

Stability Analysis of Ubiquitous Direct Time Integration Methods

M. Naguib¹, A. A. Ghaleb², F. M. Amin³

¹ Mohammed Naguib Abou El-Saad, Professor of structural analysis and mechanics
Structural Engineering Department, Faculty of Engineering, Mansoura University, Egypt.

² Ahmed Amin Ghaleb, Associate professor of concrete structures
Structural Engineering Department, Faculty of Engineering, Mansoura University, Egypt.

³ Fouad Mostafa Fouad Amin, Teaching assistant of structural analysis and mechanics.
Structural Engineering Department, Faculty of Engineering, Mansoura University, Egypt.
Foad.m.amin@gmail.com

Abstract: A brief implementation guide was introduced for direct time integration algorithms including the central difference, Houbolt, Newmark, Wilson- θ , HHT- α , WBZ- α , Generalized- α and Bathe Methods. Later, the stability and accuracy concepts were introduced with summarizing the procedures for spectral analysis for any direct time integration algorithm. Bathe method has shown superiority in different aspects based on spectral analysis. A similarity in spectral performance was found between Houbolt and a special case of WBZ- α . The central difference method also showed similar spectral performance to that shown in a special case of HHT- α .

Keywords: Direct Time Integration, Stability Analysis, Central Difference, Houbolt, Newmark, Wilson- θ , HHT- α , WBZ- α , Generalized- α , Implicit Bathe method.

1. Introduction

Direct time integration is one of the most used techniques to solve the dynamic situation included in many engineering applications. The term 'dynamic situation' refers to the presence of the inertia forces in the equilibrium equation governing the system subjected to dynamic analysis [1].

For a linear system of finite elements with N degrees of freedom, the equilibrium equation governing the dynamic response can be written as

$$M\ddot{U}(t) + C\dot{U}(t) + KU(t) = R(t) \quad (1)$$

where K , C and M are the stiffness, damping and mass matrices; $R(t)$ is the external force vector affecting on the N degrees of freedom at time t . While $U(t)$, $\dot{U}(t)$ and $\ddot{U}(t)$ are the displacement, velocity and acceleration vectors of the system degrees of freedom. The last mentioned three vectors known as state vectors describe the response of the system through the time in which the analyst is interested in.

The term 'direct' used in describing the integration algorithm means that no transformation of the equilibrium equation is done before the numerical integration is carried out. The direct time integration is based on two concepts. First, instead of satisfying (1) at any time t , the aforementioned equation is discretized to be satisfied only at discrete time intervals with time span Δt . Second, the variation between displacement, velocity and acceleration has to be assumed [2]. To apply the first concept, equation (1) will be written as follows.

$$M\ddot{U}_i + C\dot{U}_i + KU_i = R_i \quad (2)$$

where the subscript i indicates the index of step being studied through which the time range varies from t_i to t_{i+1} given that $\Delta t = t_{i+1} - t_i$.

There are two main types of direct integration algorithms: implicit and explicit. According to Noh and Bathe [3], the algorithm is implicit if it requires factorization of the effective stiffness matrix in the solution procedures. Otherwise it's an explicit one. Another definition given by Rezaiee-Pajand and Karimi-Rad [4] which states that in explicit algorithms, the responses at the end of time step are explicitly calculated based on the response of the previous time steps. While in implicit algorithms the equations used for displacement calculation at end of the time step also include velocity and acceleration at the end of time step as unknowns. Which yields a system of equations that needs to be solved (factorization) each time step. A combination between implicit and explicit techniques produces new methods such as predictor-corrector methods in which explicit equations are used to predict the response at the end of time step. Then, implicit formulae are used to correct the answer. Some methods propose repeating the correction phase till reaching the required accuracy which are called predictor-multi-corrector methods [5].

The integration algorithm is said to be unconditionally stable if the response for any initial conditions does not propagate exponentially with no limit at any time step [6]. Explicit algorithms are computationally inexpensive related to the implicit ones. The computational cost per time step is proportional to the number of degrees of freedom on applying explicit methods. However, their numerical stability is an issue. Implicit algorithms tend to be numerically stable but the computational cost per time step is relatively high as they require high storage and processing requirements. Due to the conditional stability usually possessed by explicit algorithms, small time step are selected [7]. The numerical dissipation (or algorithmic damping) introduced by different integration algorithms causes amplitude decay specially for high frequency modes. This property can be of importance in some structural problems due to the analyst desire to damp out high frequency modes [8].

Many investigations and researches have been done to analyze the stability and accuracy of well-known integration algorithms or to conduct new methods with enhanced properties [6], [8], [9], [10], [11], [12], [13].

Among the explicit algorithms the central difference formula is considered the most preferred method as it has relatively good stability properties. Also, its implementation on a computer program is very simple [14], [15]. Houbolt introduced his implicit method which can be related to the central difference method as they both used standard finite difference formulae to approximate velocity and acceleration vectors in terms of displacement vectors [16].

Later, Newmark introduced his well-known method which has two parameters γ and β to control the algorithm performance (i.e. stability and accuracy) [17]. Also, one of the most studied and analyzed methods was introduced by Wilson et al [18]. Named after the author, Wilson θ method has only one parameter θ to control the algorithm. On using $\theta = 1.0$, Wilson θ algorithm turns out to be identical to Newmark linear acceleration one ($\gamma = 1/2$ and $\beta = 1/6$).

Some improvements have been introduced to Newmark method which yielded numerous methods such as HHT- α method [19] in which the authors introduced an additional parameter α_f to modify damping and elastic forces as well as the external force vector on satisfying the equilibrium equation. A similar approach was introduced such that an additional parameter α_m was introduced to modify the inertia forces only on satisfying equilibrium equation. The later approach was named after the authors as WBZ- α method [20].

The concept of using a combination between two methods to get a new one with enhanced properties has been used over decades which produced many methods like the generalized- α method [21] which is considered a combination between HHT- α and WBZ- α methods. Bathe introduced his composite method that uses two separate algorithms to solve two consecutive time sub-steps to reach the solution at the end of time step. First, the suggested two algorithms were Newmark average acceleration ($\gamma = 1/2$ and $\beta = 1/4$) and three-point backward Euler method [22]. Later, Bathe introduced his generalized composite algorithm which is known as implicit Bathe method. Given that the only Newmark average acceleration was replaced by Newmark general algorithm [23]. Instead of splitting time step into two equal sub-steps, Bathe introduced the parameter α_s to divide the time step Δt into two unequal sub-steps $\alpha_s \Delta t$ and $(1 - \alpha_s) \Delta t$.

The following section will be used as a brief illustration of previously mentioned well-known integration algorithms. Section 3 will display the stability and accuracy concepts and our contributions to facilitate stability and accuracy analysis of integration algorithms. Results of stability and accuracy analysis done by authors will be displayed and discussed in section 4.

