THE UNIMPORTANCE OF THE SPURIOUS ROOT OF TIME INTEGRATION ALGORITHMS FOR STRUCTURAL DYNAMICS

G. M. HULBERT[1] AND J. CHUNG[2]

Department of Mechanical Engineering and Applied Mechanics, 2250 G. G. Brown, The University of Michigan, Ann Arbor, Michigan 48109–2125, U.S.A.

SUMMARY

Most commonly used second-order-accurate, dissipative time integration algorithms for structural dynamics possess a spurious root. For an algorithm to be accurate, it has been suggested that the spurious root must be small and ideally be zero in the low-frequency limit. In the paper we show that good accuracy can be achieved even if the spurious root does not tend towards zero in the low-frequency limit. This permits more flexibility in the design of time integration algorithms. As an example, we present an algorithm that has greater accuracy than several other dissipative algorithms even though for all frequencies its spurious root is non-zero. We also show that the degraded performance of the Bazzi-ρ algorithm is not due to its non-zero spurious root.

1. INTRODUCTION

For many structural dynamics problems, it is desirable to integrate the equations of motion using a time integration algorithm that possesses numerical dissipation. Numerical dissipation helps control the non-physical high-frequency oscillations that are artefacts of standard finite-element modelling of the spatial domain. While numerically dissipative algorithms exist within the Newmark family of algorithms,[1] it is well known that these dissipative methods are only first-order-accurate. To circumvent this loss of accuracy, numerous time integration methods have been developed that retain the basic algorithmic structure of the Newmark scheme and exhibit second-order accuracy and numerical dissipation, e.g. the α method of Hilber, Hughes and Taylor (HHT-α),[2-4] the α method of Wood, Bossak and Zienkiewicz (WBZ-α),[5] the ρ method of Bazzi and Anderheggen (ρ),[6] the θ_1 method of Hoff and Pahl (θ_1)[7,8] and the generalized-α method recently developed by the authors.[9] All of the above algorithms are unconditionally stable for linear problems and are clearly consistent since they are at least first-order-accurate. The basic difference between these algorithms and the Newmark algorithm is that the Newmark method is effectively a two-stage (or two-level) algorithm while the second-order-accurate dissipative algorithms are three-stage methods (see Reference 10 for the definition of a p-stage method). Three-stage methods have additional freedom, compared with two-stage methods, that enables high-frequency dissipation to be achieved without sacrificing accuracy. However, along with this additional freedom the three-stage methods have a so-called *spurious root*. The spurious root has a strictly real value whereas for underdamped

---

[1] Assistant Professor of Mechanical Engineering and Applied Mechanics.
[2] Graduate Research Assistant.

systems the roots of the exact equation are complex. Therefore, the influence of the spurious root on the numerical solution must remain small if a three-stage algorithm is to maintain accuracy. The most direct approach towards minimizing the spurious root influence is to use an algorithm with a small spurious root or a spurious root that tends towards zero in the low-frequency limit. For algorithms used to solve first-order differential equations, Crandall[11] suggested monitoring of oscillatory solutions and time-step restrictions to bound the error induced by *unstable* spurious roots. In the analysis that follows, we show that the influence of the spurious root in three-stage methods does not depend exclusively on its magnitude. Thus an alternative approach exists to minimize the spurious root influence.

2. ANALYSIS

To study the effect of the spurious root, it suffices to consider the single-degree-of-freedom model problem:

$$\ddot{u} + 2\xi\omega\dot{u} + \omega^2 u = 0 \qquad (1)$$

$$u(0) = d_0 \qquad (2)$$

$$\dot{u}(0) = v_0 \qquad (3)$$

where u is the displacement ξ is the damping ratio, ω is the undamped natural frequency, a superposed dot denotes differentiation with respect to time, and d_0 and v_0 are the initial displacement and velocity, respectively. Three-stage methods for solving (1)–(3) may be written in the form

$$\mathbf{A}_1 \mathbf{X}_{n+1} = \mathbf{A}_2 \mathbf{X}_n \qquad (4)$$

where $\mathbf{X}_n^T = \{d_n, \Delta t v_n, \Delta t^2 a_n\}$ in which d_n, v_n and a_n are the respective approximations to $u(t_n)$, $\dot{u}(t_n)$ and $\ddot{u}(t_n)$, $\Delta t = t_{n+1} - t_n$ is the time step, n is the time step number and $\mathbf{A}_1$ and $\mathbf{A}_2$ are defined by a particular time integration algorithm. Multiplying both sides of (4) by $\mathbf{A}_1^{-1}$,

