Handout # 6 (MEEN 617)

Numerical Integration to Find Time Response of SDOF mechanical system

State Space Method

The EOM for a linear system is

$$M\,\ddot{X} + D\,\dot{X} + K\,X = F(t) \qquad (1)$$

with initial conditions, at $t = 0 \rightarrow X(0) = X_o;\ \dot{X}(0) = \dot{X}_o = V_o$

Define the following variables, $\quad Y_1 = X;\ \dot{Y}_2 = X \qquad (2)$

and write EOM (1) as two first-order Eqs.

$$M\,\dot{Y}_2 + D\,Y_2 + K\,Y_1 = F(t) \quad \& \quad \dot{Y}_1 = Y_2 \qquad (3)$$

which can be written in matrix form as

$$\begin{bmatrix} \dot{Y}_1 \\ \dot{Y}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -M^{-1}K & -M^{-1}D \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -M^{-1}F \end{bmatrix} \qquad (4)$$

Or,

$$\dot{\mathbf{Y}} = \mathbf{A}\,\mathbf{Y} + \mathbf{b} \qquad (5)$$

with $\mathbf{Y} = \begin{bmatrix} Y_1 = X \\ Y_2 = \dot{X} \end{bmatrix};\ \mathbf{A} = \begin{bmatrix} 0 & 1 \\ -M^{-1}K & -M^{-1}D \end{bmatrix};\ \mathbf{b} = \begin{bmatrix} 0 \\ M^{-1}F \end{bmatrix}$

This is known as the **state-space formulation**. Eq. (5) is to be integrated numerically with initial condition vector

$\mathbf{Y}_o = \begin{bmatrix} X_o & V_o \end{bmatrix}^T$

If the applied load is NOT a function of time, then an equilibrium state is defined after a very long time as

$$\dot{\mathbf{Y}} \to 0 \;\Rightarrow\; \mathbf{Y_E} = -\mathbf{A}^{-1}\mathbf{b}_o \qquad (6)$$

Computational software such as Mathcad®, Mapple®, Mathematica®, Matlab®, etc has **built-in functions or commands** to perform the numerical integration of equations set in the form

$\dot{\mathbf{Y}} = \mathbf{A}\,\mathbf{Y} + \mathbf{b}$, even when system is nonlinear, i.e. **A=A(Y).**

A few words about numerical integration methods

Typical numerical integration methods include

a) Euler (simple) method
b) Fourth and Fifth-Order Runge-Kutta integrators,
c) Rosenbrock Method see references on page 12,
d) Adams Predictor Corrector Methods
e) Average Acceleration and Wilson-θ (Implicit) Methods

In most methods, the selection of an adequate time step is crucial for numerically stable and accurate results. (a)-(b) are favored by the young initiates into numerical computing and because of their ready availability in modern computational software. (c) –(d) are more modern (implicit) methods with automated intermediate resizing of the time step while performing the integration. Methods (e) have long been favored by structural mechanic analysts when integrating Multiple DOF (linear) systems

All methods suffer from deficiencies when nonlinearities are apparent thus forcing extremely small time steps and the ensuing

cost with lots of numerical computing (time). (Memory) Storage appears not to be an issue anymore.

State-space method for MDOF systems.

Recall the EOMS for a linear system are

$$\mathbf{M}\,\ddot{\mathbf{U}} + \mathbf{D}\,\dot{\mathbf{U}} + \mathbf{K}\,\mathbf{U}_{(t)} = \mathbf{F}_{(t)} \qquad (7)$$

where $\mathbf{U}, \dot{\mathbf{U}},$ and $\ddot{\mathbf{U}}$ are the vectors of generalized displacement, velocity and acceleration, respectively; and $\mathbf{F}_{(t)}$ is the vector of generalized (external forces) acting on the system.
$\mathbf{M}, \mathbf{D}, \mathbf{K}$ represent the matrices of inertia, viscous damping and stiffness coefficients, respectively[1].