2. Direct Integration Algorithms

For each algorithm of the following ones, some initial conditions must be known before start. At time t_0 (i.e. $t = 0$) the required initial conditions are $U_0 = U(0)$ and $\dot{U}_0 = \dot{U}(0)$. In case \ddot{U}_0 is required, it can be calculated by solving the equilibrium equation for \ddot{U}_0 at time t_0 using the following equation

$$M\ddot{U}_0 = R_0 - C\dot{U}_0 - KU_0 \quad (3)$$

2.1 Central Difference Method

The central difference formula assumes that

$$\dot{U}_i = \frac{U_{i+1} - U_{i-1}}{2\Delta t} \quad (4)$$

$$\ddot{U}_i = \frac{U_{i+1} - 2U_i + U_{i-1}}{\Delta t^2} \quad (5)$$

On substituting (4) and (5) in (2) and rearranging we get

$$\begin{aligned} \left(\frac{1}{\Delta t^2}M + \frac{1}{2\Delta t}C\right)U_{i+1} \\ = R_i + M\left(\frac{2U_i - U_{i-1}}{\Delta t^2}\right) + C\left(\frac{U_{i-1}}{2\Delta t}\right) \\ - KU_i \end{aligned} \quad (6)$$

After solving (6) for U_{i+1} , the velocity \dot{U}_i and acceleration \ddot{U}_i can be calculated easily using (4), (5) only if they are required by the analyst. On starting the algorithm at ($i = 0$), a value for U_{-1} will be required. As a special starting procedure, the following equation is used

$$U_{-1} = U_0 - \Delta t\dot{U}_0 + \frac{\Delta t^2}{2}\ddot{U}_0 \quad (7)$$

2.2 Houbolt Method

Houbolt proposed the following formulae for approximating \dot{U}_{i+1} and \ddot{U}_{i+1}

$$\ddot{U}_{i+1} = \frac{1}{\Delta t^2}(2U_{i+1} - 5U_i + 4U_{i-1} - U_{i-2}) \quad (8)$$

$$\dot{U}_{i+1} = \frac{1}{6\Delta t}(11U_{i+1} - 18U_i + 9U_{i-1} - 2U_{i-2}) \quad (9)$$

On contrary to the central difference method, equilibrium is satisfied at t_{i+1} instead of t_i . For clarity the equilibrium equation at t_{i+1} should be stated first as

$$M\ddot{U}_{i+1} + C\dot{U}_{i+1} + KU_{i+1} = R_{i+1} \quad (10)$$

then substituting (8), (9) in (10)

$$\begin{aligned} \left(\frac{2}{\Delta t^2}M + \frac{11}{6\Delta t}C + K\right)U_{i+1} \\ = R_{i+1} + \frac{1}{\Delta t^2}M(5U_i - 4U_{i-1} + U_{i-2}) \\ + \frac{1}{6\Delta t}C(18U_i - 9U_{i-1} + 2U_{i-2}) \end{aligned} \quad (11)$$

After solving (11) for U_{i+1} , If required, \dot{U}_{i+1} and \ddot{U}_{i+1} can be calculated using (8), (9). One of the drawbacks for Houbolt method is that it is not a self-starting algorithm. Which means that a special procedure is required to compute U_1 and U_2 . Simply, any self-starting algorithm can be used to calculate only the first two steps as a starting procedure for Houbolt method. Given that all the following time integration algorithms are self-starting.

2.3 Newmark Method

Newmark method's formulation starts with truncating the Taylor series expansion of displacement and velocity with introduction of two control parameters γ and β . The result equations after some mathematical manipulations are

$$\ddot{U}_{i+1} = \frac{1}{\beta\Delta t^2}(U_{i+1} - U_i) - \frac{1}{\beta\Delta t}\dot{U}_i - \frac{1-2\beta}{2\beta}\ddot{U}_i \quad (12)$$

$$\dot{U}_{i+1} = \frac{\gamma}{\beta\Delta t}(U_{i+1} - U_i) - \frac{\gamma-\beta}{\beta}\dot{U}_i - \frac{\gamma-2\beta}{2\beta}\Delta t\ddot{U}_i \quad (13)$$

Using (10) as equilibrium equation and substituting (12) and (13) into it yields

$$\begin{aligned} & \left(\frac{1}{\beta\Delta t^2}M + \frac{\gamma}{\beta\Delta t}C + K\right)U_{i+1} \\ & = R_{i+1} \\ & + M\left(\frac{1}{\beta\Delta t^2}U_i + \frac{1}{\beta\Delta t}\dot{U}_i + \frac{1-2\beta}{2\beta}\ddot{U}_i\right) \quad (14) \\ & + C\left(\frac{\gamma}{\beta\Delta t}U_i + \frac{\gamma-\beta}{\beta}\dot{U}_i + \frac{\gamma-2\beta}{2\beta}\Delta t\ddot{U}_i\right) \end{aligned}$$

Each time step calculations start with solving (14) for U_{i+1} then calculating \dot{U}_{i+1} and \ddot{U}_{i+1} using (12) and (13) respectively.

2.4 Wilson θ Method

Wilson θ method can be considered as an extension of the linear acceleration method in which the acceleration is assumed to vary linearly through time from t_i to t_{i+1} (i.e. t_i to $t_i + \Delta t$). While in Wilson θ method the acceleration is assumed to vary linearly from t_i to $t_{i+\theta}$ (t_i to $t_i + \theta\Delta t$). Equilibrium equation is satisfied at $t_{i+\theta}$ which means that it can be written as

$$M\ddot{U}_{i+\theta} + C\dot{U}_{i+\theta} + KU_{i+\theta} = R_{i+\theta} \quad (15)$$

given that

$$R_{i+\theta} = R_i + \theta(R_{i+1} - R_i) \quad (16)$$

$$\ddot{U}_{i+\theta} = \frac{6}{(\theta\Delta t)^2}(U_{i+\theta} - U_i) - \frac{6}{\theta\Delta t}\dot{U}_i - 2\ddot{U}_i \quad (17)$$

$$\dot{U}_{i+\theta} = \frac{3}{\theta\Delta t}(U_{i+\theta} - U_i) - 2\dot{U}_i - \frac{\theta\Delta t}{2}\ddot{U}_i \quad (18)$$

To solve equilibrium equation for $U_{i+\theta}$ it can be written in the following form by substituting (16), (17), (18) into (15)

$$\begin{aligned} & \left(\frac{6}{(\theta\Delta t)^2}M + \frac{3}{\theta\Delta t}C + K\right)U_{i+\theta} \\ & = \theta R_{i+1} + (1-\theta)R_i \\ & + M\left(\frac{6}{(\theta\Delta t)^2}U_i + \frac{6}{\theta\Delta t}\dot{U}_i + 2\ddot{U}_i\right) \quad (19) \\ & + C\left(\frac{3}{\theta\Delta t}U_i + 2\dot{U}_i + \frac{\theta\Delta t}{2}\ddot{U}_i\right) \end{aligned}$$

The algorithm goes straight forward by evaluating $U_{i+\theta}$ using (19) then calculating response at time t_{i+1} using the following three equations

$$\ddot{U}_{i+1} = \frac{6}{\theta^3\Delta t^2}(U_{i+\theta} - U_i) - \frac{6}{\theta^2\Delta t}\dot{U}_i + \left(1 - \frac{3}{\theta}\right)\ddot{U}_i \quad (20)$$

$$\dot{U}_{i+1} = \dot{U}_i + \frac{\Delta t}{2}(\ddot{U}_{i+1} + \ddot{U}_i) \quad (21)$$

$$U_{i+1} = U_i + \Delta t\dot{U}_i + \frac{\Delta t^2}{6}(\ddot{U}_{i+1} + 2\ddot{U}_i) \quad (22)$$

2.5 HHT- α Method

As mentioned before HHT- α method is considered a modification to Newmark general algorithm in which the satisfied equilibrium equation is written as

$$M\ddot{U}_{i+1} + C\dot{U}_{i+1-\alpha_f} + KU_{i+1-\alpha_f} = R_{i+1-\alpha_f} \quad (23)$$

given that

$$\dot{U}_{i+1-\alpha_f} = (1-\alpha_f)\dot{U}_{i+1} + \alpha_f\dot{U}_i \quad (24)$$

$$U_{i+1-\alpha_f} = (1-\alpha_f)U_{i+1} + \alpha_fU_i \quad (25)$$

$$R_{i+1-\alpha_f} = (1-\alpha_f)R_{i+1} + \alpha_fR_i \quad (26)$$

Using (12), (13) with (24), (25), (26) and substituting into (23), it can be easily found that

$$\begin{aligned} & \left(\frac{1}{\beta\Delta t^2}M + \frac{(1-\alpha_f)\gamma}{\beta\Delta t}C + (1-\alpha_f)K\right)U_{i+1} \\ & = (1-\alpha_f)R_{i+1} + \alpha_fR_i \\ & + M\left(\frac{1}{\beta\Delta t^2}U_i + \frac{1}{\beta\Delta t}\dot{U}_i + \frac{1-2\beta}{2\beta}\ddot{U}_i\right) \quad (27) \\ & + C\left(\frac{(1-\alpha_f)\gamma}{\beta\Delta t}U_i + \frac{(1-\alpha_f)\gamma-\beta}{\beta}\dot{U}_i + \frac{(\gamma-2\beta)(1-\alpha_f)}{2\beta}\Delta t\ddot{U}_i\right) - \alpha_fKU_i \end{aligned}$$

On solving (27) for U_{i+1} then substituting in (12),(13), the three state vectors (U_{i+1} , \dot{U}_{i+1} and \ddot{U}_{i+1}) will be calculated at time t_{i+1} .

