University of Trento
University of Brescia
University of Padova
University of Trieste
University of Udine
University IUAV of Venezia

Jia Chuanguo (Ph.D. Student)

MONOLITHIC AND PARTITIONED ROSENBROCK-BASED TIME INTEGRATION METHODS FOR DYNAMIC SUBSTRUCTURE TESTS

Prof. Oreste S. Bursi (Tutor)

March, 2010

UNIVERSITY OF TRENTO

Structural Engineering - Modelling, Preservation and Control of Materials and Structures

Cycle: XXII

Head of the Doctoral School: Prof. Davide Bigoni

Final Examination 19 / 04 / 2010

Board of Examiners

Prof. Mauro Da Lio (Università di Trento)

Prof. Alain Molinari (Università Paul Verlaine-Metz)

Prof. Rosario Ceravolo (Politecnico di Torino)

SUMMARY

Real-time testing with dynamic substructuring provides an efficient way to simulate the nonlinear dynamic behaviour of civil structures or mechanical facilities. In this technique, the test structure is divided onto two substructures: the relatively crucial substructure is tested physically and the other is modelled numerically in the computer. The key challenge is to ensure that both substructures interact in real-time, in order to simulate the behaviour of the emulated structure. This has special demands on the utilized integration methods and their implementations. Researchers have devoted significant effort to implement second-order integrators, such as Newmark integration methods, in a monolithic way where both substructures are integrated altogether. However, in view of large and complex structures, time integration methods are required to advance large-scale systems hence endowed with high-frequency components of the response or mixed first- and second- order systems like in the case of controlled systems. In this case, the monolithic implementation of a second-order time integration method becomes inefficient or inaccurate.

With these promises, the thesis adopts the Rosenbrock-based time integration methods for both dynamic simulations of complex systems and substructure tests, and in particular, focuses on the development of monolithic schemes with subcycling strategies for nonlinear cases and partitioned methods with staggered and parallel solution procedures for linear and nonlinear cases.

Initially, the Rosenbrock integration methods endowed with one stage to three stages are introduced and their applicabilities to second-order systems are investigated in terms of accuracy, stability and high-frequency dissipation, such as stability analysis of the Rosenbrock methods with one stage and two stages via the energy approach and numerical experiments on an uncoupled spring-pendulum system. Then, these methods are implemented in a monolithic way for real time substructure tests also considering subcycling strategies. Meanwhile, real-time substructure tests considering nonlinearities both in the numerical and physical substructures were carried out to illustrate the performances of the monolithic methods. Moreover, three types of partitioned algorithms based on the element-to-element partitioning are successively proposed. Two of them are based on acceleration continuity with a staggered solution procedure and a parallel solution procedure, respectively, and one of them is based on velocity continuity and a projection method. Both stability and accuracy properties of the proposed algorithms are examined by means of analytical techniques and numerical studies on single-, two-, three- and four-degree-of-freedom model problems and a coupled spring-pendulum system. Finally, a novel test rig conceived to perform both linear and nonlinear substructure tests with different combinations of numerical and physical substructures are presented and commented.

To my beloved parents and my wife.

ACKNOWLEDGEMENTS

First of all, I want to express my gratitude to my supervisor Prof. Oreste S. Bursi for his constant encouragement and illuminating instruction in all the time of the research. I am heartily thankful to him for all his help in my life.

I would like to thank Dr. Alessio Bonelli, Dr. Leonardo Vulcan and Dr. Leqia He who helped me a lot on time integration methods and gave me valuable advice. I also owe my sincere gratitude to Alessandro Zambonin for his contribution to the partitioned time integration methods. Moreover, I want to thank Iker.Elorza and Wang Zhen for collaborating in design the TT1 test-rig.

I want to thank the University of Trento for giving me the Ph.D. fellowship to commence this thesis and to do the necessary research work. Meanwhile, the research has benefited from the financial support of the SERIES project.

I have also enjoyed a lot of support from many people around me. For this I think the entire DIMS group of the UNITN. I owe special thanks to all of my colleagues along the way: Dr. Marco Molinari, Dr. Fabio Ferrario, Li Gu, Alireza Savadkoohi, Seema, Wu Huayong, Nicola Tondini, Md Shahin Reza and Yue Yanchao.

Lastly, I want to give special thanks to my wife, my parents and my parents-in-law, my sister and my uncle whose patient love enabled me to complete this thesis.

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Context

With the higher-rise, longer-span and smarter-material tendencies of structural systems and the higher requirements of safety and reliability, structural dynamic testing is becoming more and more important. As structures become higher and/or longer, their mathematical models and failure modes turn out to be more unpredictable with purely analytical techniques and their responses under dynamic loads, such as earthquake or wind, require advanced dynamic testing techniques. Also, with more efficient introduction of smart materials and devices to structures, their applications reduces the robustness and applicabilities of the existing design codes and need specific dynamic tests. Meanwhile, the introduction of new design concepts, such as performance-based seismic design, requires experimental techniques to validate their suitabilities.

Current dynamic testing includes a various methods: Free-vibration tests; Monitoring of ambient vibrations; Harmonic excitation tests; Shaking table tests; Quasi-static tests; Pseudodynamic tests and Real-time substructure tests (Negro and Magonette, 1998). To evaluate the dynamic performance of structures and components subjected to complex dynamic loading, such as earthquake, two basic experimental methods co-existed for a long time: the shaking table testing which frequently allows full time-scale but reduced space-scale; the Pseudodynamic testing which usually permits full space-scale but expanded time-scale. In a word, they both have inherent predominances but limitations (Williams and Blakeborough, 2001).

In order to overcome the problems of spatial and temporal scaling associated with the shaking table testing and the Pseudodynamic (PsD) testing, Real-time Testing with Dynamic substructuring (RTDS) was developed in the early 1990s (Nakashima et al., 1992) and, since then, it was used and advanced by researchers worldwide for seismic simulation studies (Nakashima and Masaoka, 1999; Dimig et al., 1999; Horiuchi et al., 1999; Darby et al., 1999; Williams, 2000; Bonnet, 2006; Bursi, 2007; Chen et al., 2007). The RTDS test method involves the combination of the real-time dynamic testing of a Physical Substructure (PS), that contains the key regions of interest, with the analytical simulation of a Numerical Substructure (NS), that contains the remainder of the emulated structure. The interaction of both substructures is achieved by imposing compatibility and equilibrium conditions at the interface.

The increasing interest of the RTDS method is motivated by two major features of this method: i) compared to the shaking table testing method, critical structural components of interest, such as portions with nonlinear behaviour or more prone to damage under dynamic loading, can be tested at full scale while the remainder is modelled numerically, which to some extent leads to significant cost savings and makes it possible to conduct full-scale test; ii) with respect to the PsD testing method, velocity-dependent phenomena can be taken into account, moreover, distributed-mass systems can be considered. Because of those two advantages, RTDS method is thus an desirable approach for earthquake engineering. However, those advantages are offset by the complexities of implementation (Nakashima, 2001; Blakeborough et al., 2001). The key challenge of the applications of the RTDS method to large or complex structures is to ensure that the PS and the NS interact in real-time. To confront this challenge, current RTDS research focuses both on the development of sophisticated control strategies (Horiuchi and Konno, 2001; Darby et al., 2001, 2002; Wallace et al., 2005; Neild et al., 2005; Gawthrop et al., 2009) and on the development of efficient numerical integration schemes (Bursi, 2007; Bonnet et al., 2007; Sajeeb et al., 2007; Shing, 2008; Bonnet et al., 2008). The research proposed herein is mainly related to numerical integration issues of the RTDS method.

Although the field of RTDS method is the main target of this thesis, both numerical strategies and methods developed for time-stepping schemes herein can also be applied to PsD testing with dynamic substructuring and pure numerical simulations. For PsD testing, the significant contribution of the substructuring methodology (Dermitzakis and Mahin, 1985) is scaling effects which extends the application to large structures. Likewise, with this method, the structure can be divided into two substructures: one is numerically simulated in computer in that it has a simple behaviour or it is not considered to be critical for the emulated structure; the remainder requires physical replication with PsD technique because it contains nonlinear behaviour or it is critical to the performance of the structure concerned (Pegon and Pinto, 2000). In the test, both substructures are solved monolithically with a direct time integration method (Bursi and Shing, 1996) or separately by using a partitioned time integration scheme (Pegon and Magonette, 2002). Differently from the conventional PsD testing methods and the direct integration algorithms, in the substructuring test the restoring forces in the NS are numerically modelled while the restoring forces in the PS are not numerically modelled but are measured from a test conducted in parallel with the direct/partitioned time integration (Williams and Blakeborough, 2001).

In this document, the time integration methods are developed for solving second-order systems, i.e. structural dynamics. But the utilized integrators are expressed in the first-order form. Besides the field of structural dynamics, it is believed that the time integration methods proposed, in particular the partitioned methods, can be implemented on both first- and second-order coupled problems in different fields of engineering or science (Nakshatrala et al., 2008; Felippa et al., 2001; Prakash and Hjelmstad, 2004).

1.2 Motivation of the research

A brief review of dynamic testing was given within the context of earthquake engineering in the previous section. Real time testing with dynamic substructuring was described and its limitations, especially its complexities of implementation, were pointed out. Moreover, the PsD testing with dynamic substructuring was introduced as well as its computational limitation. These limitations as well as the implementation difficulties of both hybrid testing methods, to some extent, reflect the high requirement for efficient and accurate integration methods. In this thesis, attempts are made to follow these requirements.

As underlined above, RTDS is efficient for modeling structures exhibiting complex nonlinear behaviour, especially if the nonlinearity is concentrated in specific regions of the structure. However, if a large structure is considered, computational model of the NS is expected to be nonlinear to take into account phenomena such as material plasticity, geometric nonlinearity and buckling. In order to integrate the nonlinear NS coupled to the PS in real time, an efficient time integration method is required capable to exhibit the following desirable properties: i) real-time compatibility; ii) unconditional stability; iii) explicit target displacements and velocities; iv) time efficiency.

Frequently, structural models for large-scale structures contain non-physical high-frequency components that are artifacts of standard finite-element modeling of the spatial domain. Moreover, physical high-frequency models are included but not accurately treated. Therefore, the equations of motion can contain stiff components of the response. In this case, an advanced integration method is required to filter out high-frequency oscillations without sacrificing the accuracy of low-frequency modes. With these considerations, we propose Rosenbrock-based L-stable Real-Time (LSRT) algorithms. These methods are linearly implicit, because they are unconditionally stable, but require only a single linearization and matrix decomposition per time step where the Jacobian is formed only at the beginning of each time step. The methods are real-time compatible and possess high-frequency dissipation capabilities.

Most of the aforementioned research works carried out on substructure tests considered structural integrators applied to the equations of motion expressed as second-order in time. Nonetheless, it is well known that the motion of the PS in a substructure test, see Fig.1.1, is driven by a transfer system -actuator- and sensors, governed by a control unit. Since the control system is typically described by first-order Differential-Algebraic Equations (DAEs), the utilized integrators have to deal with mixed first- and second-order DAEs. In order to solve this problem, there are mainly three options: i) to use different integrators for structural and control systems, respectively, Csee for instance Wu et al. (2007), that utilizes the Newmark- method for the emulated structure and a proprietary MTS controller with its own built-in time integrator; ii) to reformulate the control equations in a second-order form (Brüls and Golinval, 2006), and employ a structural integrator like the Generalized-α method (Chung and Hulbert, 1993) for both systems; to use first-order integrators like the LSRT algorithms, for both structural and control systems. Herein, we adopt the last option owing to the favourable properties of LSRT algorithms employed in control (Vulcan, 2006).

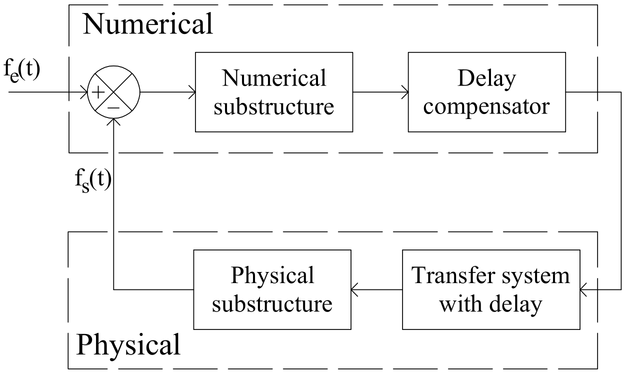


Fig. 1.1: Block diagram representation including delay for RTDS tests

With regard to applications time-stepping methods to RTDS tests, they can be broadly classified in monolithic and partitioned. In a monolithic approach, the method integrates: i) the Numerical Substructure (NS) only, whilst the Physical Substructure (PS) can be considered as a black box (Bursi et al., 2008) or as a grey box (Lamarche et al., 2009), with estimates of stiffness and damping of the PS included in the Jacobian matrix; ii) both the NS and the PS by means of stiffness estimates (Jung et al., 2007), like in a typical pseudo-dynamic (PsD) test. Conversely, a partitioned approach typically solves both NS and PS through different integrators and takes into account the interface problem, for instance by prediction, substitution and synchronization of Lagrange multipliers (Pegon and Magonette, 2002). In detail, partitioned algorithms can be applied to the Euler-Lagrange form of the equations of motion -second-order in time- (Prakash and Hjelmstad, 2004; Bonelli et al., 2008b) or to the Hamilton form of the equations of motion -first-order in time- (Nakshatrala et al., 2008). In this thesis, we consider both monolithic and partitioned approaches based on L-stable real-time compatible Rosenbrock algorithms applied to equations of motion first-order in time.

As far as complex emulated structures are concerned, numerical and control requirements impose different time steps for NS and PS, respectively. As a result, two main techniques can be identified to tackle this problem: i) model reduction, that represents an effective way to lower computation burdens related to the integration of a complex NS, but becomes very inaccurate especially for non-linear systems; ii) multi-time methods that allow to employ different time integrators in distinct subdomains. Moreover, subcycling permits to use different time steps in different subdomains. The last strategy is relatively simple to implement, but stability and accuracy properties of the original schemes can be hindered. Therefore, this thesis proposes some novel multi-time method with subcycling strategies, and also investigates relevant stability and accuracy issues.

When using subcycling strategies, the computer for RTDS tests must keep sending displacement signals without interruption to the digital servo-controller. On the other hand, the next target displacement is not ready at the instant when loading in the current integration time-step is completed. To overcome this problem, Nakashima (2001) proposed a approach in which the task of creating the target displacement (by solving the equations of motion) at an integration time step Δt and the task of creating displacement signals (to be sent to the servo-controller) at a smaller time-interval δt are separated. The signal generation task is programmed as: if the displacement target is not available, i.e. before the completion of numerical integration, the signals are generated by the extrapolation of previous displacements; once the numerical integration is completed and the target displacement is available, the signal generation task stops extrapolation and starts performing interpolation (Nakashima and Masaoka, 1999). In this thesis, a parallel solution procedure is proposed where the target displacement and velocity is provided in advance so that only interpolation is needed for signal generation task.

Hence, the objectives of this thesis can be summarised as follows:

1. To consider one- and two-stage linearly implicit Rosenbrock-based integrators applied to equations first-order in time for RTDS tests and PsD tests with DS;

2. To develop monolithic Rosenbrock-based time integration methods and subcycling strategies for RTDS;

3. To develop partitioned Rosenbrock-based time integration methods and compatible subcycling strategies for RTDS and PsD tests with DS;

4. To develop a proper framework of accuracy and stability analysis for partitioned methods;

5. To apply both monolithic and parallel partitioned integration methods to real-time substructure tests and to compare their efficiencies.

1.3 Thesis organization

With the objectives described in the previous subsection, the thesis presents the research work conducted by the author on the development of integration schemes for RTDS tests: monolithic methods and partitioned methods without and with considering subcycling strategies. The properties of the proposed methods are investigated in terms of stability, accuracy, high-frequency dissipation and implementation efficiency. Moreover, for experimental validations of the involved methods a test rig is conceived and constructed within the SERIES project. The organization is depicted in Fig. 1.2.

In detail, the thesis is organised as follows:

The first chapter focuses on the motivation of the thesis with respect to the requirements of the newly-developed RTDS technique.

The second chapter provides a detailed review of the previous work accomplished by other researchers, their contributions and the problems encountered. Firstly, the assessment of the RTDS method within the dynamic laboratory testing of structures is made in terms of advantages and limitations. Secondly, the RTDS technique is detailed as well as the problems restricting its development. Thirdly, the commonly-used integration methods are reviewed and the requirements for advanced integration schemes are underlined. Lastly, the development of partitioned methods is stated and both the GC and the PM methods are analysed in depth.

Chapter 3 introduces the Rosenbrock-based LSRT methods and their accuracy and stability are analysed when applied to second-order systems. Initially, the LSRT methods are introduced and their applications by other research are shortly discussed. Secondly, the accuracy analysis is carried out. Thirdly, the energy approach is used

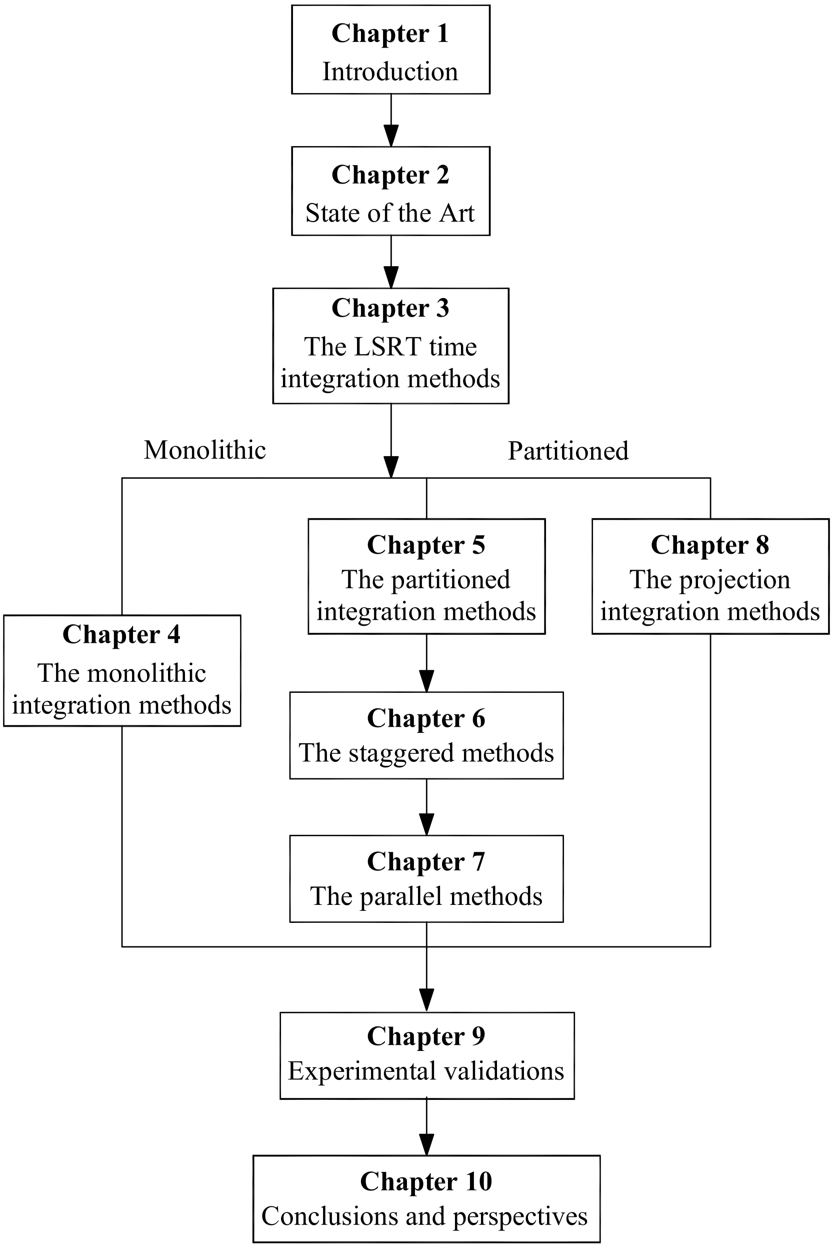


Fig. 1.2: Organization of the thesis

to investigate their stability properties with respect to second order systems. Lastly, an uncoupled spring-pendulum system is emulated numerically to validate both the theoretical analysis and the high-frequency dissipation properties of the linearly implicit time integration methods.

In Chapter 4, the monolithic integration schemes developed for RTDS are extended to nonlinear systems and subcycling strategies are developed for real time applications. Firstly partitioning and coupling techniques are introduced and applied to an uncoupled spring-pendulum system to achieve a coupled spring-pendulum system. Secondly, zero-stability analysis is performed for the coupled spring-pendulum system and both stability and accuracy analysis by means of simulations. Lastly, in order to validate the methods in real testing environment, we present RTDS test results for nonlinear Single-DoF system, and the Multiple-DoF spring-pendulum system.

From Chapter 5 to 7, the partitioned methods based on linearly implicit integrators are developed and studied. Chapter 5 mainly proposes the partitioned methods based on acceleration continuity. In chapter 6, subcycling strategies are proposed for the partitioned method which is inherently staggered. Chapter 7 extends the staggered methods to the parallel form. With respect to each type of partitioned methods, both theoretical analyses and numerical simulations of linear and nonlinear systems are carried out to investigate their performances.

Another family of partitioned methods, based on a projection solution procedure and velocity continuity, are presented in Chapter 8. Their accuracy and stability analyses are conducted. The numerical analysis is performed on a Single-DoF split-mass system and rechecked on Two- and Three-DoF systems. Finally, their applications to nonlinear systems are investigated through simulations of the coupled spring-pendulum system.

In Chapter 9, a test rig is designed and its different configurations are presented to appraise its capabilities. Finally, conclusions and future perspectives are drawn in Chapter 10.

CHAPTER 2

STATE OF THE ART

2.1 Introduction

This chapter provides a review of previous research related to this thesis. First, a briefly introduction is presented on the well-established themes of dynamic experimental testing of structures and the significance of the RTDS within the testing methods assessed. Then, the second section focuses on the RTDS in terms of control and actuator dynamics. Thirdly, the overview of the global integration methods used for solving the numerical substructure(s) are conducted in the third section, and several are emphasized which will be used for comparison. Finally, the fourth section presents the developments of the partitioned time integration methods, and the GC method and the PM method are introduced in great detail.

2.2 Experimental dynamic tests

This section provides a description of the experimental techniques which can be used for earthquake testing of civil engineering structures. In particular, the PsD test method and the RTDS method which are the focus of this thesis. A historical review is given as well as their advantages and limitations.

With the development of long-span and high-rise structures, a variety of testing methods have been developed to satisfy the strong requirement of experimental evaluation of structures. Articles by Takanashi and Nakashima (1987), Negro and Magonette (1998), and Williams and Blakeborough (2001), provide comprehensive overviews on contemporary research related to laboratory testing of structures under earthquake loads. The focus is on the context of the RTDS, for which four principal methods are introduced and discussed: quasi-static testing method, shaking table method, PsD testing method, PsD tests with dynamic substructuring and RTDS testing method. By comparison, the advantages of RTDS are discussed. Finally, key challenges ahead are detailed and possible contributions of the thesis are listed for the RTDS.

2.2.1 Quasi-static testing method

First, the most common technique, quasi-static testing method is briefly introduced. Quasi-static tests are performed by imposing predefined displacement or force histories on the specimen by actuators at an extended time scale. The specimen tested is generally composed of a series of single elements or simple portions of the emulated structure. By imposing cyclic displacements and measuring the corresponding restoring forces, one can predict the effect of systematic changes in material properties, details, boundary conditions, loading rates, and even the dynamic behaviour of the structures subjected to any dynamic input. Such tests are relatively easy and economical to execute. Since the displacement is predefined but not online, it may not cover the range of the displacements that a structure undergoes during an actual seismic event (Negro and Magonette, 1998).

2.2.2 Shaking table testing method

Shaking table method is used extensively in seismic research. In a real test, a reduced-scale model considering the law of similarity is mounted on a rigid platform, and both the specimen and the platform are excited to replicate ground motions, including recorded earthquakes time-histories (Bonnet, 2006). This testing method provides important data on the dynamic response caused by specific ground motions (either real or artificial), considering the inertial and damping characteristics of the tested structure and the phenomena of geometric nonlinearities, localized yielding and damage, and component failure. However, reduced-scale or highly simplified specimens are required for large-scale structures. This may cause problems in ensuring correct dynamic scaling: i) scale factors may not be optimized to be completely satisfied; ii) it is difficult to have confidence in the extrapolation of nonlinear dynamic response to full scale (Williams and Blakeborough, 2001); iii) scaling may result in poor representation of the behaviour of specific portions, such as connections (Negro and Magonette, 1998). In addition to the issues related to similitude, the performance of the shaking table tests requires sophisticated control system (Negro and Magonette, 1998; Williams and Blakeborough, 2001).

2.2.3 Pseudo-dynamic testing method

The third method reviewed is Pseudo-dynamic testing method, also termed as the online computer-controlled testing method or the quasi-static online testing method (Nakashima, 2001). In this experimental testing technique, a simulation is executed based on a step-by-step numerical solution of the governing equations of motion for model formulated considering both the numerical and physical components of an emulated structure:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{r}\left(\mathbf{u}\right) = \mathbf{f_e} \tag{2.1}$$

where **M** is the mass matrix, **C** the damping matrix, **u** the vector of nodal displacement, **r** is the restoring force vector, $\mathbf{f}_e$ is the vector of the external forces, and the dots represent differentiation with respect to time. In Eq. (2.1), the mass and viscous damping characteristics of the emulated structure are numerically modelled. Differently from conventional computational modelling and simulation where the entire structure is simulated analytically, the restoring force vector which contains uncertainties over the nonlinear stiffness and hysteretic damping characteristics is not evaluated numerically but directly measured on the structure at certain controlled locations. The detailed process of the PsD testing method is illustrated in Fig. 2.1. For simulating the earthquake response of a structure, a record of a real or artificially generated earthquake ground acceleration history is used to compute the external excitation f_e running the PsD algorithm. The displacement vector of the structure (where the mass of the structure can be considered to be concentrated) is calculated using a suitable integration method. The displacements are then applied to the test structure by servo-controlled hydraulic actuators fixed to the reaction wall. Load cells on the actuators measure the forces necessary to achieve the required displacements and these restoring forces are returned to the computer for use in the next time step calculation (Pegon and Pinto, 2000; Bursi, 2008). Even though Eq. (2.1) can be expressed

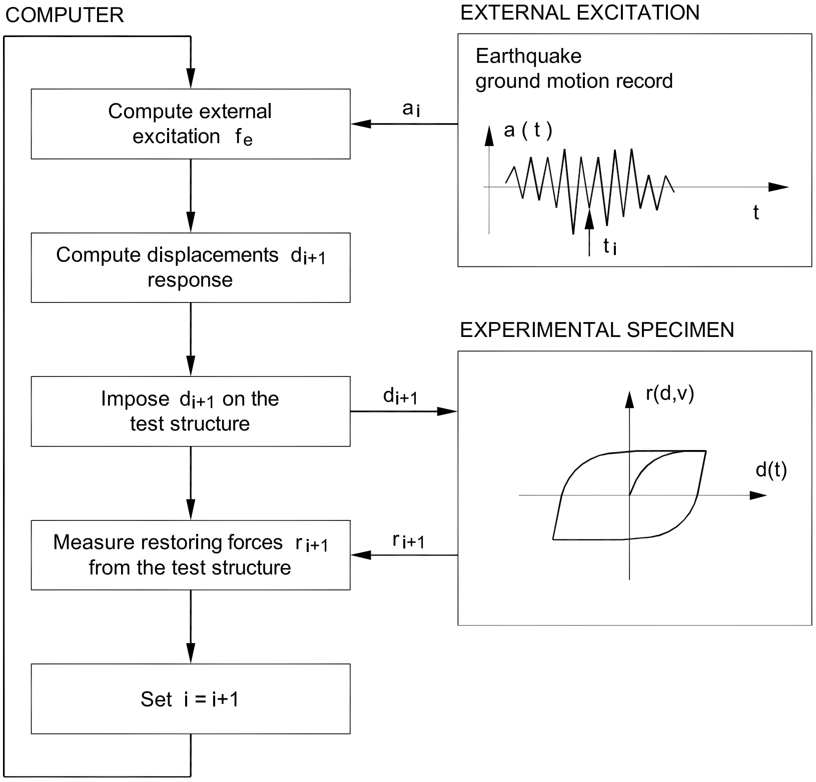

COMPUTER
EXTERNAL EXCITATION
Earthquake ground motion record
Compute external excitation fe
ai
a (t)
t
ti
Compute displacements di+1 response
EXPERIMENTAL SPECIMEN
Impose di+1 on the test structure
di+1
r(d,v)
d(t)
Measure restoring forces ri+1 from the test structure
ri+1
Set i = i+1

Fig. 2.1: Flow of the PsD testing method

for any number of degrees of freedom (DoFs), it is not feasible to test large structures. This is because the number of DoFs tested is determined by the number of available actuators and some other laboratory facilities. Even for a simple structure, the mass has to be concentrated in certain location through standard condensation techniques. If taking a example of a framed building, the slabs are generally assumed to be stiff enough for their in-plane deformability to be ignored and most of the mass is concentrated in the floor slabs. However, earthquake loading often leads to severe damage and unpredictable uncertainty only in parts of the structure. It would be much more desirable to test the critical parts of the emulated structure while also the restoring forces for the remainder are modelled in the computer. This is the main idea of the PsD testing method with dynamic substructuring.

2.2.4 PsD testing method with dynamic substructuring

It was observed that during structural testing, damage (and therefore nonlinear behaviour) often occurred in specific, limited regions of an entire structure. Hence, the substructuring technique was introduced to the PsD testing method by Dermitzakis and Mahin (1985). With the aid of the substructuring methodology, a physical model is built only of the part or parts where nonlinearity is expected (the PS) while the remainder is computationally modelled (the NS). The principle of the method is similar as the PsD testing method. The only difference is that the restoring force of the NS is numerically modelled.

With the substructuring technique, the number of DoFs of the tested structure is likely to be quite large. As far as utilized integrator is concerned, it was initially expected to use an explicit integration (Shing and Mahin, 1984). Solving large ODEs system with an explicit integrator may lead to stability problem, which blocks the application of the PsD testing to large-scale structural models. To overcome this, Dermitzakis and Mahin (1985) proposed a mixed implicit-explicit algorithm based on the work by Hughes and Liu (1978). Due to the increasing complexity of the tested structures, many efficient, unconditionally stable numerical algorithms were developed, such as Operator-splitting (OS) method (Nakashima, 1990), implicit Newmark method (Dorka and Heiland, 1991) and the alpha method (Shing et al., 1991). However, the direct application of implicit integration algorithms to PsD tests has been partially limited by the requirement to iterate with experimental substructures and difficulties in estimating the experimental tangent stiffness matrix. To implement implicit integration methods, (Shing et al., 1991), (Bursi and Shing, 1996) adopted a modified Newton-type iterative procedure with initial stiffness. The next section will provide a more detailed coverage of the development of numerical integration methods for PsD testing with substructuring.

Before moving to the RTDS testing in the next subsection, another efficient technique, the so-called continuous PsD testing method is introduced here, in which the loading rate is increased and the hold period is eliminated. This technique, by sustaining a smooth motion and a continuous loading of the test structure, eliminates or at least reduces force relaxation of structural materials. Moreover, capabilities for fast rate or near real-time also partially allows for rate-dependent effects of the tested structure. The new techniques for fast rate are built upon the same integration methods and principles developed for PsD testing in Subsection 2.2.3. As faster rate of testing with no hold period are achieved, additional challenges arise in solving this equations of motion: i) the use of shorter time step; ii) the adoption of parallel solution procedure; iii) the higher requirement to dealing with the inherent control error and response lag of servo-hydraulic systems. For this technique, Researchers at the JRC, ISpra have made a considerable effort and substantial contributions on the development both on parallel integration algorithms (Buchet and Pegon, 1994; Pegon and Magonette, 2002, 2005) and on control issues (Magonette et al., 1998).

2.2.5 Real-time testing with dynamic substructuring

When rate-dependent effects are of importance, the continuous PsD testing needs to be extended to RTDS. This approach is similar to PsD testing with substructuring, but with the testing proceeding in real time. The test principles can be better understood through a simple two-story building with a damper device as shown in Fig. 2.2. As shown in the figure, the tested structure is divided into a test specimen (the damper, which is expected to reduce the response of the overall structure and make possibly less damage to the main structure during the test) and a surrounding numerical substructure. By imposing compatibility and equilibrium conditions at the interface, the substructures are mode to interact possibly in real time, in order to emulate the dynamic behaviour of the overall structure. In detail, the test process starts from time integration in the numerical substructure, with the measured force from the PS and the external force. After integration, the desired displacement in the PS is determined and then sent to the actuator to advance the PS. At the end of the time-step the actuator loads and positions are measured and fed back to the numerical model. This loop is completed at each time incremental until the test is completed. This technique,

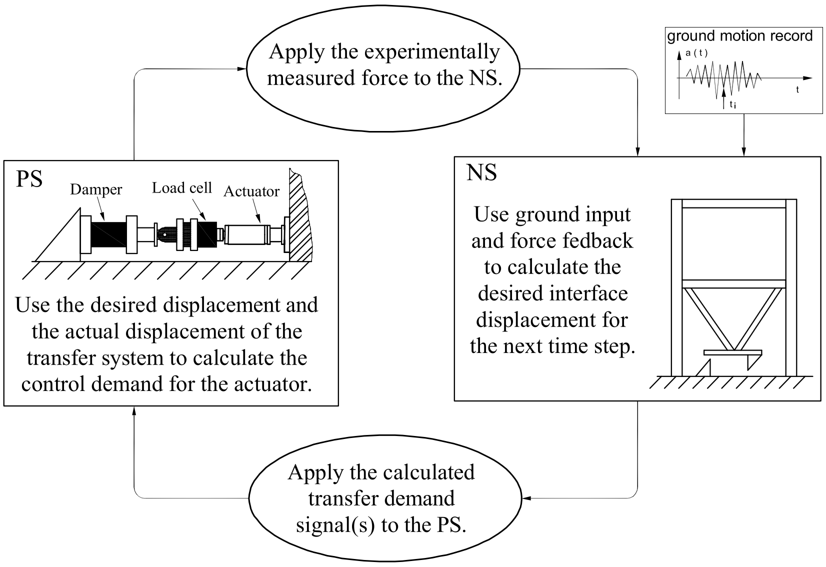

Apply the experimentally measured force to the NS.
ground motion record
a(t)
t
ti
PS
Damper
Load cell
Actuator
Use the desired displacement and the actual displacement of the transfer system to calculate the control demand for the actuator.
NS
Use ground input and force fedback to calculate the desired interface displacement for the next time step.
Apply the calculated transfer demand signal(s) to the PS.

Fig. 2.2: Control loop of the RTDS testing method

since it was developed by Nakashima et al. (1992), has been undergoing a rapid development world-widely both on control issues and integration methods (Nakashima, 2001; Williams and Blakeborough, 2001; Blakeborough et al., 2001). The increasing interest in RTDS testing is motivated by two major features: i) real time simulation enable the technique to consider rate-dependent phenomena such as strain rate effects on material properties or viscous damping forces for specific dissipative devices; ii) substructuring technique makes the possibility of large-scale structural tests with common laboratory facilities.

Meanwhile, real time testing requires rapidly imposing high loads or accurate displacements over a range of frequencies. This, on the one hand, imposes high requirement for efficient integration methods which will be detailed in the next section. On the other hand, it is a difficult task to make the actuator(s) execute exactly and continuously within a shorter period. In the case of the work presented in the thesis, the RTDS are performed in displacement control. The displacements must be conveyed rapidly from the NS to the PS. Also, the restoring forces measured form the PS must be quickly fed back to the NS. Communications between these two substructures are thus of paramount importance which, to some extent, determine the accuracy or the reliability of RTDS tests. To obtain desirable communications, efficient control schemes are required to minimize propagations of the experimental error during tests and to compensate for the actuator dynamics which is beyond emulated structures. Historically, Horiuchi et al. (1996) pointed out that the effect of the effect actuator dynamics may introduces negative damping for a linear system which may cause the test to become unstable, and utilized polynomial extrapolations to compensate for it. From then the effect has been gained sufficient attention by many researchers working on this subject (Horiuchi et al., 1999; Nakashima and Masaoka, 1999). Also, some control approaches were developed to considering actuator dynamics (Darby et al., 2001; Wagg and Stoten, 2001; Neild et al., 2005; Bursi, 2007). A fuller review of the delay compensation schemes used for RTDS testing is given in Chapter 5 of the thesis (Bonnet, 2006).

2.3 Integration methods for RTDS testing

In a RTDS test, finite element method is used to discretize the problem spatially. The resulting dynamic equations of motion are a system of second-order Ordinary Differential Equations (ODEs):

$$\mathbf{M\ddot{u}} + \mathbf{f}\left(\mathbf{u\ddot{u}}\right) = \mathbf{f_e} \tag{2.2}$$

where **f** is the assembled resisting force vector (which depend on the structural displacement vector and velocity vector). If the applied loads are entirely due to ground acceleration, the external force vector $\mathbf{f_e}$ on the right-hand side of (2.2) can be replaced by the following expression.

$$\mathbf{f_e} = -\mathbf{MB\ddot{u}_g} \tag{2.3}$$

where **B** is the ground acceleration transfer matrix and $\mathbf{\ddot{u}_g}$ is the specified support acceleration.

When considering the substructuring methodology, the resisting force vector can be divided by means of differential partitioning (Bursi, 2008) as follows

$$\mathbf{f}(\mathbf{u}\ddot{\mathbf{u}}) = \mathbf{f}^{n}(\mathbf{u}\ddot{\mathbf{u}}) + \mathbf{f}^{p}(\mathbf{u}\ddot{\mathbf{u}}) \tag{2.4}$$

where subscript n refers to the NS and subscript p stands for the PS. Note that the time integration methods discussed in this section are mainly used for the so-called monolithic integration method. If the experimental mass is negligible, the resisting force vector of the PS $\mathbf{f}^{p}$ is directly measured from the specimen by a data acquisition system. Otherwise, the force vector measured includes the inertial forces of the specimen. In order to avoid duplication, one way is to compensate for the initial forces by measuring the corresponding acceleration. Another way is to remove components with respect to the mass of the NS from the matrix $\mathbf{M}$. The RTDS test related to the coupled spring-pendulum system in Chapter 4 adopts the latter way.

Since the RTDS testing was developed, various integration methods have been implemented and validated in the past decade. In order to review them systematically, they are classified into three groups: explicit methods, implicit methods and linearly implicit methods. For an explicit method, the target displacement solution at t_{i+1} can be entirely expressed by known solutions such as the current state at t_i and $k-1$ previous solution states earlier, i.e.,

$$\mathbf{u}_{i+1} = \mathbf{f}(\mathbf{u}_i, \dot{\mathbf{u}}_i, \ddot{\mathbf{u}}_i, \dots, \mathbf{u}_{i-k+1}, \dot{\mathbf{u}}_{i-k+1}, \ddot{\mathbf{u}}_{i-k+1}) \tag{2.5}$$

where k indicates that the method belongs to k-step method. Note that in order to distinguish them from linearly implicit methods, the right hand side of 2.5 does not involves tangent stiffness matrix or its approximation. The advantages of explicit methods are that they are computationally efficient, easy to implement, and fast in their execution. However, they are conditionally stable -the second Dalquist barrier (Lambert, 1991, p.243)-. This indicates that the time step used frequently has to satisfy stability condition other than accuracy requirement, especially in presence of high-frequency components. In other words, explicit integration method is not suitable for stiff problems which have natural frequencies with different scales.

For an implicit method, the target displacement solution at t_{i+1} not only depends on known solutions as explicit methods but also unknown solutions at t_{i+1}, i.e.,

$$\mathbf{u}_{i+1} = \mathbf{f}(\mathbf{u}_{i+1}, \dot{\mathbf{u}}_{i+1}, \ddot{\mathbf{u}}_{i+1}, \mathbf{u}_i, \dot{\mathbf{u}}_i, \ddot{\mathbf{u}}_i, \dots, \mathbf{u}_{i-k+1}, \dot{\mathbf{u}}_{i-k+1}, \ddot{\mathbf{u}}_{i-k+1}) \tag{2.6}$$

Most of implicit methods are unconditionally stable, thus rendering suitability for stiff problems. It also implies that the choice of time step is due to the accuracy requirement, because this type of methods is stable for any time step. However, they are computationally more complex than explicit methods, often requiring an iterative solution process. Moreover, iterations may introduce spurious loading cycles for the PS and/or unloading process within a step which therefore causes non-realistic stiffness measurements.

Besides explicit methods and implicit methods, another type of integration methods is of interest in RTDS tests, linearly implicit methods. The methods are called linearly implicit in that they require only a single linearization and matrix decomposition per time step. This type of methods have explicit expression of target displacements as (2.5). But they also have some properties of implicit method, A-stability or L-stability, for the reason that single built-in iteration is included. A drawback of linearly implicit methods is that evaluation of tangent stiffness matrix and its inversion is required per time step. To solve this problem, users for RTDS testing frequently adopt initial stiffness matrix (Chang, 2002) or other approximations (Bursi et al., 2008; Lamarche et al., 2009).

In the following subsections a detailed review of time integration methods used for RTDS is presented following the aforementioned classification. For each method, the formulation is provided, and its stability, accuracy and efficiency for RTDS testing are detailed.

2.3.1 Central difference method

The central difference method (CDM) is the most popular time integration method for RTDS testing, especially RTDS testing applied to a Single- or Multiple-DoF systems (Nakashima et al., 1992; Horiuchi et al., 1999; Nakashima and Masaoka, 1999; Darby et al., 1999, 2001; Horiuchi and Konno, 2001; Blakeborough et al., 2001; Wu et al., 2005). It is mathematically described by:

$$\begin{aligned}
&\mathbf{M}\ddot{\mathbf{u}}_i + \mathbf{C}\dot{\mathbf{u}}_i + \mathbf{r}_i = \mathbf{f}_{e,i} \\
&\ddot{\mathbf{u}}_i = \frac{1}{\Delta t^2}\left(\mathbf{u}_{i+1} - 2\mathbf{u}_i + \mathbf{u}_{i-1}\right) \\
&\dot{\mathbf{u}}_i = \frac{1}{2\Delta t}\left(\mathbf{u}_{i+1} - \mathbf{u}_{i-1}\right)
\end{aligned} \tag{2.7}$$

where Δt is the time step chosen.

The CDM is explicit but conditionally stable. The stability limit of the CDM is $\omega_{max}\Delta t \leq 2$ where ω_{max} is the greatest natural frequency of the tested structure. Mathematically, the method is second-order accurate, with no dissipation but slight period delay with sufficiently small time step. For RTDS testing, (Nakashima and Masaoka, 1999) proposed that the choice of used time step satisfies $\omega_{max}\Delta t \leq 0.3 \sim 0.4$ for achieving accurate responses.

To implement the CDM for RTDS testing, inserting the acceleration and velocity expressions into equilibrium equation yields a formula of target displacement in terms of known solution of the two previous steps (Darby et al., 2001). The expression of the target displacement contains an inverse term but no time-dependent tangent stiffness and therefore the inversion can be provided in advance. Indeed, the method is efficient with nonlinear stiffness. If nonlinear damping forces are considered, the method becomes implicit and therefore inefficient. In this case, Wu et al. (2005) used another predictor for target velocity, i.e.,

$$\dot{\mathbf{u}}_i^P = \frac{1}{\Delta t}\left(\mathbf{u}_{i+1} - \mathbf{u}_i\right) \tag{2.8}$$

which was originally proposed by (Nakashima et al., 1992). This enables explicit form for target velocity, but deteriorates the original CDM in terms of stability and accuracy (Wu et al., 2005).

2.3.2 Explicit Newmark method

Newmark (1959) presented a family of single-step integration methods for the solution of structural dynamic problems. The method can be written in a standard form as

$$\begin{aligned}
&\mathbf{M}\ddot{\mathbf{u}}_{i+1} + \mathbf{C}\dot{\mathbf{u}}_{i+1} + \mathbf{K}\mathbf{u}_{i+1} = \mathbf{f}_{e,i+1} \\
&\mathbf{u}_{i+1} = \mathbf{u}_i + \Delta t\dot{\mathbf{u}}_n + \Delta t^2\left(\left(\frac{1}{2} - \beta\right)\ddot{\mathbf{u}}_i + \beta\ddot{\mathbf{u}}_{i+1}\right) \\
&\dot{\mathbf{u}}_{i+1} = \dot{\mathbf{u}}_i + \Delta t\left((1-\gamma)\,\ddot{\mathbf{u}}_i + \gamma\ddot{\mathbf{u}}_{i+1}\right)
\end{aligned} \tag{2.9}$$

The stability and accuracy of the method is determined by the choice of the parameters β and γ. Furthermore, numerical dissipation can be introduced by setting $\gamma \geq 0.5$, but this renders the method to be first-order accurate.

An explicit form can be derived from the Newmark method by setting $\beta = 0$, and the parameter γ is evaluated to be $1/2$ to achieve second-order accuracy. Inserting these two parameters into 2.9 yields the Explicit Newmark method as

$$\begin{aligned}
&\mathbf{M}\ddot{\mathbf{u}}_{i+1} + \mathbf{C}\dot{\mathbf{u}}_{i+1} + \mathbf{K}\mathbf{u}_{i+1} = \mathbf{f}_{e,i+1} \\
&\mathbf{u}_{i+1} = \mathbf{u}_i + \Delta t\dot{\mathbf{u}}_n + \frac{1}{2}\Delta t^2\ddot{\mathbf{u}}_i \\
&\dot{\mathbf{u}}_{i+1} = \dot{\mathbf{u}}_i + \frac{1}{2}\Delta t\left(\ddot{\mathbf{u}}_i + \ddot{\mathbf{u}}_{i+1}\right)
\end{aligned} \tag{2.10}$$

This method has the stability and numerical properties as the CDM: i) the method is second-order accurate; ii) the method is conditional stable and therefore not suitable for stiff problems. But its implementation is different from the CDM. Firstly, the explicit Newmark method is self-starting while the CDM requires a starting process (Wu et al., 2005). Second, the explicit solution of the target displacement is available in (2.10) and then the calculated displacement is sent to the PS; the experimental restoring force vector is measured and fed back into (2.10); finally, $\ddot{\mathbf{u}}_{i+1}$ and $\dot{\mathbf{u}}_{i+1}$ are achieved and the whole process is repeated to advance k to $k+1$. Third, the different implementation make a more favorable error-propagation characteristic for displacement feedback errors in an experiment (Shing and Mahin, 1990).

This method will be implemented to a spring-pendulum system in Chapter 3 and 4 for comparison. But the system exhibits nonlinearity that not only depends on displacements but also on velocities. In this case, the following predictor is used:

$$\dot{\mathbf{u}}_{i+1} = \dot{\mathbf{u}}_i + \Delta t\ddot{\mathbf{u}}_i \tag{2.11}$$

which is the same as (2.8) used by Wu et al. (2005) for the CDM. This predictor can be used for RTDS tests considering nonlinear damping force but its effects on the stability and order of accuracy have not been investigated in the literature and require further research. Moreover, the explicit Newmark method has been used in several RTDS tests considering only nonlinear stiffness forces by Bonnet (2006); Bonnet et al. (2007) and the method were proved to be accurate and stable.

2.3.3 Constant average acceleration method

Since the explicit methods are often conditionally stable, these can be limited when applied to structures with multiple DoFs or in presence high-frequency components. To overcome the aforementioned problems of explicit methods, many implicit methods were used for structural dynamics problems. In this subsection and the next subsection, two widely-used implicit methods are introduced.

The constant average acceleration method can be derived from (2.9) by introducing $\beta = 1/4$ and $\gamma = 1/2$. The equations therefore become:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{u}}_{i+1} + \mathbf{C}\mathbf{u}_{i+1} + \mathbf{K}\mathbf{u}_{i+1} &= \mathbf{f}_{e,i+1} \\ \mathbf{u}_{i+1} &= \mathbf{u}_i + \Delta t\dot{\mathbf{u}}_i + \frac{1}{4}\Delta t^2\left(\ddot{\mathbf{u}}_i + \ddot{\mathbf{u}}_{i+1}\right) \\ \dot{\mathbf{u}}_{i+1} &= \dot{\mathbf{u}}_i + \frac{1}{2}\Delta t\left(\ddot{\mathbf{u}}_i + \ddot{\mathbf{u}}_{i+1}\right) \end{aligned} \tag{2.12}$$

The method is unconditionally stable, second-order accurate, and one of the most effective and popular method for structural dynamics problems. The constant average acceleration method is also known as Trapezoidal method, as this method can be derived using the trapezoidal rule on the equivalent first-order system of equations. According to the second Dahlquist barrier (Lambert, 1991, p.243), this method is non-dissipative with lowest frequency distortion of all the unconditionally stable second-order accurate methods. To implement the method, Shing et al. (1996) proposed a modified Newton iterative algorithm for a slow PsD test and Shing (2008) extended this technique to RTDS tests. To eliminate the equilibrium error, a modified Newton iteration method was used based on the initial stiffness of the structure.

2.3.4 Generalized-α method

Compared with the constant average acceleration method which has no numerical dissipation, the Generalized-α method is more attractive for RTDS tests. The method provides proper dissipation for the high-frequency components, and the mount of the numerical dissipation can be specified through the choice of the spectral radius at infinity ρ_∞. To introduce the desirable numerical dissipation, two parameters α_m and α_f are introduced to shift the equilibrium point between t_i and t_{i+1}. Combining it with the difference formulae for displacement and velocity vectors in (2.9) yields the following method:

$$\begin{aligned}
\mathbf{M}\ddot{\mathbf{u}}_{i+1-\alpha_m} + \mathbf{C}\mathbf{u}_{i+1-\alpha_f} + \mathbf{r}_{i+1-\alpha_f} &= \mathbf{f}_{e,i+1-\alpha_f} \\
\mathbf{u}_{i+1} &= \mathbf{u}_i + \Delta t\dot{\mathbf{u}}_i + \Delta t^2\left(\left(\frac{1}{2}-\beta\right)\ddot{\mathbf{u}}_i + \beta\ddot{\mathbf{u}}_{i+1}\right) \\
\dot{\mathbf{u}}_{i+1} &= \dot{\mathbf{u}}_i + \Delta t\left((1-\gamma)\ddot{\mathbf{u}}_i + \gamma\ddot{\mathbf{u}}_{i+1}\right)
\end{aligned} \tag{2.13}$$

The time discrete combination of displacement, velocity and acceleration vectors respectively read

$$\begin{aligned}
\mathbf{u}_{i+1-\alpha_f} &= (1-\alpha_f)\,\mathbf{u}_{i+1} + \alpha_f\mathbf{u}_i \\
\dot{\mathbf{u}}_{i+1-\alpha_f} &= (1-\alpha_f)\,\dot{\mathbf{u}}_{i+1} + \alpha_f\dot{\mathbf{u}}_i \\
\ddot{\mathbf{u}}_{i+1-\alpha_m} &= (1-\alpha_m)\,\ddot{\mathbf{u}}_{i+1} + \alpha_m\ddot{\mathbf{u}}_i
\end{aligned} \tag{2.14}$$

For the internal force vector and the external force vector, Erlicher et al. (2002) employed the generalized trapezoidal, viz.

$$\begin{aligned}
\mathbf{r}_{i+1-\alpha_f} &= (1-\alpha_f)\,\mathbf{r}_{i+1} + \alpha_f\mathbf{r}_i \\
\mathbf{f}_{e,i+1-\alpha_f} &= (1-\alpha_f)\,\mathbf{f}_{e,i+1} + \alpha_f\mathbf{f}_{e,i}
\end{aligned} \tag{2.15}$$

Differently form the non-dissipative Newmark method which is two-stage method, the Generalized-α method is a three-stage method and has another two algorithmic parameters α_m and α_f. Moreover, the following relations hold:

$$\begin{aligned}
&\beta = \frac{1}{(1+\rho_\infty)^2}, \gamma = \frac{1}{2}\frac{3-\rho_\infty}{1+\rho_\infty}, \\
&\alpha_m = \frac{2\rho_\infty - 1}{1+\rho_\infty}, \alpha_f = \frac{\rho_\infty}{1+\rho_\infty}
\end{aligned} \tag{2.16}$$

Generally, $\rho_\infty \in [0, 1]$, and the choice of $\rho_\infty = 0$ corresponds to the case of asymptotic annihilation of the high-frequency response while $\rho_\infty = 1$ stands for the case of no algorithmic dissipation. The method benefits from the user-defined dissipation that limit the accumulation of experimental errors at high frequencies while not affecting the response of interest in the low frequency range. The method with proper choices of the parameters is unconditionally stable that makes it more attractive for solving systems with many DoFs and/or in presence of high-frequency components.

2.3.5 Newmark-Chang method

Even though implicit methods often lead to unconditional stable solution which is suitable for multiple-DoF structural problems, they require undesirable iterations which hampers their application to Multiple-DoF RTDS tests. For stability reasons, the utilized integration method is expected to be explicit (Blakeborough et al., 2001). To combat this, several linearly implicit methods that do not require iteration have been proposed for PsD tests (Nakashima, 1990; Chang, 2002) and RTDS tests (Chen et al., 2007; Bursi et al., 2008).

The Newmark-Chang method basically has the same procedure as Explicit Newmark method introducing two parameters β_1 and β_2 in the displacement difference equation, viz.

$$\begin{aligned}
&\mathbf{M}\ddot{\mathbf{u}}_{i+1} + \mathbf{C}\mathbf{u}_{i+1} + \mathbf{K}\mathbf{u}_{i+1} = \mathbf{f}_{e,i+1} \\
&\mathbf{u}_{i+1} = \mathbf{u}_i + \beta_1 \Delta t \dot{\mathbf{u}}_i + \beta_2 \Delta t^2 \ddot{\mathbf{u}}_i \\
&\dot{\mathbf{u}}_{i+1} = \dot{\mathbf{u}}_i + \frac{1}{2}\Delta t \left(\ddot{\mathbf{u}}_i + \ddot{\mathbf{u}}_{i+1}\right)
\end{aligned} \tag{2.17}$$

where two parameter β_1 and β_2 are given by

$$\begin{aligned}
\beta_1 &= \left[\mathbf{I} + \tfrac{1}{2}\Delta t \mathbf{M}^{-1}\mathbf{C} + \tfrac{1}{4}\Delta t^2 \mathbf{M}^{-1}\mathbf{K}_0\right]^{-1} \left[\mathbf{I} + \tfrac{1}{2}\Delta t \mathbf{M}^{-1}\mathbf{C}\right] \\
\beta_2 &= \tfrac{1}{2}\left[\mathbf{I} + \tfrac{1}{2}\Delta t \mathbf{M}^{-1}\mathbf{C} + \tfrac{1}{4}\Delta t^2 \mathbf{M}^{-1}\mathbf{K}_0\right]^{-1}
\end{aligned} \tag{2.18}$$

such that the method becomes linearly implicit. Here, $\mathbf{K}_0$ represents the initial stiffness matrix. The method is spectrally equivalent to the constant average acceleration method which is believed to be most precise unconditionally stable method in the family of Newmark methods. Nonetheless, this scheme exhibit two clear drawbacks: i) it requires a predictor like (2.11) for cases nonlinear damping forces; ii) β_1 and β_2 depend on the **C** matrix, and so these parameters need to be updated at the beginning of Δt. Moreover, Krenk and Hogsberg (2005) pointed out that structural integrators used in structural dynamics generally do not conserve the mechanical energy, but an equivalent energy involving **K**, **M**, Δt and algorithmic parameters. In this respect, we re-analysed the above-mentioned algorithms showing that the Newmark-Chang method is the only one that conserves the mechanical energy also for large Δt in the transient response of high-frequency modes; whilst the Explicit- Newmark's method (Bonnet et al., 2008) and the CR algorithm (Chen et al., 2009) exhibit large energy excursions.

2.3.6 The CR method

The explicit CR integration algorithm was developed and implemented for real-time testing by Chen and Ricles (2008). This method was derived using discrete control theory. The defining equations for the method are:

$$\begin{aligned}
&\mathbf{M}\ddot{\mathbf{u}}_{i+1} + \mathbf{C}\mathbf{u}_{i+1} + \mathbf{K}\mathbf{u}_{i+1} = \mathbf{f}_{e,i+1}\\
&\dot{\mathbf{u}}_{i+1} = \dot{\mathbf{u}}_i + \alpha_1 \Delta t \ddot{\mathbf{u}}_i\\
&\mathbf{u}_{i+1} = \mathbf{u}_i + \Delta t \dot{\mathbf{u}}_i + \alpha_2 \Delta t^2 \ddot{\mathbf{u}}_i
\end{aligned} \tag{2.19}$$

where the parameters α_1 and α_2 read:

$$\alpha_1 = \alpha_2 = 4\left[4\mathbf{M} + 2\Delta t \mathbf{C}_0 + \Delta t^2 \mathbf{K}_0\right]^{-1}\mathbf{M} \tag{2.20}$$

Differently from the Newmark Chang's method, this method also provides an explicit expression of target velocity vector while requiring the initial damping. Spectrally, the CR method is the same as the Newmark-Chang method for the reason that it is equivalent to constant average acceleration method. Therefore, the method is second-order accurate, unconditionally stable and non-dissipative with minor period distortion characteristics.

Moreover, this method has been implemented in several RTDS tests and it was demonstrated to be stable and accurate (Chen et al., 2009). Though initial stiffness and damping matrices are used if no sufficient time to update them, the method still has stable performance when applied to a soften structure.

2.3.7 Summary

In order to satisfy the application to RTDS tests for large and/or complex structures that have more possibility of high-frequency components, it is desired to develop or adopt an efficient and accurate numerical integration method. Based on the reviews of RTDS technique and numerical integration methods, some essential requirements of a desirable time integration method can be concluded as follows: i) unconditionally stable; ii) explicit expressions of target displacement and velocity vectors; iii) real-time compatible; iv) user-defined high-frequency dissipation; v) computational efficiency. Among the widely-used integration methods discussed in this section, no one can satisfy completely the five requirements. In the thesis, Rosenbrock-based LSRT algorithms developed by Bursi (2008) will be implemented in both monolithic and partitioned ways. All the aforementioned requirements are basically satisfied and the methods belong to first-order method which is suitable for mixed first-order and second-order systems or complex structures with single mass matrix.

2.4 Partitioned time integration methods

Large-scale problems may involve elements or meshing with different sizes varying over several orders of magnitude. Also, many complex problems may contains different system characteristics. In these cases, partitioned time integration methods are desirable which can divide the overall system into different subdomains and subsequently solve the subdomains separately with different integrators or different time steps. In this section, a brief review of domain decomposition techniques related to partitioned time integration methods and a short introduction to two partitioned methods are provided.

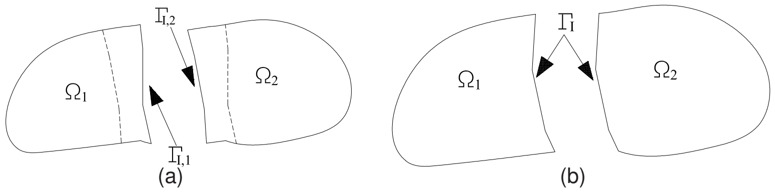


Fig. 2.3: (a) Overlapping domain decomposition method; (b) Non-overlapping domain decomposition method

2.4.1 Introduction to domain decomposition methods

Originally, Domain decomposition (DD) methods have been used to solve coupled multi-physics problems such as fluid-structure interaction. However, recently, with the advent of parallel computers and hybrid simulations, researchers have started using DD methods to divide a large-scale simulation or computation tasks into several relatively small problems which can be solved with different processors or with different approaches. The technique entails parallelism and is particularly attractive to parallel computation. DD methods, as shown in Fig. 2.3, can be classified into overlapping and non-overlapping DD methods (Elleithy and Al-Gahtani, 2000). Overlapping DD methods, as indicated by terminology, divide the overall domain into subdomains which have at least one intersection area between them. The classical *Schwarz alternating* method belongs to this class(Chan and Smith, 1996). Conversely, non-overlapping DD methods generate subdomains with no intersection between each other. The article by Xu and Zou (1998) provides a good review and a unified presentation on the theoretical aspects of non-overlapping DD methods.

Non-overlapping DD methods have been relatively wider used in the last decade, which are also referred to as substructuring or Schur complement methods. Among substructuring methods, two similar methods have emerged and are considered to be efficient parallel computing methods: the primal and dual substructuring methods (Prakash, 2007). In a primal substructuring approach, the coupling of the two substructures are achieved through a primal variable at the interface, which is either displacement or velocity or acceleration. A dual substructuring approach couples two substructures by introducing a dual variable, i.e. the Lagrange multiplier. Both of them

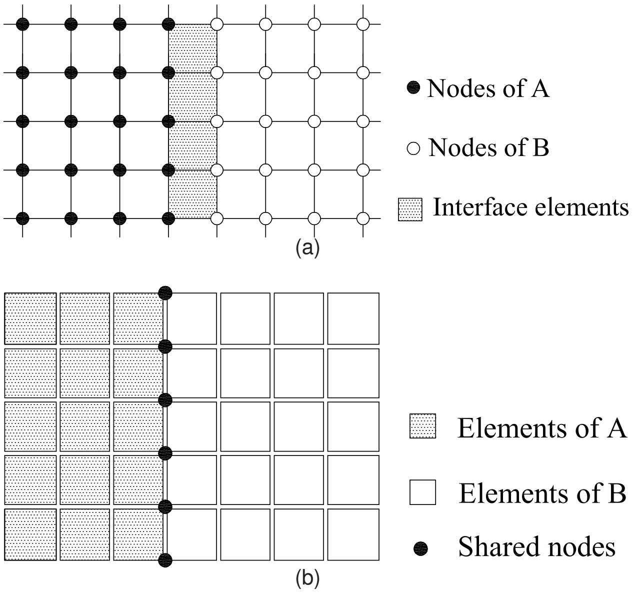


Fig. 2.4: The partitioning approach: (a) Node partitioning; (b) element partitioning

are based on very similar concepts. However, the latter one, which is currently called the Finite Element Tearing and Interconnecting (FETI) method, is becoming more and more popular due to its ability to deliver larger speed-ups on a larger number of processors (Farhat et al., 1998). All the partitioned methods in Subsection 2.4.2 and the proposed methods in the thesis are based on this technique. In structural dynamics, two types of partitioning approaches based on non-overlapping DD methods are most commonly used: node partitioning and element partitioning (Prakash and Hjelmstad, 2004). Node partitioning divides the overall structure into different sets of nodes with a layer of interface elements (see Fig. 2.4a). In the case of element partitioning (as shown in Fig. 2.4b), different sets of elements are assigned to different subdomain by a boundary of interface nodes. In order to specify both partitioning approaches, the relationships between the stiffness matrix of the overall domain and each subdomain divided with both approach can provide a distinct explanation.

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{AA} & \mathbf{K}_{AB} \\ \mathbf{K}_{BA} & \mathbf{K}_{BB} \end{bmatrix} \tag{2.21}$$

As shown in Eq. (2.21), node partitioning simply divides the system matrix into disjoint parts with the off-diagonal terms representing the interface element. In this case, the number of overall degrees of freedom is the same as the corresponding uncoupled system. However, node partitioning results in non-zero off-diagonal terms and therefore requires complex coupling algorithms, especially in the case of interface elements which are shared by more than two subdomains. Hence this approach is not suitable for real-time tests.

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{AA} & \mathbf{K}_{AI} & \\ \mathbf{K}_{IA} & \mathbf{K}_{II}^{A} + \mathbf{K}_{II}^{B} & \mathbf{K}_{IB} \\ & \mathbf{K}_{BI} & \mathbf{K}_{BB} \end{bmatrix} \tag{2.22}$$

For element partitioning, the system matrix (2.22) is composed of the elements of the corresponding subdomains with the overlap representing shared nodes which couple the individual solutions together. Differently from the node partitioning, this approach increase the overall DoFs by the number of DoFs at interfaces, but every subdomain can be solved separately and coupling can be ensured by the continuity of the solution across the interface and equilibrates tractions between the subdomains. The discussion in this thesis is restricted to element partitioning.

2.4.2 The finite element tearing and interconnecting method - the FETI method

The FETI method based on dual Schur complement approach has recently received significant attention from researchers in structural dynamics. The method enforces continuity of the solution across the interface by introducing Lagrange multipliers. This makes the method suitable for coupling two or more subdomains almost completely independent of each other. Being a relatively new method, it has been constantly undergoing enhancement in the last decade. For example, Farhat and Roux (1991) developed a FETI method for the parallel finite element solution of equilibrium equations; and its extension to dynamic problem was presented by Farhat et al. (1994). Moreover, based on differentially partitioned FETI method, Park et al. (1997) presented an algebraically partitioned FETI method for structural problems on parallel computers. In the following, we provide in brief the derivation of differential algebraic equations based on differentially partitioned FETI method.

Consider an arbitrary domain Ω partitioned into S subdomains. For these multiple subdomains, the Lagrangian can be written as

$$L = T - U = \sum_{i=1}^{S} \left[\frac{1}{2} \dot{\mathbf{u}}^i \mathbf{M}^i \dot{\mathbf{u}}^i - \frac{1}{2} \mathbf{u}^i \mathbf{K}^i \mathbf{u}^i \right] \tag{2.23}$$

where $\mathbf{M}^i$ and $\mathbf{K}^i$ are respectively the mass matrix and stiffness matrix of each subdomain. In addition, the solution must be continuous across the interface. For simplicity, we impose the continuity of displacements at the interface as:

$$\sum_{i=1}^{S} \mathbf{G}^i \mathbf{u}^i = 0 \tag{2.24}$$

where $\mathbf{G}^i$ is a Boolean matrix that operates on nodal vectors of subdomain Ω_i, picks out the degrees of freedom lying along its interface and assembles them in an interface vector. Note that the conjugate to $\mathbf{G}^i$, i.e. the matrix ${\mathbf{G}^i}^T$ operates on interface vectors, picks out the degrees of freedom that belong to Ω_i and assembles them at their corresponding position in the nodal vector for subdomain Ω_i.

The Lagrangian is augmented with the constraint equation (2.24) using a multiplier Λ as:

$$L = \sum_{i=1}^{S} \left[\frac{1}{2} \dot{\mathbf{u}}^i \mathbf{M}^i \dot{\mathbf{u}}^i - \frac{1}{2} \mathbf{u}^i \mathbf{K}^i \mathbf{u}^i \right] - \mathbf{\Lambda}^T \sum_{i=1}^{S} \mathbf{G}^i \mathbf{u}^i \tag{2.25}$$

Through the Euler-Lagrange equations

$$\begin{cases} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{u}}^i} \right) - \frac{\partial L}{\partial \mathbf{u}^i} = \mathbf{f}_e^i, \; i = 1, \dots, S \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{\Lambda}}} \right) - \frac{\partial L}{\partial \mathbf{\Lambda}} = 0 \end{cases} \tag{2.26}$$

We arrive at the system

$$\begin{cases} \mathbf{M}^i \ddot{\mathbf{u}}^i + \mathbf{K}^i \mathbf{u}^i = \mathbf{f}_e^i + {\mathbf{G}^i}^T \mathbf{\Lambda}, \; i = 1, \dots, S \\ \sum_{i=1}^{S} \mathbf{G}^i \mathbf{u}^i = 0 \end{cases} \tag{2.27}$$

where Λ can be interpreted as the interface reaction forces acting internally between all subdomains. Such a system of equations belongs to a broader class of equations called differential algebraic equations (DAEs). Actually, the system of 2.27 is DAEs of index-3. An important concept in studying DAEs is the differential index, which plays a key role in the classification and the behaviour of DAEs. The definition of the differential index can be expressed by the minimum number of differentiations needed to transform a system of DAEs into a system of ODEs. In general, it is not advisable to tackle DAEs of index 3 directly (Lunk and Simeon, 2006). Instead, the index needs to be lowered first by introducing alternative formulations.

2.4.3 Introduction to the GC method

For the sake of comparison, two partitioned methods which are appealing for PsD testing with substructuring (Gravouil and Combescure, 2001; Combescure and Gravouil, 2002; Pegon and Magonette, 2002; Pegon, 2008) are introduced. Firstly, we present the GC method which enables arbitrary numeric schemes of the Newmark family to be coupled with different time steps in each subdomain through continuity of velocities at the interface (Gravouil and Combescure, 2001). We consider here for simplicity, the GC method with only two subdomains A and B. Subdomain A is discretized in time with a coarse time step Δt_A and subdomain B with a fine time step Δt_B. It is assumed that

$$\Delta t_A = ss\Delta t_B \tag{2.28}$$

with *ss* the number of substeps. The basic principles of the method can be illustrated in Fig. 2.5. Its solution procedure can be detailed as follows:

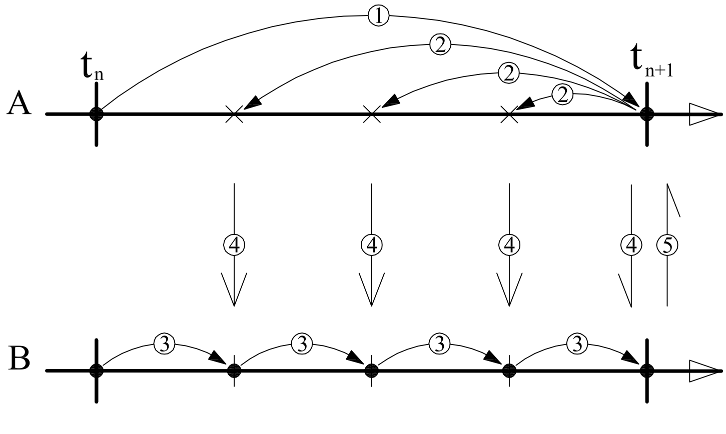


Fig. 2.5: The solution procedure of the GC method

1. Solve the free problem in subdomain *A*.

$$\begin{aligned} \tilde{\mathbf{M}}^A\ddot{\mathbf{u}}^A_{n+1_{free}} &= \mathbf{F}^A_{ext,n+1} - \mathbf{K}^A\tilde{\mathbf{u}}^A_n - \mathbf{C}^A\tilde{\dot{\mathbf{u}}}^A_n \\ \mathbf{u}^A_{n+1_{free}} &= \tilde{\mathbf{u}}^A_n + \alpha^A_1\ddot{\mathbf{u}}^A_{n+1_{free}} \\ \dot{\mathbf{u}}^A_{n+1_{free}} &= \tilde{\dot{\mathbf{u}}}^A_n + \alpha^A_2\ddot{\mathbf{u}}^A_{n+1_{free}} \end{aligned} \tag{2.29}$$

where

$$\begin{aligned} \tilde{\mathbf{M}}^A &= \mathbf{M}^A + \alpha^A_1\mathbf{K}^A + \alpha^A_2\mathbf{C}^A \\ \tilde{\mathbf{u}}^A_n &= \mathbf{u}^A_n + \Delta t\dot{\mathbf{u}}^A_n + \Delta t^2\left(\frac{1}{2} - \beta_A\right)\ddot{\mathbf{u}}^A_n \\ \tilde{\dot{\mathbf{u}}}^A_n &= \dot{\mathbf{u}}^A_n + \Delta t\left(1 - \gamma_A\right)\ddot{\mathbf{u}}^A_n \\ \alpha^A_1 &= \beta_A\Delta t^2 \\ \alpha^A_2 &= \gamma_A\Delta t \end{aligned} \tag{2.30}$$

2. Interpolate the free velocity in subdomain A.

$$\dot{\mathbf{u}}^A_{n+j/ss_{free}} = \left(1 - \frac{j}{ss}\right)\dot{\mathbf{u}}^A_{n_{free}} + \frac{j}{ss}\dot{\mathbf{u}}^A_{n+1_{free}} \tag{2.31}$$

3. Solve the free problem in subdomain B.

$$\begin{aligned} \tilde{\mathbf{M}}^B\ddot{\mathbf{u}}^B_{n+j/ss_{free}} &= \mathbf{F}^B_{ext,n+j/ss} - \mathbf{K}^B\tilde{\mathbf{u}}^B_{n+(j-1)/ss} - \mathbf{C}^B\tilde{\dot{\mathbf{u}}}^B_{n+(j-1)/ss} \\ \mathbf{u}^B_{n+j/ss_{free}} &= \tilde{\mathbf{u}}^B_{n+(j-1)/ss} + \alpha^B_1\ddot{\mathbf{u}}^B_{n+j/ss_{free}} \\ \dot{\mathbf{u}}^B_{n+j/ss_{free}} &= \tilde{\dot{\mathbf{u}}}^B_{n+(j-1)/ss} + \alpha^B_2\ddot{\mathbf{u}}^B_{n+j/ss_{free}} \end{aligned} \tag{2.32}$$

where

$$\begin{aligned} \tilde{\mathbf{M}}^B &= \mathbf{M}^B + \alpha^B_1\mathbf{K}^B + \alpha^B_2\mathbf{C}^B \\ \tilde{\mathbf{u}}^B_{n+(j-1)/ss} &= \mathbf{u}^B_{n+(j-1)/ss} + \frac{\Delta t}{ss}\dot{\mathbf{u}}^B_{n+(j-1)/ss} + \left(\frac{\Delta t}{ss}\right)^2\left(\frac{1}{2} - \beta_B\right)\ddot{\mathbf{u}}^B_{n+(j-1)/ss} \\ \tilde{\dot{\mathbf{u}}}^B_{n+(j-1)/ss} &= \dot{\mathbf{u}}^B_{n+(j-1)/ss} + \frac{\Delta t}{ss}\left(1 - \gamma_B\right)\ddot{\mathbf{u}}^B_{n+(j-1)/ss} \\ \alpha^B_1 &= \beta_B\left(\frac{\Delta t}{ss}\right)^2 \\ \alpha^B_2 &= \gamma_B\frac{\Delta t}{ss} \end{aligned} \tag{2.33}$$

4. Calculate the Lagrange multiplier at end of substep and solve the link problem in subdomain *B*.

The Lagrange multiplier can be obtained with the following equations

$$\mathbf{H}\boldsymbol{\Lambda}_{n+j/ss} = -\left(\mathbf{L}^A\dot{\mathbf{u}}^A_{n+j/ss_{free}} + \mathbf{L}^B\dot{\mathbf{u}}^B_{n+j/ss_{free}}\right) \tag{2.34}$$

where

$$\mathbf{H} = \alpha_2^A\mathbf{L}^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\left(\mathbf{L}^A\right)^T + \alpha_2^B\mathbf{L}^B\left(\tilde{\mathbf{M}}^B\right)^{-1}\left(\mathbf{L}^B\right)^T \tag{2.35}$$

With the Lagrange multiplier at the end of the substep, we can calculate the link solution.

$$\begin{aligned}\tilde{\mathbf{M}}^B\ddot{\mathbf{u}}^B_{n+j/ss_{link}} &= \left(\mathbf{L}^B\right)^T\Lambda_{n+j/ss}\\ \mathbf{u}^B_{n+j/ss_{link}} &= \alpha_1^B\ddot{\mathbf{u}}^B_{n+j/ss_{link}}\\ \dot{\mathbf{u}}^B_{n+j/ss_{link}} &= \alpha_2^B\ddot{\mathbf{u}}^B_{n+j/ss_{link}}\end{aligned} \tag{2.36}$$

Finally, the kinematic quantities in subdomain *B* can be obtained with summing the free solutions and link solutions.

$$\begin{aligned}\mathbf{u}^B_{n+j/ss} &= \mathbf{u}^B_{n+j/ss_{free}} + \mathbf{u}^B_{n+j/ss_{link}}\\ \dot{\mathbf{u}}^B_{n+j/ss} &= \dot{\mathbf{u}}^B_{n+j/ss_{free}} + \dot{\mathbf{u}}^B_{n+j/ss_{link}}\\ \ddot{\mathbf{u}}^B_{n+j/ss} &= \ddot{\mathbf{u}}^B_{n+j/ss_{free}} + \ddot{\mathbf{u}}^B_{n+j/ss_{link}}\end{aligned} \tag{2.37}$$

5. Solve the link problem in subdomain *A*.

$$\begin{aligned}\tilde{\mathbf{M}}^A\ddot{\mathbf{u}}^A_{n+1_{link}} &= \left(\mathbf{L}^A\right)^T\Lambda_{n+1}\\ \mathbf{u}^A_{n+1_{link}} &= \alpha_1^A\ddot{\mathbf{u}}^A_{n+1_{link}}\\ \dot{\mathbf{u}}^A_{n+1_{link}} &= \alpha_2^A\ddot{\mathbf{u}}^A_{n+1_{link}}\end{aligned} \tag{2.38}$$

Similarly, the kinematic quantities in subdomain A can be obtained as follows,

$$\begin{aligned}\mathbf{u}^A_{n+j/ss} &= \mathbf{u}^A_{n+j/ss_{free}} + \mathbf{u}^A_{n+j/ss_{link}}\\ \dot{\mathbf{u}}^A_{n+j/ss} &= \dot{\mathbf{u}}^A_{n+j/ss_{free}} + \dot{\mathbf{u}}^A_{n+j/ss_{link}}\\ \ddot{\mathbf{u}}^A_{n+j/ss} &= \ddot{\mathbf{u}}^A_{n+j/ss_{free}} + \ddot{\mathbf{u}}^A_{n+j/ss_{link}}\end{aligned} \tag{2.39}$$

In the GC method, the stability of the global problem depends on the stability conditions of each subdomain. Its proof was given by Gravouil and Combescure (2001) using an energy norm. The coupling between the Newmark methods does not affect the stability of the algorithms. Moreover, in multi-time-step cases the GC method entails energy dissipation at the interface, while for the case of a single time step in all the subdomains the GC method is energy preserving.

