# SECOND ORDER FINITE ELEMENT SCHEMES FOR ELASTODYNAMICS

**GONG JIAQI**

*(M.Sc., National University of Singapore)*

**A THESIS SUBMITTED**

**FOR THE DEGREE OF MASTER OF SCIENCE**

**DEPARTMENT OF MATHEMATICS**

**NATIONAL UNIVERSITY OF SINGAPORE**

**2016**

*To my mother, father and*

*To my husband*

# Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety.
I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

GONG JIAQI

Gong Jiaqi

18 Jan 2016

# Acknowledgements

First and foremost, I owe my deepest gratitude to my supervisor Assistant Professor LIU Jie. I appreciate all his contributions of time and ideas to make my M.Sc experience productive and meaningful. His constant encouragement and enlightening guidance helped me in all the time of research and writing of this thesis. His research mentality and hardworking spirit has inspired me throughout my graduate studies and will be an invaluable part of my future career.

**GONG Jiaqi**

**January 2016**

# Contents

# Summary

This thesis is concerned with second order time integration schemes for linear elastodynamics equations. In addition to those existing algorithms, we proposed two new schemes derived from the traditional DBF2 method. We analysed accuracy and stability properties of both schemes, proved that they are both second-order accurate. Moreover, we theoretically compared these two schemes with the Newmark Algorithm which is one of the most popular methods among engineers. At last, with the help of Finite Element Method, we did accuracy check to constructed elastodynamics problems with known analytic solutions, and applied them to practical problems.

# List of Tables

# List of Figures

Chapter 1

# Introduction

Linear elastodynamics is a simplification of the more general nonlinear theory of elastodynamics which is a branch of continuum mechanics. It is widely used in structural analysis and engineering design, often with the aid of Finite Element Method (FEM).

The mathematical formulation of elastodynamics problem is based on the Newton's second law, by which we can derive the equation of motion as

$$\nabla \cdot \sigma + G = \rho \ddot{u},$$

where $\sigma$ is the Cauchy stress tensor, $G$ is the body force per unit volume, $\rho$ is the mass density and $u(z,t) = \varphi(z,t) - z$ is the displacement vector.

Then, this system of differential equations is completed by a set of constitutive relations. For linear elastic material, we can choose

$$\sigma = \mu(\nabla u + \nabla u^T) + \lambda(\nabla \cdot u)I,$$

where $\mu$ and $\lambda$ are lame constants; for St. Venant-Kirchhoff material,

$$\sigma = 2\mu FE + \lambda(trE)F,$$

where $F = \nabla\varphi$ is the deformation gradient and $E = \frac{1}{2}(F^T F - I)$.

For nonlinear elastic material, $\sigma$ can be linearized by various methods, for example, by

$$\langle\sigma(\varphi^n), \nabla\phi\rangle = \langle\sigma(\varphi^{n-1}), \nabla\phi\rangle + A(\nabla\varphi^{n-1}; \nabla(\varphi^n - \varphi^{n-1}), \nabla\phi) + O((\varphi^n - \varphi^{n-1})^2),$$

where $A(\varphi; \nabla\psi, \nabla\phi) = \frac{d}{d\epsilon} \mid_{\epsilon=0} \langle\sigma(\varphi + \epsilon\psi), \nabla\phi\rangle$ and $\varphi^n$ is the numerical solution of motion $\varphi$ at $nth$ time stepping.

There are several ways that can lead us from elastodynamics equation $\nabla\cdot\sigma + G = \rho\ddot{u}$ to an ordinary differential equation in time, as

$$M\ddot{U} + C\dot{U} + KU = F, \tag{1.1}$$

where $M, C, K$ are, correspondingly, the mass, damping and stiffness matrices, $U$ is the displacement vector and $F$ is the vector of nodal load.

Commonly used space discretization methods to get equation (1.1) are the finite element method [Hughes and Hulbert (1988)], the boundary element method [Mack (1991), Dominguez (1993)], the spectral element method [Komatitsch and Vilotte (1998)] and the Smoothed Particle Hydrodynamics (SPH) method [Liu and Liu (2010)], etc.. Besides space discretization, for some specific equation like linear elastic material, where we choose $\sigma = \mu(\nabla u + \nabla u^T) + \lambda(\nabla \cdot u)I$. The ordinary differential equation (1.1)can also be obtained by using Fourier transformation, because of the property $f(\nabla g)(k) = 2\pi ikf(g)(k)$, where $f(g)$ represents the Fourier transformation of function $g$.

Due to the importance in solving practical engineering problem, and because it is the foundation for a successful and effective method for nonlinear problem, numerical methods for the time integration of linear elastodynamics is one of the most developed field in computational mechanics. The algorithms for solving linear elastodynamics equations can be classified into three main classes: (a) algorithms

based on finite difference time discretizations of the equation of motion; (b) algorithm derived applying the weighted residual method to the equation of motion; (c) algorithms stemming from a weak variational formulation.

Since when using standard finite elements to discretize the spatial domain, the spatial resolution of high-frequency modes typically is poor, thus, it would be desirable for time integration algorithms to possess controllable numerical dissipation in the higher frequency modes. Also, high-frequency numerical dissipation has been found to improve the convergence of iterative equation solvers when it is applied to solve highly nonlinear problems. Numerous second-order accurate dissipative algorithms have been developed that obtain high-frequency dissipation. For example, there are Newmark method [Chung and Hulbert (1993)], the Wilson-$\theta$ method [Wilson (1968)], the HHT-$\alpha$ method [Hilber, Hughes, and Taylor (1977)], the WBZ-$\alpha$ method [Wood, Bossak, and Zienkiewicz (1980)], the $\rho$ method [Bazzi and Anderheggen (1982)], the $\theta_1$-method [Hoff and Pahl (1988a,b)] and the Generalized-$\alpha$ method [Chung and Hulbert (1993)]. Due to the space discretization, the exact solution to equation (1.1) contains the numerical dispersion error, (see, for example, [Dauksher and Emery (2000), Guddati and Yue (2004), Marfurt (1984), Idesman and Pham (2014)]). To decrease the error, there are studies investigating the use of mesh refinement as well as the development of other special techniques, (see, for example [Babuska, Ihlenburg, Strouboulis, and Gangaraj (1997), Babuška, Strouboulis, Gangaraj, and Upadhyay (1997)]).

We would like to take Newmark method and HHT-$\alpha$ method as examples and give a general introduction.

The finite difference approximations for the Newmark method applied to equation (1.1) are

$$\begin{cases} u^{n+1} = u^n + \Delta t \dot{u}^n + \frac{1}{2}(1-2\beta)\Delta t^2 \ddot{u}^n + \beta \Delta t^2 \ddot{u}^{n+1} \\ \dot{u}^{n+1} = \dot{u}^n + (1-\gamma)\Delta t \ddot{u}^n + \gamma \Delta t \ddot{u}^{n+1} \\ M\ddot{u}^{n+1} + C\dot{u}^{n+1} + Ku^{n+1} = F^{n+1} \end{cases}$$

where $u^n$ is the numerical solution of displacement at $nth$ time stepping.

For $\gamma = \frac{1}{2}$ the Newmark method is at least second-order accurate; while it is first order accurate for all other values of $\gamma$. Some particular values can be chosen for $\gamma$ and $\beta$. For example, if $\gamma = \frac{1}{2}$, $\beta = \frac{1}{6}$, this leads to the linear interpolation of $\ddot{u}$ in time interval $[t_n, t_{n+1}]$,

$$\ddot{u}(\tau) = \ddot{u}_n + (\tau - t_n)(\frac{\ddot{u}_{n+1} - \ddot{u}_n}{\Delta t}) \qquad \forall \tau \in [t_n, t_{n+1}],$$

The concise form of Newmakr method to equation (1.1) is as the following:

$$(M+\gamma\Delta tC+\beta\Delta t^2K)\ddot{u}_{n+1} = F_n - C(\dot{u}_n+(1-\gamma)\ddot{u}_n) - K(u_n+\Delta t\dot{u}_n+(\frac{1}{2}-\beta)\Delta t^2\ddot{u}_n),$$

where $F_n$ is the value of $F$ at time $t_n$.

The HHT-$\alpha$ method is a generalization of the Newmark method and reduces to the Newmark if its parameter $\alpha = 0$. The HHT-$\alpha$ adopts the finite difference equations of the Newmark method, but modifies the equations of motion, using a parameter $\alpha$.

For the motion equation has the form like equation (1.1), then, in HHT-$\alpha$ the formula is substituted as

$$\begin{cases} u^{n+1} = u^n + \Delta t\dot{u}^n + \frac{1}{2}(1-2\beta)\Delta t^2\ddot{u}^n + \beta\Delta t^2\ddot{u}^{n+1} \\ \dot{u}^{n+1} = \dot{u}^n + (1-\gamma)\Delta t\ddot{u}^n + \gamma\Delta t\ddot{u}^{n+1} \\ M\ddot{u}^{n+1} + (1+\alpha)C\dot{u}^{n+1} - \alpha C\dot{u}^n + (1+\alpha)Ku^{n+1} - \alpha Ku^n = F^{n+1} \end{cases}$$

The HHT-$\alpha$ method is at least second-order accurate and unconditionally stable. It is widely applied to structural dynamics simulations incorporating many degrees of freedom and in which it is desirable to numerically attenuate the response at high frequencies. Which is a improvement compared with Newmark Algorithm. To achieve unconditional stability, second order accuracy and favorable numerical dissipation for linear elastic systems, the relationship between three parameters

below has been suggested:

$$-\frac{1}{3} \leq \alpha \leq 0, \qquad \beta = \frac{(1-\alpha)^2}{4}, \qquad \gamma = \frac{1}{2} - \alpha.$$

While, as for computationally effective high-order accurate time integration method for elastodynamics equation, it is still a challenging field in computational mechanics which requires further extensive research.

In this paper, we discuss two second-order accurate finite element schemes for linear elastodyanmcis equation, one of which was firstly proposed in [Liu (2015)] as the numerical algorithm for the deformable structure in a fluid-structure interaction (FSI) system. The objective of this paper is to present the accuracy as well as stability study of these two methods and compare them with Newmark Algorithm which is currently one of the most popular schemes for linear elastodynamics equation. Then, we run the accuracy check by firstly applying them to an ODE problem, then, employ them to solve constructed linear elastodynamics problems with both known and unknown exact analytic solutions. From their respective numerical performance, we conclude that the first and the second schemes are both second-order accurate, and have their certain advantages over Newmark method. For example, no need to choose any parameters makes our schemes more convenient and appealing during application.

The rest of this paper is organized as follow. We are going to discuss the spatial discretization process as well as the scalar form of equation (1.1) in following two sections of Chapter 1. We introduce the first method and analyse its accuracy and stability properties in Chapter 2 and discuss the second scheme in Chapter 3. We then compare them with Newmark Algorithm theoretically in Chapter 4. Further numerical comparison and temporal accuracy check are presented in Chapter 5.

## 1.1 Spatial discretization

In this section, we would like to briefly introduce the Finite Element Method and how it can be applied to get equation (1.1).

The finite element method was first conceived in a paper by Courant [Courant et al. (1943)], but the importance of this contribution was ignored at that time. Then the engineers independently re-invented the method in the early fifties: The earliest references generally quoted in the engineering literature are working of Argyris from 1954 to 1955 and was reprinted in 1960 [Argyris and Kelsey (1960)] and working of Turner, Clough, Martin, and Topp (1956). The name of the method was proposed by Clough (1960). Historical accounts on the development of the method, from the engineering point of view, are given in Oden (1972) and Zienkiewicz (1973). Actually, since 1967, many books have been published on the Finite Element Method, for more details, one can refer to, for example, Zienkiewicz and Taylor (2000), Norrie and De Vries (2014), Strang and Fix (1973), Alavala (2008), Brenner and Scott (2007).

Now, we will show how finite element method can lead to the ordinary different equation (1.1) below. Recall a general linear elastodynamics problem on a domain $\Omega$,

$$
\begin{aligned}
&\nabla \cdot \sigma + G = \rho \ddot{u} \qquad in\, \Omega \times I,\\
&\sigma = \mu(\nabla u + \nabla u^T) + \lambda(\nabla \cdot u)I,\\
&u(x,0) = u^0(x) \qquad in\, \Omega,\\
&v(x,0) = v^0(x) \qquad in\, \Omega,\\
&u = g(x,t) \qquad on\, \Gamma_1 \times I,\\
&(n \cdot \sigma) = T(x,t) \qquad on\, \Gamma_2 \times I,
\end{aligned}
$$

where $n$ is the outward normal vector, $(n\cdot\sigma)_i = \sum_j \sigma_{ij} n_j$, $I = (0, T_{end})$ and $\Gamma_1 \cup \Gamma_2 =$

$\partial\Omega$, $\Gamma_1 \cap \Gamma_2 = \varnothing$. Without loss of generality, we can simply assume $\rho = 1$.

Let $V = H^1(\Omega)$, we know $u$ satisfies the following weak formulation of equation

$$(\ddot{u}, v) = -(\sigma, \nabla v) + ((n \cdot \sigma), v)_{\Gamma_1} + (T, v)_{\Gamma_2} + (G, v) \qquad \forall v \in V, t \in I,$$

where $(u, v) = \int_\Omega u(x)v(x)dx$, $(\nabla u, \nabla v) = \int_\Omega \sum_i u_{x_i} v_{x_i} dx$ and $(u, v)_{\partial\Omega} = \int_{\partial\Omega} u(x)v(x)dx$.

Consider the mesh $T_h = \{K_1, \ldots, K_m\}$ which forms a partition of $\Omega$. It is required that no vertex of any $K_i$ lies on the interior of edge of another $K_j$, where $i \neq j$. Based on this partition, we can have a finite dimensional space,

$$V_h = \{v_h \in V : v_h \in \mathbb{C}^0(\bar{\Omega}) \, , v_h|_{K_i} \in \mathbb{P}^n(K_i), \forall K_i \subset T_h\}$$

where $\mathbb{P}^n$ stands for order $n$ polynomial spaces. Define, the subspace of $V_h$ as:

$$V_{h0} = \{v_h : v_h \in V_h, v_h|_{\Gamma_1} = 0\},$$

$$V_{hg} = \{v_h : v_h \in V_h, v_h|_{\Gamma_1} = g\},$$

We can restate the original problem as:

Find $u_h \in V_{hg}$, such that the following equation holds,

$$(T_h, v_h)_{\Gamma_2} - (\sigma(u_h), \nabla v_h) + (G_h, v_h) = (\ddot{u}_h, v_h) \qquad \forall v_h \in V_{h0}, t \in I,$$

where $T_h$ and $G_h$ are the corresponding interpolation of $T$ and $G$ in the space $V_{h0}$.

Assume the basis function for $V_{h0}$ is $\{\phi_i(x)\}$ and for $V_{hg}$ is $\{\varphi_i(x)\}$. For $u_h$, we can have the representation

$$u_h = \sum \xi_i(t)\varphi_i(x),$$

where $\xi_i(t)$ is the time dependent coefficient.

