Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 192 (2003) 375–394
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A partial velocity approach to subcycling structural dynamics

W.J.T. Daniel

Department of Mechanical Engineering, University of Queensland, St. Lucia Campus, 4072, Qld., Australia

Received 26 June 2002; received in revised form 26 June 2002

Abstract

Subcycling, or the use of different timesteps at different nodes, can be an effective way of improving the computational efficiency of explicit transient dynamic structural solutions. The method that has been most widely adopted uses a nodal partition, extending the central difference method, in which small timestep updates are performed interpolating on the displacement at neighbouring large timestep nodes. This approach leads to narrow bands of unstable timesteps or "statistical stability". It also can be in error due to lack of momentum conservation on the timestep interface. The author has previously proposed energy conserving algorithms that avoid the first problem of statistical stability. However, these sacrifice accuracy to achieve stability. An approach to conserve momentum on an element interface by adding partial velocities is considered here. Applied to extend the central difference method, this approach is simple, and has accuracy advantages. The method can be programmed by summing impulses of internal forces, evaluated using local element timesteps, in order to predict a velocity change at a node. However, it is still only statistically stable, so an adaptive timestep size is needed to monitor accuracy and to be adjusted if necessary. By replacing the central difference method with the explicit generalized alpha method, it is possible to gain stability by dissipating the high frequency response that leads to stability problems. However, coding the algorithm is less elegant, as the response depends on previous partial accelerations. Extension to implicit integration, is shown to be impractical due to the neglect of remote effects of internal forces acting across a timestep interface.

Keywords: Transient dynamics; Explicit integration; Subcycling; Structural dynamics

1. Introduction

Subcycling is the use of different timestep sizes to integrate different degrees of freedom in a model. This results in a spatial adaption of timestep size, as opposed to the more common temporal adaptivity. Its use arises naturally in finite element modelling of interface problems, where there are strikingly different time constants or natural frequencies either side of the interface modelled, such as in structural/acoustic interaction. Subcycling has also been used with explicit integration of structural dynamics where there is no

E-mail address: billd@mailbox.uq.edu.au (W.J.T. Daniel).

0045-7825/03/$ - see front matter
PII: S0045-7825(02)00518-2

interface, to circumvent the need for a very small timestep size at every node, dictated by some small or stiff elements. The simplest situation that can be considered is that of a major timestep, which is a multiple of a minor timestep. A partition between timesteps can take either the form of a nodal or element partition. Extension of the central difference method with an accurate nodal partition, in which the large timestep node is held at constant mid-point velocity, was considered, along with other less-effective options, in Belytschko et al. [1]. This algorithm is discussed further, with examples of its use in Neal and Belytschko [2]. Daniel [3] showed that this algorithm is in fact not stable in a classical sense, in the absence of any energy dissipation. Narrow timestep ranges are unstable, due to the nonlinearity of switching between the whole model updated once per major cycle, and the small timestep zone updated in minor cycles. As the model size increases, these unstable timestep ranges become extremely narrow, such that unstable states are very unlikely to be encountered. This situation has been labelled "statistical stability". The Belytschko et al. algorithm also has a second problem of possible inaccuracy due to a lack of momentum conservation at a timestep interface. This can occur due to the large timestep update only sampling the state at neighbouring small timestep nodes once per major cycle. This problem will be discussed further later. These problems are shared by related algorithms which use a nodal interface and update the large timestep side first, such as that due to Tamma and D'Costa [4].

Other algorithms for structural dynamics have been published, such as the constant acceleration algorithm of Belytschko and Lu [5], which is proved in Ref. [6] to be unstable, and a modified version of this algorithm, by Daniel [7], devised to avoid this problem, which is statistically stable.

Smolinski [8,9] and Daniel [10,11] have proposed energy conserving algorithms that achieve stable multi-timestep extensions of the central difference method, avoiding the statistical stability problem, but not the momentum conservation problem. The algorithms in Refs. [9] and [11] make use of mid-point estimation on either side of a nodal interface, in a leap-frog fashion. This can be accurate in an elastic case, but can sacrifice accuracy to achieve stability, if the material at the timestep interface does not deform elastically. Recently, Gravouil [12,13] has presented a relatively complicated method of subcycling the implicit or explicit Newmark algorithm using Lagrange multipliers to enforce an appropriate constraint on an element interface. Gravouil's timestep interface can be proved to be stable, but dissipative. The dissipation in the timestep interface is quite significant, especially in an implicit case.

The present paper presents an alternative approach to subcycling structural dynamics problems, using an element interface, that takes account of the impulse each subcycle acting on the timestep interface in making the major timestep update there. Partial accelerations and velocities are found at nodes on a timestep interface from the sub-problems with a free timestep interface, using differing element timesteps, then partial velocities are added. The approach is similar to that adopted by Smolinski [17] to subcycling first order diffusion problems. The partial velocity approach is first applied to the central difference method, where it leads to an elegant and accurate algorithm without dissipation in the timestep interface, that is still only statistically stable, but possesses a higher probability of stability than the Belytschko, Yen, Mullen algorithm. By limiting the timestep change at nodes of any one element to a factor of two, and assigning timesteps cautiously, a timestep limit minimizing such statistical stability problems can be set. Extension of this approach to the explicit version of the generalized alpha algorithm is then considered. The high frequency energy dissipation of this algorithm is found to convert the statistical chance of instability, into a statistical chance of inaccuracy, due to damped, but spurious oscillation. A key assumption of the approach, is that internal forces acting across the element interface between partitions integrated with different timesteps, only effect the nodes on which they act, within one minor timestep. This is shown to exclude the practical use of an implicit version of a partial velocity subcycling algorithm.

The statistical stability of the algorithm is not considered a major problem, as with use of adaptive timestepping, the unlikely event of an inaccurate or unstable choice of timestep can be detected, and the timestep adjusted to maintain accuracy. An appropriate approach to adapting the timestep size of the central difference method is presented by Noels [15,16].

2. Subcycling the central difference method

2.1. The partial velocity algorithm

Consider first an element partition into two sets of elements, S elements with the minor timestep Δt and L elements with major timestep $n\Delta t$. The subcycle states will be numbered starting from zero at the start of the major cycle. A new mid-step partial velocity $\mathbf{v}^L_{n/2}$ is first found for the L partition from the resultant internal force on L elements $\mathbf{F}^L_{\mathrm{INT}}$ and external forces on L element nodes $\mathbf{F}^L_{\mathrm{EXT}}$, treating the element interface between timestep regions as a free edge, and from forces $\mathbf{F}^I$ acting across the timestep interface. Vectors with the superscript L can be thought of as being of dimension matching the total number of degrees of freedom, but containing zero entries for nodes in the S partition that are not on the time step interface. Similarly the superscript S refers to all small timestep nodes, including those on the timestep interface, which are both S and L nodes. $\mathbf{M}$ refers to the total mass at all nodes, including contributions from S and L elements.

$$\mathbf{a}^L_0 = \mathbf{M}^{-1}(\mathbf{F}^L_{\mathrm{EXT}\,0} - \mathbf{F}^L_{\mathrm{INT}\,0} + \mathbf{F}^I_0). \tag{1}$$

In a typical cycle the previous mid-step value $\mathbf{v}^L_{-n/2}$ is available, so

$$\mathbf{v}^L_{n/2} = \mathbf{v}^L_{-n/2} + \mathbf{a}^L_0 n\Delta t. \tag{2}$$

The S partition can now be updated, for $i = 0,\ n-1$:

$$\mathbf{a}^S_i = \mathbf{M}^{-1}(\mathbf{F}^S_{\mathrm{EXT}\,i} - \mathbf{F}^S_{\mathrm{INT}\,i} - \mathbf{F}^I_i), \tag{3}$$

$$\mathbf{v}^S_{i+1/2} = \mathbf{v}^S_{i-1/2} + \mathbf{a}^S_i \Delta t. \tag{4}$$

For interface nodes, the partial velocities can now be added to update the displacements over the minor cycle. In doing this, we assume for all i that $\mathbf{F}^I_i = \mathbf{F}^I_0$. That is, internal forces acting across the element interface between S and L elements cancel. External forces at timestep interface nodes are associated with the minor timestep partition S. Hence

$$\mathbf{u}_{i+1} = \mathbf{u}_i + (\mathbf{v}^S_{i+1/2} + \mathbf{v}^L_{n/2})\Delta t, \tag{5}$$

where the partial velocities are found by evaluating the partial accelerations in (1) and (3) ignoring the $\mathbf{F}^I$ terms. Nodes not on the timestep interface are updated using the central difference method in the normal manner.

Considered in a multi-timestep context, a computational form of the algorithm can be devised that does not actually compute and store partial accelerations or velocities. At the beginning of a master timestep, all elements and nodes are set to be at the time of the clock. The clock is then advanced through the smallest timestep in use. For any element behind the clock on a particular subcycle (all of them on this first subcycle), the impulse of internal forces on the element is found using its own element timestep Δt_{e}. These impulses are assembled at the nodes to predict changes in the velocities to be used to update displacements. That is, considering node k of element l, with nodal timestep Δt_k, being the minimum of the timesteps of elements meeting at that node, the velocity used to update displacements is modified as:

$$\mathbf{v}_k = \mathbf{v}_k + \Delta\mathbf{v}_{kl} = \mathbf{v}_k + \mathbf{M}^{-1}\{\mathbf{F}_{\mathrm{EXT}\,k}\Delta t_k - \Sigma\mathbf{F}_{\mathrm{INT}\,kl}\Delta t_{\mathrm{e}\,l}\}, \tag{6}$$

by summing over all elements currently being updated. Note that external forces are now associated with the nodal timestep Δt_k, which is that used to update displacements at the node. This is equivalent to finding a separate partial acceleration describing the influence of external loading every Δt_k. Eq. (6) is relatively easy to code into an existing transient dynamics package. In essence, impulses are assembled to predict velocity changes, rather than forces being assembled to predict accelerations. In general, in an algorithm

where temporal adaptivity is present as well as subcycling, the element timestep Δt_e can be split into old and new values: i.e. $(\Delta t_{e\,OLD} + \Delta t_{e\,NEW})/2$. Such a change can be made at the start of any major cycle. In this clock-driven implementation, in which any elements with times behind the clock are updated, the element timestep Δt_{e1} need not be an integer multiple of the nodal timestep Δt_k.

