Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 5 (3): 402-410, 2005
ISSN 1812-5654

Implementation of Newmark's Method for Second Order Initial Value Problems

M. Sikander Hayat Khiyal
Department of Computer Science, International Islamic University, Sector H-10, Islamabad, Pakistan

Abstract: We discusses the conversion of the one-step Newmark's method into two-step method and proved that has same order as of one-step method. Next we consider the P-stability and phase properties of the method. We consider the implementation of the method using Newton iteration scheme. Also we discuss the estimation of local error and predictor used and presents the one step of the algorithm and convergence criteria. We discuss the stepsize changing strategy and interpolant used to calculate the back values if required. Finally we present the numerical result by applying the method to solve the stiff nonlinear differential system.

Key words: Newmark's method, P-stability, phase lag, initial value problems

INTRODUCTION

Consider the Newmark's method[1]

$$y_{n+1} = y_n + hy'_n + h^2\left\{\alpha f(t_n, y_n) + \beta f(t_{n+1}, y_{n+1})\right\} \quad (1)$$

$$y'_{n+1} = y'_n + h\left\{\gamma f(t_n, y_n) + \delta f(t_{n+1}, y_{n+1})\right\} \quad (2)$$

where, α, β, γ and δ are parameters ($\gamma = 1/2$).
We can write the equation (1)in the form

$$y_n = y_{n-1} + hy'_{n-1} + h^2\{\alpha f(t_{n-1}, y_{n-1}) + \beta f(t_n, y_n)\}$$

then

$$hy'_{n-1} = y_n - y_{n-1} - h^2\{\alpha f(t_{n-1}, y_{n-1}) + \beta f(t_n, y_n)\} \quad (3)$$

From equation (2), we have

$$hy'_n = hy'_{n-1} + h^2\{\gamma f(t_{n-1}, y_{n-1}) + \delta f(t_n, y_n)\}$$

By substituting the value of hy'_{n-1} from equation (3), we get

$$hy'_n = y_n - y_{n-1} - h^2\{(\alpha - \gamma) f(t_{n-1}, y_{n-1}) + (\beta - \delta) f(t_n, y_n)\}$$

Now substituting hy'_n in equation (1), we have

$$\begin{aligned} y_{n+1} &= y_n + y_n - y_{n-1} \\ &- h^2\{(\alpha - \gamma) f(t_{n-1}, y_{n-1}) + (\beta - \delta) f(t_n, y_n)\} \\ &+ h^2\left\{\alpha f(t_n, y_n) + \beta f(t_{n+1}, y_{n+1})\right\} \end{aligned}$$

or, equivalently,

$$y_{n+1} - 2y_n + y_{n-1} = h^2\left\{\beta n_1 + (\alpha - \beta + \delta) n_2 + (\gamma - \alpha) n_3\right\}$$

where, $n_1 = f(t_{n+1}, y_{n+1})$, $n_2 = f(t_n, y_n)$, $n_3 = f(t_{n-1}, y_{n-1})$ or,

$$y_{n+1} - 2y_n + y_{n-1} = h^2\left\{\beta \ddot{y}_{n+1} + (\alpha - \beta + \delta)\ddot{y}_n + (\gamma - \alpha)\ddot{y}_{n-1}\right\} \quad (4)$$

where:

$$\ddot{y}_{n\pm1} = f(t_{n\pm1}, y_{n\pm1}),\ \ddot{y}_n = f(t_n, y_n).$$

ORDER CONDITIONS AND P-STABILITY

Order conditions: The local truncation error for the method (4) is given by

$$\begin{aligned} \Im[y(t_n); h] &= y(t_{n+1}) - 2y(t_n) + y(t_{n-1}) \\ &- h^2\left\{\beta \ddot{y}(t_{n+1}) + (\alpha - \beta + \gamma)\ddot{y}(t_n) + (\gamma - \alpha)\ddot{y}(t_{n-1})\right\} \end{aligned} \quad (5)$$

Now expanding the terms $y(t_{n+1}), y(t_{n-1}), \ddot{y}(t_{n+1})$ and $\ddot{y}(t_{n-1})$ in Taylor series about t_n, we have

$$y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2!}\ddot{y}(t_n) + \frac{h^3}{3!}y^{(3)}(t_n) + \frac{h^4}{4!}y^{(4)}(t_n) + \ldots$$

$$y(t_{n-1}) = y(t_n) - hy'(t_n) + \frac{h^2}{2!}\ddot{y}(t_n) - \frac{h^3}{3!}y^{(3)}(t_n) + \frac{h^4}{4!}y^{(4)}(t_n) - \ldots$$

$$\ddot{y}(t_{n+1}) = \ddot{y}(t_n) + hy^{(3)}(t_n) + \frac{h^2}{2!}y^{(4)}(t_n) + \ldots$$

$$\ddot{y}(t_{n-1}) = \ddot{y}(t_n) - hy^{(3)}(t_n) + \frac{h^2}{2!}y^{(4)}(t_n) - \ldots$$

By substituting the values of $y(t_{n+1}), y(t_{n-1}), \ddot{y}(t_{n+1})$ and $\ddot{y}(t_{n-1})$ in equation (5), we have

$$\begin{aligned} \Im &= (1 - \delta - \gamma)h^2\ddot{y}(t_n) + (\gamma - \alpha - \beta)h^3 y^{(3)}(t_n) + \\ &\left[1/12 - \beta/2 - (\gamma - \alpha)/2\right]h^4 y^{(4)}(t_n) + o(h^5) \end{aligned}$$

The method (4) is consistent if in equation (5), the coefficients of h^0 and h^1 are zero such that the method is of the order greater than or equal to one. Since the coefficient of h^0 and h^1 are zero, thus the method (4) is consistent.

