**Computational Methods for Transient Coupled-Physics Problems (KAIST Fall 2005/KCP)**

| | |
|---|---|
| 18 October 2005: | Basic Transient Analysis Techniques |
| 20 October 2005: | Staggered Solution Procedures |
| 25 October 2005: | Structure-Acoustic Interaction Analysis |
| 27 October 2005: | Stabilization Strategies and their Applications |
| 01 November 2005: | Fine Points in Transient Analysis Techniques |
| 03 November 2005: | Simulation of Multi-Physics Problems |

*What is a transient problem?*

Engineering problems may be categorized according to the nature of applied forces or boundary conditions as

- Quasi-static problems;
- Periodic response problems (e.g., steady-state or vibratory problems);
- Transient problems;
  - Smooth (e.g., wind force) propagation problems.
  - Non-smooth (e.g., stress wave) propagation problems.

*Transient problems* cover, also labeled as unsteady problems, the responses of engineering systems from the initial responses to the onset of steady-state responses.

## Motivations for Studying Transient Analysis Techniques

*What Matlab and/or Commercial Software Can Provide:*

- For small-scale systems that can be described in terms of a state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}\,\mathbf{u}(t)$$

  use `"step"`, `"impulse"`, etc.

- For moderate size problems expressed as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$$

  use one of `"ode"` solvers.

*What Matlab and/or Commercial Software Can't Provide:*

- For a large-scale systems, e.g., the degrees of freedom exceeding several thousands to several millions.
- For coupled systems with vastly varying different time constants, and systems whose transient responses need a long-time duration.
- For the transient simulation of systems wherein its computational cost becomes prohitively expensive.
- For coupled systems for which the computations of coupling phenomena are difficult to accomplish both from the point of capturing the coupling mechanisms and from the point of computation efficiency. *Nearly all new emerging coupled-physics problems belong to this category!*

## What Do We Need to Compute in Transient Analysis?

- *Accurate tracing of motions or of the rate of changes:*

  *Eulerian or Lagrangian coordinates, potential functions, flux electrical, flow, thermal or their rates, and other kinamatics*

- *Accurate tracing of the state of constitutive laws*
- *Accurate tracing of momentum expressions*
- *Accurate tracing of energy or conservation principles*
- *Accurate tracing of non-smooth or dissipative mechanisms*

**Hence, transient analysis must capture accurately the frequency (or time constant) characteristics, attendant kinamatics, constitutive states, energy principles, and tracing of any discontinuities, coupling mechanisms.**

## Analytical Solution of Linear Canonical Model Equation

Linear Canonical Equation for Mechanical Systems:

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{f}(t) \tag{1}$$

$$\mathbf{p} = \mathbf{M}\dot{\mathbf{x}} + \mathbf{C}\mathbf{x}$$

$$\dot{\mathbf{p}} = \mathbf{f}(t) - \mathbf{K}\mathbf{x}$$

$$\dot{\mathbf{x}} = \mathbf{M}^{-1}\mathbf{p} - \mathbf{M}^{-1}\mathbf{C}\mathbf{x}$$

$$\Downarrow$$

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{p}} \end{Bmatrix} = \begin{bmatrix} -\mathbf{C} & \mathbf{I} \\ -\mathbf{K} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \mathbf{p} \end{Bmatrix} + \begin{Bmatrix} \mathbf{0} \\ \mathbf{f}(t) \end{Bmatrix} \tag{2}$$

$$\Downarrow$$

Canonical equation: $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}(t)$

Output equation: $\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$

**Solution of the Canonical Equation:**

$$\mathbf{x} = e^{\mathbf{A}t}\mathbf{x}(0) + \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)\, d\tau \tag{3}$$

$\boxed{e^{\mathbf{A}t}\mathbf{x}(0)}$-term: response due to the initial conditions, $\mathbf{x}(0)$.

$\boxed{\int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)\, d\tau}$-term: response due to the external (applied) forcing function, $\mathbf{f}(t)$.

**Discrete approximation:**

$$\mathbf{x}(kh) = e^{\mathbf{A}kh}\mathbf{x}(0) + \int_0^{kh} e^{\mathbf{A}(kh-\tau)}\mathbf{B}\mathbf{u}(\tau)\, d\tau$$

$$\mathbf{x}(kh+h) = e^{\mathbf{A}(kh+h)}\mathbf{x}(0) + \int_0^{kh+h} e^{\mathbf{A}(kh+h-\tau)}\mathbf{B}\mathbf{u}(\tau)\, d\tau$$

## Discrete approximation-cont'd

$$\mathbf{x}(kh+h) = e^{\mathbf{A}h}[e^{\mathbf{A}kh}\mathbf{x}(0)] + \int_0^{kh} e^{\mathbf{A}(kh+h-\tau)}\mathbf{B}\mathbf{u}(\tau)\,d\tau + \int_{kh}^{kh+h} e^{\mathbf{A}(kh+t-\tau)}\mathbf{B}\mathbf{u}(\tau)\,d\tau$$

$$\mathbf{x}(kh+h) = e^{\mathbf{A}h}[e^{\mathbf{A}kh}\mathbf{x}(0)] + e^{\mathbf{A}h}[\int_0^{kh} e^{\mathbf{A}(kh-\tau)}\mathbf{B}\mathbf{u}(\tau)\,d\tau] + \int_{kh}^{kh+h} e^{\mathbf{A}(kh+h-\tau)}\mathbf{B}\mathbf{u}(\tau)\,d\tau$$

$$\mathbf{x}(kh+h) = e^{\mathbf{A}h}\mathbf{x}(kh) + \int_{kh}^{kh+h} e^{\mathbf{A}(kh+h-\tau)}\mathbf{B}\mathbf{u}(\tau)\,d\tau$$

**Discrete approximation of the convolution term:**

$$\int_{kT}^{kh+h} e^{\mathbf{A}(kh+h-\tau)}\mathbf{B}\mathbf{u}(\tau)\, d\tau$$

(*via change of variable:* $\sigma = kh + h - \tau$)

$$\approx \int_{h}^{0} e^{\mathbf{A}\sigma}\mathbf{B}\mathbf{u}(kh)\,(-d\sigma), \quad \text{with } \mathbf{u}(\tau) \approx \mathbf{u}(kT)$$

$$= \int_{0}^{h} e^{\mathbf{A}\sigma}\mathbf{B}\mathbf{u}(kh)\,(d\sigma)$$

$$= \mathbf{A}^{-1}(e^{\mathbf{A}h} - \mathbf{I})\,\mathbf{B}\,\mathbf{u}(kh) = \mathbf{\Gamma}\,\mathbf{u}(kh)$$

where $\mathbf{\Gamma} = \mathbf{A}^{-1}(e^{\mathbf{A}h} - \mathbf{I})\,\mathbf{B}$