$$\mathbf{X}_{n+1} = \mathbf{A}\mathbf{X}_n \qquad (5)$$

in which $\mathbf{A}$ is the 3×3 amplification matrix. Assuming the eigenvalues of $\mathbf{A}$ are distinct, the discrete displacement response expression in (5) may be written as

$$d_n = c_1 \lambda_1^n + c_2 \lambda_2^n + C_3 \lambda_3^n \qquad (6)$$

in which the c_is are determined from initial conditions and λ_i denotes the ith eigenvalue of $\mathbf{A}$. Since algorithmic accuracy is of concern, we assume that the principal roots remain complex conjugate within the Nyquist sampling range, i.e. $\Delta t/T < 0\cdot5$, where T is the period of vibration. In (6), the roots are arranged such that λ_3 is the spurious root.

The solution of (1)–(3) may be written in a form similar to (6):

$$u(t_n) = c_1^\epsilon (\lambda_1^\epsilon)^n + c_2^\epsilon (\lambda_2^\epsilon)^n \qquad (7)$$

where $c_1^\epsilon = d_0$, $c_2^\epsilon = (v_0 + \xi\omega d_0)/\omega_d$, $\lambda_1^\epsilon = \exp(-\xi\omega\Delta t + i\omega_d\Delta t)$ and $\lambda_2^\epsilon = \exp(-\xi\omega\Delta t - i\omega_d\Delta t)$, in which $\omega_d = \sqrt{(1 - \xi^2)}\omega$ and $i = \sqrt{-1}$.

Comparing (6) and (7), it can be seen that the influence of the spurious root on the displacement response is dictated not only by the magnitude of the spurious root but also by c_3. Also, one may conclude that an algorithm can be spectrally accurate but still produce inaccurate solutions. That is, λ_1 and λ_2 may be quite similar to the roots of the exact solution (spectral accuracy), but errors in c_1 and c_2 could result in poor solutions.

A numerical study of the c_is was conducted for the HHT-α, WBZ-α, ρ, θ_1 and generalized-α algorithms. Also included was the trapezoidal rule algorithm since it has no spurious root and thus can be used as a baseline for comparison. In the low-frequency limit ($\Delta t/T \to 0$), the spurious root is 0, 0, $\alpha/(\alpha - 1)$, $(1 - \rho - \rho^2 - \rho^3)/2$, 0 and $(1 - 2\rho_\infty)/(2 - \rho_\infty)$ for the trapezoidal, HHT-α, WBZ-α, ρ, θ_1 and generalized-α algorithms. respectively. Thus. the spurious root is non-zero for the dissipative forms of the WBZ-α, ρ and generalized-α methods.

Given **A**, the c_is may be computed given d_0, v_0, d_1 and d_2 (see Reference 2). The following values were chosen for the model problem parameters: $\omega = 2\pi$, $\xi = 0$, $d_0 = 1$ and $v_0 = 1$. The undamped case was chosen because the ρ method is only first-order-accurate when physical damping is present. Specific values of algorithmic parameters were chosen such that the spectral radius in the high-frequency limit was $0 \cdot 8$ for all algorithms except the trapezoidal rule, for which the spectral radius is unity for all frequencies.

Figures 1–3 show the variation in the c_is as a function of $\Delta t/T$. The coefficients c_1 and c_2 are normalized with respect to c_1^e and c_2^e, respectively. As expected, the values of trapezoidal rule coefficients are nearly the same as the exact values; more precisely, for the trapezoidal rule, $c_1 = c_1^e$ and $c_3 = c_3^e = 0$. Among the numerically dissipative algorithms, coefficient errors are smallest for the generalized-α method within the temporally resolved frequency region of engineering interest ($\Delta t/T \leqslant 0 \cdot 1$, i.e. at least ten time steps per period). Since the spurious root influence on the solution from one time step to the next is given by $c_3\lambda_3$ (recall (6)), a fairer comparison of the spurious root influence is shown in Figure 4 which shows the variation in $c_3\lambda_3$. With the exception of the ρ method, the value of $c_3\lambda_3$ is similar for all dissipative algorithms shown. Of particular importance is the magnitude of $c_3\lambda_3$, compared to the errors of c_1 and c_2, the $c_3\lambda_3$ error is particularly small. Therefore, the spurious root, even if non-zero, contributes little to the total error of the algorithms studied.

