Qualitative and Quantitative Investigation of Planar Multibody Dynamics Encompassing GSSSS Family of Algorithms - A Novel In-depth Look

A THESIS

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL

OF THE UNIVERSITY OF MINNESOTA

BY

Tsung-Hui Huang

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

Master of Science in Mechanical Engineering

Dr. K. Tamma

December, 2014

Acknowledgements

I would like to thank Prof. Kumar K. Tamma for his guideness through out all my graduate school career. Without him, I won't be able to see how big the world is.

I would also like to thank Masao Shimada, Rohit Deokar, Tony Paganelli, David Tae, Tao Xue and Prof. Bing Chen for their collaboration, advice, and friendship along the way.

Dedication

To my loving family and Kuo-Yu Yang.

Abstract

Growing interest in the simulation of constrained rigid/flexible multibody systems has prompted the development of a wide variety of time integration methods for the differential-algebraic equations (DAEs) inherent to these systems. The vast majority of these methods require reformulation of the natural index 3 equations of motion leading to additional computational cost and the need to control drift phenomena, citing instability of direct index 3 methods. In the past 50 years, most researchers developed varieties of ways to decreasing the influence of instability of index 3 DAE system, and also sought for distinct formulation to equation of motion for the purposes of better physics. This research, within the generalized single step single solve (GSSSS) family of algorithms, investigates algorithm designs and framework capable of overcoming the instability and various problems encountered by previous researchers. A precise understanding of the equation of motion time level concept as well as novel methods for extending linear parent algorithms to nonlinear dynamic applications has enabled an indepth search of unique ways to the area. Also, various formulations of equations of motion and stabilization techniques are investigated which enable various approaches for simulating rigid/flexible body physics. In the end, design spaces for simulation in multibody dynamics algorithmic frameworks are identified which overcome previous limitations and are capable of providing stable, robust, and accurate simulating DAEs for both rigid and rigid/flexible multibody dynamics applications.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

- Chapter 1 introduces the goals and motivation of the research presented in this thesis.
- Chapter 2 provides background information and an overview of previous research about the GSSSS family of algorithms, and its extension to nonlinear problems.
- Chapter 3 describes the possible methods of extending these algorithms to solve differential-algebraic equations in multibody dynamics with different Index.
- Chapter 4 illustrates the results of the numerical examples for which these methods were tested.
- Chapter 5 presents a final discussion of the analyses presented in the thesis and summarizes the conclusions drawn within.

1.1 Research Objectives and Motivation

The purpose of this research to qualitatively and quantitatively investigate a new framework termed GSSSS family of methods through multibody dynamics (MBD) and illustrate the proper

development and design of robust time integration algorithms for them. The usage of GSSSS family of algorithms will help to build robust time integration algorithms for multibody dynamics. Although several publications are reviewed, it is still unclear for a beginner in this field to start proper simulation with proper integration algorithms and proper stabilization techniques. The literature is very poor in this regard from the view point of the novice and/for the beginning researcher. As a result, this work primarily focuses on investigating low frequency dynamic applications with constraints that are frequently encountered in vibration and dynamics of mechanical, aerospace, and civil engineering structures, which are frequently referred to as inertial value problems. It provides several perspectives as the following:

1) Analyse the existing key restrictions in the current methodology used in codes to simulate fully rigid systems (kinematics), as well as codes used to simulate coupled rigid and flexible systems.
2) Leverage the general nature of the GSSSS family of algorithms, and recently developed understanding of its extension to nonlinear applications, to search for algorithms and design and current approaches which may be capable of overcoming several of these existing restrictions.
3) Develop a framework with computationally attractive features which allows direct coupling of rigid and flexible bodies for efficient combined multibody system simulations.

The goal within the area of multibody dynamics is to describe the dynamic behaviour of systems interconnected through constraint equations. These bodies, which may be seen as rigid or flexible bodies, are connected through joints which serve as additional constraints to the general dynamic system. In addition to the constraints imposed by joints interconnecting multiple bodies, there may also be geometric constraints such as analytically rigid components or predefined trajectories. These constraints transform the standard set of ordinary differential equations

(ODEs) into a set of differential-algebraic equations (DAEs). The feature to simulate the motion and deformation are of significant interest to the engineering community. Problems such as vehicle dynamics, analysis of wind turbines, or robot dynamics are few examples which fall within the area of multibody dynamics.

Description of rigid/flexible body requires a deep understanding of the motion of multibodies. The motion of bodies has been the subject of some of the earlier researches pursued in three different fields, namely, rigid body mechanics, structural mechanics, and continuum mechanics. The term *rigid body* implies that the deformation of the body is assumed negligible. The motion of a rigid body in space can be completely described by generalized coordinates. However, the resulting mathematical model, in general, is highly nonlinear because of the large body rotation and translation. On the other hand, the term *structural dynamics* denotes that deformation is the main concern and large body rotations and translation may be involved. From the combination of these two studies, rigid body and structural mechanics, results in a field relevant to *continuum mechanics*, wherein the general body motion is considered. However the mathematical model has the disadvantages of the previous cases, namely nonlinearity and large dimensionality. It is also worth noting that in structural mechanics, the deformation of a body is in description of its local body coordinates. However, in rigid body mechanics, the introduction of a global frame of reference is necessary. Therefore, in the field of multibody dynamics, the selection of reference formulation becomes a big issue and worth discussion in this research to clarify the pros and cons.