## Discrete approximation -concluded

$$\boxed{\begin{array}{c} \mathbf{x}(kh+h) = e^{\mathbf{A}h}\mathbf{x}(kh) + \mathbf{\Gamma}\,\mathbf{u}(kh) \\ \Downarrow \\ x(k+1) = \mathbf{\Phi}\,\mathbf{x}(k) + \mathbf{\Gamma}\,\mathbf{u}(k), \quad \mathbf{\Phi} = e^{\mathbf{A}h} \\ \mathbf{\Gamma} = \mathbf{A}^{-1}(e^{\mathbf{A}h} - \mathbf{I})\,\mathbf{B} \end{array}} \tag{4}$$

- When the size of $\mathbf{A}$ is large and/or ill-conditioned, the exponential evaluation of $e^{\mathbf{A}h}$ is costly and often inaccurate. This is because, in effect, $e^{\mathbf{A}h}$ is evaluated by a variation of the Taylor series:

$$e^{\mathbf{A}h} = \mathbf{I} + h\mathbf{A} + \tfrac{1}{2}(h\mathbf{A})^2 + \frac{1}{6}(h\mathbf{A})^3 + ... + \frac{1}{n!}(h\mathbf{A})^n \tag{5}$$

  which requires 2n-multiplications of the term $h\mathbf{A}\mathbf{x}(k)$ and a matrix factorization for evaluating $\mathbf{A}^{-1}\mathbf{B}\mathbf{u}(k)$ per integration step $h$.

**Numerical Integration Methods: Explicit and Implicit**

Let us solve for $\mathbf{x}(t_n + h)$ of the problem stated as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, t), \quad \mathbf{A} = \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}(t_n)} \tag{6}$$

- *Explicit Algorithms:* A numerical integration scheme that yields $\mathbf{x}(t_n + h)$ by evaluating the function $\{\mathbf{f}(\mathbf{x}(t_k), t_k),\ k = 0, 1, 2 \ldots, n\}$ from the past solution states is called an explicit method.

- *Implicit Algorithms:* A numerical integration scheme that yields $\mathbf{x}(t_n + h)$ by evaluating the function $\{\mathbf{f}(\mathbf{x}(t_k), t_k),\ k = 0, 1, 2 \ldots, n\}$ *plus* the search direction $\mathbf{A}$, and by seeking Newton-like search directions of the matrix in the form $(\mathbf{I} + h\ \mathbf{A})$ is called an implicit method.

- *Explicit-Implicit Algorithms:* Combinations of the above two methods

## An Example of Explicit Algorithm: the Central Difference Method

$$
\begin{aligned}
\mathbf{p}(t_n + \tfrac{1}{2}h) &= \mathbf{p}(t_n - \tfrac{1}{2}h) + h\dot{\mathbf{p}}(t_n) \\
\mathbf{x}(t_n + h) &= \mathbf{x}(t_n) + h\dot{\mathbf{x}}(t_n + \tfrac{1}{2}h) \\
&\Downarrow \\
\mathbf{p}_{n+1/2} &= \mathbf{p}_{n-1/2} + h\dot{\mathbf{p}}_n \\
\mathbf{x}_{n+1} &= \mathbf{x}_n + h\dot{\mathbf{x}}_{n+1/2}
\end{aligned}
\tag{7}
$$

When (7) is used to integrate (2), we obtain:

$$
\begin{aligned}
\mathbf{p}_{n+1/2} &= \mathbf{p}_{n-1/2} + h(\mathbf{f}_n - \mathbf{K}\mathbf{x}_n) \\
\mathbf{x}_{n+1} &= \mathbf{x}_n + h\mathbf{M}^{-1}(\mathbf{p}_{n+1/2} - \mathbf{C}\mathbf{x}_{n+1/2})
\end{aligned}
\tag{8}
$$

As $\mathbf{x}_{n+1/2}$ is not available, we replace it by

$$
\mathbf{x}_{n+1/2} = \tfrac{1}{2}(\mathbf{x}_{n+1} + \mathbf{x}_n) \tag{9}
$$

Substituting (9) into the second of (8), we obtain

$$(\mathbf{M} + \tfrac{1}{2}h\mathbf{C})\mathbf{x}_{n+1} = (\mathbf{M} - \tfrac{1}{2}h\mathbf{C})\mathbf{x}_n + h\mathbf{p}_{n+1/2} \tag{10}$$

When $\mathbf{M}$ is diagonal (lumped mass matrix) but $\mathbf{D}$ is not, the solution of the above equation is expensive. Thus, one often employs an approximate inversion formula of the form:

$$(\mathbf{M} + \tfrac{1}{2}h\mathbf{C})^{-1} \approx \mathbf{M}^{-1}(\mathbf{I} - \tfrac{1}{2}h\mathbf{M}^{-1}\mathbf{C} + \frac{h^2}{4}(\mathbf{M}^{-1}\mathbf{C})^2) \tag{11}$$

We will discuss more on this aspect later in the lecture.