For a second order method, the coefficients of h^0, h^1, h^2 and h^3 are zero while the coefficient of h^4 is non zero. Thus we have
$1-\delta-\gamma = 0$, $\gamma-\alpha-\beta = 0$
and
$1/12-\beta/2-(\gamma-\alpha)/2 \neq 0$.
This implies that
$\delta = 1-\gamma$, $\alpha = \gamma-\beta$ and $\beta \neq 1/12$.

Thus the necessary and sufficient conditions for second order accurate methods are

$$\delta = 1-\gamma, \alpha = \gamma-\beta \text{ and } \beta \neq \frac{1}{12}. \tag{6}$$

P-stability: Lambert and Watson[2] has defined P-stability. To analyses the stability properties of the method, we apply the method (4) to the scalar test equation $\ddot{y} = -c^2 y$, c real. This yield

$$y_{n+1}-2y_n+y_{n-1} = -c^2h^2\{\beta y_{n+1}+(\alpha-\beta+\delta)y_n+(\gamma-\alpha)y_{n-1}\}$$

or, equivalently,

$$(1+\beta c^2h^2)y_{n+1}+\{-2+(\alpha-\beta+\delta)c^2h^2\}y_n + \{1+(\gamma-\alpha)c^2h^2\}y_{n-1} = 0 \tag{7}$$

Equation (7) is a difference equation of the form[3]

$$\tau_2(ch)y_{n+1}+\tau_1(ch)y_n+\tau_0(ch)y_{n-1} = 0$$

where:

$$\tau_2(ch) = 1+\beta c^2h^2,$$
$$\tau_1(ch) = -2+(\alpha-\beta+\delta)c^2h^2,$$
$$\tau_0(ch) = 1+(\gamma-\alpha)c^2h^2.$$

The general solution of this difference equation is

$$y_n = B_1\lambda_1{}^n+B_2\lambda_2{}^n$$

where, B_1 and B_2 are constants and λ_1 and λ_2 are the zeros of the stability polynomial

$$\tau_2r^2+\tau_1r+\tau_0 = 0 \tag{8}$$

Making the Routh-Herwitz transformation $r = \frac{1+z}{1-z}$ becomes

$$(\tau_2-\tau_1+\tau_0)z^2+2(\tau_2-\tau_0)z+(\tau_2+\tau_1+\tau_0) = 0$$

Thus, the necessary and sufficient conditions for P-stability are

$$(\tau_2+\tau_1-\tau_0) \geq 0, \tau_2-\tau_0 = 0, (\tau_2+\tau_1+\tau_0) \geq 0$$

Since $\tau_2 = \tau_0$ we find from (7) that

$$\beta = \gamma-\alpha \tag{9}$$

which is also a condition for second order accuracy. This is a necessary condition for P-stability. When this condition is satisfied, the transformed stability equation is:

$$(2\tau_2-\tau_1)z^2+(2\tau_2+\tau_1) = 0$$

and the necessary and sufficient conditions for P-stability are

$$2\tau_2-\tau_1 \geq 0 \text{ and } 2\tau_2+\tau_1 \geq 0 \tag{10}$$

where:

$$2\tau_2+\tau_1 = (\alpha+\beta+\delta)c^2h^2$$

From the conditions for second order accuracy (6), we have

$$2\tau_2-\tau_1 = 4+(4\beta-1)c^2h^2$$

and

$$2\tau_2+\tau_1 = c^2h^2.$$

Hence the necessary and sufficient conditions for P-stability, (10), becomes

$$c^2h^2 \geq 0 \text{ and } 4+(4\beta-1)c^2h^2 \geq 0$$

for all real values of ch. Thus

$$4\beta-1 \geq 0$$

or, equivalently,

$$\beta \geq \tfrac{1}{4} \tag{11}$$

RESULTS

The necessary and sufficient conditions for second order accuracy are

$\delta = 1-\gamma$, $\alpha = \gamma-\beta$ and $\beta \neq 1/12$.

and for P-stability

$$\beta \geq \frac{1}{4}$$

By using the second order accuracy condition (6) equation (4) can be rewritten in the form

$$y_{n+1}-2y_n+y_{n-1}=h^2\{\beta\ddot{y}_{n+1}+(1-2\beta)\ddot{y}_n+\beta\ddot{y}_{n-1}\} \quad (12)$$

which is second order accurate and P-stable if $\beta\geq 1/4$. If $\beta=1/4$, then this is the Trapezium rule.

The method (12) is Newmark's second order written as an equivalent two step method. Method (12) is equivalent to the following Newmark's method ($\gamma = 1/2$):

$$y_{n+1} = y_n+hy'_n+h^2\{(\tfrac{1}{2}-\beta)\ddot{y}_n+\beta\ddot{y}_{n+1}\} \quad (13)$$

$$y'_{n+1} = y'_n+\tfrac{h}{2}\{\ddot{y}_n+\ddot{y}_{n+1}\} \quad (14)$$

which is second order accurate and P-stable if $\beta\geq 1/4$.