The GC method (Gravouil and Combescure, 2001) as most of available methods, is in essence a sequential staggered algorithm that the integration of subdomain *B* requires information from subdomain *A* in the current time step. Consequently, the process in subdomain(s) with the fine time step has to systematically stop in order to wait for the process in the subdomain(s) with the coarse time step.

2.4.4 The interfield parallel method - the PM method

Pegon and Magonette (2002, 2005) developed and implemented an interfield parallel algorithm, the PM method, based on the GC method, but where the solution of each subdomain states advance simultaneously. In the following, we presented the formulation of the PM method. Consider for simplicity the linear elastic problem to be integrated with two subdomains A and B. In the PM method, a time step equal to $2\Delta t_A$ is exploited on subdomain A, in order to anticipate information on the subdomain B at the beginning of a new coarse time step. As depicted in Fig. 2.6, the solution procedure is described as follows:

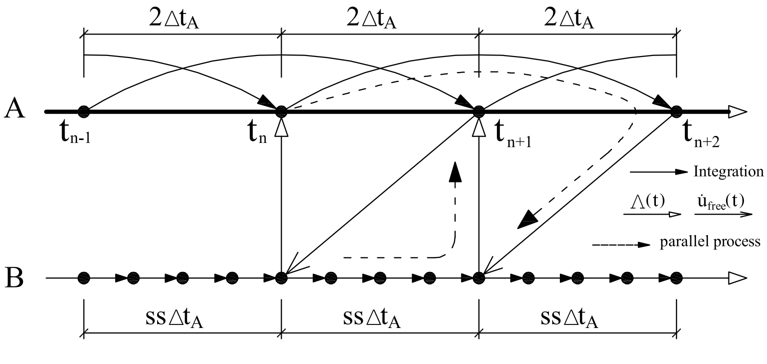


Fig. 2.6: The solution procedure of the PM method

1. Solve the free problem of subdomain *A* using $2\Delta t$ in order to advance from t_{n-1}

to t_{n+1}

$$\begin{aligned}
\tilde{\mathbf{M}}^A \ddot{\mathbf{u}}^A_{n+1_{free}} &= \mathbf{F}^A_{ext,n+1} - \mathbf{K}^A \tilde{\mathbf{u}}^A_{n-1} - \mathbf{C}^A \tilde{\dot{\mathbf{u}}}^A_{n-1} \\
\mathbf{u}^A_{n+1_{free}} &= \tilde{\mathbf{u}}^A_{n-1} + \alpha^A_1 \ddot{\mathbf{u}}^A_{n+1_{free}} \\
\dot{\mathbf{u}}^A_{n+1_{free}} &= \tilde{\dot{\mathbf{u}}}^A_{n-1} + \alpha^A_2 \ddot{\mathbf{u}}^A_{n+1_{free}}
\end{aligned} \tag{2.40}$$

where

$$\begin{aligned}
\tilde{\mathbf{M}}^A &= \mathbf{M}^A + \alpha^A_1 \mathbf{K}^A + \alpha^A_2 \mathbf{C}^A \\
\tilde{\mathbf{u}}^A_n &= \mathbf{u}^A_{n-1} + 2\Delta t \dot{\mathbf{u}}^A_{n-1} + (2\Delta t)^2 \left(\frac{1}{2} - \beta_A\right) \ddot{\mathbf{u}}^A_{n-1} \\
\tilde{\dot{\mathbf{u}}}^A_n &= \dot{\mathbf{u}}^A_{n-1} + 2\Delta t \left(1 - \gamma_A\right) \ddot{\mathbf{u}}^A_{n-1} \\
\alpha^A_1 &= \beta_A \left(2\Delta t\right)^2 \\
\alpha^A_2 &= \gamma_A 2\Delta t
\end{aligned} \tag{2.41}$$

2. Interpolate of the free velocity in subdomain A between t_n and t_{n+1}

$$\dot{\mathbf{u}}^A_{n+j/ss_{free}} = \left(1 - \frac{j}{ss}\right) \dot{\mathbf{u}}^A_{n_{free}} + \frac{j}{ss} \dot{\mathbf{u}}^A_{n+1_{free}} \tag{2.42}$$

3. Start of the loop with *ss* substeps in subdomain B

4. Solve the free problem in subdomain B at $t_{n+j/ss}$ with $j = \{1, ..., ss\}$

$$\begin{aligned}
\tilde{\mathbf{M}}^B \ddot{\mathbf{u}}^B_{n+j/ss_{free}} &= \mathbf{F}^B_{ext,n+j/ss} - \mathbf{K}^B \tilde{\mathbf{u}}^B_{n+(j-1)/ss} - \mathbf{C}^B \tilde{\dot{\mathbf{u}}}^B_{n+(j-1)/ss} \\
\mathbf{u}^B_{n+j/ss_{free}} &= \tilde{\mathbf{u}}^B_{n+(j-1)/ss} + \alpha^B_1 \ddot{\mathbf{u}}^B_{n+j/ss_{free}} \\
\dot{\mathbf{u}}^B_{n+j/ss_{free}} &= \tilde{\dot{\mathbf{u}}}^B_{n+(j-1)/ss} + \alpha^B_2 \ddot{\mathbf{u}}^B_{n+j/ss_{free}}
\end{aligned} \tag{2.43}$$

where the relative parameters are equal to those of the GC method in subdomain *B* as shown in (2.33).

5. Calculate the Lagrange multiplier vector $\mathbf{\Lambda}_{n+j/ss}$

$$\mathbf{H}\mathbf{\Lambda}_{n+j/ss} = -\left(\mathbf{L}^A \dot{\mathbf{u}}^A_{n+j/ss_{free}} + \mathbf{L}^B \dot{\mathbf{u}}^B_{n+j/ss_{free}}\right) \tag{2.44}$$

where

$$\mathbf{H}\mathbf{\Lambda}_{n+j/ss} = -\left(\mathbf{L}^A \dot{\mathbf{u}}^A_{n+j/ss_{free}} + \mathbf{L}^B \dot{\mathbf{u}}^B_{n+j/ss_{free}}\right) \tag{2.45}$$

6. Solve of the link problem on subdomain B at $t_{n+j/ss}$

$$\begin{aligned}
\tilde{\mathbf{M}}^B \ddot{\mathbf{u}}^B_{n+j/ss_{link}} &= \left(\mathbf{L}^B\right)^T \Lambda_{n+j/ss} \\
\mathbf{u}^B_{n+j/ss_{link}} &= \alpha^B_1 \ddot{\mathbf{u}}^B_{n+j/ss_{link}} \\
\dot{\mathbf{u}}^B_{n+j/ss_{link}} &= \alpha^B_2 \ddot{\mathbf{u}}^B_{n+j/ss_{link}}
\end{aligned} \tag{2.46}$$

7. Calculate the kinematic quantities of subdomain B at $t_{n+j/ss}$

$$\begin{aligned}
\mathbf{u}^B_{n+j/ss} &= \mathbf{u}^B_{n+j/ss_{free}} + \mathbf{u}^B_{n+j/ss_{link}} \\
\dot{\mathbf{u}}^B_{n+j/ss} &= \dot{\mathbf{u}}^B_{n+j/ss_{free}} + \dot{\mathbf{u}}^B_{n+j/ss_{link}} \\
\ddot{\mathbf{u}}^B_{n+j/ss} &= \ddot{\mathbf{u}}^B_{n+j/ss_{free}} + \ddot{\mathbf{u}}^B_{n+j/ss_{link}}
\end{aligned} \tag{2.47}$$

If $j = ss$, then end the loop in subdomain B

8. Solution of the problem link in subdomain A at t_{n+1}

$$\begin{aligned}
\tilde{\mathbf{M}}^A \ddot{\mathbf{u}}^A_{n+1_{link}} &= \left(\mathbf{L}^A\right)^T \Lambda_{n+1} \\
\mathbf{u}^A_{n+1_{link}} &= \alpha^A_1 \ddot{\mathbf{u}}^A_{n+1_{link}} \\
\dot{\mathbf{u}}^A_{n+1_{link}} &= \alpha^A_2 \ddot{\mathbf{u}}^A_{n+1_{link}}
\end{aligned} \tag{2.48}$$

9. Calculate the kinematic quantities of subdomain A at t_{n+1}

$$\begin{aligned}
\mathbf{u}^A_{n+j/ss} &= \mathbf{u}^A_{n+j/ss_{free}} + \mathbf{u}^A_{n+j/ss_{link}} \\
\dot{\mathbf{u}}^A_{n+j/ss} &= \dot{\mathbf{u}}^A_{n+j/ss_{free}} + \dot{\mathbf{u}}^A_{n+j/ss_{link}} \\
\ddot{\mathbf{u}}^A_{n+j/ss} &= \ddot{\mathbf{u}}^A_{n+j/ss_{free}} + \ddot{\mathbf{u}}^A_{n+j/ss_{link}}
\end{aligned} \tag{2.49}$$

As shown in Fig. 2.6, two processes (the dashed lines) are advanced independently and simultaneously; and the information exchanges are only required at the end of the loop. This method is therefore parallel but may amplify the errors in both subdomains (Pegon and Magonette, 2002). Moreover, the energy approach is not available for the PM method and the spectral analysis was done for this method by (He, 2008). And the method was discovered to be stable if the stability requirement in each subdomain could be satisfied.

2.5 Summary

The state of the art to the work of the thesis was provided in this chapter. Initially, a brief overview of several well-established testing methods used in earthquake engineering was provided, and among them, the PsD testing with dynamic substructuring and the RTDS method were stated in terms of historical development, advantages, limitation as well as some challenges. Then, several time integration algorithms involved in RTDS tests and PsD tests were introduced within three families: explicit, implicit and linearly implicit. Lastly, Section 2.4 reviewed some techniques related to partitioned time integration methods, such as domain decomposition methods and the FETI method, and elaborated two partitioned integration methods, i.e. the GC method and the PM method.

CHAPTER 3

ANALYSIS OF L-STABLE REAL TIME COMPATIBLE ALGORITHMS

3.1 Introduction

Most of the research works carried out on substructure tests considered structural integrators for the equations of motion second-order in time. Nonetheless, it is well known that the motion of the PS in a substructure test is driven by a transfer system -actuators- and sensors, governed by a control unit. Since the control system is typically described by first-order Differential-Algebraic Equations (DAEs), the utilized integrators have to deal with mixed first- and second-order DAEs. In order to solve this problem, there are mainly three options: i) to use different integrators for structural and control systems, respectively, -see for instance Wu et al. (2007), that utilizes the Newmark-β method for the emulated structure and a proprietary MTS controller with its built-in time integrator; ii) to reformulate the control equations into a second-order form (Brüls and Golinval, 2006), and employ a structural integrator like the Generalized-α (Chung and Hulbert, 1993) for both systems; iii) to use first-order integrators like the Rosenbrock algorithms, for both structural and control systems. Herein, this thesis adopts the last option owing to the favourable properties of LSRT algorithms employed in control (Vulcan, 2006).

The utilized LSRT algorithms belong to the class of the linearly implicit Rosenbrock methods which combines positive properties of the explicit and implicit methods. Furthermore, the algorithms are L-stable which enable high-frequency dissipation to be achieved without sacrificing low-frequency accuracy. Besides, the LSRT algorithms enjoy great popularity due to their simplicity of implementation for nonlinear problems.

In this chapter, the novel L-Stable Real-time (LSRT) compatible algorithms derived from the Rosenbrock methods (Bursi et al., 2008) are introduced. In particular, the one- and two-stage methods are analysed in terms of accuracy and stability, and their performance for nonlinear system is investigated through numerical experiments on an uncoupled spring-pendulum system.

3.2 Rosenbrock based algorithms

This section mainly summarizes three *real-time compatible* algorithms, *i.e.* the LSRT methods, which have been recently introduced (Bursi et al., 2008) in order to perform RTDS tests. In detail, real-time compatibility implies that to obtain the solution of the differential equation $\dot{\mathbf{y}} = \mathbf{F}(\mathbf{y}, t)$, a numerical method does not require the knowledge of the value of $\mathbf{F}$ or its derivatives at the end of the time step Δt. Moreover, to be compatible with dSPACE which operates in real time (dSPACE, 2001), the integrator has to be characterized by intermediate substeps sampled with a sample rate which is an integer number of the base sample rate. Following (Geradin and Rixen, 1997, p.389), the semi-discrete equations of motion for a generic nonlinear structure can be expressed in a general form:

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{f}\left(\mathbf{x}, \dot{\mathbf{x}}\right) = \mathbf{g}\left(t\right) \tag{3.1}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{x}$ stands for the displacement vector while $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ represent the corresponding velocity and acceleration vectors. The term $\mathbf{f}\left(\mathbf{x}, \dot{\mathbf{x}}\right)$ represents the internal force vector including the elastic forces and the internal dissipation forces, and the expression $\mathbf{g}$ defines the external force vector. Both $\mathbf{M}$ and $\mathbf{g}\left(t\right)$ do not depend on the configuration.

Driven by the idea to use only one integrator both for dynamic and control equations (Samin et al., 2007), (3.1) can be reformulated in Hamilton form:

$$\dot{\mathbf{y}} = \mathbf{F}(\mathbf{y}, t) = \left\{ \begin{array}{c} \mathbf{y}_2 \\ \mathbf{M}^{-1}\left(\mathbf{g}(t) - \mathbf{f}\left(\mathbf{y}_1, \mathbf{y}_2\right)\right) \end{array} \right\} \tag{3.2}$$

where $\mathbf{y} = \left\{\mathbf{y}_1^T \;\; \mathbf{y}_2^T\right\}^T = \left\{\mathbf{x}^T \;\; \dot{\mathbf{x}}^T\right\}^T$ defines the state vector.

The solution of (3.2) by means of an s-stage Rosenbrock method (Rosenbrock, 1963) is given by:

$$\begin{gathered}\mathbf{y}_{k+1} = \mathbf{y}_k + \textstyle\sum_{i=1}^{s} b_i \mathbf{k}_i \\ \mathbf{k}_i = [\mathbf{I} - \gamma \Delta t \mathbf{J}]^{-1} \left(\mathbf{F}\left(t_k + \alpha_i \Delta t, \mathbf{y}_k + \textstyle\sum_{j=1}^{i-1} \alpha_{ij}\mathbf{k}_j\right) + \mathbf{J}\textstyle\sum_{j=1}^{i-1} \gamma_{ij}\mathbf{k}_j\right)\Delta t\end{gathered} \tag{3.3}$$

where $\alpha_i = \sum_{j=1}^{i-1} \alpha_{ij}$, γ_{ij} and b_i are the algorithm coefficients, and **J** is the Jacobian matrix which is defined as

$$\mathbf{J} = \frac{\partial \mathbf{F}}{\partial \mathbf{y}} \tag{3.4}$$

This method is very suitable for the numerical solution of stiff and nonlinear problems (Verwer, 1982; Arnold et al., 2007b). To achieve so, it requires only a single linearization and matrix decomposition per time step, whilst **J** is formed only at the beginning of each Δt. Thus, the method is linearly implicit (Hairer and Wanner, 1996, p.102). In real-time testing, the Jacobian operator **J** is formed only on the basis of the NS, but other choices of **J** can improve the performance of these methods (Steihaug and Wolfbrandt, 1979; Lamarche et al., 2009).

In order to achieve real-time compatible, It is assumed that $\alpha_i = \sum\limits_{j=1}^{i-1} \alpha_{ij}$ so that the function f and its derivative at beginning of every inner stage only depend on the known solutions and coupling forces solved before. This summation also implies $\alpha_i = 0$ for $i = 1$.

3.2.1 L-stable real-time one-stage (LSRT1) method

The one-stage real-time method when applied to the differential equation (3.2) exploits the following formulas:

$$\mathbf{k}_1 = [\mathbf{I} - \gamma \Delta t \mathbf{J}]^{-1} \mathbf{f}(\mathbf{y}_k, t_k)\Delta t, \ \mathbf{y}_{k+1} = \mathbf{y}_k + b_1 \mathbf{k}_1 \tag{3.5}$$

The method is first-order accurate. The conditions necessary to achieve A-stability and L-stability (Hairer and Wanner, 1996) are $\gamma \geq \frac{1}{2}$ and $\gamma = 1$, respectively.

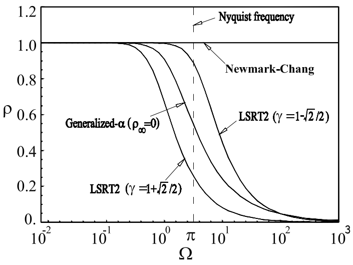


Fig. 3.1: Spectral radii ρ of the LSRT2 method with respect to the Generalized-α method and Newmark-Chang method vs. the non-dimensional frequency Ω

3.2.2 L-stable real-time two-stage (LSRT2) method

The two-stage real-time method applied to (3.2) can be expressed as:

$$\mathbf{k}_1 = [\mathbf{I} - \gamma \Delta t \mathbf{J}]^{-1} \, \mathbf{f}\,(\mathbf{y}_k, t_k)\, \Delta t, \mathbf{y}_{k+\alpha_{21}} = \mathbf{y}_k + \alpha_{21} \mathbf{k}_1 \tag{3.6}$$

$$\mathbf{k}_2 = [\mathbf{I} - \gamma \Delta t \mathbf{J}]^{-1} \left(\mathbf{f}\left(\mathbf{y}_{k+\alpha_{21}}, t_{k+\alpha_2}\right) + \mathbf{J}\gamma_{21}\mathbf{k}_1 \right) \Delta t, \mathbf{y}_{k+\alpha_2} = \mathbf{y}_k + b_1 \mathbf{k}_1 + b_2 \mathbf{k}_2 \tag{3.7}$$

For LSRT2 algorithm two sets of parameters are introduced that satisfy second-order accuracy, L-stability and real-time compatibility: $\gamma = 1 - \sqrt{2}/2$ and $\gamma = 1 + \sqrt{2}/2$, together with $\alpha_2 = \alpha_{21} = 1/2$, $\gamma_{21} = -\gamma$, $b_1 = 0$ and $b_2 = 1$. The favourable performance of the LSRT2 method with respect to the low and high-frequency components of the response can be observed from Fig. 3.1, where a comparison with the Generalized-α method (Chung and Hulbert, 1993) and Newmark-Chang method (Chang, 2002) is also illustrated.

3.2.3 L-stable real-time three-stage (LSRT3) method

The LSRT3 method applied to (3.2) can be expressed as:

$$\begin{aligned}
\mathbf{y}_{n+1} &= \mathbf{y}_n + b_1 \mathbf{k}_1 + b_2 \mathbf{k}_2 + b_3 \mathbf{k}_3 \\
\mathbf{k}_1 &= (\mathbf{I} - \gamma \Delta t \mathbf{J})^{-1} \, \mathbf{f}\,(t_n, \mathbf{y}_n)\, \Delta t \\
\mathbf{k}_2 &= (\mathbf{I} - \gamma \Delta t \mathbf{J})^{-1} \, (\mathbf{f}\,(t_n + \alpha_2 \Delta t, \mathbf{y}_n + \alpha_{21}\mathbf{k}_1) + \gamma_{21}\mathbf{k}_1\mathbf{J})\, \Delta t \\
\mathbf{k}_3 &= (\mathbf{I} - \gamma \Delta t \mathbf{J})^{-1} \, (\mathbf{f}\,(t_n + \alpha_3 \Delta t, \mathbf{y}_n + \alpha_{31}\mathbf{k}_1 + \alpha_{32}\mathbf{k}_2) + \gamma_{31}\mathbf{k}_1\mathbf{J} + \gamma_{32}\mathbf{k}_2\mathbf{J})\, \Delta t
\end{aligned} \tag{3.8}$$

In order to ensure third-order accuracy, L-stability and real-time compatibility, it is assumed that

$$\begin{aligned} b_1 = 1/4, b_2 = 0, b_3 = 3/4 \\ \alpha_2 = \alpha_{21} = 1/3, \alpha_3 = 2/3, \alpha_{31} = \alpha_{32} = 1/3 \\ \gamma_{21} = 0.1818, \gamma_{31} = -0.0428, \gamma_{32} = -0.5384, \gamma = 0.4359 \end{aligned} \tag{3.9}$$

Although this algorithm is third-order accurate for uncoupled and linear system, it is relatively complex. Therefore, it is presented here only for the monolithic case rather than the partitioned case.

It was proved by Bursi (2008) that these algorithms are unconditionally stable for uncoupled problems and entail a moderate computational cost for real-time performance. Generally, these algorithms entail five beneficial properties: i) they can be implemented in a real-time environment and exploit evenly spaced substeps; ii) they can deal with stiff systems relying on the L-stability property; iii) they do not exhibit overshoot in the velocity for large time step; (iv) they do not require the computation of the exponential of the system matrix and are easy to implement with few stages in view of real-time performance; (v) they predict explicitly the state, thus in principle, they give a better control of the acceleration of the transfer system. In addition, the LSRT algorithms were shown to be more competitive than popular Runge-Kutta methods in terms of stability, accuracy and ease of implementation.

3.3 Accuracy analysis

Some of the parameters of the utilized LSRT algorithms are determined by their required accuracy condition. Here, I reanalyse their accuracy in order to make a preliminary preparation for the accuracy analysis of the partitioned methods.

In the accuracy analysis, two types of errors are usually considered: local errors and global errors. The local error is the error introduced in a single step given the exact initial solution; while the global error is the overall error caused by repeated application of the integration approximation. Obviously, the global error is more desirable because it indicates the difference between the numerical solution and the exact solution. However, the global error is determined by local errors associated with the accumulation of local errors at each time step. Thus, we mainly investigate local errors of the partitioned methods. Local errors and global errors are sometimes defined to include round-off errors. However, round-off errors are beyond the scope of this thesis.

3.3.1 Local truncation error analysis for the LSRT1 algorithm

Here the Local truncation error of the LSRT1 method is analysed. We will let τ denote the local truncation error, and $\mathbf{y}_k$ be the exact solution.

For the sake of generality, the parameter γ of the LSRT algorithm is viewed as a variable so that not only the L-stable algorithms but also the A-stable algorithms are investigated.

The numerical solution $\mathbf{y}_{k+1}$ solved by the LSRT1 algorithm can be written in the following form

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{k}_1 = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{f}\,(t_k, y_k)\,\Delta t \tag{3.10}$$

Let us return to the Taylor series approximation and use it to estimate the error in the approximation of the LSRT1 algorithm. If we assume that we have all the data ($\mathbf{y}_k$ and its derivatives) at t_k, then the exact solution at t_{k+1} is given, according to the first-order Taylor series expansion, as follows

$$\mathbf{y}\,(t_{k+1}) = \mathbf{y}_k + \mathbf{f}_k\Delta t + \mathrm{O}\left(\Delta t^2\right) \tag{3.11}$$

By comparing (3.10) with (3.11), we can find that the numerical solution to the system (3.2) satisfies

$$\begin{aligned}\tau_{k+1}^{\mathbf{y}} &= \mathbf{y}_{k+1} - \mathbf{y}\,(t_{k+1}) = [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{f}_k\Delta t - \mathbf{f}_k\Delta t\\ &= \Delta t\gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{f}_k\Delta t + \mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{3.12}$$

which indicates that the algorithm is locally second-order accurate.

3.3.2 Local truncation error analysis for the LSRT2 algorithm

Similarly, we assume that $\mathbf{y}_k = \mathbf{y}(t_k)$ and leave the parameter γ as variable. Inserting all other parameters into (3.7) we can obtain:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t \mathbf{J}\gamma]^{-1} \left(\mathbf{f}\left(t_k + \frac{1}{2}\Delta t, \mathbf{y}_k + \frac{1}{2}\mathbf{k}_1 \right) + \mathbf{J}\gamma\mathbf{k}_1 \right) \Delta t \tag{3.13}$$

In order to investigate the local truncation error, we can simplify (3.13) into a formula which only contains $\mathbf{y}$ and its derivatives at t_k. Therefore, we adopt the first-order Taylor's series expansion of $\mathbf{f}\left(t_k + \frac{1}{2}\Delta t, \mathbf{y}_k + \frac{1}{2}\mathbf{k}_1\right)$ about t_k:

$$\begin{aligned} \mathbf{f}\left(t_k + \tfrac{1}{2}\Delta t, \mathbf{y}_k + \tfrac{1}{2}\mathbf{k}_1\right) &= \mathbf{f}_k + \tfrac{\Delta t}{2}\tfrac{\partial \mathbf{f}}{\partial t}\Big|_k + \tfrac{\mathbf{k}_1}{2}\left.\tfrac{\partial \mathbf{f}}{\partial \mathbf{y}}\right|_k + \mathrm{O}\left(\Delta t^2\right) \\ &= \mathbf{f}_k + \tfrac{\Delta t}{2}\tfrac{\partial \mathbf{f}}{\partial t}\Big|_k + \tfrac{\Delta t}{2}[\mathbf{I} - \Delta t \mathbf{J}\gamma]^{-1}\mathbf{J}\mathbf{f}_k + \mathrm{O}\left(\Delta t^2\right) \end{aligned} \tag{3.14}$$

Now, Eq. (3.13) can be rewritten as:

$$\mathbf{y}_{k+1} = y_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1} \left(\begin{array}{c} \mathbf{f}_k + \tfrac{\Delta t}{2}\tfrac{\partial \mathbf{f}}{\partial t}\Big|_k + \tfrac{\Delta t}{2}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\mathbf{J}\mathbf{f}_k - \\ -\Delta t\gamma\mathbf{J}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\mathbf{f}_k + \mathrm{O}\left(\Delta t^2\right) \end{array} \right) \Delta t \tag{3.15}$$

We then expand the Taylor's series of the exact solution $\mathbf{y}\left(t_{k+1}\right)$ up to the second order:

$$\mathbf{y}\left(t_{k+1}\right) = \mathbf{y}_k + \mathbf{f}_k\Delta t + \frac{\Delta t^2}{2}\left(\mathbf{f}_k \left.\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\right|_k + \left.\frac{\partial \mathbf{f}}{\partial t}\right|_k \right) + \mathrm{O}\left(\Delta t^3\right) \tag{3.16}$$

Comparing Eq. (3.15) with Eq. (3.16), we can get the local truncation error of this method:

$$\tau^{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \mathbf{y}\left(t_{k+1}\right) = \mathrm{O}\left(\Delta t^3\right) \tag{3.17}$$

which indicates a local truncation of order $\mathrm{O}\left(\Delta t^3\right)$.

Since the solution at the end of the first stage is required to calculate the Lagrange multipliers in the partitioned methods. Here its local truncation error is analysed as well. For LSRT2, the solution at $t_k + \frac{1}{2}\Delta t$ is:

$$\mathbf{y}_{k+\frac{1}{2}} = \mathbf{y}_k + \frac{1}{2}\mathbf{k}_1 = \mathbf{y}_k + \frac{1}{2}[1 - \Delta t\gamma\mathbf{J}]^{-1}\mathbf{f}\left(t_k, \mathbf{y}_k\right)\Delta t \tag{3.18}$$

From the Taylor series expansion, we then expand $y\left(t_{k+1/2}\right)$ up to the first order:

$$\mathbf{y}\left(t_{k+\frac{1}{2}}\right) = \mathbf{y}_k + \mathbf{f}_k\frac{\Delta t}{2} + \mathrm{O}\left(\Delta t^2\right) \tag{3.19}$$

Thus, we can obtain:

$$\tau^{\mathbf{y}}_{k+\frac{1}{2}} = \mathrm{O}\left(\Delta t^2\right) \tag{3.20}$$

which implies that the middle point only has locally second-order accuracy.

3.3.3 Global error analysis for the LSRT algorithm

When applied to linear problems, the utilized algorithms can be recast into a recursive form as

$$\mathbf{y}_{k+1} = \mathbf{R}\mathbf{y}_k + \mathbf{L}_k \tag{3.21}$$

where $\mathbf{R}$ is the amplification matrix and $\mathbf{L}_k$ is the load vector that depends on the external forces, respectively. Actually, the global error contains the local truncation error of the current step and the propagation of the previous errors, i.e.,

$$e_{k+1} = \mathbf{y}_{k+1} - \mathbf{y}\left(t_{k+1}\right) = \mathbf{R}\mathbf{y}_k + \mathbf{L}_k - \mathbf{y}\left(t_{k+1}\right) = \mathbf{R}\left(\mathbf{y}_k - \mathbf{y}\left(t_k\right)\right) + \mathbf{R}\mathbf{y}\left(t_k\right) + \mathbf{L}_k - \mathbf{y}\left(t_{k+1}\right) = \mathbf{R}e_k + \tau_{k+1} \tag{3.22}$$

Thus,

$$e_{k+1} \leq \mathbf{R}^n e_0 + \sum_{i=1}^{n-1} \mathbf{R}^{n-i}\tau_i \leq \frac{\mathbf{R}^n - 1}{\mathbf{R} - 1} \underset{i=1}{\overset{n}{Max}}\, \tau_i \tag{3.23}$$

Let us take the LSRT2 method for example. According to Eq. (3.13), we can simplify its amplification matrix as

$$\mathbf{R} = \mathbf{I} + \mathbf{\Phi}\Delta t \tag{3.24}$$

where the unitary matrix is corresponding to the first term at the RHS of Eq. (3.13) and $\mathbf{\Phi}\Delta t$ stands for the second term.

Thus, we can rewrite the global error as

$$e_{k+1} \leq \frac{\mathbf{R}^n - 1}{\mathbf{\Phi}\Delta t}\mathrm{O}\left(\Delta t^3\right) = \mathrm{O}\left(\Delta t^2\right) \tag{3.25}$$

which indicates that the LSRT2 algorithm is second-order accurate.

Roughly speaking, if the local error is $\mathrm{O}\left(\Delta t^p\right)$, then the global error will be $\mathrm{O}\left(\Delta t^{p-1}\right)$. Meanwhile, we can conclude that the LSRT1 algorithm is first-order accurate, and the LSRT2 algorithm is second-order accurate.

3.4 Stability analysis via the energy method

For physical reasons, it is desired that numerical solutions of stable system are bounded (Wood, 1990). An algorithm which exhibits this behavior is said to be stable. The stability of the LSRT algorithms have been analysed through the model approach by Vulcan (2006). In order to further realize the decaying properties of the algorithms, we introduce the energy method that guarantees the preservation of the total energy of the system.

For the stability analysis, it suffices to restrict attention to an autonomous second order problem in the linear region, characterized by the differential equation

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = 0 \tag{3.26}$$

In this equation $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are the mass, damping and stiffness matrices of the system, respectively, while $\mathbf{u}(t)$, $\dot{\mathbf{u}}(t)$ and $\ddot{\mathbf{u}}(t)$ are respectively the displacement, velocity and acceleration vectors. In order to analyse the stability properties and investigate the numerical damping of the LSRT algorithms, we assume no physical damping, i.e., $\mathbf{C} = 0$. Eq. (3.26) can be reformulated in Hamilton form:

$$\dot{\mathbf{y}} = \mathbf{J}\mathbf{y} \tag{3.27}$$

where

$$\mathbf{J} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & 0 \end{bmatrix}, \mathbf{y} = \begin{Bmatrix} \mathbf{u}(t) \\ \dot{\mathbf{u}}(t) \end{Bmatrix}, \dot{\mathbf{y}} = \begin{Bmatrix} \dot{\mathbf{u}}(t) \\ \ddot{\mathbf{u}}(t) \end{Bmatrix} \tag{3.28}$$

The discrete form of the energy balance equation involves the increment of the mechanical energy over the time interval from t_k to t_{k+1} . The increment can be expressed in terms of mean values and increments of the displacement and velocity by the following formula:

$$\Delta\mathbf{E}_{k+1} = (\mathbf{u}_{k+1} - \mathbf{u}_k)^T \mathbf{K}(\mathbf{u}_{k+1} + \mathbf{u}_k)/2 + (\dot{\mathbf{u}}_{k+1} - \dot{\mathbf{u}}_k)^T \mathbf{M}(\dot{\mathbf{u}}_{k+1} + \dot{\mathbf{u}}_k)/2 \tag{3.29}$$

which is the sum of the former potential energy increment and the latter kinetic energy increment. It is assumed that the mass matrix $\mathbf{M}$ and $\mathbf{K}$ are symmetric and positive definite. In order to deal with the first-order algorithms, Eq. (3.29) is transformed into:

$$\Delta\mathbf{E}_{k+1} = \frac{1}{2}\left\{\begin{array}{c}\mathbf{u}_{k+1}-\mathbf{u}_k\\ \dot{\mathbf{u}}_{k+1}-\dot{\mathbf{u}}_k\end{array}\right\}^T\begin{bmatrix}\mathbf{K} & 0\\ 0 & \mathbf{M}\end{bmatrix}\left\{\begin{array}{c}\mathbf{u}_{k+1}+\mathbf{u}_k\\ \dot{\mathbf{u}}_{k+1}+\dot{\mathbf{u}}_k\end{array}\right\} = \frac{1}{2}\left(\mathbf{y}_{k+1}-\mathbf{y}_k\right)^T\mathbf{P}\left(\mathbf{y}_{k+1}+\mathbf{y}_k\right) \tag{3.30}$$

Here, **P** is a sparse matrix which contains two positive definite matrices, and therefore is positive definite. Before the detailed stability analysis, we state the following formulae:

$$\begin{aligned}\mathbf{PJ} &= \begin{bmatrix}\mathbf{K} & 0\\ 0 & \mathbf{M}\end{bmatrix}\begin{bmatrix}0 & \mathbf{I}\\ -\mathbf{M}^{-1}\mathbf{K} & 0\end{bmatrix} = \begin{bmatrix}0 & \mathbf{K}\\ -\mathbf{K} & 0\end{bmatrix}\\ \mathbf{PJ}^{-1} &= \begin{bmatrix}\mathbf{K} & 0\\ 0 & \mathbf{M}\end{bmatrix}\begin{bmatrix}0 & \mathbf{I}\\ -\mathbf{M}^{-1}\mathbf{K} & 0\end{bmatrix}^{-1} = \begin{bmatrix}0 & -\mathbf{M}\\ \mathbf{M} & 0\end{bmatrix}\end{aligned} \tag{3.31}$$

$$\left\{\begin{array}{c}\mathbf{x}_1\\ \mathbf{x}_2\end{array}\right\}^T\begin{bmatrix}0 & -\mathbf{Y}\\ \mathbf{Y} & 0\end{bmatrix}\left\{\begin{array}{c}\mathbf{x}_1\\ \mathbf{x}_2\end{array}\right\} = 0 \tag{3.32}$$

which indicates that the matrices **PJ** and $\mathbf{PJ}^{-1}$ are zero-definite. These formulae are used to simplify the energy balance equations in Subsection 3.4.1 and 3.4.2.

In the following subsections energy balance equations are derived for the LSRT1 and LSRT2 algorithms. Through splitting the expressions into mean values and increments and reformulating both into linear functions of a specific vector, the energy balance equations can be transformed into a quadratic form of the specific vector multiplied by a polynomial of γ which determine the sign of the equation.

3.4.1 Stability analysis for the LSRT1 algorithm

Since an inverse matrix is included in Eq. (3.5), we rewrite it into:

$$\mathbf{y}_k = \frac{1}{\Delta t}\mathbf{J}^{-1}\left[\mathbf{I}-\Delta t\mathbf{J}\gamma\right]\mathbf{k}_1 = \frac{1}{\Delta t}\left[\mathbf{J}^{-1}-\Delta t\gamma\mathbf{I}\right]\mathbf{k}_1 \tag{3.33}$$

Therefore, the energy balance equation can be rewritten:

$$\begin{aligned}\Delta\mathbf{E}_{k+1} &= \tfrac{1}{2}\mathbf{k}_1^T\mathbf{P}\left(2\mathbf{y}_k+\mathbf{k}_1\right)\\ &= \tfrac{1}{2}\mathbf{k}_1^T\mathbf{P}\left(2\tfrac{1}{\Delta t}\left[\mathbf{J}^{-1}-\Delta t\gamma\mathbf{I}\right]\mathbf{k}_1+\mathbf{k}_1\right)\\ &= \tfrac{1}{2}\mathbf{k}_1^T\mathbf{P}\left(-2\gamma+1\right)\mathbf{k}_1+\tfrac{1}{\Delta t}\mathbf{k}_1^T\mathbf{PJ}^{-1}\mathbf{k}_1\\ &= \left(-\gamma+\tfrac{1}{2}\right)\mathbf{k}_1^T\mathbf{P}\mathbf{k}_1\end{aligned} \tag{3.34}$$

For the algorithm to be stable the energy increment must be negative semi-definite, i.e., zero or negative-definite. Obviously, if $\gamma = 1/2$, $\Delta\mathbf{E}_{k+1} = 0$ and therefore the algorithm is conservative; if $\gamma > 1/2$, the algorithm introduces positive dissipation and is therefore unconditionally stable; if $\gamma < 1/2$, the algorithm provides negative dissipation and is therefore unconditionally unstable.

3.4.2 Stability analysis for the LSRT2 algorithm

Similarly, we leave the parameter γ as variable and adopt all other parameters as shown in Subsection 3.2.2. The first step is to state the energy increment over the first stage form t_k to $t_{k+1/2}$. The increment can be expressed by the following formula:

$$\Delta\mathbf{E}_{k+1/2} = \left(\mathbf{u}_{k+1/2} - \mathbf{u}_k\right)^T \mathbf{K} \left(\mathbf{u}_{k+1/2} + \mathbf{u}_k\right)/2 + \left(\dot{\mathbf{u}}_{k+1/2} - \dot{\mathbf{u}}_k\right)^T \mathbf{M} \left(\dot{\mathbf{u}}_{k+1/2} + \dot{\mathbf{u}}_k\right)/2 \quad (3.35)$$

Similarly, Eq. (3.35) is transformed into:

$$\Delta\mathbf{E}_{k+1/2} = \frac{1}{2}\left\{\begin{array}{c}\mathbf{u}_{k+1/2} - \mathbf{u}_k \\ \dot{\mathbf{u}}_{k+1/2} - \dot{\mathbf{u}}_k\end{array}\right\}^T \left[\begin{array}{cc}\mathbf{K} & 0 \\ 0 & \mathbf{M}\end{array}\right] \left\{\begin{array}{c}\mathbf{u}_{k+1/2} + \mathbf{u}_k \\ \dot{\mathbf{u}}_{k+1/2} + \dot{\mathbf{u}}_k\end{array}\right\} = \frac{1}{2}\left(\mathbf{y}_{k+1/2} - \mathbf{y}_k\right)^T \mathbf{P} \left(\mathbf{y}_{k+1/2} + \mathbf{y}_k\right) \quad (3.36)$$

To continue, we rewrite Eq. (3.6) into:

$$\mathbf{y}_k = \frac{1}{\Delta t}\mathbf{J}^{-1}\left[\mathbf{I} - \Delta t\mathbf{J}\gamma\right]\mathbf{k}_1 = \frac{1}{\Delta t}\left[\mathbf{J}^{-1} - \Delta t\gamma\mathbf{I}\right]\mathbf{k}_1 \quad (3.37)$$

Therefore, we obtain:

$$\begin{aligned}
\Delta\mathbf{E}_{k+1/2} &= \tfrac{1}{2}\left(\tfrac{1}{2}\mathbf{k}_1\right)^T \mathbf{P}\left(2\mathbf{y}_k + \tfrac{1}{2}\mathbf{k}_1\right) \\
&= \tfrac{1}{4}\mathbf{k}_1^T\mathbf{P}\left(2\tfrac{1}{\Delta t}\left[\mathbf{J}^{-1} - \Delta t\gamma\mathbf{I}\right]\mathbf{k}_1 + \tfrac{1}{2}\mathbf{k}_1\right) \\
&= \tfrac{1}{4}\mathbf{k}_1^T\mathbf{P}\left(-2\gamma + \tfrac{1}{2}\right)\mathbf{k}_1 + \tfrac{1}{2\Delta t}\mathbf{k}_1^T\mathbf{P}\mathbf{J}^{-1}\mathbf{k}_1 \\
&= \tfrac{1}{2}\left(-\gamma + \tfrac{1}{4}\right)\mathbf{k}_1^T\mathbf{P}\mathbf{k}_1
\end{aligned} \quad (3.38)$$

Evidently, we can conclude: if $\gamma = 1/4$, the energy in the first stage is conservative; if $\gamma > 1/4$, the integration in the first stage provides positive dissipation; if $\gamma < 1/4$, the solution in the first stage provides negative dissipation.

The increment of the mechanical energy over the time interval from t_k to t_{k+1} can be expressed as:

$$\Delta\mathbf{E}_{k+1} = \frac{1}{2}\left(\mathbf{y}_{k+1} - \mathbf{y}_k\right)^T\mathbf{P}\left(\mathbf{y}_{k+1} + \mathbf{y}_k\right) = \frac{1}{2}\mathbf{k}_2^T\mathbf{P}\left(2\mathbf{y}_k + \mathbf{k}_2\right) \quad (3.39)$$

In order to avoid inverse matrices, the following transformations are required:

$$\mathbf{k}_2 = [\mathbf{I} - \Delta t \mathbf{J}\gamma]^{-1} \left(\mathbf{J} \left(\mathbf{y}_k + \frac{1}{2}\mathbf{k}_1 \right) \Delta t - \Delta t \mathbf{J} \gamma \mathbf{k}_1 \right) \tag{3.40}$$

Thus,

$$\mathbf{y}_k = \frac{1}{\Delta t} \left[\mathbf{J}^{-1} - \Delta t \gamma \mathbf{I} \right] \mathbf{k}_2 - \frac{1}{2}\mathbf{k}_1 + \gamma \mathbf{k}_1 \tag{3.41}$$

To continue, the energy balance equation can be rewritten:

$$\begin{aligned} \Delta \mathbf{E}_{k+1} &= \tfrac{1}{2}\mathbf{k}_2^T \mathbf{P} \left(2\mathbf{y}_k + \mathbf{k}_2 \right) \\ &= \tfrac{1}{2}\mathbf{k}_2^T \mathbf{P} \left(2 \left(\tfrac{1}{\Delta t} \left[\mathbf{J}^{-1} - \Delta t \gamma \mathbf{I} \right] \mathbf{k}_2 - \tfrac{1}{2}\mathbf{k}_1 + \gamma \mathbf{k}_1 \right) + \mathbf{k}_2 \right) \\ &= \tfrac{1}{2}\mathbf{k}_2^T \mathbf{P} \left(-2\gamma \mathbf{k}_2 - \mathbf{k}_1 + 2\gamma \mathbf{k}_1 + \mathbf{k}_2 \right) \end{aligned} \tag{3.42}$$

In order to eliminate the vector $\mathbf{k}_1$ in Eq. (3.42), it is necessary to express $\mathbf{k}_1$ with respect to $\mathbf{k}_2$. To achieve this, we firstly substitute $\mathbf{y}_k$ in Eq. (3.40) by the RHS of the Eq. (3.37)

$$\begin{aligned} \mathbf{k}_2 &= [\mathbf{I} - \Delta t \mathbf{J}\gamma]^{-1} \left(\mathbf{J} \left(\tfrac{1}{\Delta t} \left[\mathbf{J}^{-1} - \Delta t \gamma \mathbf{I} \right] \mathbf{k}_1 + \tfrac{1}{2}\mathbf{k}_1 \right) \Delta t - \Delta t \mathbf{J} \gamma \mathbf{k}_1 \right) \\ &= [\mathbf{I} - \Delta t \mathbf{J}\gamma]^{-1} \mathbf{J} \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right) \Delta t \mathbf{k}_1 \end{aligned} \tag{3.43}$$

Through transformation, we can obtain the following formula:

$$\mathbf{k}_1 = \frac{1}{\Delta t} \left(\frac{1}{\Delta t}\mathbf{J}^{-1} + \left(\frac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} [\mathbf{1} - \Delta t \mathbf{J}\gamma] \mathbf{k}_2 \tag{3.44}$$

Substituting k_1 in Eq. (3.42) by (3.44) yields:

$$\begin{aligned} &\Delta \mathbf{E}_{k+1} \\ &= \tfrac{1}{2}\mathbf{k}_2^T \mathbf{P} \left((2\gamma - 1) \tfrac{1}{\Delta t} \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} [\mathbf{I} - \Delta t \mathbf{J}\gamma] \mathbf{k}_2 + (1 - 2\gamma) \mathbf{k}_2 \right) \\ &= \tfrac{1}{2} (2\gamma - 1) \mathbf{k}_2^T \mathbf{P} \left(\tfrac{1}{\Delta t} \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} [\mathbf{I} - \Delta t \mathbf{J}\gamma] - \mathbf{I} \right) \mathbf{k}_2 \end{aligned} \tag{3.45}$$

The term in the bracket at the right hand side of $\mathbf{P}$ matrix in Eq. (3.45) can be simplified:

$$\begin{aligned} &\tfrac{1}{\Delta t} \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} [\mathbf{I} - \Delta t \mathbf{J}\gamma] - \mathbf{I} \\ &= \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} \left(\tfrac{1}{\Delta t} [\mathbf{I} - \Delta t \mathbf{J}\gamma] - \mathbf{J} \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right) \right) \\ &= \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \mathbf{J}^{-1} \left(-\mathbf{J}\gamma - \mathbf{J} \left(\tfrac{1}{2} - 2\gamma \right) \right) \\ &= \left(\tfrac{1}{\Delta t}\mathbf{J}^{-1} + \left(\tfrac{1}{2} - 2\gamma \right) \mathbf{I} \right)^{-1} \left(\gamma - \tfrac{1}{2} \right) \end{aligned} \tag{3.46}$$

Therefore, we have:

$$\Delta \mathbf{E}_{k+1} = -\left(\gamma - \frac{1}{2} \right)^2 \mathbf{k}_2^T \mathbf{P} \left(\left(2\gamma - \frac{1}{2} \right) \mathbf{I} - \frac{1}{\Delta t}\mathbf{J}^{-1} \right)^{-1} \mathbf{k}_2 \tag{3.47}$$

Obviously, if $\gamma = 1/4$, we can obtain

$$\Delta \mathbf{E}_{k+1} = \Delta t \left(\gamma - \tfrac{1}{2}\right)^2 \mathbf{k}_2^T \mathbf{P}\mathbf{J}\mathbf{k}_2 = 0 \tag{3.48}$$

If $\gamma = 1/2$, we can easily find:

$$\Delta \mathbf{E}_{k+1} = 0 \tag{3.49}$$

Accordingly, we can conclude that the LSRT2 algorithm with $\gamma = 1/4$ or $\gamma = 1/2$ preserves the total energy when applied to a conservative system (i.e., $\mathbf{C} = 0$ and $\mathbf{F} = 0$).

For other values of γ, the stability condition is that the following matrix must be positive definite

$$\mathbf{Q} = \mathbf{P}\left(\left(2\gamma - \frac{1}{2}\right)\mathbf{I} - \frac{1}{\Delta t}\mathbf{J}^{-1}\right)^{-1} \tag{3.50}$$

Let's analyse the matrix $\mathbf{Q}^{-1}$:

$$\begin{aligned}\mathbf{Q}^{-1} &= \left(\left(2\gamma - \tfrac{1}{2}\right)\mathbf{I} - \tfrac{1}{\Delta t}\mathbf{J}^{-1}\right)\mathbf{P}^{-1} = \tfrac{1}{\Delta t}(\mathbf{P}\mathbf{J})^{-1} + \left(2\gamma - \tfrac{1}{2}\right)\mathbf{P}^{-1} \\ &= \begin{bmatrix} \left(2\gamma - \tfrac{1}{2}\right)\mathbf{K}^{-1} & \tfrac{1}{\Delta t}\mathbf{K}^{-1} \\ -\tfrac{1}{\Delta t}\mathbf{K}^{-1} & \left(2\gamma - \tfrac{1}{2}\right)\mathbf{M}^{-1} \end{bmatrix}\end{aligned} \tag{3.51}$$

To analyse the matrix $\mathbf{Q}^{-1}$, we can rewrite it as:

$$\mathbf{Q}^{-1} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ -\mathbf{B} & \mathbf{C} \end{bmatrix} \tag{3.52}$$

where

$$\mathbf{A} = \left(2\gamma - \frac{1}{2}\right)\mathbf{K}^{-1}, \mathbf{B} = \frac{1}{\Delta t}\mathbf{K}^{-1}, \mathbf{C} = \left(2\gamma - \frac{1}{2}\right)\mathbf{M}^{-1} \tag{3.53}$$

If $\gamma > 1/4$ and $\gamma \neq 1/2$, **A**, **B** and **C** are symmetric and positive definite. Using the Blockwise inversion technique, the matrix **Q** can be expressed as:

$$\mathbf{Q} = \begin{bmatrix} \left(\mathbf{A} + \mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1} & -\left(\mathbf{A} + \mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}\mathbf{C}^{-1} \\ \mathbf{C}^{-1}\mathbf{B}\left(\mathbf{A} + \mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1} & \left(\mathbf{C} + \mathbf{B}\mathbf{A}^{-1}\mathbf{B}\right)^{-1} \end{bmatrix} \tag{3.54}$$

If we suppose $\mathbf{k}_2^T = \{\mathbf{x}_1^T, \mathbf{x}_2^T\}$, we can obtain.

$$
\begin{aligned}
&\mathbf{k}_2^T\mathbf{Q}\mathbf{k}_2 \\
&= \left\{ \begin{array}{cc} \mathbf{x}_1^T & \mathbf{x}_2^T \end{array} \right\} \left[\begin{array}{cc} \left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1} & -\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}\mathbf{C}^{-1} \\ \mathbf{C}^{-1}\mathbf{B}\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1} & \left(\mathbf{C}+\mathbf{B}\mathbf{A}^{-1}\mathbf{B}\right)^{-1} \end{array} \right] \left\{ \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \end{array} \right\} \\
&= \mathbf{x}_1^T\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_1 + \mathbf{x}_2^T\left(\mathbf{C}+\mathbf{B}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_2 \\
&+\mathbf{x}_2^T\mathbf{C}^{-1}\mathbf{B}\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_1 - \mathbf{x}_1^T\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}\mathbf{C}^{-1}\mathbf{x}_2
\end{aligned}
\tag{3.55}
$$

Since the matrices ($\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$) are symmetric and positive definite, $\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)$ and $\left(\mathbf{C}+\mathbf{B}\mathbf{A}^{-1}\mathbf{B}\right)$ are symmetric and positive definite. Meanwhile, their inverse matrices are positive definite. Hence, we can obtain:

$$\mathbf{x}_1^T\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_1 + \mathbf{x}_2^T\left(\mathbf{C}+\mathbf{B}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_2 > 0 \tag{3.56}$$

and

$$\mathbf{x}_2^T\mathbf{C}^{-1}\mathbf{B}\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{x}_1 - \mathbf{x}_1^T\left(\mathbf{A}+\mathbf{B}\mathbf{C}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}\mathbf{C}^{-1}\mathbf{x}_2 = 0 \tag{3.57}$$

Thus, we can get

$$\mathbf{k}_2^T\mathbf{Q}\mathbf{k}_2 > 0 \tag{3.58}$$

which implies that the matrix $\mathbf{Q}$ is positive definite. Then we can obtain

$$\Delta\mathbf{E}_{k+1} = -\left(\gamma - \frac{1}{2}\right)^2 \mathbf{k}_2^T\mathbf{Q}\mathbf{k}_2 < 0 \tag{3.59}$$

Finally, we can conclude that if $\gamma > 1/4$ and $\gamma \neq 1/2$, the RHS of the energy balance equation is negative, implying that the algorithm is unconditionally stable and introduces so called algorithmic damping. Conversely, the algorithm with $\gamma < 1/4$ is unconditionally unstable and provides negative dissipation.

3.5 Numerical experiments on a spring-pendulum oscillator

It is well known that systems including pendula may exhibit large amplitude subharmonic and chaotic motions as well as simple periodic behaviour (Nayfeh and Arafat,

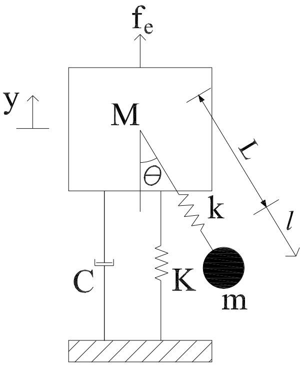


Fig. 3.2: Schematic representation of a spring-pendulum system

2001; Bayly and Virgin, 1993). Both pendula and spring-pendula have been studied theoretically, numerically and experimentally in the context of time integration and dynamic substructuring (Bursi and Mancuso, 2002; Kyrychko1 et al., 2006). In this section, a spring-pendulum system is introduced, as a nonlinear test problem, to validate the performance of the LSRT algorithms in nonlinear cases. The spring-pendulum system has aspects of both a mass-spring oscillators and a simple pendulum. Thus, it can, on the one hand, be taken as a nonlinear test system; on the other hand, it can be used to model a stiff problem by introducing different springs with distinct stiffness. In detail, the equations of motion of the spring-pendulum system are modeled by using the Lagrange approach. Subsequently, its stability is studied using the Routh-Hurwitz Criterion. Lastly, the LSRT algorithms are used to simulate the spring-pendulum system.

3.5.1 Governing equations

The complete system is shown in Fig. 3.2. It consists of a spring-pendulum with its pivot point connected to a mass *M*, belonging to a mass-spring-damper system characterized by stiffness *K* and damping *C*. The mass *M* is guided by a line and excited by an external force f_e. The pendulum mass, *m*, is assumed to act at a single point and is connected to the pivot point by a spring, *k*. The stretched length is given by *L* + *l* where *L* is the original length of the spring together with the static deflection caused by gravity of the pendulum mass. It is evident that the system includes rigid masses, springs, damper and external force. Therefore, the general form of the Lagrange equations (Leonard, 1986), also considering dissipation through the Rayleigh dissipation function and external forces using the idea of generalized forces, is required

$$\frac{\partial L}{\partial q_i} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) + \frac{\partial D}{\partial \dot{q}_i} = Q_i \tag{3.60}$$

The Lagrangian L is the difference between the kinetic energy of a system and its potential energy, i.e., $L = T - U$. For this spring-pendulum system, the kinetic energy is expressed as:

$$U = \frac{1}{2}Ky^2 + \frac{1}{2}kl^2 + mg\left(y - \frac{M+m}{k}g + L - (L+l)\cos\theta\right) + Mg\left(y - \frac{M+m}{k}g\right) \tag{3.61}$$

and the potential energy of the system is

$$T = \frac{1}{2}M\dot{y}^2 + \frac{1}{2}m\left(\dot{y}^2 + 2\dot{y}\left((L+l)\,\dot{\theta}\sin\theta - \dot{l}\cos\theta\right) + \dot{l}^2 + (L+l)^2\,\dot{\theta}^2\right) \tag{3.62}$$

Where y is the displacement of M, and l and θ is describe the spring extensional and pendulum angular motions, respectively. The superscript dots represent their time derivatives.

The Lagrangian is therefore

$$\begin{aligned} L &= T - U \\ &= \tfrac{1}{2}M\dot{y}^2 + \tfrac{1}{2}m\left(\dot{y}^2 + 2\dot{y}\left((L+l)\,\dot{\theta}\sin\theta - \dot{l}\cos\theta\right) + \dot{l}^2 + (L+l)^2\,\dot{\theta}^2\right) - \\ &-\tfrac{1}{2}Ky^2 + \tfrac{1}{2}kl^2 + mg\left(y - \tfrac{M+m}{k}g + L - (L+l)\cos\theta\right) + Mg\left(y - \tfrac{M+m}{k}g\right) \end{aligned} \tag{3.63}$$

For the viscous damping, we can introduce the Rayleigh dissipation function (Lemos, 1991) in which its contribution is analogous in form to the potential energy of a spring, i.e.,

$$V = \frac{1}{2}C\dot{y}^2 \tag{3.64}$$

The external force enters the Lagrange equations in the right hand side, in the form of generalized forces

$$Q_y = f_e \tag{3.65}$$

which is the external force along the direction of y.

Inserting all the terms of Eq. (3.60) into the following

$$\begin{cases} \frac{\partial L}{\partial y} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}}\right) + \frac{\partial V}{\partial \dot{y}} = Q_y \\ \frac{\partial L}{\partial \theta} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) + \frac{\partial V}{\partial \dot{\theta}} = 0 \\ \frac{\partial L}{\partial l} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{l}}\right) + \frac{\partial V}{\partial \dot{l}} = 0 \end{cases} \tag{3.66}$$

yields the system equations of motion

$$\begin{aligned} &(M+m)\ddot{y} + C\dot{y} + Ky + m(L+l)\left[\ddot{\theta}\sin\theta + \dot{\theta}^2\cos\theta\right] + 2m\dot{l}\dot{\theta}\sin\theta - m\ddot{l}\cos\theta = f_e \\ &m(L+l)^2\ddot{\theta} + m(L+l)(g+\ddot{y})\sin\theta + 2m(L+l)\dot{l}\dot{\theta} = 0 \\ &m\ddot{l} + kl - m\ddot{y}\cos\theta - m(L+l)\dot{\theta}^2 + mg(1-\cos\theta) = 0 \end{aligned} \tag{3.67}$$

In particular, the coordinate y is directly excited by f_e in Eq. (3.67), whereas θ and l are not. Based on the circular frequency ω of f_e an external resonance may occur when ω/ω_i is a rational number, where ω_i is a natural frequency of the system. Conversely, non-resonant interactions transfer energy from a high-frequency axial mode to a low-frequency circular mode, even though there is no special relationship between their frequencies.

3.5.2 Static analysis

The computation of static equilibria is often the first step in the analysis of a nonlinear system model so that the behavior of the system in the neighborhood of each equilibrium point can be qualitatively determined.

Before seeking for the equilibrium points, we rewrite the Lagrange equations of (3.67) in the space-state form, through the transformations

$$\mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{Bmatrix} = \begin{Bmatrix} y \\ l \\ \theta \\ \dot{y} \\ \dot{l} \\ \dot{\theta} \end{Bmatrix} \tag{3.68}$$

We will obtain the Hamilton form of the governing equations of motion

$$
\begin{aligned}
&\dot{x}_1 = x_4 \\
&\dot{x}_2 = x_5 \\
&\dot{x}_3 = x_6 \\
&(M+m)\dot{x}_4 + Cx_4 + Kx_1 + m(L+x_2)\left[\dot{x}_6 \sin x_3 + x_6^2 \cos x_3\right] + \\
&\quad + 2mx_5x_6 \sin x_3 - m\dot{x}_5 \cos x_3 = Fe \\
&m(L+x_2)\dot{x}_6 + m(g+\dot{x}_4)\sin x_3 + 2mx_5x_6 = 0 \\
&m\dot{x}_5 + kx_2 - m\dot{x}_4 \cos x_3 - m(L+x_2)x_6^2 + mg(1-\cos x_3) = 0
\end{aligned}
\tag{3.69}
$$

Equilibrium positions are determined by the equilibrium conditions $\dot{\mathbf{x}} = 0$. Hence, they must satisfy the algebraic equations

$$
\begin{aligned}
&Kx_1 = Fe \\
&mg \sin x_3 = 0 \\
&kx_2 + mg(1-\cos x_3) = 0
\end{aligned}
\tag{3.70}
$$

which have two solutions, namely, the lower equilibrium point

$$x = \left\{ \begin{matrix} Fe/k & 0 & 2j\pi & 0 & 0 & 0 \end{matrix} \right\},\ (j = 0,\ 1,\ 2,\ \cdots) \tag{3.71}$$

and the upper equilibrium point

$$x = \left\{ \begin{matrix} Fe/k & -2mg/k & (2j+1)\pi & 0 & 0 & 0 \end{matrix} \right\},\ (j = 0,\ 1,\ 2,\ \cdots) \tag{3.72}$$

Of particular interest in mechanics is the problem of stability of motion of dynamical system when they are perturbed from an equilibrium state. Moreover, the behaviour of a nonlinear system in neighborhood of an equilibrium point can be predicted on the basis of the linearized system about the equilibrium points (Leonard, 1986). With this in mind, we calculate the Jacobian matrices about the two equilibrium points, and introduce Routh-Hurwitz Criterion to judge their stability.

For the lower equilibrium point, inserting its components into the equation $\mathbf{J} = \partial\mathbf{f}/\partial\mathbf{y}$ results the Jacobian matrix

$$\mathbf{J} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -K/M & -k/M & 0 & -C/M & 0 & 0 \\ -K/M & -k/M - k/m & 0 & -C/M & 0 & 0 \\ 0 & 0 & -\mathrm{g}/L & 0 & 0 & 0 \end{bmatrix} \tag{3.73}$$

leading to the characteristic equation

$$\lambda^6+\frac{C}{M}\lambda^5+\left(\frac{K+k}{M}+\frac{k}{m}+\frac{g}{L}\right)\lambda^4+\left(\frac{Cg}{LM}+\frac{Ck}{Mm}\right)\lambda^3+\left(\frac{gk}{Lm}+g\frac{k+K}{LM}+\frac{kK}{mM}\right)\lambda^2+\frac{Cgk}{LmM}\lambda+\frac{gkK}{LmM}=0 \tag{3.74}$$

For the upper equilibrium point, the Jacobian matrix is

$$J=\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -K/M & k/M & 0 & -C/M & 0 & 0 \\ K/M & -k/M-k/m & 0 & C/M & 0 & 0 \\ 0 & 0 & g/\left(L-2mg/k\right) & 0 & 0 & 0 \end{bmatrix} \tag{3.75}$$

whose characteristic equation is

$$\begin{aligned}&\lambda^6+\tfrac{C}{M}\lambda^5+\left(\mathrm{k}\left(\tfrac{1}{\mathrm{m}}+\tfrac{\mathrm{g}}{-\mathrm{kL}+2\mathrm{gm}}+\tfrac{1}{\mathrm{M}}\right)+\tfrac{\mathrm{K}}{\mathrm{M}}\right)\lambda^4+\tfrac{(CkL-3Cgm)k}{(kL-2gm)mM}\lambda^3+\\&+\tfrac{k(-3gKm+k(KL-g(m+M)))}{m(kL-2gm)M}\lambda^2-\tfrac{Cgk^2}{(kL-2gm)mM}\lambda-\tfrac{gk^2K}{(kL-2gm)mM}=0\end{aligned} \tag{3.76}$$

Next, Let us form the Routh array

$$\begin{array}{c|cccc} \lambda^6 & a_0 & a_2 & a_4 & a_6 \\ \lambda^5 & a_1 & a_3 & a_5 & 0 \\ \lambda^4 & c_1 & c_2 & c_3 & 0 \\ \lambda^3 & d_1 & d_2 & 0 & 0 \\ \lambda^2 & e_1 & e_2 & 0 & 0 \\ \lambda^1 & f_1 & 0 & 0 & 0 \\ 1 & g_1 & 0 & 0 & 0 \end{array} \tag{3.77}$$

In the case of the lower equilibrium point, the coefficients of the characteristic polynomial are given as follows

$$\begin{array}{llll} a_0=1 & a_2=\frac{Cg}{LM}+\frac{Ck}{Mm} & a_4=\frac{gk}{Lm}+g\frac{k+K}{LM}+\frac{kK}{mM} & a_6=\frac{gkK}{LmM} \\ a_1=\frac{K+k}{M}+\frac{k}{m}+\frac{g}{L} & a_3=\frac{Cg}{LM}+\frac{Ck}{Mm} & a_5=\frac{Cgk}{LmM} & \end{array} \tag{3.78}$$

According to Routh-Hurwitz Criterion, we can compute all other entries as

$$\begin{aligned} c_1 &= \tfrac{k+K}{M} & c_2 &= \tfrac{kKL+gkm+gKm}{LmM} & c_3 &= \tfrac{gkK}{LmM} \\ d_1 &= \tfrac{Ck^2}{(k+K)mM} & d_2 &= \tfrac{Cgk^2}{(k+K)LmM} \\ e_1 &= \tfrac{kK}{mM} & e_2 &= \tfrac{gkK}{LmM} \\ f_1 &= 0 \end{aligned} \tag{3.79}$$

The necessary condition that all roots have negative real parts is that all the elements of the first column of the array have the same sign (nonzero). However, all the elements of Row f are zero and all the elements above Row f in the first column are positive. In this case, some of the roots of the polynomial are located symmetrically about the origin of the s-plane, e.g., a pair of purely imaginary roots. To determine this special case, we take the last non-zero row and form an auxiliary equation:

$$\frac{kK}{mM}\lambda^2 + \frac{gkK}{LmM}\lambda^0 = 0 \tag{3.80}$$

Now carrying out its differentiation with respect to λ gives a new characteristic equation

$$\frac{d}{d\lambda}\left(\frac{kK}{mM}\lambda^2 + \frac{gkK}{LmM}\lambda^0\right) = 2\frac{kK}{mM}\lambda = 0 \tag{3.81}$$

Then, we continue the stability analysis by replacing with

$$f_1 = 2\frac{kK}{mM} \tag{3.82}$$

Hence, the last entry can be obtained

$$e_2 = \frac{gkK}{LmM} \tag{3.83}$$

Clearly, the terms in the first column are all positive. Therefore, the motion in the neighborhood of the lower equilibrium point is stable. In the same manner, directly using Routh-Hurwitz Criterion, the case of the upper equilibrium point is proved to be unstable.

3.5.3 Numerical integration

After qualitatively studying the spring-pendulum system, we apply the LSRT algorithms, as well as two Newmark-based integrators, to this system to investigate their nonlinear performance. These two Newmark-based integrators are explicit when dealing with systems with only nonlinear stiffness. To implement them, an additional prediction for target velocity is provided and a linearization step is required for the Newmark-Chang method. In detail, a process, including an additional prediction for velocity and a linearization for the nonlinear system, is carried out before integrating. Then, the favourable properties of the LSRT algorithms are reconfirmed by comparing with the utilized Newmark-based integrators in terms of accuracy, stability and numerical dissipation.

3.5.3.1 Implementation

The state space form of the equations of motion can be directly integrated by the first-order LSRT algorithms after the Jacobian matrix can be calculated at the beginning of each time step. For the sake of comparison, two Newmark based integrators, i.e., the explicit Newmark method and the Changs method, are also implemented. These two Newmark-based integrators are explicit when dealing with systems with only nonlinear stiffness. To apply them to the spring-pendulum system, an additional prediction for target velocity is provided

$$\dot{\mathbf{x}}_{k+1} = \dot{\mathbf{x}}_k + \Delta t \ddot{\mathbf{x}}_k \tag{3.84}$$

To implement the Newmark-Chang method, a linearization of (3.1) is required to update the parameters β_1 and β_2. By means of the first-order Taylor's series expansion, the second term on the right-hand side of (3.1) can be approximated as

$$\mathbf{f}\left(\mathbf{x}, \dot{\mathbf{x}}\right) = \mathbf{f}(\mathbf{x}_0, \dot{\mathbf{x}}_0) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}|_{\mathbf{x}_0, \dot{\mathbf{x}}_0}(\mathbf{x} - \mathbf{x}_0) + \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{x}}}|_{\mathbf{x}_0, \dot{\mathbf{x}}_0}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_0) + \mathrm{O}\left((\mathbf{x} - \mathbf{x}_0)^2, (\dot{\mathbf{x}} - \dot{\mathbf{x}}_0)^2\right) \tag{3.85}$$

As a result, Eq.(3.1) becomes

$$\bar{\mathbf{M}}\ddot{\mathbf{x}} + \bar{\mathbf{C}}\dot{\mathbf{x}} + \bar{\mathbf{K}}\mathbf{x} = \bar{\mathbf{F}} \tag{3.86}$$

where

$$\begin{aligned}
\bar{\mathbf{M}} &= \mathbf{M} \\
\bar{\mathbf{C}} &= \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{x}}}\Big|_{\mathbf{x}_0, \dot{\mathbf{x}}_0} \\
\bar{\mathbf{K}} &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_{\mathbf{x}_0, \dot{\mathbf{x}}_0} \\
\bar{\mathbf{F}} &= \mathbf{g}(t) + \bar{\mathbf{C}}\dot{\mathbf{x}}_0 + \bar{\mathbf{K}}\mathbf{x}_0 - \mathbf{f}(\mathbf{x}_0, \dot{\mathbf{x}}_0)
\end{aligned} \tag{3.87}$$

Now, we recall the Newmark-Chang linearly implicit scheme. In this case, the target displacement is calculated as follows:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \beta_1 \Delta t \dot{\mathbf{x}}_k + \beta_2 \Delta t^2 \ddot{\mathbf{x}}_k \tag{3.88}$$

where two parameters β_1 and β_2, are given by

$$\begin{aligned}
\beta_1 &= \left[\mathbf{I} + \tfrac{1}{2}\Delta t \bar{\mathbf{M}}^{-1}\bar{\mathbf{C}} + \tfrac{1}{4}\Delta t^2 \bar{\mathbf{M}}^{-1}\bar{\mathbf{K}}\right]^{-1} \left[\mathbf{I} + \tfrac{1}{2}\Delta t \bar{\mathbf{M}}^{-1}\bar{\mathbf{C}}\right] \\
\beta_2 &= \tfrac{1}{2}\left[\mathbf{I} + \tfrac{1}{2}\Delta t \bar{\mathbf{M}}^{-1}\bar{\mathbf{C}} + \tfrac{1}{4}\Delta t^2 \bar{\mathbf{M}}^{-1}\bar{\mathbf{K}}\right]^{-1}
\end{aligned} \tag{3.89}$$

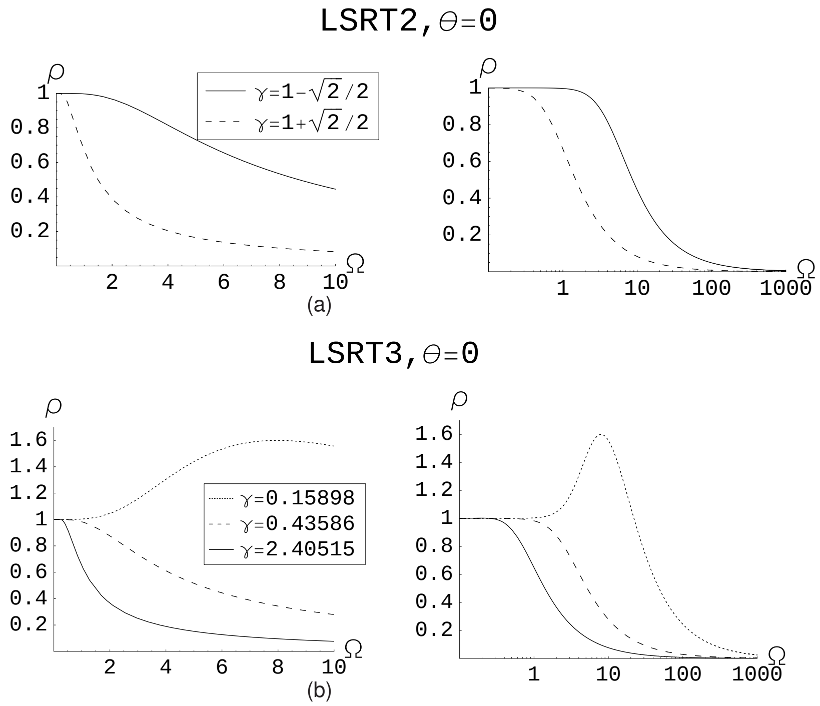
such that the method becomes linearly implicit even for the cases with nonlinear damping terms.

3.5.3.2 Absolute stability analysis

In order to analyse the convergence properties of the LSRT methods, the spring-pendulum system described by (3.67) is linearized around $\theta = 0$ under the assumption that the numerical solutions are contractive (Lambert, 1991, p.265). The spectral radius ρ of the amplification matrix of the nonstiff spring-pendulum system whose characteristics are summarized in Table 3.1 is represented in Fig. 3.3 with linear scale (left subfigures) and logarithmic scale (right subfigures), for the LSRT2 and the LSRT3 method, respectively, and for different γ values. The stability properties of the LSRT methods are similar to those of the linear uncoupled systems. The LSRT3 method with $\gamma = 0.1590$ is unstable, while the LSRT2 method with $\gamma = 1 + \sqrt{2}/2$ and the LSRT3 method with $\gamma = 2.4052$ exhibit an increased numerical damping. The high-frequency filtering capabilities of the algorithms are evident in the log-scale representation.

Table 3.1: Spring-pendula characteristics for numerical simulations

Uncoupled spring-pendulum: nonstiff problem					
M [kg]	*C* [kg/s]	*K* [N/m]	*m* [kg]	*k* [N/m]	*L* [m]
11	40	1000	0.34	400	0.167
Uncoupled spring-pendulum: stiff problem					
M [kg]	*C* [kg/s]	*K* [N/m]	*m* [kg]	*k* [N/m]	*L* [m]
11	40	1000	0.34	400000	0.167



(a)

(b)

Fig. 3.3: Spectral radii of the LSRT methods applied to the uncoupled spring-pendulum oscillator linearized around $\theta = 0$: a) the LSRT2 method; b) the LSRT3 method.

3.5.3.3 Accuracy analysis

The accuracy analysis of the uncoupled spring-pendulum nonstiff problem, whose characteristics are listed in Table 3.1, is performed by means of numerical simulations. The convergence both of displacement and velocity is depicted in Fig. 3.4a and 3.4b, respectively. As expected, all examined time-stepping methods exhibit second-order accuracy except the LSRT3 method that shows third-order accuracy. The choice of different values of γ for displacement and velocity accuracy provides similar results for the methods with the same stage number as depicted in Fig. 3.4c and Fig. 3.4d, respectively.