Define the following variables, $\mathbf{Y}_1 = \mathbf{U}\,;\ \dot{\mathbf{Y}}_2 = \dot{\mathbf{U}}$ (8)
and write EOM (7) as a set of 2n-first-order Eqs.

$$\mathbf{M}\,\dot{\mathbf{Y}}_2 + \mathbf{D}\,\mathbf{Y}_2 + \mathbf{K}\,\mathbf{Y}_1 = \mathbf{F}_{(t)} \quad \& \quad \dot{\mathbf{Y}}_1 = \mathbf{Y}_2 \qquad (9)$$

which can be written in matrix form as

$$\begin{bmatrix} \dot{\mathbf{Y}}_1 \\ \dot{\mathbf{Y}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{-1}\mathbf{F} \end{bmatrix} \qquad (10)$$

Or,

$$\dot{\mathbf{Y}} = \mathbf{A}\,\mathbf{Y} + \mathbf{b} \qquad (11)$$

with $\mathbf{Y} = \begin{bmatrix} \mathbf{U} \\ \dot{\mathbf{U}} \end{bmatrix};\ \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix};\ \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{F} \end{bmatrix}$ (12)

[1] The matrices are square with *n*-rows = *n* columns, while the vectors are n-rows.

and initial conditions $\mathbf{Y}_{(t=0)} = \begin{bmatrix} \mathbf{U_o} & \dot{\mathbf{U}}_\mathbf{o} \end{bmatrix}^T$.

A is a 2*n* x 2*n* matrix. **I** is the *n*x*n* identity matrix, and **0** is a *n*x*n* matrix full of zeroes.

Conditions for a good numerical integrator

In general a numerical integration scheme should

a) reproduce EOM as time step $\Delta t \to 0$

b) provide, as with physical model, bounded solutions for any size of time step, i.e. method should be stable

c) reproduce the physical response with fidelity and accuracy.

The numerical integration relies in representing time derivatives of a function with an algebraic approximation, for example

$$\dot{x} = \frac{dx}{dt} = \lim_{\Delta t \to 0} \frac{\Delta x}{\Delta t} \Rightarrow$$

$$\dot{x} \simeq \frac{x(t_i + \Delta t) - x(t_i)}{\Delta t} \sim \frac{x_{i+1} - x_i}{\Delta t} \qquad (13)$$

Eq. above is exact only if $\Delta t \to 0$

Numerical integration methods are usually divided into two categories, implicit and explicit.

Consider the ODE $\dot{x} = f(x,t)$ (14)

In an **explicit** numerical scheme, the ODE is represented in terms of known values at a prior time step, i.e.

$$x_{i+1} = x_i + \Delta t \; f(x_i, t_i), \qquad (15)$$

while in an **implicit numerical** scheme

$$x_{i+1} = x_i + \Delta t\ f\left(x_{i+1}, t_i\right) \tag{16}$$

Explicit numerical schemes are **conditionally stable**. That is, they provide bounded numerical solutions for (very) small time steps. For example,

$$\Delta t \le \tau_{crit} = \frac{T_n}{\pi} \tag{17}$$

where $T_n = \dfrac{2\pi}{\omega_n}$ and $\omega_n = \sqrt{K/M}$ are the natural period and natural frequency of the system, respectively. The restriction on the time step is too severe when analyzing **stiff systems**, i.e. those with large natural frequencies.

Implicit numerical schemes are **unconditionally stable**, i.e. do not impose a restriction on the size of the time step Δt. (However, accuracy may be compromised if too large time steps are used).

ANALOGY between numerical schemes and a filter

A few words of wisdom released in class

The average acceleration method[2]

for numerical integration of SDOF equation:

$$M\,\ddot{X} + D\,\dot{X} + K\,X = F(t) \qquad (1)$$

Consider a change of "thinking frame" by defining

Arithmetic ~ Continuum (function)

$$\begin{aligned} &X_i \sim X(t_i), \qquad F_i \sim F(t_i), \\ &X_{i+1} \sim X(t_{i+1}), \quad F_{i+1} \sim F(t_{i+1}) \end{aligned} \qquad (17)$$

Write Eq. (1) at two times,

$$\text{at } t = t_i \Rightarrow M\,\ddot{X}_i + D\,\dot{X}_i + K\,X_i = F_i \qquad (18a)$$

$$\text{at } t = t_{i+1} \Rightarrow M\,\ddot{X}_{i+1} + D\,\dot{X}_{i+1} + K\,X_{i+1} = F_{i+1} \qquad (18b)$$

Subtract (b) from (a) to obtain:

$$M\,\Delta\ddot{X}_i + D\,\Delta\dot{X}_i + K\,\Delta X_i = \Delta F_i \qquad (19)$$

where

$$\begin{aligned} &\Delta X_i = X_{i+1} - X_i \qquad \Delta F_i = F_{i+1} - F_i, \\ &\Delta\dot{X}_i = \dot{X}_{i+1} - \dot{X}_i, \qquad etc \end{aligned} \qquad (20)$$

Note that known quantities at $t=t_i$ are $\left\{ X_i,\ \dot{X}_i, \ddot{X}_i \right\}$.