Currently those who wish to simulate a system which has both rigid and flexible bodies must utilize recently developed codes for coupled rigid/flexible which depend almost exclusively on

particular implicit or certain explicit time integration methods. As codes intended to solve structural dynamics problems most frequently utilize implicit time integration schemes, the need to turn to only explicit or specialized schemes when coupling rigid and flexible bodies is certainly undesirable. A computational framework capable of efficiently and robustly simulating systems of both rigid and flexible bodies would be very essential for the community. As such, investigations underlying these restrictions (time integration) is one of the main objectives of this research.

As mentioned above, most multibody systems of practical interest involve large translation and rotations. As such, even for very simple linear material models, these large displacements make the problem highly geometrically nonlinear. This nonlinearity is problematic in that it greatly complicates integrating the spatially discrete equation of motion. Algorithms which for linear systems can be proven to be unconditionally stable are no longer guaranteed to provide a solution in the nonlinear case. Recently, research on the extension of the generalized single step single solve (GSSSS) family of implicit time integration algorithms has provide new insight into robust simulation of nonlinear elastodynamic systems. Concepts such as the normalized time weighted residual procedure and the importance of consistency in the equation of motion time level enable the possibility of new insights into the development of methods which may overcome the restrictions which fostered optimism that the GSSSS family may contain algorithms capable of overcoming the numerical difficulties encountered in the realm of highly nonlinear constrained dynamic simulations. The general nature of the GSSSS algorithms and the deep understanding of implementation for nonlinear equations makes this research unique to the community. This is in the following sense. A vast majority of other researchers in the area are simply reusing a handful of well-known algorithms used in linear dynamics whereas we have the unique benefit of testing new and optimal algorithms which are almost entirely unknown to others to-date.

The spatial discretization has significant impact on the resulting system of equations. In general, for structural dynamics application, the standard Galerkin finite element is used to take a continuous system and discretize it into a mesh of nodes and elements. The outcome of this descretization is often that some elements within the mesh have significantly higher frequencies, are considered to be a ”stiff” numerical system. Integration of such system has been the topic of research for many years and has led to the development of algorithms which exhibit controllable numerical dissipation. This dissipation may be thought of as artificial dampening of high frequency modes of the system, which for structural dynamic applications also play a key role in the solution. Algorithms which involve controllable numerical dissipation result in a non-physical damping of total system energy. For this reason the current research set out to determine the applications of which controllable numerical dissipation to fosters great numerical solutions, though the general nature of GSSSS algorithms always includes the capability.

As it stands to-date, there are two main families of algorithms which fall under the umbrella of numerically non-dissipative linear multistep (LMS) methods: Symplectic algorithms and algorithms which conserve energy and momentum exactly. Symplectic algorithms are characterized by the ability to conserve phase space volume which is very desirable in area such as molecular dynamics, gravitational problems, etc. In terms of total system energy however, symplectic algorithms provide bounded oscillatory energy for conservative systems for a fixed time step. On the other hand, energy and momentum conserving methods (EMM) are able to conserve total energy, linear momentum, and angular momentum exactly over a time step but do not perfectly preserve phase space volume. Implementation aspects for symplectitc algorithms are fairly well understood as the algorihm itself is not directly dependent on the material model. Conversely, implementation of EMM algorithms is still an active research area. Complexities arise from

nonlinear material models as well as problems related to the overall stiffness of the system. The key desirable quality of an EMM algorithm is that one is able to claim unconditional stability in terms of system energy, even for highly nonlinear problems. Recent research has shown each of these properties (symplectic and EMM) may be derived for structural dynamic simulations by careful extension of the parent linear time integration algorithm to nonlinear applications, each with its own benefits and drawbacks. EMM however, gives rise to unsymmetric matrices.

1.2 Overview of Existing Methods Versus Current Research

We intend to focus our attention on modelling and simulation of structural and multibody systems consisting of both rigid and flexible bodies as opposed to only rigid or only flexible bodies. The difference between structural and multibody systems is the presence of constraint equations in the formulation of the latter. These constraints equations can impose a wide variety of connections as well as analytically rigid elements. An example multibody system can be seen in Fig. 1.1.

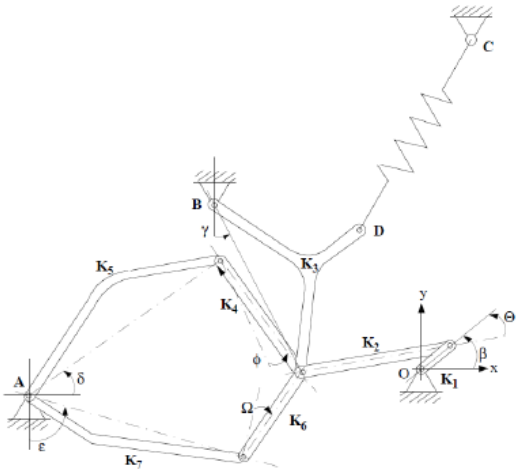


Figure 1.1: Andrew's 7-bar mechanism.[1].