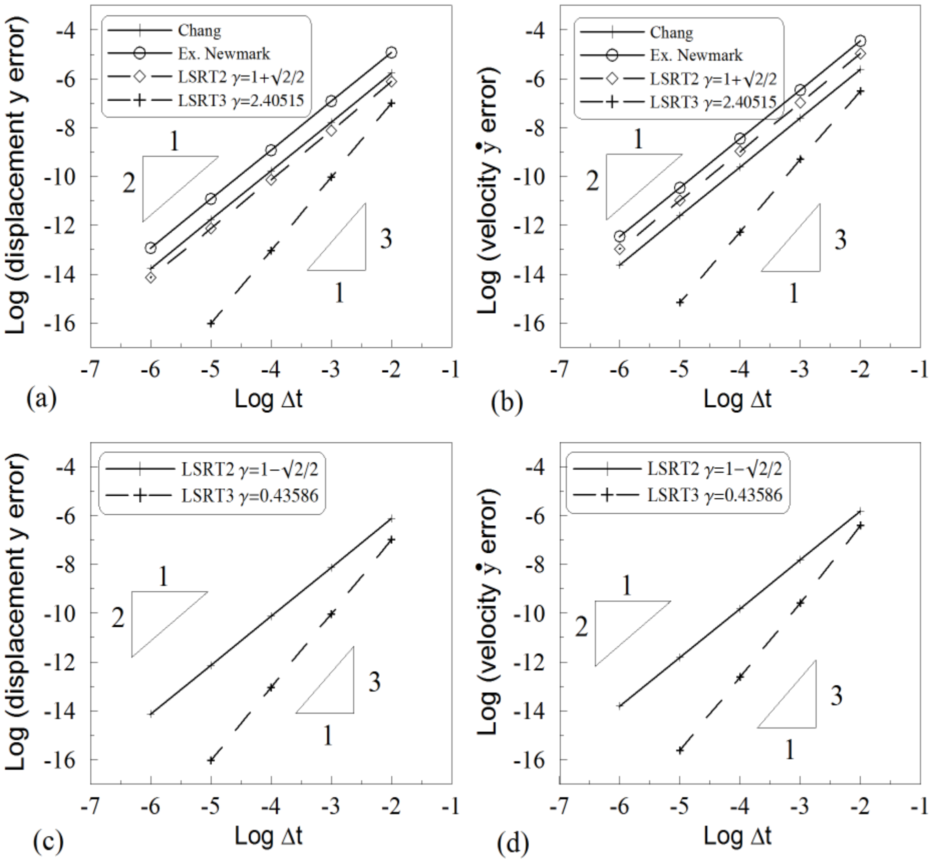


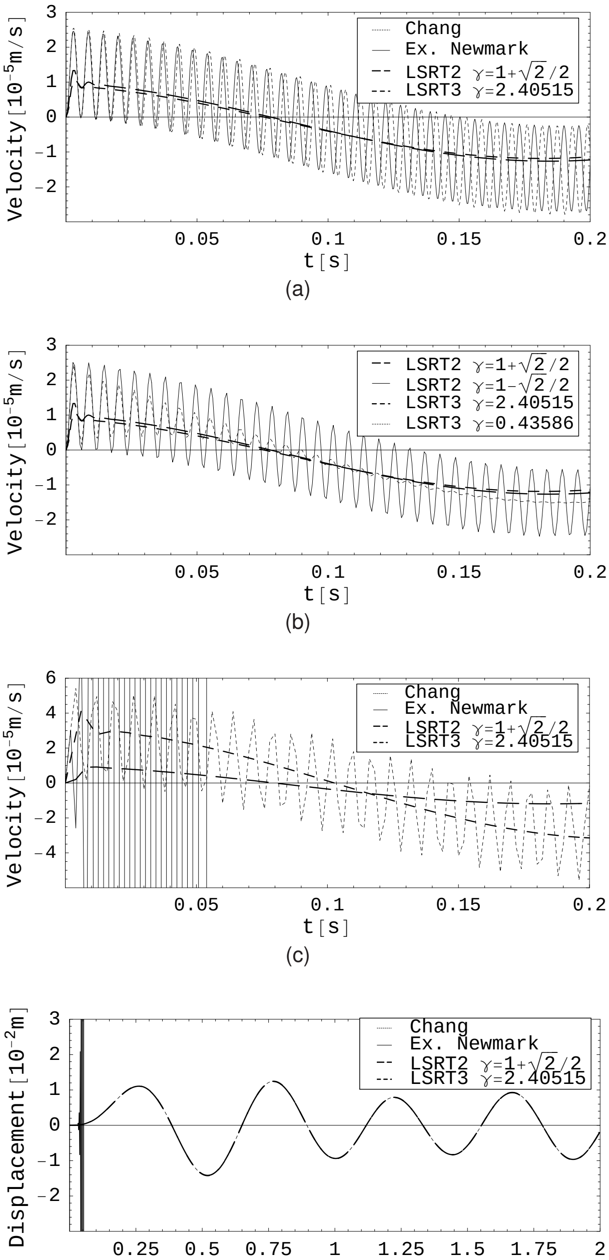
Fig. 3.4: Convergence of displacement and velocity for the uncoupled spring-pendulum nonstiff problem summarized in Table 3.1: (a) displacement y error; (b) velocity $\dot{y}$ error; (c) displacement y error for the LSRT algorithms with different γs; (d) velocity $\dot{y}$ error for the LSRT algorithms with different γs.

3.5.3.4 Numerical simulation

Some numerical simulations on the uncoupled stiff spring-pendulum system, whose characteristics are summarized in Table 3.1, are depicted in Fig. 3.5. In detail, both Fig. 3.5a and Fig. 3.5b show the axial velocity $\dot{l}$ of the spring-pendulum system integrated with $\Delta t = \Delta t_0 = 1/3$ms both for the Newmark-Chang and the Explicit-Newmark's method. Conversely for comparison, $2\Delta t_0$ is used for the LSRT2 method and $3\Delta t_0$ for the LSRT3 method, respectively. The filtering capabilities of the LSRT methods are evident. When the time step is chosen to be $\Delta t_0 = 2$ms only the LSRT methods exhibit a stable behaviour, as it is shown in Fig. 3.5c and 3.5d, respectively.

3.6 Conclusions

The LSRT algorithms were introduced and analysed in this chapter. Firstly, Considering L-stability, real-time compatibility and desirable accuracy, the LSRT algorithms derived form the linearly implicit Rosenbrock methods were introduced. Subsequently, their accuracy was restudied in a purely symbolic way, and their stability was investigated by means of the energy approach, indicating how the parameter affects the stability and dissipation property of the LSRT algorithm. Finally, the nonlinear performance of the LSRT algorithms was observed through numerical experiments on the spring-pendulum system. Compared with two Newmark-based methods, the LSRT algorithms exhibit relative higher efficiencies both on high-frequency dissipation and nonlinear implementation.



(a)

(b)

(c)

(d)

Fig. 3.5: Numerical simulations for the uncoupled spring-pendulum stiff problem summarized in Table 3.1: (a) velocity $\dot{l}$ obtained with different methods and $\Delta t_0 = 1/3$ms; (b) velocity $\dot{l}$ provided by different LSRT algorithms and $\Delta t_0 = 1/3$ms; (c) velocity $\dot{l}$ with $\Delta t_0 = 2$ms; (d) displacement y with $\Delta t_0 = 2$ms.

CHAPTER 4

MONOLITHIC TIME INTEGRATION METHODS FOR REAL-TIME SUBSTRUCTURE TESTS

4.1 Introduction

Real-time Testing based on the principle of Dynamic Substructuring and Pseudo-Dynamic testing with dynamic substructuring also known as *hybrid testing*, are becoming more and more important in testing and proof of design and control strategies for civil engineering, see for instance (Saouma and Sivaselvan, 2008) and (Bursi, 2008). In the related field of coupled mechanical systems Hardware-in-the-loop (HiL) tests are carried out, where physically and computationally heterogeneous mechanical components interact dynamically, and the response is obtained by solving simultaneously the coupled equations which model the system. Because terminology is far from standard, *heterogeneous testing* is preferred herein to *hybrid testing*, because the tested system is composed of parts of different kinds, see (Felippa et al., 2001; Gumaste et al., 2000) and references therein. In fact, hybrid is more linked to multiplatforms or multimethods in software processes Donzelli and Iazeolla (2001).

RTDS and PsD techniques with DS involve splitting of the emulated structure into two parts: the Physical Substructure (PS) or device, that contains a key region of interest and is experimentally tested; the virtual or Numerical Substructure (NS), that contains the remainder of the system and is numerically modelled as illustrated in Fig.4.1a-d. By imposing compatibility and equilibrium conditions at the interface between the NS and the PS, respectively, the substructures are made to interact possibly in *real time*, in order to emulate the dynamic behaviour of the full *emulated* system. In the special case of *real-time* testing, the computational time per time interval, *i.e.* the integration time Δt plus the time for display must be smaller or equal than the physical time taken by the actual motion of the emulated system. In order to interconnect the PS and the NS, a *transfer system* that acts on the PS is typically controlled to follow NS interface kinematic quantities or other outputs. This transfer system normally consists of an electric or electro-hydraulic actuator, or system of actuators such as a shake table, and a controller, shown in block diagram form in Fig 4.1e. The overall real-time system must be capable of interfacing the NS with the test system actuator and sensor in a seamless fashion.

Once the governing differential equations describing the NS and the PS involved are *partitioned*, for instance by means of differential partitioning (Bursi et al., 2008), different integrators can be exploited to perform the discretization in time, especially in the case of real-time applications. See, among others, papers based on linear multistep methods that are applied to the Euler-Lagrange form of equations of motions and are typically linearly implicit (Chen et al., 2009), explicit like the explicit Newmark method (Bonnet et al., 2007, 2008) or applied with sub-increments and corrections (Jung et al., 2007). Moreover, if they are A-stable they cannot exceed the accuracy order of two - Dahlquist's barrier (Lambert, 1991, p.243). Conversely, there are papers where integrators are applied to the Hamilton form of equations of motions (first-order in time) and are typically explicit (Zhang et al., 2005) or based on Runge-Kutta methods, and conceived to be linearly implicit (Bursi et al., 2008). From Butcher's theory we know that a *p*-stage explicit Runge-Kutta method cannot have accuracy order greater than *p*, but these methods reach easily accuracy of three (Bursi et al., 2008) or more (Day, 1985). Due to the nature of heterogeneous simulations, integrators are also linked with specific techniques, like the one that achieves an improved stability by means of a better estimation of the Jacobian in a coupled system (Lamarche et al., 2009) or by the approximation of the tangent operator directly from measurements (Bonelli and Bursi, 2004; Hung and El-Tawil, 2009). Moreover, problems derived from the dynamics of the transfer system have elegantly been solved with outer loop strategies (Bonnet et al., 2007) and/or delay compensation techniques (Ahmadizadeh et al., 2008). Simulations of complex substructures may also require the need of *multitasking* (Bonnet et al., 2007), or the use of partitioned integrators with *subcycling* (Bonnet et al., 2008). Moreover, for real-time applications other topics, *i.e.* operating systems, control and equipment are investigated in depth. They are beyond the scope of this paper, but relevant information can be found in the articles(Saouma and Sivaselvan, 2008; Bursi et al., 2008).

In this chapter we focus on monolithic time integrators, i.e. single time-stepping schemes which are directly applied to an emulated system. Currently limited to linear systems (Bursi et al., 2008), we extend the application of Rosenbrock-based real-time algorithms to nonlinear and possibly stiff coupled systems, for which they were originally developed (Hairer and Wanner, 1996, p.102). Two real-time L-stable integration methods with two (LSRT2) and three (LSRT3) stages based on Rosenbrock schemes (Rosenbrock, 1963), already analysed for linear problems (Bursi et al., 2008), and we note the beneficial effects of these algorithms in terms of their filtering properties beyond the Nyquist's frequency. For the sake of comparison, we also implement the linearly implicit Newmark-Chang method (Chang, 2002) and the Explicit-Newmark's method that were extensively employed by Bonnet et al.Bonnet et al. (2007, 2008) for real-time testing. The remainder of the chapter is organized as follows. Section 4.2 presents the substructuring strategy and its application to a spring-pendulum system. A stability analysis of the discretized coupled spring-pendulum system based on zero-stability approach is presented in Section 4.3. In Section 4.4, some subcycling strategies for the LSRT algorithms as well as some other real-time integrators are developed. Moreover, in order to reconfirm the performance of the methods on nonlinear problems, we present in section 4.5 some RTDS tests on the coupled spring-pendulum system in which nonlinearities are permitted in both the NS and PS. Conclusions are drawn in Section 4.6.

4.2 Substructuring strategy

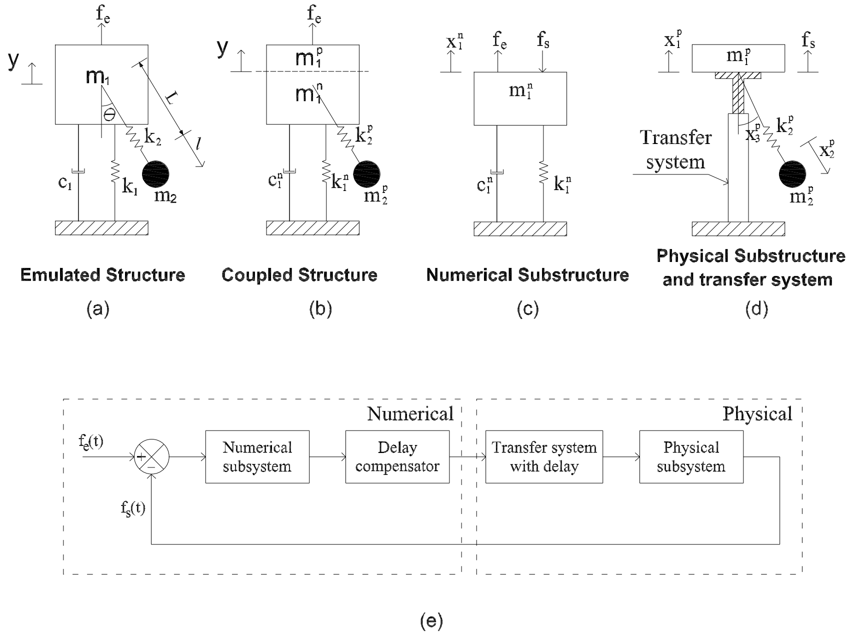
4.2.1 Partitioning and coupling

For the NS and PS to combine in order to emulate the entire structure, real-time coupling at the interconnection between the substructures is required. Coupling between the NS and the PS is achieved through interface compatibility and equilibrium conditions. With the equations of motion of the NS, the kinematic quantities can be fully computed from the previous displacement, velocity and acceleration vectors. Those controlled kinematic quantities relating to the physical/numerical substructure interface are then applied to the test specimen using a transfer system and the coupling force vector measured at the interface is feedback to the NS. This force feedback ensures the equilibrium condition, whilst the kinematic quantities applied to the NS at the interface results in the compatibility condition. In conclusion, the monolithic approach for real time tests consists in running two processes in parallel, and imposing interface compatibility and equilibrium conditions to ensure their coupling.

The partitioning technique exploited herein to partition a system is called differential partitioning (Bursi, 2008) and is based on the governing differential equations describing the NS and PS involved. In this chapter we consider a nonlinear spring-pendulum system shown in Fig. 4.1a. Through the so-called differential partitioning as shown in Fig. 4.1b, the overall structure is split into NS and PS depicted in Fig.4.1c and d, respectively. Meanwhile, the global structure is uncoupled using two sets of ordinary differential equations (as shown in Section 4.4.2) connected by a shared force vector, i.e. the coupling force at the interface, and by the interface continuities.

4.2.2 Coupled integration in real time based on displacement control

A prediction-substitution procedure based on kinematic control and without iteration to satisfy the real-time requirement is considered herein. The algorithm is schematically depicted in Fig. 4.2. It entails the integration of the differential equations of the NS using the coupling force $m_1^p \ddot{x}_{1,k-2}^p + (k_2^p x_{2,k-2}^p + m_2^p g) cos(x_{2,k-2}^p) + m_1^p g = u_{k-1}^n$



Emulated Structure (a)

Coupled Structure (b)

Numerical Substructure (c)

Physical Substructure and transfer system (d)

(e)

Fig. 4.1: (a)-(d) Schematic representation of a substructured spring-pendulum oscillator; (e) block diagram representation including delay

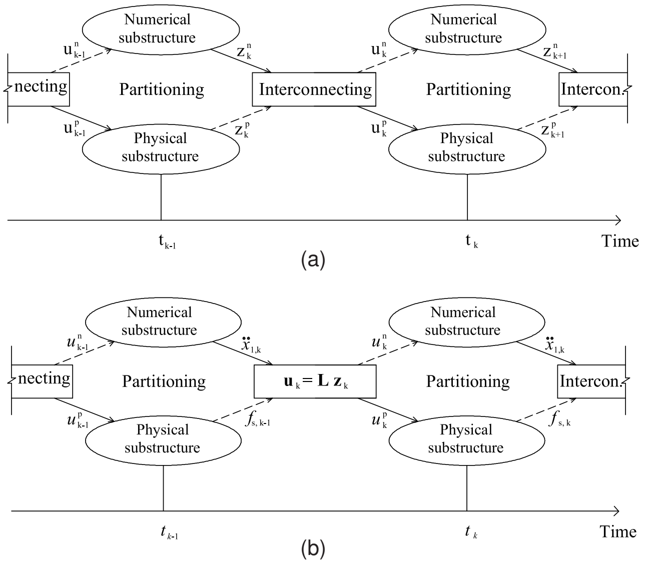


Fig. 4.2: Coupled integration in real time: (a) force-displacement(mixed) coupling strategy; and (b) mixed strategy with algebraic coupling conditions

available at t_{k-1}; the motion of the actuator governed by predicted kinematic quantity $\ddot{x}^p_{1,k} = u^p_k$ available at t_k; and the measuring of the coupling force $m^p_1\ddot{x}^p_{1,k} + (k^p_2 x^p_{2,k} + m^p_2 g)cos(x^p_{2,k}) + m^p_1 g$. This actual prediction-substitution procedure is rather different from a truly parallel scheme (Felippa et al., 2001), which at t_{k-1} would entail also a prediction of the coupling force $m^p_1\ddot{x}^p_{1,k} + (k^p_2 x^p_{2,k} + m^p_2 g)cos(x^p_{2,k}) + m^p_1 g$. This is not useful owing to the nature of the PS. Note that according to Fig. 4.2, the application of kinematic quantities and the relevant measurements entail a certain delay that in our treatment is considered in Section 4.4 by means of an approximation on the evaluation of the NS acceleration.

4.3 Subcycling strategies

Because the NS is likely to have several degrees of freedom, and may be nonlinear, it is desirable to have large Δts to integrate the entire structure in real time. On the other hand, an ideal control requires a fine time step to match the sampling rate of the digital controller (Bonnet et al., 2007). It is custom to deal with these requirements with the use of subcycling strategies. For instance, see (Weiner et al., 1993) and (Shome et al., 2004), among others, in the context of partitioned Runge-Kutta methods. Similarly, interfiled parallel algorithms without and with numerical dissipation were developed for linear multistep methods (Bonelli et al., 2008b; Bursi et al., 2009). Herein we still propose a monolithic approach, less advanced than a partitioned approach, but more effective than a basic multitasking strategy (Bonnet et al., 2007). In fact, a basic multitasking strategy adopts interpolation to solve the incompatibility between coarse and fine time steps; herein, we propose multistage algorithms -the LSRT methods- endowed with interpolations that achieve a certain order of accuracy in the NS. Similar interpolations are suggested for the Newmark's schemes.

4.3.1 Subcycling strategies for the LSRT algorithms

Along the lines of Meijaard (2003), the continuous solution of the LSRT algorithms can be described as

$$\mathbf{y}(\xi) = \mathbf{y}_k + \Delta t \sum_{i=1}^{s} b_i(\xi)\mathbf{k}_i \tag{4.1}$$

where $\xi = \dfrac{t - t_k}{\Delta t}$ with $0 \leq \xi \leq 1$.

In detail, for the LSRT2 algorithm, the weights $b_i(\xi)$ read:

$$b_1(\xi) = \frac{(2\alpha_{21} + 2\gamma + 2\gamma_{21} - \xi)\xi}{2(\alpha_{21} + \gamma_{21})}, b_2(\xi) = \frac{(-2\gamma + \xi)\xi}{2(\alpha_{21} + \gamma_{21})} \tag{4.2}$$

Conversely, for the LSRT3 algorithm the weights $b_i(\xi)$ can be evaluated as

$$\begin{aligned} b_1(\xi) &= \tfrac{(4+6\gamma+6\gamma_{31}+b_2(-2+6\gamma_{31}-6\gamma_{32})+6\gamma_{32}-3\xi)\xi}{4+6\gamma_{31}+6\gamma_{32}} \\ b_2(\xi) &= b_2\xi \\ b_3(\xi) &= \tfrac{(-6\gamma-2b_2(1+3\gamma_{21})+3\xi)\xi}{4+6\gamma_{31}+6\gamma_{32}} \end{aligned} \tag{4.3}$$

In this specific case, k_2^n and k_3^n relevant to NS are unknown; therefore, to ensure a real-time process, it is assumed $k_1^n = k_2^n = k_3^n$ in the first stage and $k_2^n = k_3^n$ in the second stage of each time step k, as shown in Fig. 4.3b. Two ways are proposed to implement the subcycling strategy of Fig. 4.3a in an actual RT test. (i) A scheme

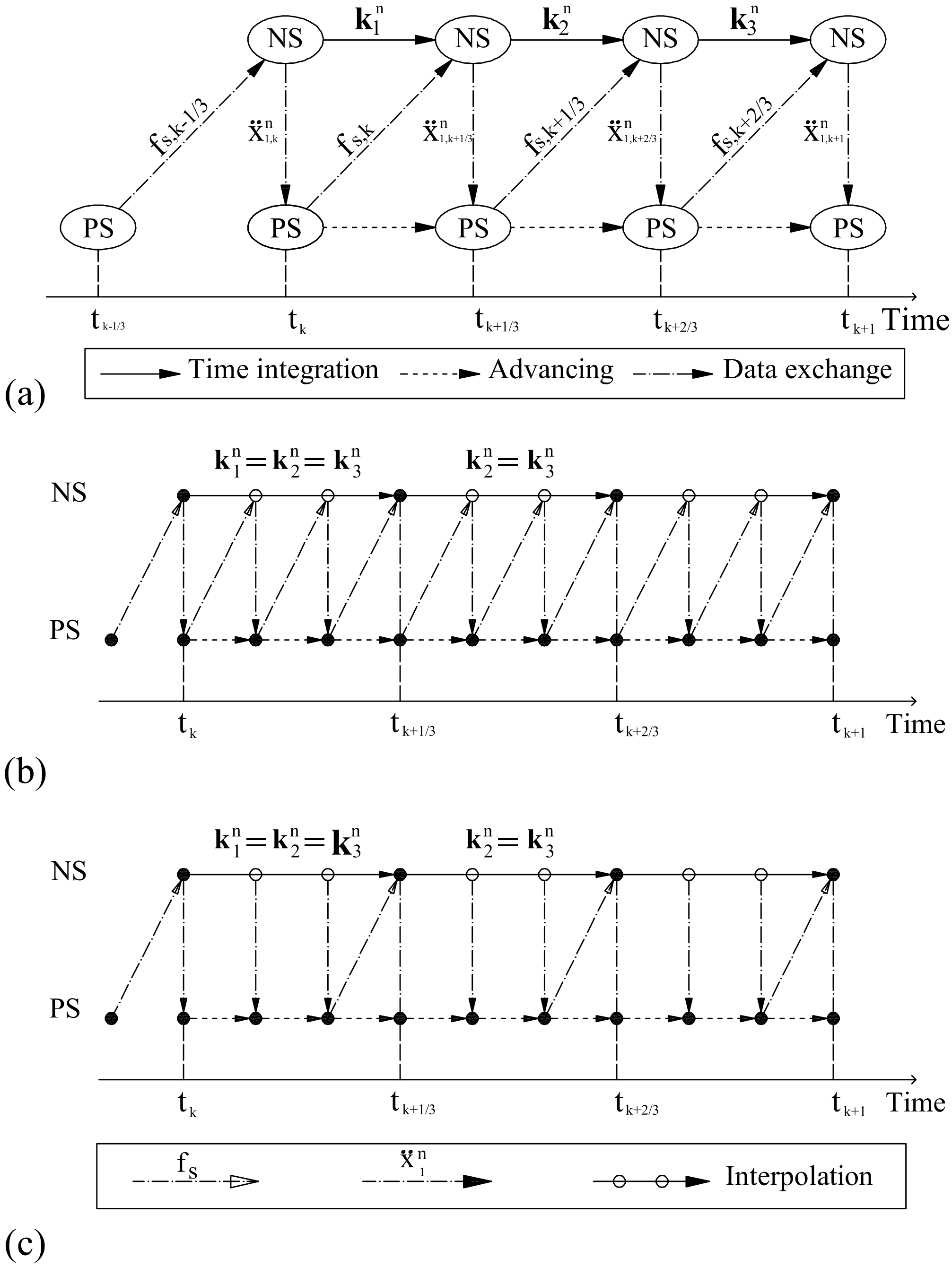


Fig. 4.3: Solution sequence of the time integration of a coupled system with the LSRT3 algorithm: (a) single time-step strategy; (b) multiple time-step strategy with equilibrium-based interpolation; (c) multiple time-step strategy with differentiation-based interpolation.

based on *Interpolation and equilibrium* where the state vector can be computed with (3.8) and (4.3); this procedure is shown in Fig. 4.3b for the LSRT method.(ii) A scheme based on *Differentiation and interpolation* where the relevant strategy for the LSRT3 algorithm is illustrated in Fig. 4.3c; in this case less information from the PS is needed.

For the time being, both subcycling strategies were only tested off-line; therefore, to evaluate the restoring force, the relevant acceleration $\ddot{x}_1$ coming from the PS in the *Interpolation and equilibrium* strategy, was obtained via equilibrium, *i.e.*

$$\ddot{x}_1(\xi) = \frac{1}{m_1^n}\left(f_e(\xi) - f_s(\xi) - c_1^n\dot{x}_1(\xi) - k_1^n x_1(\xi)\right) \tag{4.4}$$

where x_1 and $\dot{x}_1$ are scalar displacement and velocity, respectively. Conversely, $\ddot{x}_1$ in the *Differentiation and interpolation* strategy was obtained by the differentiation of (4.1).

4.3.2 Subcycling strategies for the Explicit Newmark method and the Newmark-Chang method

Along the lines of Section 3.5.3, in order to compare the Explicit Newmark method and the Newmark-Chang method with the LSRT algorithms described in Chapter 2, we used them with Δt_0. Since they are single step methods without stages, we implemented a subcycling strategy that foresees exchange of information only at the beginning and at the end of Δt_0. Nonetheless, to use subcycling that implied interpolation within Δt_0, we can proceed as follows. The evaluation of the displacement vector for the Newmark schemes was assumed to vary according to a complete third-order polynomial within each Δt as follows:

$$\mathbf{x}\left(t - t_k\right) = \mathbf{c}_0 + \mathbf{c}_1\left(t - t_k\right) + \mathbf{c}_2\left(t - t_k\right)^2 + \mathbf{c}_3\left(t - t_k\right)^3 \tag{4.5}$$

The coefficient vectors are determined for the Newmark-Chang method by means of interpolation of $\mathbf{x}$ and $\dot{\mathbf{x}}$ at time t_k and t_{k+1}, respectively:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{x}_k + \beta_1\dot{\mathbf{x}}_k\Delta t + \beta_2\Delta t^2\ddot{\mathbf{x}}_k \\ \dot{\mathbf{x}}_{k+1} &= \dot{\mathbf{x}}_k + \frac{\Delta t}{2}(\ddot{\mathbf{x}}_k + \ddot{\mathbf{x}}_{k+1}) \end{aligned} \tag{4.6}$$

Then,

$$\begin{aligned}\mathbf{c}_0 = \mathbf{x}_k, \mathbf{c}_1 = \dot{\mathbf{x}}_k, \mathbf{c}_2 &= \frac{6(\beta_1 - 1)\dot{\mathbf{x}}_k + (6\beta_2\ddot{\mathbf{x}}_k - \ddot{\mathbf{x}}_k - \ddot{\mathbf{x}}_{k+1})\Delta t}{2\Delta t}, \\ \mathbf{c}_3 &= \frac{4(1 - \beta_1)\dot{\mathbf{x}}_k + (\ddot{\mathbf{x}}_k + \ddot{\mathbf{x}}_{k+1} - 4\beta_2\ddot{\mathbf{x}}_k)\Delta t}{2\Delta t^2}\end{aligned} \tag{4.7}$$

Accordingly, the acceleration vector can be obtained by double differentiation of (4.6)a, *i.e.*

$$\begin{aligned}\ddot{\mathbf{x}}(\xi) &= \frac{6(\beta_1 - 1)\dot{\mathbf{x}}_k + (6\beta_2\ddot{\mathbf{x}}_k - \ddot{\mathbf{x}}_k - \ddot{\mathbf{x}}_{k+1})\Delta t}{\Delta t} \\ &+ 3\frac{4(1 - \beta_1)\dot{\mathbf{x}}_k + (\ddot{\mathbf{x}}_k + \ddot{\mathbf{x}}_{k+1} - 4\beta_2\ddot{\mathbf{x}}_k)\Delta t}{\Delta t}\xi\end{aligned} \tag{4.8}$$

If we set $\beta_1 = 1$, $\beta_2 = 0.5$, the acceleration vector for the Explicit-Newmark's method is recovered. Notice that the acceleration is interpolated by first-order polynomial. But it is different from a linear interpolation between $\ddot{\mathbf{x}}_k$ and $\ddot{\mathbf{x}}_{k+1}$. The proposed interpolation is derived from the original method. Therefore, this subcycling strategy preserves the original order of accuracy. In Section 4.4.3, order reduction of accuracy is discovered when implementing this subcycling strategy. However, this is not because of the subcycling strategy but because the previous coupling force is used in Eq. (4.13).

4.4 Numerical application on a coupled spring-pendulum system

4.4.1 Zero Stability analysis

Even though time stepping schemes like the LSRT algorithms are L-stable for uncoupled problems, as proved in Bursi et al. (2008) and in Subsection 3.5.3.2, for the linear and the linearized case, respectively, they can become unstable for the coupled case. This aspect of coupled systems treated in discrete time can be checked easily, through a zero-stability analysis (Kubler and Schiehlen, 2004; Bursi et al., 2008) that represents a sufficient condition for stability (Stuart and Humphries, 1996). Even though one chooses special initial conditions not to blow up the solution for a nonzero stable case, the presence of roundoff errors in the computations will still cause the numerical solution to blow up, and therefore, the zero-stability analysis becomes a necessary condition too. If one consider a vanishing step size $\Delta t \rightarrow 0$, this leads to a time invariant state vector $\mathbf{y}_{k+1} = \mathbf{y}_k = \mathbf{y}_0$. Thus, the zero-stability of the coupled substructures does not depend on the integration scheme. For the RTDS implementation with a displacement control of the actuator shown in Fig. 4.1d, the zero-stability analysis requires that the inputs, *i.e.* the coupling force u_k^n for the NS, and the acceleration u_k^p for the PS, be collected in a vector $\mathbf{u}_k$. The same can be done with the outputs through the vector $\mathbf{z}_k$. Then, one can relate the inputs and the outputs taken at the same instant at the interconnection with the relationship $\mathbf{u}_k = \mathbf{L}\mathbf{z}_k$, where $\mathbf{L}$ is a time invariant Boolean matrix. For the coupled system of Fig. 4.1d, one gets,

$$\mathbf{z}_k = \left\{ \begin{array}{c} \ddot{x}_{1,k} \\ f_{s,k-1} \end{array} \right\}, \quad \mathbf{L} = \left[\begin{array}{cc} 0 & L^p \\ L^n & 0 \end{array} \right], \quad \mathbf{u}_k = \left\{ \begin{array}{c} u_k^n \\ u_k^p \end{array} \right\} \tag{4.9}$$

where $f_{s,k-1} = m_1^p \ddot{x}_{1,k-1}^p + (k_2^p x_{2,k-1}^p + m_2^p g) cos(x_{3,k-1}^p) + m_1^p g$ and $L^p = L^n = 1$. If the output equations are time invariant and linearly dependent on the inputs, one can relate the input at time t_{k+1} to the input at time t_k through the following relation:

$$\mathbf{u}_{k+1} = \mathbf{C}_1 \mathbf{u}_k + \widetilde{\mathbf{f}}, \tag{4.10}$$

where the matrix $\mathbf{C}_1$ and the vector $\widetilde{\mathbf{f}}$ contain constant terms as a result of the vanishing time steps Δt. Hence, one obtains,

$$\begin{gathered} \mathbf{C}_1 = \left[\begin{array}{cc} 0 & m_1^p \\ 0 & -m_1^p / m_1^n \end{array} \right] \\ \widetilde{\mathbf{f}} = \left\{ \begin{array}{c} \left(k_2^p x_{2,k}^p + m_2^p g\right) cos\left(x_{3,k}^p\right) + m_1^p g \\ \frac{1}{m_1^n}\left(f_{e,k+1} - k_1^n x_{1,k+1}^n - c^n x_{2,k+1}^n + m_2^p g - \left(k_2^p x_{2,k}^p + m_2^p g\right) cos\left(x_{3,k}^p\right)\right) \end{array} \right\} \end{gathered} \tag{4.11}$$

If one uses stable integrators, zero-stability of the partitioned integration of two substructures is guaranteed, if an algebraic loop between the subsystems does not exist and if the spectral radius of $\mathbf{C}_1$ lies within the unit circle in the Argand-Gauss plane. In practice, the coupling of the NS and the PS must follow certain rules. For the coupled spring-pendulum system under exam this condition results to be:

$$m_2^p cos^2\left(x_{3,k}^p\right) + m_1^p \leq m_1^n \tag{4.12}$$

where m_1^p and m_1^n are the split-masses in Fig. 4.1b, m_2^p is the pendulum mass and x_3^p the relevant angle shown in Fig. 4.1d. If $x_{3,k}^p = 0$, the total physical masses present in the PS cannot be greater than the mass of the NS. This condition arises from the fact that we are dealing with feed-through systems and an algebraic loop arises between the acceleration u_k^p and the interface force u_k^n at the interface, because their magnitudes are both dependent on each other. The algebraic loop can only be removed through an iterative process to evaluate exactly interface quantities or the presence of filters Kubler and Schiehlen (2004). However, iterations cannot be performed in real time and some delay is always present in RTDS owing to the transfer system and/or the data acquisition process: so the algebraic loop is not eliminated and the zero-stability condition (4.12) obtained for the spring-pendulum system holds. In the previous proof and the forthcoming numerical simulations, a realistic delay of Δt between the acceleration at the interface and the coupling force is considered, i.e.

$$\ddot{x}_{1,k+1} = \frac{1}{m_1^n}(f_{e,k+1} - f_{s,k} - c^n\dot{x}_{1,k+1} - k^n x_{1,k+1} + (m_1^p + m_2^p)g) \tag{4.13}$$

This assumption and its consequences on stability and accuracy are confirmed from numerical simulations and heterogeneous tests presented herein and Section 4.5.2, respectively.

4.4.2 Numerical simulations

In numerical simulations on the coupled spring-pendulum system, the PS is also integrated with the same integrator as the NS, as shown in Fig. 4.3a. The equation of motion for the NS is represented by

$$m_1^n\ddot{x}_1^n + c_1^n\dot{x}_1^n + k_1^n x_1^n = f_e - f_s + (m_1^n + m_2^p)g \tag{4.14}$$

The kinmeatic quantity $\ddot{x}_1^n$ evaluated by a integrator in the NS is transferred to the PS whose governing equations are

$$\begin{cases} m_2^p\ddot{x}_2^p + k_2^p x_2^p - m_2^p(L + x_2^p)(\dot{x}_3^p)^2 - m_2^p\ddot{x}_1^p\cos x_3^p + m_2^p g(1 - \cos x_3^p) = 0 \\ (L + x_2^p)\ddot{x}_3^p + (g + \ddot{x}_1^p)\sin x_3^p + 2\dot{x}_2^p\dot{x}_3^p = 0 \end{cases} \tag{4.15}$$

To advance the solution of the PS, the same integrator is used to achieve the kinematic quantities at the end of the time step. The coupling force feedback to the NS is calculated with the equation

$$f_s = m_1^p\ddot{x}_1^n + k_2^p x_1^p\cos x_2^p + m_2^p g\cos x_2^p + m_1^p g \tag{4.16}$$

As shown in Fig. 4.3a, the coupling force has an inevitable delay of Δt in order to realize real time simulation, i.e., the coupling force feedback is the coupling force calculated from the previous step.

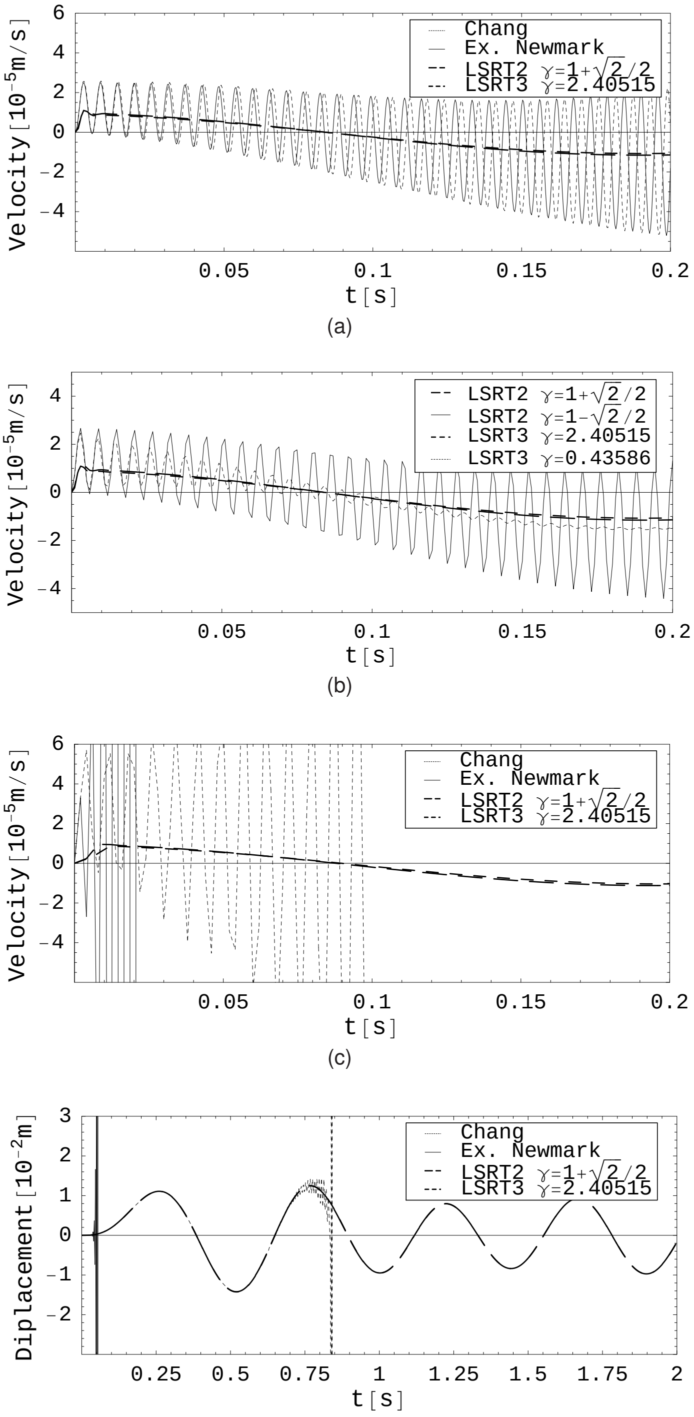
Table 4.1: Coupled Spring-pendula characteristics for numerical simulations

Coupled spring-pendulum: nonstiff problem							
m_1^n [kg]	c_1^n [kg/s]	k_1^n [N/m]	m_1^p [kg]	c_1^p [kg/s]	m_2^p [kg]	k_2^p [N/m]	L [m]
10	40	1000	1	0	0.34	400	0.167
Coupled spring-pendulum: stiff problem							
m_1^n [kg]	c_1^n [kg/s]	k_1^n [N/m]	m_1^p [kg]	c_1^p [kg/s]	m_2^p [kg]	k_2^p [N/m]	L [m]
10	40	1000	1	0	0.34	400000	0.167

Results of numerical simulations on a coupled spring-pendulum system schematically shown in Fig. 4.1(d) are depicted in Fig. 4.4. Relevant characteristics for the stiff case are summarized in Table 4.1. The time-history of the axial velocity $\dot{x}_2^p$ of the pendulum is represented in Fig. 4.4a when the system is integrated with $\Delta t_0 = 1/3$ms. Clearly, the LSRT methods exhibit higher numerical dissipation than the one relevant to the Newmark-Chang and Explicit-Newmark's methods. Moreover, if we reduce the γ values in the LSRT algorithms, they can trace the high-frequency components of the response as shown in Fig. 4.4b. When $\Delta t_0 = 2$ms, only the LSRT methods exhibit a stable behaviour, as it is evident for $\dot{x}_2^p$ and x_1^p in Fig. 4.4c and Fig. 4.4d, respectively.

4.4.3 Accuracy analysis

The accuracy analysis of the monolithic algorithms applied to the coupled spring-pendulum system is performed by means of numerical simulations on a nonstiff problem, whose characteristics are listed in Table 3.1. The convergence of displacement and velocity of the NS shown in Fig. 1c performed by different numerical integrators is shown in Fig. 4.5a and 4.5b, respectively. The reduction of accuracy to first order with respect to the uncoupled case, see Subsection 3.5.3.3 and Bursi et al. (2008), is evi-



(a)

(b)

(c)

(d)

Fig. 4.4: Numerical simulations for the coupled spring-pendulum stiff problem summarized in Table 3.1: (a) velocity $\dot{x}_2^p$ obtained with different methods and $\Delta t_0 = 1/3$ms; (b) velocity $\dot{x}_2^p$ provided by different LSRT algorithms and $\Delta t_0 = 1/3$ms; (c) velocity $\dot{x}_2^p$ with $\Delta t_0 = 2$ms; (d) displacement x_1^p with $\Delta t_0 = 2$ms.

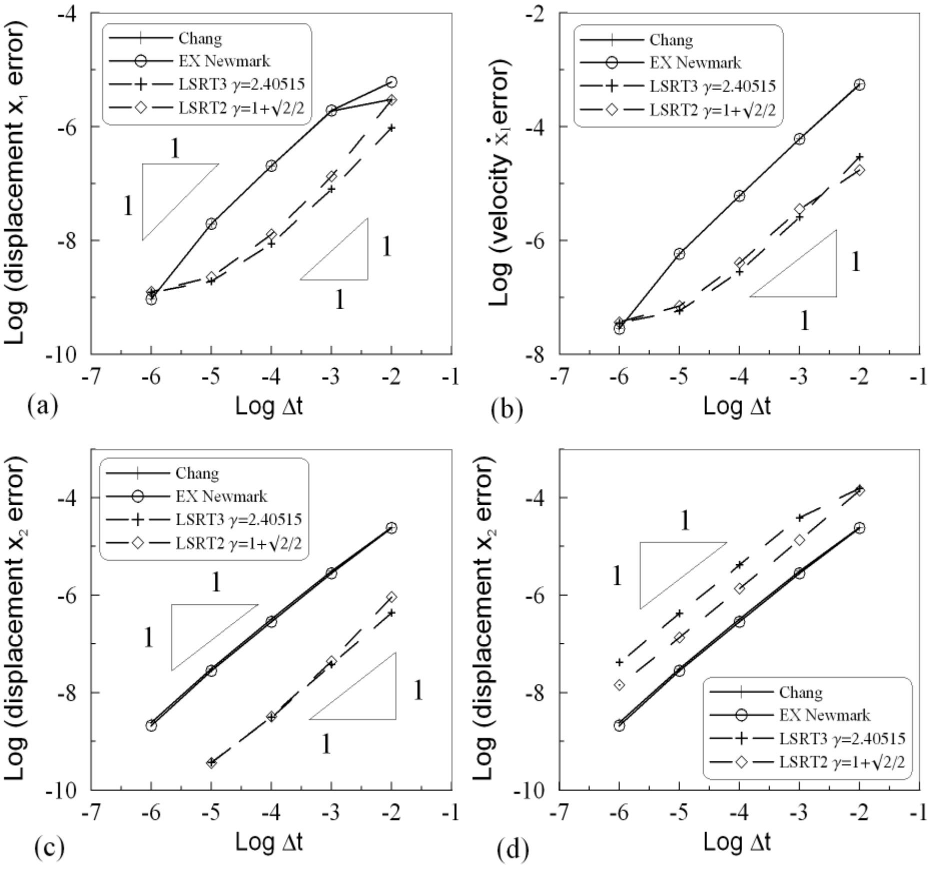


Fig. 4.5: Convergence of displacement and velocity for the coupled spring-pendulum nonstiff problem summarized in Table 3.1: (a) displacement x_1 error without subcycling; (b) velocity $\dot{x}_1$ error without subcycling; (c) displacement x_2 error with equilibrium-based interpolation; (d) displacement x_2 error with differentiation-based interpolation.

dent also for the nonlinear case. Moreover, the LSRT algorithms result to be more accurate in magnitude with respect to the Newmark-Chang and the Explicit-Newmark's method. When the subcycling strategies described in Subsection 4.3.2 are used, the *equilibrium-based* interpolation grants an increase of the axial displacement pendulum accuracy of the LSRT methods compared to the *differential-based interpolation*, as shown in Fig. 4.5c and Fig. 4.5d, respectively.

4.5 Application tests

Results of some experimental tests are described herein to confirm both analytical and numerical findings presented in the previous sections. As underlined in Section 4.3 all subcycling strategies were implemented off-line, and therefore, the basic integration methods presented in Fig 4.3a were considered. We recall that in heterogeneous tests, the NS has been implemented either as a linear or as a nonlinear spring-mass-damper system.

4.5.1 The Bouc-Wen model

In order to significantly test the performance of time-stepping schemes, in some cases, the NS was endowed with hysteretic springs governed by the Bouc-Wen model (Wen, 1976). Differently from other piecewise nonlinear models, the Bouc-Wen model offers a compact and continuous representation of hysteresis. In detail, the restoring force for an SDoF system reads

$$r\left(t\right) = K_h u\left(t\right) + z\left(t\right) \tag{4.17}$$

where the hysteretic component z is given by the solution of the nonlinear differential equation

$$\dot{z} = K_0 \dot{u} - \left|z\left(t\right)\right|^n \left[\beta sgn\left(z(t)\dot{u}\right) + \gamma\right] \dot{u} \tag{4.18}$$

where *sgn* is the signum function. The selection of suitable values for the parameters K_0, β, γ and n, entails hardening or softening nonlinearities. It can be proved that z attains an inelastic limit z_y, that reads

$$z_y = \left[\frac{K_0}{\beta + \gamma}\right]^{1/n} \tag{4.19}$$

provided that $\beta + \gamma > 0$. β, γ and n control the hysteresis shape: a hardening behaviour is simulated when $|\gamma| > |\beta|$ and $\gamma < 0$; otherwise, a softening behaviour is obtained. n modulates the sharpness of yielding and when $n \to \infty$ the elasto-perfectly plastic case is approached. The choice $n = 1$ implies that (4.18) can be analytically solved by means of exponential functions. In detail:

$$\begin{aligned} z &= \tfrac{K_0}{\beta+\gamma} - C_1 e^{-(\beta+\gamma)}u \quad & z\dot{u} > 0 \\ z &= \tfrac{K_0}{-\beta+\gamma} - C_2 e^{-(-\beta+\gamma)}u \quad & z\dot{u} < 0 \end{aligned} \tag{4.20}$$

where C_1 and C_2 depend on the initial conditions. Conversely, $n \neq 1$ implies an approximate solution of (4.18) based on hypergeometric functions.

4.5.2 Heterogeneous tests

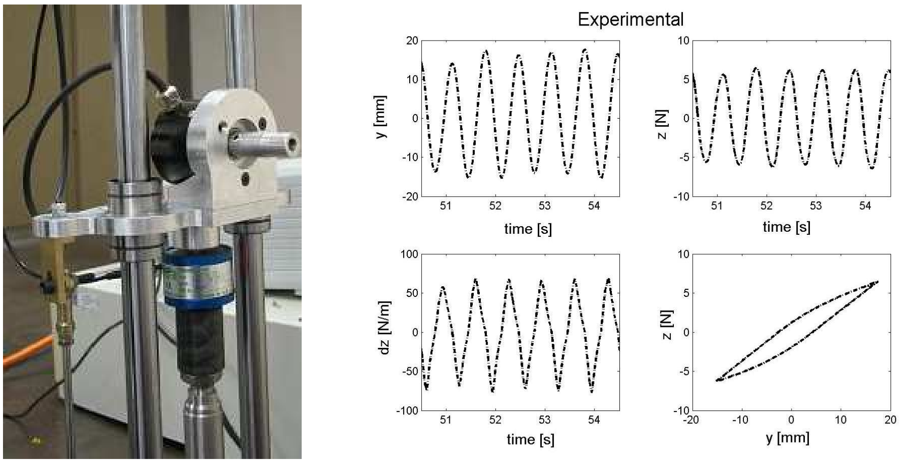
In order to perform real-time tasks a dSPACE DS1104 RD controller board was used, whilst MATLAB/Simulink was exploited to build the numerical model and implement the time-stepping schemes. The transfer system consisted of an electrically driven ball-screw actuator displacement controlled with an in-line mounted servo-motor controlled by a servo-drive. The heterogeneous system is sketched in Fig. 4.1d. The instrumentation used consisted of a load cell exploited to measure the force acting at the spring-pendulum pivot, a LVDT displacement transducer connected to the platform to track and control the actuator movement, a digital incremental encoder used to record both angular displacement and angular velocity of the pendulum and a LVDT connected to the pendulum bob, to measure the change in pendulum length. In all the experiments, the transfer system was controlled by means of a PID controller with a polynomial delay compensation technique for delay (Wallace et al., 2005).

The first test regards an SDoF system where the physical split-mass produces a purely inertial feedback force, whilst the NS is characterized by a hysteretic behaviour simulated with a Bouc-Wen model. System parameters are collected in Table 4.2. Fig. 4.6 shows the experimental results when the LSRT2 method with $\gamma = 1 - \sqrt{2}/2$ is used with $\Delta t = 0.001s$. Moreover, a sinusoidal external excitation with frequency $f = 1.5$ Hz amplitude $A = 12$ N is considered. The test is successful also when a nonlinear NS is considered.

The second test considers the spring-pendulum system, where the PS is schematically shown in Fig. 4.1d and whose parameter values are provided in Table 4.2. In this case the NS is linear. Fig. 4.7 shows the experimental results, in terms of x_1^p and x_2^p, see Fig. 1(d), when using the LSRT2 method with $\gamma = 1 - \sqrt{2}/2$ and $\Delta t = 0.002s$.

Table 4.2: Coupled system characteristics for experimental tests

SDOF system with a nonlinear Bouc-Wen NS							
m_1^n [kg]	$Bouc - Wen$	k_0 [N/m]	m_1^p [kg]	c_1^p [kg/s]	m_2^p [kg]	k_2^p [N/m]	L [m]
10	$\beta = 55, \alpha = 45$	1000	1	50	0	0	0
MDOF system with a spring-pendulum and a linear NS:							
m_1^n [kg]	c_1^n [kg/s]	k_1^n [N/m]	m_1^p [kg]	c_1^p [kg/s]	m_2^p [kg]	k_2^p [N/m]	L [m]
10	40	1000	1	0	0.34	400	0.167
MDOF system with a spring-pendulum and a nonlinear Bouc-Wen NS							
m_1^n [kg]	$Bouc - Wen$	k_0 [N/m]	m_1^p [kg]	c_1^p [kg/s]	m_2^p [kg]	k_2^p [N/m]	L [m]
10	$\beta = 55, \alpha = 45$	1000	1	0	0.34	400	0.167



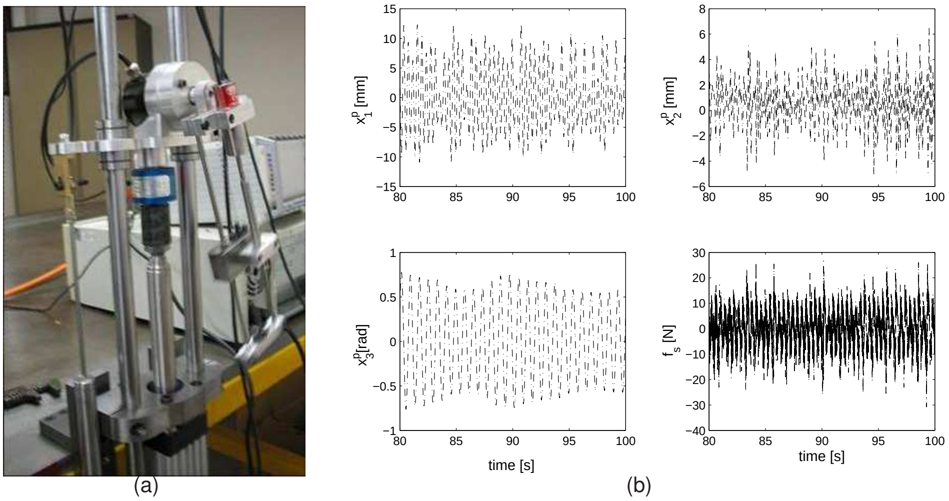
(a)

(b)

Fig. 4.6: Real-time test on an SDoF system with a nonlinear NS excited by an external sinusoidal force with f = 1.5 Hz and A = 12 N: (a) PS and set-up; (b) experimental results.

Moreover, a sinusoidal external excitation with frequency f = 2.2 Hz and amplitude A = 12 N is considered. Again experimental results show the capability of the LSRT2 method to deal with nonlinear systems and to entail a stable response.

The last RTDS test is characterized by nonlinearities in both the NS and the PS, respectively. The characteristics of the system are summarized in Table 4.2. Fig. 4.8



(a) (b)

Fig. 4.7: Real-time test on a spring-pendulum system with a linear NS excited by an external sinusoidal force with f = 2.2 Hz and A = 12 N: (a) PS and set-up; (b) experimental results.

shows the experimental results when the LSRT2 method is exploited with $\gamma = 1-\sqrt{2}/2$ and $\Delta t = 0.002s$. In addition, a sinusoidal external excitation with frequency f = 1.2 Hz and amplitude A = 20 N is considered. Though the force frequency is in resonance with the nonlinear NS, the test was successfully carried out.

4.6 Conclusions

In this chapter, a Rosenbrock-based algorithm, originally developed for real-time dynamic substructuring (RTDS), has been extended to the case of nonlinear systems. These algorithms are compatible with real-time test methods, such as RTDS, and can be applied in either a two- or three-stages linear implicit L-stable (LSRT) form. The computational overhead is kept to a minimum because they are linearly implicit, i.e. in order to avoid non-linear systems, they require only a single linearization and matrix decomposition per time step, and can use Jacobian matrices updated only at

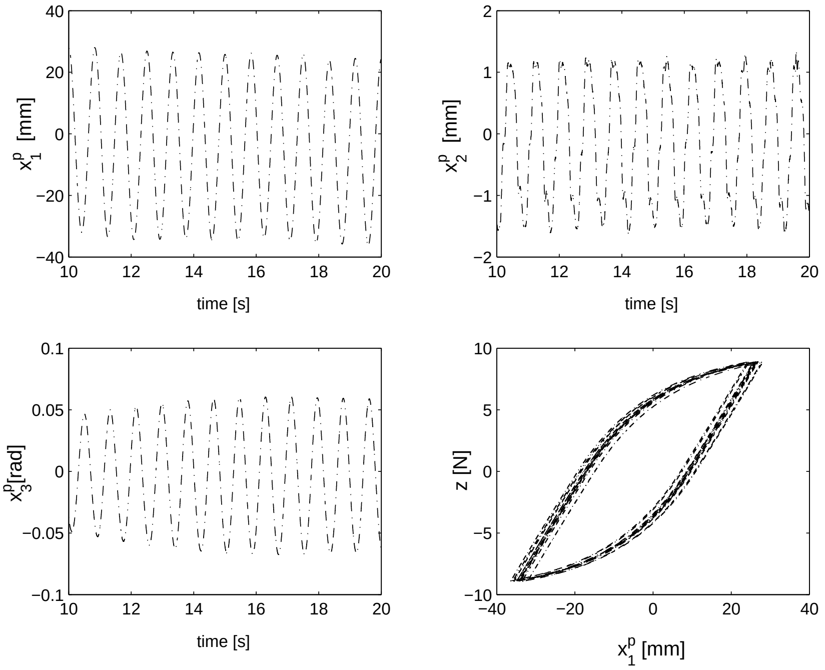


Fig. 4.8: Real-time test on a spring-pendulum system with nonlinear Numerical Substructure excited by an external sinusoidal force with *f* = 1.2 Hz and *A* = 20 N.

the beginning of each time step. The nonlinear test system used in this paper is a spring-pendulum system. Both numerical simulations and real-time substructuring test results from the spring-pendulum system, were used to demonstrate on the performance of the LSRT algorithms. This included both non-stiff and stiff nonlinear test problems. In all cases the LSRT algorithms compared very well with more established methods. In fact, both stability and accuracy properties of the proposed algorithms are typically better than the algorithms used for comparison. These favorable properties have been shown when considering nonlinearities both in the numerical and the physical substructures. A second key part of this study was to consider subcycling strategies as part of the proposed LSRT algorithms. Interpolation strategies were implemented, which showed favourable stability properties and an expected loss of accuracy in the coupled case owing to an assumed delay. As a result, the integration methods presented, can be used to obtain, stable, accurate and robust numerical integration when incorporated into heterogeneous test methods, such as RTDS.

Chapter 5

Partitioned time integration methods based on acceleration continuity

5.1 Introduction

The LSRT methods were observed to be efficient in structural dynamics in previous two chapters. Their applications to ODEs systems were well investigated in terms of accuracy, stability and nonlinear performance. The applications of these methods to RTDS tests in a monolithic way was studied through numerical analyses and experimental validations on a coupled spring-pendulum system. However, the requirement to use previous coupling force causes order reduction and stability limit. From this chapter, we start to show how the LSRT methods can be extended to partitioned time integration methods which permit independent subdomain computations.

Most of the previous works on partitioned time integration methods for structural problems were based on second-order integration methods (Geradin and Rixen, 2000; Park and Felippa, 2000; Combescure and Gravouil, 2002; Pegon and Magonette, 2002; Farhat and Chandesris, 2003; Lunk and Simeon, 2006; Brüls and Golinval, 2006; Jay and Negrut, 2007), such as the Newmark methods and the α methods. Also, there were some developed for first order problems (Nakshatrala et al., 2008, 2009; Zheng and Petzold, 2006; Zheng et al., 2009), by means of first-order integrators. In this chapter, our aim is to develop partitioned time integration methods for structural problems by the family of linearly implicit Rosenbrock methods. Attempts have been made to develop partitioned linear-implicit Euler methods for structural problems (Burgermeister et al., 2006) and to apply linearly implicit Rosenbrock methods to second-order DAEs (Arnold et al., 2007b).

Before going into details, a brief overview of some of the aforementioned works worth note is presented. To develop a partitioned time integration method for structural dynamics, one has three choices for kinematic constraints along the subdomain interfaces: continuity of displacement, velocity or acceleration. In References (Cardona and Geradin, 1989; Geradin and Rixen, 1997, 2000), the direct extensions of the Newmark methods and the α method to DAEs were introduced by enforcing the continuity of displacements or imposing the acceleration constraint. It was shown that tackling a system of index three directly by a numerical method was proponed for stability problems and probably lower global convergence; using the acceleration constraint may lead to a drift-off effect in the displacement response. To overcome both drawbacks, two prominent methods were conceived by Lunk and Simeon (2006); Jay and Negrut (2007), relying on the displacement and velocity constraints. It was illustrated that both methods had the second-order accuracy both in displacement and velocity variables, and in the article by Lunk and Simeon (2006), the method was proved to preserve the same stability condition as in the unconstrained case. However, those methods treat all differential equations and algebraic equations together which are not suitable for independent subdomain computation.

To allow independent subdomain computation, the FETI method was developed initially for quasi-static structural problems (Farhat and Roux, 1991) and later extended to structural dynamics (Farhat et al., 1995). Using this method, each subdomains can be solved independently and the inter-subdomain continuity can be enforced via Lagrange multipliers applied at the interfaces. Based on the FETI method, Gravouil and Combescure (2001) proposed a multi-time-step coupling method, labelled as the GC method. This method enables independent computation because an explicit form of Lagrange multiplier is derived from the velocity constraint after the free solutions of all subdomains. It was also shown that imposing velocity continuity at the interfaces leads to stable algorithm without considerable drift in the displacement constraint while imposing continuity of displacement or acceleration suffers from some stability and/or accuracy issues. The energy dissipation and the computation of the Lagrange multiplier at the fine time step were considered as drawbacks by Prakash and Hjelmstad (2004), who developed a variant of this method by enforcing continuity of velocities only at the coarse time step in the mesh. Another drawback is related to the staggered characteristics of the solution procedure, and Pegon and Magonette (2002) conceived a interfield parallel procedure where all subdomains can be advanced simultaneously and continuously.

In this chapter, efficient partitioned time integration methods using element-based partitioning for structural dynamics are presented. With these methods, one can firstly solve the interface Lagrange multipliers and subsequently advance the solution in all subdomains separately. Moreover, we provide a series of analytical and numerical studies to investigate the performances of the resulting partitioned time integration methods.

5.2 Derivation of an explicit Lagrange multiplier

In this section, we briefly state the derivation of an explicit Lagrange multiplier. Considering nonlinearities in the differential equations and constraint equations, Eq. (2.27) can be expressed in a general form as follows:

$$\begin{cases} \mathbf{M}^i\ddot{\mathbf{u}}^i = \mathbf{f}^i\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) + \mathbf{G}^{i^T}\left(\mathbf{u}^i\right)\Lambda, \ i = 1, \dots, S \\ \sum\limits_{i=1}^{S} \mathbf{g}^i\left(\mathbf{u}^i\right) = 0 \end{cases} \tag{5.1}$$

which are n_u equations governing the dynamic behaviour of all subdomains supplemented by n_c constraint equations, with $\mathbf{f}^i\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right)$ as the vector of applied and internal forces, $\mathbf{g}^i\left(\mathbf{u}^i\right)$ as the nonlinear holonomic interface vector and $\mathbf{G}^i\left(\mathbf{u}^i\right) = \frac{\partial \mathbf{g}^i}{\partial \mathbf{u}^i}$ as the constraint Jacobian matrix of the ith subdomain. In nonlinear finite element analysis, forces can be usually split into $\mathbf{f}^i\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) = -\mathbf{K}^i\left(\mathbf{u}^i\right) - \mathbf{C}\dot{\mathbf{u}}^i + \mathbf{f}_e^i(t)$ with the stiffness term $\mathbf{K}^i\left(\mathbf{u}^i\right)$, damping matrix $\mathbf{C}^i$ and the external force $\mathbf{f}_e^i(t)$.

In general, it is not advisable to tackle DAEs of index-three directly. In order to solve the DAEs (5.1), we lower its index by means of the following analytical differentiations. By differentiating the constraint equations with respect to time, one can get the constraints at velocity level

$$\sum_{i=1}^{S} \mathbf{G}^i \left(\mathbf{u}^i\right) \dot{\mathbf{u}}^i = 0 \tag{5.2}$$

A second differentiation step yields the constraints at acceleration level

$$\sum_{i=1}^{S} \mathbf{G}^i \left(\mathbf{u}^i\right) \ddot{\mathbf{u}}^i + \frac{\partial \mathbf{G}^i}{\partial \mathbf{u}^i} \dot{\mathbf{u}}^i \dot{\mathbf{u}}^i = 0 \tag{5.3}$$

Note that the partitioned methods in this thesis are mainly developed for the inter-subdomain linear constraint equations. It is assumed that the constraint Jacobian $\mathbf{G}^\mathbf{i}$ is constant with respect $\mathbf{u}^\mathbf{i}$. Therefore, the constraint at acceleration level can be expressed as: $\mathbf{G}^\mathbf{i}\ddot{\mathbf{u}}^\mathbf{i} = \mathbf{0}$. If we substitute in (5.1) the original constraints on displacement level by $\mathbf{G}\ddot{\mathbf{u}} = \mathbf{0}$, we get the so-called DAEs of index-one. In order to implement the LSRT methods to solve it, the resulting DAEs can be rearranged in the following form

$$\begin{cases} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^i \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \ddot{\mathbf{u}}^i \end{Bmatrix} = \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \mathbf{f}^i \left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) \end{Bmatrix} + \begin{bmatrix} \mathbf{0} \\ {\mathbf{G}^i}^T \end{bmatrix} \mathbf{\Lambda}, \ i = 1, \dots, S \\ \sum_{i=1}^{S} \left[\mathbf{0}, \mathbf{G}^i\right] \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \ddot{\mathbf{u}}^i \end{Bmatrix} = \mathbf{0} \end{cases} \tag{5.4}$$

With the assumption $\mathbf{y}^i = \begin{Bmatrix} \mathbf{u}^i \\ \dot{\mathbf{u}}^i \end{Bmatrix}$, we can obtain

$$\begin{cases} \mathbf{A}_i \dot{\mathbf{y}}^i = \mathbf{F}^i \left(\mathbf{y}^i, t\right) + \mathbf{C}_i^T \mathbf{\Lambda} \\ \sum_{i=1}^{S} \mathbf{C}_i \dot{\mathbf{y}}^i = \mathbf{0} \end{cases} \tag{5.5}$$

For ease of notation, the matrices **A**, and **C** for the *i*th subdomain are presented with subscript *i* to avoid double superscripts. We can solve the system (5.5) both for the state vectors and the Lagrange multiplier vector and finally get the explicit expressions

$$\dot{\mathbf{y}}^i = \mathbf{A}_i^{-1} \mathbf{F}^i \left(\mathbf{y}^i, t\right) + \mathbf{A}_i^{-1} \mathbf{C}_i^T \mathbf{\Lambda} \ i = 1, \dots, S \tag{5.6}$$

and

$$\mathbf{\Lambda} = -\left[\sum_{i=1}^{S} \mathbf{C}_i \mathbf{A}_i^{-1} \mathbf{C}_i^T\right]^{-1} \sum_{i=1}^{S} \mathbf{C}_i \mathbf{A}_i^{-1} \mathbf{F}^i \left(\mathbf{y}^i, t\right) \tag{5.7}$$

From a purely mathematical viewpoint, Eqs. (5.7) and (5.6) can be written in a compact form as

$$\dot{\mathbf{y}} = \mathbf{A}^{-1}\mathbf{F}(\mathbf{y}, t) + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda} \tag{5.8}$$

and

$$\mathbf{\Lambda} = -\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right]^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{F}(\mathbf{y}, t) \tag{5.9}$$

where,

$$\begin{aligned}
&\mathbf{y} = \left\{\left(\mathbf{y}^1\right)^T \cdots \left(\mathbf{y}^S\right)^T\right\}^T \\
&\mathbf{A} = \text{Blockdiagonal}\left[\mathbf{A}^1 \cdots \mathbf{A}^S\right] \\
&\mathbf{F}(\mathbf{y}, \mathbf{t}) = \left\{\left(\mathbf{F}^1\left(\mathbf{y}^1, t\right)\right)^T \cdots \left(\mathbf{F}^S\left(\mathbf{y}^S, t\right)\right)\right\}^T \\
&\mathbf{C} = \left[\mathbf{C}_1 \cdots \mathbf{C}_S\right]
\end{aligned} \tag{5.10}$$

Inserting the representation of the Lagrange multiplier vector from (5.9) into Eq. (5.8), we can obtain a system ODEs for variables **y**.

$$\dot{\mathbf{y}} = \mathbf{A}^{-1}\mathbf{F} - \mathbf{A}^{-1}\mathbf{C}^T\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right]^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{F}(\mathbf{y}, t) \tag{5.11}$$

The so-called ODEs approach is to solve the resulting ODEs. The ODEs approach solves all subdomains monolithically and therefore it is not suitable for independent simulations. But the proposed partitioned methods developed in this chapter are different from that: when the Lagrange multiplier vector achieved at the beginning of the time step or stage, each subdomain advances independently.

For the sake of comparison, we introduce another Lagrange multiplier vector Ψ to project solutions onto the velocity constraint, i.e.,

$$\begin{cases}
\mathbf{M}^i\dot{\mathbf{u}}^i = \mathbf{M}^i\dot{\mathbf{u}}^i + {\mathbf{G}^i}^T\mathbf{\Psi} \\
\mathbf{M}^i\ddot{\mathbf{u}}^i = \mathbf{f}^i\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) + {\mathbf{G}^i}^T\mathbf{\Lambda} \\
\sum\limits_{i=1}^{S}\mathbf{G}^i\dot{\mathbf{u}}^i = 0 \\
\sum\limits_{i=1}^{S}\mathbf{G}^i\ddot{\mathbf{u}}^i = 0
\end{cases} \tag{5.12}$$

Similarly, (5.12) can be written in the first order form as:

$$\begin{cases} \begin{bmatrix} \mathbf{M}^i & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^i \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \ddot{\mathbf{u}}^{\mathbf{i}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{M}^i\dot{\mathbf{u}}^i \\ \mathbf{f}^i\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) \end{Bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{G}^{i^T} \end{bmatrix} \begin{Bmatrix} \mathbf{\Psi} \\ \mathbf{\Lambda} \end{Bmatrix}, \; i = 1, \ldots, S \\ \sum_{i=1}^{S} \left[\mathbf{0}, \mathbf{G}^i\right] \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \ddot{\mathbf{u}}^i \end{Bmatrix} = \mathbf{0} \end{cases} \tag{5.13}$$

In the same manner as the previous case, (5.13) can be simplified into the same form as (5.5). Note that the differences exist in the expressions of $\mathbf{A}^i$, $\mathbf{F}^i$ and $\mathbf{C}$:

$$\begin{aligned} \mathbf{A}^i &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^i \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{M}^i & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^i \end{bmatrix} \\ \mathbf{F}^i &= \begin{Bmatrix} \dot{\mathbf{u}}^i \\ \mathbf{f}\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) \end{Bmatrix} \Rightarrow \begin{Bmatrix} \mathbf{M}^i\dot{\mathbf{u}}^i \\ \mathbf{f}\left(\mathbf{u}^i, \dot{\mathbf{u}}^i, t\right) \end{Bmatrix} \\ \mathbf{C} &= \begin{bmatrix} \mathbf{0} & \mathbf{G}^i \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{G}^i & \mathbf{0} \\ \mathbf{0} & \mathbf{G}^i \end{bmatrix} \\ \Lambda &= \Lambda \Rightarrow \begin{Bmatrix} \Psi \\ \Lambda \end{Bmatrix} \end{aligned} \tag{5.14}$$

Furthermore, we can obtain an explicit expression of the dual Lagrange multipliers vector as (5.7). In the following sections, not only the solution procedures but also the analytical investigations of accuracy and stability are based on Eqs. (5.6) and (5.7). Therefore, this case with dual Lagrange multipliers is also under consideration.

5.3 Formulations of the partitioned time integration methods

Based on the analysis of the previous section, two efficient partitioned time integration methods based on the LSRT1 method and the LSRT2 method, respectively, are presented. These methods consist in firstly solving the interface Lagrange multipliers and subsequently advancing the solutions in all subdomains separately.

In the thesis, a series of partitioned time integration methods are developed. In addition to the partitioned methods with no subcycling in this chapter, two types of

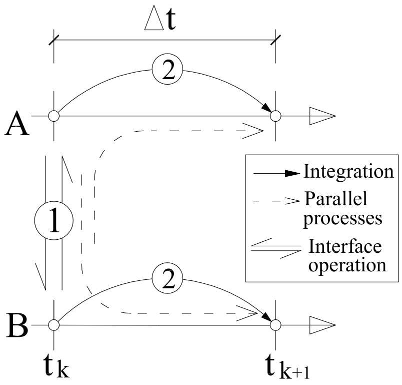


Fig. 5.1: The staggered solution procedure of the LSRT1-based partitioned method with $ss = 1$

extensions of them considering subcycling and a type of projection methods will be presented in the next three chapters, respectively. For easiness of notation, we define them in the following ways: i) based on the utilized integrator(s), we consider two classes of methods: the LSRT1-based method and the LSRT2-based method which are denoted by **R1** and **R2**, respectively; ii) based on time steps used, they can be classified into: non-subcycling cases and subcycling cases which are denoted by **N** and **S**, respectively; iii) based on solution procedures, they can be categorized into: staggered cases and parallel cases which are shorten as **S** and **P**; iv) based on interface continuity, first three types of methods are endowed with acceleration continuity and the last type with velocity continuity and they are denoted by **A** and **V**. We name the partitioned methods in inverse order. For example, the methods presented in the following subsection is abbreviated as **APNR1**, standing for Acceleration continuity, parallel solution procedure, Non-subcycling and the LSRT1 method.

5.3.1 A parallel solution procedure with no subcycling and based on the LSRT1 integrator (**APNR1**)

For simplicity, let us consider a system divided into two subdomains A and B. For notational simplicity we illustrate the solution procedure for advancing from t_k to t_{k+1} as shown in Fig. 5.1.

1. Caculate the Lagrange multiplier vector Λ_k at time t_k

$$\Lambda_k = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}_k^A + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}_k^B\right] \tag{5.15}$$

   where $\mathrm{H} = \mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{C}_A^T + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{C}_B^T$;

2. Compute $\mathbf{k}_1^i$ where $i = A, B$ and evaluate the solutions $\mathbf{y}_{k+1}^i$ at time t_{k+1} in both subdomains

$$\begin{aligned}\mathbf{k}_1^A &= [\mathbf{I} - \Delta t\gamma\mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}_k^A + \mathbf{C}_A^T\Lambda_k\right)\Delta t\\ \mathbf{y}_{k+1}^A &= \mathbf{y}_k^A + \mathbf{k}_1^A\end{aligned} \tag{5.16}$$

   and

$$\begin{aligned}\mathbf{k}_1^B &= [\mathbf{I} - \Delta t\gamma\mathbf{J}_B]^{-1}\,\mathbf{A}_B^{-1}\left(\mathbf{F}_k^B + \mathbf{C}_B^T\Lambda_k\right)\Delta t\\ \mathbf{y}_{k+1}^B &= \mathbf{y}_k^B + \mathbf{k}_1^B\end{aligned} \tag{5.17}$$

The case is defined as *parallel* because the interconnection is only done in the beginning of each stage to calculate Lagrange multipliers and then each subdomain can advance independently. Note that this method advances each subdomain with the same time step and therefore is also suitable for the cases with more than two subdomains.