---

[2] This numerical method is extremely popular among the structural dynamics community. Its extension to MDOF systems will be shown later. The other favorite method, Wilson-θ scheme, will also be given in later lectures (MDOF systems).

Now, assume the acceleration is constant within the time interval $\Delta t_i = t_{i+1} - t_i$

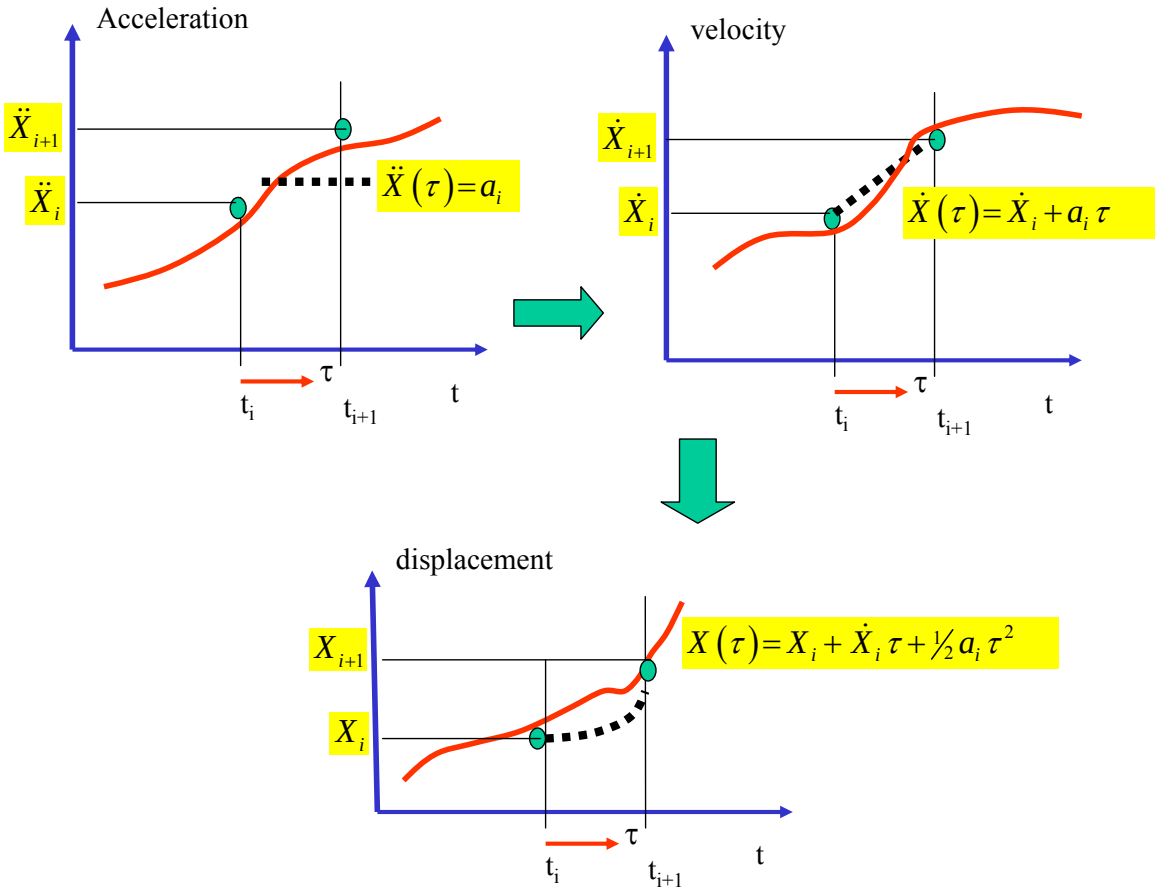
$$\ddot{X}(\tau) = a_i \qquad \text{for} \quad 0 < \tau \le \Delta t_i \qquad (21)$$

set as an average value $a_i = \tfrac{1}{2}\left(\ddot{X}_{i+1} + \ddot{X}_i\right)$.

