

5

Numerical Evaluation of Dynamic Response

PREVIEW

Analytical solution of the equation of motion for a single-degree-of-freedom system is usually not possible if the excitation—applied force $p(t)$ or ground acceleration $\ddot{u}_g(t)$ —varies arbitrarily with time or if the system is nonlinear. Such problems can be tackled by numerical time-stepping methods for integration of differential equations. A vast body of literature, including major chapters of several books, exists about these methods for solving various types of differential equations that arise in the broad subject area of applied mechanics. The literature includes the mathematical development of these methods; their accuracy, convergence, and stability properties; and computer implementation.

Only a brief presentation of a very few methods that are especially useful in dynamic response analysis of SDF systems is included here, however. This presentation is intended to provide only the basic concepts underlying these methods and to provide a few computational algorithms. Although these would suffice for many practical problems and research applications, the reader should recognize that a wealth of knowledge exists on this subject.

5.1 TIME-STEPPING METHODS

For an inelastic system the equation of motion to be solved numerically is

$$m\ddot{u} + c\dot{u} + f_S(u) = p(t) \quad \text{or} \quad -m\ddot{u}_g(t) \quad (5.1.1)$$

subject to the initial conditions

$$u_0 = u(0) \quad \dot{u}_0 = \dot{u}(0)$$

The system is assumed to have linear viscous damping, but other forms of damping, including nonlinear damping, could be considered, as will become obvious later. However, this is rarely done for lack of information on damping, especially at large amplitudes of motion. The applied force $p(t)$ is given by a set of discrete values $p_i = p(t_i)$, $i = 0$ to N (Fig. 5.1.1). The time interval

$$\Delta t_i = t_{i+1} - t_i \quad (5.1.2)$$

is usually taken to be constant, although this is not necessary. The response is determined at the discrete time instants t_i , denoted as time i ; the displacement, velocity, and acceleration of the SDF system are u_i , \dot{u}_i , and \ddot{u}_i , respectively. These values, assumed to be known, satisfy Eq. (5.1.1) at time i :

$$m\ddot{u}_i + c\dot{u}_i + (f_s)_i = p_i \quad (5.1.3)$$

where $(f_s)_i$ is the resisting force at time i ; $(f_s)_i = ku_i$ for a linearly elastic system but would depend on the prior history of displacement and on the velocity at time i if the system were nonlinear. The numerical procedures to be presented will enable us to determine the response quantities u_{i+1} , \dot{u}_{i+1} , and \ddot{u}_{i+1} at time $i + 1$ that satisfy Eq. (5.1.1) at

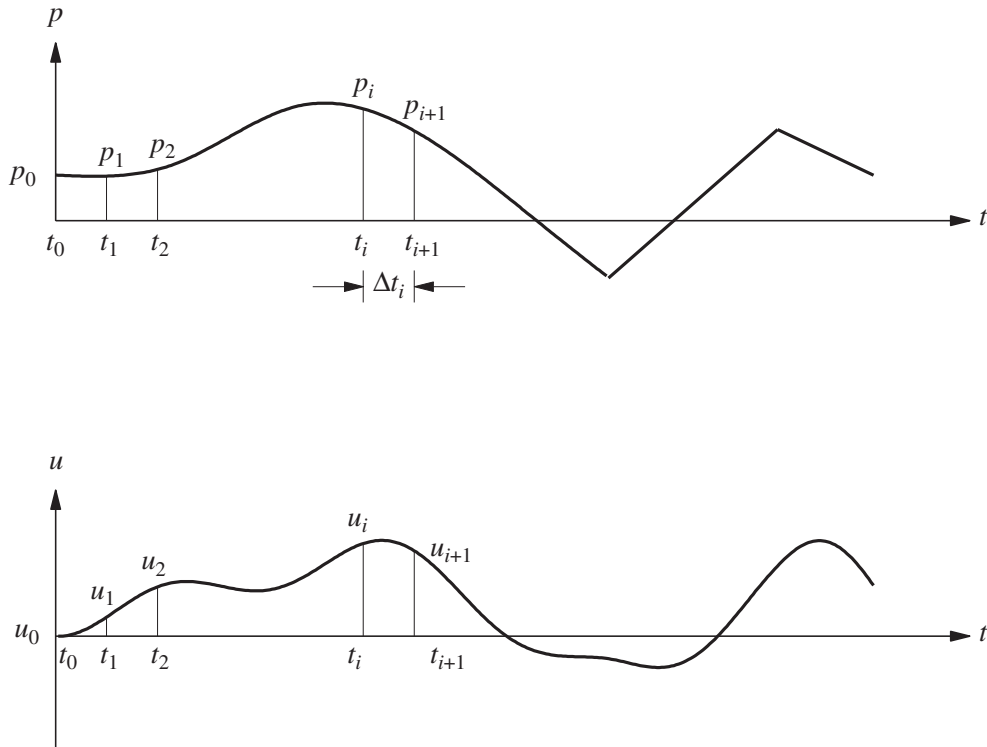


Figure 5.1.1 Notation for time-stepping methods.

time $i + 1$:

$$m\ddot{u}_{i+1} + c\dot{u}_{i+1} + (f_s)_{i+1} = p_{i+1} \quad (5.1.4)$$

When applied successively with $i = 0, 1, 2, 3, \dots$, the time-stepping procedure gives the desired response at all time instants $i = 1, 2, 3, \dots$. The known initial conditions, $u_0 = u(0)$ and $\dot{u}_0 = \dot{u}(0)$, provide the information necessary to start the procedure.

Stepping from time i to $i + 1$ is usually not an exact procedure. Many approximate procedures are possible that are implemented numerically. The three important requirements for a numerical procedure are (1) convergence—as the time step decreases, the numerical solution should approach the exact solution, (2) stability—the numerical solution should be stable in the presence of numerical round-off errors, and (3) accuracy—the numerical procedure should provide results that are close enough to the exact solution. These important issues are discussed briefly in this book; comprehensive treatments are available in books emphasizing numerical solution of differential equations.

Three types of time-stepping procedures are presented in this chapter: (1) methods based on interpolation of the excitation function, (2) methods based on finite difference expressions of velocity and acceleration, and (3) methods based on assumed variation of acceleration. Only one method is presented in each of the first two categories and two from the third group.

5.2 METHODS BASED ON INTERPOLATION OF EXCITATION

A highly efficient numerical procedure can be developed for linear systems by interpolating the excitation over each time interval and developing the exact solution using the methods of Chapter 4. If the time intervals are short, linear interpolation is satisfactory. Figure 5.2.1 shows that over the time interval $t_i \leq t \leq t_{i+1}$, the excitation function is given by

$$p(\tau) = p_i + \frac{\Delta p_i}{\Delta t_i} \tau \quad (5.2.1a)$$

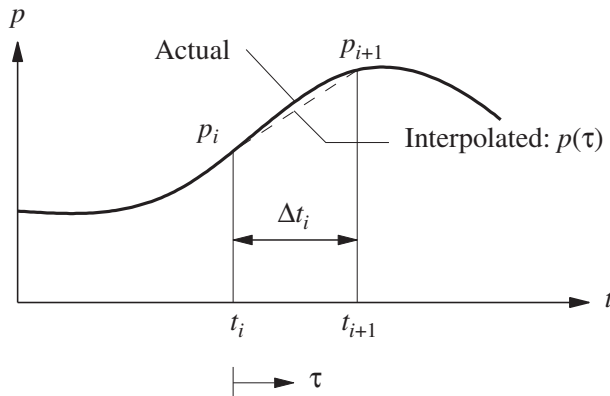


Figure 5.2.1 Notation for linearly interpolated excitation.

where

$$\Delta p_i = p_{i+1} - p_i \quad (5.2.1b)$$

and the time variable τ varies from 0 to Δt_i . For algebraic simplicity, we first consider systems without damping; later, the procedure will be extended to include damping. The equation to be solved is

$$m\ddot{u} + ku = p_i + \frac{\Delta p_i}{\Delta t_i} \tau \quad (5.2.2)$$

subject to initial conditions $u(0) = u_i$ and $\dot{u}(0) = \dot{u}_i$. The response $u(\tau)$ over the time interval $0 \leq \tau \leq \Delta t_i$ is the sum of three parts: (1) free vibration due to initial displacement u_i and velocity \dot{u}_i at $\tau = 0$, (2) response to step force p_i with zero initial conditions, and (3) response to ramp force $(\Delta p_i / \Delta t_i) \tau$ with zero initial conditions. Adapting the available solutions for these three cases from Sections 2.1, 4.3, and 4.4, respectively, gives

$$u(\tau) = u_i \cos \omega_n \tau + \frac{\dot{u}_i}{\omega_n} \sin \omega_n \tau + \frac{p_i}{k} (1 - \cos \omega_n \tau) + \frac{\Delta p_i}{k} \left(\frac{\tau}{\Delta t_i} - \frac{\sin \omega_n \tau}{\omega_n \Delta t_i} \right) \quad (5.2.3a)$$

and differentiating $u(\tau)$ leads to

$$\frac{\dot{u}(\tau)}{\omega_n} = -u_i \sin \omega_n \tau + \frac{\dot{u}_i}{\omega_n} \cos \omega_n \tau + \frac{p_i}{k} \sin \omega_n \tau + \frac{\Delta p_i}{k} \frac{1}{\omega_n \Delta t_i} (1 - \cos \omega_n \tau) \quad (5.2.3b)$$

Evaluating these equations at $\tau = \Delta t_i$ gives the displacement u_{i+1} and velocity \dot{u}_{i+1} at time $i + 1$:

$$\begin{aligned} u_{i+1} = & u_i \cos(\omega_n \Delta t_i) + \frac{\dot{u}_i}{\omega_n} \sin(\omega_n \Delta t_i) \\ & + \frac{p_i}{k} [1 - \cos(\omega_n \Delta t_i)] + \frac{\Delta p_i}{k} \frac{1}{\omega_n \Delta t_i} [\omega_n \Delta t_i - \sin(\omega_n \Delta t_i)] \end{aligned} \quad (5.2.4a)$$

$$\begin{aligned} \frac{\dot{u}_{i+1}}{\omega_n} = & -u_i \sin(\omega_n \Delta t_i) + \frac{\dot{u}_i}{\omega_n} \cos(\omega_n \Delta t_i) \\ & + \frac{p_i}{k} \sin(\omega_n \Delta t_i) + \frac{\Delta p_i}{k} \frac{1}{\omega_n \Delta t_i} [1 - \cos(\omega_n \Delta t_i)] \end{aligned} \quad (5.2.4b)$$

These equations can be rewritten after substituting Eq. (5.2.1b) as recurrence formulas:

$$u_{i+1} = Au_i + B\dot{u}_i + Cp_i + Dp_{i+1} \quad (5.2.5a)$$

$$\dot{u}_{i+1} = A'\dot{u}_i + B'\dot{u}_i + C'p_i + D'p_{i+1} \quad (5.2.5b)$$

Repeating the derivation above for under-critically damped systems (i.e., $\zeta < 1$) shows that Eqs. (5.2.5) also apply to damped systems with the expressions for the coefficients A, B, \dots, D' given in Table 5.2.1. They depend on the system parameters ω_n, k , and ζ , and on the time interval $\Delta t \equiv \Delta t_i$.

Since the recurrence formulas are derived from exact solution of the equation of motion, the only restriction on the size of the time step Δt is that it permit a close approximation to the excitation function and that it provide response results at closely spaced time intervals so that the response peaks are not missed. This numerical procedure is especially

TABLE 5.2.1 COEFFICIENTS IN RECURRENCE FORMULAS ($\zeta < 1$)

$A = e^{-\zeta \omega_n \Delta t} \left(\frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_D \Delta t + \cos \omega_D \Delta t \right)$
$B = e^{-\zeta \omega_n \Delta t} \left(\frac{1}{\omega_D} \sin \omega_D \Delta t \right)$
$C = \frac{1}{k} \left\{ \frac{2\zeta}{\omega_n \Delta t} + e^{-\zeta \omega_n \Delta t} \left[\left(\frac{1 - 2\zeta^2}{\omega_D \Delta t} - \frac{\zeta}{\sqrt{1 - \zeta^2}} \right) \sin \omega_D \Delta t - \left(1 + \frac{2\zeta}{\omega_n \Delta t} \right) \cos \omega_D \Delta t \right] \right\}$
$D = \frac{1}{k} \left[1 - \frac{2\zeta}{\omega_n \Delta t} + e^{-\zeta \omega_n \Delta t} \left(\frac{2\zeta^2 - 1}{\omega_D \Delta t} \sin \omega_D \Delta t + \frac{2\zeta}{\omega_n \Delta t} \cos \omega_D \Delta t \right) \right]$
$A' = -e^{-\zeta \omega_n \Delta t} \left(\frac{\omega_n}{\sqrt{1 - \zeta^2}} \sin \omega_D \Delta t \right)$
$B' = e^{-\zeta \omega_n \Delta t} \left(\cos \omega_D \Delta t - \frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_D \Delta t \right)$
$C' = \frac{1}{k} \left\{ -\frac{1}{\Delta t} + e^{-\zeta \omega_n \Delta t} \left[\left(\frac{\omega_n}{\sqrt{1 - \zeta^2}} + \frac{\zeta}{\Delta t \sqrt{1 - \zeta^2}} \right) \sin \omega_D \Delta t + \frac{1}{\Delta t} \cos \omega_D \Delta t \right] \right\}$
$D' = \frac{1}{k \Delta t} \left[1 - e^{-\zeta \omega_n \Delta t} \left(\frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_D \Delta t + \cos \omega_D \Delta t \right) \right]$

useful when the excitation is defined at closely spaced time intervals—as for earthquake ground acceleration—so that the linear interpolation is essentially perfect. If the time step Δt is constant, the coefficients A, B, \dots, D' need to be computed only once.