2.6 WBZ- α Method

Similar to HHT- α , WBZ- α also introduced a modified equilibrium equation

$$M\ddot{U}_{i+1-\alpha_m} + C\dot{U}_{i+1} + KU_{i+1} = R_{i+1} \quad (28)$$

given that

$$\ddot{U}_{i+1-\alpha_m} = (1-\alpha_m)\ddot{U}_{i+1} + \alpha_m\ddot{U}_i \quad (29)$$

Using Newmark approximations (12), (13) with (29) in (28), the modified equilibrium equation can be written as

$$\begin{aligned} & \left(\frac{1-\alpha_m}{\beta\Delta t^2}M + \frac{\gamma}{\beta\Delta t}C + K\right)U_{i+1} \\ & = R_{i+1} \\ & + M\left(\frac{1-\alpha_m}{\beta\Delta t^2}U_i + \frac{1-\alpha_m}{\beta\Delta t}\dot{U}_i + \frac{1-\alpha_m-2\beta}{2\beta}\ddot{U}_i\right) \quad (30) \\ & + C\left(\frac{\gamma}{\beta\Delta t}U_i + \frac{\gamma-\beta}{\beta}\dot{U}_i + \frac{\gamma-2\beta}{2\beta}\Delta t\ddot{U}_i\right) \end{aligned}$$

Like to HHT- α , the three state vectors will be calculated by solving (30) for U_{i+1} then using (12) and (13) to get \dot{U}_{i+1} and \ddot{U}_{i+1} respectively.

2.7 Generalized- α Method

The generalized- α is a combination between HHT- α and WBZ- α methods. The modified equilibrium equation can be written as

$$M\ddot{U}_{i+1-\alpha_m} + C\dot{U}_{i+1-\alpha_f} + KU_{i+1-\alpha_f} = R_{i+1-\alpha_f} \quad (31)$$

Using (12),(13) with (24),(25),(26),(29) and substituting in (31), The resultant system of equations will be given as

$$\begin{aligned} & \left(\frac{1-\alpha_m}{\beta\Delta t^2} M + \frac{(1-\alpha_f)\gamma}{\beta\Delta t} C + (1-\alpha_f)K \right) U_{i+1} \\ & = (1-\alpha_f)R_{i+1} + \alpha_f R_i \\ & + M \left(\frac{1-\alpha_m}{\beta\Delta t^2} U_i + \frac{1-\alpha_m}{\beta\Delta t} \dot{U}_i \right. \\ & \left. + \frac{1-\alpha_m-2\beta}{2\beta} \ddot{U}_i \right) \\ & + C \left(\frac{(1-\alpha_f)\gamma}{\beta\Delta t} U_i \right. \\ & \left. + \frac{(1-\alpha_f)\gamma-\beta}{\beta} \dot{U}_i \right. \\ & \left. + \frac{(\gamma-2\beta)(1-\alpha_f)}{2\beta} \Delta t \ddot{U}_i \right) - \alpha_f K U_i \end{aligned} \quad (32)$$

As a combination between HHT- α and WBZ- α methods, the generalized- α algorithm starts for each time step by calculating U_{i+1} using (32) then substituting in (12) and (13) to get \ddot{U}_{i+1} and \dot{U}_{i+1} respectively.

2.8 Bathe Implicit Method

Bathe Implicit method divided the time step calculations into two sub-steps. The first sub-step ends at ($t_{i+\alpha_s} = t_i + \alpha_s \Delta t$). Where the equilibrium equation can be given as

$$M\ddot{U}_{i+\alpha_s} + C\dot{U}_{i+\alpha_s} + KU_{i+\alpha_s} = R_{i+\alpha_s} \quad (33)$$

where the subscript $i+\alpha_s$ indicates that the vectors are calculated at time $t_{i+\alpha_s}$. The first sub-step calculations can be described using the following equations.

$$\begin{aligned} & \left(\frac{1}{\beta(\alpha_s\Delta t)^2} M + \frac{\gamma}{\beta\alpha_s\Delta t} C + K \right) U_{i+\alpha_s} \\ & = (1-\alpha_s)R_i + \alpha_s R_{i+1} \\ & + M \left(\frac{1}{\beta(\alpha_s\Delta t)^2} U_i + \frac{1}{\beta\alpha_s\Delta t} \dot{U}_i \right. \\ & \left. + \frac{1-2\beta}{2\beta} \ddot{U}_i \right) \\ & + C \left(\frac{\gamma}{\beta\alpha_s\Delta t} U_i + \frac{\gamma-\beta}{\beta} \dot{U}_i \right. \\ & \left. + \frac{(\gamma-2\beta)\alpha_s\Delta t}{2\beta} \ddot{U}_i \right) \end{aligned} \quad (34)$$

$$\dot{U}_{i+\alpha_s} = \frac{\gamma}{\beta\alpha_s\Delta t} (U_{i+\alpha_s} - U_i) - \frac{\gamma-\beta}{\beta} \dot{U}_i - \frac{(\gamma-2\beta)\alpha_s\Delta t}{2\beta} \ddot{U}_i \quad (35)$$

$$\ddot{U}_{i+\alpha_s} = \frac{1}{\beta(\alpha_s\Delta t)^2} (U_{i+\alpha_s} - U_i) - \frac{1}{\beta\alpha_s\Delta t} \dot{U}_i - \frac{1-2\beta}{2\beta} \ddot{U}_i \quad (36)$$

The second sub-step calculations compute the three state vectors at time t_{i+1} using previously calculated values at t_i and $t_{i+\alpha_s}$ which can be summarized as

$$\begin{aligned} & (c_3^2 M + c_3 C + K) U_{i+1} \\ & = R_{i+1} \\ & - M(c_1 \dot{U}_i + c_2 \dot{U}_{i+\alpha_s} + c_3 c_1 U_i \\ & + c_3 c_2 U_{i+\alpha_s}) - C(c_1 U_i + c_2 U_{i+\alpha_s}) \end{aligned} \quad (37)$$

$$\dot{U}_{i+1} = c_1 \dot{U}_i + c_2 \dot{U}_{i+\alpha_s} + c_3 \dot{U}_{i+1} \quad (38)$$

$$\ddot{U}_{i+1} = c_1 \ddot{U}_i + c_2 \ddot{U}_{i+\alpha_s} + c_3 \ddot{U}_{i+1} \quad (39)$$

where

$$c_1 = \frac{1-\alpha_s}{\alpha_s\Delta t}, \quad c_2 = \frac{-1}{(1-\alpha_s)\alpha_s\Delta t}, \quad c_3 = \frac{2-\alpha_s}{(1-\alpha_s)\Delta t} \quad (40)$$

It should be noted that for any further discussions in this paper Bathe implicit method will be denoted only as Bathe method.

3. Stability and Accuracy Analysis

Stability of a direct integration algorithm means that for any initial conditions, the response vectors should not be amplified artificially which affects the solution accuracy. Also, stability means that any round-off errors in acceleration, velocity and displacement vectors do not grow with integration progress [6].