The velocity and displacement follow from integration of Eq. (6) within the time interval, i.e.

$$\dot{X}(\tau) = \dot{X}_i + a_i\,\tau \qquad (22a)$$

$$X(\tau) = X_i + \dot{X}_i\,\tau + \tfrac{1}{2} a_i\,\tau^2 \qquad (22b)$$



At the end of the time interval, the velocity and displacement equal

$$\dot{X}_{i+1} = \dot{X}\left(\Delta t_i\right) = \dot{X}_i + a_i\,\Delta t_i \tag{23a}$$

$$X_{i+1} = X\left(\Delta t_i\right) = X_i + \dot{X}_i\,\Delta t_i + \tfrac{1}{2}\,a_i\left(\Delta t_i\right)^2 \tag{23b}$$

And the differences in velocity and displacement re

$$\begin{aligned}\Delta\dot{X}_i &= \left(\dot{X}_{i+1} - \dot{X}_i\right) = a_i\,\Delta t_i = \tfrac{1}{2}\left(\ddot{X}_{i+1} + \ddot{X}_i\right)\Delta t_i \\ &= \tfrac{1}{2}\left(\ddot{X}_{i+1} - \ddot{X}_i + 2\ddot{X}_i\right)\Delta t_i = \\ &= \tfrac{1}{2}\left(2\ddot{X}_i + \Delta\ddot{X}_i\right)\Delta t_i\end{aligned} \tag{24a}$$

$$\begin{aligned}\Delta X_i &= \left(X_{i+1} - X_i\right) = \dot{X}_i\,\Delta t_i + \tfrac{1}{2}\,a_i\left(\Delta t_i\right)^2 \\ &= \dot{X}_i\,\Delta t_i + \tfrac{1}{4}\Delta t_i^2\left(\ddot{X}_{i+1} + \ddot{X}_i - \ddot{X}_i + \ddot{X}_i\right) \\ &= \dot{X}_i\,\Delta t_i + \tfrac{1}{4}\Delta t_i^2\left(\Delta\ddot{X}_i + 2\ddot{X}_i\right)\end{aligned} \tag{25b}$$

from (25b), $$\Delta\ddot{X}_i = -2\ddot{X}_i + \frac{4}{\Delta t_i^2}\left(\Delta X_i - \dot{X}_i\,\Delta t_i\right) \tag{26a}$$

and into (24a)

$$\begin{aligned}\Delta\dot{X}_i &= \tfrac{1}{2}\left(2\ddot{X}_i + \Delta\ddot{X}_i\right)\Delta t_i \\ &= \tfrac{1}{2}\left(2\ddot{X}_i - 2\ddot{X}_i + \frac{4}{\Delta t_i^2}\left(\Delta X_i - \dot{X}_i\,\Delta t_i\right)\right)\Delta t_i\end{aligned}$$

$$\Delta\dot{X}_i = \frac{2}{\Delta t_i}\Delta X_i - \dot{X}_i \qquad (26b)$$

Note that in Eqs (26), $\left\{\Delta\dot{X}_i, \Delta\ddot{X}_i\right\}$, depend on the known values obtained at the prior time step, i.e. $\left\{\dot{X}_i, \ddot{X}_i\right\}$ and the unknown ΔX_i. Thus, replace $\left\{\Delta\dot{X}_i, \Delta\ddot{X}_i\right\}$ into the difference equation (19),

$$M\,\Delta\ddot{X}_i + D\,\Delta\dot{X}_i + K\,\Delta X_i = \Delta F_i$$

$$M\left\{-2\ddot{X}_i + \frac{4}{\Delta t_i^2}\left(\Delta X_i - \dot{X}_i\,\Delta t_i\right)\right\} + D\left\{\frac{2}{\Delta t_i}\Delta X_i - \dot{X}_i\right\} + K\,\Delta X_i = \Delta F_i$$