The exact solution of the equation of motion required in this numerical procedure is feasible only for linear systems. It is conveniently developed for SDF systems, as shown above, but would be impractical for MDF systems unless their response is obtained by the superposition of modal responses (Chapters 12 and 13).

Example 5.1

An SDF system has the following properties: $m = 0.2533 \text{ kip-sec}^2/\text{in.}$, $k = 10 \text{ kips/in.}$, $T_n = 1 \text{ sec}$ ($\omega_n = 6.283 \text{ rad/sec}$), and $\zeta = 0.05$. Determine the response $u(t)$ of this system to $p(t)$ defined by the half-cycle sine pulse force shown in Fig. E5.1 by (a) using piecewise linear interpolation of $p(t)$ with $\Delta t = 0.1 \text{ sec}$, and (b) evaluating the theoretical solution.

Solution

1. Initial calculations

$$e^{-\zeta \omega_n \Delta t} = 0.9691 \quad \omega_D = \omega_n \sqrt{1 - \zeta^2} = 6.275$$

$$\sin \omega_D \Delta t = 0.5871 \quad \cos \omega_D \Delta t = 0.8095$$

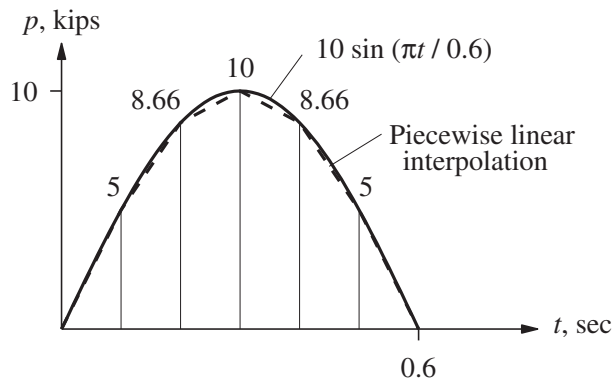


Figure E5.1

Substituting these in Table 5.2.1 gives

$$A = 0.8129 \quad B = 0.09067 \quad C = 0.01236 \quad D = 0.006352$$

$$A' = -3.5795 \quad B' = 0.7559 \quad C' = 0.1709 \quad D' = 0.1871$$

2. Apply the recurrence equations (5.2.5). The resulting computations are summarized in Tables E5.1a and E5.1b.

TABLE E5.1a NUMERICAL SOLUTION USING LINEAR INTERPOLATION OF EXCITATION

t_i	p_i	Cp_i	Dp_{i+1}	$B\ddot{u}_i$	\dot{u}_i	Au_i	u_i	Theoretical u_i
0.0	0.0000	0.0000	0.0318	0.0000	0.0000	0.0000	0.0000	0.0000
0.1	5.0000	0.0618	0.0550	0.0848	0.9354	0.0258	0.0318	0.0328
0.2	8.6602	0.1070	0.0635	0.2782	3.0679	0.1849	0.2274	0.2332
0.3	10.0000	0.1236	0.0550	0.4403	4.8558	0.5150	0.6336	0.6487
0.4	8.6603	0.1070	0.0318	0.4290	4.7318	0.9218	1.1339	1.1605
0.5	5.0000	0.0618	0.0000	0.1753	1.9336	1.2109	1.4896	1.5241
0.6	0.0000	0.0000	0.0000	-0.2735	-3.0159	1.1771	1.4480	1.4814
0.7	0.0000	0.0000	0.0000	-0.6767	-7.4631	0.7346	0.9037	0.9245
0.8	0.0000	0.0000	0.0000	-0.8048	-8.8765	0.0471	0.0579	0.0593
0.9	0.0000	0.0000	0.0000	-0.6272	-6.9177	-0.6160	-0.7577	-0.7751
1.0	0.0000				-2.5171		-1.2432	-1.2718

3. Compute the theoretical response. Equation (3.2.5)—valid for $t \leq 0.6$ sec, Eq. (2.2.4) modified appropriately—valid for $t \geq 0.6$ sec, and the derivatives of these two equations are evaluated for each t_i ; the results are given in Tables E5.1a and E5.1b.

4. Check the accuracy of the numerical results. The numerical solution based on piecewise linear interpolation of the excitation agrees reasonably well with the theoretical solution. The discrepancy arises because the half-cycle sine curve has been replaced by the series of straight lines shown in Fig. E5.1. With a smaller Δt the piecewise linear approximation would be closer to the half-cycle sine curve, and the numerical solution would be more accurate.

TABLE E5.1b NUMERICAL SOLUTION USING LINEAR INTERPOLATION OF EXCITATION

t_i	p_i	$C'p_i$	$D'p_{i+1}$	$A'u_i$	u_i	$B'\dot{u}_i$	\dot{u}_i	Theoretical \dot{u}_i
0.0	0.0000	0.0000	0.9354	0.0000	0.0000	0.0000	0.0000	0.0000
0.1	5.0000	0.8544	1.6201	-0.1137	0.0318	0.7071	0.9354	0.9567
0.2	8.6602	1.4799	1.8707	-0.8140	0.2274	2.3192	3.0679	3.1383
0.3	10.0000	1.7088	1.6201	-2.2679	0.6336	3.6708	4.8558	4.9674
0.4	8.6603	1.4799	0.9354	-4.0588	1.1339	3.5771	4.7318	4.8408
0.5	5.0000	0.8544	0.0000	-5.3320	1.4896	1.4617	1.9336	1.9783
0.6	0.0000	0.0000	0.0000	-5.1832	1.4480	-2.2799	-3.0159	-3.0848
0.7	0.0000	0.0000	0.0000	-3.2347	0.9037	-5.6418	-7.4631	-7.6346
0.8	0.0000	0.0000	0.0000	-0.2074	0.0579	-6.7103	-8.8765	-9.0808
0.9	0.0000	0.0000	0.0000	2.7124	-0.7577	-5.2295	-6.9177	-7.0771
1.0	0.0000				-1.2432		-2.5171	-2.5754

5.3 CENTRAL DIFFERENCE METHOD

This method is based on a finite difference approximation of the time derivatives of displacement (i.e., velocity and acceleration). Taking constant time steps, $\Delta t_i = \Delta t$, the central difference expressions for velocity and acceleration at time i are

$$\dot{u}_i = \frac{u_{i+1} - u_{i-1}}{2\Delta t} \quad \ddot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta t)^2} \quad (5.3.1)$$

Substituting these approximate expressions for velocity and acceleration into Eq. (5.1.3), specialized for linearly elastic systems, gives

$$m \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta t)^2} + c \frac{u_{i+1} - u_{i-1}}{2\Delta t} + ku_i = p_i \quad (5.3.2)$$

In this equation u_i and u_{i-1} are assumed known (from implementation of the procedure for the preceding time steps). Transferring these known quantities to the right side leads to

$$\left[\frac{m}{(\Delta t)^2} + \frac{c}{2\Delta t} \right] u_{i+1} = p_i - \left[\frac{m}{(\Delta t)^2} - \frac{c}{2\Delta t} \right] u_{i-1} - \left[k - \frac{2m}{(\Delta t)^2} \right] u_i \quad (5.3.3)$$

or

$$\hat{k}u_{i+1} = \hat{p}_i \quad (5.3.4)$$

where

$$\hat{k} = \frac{m}{(\Delta t)^2} + \frac{c}{2\Delta t} \quad (5.3.5)$$

and

$$\hat{p}_i = p_i - \left[\frac{m}{(\Delta t)^2} - \frac{c}{2\Delta t} \right] u_{i-1} - \left[k - \frac{2m}{(\Delta t)^2} \right] u_i \quad (5.3.6)$$

TABLE 5.3.1 CENTRAL DIFFERENCE METHOD[†]

1.0 Initial calculations

$$1.1 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m}.$$

$$1.2 \quad u_{-1} = u_0 - \Delta t \dot{u}_0 + \frac{(\Delta t)^2}{2} \ddot{u}_0.$$

$$1.3 \quad \hat{k} = \frac{m}{(\Delta t)^2} + \frac{c}{2\Delta t}.$$

$$1.4 \quad a = \frac{m}{(\Delta t)^2} - \frac{c}{2\Delta t}.$$

$$1.5 \quad b = k - \frac{2m}{(\Delta t)^2}.$$

2.0 Calculations for time step i

$$2.1 \quad \hat{p}_i = p_i - au_{i-1} - bu_i.$$

$$2.2 \quad u_{i+1} = \frac{\hat{p}_i}{\hat{k}}.$$

$$2.3 \quad \text{If required: } \dot{u}_i = \frac{u_{i+1} - u_{i-1}}{2\Delta t}; \quad \ddot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta t)^2}.$$

3.0 Repetition for the next time step

Replace i by $i + 1$ and repeat steps 2.1, 2.2, and 2.3 for the next time step.

[†] If the excitation is ground acceleration $\ddot{u}_g(t)$, according to Eq. (1.7.6), replace p_i by $-m\ddot{u}_{gi}$ in Table 5.3.1. The computed u_i , \dot{u}_i , and \ddot{u}_i give response values relative to the ground. If needed, the total velocity and acceleration can be computed readily: $\dot{u}_i^t = \dot{u}_i + \dot{u}_{gi}$ and $\ddot{u}_i^t = \ddot{u}_i + \ddot{u}_{gi}$.

The unknown u_{i+1} is then given by

$$u_{i+1} = \frac{\hat{p}_i}{\hat{k}} \quad (5.3.7)$$

The solution u_{i+1} at time $i + 1$ is determined from the equilibrium condition, Eq. (5.1.3), at time i without using the equilibrium condition, Eq. (5.1.4), at time $i + 1$. Such methods are called *explicit methods*.

Observe in Eq. (5.3.6) that known displacements u_i and u_{i-1} are used to compute u_{i+1} . Thus u_0 and u_{-1} are required to determine u_1 ; the specified initial displacement u_0 is known. To determine u_{-1} , we specialize Eq. (5.3.1) for $i = 0$ to obtain

$$\dot{u}_0 = \frac{u_1 - u_{-1}}{2\Delta t} \quad \ddot{u}_0 = \frac{u_1 - 2u_0 + u_{-1}}{(\Delta t)^2} \quad (5.3.8)$$

Solving for u_1 from the first equation and substituting in the second gives

$$u_{-1} = u_0 - \Delta t(\dot{u}_0) + \frac{(\Delta t)^2}{2} \ddot{u}_0 \quad (5.3.9)$$

The initial displacement u_0 and initial velocity \dot{u}_0 are given, and the equation of motion at time 0 ($t_0 = 0$),

$$m\ddot{u}_0 + c\dot{u}_0 + ku_0 = p_0$$

provides the acceleration at time 0:

$$\ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m} \quad (5.3.10)$$

Table 5.3.1 summarizes the above-described procedure as it might be implemented on the computer.

The central difference method will “blow up,” giving meaningless results, in the presence of numerical round-off if the time step chosen is not short enough. The specific requirement for stability is

$$\frac{\Delta t}{T_n} < \frac{1}{\pi} \quad (5.3.11)$$

This is never a constraint for SDF systems because a much smaller time step should be chosen to obtain results that are accurate. Typically, $\Delta t/T_n \leq 0.1$ to define the response adequately, and in most earthquake response analyses even a shorter time step, typically $\Delta t = 0.01$ to 0.02 sec, is chosen to define the ground acceleration $\ddot{u}_g(t)$ accurately.

Example 5.2

Solve Example 5.1 by the central difference method using $\Delta t = 0.1$ sec.