To study an algorithm performance based on accuracy and stability, it was found satisfactory to study a single degree of freedom system in its free vibration state [2]. In that case the equilibrium equation can be written as

$$m\ddot{u} + c\dot{u} + ku = r \quad (41)$$

or

$$\ddot{u} + 2\xi\omega\dot{u} + \omega^2 u = \hat{r} \quad (42)$$

given that m, c and k are the mass, damping and stiffness of a single degree of freedom system with periodic time T and response values \ddot{u}, \dot{u} and u as acceleration, velocity and displacement. r is the external load and \hat{r} is the external load divided by m . The two parameters ξ and ω are the damping coefficient and natural frequency of the single degree of freedom system which are related to the system properties as follows.

$$\omega = \sqrt{k/m}, \quad T = 2\pi/\omega \quad (43)$$

$$c = 2\xi\sqrt{km} \quad (44)$$

For any integration algorithm we can express its procedures on a single degree of freedom in one recursive function given as

$$\hat{u}_{i+1} = A\hat{u}_i + L\hat{r} \quad (45)$$

where \hat{u}_{i+1} and \hat{u}_i are vectors of response values included in the integration algorithm for one step. L is called the load operator vector and A is known as integration approximation matrix or amplification matrix. In case of external load absence, the algorithm is governed only by the amplification matrix [11]. The methodology used in calculation of the amplification matrix for direct integration algorithms studied in this paper can be found in [6] and [15]. Hilber and Hughes [9] stated that the spectral properties of the amplification matrix determine many important aspects of the algorithm. The spectral radius of the amplification matrix can be given as

$$\rho = \max \{|\lambda_1|, |\lambda_2|, |\lambda_3|\} \quad (46)$$

where λ denotes the eigenvalue of A governed by

$$\det(A - \lambda I) = 0 \tag{47}$$

and I is the identity matrix. For any convergent algorithm, two eigenvalues of the three calculated ones will be two complex conjugate eigenvalues $\lambda_{1,2}$. The two eigenvalues $\lambda_{1,2}$ are called the principle roots and so, λ_3 is called a spurious root which satisfy $|\lambda_3| \leq |\lambda_{1,2}| \leq 1$. It is worth mentioning that the spectral radius at $\Delta t/T \rightarrow \infty$ is called the ultimate spectral radius [24] and given symbol ρ_∞ .

Stability of any algorithm is assured if the spectral radius $\rho \leq 1$ for any value of time step Δt [9]. According to Noh and Bathe [23], the period elongation (PE) and amplitude decay (AD) seen in the undamped free vibration calculations of a single degree of freedom oscillator are results to numerical dispersion and dissipation respectively caused by the integration algorithm. More details and definitions for calculating PE and AD are found in [9] and [11]. Here, only the final formulae used for calculation of PE and AD are displayed.

$$\bar{\xi} = \frac{-\ln(\rho)}{\omega \Delta t} \tag{48}$$

$$\bar{\omega} = \frac{\tan^{-1}(B_\lambda/A_\lambda)}{\Delta t} \tag{49}$$

$$PE = \frac{\bar{T} - T}{T} = \frac{\omega - \bar{\omega}}{\bar{\omega}} \tag{50}$$

$$AD = 1 - \exp\left(\frac{-2\pi\bar{\xi}\omega}{\bar{\omega}}\right) \tag{51}$$

where

$$\lambda_{1,2} = A_\lambda \pm B_\lambda i \tag{52}$$

$$\rho = \sqrt{A_\lambda^2 + B_\lambda^2} \tag{53}$$

$$\bar{T} = 2\pi/\bar{\omega} \tag{54}$$

For better understanding of algorithms behavior, the eigenvalues for each amplification matrix of previously discussed integration algorithms were written as a function of $\Delta t/T$ and the algorithm parameters using a MATLAB script which implemented MATLAB symbolic toolbox. The evaluated equations were later imported to a Fortran code to calculate ρ , PE and AD for each algorithm with variation of $\Delta t/T$. It should be mentioned that many papers have studied the stability and accuracy of time integration algorithms, however the authors interest in this paper is to elaborate more details and study more parameters affecting that domain.

4. Study Parameters

Damping coefficient ξ is one of the parameters that affect the stability and accuracy study. Here, we will only focus on the undamped case in which $\xi = 0$. Some studied algorithms have no special control parameters like central difference and Houbolt methods which means in our study only one case for each of these methods is shown. Bathe method has three control parameters (γ, β, α_s). Most recent studies [2] on Bathe method only concentrated on changing α_s and setting $\gamma = 1/2$ and $\beta = 1/4$.

The ultimate spectral radius ρ_∞ was used as a control parameter in some methods in which their original control parameters can be written as a function of ρ_∞ . The following equations in Table 1 was given by [25] in which control parameters of some algorithms are given as a function of ρ_∞ .

Table 1: Control parameters as a function of ultimate spectral radius ρ_∞ .

Algorithm	α_f	α_m	γ	β
Newmark	0	0	$\frac{3 - \rho_\infty}{2\rho_\infty + 2}$	$\frac{1}{(\rho_\infty + 1)^2}$
HHT- α	$\frac{1 - \rho_\infty}{\rho_\infty + 1}$	0	$\frac{1}{2} + \alpha_f$	$\frac{(1 + \alpha_f)^2}{4}$
WBZ- α	0	$\frac{\rho_\infty - 1}{\rho_\infty + 1}$	$\frac{1}{2} - \alpha_m$	$\frac{(1 - \alpha_m)^2}{4}$
Generalized- α	$\frac{\rho_\infty}{\rho_\infty + 1}$	$\frac{2\rho_\infty - 1}{\rho_\infty + 1}$	$\frac{1}{2} + \alpha_f - \alpha_m$	$\frac{(1 + \alpha_f - \alpha_m)^2}{4}$

5. Results

For convenience abbreviations will be used for each algorithm with parameters used according to Table 2.

Table 2: Abbreviations for algorithms and control parameters

Algorithm	Abbreviation
Central difference	CD
Houbolt	HB
Wilson θ	$WS(\theta)$
Newmark	$N(\gamma)(\beta)$
Newmark*	$N(\gamma)(\beta)[\rho_\infty]$
HHT- α	$H(\gamma)(\beta)(\alpha_f)$
HHT- α^*	$H(\gamma)(\beta)(\alpha_f)[\rho_\infty]$
WBZ- α	$W(\gamma)(\beta)(\alpha_m)$
WBZ- α^*	$W(\gamma)(\beta)(\alpha_m)[\rho_\infty]$
Generalized- α	$G(\gamma)(\beta)(\alpha_f)(\alpha_m)$
Generalized- α^*	$G(\gamma)(\beta)(\alpha_f)(\alpha_m)[\rho_\infty]$
Bathe	$B(\gamma)(\beta)(\alpha_s)$

* indicates that control parameters are calculated as a function of ρ_∞ .

First, we will introduce the spectral radius curves for different integration algorithms. As a verification, some of spectral radius curves found in [2], [21] and [23] were drawn using results computed by the Fortran code. On comparing results using curves in Fig. 1 to Fig. 5 it was found to be identical with these shown by references mentioned later. After verification has been done, many comparisons were made between numerically dissipative algorithms at different values of ρ_∞ .