The concept of a rigid body, although somewhat imaginary (there is no rigid body), can be very useful if the deformations are small enough so as to not effect the gross motion of the body. Modelling a body as rigid as compared to flexible can lead to considerable computational savings. On the other hand, some situations warrant the modelling of certain bodies as flexible due to significant deformations or the desire to analyse stresses within the body. Thus a technique which has the ability to simulate both rigid and flexible bodies at the same time provides users with means to vary the degree of fidelity of a model and also leads to computationally efficient modelling of joints which are assumed to be rigid, as opposed to modelling them as flexible which requires modelling of contact. Thus for a preliminary design where computational time is a premium, a user may choose to model most of the members of the system as rigid. But in the detailed design stages, more and more members can be modelled as flexible in order to accurately capture the physics of the problem. Popular multibody software like DADS and ADAMS feature efficient simulation of systems composed of rigid bodies only. These software handle the case of flexible bodies by interfacing with finite element software to calculate the mode shapes of deformation. This current research however attempts to develop a methodology that allows the simulation of multibody systems with rigid and flexible bodies simultaneously.

Most research on flexible body dynamics is still growing and developing. Even in rigid body dynamics, there are some details that need to be addressed such as Index reduction problem, stability problem, constraint satisfaction, and time order accuracy. Here are some of the sample problems stated by the author of this research that could misinform or mislead the general reader to improper conclusions:

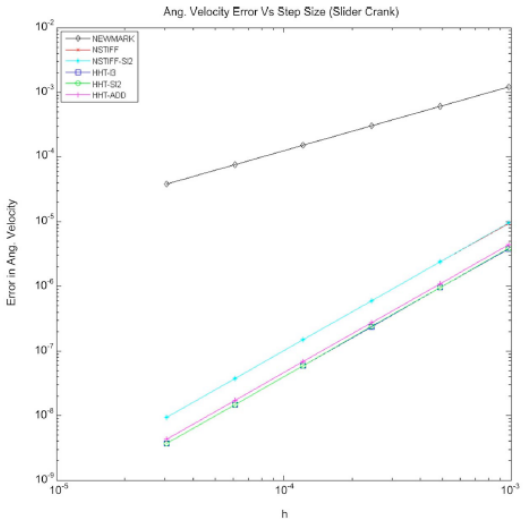


Figure 1.2: Convergence rate study of Newmark and HHT-α algorithms from [54].

Dan Negrut in [54] developed a dissipative scheme for the algorithm (HHT-α) and applied on multibody dynamic systems. He compared the solution with Newmark and order 2 BDF algorithm in Index 3 system. Then turned attention to using these schemes to solve over-determined system (solve Index 3 and 2 simultaneously). The focus of this paper is on: 1) Order of accuracy, 2) Energy conservation, 3) Constraint satisfaction, and 4) Computational cost. However Newmark, as shown in Fig. 1.2, is forced to be first order accuracy ($\gamma \neq 0.5$) due to instability issue. In fact, as demonstrated in the research, one can still preserve the same time accuracy (second order time accuracy) but not solving an over-determined system, and still satisfy all the constraint and preserve energy in good manner.

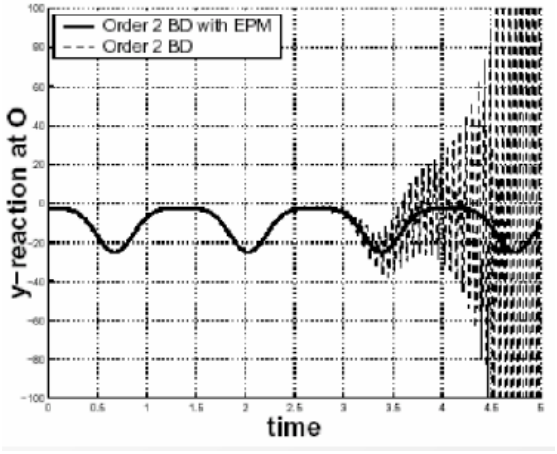


Figure 1.3: Simple Pendulum: Constraint Force Oscillations from [47].

Borri [47] demonstrates the failure for the simple pendulum by showing that the numerical solution to the Index 3 problem (without dissipation), though able to produce smooth and stable displacements and velocities, will develop oscillations in the accelerations (and therefore Lagrange multipliers) as seen in Fig. 1.3.

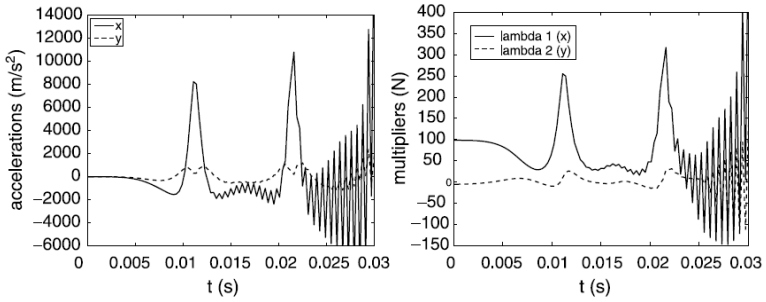


Figure 1.4: Andrew's mechanism: Acceleration and constraint force oscillations from [64].