Solution

1.0 Initial calculations

$$m = 0.2533 \quad k = 10 \quad c = 0.1592$$

$$u_0 = 0 \quad \dot{u}_0 = 0$$

$$1.1 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m} = 0.$$

$$1.2 \quad u_{-1} = u_0 - (\Delta t)\dot{u}_0 + \frac{(\Delta t)^2}{2}\ddot{u}_0 = 0.$$

$$1.3 \quad \hat{k} = \frac{m}{(\Delta t)^2} + \frac{c}{2\Delta t} = 26.13.$$

$$1.4 \quad a = \frac{m}{(\Delta t)^2} - \frac{c}{2\Delta t} = 24.53.$$

$$1.5 \quad b = k - \frac{2m}{(\Delta t)^2} = -40.66.$$

2.0 Calculations for each time step

$$2.1 \quad \hat{p}_i = p_i - au_{i-1} - bu_i = p_i - 24.53u_{i-1} + 40.66u_i.$$

$$2.2 \quad u_{i+1} = \frac{\hat{p}_i}{\hat{k}} = \frac{\hat{p}_i}{26.13}.$$

3.0 Computational steps 2.1 and 2.2 are repeated for $i = 0, 1, 2, 3, \dots$ leading to Table E5.2, wherein the theoretical result (from Table E5.1a) is also included.

TABLE E5.2 NUMERICAL SOLUTION BY CENTRAL DIFFERENCE METHOD

t_i	p_i	u_{i-1}	u_i	\hat{p}_i [Eq. (2.1)]	u_{i+1} [Eq. (2.2)]	Theoretical u_{i+1}
0.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0328
0.1	5.0000	0.0000	0.0000	5.0000	0.1914	0.2332
0.2	8.6602	0.0000	0.1914	16.4419	0.6293	0.6487
0.3	10.0000	0.1914	0.6293	30.8934	1.1825	1.1605
0.4	8.6603	0.6293	1.1825	41.3001	1.5808	1.5241
0.5	5.0000	1.1825	1.5808	40.2649	1.5412	1.4814
0.6	0.0000	1.5808	1.5412	23.8809	0.9141	0.9245
0.7	0.0000	1.5412	0.9141	-0.6456	-0.0247	0.0593
0.8	0.0000	0.9141	-0.0247	-23.4309	-0.8968	-0.7751
0.9	0.0000	-0.0247	-0.8968	-35.8598	-1.3726	-1.2718
1.0	0.0000	-0.8968	-1.3726	-33.8058	-1.2940	-1.2674

5.4 NEWMARK'S METHOD

5.4.1 Basic Procedure

In 1959, N. M. Newmark developed a family of time-stepping methods based on the following equations:

$$\dot{u}_{i+1} = \dot{u}_i + [(1 - \gamma) \Delta t] \ddot{u}_i + (\gamma \Delta t) \ddot{u}_{i+1} \quad (5.4.1a)$$

$$u_{i+1} = u_i + (\Delta t) \dot{u}_i + [(0.5 - \beta)(\Delta t)^2] \ddot{u}_i + [\beta(\Delta t)^2] \ddot{u}_{i+1} \quad (5.4.1b)$$

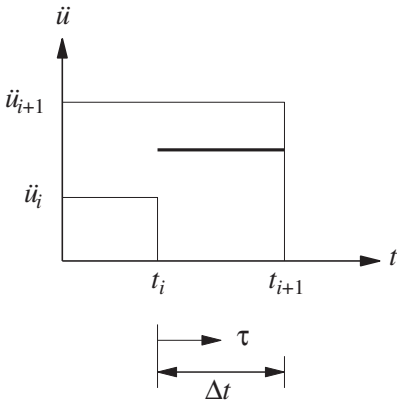
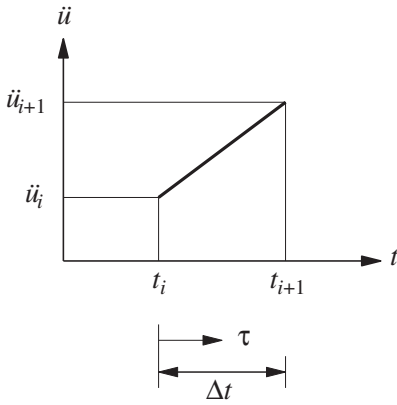
The parameters β and γ define the variation of acceleration over a time step and determine the stability and accuracy characteristics of the method. Typical selection for γ is $\frac{1}{2}$, and $\frac{1}{6} \leq \beta \leq \frac{1}{4}$ is satisfactory from all points of view, including that of accuracy. These two equations, combined with the equilibrium equation (5.1.4) at the end of the time step, provide the basis for computing u_{i+1} , \dot{u}_{i+1} , and \ddot{u}_{i+1} at time $i + 1$ from the known u_i , \dot{u}_i , and \ddot{u}_i at time i . Iteration is required to implement these computations because the unknown \ddot{u}_{i+1} appears in the right side of Eq. (5.4.1).

For linear systems it is possible to modify Newmark's original formulation, however, to permit solution of Eqs. (5.4.1) and (5.1.4) without iteration. Before describing this modification, we demonstrate that two special cases of Newmark's method are the well-known constant average acceleration and linear acceleration methods.

5.4.2 Special Cases

For these two methods, Table 5.4.1 summarizes the development of the relationship between responses u_{i+1} , \dot{u}_{i+1} , and \ddot{u}_{i+1} at time $i + 1$ to the corresponding quantities at time i . Equation (5.4.2) describes the assumptions that the variation of acceleration over a time step is constant, equal to the average acceleration, or varies linearly. Integration of $\ddot{u}(\tau)$ gives Eq. (5.4.3) for the variation $\dot{u}(\tau)$ of velocity over the time step in which

TABLE 5.4.1 AVERAGE ACCELERATION AND LINEAR ACCELERATION METHODS

Constant Average Acceleration	Linear Acceleration
	
$\ddot{u}(\tau) = \frac{1}{2}(\ddot{u}_{i+1} + \ddot{u}_i)$	$\ddot{u}(\tau) = \ddot{u}_i + \frac{\tau}{\Delta t}(\ddot{u}_{i+1} - \ddot{u}_i) \quad (5.4.2)$
$\dot{u}(\tau) = \dot{u}_i + \frac{\tau}{2}(\ddot{u}_{i+1} + \ddot{u}_i)$	$\dot{u}(\tau) = \dot{u}_i + \ddot{u}_i \tau + \frac{\tau^2}{2\Delta t}(\ddot{u}_{i+1} - \ddot{u}_i) \quad (5.4.3)$
$\dot{u}_{i+1} = \dot{u}_i + \frac{\Delta t}{2}(\ddot{u}_{i+1} + \ddot{u}_i)$	$\dot{u}_{i+1} = \dot{u}_i + \frac{\Delta t}{2}(\ddot{u}_{i+1} + \ddot{u}_i) \quad (5.4.4)$
$u(\tau) = u_i + \dot{u}_i \tau + \frac{\tau^2}{4}(\ddot{u}_{i+1} + \ddot{u}_i)$	$u(\tau) = u_i + \dot{u}_i \tau + \ddot{u}_i \frac{\tau^2}{2} + \frac{\tau^3}{6\Delta t}(\ddot{u}_{i+1} - \ddot{u}_i) \quad (5.4.5)$
$u_{i+1} = u_i + \dot{u}_i \Delta t + \frac{(\Delta t)^2}{4}(\ddot{u}_{i+1} + \ddot{u}_i)$	$u_{i+1} = u_i + \dot{u}_i \Delta t + (\Delta t)^2 \left(\frac{1}{6}\ddot{u}_{i+1} + \frac{1}{3}\ddot{u}_i \right) \quad (5.4.6)$

$\tau = \Delta t$ is substituted to obtain Eq. (5.4.4) for the velocity \dot{u}_{i+1} at time $i + 1$. Integration of $\dot{u}(\tau)$ gives Eq. (5.4.5) for the variation $u(\tau)$ of displacement over the time step in which $\tau = \Delta t$ is substituted to obtain Eq. (5.4.6) for the displacement u_{i+1} at time $i + 1$. Comparing Eqs. (5.4.4) and (5.4.6) with Eq. (5.4.1) demonstrates that Newmark's equations with $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ are the same as those derived assuming constant average acceleration, and those with $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{6}$ correspond to the assumption of linear variation of acceleration.

5.4.3 Linear Systems

For linear systems, it is possible to modify Newmark's original formulation, to permit solution of Eqs. (5.4.1) and (5.1.4) without iteration. Specialized for linear systems, Eq. (5.1.4) becomes

$$m\ddot{u}_{i+1} + c\dot{u}_{i+1} + ku_{i+1} = p_{i+1} \quad (5.4.7)$$

From Eq. (5.4.1b), \ddot{u}_{i+1} can be expressed in terms of u_{i+1} :

$$\ddot{u}_{i+1} = \frac{1}{\beta(\Delta t)^2}(u_{i+1} - u_i) - \frac{1}{\beta\Delta t}\dot{u}_i - \left(\frac{1}{2\beta} - 1\right)\ddot{u}_i \quad (5.4.8)$$

Substituting Eq. (5.4.8) into Eq. (5.4.1a) gives

$$\dot{u}_{i+1} = \frac{\gamma}{\beta \Delta t} (u_{i+1} - u_i) + \left(1 - \frac{\gamma}{\beta}\right) \dot{u}_i + \Delta t \left(1 - \frac{\gamma}{2\beta}\right) \ddot{u}_i \quad (5.4.9)$$

Next, Eqs. (5.4.8) and (5.4.9) are substituted into the governing equation (5.4.7) at time $i + 1$. This substitution gives

$$\hat{k} u_{i+1} = \hat{p}_{i+1} \quad (5.4.10)$$

where

$$\hat{k} = k + \frac{\gamma}{\beta \Delta t} c + \frac{1}{\beta (\Delta t)^2} m \quad (5.4.11)$$

and

$$\begin{aligned} \hat{p}_{i+1} = & p_{i+1} + \left[\frac{1}{\beta (\Delta t)^2} m + \frac{\gamma}{\beta \Delta t} c \right] u_i + \left[\frac{1}{\beta \Delta t} m + \left(\frac{\gamma}{\beta} - 1 \right) c \right] \dot{u}_i \\ & + \left[\left(\frac{1}{2\beta} - 1 \right) m + \Delta t \left(\frac{\gamma}{2\beta} - 1 \right) c \right] \ddot{u}_i \end{aligned} \quad (5.4.12)$$

With \hat{k} and \hat{p}_{i+1} known from the system properties m , k , and c , algorithm parameters γ and β , and the state of the system at time i defined by u_i , \dot{u}_i , and \ddot{u}_i , the displacement at time $i + 1$ is computed from

$$u_{i+1} = \frac{\hat{p}_{i+1}}{\hat{k}} \quad (5.4.13)$$

Once u_{i+1} is known, the velocity \dot{u}_{i+1} and acceleration \ddot{u}_{i+1} can be computed from Eqs. (5.4.9) and (5.4.8), respectively.

The acceleration can also be obtained from the equation of motion at time $i + 1$:

$$\ddot{u}_{i+1} = \frac{p_{i+1} - c \dot{u}_{i+1} - k u_{i+1}}{m} \quad (5.4.14)$$

rather than by Eq. (5.4.8). Equation (5.4.14) is needed to obtain \ddot{u}_0 to start the time-stepping computations [see Eq. (5.3.10)].

In Newmark's method, the solution at time $i + 1$ is determined from Eq. (5.4.7), the equilibrium condition at time $i + 1$. Such methods are called *implicit methods*. Although the resisting force is an implicit function of the unknown u_{i+1} , it was easy to calculate for linear systems.

Table 5.4.2 summarizes the time-stepping solution using Newmark's method as it might be implemented on the computer.

TABLE 5.4.2 NEWMARK'S METHOD: LINEAR SYSTEMS[†]

Special cases

(1) Constant average acceleration method ($\gamma = \frac{1}{2}, \beta = \frac{1}{4}$)(2) Linear acceleration method ($\gamma = \frac{1}{2}, \beta = \frac{1}{6}$)

1.0 Initial calculations

$$1.1 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m}.$$

1.2 Select Δt .

$$1.3 \quad a_1 = \frac{1}{\beta(\Delta t)^2}m + \frac{\gamma}{\beta\Delta t}c; \quad a_2 = \frac{1}{\beta\Delta t}m + \left(\frac{\gamma}{\beta} - 1\right)c; \quad \text{and}$$

$$a_3 = \left(\frac{1}{2\beta} - 1\right)m + \Delta t \left(\frac{\gamma}{2\beta} - 1\right)c.$$

$$1.4 \quad \hat{k} = k + a_1.$$

2.0 Calculations for each time step, $i = 0, 1, 2, \dots$

$$2.1 \quad \hat{p}_{i+1} = p_{i+1} + a_1 u_i + a_2 \dot{u}_i + a_3 \ddot{u}_i.$$

$$2.2 \quad u_{i+1} = \frac{\hat{p}_{i+1}}{\hat{k}}.$$

$$2.3 \quad \dot{u}_{i+1} = \frac{\gamma}{\beta\Delta t}(u_{i+1} - u_i) + \left(1 - \frac{\gamma}{\beta}\right)\dot{u}_i + \Delta t \left(1 - \frac{\gamma}{2\beta}\right)\ddot{u}_i.$$

$$2.4 \quad \ddot{u}_{i+1} = \frac{1}{\beta(\Delta t)^2}(u_{i+1} - u_i) - \frac{1}{\beta\Delta t}\dot{u}_i - \left(\frac{1}{2\beta} - 1\right)\ddot{u}_i.$$

3.0 Repetition for the next time step. Replace i by $i + 1$ and implement steps 2.1 to 2.4 for the next time step.

[†]If the excitation is ground acceleration $\ddot{u}_g(t)$, according to Eq. (1.7.6), replace p_i by $-m\ddot{u}_{gi}$ in Table 5.4.2. The computed u_i , \dot{u}_i , and \ddot{u}_i give response values relative to the ground. If needed, the total velocity and acceleration can be computed readily: $\dot{u}_i^t = \dot{u}_i + \dot{u}_{gi}$ and $\ddot{u}_i^t = \ddot{u}_i + \ddot{u}_{gi}$.