## An Example of Implicit Method: the Trapezoidal Rule

$$\begin{aligned} \mathbf{p}_{n+1/2} &= \mathbf{p}_n + h\dot{\mathbf{p}}_{n+1/2}, \quad \mathbf{p}_{n+1} = 2\mathbf{p}_{n+1/2} - \mathbf{p}_n \\ \mathbf{x}_{n+1/2} &= \mathbf{x}_n + h\dot{\mathbf{x}}_{n+1/2}, \quad \mathbf{x}_{n+1} = 2\mathbf{x}_{n+1/2} - \mathbf{x}_n \end{aligned} \tag{12}$$

When (12) is used to integrate (2), we obtain with $\delta = \frac{1}{2}h$:

$$\begin{bmatrix} \mathbf{M} + \delta\mathbf{C} & -\delta\mathbf{I} \\ \delta\mathbf{K} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \mathbf{p} \end{Bmatrix}_{n+1/2} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \mathbf{p} \end{Bmatrix}_{n} + \delta \begin{Bmatrix} \mathbf{0} \\ \mathbf{f}(t) \end{Bmatrix}_{n+1/2} \tag{13}$$

which can be simplified as

$$\begin{aligned} \mathbf{x}_{n+1/2} &= \mathbf{E}^{-1}(\mathbf{M}\mathbf{x}_n + \delta\mathbf{p}_n + \delta^2\mathbf{f}_{n+1/2}, \ \mathbf{E} = (\mathbf{M} + \delta\mathbf{C} + \delta^2\mathbf{K}) \\ \mathbf{p}_{n+1/2} &= \mathbf{p}_n + \delta(\mathbf{f}_{n+1/2} - \mathbf{K}\mathbf{x}_{n+1/2}) \end{aligned} \tag{14}$$

Once the half-step values are obtained by the previous expressions, the full-step values are obtained via

$$\begin{aligned} \mathbf{p}_{n+1} &= 2\mathbf{p}_{n+1/2} - \mathbf{p}_n \\ \mathbf{x}_{n+1} &= 2\mathbf{x}_{n+1/2} - \mathbf{x}_n \end{aligned} \tag{15}$$

Implementation details of the trapezoidal rule will be discussed later in the lecture.

## Computational Stability of Numerical Integration Methods

We will be guided by the following guide in assessing the computational stability: *an integration formula is stable when it is used to time-discretize physically stable diferential equations and the solution of the resulting difference equations is bounded.*

We are interested in two types of differential equations: parabolic systems whose roots lie on the negative real axis on the complex plane, and the hyperbolic systems whoseroots lie on the imaginary axis for undamped cases and lefthand complex plane for damped cases.

For the differential equations we search for its roots in the form of

$$\mathbf{x}(t) = \mathbf{c}e^{\mathbf{s}t} \tag{16}$$

where the characteristic roots $\mathbf{s}$ lie in the lefthand side of the complex plane.

The corresponding discrete characteristic roots for the difference equations can be obtained via

$$\mathbf{x}_{n+1} = \boldsymbol{\lambda}\mathbf{x}_n \tag{17}$$

For stability we must have

$$|\boldsymbol{\lambda}| \leq 1 \tag{18}$$

First, in order to assess the computational stability of a numerical integration algorithm, we obtain the characteristic value **s** by substituting (16) into the homogeneous equation of (2), viz.,

$$[\mathbf{s} - \mathbf{A}]\,\mathbf{x} = 0 \quad \Rightarrow \quad \mathbf{s} = \mathbf{TAT}, \quad \mathbf{x} = \mathbf{Tq} \tag{19}$$

so that one can obtain a set of diagonalized equation for the eigenvalue $s_k$ as

$$\dot{\mathbf{q}}_k = s_k \mathbf{q}_k \tag{20}$$

Second, we express an integration algorithm of the form

$$\sum_{i=0}^{m} \alpha_i \mathbf{q}_{n-i} - h \sum_{i=0}^{m} \beta_i \dot{\mathbf{q}}_{n-i} = 0 \tag{21}$$

Substituting (20) and (17) into the above integration formula, we obtain:

$$[\rho(\lambda) - (sh)\sigma(\lambda)]\mathbf{q}_{n-m} = 0$$

$$\rho(\lambda) = \sum_{i=0}^{m} \alpha_i \lambda^{m-i}, \qquad \sigma(\lambda) = \sum_{i=0}^{m} \beta_i \lambda^{m-i} \tag{22}$$

Hence, for a specified discrete characteristic value of $\lambda$, the characteristic value $(sh)$ can be determined from the preceding equation as

$$(sh) = \frac{\rho(\lambda)}{\sigma(\lambda)} \tag{23}$$

In particular, as $|\lambda| < 1$ yields a stable solution whereas $|\lambda| > 1$ leads to instability, the stability boundary (sb) can be determined as the contour satisfying

$$
\begin{aligned}
|\lambda| = 1 \quad &\Rightarrow \quad \lambda = e^{j\theta},\ 0 \le \theta \le 2\pi \\
&\Downarrow \\
(sh)|_{(sb)} = \frac{\rho(e^{j\theta})}{\sigma(e^{j\theta})} &= \frac{\sum_{k=0}^{m} \alpha_j e^{j(m-k)\theta}}{\sum_{k=0}^{m} \beta_j e^{j(m-k)\theta}}, \qquad 0 \le \theta \le 2\pi
\end{aligned}
\tag{24}
$$

**Table Backward Implicit Multistep Formulas**

| Formula | Gear 2-step | Gear 3-step | Park 3-step |
|---|---|---|---|
| $\alpha_0$ | 3 | 11 | 10 |
| $\alpha_1$ | -4 | -18 | -15 |
| $\alpha_3$ | 1 | 9 | 6 |
| $\alpha_3$ | 0 | -2 | -1 |
| $\beta_0$ | 2 | 6 | 6 |