In [64], Arnold shows a numerical example for Andrew's 7 bar mechanism as shown in Fig. 1.1. Also, this research uses numerical damping (dissipative algorithm: generalized-α method) to avoid numerical oscillation and however it takes the total energy artificial and eventually it will

removes the total energy slowly.

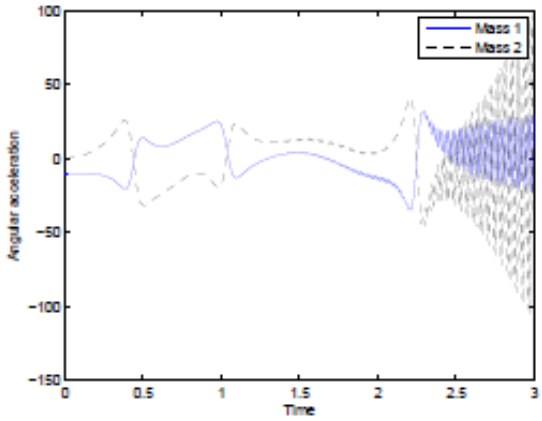
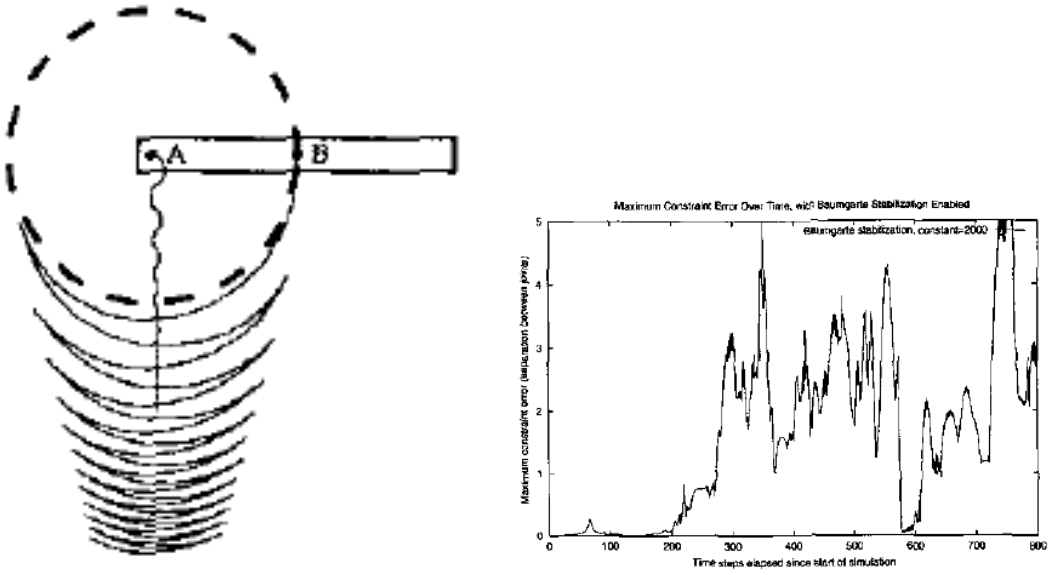


Figure 1.5: Newmark Method: Double Pendulum Acceleration blow-up from [12].

Another example, this time of a double pendulum, is discussed at length by Geradin in [12]. Using the (non-dissipative) Newmark method it is again shown that the acceleration and Lagrange multipliers exhibit unbounded oscillatory behaviour, see Fig. 1.5. These unbounded oscillations eventually lead to convergence problems in the nonlinear iterations and the solution is unable to proceed, In attempt to remedy these unstable oscillations it is later shown that using HHT-α method removes the unwanted (and non-physical) oscillations, but at the price of dissipating energy from what should be a conservative system.



(a) Drift effect of single pendulum

(b) Results of constraint satisfaction

Figure 1.6: Drift effect and imperfect constraint satisfaction of Baumgarte's stabilization from [53].

In [53], Cline and Pai show a numerical example of 6-link mechanism with Baumgarte's stabilization technique. In this paper they develop other methods to overcome drift problem. It is worth to note that many research papers still use Index 2 based Baumgarte's method to satisfy constraint; however, this result shows it is not recommended to use this technique. Also, this research will show several the imperfections of Baumgarte's stabilization.

These results from Fig. 1.2 to Fig. 1.6 shows there are several issues that need to improve in this area. The focus of this research is to investigate a new GSSSS second order framework and methodology and show the proper manner to overcome the current difficulties, and provide a design space for future research.