A further refinement of this approach is needed if rotational degrees of freedom associated with moments of inertia are present. Impulses due to gyroscopic inertia terms can be included. For instance the angular velocity about the x axis, which is a principal axis of inertia in the nodal system at a node k, can be updated as

$$\omega_{k\,x} = \omega_{k\,x} + 1/I_{xx}(M_{EXT\,k\,x}\Delta t_k - \Sigma M_{INT\,k\,x}\Delta t_e - (I_{yy} - I_{zz})\omega_{k\,y}\omega_{k\,z}\Delta t_k). \tag{7}$$

A cautious approach to assigning timesteps to elements and nodes adopted here is the following. First element timesteps are estimated, then:

(a) the timestep at each node is made the minimum of the timesteps used at attached elements;
(b) the timestep of each element is made the minimum of its nodal timesteps;
(c) the nodal timesteps are reassigned to be the minimum of timesteps used at attached elements.

This procedure has the effect of making an increase in element timestep lag spatially an increase in element size, enhancing stability.

2.2. Stability analysis

The above algorithm has statistical stability properties similar to those of the Belytschko et al. algorithm discussed in Ref. [3]. Consider a linear elastic homogeneous dual timestep case, so $\mathbf{F}^S_{INT} = \mathbf{K}^S\mathbf{u}$, $\mathbf{F}^L_{INT} = \mathbf{K}^L\mathbf{u}$ (neglecting internal forces acting across the timestep interface which cancell) and $\mathbf{F}_{EXT} = \mathbf{0}$. Write $\mathbf{M}^{-1}\mathbf{K}^S\Delta t^2$ as A and $\mathbf{M}^{-1}\mathbf{K}^L\,(n\Delta t)^2$ as B. The first minor cycle involves an update of both small and large timestep partitions. Eliminating the velocities, and dropping the use of bold font, (1)–(5) can be written as

$$\begin{bmatrix} u_1 \\ u_0 \end{bmatrix} = \begin{bmatrix} 2I - A - (1/n)B & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ u_{-1} \end{bmatrix}. \tag{8}$$

Subsequent subcycles, involving updates of the small timestep partition only, can be written as

$$\begin{bmatrix} u_{i+1} \\ u_i \end{bmatrix} = \begin{bmatrix} 2I - A & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} u_i \\ u_{i-1} \end{bmatrix}. \tag{9}$$

The resultant amplification matrix can hence be found for any number of subcycles. In the case of a non-integer timestep ratio, an update of the L partition only can occur. This is described by replacing $2I - A$ in (9) with $2I - (1/n)B$, where the timestep ratio n is not an integer.

Consider a model problem consisting of a single interface degree of freedom. The first update is a function of $A + (1/n)B = (\omega_{ns}\Delta t)^2 + n(\omega_{nl}\Delta t)^2 = \Omega^2_{SL}$, where ω_{ns} and ω_{nl} are the natural frequencies of the S and L element partitions, with the interface free. Subsequent subcycle updates depend on a Courant number $\Omega_S = \omega_{ns}\Delta t$. For instance, in the two subcycle case, the eigenvalues of the major cycle amplification matrix found by combining (8) and (9) are given by

$$\lambda = (1 - \Omega^2_S - \Omega^2_{SL} + \Omega^2_S\Omega^2_{SL}/2) \pm 1/2((\Omega^2_{SL} - 2)(\Omega^2_S - 2)(\Omega^2_S\Omega^2_{SL} - 2(\Omega^2_S + \Omega^2_{SL})))^{1/2}, \tag{10}$$

λ becomes of magnitude greater than one in the timestep range between $\Omega_{SL} = \sqrt{2}$ and $\Omega_S = \sqrt{2}$, causing instability below the expected limits of $\Omega = 2$. This is similar to the behaviour of the Belytschko et al. algorithm reported in Ref. [3]. In the multi degree of freedom two-subcycle case, a Courant number $\Omega_{SL} = \sqrt{2}$, which corresponds to the natural frequency ω_{SL} associated with the highest mode of the modified

eigenvalue problem in (11) below, gives the lowest timestep at which instability can occur, with a single timestep interface.

$$(\mathbf{K}^S + n\mathbf{K}^L)\mathbf{u} = \omega_{SL}^2 \mathbf{M}\mathbf{u}. \tag{11}$$

The factor n on $\mathbf{K}^L$ can be regarded as arising from the fact that the large timestep partition is effectively accelerated to its new velocity in one subcycle, rather than taking a major cycle.

The stability limit can be demonstrated by multiplying together the amplification matrices in (8) and (9) to give

$$\begin{bmatrix} u_2 \\ u_1 \end{bmatrix} = \begin{bmatrix} (2I - A)(2I - (A + (1/n)B)) - I & -(2I - A) \\ 2I - (A + (1/n)B) & -I \end{bmatrix} \begin{bmatrix} u_0 \\ u_{-1} \end{bmatrix}. \tag{12}$$

The displacements u are regarded as linear combinations of the modes of Eq. (11) so that $[A + (1/n)B]u_0 = \omega_{SL}^2 \Delta t^2 u_0$ with u_0 being one such mode. Let $\omega_{SL}^2 \Delta t^2$ be a maximum of 2 corresponding the u_0 being the highest mode of Eq. (11), then (12) becomes

$$\begin{bmatrix} u_2 \\ u_1 \end{bmatrix} = \begin{bmatrix} -I & -(2I - A) \\ 0 & -I \end{bmatrix} \begin{bmatrix} u_0 \\ u_{-1} \end{bmatrix}. \tag{13}$$

The amplification matrix in Eq. (13) clearly has an eigenvalue $\lambda = -1$, which corresponds to $\Delta t = \sqrt{2}/\omega_{SL}$. This real value indicates the onset of instability, as it does in the parent algorithm. The same condition can be achieved in (12) when $A = 2I$ corresponding to a timestep $\Delta t = 2/\omega_S$ associated with the highest S partition natural frequency, but this is a slightly larger timestep due to the lack of the $(1/n)B$ term in (13), corresponding to the $n\mathbf{K}^L$ term present in (11).

As the problem size increases, regions of instability are still functions of the Courant numbers Ω_S and Ω_{SL} $(= \omega_{SL}\Delta t)$ which become very similar for the higher modes. With many degrees of freedom present, the highest modes of the S partition involve localized motion of a few degrees of freedom, and are not greatly affected by the additional interaction with the L partition in (11). Hence the instability ranges that occur correspond to very narrow ranges of timesteps, and can be avoided with a small change in timestep. The same effect occurs with increasing numbers of subcycles. For instance, the stability of the single degree of freedom spring/mass system of Fig. 1(a) with 10 subcycles is plotted on Ω_S, Ω_L axes in Fig. 2, where Ω_L is $\omega_{\mathrm{nl}} n \Delta t$. Narrow timestep ranges are unstable, associated with particular Ω_S values, unstable states being plotted as a value of one and stable states as zero. The equivalent plot for the Belytschko et al. nodal interface algorithm applied to the two degree of freedom model problem of Fig. 1(b) is also shown, and can be seen to have broader unstable timestep ranges. The instabilities arise from the non-linearity of switching between an update involving the dynamic response of the whole model, and updates involving only the dynamics of the small timestep partition, causing the stiffness of the large timestep partition $n\mathbf{K}^L$ to appear and disappear.

Where more than two timestep sizes are used, the above serves as a guide, but in the absence of energy dissipation in a model, narrow ranges of unstable timesteps can occur at lower timesteps. Consider using timesteps in the ratios 1:2:4. Three different structural eigenvalue problems characterize the updates occuring.

For the first update of a major cycle (all elements updated): $(\mathbf{K}_1 + 2\mathbf{K}_2 + 4\mathbf{K}_3)\mathbf{u} = \omega_1^2 \mathbf{M}\mathbf{u}$. For the second or fourth update (partition 1 only): $\mathbf{K}_1\mathbf{u} = \omega_2^2 \mathbf{M}\mathbf{u}$. For the third update (partitions 1 and 2): $(\mathbf{K}_1 + 2\mathbf{K}_2)\mathbf{u} = \omega_3^2 \mathbf{M}\mathbf{u}$.