PHASE PROPERTIES

Phase analysis: When Newmark's method (4) is applied to the scalar test equation

$$\ddot{y}-c^2y = ve^{iwt} \quad (15)$$

for real c,v and w, we obtain a recurrence relation of the form

$$r_2y_{n+1}+r_1y_n+r_0y_{n-1} = vh^2\{b_2e^{iwt_{n+1}}+b_1e^{iwt_n}+b_0e^{iwt_{n-1}}\} \quad (16)$$

where, the $\{r_i\}$ and $\{b_i\}$ each depend on ch. The general solution of this equation is:

$$y_n = B_1\lambda^n_1+B_2\lambda^n_2+Q_2e^{iwnh}$$

where, B_1 and B_2 are constants and Q_2e^{iwnh} is a particular solution of the recurrence relation, with Q_2 satisfying:

$$Q_2\left[r_2e^{2iwh}+r_1e^{iwh}+r_0\right] = vh^2\left[b_2e^{2iwh}+b_1e^{iwh}+b_0\right] \quad (17)$$

The numerical forced oscillation is in phase with its analytical counterpart if Q_2 is also real for all real wh[4,5]. Now we will prove that Q_2 is real. We can rewrite (17) in the form:

$$Q_2[A+iB] = vh^2[C+iD]$$

where:

$$A = r_2\cos 2wh+r_1\cos wh+r_0$$
$$B = r_2\sin 2wh+r_1\sin wh$$
$$C = b_2\cos 2wh+b_1\cos wh+b_0$$
$$D = b_2\sin 2wh+b_1\sin wh$$

or

$$Q_2\left[A^2+B^2\right] = vh^2\left[(CA+DB)+i(DA-CB)\right]$$

so Q_2 is real if and only if DA-CB = 0 for all real values of wh.
Then

$$DA\text{-}CB = (b_2r_0-r_2b_0)\sin 2wh + (b_2r_1-b_1r_2+b_1r_0-r_1b_0)\sin wh$$

Thus DA-CB = 0 for all real values of wh, if $r_2 = r_0$, $b_2 = b_0$. So a symmetric method (17) for which

$$r_2 = r_0,\ b_2 = b_0 \quad (18)$$

is certainly in phase.

When Newmark's method (4) is applied to the scalar test equation (15), we obtain

$$(1+\beta c^2h^2)y_{n+1}+\left[-2+(\alpha-\beta+\delta)c^2h^2\right]y_n + \left[1+(\gamma-\alpha)c^2h^2\right]y_{n-1} = vh^2E$$

where:

$$E = \left\{\beta e^{i\omega t_{n+1}}+(\alpha-\beta+\delta)e^{i\omega t_n}+(\gamma-\alpha)e^{i\omega t_{n-1}}\right\}$$

which is of the form (17), for which,

$$r_2=1+\beta c^2h^2,\ r_1=-2+(\alpha-\beta+\delta)c^2h^2,$$
$$r_0=1+(\gamma-\alpha)c^2h^2,$$
$$b_2=\beta,\ b_1=\alpha-\beta+\delta,\ b_0=\gamma-\alpha.$$

Hence, from equation (18), it is in phase, whenever

$$\beta = \gamma-\alpha$$

which is also a necessary condition for P-stability (eq. 9).

Phase lag: The definition of Phase lag is given by Brusa and Nigro[6])
We can rewrite equation (16), with v = 0 in the form

$$\left[1+\psi_1c^2h^2\right]y_{n+1}-2\left[1+\delta_1c^2h^2\right]y_n + \left[1+\psi_1c^2h^2\right]y_{n-1}=0 \quad (19)$$

where:

$$\psi_1=\beta,\ \delta_1=-\tfrac{1}{2}(\alpha-\beta+\delta) \quad (20)$$

Substituting $y_n = e^{n\theta ch}$ in (19) gives

$$\left[1+\psi_1 H^2\right]e^{\theta h}-2\left[1+\delta_1 H^2\right]+\left[1+\psi_1 H^2\right]e^{-\theta h}=0$$

where, ch =H. Expanding in Taylor series

$$e^{\theta h}=1+\theta H+\tfrac{1}{2}\theta^2H^2+\tfrac{1}{6}\theta^3H^3$$
$$+\tfrac{1}{24}\theta^4H^4+\tfrac{1}{120}\theta^5H^5+\tfrac{1}{720}\theta^6H^6...$$
$$e^{-\theta h}=1-\theta H+\tfrac{1}{2}\theta^2H^2-\tfrac{1}{6}\theta^3H^3$$
$$+\tfrac{1}{24}\theta^4H^4-\tfrac{1}{120}\theta^5H^5+\tfrac{1}{720}\theta^6H^6...$$

we have

$$\left[1+\psi_1 H^2\right]e^{\theta h}-2\left[1+\delta_1 H^2\right]+\left[1+\psi_1 H^2\right]e^{-\theta h}$$
$$=\left(\theta^2+2\psi_1-2\delta_1\right)H^2$$
$$+\left(\tfrac{1}{12}\theta^4+\psi_1\theta^2\right)H^4+\left(\tfrac{1}{360}\theta^6+\tfrac{1}{12}\psi_1\theta^4\right)H^6 \quad (21)$$

Let $\theta=\varepsilon+\xi H+\kappa H^2$. Then this becomes

$$\left(\varepsilon^2+2\varepsilon\xi H+2\varepsilon\kappa H^2+\xi^2H^2+2\psi_1-2\delta_1\right)H^2$$
$$+\left(\tfrac{1}{12}\varepsilon^4+\psi_1\varepsilon^2\right)H^4+...$$
$$=\left(\varepsilon^2+2\psi_1-2\delta_1\right)H^2+2\varepsilon\xi H^3$$
$$+\left(2\varepsilon\kappa+\xi^2+\tfrac{1}{12}\varepsilon^4+\psi_1\varepsilon^2\right)H^2+...$$

On equating coefficients of H^j, j =2, 3, 4 to zero, we obtain

$$\varepsilon^2+2\psi_1-2\delta_1=0 \quad (22)$$

$$2\varepsilon\xi=0 \quad (23)$$

$$2\varepsilon\kappa+\xi^2+\tfrac{1}{12}\varepsilon^4+\psi_1\varepsilon^2=0 \quad (24)$$

By substituting the values of Ψ_1 and δ_1 from (20) in (22) we have

$$\varepsilon^2+\alpha+\beta+\delta=0$$

By the order condition (6), α+β+δ = 1. Thus

$$\varepsilon=i.$$

From equation (23) 2εξ = 0 implying that ξ = 0, since ε = i. From (24).