Then, let $v_h = \phi_j$, the equation becomes:

$$(T_h, \phi_j)_{\Gamma_2} - (\sigma(u_h), \nabla\phi_j) + (G_h, \phi_j) = (\ddot{u}_h, \phi_j) \qquad \forall v_h \in V_{h0}, t \in I. \tag{1.2}$$

Since

$$\begin{aligned}\sigma(u_h) &= \mu(\nabla u_h + \nabla u_h^T) + \lambda(\nabla \cdot u_h)I \\ &= \mu(\sum \xi_i(\nabla\varphi_i + \nabla\varphi_i^T)) + \lambda(\sum \xi_i(\nabla \cdot \varphi_i))I,\end{aligned}$$

by substituting above representation into equation (1.2), we get

$$\begin{aligned}\sum(\varphi_i(x), \phi_j)\ddot{\xi}_i(t) = &-\sum((\mu(\nabla\varphi_i + \nabla\varphi_i^T) + \lambda(\nabla \cdot \varphi_i I)), \nabla\phi_j)\xi_i \\ &+ (T_h, \phi_j)_{\Gamma_2} + (G_h, \phi_j).\end{aligned} \tag{1.3}$$

Assume $T = 0$ and $g(x,t) = 0$, thus $V_{h0} = V_{hg}$. If we denote the basis function as $\{\varphi_i\}_{i=1}^N$. Hence, the matrix form of equation (1.3) can be simplified as:

$$M\ddot{\xi}(t) + K\xi(t) = F, \tag{1.4}$$

where

$$\xi(t) = (\xi_1, \xi_2, \cdots, \xi_N)^T,$$

$$M_{i,j} = (\varphi_i, \varphi_j),$$

$$K_{i,j} = (\nabla\varphi_i, \mu(\nabla\varphi_j + \nabla\varphi_j^T) + \lambda(\nabla \cdot \varphi_j)I),$$

$$F = ((G_h, \varphi_1), (G_h, \varphi_2), \cdots, (G_h, \varphi_N))^T,$$

We can tell that in this problem, the matrix $C$ in equation (1.1) actually equals zero.

## 1.2 Single-degree-of-freedom problem

For the general form of elastodynamics problem as equation (1.1), it suffices to consider $F(t) = 0$ for all $t \geq 0$ in order to study the accuracy and stability properties of an algorithm. Thus, we will discuss the corresponding single-degree-of-freedom problem of elastodynamics equation

$$M\ddot{U} + C\dot{U} + KU = 0.$$

**Theorem 1.1.** *If we assume the matrix form of a elastodynamics problem is* $M\ddot{U} + C\dot{U} + KU = 0$. *Coefficient matrices* $M, C$ *and* $K$ *can be diagonalizable by a common invertible matrix, then the matrix form can be rewritten in scalar form.*

*Proof.* Assume $M, C$ and $K$ can be diagonalizable by a common invertible matrix $P$ such that

$$M = P\tilde{M}P^T, C = P\tilde{C}P^T, K = P\tilde{K}P^T,$$

where $\tilde{M}, \tilde{C}$ and $\tilde{K}$ are the respective diagonal matrices.

Thus equation(2.1) can be written as

$$P\tilde{M}P^T\ddot{U} + P\tilde{C}P^T\dot{U} + P\tilde{K}P^TU = 0,$$

Let $P^TU = \tilde{U}$, then, the above equation is given by

$$P\tilde{M}\ddot{\tilde{U}} + P\tilde{C}\dot{\tilde{U}} + P\tilde{K}\tilde{U} = 0,$$

which implies

$$\tilde{M}\ddot{\tilde{U}} + \tilde{C}\dot{\tilde{U}} + \tilde{K}\tilde{U} = 0,$$

Note that $\tilde{M}, \tilde{C}$ and $\tilde{K}$ are diagonal matrices. Hence, we have successfully reduced the matrix equation (2.1)to a much simpler system of scalar equations. □

Admittedly, the assumption on coefficient matrices $M$, $C$ and $K$ are relatively strong in theorem 1.1, however, we can deduce a more loose requirement. Assume coefficient matrices $M$ and $K$ are both nonsingular symmetric in $\mathbb{R}^N$ and one of them is non-negative definite. The Rayleigh damping is assume, which is:

$$C = aM + bK,$$

where $a$ and $b$ are parameters. We can have the following theorem.

**Theorem 1.2.** *If a elastodynamics problem has matrix form as $M\ddot{U}+C\dot{U}+KU = 0$, suppose it is Rayleigh damping, which means $C = aM + bK$, where $a$ and $b$ are parameters. In addition, assume coefficient matrices $M$, $K$ are both nonsingular symmetric in $\mathbb{R}^N$, and one of them is non-negative definite. This matrix form can be further reduced to a sing-degree-of-freedom problem.*

To prove this theorem, firstly, we need the following result.

**Theorem 1.3.** *Suppose $A$ and $B$ are n-by-n symmetric matrices, and define $C(\mu)$ by*

$$C(\mu) = \mu A + (1-\mu)B \qquad \mu \in \mathbb{R},$$

*If there exists a $\mu \in [0,1]$, such that $C(\mu)$ is non-negative definite and*

$$null(C(\mu)) = null(A) \cap null(B),$$

*then there exists a nonsingular $X$ such that both $X^TAX$ and $X^TBX$ are diagonal.*

For the proof of this theorem, one can refer to [Golub and Van Loan (2012)].

Based on our assumption, we can choose $\mu$ in theorem 1.3 as $\mu = 1$, thus, $C(\mu) = M$ which is non-negative definite. Since $M$ and $K$ are nonsingular in our assumption, thus

$$null(C(\mu)) = null(M) \cap null(K) = 0,$$

also holds. Hence, there exists a nonsingular $X$, such that both $X^TMX$ and $X^TKX$ are diagonal. Then, we can prove the theorem 1.2.

*Proof of Theorem 1.2.* With the help of theorem 1.3, we know that $X^TMX$ and $X^TKX$ are both diagonal. Denote

$$X^TMX = \Lambda_1,$$

$$X^TKX = \Lambda_2,$$

since $X$ is nonsingular, we can have

$$\begin{aligned} MX &= (X^T)^{-1}\Lambda_1 \\ &= (X^T)^{-1}\Lambda_2(\Lambda_2)^{-1}\Lambda_1 \\ &= KX(\Lambda_2)^{-1}\Lambda_1 \\ &= KX\Lambda, \end{aligned}$$

If we denote $X = (\alpha_1, \alpha_2, \cdots, \alpha_N)$ and $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_N)$, we can rewrite the above equation as

$$M\alpha_i = \lambda_i K\alpha_i.$$

Also, $\{\alpha_i\}_i^N$ constitute a basis function for $\mathbb{R}$, hence we have

$$U = \sum_{i=1}^{N} u_i(t)\alpha_i,$$

Substitute this representation into the matrix form, we have

$$\begin{aligned} 0 &= \sum_{i=1}^{N} \ddot{u}_i(t)M\alpha_i + \sum_{i=1}^{N} \dot{u}_i(t)C\alpha_i + \sum_{i=1}^{N} u_i(t)K\alpha_i \\ &= \sum_{i=1}^{N} (\lambda_i\ddot{u}_i(t) + (a\lambda_i + b)\dot{u}_i(t) + u_i(t))K\alpha_i, \end{aligned}$$

Since $\{\alpha_i\}_i^N$ is linear independent, and $K$ is invertible, thus, $\{K\alpha_i\}_i^N$ is also linear independent. From

$$\sum_{i=1}^{N}(\lambda_i\ddot{u}_i(t) + (a\lambda_i + b)\dot{u}_i(t) + u_i(t))K\alpha_i,$$

we can have, for every $i = 1, 2, \cdots, N$,

$$\lambda_i\ddot{u}_i(t) + (a\lambda_i + b)\dot{u}_i(t) + u_i(t) = 0,$$

which is the single-degree-of-freedom problem of problem $M\ddot{U} + C\dot{U} + KU = 0$. □

Recall the matrix form we get from employing finite element method to discretize the linear elastodynamics equation, which is equation (1.4). For equation (1.4), we can have the conclusion that, it can be further reduced to a single-degree-of-freedom problem.

This statement can by proved directly by using theorem (1.2). However, first of all, we need to prove these two coefficient matrices are both symmetric positive definite.

**Theorem 1.4.** *Matrices $M$ and $K$ in equation 1.4 both are symmetric positive definite.*

*Proof.* For matrix $M$, since $M_{i,j} = (\varphi_i, \varphi_j)$, it is easy to reach to this conclusion that $M$ is symmetric positive definite. For $K$, we can rewrite it as

$$\begin{aligned} K_{i,j} &= (\frac{1}{2}(\nabla\varphi_i + \nabla\varphi_i^T) + \frac{1}{2}(\nabla\varphi_i - \nabla\varphi_i^T), \mu(\nabla\varphi_j + \nabla\varphi_j^T) + \lambda(\nabla\cdot\varphi_j)I) \\ &= \frac{1}{2}(\nabla\varphi_i + \nabla\varphi_i^T, \mu(\nabla\varphi_j + \nabla\varphi_j^T)) + \frac{1}{2}(\nabla\varphi_i - \nabla\varphi_i^T, \mu(\nabla\varphi_j + \nabla\varphi_j^T)) \\ &+ (\nabla\varphi_i, \lambda(\nabla\cdot\varphi_j)I), \end{aligned}$$

In this expression, $(\nabla\varphi_i - \nabla\varphi_i^T, \mu(\nabla\varphi_j + \nabla\varphi_j^T)) = 0$, since $\nabla\varphi_i - \nabla\varphi_i^T$ is an anti-symmetric matrix and $\nabla\varphi_j + \nabla\varphi_j^T$ is symmetric.

As for $(\nabla\varphi_i, \lambda(\nabla\cdot\varphi_j)I)$, since $\lambda(\nabla\cdot\varphi_j)I$ is a diagonal matrix, we have

$$\begin{aligned}(\nabla\varphi_i, \lambda(\nabla\cdot\varphi_j)I) &= \int_\Omega \sum_k \lambda(\nabla\cdot\varphi_j)\frac{\partial\varphi_{ik}}{\partial x_k}dx \\ &= \int_\Omega \lambda(\nabla\cdot\varphi_j)\sum_k \frac{\partial\varphi_{ik}}{\partial x_k}dx \\ &= \int_\Omega \lambda(\nabla\cdot\varphi_i)(\nabla\cdot\varphi_j)dx.\end{aligned}$$

Hence, it is also symmetric positive definite.

Thus, because both $(\nabla\varphi_i+\nabla\varphi_i^T, \nabla\varphi_j+\nabla\varphi_j^T)$ and $(\nabla\varphi_i, (\nabla\cdot\varphi_j)I)$ are symmetric positive definite, $K$ is a symmetric positive definite matrix. □

Recall theorem 1.2, matrix $C$ is zero in this case, and both $M$ and $K$ are symmetric positive definite. Thus the statement the equation can be reduced to a scalar form is automatically satisfied. In fact, if we restrict the requirement on coefficient matrices $M$ and $K$ in theorem 1.2 to be both symmetric positive definite, we can prove the theorem 1.2 in another way.

*Proof of Theorem 1.2 without using theorem 1.3.* Since both $M$ and $K$ are symmetric positive definite, there exist matrices $A$ and $B$, such that $A^TA = M$, $B^TB = K$, thus

$$\begin{aligned}0 &= A^TA\ddot{\xi}(t) + aA^TA\dot{\xi}(t) + bB^TB\dot{\xi}(t) + B^TB\xi(t) \\ &= A(\ddot{\xi}(t) + a\dot{\xi}(t)) + (A^T)^{-1}B^TBA^{-1}A(b\dot{\xi}(t) + \xi(t))\end{aligned}$$

Let $A\xi(t) = \eta(t)$, $(A^T)^{-1}B^TBA^{-1} = C$, the equation then becomes,

$$(\ddot{\eta}(t) + a\dot{\eta}(t)) + C(b\dot{\eta}(t) + \eta(t)) = 0$$

where $C = (A^T)^{-1}B^TBA^{-1} = (BA^{-1})^T(BA^{-1})$ is a symmetric positive definite

matrix, which can be diagonalized as $C = P\tilde{C}P^T$. Thus

$$PP^T(\ddot{\eta}(t) + a\dot{\eta}(t)) + P\tilde{C}P^T(b\dot{\eta}(t) + \eta(t)) = 0$$

which implies that

$$P^T(\ddot{\eta}(t) + a\dot{\eta}(t)) + \tilde{C}P^T(b\dot{\eta}(t) + \eta(t)) = 0$$

Since $\tilde{C}$ is a diagonal matrix, we can even further rewrite the above equation into a scalar form. □

In this thesis, we consider it is reasonable to assume the elastodynamics problem we are solving can be reduced to a single-degree-of-freedom problem. Firstly, because the analysis presented above; secondly, there is a certain number of papers with thousands of citation also adopt this assumption (see, for example, [Chung and Hulbert (1993), Hughes (2012)]) when analysing the performance of other numerical algorithms.

Chapter 2

# The first scheme

In this chapter, we shall present our first second-order time integration algorithm, and study its four-step form as well as single-step four-stage form. Then we conduct a discussion regarding to its accuracy as well as stability properties.

In order to study the accuracy and stability properties of an algorithm, it suffices to consider $F(t) = 0$ for all $t \geq 0$. Thus, we shall work with the problem

$$M\ddot{U} + C\dot{U} + KU = 0, \tag{2.1}$$

from now onwards.

By the discussion in previous Chapter, we consider it is reasonable to assume the elastodynamics problem we are going to work on can be reduced to a scalar problem, thus, for simplicity, we now examine the following single-degree-of-freedom problem instead:

$$\ddot{u} - a\dot{u} - bu = 0.$$

By rewriting the above equation, we can have the following

$$\begin{cases} \dot{v} &= a\dot{u} + bu \\ v &= \dot{u} \end{cases} \tag{2.2}$$

## 2.1 Understanding the method

The first scheme solves equation (2.2) by the following numerical method:

$$\begin{cases} \frac{3v^{n+1}-4v^n+v^{n-1}}{2\Delta t} = av^{n+1} + b\frac{u^{n+2}+u^n}{2} \\ v^{n+1} = \frac{u^{n+2}-u^n}{2\Delta t} \end{cases} \tag{2.3}$$

where $u^n$ and $v^n$ are the approximation to the displacement and velocity respectively at time $t = n\Delta t$. This scheme is derived from DBF2, where we use $\frac{u^{n+2}+u^n}{2}$ to substitute $u^{n+1}$ and use $u^{n+2} = u^n + 2\Delta t v^{n+1}$ to update the numerical solution of displacement for next time interval.

By substituting the second equation into the first one, we can directly get this scheme's four-step scheme as

$$(3 - 2a\Delta t - 2b\Delta t^2)u^{n+2} - 4u^{n+1} + (-2b\Delta t^2 + 2a\Delta t - 2)u^n + 4u^{n-1} - u^{n-2} = 0,$$

which can be rewritten as

$$(3-2a\Delta t-2b\Delta t^2)u^{n+4}-4u^{n+3}+(-2b\Delta t^2+2a\Delta t-2)u^{n+2}+4u^{n+1}-u^n = 0, \tag{2.4}$$

Let $Y_n = (u^{n+3}, u^{n+2}, u^{n+1}, u^n)^T$ and $\Omega = 3 - 2a\Delta t - 2b\Delta t^2$, we can rewrite the above equation (2.4) in the following matrix equation form which is one-step four-stage,

$$Y_{n+1} = A_{M1}Y_n, \tag{2.5}$$

where

$$A_{M1} = \begin{pmatrix} \frac{4}{\Omega} & \frac{2b\Delta t^2 - 2a\Delta t + 2}{\Omega} & \frac{-4}{\Omega} & \frac{1}{\Omega} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

is the amplification matrix which generate information of the next time interval.