An amplification matrix describing a major cycle can be written by multiplying together amplification matrices like those in Eqs. (8) and (9), that is

$$\begin{bmatrix} u_4 \\ u_3 \end{bmatrix} = \begin{bmatrix} 2I - A_2 & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} 2I - A_3 & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} 2I - A_2 & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} 2I - A_1 & -I \\ I & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ u_{-1} \end{bmatrix}, \tag{14}$$

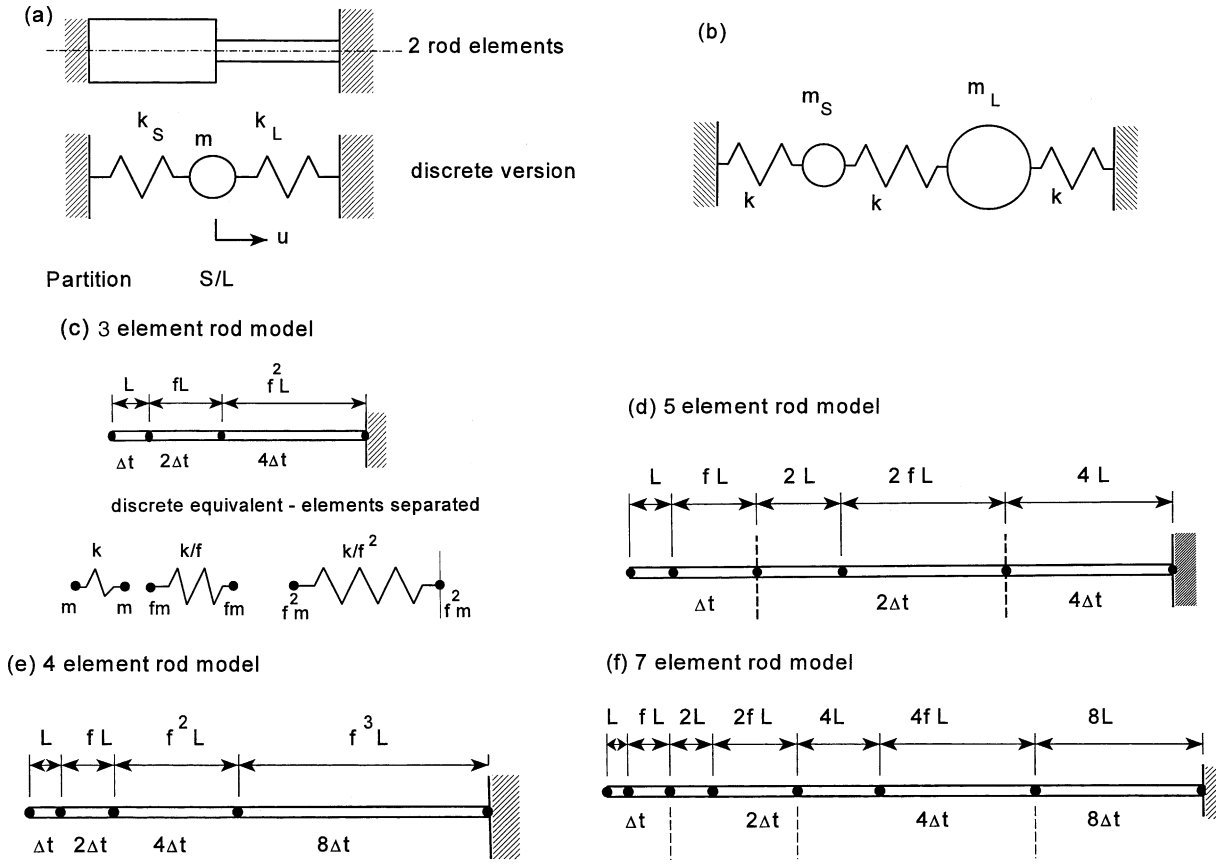


Fig. 1. One-dimensional model problems used to study comparisons between algorithms.

where $A_1 = \mathbf{M}^{-1}(\mathbf{K}_1 + 2\mathbf{K}_2 + 4\mathbf{K}_3)$, $A_2 = \mathbf{M}^{-1}\mathbf{K}_1$ and $A_3 = \mathbf{M}^{-1}(\mathbf{K}_1 + 2\mathbf{K}_2)$. A simple example studied is that of Fig. 1(c), a three degree of freedom system, consisting of three rod elements, each with its own timestep, in ratios of 2. The rods have the same properties, but different lengths. With the factor on increase in element length, f, set to 2, each element has the same maximum Courant number $\Omega = \omega_{n\,\mathrm{MAX}}\Delta t_e$, the maximum natural frequency of the element, $\omega_{n\,\mathrm{MAX}}$ being found with its connections free. Suppose single element estimates of the limits on the timesteps are made, so that the estimated Ω is $\sqrt{2}$ in each partition. The maximum timestep estimated for element 1 is that giving $\Omega_1 = 1$, where, using the symbols on the discrete system interpretation of the system in Fig. 1(c), $\Omega_1^2 = (\mathrm{k/m})\Delta t^2$. With the left-most node fixed, the first instability occurs above this estimated limit. However, with the left-most node free, numerical eigenvalue analysis of an amplification matrix for a major cycle reveals there is a very narrow range of unstable timesteps from $\Omega_1 = 0.65045$ to 0.65051. This corresponds to bifurcation of a pair of complex conjugate eigenvalues of the amplification matrix to give real values, although instability can also occur due a complex eigenvalue of modulus exceeding one. The larger real eigenvalue of the amplification matrix is -1.0002 at this Ω, causing relatively slow growth of a spurious oscillation in the solution under appropriate initial conditions, which would be easily damped out if energy dissipation were present in the model. If a more cautious assignment of timesteps is made, according to the rules given above, so the timestep increase lags the increase in element size by one element, a more stable situation results, in that the unstable timestep ranges become extremely narrow, and hard to detect. This is illustrated by the five degree of freedom model

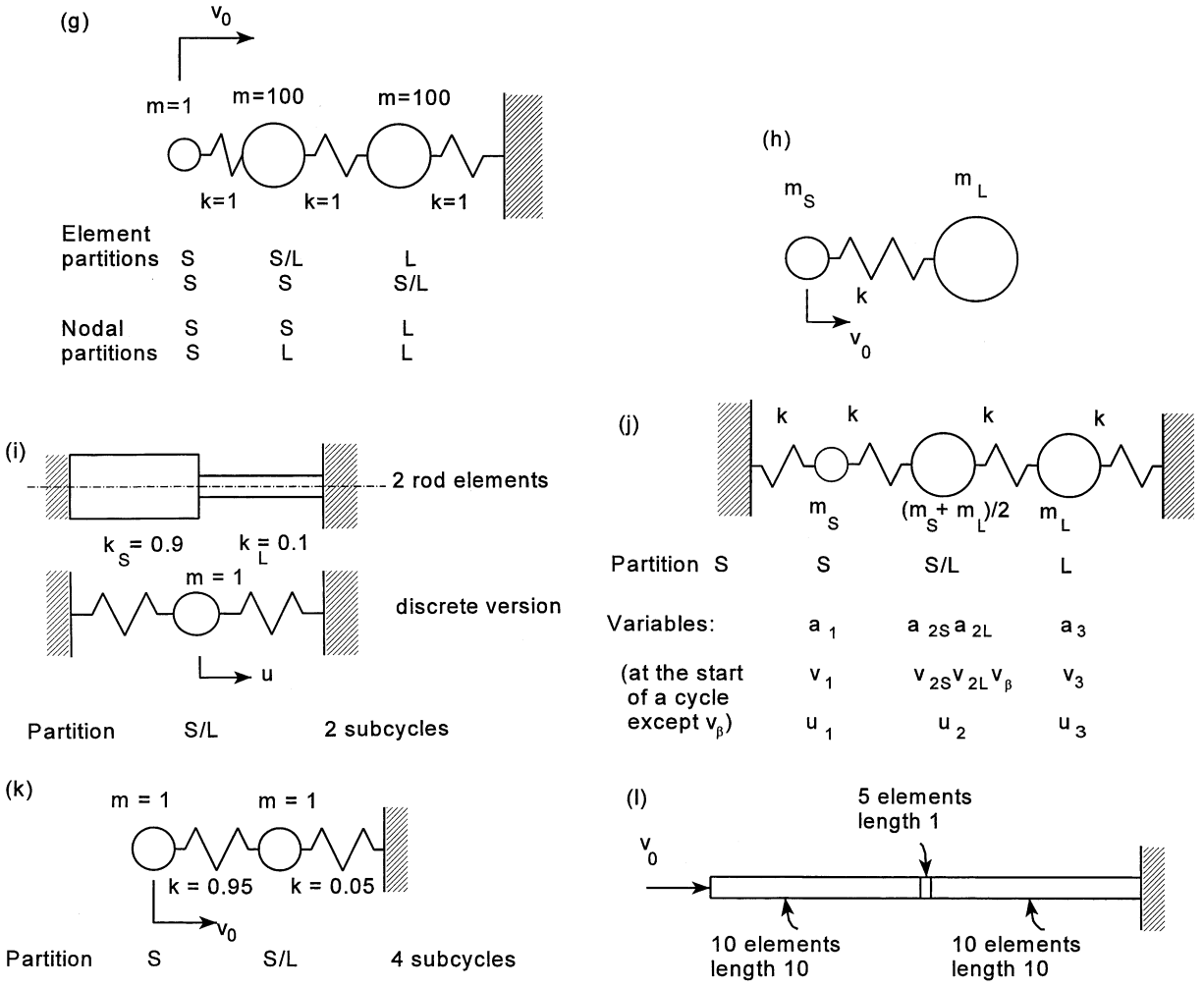


Fig. 1 (*continued*)

integrated with three timesteps of Fig. 1(d). With $f = 2$, unstable timesteps were first detected numerically by solving for the eigenvalues of a major cycle amplification matrix, at $\Omega_1 = 1.2014$, above the estimate of 1. This does not mean that lower unstable timesteps do not exist, but a finer timestep increment is required to detect them than the increment in Ω_1 of 0.0001 used. If the parameter f is varied, using the model of Fig. 1(d), this narrowing of unstable timestep ranges as f is increased can be demonstrated. Table 1 shows the changes with f in the size of one of the ranges of unstable timesteps that occur.