Rearranging terms leads to

$$K_i^*\,\Delta X_i = \Delta F_i^* \qquad (27)$$

where

$$K_i^* = \left[K + \frac{2}{\Delta t_i}D + \frac{4}{\Delta t_i^2}M\right] \qquad (28a)$$

$$\Delta F_i^* = \Delta F_i + 2M\,\ddot{X}_i + \left[2D + \frac{4}{\Delta t_i}M\right]\dot{X}_i \qquad (28b)$$

are known as **pseudo dynamic stiffness and dynamic force**, respectively

The **recipe** for the numerical integration to find the system time response is

At time t_i, known variables are $\left\{ \dot{X}_i, X_i \right\}$ (current state)

(1) find from EOM: $\ddot{X}_i = M^{-1}\left(F_i - D\,\dot{X}_i - K\,X_i \right)$

(2) form pseudo stiffness and forcing functions, $\left(K_i^*, \Delta F_i \right)$ from Eqs. (28),

(3) Calculate $\Delta X_i = \left(K_i^* \right)^{-1} \Delta F_i$, and $\Delta \dot{X}_i = \frac{2}{\Delta t_i} \Delta X_i - \dot{X}_i$;

(4) $X_{i+1} = X_i + \Delta X_i$, $\dot{X}_{i+1} = \dot{X}_i + \Delta \dot{X}_i$ at t_{i+1}

(5) Increase time to t_{i+2} and return to step (1)

The **average acceleration** method is an **implicit method**, i.e. numerically stable and consistent. The disadvantage is that it requires *memory*[3] to store $\dot{X}_i, \ddot{X}_i, \Delta \dot{X}_i, \Delta X_i, \Delta F_i$.

Average acceleration methods for numerical integration of a nonlinear system

Consider the system with EOM

$$M\,\ddot{X} + g\left(\dot{X}, X \right) = F(t) \qquad (30)$$

where $g\left(\dot{X}, X \right)$ is a nonlinear function, for example

$$g\left(\dot{X}, X \right) = g_o + k_o\,X + k_3\,X^3 + F_\mu\, sign(\dot{X})$$

As with the linear system, evaluate Eq. (30) at two times (closely spaced):

$$\text{at } t = t_i \Rightarrow M\,\ddot{X}_i + g\left(\dot{X}_i, X_i \right) = F_i \qquad (31a)$$

[3] A non-issue in the 21st century

$$\text{at } t = t_{i+1} \Rightarrow M\,\ddot{X}_{i+1} + g\left(\dot{X}_{i+1}, X_{i+1}\right) = F_{i+1} \qquad (31b)$$

Subtract (b) from (a) to obtain:

$$M\,\Delta\ddot{X}_i + g_{i+1} - g_i = \Delta F_i \qquad (32)$$

where

$$\Delta\ddot{X}_i = \ddot{X}_{i+1} - \ddot{X}_i \qquad \Delta F_i = F_{i+1} - F_i,$$
$$g_i = g\left(X_i, \dot{X}_i\right),\ g_{i+1} = g\left(X_{i+1}, \dot{X}_{i+1}\right) \qquad (33)$$

A Taylor series expansion of the nonlinear function at t_i gives

$$g_{i+1} = g_i + \left.\frac{\partial g}{\partial X}\right|_{X_i,\,\dot{X}=0} \Delta X_i + \left.\frac{\partial g}{\partial \dot{X}}\right|_{\dot{X}_i} \Delta\dot{X}_i + O\left(\Delta X_i^2, \Delta\dot{X}_i^2\right) \qquad (34)$$

define local linearized stiffness and damping coefficients as

$$K_i = \left.\frac{\partial g}{\partial X}\right|_{X_i,\,\dot{X}_i};\ D_i = \left.\frac{\partial g}{\partial \dot{X}}\right|_{X_i,\,\dot{X}_i} \qquad (35)$$

Hence,

$$g_{i+1} - g_i \approx K_i\,\Delta X_i + D_i\,\Delta\dot{X}_i$$

and the difference Eq. (32) becomes ***linear***

$$M\,\Delta\ddot{X}_i + D_i\,\Delta\dot{X}_i + K_i\,\Delta X_i = \Delta F_i \qquad (36)$$

Eq. (35) is formally identical to the one devised for a linear system. Thus, the numerical treatment is similar, except **that at each time**

step, linearized stiffness and damping coefficients need be calculated.

The recipe is thus identical; however with the apparent nonlinearity, the method does not guarantee stability for (too) large time steps.