5.3.2 A parallel solution procedure with no subcycling and based on the LSRT2 integrator (**APNR2**)

Here, both subdomains A and B are integrated by the LSRT2 algorithm with the same time step Δt. As shown in Fig. (5.2), we can describe the solution procedure as follows:

1. Calculate the Lagrange multiplier Λ_k at time t_k;

$$\Lambda_k = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}_k^A + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}_k^B\right] \tag{5.18}$$

2. Compute $\mathbf{k}_1^i$ where $i = A, B$ and evaluate the solutions $\mathbf{y}_{k+1/2}^i$ $(i = A, B)$ -First stage-;

$$\begin{aligned}\mathbf{k}_1^A &= [\mathbf{I} - \Delta t\gamma\mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}_k^A + \mathbf{C}_A^T\Lambda_k\right)\Delta t\\ \mathbf{y}_{k+1}^A &= \mathbf{y}_k^A + \frac{1}{2}\mathbf{k}_1^A\end{aligned} \tag{5.19}$$

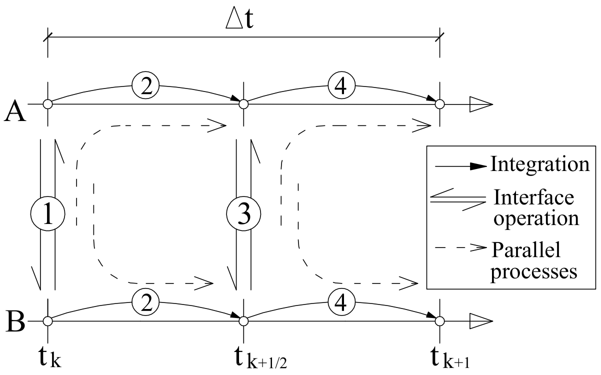


Fig. 5.2: The staggered solution procedure of the LSRT2-based partitioned method with $ss = 1$

and

$$\begin{aligned} \mathbf{k}_1^B &= [\mathbf{I} - \Delta t \gamma \mathbf{J}_B]^{-1} \, \mathbf{A}_B^{-1} \left(\mathbf{F}_k^B + \mathbf{C}_B^T \Lambda_k \right) \Delta t \\ \mathbf{y}_{k+1}^B &= \mathbf{y}_k^B + \frac{1}{2} \mathbf{k}_1^B \end{aligned} \tag{5.20}$$

3. Calculate the Lagrange multiplier $\Lambda_{k+\frac{1}{2}}$ at time $t_{k+\frac{1}{2}}$;

$$\Lambda_{k+\frac{1}{2}} = -\mathrm{H}^{-1} \left[\mathbf{C}_A \mathbf{A}_A^{-1} \mathbf{F}_{k+\frac{1}{2}}^A + \mathbf{C}_B \mathbf{A}_B^{-1} \mathbf{F}_{k+\frac{1}{2}}^B \right] \tag{5.21}$$

4. Compute $\mathbf{k}_2^i$ and advance the solutions to $\mathbf{y}_{k+1}^i$ in both subdomain, respectively, -Second stage-;

$$\begin{aligned} \mathbf{k}_2^A &= [\mathbf{I} - \Delta t \gamma \mathbf{J}_A]^{-1} \left(\mathbf{A}_A^{-1} \mathbf{F}_{k+\frac{1}{2}}^A + \mathbf{A}_A^{-1} \mathbf{C}_A^T \Lambda_{k+\frac{1}{2}}^A - \gamma \mathbf{J}_A \mathbf{k}_1^A \right) \Delta t \\ \mathbf{y}_{k+1}^A &= \mathbf{y}_k^A + \mathbf{k}_2^A \end{aligned} \tag{5.22}$$

and

$$\begin{aligned} \mathbf{k}_2^B &= [\mathbf{I} - \Delta t \gamma \mathbf{J}_B]^{-1} \left(\mathbf{A}_B^{-1} \mathbf{F}_{k+\frac{1}{2}}^B + \mathbf{A}_B^{-1} \mathbf{C}_B^T \Lambda_{k+\frac{1}{2}}^B - \gamma \mathbf{J}_B \mathbf{k}_1^B \right) \Delta t \\ \mathbf{y}_{k+1}^B &= \mathbf{y}_k^B + \mathbf{k}_2^B \end{aligned} \tag{5.23}$$

Since the LSRT2 method contains two stages, inter-domain exchange of information is not only required at the beginning of each time step but also at the beginning of the second stage. From the solution procedure of an entire time step, this method appears staggered. But if we divide the solution procedure in one time step into two stage processes, each stage process appears to be the same as the **APNR1** method. From this point of view, the **APNR2** method also enables parallel simulations.

In conclusion, the extension of the linearly implicit L-stable integrators to the DAEs with acceleration constraint is inspired by the following two points: i) the combination of the explicit expression of the multiplier vector and the explicit characteristics of the linearly implicit methods leads to partitioned methods with explicit characteristics; ii) the high-frequency dissipation property of the L-stable methods enforces numerical stabilization to the partitioned methods.

More precisely, we postulate the following properties for the partitioned methods: i) the accuracy of the adopted integrators are preserved for both state variables and Lagrange multipliers; ii) the partitioned methods are unconditionally stable in the linear case; iii) drift-off effects in displacement and velocity responses are limited and at least controllable; iv) the high-frequency dissipation property is retained and also adjustable; v) the partitioned methods permit the implementation of subcycling and further parallel extension. In the following sections, it will be shown that the first two requirements can be met. The last three requirements will be tested in the next two chapters.

5.4 Accuracy analysis

The accuracy is a very important aspect of partitioned time integration methods that needs to be considered when choosing an integrator for a DAE system. When solving DAEs, not only numerical estimates of differential solution quantities but also algebraic solution components introduce error. Moreover, these two sources of errors mostly interact even in a single step of the integration routine and propagate in the long term.

5.4.1 Local truncation error analysis of the **APNR1** method

For the sake of generality, we consider a non-autonomous system in the first order form (5.5) and we leave the parameter γ as a variable. Here, the local truncation error analysis is performed in a compact form as in (5.8) and (5.9). Suppose we know the exact solution at time t_k, *i.e.*, $\mathbf{y}_k = \mathbf{y}(t_k)$. Let us define the local truncation error as the difference between the numerical solution $\mathbf{y}_{k+1}$ and the exact solution $\mathbf{y}(t_{k+1})$:

$$\tau^{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} - \mathbf{y}(t_{k+1}) \tag{5.24}$$

Following the solution procedure as shown in Fig. 5.1, we can see that the first step is to insert the exact solution into Eq. (5.7)

$$\mathbf{\Lambda_k} = -\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right]^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{F_k} \tag{5.25}$$

where $\mathbf{F_k} := \mathbf{F}(\mathbf{y_k}, \mathbf{t_k})$. Note that the Lagrange multiplier vector is achieved by inserting the exact solution to Eq. (5.7) is exact.

Inserting the right-hand side of Eq. (5.25) into Eqs. (5.16) and (5.17) and compact all of them together, we can obtain the recursive formula

$$\mathbf{y}_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\left(\mathbf{A}^{-1}\mathbf{F}_k - \mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{F}_k\right)\Delta t \tag{5.26}$$

Assuming $\mathbf{Q} = \mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}$, the recursive formula can be simplified into

$$\mathbf{y}_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\mathbf{A}^{-1}\mathbf{F}_k\Delta t \tag{5.27}$$

Note that for the ODEs approach, the Jacobian matrix used in the inverse term of (5.27) is $(\mathbf{I} - \mathbf{Q})\mathbf{J}$ instead. This is the main difference between the proposed partitioned method and the ODEs approach.

Now, we use Taylor series expansion to express the exact solution at time t_{k+1}

$$\begin{aligned}\mathbf{y}(t_{k+1}) &= \mathbf{y}_k + \left(\mathbf{A}^{-1}\mathbf{F}_k + \mathbf{A}^{-1}\mathbf{C}^T\Lambda_k\right)\Delta t + O\left(\Delta t^2\right)\\ &= \mathbf{y}_k + \left(\mathbf{A}^{-1}\mathbf{F}_k - \mathbf{A}^{-1}\mathbf{C}^T\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right]^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{F}_k\right)\Delta t + O\left(\Delta t^2\right)\\ &= \mathbf{y}_k + (\mathbf{I} - \mathbf{Q})\mathbf{A}^{-1}\mathbf{F}_k\Delta t + O\left(\Delta t^2\right)\end{aligned} \tag{5.28}$$

By comparing (5.27) with (5.28), we can obtain the local truncation error

$$\begin{aligned}\tau^{\mathbf{y}}_{k+1} &= (\mathbf{I} - \mathbf{Q})\mathbf{A}^{-1}\mathbf{F}_k\Delta t - [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\mathbf{A}^{-1}\mathbf{F}_k\Delta t + O\left(\Delta t^2\right)\\ &= \Delta t\gamma\mathbf{J}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\mathbf{A}^{-1}\mathbf{F}_k\Delta t + O\left(\Delta t^2\right)\end{aligned} \tag{5.29}$$

which indicates a local truncation error of $\mathrm{O}\left(\Delta t^2\right)$. In the equation above, the key of the transformation is

$$\mathbf{I} - [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1} = \Delta t\gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1} \tag{5.30}$$

This transformation will be widely used hereafter.

In addition, the local truncation error of Λ needs to be investigated. Since the exact Lagrange multiplier vector can be calculated by inserting exact solution to Eq. (5.7), we can obtain:

$$\begin{aligned}\tau^{\Lambda}_{k+1} &= -\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right]^{-1}\mathbf{C}\mathbf{A}^{-1}\left(\mathbf{F}_{k+1} - \mathbf{F}\left(\mathbf{y}\left(t_{k+1}\right), t_{k+1}\right)\right)\\ &\leqslant -\left[\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right]^{-1}\mathbf{C}\mathbf{L}\tau^{\mathbf{y}}_{k+1}\end{aligned} \tag{5.31}$$

where $\mathbf{L}$ is the Lipschitz constant which must exist if a unique solution exists. Obviously, we can conclude that the order of local truncation error of the Lagrange multiplier vector is equal to the minimal order of the local truncation error of each subdomain. In this case, τ^{Λ}_{k+1} is of order $\mathrm{O}\left(\Delta t^2\right)$.

5.4.2 Local truncation error analysis of the **APNR2** method

In this case, two stages are included in one time step. Let us start with the analysis of the first stage. In the same manner as the analysis for the APNR1 method, the solution at time t_k is assumed to be exact. According to Eq. (3.6), we obtain the solution at the middle point

$$\mathbf{y}_{k+\frac{1}{2}} = \mathbf{y}_k + \frac{1}{2}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t \tag{5.32}$$

With Taylor series expansion, we can express the exact solution at the middle point

$$\mathbf{y}\left(t_{k+\frac{1}{2}}\right) = \mathbf{y}_k + \frac{1}{2}(\mathbf{I} - \mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t + \mathrm{O}\left(\Delta t^2\right) \tag{5.33}$$

Therefore, the local truncation error at $t_{k+\frac{1}{2}}$ can be obtained as

$$\begin{aligned}\tau^{\mathbf{y}}_{k+\frac{1}{2}} &= \frac{1}{2}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t - \frac{1}{2}(\mathbf{I} - \mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t - \mathrm{O}\left(\Delta t^2\right)\\ &= \frac{1}{2}[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\gamma\mathbf{J}(\mathbf{I} - \mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t^2 - \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{5.34}$$

which indicates that the local truncation error of the middle point is of $O\left(\Delta t^2\right)$. Following Eq. (5.31), the Lagrange multiplier at the middle point is also locally second-order accurate.

In the second stage, we can obtain the solution at time t_{k+1}

$$\mathbf{y}_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1} \begin{pmatrix} (\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}\left(\mathbf{y}_{k+1/2}, t_{k+1/2}\right) - \\ -\Delta t\gamma\mathbf{J}\,[\mathbf{I}-\Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k \end{pmatrix} \Delta t \tag{5.35}$$

According to Taylor series expansion, the term $\mathbf{F}\left(\mathbf{y}_{k+1/2}, t_{k+1/2}\right)$ can be expressed as follows

$$\mathbf{F}\left(\mathbf{y}_{k+1/2}, t_{k+1/2}\right) = \mathbf{F}_k + \frac{\Delta t}{2}\frac{\partial \mathbf{F}}{\partial t}\bigg|_k + \frac{\partial \mathbf{F}}{\partial \mathbf{y}}\bigg|_k \left(\mathbf{y}_{k+1/2} - \mathbf{y}_k\right) + O\left(\Delta t^2\right) \tag{5.36}$$

Since $\mathbf{y}_{k+1/2}$ is locally second-order accurate, it can be expressed as

$$\mathbf{y}_{k+1/2} = \mathbf{y}_k + \frac{1}{2}(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t + O\left(\Delta t^2\right) \tag{5.37}$$

Eq. (5.35) can be rearranged as

$$\mathbf{y}_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1} \begin{pmatrix} (\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\left(\mathbf{F}_k + \frac{\Delta t}{2}\frac{\partial \mathbf{F}}{\partial t}\bigg|_k + \frac{1}{2}\mathbf{J}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t\right) - \\ -\gamma\mathbf{J}\,[\mathbf{I}-\Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t \end{pmatrix} \Delta t \tag{5.38}$$

With Taylor series expansion, the exact solution at t_{k+1} can be expressed as

$$\begin{aligned} \mathbf{y}\left(t_{k+1}\right) &= \mathbf{y}_k + \mathbf{f}_k\Delta t + \frac{\Delta t^2}{2}\left(\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\bigg|_k \mathbf{f}_k + \frac{\partial \mathbf{f}}{\partial t}\bigg|_k\right) + O\left(\Delta t^3\right) \\ &= \mathbf{y}_k + (\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t + \frac{\Delta t^2}{2}\left((\mathbf{I}-\mathbf{Q})\,\mathbf{J}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k + \frac{\partial \mathbf{f}}{\partial t}\bigg|_k\right) + O\left(\Delta t^3\right) \end{aligned} \tag{5.39}$$

The comparison of the numerical solution and the exact solution yields

$$\begin{aligned} \tau^{\mathbf{y}}_{k+1} &= [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\left(\mathbf{F}_k + \frac{\Delta t}{2}\frac{\partial \mathbf{F}}{\partial t}\bigg|_k + \frac{1}{2}\mathbf{J}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t\right)\Delta t - \\ &- [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t\Delta t - (\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t \\ &- \frac{\Delta t^2}{2}\left((\mathbf{I}-\mathbf{Q})\,\mathbf{J}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k + \frac{\partial \mathbf{F}}{\partial t}\bigg|_k\right) + O\left(\Delta t^3\right) \\ &= \left(\mathbf{I} - [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\right)\gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t^2 + O\left(\Delta t^3\right) \\ &= \gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\gamma\mathbf{J}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I}-\mathbf{Q})\,\mathbf{A}^{-1}\mathbf{F}_k\Delta t^3 + O\left(\Delta t^3\right) \end{aligned} \tag{5.40}$$

which indicates that the local truncation for the solution at time t_{k+1} is of $O\left(\Delta t^3\right)$. Likewise, the Lagrange multiplier vector at time t_{k+1} is also locally third-order accurate.

5.4.3 Global error estimates

Since the proposed partitioned methods are self-starting, i.e., one does not need any starting procedure, the initialization error needs not to be taken into account. Moreover, the Lagrange multiplier vector is calculated from the state vector, i.e., it has no 'memory'. For these reasons, the global error analysis has mainly to deal with the error propagation in the differential variables. For linear systems, the amplification matrix is constant with time being. The result obtained in Subsection is still available here: if the local error is $\mathrm{O}\left(\Delta t^{p}\right)$, then the global error will be $\mathrm{O}\left(\Delta t^{p-1}\right)$.

5.5 Stability analysis

In inertia-type problems, only the low-frequency components of the response are usually of interest while the existence of high-frequency components of the response is inevitable. For these cases unconditionally stable algorithms are generally preferred over conditionally stable ones in that the latter ones require very small time step. Moreover, it is often advantageous for an algorithm to possess dissipation property in order to damp out any nonsignificant or spurious components of the higher modes. Conversely, dissipative algorithms may introduce dissipation at lower frequencies or do not permit user control of dissipation. Therefore, Hilber et al. (1977) suggested some requirements for a desired algorithm:

1. It should be unconditionally stable when applied to a linear problem.

2. It should possess numerical dissipation that could be controlled by a parameter other than the time step. In particular, no numerical dissipation should be possible.

3. The numerical dissipation should not affect the low-frequency components too strongly.

This section is devoted to an evaluation of Requirement 1 of the proposed partitioned algorithm. For constrained systems, model decomposition is not possible and therefore the approach to stability by Hughes (1987) to stability is not suitable. Though an energy approach was adopted for partitioned algorithms, for instance the GC method by Gravouil and Combescure (2001), a similar approach is not available for the Rosenbrock based partitioned methods, because the energy norm does not increase monotonically with time being. Another approach to stability were implemented for partitioned methods by Lunk and Simeon (2006), but the block elimination for the Rosenbrock based partitioned methods is very complicated and maybe impossible. For these reasons, we conduct an absolute stability analysis as follows: the recursive formula is expressed in a compact matrix, so that the stability can be investigated through the linear recurrences analysis by Zheng and Petzold (2006). Besides, Requirement 2 and 3 will be checked in the next chapters together with the partitioned method with $ss > 1$.

5.5.1 Stability analysis for the **APNR1** method

In order to investigate Requirement 1) of (Hilber et al., 1977), we linearize (5.5) and assume no external force, i.e., $\mathbf{F}(\mathbf{y}, t) = \mathbf{B}\mathbf{y}$. The application of the one-stage Rosenbrock algorithm leads to the following recurrence:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{k}_1 = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k\Delta t \tag{5.41}$$

$$\mathbf{J}\mathbf{y}_{k+1} = \mathbf{J}\mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k\Delta t \tag{5.42}$$

Let's assume that $\mathbf{J}\mathbf{y}_k = \mathbf{v}_k + \mathbf{w}_k$, where $\mathbf{v}_k = (\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k$ and $\mathbf{w}_k = \mathbf{Q}\mathbf{J}\mathbf{y}_k$. We rewrite Eq. (5.42) as

$$\mathbf{v}_{k+1} + \mathbf{w}_{k+1} = \mathbf{v}_k + \mathbf{w}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{J}\mathbf{v}_k\Delta t \tag{5.43}$$

Then, let's consider the following two recurrences:

$$\begin{aligned} \mathbf{v}_{k+1} &= \mathbf{v}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{J}\mathbf{v}_k\Delta t \\ \mathbf{w}_{k+1} &= \mathbf{w}_k \end{aligned} \tag{5.44}$$

Obviously, the former recurrence is the same as the monolithic one-stage Rosenbrock method. Therefore, the first recurrence is stable when $\gamma \geqslant 1/2$. The latter is constant recurrence and therefore stable. Hence following (Zheng and Petzold, 2006), the sum of these two recurrences is stable, and therefore the partitioned method is stable. This implies that the partitioned one-stage Rosenbrock method maintains the same stability condition as the progenitor method.

5.5.2 Stability analysis for the **APNR2** method

When the APNR2 method applied to a linear and nonautonomous system, one can advance the numerical solution from time t_k to time t_{k+1} as follows:

$$\mathbf{y}_{k+1} = \mathbf{y}_k + (\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1} \begin{pmatrix} (\mathbf{I} - \mathbf{Q})\,\mathbf{J}\left(\mathbf{y}_k + \frac{1}{2}\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k\Delta t\right) - \\ -\Delta t\mathbf{J}\gamma\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k \end{pmatrix} \Delta t \quad (5.45)$$

In order to adopt the linear recurrences approach to this problem, the following transformation is needed:

$$\begin{aligned} &2\mathbf{y}_{k+1} - \mathbf{y}_{k+1} = 2\mathbf{y}_k - \mathbf{y}_k + \\ &+ (\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1} \begin{pmatrix} (\mathbf{I} - \mathbf{Q})\,\mathbf{J}\left(2\mathbf{y}_k + \frac{1}{4}(\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}2\mathbf{y}_k\Delta t\right) \\ -(\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{y}_k \end{pmatrix} \Delta t \end{aligned} \quad (5.46)$$

Now, let's assume that $\mathbf{w}_k = 2\mathbf{y}_k$ and $\mathbf{v}_k = -\mathbf{y}_k$. The recurrence can be rearranged as

$$\begin{aligned} &\mathbf{w}_{k+1} + \mathbf{v}_{k+1} = \mathbf{w}_k + \mathbf{v}_k + \\ &+ (\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1} \begin{pmatrix} (\mathbf{I} - \mathbf{Q})\,\mathbf{J}\left(\mathbf{w}_k + \frac{1}{4}(\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{w}_k\Delta t\right) \\ +(\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{v}_k \end{pmatrix} \Delta t \end{aligned} \quad (5.47)$$

Then, consider the following two recurrences:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + (\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\left(\mathbf{w}_k + \frac{1}{4}(\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-1}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{w}_k\Delta t\right)\Delta t \quad (5.48)$$

and

$$\mathbf{v}_{k+1} = \mathbf{v}_k + (\mathbf{I} - \Delta t\mathbf{J}\gamma)^{-2}(\mathbf{I} - \mathbf{Q})\,\mathbf{J}\mathbf{v}_k\Delta t \quad (5.49)$$

The Recurrence (5.48) can be reformulated of the following two recurrences:

$$\begin{aligned} \mathbf{w}_{k+1/2} &= \mathbf{w}_k + \frac{1}{2}\left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-1}\left(\mathbf{I} - \mathbf{Q}\right)\mathbf{J}\mathbf{w}_k\Delta t \\ \mathbf{w}_{k+1} &= \mathbf{w}_{k+1/2} + \frac{1}{2}\left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-1}\left(\mathbf{I} - \mathbf{Q}\right)\mathbf{J}\mathbf{w}_{k+1/2}\Delta t \end{aligned} \tag{5.50}$$

Then, one can transform one step as follows

$$\mathbf{w}_{k+1/2} = \mathbf{w}_k + \left(\mathbf{I} - \left(\frac{1}{2}\Delta t\right)\mathbf{J}\left(2\gamma\right)\right)^{-1}\left(\mathbf{I} - \mathbf{Q}\right)\mathbf{J}\mathbf{w}_k\left(\frac{1}{2}\Delta t\right) \tag{5.51}$$

One can observe that this is similar to the APNR1 method, and this can be taken as the APNR1 method with the time step $\frac{1}{2}\Delta t$ and the corresponding parameter 2γ. We know that the APNR1 method is unconditionally stable when $\gamma \geqslant 1/2$. As a result, we can conclude that Recurrence (5.48) is unconditionally stable when $\gamma \geqslant 1/4$.

Now, let's analyse Recurrence (5.49). Left multiplying on both side by $\mathbf{J}$, (5.49) can be rewritten as

$$\mathbf{J}\mathbf{v}_{k+1} = \mathbf{J}\mathbf{v}_k + \left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-2}\mathbf{J}\left(\mathbf{I} - \mathbf{Q}\right)\mathbf{J}\mathbf{v}_k\Delta t \tag{5.52}$$

If one suppose that $\mathbf{J}\mathbf{v}_k = \mathbf{p}_k + \mathbf{q}_k$, where $\mathbf{p}_k = \left(\mathbf{I} - \mathbf{Q}\right)\mathbf{J}\mathbf{v}_k$ and $\mathbf{q}_k = \mathbf{Q}\mathbf{J}\mathbf{v}_k$, one can rewrite the latter recurrence as:

$$\mathbf{p}_{k+1} + \mathbf{q}_{k+1} = \mathbf{p}_k + \mathbf{q}_k + \left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-2}\mathbf{J}\mathbf{p}_k\Delta t \tag{5.53}$$

Let's consider the following two recurrences:

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-2}\mathbf{J}\mathbf{p}_k\Delta t \tag{5.54}$$

and

$$\mathbf{q}_{k+1} = \mathbf{q}_k \tag{5.55}$$

The propagation matrices of the above two recurrences are $\mathbf{I} + \left(\mathbf{I} - \Delta t\mathbf{J}\gamma\right)^{-2}\mathbf{J}\Delta t$ and $\mathbf{I}$.

For the absolute stability of Recurrence (5.54), let $z = -\Delta t\lambda_s$, where λ_s denotes an eigenvalue of the Jacobian matrix $\mathbf{J}$. Thus, we can obtain the eigenvalues of the amplification matrices:

$$R\left(z\right) = 1 - \left(1 + z\gamma\right)^{-2}z = \left(1 + z\gamma\right)^{-2}\left(\left(1 + z\gamma\right)^2 - z\right) \tag{5.56}$$

Note that $R(z)$ is the same as the usual stability function obtained from the second Dahlquists test equation (Dahlquist, 1978). If one imposes that $|R(z)| \leqslant 1$ with $z \geqslant 0$, one can obtain

$$\left|(1+z\gamma)^{-2}\left((1+z\gamma)^2 - z\right)\right| \leqslant 1 \tag{5.57}$$

Thus,

$$\gamma \geqslant 1/4 \tag{5.58}$$

which indicates that Recurrence (5.54) is unconditionally stable when $\gamma \geqslant 1/4$. Now, the two involved recurrences are both stable. The sum of them is therefore stable, which indicates that the partitioned scheme preserves the stability limit of the monolithic two-stage Rosenbrock method.

More investigations for the partitioned methods, such as numerical stability, numerical simulations and so forth, are postponed to the next chapter together with the subcycling cases.

5.6 Conclusions

In this chapter, we presented two partitioned methods, the APNR1 method and the APNR2 method. Using these methods, one can divide a large-scale system into smaller subdomains and integrate them independently. These methods also afford the possibility of solving different subdomains simultaneously with the Lagrange multiplier vectors achieved at the beginning of each time step or each stage.

In detail, an explicit expression of the Lagrange multiplier vector was derived from a system of DAEs with acceleration constraint in Section 5.2. Meanwhile, a case with an additional Lagrange multiplier vector that imposes the continuity of velocity was considered with the same form of expression. In section 5.3, two parallel solution procedures were conceived, based on the LSRT1 method and the LSRT2 method respectively. Furthermore, analytical investigations, in terms of accuracy and stability, were carried out for the partitioned methods in Section 5.4 and 5.5. The partitioned methods were observed to exhibit the same order of accuracy as their progenitors, and proved to be unconditionally stable when applied to a linear system.

The most important feature of these methods is that they enable different subdomains to advance independently which provide the possibility to be extended to the subcycling case in Chapter 6 and further to the case with a parallel solution procedure in Chapter 7.

CHAPTER 6

PARTITIONED TIME INTEGRATION METHODS WITH A STAGGERED SOLUTION PROCEDURE

6.1 Introduction

In the previous chapter, partitioned time integration methods based on the LSRT1 method and the LSRT2 method were presented for structural dynamics. With these methods, one can firstly solve the interface Lagrange multipliers and subsequently advance the solutions in all subdomains separately. Regarding complex structural problems, different subdomains may have different requirements of time step, for stability and/or accuracy considerations. In particular of RTDS tests, numerical and control requirements impose different time steps for NS and PS. In those cases, one available strategy, especially for linear problems, is model reduction that represents an effective way to lower computation burdens related to the integration of a complex NS. However, this strategy is not directly applicable to nonlinear problems and may become very inaccurate. In this chapter, the partitioned methods are extended to incorporate subcycling strategy where different subdomains are integrated with different time steps. For structural problems, considerable progress has been made in the development and analysis of subcycling strategies.

Initially, the concept of subcycling based on node partitioning was proposed by Belytschco et al. (1979), resulting in a constant-velocity subcycling algorithm. It was extended to a more general case with non-integer time step ratio in Neal and Belytschko (1989). Daniel (1998) pointed out that this subcycling algorithm suffers the *statistical stability* problem and the *momentum conservative* problem. An alternative method with a constant-acceleration assumption was devised in (Belytschko and Lu, 1993) but proved to unstable by Klisinski and Mostrom (1998). A modified version of the constant-acceleration subcycling algorithm was conceived by Daniel (1997) which is statistically stable. Wu and Smolinski (2000) proposed energy conserving algorithms which avoid the statistical stability but sacrifice accuracy. All the aforementioned subcycling strategies are based on the primal structuring formulation where an emulated structure was divided into two sets of nodes or elements and coupling of them is achieved through primal variables at the interface. Recently, Gravouil and Combescure (2001) presented a subcycling extension of the Newmark methods -the GC method- using the dual Schur substructuring, which was proved to be stable but energy dissipative. Prakash and Hjelmstad (2004) developed a variant of the GC method by enforcing continuity of velocities only at the coarse time step in the mesh.

This chapter presents a subcycling extension of the partitioned methods developed in the previous chapter. In order to advance the subdomain with the fine time step, intermediate solutions in the subdomain with the coarse time step are calculated by a linear interpolation between the initial solution and the solution at the end of time step or stage; and then the Lagrange multiplier is computed at the beginning of each substep. This chapter is organized in the order as following. Two subcycling strategies are devised for the APNR1 method and the APNR2 method, respectively, and their implementation flowcharts are proposed in detail in Section 6.2. Then, the accuracy analysis are conducted where both methods are discovered to preserve the same order of accuracy of their progenitors. Considering difficulties in analytical proof of subcycling methods, we carried out numerical stability analysis on a Single-DoF split-mass system and further on Multiple-DoF test problems. Lastly, numerical simulations are presented to validate the efficiency of the subcycling methods.

6.2 The partitioned time integration methods

6.2.1 A staggered solution procedure with subcycling and based on the LSRT1 integrator (**ASSR1**)

In this subsection, we incorporate the APNR1 method with subcycling. Let's assume that subdomain A is integrated with a coarse time step $\Delta t_A = \Delta t$ and Subdomain B with a fine time step $\Delta t_B = \Delta t/ss$, and the linear interpolation is implemented. The solution procedure for the above-mentioned system, shown in Fig. 6.1 is listed here:

1. Calculate the Lagrange multiplier vector Λ_k at time t_k;

$$\Lambda_k = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}_k^A + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}_k^B\right] \tag{6.1}$$

2. Compute $\mathbf{k}_1^A$ and evaluate the solution $\mathbf{y}_{k+1}^A$ at time t_{k+1} in Subdomain A;

$$\begin{aligned} &\mathbf{k}_1^A = [\mathbf{I} - \Delta t\gamma\mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}_k^A + \mathbf{C}_A^T\Lambda_k\right)\Delta t \\ &\mathbf{y}_{k+1}^A = \mathbf{y}_k^A + \mathbf{k}_1^A \end{aligned} \tag{6.2}$$

3. Interpolate the internal solutions of subdomain A with the linear interpolation formula;

$$\mathbf{y}^A_{k+\frac{j}{ss}} = (1 - \frac{j}{ss})\mathbf{y}_k^A + \frac{j}{ss}\mathbf{y}_{k+1}^A,\ (ss = 1,\ 2,\ \cdots,\ ss) \tag{6.3}$$

4. loops on the ss substeps of Subdomain B from t_k to t_{k+1} with $j = 1, 2, \cdots, ss$;

   a. Compute the Lagrange multiplier vector $\Lambda_{k+(j-1)/ss}$;

$$\Lambda_{k+\frac{j-1}{ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{j-1}{ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{j-1}{ss}}\right] \tag{6.4}$$

   b. Solve for the point $t_{k+j/ss}$ in Subdomain B;

$$\begin{aligned} &\mathbf{k}_1^B = \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\mathbf{A}_B^{-1}\left(\mathbf{F}_k^B + \mathbf{C}_B^T\Lambda_k\right)\frac{\Delta t}{ss} \\ &\mathbf{y}^B_{k+\frac{j}{ss}} = \mathbf{y}^B_{k+\frac{j-1}{ss}} + \mathbf{k}_1^B \end{aligned} \tag{6.5}$$

   If $j = ss$, then end the loop in Subdomain B.

Similarly to the GC method (Combescure and Gravouil, 2002), this procedure is staggered in the sense that the internal solutions calculated with the linear interpolation is required before advancing B. Differently, this partitioned method adopts the LSRT1 method which has an explicit solution of the state variables. Therefore, the interpolation involved in this method is only dependent on the target solution and the initial solution, which makes this method self-starting. But for the GC method, the interpolation is performed on the free solutions in the current and previous time step and therefore a starting solution procedure is required to initialize the GC method.

From another angle, the interpolation has an inherent similarity to the GC method. Since the initial solution in subdomain A in the current time step is also the target solution of the previous time step. From this point of view, the internal solutions is achieved through the linear interpolation between the target solutions of the current time step and that of the previous time step. For sake of notational simplicity, we define it as "parallel interpolation" in the sense that the linear interpolation is achieved between two solutions in the parallel positions of the current loop and the previous loop. In fact, both the GC method and the PM method adopt this type of interpolation: the internal solutions are calculated through the linear interpolation between the free solutions of the current loop and that of the previous loop. This will be detailed in Subsection 6.3.3. Moreover, the ASSR1 method appears unconditionally stable in all the cases considered in Subsection 6.4.2.

6.2.2 A staggered solution procedure with subcycling and based on the LSRT2 integrator (**ASSR2**)

Since the LSRT2 algorithm has two stages, we suppose that ss is even in order to avoid the case in which a substep in Subdomain B spans two stages in subdomain A. As shown in Fig. (6.2), the solution procedure can be summarized in the following pseudo-code:

1. Calculate the Lagrange multiplier vector Λ_k at time t_k;

$$\Lambda_k = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}_k^A + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}_k^B\right] \tag{6.6}$$

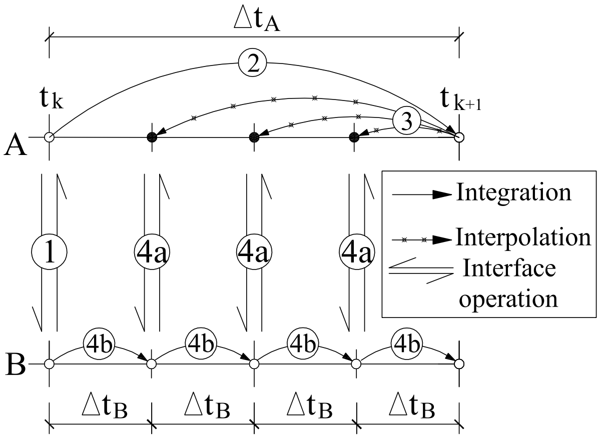


Fig. 6.1: The staggered solution procedure of the LSRT1-based partitioned method with $ss > 1$

2. Compute $\mathbf{k}_1^A$ and evaluate the solutions $\mathbf{y}^A_{k+\frac{1}{2}}$ at time $t_{k+\frac{1}{2}}$ in Subdomain A;

$$\begin{aligned}\mathbf{k}_1^A &= [\mathbf{I} - \Delta t\gamma \mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}_k^A + \mathbf{C}_A^T\Lambda_k\right)\Delta t\\ \mathbf{y}^A_{k+\frac{1}{2}} &= \mathbf{y}_k^A + \frac{1}{2}\mathbf{k}_1^A\end{aligned}\tag{6.7}$$

3. Solve the internal solutions of subdomain A with the linear interpolation formula

$$\mathbf{y}^A_{k+\frac{j}{2ss}} = (1-\frac{j}{ss})\mathbf{y}_k^A + \frac{j}{ss}\mathbf{y}^A_{k+\frac{1}{2}}\ ,\ (j = 1,\ 2,\ \cdots,\ ss)\tag{6.8}$$

4. Loops on the $ss/2$ substeps of the Subdomain B from t_k to t_{k+1} with $j = 1, 2, \cdots, ss/2$;

   a. Calculate the Lagrange multiplier vector $\Lambda_{k+\frac{j-1}{ss}}$;

$$\Lambda_{k+\frac{j-1}{ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{j-1}{ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{j-1}{ss}}\right]\tag{6.9}$$

   b. Compute $\mathbf{k}_1^B$ and evaluate the solution $\mathbf{y}^B_{k+\frac{2j-1}{2ss}}$ in Subdomain B -First stage-;

$$\begin{aligned}\mathbf{k}_1^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\mathbf{A}_B^{-1}\left(\mathbf{F}^B_{k+\frac{j-1}{ss}} + \mathbf{C}_B^T\Lambda^B_{k+\frac{j-1}{ss}}\right)\frac{\Delta t}{ss}\\ \mathbf{y}^B_{k+\frac{2j-1}{2ss}} &= \mathbf{y}^B_{k+\frac{2j-1}{2ss}} + \frac{1}{2}\mathbf{k}_1^B\end{aligned}\tag{6.10}$$

   c. Compute the lagrange multiplier vector $\Lambda_{k+\frac{2j-1}{2ss}}$;

$$\Lambda_{k+\frac{2j-1}{2ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{2j-1}{2ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{2j-1}{2ss}}\right]\tag{6.11}$$

d. Compute $\mathbf{k}_2^B$ and evaluate the solution $\mathbf{y}^B_{k+\frac{j}{ss}}$ in Subdomain B -Second stage-;

$$\begin{aligned} \mathbf{k}_2^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \left(\mathbf{A}_B^{-1}\left(\mathbf{F}^B_{k+\frac{2j-1}{2ss}} + \mathbf{C}_B^T \Lambda_{k+\frac{2j-1}{2ss}}\right) - \gamma \mathbf{J}_B \mathbf{k}_1^B\right)\frac{\Delta t}{ss} \\ \mathbf{y}^B_{k+\frac{j}{ss}} &= \mathbf{y}^B_{k+\frac{j-1}{ss}} + \mathbf{k}_2^B \end{aligned} \tag{6.12}$$

If $j = \frac{ss}{2}$, then end the loop in Subdomain B.

5. Calculate the Lagrange multiplier vector $\Lambda_{k+\frac{1}{2}}$;

$$\Lambda_{k+\frac{1}{2}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A \mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{1}{2}} + \mathbf{C}_B \mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{1}{2}}\right] \tag{6.13}$$

6. Compute $\mathbf{k}_2^A$ and evaluate the solution $\mathbf{y}^A_{k+1}$ in Subdomain A -Second stage-;

$$\begin{aligned} \mathbf{k}_2^A &= [\mathbf{I} - \Delta t \gamma \mathbf{J}_A]^{-1}\left(\mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{1}{2}} + \mathbf{A}_A^{-1}\mathbf{C}_A^T \Lambda^A_{k+\frac{1}{2}} - \gamma \mathbf{J}_A \mathbf{k}_1^A\right)\Delta t \\ \mathbf{y}^A_{k+1} &= \mathbf{y}^A_k + \mathbf{k}_2^A \end{aligned} \tag{6.14}$$

7. Interpolate the internal solutions of A with the following linear interpolation formula

$$\mathbf{y}_{k+\frac{j+ss}{2ss}^A} = (1 - \frac{j}{ss})\mathbf{y}_{k+\frac{1}{2}^A} + \frac{j}{ss}\mathbf{y}^A_{k+1}\ ,(j = 1,\ 2,\ \cdots,\ ss) \tag{6.15}$$

8. Loops on the $ss/2$ substeps of the subdomain B form $t_{k+1/2}$ to t_{k+1} with $j = 1, 2, \cdots, ss/2$;

a. Calculate the Lagrange multipliers $\Lambda_{k+\frac{1}{2}+\frac{j-1}{ss}}$;

$$\Lambda_{k+\frac{1}{2}+\frac{j-1}{ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A \mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{1}{2}+\frac{j-1}{ss}} + \mathbf{C}_B \mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{1}{2}+\frac{j-1}{ss}}\right] \tag{6.16}$$

b. Compute $\mathbf{k}_1^B$ and evaluate the solution $\mathbf{y}^B_{k+\frac{1}{2}+\frac{2j-1}{2ss}}$ in Subdomain B -Second stage-;

$$\begin{aligned} \mathbf{k}_1^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \mathbf{A}_B^{-1}\left(\mathbf{F}^B_{k+\frac{1}{2}+\frac{j-1}{ss}} + \mathbf{C}_B^T \Lambda^B_{k+\frac{1}{2}+\frac{j-1}{ss}}\right)\frac{\Delta t}{ss} \\ \mathbf{y}^B_{k+\frac{1}{2}+\frac{2j-1}{2ss}} &= \mathbf{y}^B_{k+\frac{1}{2}+\frac{2j-1}{2ss}} + \frac{1}{2}\mathbf{k}_1^B \end{aligned} \tag{6.17}$$

c. Calculate the Lagrange multipliers $\Lambda_{k+\frac{1}{2}+\frac{2j-1}{2ss}}$;

$$\Lambda_{k+\frac{1}{2}+\frac{2j-1}{2ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A \mathbf{A}_A^{-1}\mathbf{F}^A_{k+\frac{1}{2}+\frac{2j-1}{2ss}} + \mathbf{C}_B \mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{1}{2}+\frac{2j-1}{2ss}}\right] \tag{6.18}$$

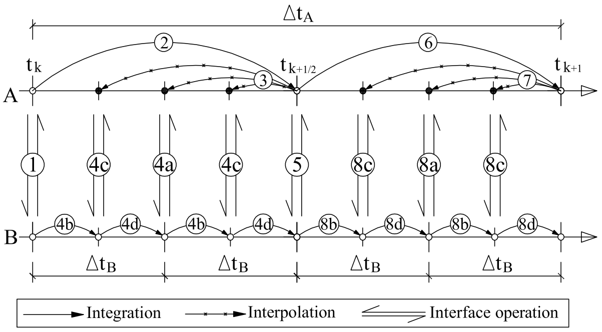


Fig. 6.2: The staggered solution procedure of the LSRT2-based partitioned method with $ss > 1$

d. Compute $\mathbf{k}_2^B$ and evaluate the solution $\mathbf{y}^B_{k+\frac{1}{2}+\frac{j}{ss}}$ in Subdomain B -Second stage-;

$$\begin{aligned} \mathbf{k}_2^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \left(\mathbf{A}_B^{-1}\left(\mathbf{F}^B_{k+\frac{1}{2}+\frac{2j-1}{2ss}} + \mathbf{C}_B^T \Lambda_{k+\frac{1}{2}+\frac{2j-1}{2ss}}\right) - \gamma \mathbf{J}_B \mathbf{k}_1^B\right)\frac{\Delta t}{ss} \\ \mathbf{y}^B_{k+\frac{1}{2}+\frac{j}{ss}} &= \mathbf{y}^B_{k+\frac{1}{2}+\frac{j-1}{ss}} + \mathbf{k}_2^B \end{aligned} \tag{6.19}$$

If $j = \frac{ss}{2}$, then end the loop in Subdomain B.

Differently from the ASSR1 method, this process contains two stages in Subdomain A and therefore two interpolations are required. Meanwhile, the socalled parallel interpolation is not available in this procedure. In the first stage of Subdomain A, the linear interpolation is performed between the initial solution and the solution at $t_{k+\frac{1}{2}}$. And in the second stage of Subdomain A, the linear interpolation is performed between the initial solution and the solution at t_{k+1}. These two interpolations will be discovered to be a sufficient condition for second-order accuracy. However, this method with $\gamma_B = 1 - \frac{\sqrt{2}}{2}$ sometime becomes conditionally stable as shown in Subsection 6.4.2.

6.3 Accuracy analysis

In this case, subcycling strategy is implemented and therefore the recursive formula for one step computation is relative more complex. It is not easy to derive the recursive in the compact form. The synthetic approach used for the APNR1 method where all subdomains are considered simultaneously is not suitable in this case. Here, we adopt an alternative way to deal with the integration of each subdomain separately.

6.3.1 Local truncation error analysis of the **ASSR1** method

First of all, let's discuss the integration in Subdomain A. Similarly, the initial solution is supposed to be exact, $\mathbf{y}_k^A = \mathbf{y}^A(t_k)$, and analogously $\Lambda_k^A = \Lambda^A(t_k)$. The integration in Subdomain A is the same as the APNR1 method. Therefore, the solution $\mathbf{y}_{k+1}^A$ has local truncation error of $\mathrm{O}\left(\Delta t^2\right)$ and it can be expressed as

$$\mathbf{y}_{k+1}^A = \mathbf{y}_k^A + \dot{\mathbf{y}}_k^A \Delta t + \mathrm{O}\left(\Delta t^2\right) \tag{6.20}$$

The solutions inside the time step in Subdomain A are solved by linear interpolation (6.3)

$$\begin{aligned}
\mathbf{y}_{k+j/ss}^A &= \left(1 - \frac{j}{ss}\right)\mathbf{y}_k^A + \frac{j}{ss}\mathbf{y}_{k+1}^A \\
&= \left(1 - \frac{j}{ss}\right)\mathbf{y}_k^A + \frac{j}{ss}\left(\mathbf{y}_k^A + \dot{\mathbf{y}}_k^A \Delta t + \mathrm{O}\left(\Delta t^2\right)\right) \\
&= \mathbf{y}_k^A + \frac{j\Delta t}{ss}\dot{\mathbf{y}}_k^A + \mathrm{O}\left(\Delta t^2\right)
\end{aligned} \tag{6.21}$$

which indicates a local truncation error of $\mathrm{O}\left(\Delta t^2\right)$.

Now, let's analyse the integration of Subdomain B. The integration of the first substep is the same as the case without subcycling too, but the time step used is not Δt but $\Delta t/ss$. At the end of the first substep, we can conclude that solutions of both subdomains, whether solved by interpolation or integration, have the local truncation error of $\mathrm{O}\left(\Delta t^2\right)$. Therefore, the Lagrange multiplier at the beginning of the second substep has a local truncation error of $\mathrm{O}\left(\Delta t^2\right)$. The solution at the end of the second substep is

$$\mathbf{y}_{k+2/ss}^B = \mathbf{y}_{k+1/ss}^B + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\mathbf{F}_{k+1/ss}^B + \mathbf{A}_B^{-1}\mathbf{C}_B^T\Lambda_k^B\right)\Delta t \tag{6.22}$$

In order to conduct a truncation error analysis, the components $\mathbf{y}^{B}_{k+2/ss}$, $\mathbf{F}^{B}_{k+1/ss}$ and Λ_B need to be expressed in terms of $\mathbf{y}_k$ and its derivatives.

$\mathbf{y}^{B}_{k+2/ss}$ and Λ_B are locally second-order accurate which can be expressed as

$$\begin{aligned}\mathbf{y}^{B}_{k+1/ss} &= \mathbf{y}^{B}_{k} + \frac{\Delta t}{ss}\dot{\mathbf{y}}^{B}_{k} + \mathrm{O}\left(\Delta t^2\right)\\ \Lambda^{B}_{k+1/ss} &= \Lambda^{B}_{k} + \frac{\Delta t}{ss}\dot{\Lambda}^{B}_{k} + \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{6.23}$$

The second component $\mathbf{F}^{B}_{k+1/ss}$ can be expressed by considering Taylor's series expansion

$$\begin{aligned}\mathbf{F}^{B}_{k+1/ss} &= \mathbf{F}^{B}_{k} + \mathbf{J}_B\left(\mathbf{y}^{B}_{k+1/ss} - \mathbf{y}^{B}_{k}\right) + \left.\frac{\partial \mathbf{F}}{\partial t}\right|_k \frac{\Delta t}{ss} + \mathrm{O}\left(\Delta t^2\right)\\ &= \mathbf{F}^{B}_{k} + \mathbf{J}_B\frac{\Delta t}{ss}\dot{\mathbf{y}}^{B}_{k} + \left.\frac{\partial \mathbf{F}}{\partial t}\right|_k \frac{\Delta t}{ss} + \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{6.24}$$

By inserting the three terms into (6.22) and simplifying it, we can obtain

$$\mathbf{y}^{B}_{k+2/ss} = \mathbf{y}^{B}_{k} + \frac{\Delta t}{ss}\dot{\mathbf{y}}^{B}_{k} + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\dot{\mathbf{y}}^{B}_{k}\frac{\Delta t}{ss} + \mathrm{O}\left(\Delta t^2\right) \tag{6.25}$$

If compared with the Taylor's series expansion $\mathbf{y}^{B}_{k+2/ss} = \mathbf{y}^{B}_{k} + \frac{2\Delta t}{ss}\dot{\mathbf{y}}^{B}_{k} + \mathrm{O}\left(\Delta t^2\right)$, we can say that the solution at the end of the second substep is locally second order accurate.

In the same manner, we can treat the following substeps. Hence, we can conclude that the solution at the end of the step is also locally second-order accurate; likewise the Lagrange multiplier vector exhibits a local truncation error of $\mathrm{O}\left(\Delta t^2\right)$.

6.3.2 Local truncation error analysis of the **ASSR2** method

In this case, the utilized LSRT2 algorithm has two stages. We can use the same strategy as the ASSR1 method, but the process will be quite complicated. Here, we follow the solution procedure and deal with each subdomain separately. Moreover, we assume a reference solution procedure with exact Lagrange multiplier vector along stages together.

For easiness of notation, we define the reference as ”unconstrained integration” because the process is equivalent to the integration of uncoupled systems, while the partitioned procedure is considered to be ”constrained integration”. Moreover, the symbols involved in ”unconstrained integration” are denoted by one bar above, such as $\bar{y}$. Therefore, the governing equations can be expressed as

$$\begin{aligned} \mathbf{A}_A \dot{\bar{\mathbf{y}}}_A &= \mathbf{F}(\bar{\mathbf{y}}_A, t) + C_A^T \Lambda(t) \\ \mathbf{A}_B \dot{\bar{\mathbf{y}}}_B &= \mathbf{F}(\bar{\mathbf{y}}_B, t) + C_B^T \Lambda(t) \end{aligned} \tag{6.26}$$

In the first stage of subdomain A, both ”constrained” and ”unconstrained” integrations have the same initial solution -exact solution $\mathbf{y}(t_k)$ and exact Lagrange multiplier vector $\Lambda(t_k)$, i.e.,

$$\begin{aligned} \mathbf{y}_k^A &= \bar{\mathbf{y}}_k^A \\ \Lambda_k &= \bar{\Lambda}_k \end{aligned} \tag{6.27}$$

From Eq. (3.6), we can easily obtain

$$\begin{aligned} \mathbf{k}_1^A &= \bar{\mathbf{k}}_1^A \\ \mathbf{y}_{k+1/2}^A &= \bar{\mathbf{y}}_{k+1/2}^A \end{aligned} \tag{6.28}$$

As we know, the middle point in the direct LSRT2 algorithm has the local truncation error of $\mathrm{O}\left(\Delta t^2\right)$. Therefore,

$$\tau_{k+1/2}^{\mathbf{y}_A} = \tau_{k+1/2}^{\bar{\mathbf{y}}_A} = \mathrm{O}\left(\Delta t^2\right) \tag{6.29}$$

As before, we can easily prove that the solutions calculated by the linear interpolation in the first stage are locally second-order accurate, i.e.,

$$\tau_{k+\frac{j}{2ss}}^{\mathbf{y}_A} = \mathrm{O}\left(\Delta t^2\right) \tag{6.30}$$

Similarly, both ”constrained” and ”unconstrained” integrations in the first stage of subdomain B can give the same result. Therefore, we can obtain

$$\begin{aligned} \mathbf{k}_1^B &= \bar{\mathbf{k}}_1^B \\ \mathbf{y}_{k+\frac{1}{2ss}}^B &= \bar{\mathbf{y}}_{k+\frac{1}{2ss}}^B \end{aligned} \tag{6.31}$$

$$\tau_{k+\frac{1}{2ss}}^{\mathbf{y}_B} = \tau_{k+\frac{1}{2ss}}^{\bar{\mathbf{y}}_B} = \mathrm{O}\left(\Delta t^2\right) \tag{6.32}$$

In agreement with (5.31), we can say that the Lagrange multiplier$\Lambda_{k+\frac{1}{2ss}}$ is also locally second-order accurate. In the second stage of the first substep in subdomain B, the solutions at the end of ”constrained” and ”unconstrained” integrations can be expressed as

$$\mathbf{y}_{k+\frac{1}{ss}}^B = \mathbf{y}_{k+\frac{1}{2ss}}^B + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \left(\mathbf{A}_B^{-1} \mathbf{F}_{k+\frac{1}{2ss}}^B + \mathbf{A}_B^{-1} \mathbf{C}_B^T \Lambda_{k+\frac{1}{2ss}}^B - \gamma \mathbf{J}_B \mathbf{k}_1^B\right) \frac{\Delta t}{ss} \tag{6.33}$$

$$\bar{\mathbf{y}}^B_{k+\frac{1}{ss}} = \bar{\mathbf{y}}^B_{k+\frac{1}{2ss}} + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \left(\mathbf{A}_B^{-1}\bar{\mathbf{F}}^B_{k+1/2/ss} + \mathbf{A}_B^{-1}\mathbf{C}_B^T\bar{\Lambda}^B_{k+\frac{1}{2ss}} - \gamma \mathbf{J}_B\bar{\mathbf{k}}^B_1\right)\frac{\Delta t}{ss} \quad (6.34)$$

From Eq. (6.31), we can easily conclude that $\mathbf{F}^B_{k+1/ss} = \bar{\mathbf{F}}^B_{k+1/ss}$. Moreover, the Lagrange multiplier $\Lambda^B_{k+\frac{1}{2ss}}$ is locally second-order accurate, i.e., $\Lambda^B_{k+\frac{1}{2ss}} - \bar{\Lambda}^B_{k+\frac{1}{2ss}} = \mathrm{O}\left(\Delta t^2\right)$. Now, if we compare the solutions $\mathbf{y}^B_{k+1/ss}$ and $\bar{\mathbf{y}}^B_{k+1/ss}$, we can find that:

$$\mathbf{y}^B_{k+1/ss} - \bar{\mathbf{y}}^B_{k+1/ss} = \mathrm{O}\left(\Delta t^3\right) \quad (6.35)$$

Since the unconstrained case is equivalent to solving the uncoupled system, the solution is locally third-order accurate, i.e.,

$$\bar{\mathbf{y}}^B_{k+1/ss} - \mathbf{y}^B\left(t_{k+1/ss}\right) = \mathrm{O}\left(\Delta t^3\right) \quad (6.36)$$

Therefore,

$$\mathbf{y}^B_{k+1/ss} - \mathbf{y}^B\left(t_{k+1/ss}\right) = \mathrm{O}\left(\Delta t^3\right) \quad (6.37)$$

However, the solution $\mathbf{y}^A_{k+1/ss}$ is only locally second-order accurate, which makes the Lagrange multiplier vector locally second-order accurate.

When the local truncation error at the end of the first substep is known, we can deal with an arbitrary substep *j*. The starting condition for Substep *j* is assumed to be locally third-order accurate and the Lagrange multiplier vector is second order accurate, which is in agreement with the end point of the first substep. This starting condition satisfies:

$$\begin{aligned} \mathbf{y}^B_{k+j/ss} - \bar{\mathbf{y}}^B_{k+j/ss} &= \mathrm{O}\left(\Delta t^3\right) \\ \Lambda^B_{k+j/ss} - \bar{\Lambda}^B_{k+j/ss} &= \mathrm{O}\left(\Delta t^2\right) \end{aligned} \quad (6.38)$$

With the aid of the Lipschitz constant, we can conclude that

$$\mathbf{F}^B_{k+j/ss} - \bar{\mathbf{F}}^B_{k+j/ss} = \mathrm{O}\left(\Delta t^3\right) \quad (6.39)$$

The solutions of the middle point given by ”constrained” and ”unconstrained” integrations are expressed separately as

$$\begin{aligned} \mathbf{y}^B_{k+\frac{2j+1}{2ss}} &= \mathbf{y}^B_{k+j/ss} + \frac{1}{2}\left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\mathbf{F}^B_{k+j/ss} + \mathbf{A}_B^{-1}\mathbf{C}_B^T\Lambda^B_{k+j/ss}\right)\frac{\Delta t}{ss} \\ \bar{\mathbf{y}}^B_{k+\frac{2j+1}{2ss}} &= \bar{\mathbf{y}}^B_{k+j/ss} + \frac{1}{2}\left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\bar{\mathbf{F}}^B_{k+j/ss} + \mathbf{A}_B^{-1}\mathbf{C}_B^T\bar{\Lambda}^B_{k+j/ss}\right)\frac{\Delta t}{ss} \end{aligned} \quad (6.40)$$

The difference of them are determined by the terms in the same positions of the two equations above. Comparing them, we have

$$\mathbf{y}^B_{k+\frac{2j+1}{2ss}} - \bar{\mathbf{y}}^B_{k+\frac{2j+1}{2ss}} = \mathrm{O}\left(\Delta t^3\right) \tag{6.41}$$

At the same time, we get:

$$\mathbf{k}^B_1 - \bar{\mathbf{k}}^B_1 = \mathrm{O}\left(\Delta t^3\right) \tag{6.42}$$

Since the LSRT2 algorithm has local second order accuracy at the end of the first stage, we have

$$\bar{\mathbf{y}}^B_{k+\frac{2j+1}{2ss}} - \mathbf{y}^B\left(t_{k+\frac{2j+1}{2ss}}\right) = \mathrm{O}\left(\Delta t^2\right) \tag{6.43}$$

The combination of Eqs. (6.41) and (6.43) yields

$$\mathbf{y}^B_{k+\frac{2j+1}{2ss}} - \mathbf{y}^B\left(t_{k+\frac{2j+1}{2ss}}\right) = \mathrm{O}\left(\Delta t^2\right) \tag{6.44}$$

which indicates a locally second-order accuracy.

Since the solution $\mathbf{y}^A_{k+\frac{2j+1}{2ss}}$ solved by linear interpolation is also locally second-order accurate, we can easily obtain by means of Eq. (5.31) that:

$$\Lambda^B_{k+\frac{2j+1}{2ss}} - \bar{\Lambda}^B_{k+\frac{2j+1}{2ss}} = \mathrm{O}\left(\Delta t^2\right) \tag{6.45}$$

$$\begin{aligned}
\mathbf{y}^B_{k+\frac{j+1}{ss}} &= \mathbf{y}^B_{k+\frac{j}{ss}} + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\mathbf{F}^B_{k+\frac{2j+1}{2ss}} + \mathbf{A}_B^{-1}\mathbf{C}_B^T\Lambda^B_{k+\frac{2j+1}{2ss}} - \gamma\mathbf{J}_B\mathbf{k}^B_1\right)\frac{\Delta t}{ss} \\
\bar{\mathbf{y}}^B_{k+\frac{j+1}{ss}} &= \bar{\mathbf{y}}^B_{k+\frac{j}{ss}} + \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\bar{\mathbf{F}}^B_{k+\frac{2j+1}{2ss}} + \mathbf{A}_B^{-1}\mathbf{C}_B^T\bar{\Lambda}^B_{k+\frac{2j+1}{2ss}} - \gamma\mathbf{J}_B\bar{\mathbf{k}}^B_1\right)\frac{\Delta t}{ss}
\end{aligned} \tag{6.46}$$

By comparing the two equations above, we can obtain

$$\mathbf{y}^B_{k+\frac{2j+1}{ss}} - \bar{\mathbf{y}}^B_{k+\frac{2j+1}{ss}} = \mathrm{O}\left(\Delta t^3\right) \tag{6.47}$$

The local truncation error of LSRT2 algorithm is of $\mathrm{O}\left(\Delta t^3\right)$. This indicates

$$\bar{\mathbf{y}}^B_{k+(j+1)/ss} - \mathbf{y}^B\left(t_{k+(j+1)/ss}\right) = \mathrm{O}\left(\Delta t^3\right) \tag{6.48}$$

Therefore, we can obtain by comparing Eq. (6.47) and (6.48) that the local truncation of an arbitrary substep within the first stage of subdomain A is $\mathrm{O}\left(\Delta t^3\right)$. Moreover, the solution $\mathbf{y}^A_{k+(j+1)/ss}$ solved by a linear interpolation is only locally second-order accurate. By considering Eq. (5.31), we can easily obtain the local truncation error of the

Lagrange multiplier vector $\mathbf{\Lambda}_{k+(j+1)/ss} = \mathrm{O}\left(\Delta t^2\right)$. They satisfy the starting condition as we assumed at the beginning of the arbitrary substep and therefore the proof process can be repeated successively to obtain the local truncation error of the partitioned algorithm at the following substeps with the first stage of subdomain A. With this in mind, we can conclude

$$\tau_{k+1/2}^{\mathbf{y}_B} = \mathrm{O}\left(\Delta t^3\right) \qquad\qquad \tau_{k+1/2}^{\mathbf{\Lambda}} = \mathrm{O}\left(\Delta t^2\right) \tag{6.49}$$

Now, we start to deal with the second stage of subdomain A. The solution at the end can be written for the "constrained" and "unconstrained" integrations

$$\begin{aligned} \mathbf{y}_{k+1}^{A} &= \mathbf{y}_{k}^{A} + \left[\mathbf{I} - \Delta t\gamma\mathbf{J}_A\right]^{-1}\left(\mathbf{A}_A^{-1}\mathbf{F}_{k+1/2}^{A} + \mathbf{A}_A^{-1}\mathbf{C}_A^{T}\Lambda_{k+1/2}^{A} - \gamma\mathbf{J}_A\mathbf{k}_1^{A}\right)\Delta t \\ \bar{\mathbf{y}}_{k+1}^{A} &= \bar{\mathbf{y}}_{k}^{A} + \left[\mathbf{I} - \Delta t\gamma\mathbf{J}_A\right]^{-1}\left(\mathbf{A}_A^{-1}\bar{\mathbf{F}}_{k+1/2}^{A} + \mathbf{A}_A^{-1}\mathbf{C}_A^{T}\bar{\Lambda}_{k+1/2}^{A} - \gamma\mathbf{J}_A\bar{\mathbf{k}}_1^{A}\right)\Delta t \end{aligned} \tag{6.50}$$

Since the differences between two terms in the same positions have been analysed, we can easily obtain the difference of the equations above

$$\mathbf{y}_{k+1}^{A} - \bar{\mathbf{y}}_{k+1}^{A} = \mathrm{O}\left(\Delta t^3\right) \tag{6.51}$$

Since the LSRT2 algorithm is locally third-order accurate, we can conclude

$$\bar{\mathbf{y}}_{k+1}^{A} - \mathbf{y}^{A}\left(t_{k+1}\right) = \mathrm{O}\left(\Delta t^3\right) \tag{6.52}$$

By comparing the two equations above, we can obtain

$$\tau_{k+1}^{\mathbf{y}_A} = \mathbf{y}_{k+1}^{A} - \mathbf{y}^{A}\left(t_{k+1}\right) = \mathrm{O}\left(\Delta t^3\right) \tag{6.53}$$

For the solutions of subdomain B with the second stage of subdomain A, the internal solutions obtained by linear interpolation are also locally second-order accurate; analogously the local truncation error of the Lagrange multiplier vector at the internal point is of $\mathrm{O}\left(\Delta t^2\right)$. Therefore, we can adopt the result of arbitrary substep in that they have the same starting condition. Finally, we can prove

$$\tau_{k+1}^{\mathbf{y}_B} = \mathbf{y}_{k+1}^{B} - \mathbf{y}^{A}\left(t_{k+1}\right) = \mathrm{O}\left(\Delta t^3\right) \tag{6.54}$$

With the aid of (5.31), we can easily obtain the local truncation error of the Lagrange multiplier vector at the end that is $\mathrm{O}\left(\Delta t^3\right)$.

6.3.3 Local truncation error analysis for the GC method

The GC method imposes continuity of velocities at the interface to couple arbitrary Newmark schemes with different time steps in different subdomains. By means of the energy approach, the method is proved to be unconditionally stable as long as all of the individual subdomains satisfy their own stability requirements. However, the method is only globally first-order accurate in subcycling case. In this subsection, we will provide an explanation for the reduction of the accuracy order. Before analysing the case with subcycling, the GC method with a single time step is analysed as follows.

The free solutions of velocity in subdomain A can be expressed as

$$\begin{aligned} \dot{\mathbf{u}}_{k+1}^{A,free} &= \tilde{\dot{\mathbf{u}}}_k^A + \alpha_2^A \ddot{\mathbf{u}}_{k+1}^{A,free} \\ &= \dot{\mathbf{u}}_k^A + \Delta t\left(1-\gamma_A\right)\ddot{\mathbf{u}}_k^A + \alpha_2^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\left(\mathbf{F}_{k+1}^A - \mathbf{K}^A\tilde{\mathbf{u}}_k^A - \mathbf{C}^A\tilde{\dot{\mathbf{u}}}_k^A\right) \end{aligned} \tag{6.55}$$

The second order Taylor series expansion of $\mathbf{F}_{k+1}^A$ can be written in the following form

$$\begin{aligned} \mathbf{F}_{k+1}^A &= \mathbf{F}_k^A + \Delta t\dot{\mathbf{F}}_k^A + \mathrm{O}\left(\Delta t^2\right) \\ &= \mathbf{M}^A\ddot{\mathbf{u}}_k^A + \mathbf{C}^A\dot{\mathbf{u}}_k^A + \mathbf{K}^A\mathbf{u}_k^A - \mathbf{G}_A^T\boldsymbol{\Lambda}_k + \\ &+\Delta t\left(\mathbf{M}^A\dddot{\mathbf{u}}_k^A + \mathbf{C}^A\ddot{\mathbf{u}}_k^A + \mathbf{K}^A\dot{\mathbf{u}}_k^A - \mathbf{G}_A^T\dot{\boldsymbol{\Lambda}}_n\right) + \mathrm{O}\left(\Delta t^2\right) \end{aligned} \tag{6.56}$$

Now, the solution can be rearranged as follows

$$\begin{aligned} \dot{\mathbf{u}}_{k+1}^{A,free} &= \dot{\mathbf{u}}_k^A + \Delta t\left(1-\gamma_A\right)\ddot{\mathbf{u}}_k^A + \\ &+\alpha_2^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\begin{pmatrix} \mathbf{M}^A\ddot{\mathbf{u}}_k^A + \mathbf{C}^A\dot{\mathbf{u}}_k^A + \mathbf{K}^A\mathbf{u}_k^A - \mathbf{G}_A^T\boldsymbol{\Lambda}_k + \\ +\Delta t\left(\mathbf{M}^A\dddot{\mathbf{u}}_k^A + \mathbf{C}^A\ddot{\mathbf{u}}_k^A + \mathbf{K}^A\dot{\mathbf{u}}_k^A - \mathbf{G}_A^T\dot{\boldsymbol{\Lambda}}_k\right) - \\ -\mathbf{K}^A\left(\mathbf{u}_k^A + \Delta t\dot{\mathbf{u}}_k^A + \Delta t^2\left(\frac{1}{2}-\beta_A\right)\ddot{\mathbf{u}}_k^A\right) - \\ -\mathbf{C}^A\left(\dot{\mathbf{u}}_k^A + \Delta t\left(1-\gamma_A\right)\ddot{\mathbf{u}}_k^A\right) + \mathrm{O}\left(\Delta t^2\right) \end{pmatrix} \\ &= \dot{\mathbf{u}}_n^A + \Delta t\ddot{\mathbf{u}}_n^A + \alpha_2^A\Delta t\dddot{\mathbf{u}}_n^A - \alpha_2^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\left(\mathbf{G}_A^T\boldsymbol{\Lambda}_n + \mathbf{G}_A^T\Delta t\dot{\boldsymbol{\Lambda}}_n\right) + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{6.57}$$

In the same manner, the free solution in subdomain B can be written as

$$\dot{\mathbf{u}}_{k+1}^{B,free} = \dot{\mathbf{u}}_k^B + \Delta t\ddot{\mathbf{u}}_k^B + \alpha_2^B\Delta t\dddot{\mathbf{u}}_k^B + \alpha_2^B\left(\tilde{\mathbf{M}}^B\right)^{-1}\left(\mathbf{G}_B^T\boldsymbol{\Lambda}_k - \mathbf{G}_B^T\Delta t\dot{\boldsymbol{\Lambda}}_k\right) + \mathrm{O}\left(\Delta t^3\right) \tag{6.58}$$

The Lagrange multiplier can be computed as follows

$$\begin{aligned}\mathbf{H}\boldsymbol{\Lambda}_{k+1} &= -\left(\mathbf{G}_A\dot{\mathbf{u}}_{k+1}^{A,free} + \mathbf{G}_B\dot{\mathbf{u}}_{n+1}^{B,free}\right)\\ &= -\begin{pmatrix}\mathbf{G}_A\left(\dot{\mathbf{u}}_k^A + \Delta t\ddot{\mathbf{u}}_k^A + \alpha_2^A\Delta t\dddot{\mathbf{u}}_k^A - \alpha_2^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\left(\mathbf{G}_A^T\boldsymbol{\Lambda}_k + \mathbf{G}_A^T\Delta t\dot{\boldsymbol{\Lambda}}_k\right) + \mathrm{O}\left(\Delta t^3\right)\right)\\ +\mathbf{G}_B\left(\dot{\mathbf{u}}_k^B + \Delta t\ddot{\mathbf{u}}_k^B + \alpha_2^B\Delta t\dddot{\mathbf{u}}_k^B - \alpha_2^B\left(\tilde{\mathbf{M}}^B\right)^{-1}\left(\mathbf{G}_B^T\boldsymbol{\Lambda}_k + \mathbf{G}_B^T\Delta t\dot{\boldsymbol{\Lambda}}_k\right) + \mathrm{O}\left(\Delta t^3\right)\right)\end{pmatrix}\end{aligned} \tag{6.59}$$

Since the solutions at t_k is exact, the continuities of all kinematic quantities are satisfied, i.e.,

$$\begin{aligned}&\mathbf{G}_A\dot{\mathbf{u}}_k^A + \mathbf{G}_B\dot{\mathbf{u}}_k^B = 0\\ &\mathbf{G}_A\ddot{\mathbf{u}}_k^A + \mathbf{G}_B\ddot{\mathbf{u}}_k^B = 0\\ &\mathbf{G}_A\dddot{\mathbf{u}}_k^A + \mathbf{G}_B\dddot{\mathbf{u}}_k^B = 0\end{aligned} \tag{6.60}$$

Moreover, $\alpha_2^A = \alpha_2^B$, because the Newmark method is second-order accurate only with $\gamma = 1/2$.

Then, Eq. (6.59) can be simplified into the following form:

$$\mathbf{H}\boldsymbol{\Lambda}_{k+1} = \mathbf{H}\boldsymbol{\Lambda}_k + \Delta t\mathbf{H}\dot{\boldsymbol{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right) \tag{6.61}$$

In the matrix $\mathbf{H} = \alpha_2^A\mathbf{G}_A\left(\tilde{\mathbf{M}}^A\right)^{-1}\mathbf{G}_A^T + \alpha_2^B\mathbf{G}_B\left(\tilde{\mathbf{M}}^B\right)^{-1}\mathbf{G}_B^T$, there is a Δt included in the terms α_2^A and α_2^B. Thus, Eq. (6.61) can be transformed into

$$\boldsymbol{\Lambda}_{k+1} = \boldsymbol{\Lambda}_k + \Delta t\dot{\boldsymbol{\Lambda}}_k + \mathrm{O}\left(\Delta t^2\right) \tag{6.62}$$

Similarly as the ASSR2 method, if the Lagrange multiplier vector used is exact (as the ”constrained” integration), the solution obtained preserves the accuracy of the Newmark method which is second order accurate. Differently from the ”constrained” integration, the GC method with $ss = 1$ is advanced with the Lagrange multiplier vector Λ of $\mathrm{O}\left(\Delta t^2\right)$. Therefore, its the local truncation error can be obtained by comparing the GC method and the ”constrained” integration:

$$\begin{aligned}\tau_{n+1}^{\mathbf{v}_A} &= \mathrm{O}\left(\Delta t^3\right) + \alpha_2^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^3\right)\\ \tau_{n+1}^{d_A} &= \mathrm{O}\left(\Delta t^3\right) + \alpha_1^A\left(\tilde{\mathbf{M}}^A\right)^{-1}\mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^3\right)\\ \tau_{n+1}^{\mathbf{v}_B} &= \mathrm{O}\left(\Delta t^3\right) + \alpha_2^B\left(\tilde{\mathbf{M}}^B\right)^{-1}\mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^3\right)\\ \tau_{n+1}^{d_B} &= \mathrm{O}\left(\Delta t^3\right) + \alpha_1^B\left(\tilde{\mathbf{M}}^B\right)^{-1}\mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^3\right)\end{aligned} \tag{6.63}$$

From this analysis, we can conclude that the case without subcycling can preserve locally third-order accurate in that the method can ensure locally second-order accurate

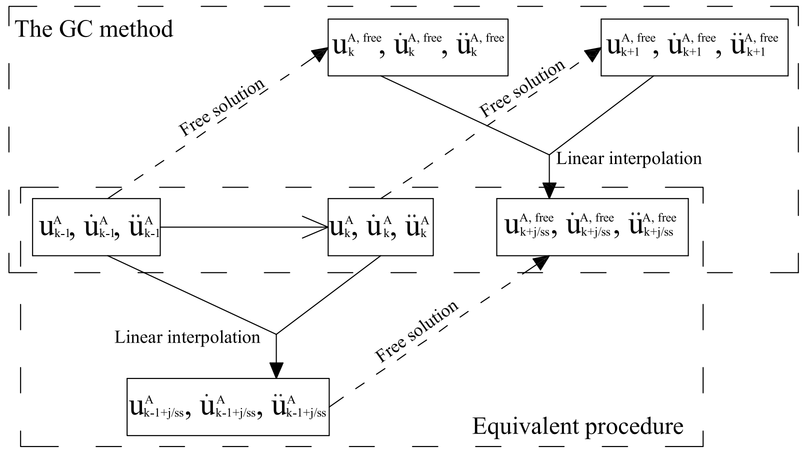

The GC method
$\mathbf{u}_k^{A,\,\text{free}}, \dot{\mathbf{u}}_k^{A,\,\text{free}}, \ddot{\mathbf{u}}_k^{A,\,\text{free}}$
$\mathbf{u}_{k+1}^{A,\,\text{free}}, \dot{\mathbf{u}}_{k+1}^{A,\,\text{free}}, \ddot{\mathbf{u}}_{k+1}^{A,\,\text{free}}$
Free solution
Free solution
Linear interpolation
$\mathbf{u}_{k-1}^{A}, \dot{\mathbf{u}}_{k-1}^{A}, \ddot{\mathbf{u}}_{k-1}^{A}$
$\mathbf{u}_{k}^{A}, \dot{\mathbf{u}}_{k}^{A}, \ddot{\mathbf{u}}_{k}^{A}$
$\mathbf{u}_{k+j/ss}^{A,\,\text{free}}, \dot{\mathbf{u}}_{k+j/ss}^{A,\,\text{free}}, \ddot{\mathbf{u}}_{k+j/ss}^{A,\,\text{free}}$
Linear interpolation
Free solution
$\mathbf{u}_{k-1+j/ss}^{A}, \dot{\mathbf{u}}_{k-1+j/ss}^{A}, \ddot{\mathbf{u}}_{k-1+j/ss}^{A}$
Equivalent procedure

Fig. 6.3: Representation of the linear interpolation in the GC method and its equivalent solution procedure

Lagrange multiplier vector. In the subcycling case, in order to use this strategy, we present a equivalent procedure to simplify the solution procedure of the GC method.

As we know, both Newmark integration method and linear interpolation are actually the linear combination of basic quantities. If one has a deep insight of the GC method, one can find a procedure which is equivalent to free solutions of subdomain A and linear interpolations as shown in Fig. 6.3.

One loop of the GC method spans two steps. Therefore, we assume that the solutions and the Lagrange multiplier vectors at t_{k-1} and t_k are exact. Here, we use the point t_k as the basis of the analysis, and the differential vectors and the Lagrange multiplier vectors are written in Taylor's series expansion based on the point t_k. Let us consider the equivalent procedure and focus on the jth substep. In the equivalent procedure, the solution in Subdomain A at $t_{k-1+\frac{j}{ss}}$ is obtained by linear interpolation, and therefore it is only second order accurate, i.e.,

$$\begin{aligned}
\dot{\mathbf{u}}^A_{k-1+\frac{j}{ss}} &= \dot{\mathbf{u}}^A_k - \frac{ss-j}{ss}\Delta t\ddot{\mathbf{u}}^A_k + \mathrm{O}\left(\Delta t^2\right)\\
\ddot{\mathbf{u}}^A_{k-1+\frac{j}{ss}} &= \ddot{\mathbf{u}}^A_k - \frac{ss-j}{ss}\Delta t\dddot{\mathbf{u}}^A_k + \mathrm{O}\left(\Delta t^2\right)
\end{aligned} \tag{6.64}$$

In Subdomain B, we suppose that the solution at the beginning of Substep j is locally second order accurate, Therefore, we have

$$\begin{aligned}\dot{\mathbf{u}}^B_{k+\frac{j-1}{ss}} &= \dot{\mathbf{u}}^B_k + \frac{j-1}{ss}\Delta t\ddot{\mathbf{u}}^B_k + \mathrm{O}\left(\Delta t^2\right)\\ \ddot{\mathbf{u}}^B_{k+\frac{j-1}{ss}} &= \ddot{\mathbf{u}}^A_k + \frac{j-1}{ss}\Delta t\dddot{\mathbf{u}}^B_k + \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{6.65}$$

The integration of Substep *j* related in the equivalent procedure includes advance of Subdomain *A* from $t_{k-1+\frac{j}{ss}}$ to $t_{k+\frac{j}{ss}}$ and advance of Subdomain *B* from $t_{k+\frac{j-1}{ss}}$ to $t_{k+\frac{j}{ss}}$. Similarly as the case with $ss = 1$, we can get:

$$\mathbf{H}\boldsymbol{\Lambda}_{k+j/ss} = \mathbf{H}\boldsymbol{\Lambda}_k + \frac{j}{ss}\Delta t\mathbf{H}\dot{\boldsymbol{\Lambda}}_k + \mathrm{O}\left(\Delta t^2\right) \tag{6.66}$$

Therefore, we can find that the Lagrange multiplier vector has one order lower than the case with $ss = 1$

$$\boldsymbol{\Lambda}_{k+j/ss} = \boldsymbol{\Lambda}_k + \mathrm{O}\left(\Delta t\right) \tag{6.67}$$

From Eqs. (6.63), we can easily conclude that the solutions in the subcycling case are only locally second-order accurate. From another viewpoint, the matrix **H** contains one Δt which makes the Lagrange multiplier vector first order accurate. This is the cause of the order reduction in the subcycling case.

6.4 Numerical analysis and simulations of a Single-DoF split mass system

Due to the complexity of the subcycling cases, both the stability analysis and convergence analyses are investigated through numerical experiments of a Single-DoF split-mass system. Moreover, some results from theoretical analyses for the cases with the same time step are validated in this section.