1.3 Literature Overview: Time Discretization

Of primary interest and the focus in this section are the numerical aspects of time integration of the equations of motion for holonomic-scleronomic system that are routinely encountered in a wide class of problems in computational dynamics of particles, meterials, and continuum structures such as elastodynamics. There exist various frameworks for developing computational formulations, namely, vector formalisms and scalar formalisms. The limitations for establishing equivalences of the strong and weak forms amongst the various frameworks (Newtonian, Lagrangian, Hamiltonian) are that the kinetic energy is quadratic in generalized velocities and the potential energy is not velocity dependent for N-body systems. The Hamiltonian framework yields a system of first order in time semi-discretized system of second order in time differential equations. Historically, most time integration methods have been derived in the sense of the single field form of representations for the equations of motion. In general, the design of computational algorithms for the integration of the equations of motion either pertain to a two-field (first order) form or a single field (second order) form of representation for the process of time integration. Consequently, the former representation does not permit a spurious root (numerically non-dissipative algorithm design). However, unless accelerations are not treated as a primary variables, for which case a spurious root does not exist, the latter naturally involves a spurious root that is associated with the underlying algorithmic amplification matrix and provides the ability to develop controllable algorithmic dissipation. This dissipation can be employed to control the amount of participation of the high frequency modes resulting from the numerical space discretization. In this section, we particularly focus attention to the class of LMS methods involving a single system of equations and a single solve within each time step as these are the most practical, economical, and commonly employed in commercial software settings. In particular we focus attention on the single field form of representations which give

rise to a 3-root system as the displacements, velocity, and accelerations are treated as the primary variables.

Traditionally, most of the classical time integration methods including linear multistep methods have been motivated from Taylor's series expansions for the approximation of the field variables such as displacement, velocity, and acceleration appearing in the space discretized equations of motion. Even though several of these methods have been developed to a certain degree of maturity and from various differing viewpoints, the geometric nature that the ordinary or partial differential systems possess has been ignored to an extent in constructing time integration algorithms in certain cases. Consequently, the time integration for highly nonlinear differential system may become unstable and blows up for certain time steps.

To address some of these concerns, geometric mechanics stemming from Lagrangian mechanics which has been pioneered by Lie (1842-1899) appears to play in important role. In particular, to ensure that certain physical attributes are conserved/preserved in the algorithm design. It has been believed that the preservation of geometric nature of the differential system can give rise to long-term simulation, especially for nonlinear differential systems. Furthermore, preserving the structure appears to provide more physically accurate time descretized algorithms [2] that are useful for a certain class of applications. Nonetheless, the goal of designing algorithms with a single system and single solve within each time step further enhances the efficiency of computations from a practical viewpoint. Hence, the focus of this research was restricted to such practical considerations.

Understanding the symmetry in physics is very important in developing time integration schemes that inherit certain desirable algorithmic attributes useful for certain classes of problems. Noether

[3] first discovered that the Lagrangian has symmetry. Conservation of quantities such as energy, linear and angular momentum results from the symmetry of the Lagrangian or the Hamiltonian. Hence, structure preserving algorithms have attracted some degree of attention, for example, the energy-momentum conserving and symplectic algorithms. Algorithms which are able to conserve fundamental physical quantities such as energy, linear momentum and angular momentum have been the topic increasingly many research papers. Beyond the basic desire to exactly conserve quantities which we know from physics should be conserved, implicit EMM algorithms have the unique ability to claim unconditional stability of energy even for nonlinear problems. That is, regardless of the size of the time step taken the algorithm will maintain an exact total system energy. This is contrary to symplectic algorithms which often suffer from jumps in energy for time step sizes are too large. However, EMM leads to unsymmetric matrices.

1.4 Literature Overview: Multibody Dynamics

In the recent years, considerable effort has been devoted to modelling, design, and control of mutibody dynamics (MBD). There are a number of publications in this field and has been steadily increasing. Lists and reviews of many contributions on formulation of MBD are given in survey paper [4]-[6] by Shabana, Bremer, and Schiehlen, and on the general area multibody system [7] by Huston. The formulation of MBD system through Lagragian equation were published [8] by Simeon, where through Hamilton system were introduced in [9]-[10] by Betsch and Steinmann. Several papers and books have been published in discussion of constraint formulation [11]-[12] by Geradin, and Shabana. Some special techniques for stabilization in DAEs were illustrated [21], [18], and [24] by Baumgarte, Petzold, and Arnold. Several commercial software were developed such as ADAMS from Mechanical dynamics Inc, DADS from CADSI Inc, RecurDyn by Function Bay Inc, and MECANO from Samtech.

In multibody dynamics, an inertial frame serves as a global reference frame for describing the motion of the multibody system. In addition, intermediate reference frames that are attached to each flexible body following the average local rigid body motion (rotation and translation) are widely used. The motion of the body relative to the intermediate frame is, approximately, due only to the deformation of the body. This simplifies the calculation of the internal forces since stress and strain measures that are not invariant under rigid body motion can be used to calculate these forces with respect to the intermediate frame. Two main types of intermediate frames are used nowadays: floating and corotational frames. In this research, we focus upon the former approach since it is the most popular approach among many MBD simulation codes. In many papers, intermediate frames are not used; instead the global inertial frame is directly used for measuring deformation. In this approach, the motion of an element consists of a combination of rigid body motion and deformation and the two types of motion are not separated. Nonlinear finite strain measures and corresponding energy conjugate stress measures, which are objective and invariant under rigid body motion, are used to calculate the internal forces with respect to the global inertial frame. The comparison of floating reference approach and inertial reference approach is worth noting for engineers and researchers.

