

COMPARATIVE STUDY ON EXPLICIT INTEGRATION ALGORITHMS FOR
STRUCTURAL DYNAMICS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

DİLARA ÇAKIR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CIVIL ENGINEERING

AUGUST 2022

Approval of the thesis:

**COMPARATIVE STUDY ON EXPLICIT INTEGRATION ALGORITHMS
FOR STRUCTURAL DYNAMICS**

submitted by **DİLARA ÇAKIR** in partial fulfillment of the requirements for the degree of **Master of Science in Civil Engineering, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Erdem Canbay
Head of the Department, **Civil Engineering**

Prof. Dr. Özgür Kurç
Supervisor, **Civil Engineering, METU**

Examining Committee Members:

Prof. Dr. Yalın Arıcı
Civil Engineering, METU

Prof. Dr. Özgür Kurç
Civil Engineering, METU

Prof. Dr. Kağan Tuncay
Civil Engineering, METU

Prof. Dr. Murat Altuğ Erberik
Civil Engineering, METU

Assist. Prof. Dr. Burcu Güldür Erkal
Civil Engineering., Hacettepe University

Date: 25.08.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Last name : Dilara akır

Signature :

ABSTRACT

COMPARATIVE STUDY ON EXPLICIT INTEGRATION ALGORITHMS FOR STRUCTURAL DYNAMICS

Çakır, Dilara
Master of Science, Civil Engineering
Supervisor : Prof. Dr. Özgür Kurç

August 2022, 87 pages

Conventional explicit integration algorithms used to solve structural dynamic problems may require too small time increments to satisfy the stability requirements in the presence of high-frequency modes. The requirement to have a too small time increment can cause extending the solution time above the tolerable limit. In this study, three different explicit integration algorithms found in the literature are compared in terms of stability, accuracy, and run-time. The examined integration methods are a two-step integration algorithm, a mass scaling method, and an unconditionally stable explicit algorithm. The performance of each algorithm has been discussed by implementing them in MATLAB and solving various structural dynamic problems. Obtained results are then compared with the solutions of Newmark's explicit integration algorithm, which is considered the reference solution method.

Keywords: Explicit Integration Algorithms, Structural Dynamics, Time History Analysis

ÖZ

YAPISAL DİNAMİK PROBLEMLERİ İÇİN BELİRTİK ENTEGRASYON ALGORİTMALARI ÜZERİNE KARŞILAŞTIRMALI BİR ÇALIŞMA

Çakır, Dilara
Yüksek Lisans, İnşaat Mühendisliği
Tez Yöneticisi: Prof. Dr. Özgür Kurç

Ağustos 2022, 87 sayfa

Yapısal dinamik problemlerini çözmek için kullanılan geleneksel belirtik entegrasyon algoritmaları, yüksek frekans modlarının varlığında kararlılık gereksinimlerini karşılamak için çok küçük zaman adımları gerektirebilir. Zaman adımlarının çok küçük olması gerekliliği, çözüm süresinin tolere edilebilir sınırın üzerine çıkmasına neden olabilir. Bu çalışmada, literatürde bulunan üç farklı belirtik entegrasyon algoritmaları kararlılık, doğruluk ve çalışma süresi açısından karşılaştırılmıştır. İncelenen yöntemler; iki aşamalı bir entegrasyon algoritması, bir kütle modifikasyon yöntemi ve koşulsuz olarak kararlı bir belirtik algoritmadır. İncelenen her bir algoritmanın başarımı, bu çalışma için MATLAB yazılımı kullanılarak geliştirilmiş bir zaman tanım alanı analiz programında uygulanarak ve çeşitli dinamik problemler çözülerek tartışılmıştır. Elde edilen çözümler, referans çözümler için seçilmiş olan Newmark belirtik entegrasyon algoritması ile karşılaştırılmıştır.

Anahtar Kelimeler: Belirtik Entegrasyon Algoritmaları, Yapısal Dinamik, Zaman Tanım Alanı Çözümü

To My Beloved Family

ACKNOWLEDGMENTS

I wish to express my deepest gratitude to my supervisor Prof. Dr. Özgür Kurç and his guidance, advice, criticism, encouragement, and insight throughout the research.

I am also grateful for all the continuous support, love, encouragement, and understanding I have received from my mother Hatice Tügsüz who inspired me to have a great passion for learning and always be there for me with her kind spirit. I am also thankful for the support and love I have received from my father Ali Rıza Çakır, who instilled in me an enthusiasm for success and always encourage me with endless trust.

I would like to thank all my friends and colleagues for their support during my research study.

TABLE OF CONTENTS

ABSTRACT.....	v
ÖZ.....	vi
ACKNOWLEDGMENTS	viii
TABLE OF CONTENTS.....	ix
LIST OF TABLES	xii
LIST OF FIGURES	xiii
CHAPTERS	
1 INTRODUCTION	1
1.1 Problem Definition.....	1
1.2 Related Work.....	3
1.3 Objective and Scope.....	6
1.4 Thesis Outline	7
2 EXPLICIT INTEGRATION METHODS	9
2.1 Introduction	9
2.2 Time History Analysis	9
2.3 Numerical Integration	11
2.3.1 Newmark’s Explicit Integration Method	12
2.3.2 The Noh-Bathe Method	12
2.3.3 Chang’s Method.....	13
2.4 Stability Limits of the Explicit Integration Methods	14
2.4.1 Newmark’s Explicit Method.....	14
2.4.2 The Noh-Bathe Method	15

2.4.3	Chang's Method	15
3	MASS SCALING	17
3.1	Introduction.....	17
3.2	Mass Scaling	17
3.2.1	The Stabilized Central Difference Method with Mass Scaling (MS)	18
3.2.2	Stability Limit of the Stabilized Central Difference Method with Mass Modification.....	18
4	IMPLEMENTATION	21
4.1	Introduction.....	21
4.2	Time History Analysis	22
4.2.1	Newmark's Explicit Method	24
4.2.2	The Noh-Bathe Method.....	26
4.2.3	Chang's Method	29
4.2.4	The Stabilized Central Difference Method with Mass Scaling (MS)	31
4.2.5	The Noh-Bathe Method with Mass Scaling (MS).....	33
4.3	Finite Element Library	35
4.3.1	2D Frame Element.....	35
4.3.2	Incompatible Membrane Element	36
4.3.3	Brick Element.....	38
5	VERIFICATION PROBLEMS	41
5.1	Introduction.....	41
5.2	SDOF Problem.....	41
5.2.1	Undamped System.....	42
5.2.2	Damped System.....	43

5.3	Frame Under Impulse Loading	45
5.4	Convergence Rate of Algorithms	49
5.5	Summary of Results	50
6	CASE STUDIES	51
6.1	Introduction	51
6.2	Case Study 1: Simple Cantilever Under Vertical Impulse	52
6.3	Case Study 2: Clamped 3D Solid	59
6.4	Case Study 3: Moving Load on Three-Span Road Bridge	70
6.5	Summary of Results	75
7	CONCLUSIONS AND FUTURE WORK	77
7.1	Conclusions	77
7.2	Future Work	78
	REFERENCES	79
	APPENDICES	
A.	Appendix - Displacement Response Plots	81

LIST OF TABLES

TABLES

Table 5.1 Model Information	46
Table 5.2 Stability Requirements	46
Table 5.3 Displacement Results	47
Table 6.1 Discretization of Case Study 1	53
Table 6.2 Stability Requirements of the Integration Methods in Case Study 1	53
Table 6.3 Accuracy Comparison from Case Study 1 – Model 1	56
Table 6.4 Accuracy Comparison from Case Study 1 – Model 2	56
Table 6.5 Solution Time Comparison from Case Study 1 – Model 1	57
Table 6.6 Solution Time Comparison from Case Study 1 – Model 2	58
Table 6.7 Discretization of Case Study 2	60
Table 6.8 Stability Requirements of the Integration Methods in Case Study 2	61
Table 6.9 Accuracy Comparison from Case Study 2 – Model 1	63
Table 6.10 Accuracy Comparison from Case Study 2 – Model 2	64
Table 6.11 Accuracy Comparison from Case Study 2 – Model 3	65
Table 6.12 Solution Time Comparison from Case Study 2 – Model 1	66
Table 6.13 Solution Time Comparison from Case Study 2 – Model 2	67
Table 6.14 Solution Time Comparison from Case Study 2 – Model 3	68
Table 6.15 Discretization of Case Study 3	71
Table 6.16 Stability Requirements of the Integration Methods in Case Study 3	71
Table 6.17 Accuracy Comparison of Algorithms from Case Study 3	73
Table 6.18 Solution Time Comparison from Case Study 3	74
Table 6.19 Summary of Solution Times	75
Table 6.20 Summary of Stability Requirements	76

LIST OF FIGURES

FIGURES

Figure 4.1 Flow Chart for Time History Analysis	23
Figure 4.2 Flow Chart for Newmark's Explicit Method.....	25
Figure 4.3 Flow Chart for the Noh-Bathe Method.....	27
Figure 4.4 Flow Chart for Chang's Method	30
Figure 4.5 Flow Chart for the Stabilized Central Difference Method with Mass Modification.....	32
Figure 4.6 Flow Chart for the Noh-Bathe Method with Mass Scaling	34
Figure 4.7 2-Node Frame Element.....	36
Figure 4.8 4-Node Quadrilateral Membrane.....	37
Figure 4.9 8-Node Brick Element	38
Figure 5.1 Single Degree of Freedom Problem	41
Figure 5.2 Displacement Response of the Undamped SDOF System	43
Figure 5.3 Displacement Response of Damped SDOF System.....	44
Figure 5.4 Frame Under Impulse Loading.....	45
Figure 5.5 Displacement Response Comparison	48
Figure 5.6 Convergence Rate of Algorithms	49
Figure 6.1 Simple Cantilever Under Vertical Impulse	52
Figure 6.2 Clamped 3D Solid under Vertical Impulse.....	59
Figure 6.3: Three-Span Road Bridge Problem.....	70
Figure A.1 Displacement Responses of the Case Study 1 - Model 1	81
Figure A.2 Displacement Responses of the Case Study 1 - Model 2	82
Figure A.3 Displacement Responses of the Case Study 2 - Model 1	83
Figure A.4 Displacement Responses of the Case Study 2 - Model 2	84
Figure A.5 Displacement Responses of the Case Study 2 - Model 3	85
Figure A.6 Displacement Responses of the Case Study 3 – A	86
Figure A.7 Displacement Responses of the Case Study 3 – B	87

CHAPTER 1

INTRODUCTION

1.1 Problem Definition

Numerical integration algorithms are usually utilized to solve the equation of motion for structural dynamics problems. These algorithms are based on dividing the problem into discrete time steps and moving forward with step-by-step direct integration. Such algorithms can be classified as explicit and implicit integration algorithms. If the algorithm needs the state of the structural system at the current and the following time step to find the state of the system at the following time step, it is called an implicit integration algorithm. On the other hand, explicit algorithms utilize the state of the current time step to compute the system's state for the following time step. In a nonlinear problem, the system's state changes as the structure deforms. In other words, the structural properties of a system can only be computed for the known, current time step. Therefore, equilibrium iterations are required to solve nonlinear problems with implicit integration algorithms at every time step as it requires the structural properties for the following time step. As the explicit algorithms use known structural properties of the current time step, it doesn't need additional equilibrium iterations. This property makes explicit integration algorithms computationally very attractive, especially in nonlinear structural dynamics problems.

While using an explicit integration algorithm to solve a dynamic problem, there is a limit on the magnitude of the time increment to obtain a solution. The step must be smaller than a certain limit to satisfy the numerical stability; otherwise, the solution diverges. Due to the conditional stability of the explicit algorithms, the critical time increment to satisfy numerical stability could be so small that obtaining results at an

acceptable solution time may not be possible, especially when the structure contains spurious high-frequency modes. When the low-frequency modes dominate the response, the high-frequency modes are out of concern. For these systems, high-frequency modes are considered spurious high-frequency modes. In order to filter out the spurious high-frequency mode and consequently improve the method's stability, numerical dissipation is commonly applied to explicit integration schemes. However, using larger time increments with improved stability criteria may not always work in favor of accuracy. In fact, the presence of high-frequency modes can severely damage the accuracy of the solution (Noh & Bathe, 2013).

The accuracy of the solution obtained by numerical integration methods cannot be directly guaranteed with any time increment. To have an accuracy of the results at a reasonable level, the rule of thumb for the time increment is $\Delta t = \frac{T_{min}}{10}$, where T_{min} is the minimum period of the system (Song et al., 2022). For the explicit integration methods selecting a time increment for the solution is an optimization problem between the solution time and accuracy within the stability range. The accuracy of the explicit integration methods should also be studied, along with improving the numerical stability limit and the solution time.

The conditional stability drawback of the explicit integration methods, especially in the presence of high-frequency modes, requires a quest for a scheme that promises reasonable solution time without compromising accuracy. The research question herein is which explicit integration approach promises the shortest solution time at the desired accuracy level. Any improvements to increase the numerical stability limit without compromising the accuracy of explicit algorithms would significantly reduce the computational time for solving nonlinear structural dynamics problems.

1.2 Related Work

Using different forms of numerical dissipation to eliminate spurious high-frequency modes is the basis of several methods found in the literature to improve the stability limit for explicit integration algorithms. One of the numerical dissipation techniques studied in literature is modifying the system mass matrix. Adjusting the mass matrix affects the system's natural frequency since it directly influences inertia (Soares & Großholz, 2018). Two main approaches are commonly employed for mass modification. The first one is directly increasing the material density. However, having an artificially high mass density can force a dynamic problem to behave as a quasi-static one (Askes et al., 2011). The second approach to modifying the system mass matrix is scaling the system stiffness matrix and adding it to the mass matrix (Askes et al., 2011). With this method, the mode shapes of the system remain unchanged while the high frequencies are reduced (Macek & Aubert, 1995). If time increment is not considered while scaling the stiffness matrix, the convergence may not be guaranteed (Soares & Großholz, 2018). Accordingly, Soares & Großholz, (2018) proposed an explicit integration scheme in which the stiffness and damping matrices are scaled by considering time increments and added to the system's mass matrix. This integration scheme was nothing but a stabilized version of the central difference method with a modified mass matrix.

The central difference method is one of the most widely used explicit integration methods in the literature because it has the largest time step stability limit among other second-order accurate explicit methods. Since it doesn't include algorithmic dissipation, the solution accuracy can be severely impaired in the presence of high-frequency modes (Noh & Bathe, 2013).

The explicit integration algorithms found in the literature can be classified as single-step, multi-step, and multi-sub-step. Having a multi-step or multi-sub-step algorithm instead of a single-step algorithm can significantly improve the stability limit and numerical accuracy (Li et al., 2021). Multi-step algorithms use the solutions of the

previous few steps in the scheme (Zhang et al., 2022). Two-step and three-step algorithms are contributed to multi-step integration scheme studies by Yang et al. (2020). The proposed two-step integration scheme is based on two previous accelerations. To find the next displacement (u_{n+1}) and velocity (\dot{u}_{n+1}), the employed accelerations are \ddot{u}_n and \ddot{u}_{n-1} . For the three-step scheme, the employed accelerations are, \ddot{u}_n , \ddot{u}_{n-1} , and \ddot{u}_{n-2} . If three previous accelerations are used in the integration scheme instead of two previous accelerations, a more accurate integration method can be obtained (Yang et al., 2020).

In multi-sub-step schemes, the time interval is split into a few sub-steps, and the equation of motion is solved at each sub-step. Multi-sub-step schemes allow broader stability regions compared to single-step methods (Zhang et al., 2022). The Noh-Bathe Method (Noh & Bathe, 2013) and Kim-Lee method (Kim et al., 2018) are examples of multi-sub-step algorithms found in the literature. Both are second-order accurate methods and have a wide stability range of numerical stability limits. The main difference between the Kim-Lee Method and the Noh-Bathe Method is the requirement of computing the initial acceleration vector. The Kim-Lee Method does not require the computation of the initial acceleration vector throughout the entire procedure. On the other hand, to start the integration procedure, additional preparation to compute the initial acceleration vector from the given initial displacement and velocity is required in the Noh-Bathe Method (Kim et al., 2018).

While most of the explicit integration methods found in the literature are conditionally stable, an unconditionally stable explicit integration algorithm is proposed by Chang, (2002). The characteristic equation presented after the stability analysis of this algorithm gives the same characteristic equation as the constant average acceleration method to guarantee that all roots are smaller than one. Consequently, it is unconditionally stable for linear systems (Chang, 2002). However, conducting matrix inversion is necessary for this proposed method. The necessity of the matrix inversion increases the computational efforts of the algorithm. When the performance of the proposed algorithm was tested in nonlinear problems,

it was seen that the method's stability depends on the instantaneous degree of nonlinearity. The instantaneous degree of nonlinearity is defined as the ratio of the system's stiffness in the following time step over the initial stiffness of the system. The algorithm is unconditionally stable for instantaneous stiffness softening systems where the instantaneous degree of nonlinearity is smaller than 1. However, it is conditionally stable for hardening systems where the instantaneous degree of nonlinearity is greater than 1 (Chang, 2010).

The other important aspect of explicit integration algorithms is the order of accuracy. Most integration algorithms are developed based on the Taylor series expansion or the weighted residual method. Accuracy will be compromised when the high-order terms are truncated in the Taylor series expansion as well as the weighted residual method (Yang et al., 2020). Extending the numerical stability limit with certain numerical dissipation method are not adequate if the level of accuracy is not within an acceptable limit. The necessity of higher-order accuracy of an algorithm becomes essential at this point. The solution should be obtained with larger time increments from higher-order accurate algorithms without compromising accuracy (Fung, 2003).

Many studies have provided a wide stability region for the explicit integration algorithms using different approaches like mass scaling, multi-sub step, or matrix inversion. Presented algorithms are usually compared to counterparts developed using similar approaches in these studies. Explicit integration algorithms that are developed with different approaches have not been compared in terms of accuracy and stability limits. On the other hand, according to the author's knowledge, such algorithms are not compared considering the run-time and accuracy. Hence a comparative study is conducted to compare the stability limits, accuracy, and run times of mass scaling, multi-sub-step, and matrix inversion algorithms.

1.3 Objective and Scope

The main objective of this study is to compare different types of existing explicit integration algorithms that propose methods for improving the numerical stability limits. The basis of comparison is computational efficiency and accuracy in structural dynamics problems. For this purpose, a new finite element analysis program is developed to perform time history analysis in the MATLAB environment. Four different explicit integration algorithms are implemented: The Noh-Bathe method, the stabilized CD method with mass scaling, Chang's method, and the Noh-Bathe method with mass scaling. The Noh-Bathe method and mass scaling technique are attractive for this study because they are used in commercial finite element analysis software ADINA and ABAQUS, respectively. The usual characteristic of an explicit algorithm is conditional stability and Chang's method promises to eliminate this main drawback of the explicit integration method. Therefore, Chang's method is included in the performance comparison to see how much is gained with unconditional stability and how much is lost with matrix inversion, in terms of computational cost.

The performance of these four explicit integration methods is compared in terms of accuracy, stability, and run-time. While only linear problems are considered, the algorithms are implemented so that they also work for nonlinear problems. In other words, computational simplifications that could be done for linear problems are excluded. This way, the conclusions of this study would be indicative of the performance of these algorithms for nonlinear problems. The performance and robustness of these methods are tested by solving various dynamic problems modeled with different finite elements under various time-dependent loading conditions.

For the accuracy comparison, errors in results are computed from each examined method with different time increments with respect to the reference accurate solution obtained from Newmark's explicit integration method. For the reasonable accuracy

of the reference solution, the selected time increment for the reference solution is taken as smaller than the rule of thumb value, i.e., $\Delta t = \frac{T_{min}}{10}$.

For the solution time comparison, errors computed for accuracy comparison are used to list the required time increment for each algorithm to reach a certain accuracy level. The solution time of the algorithms is compared while they show the same accuracy level.

1.4 Thesis Outline

The outline of this work is given as follows. After this introduction, the theory of time history analysis and the presentation of formulas of the explicit integration algorithms is presented in Chapter 2. Brief information regarding mass scaling and a stabilized central difference method with mass scaling is presented in Chapter 3. Then the implementation of the time history analysis with explicit integration algorithms and various finite elements into the platform developed in the MATLAB environment is explained in Chapter 4. In Chapter 5, verification problems are solved with the developed program to verify the implementation of the explicit integration algorithms. The obtained results are compared with the analytical solution. Case studies follow the verification problems in Chapter 6. Benchmark problems are solved with the four explicit integration algorithms: the Noh-Bathe Method, the stabilized CD method with mass scaling, Chang's Method, and the Noh-Bathe method with mass scaling. Results from these four explicit integration algorithms are compared in terms of stability, accuracy, and run-time. Finally, in Chapter 7, results obtained from these case studies are discussed, and possible future work is presented.

CHAPTER 2

EXPLICIT INTEGRATION METHODS

2.1 Introduction

This chapter presents brief information on the time history analysis and the corresponding explicit integration algorithms. In addition to the explicit Newmark integration method, the details of the Noh-Bathe Method, and Chang's Method are presented. Numerical stability analyses and discussions are also presented for each integration method.

2.2 Time History Analysis

Time history analysis is performed to see the structure's response in time under time-dependent loading. The equation of motion for the solution of the dynamic response is derived from the total potential energy functional written considering the virtual displacement method as given in Equation 2-1.

$$\delta \Pi = \delta W_{int} - \delta W_{ext} \quad (2-1)$$

Considering an element with a volume of V and surface area of S , internal and external energy can be written as in Equation 2-2 and Equation 2-3, respectively.

$$\delta W_{int} = \int (\{\delta \mathbf{u}\}^T \rho \{\ddot{\mathbf{u}}\} + \{\delta \mathbf{u}\}^T c \{\dot{\mathbf{u}}\} + \{\delta \boldsymbol{\epsilon}\}^T \{\boldsymbol{\sigma}\}) dV \quad (2-2)$$

$$\delta W_{ext} = \int \delta \mathbf{u}^T \mathbf{F} dV + \int \{\delta \mathbf{u}\}^T \{\boldsymbol{\Phi}\} dS + \sum_{i=1}^n \{\delta \mathbf{u}\}_i^T \mathbf{p}_i \quad (2-3)$$

In Equation 2-2, $\rho, c, \boldsymbol{\sigma}, \delta \mathbf{u}, \delta \boldsymbol{\epsilon}$ indicate mass density, damping parameter, stress vector, virtual displacement, and corresponding virtual strain vectors, respectively. In Equation 2-3, $\mathbf{F}, \boldsymbol{\Phi}, \mathbf{p}$ indicate, body forces, surface traction, and concentrated load vectors, respectively.

Based on the finite element discretization, expressions in Equation 2-4 are obtained.

$$\{\mathbf{u}\} = [\mathbf{N}]\{\mathbf{d}\} \quad \dot{\mathbf{u}} = [\mathbf{N}]\dot{\mathbf{d}} \quad \{\ddot{\mathbf{u}}\} = [\mathbf{N}]\{\ddot{\mathbf{d}}\} \quad \boldsymbol{\epsilon} = [\mathbf{B}]\mathbf{d} \quad (2-4)$$

In the above expressions, \mathbf{N} and \mathbf{d} denote shape functions and nodal degree of freedoms, respectively.

Inserting expressions in Equation 2-4 into total potential energy functional will yield Equation 2-5.

$$\delta \mathbf{d}^T \left(\int \rho [\mathbf{N}]^T [\mathbf{N}] dV \{\ddot{\mathbf{d}}\} + \int c [\mathbf{N}]^T [\mathbf{N}] dV \dot{\mathbf{d}} + \int [\mathbf{B}]^T \{\boldsymbol{\sigma}\} dV - \int [\mathbf{N}]^T \{\mathbf{F}\} dV - \int [\mathbf{N}]^T \{\boldsymbol{\Phi}\} dS - \sum_{i=1}^n \{\mathbf{p}\}_i \right) = 0 \quad (2-5)$$

For an arbitrary virtual displacement, $\delta \mathbf{d}$, Equation 2-5 yields Equation 2-6.

$$\int \rho [\mathbf{N}]^T [\mathbf{N}] dV \{\ddot{\mathbf{d}}\} + \int c [\mathbf{N}]^T [\mathbf{N}] dV \dot{\mathbf{d}} + \int [\mathbf{B}]^T \{\boldsymbol{\sigma}\} dV - \int [\mathbf{N}]^T \{\mathbf{F}\} dV - \int [\mathbf{N}]^T \{\boldsymbol{\Phi}\} dS - \sum_{i=1}^n \{\mathbf{p}\}_i = 0 \quad (2-6)$$

The more compact form of Equation 2-6 can be written in terms of element consistent mass and damping matrices are presented in Equation 2-7,

$$[\mathbf{m}]\{\ddot{\mathbf{d}}\} + [\mathbf{c}]\{\dot{\mathbf{d}}\} + \{\mathbf{r}^{int}\} = \{\mathbf{r}^{ext}\} \quad (2-7)$$

Where,

$$[\mathbf{m}] = \int \rho [\mathbf{N}]^T [\mathbf{N}] dV \quad (2-8)$$

$$[\mathbf{c}] = \int c [\mathbf{N}]^T [\mathbf{N}] dV \quad (2-9)$$

$$\{\mathbf{r}^{int}\} = \int [\mathbf{B}]^T \{\boldsymbol{\sigma}\} dV \quad (2-10)$$

$$\{\mathbf{r}^{ext}\} = \int [\mathbf{N}]^T \{\mathbf{F}\} dV + \int [\mathbf{N}]^T \{\boldsymbol{\Phi}\} dS + \sum_{i=1}^n \{\mathbf{p}\}_i \quad (2-11)$$

For structures having multi degrees of freedom, each element's mass and damping matrices are assembled yielding $[\mathbf{M}]$ and $[\mathbf{C}]$, respectively. Then the equation of motion in global form becomes:

$$[\mathbf{M}]\{\ddot{\mathbf{D}}\} + [\mathbf{C}]\{\dot{\mathbf{D}}\} + \{\mathbf{R}^{int}\} = \{\mathbf{R}^{ext}\} \quad (2-12)$$

2.3 Numerical Integration

The form of the equation of motion presented in Equation 2-12 can be solved using numerical direct integration algorithms. The response history is calculated using step-by-step integration in time at the discrete points divided by time increments, Δt . Numerical methods that are used to solve dynamic problems can be explicit or implicit. Explicit methods calculate the state of the system at a later time from the state of the system at the current time, whereas implicit methods calculate the state of the system at a later time by solving an equation involving both the current state of the system and the later one. In the case of a nonlinear problem, since the system's stiffness may change from one time step to the next, the tangent stiffness is unknown at the later state of a system. Therefore, equilibrium iterations are required for the solution at each time step. In terms of computational efficiency, explicit integration algorithms can be considered well-suited for nonlinear problems since the unknown state of the system is calculated from the system's current state with the known tangent stiffness. Hence, explicit integration algorithms do not require equilibrium iterations. However, one drawback of explicit algorithms is their stability condition.