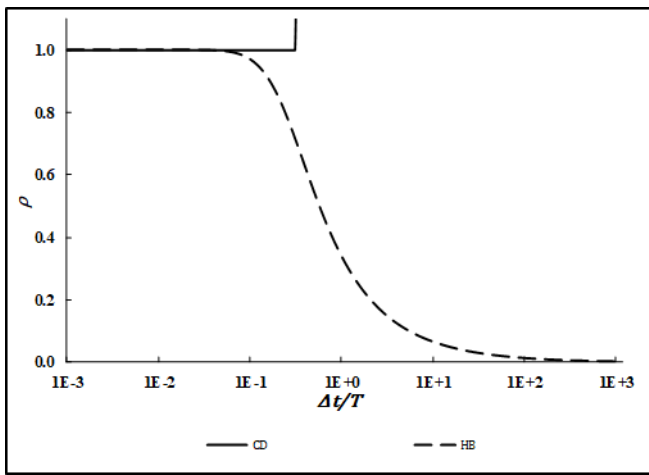


Fig. 1: Spectral Radii of Central Difference and Houbolt algorithms.

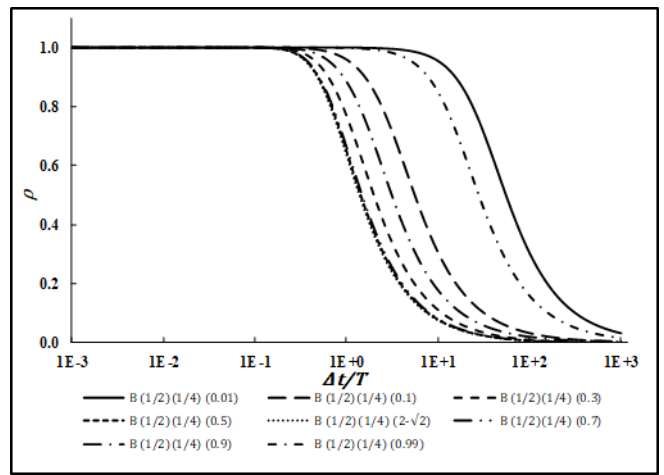


Fig. 4: Spectral Radii of Bathe Method with $\gamma = 1/2, \beta = 1/4$ and $0.0 < \alpha_s < 1.0$.

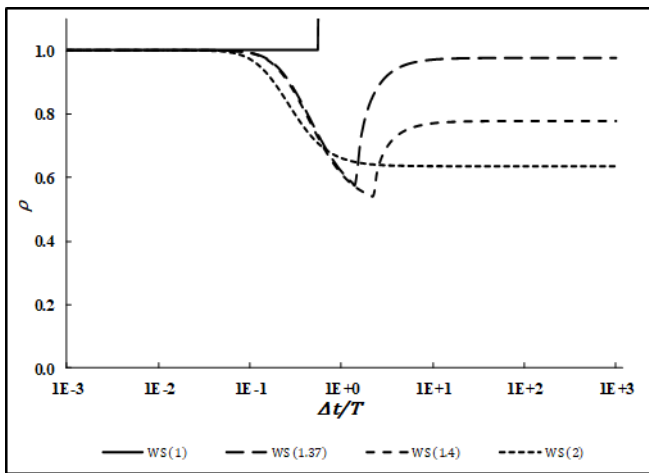


Fig. 2: Spectral Radii of Wilson θ algorithm for different values of control parameter θ .

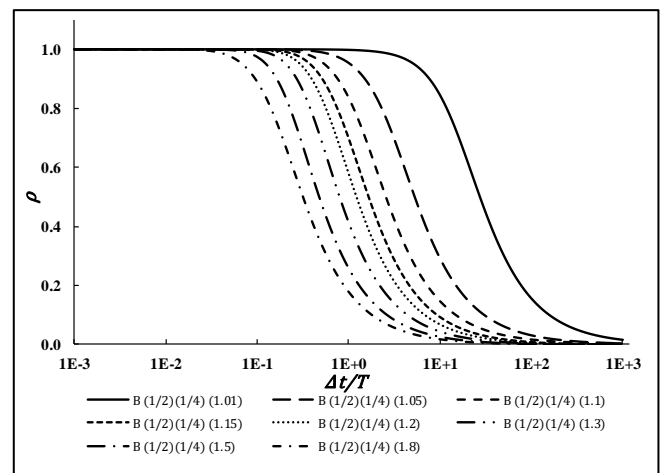


Fig. 5: Spectral Radii of Bathe Method with $\gamma = 1/2, \beta = 1/4$ and $\alpha_s > 1.0$.

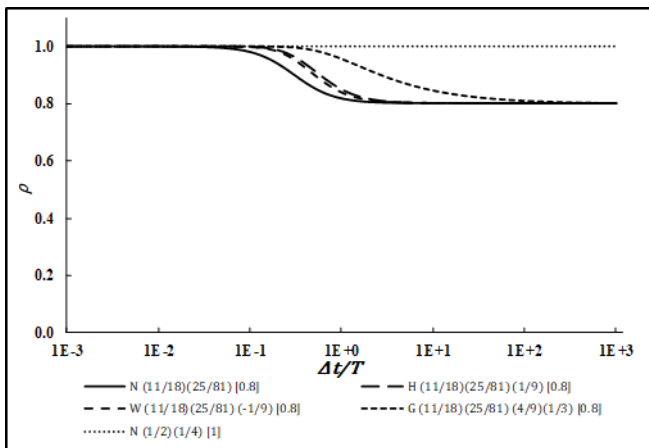


Fig. 3: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at different values of ρ_∞ .

The comparison between dissipative algorithms with the same ρ_∞ can indicate the difference between these algorithms specially for mean frequency modes. Almost all algorithms have low dissipation at low frequency modes, then the differences between algorithms elaborate clearly when dealing with higher frequency modes.

It can be seen from Fig. 1 and Fig. 2 that the CD algorithm has a conditional stability as it violates the condition of stability for values of $\Delta t/T$ higher than 0.318. A similar behavior is shown by WS(1.0) which is equivalent to $N(1/2)(1/6)$ as they are only stable for the range $(0 < \Delta t/T \leq 0.551)$. Although both the explicit (CD) and implicit (WS(1.0) and $N(1/2)(1/6)$) have a conditional stability, the difference between their stability limits gives a hint to the stability edge owned by the implicit algorithms as they provide a wider range of stability.

HHT- α algorithm did not maintain its unconditional stability on using equations in Table 1 for values of $\rho_\infty < 0.4$ as seen in Fig. 9 and Fig. 10. Actually, with numerical trials it was found that HHT- α algorithm is unconditionally stable using equations in Table 1 only with $1/3 \leq \rho_\infty \leq 1$.

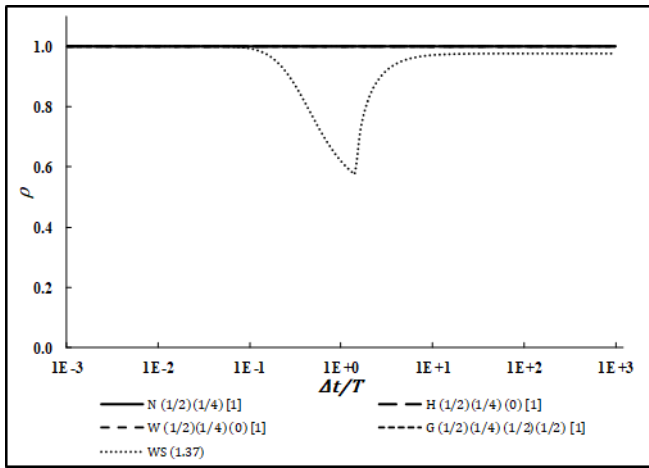


Fig. 6: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 1$) with Wilson θ at ($\theta = 1.37$).