6.4.1 The Single-DoF split mass system

In order to test the performances of the partitioned methods we consider a test problem where a Single-DoF mass-spring system is split into two Single-DoF mass-spring

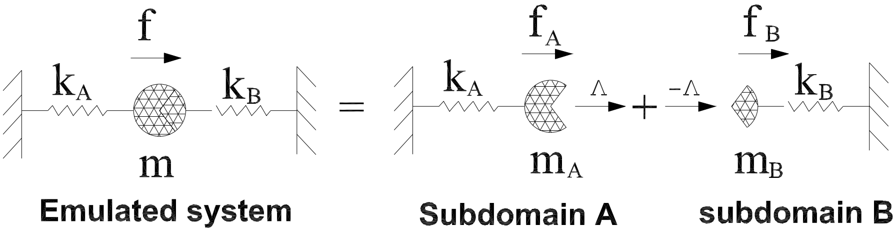


Fig. 6.4: A Single-DoF split mass system

systems A and B linked by an interface reaction force Λ as shown in Fig. 6.4. This test problem, the Single-DoF split-mass system, were widely used to validate partitioned methods (Bonelli et al., 2008a; Prakash, 2007). For a partitioned algorithm, the key is to preserve the performance of the overall system. For simplicity, we choose the following system variables $m = 1$ and $k = 1$ and introduce a parameter b_1 to partition the mass and the spring

$$b_1 = \frac{m_A}{m_B} = \frac{k_B}{k_A} \tag{6.68}$$

The external force is defined with different purposes. For stability analysis, we assume no external force in that it does not affect the stability condition. For convergence analysis, we adopt a sinusoidal external force $f_e = sin(2t)$ imposed on Subdomain B.

6.4.2 Spectral stability analysis

Although the cases with no subcycling have been proved to be unconditionally stable, the result can not be extended to the subcycling cases, because interpolation may introduce instability. In this section, the spectral analysis approach is performed by assuming no external force. By means of the spectral analysis, on the one hand, the stability condition can be qualitatively ensured and on the other hand, the high frequency dissipation and its effect on the low frequency mode can be quantitatively observed.

Here the split mass system is assumed to be linear, with no dissipation and no external force. The application of the proposed methods leads to the general recursive formula:

$$\mathbf{y}_{k+1} = \mathbf{R}\mathbf{y}_k \tag{6.69}$$

where **y** is a state vector, and **R** an amplification matrix. For the staggered methods, one can find that both subdomain *A* and *B* advanced from the same point t_k. Therefore, the starting condition of every loop is minimal and only contains the differential variables. Let λ_i denote the eigenvalues of **R**. The modulus of λ_i is written as $\mid \lambda_i \mid$. The stability condition requires that all of the eigenvalues λ_i must satisfy $\mid \lambda_i \mid \leq 1$.

Stability function of a partitioned method, differently from ODE integration, is not only dependent on system characteristics but also on the parameters *ss* and b_1. For this reason, 14 scenarios are intentionally investigated, as shown in Table 6.1. Note that the sign $\mp$ denotes that subdomain *A* is integrated with $\gamma = 1 - \sqrt{2}/2$ and sub-

Table 6.1: Spectral stability analysis of the LSRT-based partitioned methods on a Single-DoF split-mass system

Case	Integrator	ss	γ	Results
(1)	LSRT1	1	1	stable (6.5)
(2)	LSRT1	2	1	stable (6.6)
(3)	LSRT1	10	1	stable (6.7)
(4)	LSRT2	1	$1\text{-}\frac{\sqrt{2}}{2}$	stable (6.8)
(5)	LSRT2	1	$1+\frac{\sqrt{2}}{2}$	stable (6.9)
(6)	LSRT2	1	$1\mp\frac{\sqrt{2}}{2}$	stable (6.10)
(7)	LSRT2	1	$1\pm\frac{\sqrt{2}}{2}$	stable (6.11)
(8)	LSRT2	2	$1\text{-}\frac{\sqrt{2}}{2}$	stable (6.12)
(9)	LSRT2	2	$1+\frac{\sqrt{2}}{2}$	stable (6.13)
(10)	LSRT2	2	$1\mp\frac{\sqrt{2}}{2}$	stable (6.14)
(11)	LSRT2	2	$1\pm\frac{\sqrt{2}}{2}$	stable (6.15)
(12)	LSRT2	10	$1\text{-}\frac{\sqrt{2}}{2}$	sometime unstable (6.16)
(13)	LSRT2	10	$1+\frac{\sqrt{2}}{2}$	stable (6.17)
(14)	LSRT2	10	$1\mp\frac{\sqrt{2}}{2}$	stable (6.18)
(15)	LSRT2	10	$1\pm\frac{\sqrt{2}}{2}$	stable (6.19)

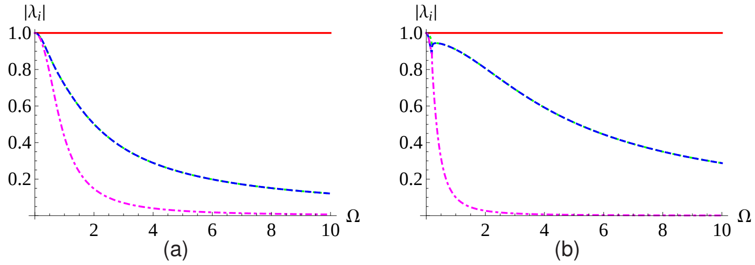


Fig. 6.5: $|\lambda_i|$ for the partitioned method integrated with LSRT1 and $ss = 1$: (a) $b_1 = 2$ and (b) $b_1 = 10$

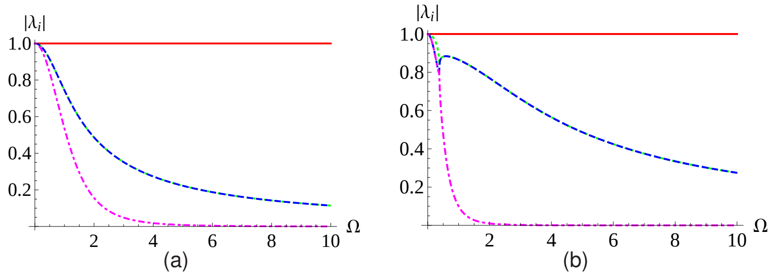


Fig. 6.6: $|\lambda_i|$ for the partitioned method integrated with LSRT1 and $ss = 2$: (a) $b_1 = 2$ and (b) $b_1 = 10$

domain B with $\gamma = 1 + \sqrt{2}/2$ and vice versa. For the cases with $ss = 1$ and $ss = 2$, two choices of b_1 are used: 2 and 10; for the cases with $ss = 10$, 4 choices of b_1 are used: 0.1, 0.5, 2 and 10. With regard to the Single-DoF test problem, the absolute values of the eigenvalues of the amplification matrices are plotted in Figs. (6.5)-(6.19). The number of eigenvalues is 4, but only two of them -complex conjugate pair- are the principal eigenvalues, whereas the others are spurious. One of the spurious eigenvalues is unit and is resulted from the acceleration constraint which will be detailed investigated in the Subsection 6.4.4. From a sufficient number of spectral radius plots, we can conclude: i) when $ss = 1$, all partitioned methods are stable, which is in agreement with the symbolic stability analysis performed in Section 5.5; ii) in the presence of subcycling, the ASSR1 method is always stable; iii) in the cases of subcycling, the ASSR2 method with $\gamma = 1 - \sqrt{2}/2$ are sometimes conditionally stable and with $\gamma = 1 + \sqrt{2}/2$ appears to be stable within all cases considered.

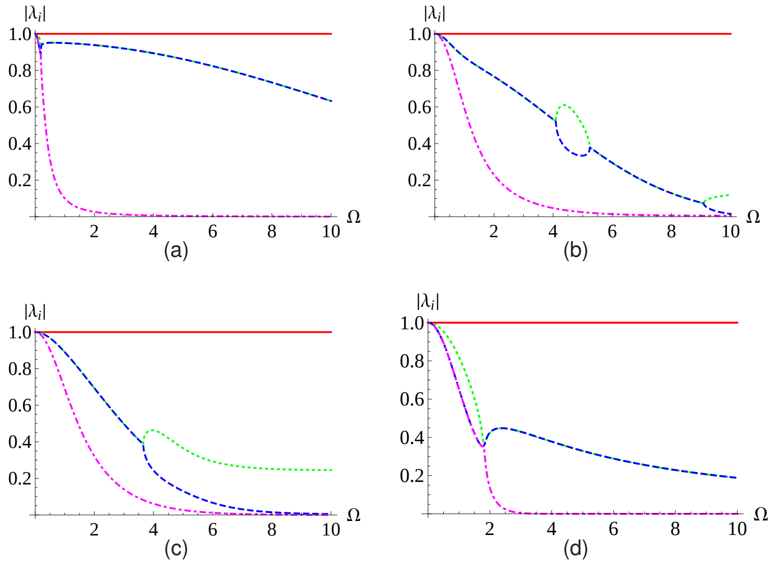


Fig. 6.7: | λ_i | for the partitioned method integrated with LSRT1 and ss = 10: (a) b_1 = 0.1, (b) b_1 = 1, (c) b_1 = 2, and (d) b_1 = 10

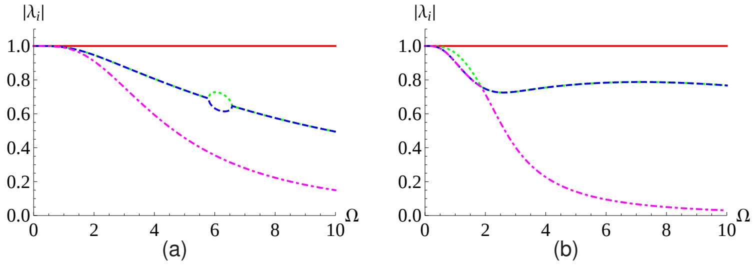


Fig. 6.8: | λ_i | for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 - \frac{\sqrt{2}}{2}$ and ss = 1: (a) b_1 = 2 and (b) b_1 = 10

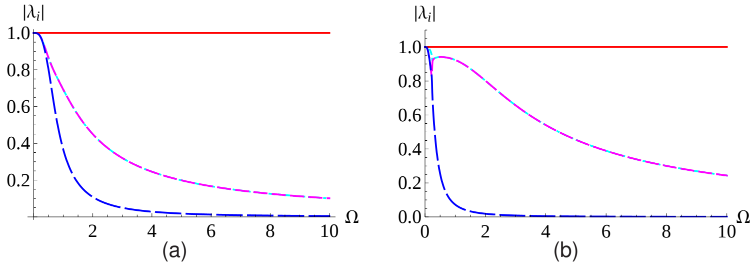


Fig. 6.9: | λ_i | for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 + \frac{\sqrt{2}}{2}$ and ss = 1: (a) b_1 = 2 and (b) b_1 = 10

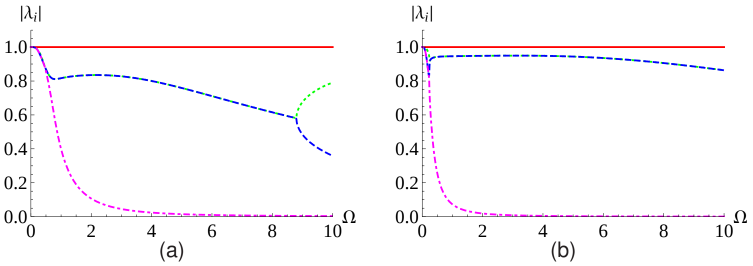


Fig. 6.10: $|\lambda_i|$ for the partitioned method integrated by LSRT2 with $\gamma_A = 1 - \frac{\sqrt{2}}{2}$, $\gamma_B = 1 + \frac{\sqrt{2}}{2}$ and $ss = 1$: (a) $b_1 = 2$ and (b) $b_1 = 10$

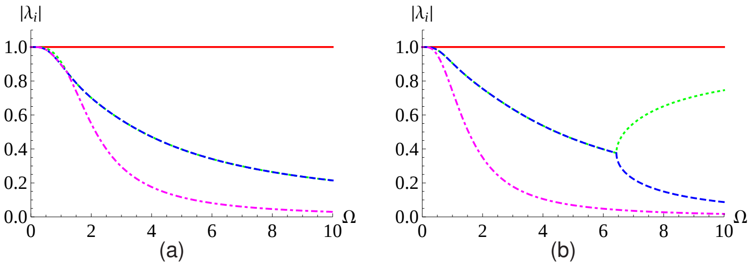


Fig. 6.11: $|\lambda_i|$ for the partitioned method integrated by LSRT2 with $\gamma_A = 1 + \frac{\sqrt{2}}{2}$, $\gamma_B = 1 - \frac{\sqrt{2}}{2}$ and $ss = 1$: (a) $b_1 = 2$ and (b) $b_1 = 10$

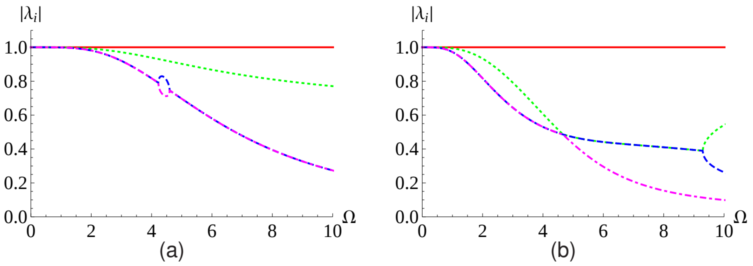


Fig. 6.12: $|\lambda_i|$ for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 - \frac{\sqrt{2}}{2}$ and $ss = 2$: (a) $b_1 = 2$ and (b) $b_1 = 10$

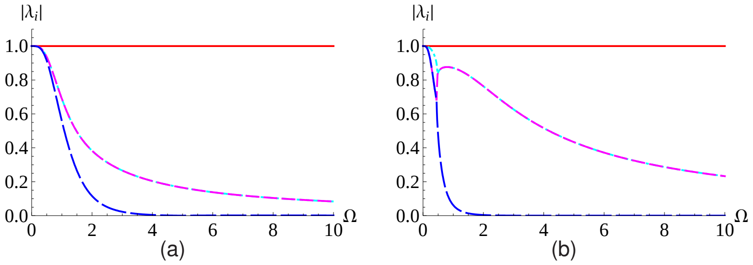


Fig. 6.13: $\mid \lambda_i \mid$ for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 + \frac{\sqrt{2}}{2}$ and $ss = 2$: (a) $b_1 = 2$ and (b) $b_1 = 10$

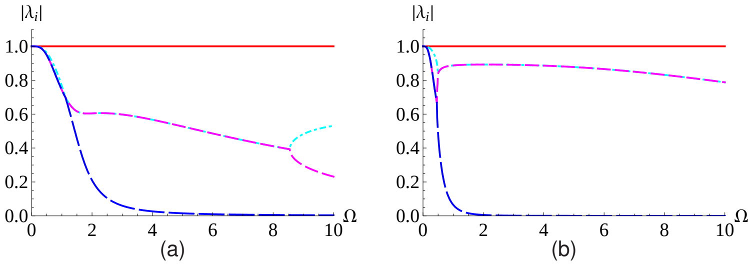


Fig. 6.14: $\mid \lambda_i \mid$ for the partitioned method integrated by LSRT2 with $\gamma_A = 1 - \frac{\sqrt{2}}{2}, \gamma_B = 1 + \frac{\sqrt{2}}{2}$ and $ss = 2$: (a) $b_1 = 2$ and (b) $b_1 = 10$

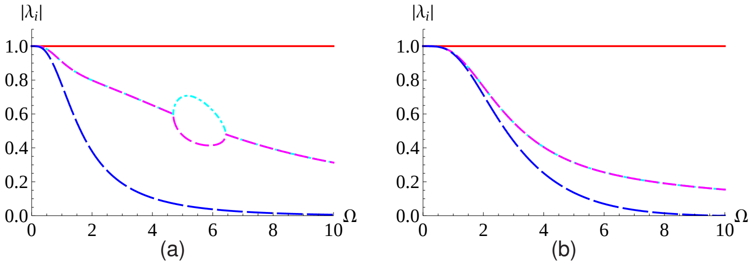


Fig. 6.15: $\mid \lambda_i \mid$ for the partitioned method integrated by LSRT2 with $\gamma_A = 1 + \frac{\sqrt{2}}{2}, \gamma_B = 1 - \frac{\sqrt{2}}{2}$ and $ss = 2$: (a) $b_1 = 2$ and (b) $b_1 = 10$

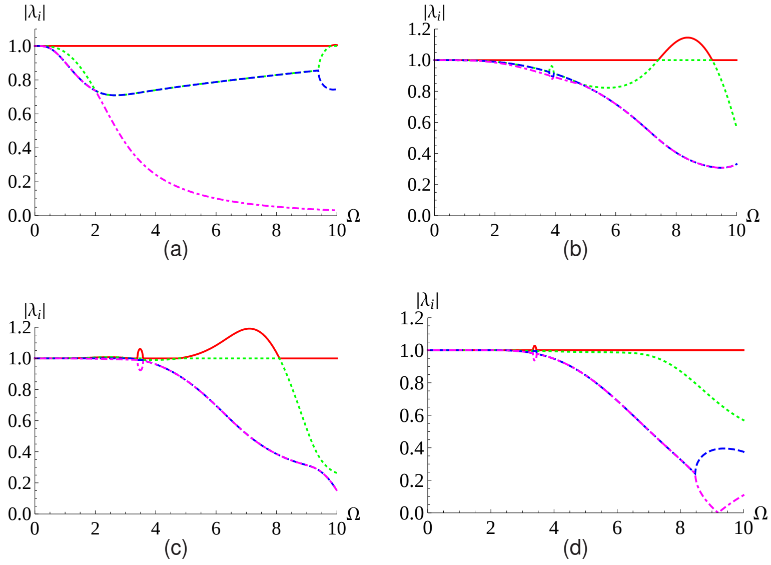


Fig. 6.16: $\mid \lambda_i \mid$ for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 - \frac{\sqrt{2}}{2}$ and ss = 10: (a) b_1 = 0.1, (b) b_1 = 1, (c) b_1 = 2, and (d) b_1 = 10

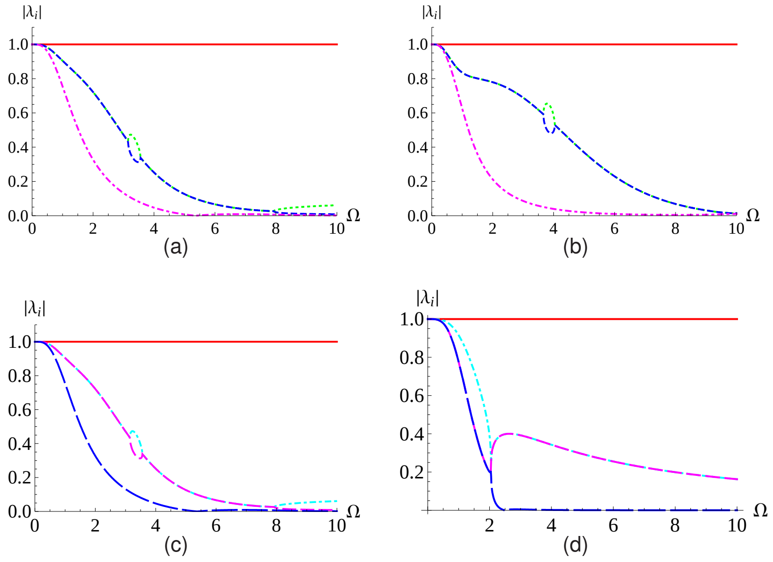


Fig. 6.17: $\mid \lambda_i \mid$ for the partitioned method integrated by LSRT2 with $\gamma_A = \gamma_B = 1 + \frac{\sqrt{2}}{2}$ and ss = 10: (a) b_1 = 0.1, (b) b_1 = 1, (c) b_1 = 2, and (d) b_1 = 10

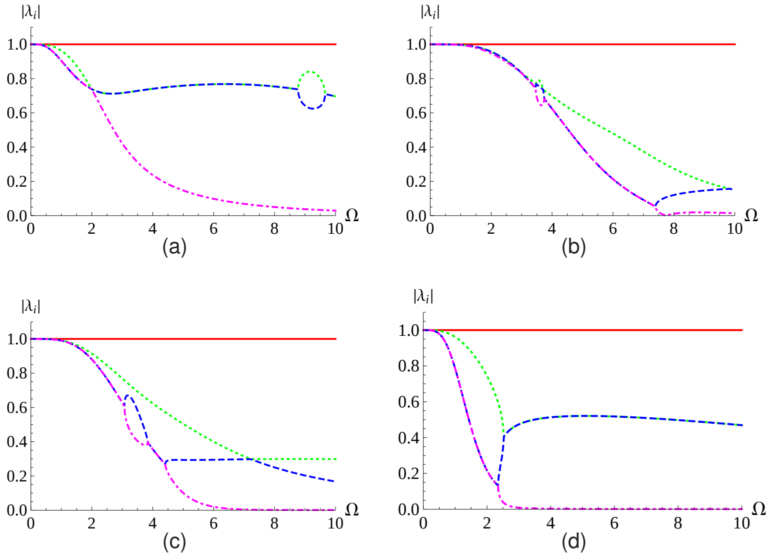


Fig. 6.18: | λ_i | for the partitioned method integrated by LSRT2 with $\gamma_A = 1 - \frac{\sqrt{2}}{2}, \gamma_B = 1 + \frac{\sqrt{2}}{2}$ and $ss = 10$: (a) $b_1 = 0.1$, (b) $b_1 = 1$, (c) $b_1 = 2$, and (d) $b_1 = 10$

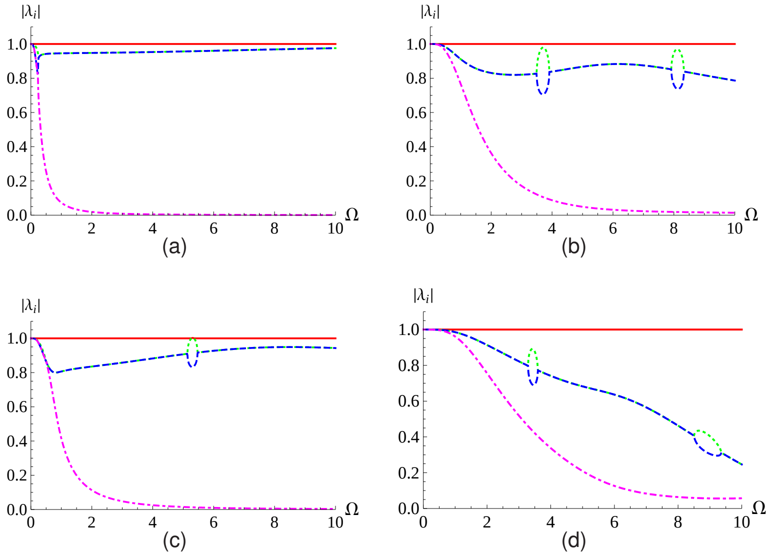


Fig. 6.19: | λ_i | for the partitioned method integrated by LSRT2 with $\gamma_A = 1 + \frac{\sqrt{2}}{2}, \gamma_B = 1 - \frac{\sqrt{2}}{2}$ and $ss = 10$: (a) $b_1 = 0.1$, (b) $b_1 = 1$, (c) $b_1 = 2$, and (d) $b_1 = 10$

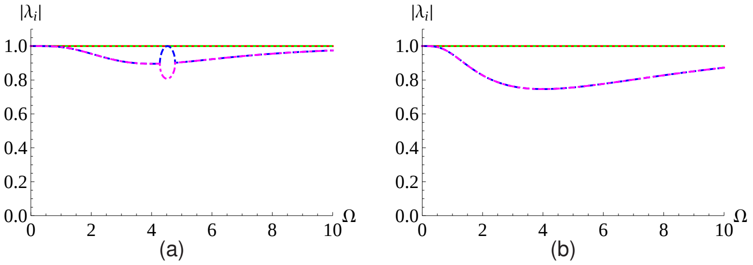


Fig. 6.20: $\mid \lambda_i \mid$ for the partitioned method integrated by two-stage Rosenbrock method with $\gamma = 1/4$ and $ss = 1$: (a) $b_1 = 2$ and (b) $b_1 = 10$

For the sake of generality, the two-stage Rosenbrock method with no numerical damping, i.e., $\gamma = 1/2$ or $\gamma = 1/4$, is also considered in the partitioned method. Herein, we consider four cases which are listed in Table 6.2. Figs. (6.20)-(6.23) show the results of the spectral analysis for the partitioned method with the conservative Rosenbrock method. For the cases with $\gamma = 1/2$, one of the spurious eigenvalues is unitary and the other one is real whose absolute values is less than one. From the stability plots, we can observe that all the cases presented are always stable. Nevertheless, both spurious eigenvalues for the cases with $\gamma = 1/4$ are unit. In these cases, the multiplicity of the unitary eigenvalue can introduce week instability which will be specifically analysed in the subsection.

Table 6.2: Spectral stability analysis of the partitioned method based on two-stage Rosenbrock method with $\gamma = 1/2$ or $\gamma = 1/4$, on a Single-DoF split-mass system

scenario	Integrator	ss	γ	Results
(1)	LSRT2	1	1/4	stable (6.20)
(1)	LSRT2	1	1/2	stable (6.21)
(3)	LSRT2	10	1/4	sometime unstable (6.22)
(4)	LSRT2	10	1/2	stable (6.23)

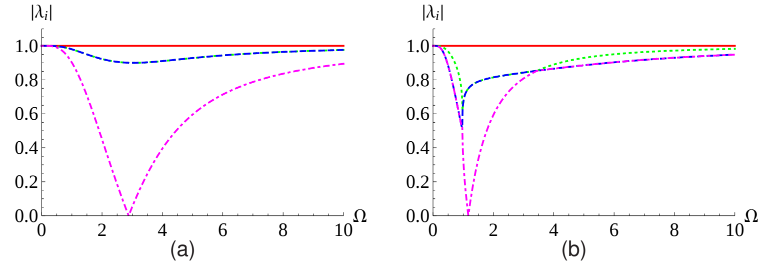


Fig. 6.21: $|\lambda_i|$ for the partitioned method integrated by two-stage Rosenbrock method with $\gamma = 1/2$ and $ss = 1$: (a) $b_1 = 2$ and (b) $b_1 = 10$

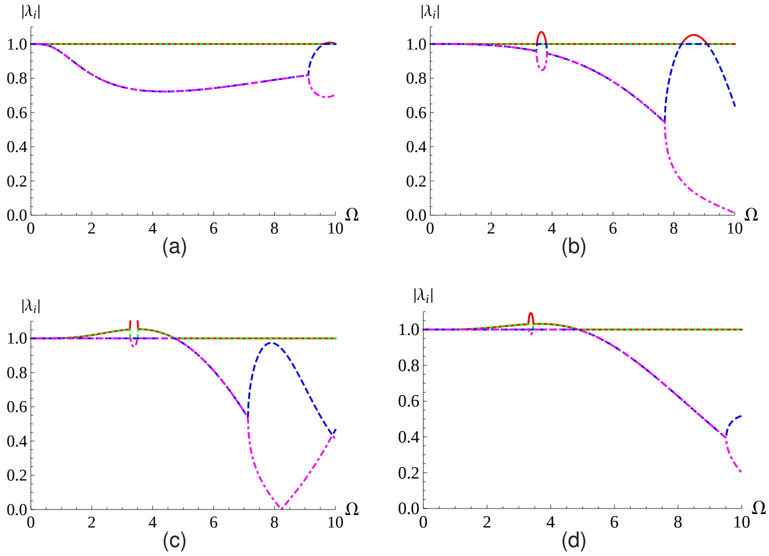


Fig. 6.22: $|\lambda_i|$ for the partitioned method integrated by two-stage Rosenbrock method with $\gamma = 1/4$ and $ss = 10$: (a) $b_1 = 0.1$, (b) $b_1 = 1$, (c) $b_1 = 2$, and (d) $b_1 = 10$

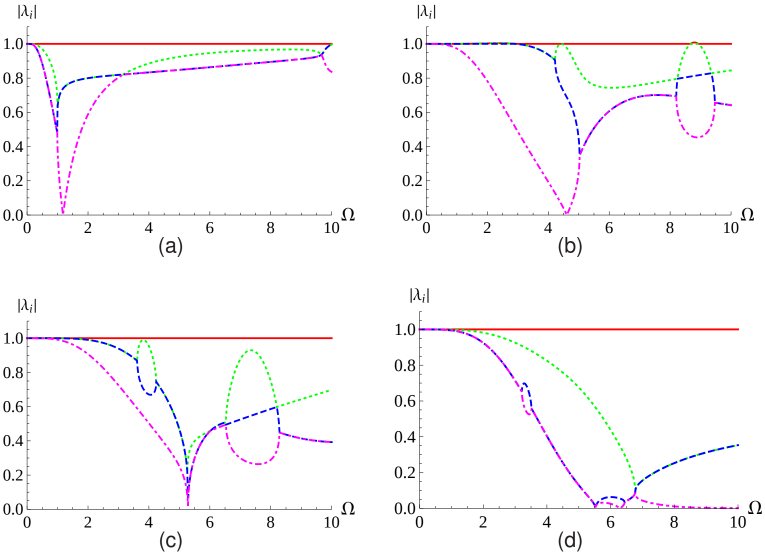


Fig. 6.23: $|\ \lambda_i\ |$ for the partitioned method integrated by two-stage Rosenbrock method with $\gamma = 1/2$ and $ss = 10$: (a) $b_1 = 0.1$, (b) $b_1 = 1$, (c) $b_1 = 2$, and (d) $b_1 = 10$

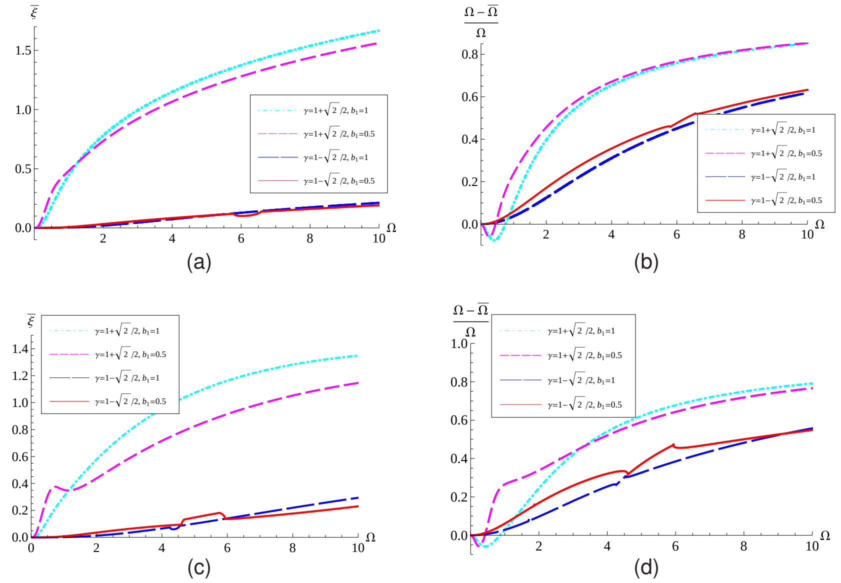


Fig. 6.24: Numerical damping ratio and relative frequency error for the staggered procedure: (a), (b) ss=1; (c), (d) ss=10.

6.4.3 Investigation of the principal eigenvalues

Here, let's consider a Single-DoF split-mass system integrated by the LSRT2-based partitioned methods. The principal complex conjugate eigenvalues of the amplification matrix can be expressed in the following form (Hughes, 1983):

$$\lambda_{1,2} = C + \mathrm{i}D = e^{-\bar{\xi}\bar{\omega}\Delta t_A \pm \mathrm{i}\bar{\omega}\Delta t_A} \tag{6.70}$$

Hence, the damped numerical frequency $\bar{\Omega}$ and the numerical damping ratio $\bar{\xi}$ read

$$\bar{\omega}\Delta t = \bar{\Omega} = \arctan\left(\frac{D}{C}\right) \tag{6.71}$$

$$\bar{\xi} = -\frac{\ln\left(C^2 + D^2\right)}{2\bar{\Omega}} \tag{6.72}$$

Both numerical damping and the frequency error are depicted in Fig. 6.24. The favourable properties of the proposed partitioned algorithm in terms of high-frequency dissipation and limited frequency error are evident.

6.4.4 Investigation of spurious eigenvalues

This subsection deals with the cause of spurious eigenvalues and their effects on the accuracy of partitioned methods. For an ODE system, numerous time integrators has been developed based on the Newmark methods, exhibiting second-order accuracy and numerical dissipation. Those methods introduce a so-called spurious root that enables high-frequency dissipation to be achieved without sacrificing low-frequency accuracy. In that case, the spurious root must be small enough to guarantee a good approximation for the low-frequency modes (Hoff and Pahl, 1988).

In order to investigate the existence of the unit eigenvalue whatever the time step is used in the partitioned integration, we consider the partitioned methods based on LSRT2 integrator. The governing equations of motion for the Single-DoF test problem can be written as:

$$\begin{cases} m_A \ddot{u}_A + k_A u_A = \Lambda \\ m_B \ddot{u}_B + k_B u_B = -\Lambda \\ \ddot{u}_A = \ddot{u}_B \end{cases} \tag{6.73}$$

By means of (5.7), the Lagrange multiplier can be expressed as

$$\Lambda = \frac{m_B k_A u_A - m_A k_B u_B}{m_A + m_B} \tag{6.74}$$

The corresponding amplification matrix can be expressed as

$$\begin{Bmatrix} u_A^{k+1} \\ \dot{u}_A^{k+1} \\ u_B^{k+1} \\ \dot{u}_B^{k+1} \end{Bmatrix} = \mathbf{R} \begin{Bmatrix} u_A^{k} \\ \dot{u}_A^{k} \\ u_B^{k} \\ \dot{u}_B^{k} \end{Bmatrix} \tag{6.75}$$

If the matrix **R** has an eigenvalue which is unitary, this indicates that there exists an eigenvector ξ which satisfies the following relationship:

$$\mathbf{R}\xi = \xi \tag{6.76}$$

From Eqs. (6.73) and (6.74), we can easily obtain that $\ddot{u}_A = 0, \ddot{u}_B = 0$, if we suppose $u_A = k_B, u_B = -k_A$.

Now, we can assume that $\xi = \left\{ \begin{array}{cccc} u_A^k & \dot{u}_A^k & u_B^k & \dot{u}_B^k \end{array} \right\} = \left\{ \begin{array}{cccc} k_B & 0 & -k_A & 0 \end{array} \right\}^T$ We can define this point as the *equilibrium point* for the constrained system for which the derivative of the vector with respect to time is equal to zero. In order to prove that this vector is one eigenvector of the matrix **R**, we advance one loop of solution from the this vector. For the case without subcycling, this process can be listed in detail as follows.

1. In the first stage, we can obtain the Lagrange multiplier at the beginning of the time step:

$$\Lambda_k = \frac{m_B k_A u_A^k - m_A k_B u_B^k}{m_A + m_B} = \frac{m_B k_A k_B + m_A k_B k_A}{m_A + m_B} = k_B k_A \tag{6.77}$$

In subdomain A, we can obtain:

$$f_A^k = \left\{ \begin{array}{c} \dot{u}_A^k \\ \ddot{u}_A^k \end{array} \right\} = \begin{bmatrix} 1 & 0 \\ 0 & m_A \end{bmatrix}^{-1} \left(\begin{bmatrix} 0 & 1 \\ k_A & 0 \end{bmatrix} \left\{ \begin{array}{c} u_A^k \\ \dot{u}_A^k \end{array} \right\} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Lambda \right) = 0 \tag{6.78}$$

Thus,

$$k_1^A = [I - \Delta t \gamma J_A]^{-1} f_A^k \Delta t = 0 \tag{6.79}$$

Accordingly, the solution at the middle of the time step in subdomain *A* can be expressed as:

$$\left\{ \begin{array}{c} u_A^{k+1/2} \\ \dot{u}_A^{k+1/2} \end{array} \right\} = \left\{ \begin{array}{c} u_A^k \\ \dot{u}_A^k \end{array} \right\} + \frac{1}{2} k_1^A = \left\{ \begin{array}{c} u_A^k \\ \dot{u}_A^k \end{array} \right\} = \left\{ \begin{array}{c} k_B \\ 0 \end{array} \right\} \tag{6.80}$$

In the same manner, we can obtain the solution at middle of the time step in subdomain *B*:

$$\left\{ \begin{array}{c} u_B^{k+1/2} \\ \dot{u}_B^{k+1/2} \end{array} \right\} = \left\{ \begin{array}{c} u_B^k \\ \dot{u}_B^k \end{array} \right\} + \frac{1}{2} k_1^B = \left\{ \begin{array}{c} u_B^k \\ \dot{u}_B^k \end{array} \right\} = \left\{ \begin{array}{c} -k_A \\ 0 \end{array} \right\} \tag{6.81}$$

2. In the second stage, we can compute the Lagrange multiplier at the middle of the time step:

$$\Lambda_{k+1/2} = \frac{m_B k_A u_A^{k+1/2} - m_A k_B u_B^{k+1/2}}{m_A + m_B} = \frac{m_B k_A k_B + m_A k_B k_A}{m_A + m_B} = k_B k_A \tag{6.82}$$

Likewise, we can obtain:

$$f_A^{k+1/2} = \left\{ \begin{array}{c} \dot{u}_A^{k+1/2} \\ \ddot{u}_A^{k+1/2} \end{array} \right\} = \begin{bmatrix} 1 & 0 \\ 0 & m_A \end{bmatrix}^{-1} \left(\begin{bmatrix} 0 & 1 \\ k_A & 0 \end{bmatrix} \left\{ \begin{array}{c} u_A^{k+1/2} \\ \dot{u}_A^{k+1/2} \end{array} \right\} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Lambda \right) = 0$$

(6.83)

$$k_2^A = [I - \Delta t\gamma J_A]^{-1}\left(f_A^k\Delta t - \Delta t\gamma J_A k_1^A\right) = 0 \tag{6.84}$$

Thus, the solution at the end of the time step in subdomain A can be expressed as:

$$\left\{\begin{array}{c} u_A^{k+1} \\ \dot{u}_A^{k+1} \end{array}\right\} = \left\{\begin{array}{c} u_A^{k} \\ \dot{u}_A^{k} \end{array}\right\} + k_2^A = \left\{\begin{array}{c} k_B \\ 0 \end{array}\right\} \tag{6.85}$$

Similarly, the solution at the end of the time step in subdomain B can be computed as:

$$\left\{\begin{array}{c} u_B^{k+1} \\ \dot{u}_B^{k+1} \end{array}\right\} = \left\{\begin{array}{c} -k_A \\ 0 \end{array}\right\} \tag{6.86}$$

3. Finally, we can obtain:

$$\left\{\begin{array}{cccc} u_A^{k+1} & \dot{u}_A^{k+1} & u_B^{k+1} & \dot{u}_B^{k+1} \end{array}\right\}^T = \left\{\begin{array}{cccc} u_A^{k} & \dot{u}_A^{k} & u_B^{k} & \dot{u}_B^{k} \end{array}\right\}^T \tag{6.87}$$

Thus, one of the eigenvalues of the amplification matrix is 1, and the corresponding eigenvector is $\left\{\begin{array}{cccc} k_B & 0 & -k_A & 0 \end{array}\right\}^T$.

In the case with subcycling, we can similarly prove the existence of the unitary eigenvalue and its corresponding eigenvector . The proof can be simplified as follows.

In the first stage of subdomain A, the solution is the same as the case with subcycling.

$$\left\{\begin{array}{c} u_A^{k+1/2} \\ \dot{u}_A^{k+1/2} \end{array}\right\} = \left\{\begin{array}{c} k_B \\ 0 \end{array}\right\} \tag{6.88}$$

Since the solutions at the middle of the time step are the same as the solutions at the beginning, the internal solutions solved by a linear interpolation are always equal to the solutions at the beginning, i.e.

$$\left\{\begin{array}{c} u_A^{k+j/2ss} \\ \dot{u}_A^{k+j/2ss} \end{array}\right\} = \left\{\begin{array}{c} k_B \\ 0 \end{array}\right\} \tag{6.89}$$

In the first stage of the substep in subdomain B, we can obtain

$$\left\{\begin{array}{c} u_B^{k+1/2ss} \\ \dot{u}_B^{k+1/2ss} \end{array}\right\} = \left\{\begin{array}{c} u_B^{k} \\ \dot{u}_B^{k} \end{array}\right\} + \frac{1}{2}k_1^B = \left\{\begin{array}{c} u_B^{k} \\ \dot{u}_B^{k} \end{array}\right\} = \left\{\begin{array}{c} -k_A \\ 0 \end{array}\right\} \tag{6.90}$$

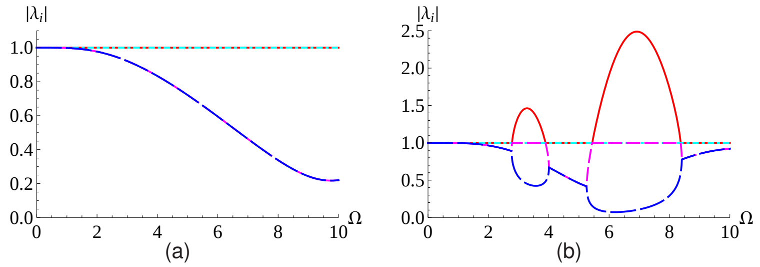


Fig. 6.25: $\mid \lambda_i \mid$ for the partitioned method using dual Lagrange multipliers and integrated with LSRT2 and $ss = 10$: (a) $b_1 = 1$ and (b) $b_1 = 5$

Similarly, the solution at the end of the first substep in the subdomain B can be obtained

$$\begin{Bmatrix} u_B^{k+2/2ss} \\ \dot{u}_B^{k+2/2ss} \end{Bmatrix} = \begin{Bmatrix} -k_A \\ 0 \end{Bmatrix} \tag{6.91}$$

Following the solution procedure of the whole loop, we can easily conclude that the solutions at any point which are obtained by either linear interpolation or LSRT2 algorithm are always equal to the initial solutions. As a result, we can summarize that

$$\begin{Bmatrix} u_A^{k+1} & \dot{u}_A^{k+1} & u_B^{k+1} & \dot{u}_B^{k+1} \end{Bmatrix}^T = \begin{Bmatrix} u_A^k & \dot{u}_A^k & u_B^k & \dot{u}_B^k \end{Bmatrix}^T \tag{6.92}$$

which indicates the existence of the unitary eigenvalue.

In sum, we can conclude that the unit spurious eigenvalue is caused by the acceleration constraint, and its corresponding eigenvector can be obtained in the DAE equations.

Another spurious eigenvalue is real and its absolute value is less than one. This is because the constraint is incomplete, only with acceleration continuity. If we introduce another Lagrange multiplier μ to impose velocity continuity, we can find both spurious become unit as shown in Fig. 6.25. However, one can observe that this case sometimes introduces instability which is not easy to avoid. Therefore the case is beyond the discussion of this thesis.

6.4.5 Numerical convergence analysis

In this subsection, a series of numerical simulations on the Single-DoF split-mass system are carried out to validate the accuracy analysis of the proposed methods using different coarse time step Δt from 10^{-4} to 10^{-1}. Similarly, we checked every case treated in the numerical stability analysis, as shown in Table 6.3. The initial conditions are chosen to be $d(t_0) = 1$ and $v(t_0) = 1$. Fig. 6.26 shows that the observed order of convergence which is in agreement with the symbolic analysis both in state variables and the Lagrange multipliers in Section 6.3.

Table 6.3: Numerical convergence analysis of the partitioned methods on a Single-DoF split-mass system

Case	Integrator	ss	γ	Results (Rate)
(1)	LSRT1	1	1	1
(2)	LSRT1	2	1	1
(3)	LSRT1	10	1	1
(4)	LSRT2	1	$1\text{-}\frac{\sqrt{2}}{2}$	2
(5)	LSRT2	1	$1+\frac{\sqrt{2}}{2}$	2
(6)	LSRT2	1	$1\mp\frac{\sqrt{2}}{2}$	2
(7)	LSRT2	1	$1\pm\frac{\sqrt{2}}{2}$	2
(8)	LSRT2	2	$1\text{-}\frac{\sqrt{2}}{2}$	2
(9)	LSRT2	2	$1+\frac{\sqrt{2}}{2}$	2
(10)	LSRT2	2	$1\mp\frac{\sqrt{2}}{2}$	2
(11)	LSRT2	2	$1\pm\frac{\sqrt{2}}{2}$	2
(12)	LSRT2	10	$1\text{-}\frac{\sqrt{2}}{2}$	2
(13)	LSRT2	10	$1+\frac{\sqrt{2}}{2}$	2
(14)	LSRT2	10	$1\mp\frac{\sqrt{2}}{2}$	2
(15)	LSRT2	10	$1\pm\frac{\sqrt{2}}{2}$	2

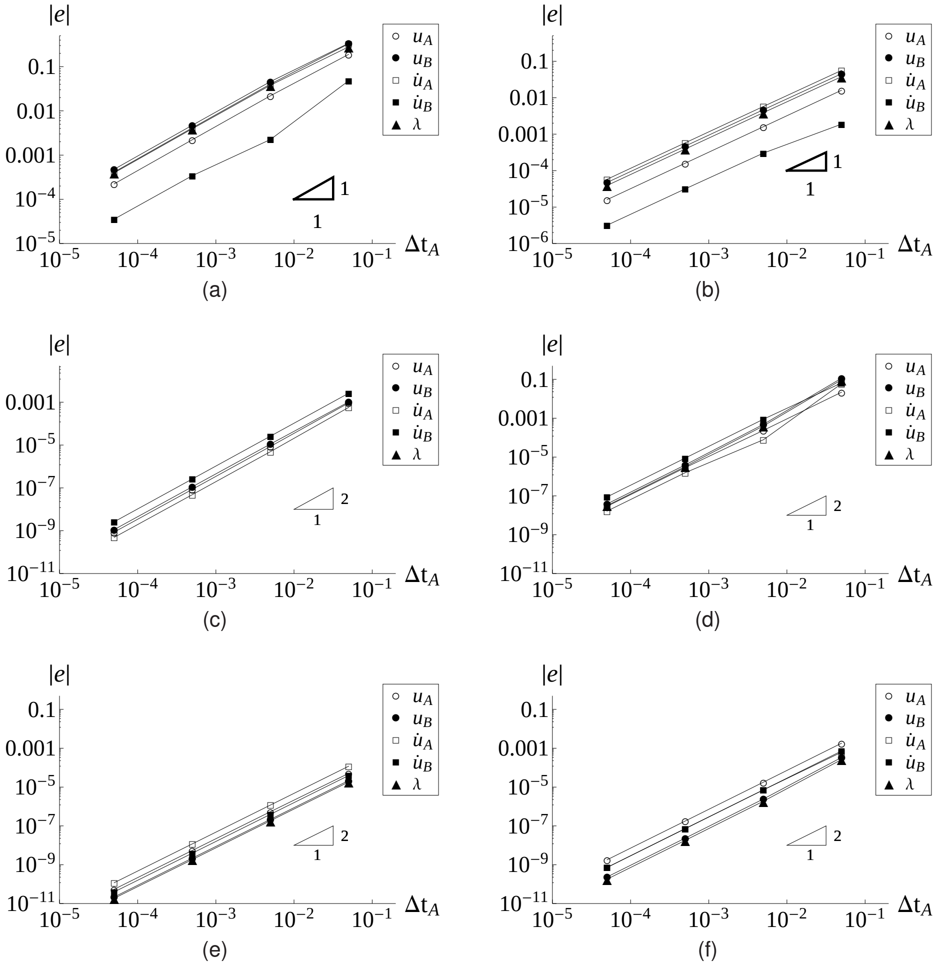
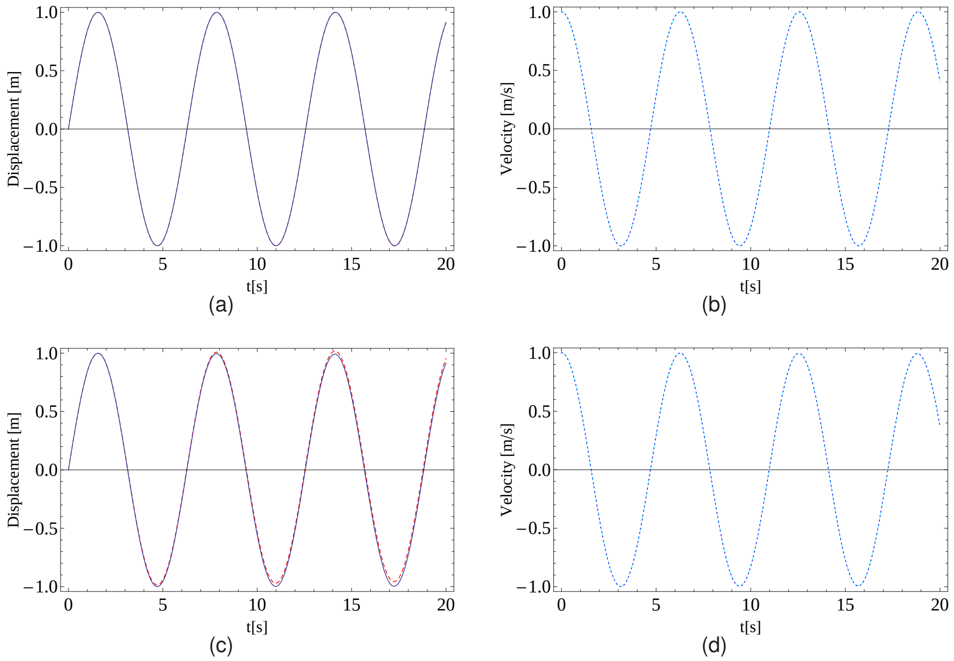


Fig. 6.26: Global error of the partitioned methods with b_1 = 10 and integrated by: (a) LSRT1 and *ss* = 1, (b) LSRT1 and *ss* = 10, (c) LSRT2 with $\gamma = 1 - \frac{\sqrt{2}}{2}$ and *ss* = 1; (d) LSRT2 with $\gamma = 1 + \frac{\sqrt{2}}{2}$ and *ss* = 1; (e) LSRT2 with $\gamma = 1 - \frac{\sqrt{2}}{2}$ and *ss* = 10; and (f) LSRT2 with $\gamma = 1 + \frac{\sqrt{2}}{2}$ and *ss* = 10



(a)

(b)

(c)

(d)

Fig. 6.27: The displacement and velocity responses in free vibration of the Single-DoF system with $b_1 = 10$ integrated by the partitioned method with $ss = 1$ and $\Delta t = 0.01$: (a), (b) LSRT2 with $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c), (d) LSRT2 with $\gamma = 1 + \frac{\sqrt{2}}{2}$

6.4.6 Numerical simulations

In this subsection, the simulations of Single-DoF system is calculated with different methods applying a relative large time step of $\Delta t = 0.01s$. From Fig. 6.27 and 6.28, we can observed that the case with $\gamma = 1 - \frac{\sqrt{2}}{2}$ and the case with subcycling relatively introduce less drift in the displacement responses. This can also be reflected in the spectral stability plots: for the cases with $\gamma = 1 + \frac{\sqrt{2}}{2}$ the non-zero spurious eigenvalue is smaller than the cases with $\gamma = 1 - \frac{\sqrt{2}}{2}$; the subcycling cases are more close to one than the cases with no subcycling.

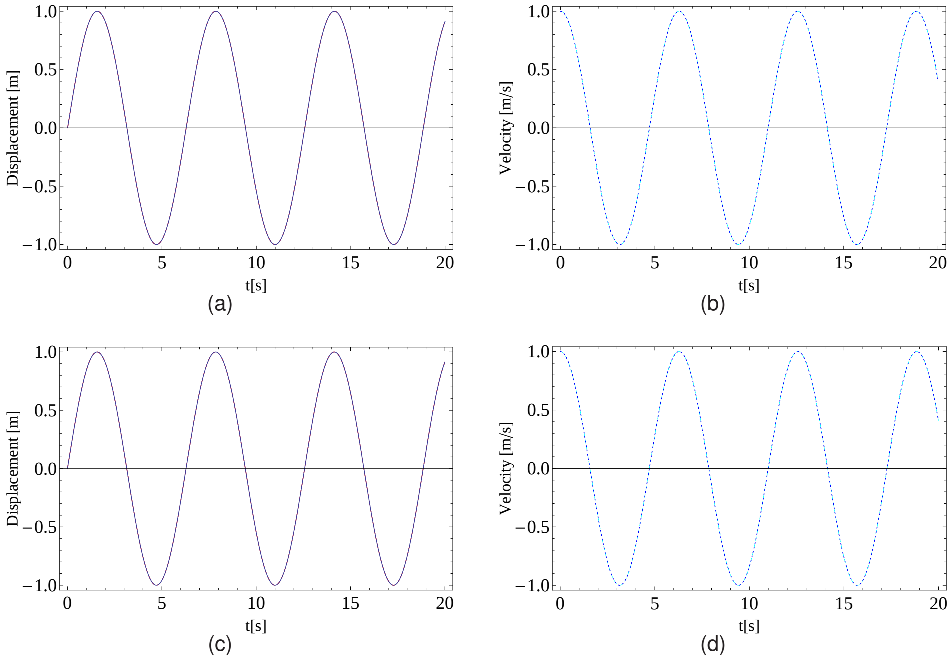


Fig. 6.28: The displacement and velocity responses in free vibration of the Single-DoF system with $b_1 = 10$ integrated by the partitioned method with $ss = 10$ and $\Delta t = 0.01$: (a), (b) LSRT2 with $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c), (d) LSRT2 with $\gamma = 1 + \frac{\sqrt{2}}{2}$

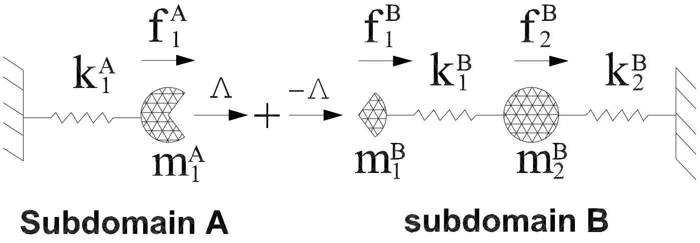


Fig. 6.29: Partitioned Two-DoF system 1 with two subdomains

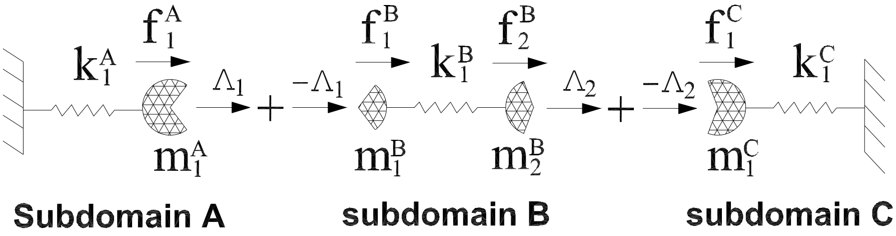


Fig. 6.30: Partitioned Two-DoF system 2 with three subdomains

6.5 Numerical simulation of partitioned Multiple-DoF linear systems

In this section, the numerical analysis of the partitioned methods are extended to Multiple-DoF systems and multiple interface problems. Several test problems including Two-, Three- and Four-DoF systems are considered to validate the proposed methods and confirm some theoretical results in previous sections.

6.5.1 Partitioned Multiple-DoF systems

For the sake of generality, the Two-DoF mass-spring system can be partitioned into two subdomains as depicted in Fig.6.29 and three subdomains as shown in Fig. 6.30. For the sake of brevity, the former case is analysed in depth and it is mainly used to validate the methods with subcycling. Conversely, the latter case is basically used to observe the performances of the partitioned method with multiple interfaces and we consider the same time step in all subdomains.

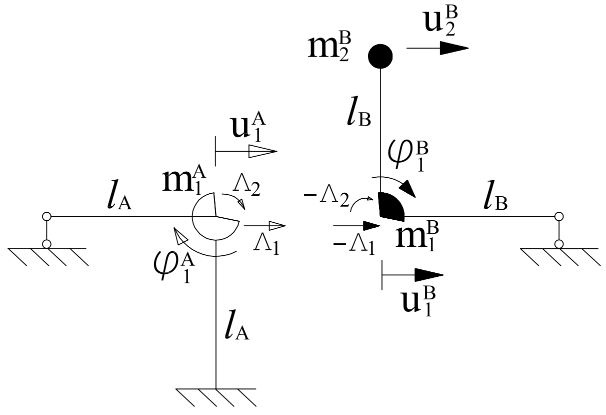


Fig. 6.31: Partitioned Three-DoF system with two DoFs at interface

In order to investigate the partitioned methods with multiple DoFs at the interface, we consider a Three-DoF system, which is split as shown in Fig. 6.31. The displacement vector of subdomain A is assumed to be $\mathbf{u}_A = \left\{u_1^A, \varphi_1^A\right\}^T$ and of subdomain B is chosen to be $\mathbf{u}_A = \left\{u_1^B, \varphi_1^B, u_2^B\right\}^T$ without considering vertical motion. The mass matrices of each subdomain read

$$\mathbf{M}_A = \frac{r}{1+r}\begin{bmatrix} m_1 & \\ & \rho^2 m_1 \end{bmatrix} \text{ and } \mathbf{M}_B = \frac{1}{1+r}\begin{bmatrix} m_1 & 0 & 0 \\ 0 & R_g^2 m_1 & 0 \\ 0 & 0 & m_2\,(1+r) \end{bmatrix} \tag{6.93}$$

and their stiffness matrix can be written as

$$\mathbf{K}_A = \frac{EI}{l_A^3}\begin{bmatrix} 12 & 6l_A \\ 6l_A & 7l_A^2 \end{bmatrix} \text{ and } \mathbf{M}_B = \frac{EI}{l_B^3}\begin{bmatrix} 3 & -3l_B & -3 \\ -3l_B & 6l_B^2 & 3l_B \\ -3 & 3l_B & 3 \end{bmatrix} \tag{6.94}$$

Here E is the Young's modulus, I is the Moment of Inertia of the cross section. All beams and columns involved are assumed to have the same cross section. m_1 and m_2 are the condensed mass and R_g is the radius of gyration for m_1. r defines the distribution of mass m_1 for two subdomains. In the simulation, we evaluate the parameters as follows: $E = 2 \times 10^8 kN/m^2$, $I = 2 \times 10^{-5} m$, $m_1 = m_2 = 20 \times 10^3 kg$, $R_g = 2m$ $l_A = 5m$ and $l_A = 3m$. The natural frequencies of the emulated system can be calculated as

$$f_1 = 1.084\,Hz,\ f_2 = 0.658\,Hz,\ f_3 = 0.103\,Hz \tag{6.95}$$

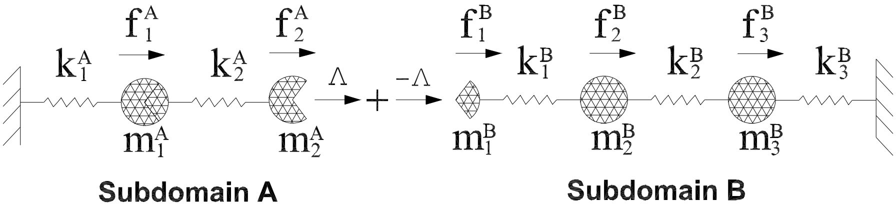


Fig. 6.32: Partitioned Four-DoF system 1 with one interface

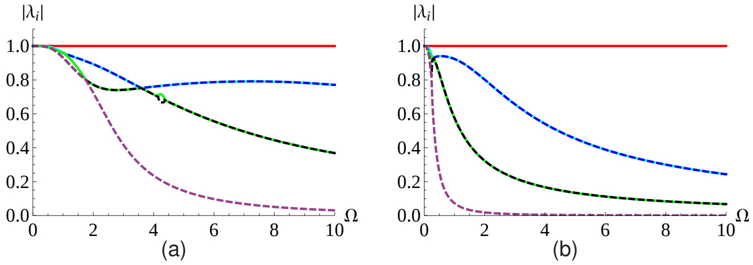
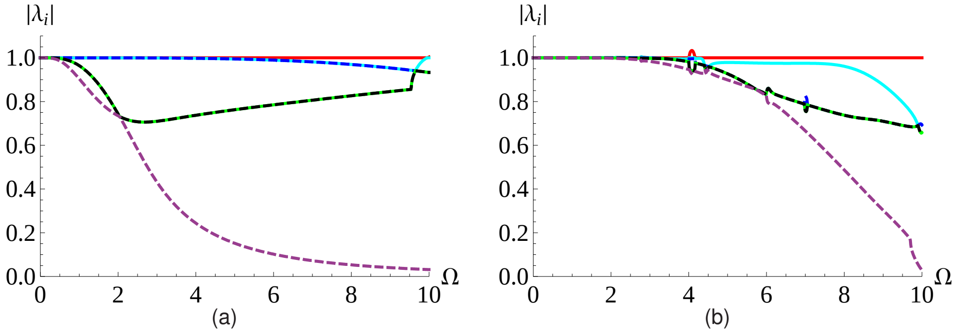


Fig. 6.33: $|\ \lambda_i\ |$ for the partitioned method applied to the partitioned Two-DoF system 1 with LSRT2, ss = 1 and b_1 = 10: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$ and (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$

In addition, a Four-DoF system which is split into two subdomains as illustrated in Fig. 6.32 is also considered to confirm the partitioned methods with one interface.

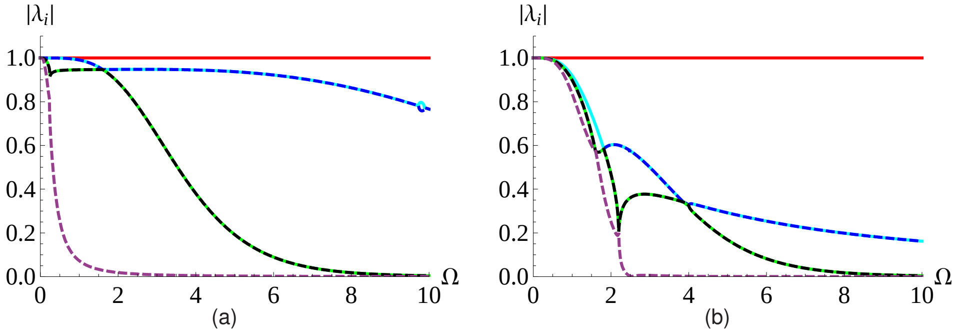
6.5.2 Spectral stability analysis for the Multiple-DoF systems

Although the spectral stability analysis was performed on the Single-DoF system, a direct extension to a Multiple-DoF system is impossible for the partitioned methods because model decomposition for DAE systems is not available. Therefore, the performance of the partitioned methods on Multiple-DoF system requires revalidation. Though numerous simulations were conducted to check the stability of the methods, only some representative results based on the LSRT2 method are presented for the sake of brevity, as shown in Table 6.4.



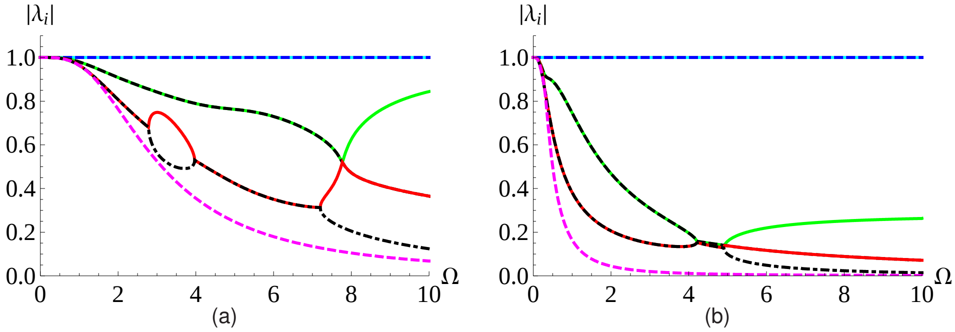
(a) (b)

Fig. 6.34: $|\ \lambda_i\ |$ for the partitioned method applied to the partitioned Two-DoF system 1 with LSRT2, ss = 10 and $\gamma = 1 - \frac{\sqrt{2}}{2}$: (a) b_1 = 0.1 and (b) b_1 = 10



(a) (b)

Fig. 6.35: $|\ \lambda_i\ |$ for the partitioned method applied to the partitioned Two-DoF system 1 with LSRT2, ss = 10 and $\gamma = 1 + \frac{\sqrt{2}}{2}$: (a) b_1 = 0.1 and (b) b_1 = 10



(a) (b)

Fig. 6.36: $|\ \lambda_i\ |$ for the partitioned method applied to the partitioned Two-DoF system 2 with LSRT2, ss = 1 (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$ and (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$

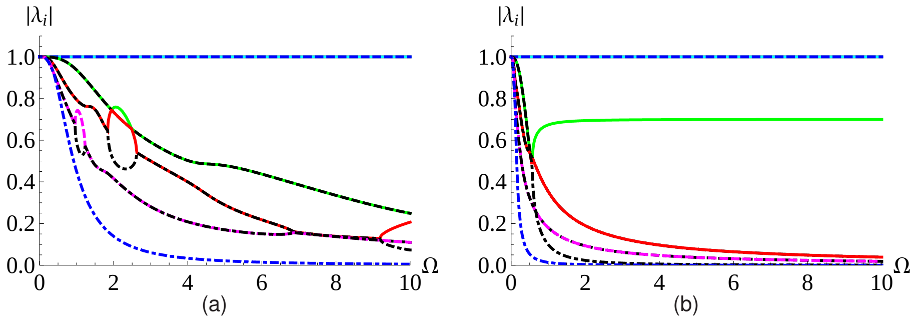


Fig. 6.37: | λ_i | for the partitioned method applied to the partitioned Three-DoF system with LSRT2, ss = 1 and r = 2: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$ and (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$

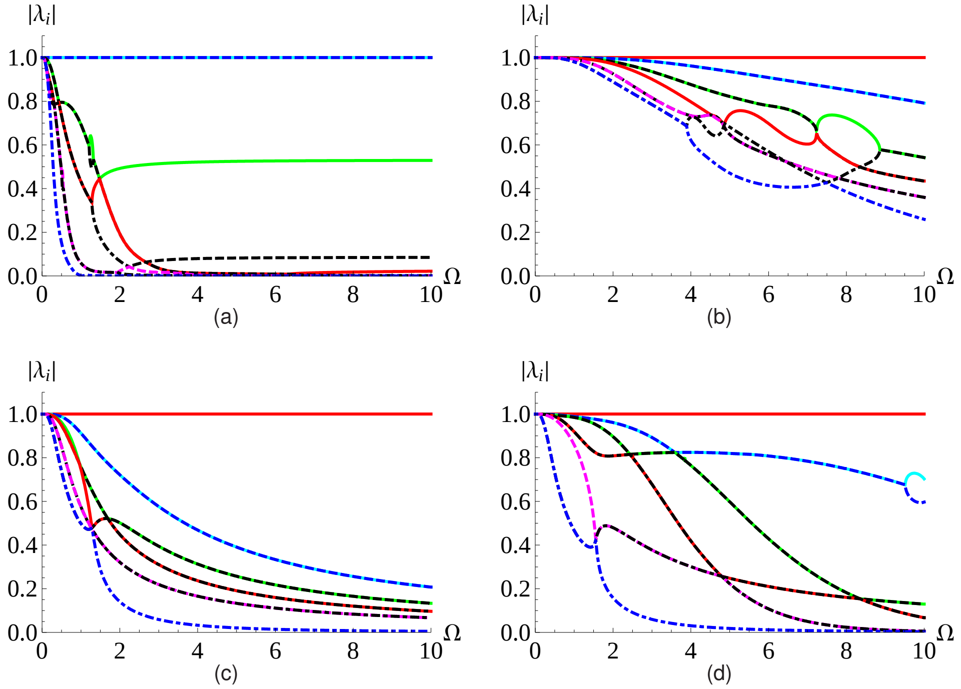


Fig. 6.38: | λ_i | for the partitioned method applied to: (a) the partitioned Three-DoF system with LSRT2, ss = 10, r = 2 and $\gamma = 1 - \frac{\sqrt{2}}{2}$, (b) the partitioned Four-DoF system with LSRT2, ss = 1 and $\gamma = 1 - \frac{\sqrt{2}}{2}$ (c) the partitioned Four-DoF system with LSRT2, ss = 1 and $\gamma = 1 + \frac{\sqrt{2}}{2}$ and (b) the partitioned Four-DoF system with LSRT2, ss = 10 and $\gamma = 1 + \frac{\sqrt{2}}{2}$

Table 6.4: Spectral stability analysis of the partitioned methods on partitioned Multiple-DoF systems

Case	system	ss	γ	Results
(1)	Fig.(6.29)	1	1-$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.33.a)
(2)	Fig.(6.29)	1	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.33.b)
(3)	Fig.(6.29)	10	1-$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.34)
(4)	Fig.(6.29)	10	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.35.a); unstable (Fig. 6.35.b)
(5)	Fig.(6.30)	1	1-$\frac{\sqrt{2}}{2}$	Conditionally stable (Fig. 6.36.a)
(6)	Fig.(6.30)	1	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.36.b)
(7)	Fig.(6.31)	1	1-$\frac{\sqrt{2}}{2}$	Conditionally stable (Fig. 6.37.a)
(8)	Fig.(6.31)	1	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.37.b)
(9)	Fig.(6.31)	10	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.38.a)
(10)	Fig.(6.32)	1	1-$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.38.b)
(11)	Fig.(6.32)	1	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.38.c)
(12)	Fig.(6.32)	10	1+$\frac{\sqrt{2}}{2}$	Stable (Fig. 6.38.d)

6.5.3 Drift analysis

In the partitioned Two-DoF system 2, a floating subdomain B is included. In this case, the stiffness matrix $\mathbf{K}_B$ is not invertible. This indicates that the floating subdomain contains one free mass mode which has no stiff term. As shown in Fig. 6.36, three unitary eigenvalues are included in the spectral plot. However, if we calculate the eigenvectors of the amplification matrix, we can find that only two linearly independent eigenvectors are corresponding to the three unitary eigenvalues.

Following the stability conditions provided by (Hughes, 1983, p.89), the case with incomplete eigenvectors leads to the so-called called weak instability and therefore the drift-off effect in the displacement and velocity responses. As we analysed before, the incomplete constraint also induces drift. Therefore, we numerically investigate the four representative cases and the results are listed in Table 6.5. Note that n_f represents the number of the floating DoFs; k_a stands for the number of unitary eigenval-

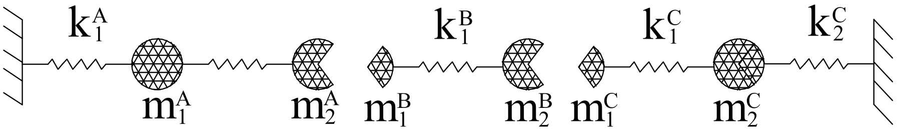


Fig. 6.39: Partitioned Four-DoF system 2 with two subdomains

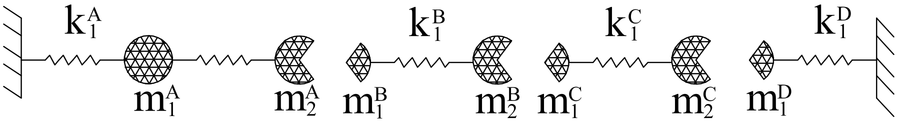
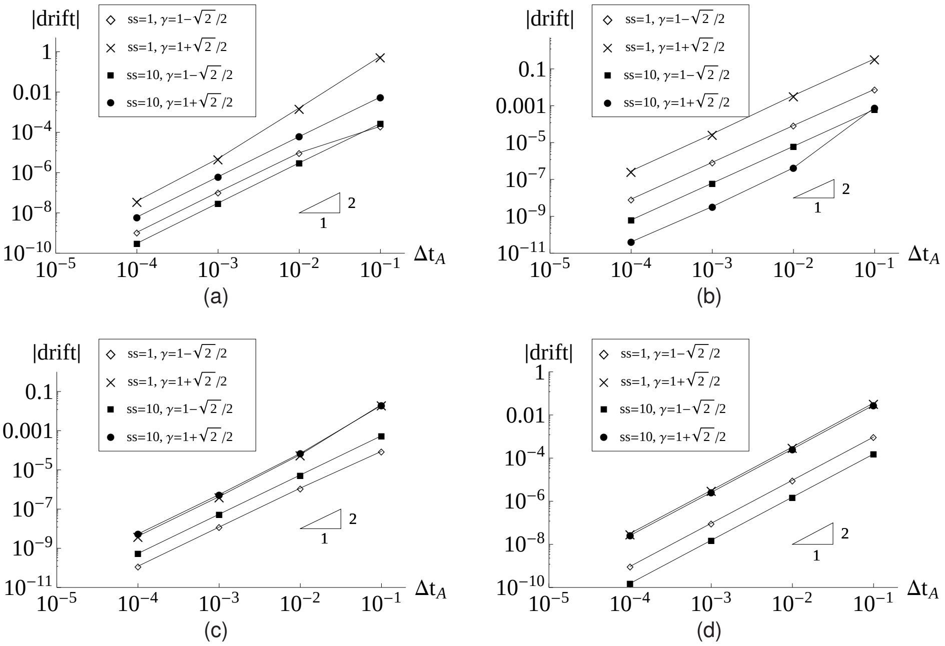


Fig. 6.40: Partitioned Four-DoF system 3 with three subdomains

ues, i.e, *algebraic multiplicity*; and k_g the maximum number of independent eigenvectors corresponding to the unitary eigenvalues, i.e., *geometric multiplicity* (Seyranian and Mailybaev, 2004). In addition, the drift in the displacement response of the aforementioned systems integrated with different partitioned methods, with respect to the time step, are depicted in Figs. (6.41) and (6.42). Note that the drift here is expressed by the different solutions at interfaces. From these figures, we can observe that the drift is quadratic with respect to the utilized time step. Moreover, the cases with subcycling exhibit smaller drift compared to the cases without subcycling. To continue, we consider again the partitioned Four-DoF system with three interfaces in depth. The Jordan decomposition for the amplification matrix of the APNR1 method applied

Table 6.5: The drift analysis of the partitioned methods on the partitioned Single- and Four-DoF systems

Case	system	n_c	n_f	k_g	k_a	Results (Rate)
(1)	Fig.6.4 ($k_B \neq 0$)	1	0	1	1	2 (Fig. 6.41.a-b)
(2)	Fig.6.4 ($k_B = 0$)	1	1	2	1	2 (Fig. 6.41.c-d)
(3)	Fig.6.39	2	1	3	2	2 (Fig. 6.42.a-b)
(4)	Fig.6.40	3	2	5	3	2 (Fig. 6.42.c-d)



(a) (b) (c) (d)

Fig. 6.41: Drift analysis for the partitioned methods with LSRT2 for different systems: (a),(b) displacement drift and velocity drift at the interface of the partitioned Single DoF system with $b_1 = 10$; (c), (d) displacement drift and velocity drift at the interface of the partitioned Single-DoF system with $b_1 = 10$ and $k_B = 0$.