Newmark's method is stable if

$$\frac{\Delta t}{T_n} \leq \frac{1}{\pi\sqrt{2}} \frac{1}{\sqrt{\gamma - 2\beta}} \quad (5.4.15)$$

For $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ this condition becomes

$$\frac{\Delta t}{T_n} \leq \infty \quad (5.4.16a)$$

This implies that the constant average acceleration method is stable for any Δt , no matter how large; however, it is accurate only if Δt is small enough, as discussed at the end of

Section 5.3. For $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{6}$, Eq. (5.4.15) indicates that the linear acceleration method is stable if

$$\frac{\Delta t}{T_n} \leq 0.551 \quad (5.4.16b)$$

However, as in the case of the central difference method, this condition has little significance in the analysis of SDF systems because a much shorter time step than $0.551 T_n$ must be used to obtain an accurate representation of the excitation and response.

Example 5.3

Solve Example 5.1 by the constant average acceleration method using $\Delta t = 0.1$ sec.

Solution

1.0 Initial calculations

$$\begin{array}{lll} m = 0.2533 & k = 10 & c = 0.1592 \\ u_0 = 0 & \dot{u}_0 = 0 & p_0 = 0 \end{array}$$

$$1.1 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m} = 0.$$

$$1.2 \quad \Delta t = 0.1.$$

$$1.3 \quad a_1 = \frac{4}{(\Delta t)^2}m + \frac{2}{\Delta t}c = 104.5; \quad a_2 = \frac{4}{\Delta t}m + c = 10.29; \quad \text{and} \\ a_3 = m = 0.2533.$$

$$1.4 \quad \hat{k} = k + a_1 = 114.5.$$

2.0 Calculations for each time step, $i = 0, 1, 2, \dots$

$$2.1 \quad \hat{p}_{i+1} = p_{i+1} + a_1 u_i + a_2 \dot{u}_i + a_3 \ddot{u}_i = p_{i+1} + 104.5 u_i + 10.29 \dot{u}_i + 0.2533 \ddot{u}_i.$$

$$2.2 \quad u_{i+1} = \frac{\hat{p}_{i+1}}{\hat{k}} = \frac{\hat{p}_{i+1}}{114.5}.$$

$$2.3 \quad \dot{u}_{i+1} = \frac{2}{\Delta t}(u_{i+1} - u_i) - \dot{u}_i.$$

$$2.4 \quad \ddot{u}_{i+1} = \frac{4}{(\Delta t)^2}(u_{i+1} - u_i) - \frac{4}{\Delta t}\dot{u}_i - \ddot{u}_i.$$

3.0 Repetition for the next time step. Steps 2.1 to 2.4 are repeated for successive time steps and are summarized in Table E5.3, where the theoretical result (from Table E5.1a) is also included.

Example 5.4

Solve Example 5.1 by the linear acceleration method using $\Delta t = 0.1$ sec.

Solution

1.0 Initial calculations

$$\begin{array}{lll} m = 0.2533 & k = 10 & c = 0.1592 \\ u_0 = 0 & \dot{u}_0 = 0 & p_0 = 0 \end{array}$$

TABLE E5.3 NUMERICAL SOLUTION BY CONSTANT AVERAGE ACCELERATION METHOD

t_i	p_i	\hat{p}_i (Step 2.1)	\ddot{u}_i (Step 2.4)	\dot{u}_i (Step 2.3)	u_i (Step 2.2)	Theoretical u_i
0.0	0.0000		0.0000	0.0000	0.0000	0.0000
0.1	5.0000	5.0000	17.4666	0.8733	0.0437	0.0328
0.2	8.6603	26.6355	23.1801	2.9057	0.2326	0.2332
0.3	10.0000	70.0837	12.3719	4.6833	0.6121	0.6487
0.4	8.6603	123.9535	-11.5175	4.7260	1.0825	1.1605
0.5	5.0000	163.8469	-38.1611	2.2421	1.4309	1.5241
0.6	0.0000	162.9448	-54.6722	-2.3996	1.4230	1.4814
0.7	0.0000	110.1710	-33.6997	-6.8182	0.9622	0.9245
0.8	0.0000	21.8458	-2.1211	-8.6092	0.1908	0.0593
0.9	0.0000	-69.1988	28.4423	-7.2932	-0.6043	-0.7751
1.0	0.0000	-131.0066	47.3701	-3.5026	-1.1441	-1.2718

$$1.1 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - ku_0}{m} = 0.$$

$$1.2 \quad \Delta t = 0.1.$$

$$1.3 \quad a_1 = \frac{6}{(\Delta t)^2}m + \frac{3}{\Delta t}c = 156.8; \quad a_2 = \frac{6}{\Delta t}m + 2c = 15.52; \quad \text{and} \\ a_3 = 2m + \frac{\Delta t}{2}c = 0.5146.$$

$$1.4 \quad \hat{k} = k + a_1 = 166.8.$$

2.0 Calculations for each time step, $i = 0, 1, 2, \dots$

$$2.1 \quad \hat{p}_{i+1} = p_{i+1} + a_1 u_i + a_2 \dot{u}_i + a_3 \ddot{u}_i = p_{i+1} + 156.8u_i + 15.52\dot{u}_i + 0.5146\ddot{u}_i.$$

$$2.2 \quad u_{i+1} = \frac{\hat{p}_{i+1}}{\hat{k}} = \frac{\hat{p}_{i+1}}{166.8}.$$

$$2.3 \quad \dot{u}_{i+1} = \frac{3}{\Delta t}(u_{i+1} - u_i) - 2\dot{u}_i - \frac{\Delta t}{2}\ddot{u}_i.$$

$$2.4 \quad \ddot{u}_{i+1} = \frac{6}{(\Delta t)^2}(u_{i+1} - u_i) - \frac{6}{\Delta t}\dot{u}_i - 2\ddot{u}_i.$$

3.0 Repetition for the next time step. Steps 2.1 to 2.4 are repeated for successive time steps and are summarized in Table E5.4, where the theoretical result (from Table E5.1a) is also included.

Observe that the numerical results obtained by the linear acceleration method are closer to the theoretical solution (Table E5.4), hence more accurate, than those from the constant average acceleration method (Table E5.3).

TABLE E5.4 NUMERICAL SOLUTION BY LINEAR ACCELERATION METHOD

t_i	p_i	\hat{p}_i (Step 2.1)	\ddot{u}_i (Step 2.4)	\dot{u}_i (Step 2.3)	u_i (Step 2.2)	Theoretical u_i
0.0	0.0000		0.0000	0.0000	0.0000	0.0000
0.1	5.0000	5.0000	17.9904	0.8995	0.0300	0.0328
0.2	8.6603	36.5748	23.6566	2.9819	0.2193	0.2332
0.3	10.0000	102.8221	12.1372	4.7716	0.6166	0.6487
0.4	8.6603	185.5991	-12.7305	4.7419	1.1130	1.1605
0.5	5.0000	246.4956	-39.9425	2.1082	1.4782	1.5241
0.6	0.0000	243.8733	-56.0447	-2.6911	1.4625	1.4814
0.7	0.0000	158.6538	-33.0689	-7.1468	0.9514	0.9245
0.8	0.0000	21.2311	0.4892	-8.7758	0.1273	0.0593
0.9	0.0000	-115.9590	31.9491	-7.1539	-0.6954	-0.7751
1.0	0.0000	-203.5678	50.1114	-3.0508	-1.2208	-1.2718

5.5 STABILITY AND COMPUTATIONAL ERROR

5.5.1 Stability

Numerical procedures that lead to bounded solutions if the time step is shorter than some stability limit are called *conditionally stable procedures*. Procedures that lead to bounded solutions regardless of the time-step length are called *unconditionally stable procedures*. The average acceleration method is unconditionally stable. The linear acceleration method is stable if $\Delta t/T_n < 0.551$, and the central difference method is stable if $\Delta t/T_n < 1/\pi$. Obviously, the latter two methods are conditionally stable.

The stability criteria are not restrictive (i.e., they do not dictate the choice of time step) in the analysis of SDF systems because $\Delta t/T_n$ must be considerably smaller than the stability limit (say, 0.1 or less) to ensure adequate accuracy in the numerical results. Stability of the numerical method is important, however, in the analysis of MDF systems, where it is often necessary to use unconditionally stable methods (Chapter 16).

5.5.2 Computational Error

Error is inherent in any numerical solution of the equation of motion. We do not discuss error analysis from a mathematical point of view. Rather, we examine two important characteristics of numerical solutions to develop a feel for the nature of the errors, and then mention a simple, useful way of managing error.

Consider the free vibration problem

$$m\ddot{u} + ku = 0 \quad u(0) = 1 \quad \text{and} \quad \dot{u}(0) = 0$$

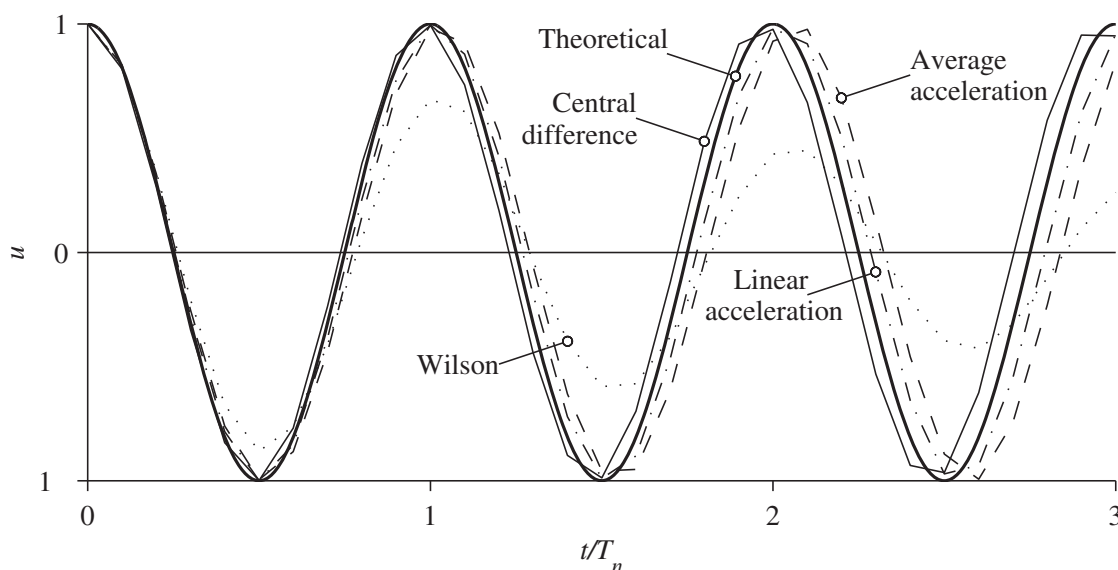


Figure 5.5.1 Free vibration solution by four numerical methods ($\Delta t/T_n = 0.1$) and the theoretical solution.

for which the theoretical solution is

$$u(t) = \cos \omega_n t \quad (5.5.1)$$

This problem is solved by four numerical methods: central difference method, average acceleration method, linear acceleration method, and Wilson's method. The last of these methods is available elsewhere; see the references at the end of the chapter. The numerical results obtained using $\Delta t = 0.1T_n$ are compared with the theoretical solution in Fig. 5.5.1. This comparison shows that some numerical methods may predict that the displacement amplitude decays with time, although the system is undamped, and that the natural period is elongated or shortened.