If another timestep size is added, giving the four timestep model of Fig. 1(e), where each element has its own timestep, instability occurs at still lower timesteps, the first unstable range of Ω_1 detected being 0.55413–0.55415. However, with the more cautious assignment of timesteps in Fig. 1(f), instability is hard to detect until $\Omega_1 = 1.2$, again above the single element estimate, and similar to the value expected from the highest natural frequency of the small timestep partition, ω_S.

Despite the fact that instability becomes unlikely in a realistic model, and can be avoided by adaptive timestepping, it is desirable to make it even more unlikely by grading of timestep changes and cautious

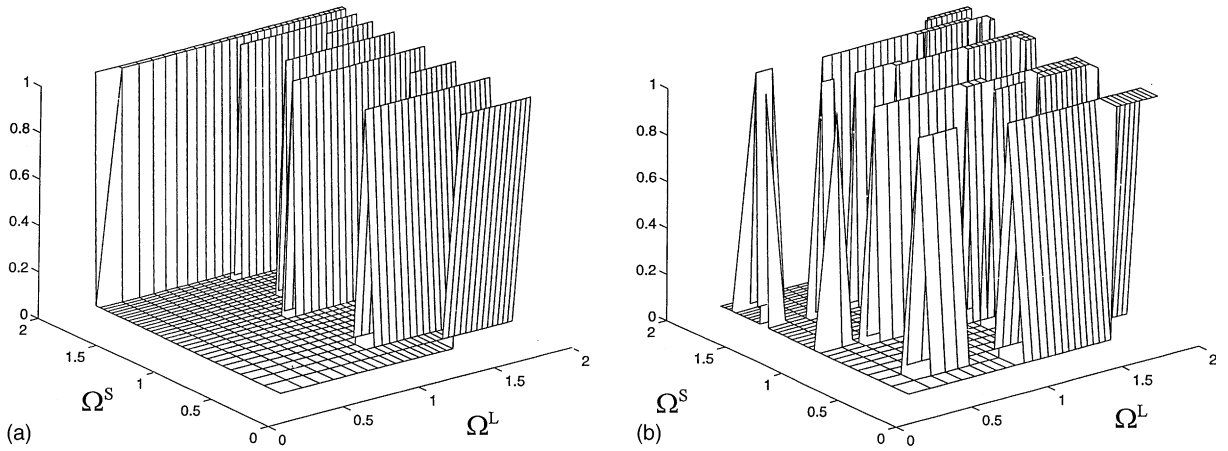


Fig. 2. Unstable timestep ranges for the partial velocity algorithm applied to the model problem of Fig. 1(a) and the Beytschko et al. algorithm applied to the model problem of Fig. 1(b), using 10 subcycles, 0 = stable, 1 = unstable. (a) Partial velocity extension of the central diference algorithm and (b) Belytschko et al. algorithm.

Table 1
Narrowing of an unstable timestep range for the problem of Fig. 1(b) with increasing length ratio f

Length ratio f	Unstable range of $\Omega_1^2 = (k/m)\Delta t^2$ (i.e. $\Delta\Omega_1^2$)
1	9×10^{-5}
1.13	7×10^{-5}
1.26	6×10^{-5}
1.3	5×10^{-5}
1.5	3×10^{-5}
2	2×10^{-5}

allocation of timesteps, and by having some dissipation of high frequencies present, to prevent these instabilities. This could be done by applying partial velocity subcycling to the "self-starting" algorithm of Tamma and D'Costa [4]. In this algorithm, new velocities are first found and then dissipation is introduced when displacements are updated, by making the update using a weighting $\mathbf{v}_\gamma$ of velocities at the start and end of the current timestep i

$$\mathbf{v}_\gamma = \gamma \mathbf{v}_{i+1} + (1-\gamma)\mathbf{v}_i \quad \text{where } \gamma \geqslant 0.5.$$

To generalise the algorithm, it is appropriate to use velocities $\mathbf{v}_\gamma$ when computing partial velocities to be added at a timestep interface node. The result is statistically stable with $\gamma = 0.5$, and similar to a subcycled central difference method. The γ parameter enables dissipation to be introduced by making $\gamma > 0.5$. The weakness of this approach is that too much dissipation occurs at lower frequencies. An effective means of adding dissipation only of the highest frequencies is to subcycle the explicit generalised alpha method as discussed in Section 3.

2.3. A cautious approach to assigning timestep sizes

Consider a partitioning where any two nodes of one element may only differ in their timesteps by a factor of 2. For n timestep regions, the eigenvalue problem describing the effective structural problem on the first subcycle when the whole model is updated is

$$(\mathbf{K}_1 + 2\mathbf{K}_2 + 4\mathbf{K}_3 + \cdots + 2^n\mathbf{K}_n)\mathbf{u} = \omega^2\mathbf{M}\mathbf{u}. \tag{15}$$

Here $\mathbf{K}_i$ is the stiffness matrix of element partition i with timestep interfaces free. We wish to estimate ω_{MAX} of this problem to set the smallest timestep, to minimize the chance of any statistical stability problems. Typically, with timesteps assigned using the rules above, a single element estimate ω_{EST} of the largest natural frequency of partition 1 is a good estimate of ω_{MAX}. Hence to be cautious, we could set ω_{MAX} equal to $\sqrt{2}_{\mathrm{EST}}$, so that half the local estimated stability limit of the central difference method is used locally. This setting helps maintain accuracy. An adaptive timestepping scheme, like that of Noels [15] could refine this estimate, if necessary.

2.4. Accuracy of partial velocity subcycling of the central difference method

Comparison of the accuracy of the algorithm of Eqs. (1)–(5) with that of the Beltytschko et al. algorithm cannot be done precisely, as a nodal interface is being compared to an element interface. A comparison of the convergence of the algorithms on one of the simplest possible dynamic systems that can be represented by both algorithms is shown in Fig. 3. The three mass system of Fig. 1(g) is subcycled using four subcycles and the nodal or element partitions marked on the figure. Convergence of the subcycled solution to the small timestep solution is shown, using an L2 norm of the difference in displacement vectors, evaluated over 1000 major cycles. If two small timestep masses are used with the Belytschko et al. algorithm it is a little more accurate than the partial velocity method which uses both timesteps at the centre mass. However, if two large timestep masses are used with the Belytschko, Yen, Mullen algorithm, then the partial velocity solution is significantly more accurate. Note both methods show noise in the convergence at large timesteps, reflecting the statistical stability of the algorithms. If a cautious approach to an element partition is used (i.e. S,S,S/L—both timesteps used at the third node), then the accuracy is a little better than that of the (S,S,L) nodal partition, and the convergence graph is smooth at large timesteps, due to unstable timestep ranges becoming too narrow to appear on the plot.

The summation over subcycle states involved in updating the displacement of nodes on the timestep interface ensures a momentum balance on the timestep interface, as expressed in (6). This means that rigid body motion is captured correctly, even in the two mass free–free system of Fig. 1(h). By contrast, the

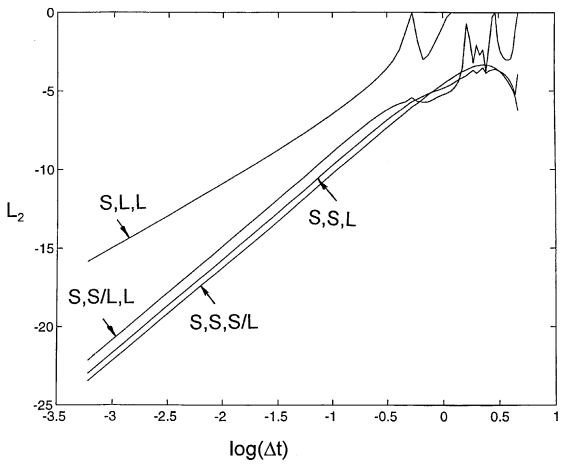
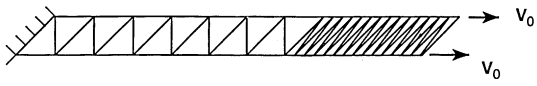


Fig. 3. Convergence of the subcycled solution to the small timestep solution, on the three mass problem of Fig. 1(g), using the central difference method. (S,S/L,L) and (S,S,S/L) partial velocity partitions. (S,S,L) Belytschko et al. nodal partition—two small timestep nodes. (S,L,L) Belytschko et al. approach—one small timestep node.

Belytschko et al. algorithm applied to this problem, given an initial velocity as shown, can lead to a rigid body component of motion in the wrong direction. This is due to the major timestep update only sampling the small timestep state once per major cycle. That is, the small timestep mass can bounce off the large timestep mass without it noticing the impact. It should be noted that this problem can be avoided simply by making the nodal timesteps to be used in this nodal partition, the minimum timestep of those estimated for elements at a node. Hence if the two-mass problem is thought of as a linear rod element with a lumped mass attached, subcycling would not be permitted under this rule. However, larger problems can also show error due to lack of momentum conservation at a timestep interface. An example is the row of triangular elements loaded in plane by an initial velocity on one end as shown in Fig. 4(a). The material is elasto-plastic with the properties shown, which leads to yielding at a stress well below that of an elastic stress wave with the initial velocity applied. The minor timestep is half the critical timestep, and four subcycles are used, with the timestep interface at the change in mesh refinement. Using the partial velocity method, the subcycled displacement of the loaded end, superimposes on the small timestep solution. However the Belytschko et al. algorithm significantly underestimates the response, as can be seen on Fig. 5. Even with a purely elastic response, the partial velocity algorithm gives a significant improvement in accuracy when tested on this problem. If the timestep interface is shifted by one element into the more coarsely meshed region, as re-



Properties:		
	Young's modulus	1.2×10^4
	Tangent modulus	1.0×10^2
	Density	1.024×10^{-6}
	Poisson's ratio	0.22
	Thickness	0.1
	Yield strength	5

Fig. 4. Plane stress mesh in tension due to an initial velocity—linear triangles used.