The floating frame approach originated out of research on rigid multibody dynamics in the late 1960s. It was used for extending rigid multibody dynamics codes to MBD. This was done by superimposing small elastic deformations on the large rigid body motion obtained using the rigid multibody dynamcis code. Initial applications of the floating frame approach includes: spinning flexible beams (for space structures application), kineto-elastodynamics of mechanisms, and flexible manipulators [13]. It was also used to extend modal analysis and experimental modal identification techniques [14] by Likins. The first n modes (where n is determined by the physics of the problem and by the required accuracy) are superposed on the rigid body motion of the component represented by the motion of the floating frame. On the other hand, inertial

frame approach has its origin in the nonlinear finite element method and continuum mechanics principles. These techniques were applied to the dynamic analysis of continuum bodies undergoing large rotations and large deformations (including both large strains and large deflections) since the early 1970s [15]-[16]. It is worth mentioning that one of the methods in the class of inertial reference approach is the the absolute nodal coordinate formulation (ANCF) by Shabana in 1996 [17]. Traditionally, the finite elements in the structural systems can be categorized into two main groups. First is isoparametric type elements such as planar triangular and rectangular elements. Since these isoparametric elements do not use rotational parameters as nodal coordinates, they are not in general recommended for use in beam, plate and shell applications. Furthermore, the lack of rotation degrees of freedom can be a source of problems when joint constraints are defined in MBD applications. The second group consists of elements in which displacements as well as infinitesimal rotations are used as nodal coordinates like beam elements, plate, and shell elements. It can be demonstrated, however, that the use of infinitesimal rotations as nodal coordinates leads to a linearization of the equation of motion of the rigid body. As a result, classical finite element formulations do not describe an exact rigid body displacement. Because of this fact, beams and plates are not considered in the finite element literature as isoparamtric elements, since an arbitrary rigid body motion of the element described in terms of infinitesimal nodal rotation does not result in zero strains. This problem can be circumvented if the nodal coordinates are expressed in terms of absolute nodal displacements and slopes. Using this new set of nodal coordinates, an absolute nodal coordinate formulation can be developed for the large deformation and rotation analysis of flexible structures that undergo an arbitrary reference displacement. Using this method, beams and plates can be treated as isoparametric elements and arbitrary rigid body motion of these elements produces zero strain. Most researchers in this field based on this method express more possibilities can be offered in the simulation of multibody dynamics. However, ANCF results in constant mass matrix and a highly nonlinear stiffness matrix which introduces numerical difficulties.

Through the above approach, one can construct the constrained multibody dynamics for both equation of motion and constraint equations. these formulation are mathematically equivalent to each other. Numerically, nonlinearity in equation of motion and constraint equations are different for the two approaches. Many difficulties for solving a DAE system may occur. Petzold and Ascher outline a number of difficulties which can arise to solve system of differential-algebraic equations in 1982 [18]. Also, they define the so-called "Index" for a differential algebraic system and pointed out that a direct discretization of DAEs with geometry constraint (Index 3) yields numerical difficulties. This is what gives rise to a multitude of other, more specific solution techniques. Typically, such a solution technique consists of steps of problem reformulation which involves reducing its Index, followed by a discretization of the resulting formulation (integrating the equation of motion with velocity or acceleration constraint). In recent works [19], [20] it has been shown that for a certain model problem, some of these formulations can be equivalent. An important consideration in selecting an appropriate solution method is the stability of the method and the subsequent stability restrictions that a chosen step size must satisfy. In this research, an investigation of the stabilization technique will be revealed and comparison of each stabilization techniques will also be shown.

Index reduction techniques are a formal mathematical procedure that reduce the Index of DAEs from 3 to 2 or 1 to obtain stability. In many cases, properties of the proposed schemes are proved mathematically. While the order of accuracy of the solution is often provided, the problem of violation of the constraints was rarely addressed. Due to the approximation and round-off errors, many numerical solutions do not satisfy constraints exactly, a phenomenon known as drift. Numerous constraint violation stabilization techniques have been developed to remedy this problem, and Baumgarte's method [21] is probably the most widely used. It presents

two shortcomings: first, constraints are never exactly satisfied, and second, no general principle exists to determine the problem dependent parameters appearing in the formulation. Consequently, this approach cannot be recommended for general used in multibody systems. Penalty based formulations have also been used to control the drift phenomenon. The augmented Lagrangian formulation is probably the most robust and efficient method [22]-[23]. The constraint violation elimination techniques have been developed to enforce satisfaction of the constraints to machine accuracy, typically by means of iterative process that projects the solution onto the constraint manifold.

Recently, new algorithms have been developed for the enforcement of constraints within the framework of the finite element method. Based on the physical concepts of energy preservation and vanishing of the work done by constraint forces, robust algorithms have been developed that present mathematical proofs of stability. Finally, scaling algorithms have been developed that appear to eliminate the ill conditioning traditionally associated with Index-3 DAEs.