Figure 5.5.2 shows the amplitude decay AD and period elongation PE in the four numerical methods as a function of $\Delta t/T_n$; AD and PE are defined in part (b) of the figure. The mathematical analyses that led to these data are not presented, however. Three of the methods predict no decay of displacement amplitude. Wilson's method contains decay of amplitude, however, implying that this method introduces *numerical damping* in the system; the equivalent viscous damping ratio $\bar{\zeta}$ is shown in part (a) of the figure. Observe the rapid increase in the period error in the central difference method near $\Delta t/T_n = 1/\pi$, the stability limit for the method. The central difference method introduces the largest period error. In this sense it is the least accurate of the methods considered. For $\Delta t/T_n$ less than its stability limit, the linear acceleration method gives the least period elongation. This property, combined with no amplitude decay, makes this method the most suitable method (of the methods presented) for SDF systems. However,

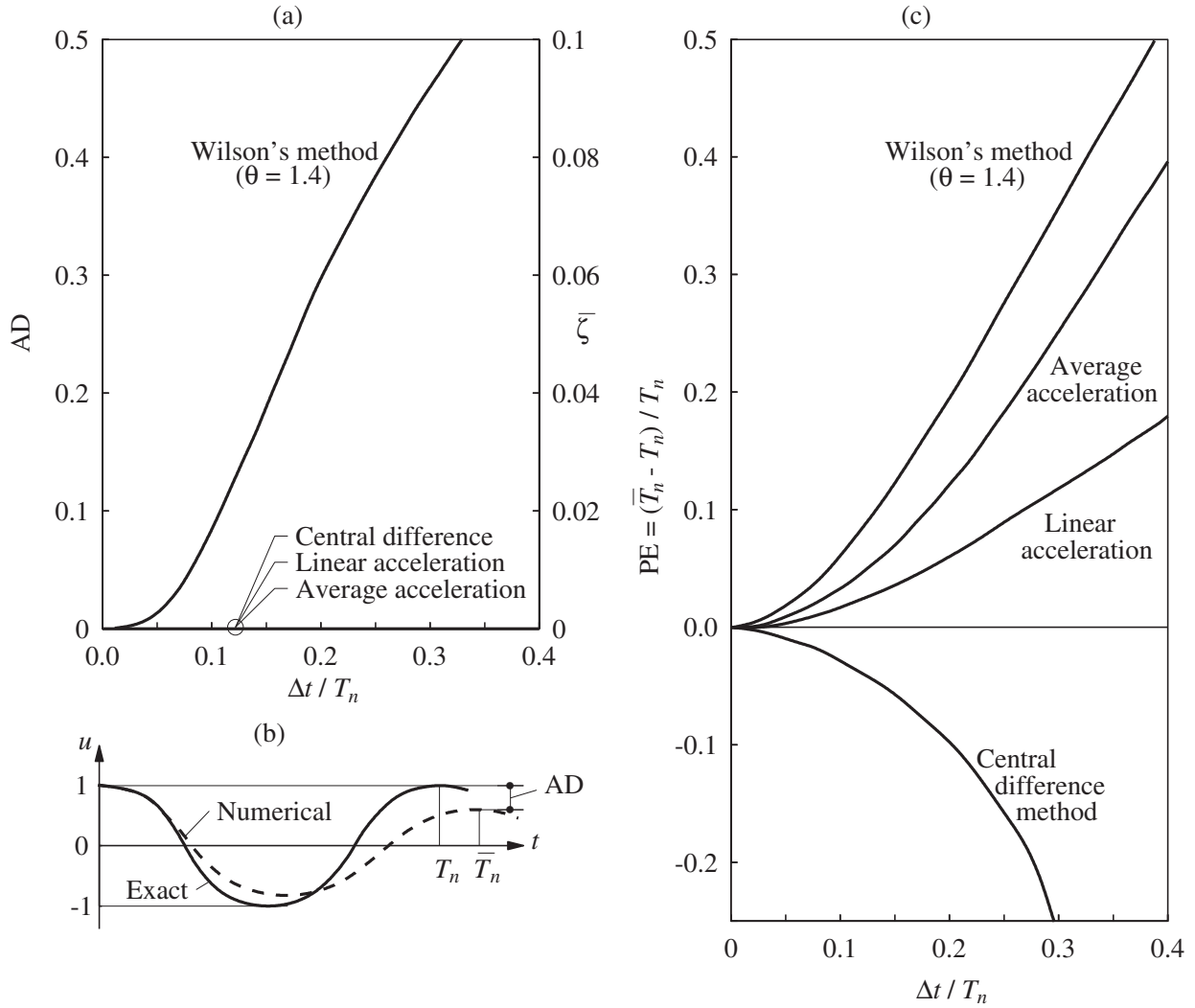


Figure 5.5.2 (a) Amplitude decay versus $\Delta t / T_n$; (b) definition of AD and PE; (c) period elongation versus $\Delta t / T_n$.

we shall arrive at a different conclusion for MDF systems because of stability requirements (Chapter 16).

The choice of time step also depends on the time variation of the dynamic excitation, in addition to the natural vibration period of the system. Figure 5.5.2 suggests that $\Delta t = 0.1T_n$ would give reasonably accurate results. The time step should also be short enough to keep the distortion of the excitation function to a minimum. A very fine time step is necessary to describe numerically the highly irregular earthquake ground acceleration recorded during earthquakes; typically, $\Delta t = 0.02$ sec and the time step chosen for computing structural response should not be longer.

One useful, although unsophisticated technique for selecting the time step is to solve the problem with a time step that seems reasonable, then repeat the solution with a slightly smaller time step and compare the results, continuing the process until two successive solutions are close enough.

The preceding discussion of stability and accuracy applies strictly to linear systems. The reader should consult other references for how these issues affect nonlinear response analysis.

5.6 NONLINEAR SYSTEMS: CENTRAL DIFFERENCE METHOD

The dynamic response of a system beyond its linearly elastic range is generally not amenable to analytical solution even if the time variation of the excitation is described by a simple function. Numerical methods are therefore essential in the analysis of nonlinear systems. The central difference method can easily be adapted for solving the nonlinear equation of motion, Eq. (5.1.3), at time i . Substituting Eqs. (5.3.1), the central difference approximation for velocity and acceleration, gives Eq. (5.3.2) with ku_i replaced by $(f_s)_i$, which can be rewritten to obtain the following expression for response at time $i + 1$:

$$\hat{k}u_{i+1} = \hat{p}_i \quad (5.6.1)$$

where

$$\hat{k} = \frac{m}{(\Delta t)^2} + \frac{c}{2\Delta t} \quad (5.6.2)$$

and

$$\hat{p}_i = p_i - \left[\frac{m}{(\Delta t)^2} - \frac{c}{2\Delta t} \right] u_{i-1} + \frac{2m}{(\Delta t)^2} u_i - (f_s)_i \quad (5.6.3)$$

Comparing these equations with those for linear systems, it is seen that the only difference is in the definition for \hat{p}_i . With this modification Table 5.3.1 also applies to nonlinear systems.

The resisting force $(f_s)_i$ appears *explicitly*, as it depends only on the response at time i , not on the unknown response at time $i + 1$. Thus it is easily calculated, making the central difference method perhaps the simplest procedure for nonlinear systems.

5.7 NONLINEAR SYSTEMS: NEWMARK'S METHOD

In this section, Newmark's method described earlier for linear systems is extended to nonlinear systems. Recall that this method determines the solution at time $i + 1$ from the equilibrium condition at time $i + 1$, i.e., Eq. (5.1.4) for nonlinear systems. Because the resisting force $(f_s)_{i+1}$ is an implicit nonlinear function of the unknown u_{i+1} , iteration is required in this method. This requirement is typical of implicit methods. It is instructive first to develop the Newton-Raphson method of iteration for static analysis of a nonlinear SDF system.

5.7.1 Newton–Raphson Iteration

Dropping the inertia and damping terms in the equation of motion [Eq. (5.1.1)] gives the nonlinear equation to be solved in a static problem:

$$f_S(u) = p \quad (5.7.1)$$

The task is to determine the deformation u due to a given external force p , where the nonlinear force–deformation relation $f_S(u)$ is defined for the system to be analyzed.

Suppose that after j cycles of iteration, $u^{(j)}$ is an estimate of the unknown displacement and we are interested in developing an iterative procedure that provides an improved estimate $u^{(j+1)}$. For this purpose, expanding the resisting force $f_S^{(j+1)}$ in Taylor series about the known estimate $u^{(j)}$ gives

$$f_S^{(j+1)} = f_S^{(j)} + \left. \frac{\partial f_S}{\partial u} \right|_{u^{(j)}} (u^{(j+1)} - u^{(j)}) + \frac{1}{2} \left. \frac{\partial^2 f_S}{\partial u^2} \right|_{u^{(j)}} (u^{(j+1)} - u^{(j)})^2 + \dots \quad (5.7.2)$$

If $u^{(j)}$ is close to the solution, the change in u , $\Delta u^{(j)} = u^{(j+1)} - u^{(j)}$, will be small and the second- and higher-order terms can be neglected, leading to the linearized equation

$$f_S^{(j+1)} \simeq f_S^{(j)} + k_T^{(j)} \Delta u^{(j)} = p \quad (5.7.3)$$

or

$$k_T^{(j)} \Delta u^{(j)} = p - f_S^{(j)} = R^{(j)} \quad (5.7.4)$$

where $k_T^{(j)} = \left. \frac{\partial f_S}{\partial u} \right|_{u^{(j)}}$ is the tangent stiffness at $u^{(j)}$. Solving the linearized equation (5.7.4) gives $\Delta u^{(j)}$ and an improved estimate of the displacement:

$$u^{(j+1)} = u^{(j)} + \Delta u^{(j)} \quad (5.7.5)$$

The iterative procedure is described next with reference to Fig. 5.7.1. Associated with $u^{(j)}$ is the force $f_S^{(j)}$, which is not equal to the applied force p , and a residual force is defined: $R^{(j)} = p - f_S^{(j)}$. The additional displacement due to this residual force is determined from Eq. (5.7.4), leading to $u^{(j+1)}$. This new estimate of the solution is used to find a new value of the residual force $R^{(j+1)} = p - f_S^{(j+1)}$. The additional displacement $\Delta u^{(j+1)}$ due to this residual force is determined by solving

$$k_T^{(j+1)} \Delta u^{(j+1)} = R^{(j+1)} \quad (5.7.6)$$

This additional displacement is used to find a new value of the displacement:

$$u^{(j+2)} = u^{(j+1)} + \Delta u^{(j+1)} \quad (5.7.7)$$

and a new value of the residual force $R^{(j+2)}$, and the process is continued until convergence is achieved. This iterative process is known as the *Newton–Raphson method*.

Convergence rate. It can be proven that near the end of the iteration process the Newton–Raphson algorithm converges with quadratic rate to the exact solution u , i.e., $|u - u^{(j+1)}| \leq c |u - u^{(j)}|^2$, where c is a constant that depends on the second derivative

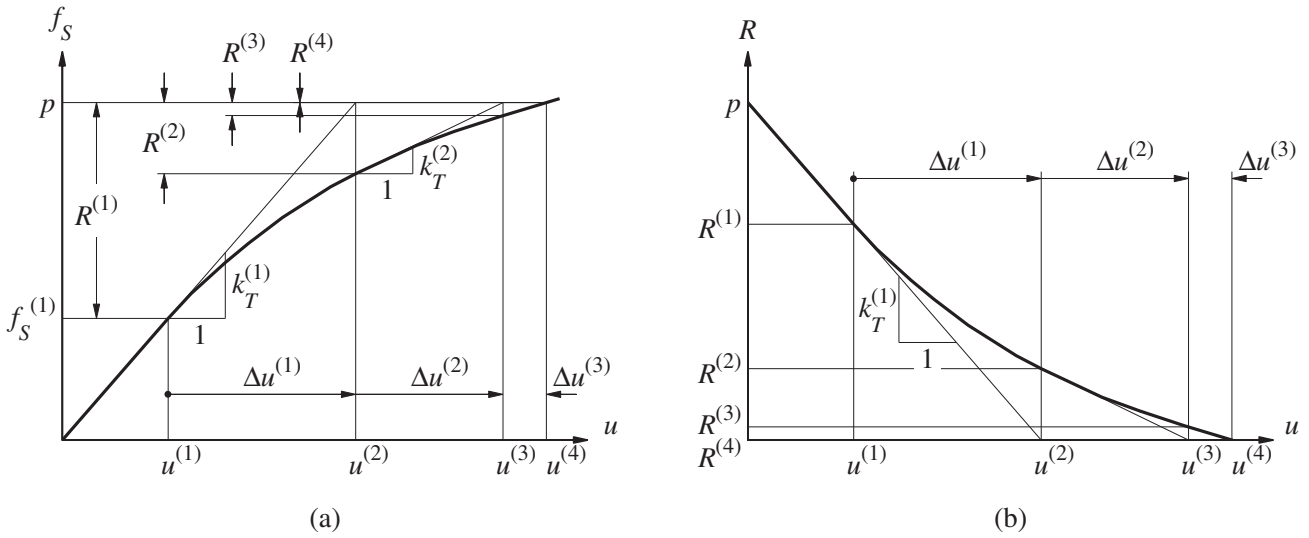


Figure 5.7.1 Newton-Raphson iteration: (a) applied and resisting forces; (b) residual force.

of the resisting force or the change in tangent stiffness. This result implies that near the solution the error in the $(j + 1)$ th iterate (equal to the difference between u and $u^{(j+1)}$) is less than the square of the error in the previous iterate $u^{(j)}$.

