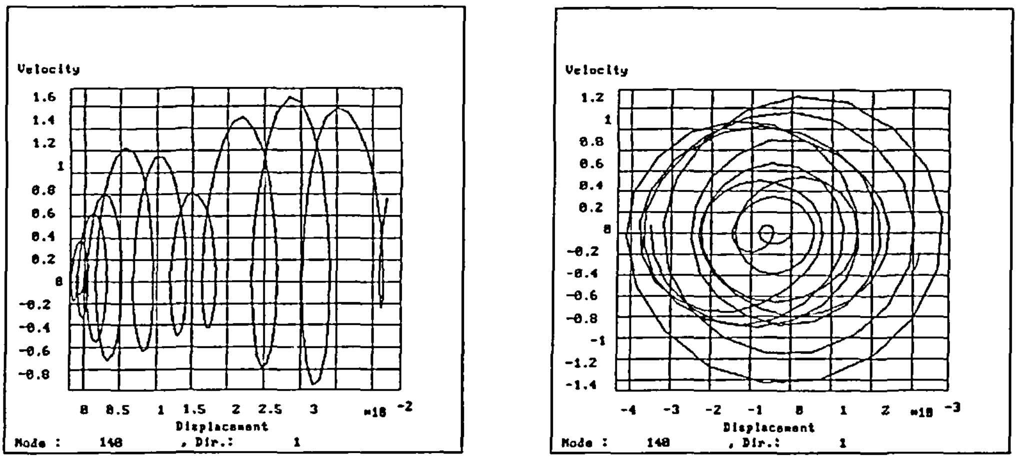

Figure 7 Phase diagram of the perturbed and unperturbed solutions

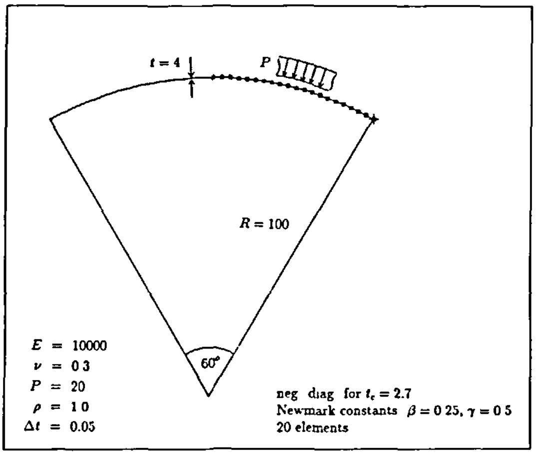

Figure 8 System and data of the spherical cap

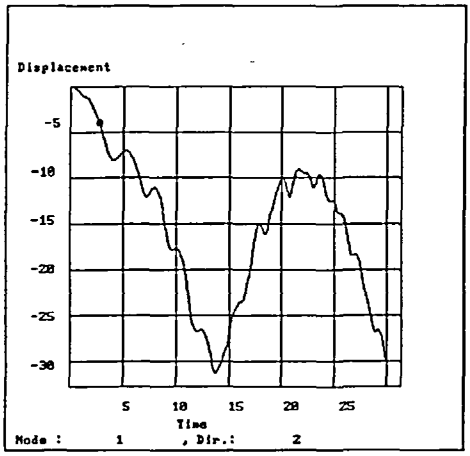

Figure 9 Apex displacement *versus* time for spherical shell

to 2 cycles for the determination of the current eigenvalue in each time step. Hence the time for the computation of the current eigenvalue was one twentieth of the total solution time.

The phase diagram in *Figure 7* shows clearly the instability of the perturbed solution.

Clamped spherical shell

This example shows the stability behaviour of a clamped spherical shell under constant pressure load of infinite duration. It has been considered by many authors, see e.g. Simitses[8], Ch. 8, and references therein. System and data are shown in *Figure 8*.

For a pressure load of $p = 20$ we observe snap-through behaviour. At time $t = 2.7$ a negative diagonal of the tangent matrix $\mathbf{K}_T$ occurred. This point is marked in the time–history plot of the displacement w at the apex, see *Figure 9*. This Figure clearly depicts the oscillations of the spherical cap around the new equilibrium state after the snap appeared.