## 2.2 Accuracy and stability properties

### 2.2.1 Accuracy

The local truncation error measures the accuracy of a method at a specific step, assuming that the method was exact at the previous step. Recall the four-step form was derived as

$$(3 - 2a\Delta t - 2b\Delta t^2)u^{n+4} - 4u^{n+3} + (-2b\Delta t^2 + 2a\Delta t - 2)u^{n+2} + 4u^{n+1} - u^n = 0,$$

Thus, by expanding to Taylor series $u((n+j)\Delta t) = u(n\Delta t) + j\Delta t\dot{u}(n\Delta t) + j^2\frac{\Delta t^2}{2!}\ddot{u}(n\Delta t) + \cdots$, we have the local truncation error $\tau$ as

$$\begin{aligned} \Delta t^2\tau =& (-4b\Delta t^2)u(n\Delta t) + \Delta t(-4a\Delta t - 12b\Delta t^2)\dot{u}(n\Delta t) \\ &+ \Delta t^2(4 - 12a\Delta t - 20b\Delta t^2)\ddot{u}(n\Delta t) + \cdots, \end{aligned}$$

Recall that $\ddot{u} - a\dot{u} - bu = 0$, thus, $\dddot{u} - a\ddot{u} - b\dot{u} = 0$. Hence, the above equation can be further simplified as

$$\Delta t^2\tau = \Delta t^4(-20b\ddot{u}(n\Delta t) + \cdots),$$

which implies $\tau = O(\Delta t^2)$, and that is to say, this method is second-order accurate.

### 2.2.2 Stability

Stability of an integration method means that numerical errors present in the solution for any initial conditions do not amplify during the integration.

#### Routh-Hurwitz Stability Condition

Firstly, we can employ the matrix form and use previously derived amplification matrix to analyse. Recall the general formula (2.5) where $Y_{n+1} = A_{M1}Y_n$, we have $Y_k = A_{M1}^k Y_0$, where $Y_0 = (u^3, u^2, u^1, u^0)^T$. The amplification matrix $A_{M1}$ for the four-stage form is given by:

$$A_1 = \begin{pmatrix} \frac{4}{\Omega} & \frac{2b\Delta t^2 - 2a\Delta t + 2}{\Omega} & \frac{-4}{\Omega} & \frac{1}{\Omega} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

For the first scheme to be unconditionally stable, $A_1^k$ has to be bounded and the von Neumann necessary condition is $\rho_{M1} \leq 1$, where $\rho_{M1}$ is the spectral radius of matrix $A_{M1}$. To verify that $A_{M1}$ satisfies von Neumann necessary condition, we shall first try to make use of the Routh-Hurtwitz Stability Conditions.

**Theorem 2.1.** *Given a equation is expressed as*

$$a_0 z^n + a_1 z^{n-1} + \cdots + a_{n-1} z + a_n = 0, \tag{2.6}$$

*Let*

$$H = \begin{pmatrix} a_1 & a_3 & a_5 & \cdots \\ a_0 & a_2 & a_4 & \cdots \\ 0 & a_1 & a_3 & \cdots \\ 0 & a_0 & a_2 & \cdots \end{pmatrix},$$

*the Routh-Hurwitz conditions for the roots of the equation (2.6) to have non-positive*

*real parts are:* $a_0 > 0$ *and the other principal minors of* $H \geq 0$.

Firstly, by simple calculation, we can easily get the characteristic equation of $A_{M1}$ of this algorithm which is

$$\rho_{M1}(\lambda) = \lambda^4 - \frac{4}{\Omega}\lambda^3 + \frac{(-2b\Delta t^2 + 2a\Delta t - 2)}{\Omega}\lambda^2 + \frac{4}{\Omega}\lambda - \frac{1}{\Omega},$$

Substituting $\lambda$ with $\frac{1+z}{1-z}$ into the characteristic equation, we can then, transform the unit circle $|\lambda| \leq 1$ onto the left half complex plane $Real(z) \leq 0$.

The characteristic equation is thus, simplified to

$$\begin{aligned} 0 = &(-4b\Delta t^2)z^4 + (8 - 8a\Delta t - 8b\Delta t^2)z^3 + (16 - 16a\Delta t - 8b\Delta t^2)z^2 \\ &+ (-8a\Delta t - 8b\Delta t^2)z + (-4b\Delta t^2), \end{aligned}$$

By our previous discussion in Introduction Chapter, $a = 0$ in this equation, hence we can have

$$(-b\Delta t^2)z^4 + (2 - 2b\Delta t^2)z^3 + (4 - 2b\Delta t^2)z^2 + (-2b\Delta t^2)z + (-b\Delta t^2) = 0,$$

In Routh-Hurwitz Stability Conditions, all roots of this equation are on the left complex plane if and only if coefficients satisfy: (a) all coefficients are positive, (b) $a_1a_2 \geq a_0a_3$ and (c) $a_1a_2a_3 \geq a_0a_3^2 + a_1^2a_4$. This yields us three inequalities of coefficients:

$$\begin{cases} b\Delta t^2 < 0 \\ (2 - 2b\Delta t^2)(4 - 2b\Delta t^2) \geq 2(b\Delta t^2)^2 \\ (2 - 2b\Delta t^2)(4 - 2b\Delta t^2)(-2b\Delta t^2) \geq (-b\Delta t^2)[(-2b\Delta t^2)^2 + (2 - 2b\Delta t)^2] \end{cases} \tag{2.7}$$

If we denote $B = b\Delta t^2$, the two latter inequalities in equation (2.7) can be further

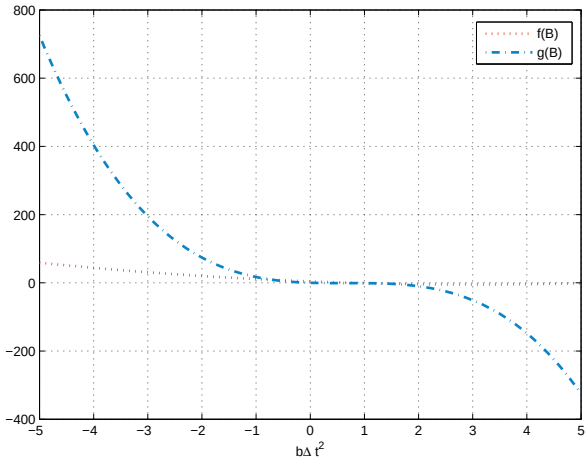


Figure 2.1: Value of two equations derived from Routh-Hurtwitz Stability Conditions to check the range of $b$ where these two inequalities can be satisfied. $x$-axis is the value of $B = b\Delta t^2$, $y$-axis is the value of function $f(B)$ and $g(B)$. When $b < 0$, $B < 0$, both inequalities in equation (2.7) are satisfied, while when $b > 0$ $B > 0$, $g(B) < 0$, hence the Routh-Hurtwitz condition is violated.

simplified into

$$\begin{cases} f(B) = B^2 - 6B + 4 \geq 0 \\ g(B) = -4B^3 + 8B^2 - 5B \geq 0 \end{cases}$$

Note that for the first quadratic equation, its two roots are given by $x_{1,2} = 3 \pm \sqrt{5}$, which are both positive. Thus, for all negative $b$, the second inequality is satisfied. As for the other inequality, it is easy to see that it is satisfied when $b < 0$, since $-B^3$, $B^2$ and $-B$ are all positive. We can also employ MATLAB to explicitly draw out the region where all three inequalities are satisfied in Figure 2.1.

Figure 2.1 clearly shows that when $b < 0$, both curves are above the $x$-axis. Thus, all three inequalities in equations (2.7) are satisfied. We could conclude that, the first scheme is unconditionally stable when $b < 0$. In fact, we could further employ the energy equation to check this conclusion.

### Energy equation

Recall the algorithm,

$$\begin{cases} \frac{3v^{n+1}-4v^n+v^{n-1}}{2\Delta t} = av^{n+1} + b\frac{u^{n+2}+u^n}{2} \\ v^{n+1} = \frac{u^{n+2}-u^n}{2\Delta t} \end{cases}$$

Thus, for any function $v$, we have

$$(\frac{3v^{n+1} - 4v^n + v^{n-1}}{2\Delta t}, v) = a(v^{n+1}, v) + b(\frac{u^{n+2} + u^n}{2}, v),$$

Especially, if we take $v = v^{n+1}$ hence, we have

$$\begin{aligned} &\frac{1}{4\Delta t}(|v^{n+1}|^2+|2v^{n+1} - v^n|^2+|v^{n+1} - 2v^n + v^{n-1}|^2-|v^n|^2-|2v^n - v^{n-1}|^2) \\ &= a|v^{n+1}|^2+b(\frac{u^{n+2} + u^n}{2}, v^{n+1}), \end{aligned}$$

Since $v^{n+1} = \frac{u^{n+2}-u^n}{2\Delta t}$, thus

$$\begin{aligned} (\frac{u^{n+2} + u^n}{2}, v^{n+1}) &= (\frac{u^{n+2} + u^n}{2}, \frac{u^{n+2} - u^n}{2\Delta t}) \\ &= \frac{1}{4\Delta t}(|u^{n+2}|^2-|u^n|^2), \end{aligned}$$

If we let $A^n = |v^n|^2+|2v^n - v^{n-1}|^2$ and $B^{n+1} = |v^{n+1} - 2v^n + v^{n-1}|^2$, and further assume $a = 0$, thus the above equation can be written as

$$A^{n+1} - A^n + B^{n+1} = b(|u^{n+2}|^2-|u^n|^2),$$

We further let $D^n = A^n - b(|u^{n+1}|^2+|u^n|^2)$, thus, we can have the following equation $D^{n+1} - D^n + B^{n+1} = 0$

Summing up both sides yields us

$$D^{n+1} - D^1 + \sum B^i = 0,$$

Since for any $i$, $B^i \geq 0$, thus

$$D^{n+1} - D^n \leq 0,$$

Which implies

$$|v^{n+1}|^2+|2v^{n+1} - v^n|^2-b(|u^{n+2}|^2+|u^{n+1}|^2) \leq |v^1|^2+|2v^1 - v^0|^2-b(|u^2|^2+|u^1|^2),$$

**Solving the characteristic equation**

Since $b$ is a real negative number, so we have unconditional stability for this method. So far, we proved that for $b$ as a real number, when $b < 0$, the first scheme is unconditionally stable.

To extend our discussion from $b$ as a real number into complex plane, we could directly solve the characteristic equation by first converting it into a depressed quartic equation through changing variables and then solve this equation by following the Ferrari's solution method.

**Theorem 2.2.** *Let $Ax^4 + Bx^3 + Cx^2 + Dx + E = 0$ be the general quartic equation we want to solve.*

*Let $x = \lambda + \frac{1}{A}$, the equation is converted into a depressed form*

$$x^4 + \alpha x^3 + \beta x^2 + \nu = 0,$$

*where*

$$\begin{cases} \alpha &= -\frac{3B^2}{8A^2} + \frac{C}{A} \\ \beta &= \frac{B^3}{8A^3} - \frac{BC}{2A^2} + \frac{D}{A} \\ \nu &= -\frac{3B^4}{256A^4} + \frac{B^2C}{16A^3} - \frac{BD}{4A^2} + \frac{E}{A} \end{cases}$$

*In addition, we can let*

$$\begin{cases} P &= -\frac{\alpha^2}{12} - \nu \\ Q &= -\frac{\alpha^3}{108} + \frac{\alpha\nu}{3} - \frac{\beta^2}{8} \\ R &= \frac{Q}{2} \pm \sqrt{\frac{Q^2}{4} + \frac{P^3}{27}}(either + or-) \\ u &= \sqrt[3]{R} \end{cases}$$

*and if we denote* $z = \sqrt{(\alpha + 2y)}$ *and*

$$y = -\frac{5}{6} - u + \begin{cases} 0 & if\ u = 0 \\ \frac{P}{3u} & if\ u \neq 0 \end{cases}$$

*the root* $x_{1,2,3,4}$ *can be expressed as*

$$x_{1,2,3,4} = -\frac{B}{4A} + \frac{\pm_s z \pm_t \sqrt{-(3\alpha + 2y \pm_s \frac{2\beta}{z})}}{2},$$

*where the two occurrences of* $\pm_s$ *must denote the same sign, but* $\pm_t$ *can be different.*

For our characteristic equation, we solve this equation as

$$(3 - 2a\Delta t - 2b\Delta t^2)\lambda^4 - 4\lambda^3 + (-2b\Delta t^2 + 2a\Delta t - 2)\lambda^2 + 4\lambda - 1 = 0, \qquad (2.8)$$

thus,

$$\begin{cases} A &= 3 - 2a\Delta t - 2b\Delta t^2 \\ B &= -4 \\ C &= -2b\Delta t^2 + 2a\Delta t - 2 \\ D &= 4 \\ E &= -1 \end{cases}$$

We still assume $a = 0$ and $b$ is complex, thus the spectral radius $\rho_{M1} = \max\{|\lambda_1|, \cdots, |\lambda_4|\}$ can be treated as the function of the variable $b\Delta t^2$. We use MATLAB to draw out the value of $\rho_{M1}$ as the function of $b\Delta t^2$ in Figure 2.2 and Figure 2.3.

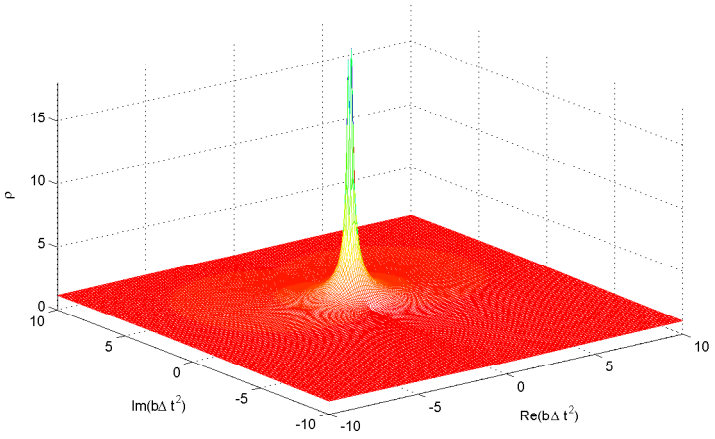


Figure 2.2: Value of $\rho_{M1} = \max\{|\lambda_1|, \cdots, |\lambda_4|\}$ as the function of complex number $b\Delta t^2$, where $Re(b\Delta t^2) \in (-10, 10)$ and $Im(b\Delta t^2) \in (-10, 10)$. Since the equation (2.8) is the characteristic equation of matrix $A_{M1}$ in equation (2.5), $\rho_{M1} \leq 1$ is the necessary condition for this scheme to be stable.

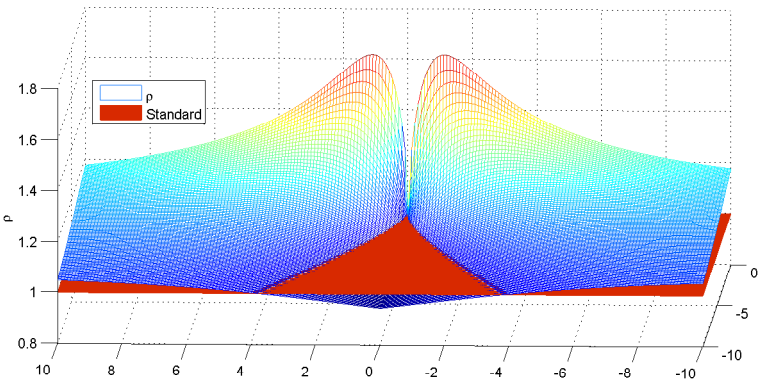


Figure 2.3: Value of $\rho_{M1} = \max\{|\lambda_1|, \cdots, |\lambda_4|\}$ as the function of complex number $b\Delta t^2$, where $Re(b\Delta t^2) \in (-10, 0)$ and $Im(b\Delta t^2) \in (-10, 10)$. On the left complex plane, compared with the standard plane $z = 1$, there is a triangle area where $\rho_{M1} \leq 1$, which implies the necessary condition for the first scheme to be stable is satisfied.