Convergence criteria. After each iteration the solution is checked and the iterative process is terminated when some measure of the error in the solution is less than a specified tolerance. Typically, one or more of the following convergence (or acceptance) criteria are enforced:

1. Residual force is less than a tolerance:

$$|R^{(j)}| \leq \varepsilon_R \quad (5.7.8a)$$

Conventional values for the tolerance ε_R range from 10^{-3} to 10^{-8} .

2. Change in displacement is less than a tolerance:

$$|\Delta u^{(j)}| \leq \varepsilon_u \quad (5.7.8b)$$

Conventional values for the tolerance ε_u range from 10^{-3} to 10^{-8} .

3. Incremental work done by the residual force acting through the change in displacement is less than a tolerance:

$$\frac{1}{2} |\Delta u^{(j)} R^{(j)}| \leq \varepsilon_w \quad (5.7.8c)$$

Tolerance ε_w must be at or near the computer (machine) tolerance because the left side is a product of small quantities.

Modified Newton-Raphson iteration. To avoid computation of the tangent stiffness for each iteration, the initial stiffness at the beginning of a time step may be used

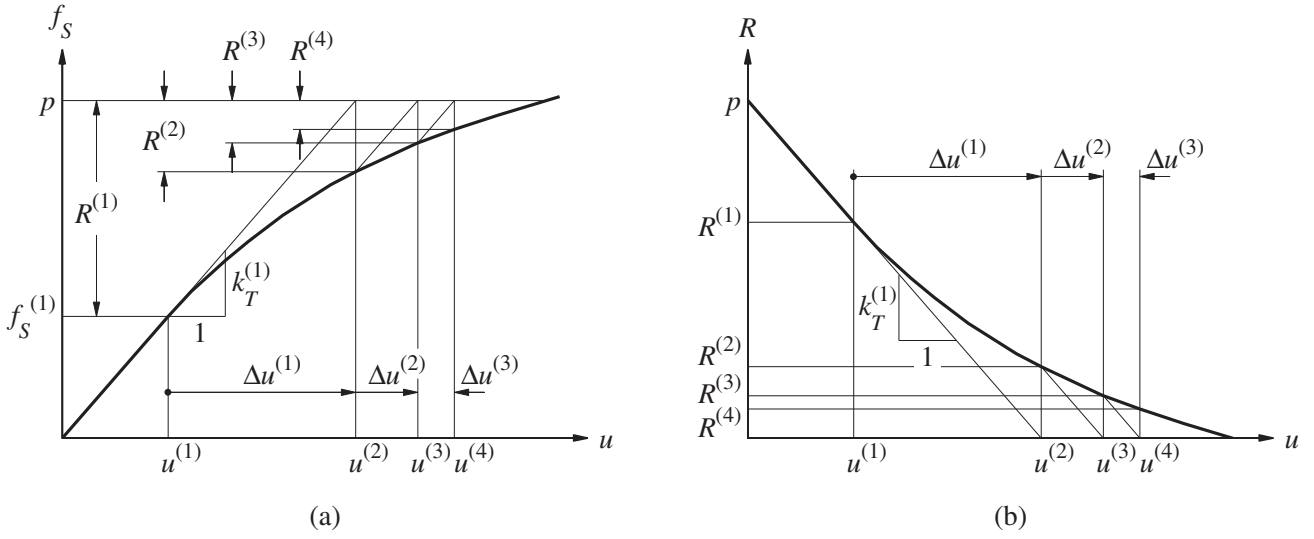


Figure 5.7.2 Modified Newton–Raphson iteration: (a) applied and resisting forces; (b) residual force.

as the constant stiffness for all iterations within the time step. This modified Newton–Raphson iteration is illustrated in Fig. 5.7.2, where it can be seen that convergence is now slower. At each iteration the residual force $R^{(j)}$ is now larger, as seen by comparing Figs. 5.7.1 and 5.7.2, and more iterations will be required to achieve convergence.

5.7.2 Newmark’s Method

We have now developed Newton–Raphson iteration to solve a nonlinear equilibrium equation [e.g., Eq. (5.7.1)] that governs the static problem. In dynamic analysis the goal is to determine response quantities u_{i+1} , \dot{u}_{i+1} and \ddot{u}_{i+1} at time $i + 1$ that satisfy Eq. (5.1.4), which can be written as

$$(\hat{f}_s)_{i+1} = p_{i+1} \quad (5.7.9)$$

where

$$(\hat{f}_s)_{i+1} = m\ddot{u}_{i+1} + c\dot{u}_{i+1} + (f_s)_{i+1} \quad (5.7.10)$$

By including the inertia and damping forces in defining the “resisting force” \hat{f}_s , the dynamic analysis equation (5.7.9) is of the same form as the static analysis equation (5.7.1).

Thus, we can adapt the Taylor series expansion of Eq. (5.7.2) to Eq. (5.7.9), interpret $(\hat{f}_s)_{i+1}$ as a function of u_{i+1} , and drop the second- and higher-order terms to obtain an equation analogous to Eq. (5.7.3):

$$(\hat{f}_s)_{i+1}^{(j+1)} \simeq (\hat{f}_s)_{i+1}^{(j)} + \frac{\partial \hat{f}_s}{\partial u_{i+1}} \Delta u^{(j)} = p_{i+1} \quad (5.7.11)$$

where

$$\Delta u^{(j)} = u_{i+1}^{(j+1)} - u_{i+1}^{(j)} \quad (5.7.12)$$

Differentiating Eq. (5.7.10) at the known displacement $u_{i+1}^{(j)}$ gives

$$\frac{\partial \hat{f}_s}{\partial u_{i+1}} = m \frac{\partial \ddot{u}}{\partial u_{i+1}} + c \frac{\partial \dot{u}}{\partial u_{i+1}} + \frac{\partial f_s}{\partial u_{i+1}}$$

where the derivatives in inertia and damping terms on the right side can be determined from Eqs. (5.4.8) and (5.4.9), respectively, which were derived from Newmark's equation (5.4.1):

$$\frac{\partial \ddot{u}}{\partial u_{i+1}} = \frac{1}{\beta(\Delta t)^2} \quad \frac{\partial \dot{u}}{\partial u_{i+1}} = \frac{\gamma}{\beta \Delta t}$$

Putting together the preceding two equations and recalling the definition of tangent stiffness (Section 5.7.1) gives

$$(\hat{k}_T)_{i+1}^{(j)} \equiv \frac{\partial \hat{f}_s}{\partial u_{i+1}} = (\hat{k}_T)_{i+1}^{(j)} + \frac{\gamma}{\beta \Delta t} c + \frac{1}{\beta(\Delta t)^2} m \quad (5.7.13)$$

With the preceding definition of $(\hat{k}_T)_{i+1}^{(j)}$, Eq. (5.7.11) can be written as

$$(\hat{k}_T)_{i+1}^{(j)} \Delta u^{(j)} = p_{i+1} - (\hat{f}_s)_{i+1}^{(j)} \equiv \hat{R}_{i+1}^{(j)} \quad (5.7.14)$$

Substituting Eqs. (5.4.8) and (5.4.9) in Eq. (5.7.10) and then combining it with the right side of Eq. (5.7.14) leads to the following expression for the residual force:

$$\begin{aligned} \hat{R}_{i+1}^{(j)} = & p_{i+1} - (f_s)_{i+1}^{(j)} - \left[\frac{1}{\beta(\Delta t)^2} m + \frac{\gamma}{\beta \Delta t} c \right] (u_{i+1}^{(j)} - u_i) + \left[\frac{1}{\beta \Delta t} m + \left(\frac{\gamma}{\beta} - 1 \right) c \right] \dot{u}_i \\ & + \left[\left(\frac{1}{2\beta} - 1 \right) m + \Delta t \left(\frac{\gamma}{2\beta} - 1 \right) c \right] \ddot{u}_i \end{aligned} \quad (5.7.15)$$

Note that the linearized equation (5.7.14) for the j th iteration in dynamic analysis is similar in form to the corresponding equation (5.7.4) in static analysis. However, there is an important difference in the two equations in that damping and inertia terms are now included in both the tangent stiffness \hat{k}_T (Eq. 5.7.13) and the residual force \hat{R} (Eq. 5.7.15). The first, fourth, and fifth terms on the right side of Eq. (5.7.15) do not change from one iteration to the next. The second and third terms need to be updated with every new estimate of displacement $u_{i+1}^{(j)}$ during iteration.

Equation (5.7.14) provides the basis for the Newton–Raphson iteration method, summarized in step 3.0 of Table 5.7.1. Once u_{i+1} is determined, the rest of the computation proceeds as for linear systems; in particular, \ddot{u}_{i+1} and \dot{u}_{i+1} are determined from Eqs. (5.4.8) and (5.4.9), respectively. Table 5.7.1 summarizes Newmark's algorithm as it might be implemented on the computer.

TABLE 5.7.1 NEWMARK'S METHOD: NONLINEAR SYSTEMS

Special cases

(1) Average acceleration method ($\gamma = \frac{1}{2}$, $\beta = \frac{1}{4}$)(2) Linear acceleration method ($\gamma = \frac{1}{2}$, $\beta = \frac{1}{6}$)**1.0 Initial calculations**1.1 State determination: $(f_S)_0$ and $(k_T)_0$.

$$1.2 \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - (f_S)_0}{m}.$$

1.3 Select Δt .

$$1.4 \quad a_1 = \frac{1}{\beta(\Delta t)^2}m + \frac{\gamma}{\beta\Delta t}c; \quad a_2 = \frac{1}{\beta\Delta t}m + \left(\frac{\gamma}{\beta} - 1\right)c; \quad \text{and}$$

$$a_3 = \left(\frac{1}{2\beta} - 1\right)m + \Delta t \left(\frac{\gamma}{2\beta} - 1\right)c.$$

2.0 Calculations for each time instant, $i = 0, 1, 2, \dots$ 2.1 Initialize $j = 1$, $u_{i+1}^{(j)} = u_i$, $(f_S)_{i+1}^{(j)} = (f_S)_i$, and $(k_T)_{i+1}^{(j)} = (k_T)_i$.

$$2.2 \quad \hat{p}_{i+1} = p_{i+1} + a_1 u_i + a_2 \dot{u}_i + a_3 \ddot{u}_i.$$

3.0 For each iteration, $j = 1, 2, 3, \dots$

$$3.1 \quad \hat{R}_{i+1}^{(j)} = \hat{p}_{i+1} - (f_S)_{i+1}^{(j)} - a_1 u_{i+1}^{(j)}.$$

3.2 Check convergence; If the acceptance criteria are not met, implement steps 3.3 to 3.7; otherwise, skip these steps and go to step 4.0.

$$3.3 \quad (\hat{k}_T)_{i+1}^{(j)} = (k_T)_{i+1}^{(j)} + a_1.$$

$$3.4 \quad \Delta u^{(j)} = \hat{R}_{i+1}^{(j)} \div (\hat{k}_T)_{i+1}^{(j)}.$$

$$3.5 \quad u_{i+1}^{(j+1)} = u_{i+1}^{(j)} + \Delta u^{(j)}.$$

3.6 State determination: $(f_S)_{i+1}^{(j+1)}$ and $(k_T)_{i+1}^{(j+1)}$.Replace j by $j + 1$ and repeat steps 3.1 to 3.6; denote final value as u_{i+1} .**4.0 Calculations for velocity and acceleration**

$$4.1 \quad \dot{u}_{i+1} = \frac{\gamma}{\beta\Delta t}(u_{i+1} - u_i) + \left(1 - \frac{\gamma}{\beta}\right)\dot{u}_i + \Delta t \left(1 - \frac{\gamma}{2\beta}\right)\ddot{u}_i.$$

$$4.2 \quad \ddot{u}_{i+1} = \frac{1}{\beta(\Delta t)^2}(u_{i+1} - u_i) - \frac{1}{\beta\Delta t}\dot{u}_i - \left(\frac{1}{2\beta} - 1\right)\ddot{u}_i.$$

5.0 Repetition for next time step. Replace i by $i + 1$ and implement steps 2.0 to 4.0 for the next time step.**Example 5.5**

An SDF system has the same properties as in Example 5.1, except that the restoring force–deformation relation is elastoplastic with yield deformation $u_y = 0.75$ in. and yield force $f_y = 7.5$ kips (Fig. E5.5). Determine the response $u(t)$ of this system (starting from rest) to the half-cycle sine pulse force in Fig. E5.1 using the constant average acceleration method with $\Delta t = 0.1$ sec and Newton–Raphson iteration.