The **recipe** for the numerical integration to find the system time response is

At time t_i, known variables are $\left\{ \dot{X}_i, X_i \right\}$ (current state)

(1) find from EOM: $\ddot{X}_i = M^{-1}\left[F_i - g\left(X_i, \dot{X}_i \right) \right]$

(2a) find local (linearized) stiffness and damping coefficients, (K_i , D_i) from eq. (35)

(2b) form pseudo stiffness and forcing functions, $\left(K_i^*, \Delta F_i \right)$ from

$$K_i^* = \left[K_i + \frac{2}{\Delta t_i} D_i + \frac{4}{\Delta t_i^2} M \right] \; ; \; \Delta F_i^* = \Delta F_i + 2M\, \ddot{X}_i + \left[2\, D_i + \frac{4}{\Delta t_i} M_i \right] \dot{X}_i$$

(3) Calculate $\Delta X_i = \left(K_i^* \right)^{-1} \Delta F_i$, and $\Delta \dot{X}_i = \frac{2}{\Delta t_i} \Delta X_i - \dot{X}_i$;

(4) Set $X_{i+1} = X_i + \Delta X_i$, $\dot{X}_{i+1} = \dot{X}_i + \Delta \dot{X}_i$ at t_{i+1}

(5) Increase time to t_{i+2} and return to step (1)

References The following are a must!

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., 1986 (1st edition), "Numerical Recipes, The Art of Scientific Computing," Cambridge University Press, UK.

Bathe, J-K, 1982 (1st ed), 2007 latest, "Finite Element Procedures," Prentice Hall.

Piche, R., and P. Nevalainen, 1999, "Variable Step Rosenbrock Algorithm for Transient Response of Damped Structures," Proc. IMechE, Vol. 213, part C, Paper C05097.

Appendix A[4]

Numerical Integration Using Modified Euler's Method

It is often difficult to solve (exactly) a nonlinear differential equation. Numerical integration is then employed to obtain results (predicitions of motion). The **Modified Euler Method** is one type of numerical integration scheme.

Solve for *q(t)* governed by

$$M\,\ddot{q}+C\,\dot{q}+K\,q=Q(q,\,\dot{q},\,t) \qquad \text{(A.1)}$$

with initial conditions q_0 and $\dot{q}_0$ at $t=0$. In Eqn. (A.1), $Q(q,\,\dot{q},\,t)$ may contain terms that are nonlinear in

Let
$$V=\frac{d\,q}{d\,t} \qquad \text{(A.2)}$$

and write Eq. (A.1) as a system of TWO first order differential equations, i.e.

$$\dot{V}=-\frac{C}{M}V-\frac{K}{M}q+\frac{Q(q,V,t)}{M}, \qquad \text{(A.3)}$$
$$\dot{q}=V$$

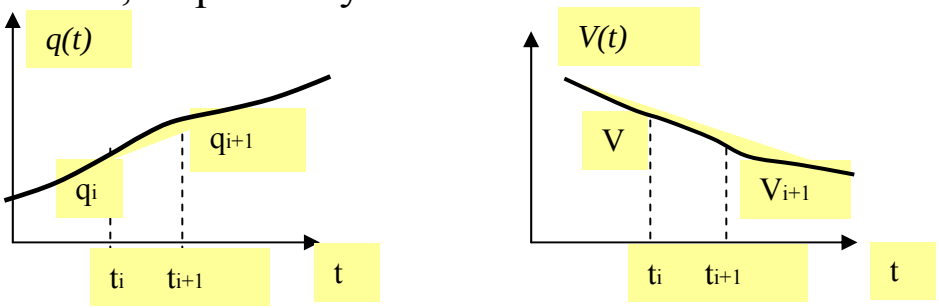
[4] This Appendix is included because most young engineering students have learned only about Euler's method. Hence, the appendix complements their education.. However, I encourage you to abandon the usage of this poor method.