Fig 2.2 displays the value of $\rho_{M1}$ for the complex $b\Delta t^2$ where $|Re(b\Delta t^2)|\leq 10$ and $|Im(b\Delta t^2)|\leq 10$. In the right plane, around zero point, the value of $\rho_{M1}$ becomes relatively high. To have a closer look at other area on left plane, Fig 2.3 presents $\rho_{M1}$'s value, with $b\Delta t^2$ only having non-positive real part where $Re(b\Delta t^2) \in (-10, 0)$ and imaginary part $Im(b\Delta t^2) \in (-10, 10)$.

With analysis above, we can conclude that when $b < 0$, the first scheme is unconditionally stable. For $b$ as a complex number, $\rho_{M1} \leq 1$ can also hold for $b$ in a certain area on left plane.

## 2.3 Numerical dissipation

For structural dynamics models using spatial discretization, the high-frequency components do not represent the actual behavior of the original system. Hence, other than convergence and stability, it is also important for a numerical approximation method to have high-frequency damping features. In the high-frequency range, $b$ goes to infinity, if we assume $\Delta t$ is finite, thus $b\Delta t^2$ will go to infinity. Recall the characteristic equation as

$$\rho_{M1}(\lambda) = \lambda^4 - \frac{4}{\Omega}\lambda^3 + \frac{(-2b\Delta t^2 + 2a\Delta t - 2)}{\Omega}\lambda^2 + \frac{4}{\Omega}\lambda - \frac{1}{\Omega},$$

where $\Omega = 3 - 2a\Delta t - 2b\Delta t^2$. Since $b\Delta t^2$ goes to infinity, therefore, $\Omega$ will go to infinity, and the characteristic equation becomes:

$$\lambda^4 + \lambda^2 = 0,$$

It is easy to have four roots of this equation as $\lambda_1 = \lambda_2 = 0$,$\lambda_3 = i$ and $\lambda_4 = -i$. Therefore, this method fails to provide effective damping in the high-frequency range. This characteristic makes it less desirable in solving dynamic problems that often require controlled high-frequency damping.

Chapter 3

# The second scheme

Having examined accuracy and stability properties of the first scheme, in this section, we present the second one as well as its accuracy and stability properties analysis following the similar process as of the first one. However, the process we derive the matrix form as well as the three-step form of this scheme has slight differences.

## 3.1 Understanding the method

The difference between this scheme and the previous one lies in the update of new displacement $u^{n+2}$. This method solves equation (2.2) by:

$$\begin{cases} \frac{3v^{n+1}-4v^n+v^{n-1}}{2\Delta t} = av^{n+1} + b\frac{u^{n+2}+u^n}{2} \\ u^{n+2} = u^{n+1} + \frac{3\Delta t}{2}v^{n+1} - \frac{\Delta t}{2}v^n \end{cases} \tag{3.1}$$

where $u^n$ and $v^n$ are the approximation to the displacement and velocity respectively at time $t = n\Delta t$.

The above algorithm can then be rewritten in a matrix form, as

$$\begin{pmatrix} -b\Delta t & 0 & 3-2a\Delta t) & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -\frac{3\Delta t}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u^{n+2} \\ u^{n+1} \\ v^{n+1} \\ v^{n} \end{pmatrix} = \begin{pmatrix} 0 & b\Delta t & 4 & -1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & -\frac{\Delta t}{2} & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ u^{n} \\ v^{n} \\ v^{n-1} \end{pmatrix},$$

Let $\tilde{Y}_{n-1}$ denote $(u^{n+1}, u^n, v^n, v^{n-1})^T$. Then, the single-step four-stage method can be expressed in the compact form as $\tilde{Y}_n = \tilde{A}\tilde{Y}_{n-1}$, where

$$\tilde{A} = \begin{pmatrix} -b\Delta t & 0 & 3-2a\Delta t & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -\frac{3\Delta t}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 0 & b\Delta t & 4 & -1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & -\frac{\Delta t}{2} & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

further calculation shows that

$$\tilde{A} = \begin{pmatrix} \frac{4a\Delta t-6}{\Omega} & -\frac{3b\Delta t^2}{\Omega} & -\frac{9\Delta t+2a\Delta t^2}{\Omega} & \frac{3\Delta t}{\Omega} \\ 1 & 0 & 0 & 0 \\ -\frac{2b\Delta t}{\Omega} & -\frac{2b\Delta t}{\Omega} & \frac{b\Delta t^2-8}{\Omega} & \frac{2}{\Omega} \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

However, when calculating the determinant of matrix $\tilde{A}$, we find that $det(\tilde{A}) = 0$, which implies this scheme can be further reduced.

Recall this second scheme (3.1), if we substitute the second equation into the first one, we can have

$$\frac{3v^{n+1} - 4v^n + v^{n-1}}{2\Delta t} = av^{n+1} + \frac{b}{2}(u^{n+1} + u^n + \frac{3\Delta t}{2}v^{n+1} - \frac{\Delta t}{2}v^n),$$

In addition, from the second equation in (3.1), we can have

$$u^{n+1} = u^n + \frac{3\Delta t}{2}v^n - \frac{\Delta t}{2}v^{n-1},$$

Thus, we can deduce the matrix form as

$$\begin{pmatrix} b\Delta t & \frac{3b\Delta t^2}{2} + 2a\Delta t - 3 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ v^{n+1} \\ v^n \end{pmatrix} = \begin{pmatrix} -b\Delta t & \frac{b\Delta t^2}{2} - 4 & 1 \\ 1 & \frac{3\Delta t}{2} & \frac{-\Delta t}{2} \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} u^n \\ v^n \\ v^{n-1} \end{pmatrix},$$

Follow the same previous calculation, we have the three stage matrix form of this scheme as $Y_{n+1} = A_{M2}Y_n$, where $Y_n = (u^n, v^n, v^{n-1})^T$ and

$$A_{M2} = \begin{pmatrix} 1 & \frac{3\Delta t}{2} & \frac{-\Delta t}{2} \\ -\frac{4b\Delta t}{\Omega} & \frac{-2b\Delta t^2 - 8}{\Omega} & \frac{2 + b\Delta t^2}{\Omega} \\ 0 & 1 & 0 \end{pmatrix}, \tag{3.2}$$

where $\Omega = 3b\Delta t^2 + 4a\Delta t - 6$.

To derive this scheme's three-step form, we firstly calculate the characteristic equation of $A_{M2}$, and then employ the Cayley-Hamilton theorem.

**Theorem 3.1.** *In linear algebra, the Cayley-Hamilton theorem states that every square matrix over a commutative ring (such as the real or complex field) satisfies its own characteristic equation. More precisely, if $A$ is a given $n \times n$ matrix and $I_n$ is the $n \times n$ identity matrix, the characteristic polynomial of $A$ is defined as $\rho(A) = det(\lambda I_n - A)$, where "det" is the determinant operation. Then $\rho(A) = 0$.*

The characteristic equation of $A_{M2}$ can be easily calculated as:

$$\lambda^3 + \frac{(14 - b\Delta t^2 - 4a\Delta t)}{\Omega}\lambda^2 + \frac{(3b\Delta t^2 - 10)}{\Omega}\lambda + \frac{(2 - b\Delta t^2)}{\Omega} = 0, \tag{3.3}$$

By the Cayley-Hamilton Theorem we have

$$\begin{aligned} 0 =&(3b\Delta t^2 + 4a\Delta t - 6)A^3 + (14 - b\Delta t^2 - 4a\Delta t)A^2 \\ &+ (3b\Delta t^2 - 10)A + (2 - b\Delta t^2)I, \end{aligned} \tag{3.4}$$

Recall $Y_{n+1} = A_{M2}Y_n$ where $Y_n = (u^n, v^n, v^{n-1})^T$. We can have the general formula as $Y_{n+k} = A_{M2}^k Y_n$. Thus, for equation (3.4), if we multiply both sides with $Y_n$, we can get

$$\begin{aligned} &(3b\Delta t^2 + 4a\Delta t - 6)Y_{n+3} + (14 - b\Delta t^2 - 4a\Delta t)Y_{n+2} \\ &+ (3b\Delta t^2 - 10)Y_{n+1} + (2 - b\Delta t^2)Y_n = 0, \end{aligned}$$

In particular,

$$\begin{aligned} &(3b\Delta t^2 + 4a\Delta t - 6)u_{n+3} + (14 - b\Delta t^2 - 4a\Delta t)u_{n+2} \\ &+ (3b\Delta t^2 - 10)u_{n+1} + (2 - b\Delta t^2)u_n = 0, \end{aligned} \tag{3.5}$$

Thus, equation (3.5) presents the three-step form of the second time-integration scheme.

## 3.2 Accuracy and stability properties

### 3.2.1 Accuracy

To discuss the accuracy of this scheme, we still employ its three-step form (3.5) to calculate the local truncation error $\tau$, which is

$$\begin{aligned} \Delta t^2\tau =&(4b\Delta t^2)u(n\Delta t) + \Delta t(4a\Delta t + 10b\Delta t^2)\dot{u}(n\Delta t) \\ &+ \Delta t^2(-4 + 10a\Delta t + 13b\Delta t^2)\ddot{u}(n\Delta t) + \cdots, \end{aligned}$$

Recall that $\ddot{u} - a\dot{u} - bu = 0$, thus, $\dddot{u} - a\ddot{u} - b\dot{u} = 0$. Hence, the above equation can be further simplified as

$$\Delta t^2 \tau = \Delta t^4 (13 b\ddot{u}(n\Delta t) + \cdots),$$

which implies $\tau = O(\Delta t^2)$, and this is to say, again, this method has second-order accuracy.

### 3.2.2 Stability

To examine the stability of this second-order scheme, we follow the routine of the first method.

#### Routh-Hurwitz Stability Condition

Recall, the characteristic equation (3.3) as

$$\lambda^3 + \frac{(14 - b\Delta t^2 - 4a\Delta t)}{\Omega}\lambda^2 + \frac{(3b\Delta t^2 - 10)}{\Omega}\lambda + \frac{(2 - b\Delta t^2)}{\Omega} = 0,$$

As the previous process, we let $\lambda = \frac{1+z}{1-z}$ and transform the unit circle onto the left half complex plane. Thus, the characteristic equation becomes the following,

$$(32 - 8b\Delta t^2 - 8a\Delta t)z^3 + (20 - 4b\Delta t^2 - 16a\Delta t)z^2 + (-8b\Delta t^2 - 8a\Delta t)z - 4b\Delta t^2 = 0,$$

Let $a = 0$ and $B = b\Delta t^2$, then, the equation is simplified as

$$(8 - 2B)z^3 + (5 - B)z^2 + (-2B)z - B = 0,$$

By Routh-Hurwitz Stability Condition, in order to have all roots with non-positive real parts, coefficients need meets: (a) all positive and (b) $a_1 a_2 \geq a_0 a_3$.

Thus, we can have following two inequalities:

$$\begin{cases} b\Delta t^2 < 0 \\ (5-B)(-2B) \geq (-B)(8-2B) \end{cases} \tag{3.6}$$

After simplification, the second inequality in equation (3.6) actually yields the same result as the first inequality, $b\Delta t^2 < 0$. Thus, we can conclude, for all $b < 0$, the second scheme is unconditionally stable.

**Energy equation**

If we use energy function to verify, let $a = 0$, recall the second scheme as

$$\begin{cases} \frac{3v^{n+1}-4v^n+v^{n-1}}{2\Delta t} = b\frac{u^{n+2}+u^n}{2} \\ u^{n+2} = u^{n+1} + \frac{3\Delta t}{2}v^{n+1} - \frac{\Delta t}{2}v^n \end{cases}$$

From the second equation, we can get the relationship $u^{n+2}-u^{n+1} = \frac{3\Delta t}{2}v^{n+1} - \frac{\Delta t}{2}v^n$. The left-hand side of the first equation is exactly

$$\frac{3v^{n+1}-4v^n+v^{n-1}}{2\Delta t} = \frac{1}{\Delta t^2}[(u^{n+2}-u^{n+1}) - (u^{n+1}-u^n)],$$

We choose $v = [(u^{n+2}-u^{n+1}) + (u^{n+1}-u^n)] = u^{n+1} - u^n$ to be our test function. Thus, we can get the following equation

$$\begin{aligned} \frac{1}{\Delta t^2}(|u^{n+2}-u^{n+1}|^2 - |u^{n+1}-u^n|^2) &= \frac{b}{2}(|u^{n+2}|^2 - |u^n|^2) \\ &= \frac{b}{2}[(|u^{n+2}|^2 + |u^{n+1}|^2) - (|u^{n+1}|^2 + |u^n|^2)], \end{aligned}$$

By letting $A^{n+1} = \frac{1}{\Delta t^2}|u^{n+1}-u^n|^2 - \frac{b}{2}(|u^{n+1}|^2 + |u^n|^2)$, the above equation can be simplified as

$$A^{n+1} - A^n = 0,$$

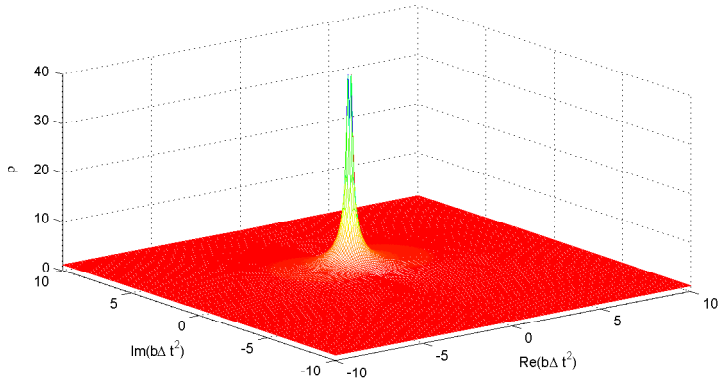


Figure 3.1: Value of $\rho_{M2} = \max\{|\lambda_1|, \cdots, |\lambda_3|\}$ as the function of complex number $b\Delta t^2$, where $Re(b\Delta t^2) \in (-10, 10)$ and $Im(b\Delta t^2) \in (-10, 10)$. Since we are solving the equation (3.3) which is the characteristic equation of matrix $A_{M2}$ in equation (3.2), $\rho_{M2} \leq 1$ is the necessary condition for this scheme to be stable.

Sum up both sides, we can get

$$\frac{1}{\Delta t^2}|u^{n+1} - u^n|^2 - \frac{b}{2}(|u^{n+1}|^2 + |u^n|^2) = \frac{1}{\Delta t^2}|u^1 - u^0|^2 - \frac{b}{2}(|u^1|^2 + |u^0|^2),$$

This implies when $b < 0$, we can have

$$|u^{n+2}|^2 + |u^{n+1}|^2 \leq |u^1|^2 + |u^0|^2 - \frac{2}{b\Delta t^2}|u^1 - u^0|^2,$$

With the above analysis we can safely conclude that the second scheme is unconditionally stable when $b < 0$.

### Solving the characteristic equation

Furthermore, if we extend from real number to complex plane, we can still directly solve the characteristic equation of $A_{M2}$ and regard its spectral radius $\rho_{M2}$ as the function of variable $b\Delta t^2$, and we draw out Figure 3.1 and Figure 3.2. The Figure 3.2 displays more clearly the value of $\rho_{M2}$ around zero point.

