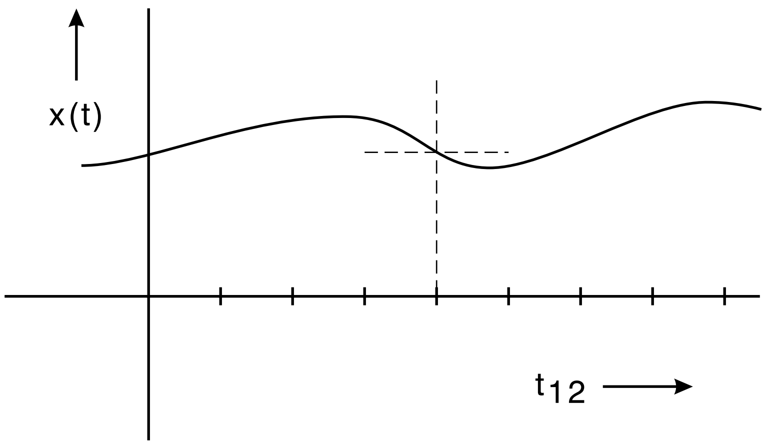
MISN-0-39

COMPUTER ALGORITHM FOR THE DAMPED DRIVEN OSCILLATOR



Project PHYSNET Physics Bldg. Michigan State University East Lansing, MI

COMPUTER ALGORITHM FOR
THE DAMPED DRIVEN OSCILLATOR

by
Peter Signell

Title: **Computer Algorithm for the Damped Driven Oscillator**

Author: Peter Signell, Dept. of Physics, Mich. State Univ

Version: 2/1/2000 Evaluation: Stage 0

Length: 1 hr; 12 pages

Input Skills:

1. Expand a given function about a given point using Taylor's series (MISN-0-4).

Output Skills (Knowledge):

K1. Given the force acting on a damped driven oscillator along with the oscillator's position and velocity at a specified time, derive a Numerov type algorithm for the approximate numerical calculation of the oscillator's position at all past and future times.

Post-Options:

1. "Response of a Damped Driven Oscillator" (MISN-0-30).
2. "Damped Driven Oscillations; Mechanical Resonances" (MISN-0-31).
3. "Laplace Transform Solution for the Damped Driven Oscillator" (MISN-0-47).

COMPUTER ALGORITHM FOR THE DAMPED DRIVEN OSCILLATOR

by

Peter Signell

1. Derivation

1a. Equation for a Damped Driven Oscillator. The equation to be solved is that of the damped harmonically-driven oscillator[1] acted on by the force:

$$F(t) = -kx(t) - \lambda v(t) + F_0 \cos(\omega t).$$

Since[2]

$$F(t) = ma(t) = m\frac{dx(t)}{dt} \equiv mx''(t),$$

and

$$v(t) = \frac{dx(t)}{dt} \equiv x'(t),$$

our equation to be solved is:

$$mx''(t) + kx(t) + \lambda x'(t) - f(t) = 0, \tag{1}$$

where

$$f(t) \equiv F_0 \cos(\omega t).$$

1b. Approximate Derivatives by Finite Differences. We now make two power series expansions[3] about time t:

$$x(t + \Delta) = x(t) + \Delta x'(t) + (\Delta^2/2)x''(t) + (\Delta^3/6)x'''(t) + \ldots$$

$$x(t - \Delta) = x(t) - \Delta x'(t) + (\Delta^2/2)x''(t) - (\Delta^3/6)x'''(t) + \ldots$$

We will choose Δ sufficiently small so that we can disregard all terms after Δ^2 without incurring much error. Then add and subtract the above equations to obtain:

$$x(t + \Delta) + x(t - \Delta) \simeq 2x(t) + \Delta^2 x''(t)$$

[1]See "Damped Driven Oscillations; Mechanical Resonances" (MISN-0-31).
[2]See "Particle Dynamics - The Laws of Motion" (MISN-0-14).
[3]See "Taylor's Series for the Expansion of a Function About a Point" (MISN-0-4).

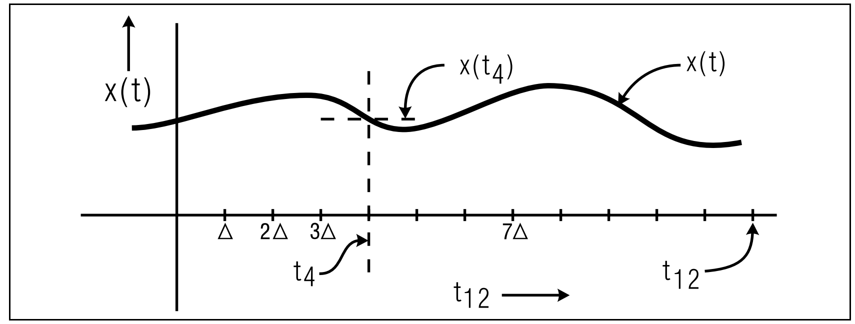


Figure 1. The labeling system for time net-points.

$$x(t + \Delta) - x(t - \Delta) \simeq 2\Delta x'(t)$$

or

$$x''(t) \simeq \frac{1}{\Delta^2}\left[x(t + \Delta) - 2x(t) + x(t - \Delta)\right]$$

$$x'(t) \simeq \frac{1}{2\Delta}\left[x(t + \Delta) - x(t - \Delta)\right]$$

which are often quoted in calculus courses.[4]