Most of the explicit integration algorithms are conditionally stable. In other words, there is a limit on the magnitude of the time increment. The solution diverges for time increments that are larger than the limit value. Using smaller time increments for a stable solution may significantly increase the analysis time. Hence, the need for increasing the speed of analysis leads to improving the stability limit on the time increment. In the literature, there are different approaches for increasing the stability limit for explicit integration algorithms. Within the scope of this study, two different approaches are implemented and compared: the Noh-Bathe method (Noh & Bathe, 2013), and Chang's method (Chang, 2002). First, the basic explicit integration method, Newmark's explicit integration algorithm is discussed.

2.3.1 Newmark's Explicit Integration Method

Newmark's explicit integration method is based on the following equations:

$$\mathbf{U}_{t+1} = \mathbf{U}_t + \dot{\mathbf{U}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{U}}_t \Delta t^2 \quad (2-13)$$

$$\dot{\mathbf{U}}_{t+1} = \dot{\mathbf{U}}_t + \frac{1}{2} \ddot{\mathbf{U}}_t \Delta t + \frac{1}{2} \ddot{\mathbf{U}}_{t+1} \Delta t \quad (2-14)$$

$$\ddot{\mathbf{U}}_{t+1} \left(\mathbf{M} + \frac{\Delta t}{2} \mathbf{C} \right) = \mathbf{P}_{t+1} - (\mathbf{C} \dot{\mathbf{U}}_t + \mathbf{K} \mathbf{U}_t + \mathbf{K} \dot{\mathbf{U}}_t \Delta t) - \left(\frac{\Delta t}{2} \mathbf{C} \ddot{\mathbf{U}}_t + \frac{\Delta t^2}{2} \mathbf{K} \ddot{\mathbf{U}}_t \right) \quad (2-15)$$

where \mathbf{K} , \mathbf{M} , and \mathbf{C} are stiffness, mass, and damping matrices; \mathbf{P}_{t+1} external force vector, \mathbf{U} , $\dot{\mathbf{U}}$, and $\ddot{\mathbf{U}}$ are displacement, velocity, and acceleration vectors, respectively.

2.3.2 The Noh-Bathe Method

A two-sub-step explicit time integration scheme has been proposed to solve the structural system's displacement, velocity, and acceleration response by Noh & Bathe (2013). A parameter, p , has been introduced to identify the time step sizes for the first and second sub-step. The system response is calculated considering a time interval, Δt consists of two sub-steps. The time increment for the first and second sub-step is $p\Delta t$ and $(1-p)\Delta t$, respectively, where $p \in (0,1)$ (Noh & Bathe, 2013).

The proposed integration scheme by Noh & Bathe (2013) to solve Equation 2-12 is as follows,

First sub-step,

$$[\mathbf{M}]^{t+p\Delta t} \{ \ddot{\mathbf{U}} \} + [\mathbf{C}]^{t+p\Delta t} \{ \tilde{\dot{\mathbf{U}}} \} + [\mathbf{K}]^{t+p\Delta t} \{ \mathbf{U} \} = {}^{t+p\Delta t} \{ \hat{\mathbf{R}} \} \quad (2-16)$$

$${}^{t+p\Delta t} \{ \mathbf{U} \} = {}^t \{ \mathbf{U} \} + [p\Delta t] {}^t \{ \dot{\mathbf{U}} \} + \frac{1}{2} [p\Delta t]^2 {}^t \{ \ddot{\mathbf{U}} \} \quad (2-17)$$

$${}^{t+p\Delta t}\{\widehat{\mathbf{U}}\} = {}^t\{\dot{\mathbf{U}}\} + \frac{1}{2}[p\Delta t] {}^t\{\ddot{\mathbf{U}}\} \quad (2-18)$$

$${}^{t+p\Delta t}\{\dot{\mathbf{U}}\} = {}^{t+p\Delta t}\{\widehat{\mathbf{U}}\} + \frac{1}{2}[p\Delta t]{}^{t+p\Delta t}\{\ddot{\mathbf{U}}\} \quad (2-19)$$

Second sub-step,

$$[\mathbf{M}]^{t+\Delta t}\{\ddot{\mathbf{U}}\} + [\mathbf{C}]^{t+\Delta t}\{\tilde{\mathbf{U}}\} + [\mathbf{K}]^{t+\Delta t}\{\mathbf{U}\} = {}^{t+\Delta t}\mathbf{R} \quad (2-20)$$

$${}^{t+\Delta t}\{\mathbf{U}\} = {}^{t+p\Delta t}\{\mathbf{U}\} + [(1-p)\Delta t] {}^{t+p\Delta t}\{\dot{\mathbf{U}}\} + \frac{1}{2}[(1-p)\Delta t]^2{}^{t+p\Delta t}\{\ddot{\mathbf{U}}\} \quad (2-21)$$

$${}^{t+\Delta t}\{\widehat{\mathbf{U}}\} = {}^{t+p\Delta t}\{\dot{\mathbf{U}}\} + \frac{1}{2}[(1-p)\Delta t]{}^{t+p\Delta t}\{\ddot{\mathbf{U}}\} \quad (2-22)$$

$${}^{t+\Delta t}\{\dot{\mathbf{U}}\} = {}^{t+\Delta t}\{\widehat{\mathbf{U}}\} + [(1-p)\Delta t](q_0 {}^t\{\ddot{\mathbf{U}}\} + q_1 {}^{t+p\Delta t}\{\ddot{\mathbf{U}}\} + q_2 {}^{t+\Delta t}\{\ddot{\mathbf{U}}\}) \quad (2-23)$$

where,

$${}^{t+p\Delta t}\{\tilde{\mathbf{U}}\} = (1-s){}^{t+p\Delta t}\{\widehat{\mathbf{U}}\} + s {}^t\{\dot{\mathbf{U}}\} \quad (2-24)$$

$${}^{t+\Delta t}\{\tilde{\mathbf{U}}\} = (1-s){}^{t+\Delta t}\{\widehat{\mathbf{U}}\} + s {}^{t+p\Delta t}\{\dot{\mathbf{U}}\} \quad (2-25)$$

\mathbf{M} , \mathbf{K} , and \mathbf{C} are mass, stiffness, and damping matrices, respectively and \mathbf{R} represents the external force vector. Also, \mathbf{U} , $\dot{\mathbf{U}}$, and $\ddot{\mathbf{U}}$ are displacement, velocity, and acceleration vectors, respectively. q_0, q_1, q_2 and s parameters are dependent on the selected p value. Selected p value and dependent parameters affect the stability and accuracy characteristics of the method (Noh & Bathe, 2013).

2.3.3 Chang's Method

Chang's method is an unconditionally stable explicit algorithm (Chang, 2002). In other words, there is no stability limit for the magnitude of the time increment.

For a multi-degree of freedom system, the formulation of the proposed algorithm by Chang (2002) can be expressed as,

$$[\mathbf{M}]\{\ddot{\mathbf{U}}\}_{t+\Delta t} + [\mathbf{C}]\{\dot{\mathbf{U}}\}_{t+\Delta t} + \{\mathbf{r}\}_{t+\Delta t} = \{\mathbf{f}\}_{t+\Delta t} \quad (2-26)$$

$$\{\mathbf{U}\}_{t+\Delta t} = \{\mathbf{U}\}_t + \beta_1(\Delta t)\{\dot{\mathbf{U}}\}_t + \beta_2(\Delta t)^2\{\mathbf{U}\}_t \quad (2-27)$$

$$\{\dot{\mathbf{U}}\}_{t+\Delta t} = \{\dot{\mathbf{U}}\}_t + \frac{1}{2}(\Delta t) (\{\ddot{\mathbf{U}}\}_t + \{\ddot{\mathbf{U}}\}_{t+\Delta t}) \quad (2-28)$$

where \mathbf{M} and \mathbf{C} are mass and damping matrices; \mathbf{r} and \mathbf{f} are internal force vector and external force vector, \mathbf{U} , $\dot{\mathbf{U}}$, and $\ddot{\mathbf{U}}$ are displacement, velocity, and acceleration vectors, respectively.

β_1 and β_2 coefficients are defined as,

$$\beta_1 = \left[\mathbf{I} + \frac{1}{2}(\Delta t)\mathbf{M}^{-1}\mathbf{C} + \frac{1}{4}(\Delta t)^2\mathbf{M}^{-1}\mathbf{K}_0 \right]^{-1} \times \left[\mathbf{I} + \frac{1}{2}(\Delta t)\mathbf{M}^{-1}\mathbf{C} \right] \quad (2-29)$$

$$\beta_2 = \left(\frac{1}{2} \right) \left[\mathbf{I} + \frac{1}{2}(\Delta t)\mathbf{M}^{-1}\mathbf{C} + \frac{1}{4}(\Delta t)^2\mathbf{M}^{-1}\mathbf{K}_0 \right]^{-1} \quad (2-30)$$

where \mathbf{I} is the identity matrix and \mathbf{K}_0 is the initial system stiffness matrix.

2.4 Stability Limits of the Explicit Integration Methods

2.4.1 Newmark's Explicit Method

Newmark's explicit integration method is a conditionally stable explicit algorithm. The stability condition for Newmark's method is given as follows,

$$\Delta t < \frac{2}{\omega_{n,max}} \quad (2-31)$$

where $\omega_{n,max}$ is the maximum natural frequency of the system.

2.4.2 The Noh-Bathe Method

The maximum stability limit of the Noh-Bathe method is presented in Equation 2-32 (Noh & Bathe, 2013). In this equation Ω_s indicates $\omega_0 \Delta t_{cr}$ where Δt_{cr} is the critical time increment for numerical stability and ω_0 is the maximum natural frequency of the system.

$$\Omega_s^2 = \frac{1}{\gamma p(1-p)} \quad (2-32)$$

where,

$$\gamma = 0.25 - 0.5(1-p)q_1 \quad (2-33)$$

$$q_1 = \frac{1-2p}{2p(1-p)} \quad (2-34)$$

$$0.5 \leq p \leq 2 - \sqrt{2} \quad (2-35)$$

Noh, G, & Bathe, K. J., indicates that p is related to the amount of the numerical dissipation in the high frequencies when $p = 0.5$, no numerical dissipation occurs and when $p = 2 - \sqrt{2}$, the maximum numerical dissipation occurs. The maximum critical time increment for the stability is obtained when p value equals 0.5. By increasing p value within a defined range, a convergence in the solution can be obtained just below the stability limit, and the accuracy of the solution will be improved. The suggested value for p is 0.54 (Noh & Bathe, 2013).

2.4.3 Chang's Method

The characteristic equation of the proposed algorithm is given as follows (Chang, 2002):

$$\lambda \left(\lambda^2 - \frac{2-0.5\Omega^2}{1+0.25\Omega^2} \lambda + 1 \right) = 0 \quad (2-36)$$

Since this characteristic equation is the same as the characteristic equation of the average acceleration method, Chang's Method has the numerical properties of the average acceleration method, and consequently, the proposed algorithm by Chang is unconditionally stable.

CHAPTER 3

MASS SCALING

3.1 Introduction

This chapter presents brief information on mass scaling and the corresponding explicit integration algorithm, the stabilized CD method with mass scaling. Numerical stability analyses and discussions are also given for this integration method.

3.2 Mass Scaling

Mass scaling can be performed by directly increasing the density of the material or scaling the system stiffness matrix and adding it to the system mass matrix. Mass scaling can be applied to any algorithm to increase its stability. With mass scaling instead of the original mass matrix of the system an increased mass matrix is used. An increase in the mass matrix results in a decrease in the maximum frequency of the system and consequently increases the critical time increment requirement for the stability of the algorithm. Since an increased mass matrix instead of the original system's mass matrix is used in the solution of the equation of motion, this method is an approximation, and the error in results can be larger than the methods without mass scaling. On the other hand, this approach can be useful when the accuracy target is not too precise since the approach enables the use of larger time increments outside of the stability range of the algorithm without mass scaling. A stabilized version of the central difference method with a mass scaling technique (Soares & Großholz, 2018) is included in this comparative study. Also, the proposed mass scaling by, Soares & Großholz (2018) is applied to the Noh-Bathe method to increase its stability limit, and its performance is studied.

3.2.1 The Stabilized Central Difference Method with Mass Scaling (MS)

In this method, the stability of the standard central difference method is enhanced by modifying the system mass matrix (Soares & Großholz, 2018). The modified mass matrix is given in Equation 3-1.

$$[\mathbf{M}]' = [\mathbf{M}] + 0.5\Delta t[\mathbf{C}] + a\Delta t^2[\mathbf{K}] \quad (3-1)$$

In Equation 3-1, \mathbf{M} , \mathbf{C} , \mathbf{K} , Δt and a are, system mass matrix, damping matrix, system stiffness matrix, and integration constant, respectively.

Acceleration and velocity can be computed according to the standard central difference method as in Equations 3-2 and 3-3.

$$\{\ddot{\mathbf{U}}\}_t = \frac{1}{\Delta t^2}(\{\mathbf{U}\}_{t+\Delta t} - 2\{\mathbf{U}\}_t + \{\mathbf{U}\}_{t-\Delta t}) \quad (3-2)$$

$$\{\dot{\mathbf{U}}\}_t = \frac{1}{2\Delta t}(\{\mathbf{U}\}_{t+\Delta t} - \{\mathbf{U}\}_{t-\Delta t}) \quad (3-3)$$

By inserting the modified mass matrix, Equation 3-2 and 3-3 into the equation of motion given in Equation 2-12, the following expression for the displacement solution is obtained.

$$\{\mathbf{U}\}_{t+\Delta t} = 2\{\mathbf{U}\}_t - \{\mathbf{U}\}_{t-\Delta t} + ([\mathbf{M}] + \Delta t[\mathbf{C}] + a\Delta t^2[\mathbf{K}])^{-1}(\Delta t^2(\{\mathbf{F}\}_t - \{\mathbf{P}\}_t) - \Delta t[\mathbf{C}](\{\mathbf{U}\}_t - \{\mathbf{U}\}_{t-\Delta t})) \quad (3-4)$$

where, \mathbf{F} and \mathbf{P} are external force and internal force vectors, \mathbf{U} , $\dot{\mathbf{U}}$, and $\ddot{\mathbf{U}}$ are displacement, velocity, and acceleration vectors, respectively.

3.2.2 Stability Limit of the Stabilized Central Difference Method with Mass Modification

The spectral radius of this algorithm, obtained from the stability analysis (Soares & Großholz, 2018) in the case where complex eigenvalues are presented is shown in Equation 3-5.

$$\rho^2 = \frac{M+a\Delta t^2 K_T}{M+a\Delta t^2 K_T+\Delta t C} \quad (3-5)$$

If

$$C^2 - 4MK' - 2\Delta t CK' - 4a\Delta t^2 K_T K' + \Delta t^2 K'^2 < 0 \quad (3-6)$$

It can be seen from Equation 3-5 that the spectral radius is smaller than 1 when $C \neq 0$, and it is equal to 1 when $C = 0$. For this method, the undamped vibration case is the most critical case for numerical stability.

For an undamped system, where $M \equiv 1$, $K_T \equiv K' \equiv \omega^2$, and $\Omega = \omega\Delta t$, the numerical stability condition is obtained as in Equation 3-7.

$$\gamma = -1 - (a - 0.25)\Omega^2 < 0 \quad (3-7)$$

Equation 3-7 can be ensured when $a = 0.25$. To have more accurate numerical technique, a smaller “ a ” value is suggested with the use of the following expression where $a \in (0,0.25)$ (Soares & Großholz, 2018).

$$a = 0.25 \tanh(0.25\omega\Delta t) \quad (3-8)$$

where ω is the maximum natural frequency of the system. By using the proposed a , the stability of the method is guaranteed.

CHAPTER 4

IMPLEMENTATION

4.1 Introduction

In this study, a finite element analysis program is developed in the MATLAB environment to compare the run-time of different explicit time integration methods for time history analysis. The program can solve the displacement, velocity, and acceleration responses with time history analysis, including the Noh-Bathe method, the stabilized central difference (CD) method with mass scaling (MS), Chang's method, and the Noh-Bathe method with mass scaling (MS) as explicit solution algorithms. In addition to these methods, the Newmark explicit integration method is also implemented and is used as a reference case while comparing the run-time and accuracy of other explicit integration methods.

The finite element program has a finite element library including 2D frame, 4-node incompatible membrane, and 8-node brick elements. These elements can be used in various structural models where the performance of explicit algorithms is studied.

In this chapter, the implementation of the time history analysis with explicit integration algorithms and finite elements is presented. All integration algorithms perform stiffness matrix calculations within each time step to mimic the solution approach for nonlinear problems, although the structural problem is linear.

4.2 Time History Analysis

The main algorithm for the time history analysis is presented in Figure 4.1. The analysis starts by taking inputs for geometry and material properties, finite element type, loading properties, and analysis options like damping ratio, time increment, and initial and final time. For all dynamic problems, initial conditions are considered as \mathbf{u}_0 , \mathbf{v}_0 and \mathbf{a}_0 are equal to 0. According to the finite element type utilized in the structural model, the element stiffness and lumped mass matrices are assembled to form system stiffness, \mathbf{K} , and mass matrices, \mathbf{M} . If the value of the damping ratio ξ is nonzero, the Rayleigh damping matrix, \mathbf{C} , is calculated.

As the damping, structural stiffness, and mass matrices are computed, the analysis is initiated by calling the subroutine for the chosen explicit integration algorithm. This explicit integration algorithm then calculates displacement, velocity, and acceleration responses.

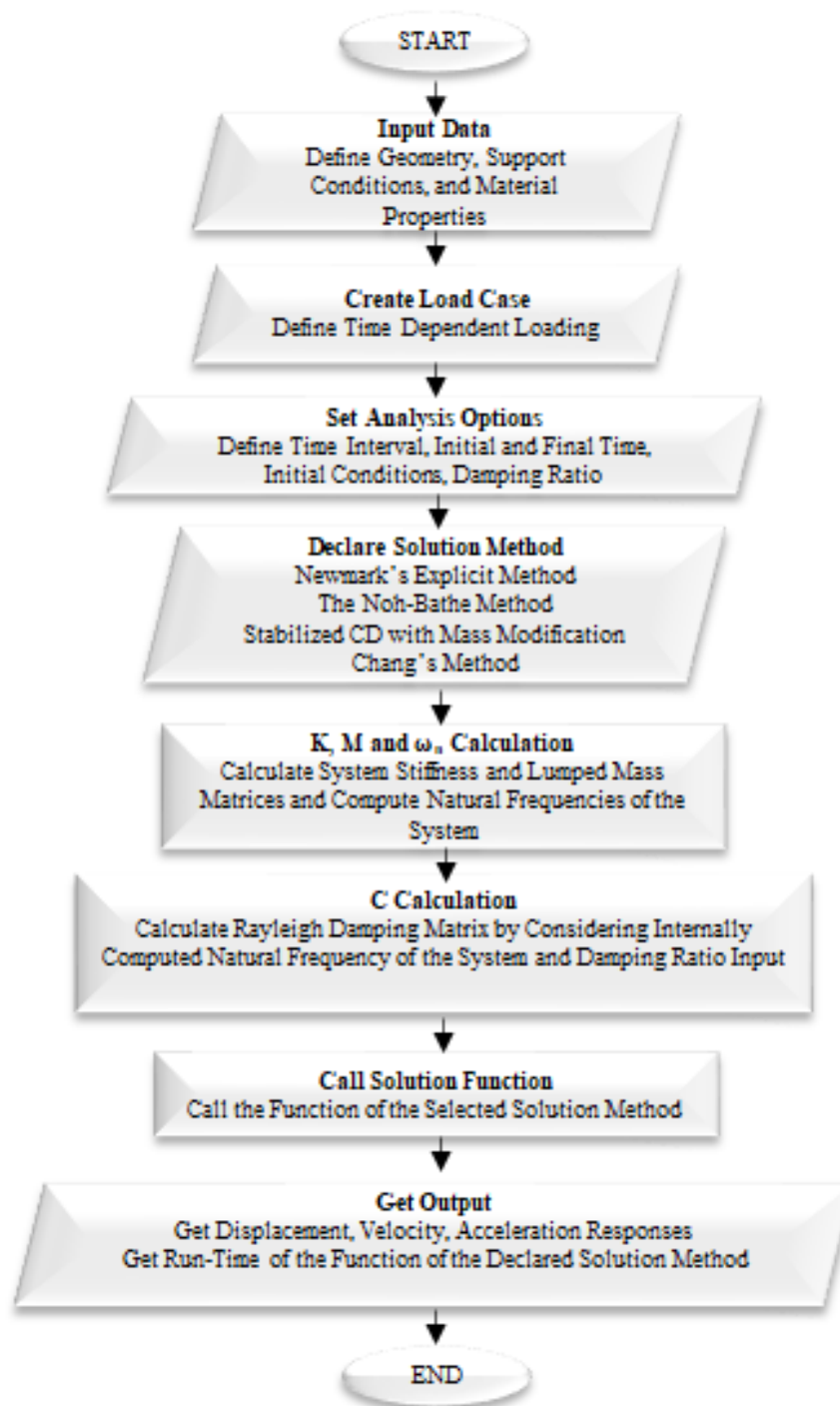


Figure 4.1 Flow Chart for Time History Analysis

4.2.1 Newmark's Explicit Method

The main steps of Newmark's explicit method are presented in Figure 4.2. Newmark's explicit integration method takes \mathbf{K} , \mathbf{M} , and \mathbf{C} matrices as input.

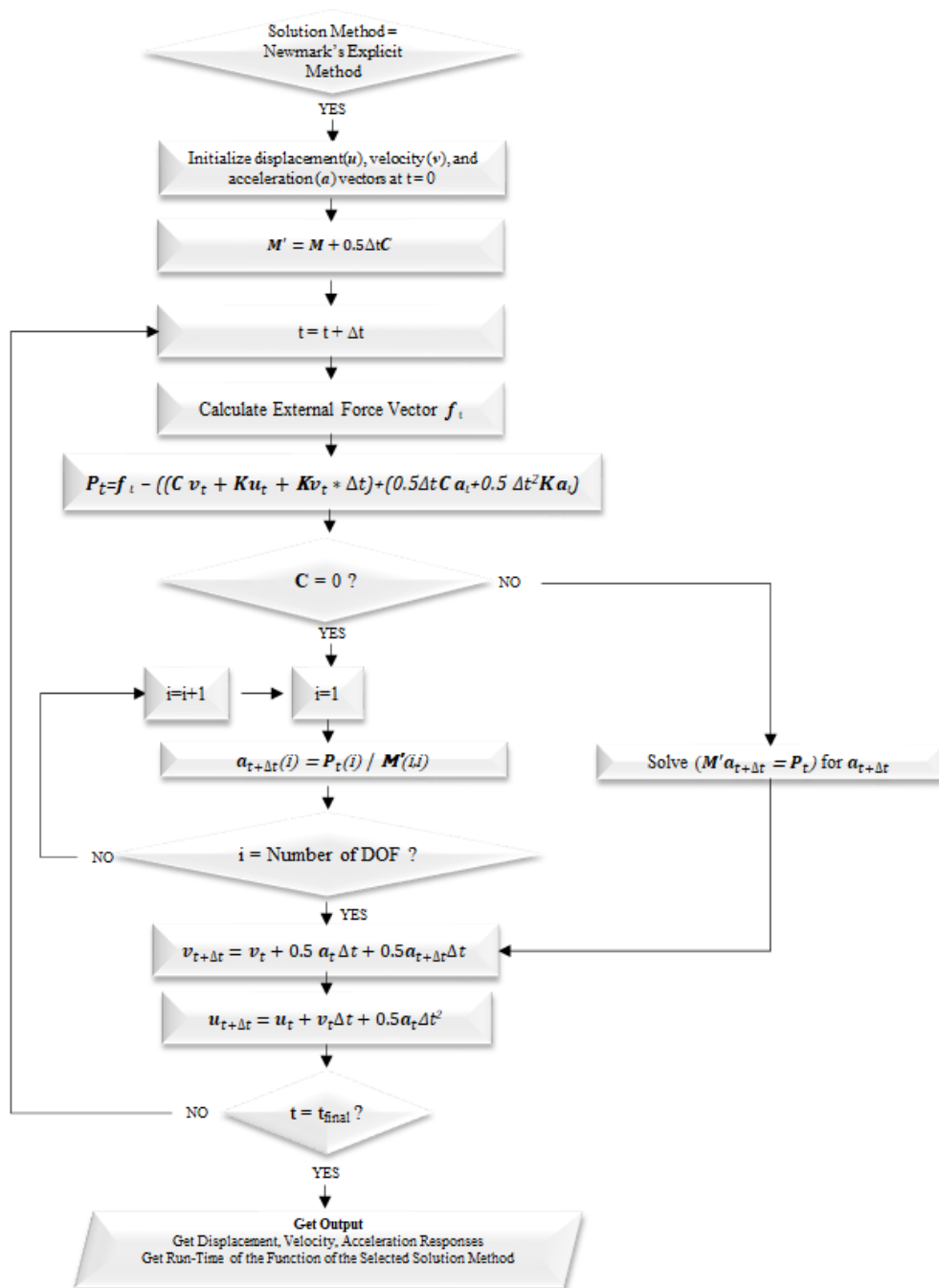


Figure 4.2 Flow Chart for Newmark's Explicit Method

The algorithm presented in Figure 4.2 starts with the initialization of the displacement, velocity, and acceleration vectors at $t = 0$. Before the time stepping is started, M' is calculated as $M + 0.5\Delta t C$. For an undamped system M' is diagonal

and for the computational efficiency, the acceleration vector is computed by element-by-element division instead of matrix inversion. The time stepping loop continues until the final time is reached. The output of this subroutine is the displacement, velocity, and acceleration responses and the run-time.