To detect the singularity of the tangent matrix the algorithm needed in each time step in the average 2 iterations when $\omega = 1.4$ and $\varepsilon = 10^{-3}$ where used. A tighter tolerance was not needed since only the change of sign of the eigenvalue had to be investigated for the detection of the singularity.

CONCLUSIONS

The algorithm developed in this paper is able to detect singular points of the tangent matrix within an implicit time stepping scheme in an efficient manner. Since the loss of stability of autonomous systems is denoted by the singularity of the tangent operator this method can be used as an indicator for this type of stability behaviour. Due to the structure of the algorithm one has to initialize it with an eigenvector which can be computed efficiently by an inverse iteration. Then it turned out that the use of an relaxation parameter $\omega = 1.4$ and a solution tolerance of $\varepsilon = 10^{-3}$ were sufficient to detect the singularities with 2 to 3 iteration cycles per time step.

REFERENCES

1 Riks, E. The application of Newton's method to the problem of elastic stability, *J. Appl. Mech.*, 39, 1060–1066 (1972)
2 Crisfield, M. A. A fast incremental/iterative solution procedure that handles snap through, *Comp. Struct.*, 13, 55–62 (1981)
3 Ramm, E. Strategies for tracing the nonlinear response near limit points, in *Nonlinear Finite Element Analysis in Structural Mechanics*, (Eds Wunderlich, Stein, Bathe), Springer, Berlin-Heidelberg-New-York (1981)
4 Moore, G. and Spence, A. The calculation of turning points of nonlinear equations, *SIAM J. Num. Anal.*, 17, 567–576 (1980)
5 Wriggers, P. and Simo, J. C. A general procedure for the direct computation of turning and bifurcation points, *Int. J. Num. Meth. Eng.*, 30, 155–176 (1990)
6 Shilkrut, D. Investigation of axisymmetric deformation of geometrically nonlinear, rotationally, orthotropic, circular plates, *Int. J. Non-Linear Mech.*, 18, 95–118 (1983)
7 Riks, E., Brogan, F. A. and Rankin, C. C. Numerical aspects of shell stability analysis, in *Computational Mechanics of Nonlinear Response of Shells*, (Eds W. Krätzig and E. Quate), Springer, Berlin (1990)
8 Simitses, G. J. *Dynamic Stability of Suddenly Loaded Structures*, Springer, New York, Berlin; Heidelberg (1990)
9 Budiansky, B. and Roth, E. S. Axisymmetric dynamic buckling of clamped shallow spherical shells, *Collected Papers on Instability of Shell Structures, NASA TND-1510* (1962)
10 Kleiber, M., Kotula, W. and Saran, M. Numerical analysis of dynamic quasi-bifurcation, *Eng. Comput.*, 4, 48–52 (1987)
11 Krätzig, W. B. Eine einheitliche statische und dynamische Stabilitätstheorie für Pfadverfolgungsalgorithmen in der numerischen Festkörpermechanik, *Z. Angew. Math. Mech.*, 69, 203–207 (1989)
12 Burmeister, A. and Ramm, E. Dynamic stability analysis of shell structures, in *Computational Mechanics of Nonlinear Response of Shells*, (Eds W. Krätzig and E. Onate), Springer, Berlin (1990)
13 Hughes, T. R. J. *The Finite Element Method*, Prentice Hall, Englewood Cliffs, NJ (1987)
14 Schwartz, H.-R. The eigenvalue problem $(A - \lambda B)x = 0$ for symmetric matrices of high order, *Comp. Meth. Appl. Mech. Eng.*, 3, 11–28 (1974)
15 Carstensen, C. and Wriggers, P. On pertubations in non-linear dynamics, *Commun. Appl. Mech.* in press
16 Zienkiewicz, O. C. and Taylor, R. L. *The Finite Element Method*, Vol. I, McGraw-Hill, London (1989)