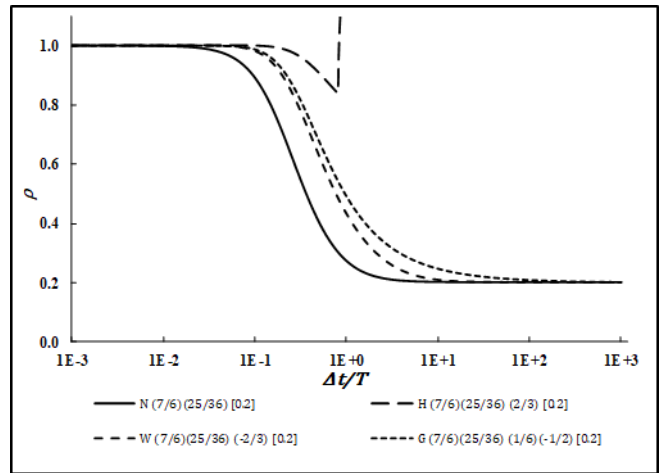


Fig. 9: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.2$).

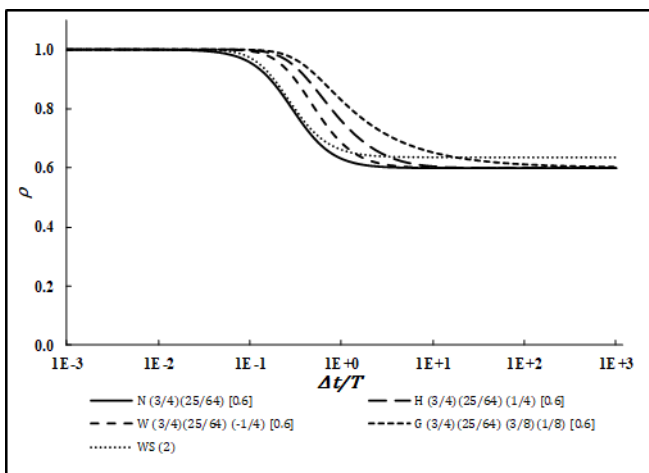


Fig. 7: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.6$) with Wilson θ at ($\theta = 2.0$).

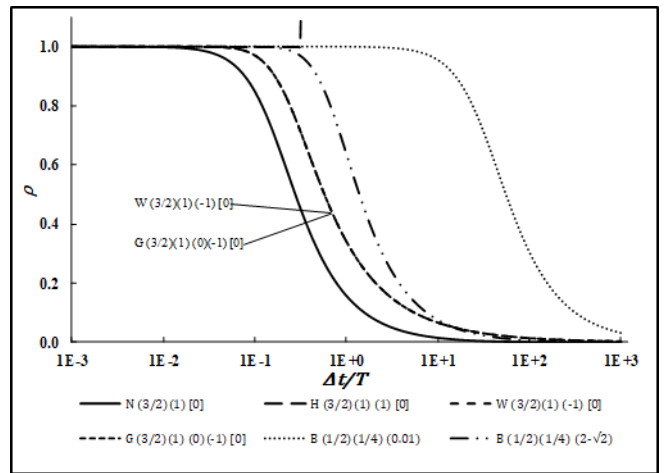


Fig. 10: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.0$) with Bathe at ($\gamma = 1/2, \beta = 1/4, \alpha_s = 0.01, 2 - \sqrt{2}$).

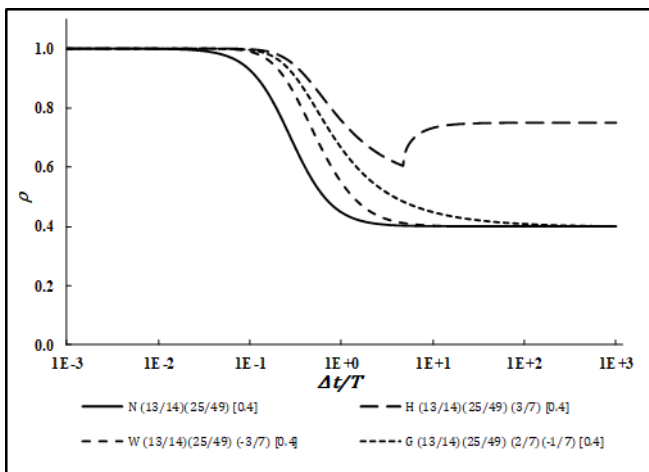


Fig. 8: Spectral Radii of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.4$).

Some unconditionally stable cases show a cusp as the spectral radius curve bifurcates from its smooth bath like in $WS(1.37)$, $WS(1.4)$ and $H(13/14)(25/49)(3/7)[0.4]$.

That bifurcation means that the analyst cannot predict a relation on which the algorithm damps out the modes as their frequency increase. It was found by numerical trials that spectral radius curve for Wilson θ is almost smooth with $\theta > 1.42$ while for HHT- α the condition for smooth spectral radius curve is $1/2 \leq \rho_\infty \leq 1$ given that ρ_∞ is used as a control parameter (i.e. Table 1 is implemented).

For a better judgement on algorithms` performance, PE and AD are calculated and drawn in charts similar to that used for spectral radius. Only a minor change will be done on the horizontal axis as it will have a smaller range in favor of results` clarity.

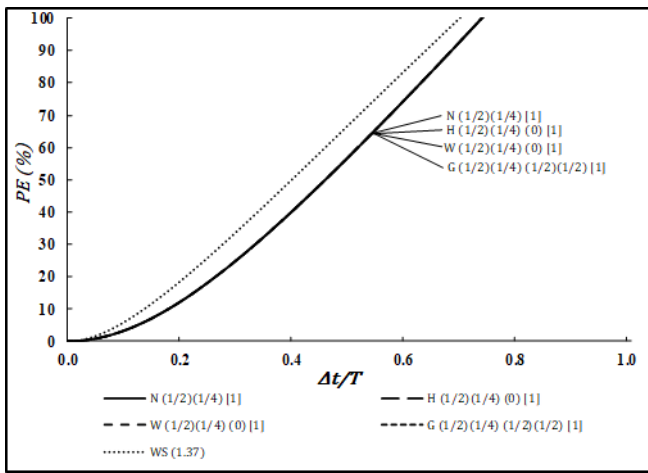


Fig. 11: Period elongation percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 1.0$) with Wilson θ at ($\theta = 1.37$).

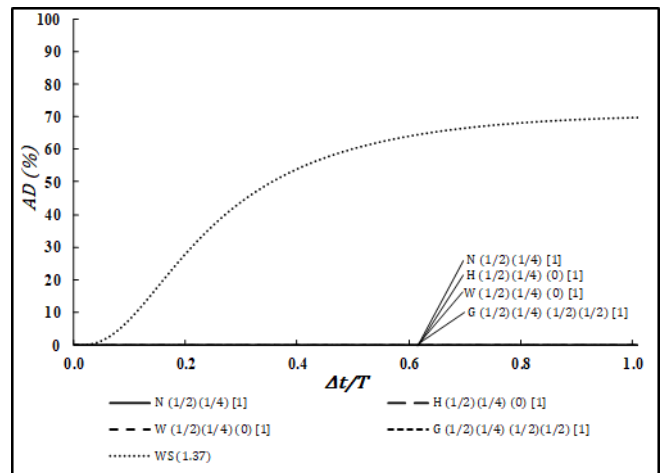


Fig. 14: Amplitude decay percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 1.0$) with Wilson θ at ($\theta = 1.37$).

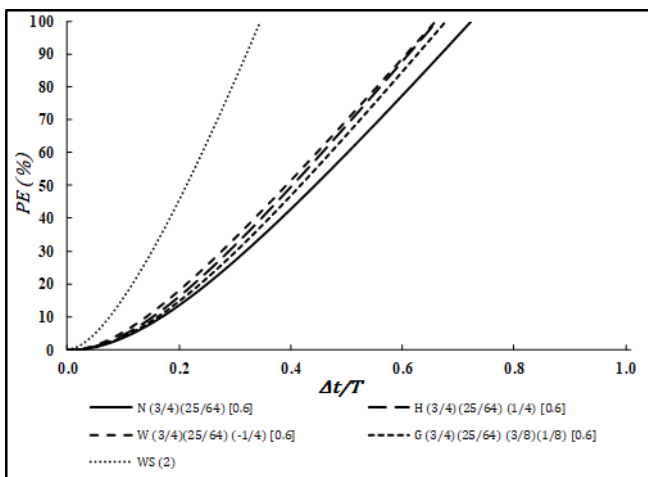


Fig. 12: Period elongation percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.6$) with Wilson θ at ($\theta = 2$).