4.2.2 The Noh-Bathe Method

The main steps of the Noh-Bathe Method are presented in Figure 4.3. The inputs for this function are structural stiffness matrix \mathbf{K} , lumped system mass matrix \mathbf{M} , Rayleigh damping matrix \mathbf{C} , time step size, final time, and “ p ” value. “ p ” value is taken as 0.54 for all problems in this study.

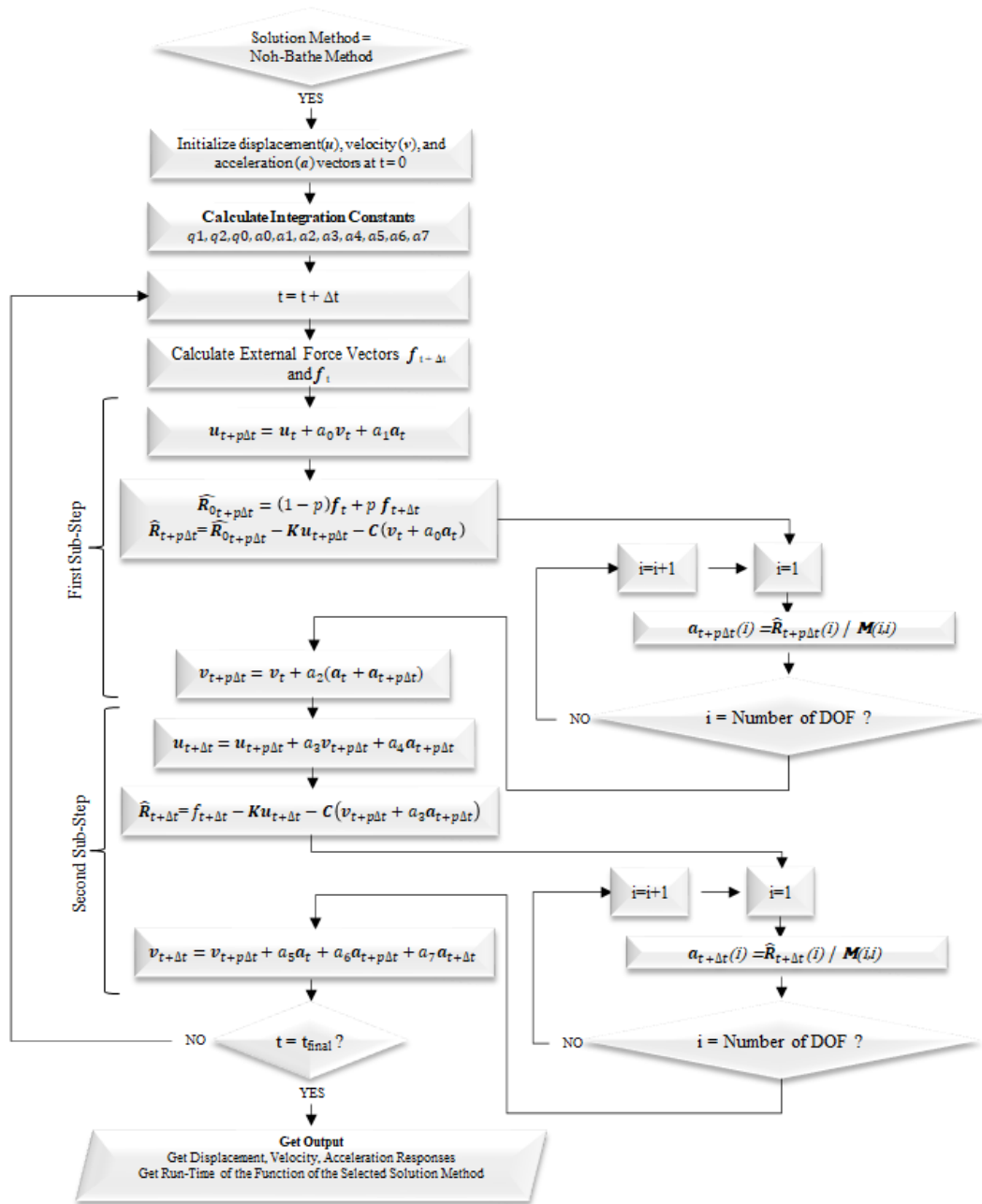


Figure 4.3 Flow Chart for the Noh-Bathe Method

As it is presented in Figure 4.3, the algorithm starts with the initialization of the displacement, velocity, and acceleration vectors at $t = 0$. After computing the integration constants, the loop for time stepping begins. Inside the time stepping loop, when t equals to $t + \Delta t$, primarily the external force vector is computed for

$t = t$ and $t = t + \Delta t$. In the first sub-step, the displacement, acceleration, and velocity vectors at the time equal to $t + p\Delta t$ is computed in order using displacement, acceleration, and velocity vectors at the time equal to t . Then second sub-step is started. In second sub-step, by using the displacement, acceleration, and velocity vectors at the time $t = t + p\Delta t$ computed in the first sub-step, displacement, acceleration, and velocity at $t = \Delta t$, is computed. The time stepping loop continues until the final time is reached. The output of this subroutine is the displacement, velocity, and acceleration responses and the run-time.

Integration constants for the Noh-Bathe Method, used above, are given as follows:

$$\begin{aligned}
 q_1 &= \frac{(1-2p)}{2p(1-p)} \quad , \quad q_2 = 0.5 - pq_1 \quad , \quad q_0 = -q_1 - q_2 + 0.5 \\
 a_0 &= p\Delta t \quad , \quad a_1 = 0.5(p\Delta t)^2 \quad , \quad a_2 = \frac{a_0}{2} \\
 a_3 &= (1-p)\Delta t \quad , \quad a_4 = 0.5((1-p)\Delta t)^2 \quad , \quad a_5 = q_0 a_3 \\
 a_6 &= (0.5 + q_1)a_3 \quad , \quad a_7 = q_2 a_3
 \end{aligned}
 \tag{4-1}$$

4.2.3 Chang's Method

The main steps of Chang's Method are presented in Figure 4.4. The algorithm of Chang's method takes \mathbf{K} , \mathbf{M} , and \mathbf{C} matrices as input.

The algorithm presented in Figure 4.4 starts with the initialization of the displacement, velocity, and acceleration vectors at $t = 0$. As the next step, β_1 and β_2 coefficients whose calculation required the matrix inversion is computed. Then, the time stepping begins. For an undamped system, \mathbf{MM} is diagonal and for the computational efficiency, the velocity vector is computed by element-by-element division instead of matrix inversion. The time stepping loop continues until the final time is reached. The output of this subroutine is the displacement, velocity, and acceleration responses and the run-time.



Figure 4.4 Flow Chart for Chang's Method

4.2.4 The Stabilized Central Difference Method with Mass Scaling (MS)

The main steps of the stabilized central difference (CD) method with mass scaling (MS), are presented in Figure 4.5. The algorithm takes the maximum natural frequency ω_{max} , \mathbf{K} , \mathbf{M} , and \mathbf{C} matrices as input. ω_{max} is used to compute the integration constant a as proposed by Soares & Großholz (2018).

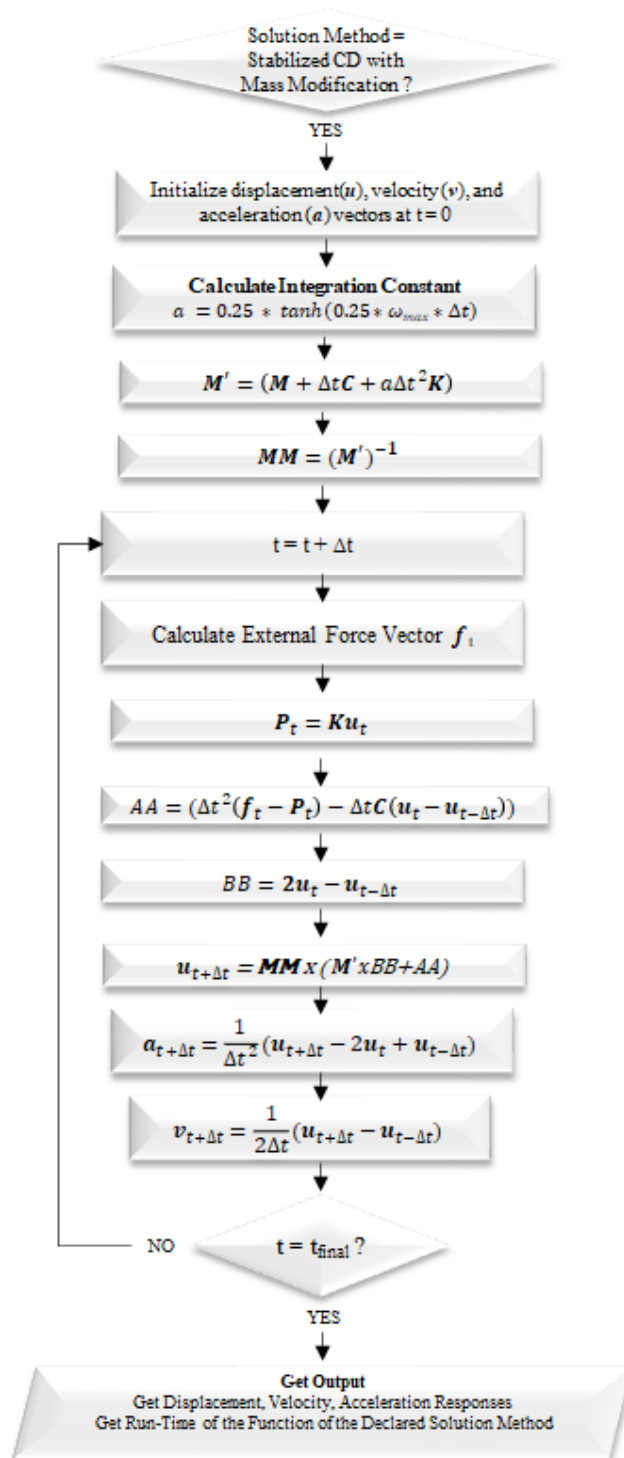


Figure 4.5 Flow Chart for the Stabilized Central Difference Method with Mass Modification

The algorithm presented in Figure 4.5 starts with the initialization of the displacement, velocity, and acceleration vectors at $t = 0$ and $t = \Delta t$. Displacement, velocity, and acceleration vectors at $t = 0$ and $t = \Delta t$ are taken as 0. The stability-related constant a using the maximum natural frequency of the system is computed. Then, mass scaling is performed and the inverse of the scaled mass matrix is calculated and kept before time stepping. Time stepping begins after mass scaling. When $t = t + \Delta t$, the external and internal force vectors at $t = t$ are computed. Displacement at $t + \Delta t$ is calculated by matrix multiplication with the pre-computed inverse scaled mass matrix. To solve displacement at $t + \Delta t$, the displacement at $t = t$ and $t = t - \Delta t$ is used. Then with known displacement at $t = t$, $t = t + \Delta t$ and $t = t - \Delta t$, acceleration and velocity at $t = t + \Delta t$ is solved. The time stepping loop continues until the final time is reached. The output of this subroutine is the displacement, velocity, and acceleration responses and the run-time.

4.2.5 The Noh-Bathe Method with Mass Scaling (MS)

The main steps of the Noh-Bathe method with mass scaling are presented in Figure 4.6. The inputs for this function are structural stiffness matrix \mathbf{K} , lumped system mass matrix \mathbf{M} , Rayleigh damping matrix \mathbf{C} , time step size, final time, and “ p ” value and maximum natural frequency ω_{max} . ω_{max} is used to compute the integration constant a as proposed by Soares & Großholz (2018).

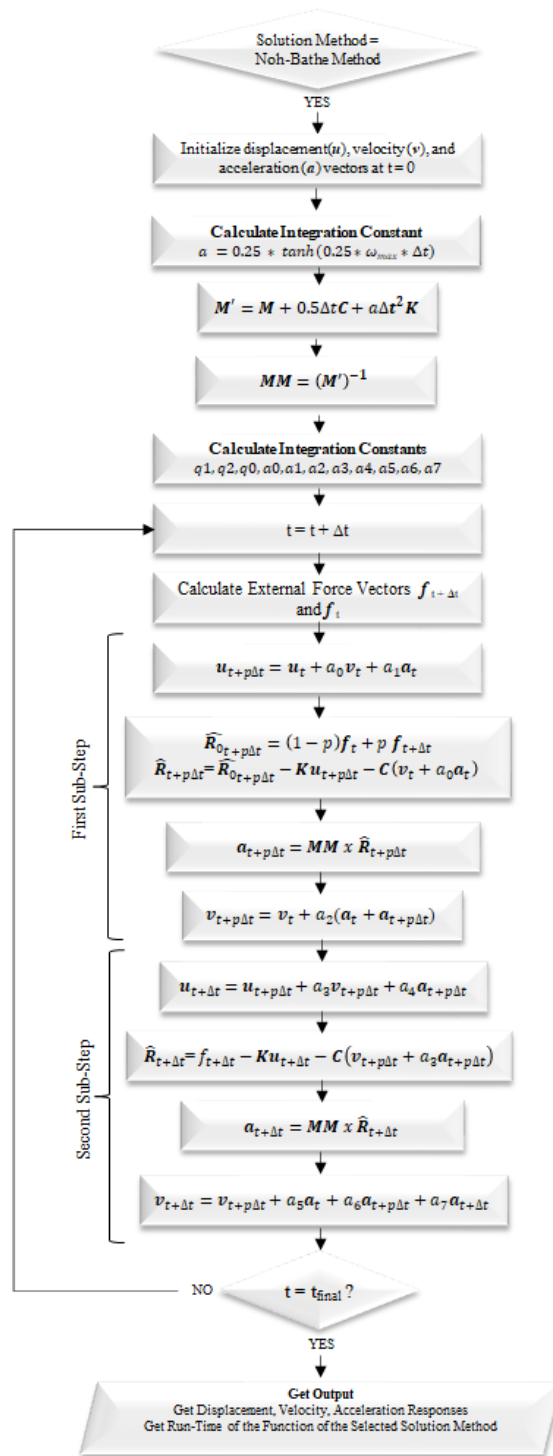


Figure 4.6 Flow Chart for the Noh-Bathe Method with Mass Scaling

The algorithm presented in Figure 4.6 is the Noh-Bathe method with the addition of mass scaling proposed by Soares & Großholz (2018). The stability-related constant α using the maximum natural frequency of the system is computed. Then, mass scaling is performed and the inverse of the scaled mass matrix is calculated and kept before time stepping. Time stepping begins after mass scaling. The first and second substep run similarly to the classical Noh-Bathe method. However, since the used mass matrix is non-diagonal due to mass scaling, matrix multiplication with the inverse of the pre-calculated scaled mass matrix is used to compute acceleration. The time stepping loop continues until the final time is reached. The output of this subroutine is the displacement, velocity, and acceleration responses and the run-time. Matrix multiplication due non-diagonality of the scaled mass matrix is expected to increase the computational cost compared to the classical Noh-Bathe method.

4.3 Finite Element Library

The analysis program has three types of finite elements: 2D frame element, 4-node incompatible membrane element, and 8-node brick element.

4.3.1 2D Frame Element

2D frame element has two nodes and 3 degrees of freedom at each node as presented in Figure 4.7. The stiffness matrix of this frame element is the combination of truss and beam element formulations so that it can undergo axial and bending deformations. This element doesn't consider shear deformations. The stiffness matrix and lumped mass matrices are presented in Equations 4-2 and 4-3, respectively. A small rotational mass is introduced by multiplying mass density with 10^{-6} to eliminate "division by zero error" during the analysis.

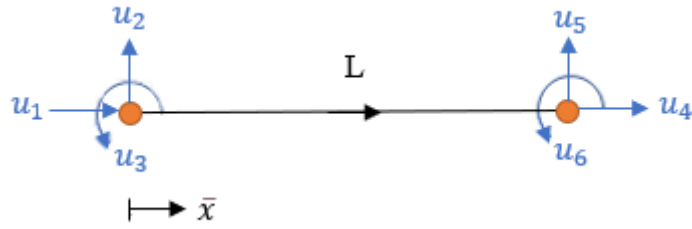


Figure 4.7 2-Node Frame Element

$$K = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (4-2)$$

$$M = \begin{bmatrix} 0.5\rho AL & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.5\rho AL & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho 10^{-6} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5\rho AL & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5\rho AL & 0 \\ 0 & 0 & 0 & 0 & 0 & \rho 10^{-6} \end{bmatrix} \quad (4-3)$$

4.3.2 Incompatible Membrane Element

Quadrilateral membrane elements with bilinear shape functions suffer from the shear locking deficiency. When these elements undergo bending deformation, the shape functions cannot represent the true bent shape of the element. Because of this reason, they behave extremely stiff under bending deformations. To improve this deficiency, higher-order general interpolation terms are to the displacement field definition in

addition to the bilinear terms (N_5 to N_8 in Equation 4-5). Such an element is called incompatible membrane element in this study (Cook, 2007).

For the plane stress problem where $\sigma_{zz} = 0$, displacements are assumed to be uniform through the thickness of the element. The in-plane displacements consist of two components, and deformation in the z -direction is nonzero due to Poisson's Ratio and can be obtained from in-plane displacements.

$$u = \begin{Bmatrix} u_x \\ u_y \end{Bmatrix} \quad (4-4)$$

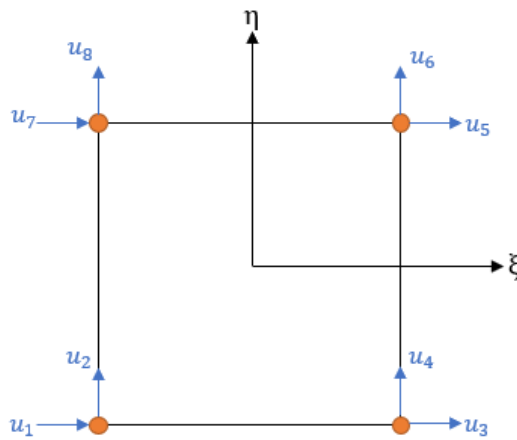


Figure 4.8 4-Node Quadrilateral Membrane

Shape Functions

$$N_1 = \frac{1}{4}(1 - \xi)(1 - \eta) \quad , \quad N_2 = \frac{1}{4}(1 + \xi)(1 - \eta)$$

$$N_3 = \frac{1}{4}(1 + \xi)(1 + \eta) \quad , \quad N_4 = \frac{1}{4}(1 - \xi)(1 + \eta)$$

$$\begin{aligned}
 N_5 &= (1 - \xi^2) & , & & N_6 &= (1 - \eta^2) \\
 N_7 &= (1 - \xi^2) & , & & N_8 &= (1 - \eta^2)
 \end{aligned}
 \tag{4-5}$$

4.3.3 Brick Element

3D Brick element with 8-node is implemented. It has 3 degrees of freedom for translation in X, Y, and Z directions at each node. 3D brick elements are basically an extension of the bilinear membrane. The element displacement field of the 3D solid element is similar to the bilinear membrane with the addition of translation in the z-direction (Cook, 2007).

$$u = \begin{Bmatrix} u_x \\ u_y \\ u_z \end{Bmatrix}
 \tag{4-6}$$

The implemented 3D solid element is a trilinear element since its shape functions are the product of three linear functions.

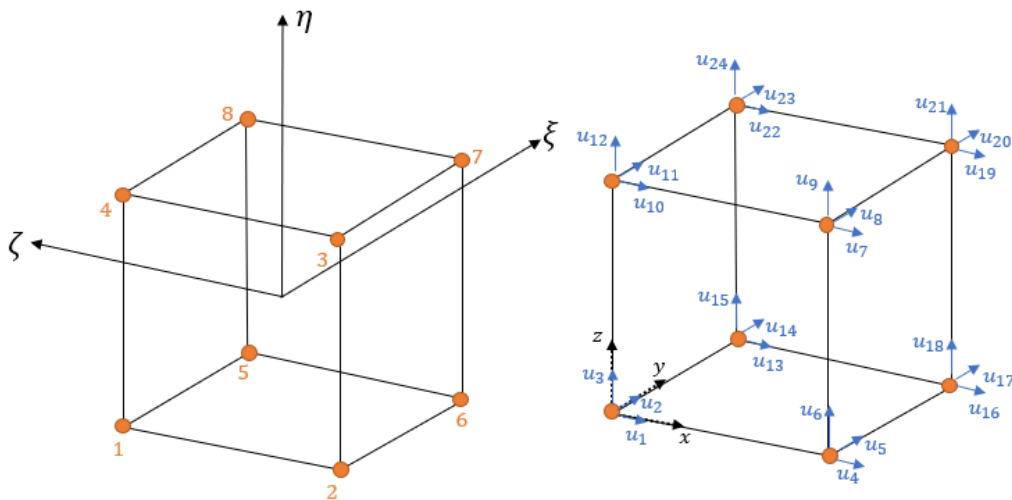


Figure 4.9 8-Node Brick Element

Shape Functions

$$\begin{aligned} N_1 &= \frac{1}{8}(1 - \xi)(1 - \eta)(1 + \zeta) & , & & N_2 &= \frac{1}{8}(1 - \xi)(1 - \eta)(1 - \zeta) \\ N_3 &= \frac{1}{8}(1 - \xi)(1 + \eta)(1 - \zeta) & , & & N_4 &= \frac{1}{8}(1 - \xi)(1 + \eta)(1 + \zeta) \\ N_5 &= \frac{1}{8}(1 + \xi)(1 - \eta)(1 + \zeta) & , & & N_6 &= \frac{1}{8}(1 + \xi)(1 - \eta)(1 - \zeta) \\ N_7 &= \frac{1}{8}(1 + \xi)(1 + \eta)(1 - \zeta) & , & & N_8 &= \frac{1}{8}(1 + \xi)(1 + \eta)(1 + \zeta) \end{aligned}$$

(4-7)

CHAPTER 5

VERIFICATION PROBLEMS

5.1 Introduction

This chapter presents verification of the implemented explicit integration algorithms. One of the verification problems is a single degree of freedom problem under constant loading. The second verification problem is a frame under impulse loading (Cook, 2007). Solving these problems aims to verify the implementation of explicit integration algorithms. For both problems, obtained results are compared with analytical solutions.

5.2 SDOF Problem

A single-degree-of-freedom problem is presented in Figure 5.1. The system's stiffness is 3240 kN/m, and the mass is 18 kg. The applied load is constant in time and equals to 0.1 kN.

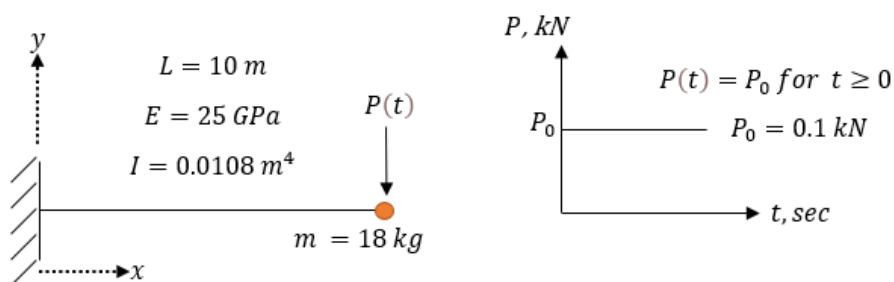


Figure 5.1 Single Degree of Freedom Problem

The problem presented in Figure 5.1 is modeled as a cantilever with a single 2D frame element. Restraints are introduced in translation- x and rotation- z directions at

the free end of the element. The time increment used for all explicit integration algorithms to solve this problem is $\Delta t = 10^{-5}$ sec. The reason for using a small time step for this verification problem is not to have accuracy problems due to the selected time step. For the Noh-Bathe Method, the integration constant “ p ” is taken as $p = 0.54$. For the stabilized CD method with mass modification, the integration constant “ a ” is calculated with the maximum natural frequency, $\omega_n = 4.24 \text{ rad/sec}$ as $a = 2.65 \times 10^{-4}$.