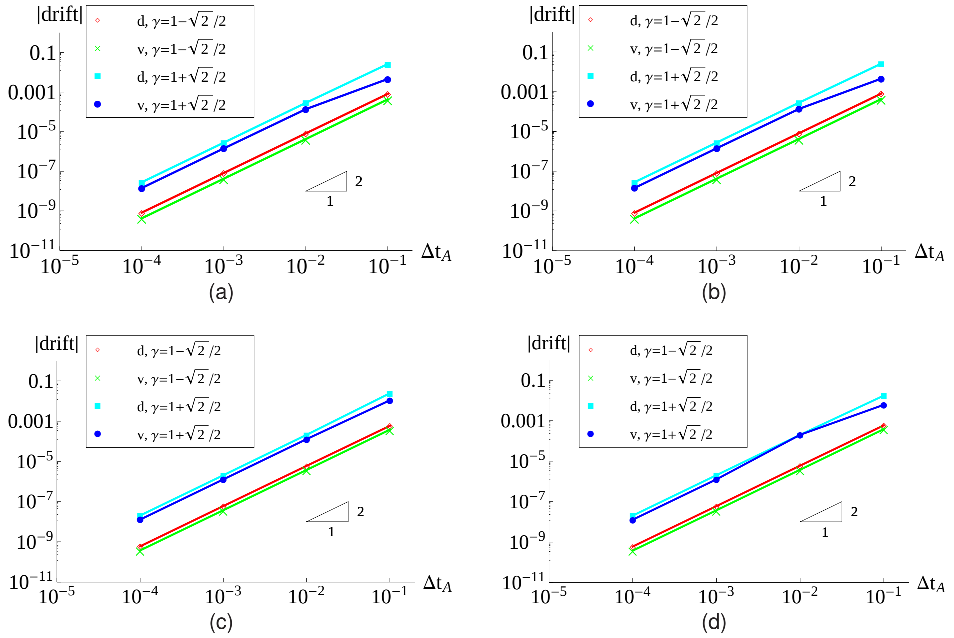


Fig. 6.42: Drift analysis for the partitioned methods with LSRT2 for different systems: (a) drifts at the first interface of the partitioned Four-DoF system with two interfaces; (b) drifts at the second interface of the partitioned Four-DoF system with two interfaces; (c) drifts at the first interface of the partitioned Four-DoF system with three interfaces; (d) drifts at the second interface of the partitioned Four-DoF system with three interfaces.

to this system leads to a Jordan block as

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.96}$$

which is actually composed of two 2×2 Jordan blocks with unitary eigenvalue. This is different from the following case

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.97}$$

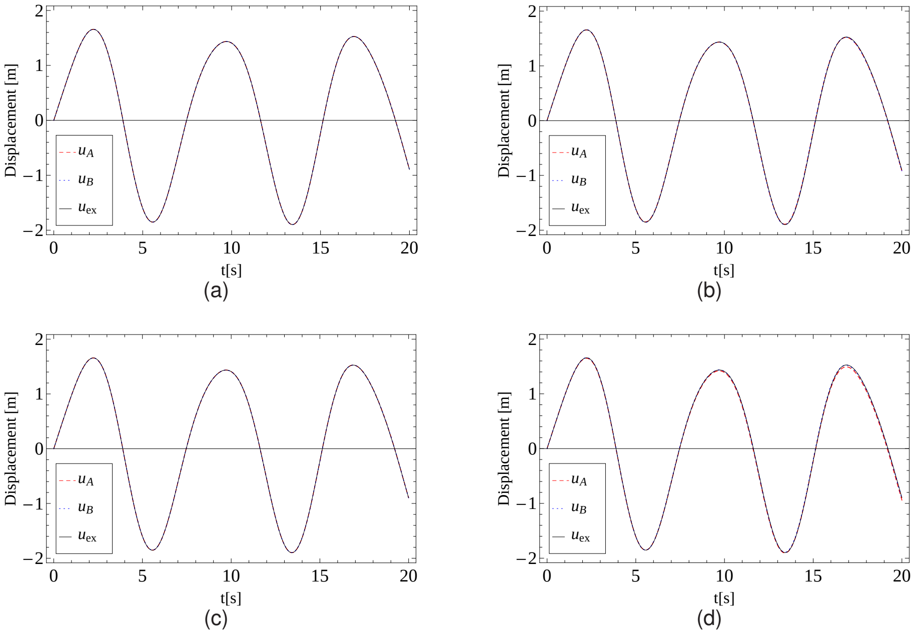
This form is a 3×3 Jordan block with unitary eigenvalues. The difference between (6.96) and (6.97 can be derived in their nth power:

$$\begin{bmatrix} 1 & n & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & n \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & n & \frac{n^2+n}{2} & 0 \\ 0 & 1 & n & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.98}$$

In general, the amplification matrix usually contains $(k_g - ka)$ Jordan blocks of 2×2 as (6.96). Therefore, we can conclude that the drift-off effect is not aggravated with the increasing number of the floating DoFs.

6.5.4 Simulations

In order to illustrate the stability properties of the partitioned methods when applied to Multiple-DoF systems, several free vibration simulations for a partitioned Two-DoF system were conducted, as shown in Fig. 6.43. The initial conditions were chosen to be unit velocity and zero displacement for every DoF. For the partitioned methods with no subcycling, the time step is fixed to 10ms. For the subcycling cases, the coarse time step and the fine time step is chosen to be $\Delta t_A = 10ms$ and $\Delta t_B = 1ms$, respectively. It can be clearly observed that the displacement drift is not noticeable. The drift-off effect in the velocity is relatively smaller which are omitted for brevity.



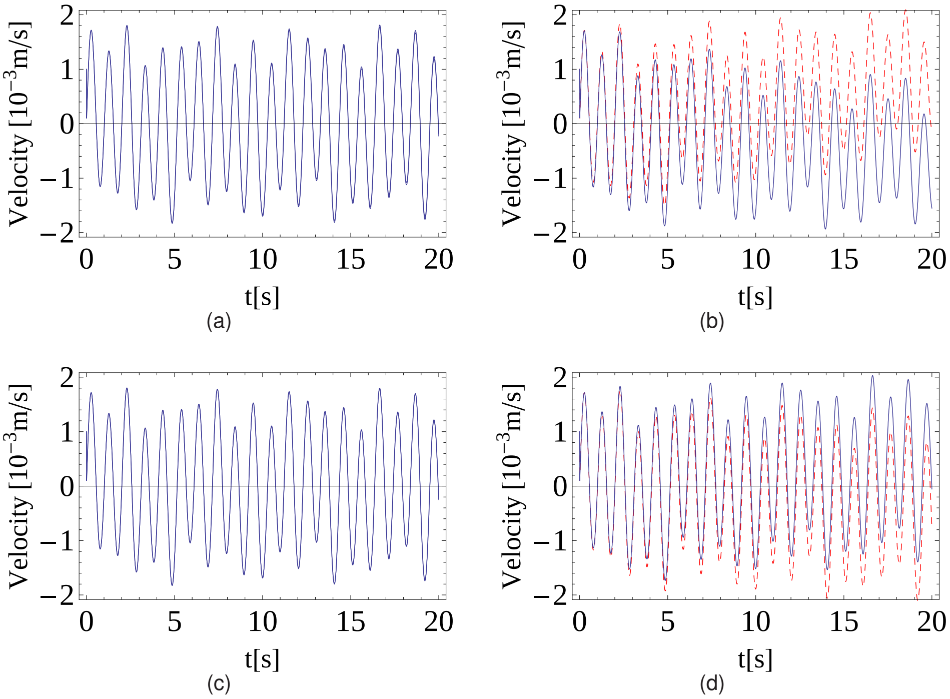
(a)

(b)

(c)

(d)

Fig. 6.43: Numerical simulations for the partitioned Two-DoF system with one interface: (a) displacements at the interface obtained with ss = 1 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) displacements at the interface obtained with ss = 1 and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) displacements at the interface obtained with ss = 10 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (d) displacements at the interface obtained with ss = 10 and $\gamma = 1 + \frac{\sqrt{2}}{2}$.



(a)

(b)

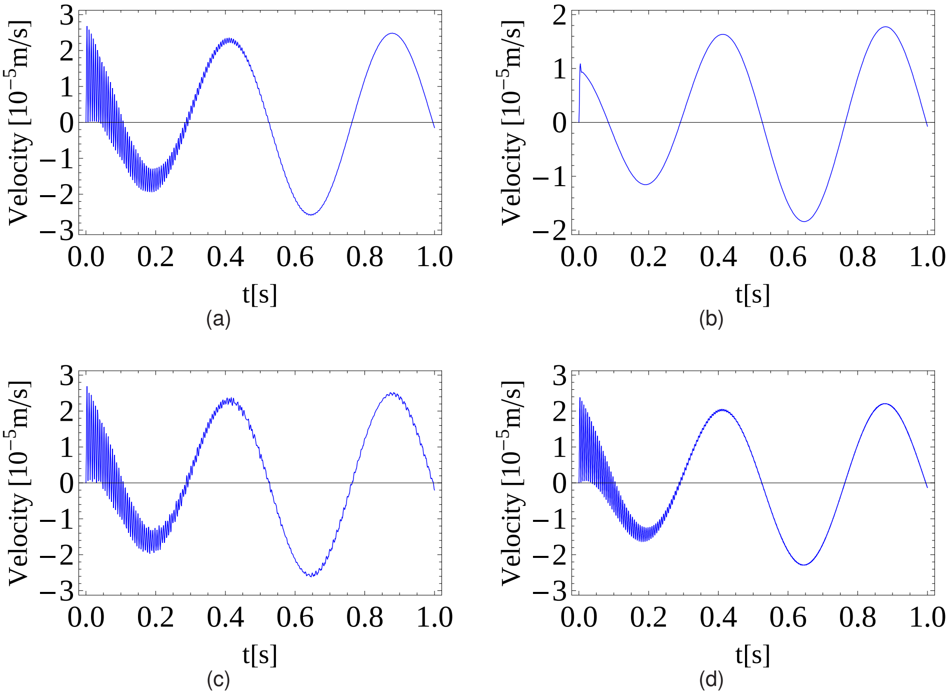
(c)

(d)

Fig. 6.44: Numerical simulations for the partitioned Three-DoF system with one interface: (a) velocities of the translational DoF at the interface obtained with *ss* = 1 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) velocities of the translational DoF at the interface obtained with *ss* = 1 and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) velocities of the translational DoF at the interface obtained with *ss* = 10 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (d) velocities of the translational DoF at the interface obtained with *ss* = 10 and $\gamma = 1 + \frac{\sqrt{2}}{2}$.

The second example is the partitioned Three-DoF system. The objective is to observe the drift induced by the weak instability. The numerical results, displayed in Fig. 6.44, also indicate that the partitioned method with $\gamma = 1 - \frac{\sqrt{2}}{2}$ has less drift at interface. In the case with floating subdomain, the weak instability becomes the foremost limit of the partitioned method. Therefore, the parameter $\gamma = 1 - \frac{\sqrt{2}}{2}$ is preferred.

To illustrate the nonlinear performance and high-frequency dissipation of the partitioned methods, let us consider again the coupled spring-pendulum stiff system as described in Fig. 4.1 of Section 4.2. The system characteristics and the external force is also chosen as shown in Table 3.1. The time step used in subdomain *NS* is



(a) (b) (c) (d)

Fig. 6.45: Numerical simulations for the partitioned spring-pendulum stiff system with one interface: (a) extensional velocities of the pendulum obtained with $ss = 1$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) extensional velocities of the pendulum obtained with $ss = 1$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) extensional velocities of the pendulum obtained with $ss = 10$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (d) extensional velocities of the pendulum obtained with $ss = 10$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$.

fixed to 1ms. Clearly, high-frequency dissipation property of their progenitor methods is maintained in the partitioned case. Similarly, the methods with $\gamma = 1 + \frac{\sqrt{2}}{2}$ exhibit higher numerical dissipation. However, as shown in the simulations of the partitioned Three-DoF system, this also bring about higher drift at interface so that this makes long term simulation impossible. Accordingly, the choice of the parameter γ depends on the period of the simulation, the system characteristics, the accuracy requirement and so on.

The partitioned methods based on acceleration continuity were also extended in this chapter to incorporate subcycling strategies, resulting in novel staggered partitioned methods. In these methods, the emulated system is partitioned into two subdomains: one is integrated with the coarse time step and the other with the fine time step. In addition to the computation of the Lagrange multipliers and the integration of the subdomains, a linear interpolation is required before advancing the solution of the subdomain with the fine time step. This prevents the partitioned methods to be implemented in parallel simulations.

For these staggered partitioned methods, the accuracy analyses were performed by setting a reference solution procedure by which each subdomain advanced separately with exact Lagrange multipliers and both the differential vectors and the Lagrange multiplier vector were proved to preserve the same order of accuracy as the progenitor monolithic method. For comparison, the GC method was also proved in a similar way to have a reduction of accuracy order with respect to its progenitor Newmark method. This is because the inverse term **H** in the formula of the Lagrange multiplier vector of the GC method includes a Δt factor while the proposed staggered methods do not.

The stability of the proposed staggered methods were studied by means of the spectral approach on a Single-DoF split-mass system, and these methods frequently appears to be stable. Meanwhile, the components of the eigenvalues of the amplification matrix was investigated in depth. Four non-zero eigenvalues were included in the amplification matrix. Among them, only one pair of complex conjugate eigenvalues are principle eigenvalues, whereas the other two are spurious. For the complex conjugate eigenvalues, it was studied in terms of numerical damping ratio and frequency error. For the spurious eigenvalues, the one unitary is unitary was found to be corresponding to the ”equilibrium point” of the continuous DAE system and the other that was real and less than one was observed to be owing to incomplete constraint at the interface.

When dealing with a partitioned system in the presence of floating DoFs, the number of unitary eigenvalues is greater than one which indicates weak instability and therefore the displacement solutions exhibit slightly drift phenomena. But it was observed that with proper choice of the parameter γ, the drift effect could be limited to be so small as to be negligible. Moreover, it was proved that the drift effect would not accumulate with the increasing number of the floating DoFs.

CHAPTER 7

PARTITIONED TIME INTEGRATION METHODS WITH AN INTERFIELD PARALLEL SOLUTION PROCEDURE

7.1 Introduction

The partitioned methods mentioned in the previous chapter provide an efficient and accurate approach to solving large-scale structural dynamics problem. In those methods, subcycling strategies are adopted. The subcycling strategies, on the one hand, enable different subdomains to be advanced with different time steps; on the other hand, they make the solution procedure be sequential in that the integration of subdomain B is dependent on the interpolation achieved after the integration of subdomain A. Like the GC method (Combescure and Gravouil, 2002), these methods are not suitable for RTDS tests and continuous PsD tests.

In order to solve this problem, Pegon and Magonette (2002) developed and implemented an interfield parallel solution procedure, the PM method, where both subcycling and interfield parallelism are allowed. Along the same line, the Rosenbrock-based partitioned methods developed in the previous chapter are extended to an interfield parallel solution procedure in this chapter.

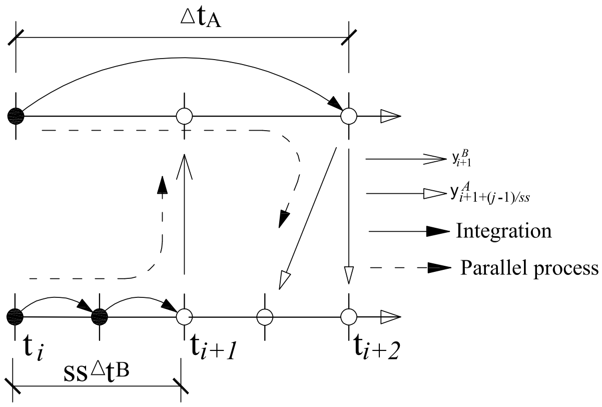


Fig. 7.1: The interfield parallel solution procedure of the LSRT1-based partitioned method

7.2 Formulations of parallel partitioned methods

7.2.1 An interfield parallel solution procedure based on the LSRT1 time-stepping method (**APSR1**)

In order to illustrate the procedure, it is assumed that the system under consideration is split into two subdomain A and B. As shown in Fig. 7.1, Subdomain A is advanced with the coarse time step Δt_A, while Subdomain B with the fine time step Δt_B. Actually in this procedure, the system time step is $\Delta t = 1/2\Delta t_A$. The solution of Subdomain A is advanced further in order to provide the required solution for the next time step. It is noted that the parameter ss is related to the ratio between the system time step and the fine time step, i.e., $ss = \Delta t/\Delta t_B$. The solution procedure from t_i to t_{i+1} can be understood from the flowchart of Fig. 7.2. In detail, the procedure contains two independent processes and one information exchange at the end. Initial solutions are required from the previous two loops in subdomain A : $\mathbf{y}^A_{i+\frac{j-1}{ss}}$ and $\mathbf{y}^A_i$ while one is taken from the previous loop in subdomain B: $\mathbf{y}^B_i$. For Subdomain A, the solution procedure can be elaborated as follows:

1. Evaluate $\mathbf{F}^A_i$ and $\mathbf{F}^B_i$ with the initial solution $\mathbf{y}^A_i$ and $\mathbf{y}^B_i$, and then calculate the

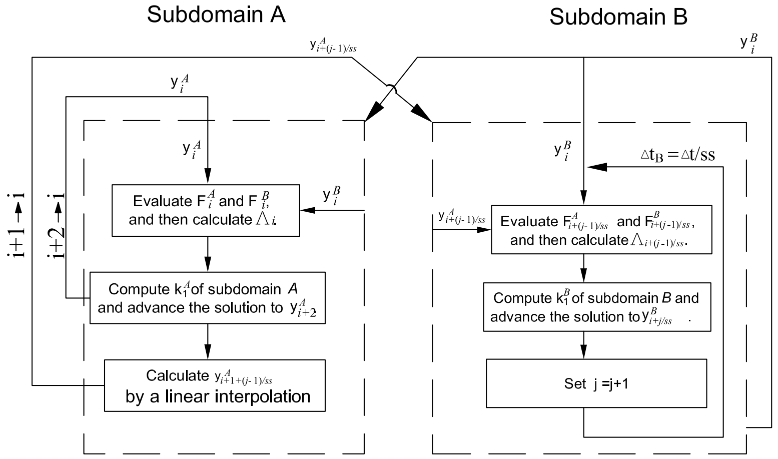

Subdomain A
Subdomain B
$y^A_{i+(j-1)/ss}$
y^B_i
y^A_i
y^A_i
y^B_i
$\Delta t_B = \Delta t/ss$
$i+1 \rightarrow i$
$i+2 \rightarrow i$
Evaluate F^A_i and F^B_i, and then calculate Λ_i.
y^B_i
$y^A_{i+(j-1)/ss}$
Evaluate $F^A_{i+(j-1)/ss}$ and $F^B_{i+(j-1)/ss}$, and then calculate $\Lambda_{i+(j-1)/ss}$.
Compute k^A_1 of subdomain A and advance the solution to y^A_{i+2}
Compute k^B_1 of subdomain B and advance the solution to $y^B_{i+j/ss}$.
Calculate $y^A_{i+1+(j-1)/ss}$ by a linear interpolation
Set j =j+1
$i+1 \rightarrow i$

Fig. 7.2: Flowchart for the interfield parallel solution procedure of the LSRT1-based partitioned method

Lagrange multiplier vector Λ_i as follows

$$\Lambda_i = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}_i^A + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}_i^B\right] \tag{7.1}$$

2. Compute $\mathbf{k}_1^A$ and advance the solution to $\mathbf{y}_{i+2}^A$

$$\begin{aligned} \mathbf{k}_1^A &= [\mathbf{I} - 2\Delta t\gamma\mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}_i^A + \mathbf{C}_A^T\Lambda_i\right)2\Delta t \\ \mathbf{y}_{i+2}^A &= \mathbf{y}_i^A + \mathbf{k}_1^A \end{aligned} \tag{7.2}$$

3. Calculate $\mathbf{y}^A_{i+\frac{j}{ss}}$ by means of a linear interpolation

$$\mathbf{y}_{i+j/ss}^A = \left(1 - \frac{j}{ss}\right)\mathbf{y}_i^A + \frac{j}{ss}\mathbf{y}_{i+1}^A \tag{7.3}$$

The solution procedure in Subdomain *B* involves *ss* substeps, and each substep advances as follows:

1. Evaluate $\mathbf{F}^A_{i+\frac{j-1}{ss}}$ and $\mathbf{F}^B_{i+\frac{j-1}{ss}}$, and calculate the Lagrange multiplier vector $\Lambda_{i+\frac{j-1}{ss}}$ as follows

$$\Lambda_{i+\frac{j-1}{ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{i+\frac{j-1}{ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{i+\frac{j-1}{ss}}\right] \tag{7.4}$$

2. Calculate $\mathbf{k}_1^B$ and advance the solution to $\mathbf{y}^B_{i+\frac{j}{ss}}$

$$\begin{aligned} \mathbf{k}_1^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma \mathbf{J}_B\right]^{-1} \mathbf{A}_B^{-1} \left(\mathbf{F}_i^B + \mathbf{C}_B^T \Lambda_i\right) \frac{\Delta t}{ss} \\ \mathbf{y}^B_{i+\frac{j}{ss}} &= \mathbf{y}^B_{i+\frac{j-1}{ss}} + \mathbf{k}_1^B \end{aligned} \tag{7.5}$$

3. If $j = ss$, then end the loop in subdomain B.

With this procedure, we can easily find that the initial solutions needed for the each subdomain were provided in the previous step and the interfield communications are needed only at the end of each time step. This makes the solution of each subdomain independent, and therefore allows implementing parallel computations. When compared to the staggered partitioned methods, the parallelism is achieved at the expense of using large time step of subdomain A which certainly amplifies integration errors. Moreover, the process needs more initial solutions which increase the memory load and, subsequently, the computational cost.

Similarly as the PM method, this procedure is not self-starting. Herein, the LSRT1-based partitioned method without subcycling is adopted to initiate the procedure with the system time step, i.e., advancing the solutions from $\mathbf{y}_0^A$ and $\mathbf{y}_0^B$ to $\mathbf{y}_1^A$ and $\mathbf{y}_1^B$. Moreover, this choice does not destroy the accuracy of the method. This is because both of them have the same order for the local truncation error. In addition, this scheme preserves the parallel characteristics of the overall solution procedure.

7.2.2 An interfield parallel solution procedure based on the LSRT2 time-stepping method (**APSR2**)

Since the LSRT2 method contains two stages, the integration in subdomain A is expected to advance with the time step $\Delta t_A = 4\Delta t$ as shown in Fig. 7.3. Moreover, the solution procedure from t_i to t_{i+1} can be understood from the flowchart depicted in Fig. 7.4. The integration of both subdomains is independent and this provides the possibility for parallel computations. Differently from the previous case, the integration for subdomain A not only needs the Lagrange multiplier at the beginning of the first stage but also that at the beginning of the second stage. Therefore, the solution

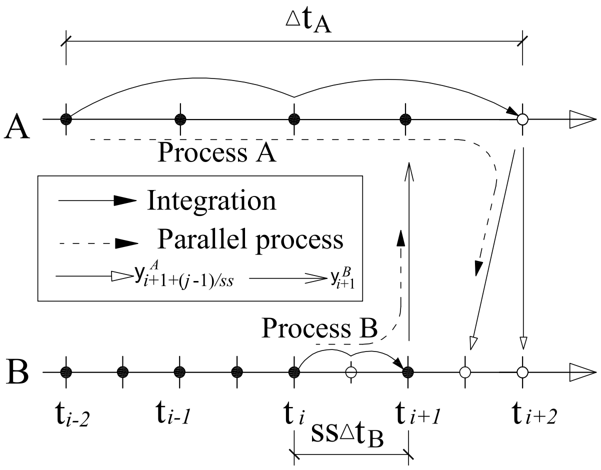


Fig. 7.3: The interfield parallel solution procedure of the LSRT2-based partitioned method

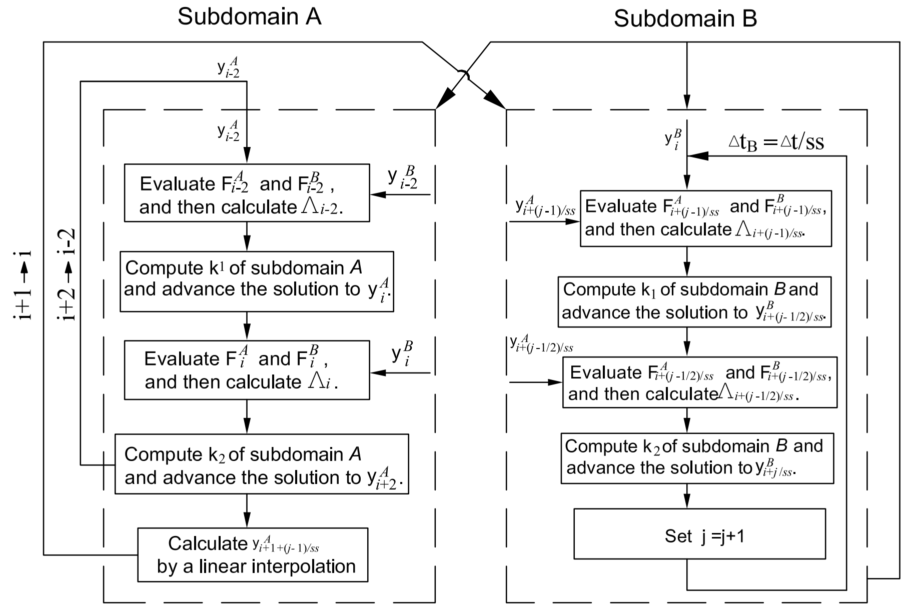

Subdomain A
Subdomain B
y^A_{i-2}
y^A_{i-2}
Evaluate F^A_{i-2} and F^B_{i-2}, and then calculate Λ_{i-2}.
y^B_{i-2}
Compute k_1 of subdomain A and advance the solution to y^A_i.
Evaluate F^A_i and F^B_i, and then calculate Λ_i.
y^B_i
Compute k_2 of subdomain A and advance the solution to y^A_{i+2}.
Calculate $y^A_{i+1+(j-1)/ss}$ by a linear interpolation
$i+1 \to i$
$i+2 \to i-2$
y^B_i
$\Delta t_B = \Delta t/ss$
$y^A_{i+(j-1)/ss}$
Evaluate $F^A_{i+(j-1)/ss}$ and $F^B_{i+(j-1)/ss}$, and then calculate $\Lambda_{i+(j-1)/ss}$.
Compute k_1 of subdomain B and advance the solution to $y^B_{i+(j-1/2)/ss}$.
$y^A_{i+(j-1/2)/ss}$
Evaluate $F^A_{i+(j-1/2)/ss}$ and $F^B_{i+(j-1/2)/ss}$, and then calculate $\Lambda_{i+(j-1/2)/ss}$.
Compute k_2 of subdomain B and advance the solution to $y^B_{i+j/ss}$.
Set j =j+1

Fig. 7.4: Flowchart for the interfield parallel solution procedure of the LSRT2-based partitioned method

procedure in subdomain A starts earlier at time t_{i-2} and ends further at time t_{i+2}. For this method, the solution procedure in subdomain A can be described as follows:

1. Evaluate $\mathbf{F}^A_{i-2}$ and $\mathbf{F}^B_{i-2}$ with the initial solutions $\mathbf{y}^A_{i-2}$ and $\mathbf{y}^B_{i-2}$ and then calculate the Lagrange multiplier vector Λ_{i-2}

$$\Lambda_{i-2} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{i-2} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{i-2}\right] \tag{7.6}$$

2. Compute $\mathbf{k}^A_1$ and advance the solution to $\mathbf{y}^A_i$

$$\begin{aligned}\mathbf{k}^A_1 &= [\mathbf{I} - 4\Delta t\gamma\mathbf{J}_A]^{-1}\,\mathbf{A}_A^{-1}\left(\mathbf{F}^A_{i-2} + \mathbf{C}_A^T\Lambda_{i-2}\right)4\Delta t\\ \mathbf{y}^A_i &= \mathbf{y}^A_{i-2} + \frac{1}{2}\mathbf{k}^A_1\end{aligned} \tag{7.7}$$

3. Evaluate $\mathbf{F}^A_i$ and $\mathbf{F}^B_i$ with the initial solution $\mathbf{y}^A_i$ and $\mathbf{y}^B_i$ and then calculate the Lagrange multiplier vector Λ_i

$$\Lambda_i = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_i + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_i\right] \tag{7.8}$$

4. Compute $\mathbf{k}^A_2$ and advance the solution to $\mathbf{y}^A_{i+2}$

$$\begin{aligned}\mathbf{k}^A_2 &= [\mathbf{I} - 4\Delta t\gamma\mathbf{J}_A]^{-1}\left(\mathbf{A}_A^{-1}\left(\mathbf{F}^A_i + \mathbf{C}_A^T\Lambda_i\right) - \gamma\mathbf{J}_A\mathbf{k}^A_1\right)4\Delta t\\ \mathbf{y}^A_{i+2} &= \mathbf{y}^A_i + \mathbf{k}^A_2\end{aligned} \tag{7.9}$$

5. Calculate $\mathbf{y}^A_{i+\frac{j}{ss}}$ by means of a linear interpolation

$$\mathbf{y}^A_{\frac{j}{2ss}} = \left(1 - \frac{j}{2ss}\right)\mathbf{y}^A_i + \frac{j}{2ss}\mathbf{y}^A_{i+1} \tag{7.10}$$

At the same time, the procedure in subdomain B is performing ss substeps of solution processes, each of which can be detailed as follows:

1. Evaluate $\mathbf{F}^A_{i+\frac{j-1}{ss}}$ and $\mathbf{F}^B_{i+\frac{j-1}{ss}}$, and calculate the Lagrange multiplier vector $\Lambda_{i+\frac{j-1}{ss}}$

$$\Lambda_{i+\frac{j-1}{ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{i+\frac{j-1}{ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{i+\frac{j-1}{ss}}\right] \tag{7.11}$$

2. Calculate $\mathbf{k}^B_1$ and advance the solution to $\mathbf{y}^B_{i+\frac{2j-1}{2ss}}$

$$\begin{aligned}\mathbf{k}^B_1 &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\mathbf{A}_B^{-1}\left(\mathbf{F}^B_{i+\frac{j-1}{ss}} + \mathbf{C}_B^T\Lambda_{i+\frac{j-1}{ss}}\right)\frac{\Delta t}{ss}\\ \mathbf{y}^B_{i+\frac{2j-1}{2ss}} &= \mathbf{y}^B_{i+\frac{j-1}{ss}} + \frac{1}{2}\mathbf{k}^B_1\end{aligned} \tag{7.12}$$

3. Evaluate $\mathbf{F}^A_{i+\frac{2j-1}{2ss}}$ and $\mathbf{F}^B_{i+\frac{2j-1}{2ss}}$ and calculate the Lagrange multiplier vector $\Lambda_{i+\frac{2j-1}{2ss}}$

$$\Lambda_{i+\frac{2j-1}{2ss}} = -\mathrm{H}^{-1}\left[\mathbf{C}_A\mathbf{A}_A^{-1}\mathbf{F}^A_{i+\frac{2j-1}{2ss}} + \mathbf{C}_B\mathbf{A}_B^{-1}\mathbf{F}^B_{i+\frac{2j-1}{2ss}}\right] \tag{7.13}$$

4. Calculate $\mathbf{k}_2^B$ and advance the solution to $\mathbf{y}^B_{i+\frac{j}{ss}}$

$$\begin{aligned}\mathbf{k}_2^B &= \left[\mathbf{I} - \frac{\Delta t}{ss}\gamma\mathbf{J}_B\right]^{-1}\left(\mathbf{A}_B^{-1}\left(\mathbf{F}^B_{i+\frac{2j-1}{2ss}} + \mathbf{C}_B^T\Lambda_{i+\frac{2j-1}{2ss}}\right) - \gamma\mathbf{J}_B\mathbf{k}_1^B\right)\frac{\Delta t}{ss} \\ \mathbf{y}^B_{i+\frac{j}{ss}} &= \mathbf{y}^B_{i+\frac{j-1}{ss}} + \mathbf{k}_2^B\end{aligned} \tag{7.14}$$

5. If $j = ss$, then end the loop in subdomain B.

Differently from the LSRT2-based staggered case, the parameter ss here can be odd or even. Similarly, this method is not self-starting and the LSRT2-based partitioned method with no subcycling is chosen to preserve the second-order accuracy and the parallel characteristics. In this case, three loops of initial procedures shown in Fig. 6.3 are required with the time step Δt to achieve the solutions at time t_1, t_2 and t_3. Subcycling is introduced when the two subdomains have different requirements of time step. The initial procedure without subcycling may not satisfy these requirements. Frequently, emulated structures exhibit linear responses at the beginning. Therefore, the computational cost at the beginning is lower with respect to the subsequent simulation in the presence of nonlinearities. From this point of view, the adoption of the partitioned method with no subcycling is feasible and does not affect the applicability range of this novel partitioned method.

7.3 Accuracy analysis

Differently from the staggered partitioned methods, the parallel methods require special starting procedures with the same or higher order of accuracy to provide initial solutions. As a consequence, the accuracy analysis for the parallel methods needs to take into account not only the local truncation error induced by the regular process but also the initialization error resulting from the starting procedure.

Since the number of steps in the starting procedure is finite, the propagation error is limited providing that the utilized time step is small. Therefore, the initialization error analysis is omitted here. The truncation error of the parallel procedure can be analysed with the same strategies as the staggered cases. Here, for simplicity, we only present the analysis for the parallel method based on the LSRT2 integrator.

Let us first analyse the integration of subdomain A and assume that the initial solutions for the parallel procedure, $\mathbf{y}_{i-2}^{A}$, $\mathbf{y}_{i-2}^{B}$ and $\mathbf{y}_{i}^{A}$, are exact. Similarly, the Lagrange multiplier Λ_{i-2} achieved at time t_{i-2} is also exact.

With the exact solutions and the exact Lagrange multiplier vector, the solution obtained at time t_i in the first stage is locally second-order accurate, i.e.,

$$\tau_{i}^{y_A} = \mathrm{O}\left(\Delta t^2\right) \tag{7.15}$$

Likewise, the Lagrange multiplier vector Λ_i at time t_i is also locally second-order accurate. Similarly to subsection (6.3.4), we can prove that the solutions t_{i+2} in the second stage is locally third-order accurate, i.e.,

$$\tau_{i+2}^{y_A} = \mathrm{O}\left(\Delta t^3\right) \tag{7.16}$$

Now let us start the integration of subdomain B and similarly suppose that the initial solutions, $\mathbf{y}_{i}^{A}$, $\mathbf{y}_{i}^{B}$ and $\mathbf{y}_{i+1}^{A}$, are exact. Then, the solution of subdomain B obtained at the end of the first stage can be proven to be locally second-order accurate. The internal solutions between time t_i and t_{i+1} obtained by linear interpolation, $\mathbf{y}_{i+(j-1)/ss}^{A}$ and $\mathbf{y}_{i+(j-1/2)/ss}^{A}$, are locally second-order accurate. Therefore, the Lagrange multiplier vector obtained at the end of the first stage of the first substep in subdomain B is locally second-order accurate. Then, the solution obtained at the end of the first substep is locally third-order accurate. In the same manner, the solution at the end of this loop in subdomain B can be proven to be locally third-order accurate. Hence, we can conclude that the local truncation error has the same order of accuracy as the staggered case. However, the accuracy of these numerical methods is not only determined by the order p but also the constant C, as shown in the terminology $C(\Delta t^p)$. In order to efficiently compare the accuracies of different cases, a numerical accuracy analysis will be carried out in the following sections.

Similarly to the staggered partitioned methods, the classical stability analysis approaches, such as the model decomposition approach and the energy approach, are not suitable for the parallel partitioned methods. In fact, stability is not only related to the overall system, but even for the linear system it depends on the partitions. Therefore, the performances of the parallel partitioned methods are investigated by means of spectral stability analysis applied to a Single-DoF split mass system.

7.4.1 Spectral stability analysis

For the parallel partitioned methods, let us consider again a Single-DoF split mass system. Since the LSRT2-based method is relatively more complicated and more desirable in practice, we mainly deal with the method based on the LSRT2 algorithm and only some results will be represented for the LSRT1-based method. The parallel partitioned methods are not self-starting and therefore the choice of sufficient state variables for the stability analysis is the main difficulty. It is of crucial importance to choose sufficient variables. Here, two codes are taken into account: i) the output x_{i+1} and the input x_i are parallel as depicted in Fig. 7.5; ii) the output block contains all the target variables while the input block includes all the initial variables in one loop of the solution procedure. The initial solutions for one loop are composed of $\mathbf{y}_{i-2}^A$, $\mathbf{y}_{i-2}^B$, $\mathbf{y}_i^A$, $\mathbf{y}_i^B$ and $\mathbf{y}_{i+1}^A$ while the output contains only $\mathbf{y}_{i+1}^B$ and $\mathbf{y}_{i+2}^A$. In order to study the amplification factor of the loop, another pair of solutions $\mathbf{y}_{i-1}^A$, $\mathbf{y}_{i-1}^B$, as intermediates, are also considered. As a consequence, the state vector involved in the spectral stability analysis can be considered as follows:

$$\mathrm{X}_i = \begin{pmatrix} (\mathbf{y}_{i-2}^A)^T & (\mathbf{y}_{i-2}^B)^T & (\mathbf{y}_{i-1}^A)^T & (\mathbf{y}_{i-1}^B)^T & (\mathbf{y}_i^A)^T & (\mathbf{y}_i^B)^T & (\mathbf{y}_{i+1}^A)^T \end{pmatrix}^T \quad (7.17)$$

where $\mathbf{y}_i^A$ collects the kinematic quantities of subdomain A

$$\mathbf{y}_i^A = \begin{pmatrix} u_i^A & \dot{u}_i^A \end{pmatrix}^T \quad (7.18)$$

and similarly

$$\mathbf{y}_i^B = \begin{pmatrix} u_i^B & \dot{u}_i^B \end{pmatrix}^T \quad (7.19)$$

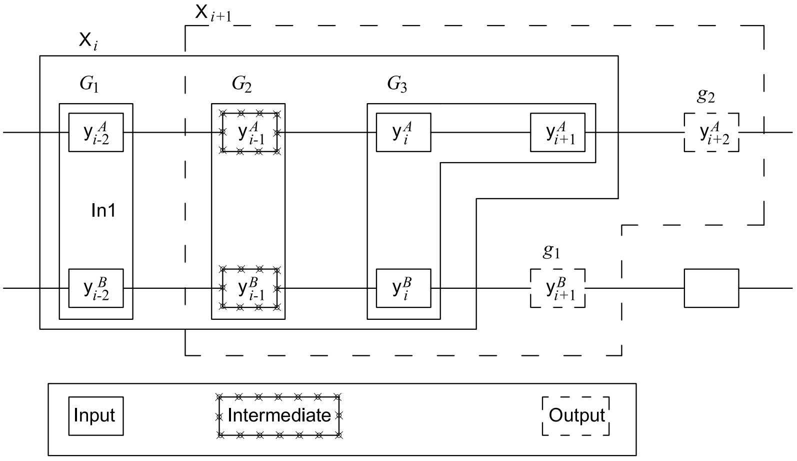


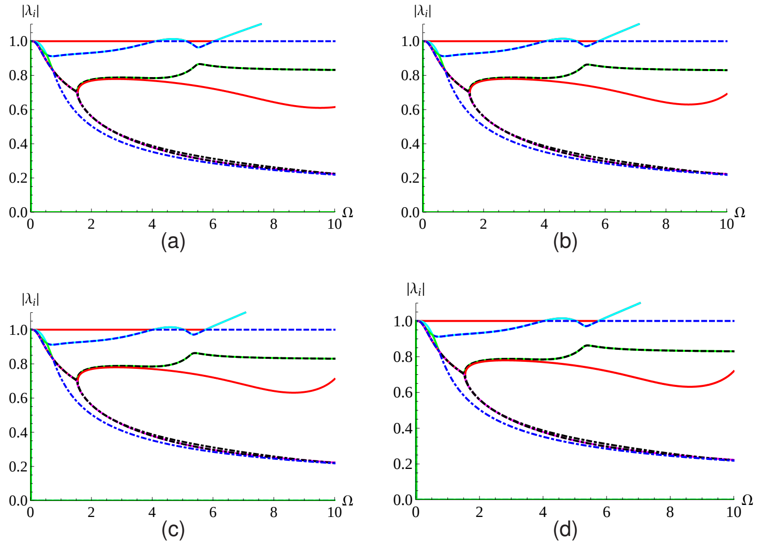
Fig. 7.5: The input and output representation of the integration loop for the interfield parallel solution procedure of the LSRT2-based partitioned method

As a result, X_i generally has the dimension $8n_A + 6n_B$ with n_A and n_B the DoFs in the two subdomains. For the Single-DoF split mass system, the dimension is 14. The application of the LSRT2-based method leads to the following recurrence formula:

$$X_{i+1} = \begin{bmatrix} 0_{4\times4} & \mathbf{I}^{G_2}_{4\times4} & 0_{4\times6} \\ 0_{6\times4} & 0_{6\times4} & \mathbf{I}^{G_3}_{6\times6} \\ 0_{2\times4} & 0_{2\times4} & \mathbf{R}^{g_1/G_3}_{2\times6} \\ \mathbf{R}^{g_2/G_1}_{2\times4} & 0_{2\times4} & \mathbf{R}^{g_2/G_3}_{2\times6} \end{bmatrix} X_i \qquad (7.20)$$

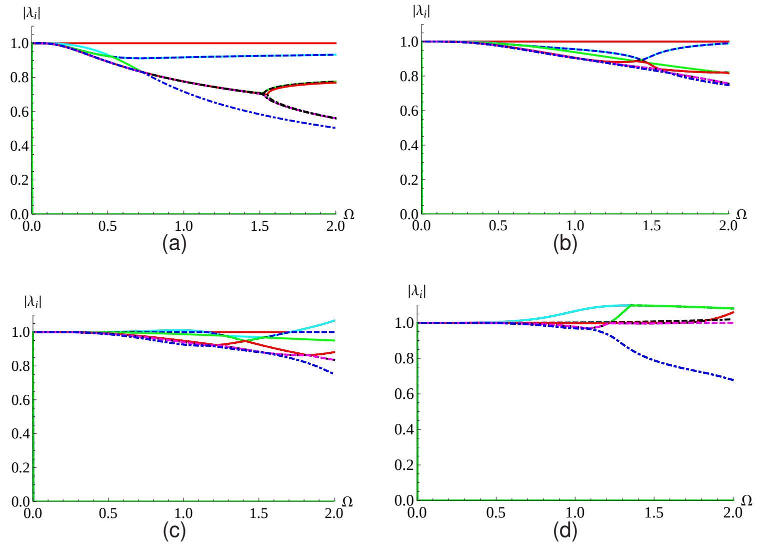
Note that the amplification matrix is written in such a way that allows us to explain the causes of special eigenvalues. In order to clearly represent the amplification, the input and output solutions are classified into different groups as depicted in Fig. 7.5. In the amplification matrix, **I**s are identity matrices that stand for the solutions in the overlapped domain while **R**s are non-zeroes that represent the actual recurrence involved in the parallel loop.

With respect to the Single-DoF system endowed with the system characteristics addressed in Subsection 6.4.1, the absolute values of the eigenvalues of the amplification matrices are plotted in Figs. (7.6-7.10).



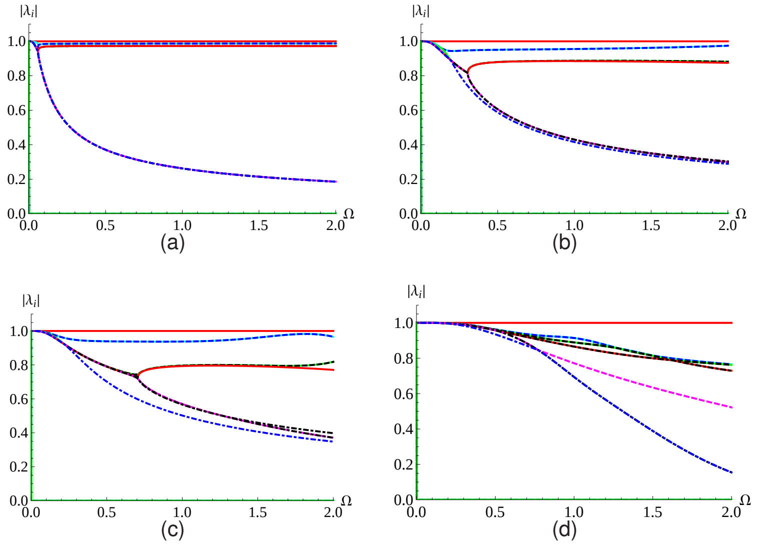
(a) (b) (c) (d)

Fig. 7.6: $|\ \lambda_i\ |$ for the parallel partitioned method integrated by LSRT2 with $b_1 = 0.1$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$: (a) ss = 2; (b) ss = 5; (c) ss = 10; (d)ss = 50



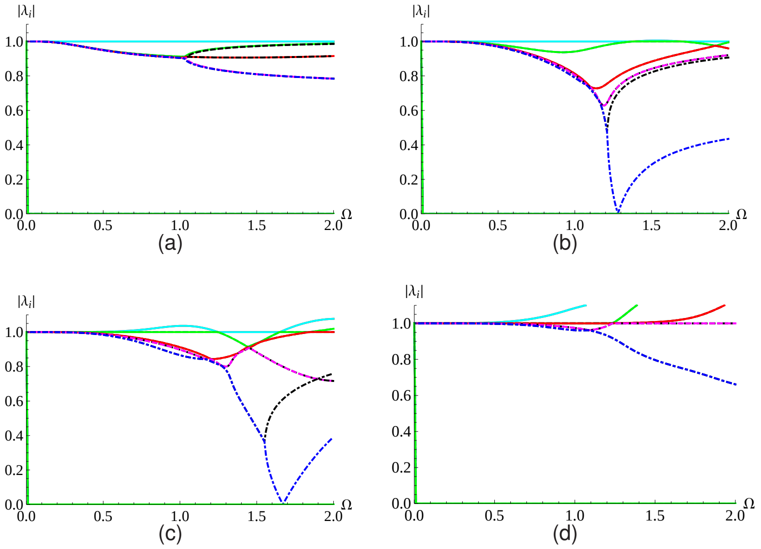
(a) (b) (c) (d)

Fig. 7.7: $|\ \lambda_i\ |$ for the parallel partitioned method integrated by LSRT2 with ss = 10 and $\gamma = 1 - \frac{\sqrt{2}}{2}$: (a) $b_1 = 0.1$; (b) $b_1 = 0.5$; (c) $b_1 = 1$; (d)$b_1 = 10$



(a) (b) (c) (d)

Fig. 7.8: $|\lambda_i|$ for the parallel partitioned method integrated by LSRT2 with $ss = 10$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$: (a) $b_1 = 0.1$; (b) $b_1 = 0.5$; (c) $b_1 = 1$; (d)$b_1 = 10$



(a) (b) (c) (d)

Fig. 7.9: $|\lambda_i|$ for the parallel partitioned method integrated by LSRT2 with $ss = 10$ and $\gamma = 1/4$: (a) $b_1 = 0.1$; (b) $b_1 = 0.5$; (c) $b_1 = 1$; (d)$b_1 = 10$

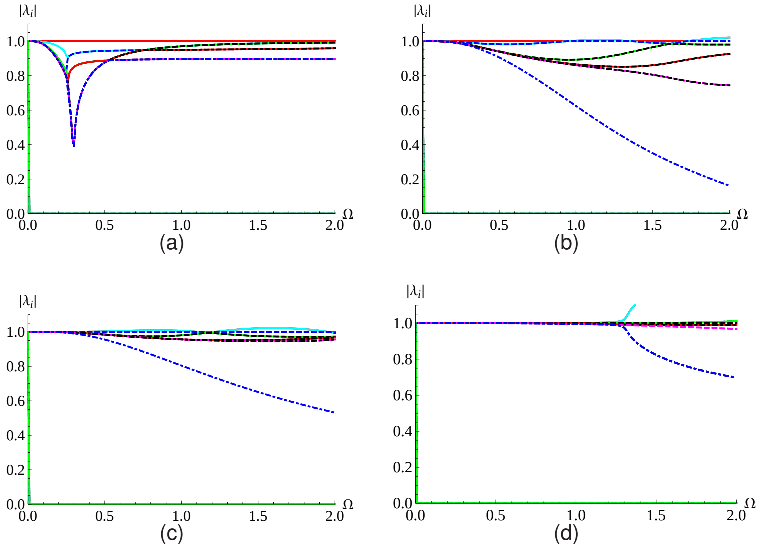


Fig. 7.10: $\mid \lambda_i \mid$ for the parallel partitioned method integrated by LSRT2 with $ss = 10$ and $\gamma = 1/2$: (a) $b_1 = 0.1$; (b) $b_1 = 0.5$; (c) $b_1 = 1$; (d)$b_1 = 10$

The number of nonzero eigenvalues is found to be 10: one is unitary, 4 pairs of them are complex conjugate and the other one is frequently less than 1. Besides, four zero eigenvalues are included. First of all, let us investigate the existence of the zero eigenvalues. If we have a deep look at the amplification matrix defined in 7.20, we can find that the third term of the first column is originally $\mathbf{R}_{2\times4}^{g_2/G_1}$ but here is zero block. This is because the solution $\mathbf{y}_{i+1}^{B}$ is not dependent on the initial solution $\mathbf{y}_{i-2}^{A}$ and $\mathbf{y}_{i-2}^{B}$. For this reason, there are two zero eigenvalues and two corresponding eigenvectors. Let us suppose that one of the corresponding eigenvector is:

$$\xi_1 = \begin{pmatrix} \psi_{1\times4} & \mathbf{0}_{1\times10} \end{pmatrix}^T \tag{7.21}$$

As a result of the intermediate block, we can obtain the following equation

$$\mathbf{R}\begin{pmatrix} \mathbf{0}_{1\times4} & \psi_{1\times4} & \mathbf{0}_{1\times6} \end{pmatrix}^T = \begin{pmatrix} \psi_{1\times4} & \mathbf{0}_{1\times10} \end{pmatrix}^T \tag{7.22}$$

If we assume $\xi_1 = \begin{pmatrix} \mathbf{0}_{1\times4} & \psi_{1\times4} & \mathbf{0}_{1\times6} \end{pmatrix}^T$, we can finally achieve:

$$\begin{aligned} \mathbf{R}\xi_1 &= 0\xi_1 \\ \mathbf{R}\xi_2 &= 0\xi_2 + \xi_1 \end{aligned} \tag{7.23}$$

and, therefore,

$$\mathbf{R}^2\xi_2 = \mathbf{R}\xi_1 = 0 \tag{7.24}$$

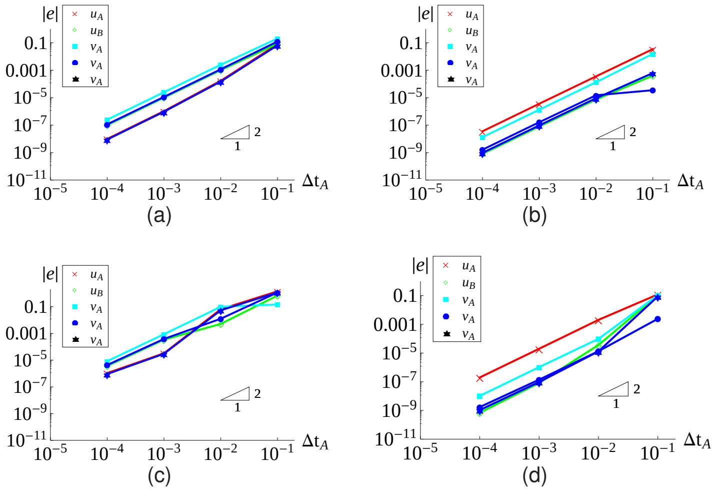
Accordingly, the eigenvector ξ_2 is the eigenvector of the amplification matrix $\mathbf{R}$ with respect to one zero eigenvalue, while the vector ξ_2 is the so-called associated vector. In the same manner, we can analyse the other zero eigenvector and find its corresponding associated vector. Generally, the number of the zero eigenvalues is determined by the DoFs of subdomain B and therefore is equal to $4n_B$. Finally, we can conclude that the zero eigenvalues are caused by *redundant* inputs involved in the construction of the amplification matrix and they do not affect the accuracy of the methods. Note that the so-called *redundant* is not related to the necessary condition for the integration but the *redundant* solutions are inevitable to achieve this amplification matrix.

As for the unit eigenvalue, we can also figure out the corresponding eigenvector from the "equilibrium point" of the Index-one DAEs system

$$\xi = \left\{ \begin{array}{cccccccccccccccccc} k_B & 0 & -k_A & 0 & k_B & 0 & -k_A & 0 & k_B & 0 & -k_A & 0 & k_B & 0 & -k_A & 0 & k_B & 0 \end{array} \right\}^T \tag{7.25}$$

Although the parallel procedure is relatively more complex with respect to the staggered case, we can also, by following the procedure, achieve $\mathbf{R}\xi = \xi$. As a result, the unitary eigenvalue is always existent whatever the time step and the parameter γ are used. In addition, there are 9 eigenvalues which are not constant. Among them, 4 pairs are complex conjugate and one is real. Note that the composition of the eigenvalues discussed here is beyond the consideration of bifurcations.

Fig. 7.6 shows the absolute values of the eigenvalues of the amplification matrices for the cases with different values of the parameter *ss*. Notice that all the cases in Fig. 7.6 have an excellent similarity. This is because, with different *ss*, the only difference exists in the $\mathbf{R}_{2\times 6}^{g_1/G_3}$ block of the amplification matrix while all the other blocks remain the same. Therefore, we only elaborate the cases with $ss = 10$ which are relatively common, only considering $\Omega \in [0, 2]$. Similarly to the staggered methods, the method with $\gamma = 1 + \frac{\sqrt{2}}{2}$ exhibits unconditional stability while the one with $\gamma = 1 - \frac{\sqrt{2}}{2}$ sometimes is only conditionally stable. Nevertheless, we also must be aware that all the methods are stable when the time step is sufficient small. Moreover, we would like to point out a



(a)

(b)

(c)

(d)

Fig. 7.11: Global error of the Single-DoF system for the parallel partitioned method integrated by LSRT2 with $ss = 10$: (a) $b_1 = 0.1$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $b_1 = 10$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c) $b_1 = 0.1$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (d)$b_1 = 10$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$

trend that must be taken into account to ensure the stability of the parallel integration: the stability of the partitioned methods depends on the partitioning parameter b_1. Larger value of b_1 introduces more damping, and therefore the method becomes more stable.

7.4.2 Numerical convergence analysis

For the parallel partitioned methods, the global error e_k at time t_k is determined by both the starting procedure and the regular parallel procedure. Although the local truncation errors for both procedures are known, it is not immediately obvious how the errors will accumulate over k time steps. In order to cope with this we conduct the numerical convergence analysis.

Figs. 7.11 and 7.12 show the global error versus for the Single-DoF split mass system, with the initial condition $d_0 = 0$ and $v_0 = 1$. Numerical results coincide well with the theoretical accuracy analysis: the LSRT2-based method is globally second-order accurate. In addition, the comparison of the cases with different algorithm parameter γ and with different partitioning parameter b_1 shows that the method with $\gamma = 1 + \sqrt{2}/2$

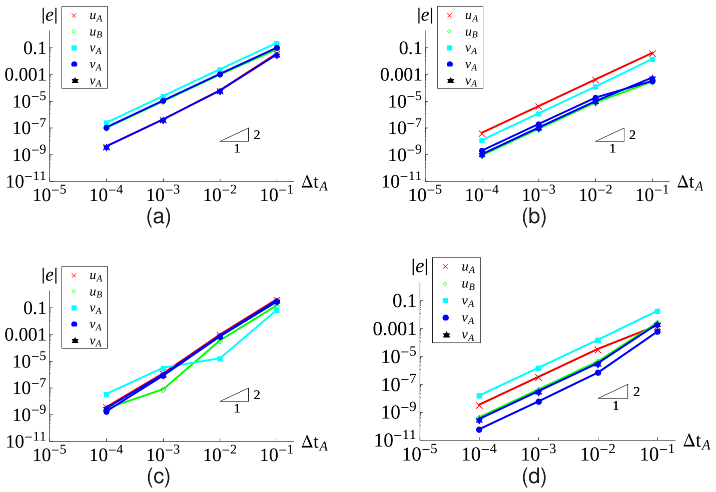
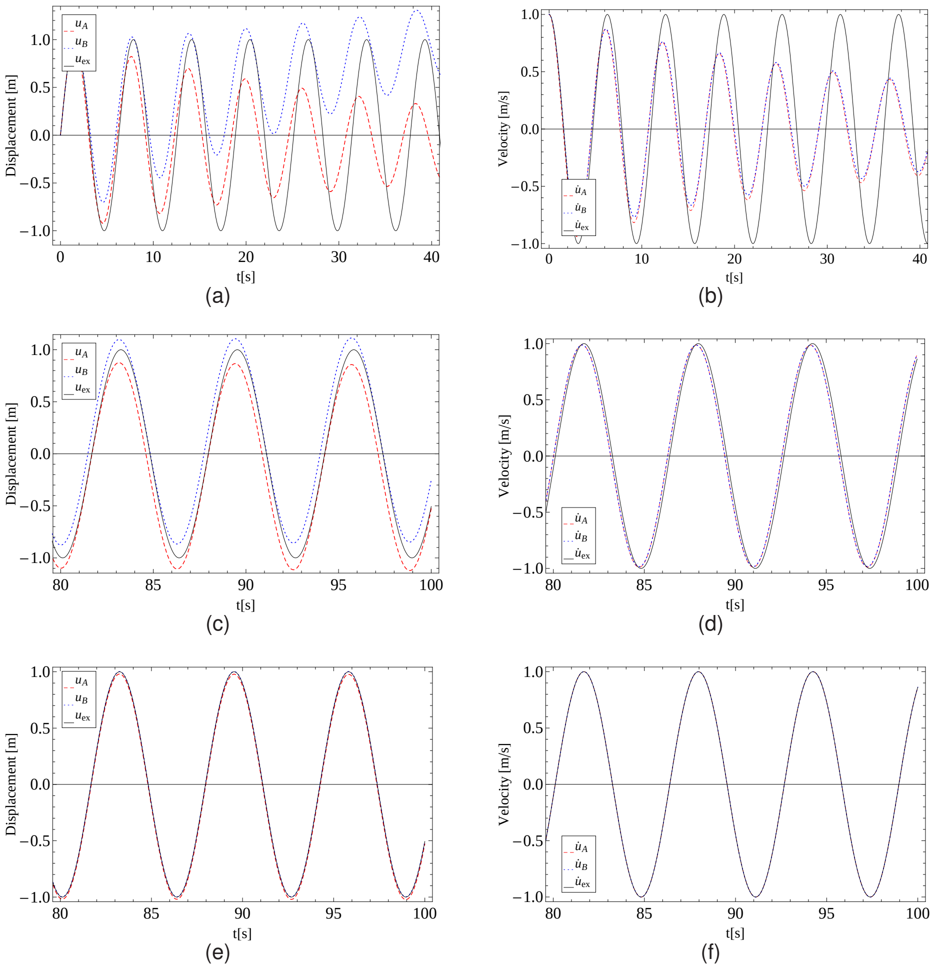


Fig. 7.12: Global error of the Single-DoF system for the parallel partitioned method integrated by LSRT2 with ss = 10: (a) b_1 = 0.1 and $\gamma = 1/4$; (b) b_1 = 10 and $\gamma = 1/4$; (c) b_1 = 0.1 and $\gamma = 1/2$; (d)b_1 = 10 and $\gamma = 1/2$

is more dissipative in the low frequency range and therefore introduces more error than the method with $\gamma = 1 - \sqrt{2}/2$. As analysed in the previous subsection, higher numerical dissipation is benefit to the stabilization of the partitioned method. If the parameter $\gamma = 1 + \sqrt{2}/2$ is inevitable for stability limit, on the one hand, we can use a smaller time step for required accuracy; on the other hand, we can choose another suitable partitioning parameter b_1 to ensure the accuracy in the low frequency rang. Besides, the partitioned methods with $\gamma = 1/4$ and $\gamma = 1/2$ are simulated as shown in Fig. 7.12. Both cases have no evident high-frequency dissipation except the numerical dissipation due to the coupling at the interface. But they are relative more accurate than the cases with dissipative values of γ.

7.4.3 Numerical simulation

For the parallel methods, four pairs of complex conjugate eigenvalues are included in the stability analysis. The numerical damping and the frequency error are not easy to be depicted as the staggered methods. In addition, the inherent drift due to the acceleration constraint is also inevitable for the parallel method. That is why many



(a) (b)

(c) (d)

(e) (f)

Fig. 7.13: The numerical simulations the Single-DoF system integrated by the parallel method with $ss = 10$, $\Delta t = 0.01$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$: (a), (b) the displacement and velocity responses with $b_1 = 0.1$; (c), (d) the displacement and velocity responses with $b_1 = 1$; (e), (f) the displacement and velocity responses with $b_1 = 10$

simulations will be enumerated considering different parameters. With this purpose in mind, we take no account of the physical damping and external force. The simulations of the Single-DoF split mass system are shown in Figs. 7.13 - 7.15. The evident incoincidence of the numerical solution on the displacements with the theoretical solution depicted in Fig. 7.13 (a) concerns the oscillation decay and period delay as well as the drift at the interface. In case the of velocities (Fig. 7.13 b), the effect of the drift

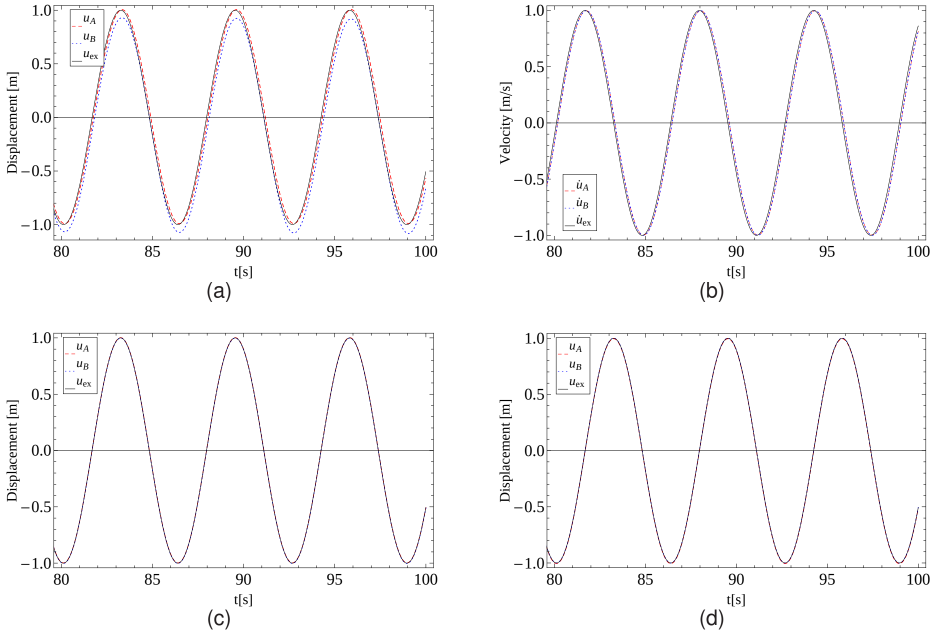
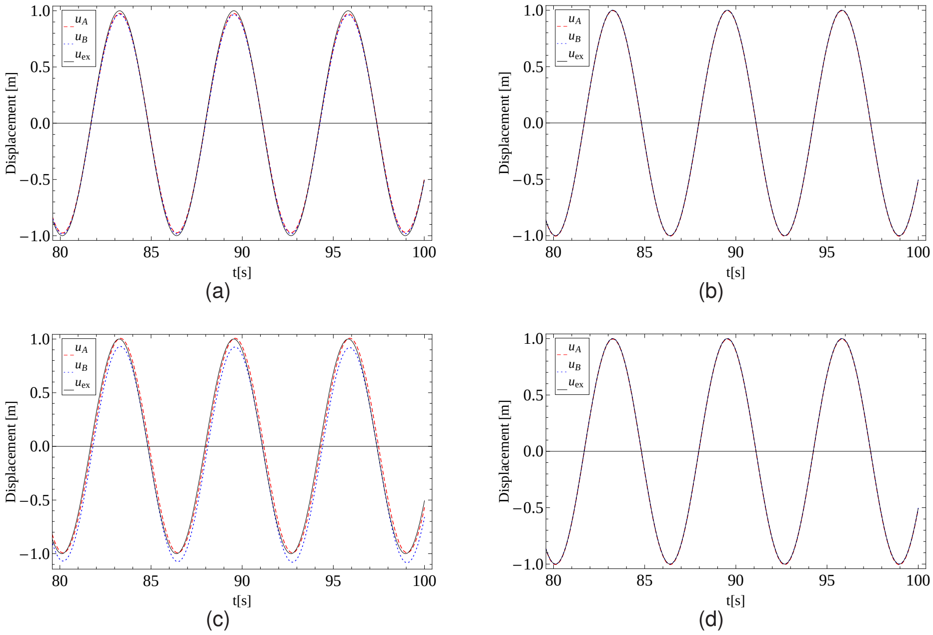


Fig. 7.14: The numerical simulations the Single-DoF system integrated by the parallel method with $ss = 10$, $\Delta t = 0.01$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$: (a), (b) the displacement and velocity responses with $b_1 = 0.1$; (c) the displacement response with $b_1 = 1$; (d) the displacement response with $b_1 = 10$



(a)

(b)

(c)

(d)

Fig. 7.15: The numerical simulations the Single-DoF system integrated by the parallel method with $ss = 10$ and $\Delta t = 0.01$: (a)the displacement response with $b_1 = 0.1$ and $\gamma = 1/4$; (b) the displacement response with $b_1 = 10$ and $\gamma = 1/4$; (c) the displacement response with $b_1 = 0.1$ and $\gamma = 1/2$; (d) the displacement response with $b_1 = 10$ and $\gamma = 1/2$

becomes unobservable while the other effects remain at the same level in the velocity responses. When we adopt the parameter $b_1 = 1$, the responses will be certainly improved. Now let us proceed with these calculations a little bit longer to compare the responses. As shown in Fig. 7.13 (c) and (d), the displacements reveal significant drift effect as well as a little period delay. Similarly, the drift effect disappears in the velocity while the period delay remains. For $b_1 = 10$, even when the calculations continue to 100s, the responses only exhibit slight difference in the displacement, which is mainly the contribution of the drift at the interface, and an excellent agreement for the velocity.

Compared with the cases with $\gamma = 1 - \frac{\sqrt{2}}{2}$, Fig. 7.14 simulated with $\gamma = 1 - \frac{\sqrt{2}}{2}$ has less error in both displacement and velocity responses, especially for velocity. The LSRT2 algorithm with $\gamma = 1 - \frac{\sqrt{2}}{2}$ for uncoupled system introduces less dissipation for the low frequency response. As shown in Fig. 7.14 (a), this disagreement is mainly induced by the drift at the interface. For the other two cases with $b_1 = 1$ and $b_1 = 10$, their displacement responses hardly exhibit noticeable differences while less for their velocity responses which are omitted for simplicity.

In addition, some simulations are carried out for the conservative values of $\gamma = 1/4$ and $\gamma = 1/2$. Their performances are plotted in Fig. 7.15. Even though the global integration with these values of γ has no dissipation at all, this only indicates zero amplification decay but not zero period delay. If $b_1 \neq 1$, the time variations in both subdomain are different. Moreover, the time steps used in different subdomain are unequal. Therefore, the integrations in both subdomains have different period delay. The unbalance period delay aggravates the deviation of the Lagrange multiplier from its exact solution and therefore induces the amplification decay as shown in Fig. 7.15 (a).

The parallel methods applied on a Single-DoF system can be explored further with more simulations, but this can not be extend to the Multiple-DoF system directly as the ODE solver. Before moving to numerical analysis for Multiple-DoF systems, let us make the following conclusions:

1. Using sufficient small time steps, all the cases involved are stable.
2. The methods with higher dissipative value of γ will be stable but may be inaccurate, especially when b_1 is not suitably chosen.

3. The drift effect is mainly related to be displacement response, its effect on the velocity response is relatively slight.

7.5 Numerical analysis and simulations for Multiple-DoF split mass systems

Both theoretical consideration and numerical examples have shown some problems in the Single-DoF system simulations: uncontrollable numerical dissipation and conditionally stability. But their effects can be reduced by considering proper choices of the parameters γ and b_1. Some examples are provided in this section to ensure that their effects are not aggravated with increasing the number of DoFs.

7.5.1 Spectral stability analysis

In this subsection, the partitioned Two-DoF system is considered. For this system, the number of the initial values provided for the parallel procedure is $8n_A + 6n_B = 20$. Fig. 7.16 depicts the absolute values of the eigenvalues of the amplification matrices. Similarly, the parallel method with $\gamma = 1 + \frac{\sqrt{2}}{2}$ indicates unconditionally stability while with $\gamma = 1 + \frac{\sqrt{2}}{2}$ only conditionally stability when $b_1 = 10$. Here, we can find that 8 of the eigenvalues are zero. This is because the number of DoFs in subdomain B is 2. The numerical stability for Multiple-DoF systems are investigated for the other cases which are not represented for simplicity. The results are similar: the method with $\gamma = 1 - \frac{\sqrt{2}}{2}$ sometimes is only conditionally stable. But the non-zero eigenvalues of the amplification matrix with $\gamma = 1 - \frac{\sqrt{2}}{2}$ is more approximate to one in modular which indicates higher accuracy.

7.5.2 Numerical simulations

In this subsection, the GC method, the PM method and the proposed staggered method will be implemented for comparison. First, let us consider the partitioned

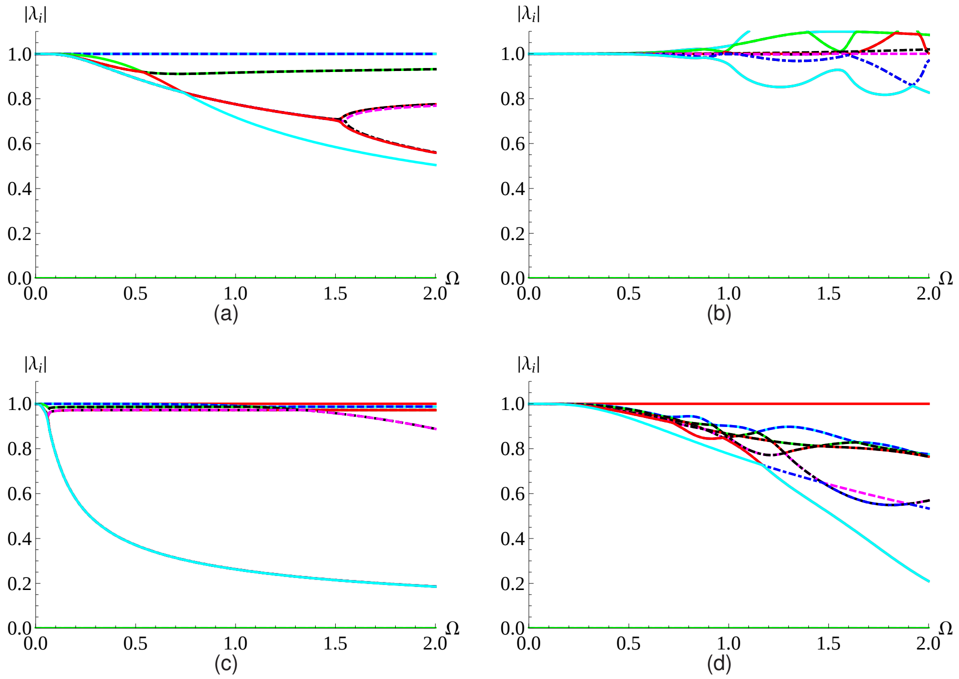
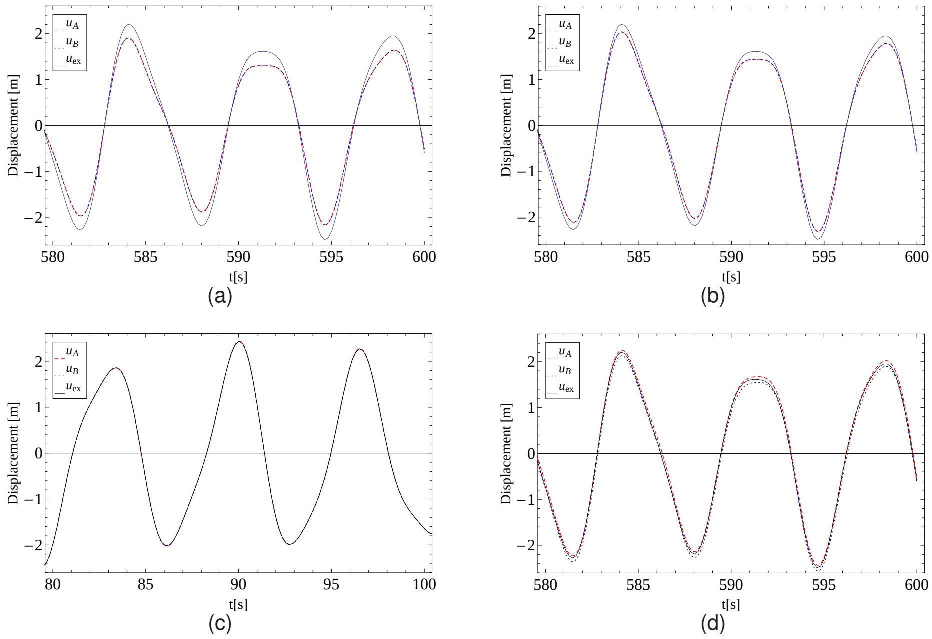


Fig. 7.16: $\mid \lambda_i \mid$ for the parallel method appled to the paritioned Two-DoF system 1 with LSRT2 and ss = 10: (a) b_1 = 0.1 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) b_1 = 10 and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c) b_1 = 0.1 and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (d) b_1 = 10 and $\gamma = 1 + \frac{\sqrt{2}}{2}$



(a) (b) (c) (d)

Fig. 7.17: The displacement responses of the partitioned Two-DoF system 1 integrated with: (a) the GC method; (b) the PM method; (c) the LSRT2-based staggered method; and the LSRT2-based parallel method

Two-DoF system 1. All the methods are implemented with time step $\Delta t = 10ms$. Here, the system variables are chosen the same as before and the parameter $b_1 = 1$. For the proposed methods, staggered or parallel, we adopt the parameter $\gamma = 1 - \frac{\sqrt{2}}{2}$. For the GC method and the PM method, we use the parameters $\beta = 1/4$ and $\gamma = 1/2$ for both subdomain. Fig. 7.17 and 7.18 collect the numerical solutions of the period from $580ms$ to $600ms$ in order to highlight the different numerical properties of the partitioned methods. The LSRT2-based parallel method exhibits more drift effect in the displacement responses than the other methods. This is consistent with the results for the Single-DoF system in the previous section. For the GC method and the PM method, no noticeable drifts are included in the displacement responses, but the deviations of the displacement responses from the exact solutions are more or less to the same extent as the proposed parallel method. Moreover, the proposed parallel methods generates more accurate velocity solutions with no visible drift and deviation while the GC method and the PM method also lead to noticeable errors in velocity re-

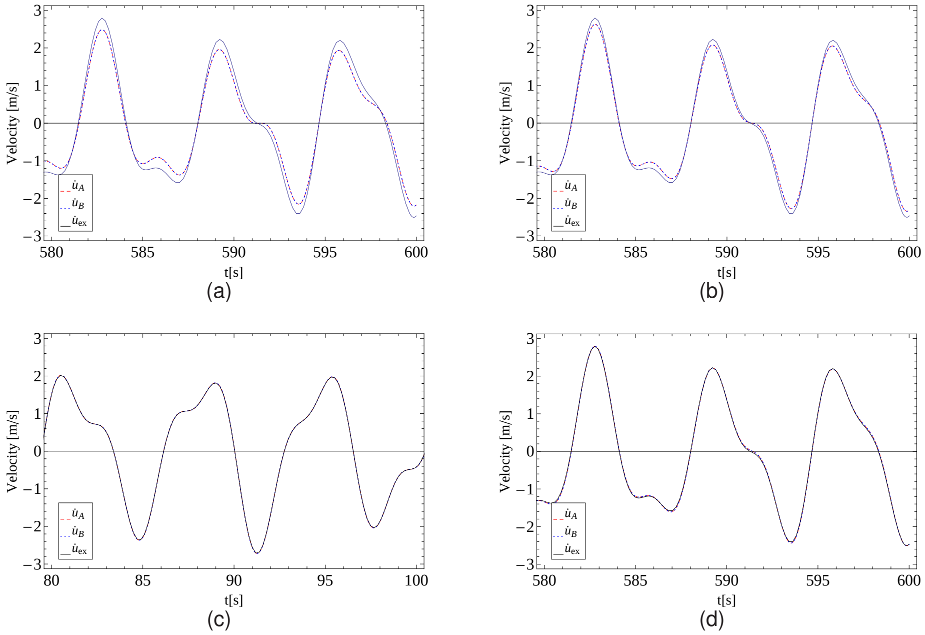


Fig. 7.18: The velocity responses of the partitioned Two-DoF system 1 integrated with: (a) the GC method; (b) the PM method; (c) the LSRT2-based staggered method; and the LSRT2-based parallel method

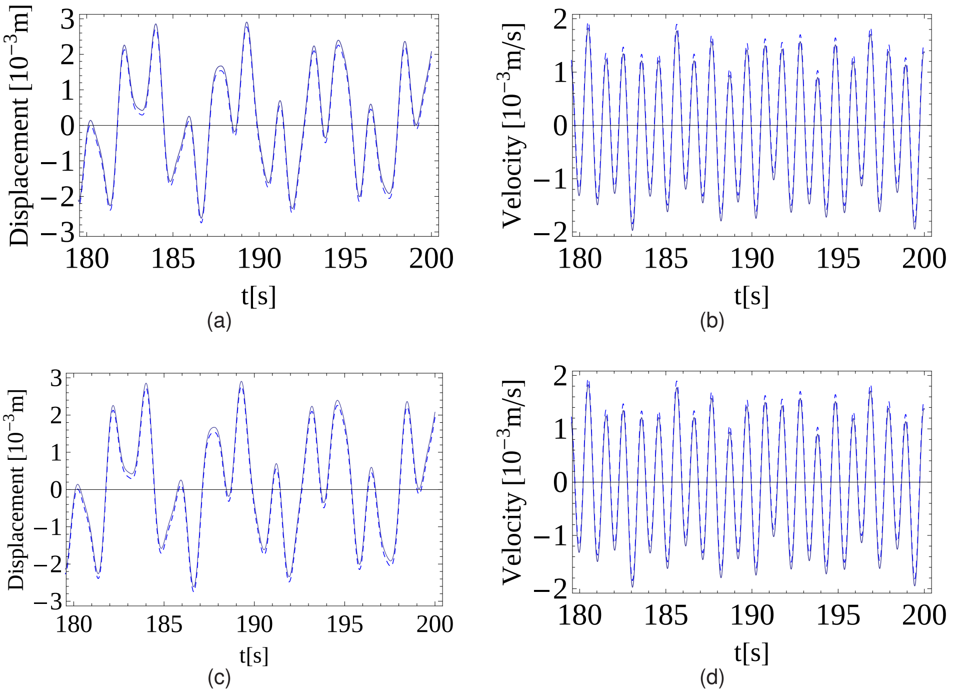


Fig. 7.19: The displacement and velocity responses of the partitioned Three-DoF system integrated with: (a) and (b) the LSRT2-based staggered method; (c) and (d) the LSRT2-based parallel method

sponses. In addition, the LSRT-based staggered method provides the solutions with excellent agreement both for the displacement and velocity responses.