$$2\varepsilon\kappa+\xi^2+\tfrac{1}{2}\varepsilon^4+\psi_1\varepsilon^2=0$$

implying that

$$\kappa=\tfrac{i}{2}\left(\tfrac{1}{12}-\beta\right)$$

Thus

$$\theta=i+\tfrac{i}{2}\left(\tfrac{1}{12}-\beta\right)H^2+O\left(H^3\right)$$

Thus the phase lag is

$$b-1=\tfrac{i}{2}\left(\tfrac{1}{12}-\beta\right)H^2+O\left(H^4\right)$$

If β = 1/12 (in which case the method is not P-stable), then we take

$$\theta=i+\zeta H^3+\sigma H^4$$

Substituting in (21) yields

$$\left(-1+2i\zeta H^3+2i\sigma H^4+2\psi_1-2\delta_1\right)H^2$$
$$+\left(\tfrac{1}{12}-\psi_1\right)H^4+\left(-\tfrac{1}{360}+\tfrac{1}{12}\psi_1\right)H^6+...$$
$$=2i\zeta H^5+\left(2i\sigma-\tfrac{1}{360}+\tfrac{1}{12}\psi_1\right)H^6+...$$

On equating the coefficients of H^j, j = 5, 6 to zero, we have

$$\zeta=0,\ 2i\sigma-\tfrac{1}{360}+\tfrac{1}{12}\psi_1=0$$

then

$$\theta=i+\tfrac{i}{2}\left[-\tfrac{1}{360}+\tfrac{1}{12}\beta\right]H^4+O\left(H^5\right).$$

Hence the phase lag is

$$b-1=\tfrac{i}{2}\left[-\tfrac{1}{360}+\tfrac{1}{12}\beta\right]H^4+O\left(H^5\right)$$

IMPLEMENTATION

Iterative scheme: Consider the P-stable, in phase second order one step version of the Newmark method given by (13) and (14)

$$y'_{n+1}=y'_n+\frac{h}{2}[\ddot{y}_{n+1}+\ddot{y}_n]$$
$$y_{n+1}=y_n+hy'_n+h^2\left[\beta\ddot{y}_{n+1}+\left(\frac{1}{2}-\beta\right)\ddot{y}_n\right]$$

for any $\beta\geq\frac{1}{4}$ where, $\ddot{y}_{n+1}=f\left(t_{n+1},y_{n+1}\right)$, $\ddot{y}_n=f\left(t_n,y_n\right)$

When this method is applied to a nonlinear differential system

$$\ddot{y}=f\left(t,y\right),y\left(a\right)=\eta,y'\left(a\right)=\eta' \quad (25)$$

a nonlinear algebraic system must be solved at each step. This may be solved by using the Newton iteration scheme. Thus on defining

$$F(y)=y-y_n-hy'_n-h^2\left[\beta f\left(t_{n+1},y\right)+\left(\frac{1}{2}-\beta\right)\ddot{y}_n\right]$$

the Newton iteration scheme is

$$\dot{F}\left(y_{n+1}^{(p-1)}\right)\left(y_{n+1}^{(p)}-y_{n+1}^{(p-1)}\right) = F\left(y_{n+1}^{(p-1)}\right),\ p = 1,2,..., \quad (26)$$

where:

$$\dot{F}(y) = I-h^2\beta\frac{\partial f(t_{n+1},y)}{\partial y}$$

and $\frac{\partial f}{\partial y}$ is the Jacobian matrix of f with respect to y. Suppose that J is an approximation for the Jacobian matrix, then

$$\dot{F}(y) = I-h^2\beta J$$

Thus the Newton iteration scheme becomes

$$\left(I-h^2\beta J\right)\left(y_{n+1}^{(p)}-y_{n+1}^{(p-1)}\right) = -F\left(y_{n+1}^{(p-1)}\right)\ p = 1,2,..., \quad (27)$$

where,

$$F\left(y_{n+1}^{(p-1)}\right) = y_{n+1}^{(p-1)}-y_n-hy'_n-h^2\left[\beta f\left(t_{n+1},y_{n+1}^{(p-1)}\right)+\left(\frac{1}{2}-\beta\right)\ddot{y}_n\right]$$

The formation of y'_n is discussed below after the one-step of algorithm.

Next, we consider the P-stable, in phase second order two-step version of the Newmark method given by (12)

$$y_{n+1}-2y_n+y_{n-1} = h^2\left[\beta\ddot{y}_{n+1}+(1-2\beta)\ddot{y}_n+\beta\ddot{y}_{n-1}\right]$$

for any $\beta \geq \frac{1}{4}$ where $\ddot{y}_{n\pm1} = f(t_{n\pm1},y_{n\pm1}), \ddot{y}_n = f(t_n,y_n)$

When this method is applied to a nonlinear differential system (25), a Newton iteration scheme is,

$$\left(I-h^2\beta J\right)\left(y_{n+1}^{(p)}-y_{n+1}^{(p-1)}\right) = -F\left(y_{n+1}^{(p-1)}\right),\ p = 1,2,...$$

where,

$$F\left(y_{n+1}^{(p-1)}\right) = y_{n+1}^{(p-1)}-2y_n+y_{n-1}$$
$$-h^2\left[\beta f\left(t_{n+1},y_{n+1}^{(p-1)}\right)+(1-2\beta)\ddot{y}_n+\beta\ddot{y}_{n-1}\right]$$

Local error estimation and predictors: To estimate the local error, we will consider an approach based on comparing the predicted and corrected values. As the corrector is a second order Newmark's method, the order of the predictor must be one to enable us to compute a local error estimate. We use a predictor based on information available on the current step. We can use the predictor

$$y_{n+1}^{(0)} = y_n+hy'_n \quad (28)$$

which is just the Euler's method.