5.2.1 Undamped System

The analytical solution of the undamped single degree of freedom system is given in Equation 5-1.

$$u(t) = (u_{st})_0(1 - \cos \omega_n t) \quad (5-1)$$

$$(u_{st})_0 = \frac{p_0}{k} \quad (5-2)$$

where ω_n is the natural frequency of the system.

In Figure 5.2, the displacement response of the SDOF system is presented. The plotted solutions are obtained from the Noh-Bathe method, the Stabilized CD Method with mass scaling (MS), Chang’s method, the Noh-Bathe method with mass scaling (MS), and the analytical solution.

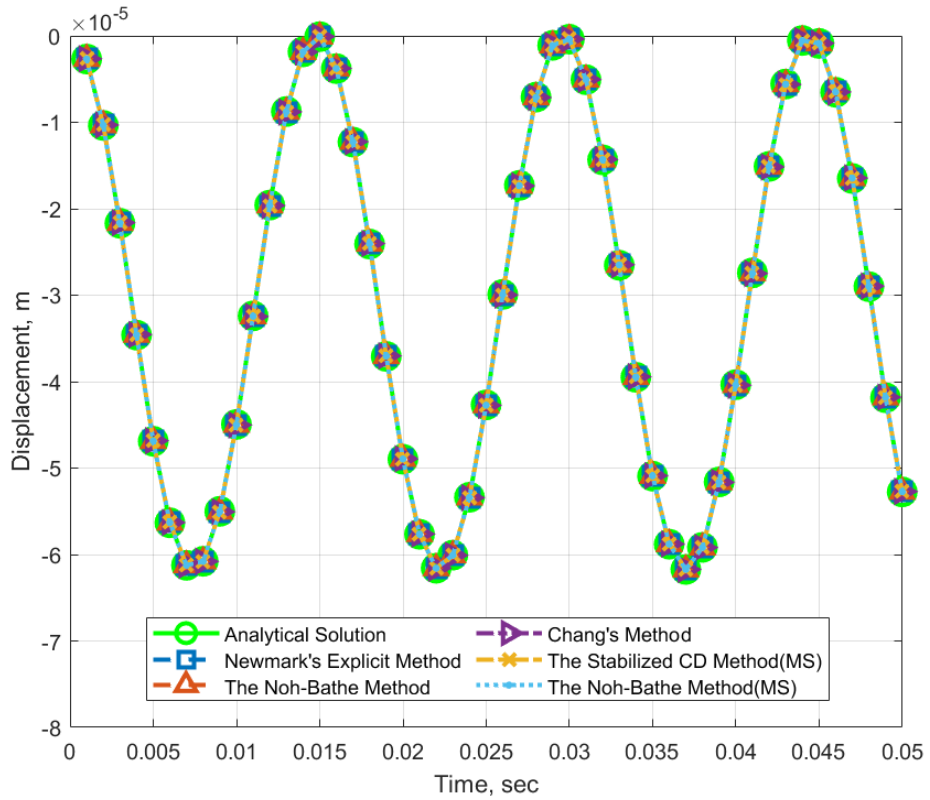


Figure 5.2 Displacement Response of the Undamped SDOF System

The data points for displacement response at every 100 steps are recorded for the displacement plot shown in Figure 5.2. Figure 5.2 shows that the displacement responses from all algorithms match the analytical solution.

5.2.2 Damped System

The analytical solution for the damped single-degree-of-freedom system is presented below.

$$u(t) = (u_{st})_0 \left[1 - e^{-\xi \omega_n t} \left(\cos \omega_D t + \frac{\xi}{\sqrt{1-\xi^2}} \sin \omega_D t \right) \right] \quad (5-3)$$

$$(u_{st})_0 = \frac{p_0}{k} \quad (5-4)$$

$$\omega_D = \omega_n \sqrt{1 - \xi^2} \quad (5-5)$$

where ξ is the damping ratio and ω_n is the natural frequency of the system.

In Figure 5.3, the displacement response from the Noh-Bathe method, the Stabilized CD Method with mass scaling (MS), Chang's method, the Noh-Bathe method with mass scaling (MS), and the analytical solution with a damping ratio, $\xi = 0.05$ is presented.

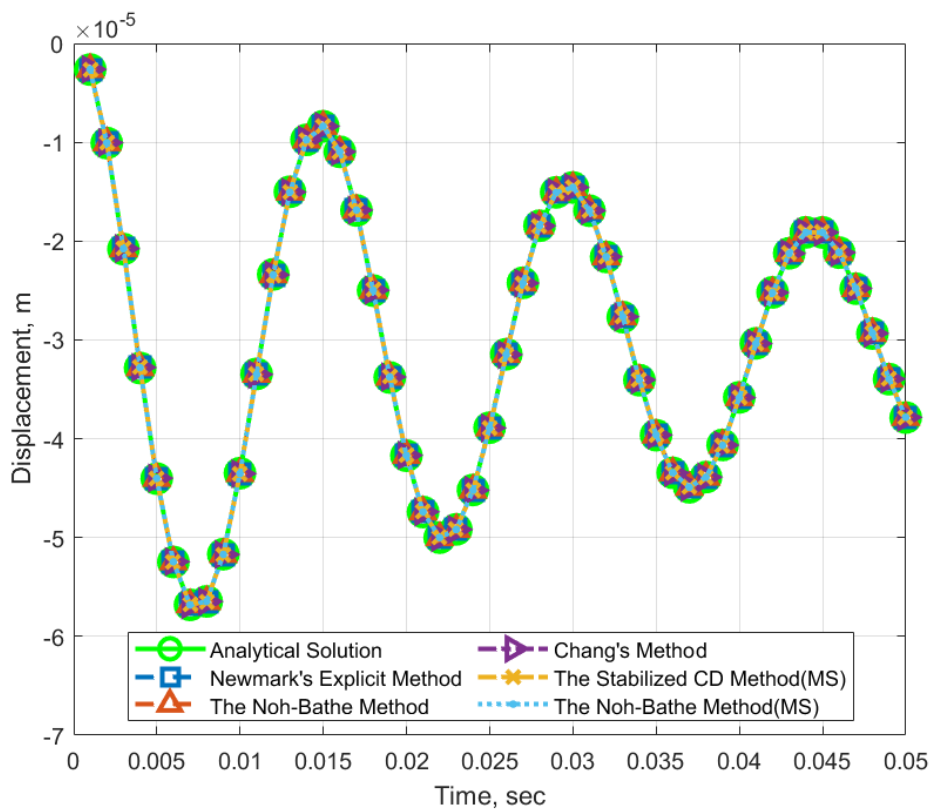


Figure 5.3 Displacement Response of Damped SDOF System

The data points at every 100 steps are recorded for the displacement plot shown in Figure 5.3. Figure 5.3 shows that the displacement responses from all algorithms match the analytical solution.

5.3 Frame Under Impulse Loading

Figure 5.4 presents a frame problem under impulse loading (Cook, 2007). The loading is applied to the right end of the frame in the X direction.

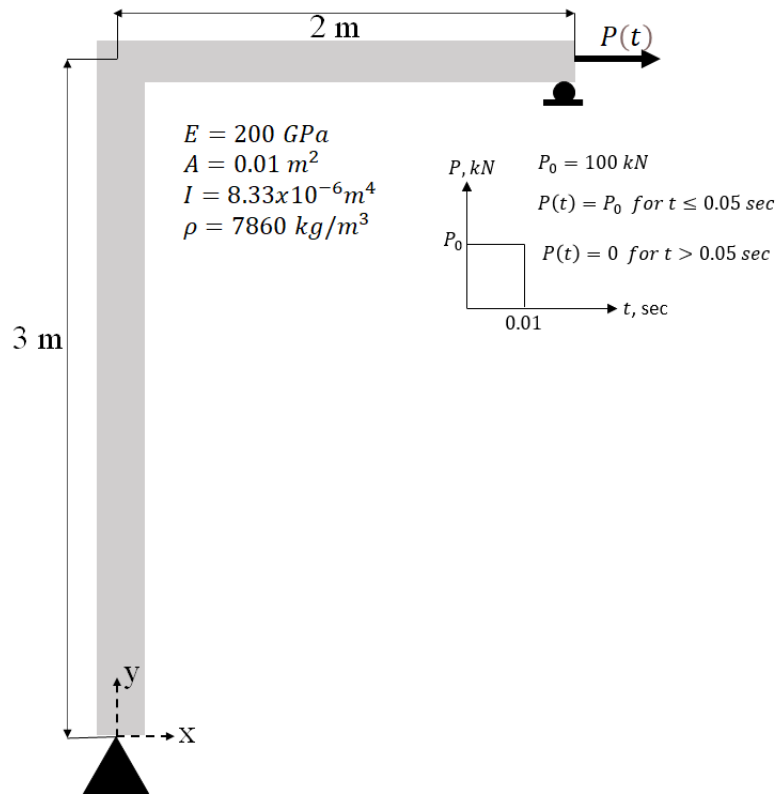
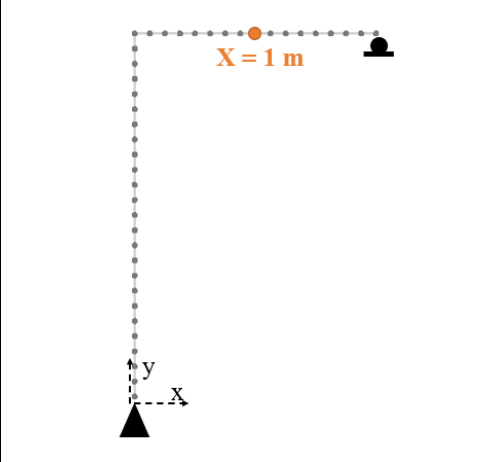


Figure 5.4 Frame Under Impulse Loading

The problem shown in Figure 5.4 is discretized by 2D frame elements with a size of 0.125 m. Model information is presented in Table 5.1. Table 5.1 shows that accurate results can be obtained by using time increments smaller than $6.01 \times 10^{-6} \text{ sec}$. For the reference solution, the problem is modeled in LARSA 4D by using *beam* elements with zero shear area. Linear time history analysis is performed in LARSA 4D environment to obtain reference displacement response.

Table 5.1 Model Information

	<table border="1"> <tr> <td>Number of DOF</td> <td>120</td> <td></td> </tr> <tr> <td>ω_{min}</td> <td>20.78</td> <td>rad/sec</td> </tr> <tr> <td>ω_{max}</td> <td>104545.5</td> <td>rad/sec</td> </tr> <tr> <td>$\Delta t_{max\ accurate} = \frac{T_{min}}{10}$</td> <td>6.01E-06</td> <td>sec</td> </tr> </table>	Number of DOF	120		ω_{min}	20.78	rad/sec	ω_{max}	104545.5	rad/sec	$\Delta t_{max\ accurate} = \frac{T_{min}}{10}$	6.01E-06	sec
Number of DOF	120												
ω_{min}	20.78	rad/sec											
ω_{max}	104545.5	rad/sec											
$\Delta t_{max\ accurate} = \frac{T_{min}}{10}$	6.01E-06	sec											

The stability requirement of the algorithms to solve this problem is presented in Table 5.2. Chang’s method is unconditionally stable, and the stabilized CD method with mass scaling is also stable for all time increments. For the stabilized CD method (MS), the stability-related parameter α is between 0 and 0.25 for all time increments when the recommended equation is used to calculate this parameter Soares & Großholz (2018). The stability limit of the Noh-Bathe method is approximately 1.9 times larger than the stability limit of Newmark’s explicit method.

Table 5.2 Stability Requirements

Method	$\Delta t_{critical}$
Newmark's Explicit Method	1.91E-05 sec
The Stabilized CD Method (MS)	$\alpha \in [0,0.25]$, stable for all Δt
Chang's Method	Unconditionally Stable
The Noh-Bathe Method, $p=0.54$	3.58E-05 sec

In Table 5.3, the displacement results at the $x = 1$ m on the horizontal portion of the frame element in the translation y direction are presented at discrete times. The results are obtained by using time increments as $1 \times 10^{-6} \text{ sec}$ for all algorithms and in the LARSA 4D model.

Table 5.3 Displacement Results

$\Delta t = 10^{-6} \text{sec}$	LARSA 4D	Newmark's Explicit Method	The Noh- Bathe Method	The Noh- Bathe Method (MS)	Chang's Method	The Stabilized CD Method (MS)
Time, sec	Displacement, m					
0.01	-0.0042	-0.0042	-0.0042	-0.0042	-0.0042	-0.0042
0.02	-0.0018	-0.0018	-0.0018	-0.0018	-0.0018	-0.0018
0.03	-0.0020	-0.0020	-0.0020	-0.0020	-0.0020	-0.0020
0.04	-0.0100	-0.0100	-0.0100	-0.0100	-0.0100	-0.0100
0.05	-0.0049	-0.0050	-0.0050	-0.0050	-0.0050	-0.0050
0.06	-0.0064	-0.0063	-0.0063	-0.0063	-0.0063	-0.0063
0.07	-0.0129	-0.0128	-0.0128	-0.0128	-0.0128	-0.0128
0.08	-0.0059	-0.0059	-0.0059	-0.0059	-0.0059	-0.0059
0.09	-0.0082	-0.0082	-0.0082	-0.0082	-0.0082	-0.0082
0.1	-0.0118	-0.0119	-0.0119	-0.0119	-0.0119	-0.0119

Figure 5.5 presents the displacement response plots drawn with data points at every 10^{-6} sec. The results for this displacement plots are also obtained by using time increments as 1×10^{-6} sec for all algorithms and in the LARSA 4D model.

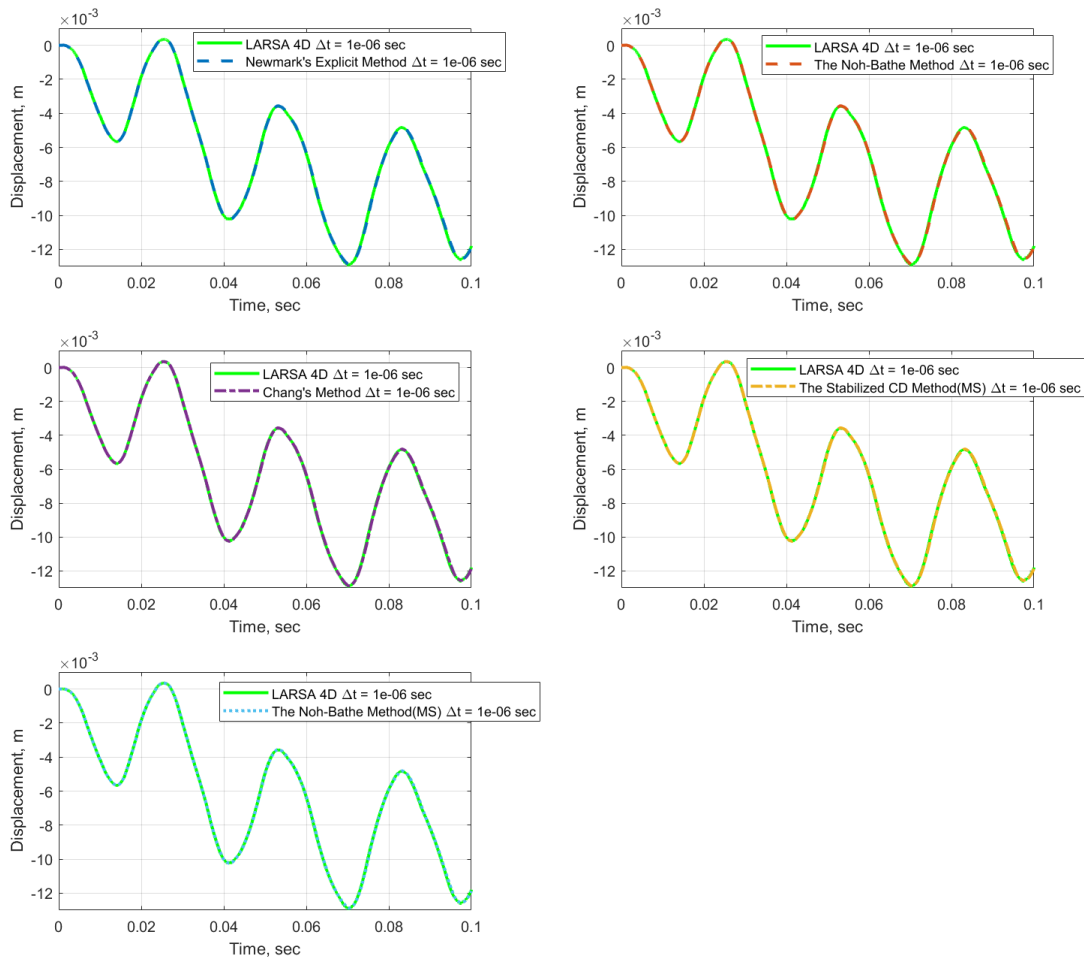


Figure 5.5 Displacement Response Comparison

Table 5.3 and Figure 5.5 shows that the results obtained from all algorithms at discrete times are matched with the reference solution obtained from the LARSA 4D model, meaning that implementations of all algorithms are verified.

5.4 Convergence Rate of Algorithms

In Figure 5.6, the relative displacement errors are plotted for different time steps for each integration algorithm. The slope of each plot is considered to be the convergence rate. These plots are drawn using the results obtained from the frame problem under impulse loading. All algorithms are run when $\Delta t = 1 \times 10^{-8} \text{ sec}$, $\Delta t = 1 \times 10^{-7} \text{ sec}$, $\Delta t = 1 \times 10^{-6} \text{ sec}$, and $\Delta t = 1 \times 10^{-5} \text{ sec}$. The displacement results at $x=1 \text{ m}$ in translation y direction at $t=0.001 \text{ sec}$ obtained from each algorithm are compared with the reference solution, and errors are calculated. The reference solution for the errors presented in Figure 5.6 is obtained from Newmark's explicit method when the time increment is used as $1 \times 10^{-9} \text{ sec}$.

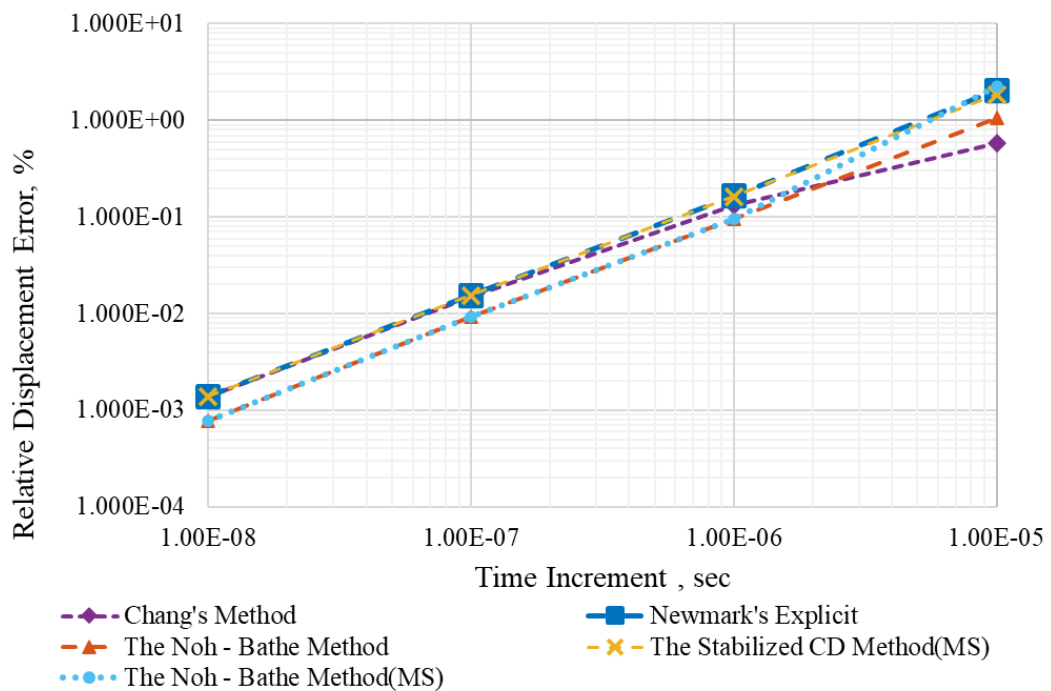


Figure 5.6 Convergence Rate of Algorithms

Figure 5.6 shows that the convergence rates of all algorithms are very close to each other. Still, the results obtained from the Noh-Bathe method are closer to the reference explicit algorithm's results.

5.5 Summary of Results

In this chapter, the first verification problem shows that the results obtained from implemented algorithms match the analytical solution for a damped and undamped single-degree-of-freedom system. In the second verification problem, a frame structure is analyzed and the displacement responses are compared with the results obtained from LARSA 4D model. The second verification problem shows that the displacement results from all algorithms' discrete times match the reference results obtained from the LARSA 4D results.

The convergence rate of implemented algorithms is calculated using the frame under the impulse loading model, and all algorithms' convergence rate is found to be close to each other. The Noh-Bathe method gives closer results to the reference solution for the tested time increments.

CHAPTER 6

CASE STUDIES

6.1 Introduction

In this chapter, various dynamic problems are solved to compare the stability, accuracy, and run-time of the examined explicit integration algorithms. The first case study is a simple cantilever under vertical impulse. The cantilever is modeled in two ways by using, membrane, and brick elements. The aim of solving this problem is to compare the performance of the solution of the algorithms for a simple system.

The second case study is 3D clamped solid problem. The solid is modeled with 8-node brick elements. The solid is clamped from all edges; hence it has a very stiff behavior in the axial direction. This benchmark problem aims to push the examined explicit integration methods beyond their accuracy and stability limits. A mesh study is performed in this problem to increase degrees of freedom and see many high-frequency modes.

The third case study, moving load on a three-span road bridge, is analyzed with membrane elements to see the explicit integration algorithms' performance on a real-life problem.

The variety in loading types and used element formulation is intentionally provided in selected benchmark problems to see the applicability of all examined methods for different types of structural dynamic problems. This preference also supported the precision of the conclusions regarding the robustness of the studied explicit integration methods.

Accuracy comparison is performed for a certain time range for each problem. The errors in results are computed with respect to reference results. The reference results

(x_{ref}) are obtained from the Newmark's Explicit Integration Method with a time step smaller than $\Delta t = \frac{T_{min}}{10}$. In this chapter, errors are calculated as follows:

$$error \% = \left| \frac{x - x_{ref}}{x_{ref}} \right| \times 100 \quad (6-1)$$

where x is the result obtained from the examined integration algorithm.

6.2 Case Study 1: Simple Cantilever Under Vertical Impulse

In Figure 6.1, a simple cantilever problem is presented. The modulus of elasticity and density of the 2 m cantilever are 200 GPa and 7840 kg/m³, respectively. The cross-section of the cantilever is a 0.2 m x 1 m rectangle. At the free end of the cantilever, 100 kN force is applied vertically. The loading is removed at $t = 0.05$ sec.

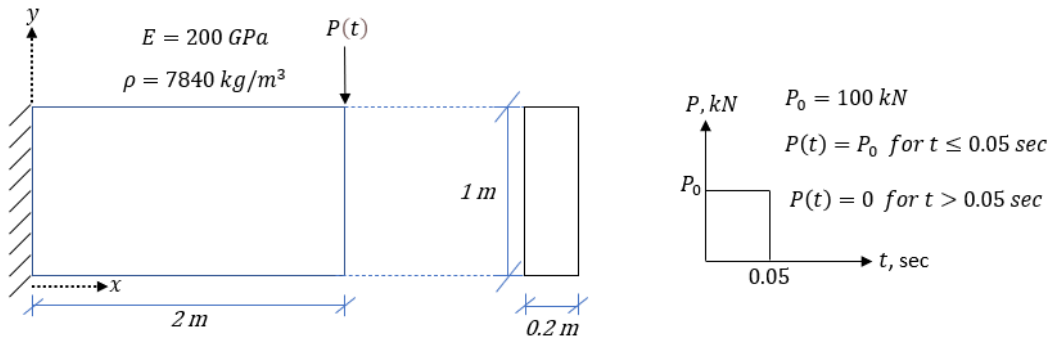
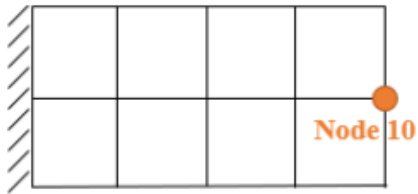
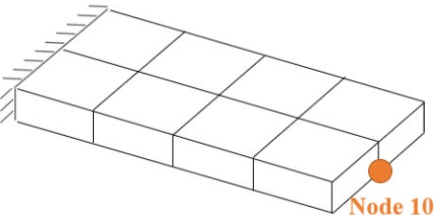


Figure 6.1 Simple Cantilever Under Vertical Impulse

Discretization of the simple cantilever under vertical impulse is presented in Table 6.1. The simple cantilever problem is modeled with 4-Node Membranes and 8-Node Brick elements. Model 1 and Model 2 presented in Table 6.1 correspond to 4-Node Membrane and 8-Node Brick models, respectively. Table 6.1 also shows the total number of degrees of freedom, the system's minimum and maximum natural frequencies, and the maximum time increment for the accurate reference results, which corresponds to the minimum period of the system over 10. In this case study,

the goal was to see the effect of the element formulation on the performance of implemented explicit integration algorithms.