# An Example: One-step formula

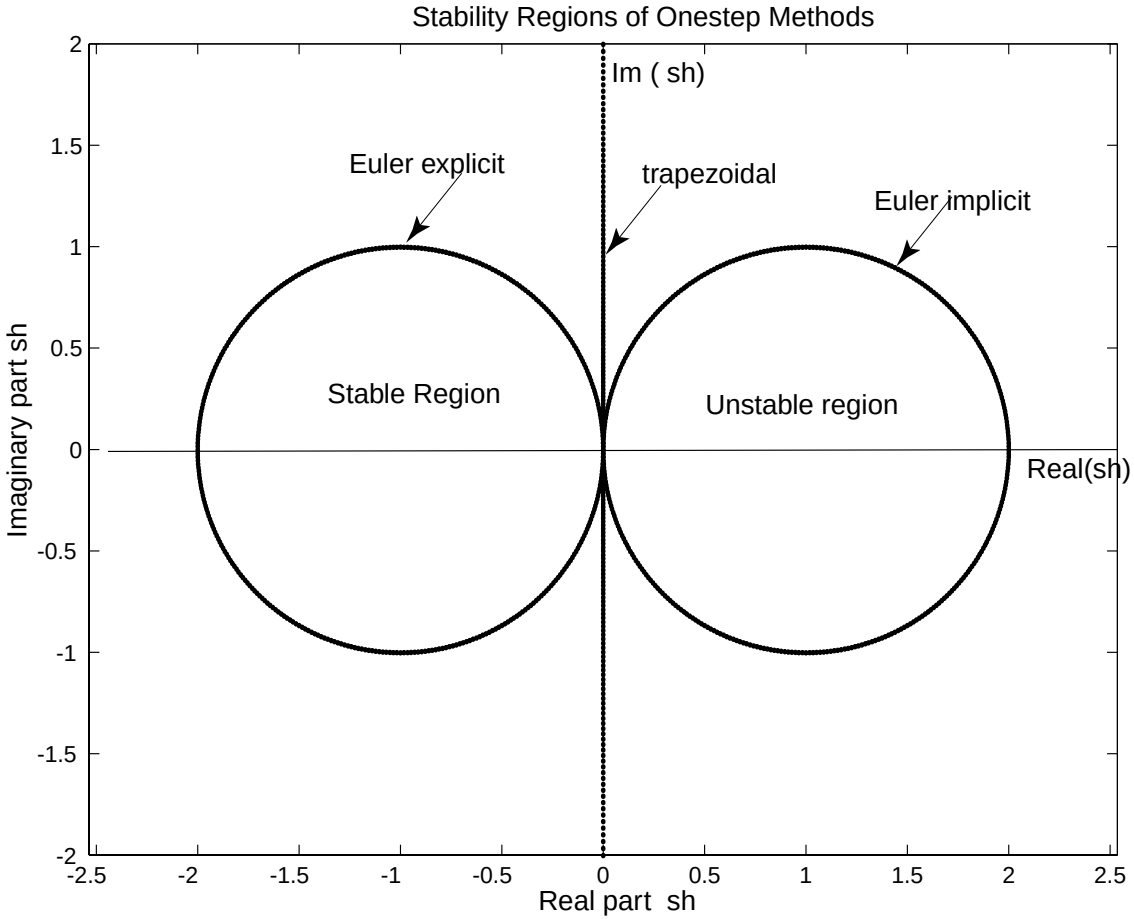


Fig. 1 Stability Regions of Onestep Formulas

$$(\mathbf{s}h)_b = \frac{e^{i\theta} - 1}{\alpha e^{i\theta} + (1-\alpha)}, \quad 0 \le \theta \le 2\pi \tag{25}$$