We can also use the predictor based on the Lagrange interpolant which interpolates to (t_n, y_n) and (t_{n-1},y_{n-1}), we obtain

$$y_{n+1}^{(0)} = 2y_n-y_{n-1} \quad (29)$$

(this predictor can be used only if two back values are known). For the two step version of the Newmark method, we will use this predictor.

Suppose u(t) is the local solution, $y^{(0)}_{n+1}$ is the predicted value of y_{n+1} obtained from (28) or (29) and $y^{(Q)}_{n+1}$ is the corrected value obtained from the iteration (27). Then, provided the local error in $y^{(0)}_{n+1}$ is $O(h^2)$, we have

$$u(t_{n+1})-y_{n+1}^{(0)} = Ah^2+O(h^3)$$
$$u(t_{n+1})-y_{n+1}^{(Q)} = O(h^3)$$

where, A is independent of h. Subtracting these equations, we have

$$y_{n+1}^{(Q)}-y_{n+1}^{(0)} = Ah^2+O(h^3)$$

and hence the local error in $y^{(0)}_{n+1}$ is given by

$$u(t)-y_{n+1}^{(0)} = y_{n+1}^{(Q)}-y_{n+1}^{(0)}+O(h^3)$$

Thus the local error may be estimated from

$$Le_{n+1} = y_{n+1}^{(Q)}-y_{n+1}^{(0)} \quad (31)$$

One-step of the algorithm: Suppose h is the stepsize to be used for the next step. If we use the one step version of Newmark's method, we need y_n, y'_n and $\ddot{y}_n$ which are available from the previous computation. For the two-step version of Newmark's method, we need $y_{n-1}, y'_n, \ddot{y}_{n-1}$ and $\ddot{y}_n$ which are available from previous computations. Also an approximation J to $\partial f/\partial y$ is known. Suppose that the triangular factors of $I-h^2\beta J$ are available and that a predicted value $y^{(0)}_{n+1}$ has already been computed from (28) or (29). The iteration (27) for the Newmark method can be implemented as shown in the following algorithm.

Algrithm: (Second order P-stable Newmark's method)

Set p = 1

Repeat

Evaluate $Z_1 = f(t_{n+1}, y_{n+1}^{(p-1)})$

Calculate

$$G(y_{n+1}^{(p-1)})$$

Solve

$$\left(I-h^2\beta J\right)v = G(y_{n+1}^{(p-1)})$$

Set

$$y_{n+1}^{(p)} = y_{n+1}^{(p-1)} + v$$

$$p = p+1$$

until convergence (Q iteration, say)

(a) When we use the one-step version of Newmark's method, we take

$$G(y_{n+1}^{(p-1)}) = -y_{n+1}^{(p-1)} + y_n + hy'_n + \left(\frac{1}{2}-\beta\right)\ddot{y}_n + \beta h^2 Z_1$$

Before proceeding to the next step, we set $y_{n+1} = y_{n+1}^{(Q)}$.

We wish to calculate $\ddot{y}_{n+1}$ and y'_{n+1} without evaluating $f\left(t_{n+1}, y_{n+1}^{(Q)}\right)$. To do so, we use the (13) to compute $\ddot{y}_{n+1}$ and then (14) to compute y'_{n+1}. Thus

$$\ddot{y}_{n+1} = \frac{1}{\beta}\left[\frac{1}{h^2}\left(y_{n+1}-y_n\right) - \frac{1}{h}y'_n - \left(\frac{1}{2}-\beta\right)\ddot{y}_n\right]$$

and

$$y'_{n+1} = y'_n + \frac{h}{2}\left[\ddot{y}_n + \ddot{y}_{n+1}\right]$$

(b) When we use the two-step version of Newmark's method, we take

$$G(y_{n+1}^{(p-1)}) = -y_{n+1}^{(p-1)} + 2y_n - y_{n-1} + h^2\left[\beta\left(Z_1 + \ddot{y}_{n-1}\right) + \left(1-2\beta\right)\ddot{y}_n\right]$$

Before proceeding to the next step, we set $y_{n+1} = y_{n+1}^{(Q)}$. To calculat $\ddot{y}_{n+1}$ without evaluating $f\left(t_{n+1}, y_{n+1}^{(Q)}\right)$ we use (12). Thus

$$\ddot{y}_{n+1} = \frac{1}{\beta h^2}\left[y_{n+1} - 2y_n + y_{n-1} - \beta h^2\ddot{y}_{n-1} - \left(1-2\beta\right)h^2\ddot{y}_n\right]$$

To test for convergence in the algorithm, we check the condition $||v|| \leq \varepsilon$. If this absolute error test is satisfied then the iteration has converged. We take $\varepsilon = c^*Tol$, where, Tol is the local error tolerance supplied by the user and c is an appropriate constant. We choose c = 1.0. If $||v|| > \varepsilon$ and $p \geq 2$, we form an estimate of the rate of convergence,

$$\text{Rate} = \frac{\left\|y_{n+1}^{(p)} - y_{n+1}^{(p-1)}\right\|}{\left\|y_{n+1}^{(p-1)} - y_{n+1}^{(p-2)}\right\|}.$$

To ensure that convergence is not too slow, we assume that the iteration fails to converge whenever Rate>0.9. If Rate≤0.9 then we perform another step of the iteration and check the above condition. Continue in this way until either the absolute error test is satisfied or the number of iterations exceeds some upper limit, Pmax = 5, (to obtain our results, we choose Pmax = 5). Thus if p>Pmax the iteration is deemed to have failed to converge. If the iteration diverges, some combination of stepsize reduction and reassemble and refactorisation of the iteration matrix $(1-\beta h^2 J)$, with or without reevaluation of the Jacobian matrix J, should be used. For the numerical results obtained below, we have tested two strategies.