1c. Net-Point Times. We need a more succinct labeling system at this point or we won't be able to see the forest for the trees. We define discrete "net-point" times as: $t_n \equiv n\Delta$, where Δ is some small time interval (see Fig. 1). We can then rewrite the derivatives at time t_n as:

$$x''_n \simeq \frac{1}{\Delta^2}(x_{n+1} - 2x_n + x_{n-1}) \tag{2}$$

$$x'_n \simeq \frac{1}{2\Delta}(x_{n+1} - x_{n-1}) \tag{3}$$

Then at time t_n our basic Eq. (1) becomes:

$$mx''_n + kx_n + \lambda x'_n - f_n = 0.$$

Substituting for x''_n and x'_n:

$$\frac{m}{\Delta^2}\left(x_{n+1} - 2x_n + x_{n-1}) + kx_n + \frac{\lambda}{2\Delta}(x_{n+1} - x_{n-1}\right) - f_n = 0$$

[4]The second equation above is used in "Response of a Damped Driven Oscillator" (MISN-0-30) where a microcomputer is utilized in finding solutions.

Collecting coefficients of the x values, we get:

$$x_{n+1} = Ax_n + Bx_{n-1} + \Delta^2 C_n \qquad (4)$$

where:

$$\begin{aligned} A &= \left(2 - \Delta^2 \frac{k}{m}\right)/D \\ B &= -\left(1 - \Delta\frac{\lambda}{2m}\right)/D \\ C_n &= \left(\frac{f_n}{m}\right)/D \\ D &= 1 + \Delta\frac{\lambda}{2m} \end{aligned}$$

2. Application

2a. Iterating the Recurrence Relation. Equation (4) is called a three point recurrence relation because it relates x at three successive time net-points. If you know x_0 and x_1, you can use this relation to calculate x_2. Then knowing x_1 and x_2, you can calculate x_3, etc. You can keep up this recurrence procedure until you have reached the time for which you wish to know the solution. For example, this might be a time sufficiently large so that transient effects have died away and only the steady state solution remains.

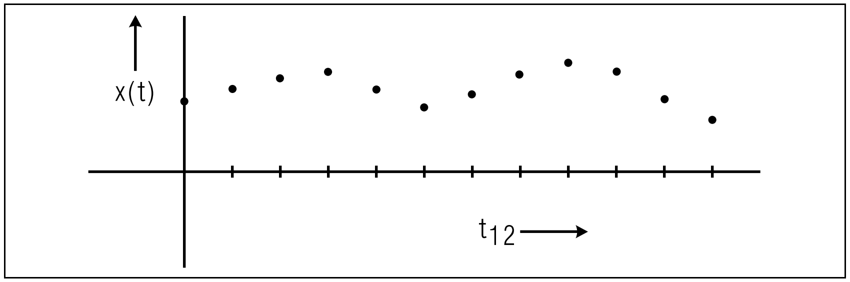


Figure 2. The Calculated Curve. Draw a smooth curve through the plotted points and you have $x(t)$.

2b. Starting the Iteration. With a three-point recurrence relation one can calculate values at successive time net-points, but one needs two adjacent values in order to get started. In the example cited in the preceding paragraph, one needed x_0 and x_1 in order to start the procedure. Those two initial values generally come from a specification of a "complete set of initial conditions." Our Eq. (1), being a second order equation for x, always requires that one specify two independent conditions on x.

A common case is where both the position and velocity are known at some instant of time. Here we will call that time $t = 0$; hence we know x_0 and x_1 by solving Eq. (3) at $n = 0$ for x_{-1} (Note: $v_0 = x_0'$), and using that to eliminate x_{-1} from Eq. (4) for $n = 0$. We thus obtain x_1 in terms of x_0 and v_0 (usually given quantities):

$$x_1 = \left(1 - \Delta^2 \frac{k}{2m}\right) x_0 + \left(\Delta - \Delta^2 \frac{\lambda}{2m}\right) v_0 + \Delta^2 \frac{f_0}{2m}.$$

Now the algorithm is complete. Put in x_0 and v_0 to get x_0 and x_1, x_0 and x_1 to get x_2, x_1 and x_2 to get x_3, etc.

▷ How would you obtain position values for times earlier than the one for which x and v are initially known? Answer: To see the answer, replace each of the following letters by its successor in the alphabet: mdfzshud cdksz.

Acknowledgments

Preparation of this module was supported in part by the National Science Foundation, Division of Science Education Development and Research, through Grant #SED 74-20088 to Michigan State University.

MODEL EXAM

1. The force acting on a damped driven oscillator is $F = -kx - \lambda v + F_0 \cos(\omega t)$; the oscillator's initial position and speed are x_0 and v_0. Derive without notes, a complete Numerov-type algorithm for the (approximate) calculation of the oscillator's position at all times. NOTE: The algorithm is to include the method of use of any three point recurrence relation such as:

$$x_{n+1} = Ax_n + Bx_{n-1} + \Delta^2 C_n$$

where:

$$\begin{aligned} A &= \left(2 - \Delta^2 \frac{k}{m}\right)/D \\ B &= -\left(1 - \Delta \frac{\lambda}{2m}\right)/D \\ C_n &= \left(\frac{f_n}{m}\right)/D \\ D &= 1 + \Delta \frac{\lambda}{2m} \end{aligned}$$

and boundary conditions such as:

$$x_1 = \left(1 - \Delta^2 \frac{k}{2m}\right) x_0 + \left(\Delta - \Delta^2 \frac{\lambda}{2m}\right) v_0 + \Delta^2 \frac{f_0}{2m}.$$

Brief Answers:

1. See this module's *text*.