These two figures of the second scheme share similarity with those of the first scheme. From the Figure 3.2, we could see that for $b\Delta t^2$ with non-positive real part,

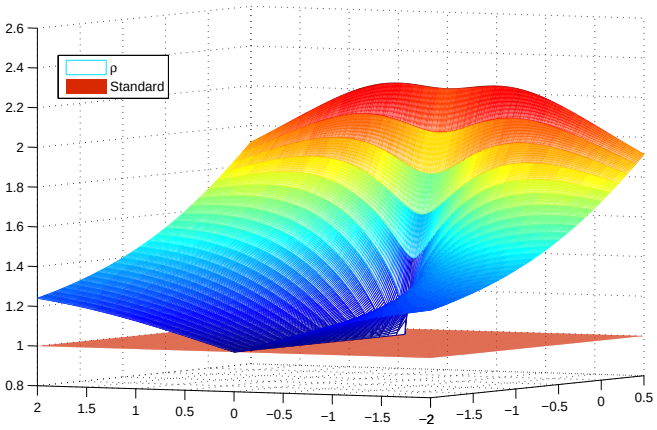


Figure 3.2: Value of $\rho_{M2} = \max\{|\lambda_1|, \cdots, |\lambda_3|\}$ as the function of complex number $b\Delta t^2$, where $Re(b\Delta t^2) \in (-2, 0.5)$ and $Im(b\Delta t^2) \in (-2, 2)$. On the left complex plane, the value of $\rho_{M2}$ intersects with the standard plane, which is $z = 1$, only on the $x$-axis. Hence, for $b$ as a complex number, only when $b$ is real negative, the necessary condition for the second scheme to be stable is satisfied.

$\rho_{M2} \leq 1$ only for $b$ as real numbers. We directly plot the value of $\rho_{M2}$ in Figure 3.3. This figure confirms what we observe from the Figure 3.2, that for $b$ which is non-positive real number, $\rho_{M2} \leq 1$.

Compared with the first scheme, the region where $\rho_{M2} \leq 1$ is smaller for this scheme. In general, if $b$ is a pure complex number, only the first scheme can be stable. For $b$ as a non-positive real number, both schemes are unconditionally stable.

## 3.3 Numerical dissipation

Recall the characteristic equation as

$$\rho_{M2}(\lambda) = -\lambda^3 - \frac{(14 - b\Delta t^2 - 4a\Delta t)}{\Omega}\lambda^2 - \frac{(3b\Delta t^2 - 10)}{\Omega}\lambda - \frac{(2 - b\Delta t^2)}{\Omega} = 0,$$

To examine the numerical dissipation of the method in the high-frequency range, let $\Omega$ goes to $\infty$, thus, we have

$$\rho_{M2}(\lambda) = -\lambda^3 + \frac{1}{3}\lambda^2 - \lambda + \frac{1}{3} = 0,$$

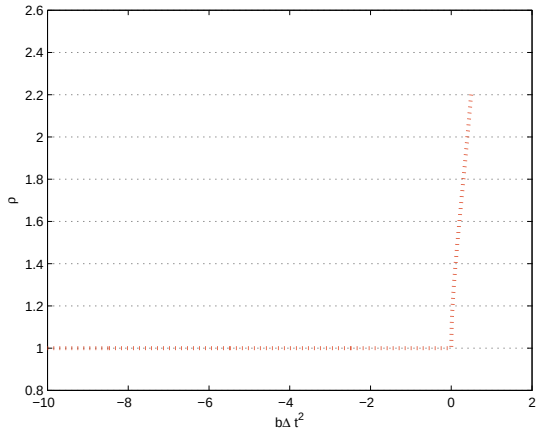


Figure 3.3: Value of $\rho_{M2} = \max\{|\lambda_1|, \cdots, |\lambda_3|\}$ as the function of complex number $b\Delta t^2$, where $(b\Delta t^2) \in (-10, 0.5)$. When $b\Delta t^2 \leq 0$, $\rho_{M2} \leq 1$ satisfies.

We can have roots of the above equation as $x_1 = 1/3, x_2 = i$ and $x_3 = -i$ which implies $\rho_{M2} = 1$. Therefore, this method also fails to perform effectively in the high-frequency range as the first one does.

Chapter 4

# Comparison with Newmark Algorithm

Newmark Algorithm is developed in 1959 for use in structural dynamics. It is widely used in numerical evaluation of the dynamic response of structures and solids such as in finite element analysis to model dynamic system. It has become one of the most popular methods among engineers in solving the second order elastodynamics equation.

Having examined the accuracy and stability properties of newly proposed two second-order accurate methods, we now compare them to the Newmark scheme.

The Newmark Algorithm solves equation(2.2) by the following numerical method:

$$\begin{cases} u^{n+1} = u^n + \Delta t\dot{u}^n + \frac{1}{2}(1-2\beta)\Delta t^2\ddot{u}^n + \beta\Delta t^2\ddot{u}^{n+1} \\ \dot{u}^{n+1} = \dot{u}^n + (1-\gamma)\Delta t\ddot{u}^n + \gamma\Delta t\ddot{u}^{n+1} \\ \ddot{u}^{n+1} = a\dot{u}^{n+1} + bu^{n+1} \end{cases} \tag{4.1}$$

This method is modified from the Taylor series expansions of both displacements $u(n\Delta t)$ and velocities $\dot{u}(n\Delta t)$ from step $n$ to step $n+1$. There are two parameters $\beta$ and $\gamma$ introduced to indicate how much of the acceleration at the end of the interval enters into the relations for velocity and displacement at the end of the interval. From previous work we know the Newmark Algorithm achieves second-order accuracy when $\gamma = \frac{1}{2}$, therefore we shall examine this algorithm using $\gamma = \frac{1}{2}$

from now onwards.

## 4.1 Stability region

By substituting $\ddot{u}^{n+1} = a\dot{u}^{n+1} + bu^{n+1}$ into the first two equations, we can rewrite equation (4.1) in a matrix form, like what we did for the second scheme,

$$\begin{pmatrix} 1-\Delta t^2\beta b & -\Delta t^2\beta a \\ -\frac{1}{2}\Delta t b & 1-\frac{1}{2}\Delta t a \end{pmatrix} \begin{pmatrix} u^{n+1} \\ \dot{u}^{n+1} \end{pmatrix}$$

$$= \begin{pmatrix} 1+\frac{1}{2}\Delta t^2(1-2\beta)b & \Delta t + \frac{1}{2}\Delta t^2(1-2\beta)a \\ \frac{1}{2}\Delta t b & 1+\frac{1}{2}\Delta t a \end{pmatrix} \begin{pmatrix} u^n \\ \dot{u}^n \end{pmatrix},$$

In order to compare the stability region of Newmark Algorithm with the previous two methods, we directly let $a = 0$, thus the above equation is written as:

$$\begin{pmatrix} 1-\Delta t^2\beta b & 0 \\ -\frac{1}{2}\Delta t b & 1 \end{pmatrix} \begin{pmatrix} u^{n+1} \\ \dot{u}^{n+1} \end{pmatrix} = \begin{pmatrix} 1+\frac{1}{2}\Delta t^2(1-2\beta)b & \Delta t \\ \frac{1}{2}\Delta t b & 1 \end{pmatrix} \begin{pmatrix} u^n \\ \dot{u}^n \end{pmatrix},$$

By simple calculation, we can write the above equation in a more compact form $Y_{n+1} = A_{New}Y_n$, where $Y_n = (u^n, \dot{u}^n)^T$ and

$$A_{New} = \begin{pmatrix} 1-\frac{\Delta t^2 b}{2\Omega} & -\frac{\Delta t}{\Omega} \\ b\Delta t - \frac{b^2\Delta t^3}{4\Omega} & 1-\frac{b\Delta t^2}{2\Omega} \end{pmatrix}, \tag{4.2}$$

where $\Omega = \beta b\Delta t^2 - 1$.

The characteristic equation of matrix $A_{New}$ is easy to calculated as

$$\begin{aligned} \rho_{New}(\lambda) &= \lambda^2 - 2(1-\frac{b\Delta t^2}{2\Omega})\lambda + [(1-\frac{b\Delta t^2}{2\Omega})^2 + \frac{\Delta t}{\Omega}(\frac{b\Delta t}{2} - \frac{b^2\Delta t^3}{4\Omega})] \\ &= \lambda^2 + (\frac{b\Delta t^2}{\Omega} - 2)\lambda + 1, \end{aligned} \tag{4.3}$$

Next, we shall try to make use of the Routh-Hurwitz Stability Condition to

obtain necessary condition on $\beta$.

In order to let $|\lambda| \leq 1$, where $\lambda$ is the root of characteristic equation(4.3).We substitute $\lambda$ with $\frac{1+z}{1-z}$ in to the equation.

The characteristic equation (4.3) then can be simplified to

$$(4 - \frac{B}{\beta B - 1})z^2 + \frac{B}{\beta B - 1} = 0,$$

where $B = b\Delta t^2$.

For the second-order polynomial, the matrix $H$ in the Routh-Hurwitz condition is $\begin{pmatrix} \lambda_1 & 0 \\ \lambda_0 & \lambda_2 \end{pmatrix}$. Thus, the Routh-Hurwitz condition for this quadratic polynomial to have roots with non-positive real parts are $4 > \frac{B}{\beta B - 1} \geq 0$. If we assume $b$ is a non-positive real number, then, this inequality can further yield the condition of $\beta$ is:

$$\beta \geq \frac{1}{4},$$

Thus, for the Newmark Algorithm to be second-order accurate and unconditionally stable given $b \leq 0$, we need $\beta \geq \frac{1}{4}$.

There is another way could be employed to analyse this characteristic equation. If we denote roots of this characteristic equation as $\lambda_1$ and $\lambda_2$, according to the property of quadratic equations' roots, we have

$$\begin{cases} \lambda_1 + \lambda_2 = 2 - \frac{b\Delta t^2}{\Omega} \\ \lambda_1 \lambda_2 = 1 \end{cases}$$

If $\rho_{New} \leq 1$, which means $\mid \lambda_1 \mid \leq 1$ and $\mid \lambda_2 \mid \leq 1$. However, since $\lambda_1 \lambda_2 = 1$, thus, we can get $\mid \lambda_1 \mid = 1$ and $\mid \lambda_2 \mid = 1$. Therefore, we can assume $\lambda_1 = e^{i\theta}$ and $\lambda_2 = e^{-i\theta}$. By $\lambda_1 + \lambda_2 = 2 - \frac{b\Delta t^2}{\Omega}$, we can have

$$2\cos(\theta) = 2 - \frac{b\Delta t^2}{\Omega} = 2 - \frac{b\Delta t^2}{b\Delta t^2 \beta - 1},$$

Thus,

$$b\Delta t^2 = \frac{2\cos(\theta) - 2}{1 + (2\cos(\theta) - 2)\beta},$$

Which implies that Newmark method is stable only for $b$ as a real number.

Compared with two newly proposed schemes discussed in this paper, the first method is the only one which can have stability when $b$ has non-zero imaginary part. Both Newmark Algorithm and our second scheme can only be stable with $b$ as a real number. And all three schemes are unconditionally stable when $b < 0$. Another advantage of these two schemes compared with Newmark is that, there is no need to choose any parameters, which is more convenient to use in practice.

## 4.2 Numerical dissipation

As for the numerical dissipation of Newmark Algorithm, when $\Omega$ approaches to infinity, the characteristic equation becomes $\lambda^2 + 1 = 0$. hence, two roots are $i$ and $-i$, which also makes Newmark undesirable in solving dynamic problems that require controlled high-frequency damping.

Thus, when it comes to numerical dissipation, all three schemes need further study to investigate improvements.

Chapter 5

# Analytic solution and temporal accuracy checks

## 5.1 Accuracy check with ODE problems

Now, we would like to have an accuracy check to the first and the second schemes discussed previously. Firstly, we directly test the numerical performance by analytic solutions for the ODE equation (2.2) and compare their numerical performance to Newmark Algorithm.

We choose the equation that governs a mass-spring system while in motion. A mass $m$ is suspended at the end of a spring, its weight stretches the spring by a length $L$ to reach a static state. Let $u(t)$ denote the displacement, as a function of time, of the mass relative to its equilibrium position and follow the convention that downward is positive. Then, $u > 0$ means the spring is stretched beyond its equilibrium length, while $u < 0$ means that the spring is compressed.

The problem has a general form as

$$m\ddot{u} + \gamma\dot{u} + ku = F(t),$$

where $u$ is the displacement of the mass spring relative to its equilibrium position,

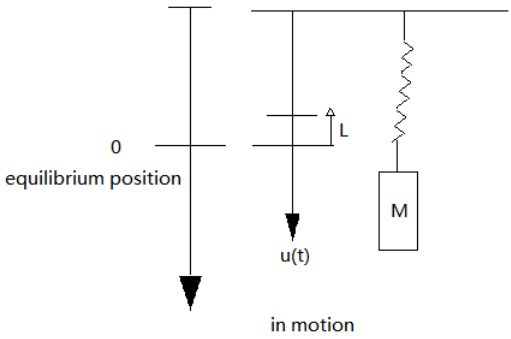


Figure 5.1: A mass-spring system, mass M is suspended at the end of a spring, by a length L, its weight stretches the spring to reach a equilibrium position. $u(t)$ as a function of time, is the displacement of mass relative to its equilibrium position.

$m$ ($m > 0$) is the mass, $\gamma$ ($\gamma \geq 0$) is the damping constant and $k$ ($k > 0$) is the spring (Hooke's) constant. When $\gamma = 0$, it is undamped free vibration, while $\gamma > 0$, is damped.

The characteristic equation of this problem is $mr^2 + \gamma r + k = 0$. Its solution will be either negative real numbers, or complex numbers with negative real parts. The displacement $u(t)$ behaves differently depending on the size of $\gamma$ relative to $m$ and $k$. In this thesis, we choose one analytic solution from each of damped and undamped vibration to analyse.

Firstly, considering an undamped free vibration, in which $\gamma = 0$, $F(t) = 0$. Thus, the motion equation should be

$$m\ddot{u} + ku = 0,$$

By solving the characteristic equation, we can have the roots $r = \pm i\sqrt{\frac{k}{m}}$. Denote

$\omega_0 = \sqrt{\frac{k}{m}}$, the general solution to this ODE can be represented as

$$u(t) = C_1 \cos \omega_0 t + C_2 \sin \omega_0 t,$$

We let the exact solution to be

$$u(t) = 2 \cos 5t + 5 \sin 5t,$$

Thus, the ODE problem we constructed becomes

$$\ddot{u} + 25u = 0, \tag{5.1}$$

and the initial condition for the problem (5.1) is

$$\begin{cases} u_0 = 2 \\ v_0 = 25 \end{cases}$$

For the accuracy check, we simulated our numerical results until $t_{end} = 2.0$ in Matlab and listed errors for different time steps in the Table 5.1.