Strategy 1: If the iteration diverges then reevaluate the Jacobian (if it has not already been evaluated at this point), halve the stepsize and refactorise the iteration matrix and repeat the step. If it diverges again then halve the stepsize and refactorise the iteration matrix and repeat the step. Continue halving the stepsize and refactorising the iteration matrix until the iteration converges.

If the iteration converges but the step has to be rejected because the local error test is not satisfied, then repeat the step with a smaller stepsize. In this case reevaluate the Jacobian (if it has not already been evaluated at this point) and refactorise the iteration matrix. If the local error test is satisfied and the stepsize is changed the again reevaluate the Jacobian (since this is a new point, the Jacobian could not have been evaluated here before) and refactorise the iteration matrix.

Strategy 2: The Jacobian is reevaluated the first time an iteration diverges and also the iteration matrix is refactorised. Then we repeat the integration step. If divergence occurs again then we halve the stepsize and refactorise the iteration matrix, without reevaluating the Jacobian and repeat the step. We continue halving the stepsize and refactorising the iteration matrix until the iteration converges.

If the iterations converges but the step has to be rejected because the local error test is not satisfied, that is $||Le_{n+1}|| > Tol$, the step is repeated with a smaller stepsize and the iteration matrix is refactorised without reevaluating the Jacobian. If the local error test $||Le_{n+1}|| \leq Tol$ is satisfied, we take $y_{n+1} = y_{n+1}^{(Q)}$ but for the next step if the stepsize is changed then we refactorise the iteration matrix without reevaluating the Jacobian such that we refactorise the iteration matrix whenever the stepsize is changed.

Changing the stepsize: The stepsize to be used for the next (or repeated step) may be calculated from the local error estimate and local error tolerance. A local error estimate for the second order Newmark's method may be obtained from the relation (31):

$$Le_{n+1} = y_{n+1}^{(Q)} - y_{n+1}^{(0)}$$

where, $y_{n+1}^{(Q)}$ is the corrected value and $y_{n+1}^{(0)}$ is the first order predictor. Then, we have from (30) that

$$Le_{n+1} = Ah^2 + O(h^3)$$

This implies that

$$\|A\| \approx \frac{\|Le_{n+1}\|}{h^2}$$

If we choose the stepsize, $\bar{h}$, so that the error estimate on the next (or repeated) step is expected to equal the user supplied local error tolerance Tol, we find that

$$Le_{n+1} = \|A\|\bar{h}^2 + O(\bar{h}^3)$$

or,

$$Le_{n+1} \approx \bar{h}^2 \frac{\|Le_{n+1}\|}{h^2}$$

This implies that

$$\bar{h} \approx h\left\{\frac{Tol}{\|Le_{n+1}\|}\right\}^{\frac{1}{2}}$$

In practice, since the above analysis only holds asymptotically as $h \to 0$, we take

$$\bar{h} = \rho h\left\{\frac{Tol}{\|Le_{n+1}\|}\right\}^{\frac{1}{2}} \quad (32)$$

where, ρ is a safety factor, whose purpose is to avoid failed step (ρ is often taken to be $2^{-\frac{1}{p+1}}$ where p is the order of the predictor, or sometimes 0.8 or 0.9).

To avoid large fluctuations in the stepsize caused by local changes in the error estimate, we put a restriction on the amount by which the stepsize may be increased or decreased. We do not allow the stepsize to decrease by more than a factor ρ_1 or increase by more than a factor ρ_3. Also to avoid the extra function evaluations, Jacobian evaluation and matrix factorisations involved in changing the stepsize, we do not increase the stepsize at all unless it can be increased by a factor of at least ρ_2, where $\rho_2<\rho_3$.

For the numerical results, we have considered $\rho = 2^{-0.5}$ and $\rho_1 = 0.2$, $\rho_2 = 2.0$ and $\rho_3 = 5.0$.

When we apply the two step version of Newmark's method, suppose y_{n+1} is accepted and a new stepsize $\bar{h}$ is predicted for the next step. We need to find the approximation to $y(t_{n+1}-\bar{h})$ and $\ddot{y}(t_{n+1}-\bar{h})$. We can take $y(t_{n+1}-\bar{h}) \approx P_{1,n+1}(t_{n+1}-\bar{h})$ and evaluate the function $f(y(t_{n+1}-\bar{h}))$ where, $P_{1,n+1}(t)$ is the interpolating polynomial of degree one satisfying the conditions $P_{1,n+1}(t_n) = y_n$ and $P_{1,n+1}(t_{n+1}) = y_{n+1}$. We find that

$$P_{1,n+1}(t_{n+1}-\bar{h}) = \left[1-\frac{\bar{h}}{h}\right]y_{n+1} + \frac{\bar{h}}{h}y_n \quad (33)$$