In Reference 7, the ρ method was noted to have poor performance even when physical damping was absent; it was suggested that its non-zero spurious root is the cause of the large
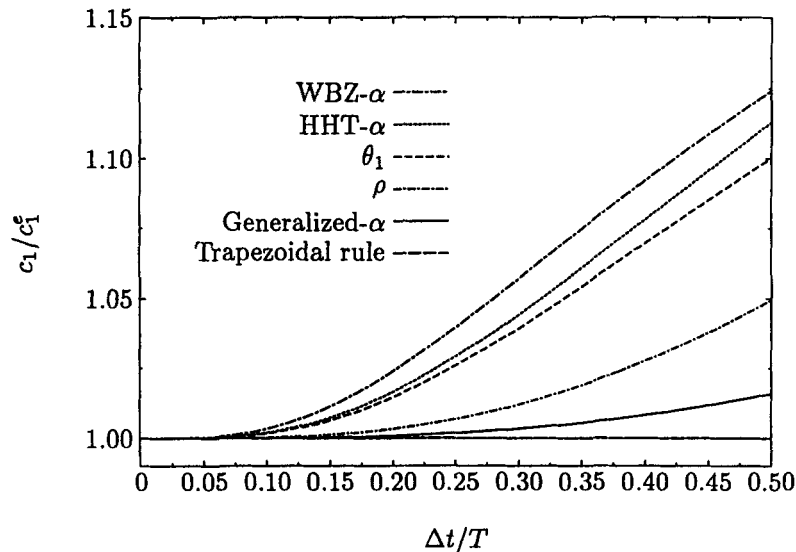

Figure 1. Variation in c_1 coefficient of discrete displacement response expression

errors. From Figures 2 and 4, it can be seen that the substantial error in c_2 is more responsible for the inaccurate computations obtained with the ρ method. To test this assertion, the model problem was solved using the generalized-α and ρ methods; the model parameters given previously were employed and $\Delta t = 0 \cdot 1$. The generalized-α method was chosen as the dissipative baseline algorithm because we have found it has the smallest error among the dissipative algorithm of this study. Also, a 'modified' ρ method was constructed by replacing c_2 of the ρ method with c_2 of the generalized-α method; the other values in the discrete
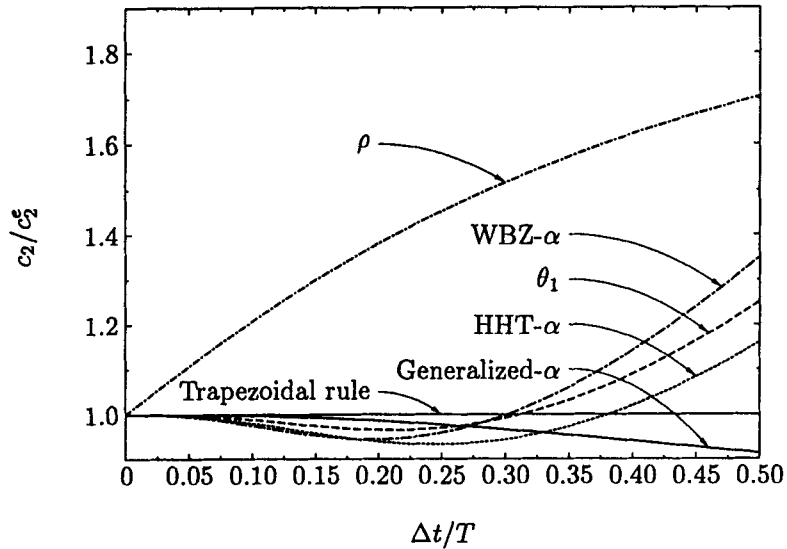

Figure 2. Variation in c_2 coefficient of discrete displacement response expression
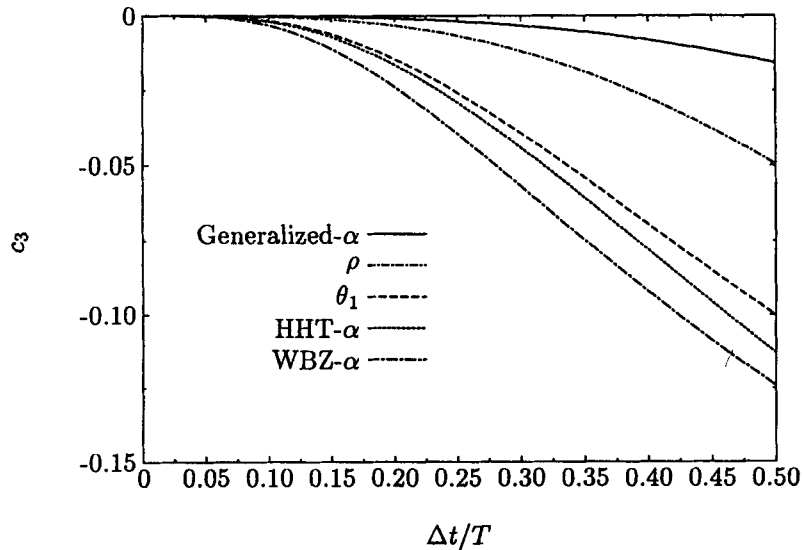