Table 5.1: Computational errors $E$ using the first, the second and Newmark schemes for the undamped vibration problem (5.1) (and local order $\alpha$), calculated until time $t_{end} = 2.0$ with different time steps. $\Delta t = 1e-2$, $\alpha = \frac{\log_{10}(E_{k-1}/E_k)}{\log_{10}(h_{k-1}/h_k)}$.Newmark1 stands for Newmark method with $\beta = 1/3$, Newmark2 is Newmark method with $\beta = 1$.

| Method | $\Delta t$ <br> $\Delta t$ | $\frac{\Delta t}{4}$ | $\frac{\Delta t}{16}$ | $\frac{\Delta t}{64}$ | $\frac{\Delta t}{256}$ |
|---|---|---|---|---|---|
| Method1 | 0.017 | 9.676e-04 (2.06) | 5.934e-05(2.01) | 3.692e-06(2.00) | 2.168e-07(2.04) |
| Method2 | 0.005 | 3.057e-04(2.07) | 1.865e-05(2.02) | 1.159e-06(2.00) | 7.233e-08(2.00) |
| Newmark1 | 0.010 | 6.247e-04(2.00) | 3.904e-05(2.00) | 2.440e-06(2.00) | 1.525e-07(2.00) |
| Newmark2 | 0.038 | 0.0024(2.00) | 1.490e-04(2.00) | 9.315e-06(2.00) | 5.822e-07(2.00) |

Local errors $\alpha$ in Table 5.1 give us clean second-order accuracy for all three schemes. In addition, we can also tell that, with different $\beta$, numerical performance from Newmark method is different. Thus, we further draw out errors along with time for Newmark method with various value of $\beta$ in the Figure 5.2. Figure 5.2

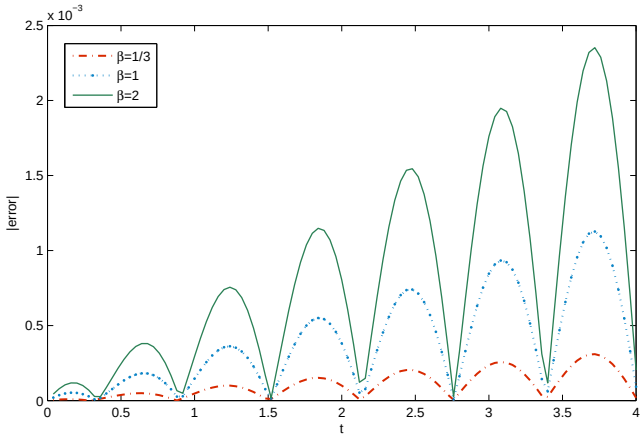


Figure 5.2: Errors for the Newmark Algorithm with different value of $\beta$ along with the time when solving ODE problem (5.1) with different time step.

shows that bigger $\beta$ yields bigger error when solving ODE problem (5.1).

Errors along with time for the first, the second methods compared to Newmark with $\beta = 1$ are displayed in the Figure 5.3.

For damped free vibration, where $\gamma > 0$ and $F(t) = 0$, we choose to solve the vibration whose characteristic equation has two distinct real roots, which means $\gamma^2 > 4mk$. Assuming two distinctive roots for the characteristic equation are $r_1$ and $r_2$, the displacement has the following form

$$u(t) = C_1 e^{r_1 t} + C_2 e^{r_2 t},$$

where $C_1$ and $C_2$ are determined by initial conditions. A mass-spring with this type of displacement function is called overdamped.

We choose the exact solution to be $u(t) = e^{-t} - e^{-2t}$. Thus, the constructed ODE equation becomes

$$\ddot{u} + 3\dot{u} + 2u = 0, \tag{5.2}$$

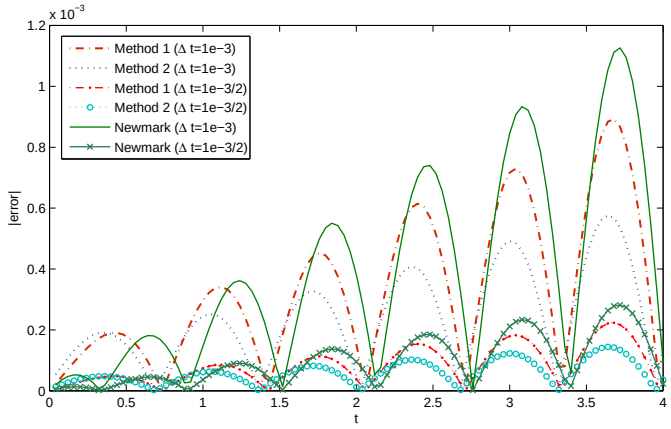


Figure 5.3: Errors for the first, the second and Newmark schemes along with the time when solving ODE problem (5.1) with different time steps. Let $\beta = 1$ when employing Newmark Algorithm.

The initial displacement and velocity are

$$\begin{cases} u_0 = 0 \\ v_0 = 1 \end{cases}$$

To check the accuracy, we simulate our numerical results to $t_{end} = 1.0$ and listed errors for different time steps in Table 5.2. Local orders $\alpha$ give clear and clean second-order accuracy in time for three methods.

Table 5.2: Computational errors $E$ using the first, the second and Newmark schemes for the undamped vibration problem (5.2) (and local order $\alpha$), calculated until time $t_{end} = 1.0$ with different time steps. $\Delta t = 1e-2$, $\alpha = \frac{\log_{10}(E_{k-1}/E_k)}{\log_{10}(h_{k-1}/h_k)}$. Newmark1 represents the Newmark method with $\beta = 1/3$, and Newmark2 is the Newmark method with $\beta = 1$.

| Method | $\Delta t$ | $\frac{\Delta t}{4}$ | $\frac{\Delta t}{16}$ | $\frac{\Delta t}{64}$ | $\frac{\Delta t}{256}$ |
|---|---|---|---|---|---|
| Method1 | 2.1234e-04 | 1.314e-05(2.01) | 8.195e-07(2.00) | 5.134e-08(2.00) | 1.279e-09(2.66) |
| Method2 | 1.4418e-04 | 8.954e-06(2.00) | 5.587e-07(2.00) | 3.491e-08(2.00) | 2.181e-09(2.00) |
| Newmark1 | 2.3747e-05 | 1.484e-06(2.00) | 9.276e-08(2.00) | 5.798e-09(2.00) | 3.628e-10(2.00) |
| Newmark2 | 1.6607e-04 | 1.038e-05(2.00) | 6.487e-07(2.00) | 4.054e-08(2.00) | 2.534e-09(2.00) |

Figure 5.4 shows errors along with different time for Newmark method with three different choices of $\beta$ and Figure 5.5 shows errors for the first, the second and Newmark Algorithm along with time.

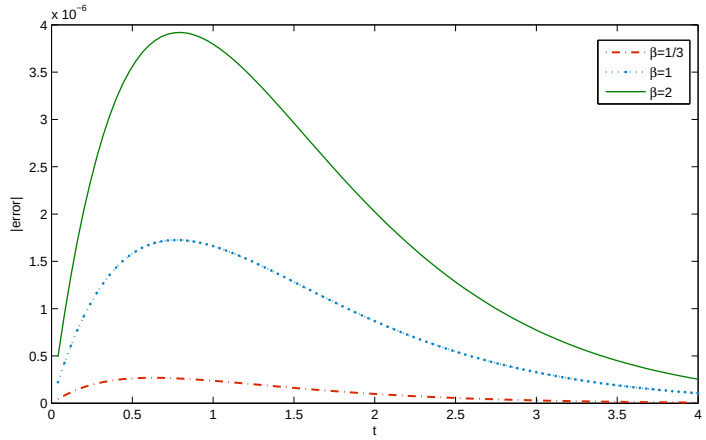


Figure 5.4: Errors for the Newmark schemes with different value of $\beta$ along with the time when solving ODE problem (5.2) with different time steps.

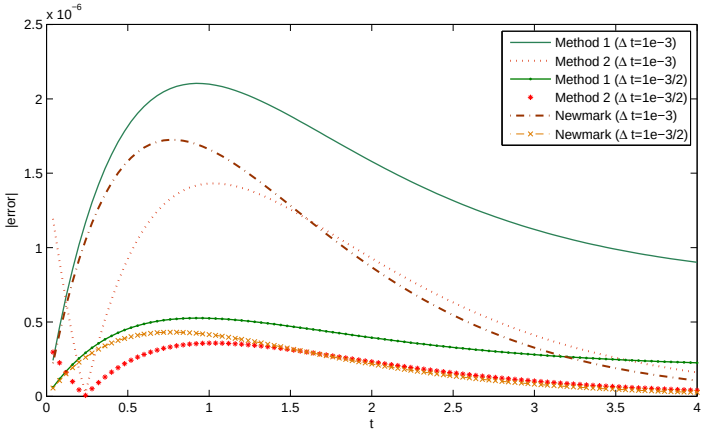


Figure 5.5: Errors for the first, the second and Newmark schemes along with the time when solving ODE problem (5.2) with different time steps. Choose $\beta = 1$ when applying Newmark method.

From Figure 5.3 and Figure 5.5, in general, the second scheme performed slightly better than the first one did numerically in these two ODE problems. While from Figure 5.2 and Figure 5.4 we can conclude that, the choice of $\beta$ when employing Newmark method will exert impact on the numerical result. When solving these two constructed ODE problems, the bigger $\beta$ yields bigger error.

## 5.2 Accuracy check with linear elastodynamics

After employing all three methods to solve the ODE problem, now, we will only focus on employing the first and the second schemes and would like to construct a elastodynamics problem whose exact analytic solution is known, and thus can be used for code validation. Recall the general elastodynamics equation,

$$\rho \frac{\partial^2 u}{\partial t^2} = \nabla \cdot \sigma + G,$$

Without loss of generality, we let $\rho = 1$, and assume the displacement has the expression $u = (u_1, u_2)$ in a two-dimensional domain, where

$$\begin{cases} u_1 = \sin(2\pi x + y)\sin(t) \\ u_2 = [1 + \sin(x + t)]\sin(2\pi y) \end{cases}$$

Thus the velocity $v$ should be

$$\begin{cases} v_1 = \sin(2\pi x + y)\cos(t) \\ v_2 = \sin(2\pi y)\cos(x + t) \end{cases}$$

Let lame constant $\mu = 10, \lambda = 10$. By calculation, we can have the body force $G =$

$$\begin{pmatrix} (-40\pi\cos(2\pi y)\cos(t + x) + 120\pi^2\sin(2\pi x + y)\sin(t) + 9\sin(2\pi x + y)\sin(t)) \\ 9\sin(2\pi y)\sin(t + x) + 40\pi\sin(2\pi x + y)\sin(t) + 120\pi^2\sin(2\pi y)(1 + \sin(t + x)) \end{pmatrix},$$

We choose the unit square as our domain, and give the right boundary, where $x = 1$, Neumann condition which is $T = n \cdot \sigma$, and others Dirichlet boundary condition. Thus, the two-dimensional linear elastodynamics problem we constructed can be stated as blow:

$$\begin{aligned}
&\frac{\partial^2 u}{\partial t^2} = \nabla \cdot \sigma + G \\
&\sigma = 10(\nabla u + \nabla u^T) + 10(\nabla \cdot u)I \\
&u_0 = (0, [1 + \sin(x)] \sin(2\pi y)) \qquad in\, \Omega \\
&v_0 = (\sin(2\pi x + y), \sin(2\pi y) \cos(x)) \qquad in\, \Omega \\
&u = (\sin(2\pi x + y) \sin(t), [1 + \sin(x + t)] \sin(2\pi y)) \qquad on\, \Gamma_1 \times I \\
&(n \cdot \sigma) = T \qquad on\, \Gamma_2 \times I
\end{aligned} \tag{5.3}$$

where

$$G = \begin{pmatrix} (-40\pi \cos(2\pi y) \cos(t + x) + 120\pi^2 \sin(2\pi x + y) \sin(t) + 9 \sin(2\pi x + y) \sin(t)) \\ 9 \sin(2\pi y) \sin(t + x) + 40\pi \sin(2\pi x + y) \sin(t) + 120\pi^2 \sin(2\pi y)(1 + \sin(t + x)) \end{pmatrix},$$

$$T = \begin{pmatrix} 10(6\pi \cos(y + 2\pi) + 2\pi \cos(2\pi y)(1 + \sin(1 + t))) \\ 10(\cos(y + 2\pi) \sin(t) + \sin(2\pi y) \cos(t + 1)) \end{pmatrix},$$

To solve this problem, we firstly apply Finite Element Method to semi-discretize the equation. We denote the finite element space with dimension $N$ to be $V_h$, and basis function to be $\psi_j$, $j = 1 \cdots N$. Want to find $\varphi_h \in V_h$ such that $\varphi_h = \sum_{i=1}^{n} \xi_i \psi_i(x)$ and

$$(T, \nabla \psi_j)_{\Gamma_2} - (\sigma, \nabla \psi_j) + (G, \psi_j) = (\ddot{\varphi}_h, \psi_j) \qquad \forall \psi_j \in V_h, t \in I, \tag{5.4}$$

The software we choose to use is called FEniCS. It is a software which enables automated solution of differential equations by providing scientific computing, tools for working with computational meshes, finite element variational formulations of both ordinary and partial differential equations. The FEniCS Project was initiated in 2003 as a research collaboration between the University of Chicago and Chalmers

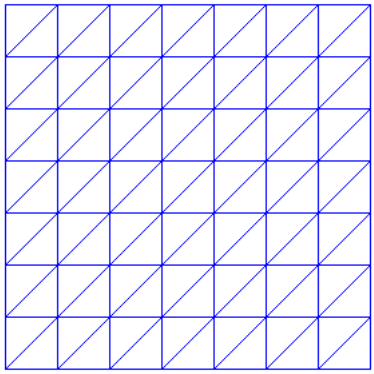

Figure 5.6: Computational mesh generated by FEniCS when applying Finite Element Method to solve linear elastodyanmics problem (5.3)

University of Technology. For more information, one can refer to [Logg, Mardal, and Wells (2012), Alnæs, Hake, Kirby, Langtangen, Logg, and Wells (2011)]. In FEniCS, we let the function space to be Lagrange family elements of degree 2. And the mesh generated during calculation is plotted in Figure 5.6.

Then, our first and second schemes were applied on equation (5.4). However, before starting the iteration, we firstly need to use the initial condition to obtain some necessary values, $v^1, u^1$ and $u^2$ which are the numerical solutions of both velocity and displacement at time $dt$ and displacement at $2dt$ respectively. We would like to use the following scheme to calculate,

$$\begin{cases} \frac{v^1 - v^0}{\Delta t} = av^1 + b\frac{u^2+u^0}{2} \\ u^1 = u^0 + \Delta t v^0 \\ u^2 = u^0 + 2\Delta t v^0 \end{cases}$$

During iteration, at each time stepping, we used the default linear solver package within FEniCS distribution for Ubuntu, which is sparse LU decomposition, to solve this linear problem. The final iteration ended until the time $t_{end} = 1.0$. We listed numerical results in Table 5.3, and draw out the log-log plot in Figure 5.7. Both

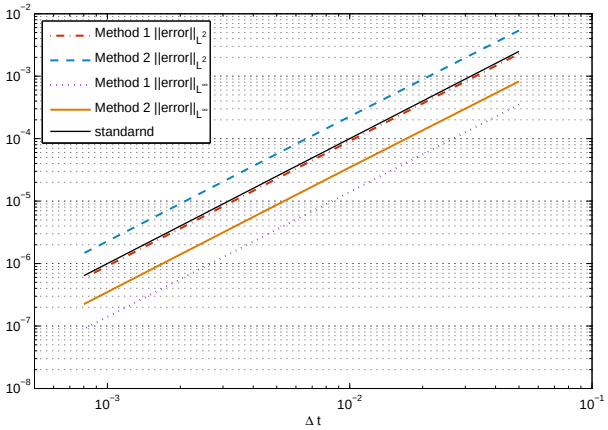


Figure 5.7: The log-log figure of errors calculated by using the first and the second second-order accurate schemes to solve the elastodynamics problem (5.3) in FEniCS.

local order in the table as well as slope of each line give us clear and clean second-order accuracy for the first and the second scheme. In addition, when solving this elastodynamics equation, the first scheme gives slightly better performance.