Table 6.1 Discretization of Case Study 1

<p style="text-align: center;"><u>Model 1: 4-Node Membrane</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td style="text-align: right;">Number of DOF</td> <td style="text-align: center;">24</td> </tr> <tr> <td style="text-align: right;">ω_{min}</td> <td style="text-align: center;">1099.5767 rad/sec</td> </tr> <tr> <td style="text-align: right;">ω_{max}</td> <td style="text-align: center;">19945.09 rad/sec</td> </tr> <tr> <td style="text-align: right;">$\Delta t_{max\ accurate}$</td> <td style="text-align: center;">3.1502E-05 sec</td> </tr> <tr> <td style="text-align: right;">$= \frac{T_{min}}{10}$</td> <td></td> </tr> </tbody> </table>	Number of DOF	24	ω_{min}	1099.5767 rad/sec	ω_{max}	19945.09 rad/sec	$\Delta t_{max\ accurate}$	3.1502E-05 sec	$= \frac{T_{min}}{10}$		 <p style="text-align: center;">0.5 m x 0.5 m</p>
Number of DOF	24										
ω_{min}	1099.5767 rad/sec										
ω_{max}	19945.09 rad/sec										
$\Delta t_{max\ accurate}$	3.1502E-05 sec										
$= \frac{T_{min}}{10}$											
<p style="text-align: center;"><u>Model 2: 8-Node Brick</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr> <td style="text-align: right;">Number of DOF</td> <td style="text-align: center;">72</td> </tr> <tr> <td style="text-align: right;">ω_{min}</td> <td style="text-align: center;">501.5008 rad/sec</td> </tr> <tr> <td style="text-align: right;">ω_{max}</td> <td style="text-align: center;">49888.3519 rad/sec</td> </tr> <tr> <td style="text-align: right;">$\Delta t_{max\ accurate}$</td> <td style="text-align: center;">1.2594E-05 sec</td> </tr> <tr> <td style="text-align: right;">$= \frac{T_{min}}{10}$</td> <td></td> </tr> </tbody> </table>	Number of DOF	72	ω_{min}	501.5008 rad/sec	ω_{max}	49888.3519 rad/sec	$\Delta t_{max\ accurate}$	1.2594E-05 sec	$= \frac{T_{min}}{10}$		 <p style="text-align: center;">0.5 m x 0.5 m x 0.2 m</p>
Number of DOF	72										
ω_{min}	501.5008 rad/sec										
ω_{max}	49888.3519 rad/sec										
$\Delta t_{max\ accurate}$	1.2594E-05 sec										
$= \frac{T_{min}}{10}$											

In Table 6.2 the stability requirements of implemented explicit integration algorithms are presented. The a value for the Stabilized CD Method (MS) is calculated from the recommended equation (Soares & Großholz, 2018). The calculated a value is always between $[0,0.25]$, meaning that the algorithm is stable for all Δt . The critical time increment for Newmark's Explicit Method and the Noh-Bathe Method are calculated and presented in Table 6.2. Chang's Method is unconditionally stable.

Table 6.2 Stability Requirements of the Integration Methods in Case Study 1

	Model 1: 4-Node Membrane	Model 2: 8-Node Brick
Method	$\Delta t_{critical}$	$\Delta t_{critical}$
Newmark's Explicit Method	1.00E-04 sec	4.01E-05 sec
The Stabilized CD Method (MS)	$a \in [0,0.25]$, stable for all Δt	$a \in [0,0.25]$, stable for all Δt
Chang's Method	Unconditionally Stable	Unconditionally Stable
The Noh-Bathe Method, p=0.54	1.88E-04 sec	7.51E-05 sec

Table 6.1 and Table 6.2 show that by changing the used element from 4-node membranes to 8-node solid elements, the maximum natural frequency of the system increased with the increase in the number of degrees of freedom. This causes a smaller time step required for the stability of the Newmark's Explicit Method and the Noh-Bathe Method in the model with 8-node solid elements. Table 6.2 also shows that the critical time increment for the stability of the Noh-Bathe method is approximately 1.9 times larger than Newmark's explicit method.

The time history analysis is performed on the developed finite element analysis program between t equal to 0 and t equal to 0.016 sec. For the accuracy comparison, 7 test runs using different Δt values are performed for the 4-node membrane model. The time increments for test runs of the 4-node membrane model are $\Delta t = 1 \times 10^{-6} \text{ sec}$, $\Delta t = 5 \times 10^{-6} \text{ sec}$, $\Delta t = 1 \times 10^{-5} \text{ sec}$, $\Delta t = 5 \times 10^{-5} \text{ sec}$, $\Delta t = 1 \times 10^{-4} \text{ sec}$, $\Delta t = 1.5 \times 10^{-4} \text{ sec}$, and $\Delta t = 2 \times 10^{-4} \text{ sec}$ for Test 1, Test 2, Test 3, Test 4, Test 5, Test 6, and Test 7, respectively. For the 8-node brick model, 8 test runs using different Δt values are performed. The time increments for test runs of the 8-node brick model are $\Delta t = 1 \times 10^{-6} \text{ sec}$, $\Delta t = 2.5 \times 10^{-6} \text{ sec}$, $\Delta t = 5 \times 10^{-6} \text{ sec}$, $\Delta t = 7.5 \times 10^{-6} \text{ sec}$, $\Delta t = 1 \times 10^{-5} \text{ sec}$, $\Delta t = 2.5 \times 10^{-5} \text{ sec}$, $\Delta t = 5 \times 10^{-5} \text{ sec}$, and $7.5 \times 10^{-5} \text{ sec}$ for Test 1, Test 2, Test 3, Test 4, Test 5, Test 6, Test 7, and Test 8 respectively.

Table 6.3 and Table 6.4 show maximum displacements in the y direction at "node 10" obtained for the analyzed time range from all test runs reported for both membrane and brick models. Table 6.3 and Table 6.4 also show the time when the maximum displacement in the y direction at "node 10" is obtained, and the displacement error and phase difference with respect to the accurate reference solution obtained from Newmark's Explicit Method. For the reference solution, the used time increment is $\Delta t = 1 \times 10^{-6} \text{ sec}$ which is a smaller time step than the time step obtained from the rule of thumb, $\frac{T_{min}}{10}$ presented in Table 6.1. The displacement error is calculated by dividing the difference between the obtained value and the

reference value by the reference value, taking the absolute of the ratio, and multiplying it by 100. The phase difference is calculated by taking the absolute difference between the reference value and the obtained value. The displacement response plots from all algorithms for all test runs can be seen in Figure A.1 and Figure A.2 in Appendix A.

Table 6.3 shows that in Model 1, the displacement error from the Noh-Bathe method is zero in Test 1, Test 2, and Test 3, and this algorithm is unstable in Test 7. Table 6.4 shows that, in Model 2, the displacement error from the Noh-Bathe method is zero in Test 1, Test 2, Test 3, and Test 4. The time increment in Test 3, in Model 1, is 1.33 times larger than the time increment in Test 4, in Model 2. The displacement error from the stabilized CD method with mass scaling (MS) is zero up to Test 2 in both Model 1 and Model 2. The time increment in Test 2, in Model 1, is two times larger than the time increment in Test 2, in Model 2. The displacement error in Chang's method is zero at Test 1 in Model 1 only. These results show that good accuracy can be achieved from all algorithms in the 4-node membrane model (Model 1) with greater time increments than the time increment required for good accuracy in the 8-node brick model (Model 2). This validates the increase in the number of degrees of freedom causing smaller time increments required for good accuracy.

The comparison shows that explicit integration algorithms have different accuracy characteristics and may yield different maximum displacement errors with different time increments. The solution time of algorithms is compared by using the largest possible time increments that give a closer maximum displacement error to the selected allowable maximum displacement error. The analyzed time range is kept fixed for all algorithms between 0-0.016 sec. The solution time comparison for case study 1 is presented in Table 6.5 and Table 6.6.

Table 6.3 Accuracy Comparison from Case Study 1 – Model 1

Model 1: 4-Node Membrane						
	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error, %	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-06	-1.7557E-04	0.014236	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-06	-1.7557E-04	0.014236	0.000	0.00E+00
	Chang's Method	1.00E-06	-1.7557E-04	0.014236	0.000	0.00E+00
	The Noh - Bathe Method	1.00E-06	-1.7557E-04	0.014236	0.000	0.00E+00
Test 2	The Stabilized CD Method(MS)	5.00E-06	-1.7557E-04	0.014240	0.000	4.00E-06
	Chang's Method	5.00E-06	-1.7556E-04	0.014235	0.006	1.00E-06
	The Noh - Bathe Method	5.00E-06	-1.7557E-04	0.014235	0.000	1.00E-06
Test 3	The Stabilized CD Method(MS)	1.00E-05	-1.7558E-04	0.014240	0.006	4.00E-06
	Chang's Method	1.00E-05	-1.7555E-04	0.014230	0.011	6.00E-06
	The Noh - Bathe Method	1.00E-05	-1.7557E-04	0.014240	0.000	4.00E-06
Test 4	The Stabilized CD Method(MS)	5.00E-05	-1.7594E-04	0.014250	0.211	1.40E-05
	Chang's Method	5.00E-05	-1.7568E-04	0.014300	0.063	6.40E-05
	The Noh - Bathe Method	5.00E-05	-1.7570E-04	0.014250	0.074	1.40E-05
Test 5	The Stabilized CD Method(MS)	1.00E-04	-1.7636E-04	0.014300	0.450	6.40E-05
	Chang's Method	1.00E-04	-1.7666E-04	0.014300	0.621	6.40E-05
	The Noh - Bathe Method	1.00E-04	-1.7543E-04	0.014200	0.080	3.60E-05
Test 6	The Stabilized CD Method(MS)	1.50E-04	-1.7707E-04	0.014400	0.854	1.64E-04
	Chang's Method	1.50E-04	-1.7671E-04	0.014400	0.649	1.64E-04
	The Noh - Bathe Method	1.50E-04	-1.7540E-04	0.014250	0.097	1.40E-05
Test 7	The Stabilized CD Method(MS)	2.00E-04	-1.7690E-04	0.014400	0.758	1.64E-04
	Chang's Method	2.00E-04	-1.7439E-04	0.014400	0.674	1.64E-04
	The Noh - Bathe Method	2.00E-04	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE

Table 6.4 Accuracy Comparison from Case Study 1 – Model 2

Model 2: 8 -Node Brick						
	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error, %	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-06	-1.5923E-04	0.013554	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-06	-1.5923E-04	0.013554	0.000	0.00E+00
	Chang's Method	1.00E-06	-1.5924E-04	0.013554	0.006	0.00E+00
	The Noh - Bathe Method	1.00E-06	-1.5923E-04	0.013554	0.000	0.00E+00
Test 2	The Stabilized CD Method(MS)	2.50E-06	-1.5923E-04	0.013555	0.000	1.00E-06
	Chang's Method	2.50E-06	-1.5924E-04	0.013555	0.006	1.00E-06
	The Noh - Bathe Method	2.50E-06	-1.5923E-04	0.013555	0.000	1.00E-06
Test 3	The Stabilized CD Method(MS)	5.00E-06	-1.5922E-04	0.013555	0.006	1.00E-06
	Chang's Method	5.00E-06	-1.5927E-04	0.013560	0.025	6.00E-06
	The Noh - Bathe Method	5.00E-06	-1.5923E-04	0.013555	0.000	1.00E-06
Test 4	The Stabilized CD Method(MS)	7.50E-06	-1.5921E-04	0.013552	0.013	2.00E-06
	Chang's Method	7.50E-06	-1.5931E-04	0.013567	0.050	1.30E-05
	The Noh - Bathe Method	7.50E-06	-1.5923E-04	0.013552	0.000	2.00E-06
Test 5	The Stabilized CD Method(MS)	1.00E-05	-1.5919E-04	0.013550	0.025	4.00E-06
	Chang's Method	1.00E-05	-1.5934E-04	0.013570	0.069	1.60E-05
	The Noh - Bathe Method	1.00E-05	-1.5922E-04	0.013560	0.006	6.00E-06
Test 6	The Stabilized CD Method(MS)	2.50E-05	-1.5919E-04	0.013550	0.025	4.00E-06
	Chang's Method	2.50E-05	-1.5889E-04	0.013650	0.214	9.60E-05
	The Noh - Bathe Method	2.50E-05	-1.5915E-04	0.013550	0.050	4.00E-06
Test 7	The Stabilized CD Method(MS)	5.00E-05	-1.5874E-04	0.013650	0.308	9.60E-05
	Chang's Method	5.00E-05	-1.5901E-04	0.013600	0.138	4.60E-05
	The Noh - Bathe Method	5.00E-05	-1.5887E-04	0.013550	0.226	4.00E-06
Test 8	The Stabilized CD Method(MS)	7.50E-05	-1.5893E-04	0.013650	0.188	9.60E-05
	Chang's Method	7.50E-05	-1.5955E-04	0.013650	0.201	9.60E-05
	The Noh - Bathe Method	7.50E-05	-1.5884E-04	0.013650	0.245	9.60E-05

Table 6.5 Solution Time Comparison from Case Study 1 – Model 1

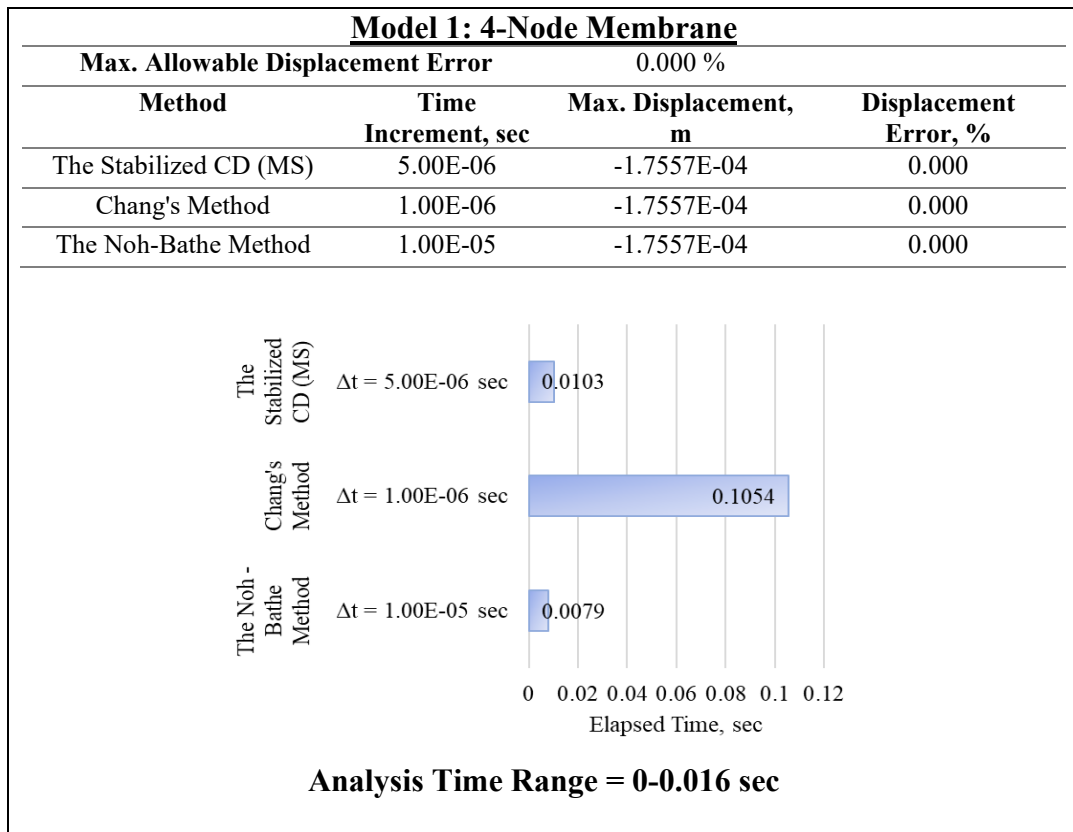


Table 6.6 Solution Time Comparison from Case Study 1 – Model 2

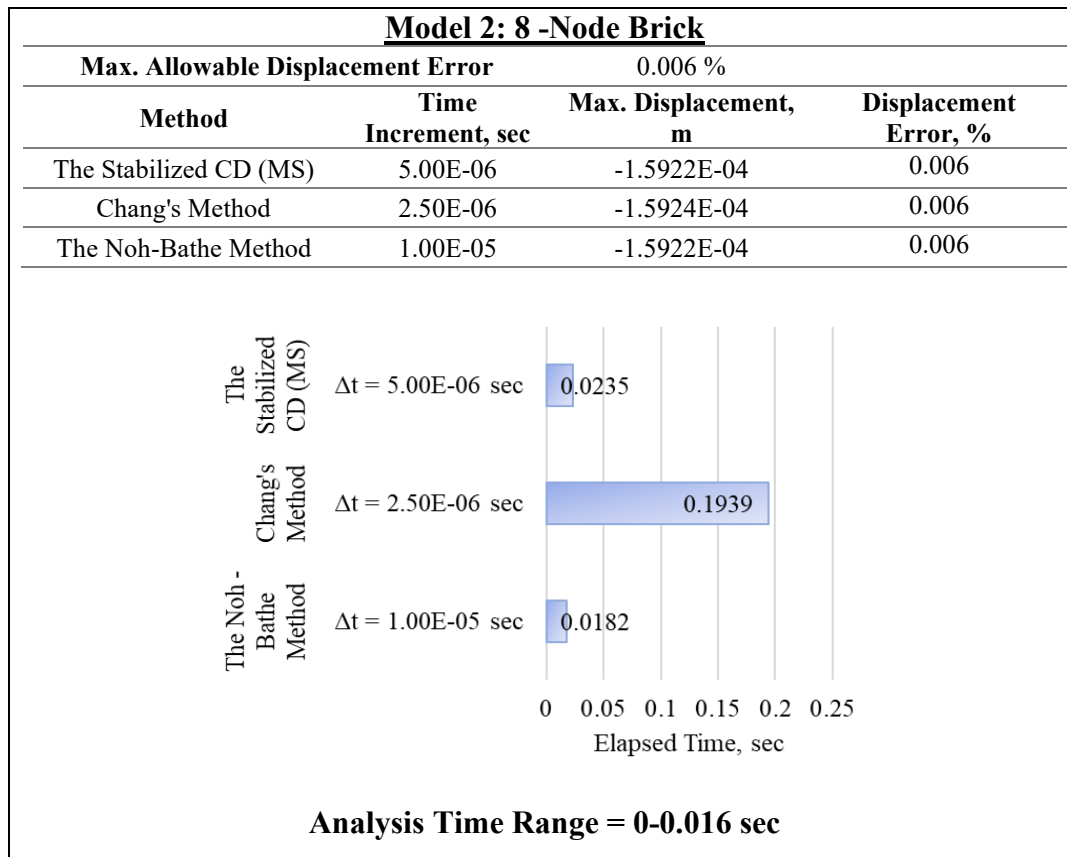


Table 6.5 shows that in Model 1, the allowable maximum displacement error is reached with the Noh-Bathe method by using 2 times and 10 times larger time increments than the stabilized CD method with mass scaling (MS) and Chang’s method, respectively. Table 6.6 shows that in Model 2, the allowable maximum displacement error is reached with the Noh-Bathe method by using 2 times and 4 times larger time increments than the stabilized CD method with mass scaling (MS) and Chang’s method, respectively. Although the stabilized CD method and Chang’s method are stable with large time increments, when high accuracy is aimed, the required time increment for these methods is within the stability limit of the Noh-Bathe method. Table 6.5 and Table 6.6 shows that the Noh-Bathe method is the fastest, and Chang’s method is the slowest.

6.3 Case Study 2: Clamped 3D Solid

In Figure 6.2, a clamped 3D solid problem is presented. The modulus of elasticity and density of the rectangle solid are 28 GPa and 2400 kg/m³, respectively. The size of the rectangle solid is 22 m x 22 m x 2.75 m. A time-dependent loading is applied to the middle node of the rectangle in the x direction. At time equals 1, loading is removed. High-frequency responses are aimed in this in-plane vibration study to see the effect of the change in time increment on the accuracy of the maximum displacement result seen in the analyzed time range.

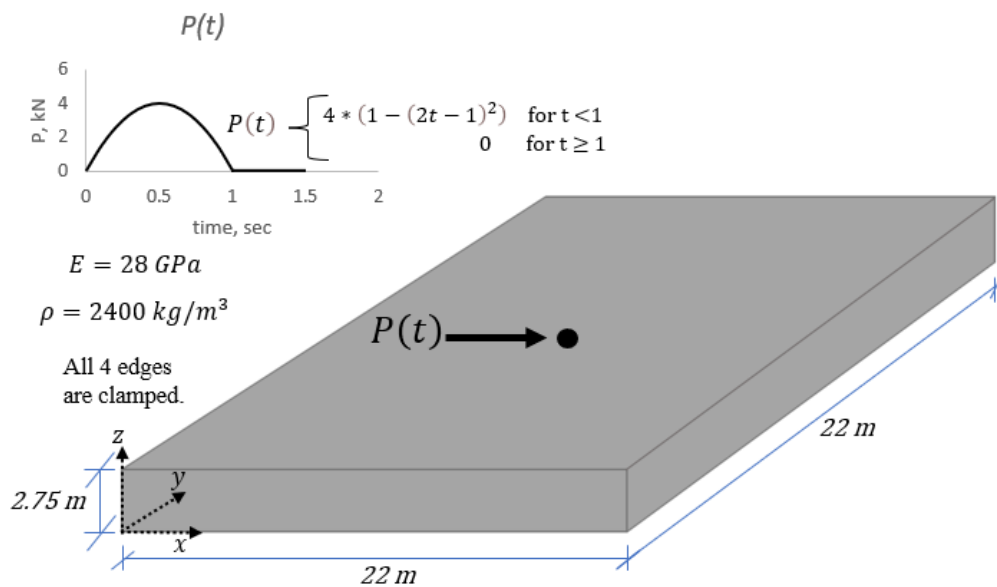
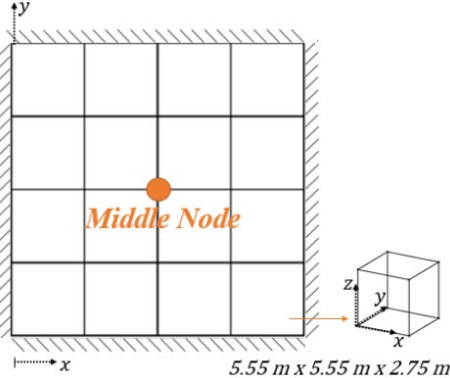
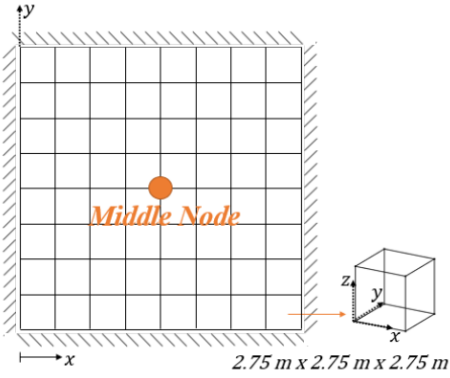
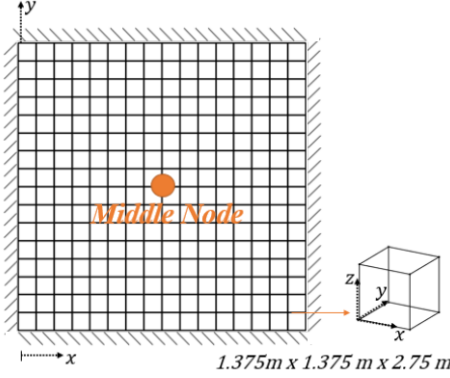


Figure 6.2 Clamped 3D Solid under Vertical Impulse

Discretization of the clamped 3D solid is presented in Table 6.7. The 3D solid is modeled by using 8-node brick elements. In Model 1, 4x4 mesh is used where the size of each element is 5.55 m x 5.55 m x 2.75 m. In Model 2, 8x8 mesh is used where the size of each element is 2.75 m x 2.75 m x 2.75 m. In Model 3, 16x16 mesh is used where the size of each element is 1.375 m x 1.375 m x 2.75 m. Table 6.7 also shows the total number of degrees of freedom, the system's minimum and maximum natural frequency, and the maximum time increment for the accurate reference

results, which corresponds to the minimum period of the system over 10. The goal was to see the effect of the mesh size on the performance of implemented explicit integration algorithms.