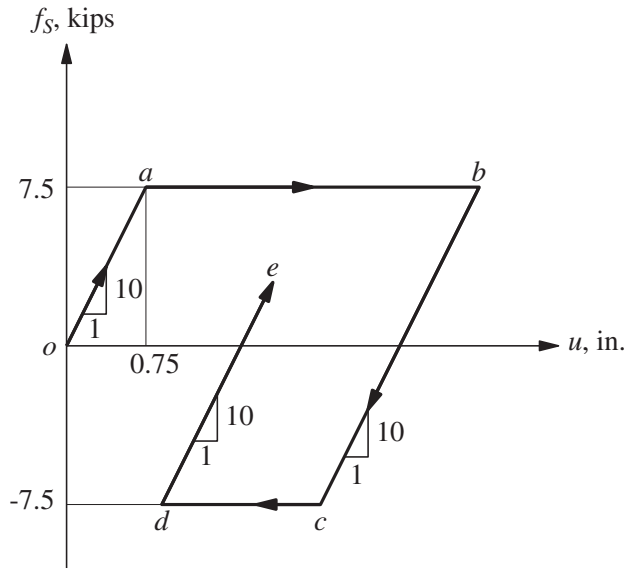


Figure E5.5

Solution**1.0 Initial calculations**

$$m = 0.2533 \quad k = 10 \quad c = 0.1592$$

$$u_0 = 0 \quad \dot{u}_0 = 0 \quad p_0 = 0$$

1.1 State determination: $(f_S)_0 = 0$ and $(k_T)_0 = k = 10$.

$$\mathbf{1.2} \quad \ddot{u}_0 = \frac{p_0 - c\dot{u}_0 - (f_S)_0}{m} = 0.$$

1.3 $\Delta t = 0.1$.

$$\mathbf{1.4} \quad a_1 = \frac{4}{(\Delta t)^2}m + \frac{2}{\Delta t}c = 104.5040; \quad a_2 = \frac{4}{\Delta t}m + c = 10.2912; \quad \text{and} \\ a_3 = m = 0.2533.$$

As an example, the calculations of steps 2.0, 3.0 and 4.0 in Table 5.7.1, are implemented as follows for the time step that begins at 0.3 sec and ends at 0.4 sec.

2.0 Calculations for $i = 3$

2.1 Initialize $j = 1$

$$u_{i+1}^{(1)} = u_i = 0.6121, (f_S)_{i+1}^{(1)} = (f_S)_i = 6.1206, \text{ and } (k_T)_{i+1}^{(1)} = (k_T)_i = 10.$$

$$\mathbf{2.2} \quad \hat{p}_{i+1} = p_{i+1} + 104.5u_i + 10.29\dot{u}_i + 0.2533\ddot{u}_i = 123.9535.$$

3.0 First iteration, $j = 1$

$$\mathbf{3.1} \quad \hat{R}_{i+1}^{(1)} = \hat{p}_{i+1} - (f_S)_{i+1}^{(1)} - 104.5u_{i+1}^{(1)} \\ = 123.9535 - 6.1206 - 63.9630 = 53.8698.$$

3.2 Check of convergence: Because $|\hat{R}_{i+1}^{(1)}| = 53.8698$ exceeds $\varepsilon_R = 10^{-3}$, chosen for this example, implement steps 3.3 to 3.7.

$$3.3 \quad (\hat{k}_T)_{i+1}^{(1)} = (k_T)_{i+1}^{(1)} + a_1 = 10 + 104.5040 = 114.5040.$$

$$3.4 \quad \Delta u^{(1)} = \hat{R}_{i+1}^{(1)} \div (\hat{k}_T)_{i+1}^{(1)} = 53.8698 \div 114.5040 = 0.4705.$$

$$3.5 \quad u_{i+1}^{(2)} = u_{i+1}^{(1)} + \Delta u^{(1)} = 0.6121 + 0.4705 = 1.0825.$$

$$3.6 \quad \text{State determination: } (f_S)_{i+1}^{(2)} \text{ and } (k_T)_{i+1}^{(2)}$$

$$(f_S)_{i+1}^{(2)} = (f_S)_i + k(u_{i+1}^{(2)} - u_i) = 6.1206 + (10 \times 0.4705) = 10.8253.$$

$$\text{Because } (f_S)_{i+1}^{(2)} > f_y, (f_S)_{i+1}^{(2)} = f_y = 7.5 \text{ and } (k_T)_{i+1}^{(2)} = 0.$$

3.0 *Second iteration, $j = 2$*

$$3.1 \quad \hat{R}_{i+1}^{(2)} = \hat{p}_{i+1} - (f_S)_{i+1}^{(2)} - 104.5u_{i+1}^{(2)}$$

$$= 123.9535 - 7.5 - 113.1282 = 3.3253.$$

$$3.2 \quad \text{Check of convergence: Because } \left| \hat{R}_{i+1}^{(2)} \right| = 3.3253 \text{ exceeds } \varepsilon_R, \text{ implement steps 3.3 to 3.7.}$$

$$3.3 \quad (\hat{k}_T)_{i+1}^{(2)} = (k_T)_{i+1}^{(2)} + a_1 = 0 + 104.5040 = 104.5040.$$

$$3.4 \quad \Delta u^{(2)} = \hat{R}_{i+1}^{(2)} \div (\hat{k}_T)_{i+1}^{(2)} = 3.3253 \div 104.5040 = 0.0318.$$

$$3.5 \quad u_{i+1}^{(3)} = u_{i+1}^{(2)} + \Delta u^{(2)} = 1.0825 + 0.0318 = 1.1143.$$

$$3.6 \quad \text{State determination: } (f_S)_{i+1}^{(3)} \text{ and } (k_T)_{i+1}^{(3)}$$

$$(f_S)_{i+1}^{(3)} = (f_S)_i + k(u_{i+1}^{(3)} - u_i) = 6.1206 + (10 \times 0.5023) = 11.1434.$$

$$\text{Because } (f_S)_{i+1}^{(3)} > f_y, (f_S)_{i+1}^{(3)} = f_y = 7.5 \text{ and } (k_T)_{i+1}^{(3)} = 0.$$

3.0 *Third iteration, $j = 3$*

$$3.1 \quad \hat{R}_{i+1}^{(3)} = \hat{p}_{i+1} - (f_S)_{i+1}^{(3)} - 104.5u_{i+1}^{(3)}$$

$$= 123.9535 - 7.5 - 116.4535 = 0.$$

$$3.2 \quad \text{Check of convergence: Because } \left| \hat{R}_{i+1}^{(3)} \right| = 0 \text{ is less than } \varepsilon_R, \text{ skip steps 3.3 to 3.7;}$$

$$\text{set } u_4 = u_4^{(3)} = 1.1143.$$

4.0 *Calculations for velocity and acceleration*

$$4.1 \quad \dot{u}_{i+1} = \frac{2}{\Delta t}(u_{i+1} - u_i) - \dot{u}_i = \frac{2}{0.1}(1.1143 - 0.6121) - 4.683 = 5.3624.$$

$$4.2 \quad \ddot{u}_{i+1} = \frac{4}{(\Delta t)^2}(u_{i+1} - u_i) - \frac{4}{\Delta t}\dot{u}_i - \ddot{u}_i$$

$$= \frac{4}{(0.1)^2}(1.1143 - 0.6121) - \frac{4}{0.1}4.6833 - 12.3719 = 1.2103.$$

These calculations for the time step 0.3 to 0.4 sec are summarized in Table E5.5.

5.0 *Repetition for next time step.* After replacing i by $i + 1$, steps 2.0 to 4.0 are repeated for successive time steps and are summarized in Table E5.5.

TABLE E5.5 NUMERICAL SOLUTION BY CONSTANT AVERAGE ACCELERATION METHOD WITH NEWTON–RAPHSOON ITERATION

t_i	p_i	\hat{R}_i or $\hat{R}_i^{(j)}$	$(k_T)_i$ or $(k_T)_i^{(j)}$	$(\hat{k}_T)_i$ or $(\hat{k}_T)_i^{(j)}$	$\Delta u^{(j)}$	u_i or $u_i^{(j+1)}$	$(f_S)_i^{(j+1)}$	\dot{u}_i	\ddot{u}_i
0.0	0.0000		10			0.0000		0.0000	0.0000
0.1	5.0000	5.0000	10	114.504	0.0437	0.0437	0.4367	0.8733	17.4666
0.2	8.6603	21.6355	10	114.504	0.1889	0.2326	2.3262	2.9057	23.1801
0.3	10.0000	43.4481	10	114.504	0.3794	0.6121	6.1206	4.6833	12.3719
0.4	8.6603	53.8698	10	114.504	0.4705	1.0825	7.5000		
		3.3253	0	104.504	0.0318	1.1143	7.5000	5.3624	1.2103
0.5	5.0000	55.9918	0	104.504	0.5071	1.6214	7.5000	4.7792	−12.8735
0.6	0.0000	38.4230	0	104.504	0.3677	1.9891	7.5000	2.5742	−31.2270
0.7	0.0000	11.0816	0	104.504	0.1060	2.0951	7.5000	−0.4534	−29.3242
0.8	0.0000	−19.5936	0	104.504	−0.1875	1.9076	5.6251		
		1.8749	10	114.504	0.0164	1.9240	5.7888	−2.9690	−20.9876
0.9	0.0000	−41.6593	10	114.504	−0.3638	1.5602	2.1506	−4.3075	−5.7830
1.0	0.0000	−47.9448	10	114.504	−0.4187	1.1415	−2.0366	−4.0668	10.5962

During the next three time steps (after 0.4 sec), the system is on the yielding branch *ab*. In other words, the stiffness $\hat{k}_i = 0$ remains constant, and no iteration is necessary. Between 0.6 and 0.7 sec the velocity changes sign from positive to negative, implying that the deformation begins to decrease, the system begins to unload along the branch *bc*, and the stiffness $\hat{k}_i = 10$. However, we have ignored this change during the time step, implying that the system stays on the branch *ab* and no iteration is necessary.

The computation for the time step starting at 0.6 sec can be made more accurate by finding, by a process of iteration, the time instant at which $\dot{u} = 0$. Then the calculations can be carried out with stiffness $\hat{k}_i = 0$ over the first part of the time step and with $\hat{k}_i = 10$ over the second part of the time step. Alternatively, a smaller time step can be used for improved accuracy.

Note that the solution over a time step is not exact because equilibrium is satisfied only at the beginning and end of the time step, not at all time instants within the time step. This implies that the energy balance equation (Chapter 7) is violated. The discrepancy in energy balance, usually calculated at the end of the excitation, is an indication of the error in the numerical solution.

Example 5.6

Repeat Example 5.5 using modified Newton–Raphson iteration within each time step of $\Delta t = 0.1$ sec.

Solution The procedure of Table 5.7.1 is modified to use the initial stiffness at the beginning of a time step as the constant stiffness for all iterations within the time step. The computations in steps 1.0 and 2.0 are identical to those presented in Example 5.5, but step 3.0 is now different. To illustrate these differences, step 3.0 in the modified Table 5.7.1 is implemented for the time step that begins at 0.3 sec and ends at 0.4 sec.