Figure 3. Variation in c_3 coefficient of discrete displacement response expression

response expression (6) of the ρ method were not changed. While the modified ρ method has no direct three-stage counterpart, it provides a useful test of the relative influence of the c_2 and spurious root error contributions. The displacement errors of these three algorithms are shown in Figure 5. Note that the error of the modified ρ method is nearly the same as the generalized-α method. Thus, the poor performance of the ρ method is attributed to the error in c_2 rather than its non-zero spurious root.
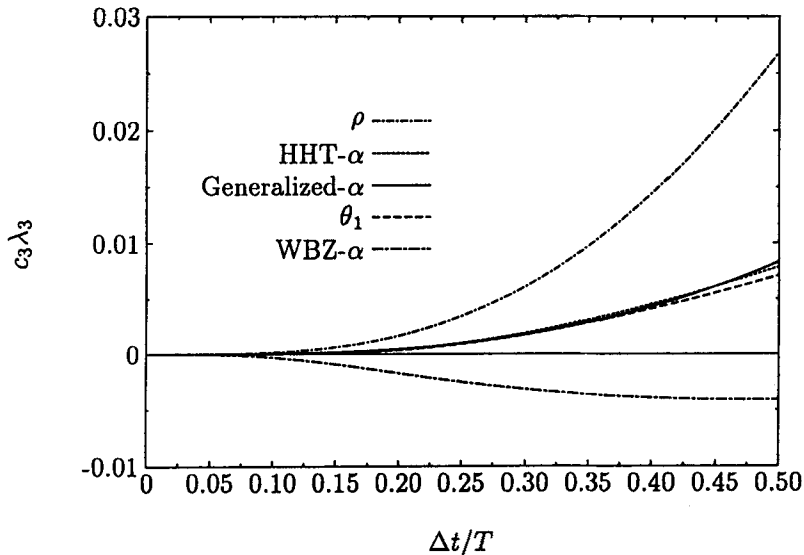

Figure 4. Variation in $c_3\lambda_3$ of discrete displacement response expression
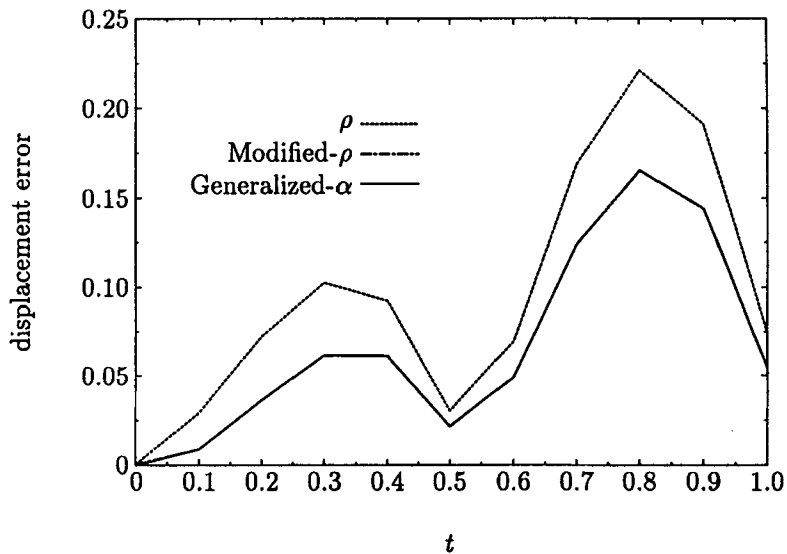