Table 6.7 Discretization of Case Study 2

<p style="text-align: center;"><u>Model 1: 4 x 4</u></p> <hr/> <table border="0"> <tr> <td>Number of DOF</td> <td>54</td> <td></td> </tr> <tr> <td>ω_{min}</td> <td>268.3592</td> <td>rad/sec</td> </tr> <tr> <td>ω_{max}</td> <td>2256.6748</td> <td>rad/sec</td> </tr> <tr> <td>$\Delta t_{max\ accurate}$</td> <td>2.7843E-04</td> <td>sec</td> </tr> <tr> <td>$= \frac{T_{min}}{10}$</td> <td></td> <td></td> </tr> </table> <hr/>	Number of DOF	54		ω_{min}	268.3592	rad/sec	ω_{max}	2256.6748	rad/sec	$\Delta t_{max\ accurate}$	2.7843E-04	sec	$= \frac{T_{min}}{10}$			
Number of DOF	54															
ω_{min}	268.3592	rad/sec														
ω_{max}	2256.6748	rad/sec														
$\Delta t_{max\ accurate}$	2.7843E-04	sec														
$= \frac{T_{min}}{10}$																
<p style="text-align: center;"><u>Model 2: 8 x 8</u></p> <hr/> <table border="0"> <tr> <td>Number of DOF</td> <td>294</td> <td></td> </tr> <tr> <td>ω_{min}</td> <td>207.0125</td> <td>rad/sec</td> </tr> <tr> <td>ω_{max}</td> <td>2436.769</td> <td>rad/sec</td> </tr> <tr> <td>$\Delta t_{max\ accurate}$</td> <td>2.5785E-04</td> <td>sec</td> </tr> <tr> <td>$= \frac{T_{min}}{10}$</td> <td></td> <td></td> </tr> </table> <hr/>	Number of DOF	294		ω_{min}	207.0125	rad/sec	ω_{max}	2436.769	rad/sec	$\Delta t_{max\ accurate}$	2.5785E-04	sec	$= \frac{T_{min}}{10}$			
Number of DOF	294															
ω_{min}	207.0125	rad/sec														
ω_{max}	2436.769	rad/sec														
$\Delta t_{max\ accurate}$	2.5785E-04	sec														
$= \frac{T_{min}}{10}$																
<p style="text-align: center;"><u>Model 3: 16 x 16</u></p> <hr/> <table border="0"> <tr> <td>Number of DOF</td> <td>1350</td> <td></td> </tr> <tr> <td>ω_{min}</td> <td>186.4596</td> <td>rad/sec</td> </tr> <tr> <td>ω_{max}</td> <td>4932.5719</td> <td>rad/sec</td> </tr> <tr> <td>$\Delta t_{max\ accurate}$</td> <td>1.2738E-04</td> <td>sec</td> </tr> <tr> <td>$= \frac{T_{min}}{10}$</td> <td></td> <td></td> </tr> </table> <hr/>	Number of DOF	1350		ω_{min}	186.4596	rad/sec	ω_{max}	4932.5719	rad/sec	$\Delta t_{max\ accurate}$	1.2738E-04	sec	$= \frac{T_{min}}{10}$			
Number of DOF	1350															
ω_{min}	186.4596	rad/sec														
ω_{max}	4932.5719	rad/sec														
$\Delta t_{max\ accurate}$	1.2738E-04	sec														
$= \frac{T_{min}}{10}$																

In Table 6.8, stability requirements for implemented explicit integration methods are presented. The a value for the stabilized CD method with mass scaling (MS) is

calculated from the recommended equation (Soares & Großholz, 2018). The calculated a value is always between $[0,0.25]$, meaning that the algorithm is stable for all Δt . The critical time increment for Newmark's explicit method and the Noh-Bathe method are calculated and presented in Table 6.8. Chang's method is unconditionally stable.

Table 6.8 Stability Requirements of the Integration Methods in Case Study 2

	Model 1: 4 x 4	Model 2: 8 x 8	Model 3: 16 x 16
Method	$\Delta t_{critical}$	$\Delta t_{critical}$	$\Delta t_{critical}$
Newmark's Explicit Method	8.86E-04 sec	8.21E-04 sec	4.05E-04 sec
The Stabilized CD Method (MS)	$a \in [0,0.25]$, stable for all Δt	$a \in [0,0.25]$, stable for all Δt	$a \in [0,0.25]$, stable for all Δt
Chang's Method	Unconditionally Stable	Unconditionally Stable	Unconditionally Stable
The Noh-Bathe Method, $p=0.54$	1.66E-03 sec	1.54E-03 sec	7.59E-04 sec

Table 6.7 show that with increased refinement, the number of degrees of freedom is increased and the maximum natural frequency of the system is increased. When the system has higher frequencies due to refined mesh size, the maximum time increment required for accurate results is smaller than in less refined systems. Similarly, Table 6.8 shows that Newmark's explicit method and the Noh-Bathe method require a smaller critical time increment for stability in Model 3, which is the most refined model. Table 6.8 also shows that the critical time increment for the stability of the Noh-Bathe method is approximately 1.9 times larger than Newmark's explicit method.

Table 6.9, Table 6.10, and Table 6.11 show the maximum displacements in the x direction at the "middle node" from all models' test runs. Table 6.9, Table 6.10, and Table 6.11 also present the corresponding time when the maximum displacement in the x direction at the "middle node" is obtained, and the displacement error and phase difference with respect to the reference solution obtained from Newmark's explicit method. For the reference solution, the used time increment is $\Delta t = 1 \times 10^{-4} \text{ sec}$

which is a smaller time step than the time step obtained from the rule of thumb, $\frac{T_{min}}{10}$ presented in Table 6.7. The displacement error is calculated by dividing the difference between the obtained value and the reference value by the reference value, taking the absolute of the ratio, and multiplying it by 100. The phase difference is calculated by taking the absolute difference between the reference value and the obtained value. The displacement response plots from all algorithms for all test runs can be seen in Figure A.3, Figure A.4 and Figure A.5 in Appendix A. In this case study, the modified Noh-Bathe method with mass scaling (MS) is included in the performance comparison. The expectation is obtaining results with this modified algorithm by using time increments which make the classical Noh-Bathe method unstable.

The time history analysis is performed on the developed finite element analysis program between t equal to 0 and t equal to 2 sec. For the accuracy comparison, 8 test runs using different Δt values are performed for Model 1, Model 2, and Model 3. The time increments for test runs of Model 1 and Model 2 are $\Delta t = 1 \times 10^{-4} \text{sec}$, $\Delta t = 2.5 \times 10^{-4} \text{sec}$, $\Delta t = 5 \times 10^{-4} \text{sec}$, $\Delta t = 7.5 \times 10^{-4} \text{sec}$, $\Delta t = 1 \times 10^{-3} \text{sec}$, $\Delta t = 1.5 \times 10^{-3} \text{sec}$, $\Delta t = 2 \times 10^{-3} \text{sec}$ and $\Delta t = 2.5 \times 10^{-3} \text{sec}$ for Test 1, Test 2, Test 3, Test 4, Test 5, Test 6, Test 7, and Test 8, respectively. The time increments for test runs of Model 3 are $\Delta t = 1 \times 10^{-4} \text{sec}$, $\Delta t = 7.5 \times 10^{-4} \text{sec}$, $\Delta t = 8 \times 10^{-4} \text{sec}$, $\Delta t = 8.5 \times 10^{-4} \text{sec}$, $\Delta t = 9 \times 10^{-4} \text{sec}$, $\Delta t = 1 \times 10^{-3} \text{sec}$, $\Delta t = 1.5 \times 10^{-3} \text{sec}$, and $1.75 \times 10^{-3} \text{sec}$ for Test 1, Test 2, Test 3, Test 4, Test 5, Test 6, Test 7, and Test 8 respectively.

Table 6.9 shows that in Model 1 the Noh-Bathe method with mass scaling is stable within the range of the classical Noh-Bathe method. When the mesh density is increased, the effect of mass scaling is observed. Table 6.10 shows that in Model 2, the Noh-Bathe method with mass scaling is stable when using about 1.6 times larger time increment than the critical time increment for the classical Noh-Bathe method. Similarly, Table 6.11 shows that in Model 3, the Noh-Bathe method with mass

scaling is stable when using about 1.1 times larger time increment than the critical time increment for the classical Noh-Bathe method.

Table 6.9 Accuracy Comparison from Case Study 2 – Model 1

Model 1: 4 x 4						
	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error, %	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-04	3.4406E-08	0.4895	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-04	3.4406E-08	0.4895	0.000	0.00E+00
	The Noh - Bathe Method(MS)	1.00E-04	3.4409E-08	0.4895	0.009	0.00E+00
	Chang's Method	1.00E-04	3.4414E-08	0.4899	0.023	4.00E-04
	The Noh - Bathe Method	1.00E-04	3.4409E-08	0.4895	0.009	0.00E+00
Test 2	The Stabilized CD Method(MS)	2.50E-04	3.4396E-08	0.4893	0.029	2.50E-04
	The Noh - Bathe Method(MS)	2.50E-04	3.4414E-08	0.4898	0.023	2.50E-04
	Chang's Method	2.50E-04	3.4409E-08	0.4913	0.009	1.75E-03
	The Noh - Bathe Method	2.50E-04	3.4409E-08	0.4893	0.009	2.50E-04
Test 3	The Stabilized CD Method(MS)	5.00E-04	3.4393E-08	0.4890	0.038	5.00E-04
	The Noh - Bathe Method(MS)	5.00E-04	3.4429E-08	0.4925	0.067	3.00E-03
	Chang's Method	5.00E-04	3.4414E-08	0.5045	0.023	1.50E-02
	The Noh - Bathe Method	5.00E-04	3.4403E-08	0.4885	0.009	1.00E-03
Test 4	The Stabilized CD Method(MS)	7.50E-04	3.4420E-08	0.4913	0.041	1.75E-03
	The Noh - Bathe Method(MS)	7.50E-04	3.4388E-08	0.4868	0.052	2.75E-03
	Chang's Method	7.50E-04	3.4388E-08	0.4973	0.052	7.75E-03
	The Noh - Bathe Method	7.50E-04	3.4394E-08	0.5123	0.035	2.28E-02
Test 5	The Stabilized CD Method(MS)	1.00E-03	3.4425E-08	0.5050	0.055	1.55E-02
	The Noh - Bathe Method(MS)	1.00E-03	3.4374E-08	0.5010	0.093	1.15E-02
	Chang's Method	1.00E-03	3.4427E-08	0.5010	0.061	1.15E-02
	The Noh - Bathe Method	1.00E-03	3.4414E-08	0.5110	0.023	2.15E-02
Test 6	The Stabilized CD Method(MS)	1.50E-03	3.4477E-08	0.5040	0.206	1.45E-02
	The Noh - Bathe Method(MS)	1.50E-03	3.4314E-08	0.4890	0.267	5.00E-04
	Chang's Method	1.50E-03	3.4428E-08	0.5055	0.064	1.60E-02
	The Noh - Bathe Method	1.50E-03	3.4382E-08	0.4980	0.070	8.50E-03
Test 7	The Stabilized CD Method(MS)	2.00E-03	3.4432E-08	0.4860	0.076	3.50E-03
	The Noh - Bathe Method(MS)	2.00E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	2.00E-03	3.4442E-08	0.4840	0.105	5.50E-03
	The Noh - Bathe Method	2.00E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 8	The Stabilized CD Method(MS)	2.50E-03	3.4520E-08	0.4825	0.331	7.00E-03
	The Noh - Bathe Method(MS)	2.50E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	2.50E-03	3.4499E-08	0.4800	0.270	9.50E-03
	The Noh - Bathe Method	2.50E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE

Table 6.10 Accuracy Comparison from Case Study 2 – Model 2

Model 2: 8 x 8						
	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error,%	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-04	4.2799E-08	0.4848	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-04	4.2801E-08	0.4849	0.005	1.00E-04
	The Noh - Bathe Method(MS)	1.00E-04	4.2808E-08	0.4851	0.021	3.00E-04
	Chang's Method	1.00E-04	4.2800E-08	0.4859	0.002	1.10E-03
	The Noh - Bathe Method	1.00E-04	4.2807E-08	0.4850	0.019	2.00E-04
Test 2	The Stabilized CD Method(MS)	2.50E-04	4.2805E-08	0.4965	0.014	1.17E-02
	The Noh - Bathe Method(MS)	2.50E-04	4.2814E-08	0.4860	0.035	1.20E-03
	Chang's Method	2.50E-04	4.2829E-08	0.5078	0.070	2.30E-02
	The Noh - Bathe Method	2.50E-04	4.2802E-08	0.4845	0.007	3.00E-04
Test 3	The Stabilized CD Method(MS)	5.00E-04	4.2817E-08	0.4965	0.042	1.17E-02
	The Noh - Bathe Method(MS)	5.00E-04	4.2804E-08	0.5030	0.012	1.82E-02
	Chang's Method	5.00E-04	4.2817E-08	0.4885	0.042	3.70E-03
	The Noh - Bathe Method	5.00E-04	4.2809E-08	0.4960	0.023	1.12E-02
Test 4	The Stabilized CD Method(MS)	7.50E-04	4.2825E-08	0.5078	0.061	2.30E-02
	The Noh - Bathe Method(MS)	7.50E-04	4.2780E-08	0.4958	0.044	1.10E-02
	Chang's Method	7.50E-04	4.2816E-08	0.5033	0.040	1.85E-02
	The Noh - Bathe Method	7.50E-04	4.2817E-08	0.4950	0.042	1.02E-02
Test 5	The Stabilized CD Method(MS)	1.00E-03	4.2813E-08	0.4990	0.033	1.42E-02
	The Noh - Bathe Method(MS)	1.00E-03	4.2759E-08	0.4960	0.093	1.12E-02
	Chang's Method	1.00E-03	4.2847E-08	0.4980	0.112	1.32E-02
	The Noh - Bathe Method	1.00E-03	4.2818E-08	0.4940	0.044	9.20E-03
Test 6	The Stabilized CD Method(MS)	1.50E-03	4.2821E-08	0.4905	0.051	5.70E-03
	The Noh - Bathe Method(MS)	1.50E-03	4.2713E-08	0.4920	0.201	7.20E-03
	Chang's Method	1.50E-03	4.2837E-08	0.5025	0.089	1.77E-02
	The Noh - Bathe Method	1.50E-03	4.2791E-08	0.4920	0.019	7.20E-03
Test 7	The Stabilized CD Method(MS)	2.00E-03	4.2849E-08	0.5000	0.117	1.52E-02
	The Noh - Bathe Method(MS)	2.00E-03	4.2701E-08	0.4940	0.229	9.20E-03
	Chang's Method	2.00E-03	4.2856E-08	0.5000	0.133	1.52E-02
	The Noh - Bathe Method	2.00E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 8	The Stabilized CD Method(MS)	2.50E-03	4.2892E-08	0.5025	0.217	1.77E-02
	The Noh - Bathe Method(MS)	2.50E-03	4.2704E-08	0.4950	0.222	1.02E-02
	Chang's Method	2.50E-03	4.2905E-08	0.5100	0.248	2.52E-02
	The Noh - Bathe Method	2.50E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE

Table 6.11 Accuracy Comparison from Case Study 2 – Model 3

Model 3: 16 x 16						
	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error,%	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-04	5.1353E-08	0.4893	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-04	5.1350E-08	0.4896	0.006	3.00E-04
	The Noh - Bathe Method(MS)	1.00E-04	5.1352E-08	0.4896	0.002	3.00E-04
	Chang's Method	1.00E-04	5.1353E-08	0.4898	0.000	5.00E-04
	The Noh - Bathe Method	1.00E-04	5.1351E-08	0.4996	0.004	1.03E-02
Test 2	The Stabilized CD Method(MS)	7.50E-04	5.1371E-08	0.5040	0.035	1.47E-02
	The Noh - Bathe Method(MS)	7.50E-04	5.1303E-08	0.5025	0.097	1.32E-02
	Chang's Method	7.50E-04	5.1397E-08	0.4965	0.086	7.20E-03
	The Noh - Bathe Method	7.50E-04	5.1378E-08	0.4988	0.049	9.45E-03
Test 3	The Stabilized CD Method(MS)	8.00E-04	5.1364E-08	0.5064	0.021	1.71E-02
	The Noh - Bathe Method(MS)	8.00E-04	5.1299E-08	0.4992	0.105	9.90E-03
	Chang's Method	8.00E-04	5.1365E-08	0.4872	0.023	2.10E-03
	The Noh - Bathe Method	8.00E-04	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 4	The Stabilized CD Method(MS)	8.50E-04	5.1367E-08	0.4854	0.027	3.95E-03
	The Noh - Bathe Method(MS)	8.50E-04	5.1292E-08	0.4956	0.119	6.25E-03
	Chang's Method	8.50E-04	5.1367E-08	0.4981	0.027	8.80E-03
	The Noh - Bathe Method	8.50E-04	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 5	The Stabilized CD Method(MS)	9.00E-04	5.1385E-08	0.4959	0.062	6.60E-03
	The Noh - Bathe Method(MS)	9.00E-04	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	9.00E-04	5.1388E-08	0.4995	0.068	1.02E-02
	The Noh - Bathe Method	9.00E-04	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 6	The Stabilized CD Method(MS)	1.00E-03	5.1380E-08	0.4990	0.053	9.70E-03
	The Noh - Bathe Method(MS)	1.00E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	1.00E-03	5.1366E-08	0.5020	0.025	1.27E-02
	The Noh - Bathe Method	1.00E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 7	The Stabilized CD Method(MS)	1.50E-03	5.1373E-08	0.5040	0.039	1.47E-02
	The Noh - Bathe Method(MS)	1.50E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	1.50E-03	5.1408E-08	0.4965	0.107	7.20E-03
	The Noh - Bathe Method	1.50E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
Test 8	The Stabilized CD Method(MS)	1.75E-03	5.1403E-08	0.4935	0.097	4.20E-03
	The Noh - Bathe Method(MS)	1.75E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
	Chang's Method	1.75E-03	5.1398E-08	0.5058	0.088	1.65E-02
	The Noh - Bathe Method	1.75E-03	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE

The solution time of algorithms is compared by using the largest possible time increments that give the closest maximum displacement error to the selected allowable maximum displacement error. The analyzed time range is kept fixed for all algorithms between 0-2 sec. The solution time comparison for case study 2 is presented in Table 6.12, Table 6.13, and Table 6.14.

Table 6.12 Solution Time Comparison from Case Study 2 – Model 1

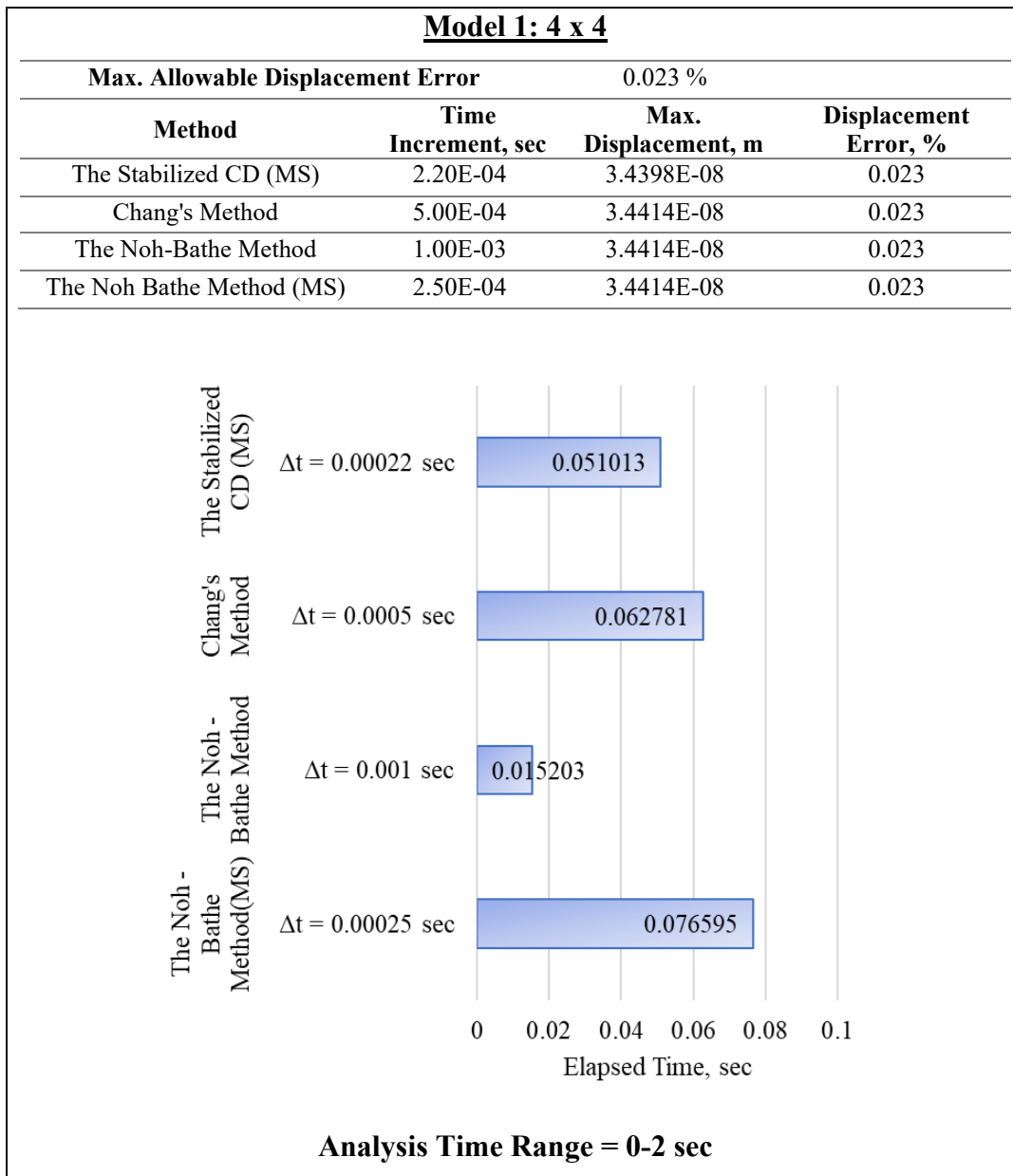


Table 6.13 Solution Time Comparison from Case Study 2 – Model 2

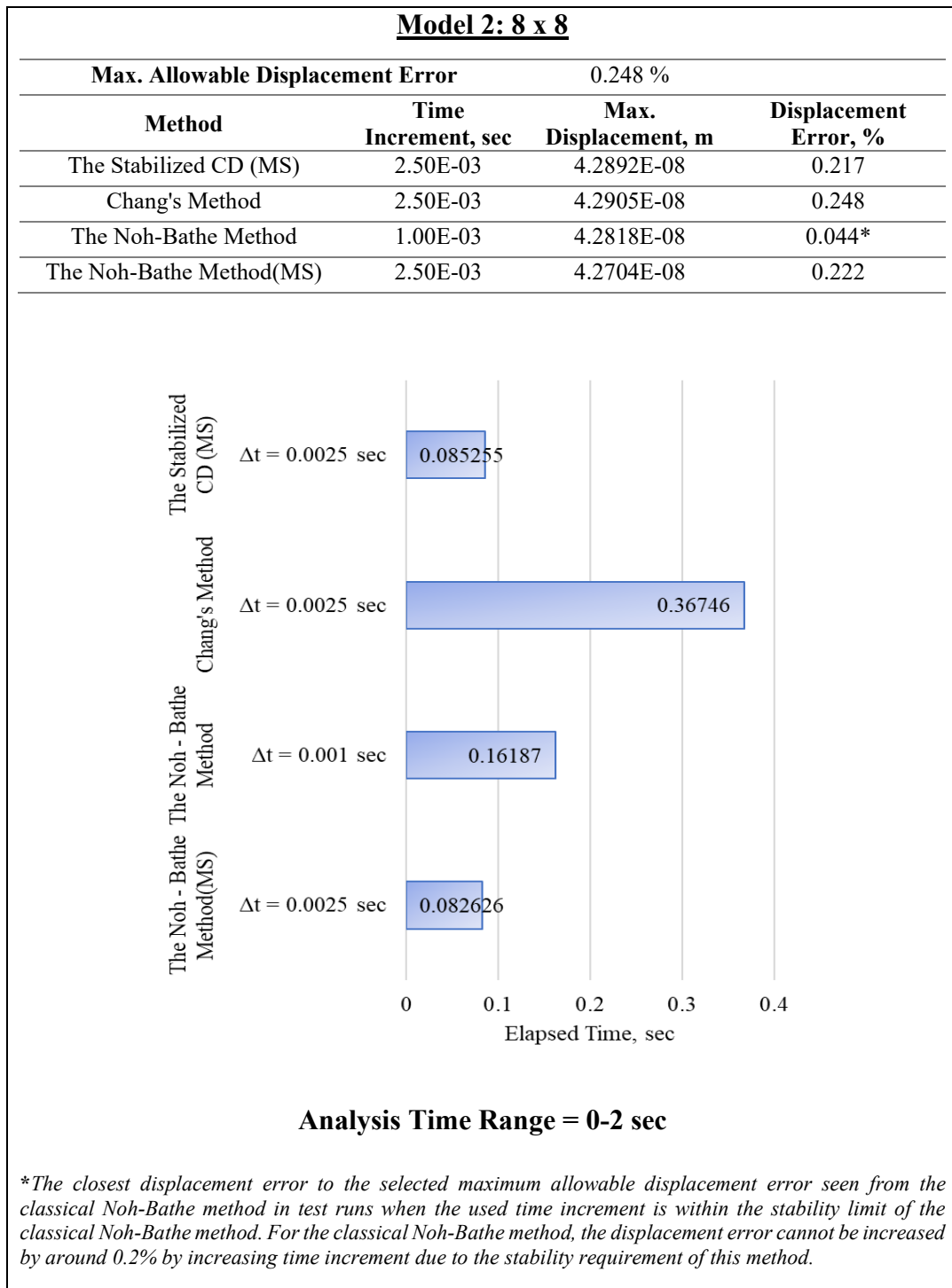


Table 6.14 Solution Time Comparison from Case Study 2 – Model 3

Model 3: 16 x 16			
Max. Allowable Displacement Error		0.107 %	
Method	Time Increment, sec	Max. Displacement, m	Displacement Error, %
The Stabilized CD (MS)	1.75E-03	5.1403E-08	0.097
Chang's Method	1.50E-03	5.1408E-08	0.107
The Noh-Bathe Method	7.50E-04	5.1378E-08	0.049*
The Noh-Bathe Method(MS)	8.00E-04	5.1299E-08	0.105

Method	Time Increment (Δt), sec	Elapsed Time, sec
The Stabilized CD (MS)	0.00175	3.6434
Chang's Method	0.0015	22.2177
The Noh - Bathe Method	0.00075	5.7335
The Noh - Bathe Method(MS)	0.0008	10.325

Analysis Time Range = 0-2 sec

**The closest displacement error to the selected maximum allowable displacement error seen from the classical Noh-Bathe method in test runs when the used time increment is within the stability limit of the classical Noh-Bathe method. For the classical Noh-Bathe method, the displacement error cannot be increased by around 0.1% by increasing time increment due to the stability requirement of this method.*

Table 6.12 shows that in Model 1 when the target is the high accuracy level, around 0.02%, the fastest algorithm is the classical Noh-Bathe Method. To reach this high accuracy level, algorithms with mass scaling require smaller time increments. Moreover, the non-diagonal scaled mass matrix increases the computation time. Therefore, both stabilized CD method with mass scaling (MS) and the Noh-Bathe method with mass scaling (MS) are slower than the classical Noh-Bathe Method, when high accuracy level is expected.