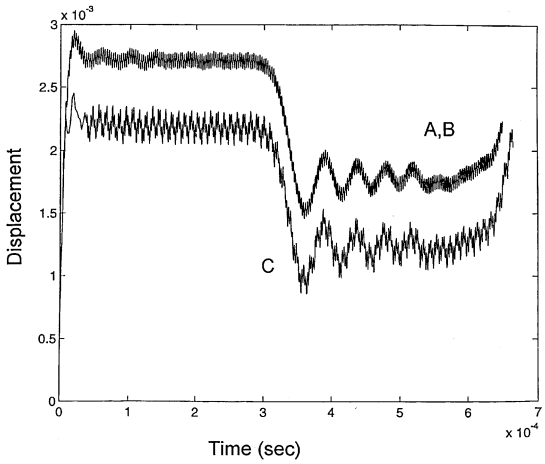


Fig. 5. Response at the right end of the mesh of Fig. 4 to an initial velocity: (A) small timestep solution, (B) solution subcycled using the partial velocity method and (C) solution subcycled using the Belytschko et al. algorithm.

quired by the rules for allocating timesteps given above, then the difference between algorithms is much less dramatic.

Good results can be obtained with the present algorithm on elastic problems, especially if flexural response, dominated by low frequencies is involved. These results are academic as modal analysis would probably be used in such cases rather than explicit integration (unless other non-linearities such as contact-separation are involved). It is of more interest to test how well the algorithm captures plastic deformation. A realistic trial problem using triangular shell elements is shown in Fig. 6. The pipe is given an initial velocity at A of 10 m s^{-1} radially inward, causing a stress wave of 390 MPa in the small cylinder, just below the yield strength of 400 MPa, and causing plastic deformation in the large cylinder. A subcycled solution is an effective option, due to the range of element sizes present. Element timesteps are assigned based on single element estimates of stability, in the ratios 1:2:4:8:16, the smallest being 0.48 of the local critical timestep. The response over 1008 minor cycles of the radial displacement at A is shown in Fig. 7. It can be seen to be quite similar to the small timestep solution.

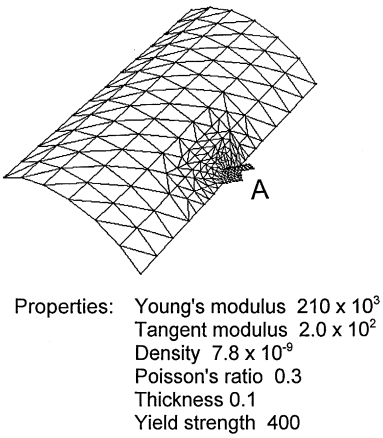


Fig. 6. A quarter model of a cylinder with a pipe connected, using discrete Kirchoff triangles. Initial velocity applied at A.

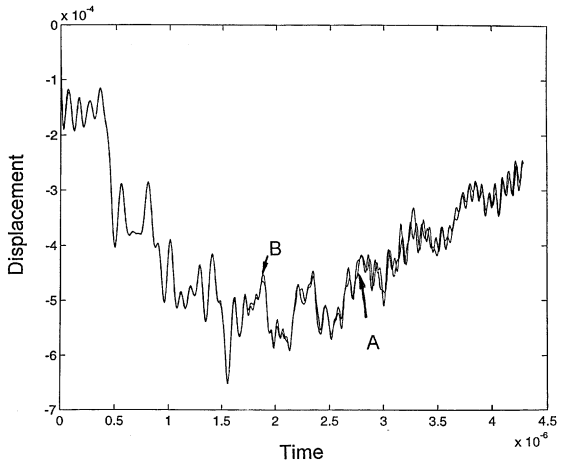


Fig. 7. Radial motion of the pipe at A of Fig. 6: (A) not subcycled—central difference solution and (B) subcycled—partial velocity method with timesteps in multiples of 2.

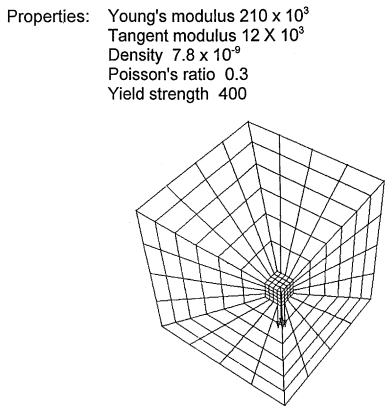


Fig. 8. A trial problem using linear hexahedral solid elements.

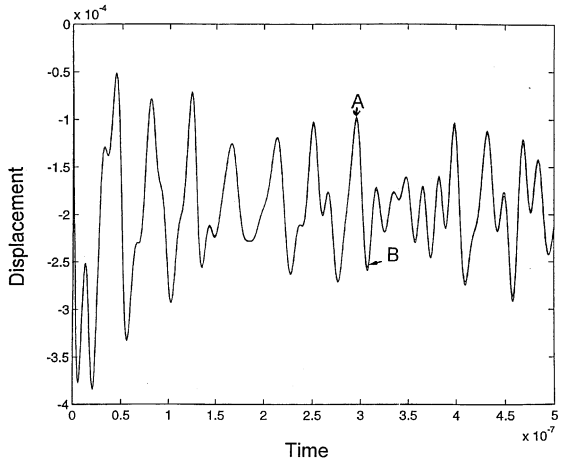


Fig. 9. Accuracy of the partial velocity method on the response to an initial velocity of the model of Fig. 8: (A) not subcycled—central difference solution. (B) subcycled—partial velocity method with timesteps in multiples of 2.

A trial problem using hexahedral linear solid elements is shown in Fig. 8. Faces A and B are planes of symmetry. Faces C and D are fixed. The arrows represent initial velocities applied to the nodes shown. Again, the material is elasto-plastic, with the properties on the figure, and the initial velocities are high enough to cause substantial plastic deformation. Element timesteps are chosen as 0.48 the local critical timestep and sorted into multiples of 2 as in the previous example. The accuracy achieved with this strategy is illustrated in Fig. 9, which plots the deformation at the corner node impacted. Slight differences in the solution are visible only towards the end of the simulation time of 706 minor timesteps.

To develop a robust code based upon this algorithm and to ensure accuracy, an adaptive approach is needed, based on a measure of error due to integration like that of Noels [15]. In addition, the energy of the solution can be monitored (e.g. each major cycle) in order to monitor stability and change the timestep size if necessary.

3. Subcycling the explicit generalized alpha method

3.1. The partial velocity algorithm

The generalized alpha method was developed for implicit time integration by Chung and Hulbert [18] to dissipate high frequencies, which are artefacts of the discretization of a finite element model. The method uses two 'alpha' parameters to generalize the Newmark algorithm. The same authors extended their approach to explicit integration in Ref. [14]. The explicit algorithm uses a single parameter α to control high frequency dissipation, while avoiding filtering of intermediate frequencies. α can be expressed in terms of a parameter ρ_b measuring the spectral radius at the bifurcation point, at which maximum dissipation is obtained. The Newmark parameters β and γ used to weight accelerations also depend on ρ_b.

$$\alpha = (2\rho_b - 1)/(\rho_b + 1), \tag{16}$$

$$\gamma = 3/2 - \alpha, \tag{17}$$

$$\beta = (5 - 3\rho_b)/\{(1 + \rho_b)^2(2 - \rho_b)\}. \tag{18}$$

If $\rho_b = 1$ then the energy conserving mid-point algorithm is obtained, with the stability limit of the central difference method. If $\rho_b = 0$ then the highest frequency that can be represented is annihilated in one cycle, the maximum timestep that can be used being that at the bifurcation point, which is $(1/\sqrt{2})\ \Delta t_{\mathrm{CRIT}}$ of the central difference method.

The explicit generalized alpha method can be subcycled using a partial velocity approach.

The appropriate velocities to add, at timestep interface nodes, are those associated with the parameter β of the Newmark algorithm, as these velocities are used to update displacement in the parent algorithm. E.g. for the S partition of a two timestep case.

$$\mathbf{v}^S_\beta = \mathbf{v}^S_i + (\beta \mathbf{a}^S_{i+1} + (1/2 - \beta)\mathbf{a}^S_i)\Delta t. \tag{19}$$

Thus the displacement update for subcycle i becomes

$$\mathbf{u}_{i+1} = \mathbf{u}_i + (\mathbf{v}^S_\beta + \mathbf{v}^L_\beta)\Delta t, \tag{20}$$

where the velocity held constant is

$$\mathbf{v}^L_\beta = \mathbf{v}^L_0 + (\beta \mathbf{a}^L_n + (1/2 - \beta)\mathbf{a}^L_0)n\Delta t. \tag{21}$$

Partial velocities either side of the timestep interface are updated as per the Newmark algorithm e.g.

$$\mathbf{v}^S_{i+1} = \mathbf{v}^S_i + (\gamma \mathbf{a}^S_{i+1} + (1 - \gamma)\mathbf{a}^S_i)\Delta t. \tag{22}$$

The new partial acceleration $\mathbf{a}^S_{i+1}$ in a subcycle is found as

$$\mathbf{a}^S_{i+1} = \{\mathbf{M}^{-1}(\mathbf{F}^S_{\mathrm{EXT}\,i} - \mathbf{F}^S_{\mathrm{INT}\,i}) - \alpha \mathbf{a}^S_i\}/(1 - \alpha). \tag{23}$$