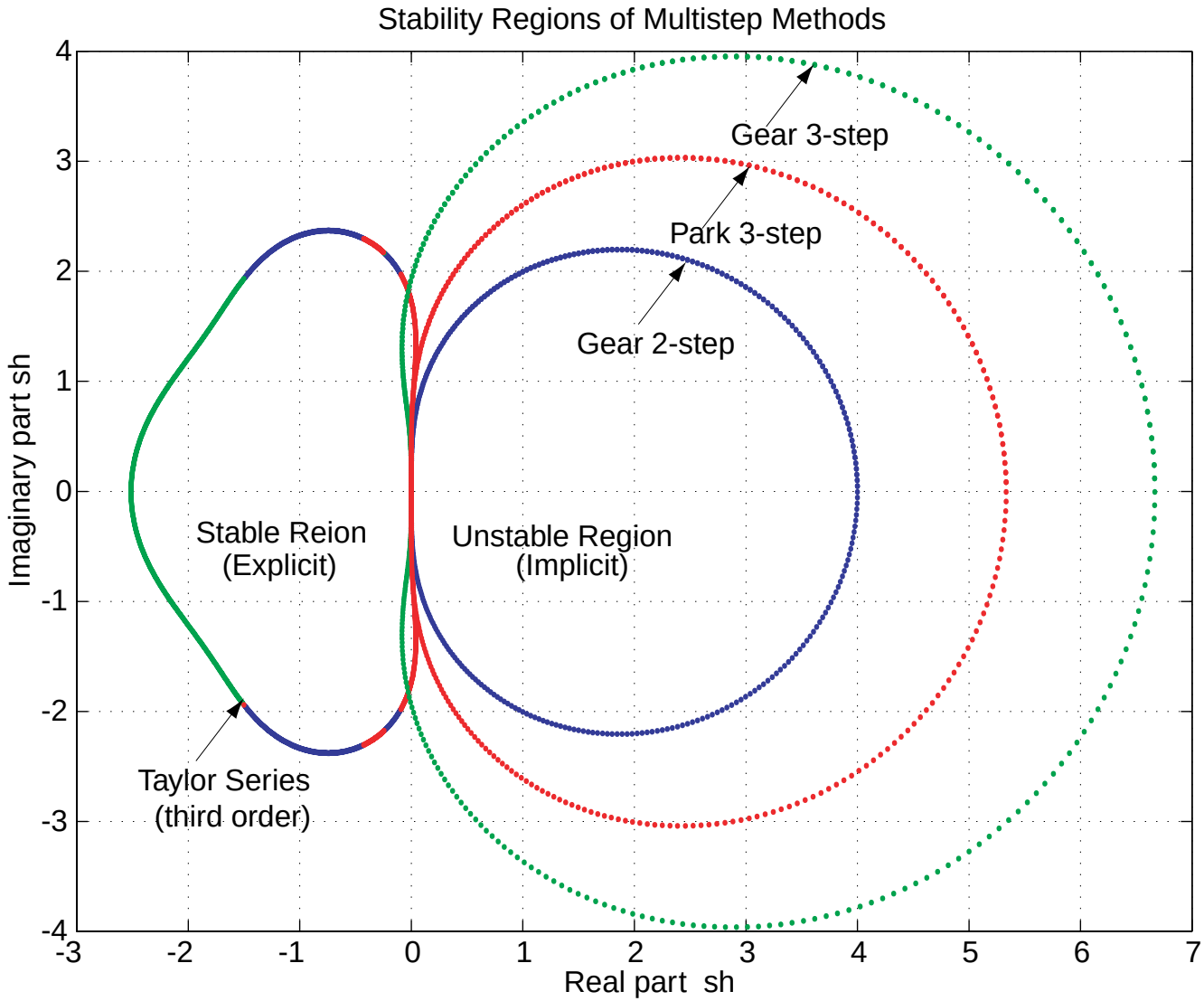


Fig. 2 Stability Regions of Multistep Formulas - Overall

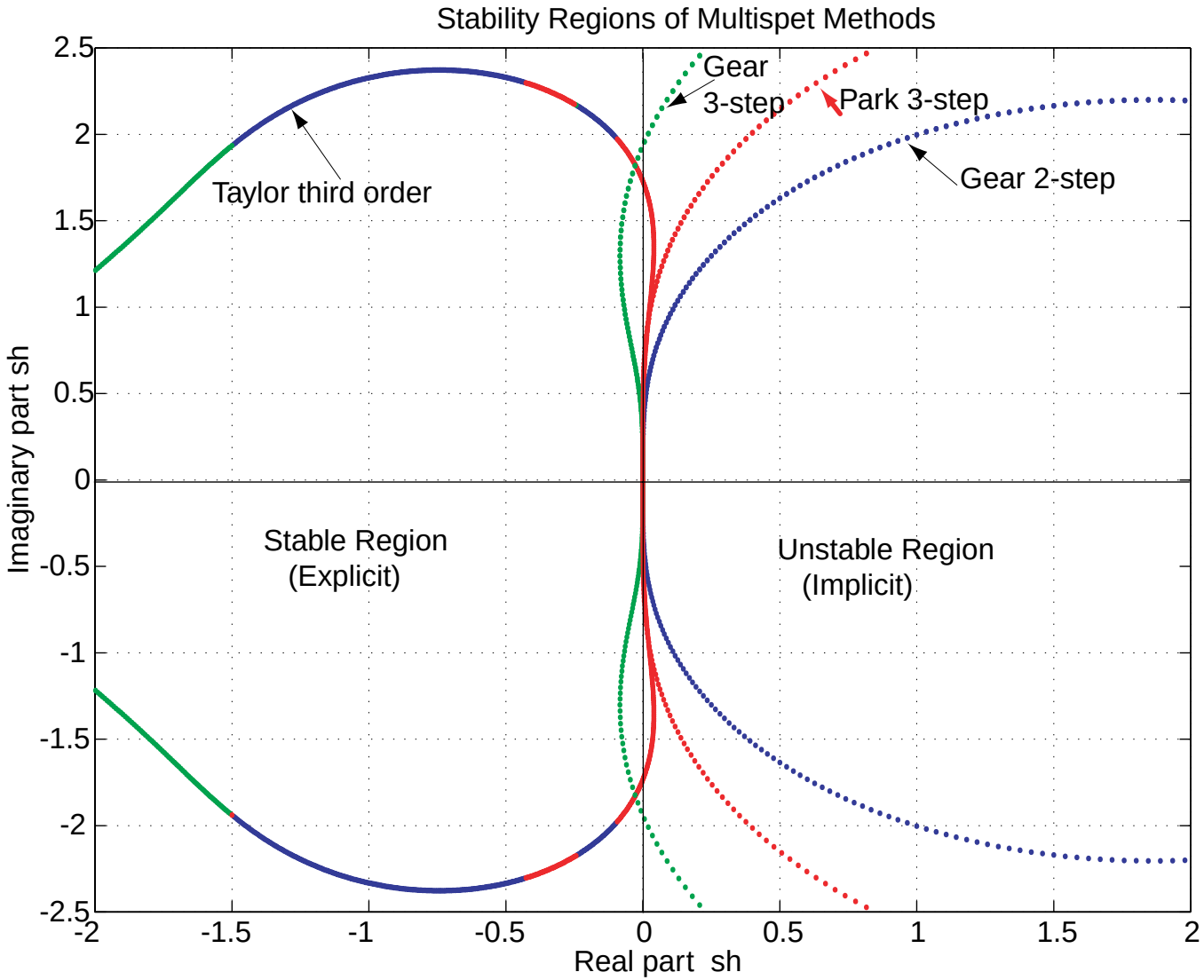


Fig. 3 Stability Regions of Multistep Formulas - Enlarged Around the Origin

## Accuracy of Integration formulas

*Errors in the Characteristic Domain*

Let us recall the linear multistep integration formula

$$\sum_{j=0}^{m}(\alpha_j q_{n-j} - h\beta_j \dot{q}_{n-j}) = 0 \tag{22}$$

and its characteristic equation:

$$L(\lambda) = \rho(\lambda) - \mathbf{s}h\sigma(\lambda) = 0 \tag{26}$$

If we integrate the first-order canonical equation (2) by (22), the solution at time $t = nh$ can be expressed as

$$q_n = \sum_{j=l}^{m} \lambda_j a_j \tag{27}$$

where $\lambda_j$ are roots of $L(\lambda) \equiv 0$ and $a_j$ are constants that depend on the initial conditions at $t = h(n - j),\ \ j = 1...m$.

Therefore, for one-step formulas, $\lambda_0$ approximates $e^{\mathbf{s}h}$ (why?). If $m > 1$, i.e.,for multistep formulas, the computed solution contains $m$ characteristic roots. Of the $m$ roots, we call the one that approximates $e^{\mathbf{s}h}$ most closely, the principal root, $\lambda_p$, and the rest $(m - 1)$ roots are called the extraneous roots.

The error in the characteristic domain, $E_c$, is therefore defined as the difference between the principal root and its corresponding exact component $e^{\mathbf{s}h}$, *viz.*,

$$E_c = e^{\mathbf{s}h} - \lambda_p \tag{28}$$

**An Example:**

For one-step formulas

$$q_n = q_{n-1} + h(\alpha \dot{q}_n + (1 - \alpha)\dot{q}_{n-1}) \tag{29}$$

the difference operator $L(\lambda)$ reads

$$L(\lambda) = (\lambda - 1) - \mathbf{s}h(\alpha\lambda + (1-\alpha)) \tag{30}$$

The principal root is obtained from $L(\lambda) = 0$, therefore, as

$$\lambda_p = \frac{1 + (1-\alpha)\mathbf{s}h}{1 - \alpha\mathbf{s}h} \tag{31}$$

Hence, the error in the characteristic domain becomes

$$E_c = e^{\mathbf{s}h} - \frac{1 + (1-\alpha)\mathbf{s}h}{1 - \alpha\mathbf{s}h} \tag{32}$$

The truncation error in the characteristic domain, $E_{TC}$, is then obtained as the non-vanishing leading term in the above equation in its Taylor series around $\mathbf{s}h \to 0$, *viz*,

$$E_{TC} = (\tfrac{1}{2} - \alpha)(\mathbf{s}h)^2 + (\frac{1}{6} - \alpha^2)(\mathbf{s}h)^3 \tag{33}$$