Table 6.13 shows that in Model 2 the target accuracy level is around ten times lower than in Model 1, around 0.2%. When a relatively low accuracy level is selected in Model 2, larger time increments can be used for the algorithms with mass scaling and Chang's method. The classical Noh-Bathe method cannot show the selected low accuracy level since the time increment cannot be increased due to the stability requirement of this method. When a lower accuracy level is preferred, the solution times of the stabilized CD method and the Noh-Bathe method with mass scaling are very close and the fastest.

Table 6.14 shows that Model 3 is the most refined model among created 3 models. The accuracy change with increased time increment is seen more clearly in this model. In Model 3, for the target accuracy level, the solution time is 5 times lower than in Model 1. In the less refined model, Model 2, a similar accuracy level is seen from all algorithms using the same time increment. However, in the more refined model, Model 3, the selected accuracy level is reached in the stabilized CD method with mass scaling using around 2.2 times larger time increment than the Noh-Bathe method with mass scaling. Also, the selected accuracy level is seen in Chang's method by using around 1.9 times larger time increment than the Noh-Bathe method with mass scaling. Similar to the results of Model 2, a larger time increment cannot be utilized for the classical Noh-Bathe method for a lower accuracy level since the time increment cannot be further increased due to the stability requirement of this method. In Model 3, the stabilized CD method with mass scaling is the fastest algorithm.

6.4 Case Study 3: Moving Load on Three-Span Road Bridge

Figure 6.3 presents a three-span road bridge problem under 30 m/sec moving load with a magnitude of 5 kN.

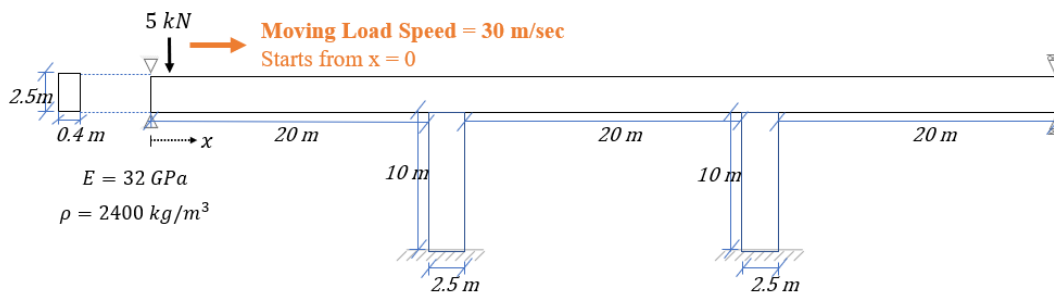
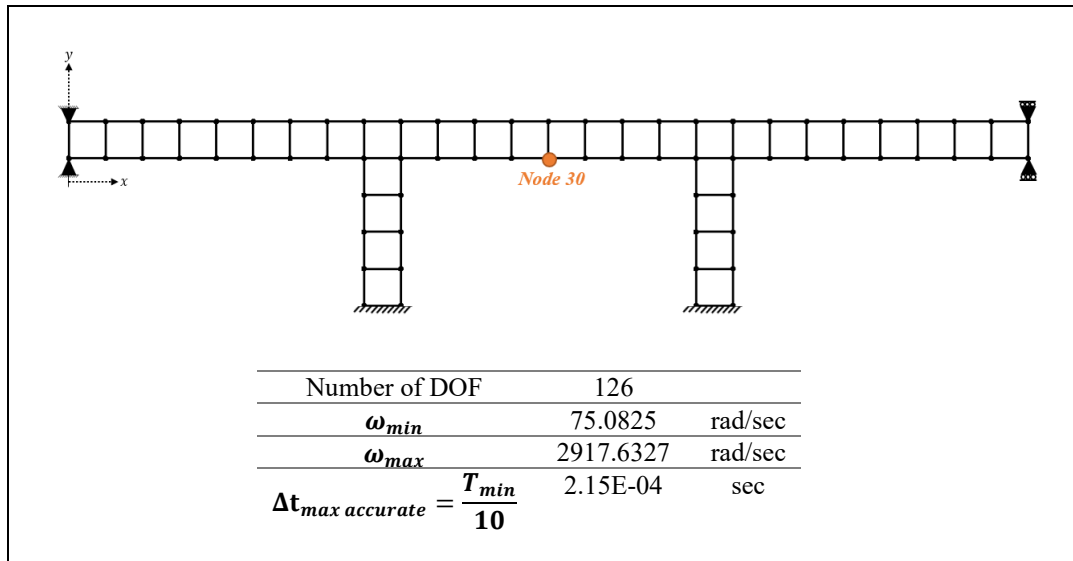


Figure 6.3: Three-Span Road Bridge Problem

Table 6.15 presents the discretization of the three-span road bridge with quadrilateral membranes. 2.5 m x 2.5 m quadrilateral elements are used to mesh the structure model. The “node 30” is located at $x = 32.5 \text{ m}$. Table 6.15 also shows the total number of degrees of freedom, the system’s minimum and maximum natural frequencies, and the maximum time increment for the accurate reference results, which corresponds to the minimum period of the system over 10. This case study aims to see the performance of algorithms in a real-life problem.

Table 6.15 Discretization of Case Study 3



In Table 6.16, stability requirements for implemented explicit integration methods are presented. The a value for the stabilized CD method with mass scaling (MS) is calculated from the recommended equation (Soares & Großholz, 2018). The calculated a value is always between $[0,0.25]$, meaning that the algorithm is stable for all Δt . The critical time increment for Newmark's explicit method and the Noh-Bathe method are calculated and presented in Table 6.16. Chang's method is unconditionally stable.

Table 6.16 Stability Requirements of the Integration Methods in Case Study 3

Method	$\Delta t_{critical}$
Newmark's Explicit Method	6.85E-04 sec
The Stabilized CD Method (MS)	$a \in [0,0.25]$, stable for all Δt
Chang's Method	Unconditionally Stable
The Noh-Bathe Method, $p=0.54$	1.28E-03sec

Table 6.16 shows that the critical time increment for the stability of the Noh-Bathe method is approximately 1.9 times larger than Newmark's explicit method.

In Table 6.17, maximum displacements in the x direction at “node 30” observed within the analyzed time range from all test runs are reported for all models. Table 6.17 also shows the corresponding time when the maximum displacement in the x direction at “node 30” is observed, and the displacement error and phase difference with respect to the accurate reference solution obtained from Newmark’s explicit method. For the reference solution, the used time increment is $\Delta t = 1 \times 10^{-5} \text{ sec}$ which is a smaller time step than the time step obtained from the rule of thumb, $\frac{T_{min}}{10}$ presented in Table 6.15. The displacement error is calculated by dividing the difference between the obtained value and the reference value by the reference value, taking the absolute of the ratio, and multiplying it by 100. The phase difference is calculated by taking the absolute difference between the reference value and the obtained value. The displacement response plots from all algorithms for all test runs can be seen in Figure A.6 and Figure A.7 in Appendix A.

In this case study, Newmark’s explicit method is included in the performance comparison. The high accuracy is the target, therefore, the tested time increments within the stability of Newmark’s method. The time history analysis is performed on the developed finite element analysis program between t equal to 0 and t equal to 2 sec. For the accuracy comparison, 13 test runs using different Δt values are performed. The time increment used in the first test run is $\Delta t = 1 \times 10^{-5} \text{ sec}$ and the used time increment is increased $1 \times 10^{-5} \text{ sec}$ for each test run up to Test 10 and it is increased $1 \times 10^{-4} \text{ sec}$ between Test 11 and Test 13.

Table 6.17 Accuracy Comparison of Algorithms from Case Study 3

	Method	Time Increment, sec	Max. Displacement, m	Corresponding Time, sec	Displacement Error, %	Phase Difference, sec
	Newmark's Explicit(REF)	1.00E-05	2.0771E-06	1.7344	REFERENCE	REFERENCE
Test 1	The Stabilized CD Method(MS)	1.00E-05	2.0770E-06	1.7344	0.005	0.00E+00
	Chang's Method	1.00E-05	2.0769E-06	1.7344	0.010	0.00E+00
	The Noh - Bathe Method	1.00E-05	2.0770E-06	1.7344	0.005	0.00E+00
Test 2	Newmark's Explicit	2.00E-05	2.0772E-06	1.7344	0.005	0.00E+00
	The Stabilized CD Method(MS)	2.00E-05	2.0772E-06	1.7344	0.005	0.00E+00
	Chang's Method	2.00E-05	2.0765E-06	1.7344	0.029	0.00E+00
Test 3	The Noh - Bathe Method	2.00E-05	2.0770E-06	1.7344	0.005	0.00E+00
	Newmark's Explicit	3.00E-05	2.0775E-06	1.7343	0.019	1.00E-04
	The Stabilized CD Method(MS)	3.00E-05	2.0775E-06	1.7343	0.019	1.00E-04
Test 4	Chang's Method	3.00E-05	2.0759E-06	1.7345	0.058	1.00E-04
	The Noh - Bathe Method	3.00E-05	2.0771E-06	1.7344	0.000	0.00E+00
	Newmark's Explicit	4.00E-05	2.0778E-06	1.7343	0.034	1.00E-04
Test 5	The Stabilized CD Method(MS)	4.00E-05	2.0778E-06	1.7343	0.034	1.00E-04
	Chang's Method	4.00E-05	2.0748E-06	1.7346	0.111	2.00E-04
	The Noh - Bathe Method	4.00E-05	2.0771E-06	1.7343	0.000	1.00E-04
Test 6	Newmark's Explicit	5.00E-05	2.0783E-06	1.7342	0.058	2.00E-04
	The Stabilized CD Method(MS)	5.00E-05	2.0782E-06	1.7343	0.053	1.00E-04
	Chang's Method	5.00E-05	2.0732E-06	1.7347	0.188	3.00E-04
Test 7	The Noh - Bathe Method	5.00E-05	2.0772E-06	1.7343	0.005	1.00E-04
	Newmark's Explicit	6.00E-05	2.0788E-06	1.7341	0.082	3.00E-04
	The Stabilized CD Method(MS)	6.00E-05	2.0786E-06	1.7342	0.072	2.00E-04
Test 8	Chang's Method	6.00E-05	2.0748E-06	1.7333	0.111	1.10E-03
	The Noh - Bathe Method	6.00E-05	2.0773E-06	1.7343	0.010	1.00E-04
	Newmark's Explicit	7.00E-05	2.0792E-06	1.7340	0.101	4.00E-04
Test 9	The Stabilized CD Method(MS)	7.00E-05	2.0790E-06	1.7341	0.091	3.00E-04
	Chang's Method	7.00E-05	2.0765E-06	1.7335	0.029	9.00E-04
	The Noh - Bathe Method	7.00E-05	2.0774E-06	1.7342	0.014	2.00E-04
Test 10	Newmark's Explicit	8.00E-05	2.0793E-06	1.7339	0.106	5.00E-04
	The Stabilized CD Method(MS)	8.00E-05	2.0793E-06	1.7340	0.106	4.00E-04
	Chang's Method	8.00E-05	2.0770E-06	1.7338	0.005	6.00E-04
Test 11	The Noh - Bathe Method	8.00E-05	2.0775E-06	1.7342	0.019	2.00E-04
	Newmark's Explicit	9.00E-05	2.0790E-06	1.7338	0.091	6.00E-04
	The Stabilized CD Method(MS)	9.00E-05	2.0793E-06	1.7338	0.106	6.00E-04
Test 12	Chang's Method	9.00E-05	2.0765E-06	1.7339	0.029	5.00E-04
	The Noh - Bathe Method	9.00E-05	2.0776E-06	1.7342	0.024	2.00E-04
	Newmark's Explicit	1.00E-04	2.0782E-06	1.7336	0.053	8.00E-04
Test 13	The Stabilized CD Method(MS)	1.00E-04	2.0791E-06	1.7338	0.096	6.00E-04
	Chang's Method	1.00E-04	2.0768E-06	1.7340	0.014	4.00E-04
	The Noh - Bathe Method	1.00E-04	2.0777E-06	1.7342	0.029	2.00E-04
Test 14	Newmark's Explicit	2.00E-04	2.0799E-06	1.7330	0.135	1.40E-03
	The Stabilized CD Method(MS)	2.00E-04	2.0812E-06	1.7336	0.197	8.00E-04
	Chang's Method	2.00E-04	2.0751E-06	1.7328	0.096	1.60E-03
Test 15	The Noh - Bathe Method	2.00E-04	2.0788E-06	1.7338	0.082	6.00E-04
	Newmark's Explicit	3.00E-04	2.0816E-06	1.7325	0.217	1.90E-03
	The Stabilized CD Method(MS)	3.00E-04	2.0797E-06	1.7334	0.125	1.00E-03
Test 16	Chang's Method	3.00E-04	2.0825E-06	1.7337	0.260	7.00E-04
	The Noh - Bathe Method	3.00E-04	2.0792E-06	1.7337	0.101	7.00E-04
	Newmark's Explicit	4.00E-04	2.0698E-06	1.7380	0.351	3.60E-03
Test 17	The Stabilized CD Method(MS)	4.00E-04	2.0804E-06	1.7332	0.159	1.20E-03
	Chang's Method	4.00E-04	2.0840E-06	1.7340	0.332	4.00E-04
	The Noh - Bathe Method	4.00E-04	2.0797E-06	1.7332	0.125	1.20E-03

The solution time of algorithms is compared by using the largest possible time increments that give closer maximum displacement error to the selected allowable maximum displacement error. The analyzed time range is kept fixed for all algorithms between 0-2 sec. The solution time comparison for case study 3 is

presented in Table 6.18. In this case study, Newmark’s explicit method is also included in the solution time comparison.

Table 6.18 Solution Time Comparison from Case Study 3

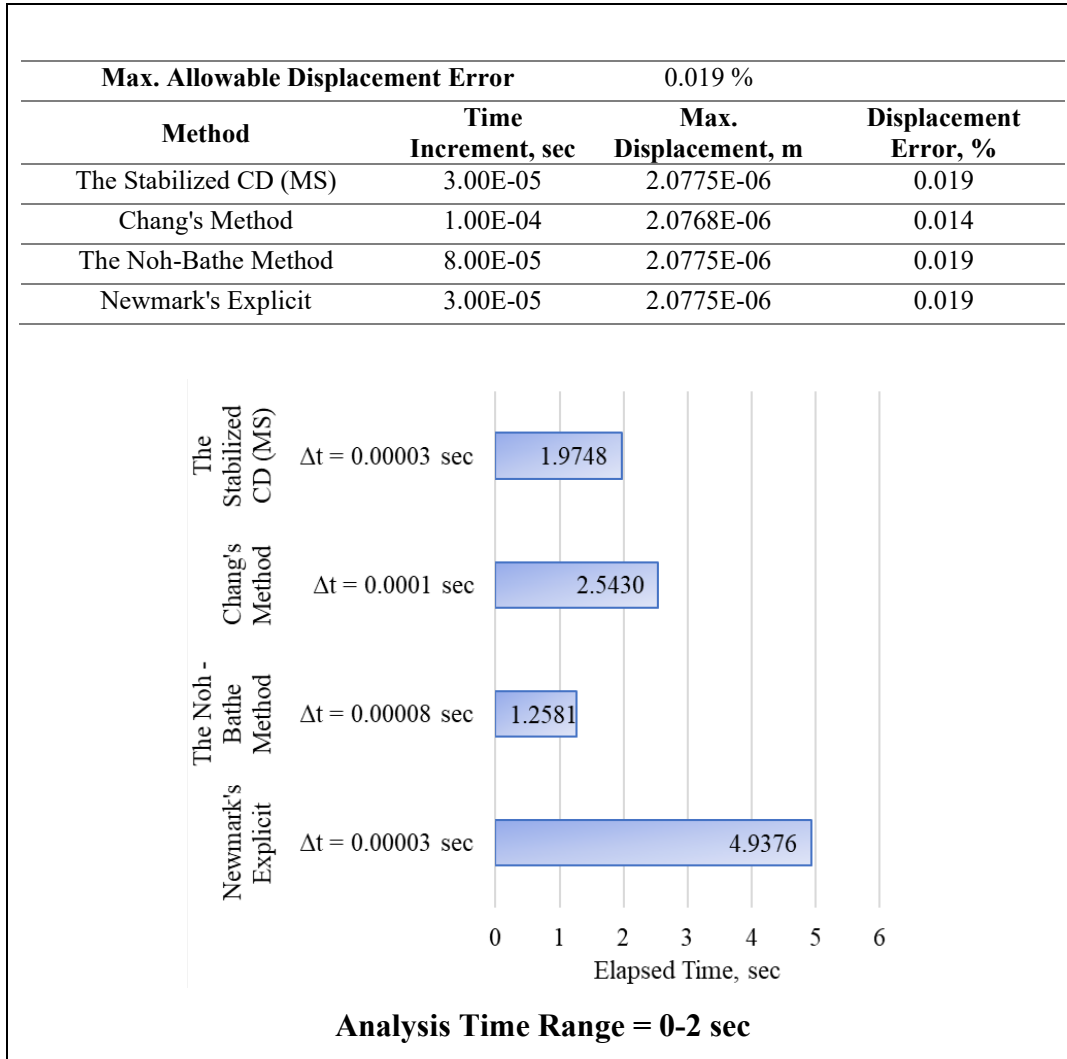


Table 6.18 shows that the largest time increment used to obtain the selected accuracy level is for Chang’s method, yet, the largest used time increment in this comparison can make this algorithm faster than Newmark’s explicit method but not make it faster than other methods. The selected accuracy level is a high-accuracy level and this causes a smaller time step required for the stabilized CD method with mass scaling

(MS) than the Noh-Bathe method. Consequently, the Noh-Bathe method is a faster algorithm than the stabilized CD method for high accuracy level. Table 6.18 also shows that a smaller time step is required for Newmark’s explicit method than the Noh-Bathe method to achieve the same accuracy level. For the preferred accuracy level, Newmark’s method is approximately 3.9 times slower than the Noh-Bathe method.

6.5 Summary of Results

The summary of the solution times obtained from all case studies is presented in Table 6.19. Table 6.19 shows that when a low level of accuracy is preferred (smaller than 0.023%), the Noh-Bathe method is the fastest algorithm. When a higher displacement error is required (smaller than 0.25%), the stabilized CD method is the fastest algorithm.

Table 6.19 Summary of Solution Times

Case Study	Fastest Method	Solution Time, sec	Max. Allowable Displacement Error, %
Case Study 1 – Model 1	The Noh-Bathe Method	0.0079	0.000
Case Study 1 – Model 2	The Noh-Bathe Method	0.0182	0.006
Case Study 2 – Model 1	The Noh-Bathe Method	0.0152	0.023
Case Study 2 – Model 2	The Stabilized CD (MS)	0.0852	0.248
	----- The Noh-Bathe Method (MS)	0.0826	
Case Study 2 – Model 3	The Stabilized CD (MS)	3.6434	0.107
Case Study 3	The Noh-Bathe Method	1.2581	0.019

The summary of the stability requirements obtained from all case studies is presented in Table 6.20. Table 6.20 shows that the critical time increment for the stability of the Noh-Bathe method is approximately 1.9 times larger than Newmark’s explicit method in all case studies. Also for all case studies, Chang’s method is unconditionally stable and the stabilized CD method is stable for all time increments because the stability-related parameter, α , is calculated with suggested equation (Soares & Großholz, 2018) and it is always between 0 and 0.25.

Table 6.20 Summary of Stability Requirements

Case Study	ω_{max} , rad/sec	$\Delta t_{critical}$, sec	
		Newmark's Explicit Method	The Noh-Bathe Method, $\rho=0.54$
Case Study 1 – Model 1	19945.09	1.00E-04	1.88E-04
Case Study 1 – Model 2	49888.35	4.01E-05	7.51E-05
Case Study 2 – Model 1	2256.67	8.86E-04	1.66E-03
Case Study 2 – Model 2	2436.77	8.21E-04	1.54E-03
Case Study 2 – Model 3	4932.57	4.05E-04	7.59E-04
Case Study 3	2917.63	6.85E-04	1.28E-03

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

The main conclusions of this study within the view of the test cases examined are summarized below:

- The best performance in terms of solution time is observed in the Noh-Bathe method when high-level accuracy is required for all the test cases. Although the stabilized CD method with mass scaling and Chang's method is stable for large time increments, using larger time increments did not produce highly accurate results. In fact, in the case studies, for the same level of accuracy, the smaller time step is required for the stabilized CD method with mass scaling when compared with the Noh-Bathe method.
- The stability limit of the classical Noh-Bathe method can be increased with mass scaling proposed by Soares & Großholz (2018). The classical Noh-Bathe method gives highly accurate results within its stability limit. When a low accuracy level is preferred, the Noh-Bathe method with mass scaling, the stabilized CD method with mass scaling, and Chang's method can be used with larger time increments. The stabilized CD method is the fastest algorithm for a low-accuracy target.
- All algorithms are robust and have a consistent convergence rate. They all produce acceptable results for different types of structures and loading cases.

7.2 Future Work

In this study, the solution algorithms are implemented in a way to simulate the nonlinear solution procedure by including terms with stiffness matrix into the time stepping loop. This approach gave an idea about the solution methods in terms of solution times in the case of a nonlinear problem where the stiffness matrix has to be updated at each time step.

Solution accuracy and the range of stability of the methods can change when the problem is nonlinear. Chang's method is unconditionally stable with only a certain degree of nonlinearity (Chang, 2010). To compare all methods, actual nonlinear benchmark problems should be analyzed in terms of accuracy and stability in nonlinear cases.

REFERENCES

- Askes, H., Nguyen, D. C., & Tyas, A. (2011). Increasing the critical time step: micro-inertia, inertia penalties and mass scaling. *Computational Mechanics*, 657-667.
- Chang, S. Y. (2002). Explicit pseudodynamic algorithm with unconditional stability. *Journal of Engineering Mechanics*, 935-947.
- Chang, S. Y. (2010). Nonlinear Performance of Explicit Pseudodynamic Algorithms. *Journal of Earthquake Engineering*, 211-230.
- Chopra, A. K. (2012). *Dynamics of Structures*. Pearson Education.
- Cook, R. D. (2007). *Concepts and applications of finite element analysis*. John Wiley & Sons.
- Fung, T.C. (2003). Numerical dissipation in time-step integration algorithms for structural dynamic analysis. *Progress in Structural Engineering and Materials*, 167-180.
- Kim, Wooram, Jin Ho Lee. (2018). An improved explicit time integration method for linear and nonlinear structural dynamics. *Computers & Structures*, 42-53.
- Li, J., Yu, K., & Li, X. (2021). An identical second-order single step explicit integration algorithm with dissipation control for structural dynamics. *International Journal for Numerical Methods in Engineering*, 1089-1132.
- Macek, R. W., & Aubert, B. H. (1995). A mass penalty technique to control the critical time increment in explicit dynamic finite element analyses. *Earthquake engineering & structural dynamics*, 1315-1331.
- Noh, G, & Bathe, K. J. (2013). An explicit time integration scheme for the analysis of wave propagations. *Computers & Structures*, 178-193.

- Soares Jr, D., & Großholz, G. (2018). Nonlinear structural dynamic analysis by a stabilized central difference method. *Engineering Structures*, 383-392.
- Song, C., Eisenträger, S., & Zhang, X. (2022). High-order implicit time integration scheme based on Padé expansions. *Computer Methods in Applied Mechanics and Engineering*.
- Yang, C., Li, Q., & Xiao, S. (2020). Non-iterative explicit integration algorithms based on acceleration time history for nonlinear dynamic systems. *Archive of Applied Mechanics*, 397-41.
- Zhang, H., Zhang, R., Zannoni, A., Xing, Y., & Masarati, P. (2022). A novel explicit three-sub-step time integration method for wave propagation problems. *Archive of Applied Mechanics*, 821-852.

A. Appendix - Displacement Response Plots

Displacement Responses of Case Study 1 - Model 1 are presented in Figure A.1.