Define $$\omega_n = \sqrt{\frac{K}{M}} \text{ and } \zeta = \frac{C}{2\sqrt{K\,M}} \qquad \text{(A.4)}$$

as the **natural frequency** and **viscous damping ratio**, respectively. With the noted definitions Eqs. (A.3) become

$$\dot{V} = \frac{d\,V}{d\,t} = F_1(q, V, t) = -2\zeta\,\omega_n^2 V - \omega_n^2\,q + \frac{Q(q, V, t)}{M}$$
$$\dot{q} = \frac{d\,q}{d\,t} == F_2(q, V, t) = V \qquad \text{(A.5)}$$

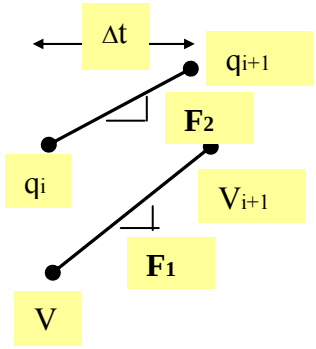
Note that F_1 and F_2 are the **slopes** of the *(V* vs *t)* and *(q* vs *t)* curves, respectively.



Let the numerical approximations (arithmetic values) be

$$V_i \approx V(t_i), \text{and } q_i \approx q(t_i) \qquad \text{(A.6)}$$

where $t_i = i\,\Delta t;\ _{i=0,1,\ldots.}$ and Δt is a suitably small time step for numerical integration.



In ***Euler's numerical scheme,*** approximate the time derivatives (or slopes) as:

$$V_{i+1} \sim V_i + \Delta t\ f_{1_i}, \quad f_{1_i} = F_1\left(q_i, V_i, t_i\right),$$
$$q_{i+1} \sim q_i + \Delta t\ f_{2_i}, \quad f_{2_i} = F_2\left(q_i, V_i, t_i\right)$$

for i=1,2 ….. (A.7)

with initial conditions q_0 and $\dot{q}_0$ at $t_0 = 0$

Eq. (7) offer a recursive relation to calculate the numerical (arithmetic) values of the variables V_i and q_i at successive times *t).* The *regular* ***Euler method*** is first order with a truncation error of order (Δt^2).

A modified Euler method (second order accurate) with error order (Δt^3) follows:

(a) Compute preliminary estimates of $\left(\underline{V}_{i+1}, \underline{q}_{i+1}\right)$ as

$$\underline{V}_{i+1} \sim V_i + \Delta t\, F_1\left(q_i, V_i, t_i\right),$$
$$\underline{q}_{i+1} \sim q_i + \Delta t\, F_2\left(q_i, V_i, t_i\right) \qquad \text{(A8.a)}$$

(b) Use these preliminary estimates to obtain improved **slopes** as

$$f_{1_{i+1}} = F_1\left(\underline{q}_{i+1}, \underline{V}_{i+1}, t_{i+1}\right),$$
$$f_{2_{i+1}} = F_2\left(\underline{q}_{i+1}, \underline{V}_{i+1}, t_{i+1}\right), \qquad \text{(A8.b)}$$

(c) Define **average slopes** as per

$$f_{1_i} = \frac{1}{2}\left(f_{1_{i+1}} + f_{1_i}\right)$$
$$f_{2_i} = \frac{1}{2}\left(f_{2_{i+1}} + f_{2_i}\right) \qquad \text{(A8.c)}$$

(c)and obtain **new estimations** using the averaged slopes, i.e.

$$V_{i+1} \sim V_i + \Delta t\ f_{1_i},$$
$$q_{i+1} \sim q_i + \Delta t\ f_{2_i} \qquad (A8.d)$$

(d) **Repeat steps (a) through (d) at** each time step, $t_i = i\,\Delta t;\ _{i=0,1,....}$, and starting with the initial conditions q_0 and $\dot{q}_0$ at $t_0 = 0$

The size of the time step Δt is very important to obtain accurate and numerically stable results.

If Δt is too large, then numerical predictions will be in error and very likely show unstable (oscillating) results.

If Δt is too small, then the numerical method will be slow and costly since the number of operations increases accordingly.

In practice, a time step of the order $\Delta t = T_n/60$, where T_n is the natural period of motion given as ($2\pi/\omega_n$).

Euler's method is **most times NOT** a good choice to perform the numerical integration of linear or nonlinear ODES. Alas, it is widely used by rookie engineers and engineering students because it is easy to implement. Often enough, however, predictions can be wrong and misleading.

I call Euler's method a BRUTE FORCE approach,

Often regarded as an ***art***, numerical computing is in actuality a well established branch of mathematics.