*Truncation Error in the Time Domain*

The prevalent definition of the term *truncation error* in the time domain $E_{TT}$ is obtained by expanding the solution in terms of the Taylor series and retaining the non-vanishing leading term in the form

$$E_{TT} = c_1 h\dot{q}_{n-1} + c_2 h^2 \ddot{q}_{n-1} + c_3 h^3 \dddot{q}_{n-1} + .. \tag{34}$$

Note that the above time-domain truncation error is easily obtained from the truncation error in the characteristic domain $E_{TC}$ in (33) since the leading error at the n-th step, $E$, is given by

$$E = E_{TC} q_{n-1} \tag{35}$$

and

$$\frac{du^r}{dt^r} = \mathbf{s}^r q_{n-1} \tag{36}$$

Therefore, for one-step formulas, the truncation error in the time domain becomes

$$E_{TT} = (\tfrac{1}{2} - \alpha)h^2 \ddot{q}_{n-1} + (\frac{1}{6} - \alpha^2)h^3 \dddot{q}_{n-1} \tag{37}$$

The evaluation of the truncation error in the time domain requires in practice approximate calculations of high derivatives (which is dependent on computer implementation) such as $\ddot{\ddot{q}}_{n-1}$ at every integration step. Hence, the truncation error in the characteristic domain can be viewed as the most accurate evaluation of the *relative* truncated error.

## Practical Considerations on the Truncation Error

In the preceding discussions, the truncation error was defined as the non-vanishing leading term of the Taylor series expansion of the given difference equation. The fundamental assumption for the validity of the truncation error is the differentiability of the solution up to the order of the truncation error. However, in second-order dynamics problems, the derivatives (or the derivative expressions from a difference formula) of the corresponding order can often become discontinuous due to boundary, initial and/or loading conditions. The contact-impact problem, for instance, is one such example. Experience has shown that the truncation error is not reliable for measuring the accuracy of the computed solution for these problems.

Another practical restriction on the truncation error is that it does not distinguish whether the error is due to the phase distortion or amplitude decay. Therefore, for oscillatory problems like the structural dynamics equations, the task of estimating the global error (say, after one period) by the truncation error estimates is often impractical as the latter essentially

govern only the local error magnitude.

Finally, for explicit formulas, the relative magnitude of the truncation error with respect to the solution can remain small even if the stepsize exceeds the stability limits.

## A Global Accuracy Measure

The desirable measure of accuracy is the one which can provide not only the local errors committed at each integration step, but also the error accumulated several steps (say, one period) after the current step. This is because the composition of the dominant frequency response changes in time for both linear and nonlinear problems; hence, the accuracy measure that meets such requirements can assure us the desired global accuracy with respect to the dominant response components provided they can be estimated accurately.

Fortunately, there exists one such measure of accuracy if the following assumptions are valid:

- The number of the characteristic roots of the difference equation is equal to the order of differential equation if the formula is explicit.
- The magnitudes of extraneous roots are considerably smaller than those of the principal roots if the formula is implicit.
- The bulk of the system energy is contained within the dominant response modes.

If the above assumptions hold, one can obtain the accuracy of the computed solution by expressing the principal root in the form

$$\lambda_\rho = \Re(\lambda_p) + \jmath\Im(\lambda_p) \tag{38}$$

and defining for undamped systems:

*Artificial damping ratio*, $\xi_a$

$$\xi_a = \frac{1}{2\omega h}\ln\{(\Re(\lambda_p))^2 + (\Im(\lambda_p))^2\} \tag{39}$$

*Artificial frequency distortion,* $f_a$

$$f_a = \frac{1}{\omega h} \tan^{-1}\{\frac{\Im(\lambda_p)}{\Re(\lambda_p)}\} \tag{40}$$

Hence, (38) can be written in the form

$$\lambda_p = e^{(\xi_a + J f_a)\omega h} \tag{41}$$

Note that, even for linear problems, $\xi_a$ and $f_a$ are dependent upon $\omega h$! This can be compared with the exact solution