Similar equations apply to the L partition, although external forces at timestep interface nodes are associated with the minor timestep side of the interface. To implement this updating in a multi-timestep context, for cycle i we first find for each element l, impulses $\mathbf{F}_{\mathrm{INT}}\Delta t_{l\,i}/(1 - \alpha)$ where $\Delta t_{l\,i}$ is the element timestep. Then the new partial velocity changes $\mathbf{a}_{i+1}\Delta t_{l\,i}$ at the nodes of each timestep interface element can be found using (23), but modifying the external force term, as the impulse due to external forces can again be associated

with the nodal timestep Δt_k, being the minimum of the local element timesteps, as in (6). Combining (19) and (22) velocity changes being summed over elements l being updated gives:

$$\mathbf{v}_{\beta k\, i} = \mathbf{v}_{\beta k\, i-1} + \Sigma\{\beta(\mathbf{a}_{i+1\, kl}\Delta t_{l\, i}) + ((1/2-\beta)\Delta t_{l\, i} + (\gamma-\beta)\Delta t_{l\, i-1})\mathbf{a}_{i\, kl} - (1/2+\beta-\gamma)(\mathbf{a}_{i-1\, kl}\Delta t_{l\, i-1})\}. \tag{24}$$

If $\rho_b = 1$ corresponding to $\alpha = 1/2$, $\beta = 1/2$ and $\gamma = 1$, then $\mathbf{a}_\alpha = (\mathbf{a}_i + \mathbf{a}_{i+1})/2$ is used to make the update and (24) becomes equivalent to (6), the velocity change depending only on impulse sums of internal or external forces. Otherwise, the need to use and store two previous partial accelerations (or else store the partial velocities and accelerations at the start of the previous timestep) is an extra burden of computation and of memory requirements, reflecting the memory of past states needed to achieve dissipation of high frequencies. It is possible, but extravagant, to compute and store partial accelerations of each node of every element. A more practical option is to use a nodal data structure, storing a list of acceleration components at each node, each list being of variable length determined from an integer index array. Hence a node exposed to three element timesteps would have a three entry list of acceleration components in the array of $\mathbf{a}_{i\ kl}$ values, for each coordinate direction. A counter is also necessary to track which entries in each list of accelerations have already been processed, as the list of active finite elements is processed to update accelerations and accumulate changes to $\mathbf{v}_\beta$. Elements must be processed in a predictable order, matching that of the stored values, such as in order of decreasing element timestep.

3.2. Stability and spectral properties of the subcycled explicit generalized alpha method

Eqs. (19)–(23) can be arranged to give the amplification matrices for the large timestep update or subsequent subcycles given in Appendix A These describe the single interface degree of freedom model problem of Fig. 1(a). When multiplied together to describe a major cycle, these matrices enable spectral radius to be plotted, varying the major and minor timesteps. In estimating the spectral radius, one real eigenvalue with magnitude 1 is ignored, as it corresponds to an eigenvector in which the partial velocities are the only non-zero terms and they cancel each other.

An example of the dissipative properties of the subcycled generalized alpha method is shown in Fig. 10. It applies to the particular case of a single degree of freedom problem of Fig. 1(i), the horizontal Courant

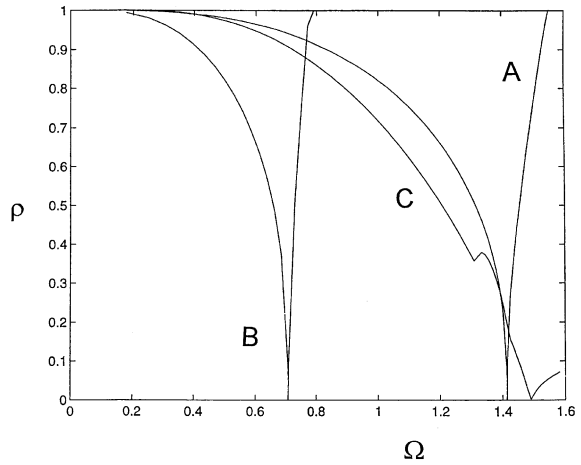


Fig. 10. Example of decay of spectral radius with increasing timestep size. Explicit generalized alpha method applied to the problem of Fig. 1(i): (A) small timestep only. (B) large timestep only and (C) subcycled solution—partial velocity method with two subcycles.

number axis being (natural frequency of the whole model) × (minor timestep). Two subcycles and $\rho_b = 0$ is used. The behaviour of the parent algorithm with small and large timesteps is shown. As the natural frequency of the small timestep partition, using the full mass of the interface node, is less than that of the full model, the stability of the subcycled solution is slightly better than that of a small timestep solution. The energy dissipation, is similar to that of the small timestep, but less precisely controlled. This disturbance in the smooth reduction in spectral radius designed into the parent algorithm increases as Ω^L increases. For the two subcycle case, this is shown in the plot of spectral radius versus Ω^S and Ω^L of Fig. 11. This figure applies to a case where $\rho_b = 0$ in the small timestep partition, but $\rho_b = 1$ in the large timestep partition. Note, in both cases, that the algorithmic energy dissipation does remove the problem of statistical stability. To achieve this, it is more important to dissipate the high frequencies in the small timestep partition than in the large timestep partition. Numerical experimentation indicates that the condition $\rho_b = 0$ is necessary in the small timestep partition to remove all unstable timestep bands. The ridge visible in Fig. 11 does eventually lead to instability, but only with $\Omega^L > \Omega^S$. From an accuracy viewpoint as well, it is best to avoid this condition. With larger numbers of subcycles, plots like Fig. 11 show less deviation from the expected variation of spectral radius with the small timestep. Closer examination of the eigenvalues of the amplification matrix shows that the disturbance to the reduction in spectral radius is associated with the existence of real, negative eigenvalues—i.e. spurious damped oscillation at the highest frequency that the major timestep can represent. The original algorithm is designed so that the real eigenvalue of the amplification matrix λ_3 does not become negative, and does not exceed the complex conjugate eigenvlaues $\lambda_{1,2}$. This is not necessarily the case with a subcycled version, using the parameters of the original algorithm. In fact, where the $\rho_b = 1$ case would have become unstable, the dissipative case shows accuracy problems, due to real eigenvalues.

Some more insight can be obtained by studying the eigenvalues of the amplification matrix for a slightly larger problem, the three mass system of Fig. 1(j). A 12×12 amplification matrix was written, using the unknowns listed on Fig. 1(j). The redundant variable $\mathbf{v}_\beta^L$ and partial acceleration unknowns were used for convenience, leading to zero eigenvalues of the amplification matrix. The use of separate partial velocity unknowns at the timestep interface leads to an real eigenvalue of magnitude 1, corresponding to an eigenvector where these velocities cancell. There are typically three pairs of complex conjugate eigenvalues, corresponding to eigenvectors reflecting approximations to the three modes of vibration of the physical system, and four real eigenvalues. The magnitude of each complex eigenvalue is plotted in Fig. 12, using the conservative "single element" estimates of the stability limits in each timestep partition on Fig. 1(j). With $\rho_b = 0$ in both partitions, the lowest mode is only slightly dissipated, its eigenvalue being just below

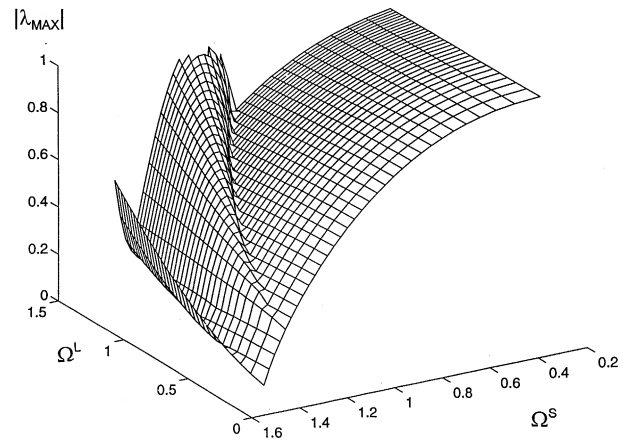


Fig. 11. Spectral radius of the single interface degree of freedom subcycled using $\rho_b^S = 0$ and $\rho_b^L = 1$.