Secondly, we perform a series of simulations of the partitioned Three-DoF system. Similarly, the time step $\Delta t = 10ms$ is used and the system characteristics is chosen the same as Subsection 6.6.1. The numerical solutions simulated by the LSRT-based methods, staggered and parallel, with $\gamma = 1 - \frac{\sqrt{2}}{2}$ are reported in Fig. 7.19. Differently from the simulations for the Single-DoF system, the velocity responses solved by both methods also reveal slight drift effect. This drift effect is not only because of the free mass model, but also due to the weak instability induced by the floating DoFs. Even though only one floating subdomain is included in the system, two floating DoFs are actually hidden in Subdomain B: one is the horizontal motion, the other one is the rotational motion. Nevertheless, we also must be aware that the numerical solutions are still within an acceptable range if the simulation period is not too long.

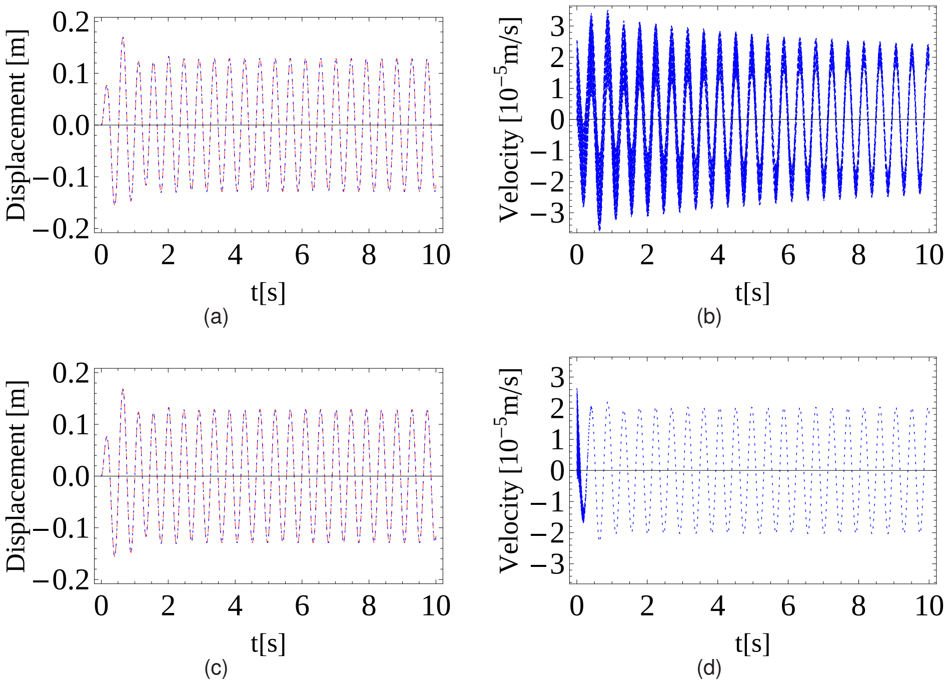


Fig. 7.20: The displacement and velocity responses of the coupled spring-pendulum system integrated by the LSRT2-based parallel method with: (a) and (b) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c) and (d) $\gamma = 1 + \frac{\sqrt{2}}{2}$

Finally, we perform a couple of simulations of the coupled spring-pendulum system in order to check the possibility of the nonlinear application of the LSRT2-based parallel method. The same stiff problem is solved here with the time step $\Delta t = 1ms$ for the purpose to investigate the high-frequency dissipation property of the Rosenbrock-based partitioned method. Fig. 7.20 shows that the displacement responses remain excellent and no drift effect is visible. The difference in the velocity responses reveals the high-frequency dissipation property of the method. Similar, the method with $\gamma = 1 + \frac{\sqrt{2}}{2}$ introduces higher numerical damping to the stiff problem.

7.6 Conclusions

The partitioned methods proposed in the previous chapter are of great potential for solving complex structures, especially when subcycling is used. However, the methods with subcycling are inherently staggered, in the sense that all the subdomains have to be advanced sequentially, and not simultaneously at each time step. More precisely, the time integration with the fine time step requires the information (the internal solution interpolated from the integration with the coarse time step) which is not available at the beginning of the time step. This limits the application of the method to some parallel computations, such as, real-time substructure testing.

In this chapter, we have focused exclusively on their extension to the parallel case. The accuracy analysis for the staggered method provides us a powerful access to implementation of parallel solution procedures without loss of the order of accuracy. For the sake of lack of analytical investigation, we present a mathematical framework for assessing some important numerical properties of the parallel methods, such as stability and accuracy. Our numerical framework is based on the simulations on a Single-DoF test problem, and then some Multiple-DoF test problems considering floating subdomains, multiple Lagrange multipliers and nonlinearities are utilized to test the numerical behaviour of parallel method.

In particular, numerical stability analysis was carried out by means of the spectral analysis. The amplification matrix was based on two assumptions: i) the output x_{i+1} and the input x_i are parallel; ii) the output includes all the target variables while the input includes all the initial variables in one loop of the solution procedure. The number of the eigenvalues of the obtained amplification matrix was 18 for a Single-DoF split mass problem. All the eigenvalues were classified into four groups, i.e. principle, unitary, zero and real, and they were analysed in terms of their causes and their effects on the stability and accuracy.

In addition, it was noticed that the methods exhibited drift effects. The drift effects were primarily caused by the acceleration constraint and were observed to be limited. When solving a partitioned system in the presence of floating subdomains or DoFs, the drift effect was additionally induced by the weak instability, and could not be bounded with the time approximating to infinity. With careful choices of the parameter γ and the utilized time step, the drift effect of both cases could be reduced to an acceptable level. In conclusion, the parallel methods were discovered to offer sufficiently accurate numerical solutions and preserve some numerical properties, such as high-frequency dissipation, as their progenitor methods.

CHAPTER 8

PARTITIONED TIME INTEGRATION METHODS BASED ON PROJECTION METHODS

8.1 Introduction

In this chapter, partitioned time integration methods are formulated by considering a projection onto the velocity constraint. The methods proposed in the last two chapters to some extent introduce slight but unavoidable drift-off effects in the displacement responses because of the acceleration constraint. Therefore, we develop an alternative partitioned method based on the velocity constraint.

Often a DAEs system is implicitly solved inside the framework of projection methods. In Yen et al. (1998), the HHT-α method was extended to DAEs holonomically constrained systems and projection methods were proposed to project the solution of the underlying ODEs onto the velocity and displacement constraints; but the DAE solvers were only first order convergent. A second order extension of the HHT-α method to the DAEs system was made by means of an implicit projection without relying on underlying ODEs (Jay and Negrut, 2007). Note that both of them are suitable for integrating DAEs systems, but not efficient in sequential or parallel computations. In recent years, a variety of explicit projection methods were developed. Actually, the GC method and the PM method belong to this type of projection method in that the free solution is obtained by integration of all subdomains without considering constraints, and then the link problem is solved to update the Lagrange multiplier and project the free solution onto the velocity constraint. Both methods provide the possibility for sequential and parallel computation, respectively, and second-order convergence can be achieved without subcycling, while first-order convergence with subcycling. In addition, Zheng and Petzold (2006) proposed an explicit projection method to solve first-order DAEs systems by means of a first-order integrator, Runge-Kutta-Chebysheve method. The method is second order accurate for the state variable and the Lagrange multiplier, and is suitable for parallel computation. In this chapter, a couple of projection methods based on first order integrators, i.e. the LSRT1 method and the LSRT2 method, are applied to second-order systems. We show that they preserve the order of convergence and the favourable properties of their progenitor methods.

8.2 Formulation of the projection methods

Motivated by the projection method developed by Zheng and Petzold (2006), we develop two partitioned methods with the LSRT1 and LSRT2 methods, respectively, for solving second order equations of motion in first-order form constrained by the inter-subdomain velocity continuity.

$$\begin{cases} \dot{\mathbf{y}} = \mathbf{F}(\mathbf{y}, t) + \mathbf{C}^T\mathbf{\Lambda} \\ \mathbf{C}\mathbf{y} = 0 \end{cases} \tag{8.1}$$

which is an index-2 DAEs system.

8.2.1 Derivation of the LSRT1-based projection method (**VPNR1**)

For the application of the LSRT1 method which is characterized by only one stage, the general projection procedure (Brown et al., 2001) can be followed. The numerical procedure consists of the following three steps:

1. Solve for the intermediate solution at time t_k

$$\begin{aligned} \mathbf{k}_1 &= \Delta t\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\left(\mathbf{F}(\mathbf{y}_k, t_k) + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right) \\ \mathbf{y}^*_{k+1} &= \mathbf{y}_k + \mathbf{k}_1 \end{aligned} \tag{8.2}$$

2. Perform the projection

$$\begin{aligned} \Phi_{k+1} &= -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right)^{-1}\mathbf{C}\mathbf{y}^{*}_{k+1} \\ \mathbf{y}_{k+1} &= \mathbf{y}^{*}_{k+1} + \mathbf{A}^{-1}\mathbf{C}^{T}\Phi_{k+1} \end{aligned} \tag{8.3}$$

3. Update the Lagrange multiplier

$$\mathbf{\Lambda}_{k+1} = \mathbf{\Lambda}_{k} + \frac{2}{\Delta t}\Phi_{k+1} \tag{8.4}$$

In the design of such a method, the key factor is the choice of the Lagrange multiplier update. Here, Eq. (8.4) is set to achieve first order accuracy for the Lagrange multiplier, which will be analysed in detail in Subsection 8.3.1.

8.2.2 Derivation of the LSRT2-based projection method (**VPNR2**)

The LSRT2 method is a two-stage method. Direct application to the projection procedure as proposed by Zheng and Petzold (2006) leads to instability for second order systems. This will be detailed analysed in Section 8.4. Here, another projection is performed at the end of the first stage per time step and the Lagrange multiplier is updated simultaneously. The projections involved are standard, with the same way as Eq. (8.3). How to update the Lagrange multipliers at both the end of the first stage and the end of the time step is the crucial point to design this method. This determines the accuracy and stability of the method. The details of the design of the method are postponed to Subsection 8.3.2 and Section 8.4. Here, the solution procedure of the method is presented as follows:

1. Solve for the intermediate solution at time $t_{k+1/2}$

$$\begin{aligned} \mathbf{k}_1 &= \Delta t\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\left(\mathbf{F}\left(\mathbf{y}_k, t_k\right) + \mathbf{A}^{-1}\mathbf{C}^{T}\mathbf{\Lambda}_k\right) \\ \mathbf{y}^{*}_{k+1/2} &= \mathbf{y}_k + \frac{1}{2}\mathbf{k}_1 \end{aligned} \tag{8.5}$$

2. Perform the projection

$$\begin{aligned} \Phi_{k+1/2} &= -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right)^{-1}\mathbf{C}\mathbf{y}^{*}_{k+1/2} \\ \mathbf{y}_{k+1/2} &= \mathbf{y}^{*}_{k+1/2} + \mathbf{A}^{-1}\mathbf{C}^{T}\Phi_{k+1/2} \end{aligned} \tag{8.6}$$

3. Update the Lagrange multiplier

$$\mathbf{\Lambda}_{k+1/2} = \mathbf{\Lambda}_k - \frac{1}{\Delta t}\Phi_{k+1/2} \tag{8.7}$$

4. Solve for the intermediate solution at time t_{k+1}

$$\begin{aligned} \mathbf{k}_2 &= \Delta t \left[\mathbf{I} - \Delta t \gamma \mathbf{J}\right]^{-1} \left(\mathbf{F}\left(\mathbf{y}_{k+1/2}, t_{k+1/2}\right) + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_{k+1/2} - \gamma \mathbf{J}\mathbf{k}_1\right) \\ \mathbf{y}^*_{k+1} &= \mathbf{y}_k + \mathbf{k}_2 \end{aligned} \tag{8.8}$$

5. Perform the projection

$$\begin{aligned} \Phi_{k+1} &= -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1} \mathbf{C}\mathbf{y}^*_{k+1} \\ \mathbf{y}_{k+1} &= \mathbf{y}^*_{k+1} + \mathbf{A}^{-1}\mathbf{C}^T\Phi_{k+1} \end{aligned} \tag{8.9}$$

6. Update the Lagrange multiplier

$$\mathbf{\Lambda}_{k+1} = \mathbf{\Lambda}_{k+1/2} + \frac{1}{\Delta t}\Phi_{k+1} \tag{8.10}$$

This solution procedure is parallel in the sense that the interface problem related to the projections and the updates of the Lagrange multiplier is only conducted at the end of each stage.

8.3 Accuracy analysis

8.3.1 Local truncation error analysis for the LSRT1-based projection method

In this section we will show that the projection method based on the LSRT1 method is locally second order accurate in time for both the state vectors and the coupling force. We consider one time step of the projection method from t_k to t_{k+1}. Suppose we have the exact solutions as the initial conditions at time t_k, i.e., $\mathbf{y}_k = \mathbf{y}(t_k)$ and $\mathbf{\Lambda}_k = \mathbf{\Lambda}(t_k)$. We can have the following equations satisfied:

$$\begin{aligned} \mathbf{C}\mathbf{y}_k &= 0 \\ \mathbf{C}\dot{\mathbf{y}}_k &= 0 \end{aligned} \tag{8.11}$$

In the first step of the projection method, the LSRT1 algorithm is used to solve for the equations of motion without regard to the constraint equations to obtain the intermediate state vector $\mathbf{y}^*_{k+1}$. In order to continue, we present another an auxiliary ODE with constant Lagrange multiplier.

$$\mathbf{A}\dot{\mathbf{y}}^* = \mathbf{F}(\mathbf{y}^*, t) + \mathbf{C}^T\mathbf{\Lambda}^* \tag{8.12}$$

where $\mathbf{\Lambda}^*$, the Lagrange multiplier vector, is assumed to be constant. We choose $\mathbf{\Lambda}^* = \mathbf{\Lambda}_k$. Therefore $\mathbf{y}^*_{k+1}$ is also the numerical solution of Eq. (8.12), and $\mathbf{y}^*(t_{k+1})$ is the exact solution of (8.12) at t_{k+1}. Since the LSRT1 method is first order accurate, we have

$$\mathbf{y}^*_{k+1} - \mathbf{y}^*(t_{k+1}) = \mathrm{O}\left(\Delta t^2\right) \tag{8.13}$$

The next step in the derivation is to compare $\mathbf{y}^*(t_{k+1})$ and $\mathbf{y}(t_{k+1})$ which are exact solutions of the auxiliary ODE and the original DAE, respectively. Herein, we begin by applying simple Taylor series analyses as follows:

$$\begin{aligned} \mathbf{y}^*(t_{k+1}) &= \mathbf{y}^*_k + \Delta t\dot{\mathbf{y}}^*_k + \frac{\Delta t^2}{2}\ddot{\mathbf{y}}^*_k + \mathrm{O}\left(\Delta t^3\right) \\ \mathbf{y}(t_{k+1}) &= \mathbf{y}_k + \Delta t\dot{\mathbf{y}}_k + \frac{\Delta t^2}{2}\ddot{\mathbf{y}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.14}$$

where $\mathbf{y}^*_k$ is supposed to be exact, i.e. $\mathbf{y}^*_k = \mathbf{y}_k$. And we assume that the auxiliary ODE is sufficient smooth, i.e.

$$\begin{aligned} \dot{\mathbf{y}}^*_k &= \dot{\mathbf{y}}_k = \mathbf{A}^{-1}\left(\mathbf{F}(\mathbf{y}_k, t_k) + \mathbf{C}^T\mathbf{\Lambda}_k\right) \\ \ddot{\mathbf{y}}_k &= \mathbf{A}^{-1}\left(\dot{\mathbf{F}}(\mathbf{y}_k, t_k) + \mathbf{C}^T\dot{\mathbf{\Lambda}}_k\right) \\ \ddot{\mathbf{y}}^*_k &= \mathbf{A}^{-1}\dot{\mathbf{F}}(\mathbf{y}_k, t_k) \end{aligned} \tag{8.15}$$

Thus, we can obtain

$$\begin{aligned} &\mathbf{y}(t_{k+1}) - \mathbf{y}^*(t_{k+1}) = \frac{\Delta t^2}{2}\left(\ddot{\mathbf{y}}_k - \ddot{\mathbf{y}}^*_k\right) + \mathrm{O}\left(\Delta t^3\right) \\ &= \frac{\Delta t^2}{2}\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.16}$$

By combining Eq.(8.16) with Eq. (8.13), we can find

$$\mathbf{y}^*_{k+1} - \mathbf{y}(t_{k+1}) = \mathrm{O}\left(\Delta t^2\right) \tag{8.17}$$

Before moving to the projection step, we can express the intermediate solution at t_{k+1} from Eq. (8.2) as follows:

$$\mathbf{y}^*_{k+1} = \mathbf{y}_k + \Delta t\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\dot{\mathbf{y}}_k \tag{8.18}$$

If both sides of (8.18) are left multiplied by **C**, we can obtain

$$\begin{aligned}
\mathbf{C}\mathbf{y}^*_{k+1} &= \mathbf{C}\mathbf{y}_k + \mathbf{C}\Delta t\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\dot{\mathbf{y}}_k \\
&= \mathbf{C}\Delta t\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\dot{\mathbf{y}}_k \\
&= \mathbf{C}\Delta t\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,(\mathbf{I} - \Delta t\gamma\mathbf{J} + \Delta t\gamma\mathbf{J})\,\dot{\mathbf{y}}_k \\
&= \Delta t\mathbf{C}\dot{\mathbf{y}}_k + \Delta t\mathbf{C}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\Delta t\gamma\mathbf{J}\dot{\mathbf{y}}_k \\
&= \Delta t\mathbf{C}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\Delta t\gamma\mathbf{J}\dot{\mathbf{y}}_k \\
&= \Delta t^2\gamma\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k + \Delta t^3\gamma^2\mathbf{C}\,[\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\,\mathbf{J}\mathbf{J}\dot{\mathbf{y}}_k \\
&= \Delta t^2\gamma\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k + \mathrm{O}\left(\Delta t^3\right)
\end{aligned} \tag{8.19}$$

In order to evaluate the term $\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k$, we introduce the continuity of the second derivative of the differential vector

$$\mathbf{C}\ddot{\mathbf{y}}_k = 0 \tag{8.20}$$

Moreover, the second derivative of the differential vector can be expressed as

$$\ddot{\mathbf{y}}_k = \frac{\partial}{\partial t}\left(\mathbf{A}^{-1}\mathbf{F}\,(\mathbf{y}_k, t) + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right) = \mathbf{J}\dot{\mathbf{y}}_k + \left.\frac{\partial\mathbf{F}}{\partial t}\right|_k + \mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}} \tag{8.21}$$

The combination Eq. (8.20) and (8.21) results in

$$\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k = -\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_k - \mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}} \tag{8.22}$$

Thus, we can continue Eq. (8.19) as follows,

$$\mathbf{C}\mathbf{y}^*_{k+1} = -\Delta t^2\gamma\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_k - \Delta t^2\gamma\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}} + \mathrm{O}\left(\Delta t^3\right) \tag{8.23}$$

Now, let us discuss the projection step.

$$\begin{aligned}
\Phi_{k+1} &= -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{y}^*_{k+1} \\
&= \Delta t^2\gamma\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_k + \Delta t^2\gamma\dot{\mathbf{\Lambda}} + \mathrm{O}\left(\Delta t^3\right)
\end{aligned} \tag{8.24}$$

Now, the local truncation error of the differential vector can be obtained by combining Eq. (8.17) and (8.24)

$$\mathbf{y}_{k+1} - \mathbf{y}\,(t_{k+1}) = \mathbf{y}_{k+1} - \mathbf{y}^*_{k+1} + \mathbf{y}^*_{k+1} - \mathbf{y}\,(t_{k+1}) = \Phi_{k+1} + \mathrm{O}\left(\Delta t^2\right) = \mathrm{O}\left(\Delta t^2\right) \tag{8.25}$$

which indicate that it is of second order.

Furthermore, the algebraic vector can be evaluated in term of local truncation error as

$$\begin{aligned}\mathbf{\Lambda}_{k+1} &= \mathbf{\Lambda}_k + \frac{2}{\Delta t}\left(\Delta t^2\gamma\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \Delta t^2\gamma\dot{\mathbf{\Lambda}} + \mathrm{O}\left(\Delta t^3\right)\right) \\ &= \mathbf{\Lambda}_k + 2\Delta t\gamma\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + 2\Delta t\gamma\dot{\mathbf{\Lambda}} + \mathrm{O}\left(\Delta t^3\right)\end{aligned} \tag{8.26}$$

Which is of only first order.

The method has different order of local truncation error in differential vector and algebraic vector. But the numerical convergence analysis performed in Subsection 8.5.2 indicates first order global error in differential vector. This is because the projection in next time step can filter the second order term in the algebraic vector, $2\Delta t\gamma\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + 2\Delta t\gamma\dot{\mathbf{\Lambda}}$. The similar phenomenon can be observed in the analysis of the LSRT2-based projection method and thereof the detailed investigation is omitted here.

8.3.2 Local truncation error analysis for the LSRT2-based projection method

In this section we will show that the projection method based on LSRT2 is locally third order accurate in time for the differential vector and locally second-order accurate for the algebraic vector. We consider one time step of the projection method from t_k to t_{k+1}. Suppose we have the exact solutions as the initial conditions at time t_k, i.e., $\mathbf{y}_k = \mathbf{y}(t_k)$, and $\Lambda_k = \Lambda(t_k)$. In the same way, we can have Eqs.(8.11) satisfied.

In the first stage, the LSRT2 algorithm is firstly used to solve for the equations of motion without regard to the constraint equations to obtain the intermediate state vector $\mathbf{y}^*_{k+1/2}$.

For the intermediate solution in the first stage, we present an auxiliary ODEs with the same form as (8.12), where the Lagrange multiplier in the first stage is assumed to be constant and equal to $\mathbf{\Lambda}_k$. Suppose $\mathbf{y}^*_{k+1/2}$ is the numerical solution of (8.12), and $\mathbf{y}^*\left(t_{k+1/2}\right)$ is the exact solution of the auxiliary ODEs at $t_{k+1/2}$. Because the LSRT2 method in the first stage is similar as the LSRT1 method and therefore the intermediate solution is of second order local truncation error, i.e.

$$\mathbf{y}^*_{k+1/2} - \mathbf{y}^*\left(t_{k+1/2}\right) = \mathrm{O}\left(\Delta t^2\right) \tag{8.27}$$

Then we deal with the projection at the end of first stage, and we try to compare the exact solutions of the original DAEs and the auxiliary ODEs. Herein, we begin by applying simple Taylor series analyses as follows:

$$\begin{aligned} &\mathbf{y}^*\left(t_{k+1/2}\right) = \mathbf{y}_k^* + \frac{\Delta t}{2}\dot{\mathbf{y}}_k^* + \frac{\Delta t^2}{8}\ddot{\mathbf{y}}_k^* + \mathrm{O}\left(\Delta t^3\right) \\ &\mathbf{y}\left(t_{k+1/2}\right) = \mathbf{y}_k + \frac{\Delta t}{2}\dot{\mathbf{y}}_k + \frac{\Delta t^2}{8}\ddot{\mathbf{y}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.28}$$

Similarly, initial solutions for both the auxiliary ODEs and the original DAEs can be expressed as Eqs. (8.15).

The comparison between Eqs. (8.28) leads to the difference between the exact solutions at t_{k+1} of the original DAEs and the auxiliary ODEs.

$$\begin{aligned} &\mathbf{y}\left(t_{k+1/2}\right) - \mathbf{y}^*\left(t_{k+1/2}\right) = \frac{\Delta t^2}{8}\left(\ddot{\mathbf{y}}_k - \ddot{\mathbf{y}}_k^*\right) + \mathrm{O}\left(\Delta t^3\right) \\ &= \frac{\Delta t^2}{8}\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.29}$$

Considering Eq. (8.1), the intermediate solution (8.5) can be written in the form

$$\mathbf{y}_{k+1/2}^* = \mathbf{y}_k + \frac{1}{2}\Delta t\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\dot{\mathbf{y}}_k \tag{8.30}$$

With the similar derivation for the LSRT1-based projection method, we can express the term $\mathbf{C}\mathbf{y}_{k+1/2}^*$ in the form

$$\mathbf{C}\mathbf{y}_{k+1/2}^* = \frac{1}{2}\Delta t^2\gamma\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k + \mathrm{O}\left(\Delta t^3\right) \tag{8.31}$$

As result of the continuity of the second derivative of the differential vector, i.e. $\mathbf{C}\ddot{\mathbf{y}}_k = 0$, the expression of (8.21), we can similarly obtain

$$\begin{aligned} &\mathbf{C}\mathbf{y}_{k+1/2}^* = \frac{1}{2}\Delta t^2\gamma\mathbf{C}\mathbf{J}\dot{\mathbf{y}}_k + \mathrm{O}\left(\Delta t^3\right) \\ &= -\frac{1}{2}\Delta t^2\gamma\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_k - \frac{1}{2}\Delta t^2\gamma\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.32}$$

By inserting (8.32) into Eq. (8.6a), we can obtain

$$\begin{aligned} &\Phi_{k+1/2} = -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{y}_{k+1/2}^* \\ &= \frac{1}{2}\Delta t^2\gamma\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_k + \frac{1}{2}\Delta t^2\gamma\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right) \end{aligned} \tag{8.33}$$

Finally, it follows form equation (8.33)and (8.27) that

$$\mathbf{y}_{k+1/2} - \mathbf{y}\left(t_{k+1/2}\right) = \mathrm{O}\left(\Delta t^2\right) \tag{8.34}$$

By inserting (8.33) into Eq. (8.7), the Lagrange multiplier vector can be expressed as

$$\begin{aligned}\mathbf{\Lambda}_{k+1/2} &= \mathbf{\Lambda}_k - \frac{1}{\Delta t}\Phi_{k+1/2} \\ &= \mathbf{\Lambda}_k - \frac{1}{2}\Delta t\gamma\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k - \frac{1}{2}\Delta t\gamma\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{8.35}$$

which is locally first order accurate in time to the exact solution.

Before analysing the second stage, some terms included in Eq. (8.8) are expressed by means of Taylor series expansion about the initial solution which are useful in the following proof process.

$$\begin{aligned}&\mathbf{F}\left(t_{k+1/2}, \mathbf{y}_{k+1/2}\right) = \mathbf{F}_k + \frac{\Delta t}{2}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \mathbf{J}\frac{\mathbf{k}_1}{2} + \mathrm{O}\left(\Delta t^2\right) \\ &\mathbf{k}_1 = \Delta t\mathbf{F}_k + \Delta t\mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k + \mathrm{O}\left(\Delta t^2\right)\end{aligned} \tag{8.36}$$

Following Eq. (8.8), the intermediate solution at t_{k+1} can be expressed as

$$\begin{aligned}&\mathbf{y}^*_{k+1} = \mathbf{y}_k + [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\left(\mathbf{F}_k + \frac{\Delta t}{2}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \frac{1}{2}\mathbf{J}\Delta t\mathbf{F}_k + \right. \\ &+\frac{1}{2}\mathbf{J}\Delta t\mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k - \frac{1}{2}\Delta t\gamma\mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k - \\ &\left.-\frac{1}{2}\Delta t\gamma\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k - \gamma\mathbf{J}\Delta t\mathbf{F}_k - \gamma\mathbf{J}\Delta t\mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right)\Delta t + \mathrm{O}\left(\Delta t^3\right)\end{aligned} \tag{8.37}$$

According to Taylor series expansion, we can express the exact solution at time t_{k+1} to the third order about the initial solution

$$\begin{aligned}&\mathbf{y}\left(t_{k+1}\right) = \mathbf{y}_k + \mathbf{F}_k\Delta t + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\Delta t + \mathrm{O}\left(\Delta t^3\right) + \\ &+\frac{\Delta t^2}{2}\left(\mathbf{J}\left(\mathbf{F}_k + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right) + \left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k\right)\end{aligned} \tag{8.38}$$

Therefore

$$\begin{aligned}&\mathbf{y}^*_{k+1} - \mathbf{y}\left(t_{k+1}\right) \\ &= [\mathbf{I} - \Delta t\gamma\mathbf{J}]^{-1}\left(\mathbf{F}_k + \frac{\Delta t}{2}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \frac{1}{2}\mathbf{J}\Delta t\mathbf{F}_k + \frac{1}{2}\mathbf{J}\Delta t\mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k - \right. \\ &\left.-\frac{1}{2}\Delta t\gamma\mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k - \frac{1}{2}\Delta t\gamma\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k - \gamma\mathbf{J}\Delta t\mathbf{F}_k - \gamma\mathbf{J}\Delta t\mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right)\Delta t - \\ &-\mathbf{F}_k\Delta t - \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\Delta t - \frac{\Delta t^2}{2}\left(\mathbf{J}\left(\mathbf{F}_k + \mathbf{A}^{-1}\mathbf{C}^T\mathbf{\Lambda}_k\right) + \left.\frac{\partial \mathbf{F}}{\partial t}\right|_k + \mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k\right) + \mathrm{O}\left(\Delta t^3\right) \\ &= -\frac{1}{2}\Delta t^2\gamma\mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{CA}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\left.\frac{\partial \mathbf{F}}{\partial t}\right|_k - \frac{1}{2}\Delta t^2\left(\gamma + 1\right)\mathbf{A}^{-1}\mathbf{C}^T\dot{\mathbf{\Lambda}}_k + \mathrm{O}\left(\Delta t^3\right)\end{aligned} \tag{8.39}$$

Left multiplying (8.39) by the term $\mathbf{A}^{-}1\mathbf{C}$ results in

$$\begin{aligned}&\mathbf{A}^{-1}\mathbf{C}\left(\mathbf{y}_{k+1}^{*}-\mathbf{y}\left(t_{k+1}\right)\right)\\&=-\frac{1}{2}\Delta t^{2}\gamma\mathbf{A}^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_{k}-\frac{1}{2}\Delta t^{2}\left(\gamma+1\right)\mathbf{A}^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\dot{\mathbf{\Lambda}}_{k}+\mathrm{O}\left(\Delta t^{3}\right)\end{aligned}\tag{8.40}$$

Since the $\mathbf{y}\left(t_{k+1}\right)$ is the exact solution, we have $\mathbf{A}^{-}1\mathbf{C}\mathbf{y}\left(t_{k+1}\right)=0$. Meanwhile, we can get

$$\mathbf{A}^{-1}\mathbf{C}\mathbf{y}_{k+1}^{*}=-\frac{1}{2}\Delta t^{2}\gamma\mathbf{A}^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_{k}-\frac{1}{2}\Delta t^{2}\left(\gamma+1\right)\mathbf{A}^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\dot{\mathbf{\Lambda}}_{k}+\mathrm{O}\left(\Delta t^{3}\right)\tag{8.41}$$

Thus, by inserting Eq. (8.41) into Eq. (8.9), we can obtain

$$\Phi_{k+1}=\frac{1}{2}\Delta t^{2}\gamma\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right)^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_{k}+\frac{1}{2}\Delta t^{2}\left(\gamma+1\right)\dot{\mathbf{\Lambda}}_{k}+\mathrm{O}\left(\Delta t^{3}\right)\tag{8.42}$$

Now, the local truncation error of the differential vector can be obtained

$$\begin{aligned}\tau_{k+1}&=\mathbf{y}_{k+1}^{*}+\mathbf{A}^{-1}\mathbf{C}^{T}\mathbf{\Phi}_{k+1}-\mathbf{y}\left(t_{k+1}\right)=\mathbf{y}_{k+1}^{*}-\mathbf{y}\left(t_{k+1}\right)+\mathbf{A}^{-1}\mathbf{C}^{T}\mathbf{\Phi}_{k+1}\\&=-\frac{1}{2}\Delta t^{2}\gamma\mathbf{A}^{-1}\mathbf{C}^{T}\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right)^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_{k}-\frac{1}{2}\Delta t^{2}\left(\gamma+1\right)\mathbf{A}^{-1}\mathbf{C}^{T}\dot{\mathbf{\Lambda}}_{k}+\mathrm{O}\left(\Delta t^{3}\right)\\&+\frac{1}{2}\Delta t^{2}\gamma\mathbf{A}^{-1}\mathbf{C}^{T}\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{T}\right)^{-1}\mathbf{C}\left.\frac{\partial\mathbf{F}}{\partial t}\right|_{k}+\frac{1}{2}\Delta t^{2}\left(\gamma+1\right)\mathbf{A}^{-1}\mathbf{C}^{T}\dot{\mathbf{\Lambda}}_{k}+\mathrm{O}\left(\Delta t^{3}\right)\\&=\mathrm{O}\left(\Delta t^{3}\right)\end{aligned}\tag{8.43}$$

which indicates a local truncation error of $\mathrm{O}\left(\Delta t^{3}\right)$. By inserting Eq. (8.42) into Eq. (8.10), we can obtain the local truncation error of the algebraic vector, i.e.

$$\begin{aligned}\mathbf{\Lambda}_{k+1}&=\mathbf{\Lambda}_{k+1/2}+\frac{1}{\Delta t}\Phi_{k+1}\\&=\mathbf{\Lambda}_{k}+\mathrm{O}\left(\Delta t^{2}\right)\end{aligned}\tag{8.44}$$

which is only locally first order accurate.

In this proof process, we can find that the second order terms in the expression of the Lagrange multiplier vector, as shown in Eq. (8.35), is eliminated completely in the calculation of the second stage as implied in Eq. (8.43). Likewise, the unbalanced second order term in Eq. (8.44) can be filtered out by the projection in the first stage of the next time step. This makes the differential vector achieves globally second order accuracy even in the presence of the lower order accuracy in the algebraic vector. Moreover, similar explanation can be used for the similar phenomenon in the LSRT1-based projection method.

8.4 Stability analysis

In this section, the analytical stability analyses are performed for the proposed projection method. For the LSRT1-based projection method, the linear recurrence approach is used and the method is proved to be unconditionally stable when applied to second order systems. For the LSRT2-based projection method, the method has two projection and therefore the linear recurrence approach become infeasible. But, this approach is utilized to study the projection method with only one projection at the end of the time step which is included in the common framework proposed by (Zheng et al., 2009). The method is proved to be unconditionally unstable when applied to undamped second order systems. This indicates the necessity to introduction another projection at the end of the first stage.

In order to analyse the stability properties of the methods, we use a linear second order system and assume no damping term and no external force term. The governing equations can be expressed in the first order form as

$$\begin{aligned}\dot{\mathbf{y}} &= \mathbf{J}\mathbf{y} + \mathbf{A}^{-1}\mathbf{C}^T\boldsymbol{\Lambda} \\ \mathbf{C}\mathbf{y} &= 0\end{aligned} \tag{8.45}$$

For brevity, we introduce an linear ODEs system, $\dot{\mathbf{y}}^* = \mathbf{J}\mathbf{y}^*$. For the LSRT1 method, its application to the linear system leads to the amplification matrix as

$$\mathbf{R} = \mathbf{I} + \Delta t\mathbf{J}\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1} \tag{8.46}$$

The application of the LSRT2 method to the system results in the amplification matrix as

$$\mathbf{R} = \mathbf{I} + \Delta t\mathbf{J}\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\left(\mathbf{I} + \left(\frac{1}{2} - \gamma\right)\Delta t\mathbf{J}\left[\mathbf{I} - \Delta t\gamma\mathbf{J}\right]^{-1}\right) \tag{8.47}$$

The two amplification matrices are expressed in the function of $\Delta t\mathbf{J}$. Now, we simultaneously analyse both the LSRT1-based projection method and the LSRT2-based projection method with only one projection step at the end of each time step. Here, we shortly describe the solution procedure of the latter case.

1. Solve for the intermediate solution at time t_{k+1}

$$\begin{aligned}
\mathbf{k}_1 &= \Delta t \left[\mathbf{I} - \Delta t \gamma \mathbf{J}\right]^{-1} \left(\mathbf{F}\left(\mathbf{y}_k, t_k\right) + \mathbf{A}^{-1}\mathbf{C}^T \boldsymbol{\Lambda}_k\right) \\
\mathbf{y}_{k+1/2} &= \mathbf{y}_k + \frac{1}{2}\mathbf{k}_1 \\
\mathbf{k}_2 &= \Delta t \left[\mathbf{I} - \Delta t \gamma \mathbf{J}\right]^{-1} \left(\mathbf{F}\left(\mathbf{y}_{k+1/2}, t_{k+1/2}\right) + \mathbf{A}^{-1}\mathbf{C}^T \boldsymbol{\Lambda}_k - \gamma \mathbf{J}\mathbf{k}_1\right) \\
\mathbf{y}^*_{k+1} &= \mathbf{y}_k + \mathbf{k}_2
\end{aligned} \tag{8.48}$$

2. Perform the projection

$$\begin{aligned}
\Phi_{k+1} &= -\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1} \mathbf{C}\mathbf{y}^*_{k+1} \\
\mathbf{y}_{k+1} &= \mathbf{y}^*_{k+1} + \mathbf{A}^{-1}\mathbf{C}^T \Phi_{k+1}
\end{aligned} \tag{8.49}$$

3. Update the Lagrange multiplier

$$\boldsymbol{\Lambda}_{k+1} = \boldsymbol{\Lambda}_{k+1/2} + \frac{1}{\Delta t}\Phi_{k+1} \tag{8.50}$$

This solution procedure is the similar as the LSRT1-based projection method as shown in Subsection 8.2.1. The first step of both cases, i.e. Eqs. (8.2) and (8.48), results in the common formula as follows

$$\mathbf{y}^*_{k+1} = \mathbf{R}\mathbf{y}_k - \left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{A}^{-1}\mathbf{C}\Lambda_k \tag{8.51}$$

It is derived from Eq. (8.51) that

$$\mathbf{y}^*_{k+1} - \mathbf{y}^*_k = \mathbf{R}\left(\mathbf{y}_k - \mathbf{y}_{k-1}\right) - \left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{A}^{-1}\mathbf{C}\left(\boldsymbol{\Lambda}_k - \boldsymbol{\Lambda}_{k-1}\right) \tag{8.52}$$

In the projection step, combining Eqs. (8.3) leads to

$$\begin{aligned}
\mathbf{y}_{k+1} &= \mathbf{y}^*_{k+1} - \mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{y}^*_{k+1} \\
&= \left(\mathbf{I} - \mathbf{Q}\right)\mathbf{y}^*_{k+1}
\end{aligned} \tag{8.53}$$

where $\mathbf{Q} = \mathbf{A}^{-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}$.

From the last step, we can obtain

$$\boldsymbol{\Lambda}_{k+1} - \boldsymbol{\Lambda}_k = \frac{2}{\Delta t}\Phi_{k+1} \tag{8.54}$$

Likewise, we can obtain

$$\boldsymbol{\Lambda}_k - \boldsymbol{\Lambda}_{k-1} = \frac{2}{\Delta t}\Phi_k \tag{8.55}$$

Combining Eq. (8.52) and Eq. (8.55) leads to

$$\begin{aligned}
\mathbf{y}^*_{k+1} - \mathbf{y}^*_k &= \mathbf{R}\left(\mathbf{y}_k - \mathbf{y}_{k-1}\right) - \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{A}^{-1}\mathbf{C}\Phi_k \\
&= \mathbf{R}\left(\mathbf{I} - \mathbf{Q}\right)\left(\mathbf{y}^*_k - \mathbf{y}^*_{k-1}\right) + \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{A}^{-1}\mathbf{C}\left(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{y}^*_k \\
&= \mathbf{R}\left(\mathbf{I} - \mathbf{Q}\right)\left(\mathbf{y}^*_k - \mathbf{y}^*_{k-1}\right) + \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{Q}\mathbf{y}^*_k
\end{aligned} \tag{8.56}$$

Let $\mathbf{y}^*_k = \mathbf{w}_k + \mathbf{v}_k$, where $\mathbf{w}_k = \mathbf{Q}\mathbf{y}^*_k$ and $\mathbf{v}_k = \left(\mathbf{I} - \mathbf{Q}\right)\mathbf{y}^*_k$. We can reform Eq. (8.56) as

$$\mathbf{w}_{k+1} + \mathbf{v}_{k+1} - \mathbf{w}_k - \mathbf{v}_k = \mathbf{R}\left(\mathbf{v}_k - \mathbf{v}_{k-1}\right) + \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{w}_k \tag{8.57}$$

Consider the following two recurrences:

$$\mathbf{v}_{k+1} - \mathbf{v}_k = \mathbf{R}\left(\mathbf{v}_k - \mathbf{v}_{k-1}\right) \tag{8.58}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1}\mathbf{w}_k \tag{8.59}$$

The propagation matrices for the recurrences (8.58) and (8.58) are $\mathbf{R}_1 = \mathbf{R}$ and

$$\mathbf{R}_2 = \mathbf{I} + \frac{2}{\Delta t}\left(\mathbf{I} - \mathbf{R}\right)\mathbf{J}^{-1} \tag{8.60}$$

Now, the stability properties for the projection methods can be achieved through the stability analyses of the two propagation matrices (Zheng and Petzold, 2006). Zheng and Petzold (2006) introduced a first order test problem to study their absolute stabilities of the propagation matrices. Using the linear first order test equation (Dahlquist, 1963), i.e. $\dot{y} = \lambda y$, we can define $z = -\Delta t \lambda$ with $z \geqslant 0$. For the LSRT1-based projection method, substituting $\Delta t \mathbf{J} = -z$ into Eq. (8.46) results in the stability funcation as

$$R_1\left(z\right) = R\left(z\right) = 1 - z/\left(1 + \gamma z\right) \tag{8.61}$$

The stability requirement is obtained by imposing

$$\left|1 - z/\left(1 + \gamma z\right)\right| \leqslant 1 \tag{8.62}$$

which indicates that $\gamma \geqslant 1/2$.

Inserting Eq. (8.61) into (8.60) leads to the stability function as

$$R_2\left(z\right) = 1 - \frac{2}{1 + \gamma z} \tag{8.63}$$

With the same manner, its stability requirement is determined by

$$\left|1 - \frac{2}{1 + \gamma z}\right| \leqslant 1 \tag{8.64}$$

which indicates that the second matrix is unconditionally stable when $\gamma \geqslant 1/2$.

However, the stability analysis based on the first order test equation is not sufficient. Since the projection methods are proposed for mechanical second-order systems, it is necessary to consider the second-order test problem (Dahlquist, 1978), i.e. $\ddot{u} = -\omega^2 u$. In order to analyse the stability of recurrences (8.58) and (8.59) for a second-order system, we consider the model problem

$$\begin{aligned} \dot{u} &= p \\ \dot{p} &= -\omega^2 u \end{aligned} \tag{8.65}$$

As a result, $\lambda^2 = -\omega^2$; moreover, the stability function of Recurrence (8.58) based on the second-order test problem can be obtained by substituting into (8.46).

$$R_1\left(z\right) = R\left(z\right) = 1 - \frac{zi}{(1 + \gamma zi)} \tag{8.66}$$

The stability condition is imposed by $\|R_1\| \leqslant 1$, i.e.

$$\sqrt{\frac{1 + z^2(-1 + \gamma)^2}{1 + z^2\gamma^2}} \leqslant 1 \tag{8.67}$$

which indicates that $\gamma \geqslant 1/2$.

Likewise, inserting (8.66) and $\Delta t\mathbf{J} = -zi$ into (8.60) results in the stability function of Recurrence (8.59) based on the second-order test problem

$$R_2\left(z\right) = 1 - \frac{1}{(1 + \gamma zi)} \tag{8.68}$$

The stability condition is imposed by $\|R_2\| \leqslant 1$, i.e.

$$\sqrt{\frac{z^2\gamma^2}{1 + z^2\gamma^2}} \leqslant 1 \tag{8.69}$$

This condition is satisfied with an arbitrary value of γ.

Based on the analysis above, the following conclusion can be made: when $\gamma \geqslant 1/2$, both recurrences (8.58) and (8.59) are stable based on both the first-order and second-order test problems. Hence following (Zheng and Petzold, 2006), the sum of these two recurrences is stable, and therefore the LSRT1-based projection method is stable.

Now, we move to the LSRT2-based projection method with only one projection step. Here we use the second-order test problem to prove that this case is not stable for structural problems.

The stability function can be obtained by substituting $\Delta t\mathbf{J} = -zi$ into (8.47)

$$R_1 = R = 1 - zi\,[1 + \gamma zi]^{-1}\left(1 - \left(1/2 - \gamma\right) zi\,[1 + \gamma zi]^{-1}\right) \tag{8.70}$$

The stability condition for the first propagation matrix is imposed by $\|R_1\| \leqslant 1$, i.e.,

$$\frac{1}{2}\sqrt{\frac{4 + 8z^2\gamma^2 + z^4\left(1 - 4\gamma + 2\gamma^2\right)^2}{\left(1 + z^2\gamma^2\right)^2}} \leqslant 1 \tag{8.71}$$

which indicates that $\gamma \geqslant 1/4$.

Note that the solution of $\|R_1\| = 1$ contains two values of γ, $\gamma = 1/4$ and $\gamma = 1/2$ which is multiple solution. This is consistent to the stability analysis via the energy method in Chapter 3.

Now, we know that the first propagation matrix is stable when $\gamma \geqslant 1/4$. Let us analyse the second propagation matrix. Inserting (8.70) into Eq. (8.60) gives

$$R_2 = 1 - 2\,[1 + \gamma zi]^{-1}\left(1 - \left(1/2 - \gamma\right) zi\,[1 + \gamma zi]^{-1}\right) \tag{8.72}$$

whose stability is guaranteed by imposing $\|R_2\| \leqslant 1$, i.e.,

$$\sqrt{\frac{1 + z^4\gamma^4 + z^2(1 - 4\gamma + 6\gamma^2)}{(1 + z^2\gamma^2)^2}} \leqslant 1 \tag{8.73}$$

which can be simpified into

$$z^2(1 - 2\gamma)^2 \leqslant 0 \tag{8.74}$$

Now, we can find that the second matrix is unconditionally unstable except $\gamma = 1/2$. That's why we introduce another projection with the purpose to have desirable stability property. But for the LSRT2-based projection method with two projections, the linear recurrences approach is not suitable in that the additional projection blocks the process to split the original recurrence into different but dependent recurrences. The stability of the proposed method will be investigated by means of spectral stability approach in the next section.

8.5 Numerical analysis and simulations of a Single-DoF split mass system

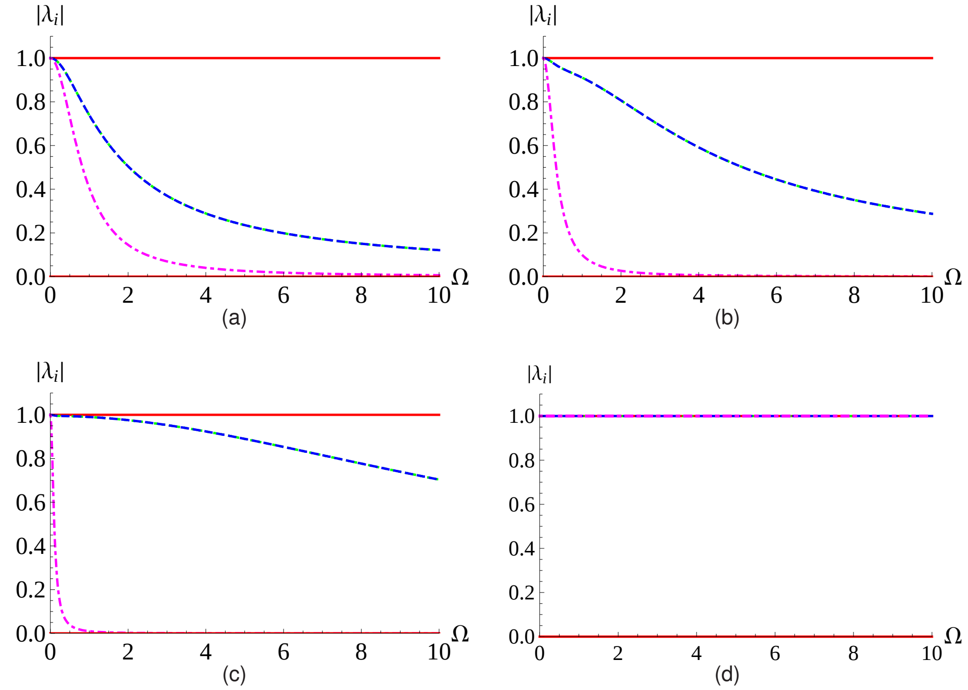
We present here some numerical validation of the projection methods via applications to the Partitioned Single-DoF system, and in particular, the numerical stability analysis by means of the spectral approach.

8.5.1 Spectral stability analysis

The LSRT1-based projection method is analytically proved to be unconditionally stable. Therefore, only a few of simulation results will be presented to revalidate the theoretical analysis in the previous section and greater effect will be taken to provide sufficient numerical results to guarantee the stability of the LSRT2-based projection method. Let us once more consider the Partitioned Single-DoF system in Chapter 6 and adopt the same system characteristics. For both projection methods, the amplification matrices are constructed with 4 initial values, viz. 3 components from the differential vector and 1 component from the algebraic vector. Although the differential vector contains 4 components but only three independent components in that the algebraic equation entails velocity continuity. Hence the following state vector is considered

$$\left\{ \begin{array}{cccc} u_A & \dot{u}_A & u_B & \lambda \end{array} \right\}^T \tag{8.75}$$

Firstly, we check the absolute values of the eigenvalues of the amplification matrices for the LSRT1-based projection method in Fig. 8.1. The number of the nonzero eigenvalues is 4. Among them, one pair of complex conjugate eigenvalues are the principal eigenvalues while the real one is the spurious eigenvalue. Likewise, the principal eigenvalues enable numerical simulations emulate the overall system. The other two are real. One of them is unitary whose associated eigenvector can be obtained by considering ”equilibrium point”. If the equilibrium solution $\left\{ \begin{array}{cccc} k_B & 0 & k_A & k_A k_B \end{array} \right\}^T$ is taken as the initial solution, the solution procedure in one time step from t_k to t_{k+1} follows: the intermediate solution maintain unchanged in the first step through (8.2); the zero velocity term in the intermediate solution induces zero projection term Φ_{k+1} which makes unitary projection; finally the zero term Φ_{k+1} results in no update of the



(a)

(b)

(c)

(d)

Fig. 8.1: | λ_i | for the LSRT1-based projection method integrated without subcycling: (a) b_1 = 2 and γ = 1; (b) b_1 = 10 and γ = 1; (c) b_1 = 100 and γ = 1; (b) b_1 = 10 and γ = 1/2.

Lagrange multiplier. Therefore, one unitary eigenvalue exists. The other real eigenvalue is spurious and less than 1. Therefore, it does not directly affect the physical nature of the solution only induces slight displacement incoincidence at the interface.

Fig. 8.1 also brings to mind that the projection method is unconditionally stable which is consistent with the theoretical results in the previous section. Moreover, the projection method associated with the conservative integrator γ = 1/2 has no numerical dissipation. This indicates that the coupling does not introduce any numerical damping which is one evident difference from the partitioned method with acceleration constraint in the previous two chapters.

Fig. 8.2 (a) displays the numerical damping ratios associated with the principal eigenvalues of the LSRT1-based projection method. One can observe that the numerical damping ratio remains closer to zero and thus induces less damping in the low frequency oscillations. Moreover, the method produces desirable numerical damping

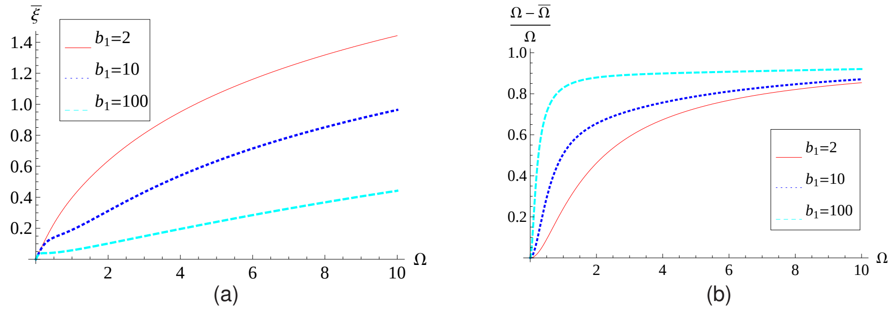


Fig. 8.2: Numerical damping ratio and relative frequency error for the LSRT1-based projection methods: (a) numerical damping ratio; (b) relative frequency error.

in the higher frequency range which follows the progenitor LSRT1 method very well. It is also noticeable that the higher value of the partitioned parameter b_1 gives less low frequency dissipation. From Fig. 8.2 (b), the price paid for the less dissipation is prone to higher frequency error in the low frequency range.

Next, we conduct a series of tests to investigate the stability properties of the LSRT2-based projection method. The absolute values of the eigenvalues of the amplification matrices for the LSRT2-based projection method are presented in Figs. 8.3-8.4. Clearly, the various cases considered are unconditionally stable. Differently from the LSRT1-based projection method, the moduli of the spurious eigenvalue reduces from 0.5 (not one) to zero along with increasing Ω. This is determined by the updates of the Lagrange multiplier but does not affect the accuracy of the projection method. As we analysed in the section of accuracy analysis, the error induced in the update of the Lagrange multiplier vector can be corrected with the later projection process. Moreover, the principal eigenvalues reduces smoothly from one to zero along with Ω. The desirable high frequency dissipation property of the progenitor LSRT2 method is preserved as shown in Fig. 8.6. However, the method gives rise to an evitable higher periodicity error in the high frequency range. For the projection method based on the conservative Rosenbrock method, the method introduces slight numerical damping.

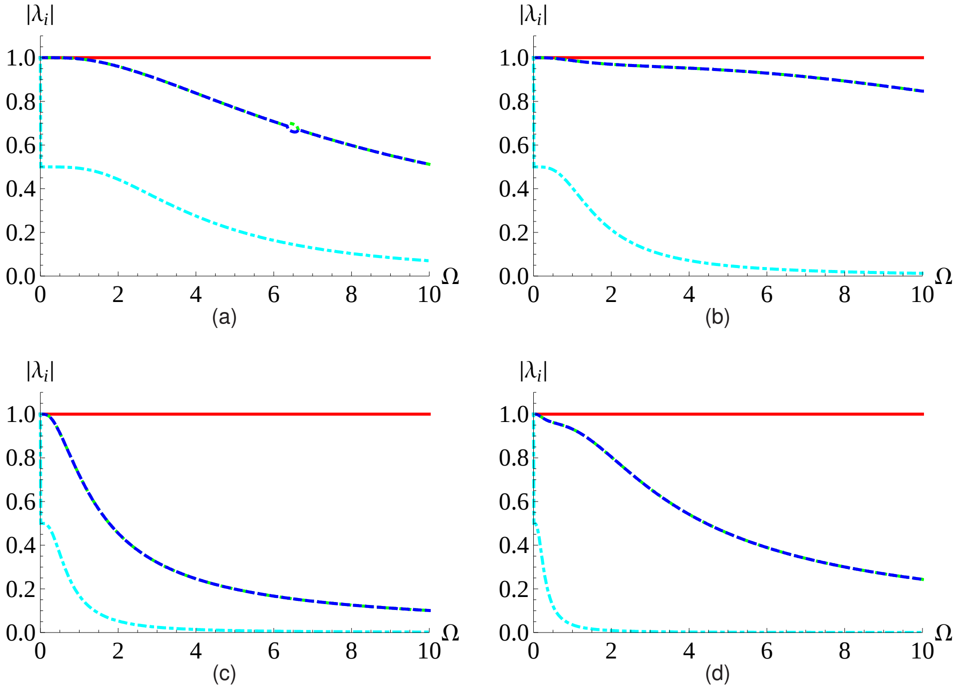


Fig. 8.3: $|\lambda_i|$ for the LSRT2-based projection method integrated without subcycling: (a) $b_1 = 2$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $b_1 = 10$ and $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c) $b_1 = 2$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$; (b) $b_1 = 10$ and $\gamma = 1 + \frac{\sqrt{2}}{2}$.

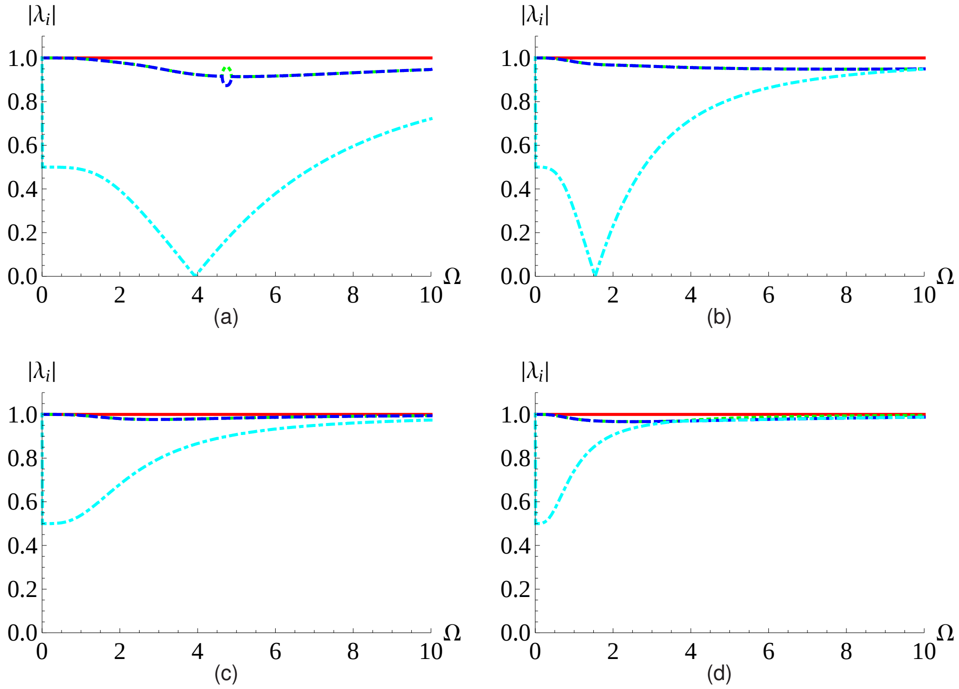


Fig. 8.4: $\mid \lambda_i \mid$ for the projection method integrated with conservative integrators: (a) b_1 = 2 and γ = 1/4; (b) b_1 = 10 and γ = 1/4; (c) b_1 = 2 and γ = 1/2; (b) b_1 = 10 and γ = 1/2.

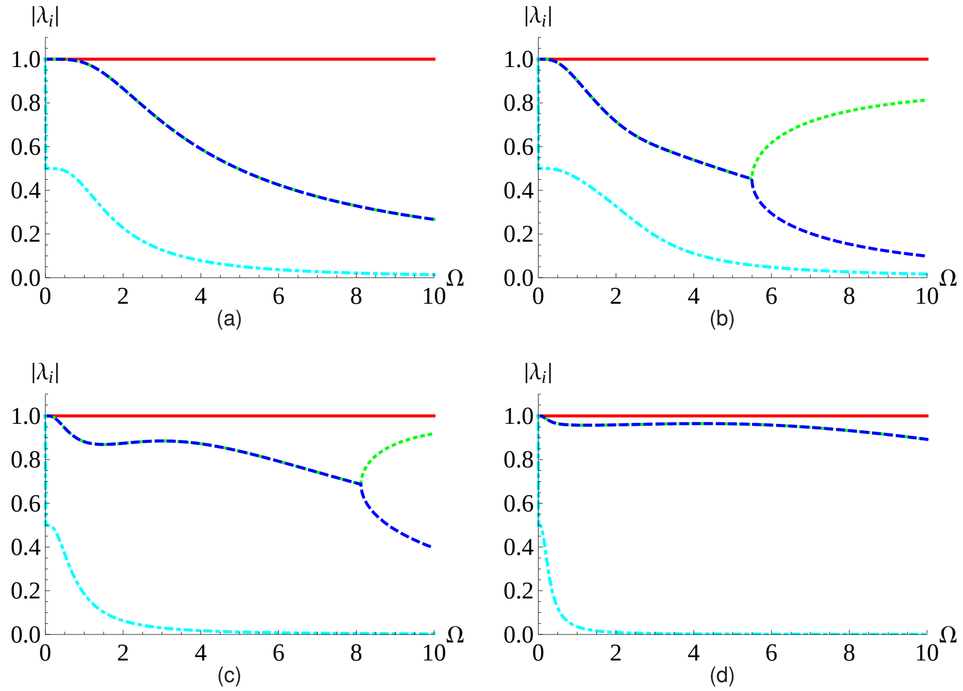


Fig. 8.5: $|\ \lambda_i\ |$ for the LSRT2-based projection method integrated with $\gamma_A = 1 - \frac{\sqrt{2}}{2}$ and $\gamma_B = 1 + \frac{\sqrt{2}}{2}$: (a) $b_1 = 0.1$; (b) $b_1 = 0.5$; (c) $b_1 = 2$; (b) $b_1 = 10$.

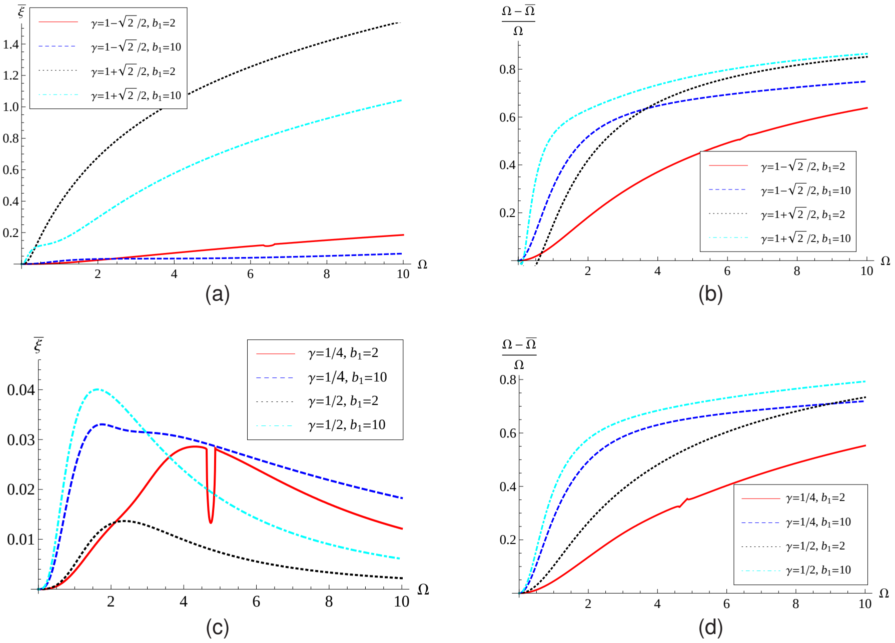


Fig. 8.6: Numerical damping ratio and relative frequency error for the LSRT2-based projection methods with: (a), (b) L-stable integrator; (c), (d) conservative integrator.

8.5.2 Numerical convergence analysis

This subsection, the accuracy analysis is revalidated through the simulation on the Partitioned Single-DoF system. Here, the system is subjected to a harmonic base excitation which is imposed on subdomain A. The numerical results for a range of time steps from $\Delta t = 1.0 \times 10^{-5}$ to $\Delta t = 0.1$ are compared with the exact solution at $t = 0.5$.

For the LSRT1-based projection method, the global error is examined in Fig. 8.7. It can be observed that both the differential vector and the algebraic vector are first order convergent. The LSRT2-based projection method is checked in Fig. 8.8. Only second-order convergence of the differential vector is observed while the algebraic vector is first-order convergent. Moreover, one can observe from both Figs. 8.7 and 8.8 that the solutions of both subdomains in the displacement and the velocity are overlapped. This, on the one hand, shows that the solutions in both subdomains have similar accuracy; on the other hand, this indicates that the drift effect is limited.

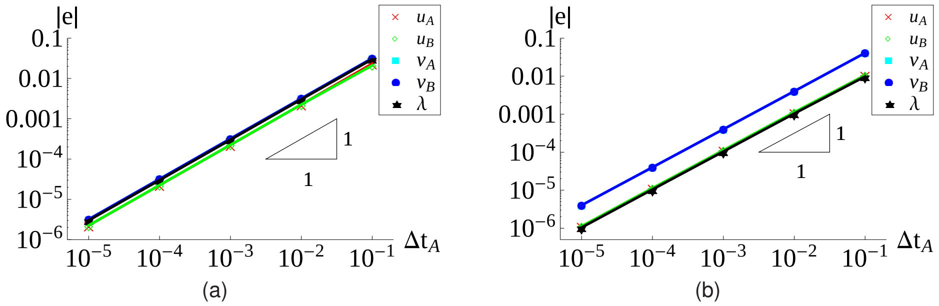


Fig. 8.7: Global error of the Partitioned Single-DoF system emulated with the LSRT1-based projection method: (a) $\gamma = 1$; (b) $\gamma = 1/2$.

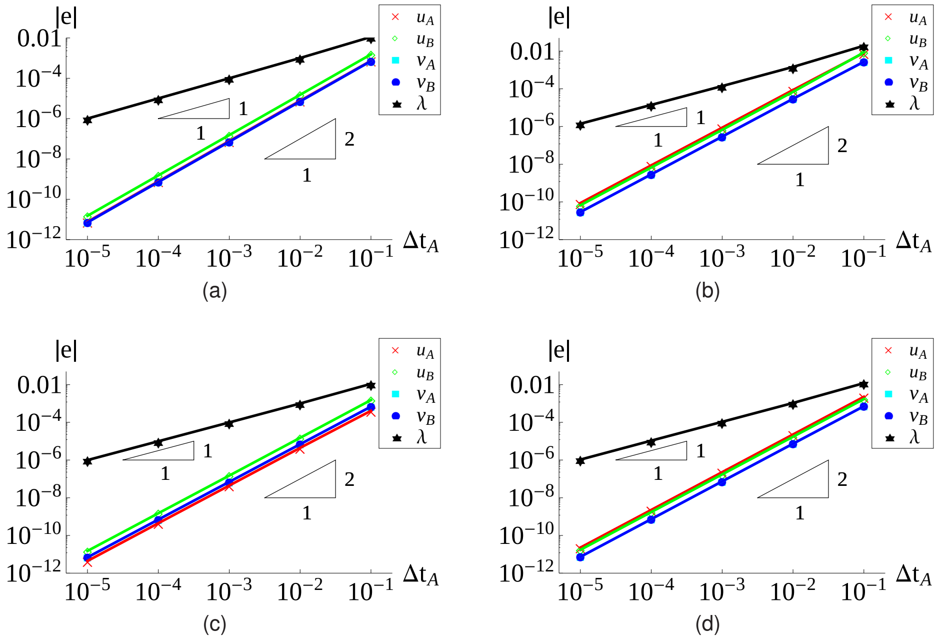
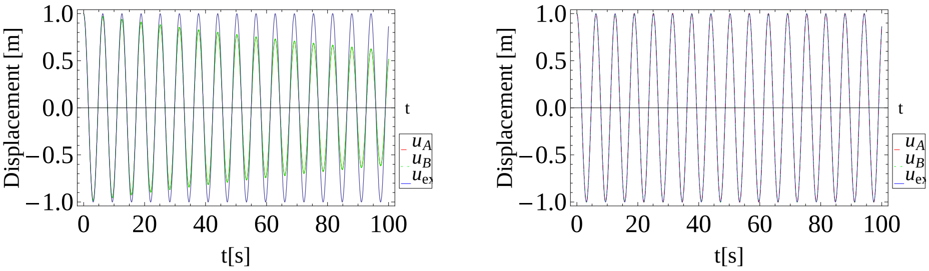


Fig. 8.8: Global error of the Partitioned Single-DoF system emulated with the LSRT2-based projection method: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$.

Therefore, the drift analysis for this case is omitted.



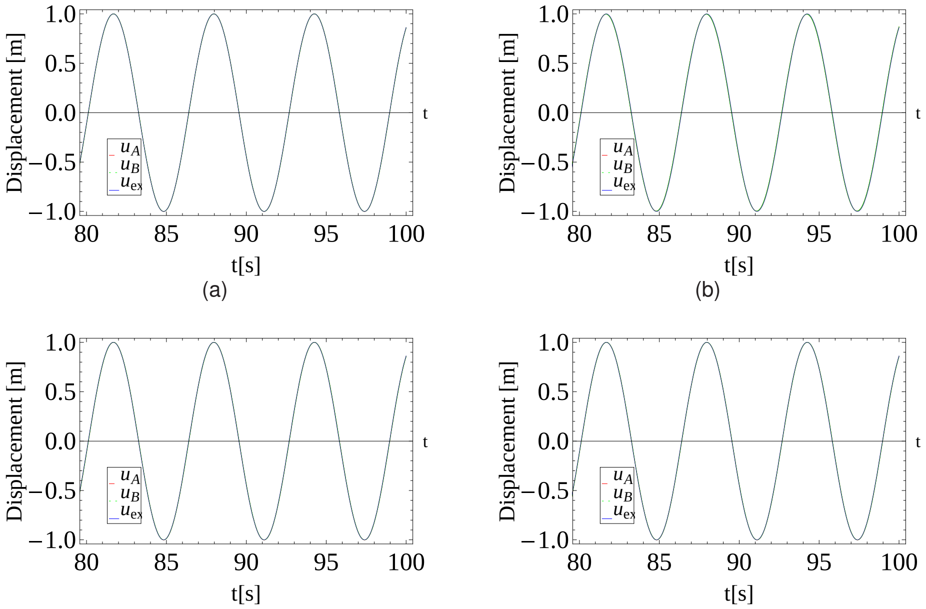
(a)

(b)

Fig. 8.9: The displacement responses of the LSRT1-based projection method applied to the Partitioned Single-DoF system emulated with : (a) $\gamma = 1$; (b) $\gamma = 1/2$.

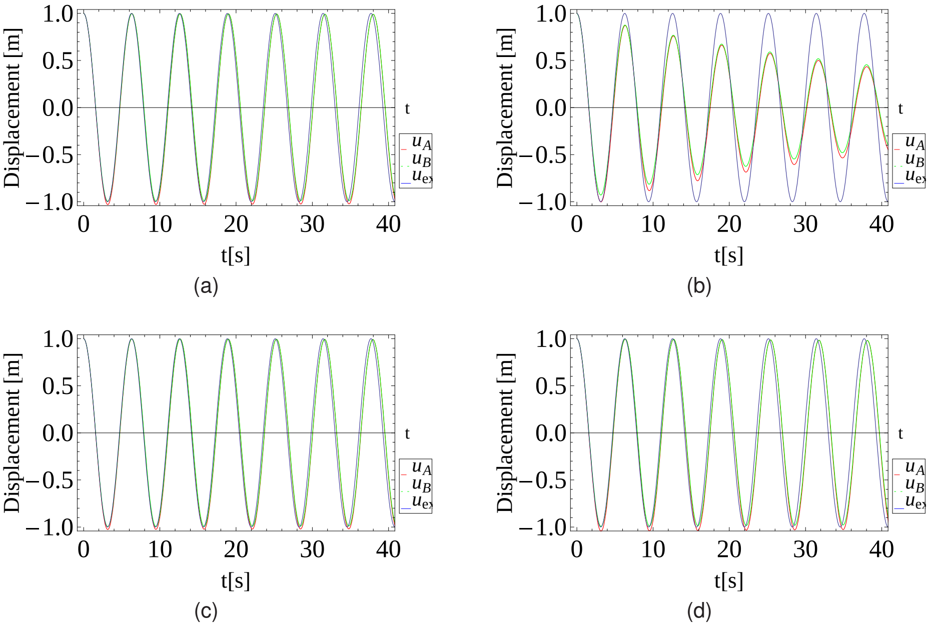
8.5.3 Numerical simulations

The Partitioned Single-DoF system without external force is considered here. The time step during the simulation is chosen to be 10ms. For the LSRT1-based projection method, the simulation results in the displacement are plotted in Fig. 8.9. The case with $\gamma = 1$ shows oscillatory decay, which is only due to the numerical damping introduced by the L-stable integrator. When using $\gamma = 1/2$, the method captures the response very accurately with no visible oscillatory decay and period delay. For the LSRT2-based projection method, the numerical solutions by means of different cases are observed to stay stable and accurate for a total time of 100s as shown in Fig. 8.10. Also, in our experience, the methods remain sufficient accurate even when using a total simulation period of 2000s. In order to compare the different cases, a larger time step, $\Delta t = 100ms$, is used. The displacement responses are displayed in Fig 8.11. One can observe that all the cases remain stable but presenting slight numerical error. Especially, the cases with $\gamma = 1 + \frac{\sqrt{2}}{2}$ introduces greater numerical damping as well as sight period error. For all the other cases, the numerical errors are mainly linked to period delay. Moreover, it is noticeable that all the cases with $\Delta t = 100ms$ exhibit slight drift effect in the displacement responses.