3.0 First iteration, $j = 1$

$$\begin{aligned} \mathbf{3.1} \quad \hat{R}_{i+1}^{(1)} &= \hat{p}_{i+1} - (fs)_{i+1}^{(1)} - 104.5 u_{i+1}^{(1)} \\ &= 123.9535 - 6.1206 - 63.9630 = 53.8698. \end{aligned}$$

3.2 Check of convergence: Because $\left| \hat{R}_{i+1}^{(1)} \right| = 53.8698$ exceeds ε_R , implement steps 3.3 to 3.7.

$$\mathbf{3.3} \quad (\hat{k}_T)_{i+1} = (k_T)_{i+1} + a_1 = 10 + 104.5040 = 114.5040.$$

$$\mathbf{3.4} \quad \Delta u^{(1)} = \hat{R}_{i+1}^{(1)} \div (\hat{k}_T)_{i+1} = 53.8698 \div 114.5040 = 0.4705.$$

$$\mathbf{3.5} \quad u_{i+1}^{(2)} = u_{i+1}^{(1)} + \Delta u^{(1)} = 0.6121 + 0.4705 = 1.0825.$$

3.6 State determination: $(fs)_{i+1}^{(2)}$

$$(fs)_{i+1}^{(2)} = (fs)_i + k \left(u_{i+1}^{(2)} - u_i \right) = 6.1206 + (10 \times 0.4705) = 10.8253.$$

$$\text{Because } (fs)_{i+1}^{(2)} > f_y, (fs)_{i+1}^{(2)} = f_y = 7.5.$$

3.0 Second iteration $j = 2$

$$\begin{aligned} \mathbf{3.1} \quad \hat{R}_{i+1}^{(2)} &= \hat{p}_{i+1} - (fs)_{i+1}^{(2)} - 104.5 u_{i+1}^{(2)} \\ &= 123.9535 - 7.5 - 113.1282 = 3.3253. \end{aligned}$$

3.2 Check of convergence: Because $\left| \hat{R}_{i+1}^{(2)} \right| = 3.3253$ exceeds ε_R , implement steps 3.3 to 3.7.

$$\mathbf{3.3} \quad (\hat{k}_T)_{i+1} = 114.5040.$$

$$\mathbf{3.4} \quad \Delta u^{(2)} = \hat{R}_{i+1}^{(2)} \div (\hat{k}_T)_{i+1} = 3.3253 \div 114.5040 = 0.0290.$$

$$\mathbf{3.5} \quad u_{i+1}^{(3)} = u_{i+1}^{(2)} + \Delta u^{(2)} = 1.0825 + 0.0290 = 1.1116.$$

3.6 State determination: $(fs)_{i+1}^{(3)}$

$$(fs)_{i+1}^{(3)} = (fs)_i + k(u_{i+1}^{(3)} - u_i) = 6.1206 + (10 \times 0.5000) = 11.1157.$$

$$\text{Because } (fs)_{i+1}^{(3)} > f_y, (fs)_{i+1}^{(3)} = f_y = 7.5.$$

3.0 Third iteration, $j = 3$

$$\begin{aligned} \mathbf{3.1} \quad \hat{R}_{i+1}^{(3)} &= \hat{p}_{i+1} - (fs)_{i+1}^{(3)} - 104.5 u_{i+1}^{(3)} \\ &= 123.9535 - 7.5 - 116.1631 = 0.2904. \end{aligned}$$

3.2 Check of convergence: Because $\left| \hat{R}_{i+1}^{(3)} \right| = 0.2904$ exceeds ε_R , implement steps 3.3 to 3.7.

$$\mathbf{3.3} \quad (\hat{k}_T)_{i+1} = 114.5040.$$

$$\mathbf{3.4} \quad \Delta u^{(3)} = \hat{R}_{i+1}^{(3)} \div (\hat{k}_T)_{i+1} = 0.2904 \div 114.5040 = 0.0025.$$

$$3.5 \quad u_{i+1}^{(4)} = u_{i+1}^{(3)} + \Delta u^{(3)} = 1.1116 + 0.0025 = 1.1141.$$

$$3.6 \quad \text{State determination: } (fs)_{i+1}^{(4)}$$

$$(fs)_{i+1}^{(4)} = (fs)_i + k(u_{i+1}^{(4)} - u_i) = 6.1206 + (10 \times 0.5020) = 11.1410.$$

$$\text{Because } (fs)_{i+1}^{(4)} > f_y, (fs)_{i+1}^{(4)} = f_y = 7.5.$$

These calculations and those for additional iterations during the time step 0.3 to 0.4 sec are shown in Table E5.6.

TABLE E5.6 NUMERICAL SOLUTION BY CONSTANT AVERAGE ACCELERATION METHOD WITH MODIFIED NEWTON–RAPHSON ITERATION

t_i	p_i	\hat{R}_i or $\hat{R}_i^{(j)}$	$(k_T)_i$ or $(k_T)_i^{(j)}$	$(\hat{k}_T)_i$ or $(\hat{k}_T)_i^{(j)}$	$\Delta u^{(j)}$	u_i or $u_i^{(j+1)}$	$(fs)_i^{(j+1)}$	\dot{u}_i	\ddot{u}_i
0.0	0.0000		10			0.0000		0.0000	0.0000
0.1	5.0000	5.0000	10	114.504	0.0437	0.0437	0.4367	0.8733	17.4666
0.2	8.6603	21.6355	10	114.504	0.1889	0.2326	2.3262	2.9057	23.1801
0.3	10.0000	43.4481	10	114.504	0.3794	0.6121	6.1206	4.6833	12.3719
0.4	8.6603	53.8698	10	114.504	0.4705	1.0825	7.5000		
		3.3253			0.02904	1.1116	7.5000		
		0.2904			2.536E-3	1.1141	7.5000		
		2.536E-2			2.215E-4	1.1143	7.5000		
		2.215E-3			1.934E-5	1.1143	7.5000	5.3623	1.2095
0.5	5.0000	55.9912	0	104.504	0.5071	1.6214	7.5000	4.7791	−12.8734
0.6	0.0000	38.4222	0	104.504	0.3677	1.9891	7.5000	2.5741	−31.2270
0.7	0.0000	11.0810	0	104.504	0.1060	2.0951	7.5000	−0.4534	−29.3242
0.8	0.0000	−19.5936	0	104.504	−0.1875	1.9076	5.6250		
		1.8750			1.794E-2	1.9256	5.8044		
		−0.1794			−1.717E-3	1.9238	5.7873		
		1.717E-2			1.643E-4	1.9240	5.7889		
		1.643E-3			−1.572E-5	1.9240	5.7888	−2.9690	−20.9879
0.9	0.0000	−41.6600	10	114.504	−0.3638	1.5602	2.1505	−4.3076	−5.7824
1.0	0.0000	−47.9451	10	114.504	−0.4187	1.1414	−2.0367	−4.0668	10.5969

The original Newton–Raphson iteration converges more rapidly than the modified Newton–Raphson iteration, as is apparent by comparing Tables E5.5 and E5.6 that summarize results from the two methods, respectively. Observe the following: (1) The results of the first iteration are identical in the two cases because both use the initial tangent stiffness. Consequently, the resisting force $(fs)_{i+1}^{(2)}$ and the residual force $\hat{R}_{i+1}^{(2)}$ are identical. (2) By using the current tangent stiffness $(k_T)_{i+1}^{(2)}$ and the associated value of $(\hat{k}_T)_{i+1}^{(2)}$ from Eq. (5.7.13) in the second iteration, the original Newton–Raphson method leads to a smaller residual force

$\hat{R}_{i+1}^{(3)} = 0$ (Example 5.5) compared to $\hat{R}_{i+1}^{(3)} = 0.2904$ from the modified Newton–Raphson method (Example 5.6). (3) Because at each iteration the residual force $\hat{R}_{i+1}^{(j)}$ is now smaller, convergence is achieved in fewer iterations; for this time step of this example, two iterations are required in the original Newton–Raphson method (Example 5.5) compared to five iterations in the modified Newton–Raphson method (Example 5.6).

FURTHER READING

- Bathe, K.-J., *Finite Element Procedures*, Prentice Hall, Englewood Cliffs, N.J., 1996, Chapter 9.
- Filippou, F. C., and Fenves, G. L., “Methods of Analysis for Earthquake-Resistant Structures,” in: *Earthquake Engineering: From Engineering Seismology to Performance-Based Engineering* (eds. Y. Bozorgnia and V. V. Bertero), CRC Press, New York, 2004, Chapter 6.
- Hughes, T. J. R., *The Finite Element Method*, Prentice Hall, Englewood Cliffs, N.J., 1987, Chapter 9.
- Humar, J. L., *Dynamics of Structures*, 2nd ed., A. A. Balkema Publishers, Lisse, The Netherlands, 2002, Chapter 8.
- Newmark, N. M., “A Method of Computation for Structural Dynamics,” *Journal of the Engineering Mechanics Division, ASCE*, **85**, 1959, pp. 67–94.

PROBLEMS

- 5.1** In Section 5.2 we developed recurrence formulas for numerical solution of the equation of motion of a linear SDF system based on linear interpolation of the forcing function $p(t)$ over each time step. Develop a similar procedure using a piecewise-constant representation of the forcing function wherein the value of the force in the interval t_i to t_{i+1} is a constant equal to \tilde{p}_i (Fig. P5.1). Show that the recurrence formulas for the response of an undamped system are

$$u_{i+1} = u_i \cos(\omega_n \Delta t_i) + \dot{u}_i \frac{\sin(\omega_n \Delta t_i)}{\omega_n} + \frac{\tilde{p}_i}{k} [1 - \cos(\omega_n \Delta t_i)]$$

$$\dot{u}_{i+1} = u_i [-\omega_n \sin(\omega_n \Delta t_i)] + \dot{u}_i \cos(\omega_n \Delta t_i) + \frac{\tilde{p}_i}{k} \omega_n \sin(\omega_n \Delta t_i)$$

Specialize the recurrence formulas for the following definition of the piecewise-constant force: $\tilde{p}_i = (p_i + p_{i+1})/2$. Write the recurrence formulas in the following form:

$$u_{i+1} = Au_i + B\dot{u}_i + Cp_i + Dp_{i+1}$$

$$\dot{u}_{i+1} = A'u_i + B'\dot{u}_i + C'p_i + D'p_{i+1}$$

with equations for the constants A, B, C, \dots, D' .

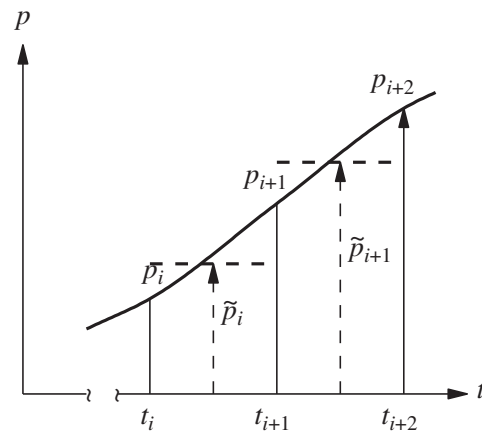


Figure P5.1

- *5.2 Solve Example 5.1 using the piecewise-constant approximation of the forcing function; neglect damping in the SDF system.
- *5.3 Solve the problem in Example 5.1 by the central difference method, implemented by a computer program in a language of your choice, using $\Delta t = 0.1$ sec. Note that this problem was solved as Example 5.2 and that the results were presented in Table E5.2.
- *5.4 Repeat Problem 5.3 using $\Delta t = 0.05$ sec. How does the time step affect the accuracy of the solution?
- *5.5 An SDF system has the same mass and stiffness as in Example 5.1, but the damping ratio is $\zeta = 20\%$. Determine the response of this system to the excitation of Example 5.1 by the central difference method using $\Delta t = 0.05$ sec. Plot the response as a function of time, compare with the solution of Problem 5.3, and comment on how damping affects the peak response.
- *5.6 Solve the problem in Example 5.1 by the central difference method using $\Delta t = \frac{1}{3}$ sec. Carry out your solution to 2 sec, and comment on what happens to the solution and why.
- *5.7 Solve the problem in Example 5.1 by the constant average acceleration method, implemented by a computer program in a language of your choice, using $\Delta t = 0.1$ sec. Note that this problem was solved as Example 5.3, and the results are presented in Table E5.3. Compare these results with those of Example 5.2, and comment on the relative accuracy of the constant average acceleration and central difference methods.
- *5.8 Repeat Problem 5.7 using $\Delta t = 0.05$ sec. How does the time step affect the accuracy of the solution?
- *5.9 Solve the problem in Example 5.1 by the constant average acceleration method using $\Delta t = \frac{1}{3}$ sec. Carry out the solution to 2 sec, and comment on the accuracy and stability of the solution.
- *5.10 Solve the problem of Example 5.1 by the linear acceleration method, implemented by a computer program in a language of your choice, using $\Delta t = 0.1$ sec. Note that this problem was solved as Example 5.4 and that the results are presented in Table E5.4. Compare with the solution of Example 5.3, and comment on the relative accuracy of the constant average acceleration and linear acceleration methods.
- *5.11 Repeat Problem 5.10 using $\Delta t = 0.05$ sec. How does the time step affect the accuracy of the solution?

*Denotes that a computer is necessary to solve this problem.

- ***5.12** Solve the problem of Example 5.5 by the central difference method, implemented by a computer program in a language of your choice, using $\Delta t = 0.05$ sec.
- ***5.13** Solve Example 5.5 by the constant average acceleration method with Newton–Raphson iteration, implemented by a computer program in a language of your choice. Note that this problem was solved as Example 5.5 and the results were presented in Table E5.5.
- ***5.14** Solve Example 5.6 by the constant average acceleration method with modified Newton–Raphson iteration, implemented by a computer program in a language of your choice. Note that this problem was solved as Example 5.6 and the results were presented in Table E5.6.
- ***5.15** Solve Example 5.5 by the linear acceleration method with Newton–Raphson iteration using $\Delta t = 0.1$ sec.
- ***5.16** Solve Example 5.5 by the linear acceleration method with modified Newton–Raphson iteration using $\Delta t = 0.1$ sec.

*Denotes that a computer is necessary to solve this problem.