Table 5.3: Computational errors E using the first and the second second-order accurate finite element schemes respectively to the elstodynamics problem (5.3) (and local order $\alpha$), calculated until the time $t_{end} = 1$ with various time steps. $u_h^1$ and $u_h^2$ are numerical solutions calculated by the first and the second schemes, $\Delta t = 0.04$, $\alpha = \frac{\log_{10}(E_{k-1}/E_k)}{\log_{10}(h_{k-1}/h_k)}$.

| Error | $\frac{\Delta t}{\Delta t}$ | $\frac{\Delta t}{2}$ | $\frac{\Delta t}{4}$ | $\frac{\Delta t}{8}$ |
|---|---|---|---|---|
| $\|\|u - u_h^1\|\|_{L^2}$ | 0.00229988 | 0.000365340(2.6542) | 9.13318e-05(2.0001) | 2.28328e-05(2.0000) |
| $\|\|u - u_h^2\|\|_{L^2}$ | 0.00349157 | 0.000893172(1.9669) | 0.000225815(1.9838) | 5.67680e-05(1.9920) |
| $\|\|u - u_h^1\|\|_{L^\infty}$ | 0.000655078 | 5.60533e-05(3.5468) | 1.40128e-05(2.0001) | 3.50318e-06(2.0000) |
| $\|\|u - u_h^2\|\|_{L^\infty}$ | 0.000535704 | 0.000137038(1.9669) | 3.46462e-05(1.9838) | 8.70979e-06(1.9920) |

## 5.3 Solving practical elastodynamics problems

Next, we would like to employ the second method to solve some practical problems both linear and nonlinear. Consider a solid material in a force field with part of its boundary fixed and the rest part free to move. One example is a beam with one of its end sticked to the wall and the other is free. Suppose we know its shape

before the deformation and we want to determine its shape once the body is lifted up or stands up. The solid material can be either linear elastic or nonlinear elastic. In this section, we would like to employ the second method to solve the linear and nonlinear elastodynamics problem of such a solid beam which we do not know the analytic formula for the exact solution.

The original configuration is called reference configuration and is denoted by $\Omega$. We use $x$ to denote any point in $\Omega$. When the force applied, the body deform, and $x$ eventually moves to $\varphi(x)$. We assume that $\Omega = [0,5] \times [0,1] \times [0,1]$. The initial displacement and velocity are both zero. There is no body force. The left facet is fixed, while there is a traction force only on the right facet, which is depend on time $T = (0, 0, 200t)$. We assume the density $\rho = 1$, and lame constant $\mu = 2.3 \times 10^4$, $\lambda = 1.05 \times 10^5$.

Hence, for the linear elastic solid material, the elastodynamics problem can be presented as

$$\begin{aligned}
&\frac{\partial^2 u}{\partial t^2} = \nabla \cdot \sigma \qquad in\, \Omega \times I \\
&\sigma = 2.3 \times 10^4 (\nabla u + \nabla u^T) + 1.05 \times 10^5 (\nabla \cdot u) I \\
&u(x,0) = 0 \qquad in\, \Omega \\
&v(x,0) = 0 \qquad in\, \Omega \\
&u = 0 \qquad on\, \Gamma_1 \times I \\
&(n \cdot \sigma) = (0,0,200t) \qquad on\, \Gamma_2 \times I \\
&(n \cdot \sigma) = (0,0,0) \qquad on\, \Gamma_3 \times I
\end{aligned} \tag{5.5}$$

where $\Gamma_1$ is the facet where $x_1 = 0$, $\Gamma_2$ is $x_1 = 5$ and $\Gamma$ is the rest.

Figure 5.8 displays the original configuration of our elastic beam at time zero as well as the computational mesh used in calculation when employing finite element method.

We solve this problem by the second method, and choose Lagrange elements of

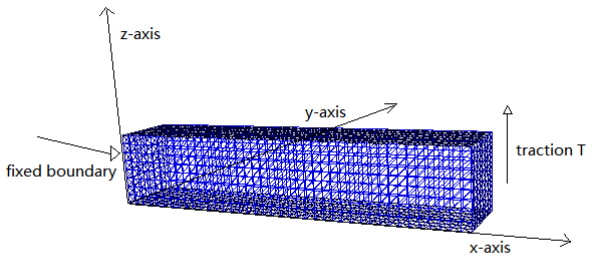


Figure 5.8: Original configuration of the elastic beam in the constructed elastodynamics problem with unknown analytic solution whose left facet is fixed. The mesh plotted is the computational mesh used in Finite Element Method.

degree 2 as function space when applying finite element method. Table 5.4 displays the numerical results of the displacement along $z$-axis at point $(5, 1, 1)$ with time step $\Delta t = 0.01$ and $\Delta t = 0.005$.

Table 5.4: Displacement along $z$˙axis at point $(5, 1, 1)$, calculated by time step $\Delta t = 0.01$ and $\Delta t/2 = 0.005$ respectively at time $t = 0.25$, $t = 0.375$, $t = 0.5$, $t = 0.625$ and $t = 0.75$

| | $t = 0.25$ | $t = 0.375$ | $t = 0.5$ | $t = 0.625$ | $t = 0.75$ | | |
|---|---|---|---|---|---|---|---|
| linear ($\Delta t$) | 0.31682 | 0.64980 | 0.79795 | 0.81837 | 0.93600 | 1.25316 | 1.52569 |
| linear ($\Delta t/2$) | 0.30512 | 0.62845 | 0.79458 | 0.81610 | 0.93006 | 1.22823 | 1.51643 |

We draw out the position of this elastic beam at time $t = 0.25$, $t = 0.5$, $t = 0.75$ and $t = 1.0$ respectively with time step $\Delta t = 0.01$ and $\Delta t = 0.005$ in Figure 5.9. The elastic beam is gradually lifted along with time. Since the traction force increases as time increases, the speed of the beam's lifting is accelerated.

Besides linear elasticity, we can also assume the material to be nonlinear elastic while other conditions remain the same. We let the material to be Venant-Kirchhoff elastic, thus the strain energy is

$$I(\varphi(\cdot, t)) = \int_\Omega W(\nabla\varphi)dx,$$

where $W = \mu tr(E^2) + \frac{\lambda}{2}(trE)^2$, $F = \nabla\varphi$ and $E = \frac{1}{2}(F^TF - I)$.

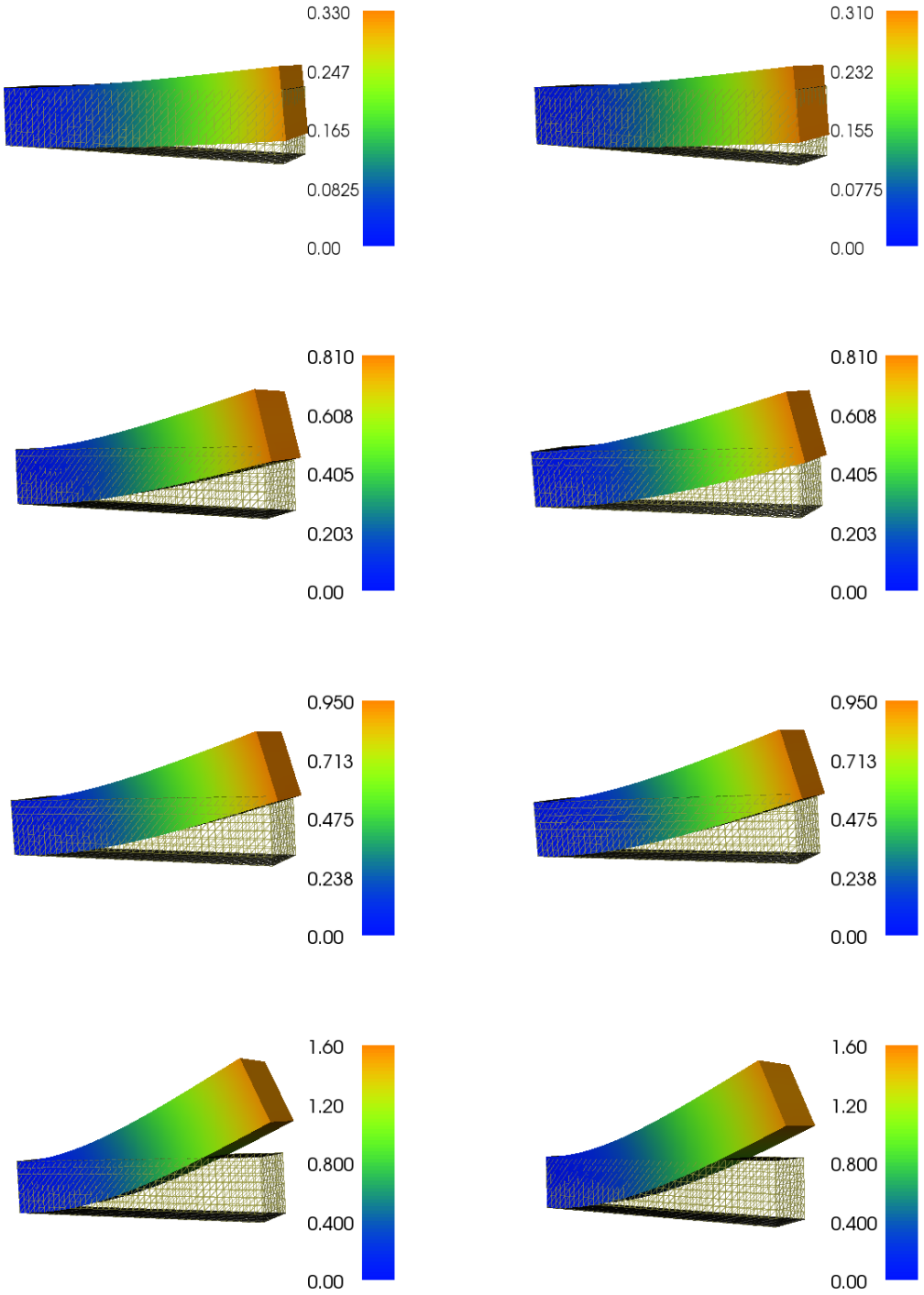


Figure 5.9: Position of the elastic beam at time $t_{end} = 0.25$, $t_{end} = 0.5$, $t_{end} = 0.75$ and $t_{end} = 1.0$. Left plot: Result calculated by using time step $\Delta t = 0.01$. Right plot: Result calculated by using time step $\Delta t = 0.005$.

$\sigma$ satisfies

$$\frac{d}{d\varepsilon}|_{\varepsilon=0} I(\varphi + \varepsilon\phi) = (\sigma(\varphi), \nabla\phi),$$

Thus, simple calculation shows that

$$\sigma(\varphi) = 2\mu FE + \lambda(trE)F,$$

Our elastodynamics problem can thus be presented as

$$\begin{aligned}
&\frac{\partial^2\varphi}{\partial t^2} = \nabla\cdot\sigma \qquad in\,\Omega\times I\\
&\sigma(\varphi) = 4.6\times 10^4 FE + 1.05\times 10^5 (trE)F\\
&u(x,0) = 0 \qquad in\,\Omega\\
&v(x,0) = 0 \qquad in\,\Omega\\
&u = 0 \qquad on\,\Gamma_1\times I\\
&(n\cdot\sigma) = (0,0,200t) \qquad on\,\Gamma_2\times I\\
&(n\cdot\sigma) = (0,0,0) \qquad on\,\Gamma_3\times I
\end{aligned} \tag{5.6}$$

where $\Gamma_1$ is the facet where $x_1 = 0$, $\Gamma_2$ is $x_1 = 5$, $\Gamma_3$ is the rest boundary, $u = \varphi - x$, $F = \nabla\varphi$ and $E = \frac{1}{2}(F^TF - I)$.

When numerically solving a nonlinear problem, there is one more thing we need to pay attention to which is linearization. In this nonlinear elastodynamics problem, after the application of Finite Element Method, we can have the following equation,

$$(T, \nabla\psi_j)_{\Gamma_2} - (\sigma, \nabla\psi_j) + (G, \psi_j) = (\ddot{\varphi}_h, \psi_j) \qquad \forall\psi_j \in V_h, t\in I, \tag{5.7}$$

Then, directly applying the second scheme to above equation (5.7), we can have a collection at equations for every time stepping, for each row $j$,

$$(T_{n+1}, \nabla\psi_j)_{\Gamma_2} - (\sigma(\frac{\varphi_h^{n+2} + \varphi_h^n}{2}), \nabla\psi_j) + (G_{n+1}, \psi_j) = (\frac{\varphi_h^{n+2} - 2\varphi_h^{n+1} + \varphi_h^n}{\Delta t^2}, \psi_j),$$

where $\varphi_h^{n+2} = \sum_{i=1}^{n} \xi_i^{n+2}\psi_i(x)$.

This equation can be simplified as

$$L_j(\varphi_h^{n+2}) = 0,$$

where

$$L_j(\varphi_h^{n+2}) = (T_{n+1}, \nabla\psi_j)_{\Gamma_2} - (\sigma(\frac{\varphi_h^{n+2} + \varphi_h^n}{2}), \nabla\psi_j) + (G_{n+1}, \psi_j) - (\frac{\varphi_h^{n+2} - 2\varphi_h^{n+1} + \varphi_h^n}{\Delta t^2}, \psi_j),$$

This equation can be solved by Newton method,

$$\varphi_h^{n+2} = \varphi_h^{n+1} - (\nabla L(\varphi_h^{n+1}))^{-1} L(\varphi_h^{n+1}),$$

where $L = (L_1, L_2, \cdots)^T$.

To generate the matrix $\nabla L(\varphi_h^{n+1})$, we can use variation instead of direct calculation. By simple calculation, it is easy to verify the following equation holds

$$\frac{d}{d\varepsilon}|_{\varepsilon=0} L_j(\varphi_h^{n+1} + \varepsilon\psi) = (\nabla L_j(\varphi_h^{n+1}))\psi,$$

By choosing $\psi = e_k = (0, \cdots, 0, 1, 0, \cdots, 0)^T$, we can easily recover the $(j, k)$ component of $\nabla L(\varphi_h^{n+1})$.

We still use FEniCS to solve this nonlinear problem (5.6) and the package used for solving is coded to use Newton method as discussed above. To compare with the linear elastic material, we also draw out the position of the elastic beam at time $t = 0.25$, $t = 0.5$ and $t = 0.75$ in Figure 5.10.

Positions of both linear and nonlinear elastic beams are very similar, thus, to have a clearer comparison, we listed the displacement along $z$ axis at point $(5, 1, 1)$ for these two beams in Table 5.5. From the the differences' value in Tale 5.5, we can tell that, even though at the very beginning, displacements of linear and nonlinear beams are alike, but as time increases, their difference is also increasing.