We can take $y(t_{n+1}-\bar{h}) \approx P_{2,n+1}(t_{n+1}-\bar{h}), \ddot{y}(t_{n+1}-\bar{h}) \approx \ddot{P}_{2,n+1}(t_{n+1}-\bar{h})$
Where $P_{2,n+1}$ (t) is the interpolating polynomial of degree two satisfying the conditions $P_{2,n+1}(t_n) = y_n$ and $P_{2,n+1}(t_{n+1}) = y_{n+1}$ and either (a) $\ddot{P}_{2,n+1}(t_{n+1}) = \ddot{y}_{n+1}$ or (b) $\ddot{P}_{2,n+1}(t) = \ddot{y}_n$ we find that

$$\text{(a)} \quad P_{2,n+1}(t_{n+1}-\bar{h}) = \left[1-\frac{\bar{h}}{h}\right]y_{n+1} + \frac{\bar{h}}{h}y_n - \frac{\bar{h}}{2}(h-\bar{h})\ddot{y}_{n+1} \quad (34)$$

$$\text{(b)} \quad \ddot{P}_{2,n+1}(t_{n+1}-\bar{h}) = 2\ddot{y}_{n+1} \quad (35)$$

$$\text{and} \quad P_{2,n+1}(t_{n+1}-\bar{h}) = \left[1-\frac{\bar{h}}{h}\right]y_{n+1} + \frac{\bar{h}}{h}y_n - \frac{\bar{h}}{2}(h-\bar{h})\ddot{y}_n \quad (36)$$

$$\ddot{P}_{2,n+1}(t_{n+1}-\bar{h}) = 2\ddot{y}_n \quad (37)$$

We can take $y(t_{n+1}-\bar{h}) \approx P_{3,n+1}(t_{n+1}-\bar{h}), \ddot{y}(t_{n+1}-\bar{h}) \approx \ddot{P}_{3,n+1}(t_{n+1}-\bar{h})$ is the interpolating polynomial of degree three satisfying the conditions $P_{3,n+1}(t_n) = y_n$, $P_{3,n+1}(t_{n+1}) = y_{n+1}$ and $\ddot{P}_{3,n+1}(t_n) = \ddot{y}_n$ and $\ddot{P}_{3,n+1}(t_{n+1}) = \ddot{y}_{n+1}$.
We find that

$$P_{3,n+1}(t_{n+1}-\bar{h}) = \left[1-\frac{\bar{h}}{h}\right]y_{n+1} + \frac{\bar{h}}{h}y_n + \left[\frac{\bar{h}^2}{2} - \frac{h\bar{h}}{3} - \frac{\bar{h}^3}{6h}\right]\ddot{y}_{n+1} + \left[\frac{\bar{h}^3}{6h} - \frac{h\bar{h}}{6}\right]\ddot{y}_n \quad (38)$$

and

$$\ddot{P}_{3,n+1}\left(t_{n+1}-\bar{h}\right) = \left[1-\frac{\bar{h}}{h}\right]\ddot{y}_{n+1}+\frac{\bar{h}}{h}\ddot{y}_n \tag{39}$$

We can use any one interpolating polynomial discussed above to find approximation to $y\left(t_{n+1}-\bar{h}\right)$ and $\ddot{y}\left(t_{n+1}-\bar{h}\right)$. A similar approach can be adopted when y_{n+1} has to be rejected (because of error test failure) and the step repeated with a new stepsize h and also when the iteration diverges and the stepsize is halved.

Starting technique and numerical results: Since the two step version of Newmark's method requires the initial calculation of y_0 and y_1, given y_0 and $\dot{y}_0$ (initial conditions), we need to calculate the starting value y_1 before applying the two-step version of Newmark's method and form the error estimate. If the error test is satisfied and y_1 is accepted then we proceed to the two step version of Newmark's method. For the one-step version of Newmark's method, we use the predictor given by (28) while (29) for the two-step version of Newmark's method.

On change of stepsize, when we apply the two-step version of Newmark's method, we have considered different choices of interpolating polynomial discussed above. For the numerical results, we have used the following notation for these choices.

NM1ST: One step version of Newmark method.

NM2P1: Two step version of Newmark method and apply the interpolating polynomial of degree one given by (33) to approximate $y(t-\bar{h})$ and evaluate $f(y(t-\bar{h}))$

NM2P2: Two step version of Newmark method and apply the interpolating polynomial of degree two given by (34) and (35) to approximate $y(t-\bar{h})$ and $\ddot{y}(t-\bar{h})$

NM2P3: Two step version of Newmark method and apply the cubic Hermite interpolant given by (38) and (39) to approximate $y(t-\bar{h})$ and $\ddot{y}(t-\bar{h})$

For the numerical results, we have solved the following initial value problems

Exmple 1: The scalar nonlinear problem

$$\ddot{y}+\sinh y = 0,\ y(0) = 1,\ \dot{y}(0) = 0.$$

The next two problems both use the same differential equation but with different initial conditions. These nonlinear differential equations are

$$\ddot{y}_1+\sinh\left(y_1+y_2\right) = 0,\ \ddot{y}_2+10^4 y_2 = 0.$$

Exmple 2: $y_1(0) = 1$, $\dot{y}_1(0) = 0$, $y_2(0) = 10^{-4}$, $\dot{y}_2(0) = 0$.
Exmple 3: $y_1(0) = 1$, $\dot{y}_1(0) = 0$, $y_2(0) = 10^{-8}$, $\dot{y}_2(0) = 0$.

The above problems have been solved for $t \in [0,6]$. The error at the end point is obtained by comparing the computed solution with the solution obtained by using a fixed step code with a small stepsize for example 1 and for the first equation of examples 2 and 3. For the second equation of example 2 and 3 we have use the exact solution. We denote the error at the end point by MAXERR where for the scalar equation it is $|\text{Error at } t=6|$ and for the system it is $||\text{Error at } t=6||_\infty$. For these examples the stepsize is chosen initially to be h = 1.0. The results obtained for the cases where the maximum number of iteration permitted, Pmax, is 5.