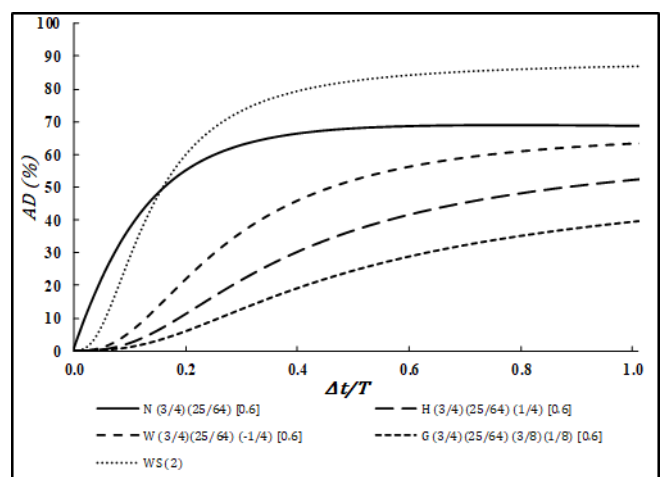


Fig. 15: Amplitude decay percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.6$) with Wilson θ at ($\theta = 2$).

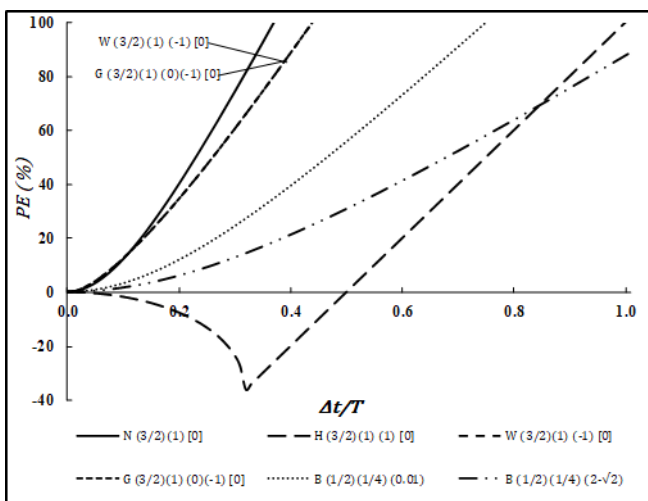


Fig. 13: Period elongation percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.0$) with Bathe at ($\gamma = 1/2, \beta = 1/4, \alpha_s = 0.01, 2 - \sqrt{2}$).

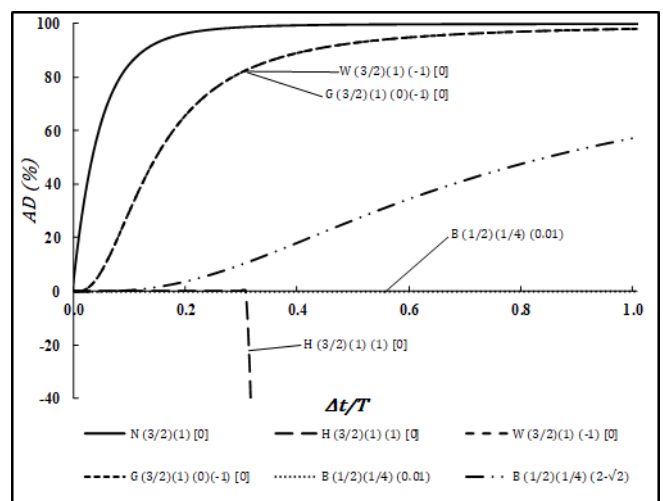


Fig. 16: Amplitude decay percentage of Newmark, HHT- α , WBZ- α and Generalized- α algorithms at ($\rho_\infty = 0.0$) with Bathe at ($\gamma = 1/2, \beta = 1/4, \alpha_s = 0.01, 2 - \sqrt{2}$).

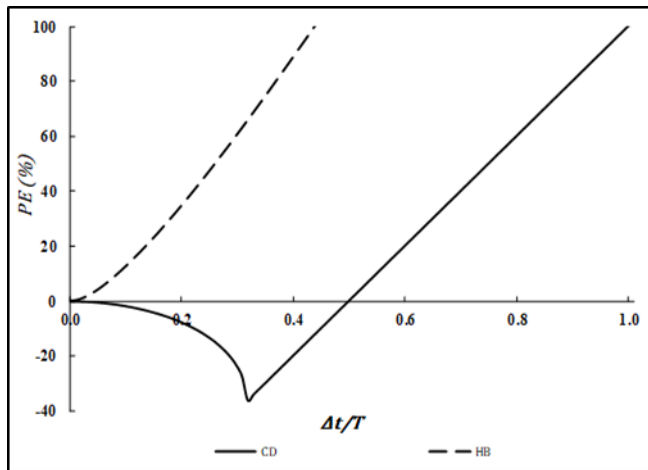


Fig. 17: Period elongation percentage of central difference and Houbolt algorithms.

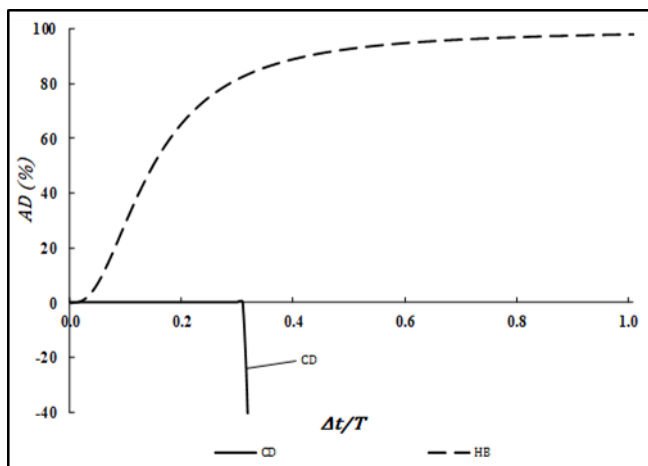


Fig. 18: Amplitude decay percentage of central difference and Houbolt algorithms.

On using ρ_∞ as a control parameter, it was found that more than one algorithm reduces to be equivalent to another algorithm. This feature is inherent in the generalized- α method as on setting $\alpha_m = 0$ the algorithm reduces to be HHT- α while setting $\alpha_f = 0$ reduces the generalized- α to WBZ- α . Consequently, setting both $\alpha_m = 0$ and $\alpha_f = 0$ reduces generalized- α method to the standard Newmark family. For example, $H(1/2)(1/4)(0)[1]$ and $W(1/2)(1/4)(0)[1]$ are exactly the same as $N(1/2)(1/4)[1]$ algorithm which is known as the trapezoidal method.

The spectral performance of the unconditionally stable algorithms with $\rho_\infty = 1$ can be found in Fig. 6, Fig. 11 and Fig. 14. These algorithms showed that they have the same period elongation percentage as well as zero amplitude decay, which is a very important aspect for the analyst who wants to keep all modes in the response without damping out the higher ones. On using Bathe method $B(1/2)(1/4)(0.01)$ it approximately has the same spectral performance as that shown by the trapezoidal method and $G(1/2)(1/4)(1/2)(1/2)[1]$.