$$\lambda_e = e^{i\omega h} \tag{42}$$

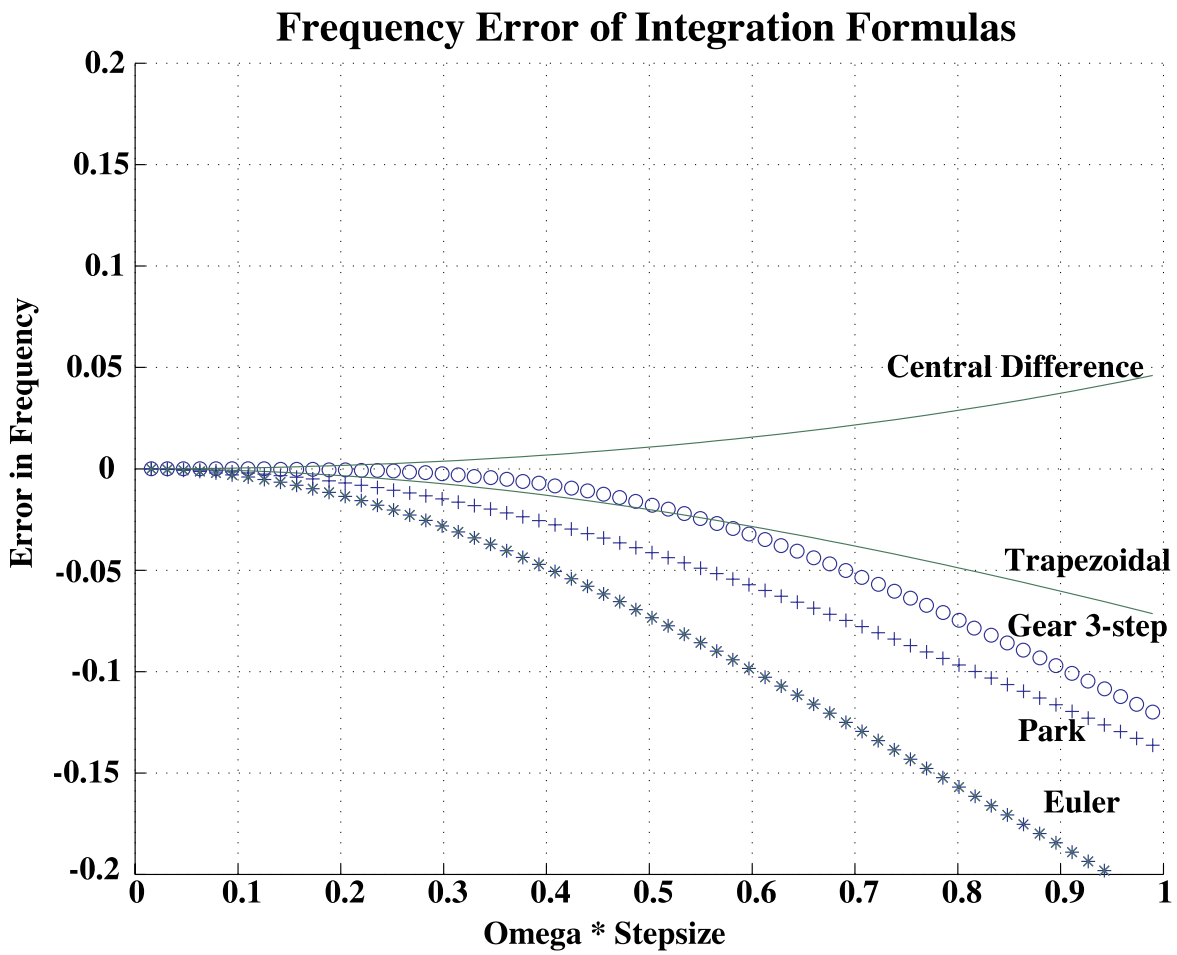


Fig. 4 Frequency error of various formulas

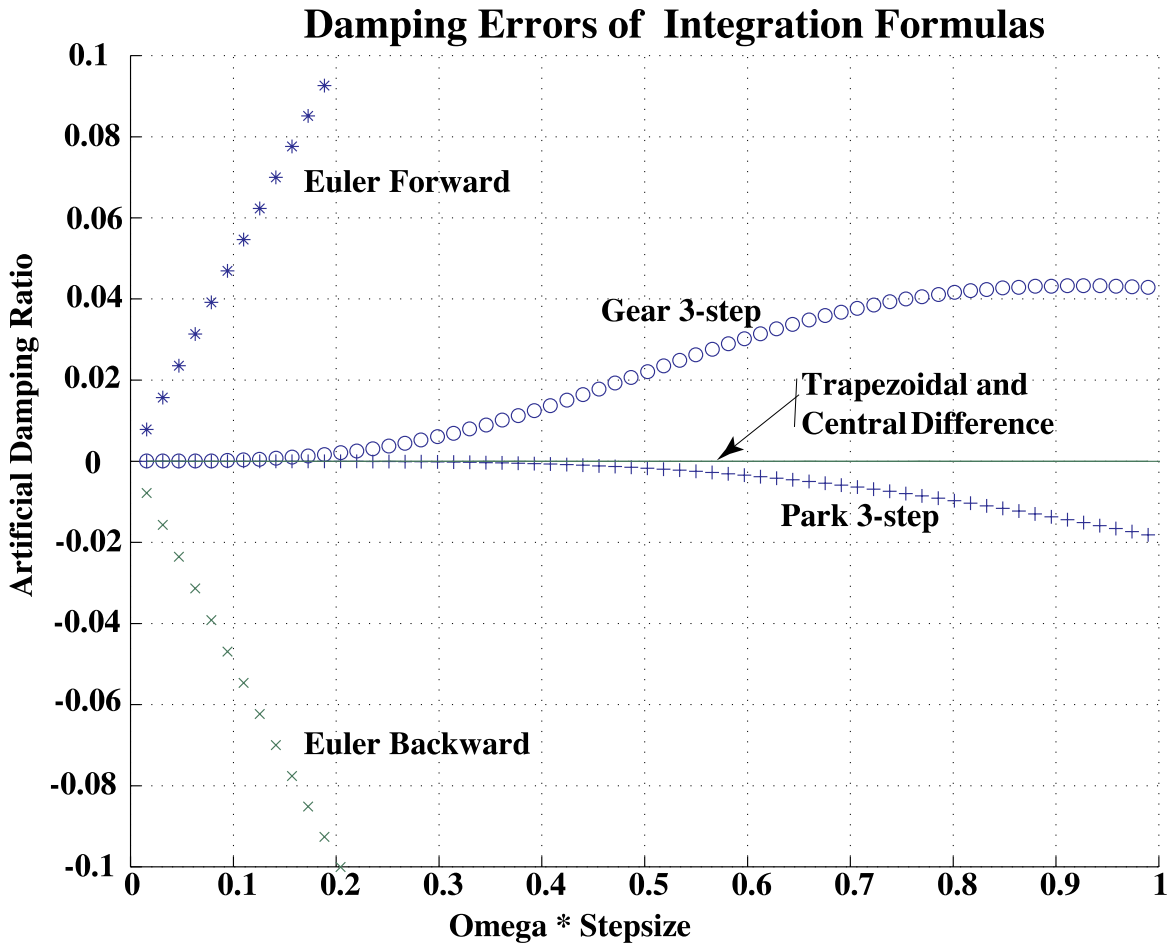


Fig. 5 Damping error of various formulas

Figures 4 and 5 illustrate the artificial damping ratio and the artificial frequency distortion for several popular direct time integration formulas. Observe that both the trapezoidal rule and the central difference formula do not inject any artificial damping. In addition, the trapezoidal rule

stretches the period whereas the central difference formula shortens the period. In general, implicit formulas elongate the period and explicit formulas shortens it. This has an important bearing on the choice of mass matrics, viz, lumped mass vs consistent mass matrix. It can be shown that the lumped mass lowers the system frequencies whereas the consistent mass matrix increases the frequencies. Hence, it's best to employ the lumped mass matrix in conjunction with explicit formulas. On the other hand, it's preferable to employ the consistent mass matrix when one employs implicit formulas. This is fortunate because the computational overhead associated with explicit formulas would have been prohitively expensive if the consistent mass matrix were to be used.