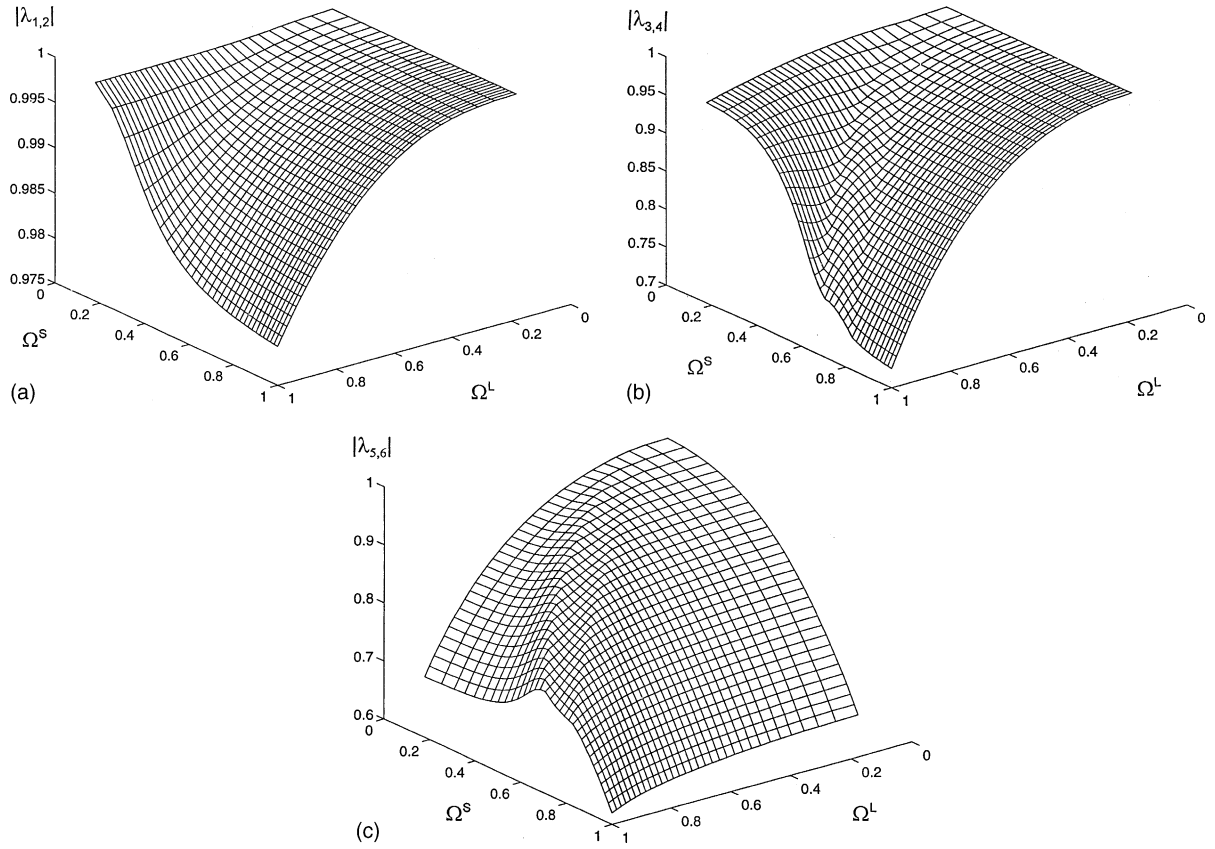


Fig. 12. Damping of each mode of the problem of Fig. 1(j), measured by the complex eigenvalues of an amplification matrix. $\rho_b^S = 0$ and $\rho_b^L = 0$. (a) Mode 1 (lowest), (b) Mode 2 and (c) Mode 3 (highest).

one in magnitude and consistently the highest present, indicating no stability problems with the timestep limits in use. The other modes are dissipated more, as expected, the case of $\rho_b = 0$ in both partitions being plotted in Fig. 12. If however, the timestep limits are increased, narrow ridges start to appear in these plots, analogous to the statistical stability problem, where an eigenvalue that is real, negative, but still less than one, occurs over a narrow range of timesteps, before unstable conditions are encountered. An example is the plot in Fig. 13 of the minimum complex eigenvalue of the amplification matrix. This generally represents damping of the highest mode, and is plotted for the case $\rho_b = 0$ in the small timestep partition and $\rho_b = 0$ in the large timestep partition. However, for a narrow range of the minor timestep, the complex eigenvalue for this mode has bifurcated, becoming two real eigenvalues, and hence the search for the lowest complex eigenvaue picks a different one. At larger timesteps than those used in Fig. 12, too high a ρ_b (e.g. above about 0.8) used with the major timestep, can lead to instability, when both Ω^S and Ω^L are large.

3.3. Accuracy of the subcycled explicit generalized alpha method

In the limit of $\rho_b = 1$, the present algorithm is comparable to the subcycled central difference method discussed earlier. With energy dissipation added, it behaves on elastic and elastoplastic problems in a manner similar to its parent algorithm. An example of the accuracy achieved on a simple problem is provided by the

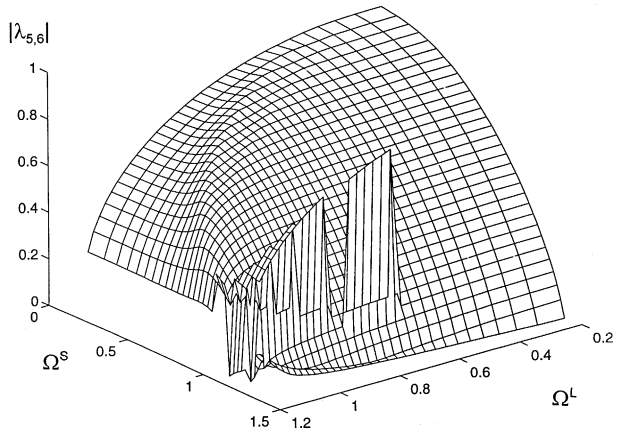


Fig. 13. Damping of mode 3 of the problem of Fig. 1(j), with $\rho_b^S = 0$ and $\rho_b^L = 0$.

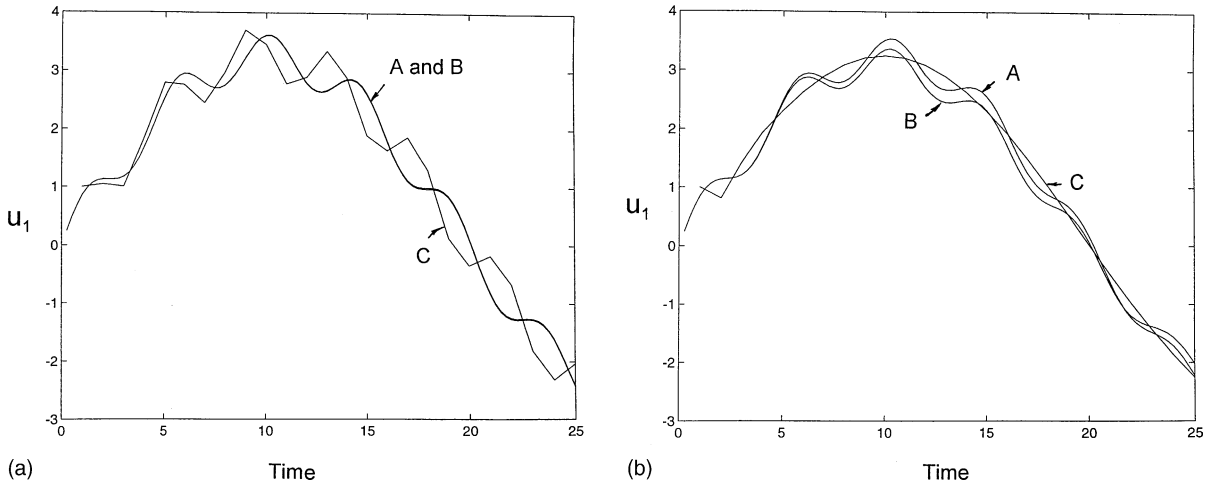


Fig. 14. Alternative approaches to subcycling the explicit generalized alpha method compared on the model problem of Fig. 1(k). (A) small timestep only, (B) partial velocity method and (C) large timestep only. (a) $\rho_b = 1$ case and (b) $\rho_b = 0$ case.

two-mass system of Fig. 1(k), using four subcycles, and timesteps close to the stability limits—a minor timestep of 0.25 was used. The motion of the left mass due to an initial velocity is plotted on Fig. 14. With $\rho_b = 1$ in both partitions, the subcycled partial velocity solution superimposes on the small timestep solution. However, if this problem is changed to one with a mass change, rather than a stiffness change, the accuracy is not as good. A limitation of the algorithm is revealed by the elastic/perfectly plastic one dimensional rod model of Fig. 1(l), with a fine mesh mid-length along a rod, which is impacted on one end. A small timestep is used for the fine mesh. The elastic subcycled solution to this problem, using $\Delta t^S = 1/\sqrt{2}\Delta t^S_{\mathrm{CRIT}}$, with $\rho_b = 0$, is a filtered version of the small timestep solution (Fig. 15). However, with an initial velocity on the left end sufficient to cause all rod elements to yield, there is an early reflection visible on Fig. 16. This error disappears if the large timestep elements are integrated using a $\rho_b = 1$ setting at their nodes. This setting seems appropriate, as the coarser mesh in the large timestep partition already filters high frequencies.

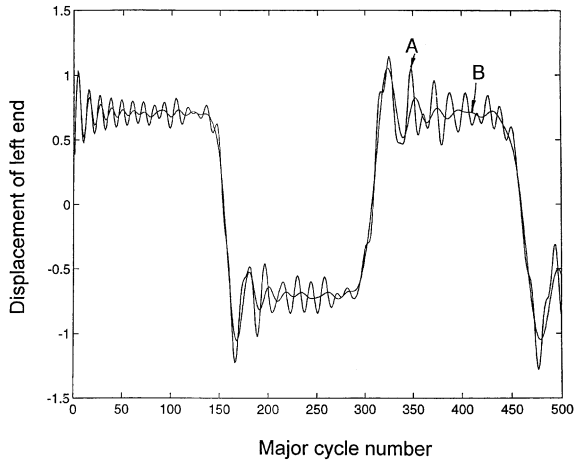


Fig. 15. The partial velocity extension of the explicit generalized alpha method applied to the mesh of linear rod elements of Fig. 1(l). Elastic solution using $\rho_b = 0$. (A) small timestep only and (B) four subcycles.