In case of using $\rho_\infty = 0$ for Newmark, HHT- α , WBZ- α and generalized- α algorithms and comparing them with $B(1/2)(1/4)(0.01)$ and $B(1/2)(1/4)(2 - \sqrt{2})$ as shown in Fig. 10, Fig. 13 and Fig. 16 it can be seen that $B(1/2)(1/4)(0.01)$ has almost no amplitude decay for low frequency modes and an acceptable period elongation behavior among the other algorithms introduced in that context. On applying $H(3/2)(1)(1)[0]$ on a structural system with Δt that keeps on the condition ($0 < \Delta t/T \leq 0.31$) for all modes which is the stability condition, the algorithm has no amplitude decay but a recognizable period elongation compared to $B(1/2)(1/4)(0.01)$.

On using $\rho_\infty = 0.6$ for algorithms shown in Fig. 7, Fig. 12 and Fig. 15 it is found that the generalized- α algorithm $G(3/4)(25/64)(3/8)(1/8)[0.6]$ has the least amplitude decay among other methods with in the comparison, although it has a slightly higher period elongation than that of $N(3/4)(25/64)[0.6]$.

The central difference method on the spectral performance basis shown in Fig. 1, Fig. 17 and Fig. 18 has no special edge as there are algorithms such as $N(1/2)(1/4)[1]$ which has no algorithmic damping and moreover is unconditionally stable. A new interesting finding was that the central difference's spectral performance is nearly the same as that of $H(3/2)(1)(1)[0]$. A similar behavior is found between the Houbolt method and $W(3/2)(1)(-1)[0]$.

6. Conclusions

Stability and accuracy analysis of direct integration algorithms based on their spectral characteristics has shown many important aspects that any analyst would be interested in before using the algorithm. Some of the study results can be summarized in the following paragraphs.

The newly proposed bathe method showed promising spectral properties compared to the historically dominating generalized- α . More combinations of control parameters used in Bathe and generalized- α methods should be studied. Also, it should be noted that Bathe method approximately requires twice the computational effort required on applying generalized- α method which reveals that a computational performance studies should be carried on clarifying this issue.

Houbolt and central difference methods showed equivalent spectral properties to those of $W(3/2)(1)(-1)[0]$ and $H(3/2)(1)(1)[0]$ respectively. Numerical examples should be carried on finding out if there is a similar equality in responses calculated using these methods or not.

Acknowledgment

The authors wish to thank Mansoura University, Information Technology Institute in Cairo and the Egyptian Supreme Council of Universities for their technical and logistic support.

References

- [1] M. A. Dokainish and K. Subbaraj, "A survey of direct time-integration methods in computational structural dynamics—I. Explicit methods," *Comput. Struct.*, vol. 32, no. 6, pp. 1371–1386, Jan. 1989.
- [2] K. Bathe, *Finite Element Procedures Second Edition* Bathe Finite Element Procedures Klaus-Jürgen Bathe Second Edition. 2006.
- [3] G. Noh and K. J. Bathe, "An explicit time integration scheme for the analysis of wave propagations," *Comput. Struct.*, vol. 129, pp. 178–193, 2013.
- [4] M. Rezaiee-Pajand and M. Karimi-Rad, "An accurate predictor-corrector time integration method for structural dynamics," *Int. J. Steel Struct.*, vol. 17, no. 3, pp. 1033–1047, 2017.
- [5] X. D. LI and N.-E. WIBERG, "STRUCTURAL DYNAMIC ANALYSIS BY A TIME-DISCONTINUOUS GALERKIN FINITE ELEMENT METHOD," *Int. J. Numer. Methods Eng.*, vol. 39, no. 12, pp. 2131–2152, Jun. 1996.
- [6] K. J. Bathe and E. L. Wilson, "Stability and accuracy analysis of direct integration methods," *Earthq. Eng. Struct. Dyn.*, vol. 1, no. 3, pp. 283–291, 1972.
- [7] K. Subbaraj and M. A. Dokainish, "A survey of direct time-integration methods in computational structural dynamics—II. Implicit methods," *Comput. Struct.*, vol. 32, no. 6, pp. 1387–1401, Jan. 1989.
- [8] T. C. Fung, "Numerical dissipation in time-step integration algorithms for structural dynamic analysis," *Prog. Struct. Eng. Mater.*, vol. 5, no. 3, pp. 167–180, Jul. 2003.
- [9] H. M. Hilber and T. J. R. Hughes, "Collocation, dissipation and [overshoot] for time integration schemes in structural dynamics," *Earthq. Eng. Struct. Dyn.*, vol. 6, no. 1, pp. 99–117, Jan. 1978.
- [10] C. Chen and J. M. Ricles, "Stability Analysis of Direct Integration Algorithms Applied to MDOF Nonlinear Structural Dynamics," *J. Eng. Mech.*, vol. 136, no. 4, pp. 485–495, Apr. 2010.
- [11] J. M. Benítez and F. J. Montáns, "The value of numerical amplification matrices in time integration methods," *Comput. Struct.*, vol. 128, pp. 243–250, Nov. 2013.
- [12] X. Liang and K. M. Mosalam, "Lyapunov Stability and Accuracy of Direct Integration Algorithms Applied to Nonlinear Dynamic Problems," *J. Eng. Mech.*, vol. 142, no. 5, p. 04016022, May 2016.
- [13] J. Zhang, Y. Liu, and D. Liu, "Accuracy of a composite implicit time integration scheme for structural dynamics," *Int. J. Numer. Methods Eng.*, vol. 109, no. 3, pp. 368–406, Jan. 2017.
- [14] K. C. Park, "Practical aspects of numerical time integration," *Comput. Struct.*, vol. 7, no. 3, pp. 343–353, Jun. 1977.
- [15] D. Roy and G. V. Rao, *Elements of Structural Dynamics*. Chichester, UK: John Wiley & Sons, Ltd, 2012.
- [16] J.C. Houbolt, "A Recurrence Matrix Solution for the Dynamic Response of Elastic Aircraft," *J. Aeronaut. Sci.*, vol. 17, no. 9, pp. 540–550, Sep. 1950.
- [17] N. M. Newmark, "A Method of Computation for Structural Dynamics," *J. Eng. Mech. Div.*, vol. 85, no. 3, pp. 67–94, 1959.
- [18] E. L. Wilson, I. Farhoomand, and K. J. Bathe, "Nonlinear dynamic analysis of complex structures," *Earthq. Eng. Struct. Dyn.*, vol. 1, no. 3, pp. 241–252, 1972.
- [19] H. M. Hilber, T. J. R. Hughes, and R. L. Taylor, "Improved numerical dissipation for time integration algorithms in structural dynamics," *Earthq. Eng. Struct. Dyn.*, vol. 5, no. 3, pp. 283–292, Jul. 1977.
- [20] W. L. Wood, M. Bossak, and O. C. Zienkiewicz, "An alpha modification of Newmark's method," *Int. J. Numer. Methods Eng.*, vol. 15, no. 10, pp. 1562–1566, Oct. 1980.
- [21] J. Chung and G. M. Hulbert, "A Time Integration Algorithm for Structural Dynamics With Improved Numerical Dissipation: The Generalized- α Method," *J. Appl. Mech.*, vol. 60, no. 2, p. 371, 1993.
- [22] K. J. Bathe and M. M. I. Baig, "On a composite implicit time integration procedure for nonlinear dynamics," *Comput. Struct.*, vol. 83, no. 31–32, pp. 2513–2524, 2005.
- [23] G. Noh and K.-J. Bathe, "Further insights into an implicit time integration scheme for structural dynamics," *Comput. Struct.*, vol. 202, pp. 15–24, Jun. 2018.
- [24] T. C. Fung, "Complex-time-step newmark methods with controllable numerical dissipation," *Int. J. Numer. Methods Eng.*, vol. 41, no. 1, pp. 65–93, 1998.
- [25] D. Kuhl and M. A. Crisfield, "Energy-conserving and decaying Algorithms in non-linear structural dynamics," *Int. J. Numer. Methods Eng.*, vol. 45, no. 5, pp. 569–599, Jun. 1999.