For linear systems, one would like to assess the accumulated artificial damping and artificial frequency distortion. The accumulated error after $k$ integration steps can be assessed by the following relations:

*Artificial damping after k steps:*

$$| \lambda_p |^k = e^{\xi_a k \omega h} \tag{43}$$

*Period error after k steps:*

$$\triangle T = (\frac{1}{f_a} - 1)kh \tag{44}$$

The above two expressions can thus provide the global error of $u^n$ after $k$ integration steps if $q_{n-k}$ was the true initial condition.

Numerical Damping and Frequency Errors of One-Step Formulas

The principal root for the one-step formula given by (-22) becomes with $\mathbf{s} = \pm j\omega$

$$\lambda_p = \frac{1 - \alpha(1-\alpha)\omega^2h^2}{1 + \alpha^2\omega^2h^2} + J\frac{\omega h}{1 + \alpha^2\omega^2h^2} \tag{45}$$

so that one obtains

$$\xi_\beta = \frac{1}{(\omega h)} \ln\{\frac{(1 + \alpha(1-\alpha)\omega^2h^2)}{(1 + \alpha^2\omega^2h^2)}\}^{\frac{1}{2}} \tag{46}$$

for the artificial damping ratio, and

$$f_\beta = \frac{1}{(\omega h)} \tan^{-1}\{\frac{(\omega h}{1-\alpha(1-\alpha)\omega^2 h^2)}\} \tag{47}$$

for the artificial frequency distortion.

The artificial damping and the artificial frequency distortion curves for some of the popular one-step formulas are illustrated in Figs. 4 and 5.

## Quantitive Analysis of Stability

So far we have learned how to draw the stability regions of integration formulas and map them onto the **s***h*-plane of the characteristic roots of the governing equations. We have also learned how the two combined characteristics of the formulas and the governing equations of motion manifest themselves in the numerical solutions. Although it is not an intention to cover why certain formulas become only conditionally stable and others possess unconditional stability, it is instructive to examine why the solution of the resulting difference equations becomes stable or unstable, and if conditionally stable, how to determine the stability limit. We will call such an analysis *quantitative stability analysis*.

To this end, let us recall that integration formulas can be characterized by the equation

$$\rho(\lambda)q^n - h\sigma(\lambda)\dot{q}^n = 0 \tag{48}$$

$$\hat{\rho}(\lambda)\dot{q}^n - h\hat{\sigma}(\lambda)\ddot{q}^n = 0 \tag{49}$$

which is applied to solve

$$\mathbf{m}\ddot{q}^n + \mathbf{d}\dot{q}^n + \mathbf{k}q^n = f \tag{50}$$

First, substitute (49) into (50) to yield

$$\mathbf{m}\frac{\hat{\rho}}{\hat{\sigma}}\dot{q}^n + h\mathbf{d}\dot{q}^n + h\mathbf{k}q^n = 0 \tag{51}$$

Second, substitute (48) into (51) to obtain

$$\{\{\mathbf{m}\frac{\hat{\rho}}{\hat{\sigma}} + h\mathbf{d}\}\frac{\rho}{\sigma} + h^2\mathbf{k}\}q^n = 0 \tag{52}$$

Rearranging the above equation we obtain the following characteristic equation

$$\mathbf{m}\hat{\rho}\rho + h\mathbf{d}\rho\hat{\sigma} + h^2\mathbf{k}\sigma\hat{\sigma} = 0 \tag{53}$$

Now, instead of drawing the stability region on the $\mathbf{s}h$-plane as we have done in the preceding sections, we first map the unit circle, $\lambda = 1$, onto another plane by the following transformation

$$\lambda = \frac{z+1}{z-1} \tag{54}$$

which effectively occupies the entire left-hand $z$-plane. Obviously, there exists a unique relation between the $z$-plane and $\mathbf{s}h$-plane. What we are after is that we would like to cast (53) into a $z$-polynomial equation with real coefficients. Observe that the unit circle boundary on the $\lambda$-plane is mapped onto the imaginary axis on the $z$-plane and the entire region inside of the $\lambda$-unit circle is mapped onto the entire left-hand $z$-plane. Hence, if any root of the $z$-polynomial equation has positive real part, we will know that the formulas (48) and (49) at that frequency and at that damping ratio will give unbounded solution. It is this observation that we will exploit here.

Let us introduce the transformation given by (54) into (53) so that we

have

$$\mathbf{m}\hat{\rho}(z)\rho(z) + h\mathbf{d}\rho(z)\hat{\sigma}(z) + h^2\mathbf{k}\sigma(z)\hat{\sigma}(z) = 0 \tag{55}$$

Our task is now reduced to how to determine whether any roots will yield positive real parts in the above polynomial equation without actually solving it! It turns out that there exists a procedure to do just that developed by Routh in 1877 and subsequently simplified by Hurwitz in 1895. Let us summarize for our present purposes their stability criterion as:

*For a quadratic equation*

$$a_0 z^2 + a_1 z + a_2 = 0 \tag{56}$$

*in order for the two roots to have semi-negative real parts, we must have*

$$a_0 \geq 0, \quad a_1 \geq 0, \quad a_2 \geq 0 \tag{57}$$

*For a fourth-order equation*

$$a_0 z^4 + a_1 z^3 + a_2 z^2 + a_3 z + a_4 = 0 \tag{58}$$

*we must have*

$$\{a_i \geq 0, i = 0 \ldots 4\}, \qquad a_3(a_1 a_2 - a_0 a_3) > a_1^2 a_4 \tag{59}$$

*the cubic case can be obtained by setting* $a_0 = 0$ *in the above criterion.*

For high-order cases the reader is referred to the text by Gantmacher.

**Example: The Trapezoidal Rule**

For this case we have

$$\hat{\rho} = \rho = \lambda - 1, \qquad \hat{\sigma} = \sigma = \frac{1}{2}(\lambda + 1) \tag{60}$$

Upon transforming from $\lambda$ to the $z$-plane, we have

$$\hat{\rho} = \rho = \frac{2}{z-1}, \qquad \hat{\sigma} = \sigma = \frac{z}{z-1} \tag{61}$$

whose substitution into (55) yields

$$4\mathbf{m} + 2h\mathbf{d}z + h^2\mathbf{k}z^2 = 0 \tag{62}$$

Hence, the trapezoidal rule is unconditionally stable provided $\mathbf{m} \geq 0, \quad \mathbf{d} \geq 0, \quad \mathbf{k} \geq 0$.