Table 1: Methods are compared for example 1 with Pmax = 5, Tol = 10^{-2} and $\epsilon = 10^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	F1
NM1ST	1.48×10^{-3}	66	1	65	52	58	52	9	6	-
NM2P1	4.91×10^{-2}	101	1	88	69	78	69	13	9	12
NM2P2	2.00×10^{-2}	83	1	82	67	74	67	10	7	-
NM2P3	1.66×10^{-3}	92	1	91	72	81	72	13	9	-

Table 2: Methods are compared for example 1 with Pmax = 5, Tol = 10^{-4} and $\epsilon = 10^{-4}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	F1
NM1ST	3.17×10^{-5}	488	1	487	463	474	463	15	11	-
NM2P1	8.39×10^{-3}	738	1	720	692	705	692	19	13	17
NM2P2	1.55×10^{-3}	679	1	678	650	663	650	18	13	-
NM2P3	2.89×10^{-5}	681	1	680	652	665	652	18	13	-

Table 3: Methods are compared for example 2 with Pmax = 5, Tol = 10^{-2} and $\epsilon = {}^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	1.47×10^{-3}	66	1	65	52	58	52	9	6	10	-
NM2P1	2.16×10^{-2}	93	1	85	72	78	72	8	6	9	7
NM2P2	4.75×10^{-3}	379	1	378	269	323	269	103	54	104	-
NM2P3	7.08×10^{-4}	88	1	87	72	79	72	11	7	12	-

Table 4: Methods are compared for example 2 with Pmax = 5, Tol = 10^{-4} and $\epsilon = ^{-4}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	1.88×10^{-5}	567	1	566	540	552	540	18	12	19	-
NM2P1	2.06×10^{-3}	846	1	816	770	792	770	31	22	32	29
NM2P2	1.79×10^{-4}	733	1	732	706	718	706	18	12	19	-
NM2P3	5.39×10^{-5}	803	1	802	738	769	738	46	31	47	-

Table 5: Methods are compared for example 3 with Pmax = 5, Tol = 10^{-2} and $\epsilon = ^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	1.48×10^{-2}	66	1	65	52	58	52	9	6	10	-
NM2P1	4.91×10^{-2}	101	1	88	69	78	69	13	9	14	12
NM2P2	2.01×10^{-2}	83	1	82	67	74	67	10	7	11	-
NM2P3	1.66×10^{-2}	92	1	91	72	81	72	13	9	14	-

Table 6: Methods are compared for example 3 with Pmax = 5, Tol = 10^{-2} and $\epsilon = ^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	3.17×10^{-2}	488	1	487	463	474	463	15	11	16	-
NM2P1	8.39×10^{-2}	738	1	720	692	705	692	19	13	20	17
NM2P2	1.55×10^{-2}	679	1	678	650	663	650	18	13	19	-
NM2P3	2.89×10^{-2}	681	1	680	652	665	652	18	13	19	-

Table 7: Methods are compared for example 1 with Pmax = 5, Tol = 10^{-2} and $\epsilon = ^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	F1
NM1ST	2.08×10^{-3}	66	8	65	52	58	52	9	6	-
NM2P1	4.73×10^{-2}	101	10	88	69	78	69	13	9	12
NM2P2	2.28×10^{-2}	83	8	82	67	74	67	10	7	-
NM2P3	1.56×10^{-3}	92	10	91	72	81	72	13	9	-

Table 8: Methods are compared for example 2 with Pmax = 5, Tol = 10^{-2} and $\epsilon = ^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	2.08×10^{-3}	66	8	65	52	58	52	9	6	10	-
NM2P1	2.03×10^{-2}	93	6	85	72	78	72	8	6	9	7
NM2P2	5.85×10^{-3}	514	109	513	292	402	292	215	110	216	-
NM2P3	1.76×10^{-4}	88	8	87	72	79	72	11	7	12	-

Table 9: Methods are compared for example 3 with Pmax = 5, Tol = 10^{-2} and $\epsilon = ^{-2}$

Methods	MAXERR	FCN	JC	NIT	NST	STP	SST	CS	FS	FT	F1
NM1ST	2.08×10^{-3}	66	8	65	52	58	52	9	6	10	-
NM2P1	4.73×10^{-2}	101	10	88	69	78	69	13	9	14	12
NM2P2	2.28×10^{-2}	83	8	82	67	74	67	10	7	11	-
NM2P3	1.56×10^{-3}	92	10	91	72	81	72	13	9	14	-

In Tables, we give the results of solving these problems with Tol = 10^{-2} and 10^{-4}.
In each case, we present the following statistics:

- Number of evaluation of the differential equation right hand side f, FCN;
- Number of evaluation of the Jacobian $\partial f/\partial y$, JC;
- Number of iterations overall, NIT;
- Number of iterations in steps where iteration converged, NST;
- Number of steps overall, STP;
- Number of successful steps to complete the integration, SST;
- Number of steps where the stepsize is changes, CS;
- Number of failed steps, FS;
- Number of LU factorisation of the iteration matrix, FT;
- Number of function evaluations by using the interpolating polynomial of degree one given by (30) to find back values on change of stepsize, FP1.

The results are presented in Table1-6 for strategy 1 and in Table 7-9 for strategy 2.

REFERENCES

1. Newmark, N.M., 1959. A method of computation for structural dynamics. J. Eng. Mech. Division. ASCE., 8: 67-94.
2. Lambert, J.D. and I.A. Watson, 1976. Symmetric multistep methods for periodic initial value problems. J. Inst. Maths. Applics., 18: 189-202.
3. Coleman, J.P., 1989. Numerical method for y = f(x, y) via rational approximation for the cosine. IMA J. Numer. Anal, 9: 145-165.
4. Gladwell, I. and R.M. Thomas, 1983. Damping and phase analysis for some methods for solving second order ordinary differential equation. Intl. J. Num. Meth. Eng., 19: 495-503.
5. Khiyal, M.S.H., 1991. Efficient algorithms based on direct hybrid methods for second order initial value problems. Ph.D Thesis UMIST.
6. Brusa, L. and L. Nigro, 1980. A one step method for direct integration of structural dynamics equations. Intl. J. Num. Meth. Engng., 15: 685-699.