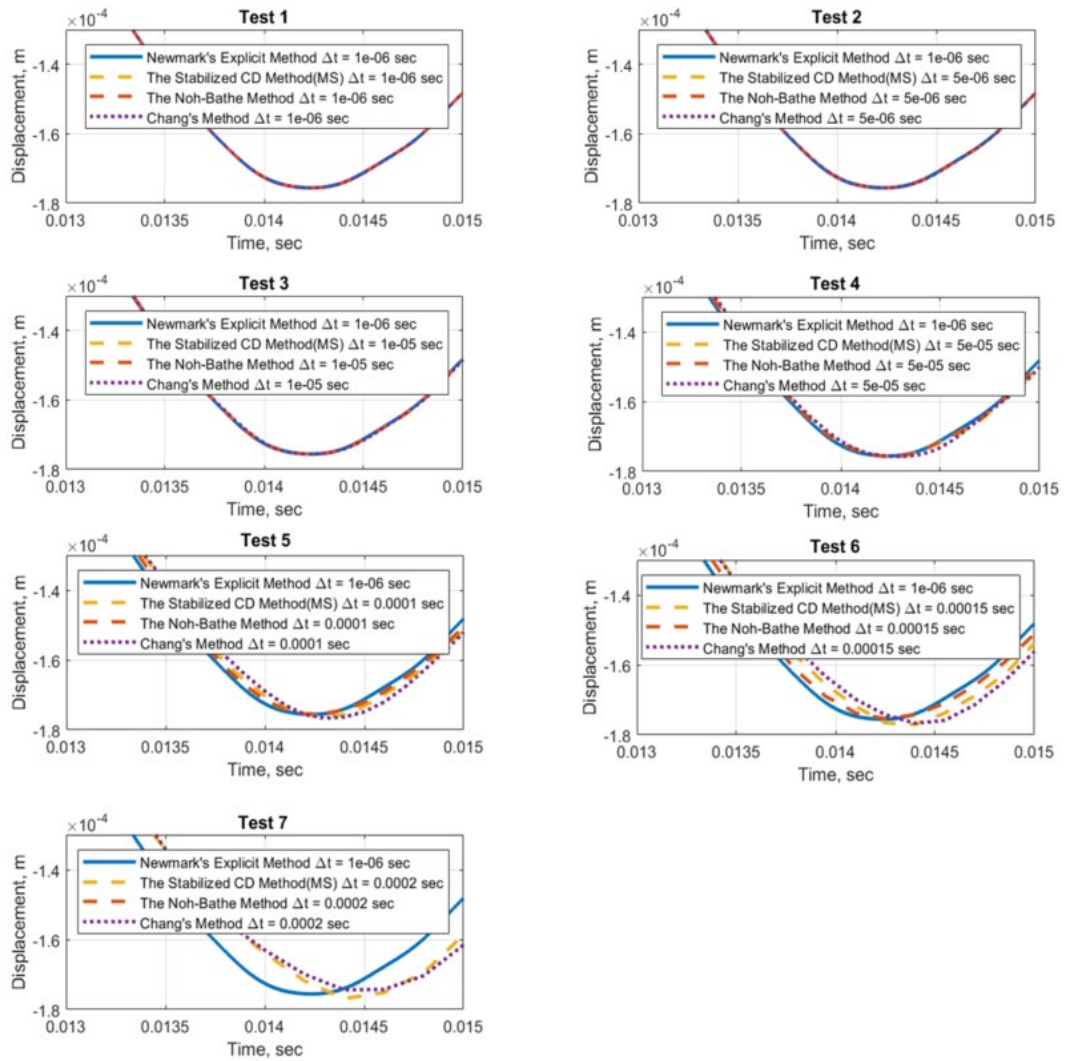


Figure A.1 Displacement Responses of the Case Study 1 - Model 1

Displacement Responses of Case Study 1 - Model 2 are presented in Figure A.2.

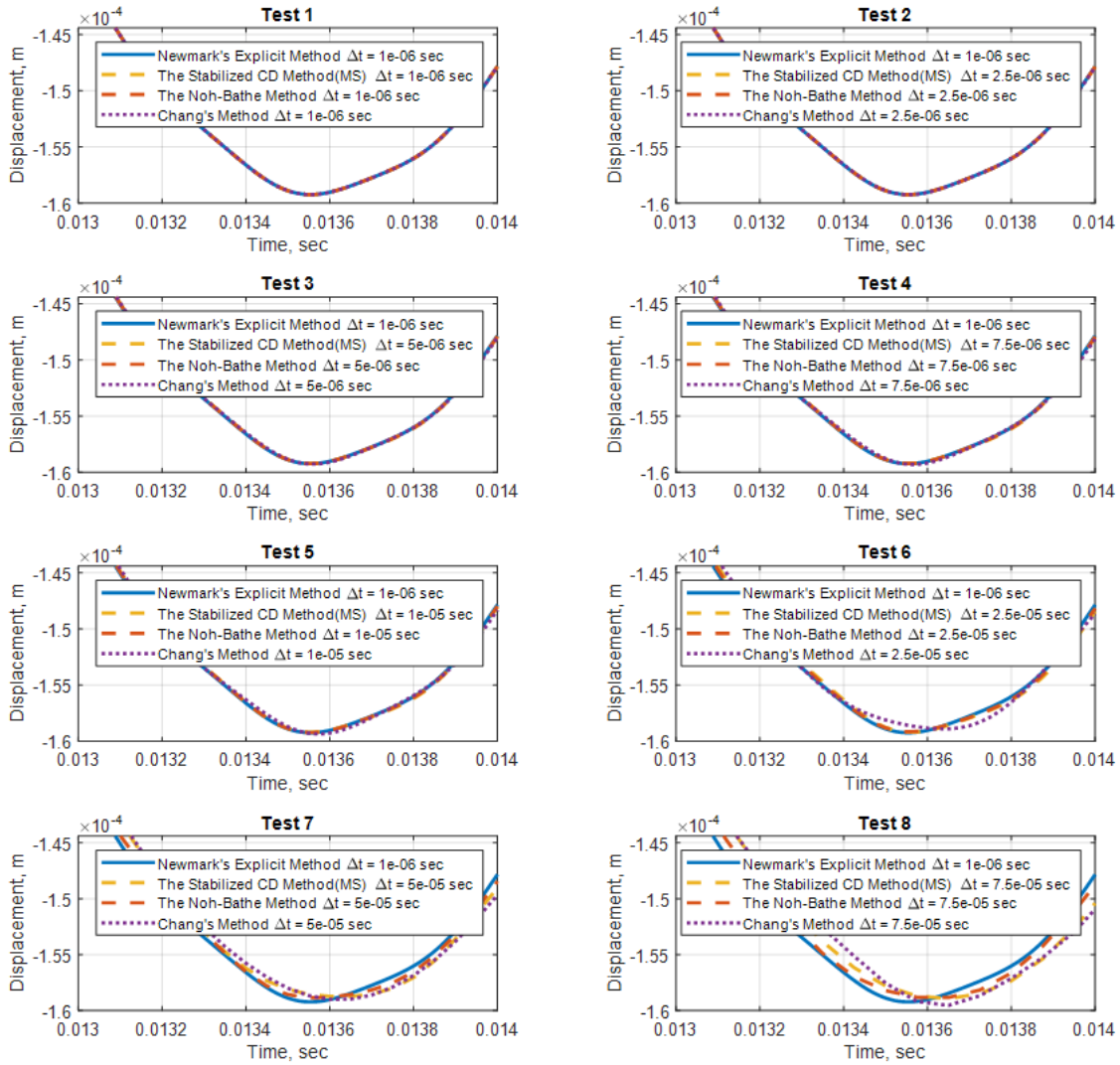


Figure A.2 Displacement Responses of the Case Study 1 - Model 2

Displacement Responses of Case Study 2 - Model 1 are presented in Figure A.3.

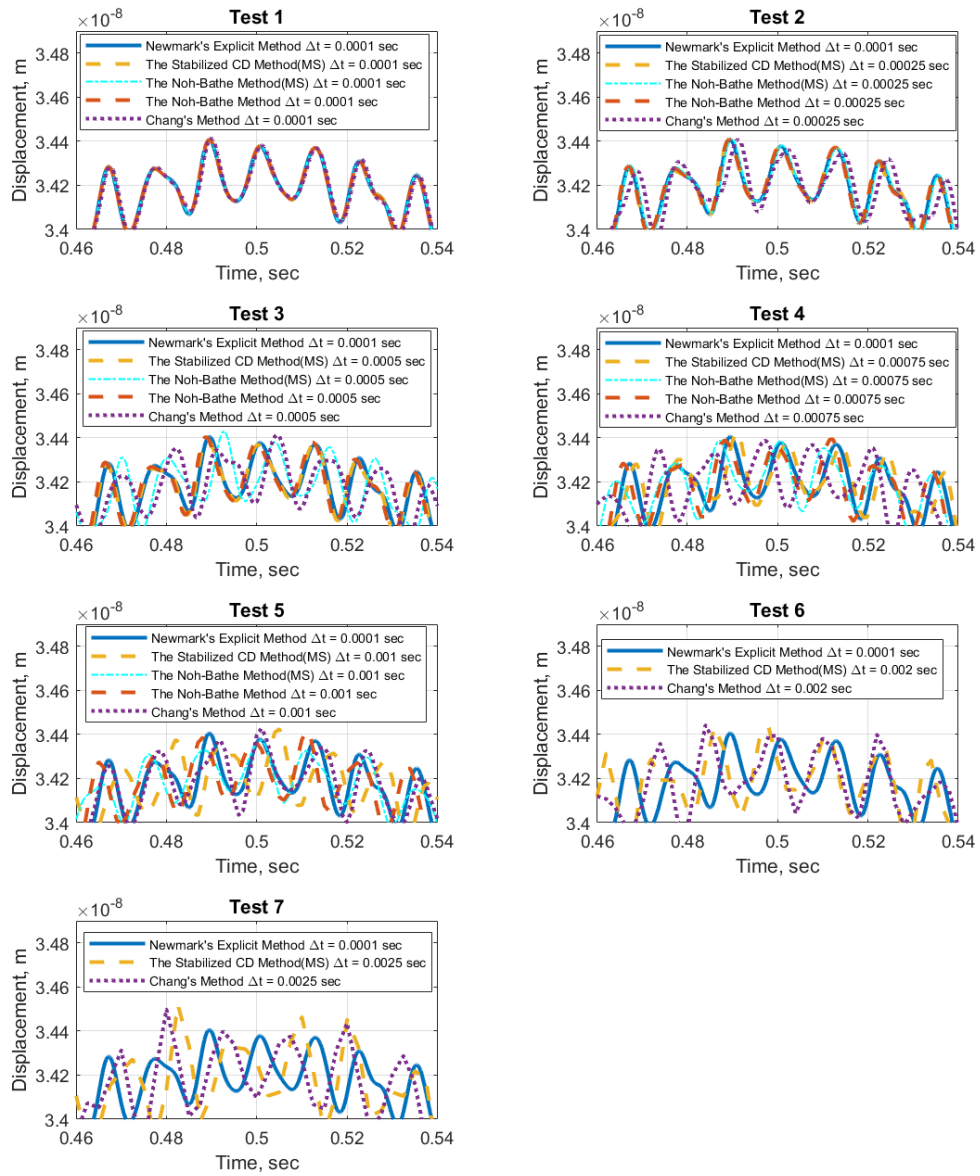


Figure A.3 Displacement Responses of the Case Study 2 - Model 1

Displacement Responses of Case Study 2 - Model 2 are presented in Figure A.4.

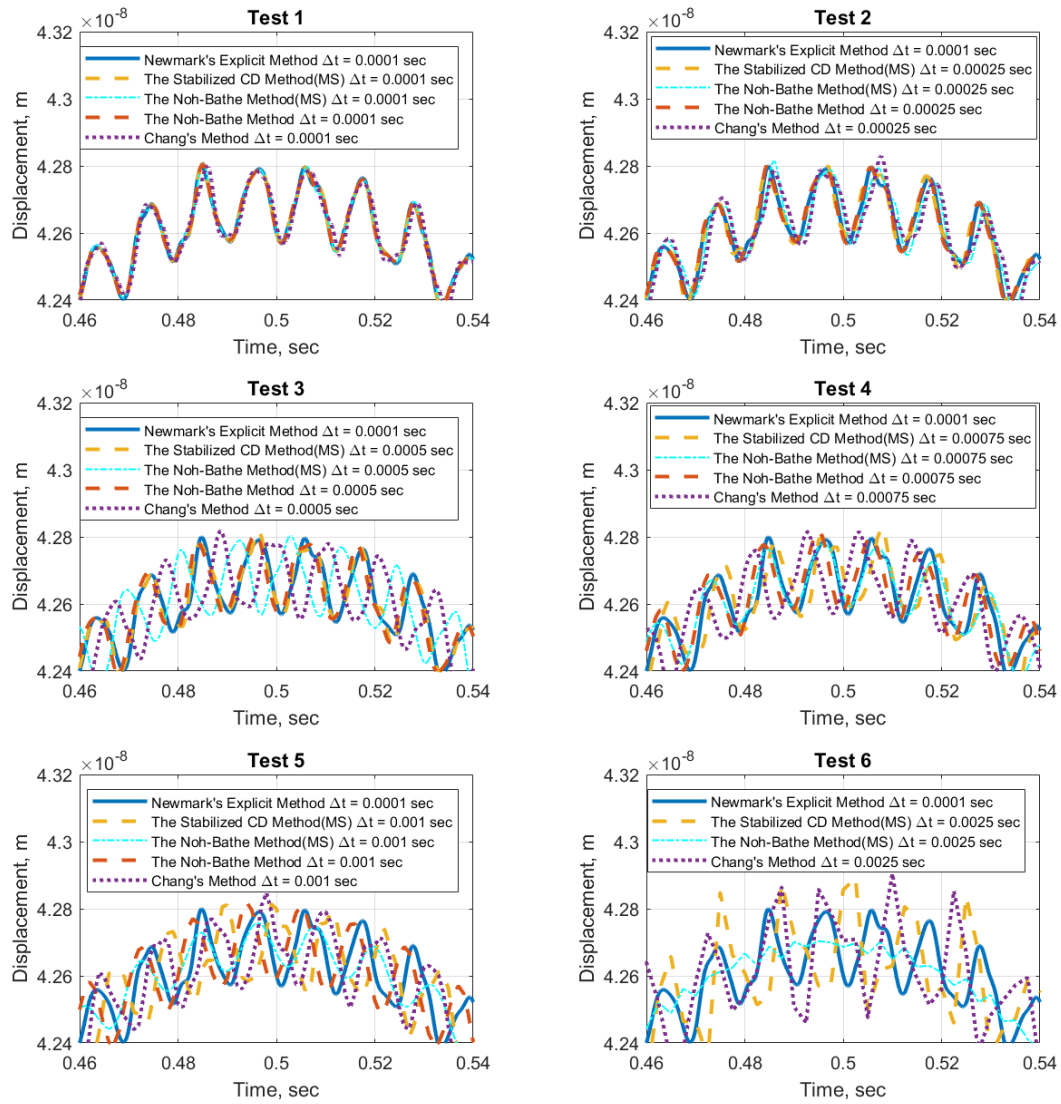


Figure A.4 Displacement Responses of the Case Study 2 - Model 2

Displacement Responses of Case Study 2 - Model 3 are presented in Figure A.5.

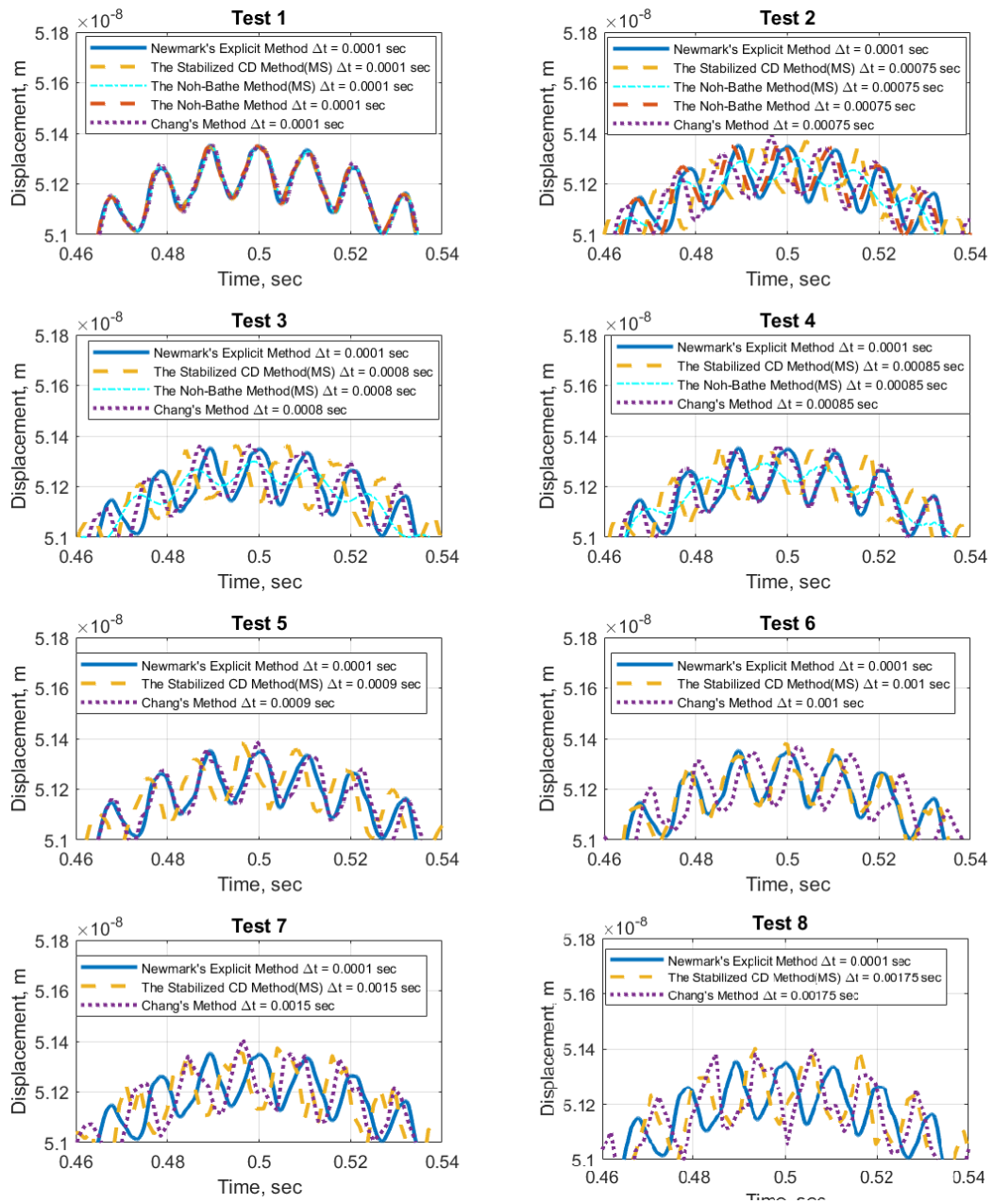


Figure A.5 Displacement Responses of the Case Study 2 - Model 3

Displacement Responses of Case Study 3 are presented in Figure A.6.

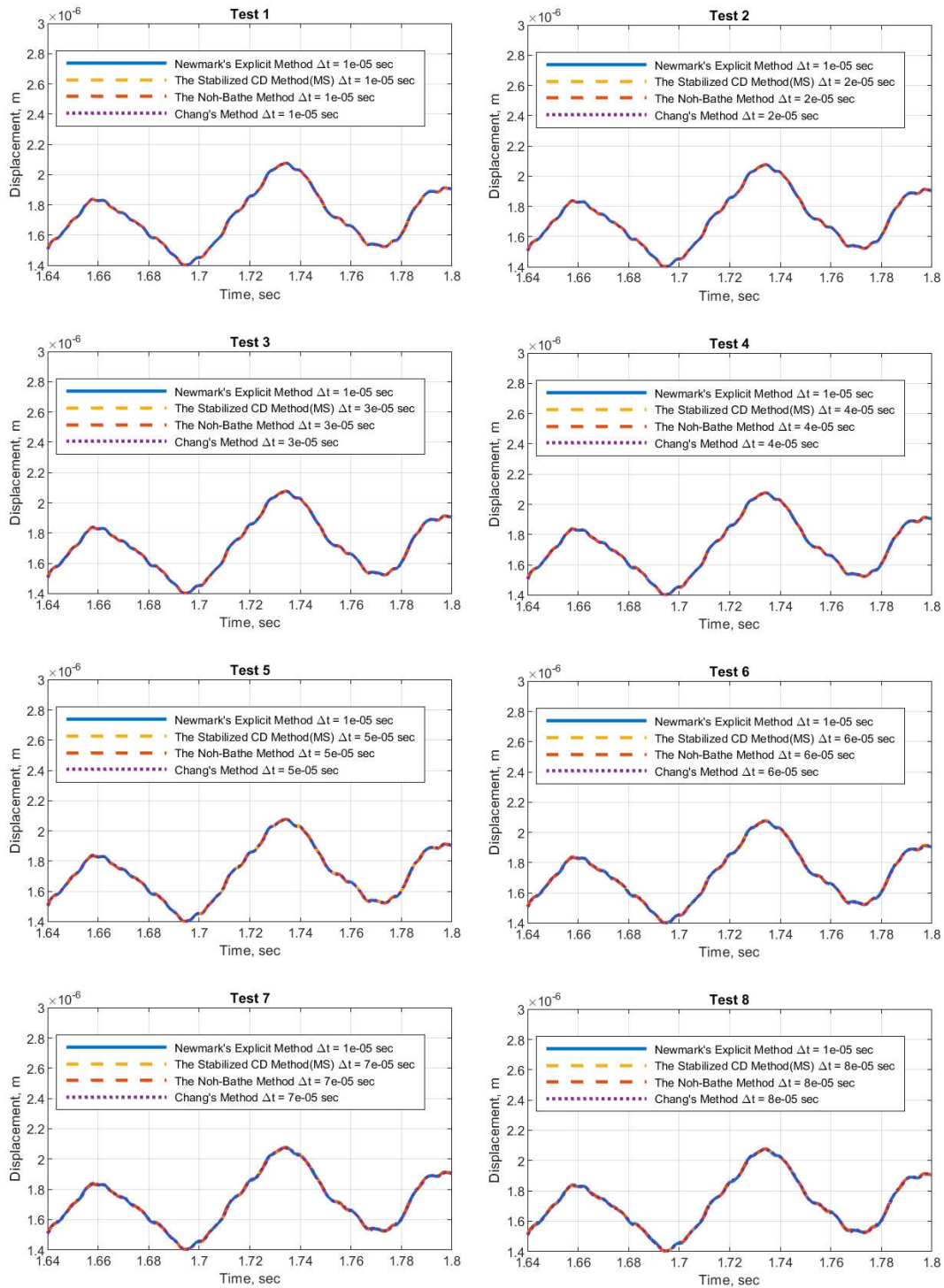


Figure A.6 Displacement Responses of the Case Study 3 – A

Displacement Responses of Case Study 3 are presented in Figure A.7.

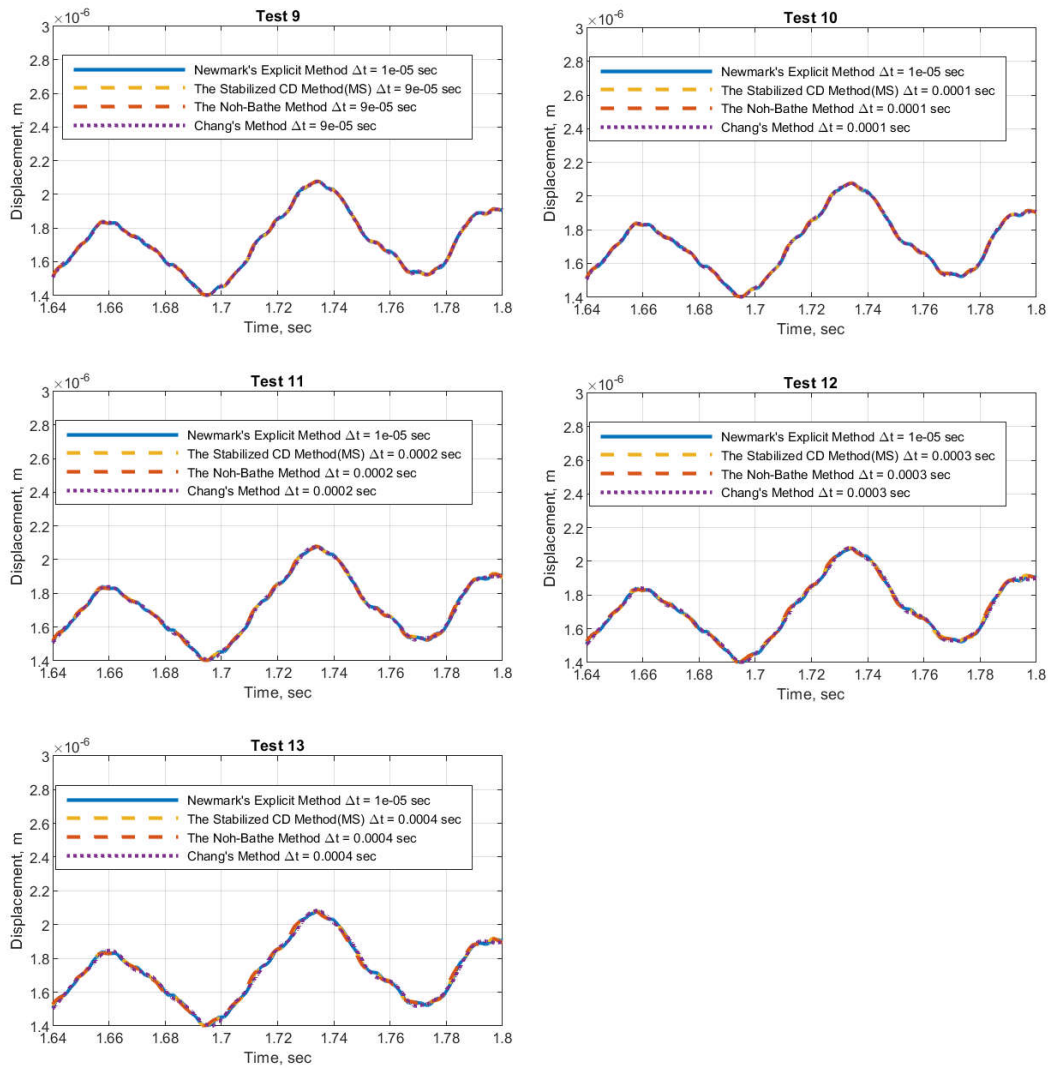


Figure A.7 Displacement Responses of the Case Study 3 – B