1.5 Research Goals

In recent years there has been a variety of authors publishing numerous papers mimicking the applications of well known structural dynamics algorithms to multibody dynamic simulations [54],[12], and[56]; some of which attempt to leave the DAE in its natural Index 3 form as well as include the capability of flexible elements. On the other hand, research in [55], [24], and [57] shows that from numerical stability point of view, DAEs from multibody dynamic simulation are mostly done within lower Index. Also, another point of view from [41] and [11] tell us that different reference frame approaches (inertial reference frame with/without intermediate frame) results in different numerical difficulties. This research does not express which combination is better than the other, but shows the big picture of the design space for those who enter this

field and to simulate the MBD application in a proper manner. Such is the goal of the current research: to utilize the GSSSS family of algorithms to identify a method which can provide a stable solution and satisfy physical constraints (Index 3/2/1). Also, to help researchers answer the following questions:

1) What are the numerical advantages and disadvantages for using FRF or IRF or others such as ANCF?

2) Which Index is preferable? Under which condition?

3) How to simulate the DAEs from MBD problem properly (satisfaction of constraints, second order time accuracy, and numerical stability)?

Chapter 2

Structural Dynamics

The purpose of this chapter is to provide a basic background of the extension of the GSSSS family of algorithms to nonlinear system of ordinary differential equations(ODEs). This background is required to understand further extensions to the solution of differential-algebraic equations (DAEs), the main objective of this research. First, an overview of GSSSS algorithms is outlined; then the usage of GSSSS, and finally the importance of consistency of equation of motion time level in described.

2.1 The Framework of GSSSS Family of Algorithms Encompassing LMS methods

The importance of GSSSS is its unique concept that provides new avenues to design time integration operators for dynamic problems through a unified framework, which contains those existing algorithms known so far and including new designs of algorithms encompassing the class of LMS methods. This concept was originally introduced by Zhou et al [25]-[29], under

algorithms by design.

By choosing specific algorithmic parameters related to the algorithmic DNA markers, the corresponding generalized single solve single step (GSSSS) time integration algorithms were uniquely design under its framework. Two principal roots and the spurious root act as an essential rule in generating algorithms. Once three roots are properly selected, the underlying design enables one to not only produce existing algorithms in the sense of LMS methods (Newmark, Midpoint rule with endpoint acceleration and midpoint acceleration, etc.)[30][31], but also provide new avenues toward new algorithmic designs in the sense of the least amount of algorithmic dissipation, dispersion, and overshoot behaviour. In general, for the single field form of representation in the sense of LMS methods, there are three roots participating in the algorithm, namely, the two principal roots ρ_∞^{min}, ρ_∞^{max} and the spurious root ρ_∞^s. It has been shown in [27] and [29] that the second-order time accurate, unconditionally stable LMS framework contains two distinct algorithmic structures: constrained U (displacement overshooting aspect, or U0) and constrained V (velocity overshooting aspect, or V0) family of algorithms for linear structural dynamic systems. Most numerically non-dissipative and dissipative algorithms pertain to the constrained U-family of algorithms within the class of LMS methods for finite element computations (non-dissipative schemes such as Newmark method [31], midpoint rule, and controllable dissipative algorithms such as WBZ [32] and HHT-α [33] etc.).

This means that the relatively unknown V0 family has, for the most part, not been the focus of significant research and almost certainly has not been considered in the case of DAEs.

2.1.1 The U0 and V0 Family

The U0 family of algorithms contain most of the algorithms with zero-order displacement overshooting behaviour, and the V0 family of algorithms contains most of the algorithms with zeros-order velocity overshooting behaviour.

Consider the semi-discretized system of equations of linear structural dynamic problems by space discretization of the single form as :

$$\begin{aligned} &\mathbf{M}a(t)) + \mathbf{C}v(t) + \mathbf{K}u(t) = \mathbf{F}(t) \\ &u(0) = u_0 \quad , \quad \dot{u} = v_0 \end{aligned} \tag{2.1}$$

where **M** is the mass matrix, **C** is the damping matrix, and **K** is the stiffness matrix. The U0 and V0 family of algorithms are given as follows.

U0/V0 Family of Algorithms

Given u_n, v_n, and a_n, find u_{n+1}, v_{n+1}, and a_{n+1} from

$$\begin{aligned} (\Lambda_6 W_1 \mathbf{M} + \Lambda_5 W_2 \Delta t \mathbf{C} + \Lambda_4 W_4 \Delta t^2 \mathbf{K})\Delta a = &- \mathbf{M}a_n - \mathbf{C}(v_n + \Lambda_4 W_1 \Delta t a_n) \\ &- \mathbf{K}(u_n + \Lambda_1 W_1 \Delta t v_n + \Lambda_2 W_2 \Delta t^2 a_n) \\ &+ (1 - W_1)\mathbf{F}_n + W_1 \mathbf{F}_{n+1} \end{aligned} \tag{2.2}$$

followed by updating the variables at the end of each time level as follows

$$
\begin{aligned}
u_{n+1} &= u_n + \lambda_1 v_n \Delta t + \lambda_2 v \Delta t^2 + \lambda_3 \Delta a \Delta t^2 \\
v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 \Delta a \Delta t \\
a_{n+1} &= a_n + \Delta a
\end{aligned}
\tag{2.3}
$$