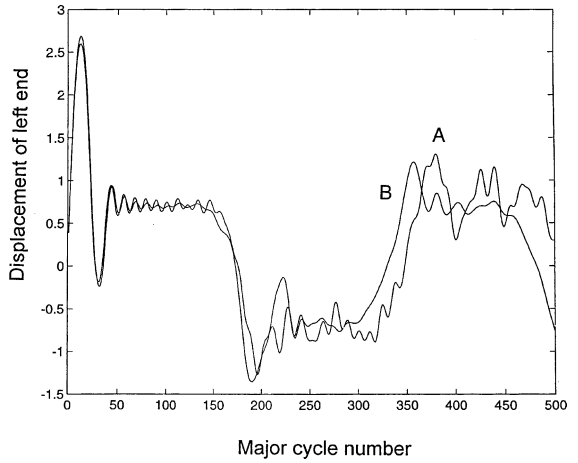


Fig. 16. The partial velocity extension of the explicit generalized alpha method applied to the mesh of linear rod elements of Fig. 1(l). Elastic/perfectly–plastic solution using $\rho_b = 0$ in both partitions. (A) small timestep only and (B) four subcycles.

4. Implicit subcycling

The case for implicit subcycling is not as strong as that for explicit integration, as there is not a problem with stability limits, except for the effects of non-linearities. A subcycled solution could however give extra local detail of a response in which there is genuine spatial localization of deformation. Low frequencies cannot be captured any better than with use of the major timestep only, as thinking elastically, low frequency modes will typically involve motion of the whole model, including the large timestep nodes. Hence, a smaller timestep is of use only to enhance the frequency response locally. Any partial velocity implicit method suffers from the basic flaw that the internal forces acting across the timestep interface that are neglected in a partial velocity formulation can cause error at large timesteps. In an explicit case, these neglected forces only have time to affect the motion of their own interface nodes. Hence if the impulses due to such forces can be assumed to cancell over a major cycle, the resulting total velocities of such nodes are

correctly updated by Eq. (6). In an implicit case, however, using larger timesteps, neglected internal forces acting across a timestep interface can have effects on other nodes remote from the timestep interface. These effects are not corrected, and lead to incorrect prediction of the static component of a response at timesteps above explicit stability limits: for instance, the wrong late-time level of response to a step load.

5. Conclusions

Subcycling explicit integration of structural dynamics problems can be done by summing partial velocities on an element interface. When applied to the central difference method, this approach shows improvements in accuracy and stability over the nodal interface with linear interpolation of large timestep displacements, which is used in the Belytschko et al. algorithm. The partial velocity algorithm is attractive as it is elegant. It can be coded simply by computing sums of impulses due to internal forces, using local element timesteps, and sums of impulses due to external forces, using nodal timesteps. The stability is still statistical in nature, due to the nonlinearity of effectively switching the large timestep partition on and off introducing higher frequencies into the solution. However, by allocating timesteps cautiously in local ratios of two, stability problems can be minimized. Timestep adaptivity can be used to detect and avoid stability problems, and more work is planned on this. Introducing high frequency energy dissipation, by replacing the central difference method with the explicit generalized alpha method, can remove the statistical stability problem, but makes the algorithm less elegant to code. As well, such a dissipative algorithm still has a statistical chance of inaccuracy, due to spurious, damped high frequency oscillations. An alternative is to make the timestep interface itself dissipative, in the manner of Gravouil. This is difficult to do without introducing more dissipation of intermediate frequencies than that of the generalized alpha method. Extension to implicit integration of the partial velocity approach is prevented by the implication that when summing partial velocities, the effects of internal forces acting across the element interface between timestep regions are adequately cancelled. This assumption is only appropriate in an explicit case.

This approach to subcycling seems well suited to particle dynamics problems, where it is important to capture the duration of contact with sufficient accuracy. In this case an "element" is a particular interaction between particles. The element interface of the present algorithm is more appropriate than a nodal interface for this class of problem.

Appendix A

To describe a large timestep update of the single interface degree of freedom model problem, using the explicit generalized alpha method, the following matrix equation of the form $\mathbf{Ay} = \mathbf{Bx}$ can be written as $\mathbf{y} = \mathbf{A}^{-1}\mathbf{Bx}$. $\Omega^L = n\Delta t\sqrt{(k^L/m)}$. the old partial acceleration and partial velocity are retained to form v_β in subcycle updates.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -\gamma_L & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ v_0^S\Delta t \\ a_0^S\Delta t \\ v_n^L n\Delta t \\ a_n^L n^2\Delta t^2 \\ v_0^L n\Delta t \\ a_0^L n^2\Delta t^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1-\gamma_L & 0 & 0 \\ -\dfrac{\Omega^{L^2}}{1-\alpha_L} & 0 & 0 & 0 & -\dfrac{\alpha_L}{1-\alpha_L} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ v_0^S\Delta t \\ a_0^S\Delta t^2 \\ v_0^L n\Delta t \\ a_0^L n^2\Delta t^2 \\ v_0^L n\Delta t \\ a_0^L n^2\Delta t^2 \end{bmatrix}.$$

A small timestep update can be described as follows, using $\Omega^S = \Delta t\sqrt{(k^S/m)}$.

$$\begin{bmatrix} 1 & 0 & -\beta_S & 0 & -\dfrac{\beta_L}{n} & 0 & 0 \\ 0 & 1 & -\gamma_S & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{i+1} \\ v^S_{i+1}\Delta t \\ a^S_{i+1}\Delta t^2 \\ v^L_n n\Delta t \\ a^1_n n^2\Delta t^2 \\ v^L_0 n\Delta t \\ a^L_0 n^2\Delta t^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dfrac{1}{2}-\beta_S & 0 & 0 & \dfrac{1}{n} & \dfrac{\frac{1}{2}-\beta_L}{n} \\ 0 & 1 & 1-\gamma_S & 0 & 0 & 0 & 0 \\ -\dfrac{\Omega^{S^2}}{1-\alpha_S} & 0 & -\dfrac{\alpha_S}{1-\alpha_S} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_i \\ v^S_i\Delta t \\ a^S_i\Delta t^2 \\ v^L_n n\Delta t \\ a^L_n n^2\Delta t^2 \\ v^L_0 n\Delta t \\ a^L_0 n^2\Delta t^2 \end{bmatrix}$$

When these matrices are multiplied together to describe a major cycle, the last two rows and columns can be discarded. The use of separate partial velocities above leads to a redundant mode with a unit eigenvalue, in which mode the partial velocities cancell.

References

[1] T. Belytschko, H.-J. Yen, R. Mullen, Mixed methods for time integration, Comput. Methods Appl. Mech. Engrg. 17/18 (1979) 259–275.

[2] M.O. Neal, T. Belytschko, Explicit–explicit subcycling with non-integer time step ratios for structural dynamic systems, Comput. Struct. 31 (6) (1989) 871–880.

[3] W.J.T. Daniel, A study of the stability of subcycling algorithms in structural dynamics, Comput. Methods Appl. Mech. Engrg. 156 (1998) 1–13.

[4] K.K. Tamma, J.F. D'Costa, A new explicit variable time-integration self-starting methodology for computational structural dynamics, Int. J. Num. Meth. Engrg. 33 (1992) 1165–1180.

[5] T. Belytschko, Y.Y. Lu, An explicit multi-time step integration for parabolic and hyperbolic systems, in: New Methods Trans. Anal., PVP, vol. 246/AMD, vol. 143, ASME, New York, 1992, pp. 25–39.

[6] M. Klisinski, A. Mostrom, On stability of multitime-step integration procedures, J. Engrg. Mech. 124 (7) (1998) 783–793.

[7] W.J.T. Daniel, Analysis and implementation of a new constant acceleration algorithm, Int. J. Num. Meth. Engrg. 40 (1997) 2841–2855.

[8] P. Smolinski, S. Sleith, T. Belytschko, Stability of an explicit multi-timestep integration algorithm for linear structural dynamics equations, Computat. Mech. 18 (1996) 236–244.

[9] Y.S. Wu, P. Smolinski, Multi-time step integration algorithm for structural dynamics based on the modified trapezoidal rule, Comput. Meth. Appl. Mech. Engrg. 187 (3) (2001) 641–660.

[10] W.J.T. Daniel, Subcycling first- and second-order generalizations of the trapezoidal rule, Int. J. Num. Meth. Engrg. 42 (1998) 1091–1119.

[11] W.J.T. Daniel, Multi-timestep integration in computational dynamics, in: J.A.C. Ambrosio, M. Kleiber (Eds.), Computational Aspects of Nonlinear Structural Systems with Large Rigid Body Motion, NATO Science Series, IOS Press, Amsterdam, 2001.

[12] A. Gravouil, A. Combescure, Multi-time-step explicit-implicit method for non-linear structural dynamics, Int. J. Num. Meth. Engrg. 50 (2001) 199–225.

[13] A. Combescure, A. deGayffier, A. Gravouil, N.A. Greffet, A Lagrange multiplier based domain decomposition method for time-dependent problems involving several time-scales. IV World Congress on Computational Mechanics 1998.

[14] G.M. Hulbert, J. Chung, Explicit time integration algorithms for structural dynamics with optimal numerical dissipation, Comput. Meth. Appl. Mech. Engrg. 137 (1996) 175–188.

[15] L. Noels, L. Stainier, J. Ponthot, Self-adapting time integration strategies for non-linear structural dynamics, in: S. Valliappan, N. Khalili, (Eds.), Proc. Computational Mechanics—New Frontiers for New Millenium, 2001.

[16] L. Noels, Determination automatique de la taille du pas de temps pour les schemas implicites en dynamic non-lineaire. Travail de fin d'etude, universite de Liege, Liege.

[17] P. Smolinski, Y.S. Wu, Stability of explicit subcycling time integration with linear interpolation for first-order finite element discretizations, Comput. Methods Appl. Mech. Engrg. 151 (1998) 311–324.

[18] J. Chung, G.M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method, ASME J. Appl. Mech. 60 (1993) 371–375.