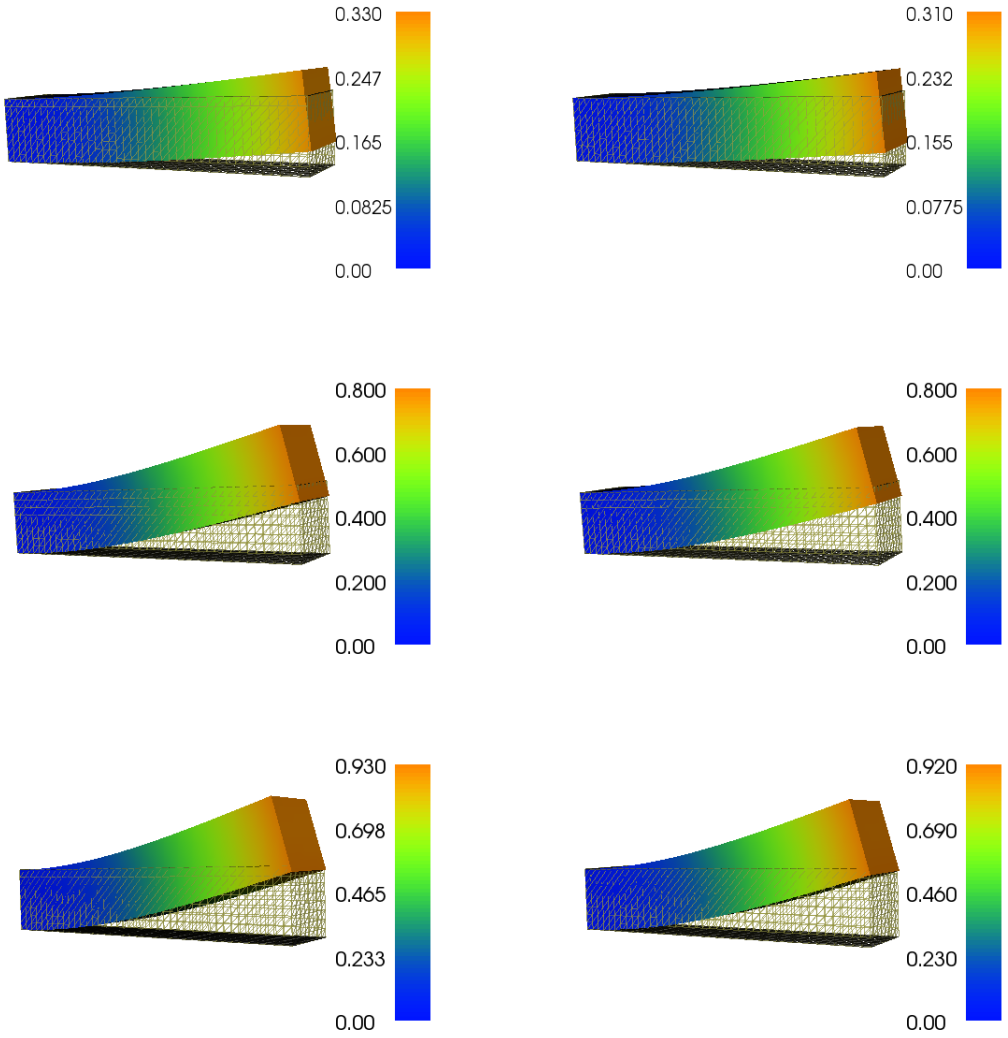


Figure 5.10: Position of the nonlinear elastic beam at time $t_{end} = 0.25$, $t_{end} = 0.5$ and $t_{end} = 0.75$. Left plot: Result calculated by using time step $\Delta t = 0.01$. Right plot: Result calculated by using time step $\Delta t = 0.005$.

Table 5.5: Displacement along $z$ axis at point $(5, 1, 1)$ for both linear and nonlinear elastic beams which are computational results of problem (5.5) and problem (5.6), calculated by different time step at time $t = 0.25$, $t = 0.5$ and $t = 0.75$. $\Delta t = 0.01$.The difference is calculated as the displacement of linear beam minus the one of nonlinear beam.

| | $t = 0.25$ | $t = 0.5$ | $t = 0.75$ |
|---|---|---|---|
| linear ($\Delta t$) | 0.31681757 | 0.79794951 | 0.93599893 |
| linear ($\Delta t/2$) | 0.30512216 | 0.79458233 | 0.93006166 |
| nonlinear ($\Delta t$) | 0.31408305 | 0.77090623 | 0.88662774 |
| nonlinear ($\Delta t/2$) | 0.3025707 | 0.76789883 | 0.88134227 |
| difference($\Delta t$) | 0.0027 | 0.0270 | 0.0494 |
| difference ($\Delta t/2$) | 0.0026 | 0.0267 | 0.0487 |

In FEniCS, there are several packages designed to especially solve nonlinear elastic problems like CBC.Twist which is a DOLFIN module written in UFL syntax. With the help of these packages, things like defining constitutive relationships for different materials, describing boundary conditions can be further simplified. More packages can be found in [Logg, Mardal, and Wells (2012)].

At last, we would like to change the condition to be more practical. Still use the beam in above two problems. Now consider the linear elastic beam in the gravitational field with one of its end, the left facet, fixed to the wall and the rest part free to move. Thus there is a body force $G = (0, 0, -9.8)$. Assume the traction force to be $T = (0, 0, -5)$, density $\rho = 10$ and lame constant $\mu = 0.23 \times 10^5$, $\lambda = 0.105 \times 10^6$. The statement of this problem is very similar to above two, thus, we will not restate it again. Figure 5.11 displays the position of this linear elastic beam at different time and Table 5.6 gives the displacement along $z$ axis at point $(5, 1, 1)$. Figure 5.11 clearly displays the full cycle of the movement of the elastic beam in the gravitational field. At time $t = 2.25$ the elastic beam almost bounces back to its initial position.

Table 5.6: Displacement along $z$ axis at point $(5, 1, 1)$ for the linear elastic beams in gravitational field, calculated by different time step at time $t = 0.5$, $t = 1.0$, $t = 2.0$ and $t = 2.25$. $\Delta t = 0.01$.

| | $t = 0.5$ | $t = 1.0$ | $t = 2.0$ | $t = 2.25$ |
|---|---|---|---|---|
| $dt = \Delta t$ | -0.69212734 | -2.08849109 | -2.92580695 | -0.69574282 |
| $dt = \Delta t/2$ | -0.63320064 | -2.04031687 | -3.00177963 | -0.67443876 |

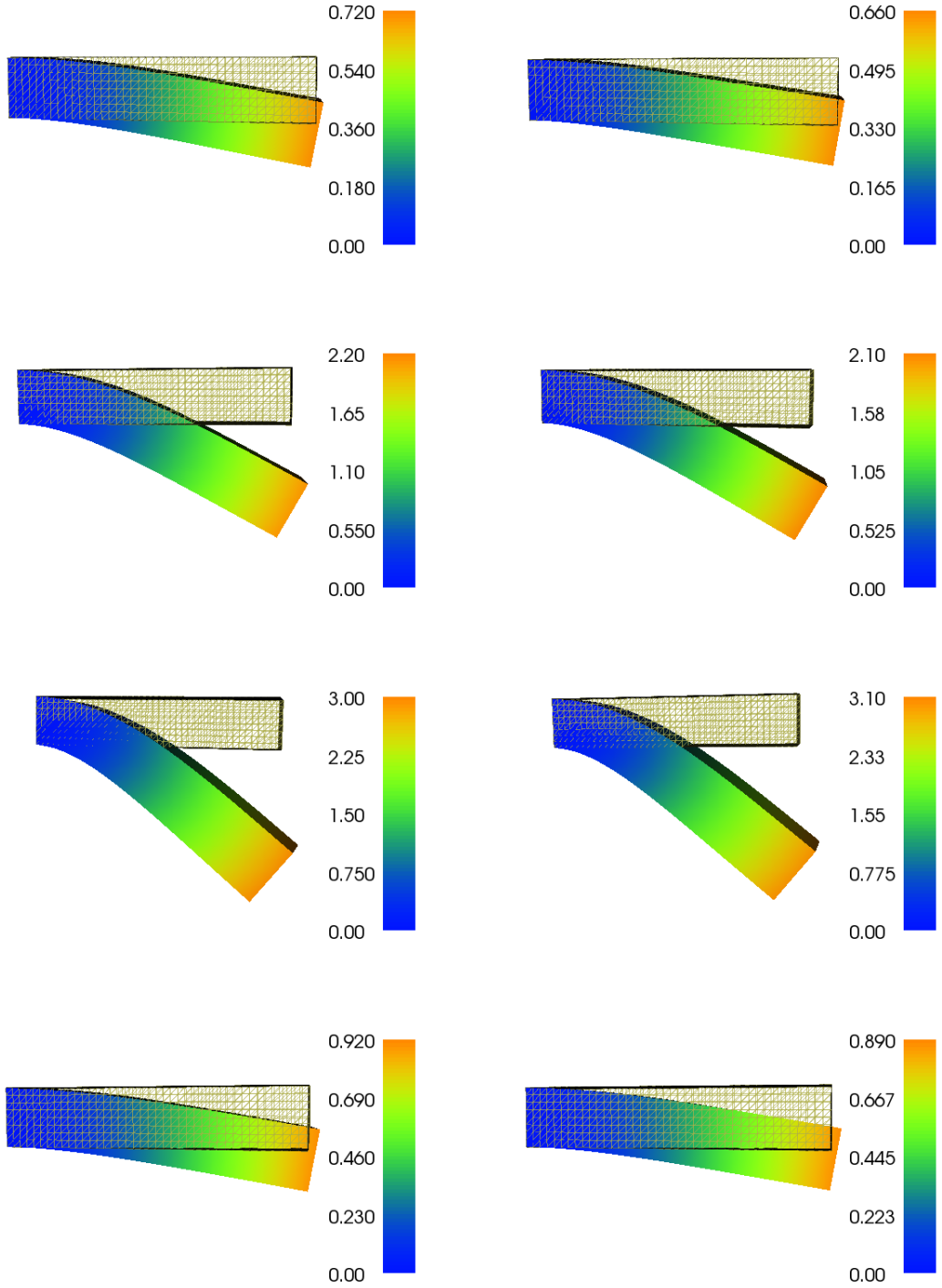


Figure 5.11: Position of the linear elastic beam in the gravitational field at time $t_{end} = 0.5$, $t_{end} = 1.0$, $t_{end} = 2.0$ and $t_{end} = 2.25$. Left plot: Result calculated by using time step $\Delta t = 0.01$. Right plot: Result calculated by using time step $\Delta t = 0.005$.

In conclusion, in this thesis, we numerically test the first and the second schemes to be second-order accurate. And compare these two methods with one of the most frequently used scheme, Newmark Algorithm by their numerical performance of solving ODE problems. Since the numerical result of Newmark scheme depends heavily on the choice of one of the parameter $\beta$, wise choice of $\beta$ can make Newmark performs as well as our two new schemes. However, in general, our first and second schemes have their advantages and convenience in practical application.

# Bibliography

Alavala, C. R., 2008, *Finite element methods: Basic concepts and applications*. PHI Learning Pvt. Ltd.

Alnæs, M. S., J. Hake, R. C. Kirby, H. P. Langtangen, A. Logg, and G. N. Wells, 2011, "The FEniCS Manual," .

Argyris, J. H., and S. Kelsey, 1960, *Energy theorems and structural analysis*, vol. 960. Springer.

Babuska, I., F. Ihlenburg, T. Strouboulis, and S. Gangaraj, 1997, "A Posteriori Error Estimation for Finite Element Solutions of Helmholtz'Equation-Part II: Estimation of the Pollution Error," *International Journal for Numerical Methods in Engineering*, 40(21), 3883–3900.

Babuška, I., T. Strouboulis, S. Gangaraj, and C. Upadhyay, 1997, "Pollution error in the h-version of the finite element method and the local quality of the recovered derivatives," *Computer Methods in Applied Mechanics and Engineering*, 140(1), 1–37.

Bazzi, G., and E. Anderheggen, 1982, "The $\rho$-family of algorithms for time-step integration with improved numerical dissipation," *Earthquake Engineering & Structural Dynamics*, 10(4), 537–550.

Brenner, S., and R. Scott, 2007, *The mathematical theory of finite element methods*, vol. 15. Springer Science & Business Media.

Chung, J., and G. Hulbert, 1993, "A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-$\alpha$ method," *Journal of applied mechanics*, 60(2), 371–375.

Clough, R., 1960, *The Finite Element Method in Plane Stress Analysis*. American Society of Civil Engineers.

Courant, R., et al., 1943, "Variational methods for the solution of problems of equilibrium and vibrations," *Bull. Amer. Math. Soc*, 49(1), 1–23.

Dauksher, W., and A. Emery, 2000, "The solution of elastostatic and elastodynamic problems with Chebyshev spectral finite elements," *Computer methods in applied mechanics and engineering*, 188(1), 217–233.

Dominguez, J., 1993, *Boundary elements in dynamics*. Wit Press.

Golub, G. H., and C. F. Van Loan, 2012, *Matrix computations*, vol. 3. JHU Press.

Guddati, M. N., and B. Yue, 2004, "Modified integration rules for reducing dispersion error in finite element methods," *Computer methods in applied mechanics and engineering*, 193(3), 275–287.

Hilber, H. M., T. J. Hughes, and R. L. Taylor, 1977, "Improved numerical dissipation for time integration algorithms in structural dynamics," *Earthquake Engineering & Structural Dynamics*, 5(3), 283–292.

Hoff, C., and P. Pahl, 1988a, "Development of an implicit method with numerical dissipation from a generalized single-step algorithm for structural dynamics," *Computer Methods in Applied Mechanics and Engineering*, 67(3), 367–385.

——— , 1988b, "Practical performance of the $\theta 1$-method and comparison with other dissipative algorithms in structural dynamics," *Computer Methods in Applied Mechanics and Engineering*, 67(1), 87–110.

Hughes, T. J., 2012, *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation.

Hughes, T. J., and G. M. Hulbert, 1988, "Space-time finite element methods for elastodynamics: formulations and error estimates," *Computer methods in applied mechanics and engineering*, 66(3), 339–363.

Idesman, A., and D. Pham, 2014, "Finite element modeling of linear elastodynamics problems with explicit time-integration methods and linear elements with the reduced dispersion error," *Computer Methods in Applied Mechanics and Engineering*, 271, 86–108.

Komatitsch, D., and J.-P. Vilotte, 1998, "The spectral element method: an efficient tool to simulate the seismic response of 2D and 3D geological structures," *Bulletin of the seismological society of America*, 88(2), 368–392.

Liu, J., 2015, "A second-order stable explicit interface advancing scheme for FSI with both rigid and elastic structures and its application to fish swimming simulations," *Computers & Fluids*, 118, 274–292.

Liu, M., and G. Liu, 2010, "Smoothed particle hydrodynamics (SPH): an overview and recent developments," *Archives of computational methods in engineering*, 17(1), 25–76.

Logg, A., K.-A. Mardal, and G. Wells, 2012, *Automated solution of differential*

*equations by the finite element method: The FEniCS book*, vol. 84. Springer Science & Business Media.

Mack, M. G., 1991, *A three-dimensional boundary element method for elastodynamics.*

Marfurt, K. J., 1984, "Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations," *Geophysics*, 49(5), 533–549.

Norrie, D. H., and G. De Vries, 2014, *The finite element method: fundamentals and applications*. Academic Press.

Oden, J. T., 1972, "Finite elements of nonlinear continua," .

Strang, G., and G. J. Fix, 1973, *An analysis of the finite element method*, vol. 212. Prentice-Hall Englewood Cliffs, NJ.

Turner, M. J., R. W. Clough, H. C. Martin, and L. J. Topp, 1956, "Stiffness and Deflection Analysis of Complex Structures," *Journal of Aeronautical Science*, 23(9), 805–823.

Wilson, E. L., 1968, "A computer program for the dynamic stress analysis of underground structures," working paper, DTIC Document.

Wood, W., M. Bossak, and O. Zienkiewicz, 1980, "An alpha modification of Newmark's method," *International Journal for Numerical Methods in Engineering*, 15(10), 1562–1566.

Zienkiewicz, O., 1973, "Finite Elements, the Background Story," *Whiteman (73)*, pp. 1–35.

Zienkiewicz, O. C., and R. L. Taylor, 2000, *The finite element method: solid mechanics*, vol. 2. Butterworth-heinemann.

# SECOND ORDER FINITE ELEMENT SCHEMES FOR ELASTODYNAMICS

GONG JIAQI

NATIONAL UNIVERSITY OF SINGAPORE

2016