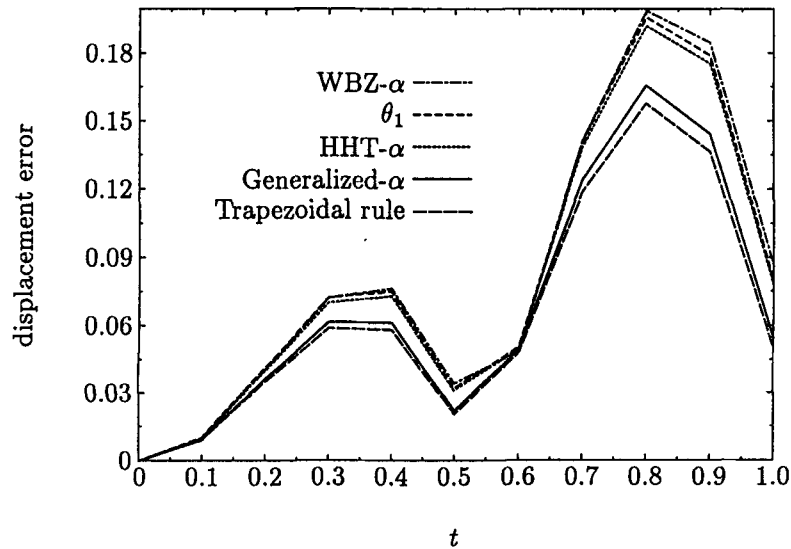Figure 5. Displacement error time histories of generalized-α, ρ and modified ρ algorithms

Figure 6. Displacement error time histories of trapezoidal, HHT-α, WBZ-α θ_1 and generalized-α algorithms

To further substantiate the unimportance of a non-zero spurious root in the low-frequency limit, displacement errors are compared in Figure 6 for the trapezoidal, HHT-α, WBZ-α, θ_1 and generalized-α algorithms (the model problem employed is the same as described in the preceding paragraph). It may be seen that the error of the generalized-α method is closer to that of the trapezoidal rule than the other algorithms for which the spurious root is zero in the low-frequency limit; this is the expected result based upon the c_i errors shown in Figures 1, 2 and 4.

3. CLOSING REMARKS

When designing time integration methods to solve structural dynamics problems, it is essential to assess the influence of spurious roots that are present in three-stage (and higher-stage) algorithms. While algorithms can be designed such that the spurious root is zero in the low-frequency limit, we have found that this design condition is too restrictive and not necessary to minimize undesirable spurious root behaviour. An alternative design requirement is to ensure that the c_3 coefficient that multiplies the spurious root is small and tends to zero in the low-frequency limit. The generalized-α method is one such algorithm that possesses a non-zero spurious root yet exhibits better performance than other numerically dissipative algorithms studied in this paper. A more detailed discussion and comparison of the generalized-α method may be found in Reference 9.

REFERENCES

1. N. M. Newmark, 'A method of computation for structural dynamics,' *J. Eng. Mech. Div. ASCE*, **85** (EM3), 67–94 (1959).
2. H. M. Hilber, 'Analysis and design of numerical integration methods in structural dynamics,' *PhD. thesis*, Earthquake Engineering Research Center, University of California, Berkeley, 1976.

3. H. M. Hilber, T. J. R. Hughes and R. L. Taylor, 'Improved numerical dissipation for time integration algorithms in structural dynamics,' *Earthquake eng. struct. dyn.*, **5**, 283–292 (1977).
4. H. M. Hilber and T. J. R. Hughes, 'Collocation, dissipation and 'overshoot' for time integration schemes in structural dynamics,' *Earthquake eng. struct. dyn.*, **6**, 99–118 (1978).
5. W. L. Wood, M. Bossak and O. C. Zienkiewicz, 'An alpha modification of Newmark's method,' *Int. j. numer. methods eng.*, **15**, 1562–1566 (1981).
6. G. Bazzi and E. Anderheggen, 'The ρ-family of algorithms for time-step integration with improved numerical dissipation,' *Earthquake eng. struct. dyn.*, **10**, 537–550 (1982).
7. C. Hoff and P. J. Pahl, 'Practical performance of the θ_1 method and comparison with other dissipative algorithms in structural dynamics,' *Comput. methods appl. mech. eng.*, **67**, 87–110 (1988).
8. C. Hoff and P. J. Pahl, 'Development of an implicit method with numerical dissipation from a generalized single-step algorithm for structural dynamics' *Comput. methods appl mech. eng.*, **67**, 367–385 (1988).
9. J. Clung and G. M. Hulbert, 'A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method,' *ASME J. Appl. Mech.*, **60**, 371–375 (1993).
10. W. L. Wood, *Practical Time-Stepping Schemes*, Clarendon Press, Oxford, 1990.
11. S. H. Crandall, *Engineering Analysis. A Survey of Numerical Procedures*, McGraw-Hill, New York, 1956.