(a) (b) (c) (d)

Fig. 8.10: The displacement responses of the LSRT2-based projection method applied to the partitioned Single-DoF system with $b_1 = 10$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$



(a)

(b)

(c)

(d)

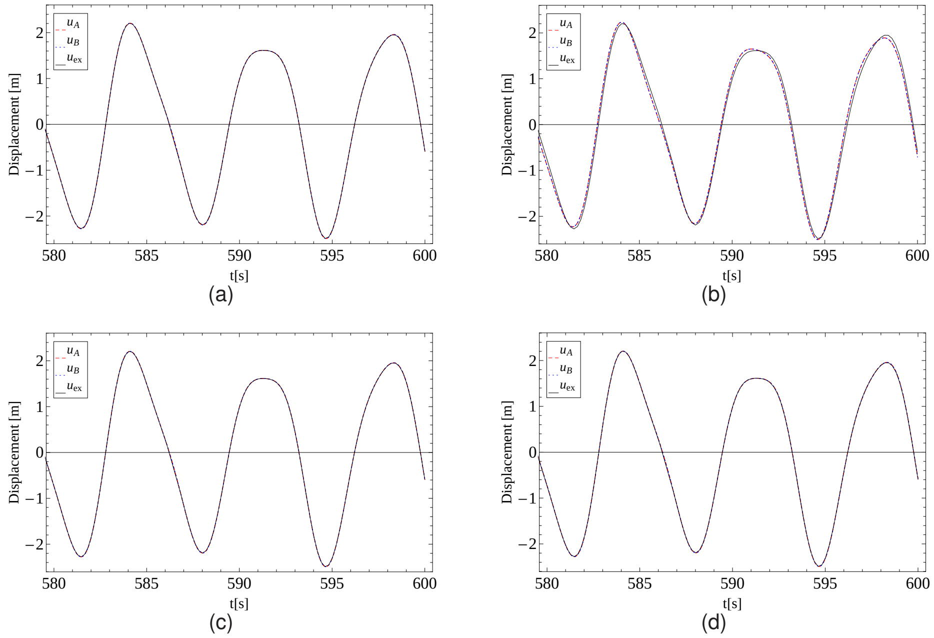
Fig. 8.11: The displacement responses of the LSRT2-based projection method applied to the partitioned Single-DoF system with $b_1 = 10$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$

The projection methods, DAEs time integration methods, have been analytically investigated in terms of accuracy and stability. But this is not sufficient with respect to ODE integrators (Arnold et al., 2007a). In order to validate the efficiency and applicability of the methods, the numerical tests are performed for the Multiple-DoF systems utilized in Chapter 6. In this section, we mainly deal with the LSRT2-based projection method for simplicity.

For the Partitioned Two-DoF system 1, Fig. 8.12 shows the displacement of the interface DoF computed with time step $\Delta t = 10ms$ using the LSRT2-based integration method with different values of parameter γ. The difference in the displacement responses becomes evident, of course, a small amount of numerical damping is introduced in the case with $\gamma = 1 + \frac{\sqrt{2}}{2}$. In order to extend the comparison subsection in Chapter 7, the external forces and the partitioned parameter b_1 are the same as those used in the comparison subsection. By comparison, the projection method exhibits higher accuracy, especially the case $\gamma = 1 + \frac{\sqrt{2}}{2}$, with excellent agreement with the exact solution.

The projection method, without subcycling involved, has high applicability to the multiple-interface problem. In order to study its application, the Partitioned Tow-DoF system 2 is once more considered. For comparison and simplicity, the system characteristics and the initial condition are chosen to be equivalent to the partitioned Two-DoF system 1 by splitting the second DoF. The displacement responses are shown in Fig. 8.13. We can notice that the method with $\gamma = 1 - \frac{\sqrt{2}}{2}$ gives sufficient accuracy that the displacement can follow the exact solution seamlessly along with increasing time until 600s. To the contrary, the method with $\gamma = 1 + \frac{\sqrt{2}}{2}$ introduces slight numerical damping that the displacement can preserve sufficient accuracy within a duration of 100s.

In the third test cases considered, the Partitioned Three-DoF system is simulated with no external force. As in the previous two chapters, floating subdomains lead to weak instability and therefore aggravate drift effect. The Partitioned Three-DoF system has one floating DoF which can be used to check if these phenomena affect the projection methods. The results of the coupled simulations of the Partitioned



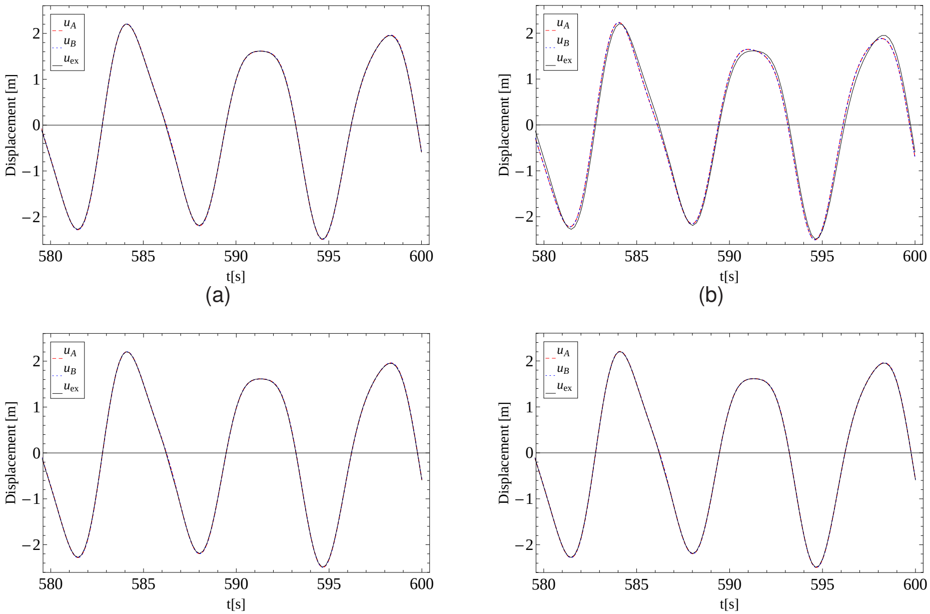
(a)

(b)

(c)

(d)

Fig. 8.12: The displacement responses of the LSRT2-based projection method applied to the partitioned Two-DoF system 1 with $b_1 = 10$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$



(a)

(b)

(c)

(d)

Fig. 8.13: The displacement responses of the LSRT2-based projection method applied to the partitioned Two-DoF system 2 with $b_1 = 10$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$

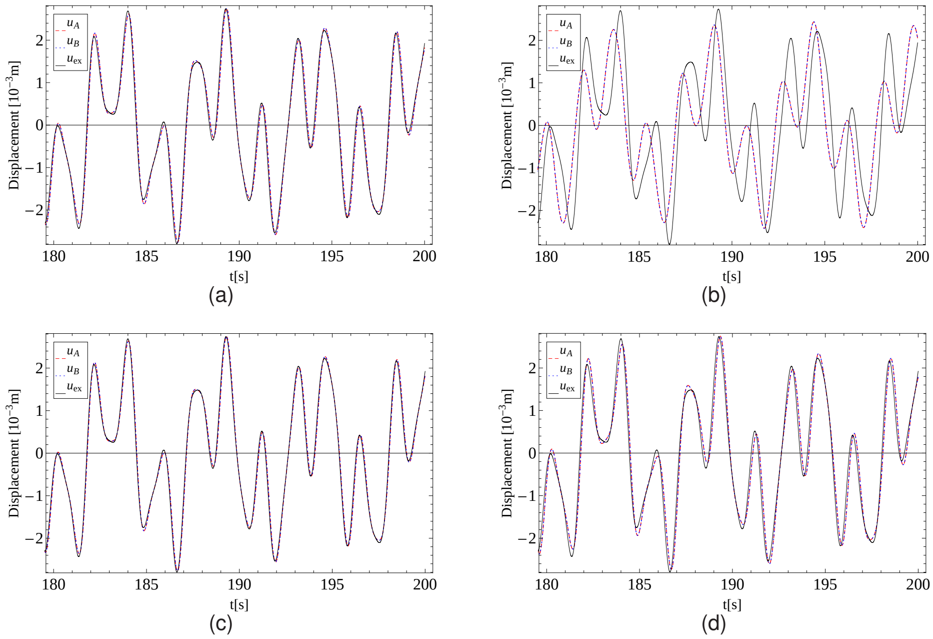


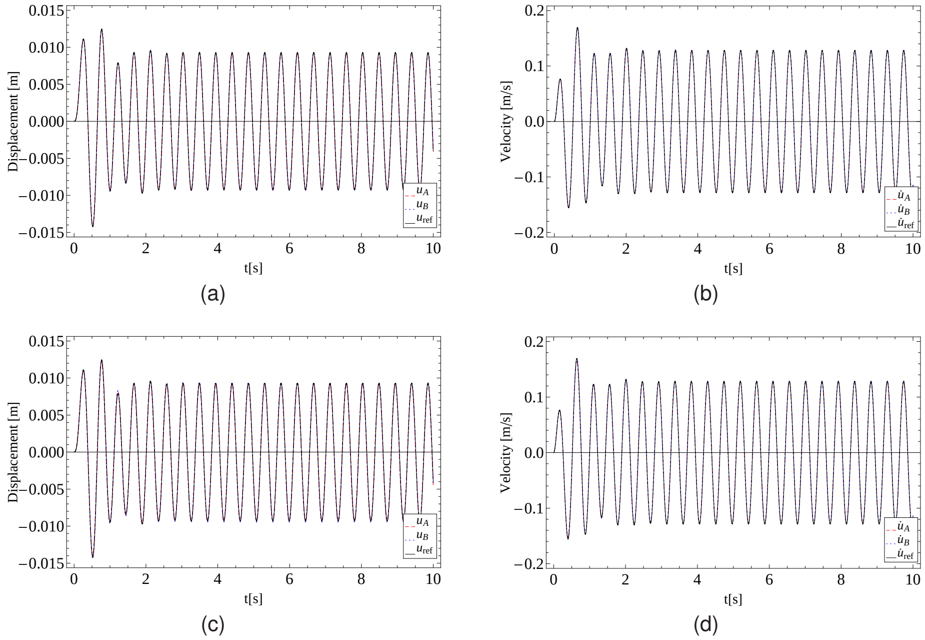
Fig. 8.14: The displacement responses of the LSRT2-based projection method applied to the partitioned Three-DoF system with $r = 2$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$

Three-DoF system are shown in Fig (8.14), with the time step $\Delta t = 10ms$. The reference solutions are obtained through simulation of the global system using the LSRT2 method with $\gamma = 1/2$ and $\Delta t = 0.1ms$. It is observed that the numerical responses and exact solution show acceptable agreement except the case with $\gamma = 1 + \frac{\sqrt{2}}{2}$. The case with $\gamma = 1 + \frac{\sqrt{2}}{2}$ performs well for a certain length of time after which the numerical solutions exhibit moderate amplitude decay and slight period delay as shown (8.14a). In conclusion, these solutions illustrate the superior and more efficient performance of the projection method in long-time duration problems. In particular, it is of interest to note by combining the all the simulations, from the partitioned Single-DoF system to the Partitioned Three-DoF system, that the errors do not show any systematic growth along with the system complexity. This indicates that the projection methods do not lead to an undesirable error when applied to more complex system.

The simulations in the previous three test problems involved linear systems. Here, we consider a stiff nonlinear problem, the coupled spring-pendulum system, whose system characteristics are collected in Table 4.1. The problem is solved using the LSRT2-based projection method with $\Delta t = 10ms$. The proposed method performs well, giving a excellent displacement and velocity responses, as seen in Fig. 8.15. In this figure, the reference solution is obtained through integrating the uncoupled spring-pendulum system with the LSRT2 method with $\gamma = 1 - \sqrt{2}/2$ and $\Delta t = 0.1ms$. Moreover, the velocity responses of the elongation of the pendulum is plotted in Fig 8.16. The results indicate that the projection method with both values of γ has enough high frequency dissipation that the high frequency oscillation can be damped out in a few time steps. Although the responses for only the first 10s are shown, the problem was actually run for a total time of 200s, and the solution was observed to be stable and sufficiently accurate. Also, in our experience, the method remains stable if a larger time step is used or a longer simulation period is expected. But more numerical damping will be introduced with larger time step and the numerical damping will be accumulated to an inacceptable amplitude decay, especially with longer simulation duration. Altogether, the numerical dissipation requires high attention when dealing with stiff coupled problems. On the one hand, it is determined by the choices of the parameter γ and the time step and the partitioning of the coupled system. On the other hand, it determines low-frequency accuracy and high-frequency dissipation and therefore the efficiency and accuracy of the proposed method. Therefore, we need take into account the accuracy tolerance and the length of simulation when choosing the parameter γ and the time step, and we need avoid introducing additional high-frequency models when partitioning the system.

8.7 Conclusions

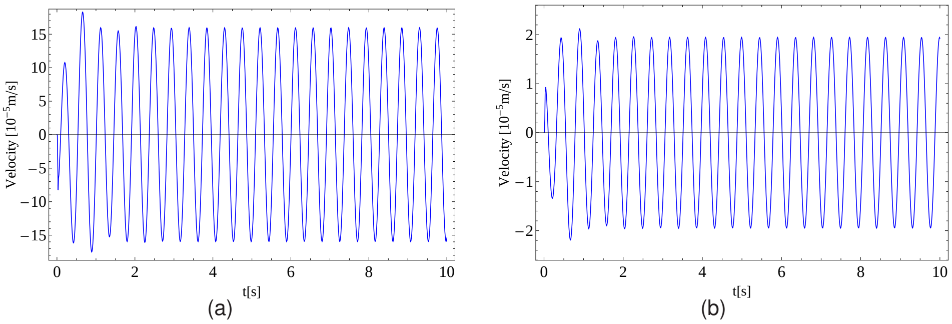
A new form of partitioned time integration methods was proposed in this chapter. Using these methods, all subdomains can be integrated independently and coupling is achieved by projecting the numerical solutions onto the velocity constraint, simultaneously updating the Lagrange multiplier vector at interfaces. Following this strategy, two projection methods based on the LSRT1 and the LSRT2 algorithms, respectively,



(a) (b)

(c) (d)

Fig. 8.15: The displacement and velocity responses of the LSRT2-based projection method applied to the coupled spring-pendulum stiff problem with: (a), (b) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (c), (d) $\gamma = 1 + \frac{\sqrt{2}}{2}$



(a) (b)

Fig. 8.16: The elongational velocity responses of the LSRT2-based projection method applied to the partitioned Three-DoF system 1 with $b_1 = 10$: (a) $\gamma = 1 - \frac{\sqrt{2}}{2}$; (b) $\gamma = 1 + \frac{\sqrt{2}}{2}$; (c) $\gamma = 1/4$; (d) $\gamma = 1/2$

were formulated to achieve certain order of accuracy and stability requirements. In order to validate their efficiency and properties, we conducted systematic numerical analyses for the proposed methods, e.g. spectral stability, numerical convergence and high-frequency dissipation. Moreover, a series of test problems were used to check both theoretical and numerical analyses. The numerical results showed that the methods remained stable and sufficiently accurate with certain time step sizes. Moreover, they preserve their progenitor high-frequency dissipation. Lastly, we underlined that the proposed methods can be implemented within a parallel computing environment, where each subdomain can be independently integrated using a separate processor. This observation provides the possibility to apply them to RTDS tests for very large structures. Nonetheless, subcycling was not included and investigated in these methods. A future work, believed to be of immense significance, is to combine the efficient methods with subcycling strategies and thus, hopefully, enlarge the applications of the proposed methods.

Chapter 9

Experimental validation and real-time implementation

9.1 Introduction

Both theoretical and numerical analyses for the partitioned methods have been presented in the previous four chapters. The partitioned methods with parallel solution procedure were demonstrated to be efficient for real-time applications. The monolithic methods with no subcycling were validated numerically and experimentally through the application to a coupled spring-pendulum system in Chapter 4 while the cases with subcycling were done only numerically but not experimentally.

With this in mind, a test rig is designed to provide a series of experimental validations for both monolithic and partitioned methods which are suitable for real-time implementations. This chapter describes the design and construction of the test rig.

9.2 General test rig design

Experiments were conceived to be performed using a small-scale but robust test rig to study the efficiencies and accuracies of the proposed monolithic and partitioned time integration methods, before being applied to large-scale structures. This rig is shown schematically in Fig. 9.1. The masses are guided on a linear bearing and linked by coil springs or other nonlinear elements (for example, damping devices and non-smooth spring). The rotations of the masses are allowed to consider geometric nonlinearity, while the tested structure can be also considered to be linear by constraining the rotation and installing only linear damping devices and springs.

The damping except the specific damping devices has two main sources in the system: the spring/damping elements connecting the masses together and the linear bearings. The damping provided by a spring/damping element is proportional to the relative velocity between the connected degrees of freedom, whereas the damping provided by the bearing is proportional to the absolute velocity of the supported mass (Bonnet et al., 2007). In addition to the geometric nonlinearity derived by the mass rotations, another two sources of nonlinearities are considered: springs with discontinuous supports and nonlinear damping devices. Also, nonlinearities can be included in the NS(s), such as the Bouc-Wen model. As a result, RTDS tests can be implemented in a variety of ways by installing different combinations of spring and/or damping devices and mass(es), and by considering linear/nonlinear models of the NS(s).

To perform RTDS tests, four electro-thrust actuators are connected to the PS (as shown Fig. 9.1). The translational motion of each mass is activated by average displacement of each pair of actuators attached while the rotational motion is achieved their relative displacement. Together with each actuator, each DoF activated is fitted with a load cell, a displacement transducer and an accelerometer.

The test rig was designed to take into account geometric nonlinearity by allowing the rotations of the masses. But it to some extent hardens the operation complexity of the test rig. Moreover, the rotation of the actuator introduces additional inertia forces into the system. A important design principle of the rig was to keep the rotations of the actuators as low as possible to ease the real-time implementation. As shown in Fig 9.2, the rotation range of the actuators depends primarily on the distance between the fixed points of the actuators in parallel, their lengths and their operating stroke range. The displacement stroke of the actuator is 250mm and the distance d_{OA} = 400 which determine the maximal rotation of each mass is $\theta_1 \approx 17$. The minimal radius of the rotation of the actuator is $d_{AB} = 990mm$ which indicates the maximal angle $\theta_2 \approx 9$. In this case, we neglect the rotation of the actuators and therefore their inertia forces. Meanwhile, the directions of the coupling forces measured from the load cells are assumed to be constant and parallel with the translational movement.

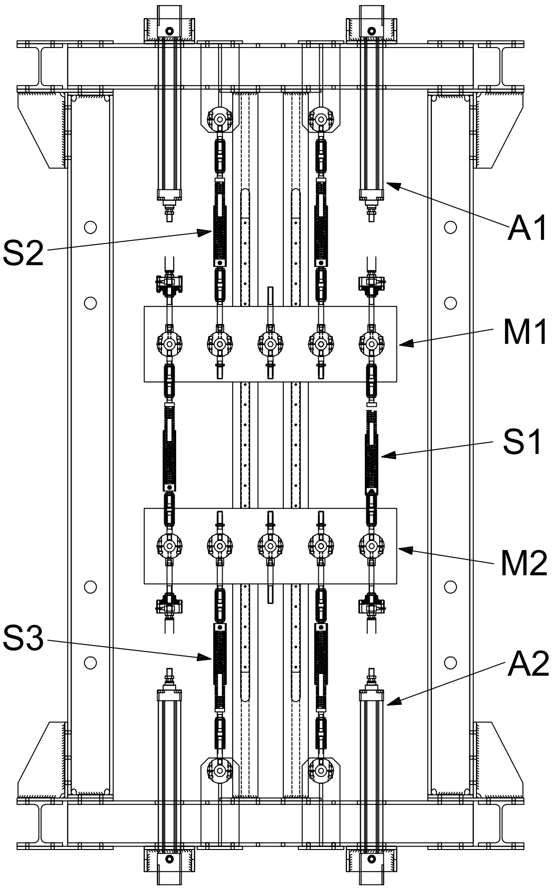


Fig. 9.1: Test set-up

9.3 Detailed experimental equipment

As shown in Fig. 9.1, it was intended to design the rig for a maximum capacity of two masses (considering their rotations) activated by two pairs of actuators as well as some spring and damping devices. In this section the specifications of the test components used in the test are emphasized. Before the description and choices of different test components, purely numerical simulations have been performed on the overall system shown in Fig. 9.3. The equations of motion for the system can be

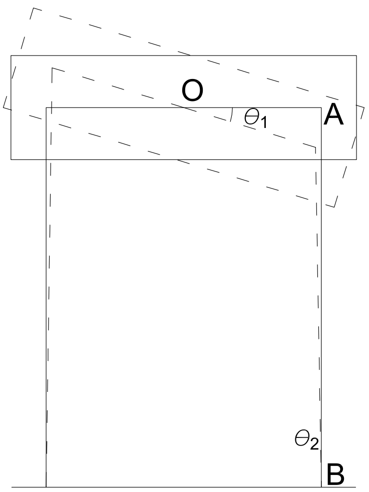


Fig. 9.2: Schematic representation of the rotation of the mass

found by means of Lagrangian mechanics:

$$
\begin{aligned}
& m_1\ddot{h}_1 + f_1 + f_2 - f_3 - f_4 + f_{cm,1} = F_1 \\
& m_2\ddot{h}_2 + f_3 + f_4 - f_5 - f_6 + f_{cm,2} = F_2 \\
& J_1\ddot{\theta}_1 + (f_1e_1 - f_2e_2 + f_4e_4 - f_3e_3)\cos\theta_1 = M_1 \\
& J_2\ddot{\theta}_2 + (f_3e_3 - f_4e_4 - f_5e_5 + f_6e_6)\cos\theta_2 = M_2
\end{aligned}
\tag{9.1}
$$

where h_i and θ_i are the translational displacement and the angular displacement of the mass m_i, respectively, and J_i is its moment of inertia, and F_i and M_i are the external force and moment. f_i is the sum of the restoring force and damping force of the ith spring/damping element with the eccentricity e_i, and $f_{cm,i}$ is the damping force provided by the bearing of the supported mass. For the preliminary simulations, we assume the following mechanical properties of the system:

$$
\begin{aligned}
& m_1 = m_2 = 237\text{kg},\ J_1 = J_2 = 5275\text{kgm}^2 \\
& k_1 = 14.8\text{kN/m},\ k_2 = 1.1k_1,\ k_3 = 0.9k_2, \\
& k_4 = 1.2k_3,\ k_5 = 0.95k_4,\ k_6 = 1.05k_5, \\
& c_i = 10\text{kg/s}\ (i = 1, \ldots 6),\ c_{m,1} = c_{m,2} = 75\text{kg/s}
\end{aligned}
\tag{9.2}
$$

Assuming the initial values of $h_1 = 0.03m$ and $\theta_1 = 5\pi/180$, Fig.9.4 shows the free vibrations of the model of the form shown in Fig.9.3. Meanwhile, the natural frequen-

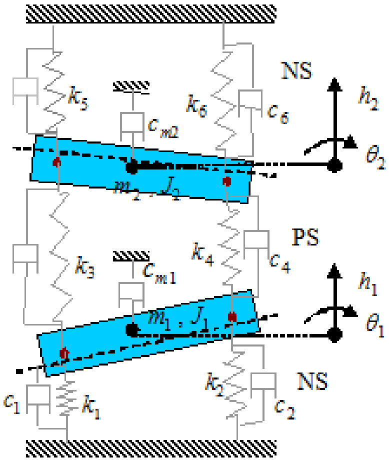


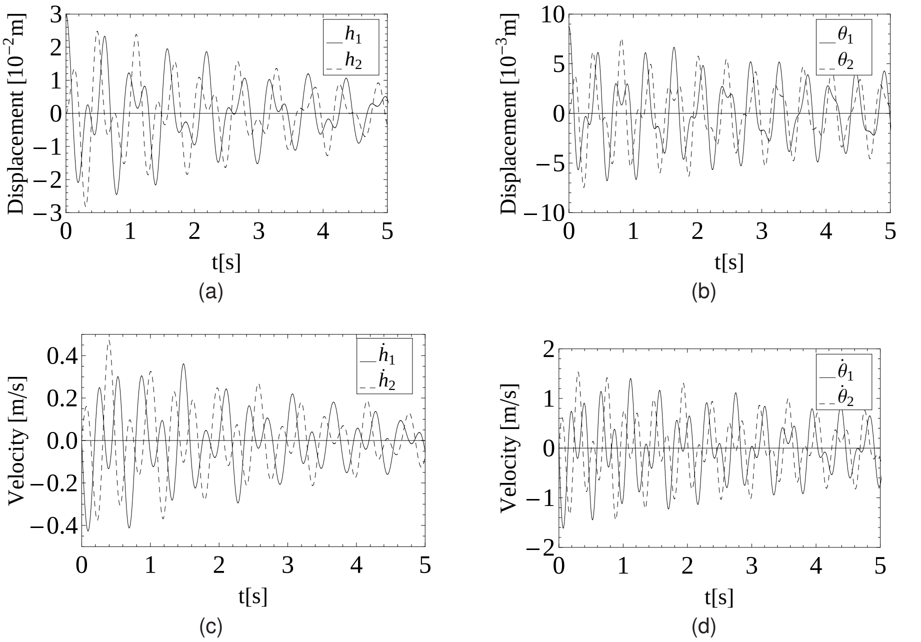
Fig. 9.3: Schematic representation of the emulated four-DoF system

cies of the four-DoF system linearized at the equilibrium point $0, 0, 0, 0, 0, 0, 0, 0$ are 4.27, 3.20, 2.45 and $1.85Hz$.

9.3.1 The masses and the bearings

The rig contains two masses moving translational on two linear guides as shown in Fig. 9.5. Each mass is composed of four rotational blocks, a translational block and some other steel elements above the linear bearings. The masses are adjustable through changing the number of the rotational blocks. Two translational blocks, similarly as two carts, are fixed on four sliding blocks which allow smooth movement on two linear guides as shown in Fig. 9.5. For each mass, four rotational blocks are fixed to one translational block by means of a rotational bearing and a shaft. Besides, there are some bearings between the rotational blocks that are used to connect the rotational blocks to some devices, such as actuators, springs and damping devices.

The test rig is design to be flexible so that a series of the configurations, from linear to nonlinear, from one- to four-DoF, can be considered. This requires the rig to be easily reconfigurable. To achieve this, the framework is fixed horizontally to the base of the laboratory. However, this will cause much out-plane bending on each mass. To reduce this bending to a permitted level, the four rotational blocks are expected to be



(a)

(b)

(c)

(d)

Fig. 9.4: Numerical simulations for the four-DoF system as shown in Fig. 9.3 using the LSRT2 time integration method with $\gamma = 1 - \frac{\sqrt{2}}{2}$: (a) translational displacements h_1 and h_2; (b) rotational displacements θ_1 and θ_2; (c) translational velocities $\dot{h}_1$ and $\dot{h}_2$; (b) rotational velocities $\dot{\theta}_1$ and $\dot{\theta}_2$;

fixed together rigidly to provide a greater bending stiffness. Likewise, the rotational bearings as well as their corresponding shafts are designed to be rigid enough to support the unbalance of the rotational blocks. With respect to vertical orientation, the horizontal orientation also introduces more friction. This imposes higher requirements on the linear bearings. The used linear bearings have the following physical details: dynamic loading 38740N and static loading 83060N. This is enough to support the masses on them either statically or dynamically. Moreover, the friction of the bearings will be considered in the parameter $c_{m,1}$ and $c_{m,2}$ that will be identified before real-time tests.

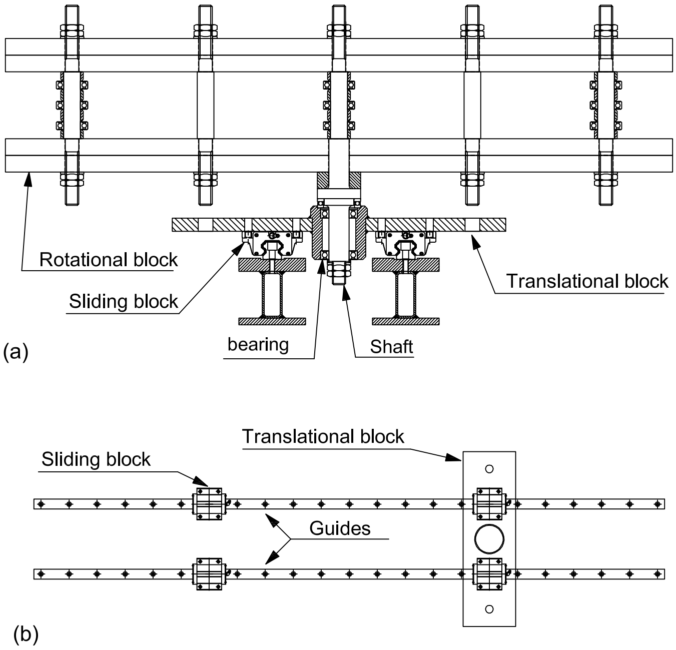
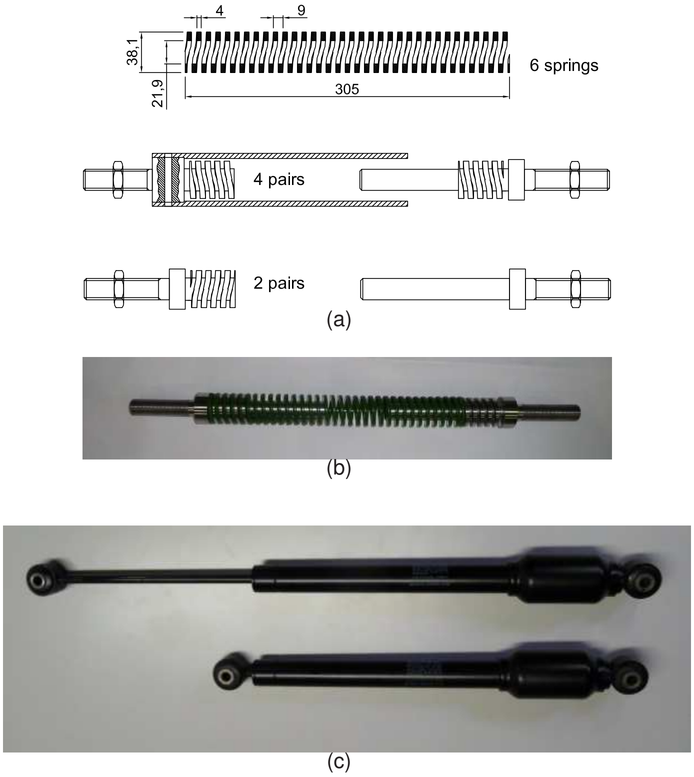


Fig. 9.5: Main configurations of the masses and the bearings

9.3.2 Springs and damping devices

The masses of the PS are linked by springs and damping devices. Six springs were considered for the rig. The springs selected are of the rectangular wire section type with a spring rate of 14.8kN/m, wire section of $b \times h$ = 8.1$mm \times$4.0mm, outer coil diam-



(a)

(b)

(c)

Fig. 9.6: (a) A coil spring and its configurations with continuous and discontinuous supports; (b) detail of mounting spring; and (c) dampers

eter of 40mm and a free length of 305mm. The springs provide the same maximum working deflection of 112.0mm. Among them, four are constructed with continuous support while two are constructed with discontinuous support as shown in Fig. 9.6(a) to introduce nonsmooth oscillations. In addition to the springs, two nonlinear dampers were selected as shown in Fig. 9.6(b). Both of them are of the same type, STAB-O-SHOC TA 20, with an maximum damping force of 550N, and a stroke of 200mm.

9.3.3 Electro-thrust actuators

There are four electro-thrust actuators produced by Parker. To gain a better idea of their sizes and performances, here are a few more specific physical details:

1. Type: ETB80;

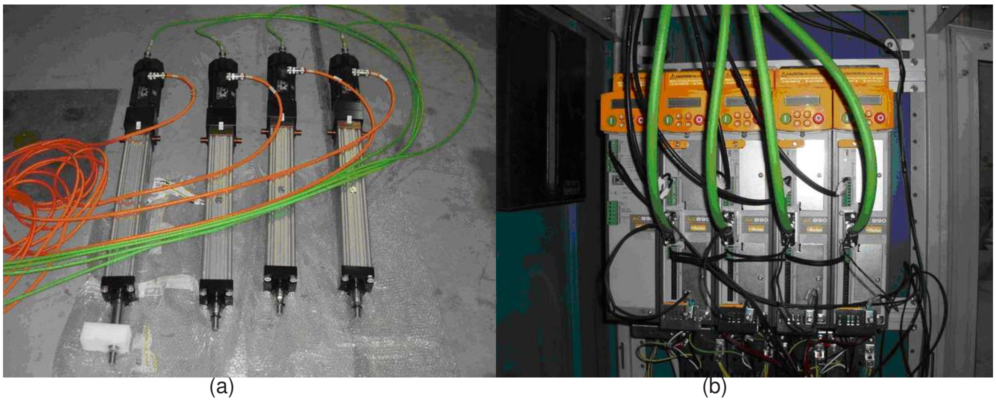


Fig. 9.7: (a) Electro-thrust actuators; (b) AC890 units

2. Available stroke: 500mm;
3. Max. permissible speed: 1340;
4. Max. thrust force: 8.3kN;

As shown in Fig. 9.7(a), Each of The actuators is equipped with a 3000rpm 400V permanent magnet motor, with a maximum torque of about 4Nm. Considering the reduction introduced by the ball-screw mechanism of the actuators, this translates into around 2kN and 0.5m/s.

The motors are synchronous electric machines with permanent magnets of 8 poles in the rotor, so they are operated by feeding them with a three-phase voltage of amplitude between 0 and 400V and a frequency between 0 and 200Hz. They are also equipped with an electro-mechanical brake, with coils that need to be energized with a 24Vdc supply to release the brake, as well as with a resolver, which is a position sensor for the motor and provides with a means of measuring the rotational speed.

The motors of the actuators are powered by four controlled inverters of the AC890 series as shown in Fig 9.7(b). The AC890 system generates the three-phase voltage of the necessary amplitude and frequency to make each of the motors rotate at the specified speed. The direct current supply required by the drives is obtained by a three-phase rectifier from the main supply of the laboratory.

It is possible to connect each of the drives to the PC via USB for configuration, for which the proprietary software DSELite is used.

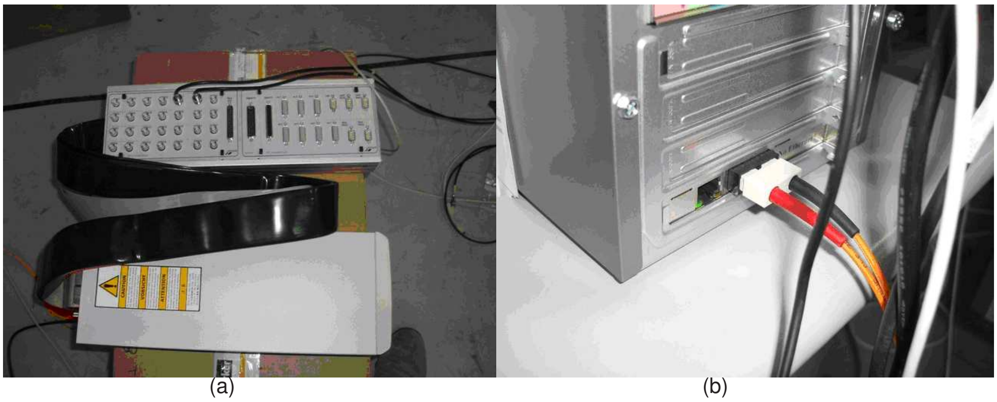


Fig. 9.8: (a) PPC 1103 unit and its connector panel; (b) detail of optical fiber connection to PC

9.3.4 Digital control and data-acquisition system

The real-time computing is done by the real-time system Power PC 1103 of dSpace, which is connected via optical fibre to a PC for programming and supervision purposes (shown in Fig. 9.8). It presents 8 digital to analogue converters, 4 parallel analogue to digital converters and 4 multiplexed (x4) analogue to digital converters, all of them for analogue voltage signals between -10 and 10V. It is therefore possible to acquire 20 different signals.

The programming is done with Matlab, Simulink, Real Time Workshop and Real Time Interface with great ease and the supervision is done with Control Desk.

9.3.5 Instrumentation package

There are four optical displacement sensors, each of which is fitted on one actuator body as shown in Fig 9.9. The devices measure their distances from any object placed in the way of their laser beams, as long as the distances are between 60 and 260mm. Each of them requires a 24Vdc power supply and provides with an analogue output between 4 and 20mA.

The output is transmitted along a shielded cable into a metallic connection box con-

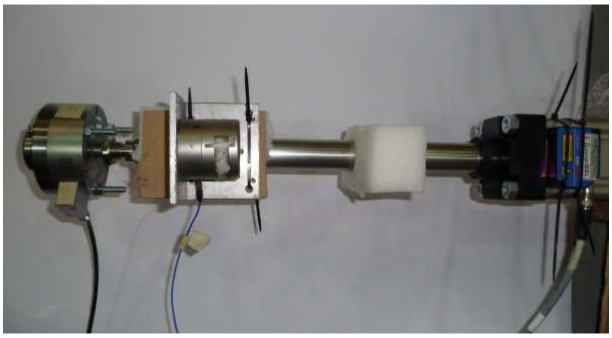


Fig. 9.9: Laser sensor mounted on actuator for measurement of displacement..

nected to the ground, from where it continues along a coaxial cable to the connection board of the Power PC 1103 unit.

In addition to the distance sensors, there are also four load cells and four accelemeters to that are connected the end of the piston to the tested specimen.

9.4 Different test configurations

To allow accurate evaluation of the time integration methods, some simple linear systems were required, in which the mass and stiffness properties can be clearly and accurately determined and the overall system can be reliably computed. In this case, the performance of the proposed integration methods can be studied by comparing between numerical simulations and experimental results. On the other hand, many complex nonlinear system were required to check the applicability of the integration methods to nonlinear and/or complex systems. In this section, two relatively complex and representative configurations for a four-DoF emulated system are presented in detail.

The first case is a four-DoF emulated system partitioned into single PS and two NSs as shown in Fig. 9.10. In this case, two NSs are considered each of which presents a Two-DoF subsystem while single NS is tested as a floating subsystem which has two masses connected by two parallel springs. For this case, the test rig in Fig. 9.1 is required to be rearranged: uninstalling springs noted S2 and S3.

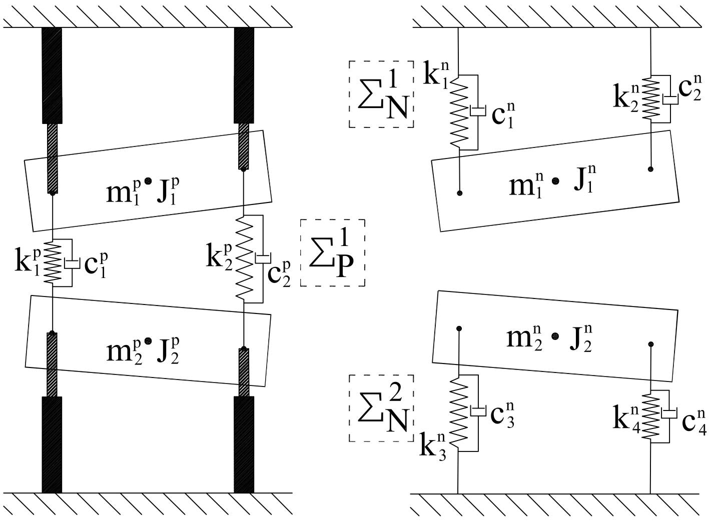


Fig. 9.10: Schematic test configuration with single physical and two numerical substructures.

Differently from the first configuration, the second case contains two physical and single numerical substructures as shown in Fig. 9.11. For this case, the NS has the same configuration as the PS of the first case. For the PS, the springs of each mass have to be connected to the framework. For the sake of ease of implementation, these springs S2 and S3 are connected to the inner joints as shown in Fig 9.1 and the springs between two masses are required to removed.

9.5 Conclusions

In order to evaluate the performance of the partitioned algorithms for RTDS, a novel test rig has been designed in this chapter. It is capable to perform both linear and nonlinear substructure tests on Multiple-DoF systems in real time. It is characterized by two masses and four DoFs. The nonlinearities at these stages derive from mass

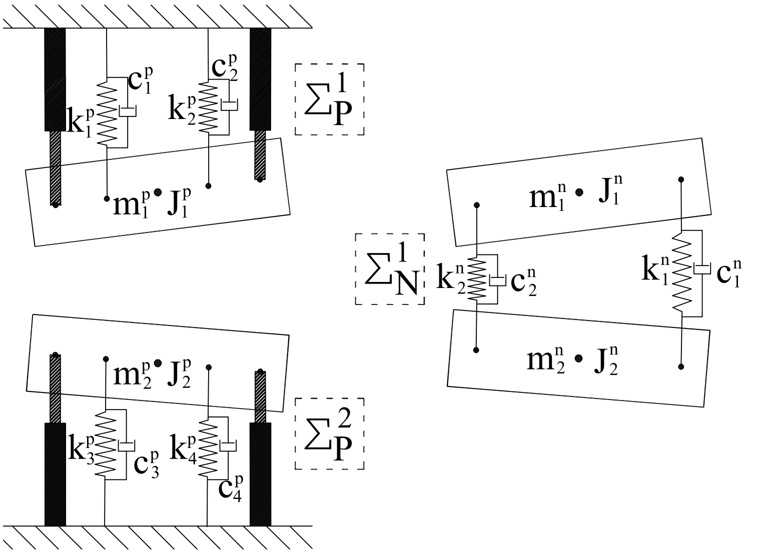


Fig. 9.11: Schematic test configuration with two physical and single numerical substructures.

rotations, springs with discontinuous supports and nonlinear dampers. This chapter focused on the design of this test rig and described the equipment that was prepared to conduct this research.

CHAPTER 10

CONCLUSIONS AND FUTURE PERSPECTIVES

10.1 Summary

The main objective of this work is to implement first-order integrators, the LSRT methods, in monolithic and partitioned ways for dynamic simulations and substructure tests. Four main contributions of this thesis can be summarized as follows:

1. To study the first-order time integration methods in terms of accuracy, stability as well as high-frequency dissipation property;
2. To extend the monolithic time integration methods to the nonlinear case and adopt subcycling strategies;
3. To develop partitioned time integration methods based on acceleration continuity and extend them to two subcycling cases with staggered and parallel solution procedures;
4. To develop partitioned time integration methods based on velocity continuity and a projection method.
5. To design and construct a flexible test rig for experimental validations of all the monolithic and partitioned time integration methods.

In this process, a great number of analytical approaches were presented or flexibly adopted to study some numerical aspects of the proposed integration methods. Subsequently, a set of linear and nonlinear test problems were numerically simulated to validate the analytical results and to further investigate the performance of these methods. In addition, several substructure tests of the coupled spring-pendulum system were carried out and a test rig of a Four-DoF system was designed to experimentally investigate the behaviour of the proposed methods. From the available numerical or experimental results, it was illustrated that the proposed methods exhibited many favourable properties: accuracy, stability, high-frequency dissipation, time efficiency, ease of implementation and so forth.

10.2 Conclusions

As presented in the previous section, this thesis presented a series of monolithic and partitioned time integration methods for substructure tests and numerical simulations. The first chapter is the introduction, including the context and the objective of this research. In Chapter 2, a literature overview related to the thesis was presented in the order: the widely-used testing methods; the RTDS method and the PsD testing with dynamic substructuring; several direct time integration methods; and partitioned time integration methods. This overview, on the one hand, provided some challenges posed by the RTDS and the PsD testing with dynamic substructuring; on the other hand, introduced in length some integration methods which were used for comparison in the other chapters.

In Chapter 3, the utilized time integration methods were introduced. Their applicabilities to structural problems were checked in terms of accuracy and stability. Based on the first-order form of the structural problem, the direct integration methods were observed to have the same order of accuracy as dealing with first-order systems. For the stability analysis of one- and two-stage methods, the energy approach was used. Through serious reformulations, the energy balance equation of each method applied to an undamped free vibration problem was simplified to be a formula whose sign is dependent on the parameter γ, and therefore, the stabilities of the methods were judged. The nonlinear performances of the time integration methods were investigated through numerical simulations of an uncoupled spring-pendulum system. Compared with the Explicit Newmark method and the Newmark-Chang method, the

LSRT methods were discovered to be more efficient, easy to implement, unconditionally stable and endowed with high-frequency dissipation.

For the monolithic time integration methods, the first contribution of the thesis was to extend them to nonlinear cases. The spring-pendulum system was again adopted and partitioned into physical and numerical substructures. The stabilities of the monolithic methods applied to the coupled spring-pendulum system were studied by means of the zero stability analysis. Through numerical simulations on the coupled spring-pendulum system, all the monolithic methods were proved to be first-order accurate and it was observed that the reduction of the accuracy is caused by the adoption of the previous coupling force. These favourable properties were shown when considering nonlinearities both in the numerical and the physical substructures.

The second contribution of the thesis to the monolithic methods was the adoption of subcycling strategies. For the LSRT methods, two subcycling strategies, the equilibrium-based interpolation and the differentiation-based interpolation, were developed. For comparison, the Explicit Newmark method and the Newmark Chang method were also endowed with a subcycling strategy that was developed to keep the original order of accuracy. All the subcycling strategies were validated through numerical simulations on the coupled spring-pendulum system.

In view of hybrid testing of complex emulated structures, several novel partitioned methods were developed based on the LSRT1 method and the LSRT2 method, respectively. The partitioned methods use the FETI technique to split the overall system into two or more subdomains coupled to each other with Lagrange multipliers. These Lagrange multipliers ensured the continuity of the coupled solutions along the interfaces.

In Chapter 5, a novel type of partitioned integration methods were conceived based on the continuity of the acceleration. This type of partitioned methods was implemented in as follows: i) solve the Lagrange multipliers at the beginning of the step or the inner stage; ii) advance each subdomain independently and simultaneously. It was observed that only one communication at the beginning was needed and this made it possible for parallel simulations. The accuracy analyses of the partitioned methods were performed in a compact form in order to include al subdomains altogether. Both the differential and algebraic vector was proved to have the same order of accuracy of the progenitor time integration methods. Their stability analyses were conducted by using the linear recurrence approach and the partitioned methods were proved to be unconditionally stable. After these theoretical analyses, the favourable properties were rechecked through numerical simulations on a Single-, Two-, Three-, and Four-DoF split-mass systems as well as the coupled spring-pendulum system.

The partitioned methods based on acceleration continuity were also extended in Chapter 6 to incorporate subcycling strategies, resulting in novel staggered partitioned methods. In these methods, the emulated system is partitioned into two subdomains: one is integrated with the coarse time step and the other with the fine time step. In addition to the computation of the Lagrange multiplier vector and the integration of the subdomains, a linear interpolation is required before advancing the solution of the subdomain with the fine time step. This prevents the partitioned methods to be implemented in parallel simulations.

For these staggered partitioned methods, the accuracy analyses were performed by setting a reference solution procedure by which each subdomain advanced separately with exact Lagrange multipliers and both the differential vectors and the Lagrange multiplier vector were proved to preserve the same order of accuracy as the progenitor monolithic method. For comparison, the GC method was also proved in a similar way to have a reduction of accuracy order with respect to its progenitor Newmark method. The inverse term **H** in the formula of the Lagrange multiplier vector of the GC method includes a Δt factor while the proposed staggered methods do not. It was illustrated in the accuracy analysis that this implied an order reduction of the GC method.

The stability of the proposed staggered methods were studied by means of the spectral approach on a Single-DoF split-mass system, and these methods frequently appears to be stable. Meanwhile, the components of the eigenvalues of the amplification matrix were investigated in depth. Four non-zero eigenvalues were included in the amplification matrix. Among them, only one pair of complex conjugate eigenvalues are principle eigenvalues, whereas the other two are spurious. For the complex conjugate eigenvalues, it was studied in terms of numerical damping ratio and frequency error. For the spurious eigenvalues, the one unitary is unitary was found to be corresponding to the ”equilibrium point” of the continuous DAE system and the other that was real and less than one was observed to be so owing to incomplete constraint.

When dealing with a partitioned system in the presence of floating DoFs, the number of unitary eigenvalues is greater than one which indicates weak instability and therefore the displacement solutions exhibit slightly drift phenomena. This is because the continuous DAEs system of index 1 has null eigenvectors. But it was observed that with proper choice of the parameter γ, the drift effect could be limited to be so small as to be negligible. Moreover, it was proved that the drift effect would not accumulate with the increasing number of the floating DoFs.

For RTDS tests, a type of parallel partitioned methods was conceived in Chapter 7. In these methods, the integration in the subdomain with the coarse time step advances 2 times (for the LSRT1-based case) or 4 times (for the LSRT2-based case) further than the system time step, in order to provide the information required in the integration of the subdomain with the fine time step in the next system time step. This makes the integration of both subdomains to advance simultaneously but independently. A series of analytical and numerical analyses were conducted for the proposed methods.

Differently, the parallel methods are not self-starting, and therefore, it was of crucial importance to choose sufficient variables when building the amplification matrix. Two codes were followed: the output x_{i+1} and the input x_i were parallel and the output included all the target variables while the input included all the initial variables in one loop of the solution procedure. The number of the eigenvalues of the obtained amplification matrix was 18 for a Single-DoF split mass problem. All the eigenvalues were classified into four groups, i.e. principle, unitary, zero and real, and they were analysed in terms of their causes and their effects on the stability and accuracy.

The fourth type of partitioned methods developed in Chapter 8 was based on a projection solution procedure as well as velocity continuity. These methods contain three steps in every stage: i) solve the subdomains independently; ii) project the solutions to the velocity constraint; and iii) update the Lagrange multipliers. The computation cost for the last two steps is negligible with respect to the first step. Therefore, methods enable parallel simulations in that all the subdomains advance independently only with information exchange at the end of each stage/step.

For this type of methods, the accuracy analysis was also conducted by using a reference solution procedure with constant Lagrange multipliers. Differently from previous methods, the local truncation error of the Lagrange multipliers was discovered to be one order lower for either LSRT1- or LSRT2-based methods. However, it was observed that this error had no propagation on the differential vectors and this made the differential vectors preserve the same order of accuracy of the progenitor integration method.

With regard to the stability analyses, the LSRT1-based projection method was proved to be unconditionally stable by using the linear recurrence approach. For the LSRT2-based projection method, the spectral analysis was adopted and the method appears to be unconditionally stable for all cases considered. Moreover, the linear recurrence approach was used to analyse a projection method based on the LSRT2 method but with only one projection. The resulting recurrence was observed to be stable for a first-order test problem but unstable for a second-order test problem. Besides, numerical analyses and simulations were carried out and the projection methods were validated to be accurate and efficient.

In order to evaluate the performance of the partitioned algorithms for RTDS, a novel test rig has been designed in Chapter 9. It is capable to perform both linear and nonlinear substructure tests on Multiple-DoF systems in real time. It is characterized by two masses and four DoFs. The nonlinearities at these stages derive from mass rotations and springs with discontinuous supports. Also nonlinear dampers can be introduced.

10.3 Future perspectives

Recommendations for future work are expressed herein regarding validations, improvements and applications of the proposed methods.

Regarding the validations of the proposed methods, a novel test rig has been designed and different tests will be performed in a near future. In this thesis, all numerical analyses were conducted without considering experimental errors and actuator dynamics. But these effects in typical situation of RTDS are large enough to have significant influence on the experiment results and in certain cases can lead to instability. In the tests, some linear/non-linear control techniques will be implemented and the integration methods will be investigated in the presence of experimental error and actuator dynamics.

With regard to the partitioned methods with acceleration continuity, the effect of the displacement drift-off error is due primarily to the index-1 DAE characteristics and is aggravated in the presence of floating DoFs. It was illustrated in Chapter 6 that this effect could be reduced by a proper choice of the parameter γ and the utilized time step. In order to reduce this effect, one way is to investigate the relationship of the drift-off effect with respect to the value of parameter γ and further design an estimate of the drift-off effect with the hope that this effect can be controlled or minimized. Another way is to adopt some stabilization techniques, such as the Baumgarte stabilization. Developments in reducing displacement drift-off error would be beneficial to extend the application of the partitioned method to longer-period simulations. Regarding the projection methods proposed, the aforementioned drift-off effect is not noticeable. But this type of methods treat all subdomains with the same time step. Future work is required to endow the methods with subcycling strategies, and furthermore, extend them to the case with a parallel solution procedure.

In addition, we want to extend the applicability range of the partitioned methods. The thesis provided a series of time integration methods and every method has different choices of the parameter γ. More work is required to compare the performances of different cases so as to provide some effective guides for their applications. Moreover, the systems studied in this thesis are of second-order. The objective to implement the first-order integrator is to solve mixed first- and second-order system. The concept of the partitioned integration methods can also be extended to couple the discretization in time of controller(s) and the solution of NS(s) together.

BIBLIOGRAPHY

M. Ahmadizadeh, G. Mosqueda, and A.M. Reinhorn. Compensation of actuator delay and dynamics for real-time hybrid structural simulation. *Earthquake engineering & structural dynamics*, 37(1):21–42, 2008.

M. Arnold, B. Burgermeister, and A. Eichberger. Linearly implicit time integration methods in real-time applications: DAEs and stiff ODEs. *Multibody System Dynamics*, 17:99–117, 2007a.

M. Arnold, B. Burgermeister, and A. Eichberger. Linearly implicit time integration methods in real-time applications: DAEs and stiff ODEs. *Multibody System Dynamics*, 17:99–117, 2007b.

P.V. Bayly and L.N. Virgin. An empirical study of the stability of periodic motion in the forced spring-pendulum. *Proceedings: Mathematical and Physical Sciences*, 443 (1918):391–408, 1993.

T. Belytschco, H-J. Yen, and Mullen R. Mixed methods for time integration. *Computer Methods in Applied Mechanics Engineering*, 17:259–275, 1979.

T. Belytschko and Y. Y. Lu. Explicit multi-time step integration for first and second order finite element semidiscretizations. *Computer Methods in Applied Mechanics and Engineering*, 108(3-4):353–383, 1993.

A. Blakeborough, M. S. Williams, A. P. Darby, and D. M. Williams. The development of real-time substructure testing. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 359(1786):1869–1891, 2001.

A. Bonelli and O. S. Bursi. Generalized-α methods for seismic structural testing. *Earthquake Engineering and Structural Dynamics*, 33:1067–1102, 2004.

A. Bonelli, O. S. Bursi, L. He, P. Pegon, and G. Magonette. Convergence analysis of a parallel interfield method for heterogeneous simulations with dynamic substructuring. *International Journal for Numerical Methods in Engineering*, 2008a. DOI: 10.1002/nme.2285.

A. Bonelli, O.S. Bursi, L. He, G. Magonette, and P. Pegon. Convergence analysis of a parallel interfield method for heterogeneous simulations with dynamic substructuring. *International Journal for Numerical Methods in Engineering*, 75:800–825, 2008b.

P. A. Bonnet, M. S. Williams, A. Blakeborough, S. A. Neild, D. P. Stoten, and C. A. Taylor. Real-time hybrid experiments with newmark integration, mcsmd outer-loop control and multi-tasking strategies. *Earthquake Engineering and Structural Dynamics*, 36(1):119–141, 2007.

P. A. Bonnet, M. S. Williams, and A. Blakeborough. Evaluation of numerical time-integration schemes for real-time hybrid testing. *Earthquake Engineering and Structural Dynamics*, 37(13):1467–1490, 2008.

P.A. Bonnet. *The development of multi-axis real-time substructure testing*. PhD thesis, University of Oxford, 2006.

D.L. Brown, R. Cortez, and M.L. Minion. Accurate projection methods for the incompressible navier-stokes equations. *J. Comput. Phys*, 168:464–499, 2001.

O. Brüls and J.C. Golinval. The generalized-α method in mechatronic applications. *ZAMM - Journal of Applied Mathematics and Mechanics*, 86(10):748–758, 2006.

Ph. Buchet and P. Pegon. PSD testing with substructuring: implementation and use. Technical Report I.94.25, E.C., Joint Research Centre, ELSA, Ispra, Italy, 1994.

B. Burgermeister, M. Arnold, and B. Esterl. Dae time integration for real-time applications in multi-body dynamics. *ZAMM - Journal of Applied Mathematics and Mechanics*, 86:759–771, 2006.

O. S. Bursi and P. B. Shing. Evaluation of some implicit time-stepping algorithms for pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, 25: 333–355, 1996.

O. S. Bursi, A. Gonzalez-Buelga, L. Vulcan, S. A. Neild, and D. J. Wagg. Novel coupling Rosenbrock-based algorithms for real-time dynamic substructure testing. *Earthquake Engineering and Structural Dynamics*, 37:271–288, 2008.

O.S. Bursi. *Analysis of control techniques and development of numerical methods for heterogeneous simulations with dynamic substructuring*. PhD thesis, Department of Mechanical Engineering, University of Bristol, March 2007.

O.S. Bursi. *New Approaches to Analysis and Testing of Mechanical and Structural Systems (eds. O.S. Bursi and D.J. Wagg)*, chapter Computational techniques for simulation of monolithic and heterogeneous structural dynamic systems. CISM, 2008.

O.S. Bursi and M. Mancuso. Analysis and performance of a predictor-multicorrector time discontinuous galerkin method in non-linear elastodynamics. *Earthquake engineering & structural dynamics*, 31(10):1793–1814, 2002.

O.S. Bursi, L. He, A. Bonelli, and P. Pegon. Novel generalized-α methods for interfield parallel integration of heterogeneous structural dynamic systems. *Journal of Computational and Applied Mathematics*, In Press, 2009.

A. Cardona and M. Geradin. Time integration of the equations of motion in mechanism analysis. *Computers & Structures*, 33:801–820, 1989.

T. F. Chan and B. Smith. Overlapping schwarz methods on unstructured meshes using non-matching coarse grids. *Numer. Math*, 73:149–167, 1996.

S. Y. Chang. Explicit pseudodynamic algorithm with unconditional stability. *Journal of engineering mechanics*, 128(9):935–947, 2002.

C. Chen and J.M. Ricles. Development of direct integration algorithms for structural dynamics using discrete control theory. *Journal of Engineering Mechanics*, 134(8): 676–683, 2008.

C. Chen, J. M. Ricles, and T. M. Marullo. Real-time hybrid testing using the unconditionally stable explicit cr integration algorithm. In *Proceedings of the 2nd International Conference on Advances in Experimental Structural Engineering*, pages 725–732, Dec. 2007.

C. Chen, J.M. Ricles, T.M. Marullo, and O. Mercan. Real-time hybrid testing using the unconditionally stable explicit cr integration algorithm. *Earthquake Engineering and Structural Dynamics*, 38:23–44, 2009.

J. Chung and G. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, 60:371–375, 1993.

A. Combescure and A. Gravouil. A numerical scheme to couple subdomains with different time-steps for predominantly linear transient analysis. *Computer Methods in Applied Mechanics and Engineering*, 191:1129–1157, 2002.

G. Dahlquist. A special stability problem for linear multistep methods. *BIT Numerical Mathematics*, 3:27–43, 1963.

G. Dahlquist. On accuracy and unconditional stability of linear multistep methods for second order differential equations. *BIT Numerical Mathematics*, 18:133–136, 1978.

W. J. T. Daniel. Subcycling first- and second-order generalizations of the trapezoidal rule. *International Journal for Numerical Methods in Engineering*, 42:1091–1119, 1998.

W.J.T. Daniel. Analysis and implementation of a new constant acceleration subcycling algorithm. *International Journal for Numerical Methods in Engineering*, 40:2841–2855, 1997.

A. P. Darby, A. Blakeborough, and D. M. Williams. Real-time substructure tests using hydraulic actuator. *Journal of Structural Engineering*, 125(10):1133–1139, 1999.

A. P. Darby, A. Blakeborough, and M. S. Williams. Improved control algorithm for real-time substructure testing. *Earthquake engineering and structural dynamics*, 30(3): 431–448, 2001.

A. P. Darby, D. M. Williams, and A. Blakeborough. Stability and delay compensation for real-time substructure testing. *Journal of Engineering Mechanics*, 128(12): 1276–1284, 2002.

J.D. Day. A minimum configuration l-stable fourth-order non-autonomous rosenbrock method for stiff differential equations. *Communications in applied numerical methods*, 1(6):293–297, 1985.

S. Dermitzakis and S. Mahin. Development of substructuring techniques for on-line computer controlled seismic performance testing. Technical Report UCB/EERC-85/04, Earthquake Engineering Research Center, University of California, Berkeley, 1985.

J. Dimig, C. Shield, C. French, F. Bailey, and A. Clark. Effective force testing: A method of seismic simulation for structural testing. *Journal of Structural Engineering*, 125(9):1028–1037, 1999.

P. Donzelli and G. Iazeolla. A dynamic simulator of software processes to test process assumptions. *Journal of Systems and Software*, 56(1):81 – 90, 2001.

U.E. Dorka and D. Heiland. Fast online earthquake simulation using a novel pc supported measurement and control concept. In *Proc. 4th Int. Conf. Structural Dynamics*, pages 636–645, Southhampton, 1991.

dSPACE. *Real-time interface, Implementation guide, Release 3.2*. Paderborn, Germany, 2001.

W. M. Elleithy and H. J. Al-Gahtani. An overlapping domain decomposition approach for coupling the finite and boundary element methods. *Engineering Analysis with Boundary Elements*, 24(5):391 – 398, 2000.

S. Erlicher, L. Bonaventura, and O. S. Bursi. The analysis of the Generalized-alpha method for non-linear dynamic problems. *Computational Mechanics*, 28(2):83–104, 2002.

C. Farhat and M. Chandesris. Time-decomposed parallel time-integrators: theory and feasibility studies for fluid, structure, and fluid-structure applications. *International Journal for Numerical Methods in Engineering*, 58:1397–1434, 2003.

C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32:1205–1227, 1991.

C. Farhat, L. Crivelli, and F. X. Roux. A transient feti methodology for large-scale implicit computations in structural mechanics. *International Journal for Numerical Methods in Engineering*, 37:1945–1975, 1994.

C. Farhat, L. Crivelli, M. Géradin, and J. Mandel. A scalable Lagrange multiplier based domain decomposition method for time-dependent problems. *International Journal for Numerical Methods in Engineering*, 37:71–107, 1995.

C. Farhat, C. Lacour, and D. Rixen. Incorporation of linear multipoint constraints in substructure based iterative solvers. part i: a numerically scalable algorithm. *International Journal for Numerical Methods in Engineering*, 34:997–C1016, 1998.

C. A. Felippa, K. C. Park, and C. Farhat. Partitioned analysis of coupled mechanical systems. *Computer Methods in Applied Mechanics and Engineering*, 190:3247–3270, 2001.

P. J. Gawthrop, S. A. Neild, A. Gonzalez-Buelga, and D. J. Wagg. Causality in real-time dynamic substructure testing. *Mechatronics*, 19(7):1105–1115, 2009.

M. Geradin and D. Rixen. *Flexible Multibody Dynamics*. John Wiley & Sons Ltd, 2000.

M. Geradin and D. Rixen. *Mechanical Vibrations: Theory and Application to Structural Dynamics*. John Wiley & Sons Ltd, 1997.

A. Gravouil and A. Combescure. Multi-time-step explicit-implicit method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 50:199–225, 2001.

U.A. Gumaste, K.C. Park, and K.F. Alvin. A family of implicit partitioned time integration algorithms for parallel analysis of heterogeneous structural systems. *Computational Mechanics*, 24(6):463–475, 2000.

E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer, New York, 2nd edition, 1996.

L. He. *Development of partitioned time integration schemes for parallel simulation of heterogeneous systems*. PhD thesis, University of Trento, 2008.

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

C. Hoff and P. J. Pahl. Practical performance of the θ1-method and comparison with other dissipative algorithms in structural dynamics. *Comput. Methods Appl. Mech. Eng.*, 67(1):87–110, 1988.

T. Horiuchi and T. Konno. A new method for compensating actuator delay in real-time hybrid experiments. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 359(1786):1893–1909, 2001.

T. Horiuchi, M. Nakagawa, M. Sugano, and T. Konno. Development of a real-time hybrid experimental system with actuator delay compensation. In *Proc. 11th World Conf. on Earth- quake Engineering*, pages 3088–3093, Acapulco, Mexico, June 23–28 1996.

T. Horiuchi, M. Inoue, T. Konno, and Namita Y. Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Eng. and Struct. Dyn.*, 28:1121–1141, 1999.

T. J. R. Hughes. *The Finite Element Method, Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1987.

T. J. R. Hughes and W. K. Liu. Implicit-explicit finite elements in transient analysis: stability theory. *Journal of Applied Mechanics*, 45:371–374, 1978.

T.J.R. Hughes. *Computational methods for transient analysis*, chapter Analysis of Transient Algorithms with Particular Reference to Stability Behavior. NORTH-HOLLAND, 1983.

C. Hung and S. El-Tawil. A method for estimating specimen tangent stiffness for hybrid simulation. *Earthquake engineering & structural dynamics*, 38(1):115–134, 2009.

L.O. Jay and D. Negrut. Extensions of the hht-α method to differential-algebraic equations in mechanics. *Electronic Transactions on Numerical Analysis*, 26:190–208, 2007.

R. Y. Jung, P. B. Shing, E. Stauffer, and B. Thoen. Performance of a real-time pseudodynamic test system considering nonlinear structural response. *Earthquake Engineering and Structural Dynamics*, 2007. in press.

M. Klisinski and A. Mostrom. On stability of multitime step integration procedures. *Journal of Engineering Mechanics*, 124:783–793, 1998.

S. Krenk and J.R. Hogsberg. Properties of time integration with first order filter damping. *International Journal for Numerical Methods in Engineering*, 64:547–566, 2005.

R. Kubler and W. Schiehlen. Modular simulation in multibody system dynamics. *Multibody System Dynamics*, 4:107–127, 2004.

K.B. Kyrychko1, Y.N . andBlyuss, A. Gonzalez-Buelga, S.J. Hogan, and D.J. Wagg. Real-time dynamic substructuring in a coupled oscillatorcpendulum system. *Proceedings-Royal Society. Mathematical, physical and engineering sciences*, 462 (2068):1271–1294, 2006.

C.P. Lamarche, A. Bonelli, O.S. Bursi, and R. Tremblay. A rosenbrock-w method for real-time dynamic substructuring and pseudo-dynamic testing. *Earthquake Engineering and Structural Dynamics*, 38:1071–1092, 2009.

J. D. Lambert. *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*. Wiley, 1991.

N.A. Lemos. Remark on rayleigh's dissipation function. *American Journal of Physics*, 59(7):660–661, 1991.

M. Leonard. *Elements Of Vibration Analysis*. Mcgraw Hill, 1986.

C. Lunk and B. Simeon. Solving constrained mechanical systems by the family of newmark and α -methods. *Zeitschrift für angewandte Mathematik und Mechanik*, (86):747–840, 2006.

G. Magonette, P. Pegon, J. F. Molina, and Ph. Buchet. Development of fast continuous pseudodynamic substructuring tests. In *Second World Conference on Structural Control*, Kyoto, Japan, 28 June - 1 July 1998.

J.P. Meijaard. Application of runge-kutta-rosenbrock methods to the analysis of flexible multibody systems. *Multibody System Dynamics*, 10:263–288(26), 2003.

M. Nakashima. Integration techniques for substructure pseudo dynamic test. In *Proceedings of the Fourth U.S. National Conference on Earthquake Engineering*, volume 12, pages 515–524, Palm Springs, USA, May 20-24 1990.

M. Nakashima. Development, potential, and limitations of real-time online (pseudo-dynamic) test. *Philosophical Transactions of the Royal Society*, 359:1851–1867, 2001.

M. Nakashima and N. Masaoka. Real-time on-line test for mdof systems. *Earthquake Engineering and Structural Dynamics*, 28:393–420, 1999.

M. Nakashima, H. Kato, and E. Takaoka. Development of real-time pseudo dynamic testing. *Earthquake Engineering and Structural Dynamics*, 21:79–92, 1992.

K. B. Nakshatrala, K. D. Hjelmstad, and D. A. Tortorelli. A feti-based domain decomposition technique for time-dependent first-order systems based on a dae approach. *International Journal for Numerical Methods in Engineering*, 75:1385–1415, 2008.

P. B. Nakshatrala, K. B. Nakshatrala, and D. A. Tortorelli. A time-staggered partitioned coupling algorithm for transient heat conduction. *International Journal for Numerical Methods in Engineering*, 78:1387–1406, 2009.

A.H. Nayfeh and H.N. Arafat. An overview of non-linear system dynamics. In Inman D.J. Ewins, D.J. and, editor, *Structural Dynamics @ 2000: Current Status and Future Directions*. Research Studies Press Ltd, 2001.

M.O. Neal and T. Belytschko. Explicit-explicit subcycling with non-integer time step ratios for structural dynamics systems. *Computers and Structures*, 31:871–880, 1989.

P. Negro and G. Magonette. Experimental methods in structural dynamics. *European Earthquake Engineering*, 12(1):29–39, 1998.

S. A. Neild, D. P. Stoten, D. Drury, and D. J. Wagg. Control issues relating to real-time substructuring experiments using a shaking table. *Earthquake engineering and structural dynamics*, 34(9):1171–11192, 2005.

N. N. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division ASCE*, 85:67–94, 1959.

K. C. Park and C. A. Felippa. A variational principle for the formulation of partitioned structural systems. *International journal for numerical methods in engineering*, 47: 395–418, 2000.

K. C. Park, M. R.. Justino, and C. A. Felippa. An algebraically partitioned FETI method for parallel structural analysis: Algorithm description. *International Journal for Numerical Methods in Engineering*, (40):2717–2737, 1997.

P. Pegon. *New Approaches to Analysis and Testing of Mechanical and Structural Systems (eds. O.S. Bursi and D.J. Wagg)*, chapter Continuous PsD Testing With Substructuring. CISM, 2008.

P. Pegon and G. Magonette. Continuous PSD testing with non-linear substructuring: Presentation of a stable parallel inter-field procedure. Technical Report I.02.167, E.C., JRC, ELSA, Ispra, Italy, 2002.

P. Pegon and G. Magonette. Continuous PsD testing with non-linear substructuring: using the operator splitting technique to avoid iterative procedures. Technical Report SPI.05.30, E.C., JRC, ELSA, Ispra, Italy, 2005.

P. Pegon and A.V. Pinto. Pseudo-dynamic testing with substructuring applications at the ELSA laboratory. *Earthquake Engineering and Structural Dynamics*, 29(7): 905–925, 2000.

A. Prakash. *Multi-time-step Domain Decomposition and Coupling Methods for Non-Linear Structural Dynamics*. PhD thesis, University of Illinois at Urbana-Champaign, 2007.

A. Prakash and K. D. Hjelmstad. A FETI-based multi-time-step coupling method for Newmark schemes in structural dynamics. *International journal for numerical methods in engineering*, 61:2183–2204, 2004.

H.H. Rosenbrock. Some general implicit processes for the numerical solution of differential equations. *Computer Journal*, 5:329–330, 1963.

R. Sajeeb, D. Roy, and C. S. Manohar. Numerical aspects of a real-time substructuring technique in structural dynamics. *International journal for numerical methods in engineering*, 72(11):1261–1313, 2007.

J. C. Samin, O. Brüls, J. F. Collard, L. Sass, and P. Fisette. Multiphysics modeling and optimization of mechatronic multibody systems. *Multibody System Dynamics*, 18: 345–373, 2007.

V. Saouma and M. V. Sivaselvan. *Hybrid Simulation Theory, Implementation and Applications*. Longman Scientific and Technical, Routledge, USA, 2008.

A.P. Seyranian and A.A. Mailybaev. *Multiparameter Stability Theory With Mechanical Applications (series On Stability, Vibration And Control Of Systems, Series A Vol. 13)*. World Scientific Publishing Company, 2004.

P. B. Shing and S.A. Mahin. Experimental error effects in pseudodynamic testing. *Journal of Engineering Mechanics, ASCE*, 116:805–821, 1990.

P. B. Shing, M. Nakashima, and O.S. Bursi. Application of pseudodynamic test method to structural research. *Earthquake Spectra*, 12:29–56, 1996.

P.B. Shing. *New Approaches to Analysis and Testing of Mechanical and Structural Systems (eds. O.S. Bursi and D.J. Wagg)*, chapter Real-time hybrid testing techniques. CISM, 2008.

P.B. Shing and S.A. Mahin. Pseudodynamic test method for seismic performance evaluation: theory and implementation. Technical Report UCB/EERC-84/01, Earthquake Engineering Research Center, University of California, Berkeley, 1984.

P.B. Shing, M. Vannan, and E. Cater. Implicit time integration for pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, 20:551–76, 1991.

S. S. Shome, E. J. Haug, and L. O. Jay. Dual-rate integration using partitioned Runge-Kutta methods for mechanical systems with interacting subsystems. *Mechanics Based Design of Structures and Machines*, 32:253–282, 2004.

T. Steihaug and A. Wolfbrandt. An attempt to avoid exact jacobian and nonlinear equations in the numerical solution of stiff differential equations. *Math. Comp.*, 33 (521-534), 1979.

A. Stuart and A.R. Humphries. *Dynamical Systems and Numerical Analysis*. Cambridge Monographs on Applied and Computational Mathematics, 1996.

K. Takanashi and M. Nakashima. Japanese activities on on-line testing. *Journal of Engineering Mechanics*, 113(7):1014–1032, 1987.

J.G. Verwer. An analysis of rosenbrock methods for nonlinear stiff initial value problems. *SIAM Journal on Numerical Analysis*, 19(1):155–170, 1982.

L. Vulcan. *Discrete-time analysis of integrator algorithms applied to S.I.S.O. adaptive controllers with minimal control synthesis*. PhD thesis, University of Trento, 2006.

D. J. Wagg and D. P. Stoten. Substructuring of dynamical systems via the adaptive minimal control synthesis algorithm. *Earthquake Engineering and Structural Dynamics*, 30:865–877, 2001.

M.I. Wallace, D.J. Wagg, and S.A. Neild. An adaptive polynomial based forward prediction algorithm for multi-actuator real-time dynamic substructuring. *Proceedings of the Royal Society A*, 2064(461):3807–3826, 2005.

R. Weiner, M. Arnold, P. Rentrop, and K. Strehmel. Partitioning strategies in runge-kutta type methods. *IMA journal of numerical analysis*, 13:303–319, 1993.

Y.K. Wen. Method for random vibration of hysteretic systems. *Journal of Engineering Mechanics*, 102:249–263, 1976.

D. M. Williams. *Numerical modelling of real-time sub-structure testing*. PhD thesis, Mathematical,Physical & Life Sciences Division - Engineering Science, University of Oxford, 2000.

M. S. Williams and A. Blakeborough. Laboratory testing of structures under dynamic loads: An introductory review. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 359(1786):1651–1669, 2001.

W.L. Wood. *Practical Time-Stepping Schemes*. Clarendon Press, Oxford, 1990.

B. Wu, H. Bao, J. Ou, and S. Tian. Stability and accuracy analysis of the central difference method for real-time substructure testing. *Earthquake Engineering and Structural Dynamics*, 34:705–718, 2005.

B. Wu, Q. Wang, P. B. Shing, and J. Ou. Equivalent force control method for generalized real-time substructure testing with implicit integration. *Earthquake Engineering and Structural Dynamics*, 36:1127–1149, 2007.

Y. S. Wu and P. Smolinski. A multi-time step integration algorithm for structural dynamics based on the modified trapezoidal rule. *Computer Methods in Applied Mechanics and Engineering*, 187(3-4):641 – 660, 2000.

J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM Review*, 40:857–914, 1998.

J. Yen, L. Petzold, and S. Raha. A time integration algorithm for flexible mechanism dynamics: The dae alpha-method. Technical Report 3-4, 1998.

Y. Zhang, R. Sause, J. Ricles, and C. Naito. Modified predictor-corrector numerical scheme for real-time pseudo dynamic tests using state-space formulation. *Earthquake engineering & structural dynamics*, 34(3):271–288, 2005.

Z. Zheng and L. Petzold. Runge-kutta-chebyshev projection method. *J. Comput. Phys.*, 219(2):976–991, 2006.

Z. Zheng, B. Simeon, and L. Petzold. A framework for the analysis of second order projection methods. *Applied Numerical Mathematics*, 2009. submitted.