where for U0 family we have

$$
\begin{aligned}
\Lambda_1 W_1 &= \frac{1}{1+\rho_\infty^s}, \quad \lambda_1 = 1 \\
\Lambda_2 W_2 &= \frac{1}{2(1+\rho_\infty^s)}, \quad \lambda_2 = 1/2 \\
\Lambda_3 W_3 &= \frac{1}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^s)}, \quad \lambda_3 = \frac{1}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})} \\
\Lambda_4 W_1 &= \frac{1}{1+\rho_\infty^s}, \quad \lambda_4 = 1 \\
\Lambda_5 W_2 &= \frac{3+\rho_\infty^{min}+\rho_\infty^{max}-\rho_\infty^{min}\rho_\infty^{max}}{2(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^s)}, \quad \lambda_5 = \frac{3+\rho_\infty^{min}+\rho_\infty^{max}-\rho_\infty^{min}\rho_\infty^{max}}{2(1+\rho_\infty^{min})(1+\rho_\infty^{max})} \\
\Lambda_6 W_1 &= \frac{2+\rho_\infty^{min}+\rho_\infty^{max}+\rho_\infty^s-\rho_\infty^{min}\rho_\infty^{max}\rho_\infty^s}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^s)} \\
W_1 &= \frac{1}{1+\rho_\infty^s}
\end{aligned}
\tag{2.4}
$$

and for V0 family we have

$$
\begin{aligned}
\Lambda_1 W_1 &= \frac{3+\rho_\infty^{min}+\rho_\infty^{max}-\rho_\infty^{min}\rho_\infty^{max}}{2(1+\rho_\infty^{min})(1+\rho_\infty^{max})}, \quad \lambda_1 = 1 \\
\Lambda_2 W_2 &= \frac{1}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})}, \quad \lambda_2 = 1/2 \\
\Lambda_3 W_3 &= \frac{1}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^{s})}, \quad \lambda_3 = \frac{1}{2(1+\rho_\infty^{s})} \\
\Lambda_4 W_1 &= \frac{3+\rho_\infty^{min}+\rho_\infty^{max}-\rho_\infty^{min}\rho_\infty^{max}}{2(1+\rho_\infty^{min})(1+\rho_\infty^{max})}, \quad \lambda_4 = 1 \\
\Lambda_5 W_2 &= \frac{2}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^{s})}, \quad \lambda_5 = \frac{1}{1+\rho_\infty^{s}} \\
\Lambda_6 W_1 &= \frac{2+\rho_\infty^{min}+\rho_\infty^{max}+\rho_\infty^{s}-\rho_\infty^{min}\rho_\infty^{max}\rho_\infty^{s}}{(1+\rho_\infty^{min})(1+\rho_\infty^{max})(1+\rho_\infty^{s})} \\
W_1 &= \frac{3+\rho_\infty^{min}+\rho_\infty^{max}-\rho_\infty^{min}\rho_\infty^{max}}{2(1+\rho_\infty^{min})(1+\rho_\infty^{max})}
\end{aligned}
\tag{2.5}
$$

ρ_∞^{min}, ρ_∞^{max}, and ρ_∞^{s} are the first principal root, the second principal root, and the spurious root at the high-frequency limit, respectively, and they satisfy the following relation:

$$
0 \le \rho_\infty^{s} \le \rho_\infty^{min} \le \rho_\infty^{max} \le 1 \tag{2.6}
$$

2.1.2 Usage of the GSSSS Family of Algorithms

The research here is based on the above two families of linear dynamic algorithms. Under the framework of the GSSSS family of algorithms, all of the currently existing and new algorithms that are second order time accurate encompassing LMS methods can be constructed by first selecting the family (either the so-called U0 and V0 family), and then the three parameters ρ_∞^{min}, ρ_∞^{max}, and ρ_∞^s specify the particular algorithm that is selected. Simply for illustration, Table 2.1 expresses some of the well-known algorithms under the GSSSS framework.

Table 2.1: Commonly known algorithms

$(\rho_\infty^{min}, \rho_\infty^{max}, \rho_\infty^s)$	Common Name	Conditions
U0(1,1,0)	Newmark	-
U0/V0(1,1,1)	Classical Midpoint Rule [MPR-EPA]	-
V0(1,1,0)	New Version Midpoint Rule [MPR-MPA]	-
U0(ρ_∞^{min},ρ_∞^{max},ρ_∞^s)	Generalized-α	$\rho_\infty^{min} = \rho_\infty^{max} = \rho_\infty^s = \rho_\infty$
U0(ρ_∞^{min},ρ_∞^{max},0)	WBZ	$\rho_\infty^{min} = \rho_\infty^{max} = \rho_\infty$
U0(ρ_∞^{min},ρ_∞^{max},ρ_∞^s)	HHT-α	$\rho_\infty^{min} = \rho_\infty^{max}$, $\rho_\infty = \frac{1-\rho_\infty^{min}\rho_\infty^{max}}{\rho_\infty^{min}+\rho_\infty^{max}+2\rho_\infty^{min}\rho_\infty^{max}}$
U0/V0(ρ_∞^{min},1,ρ_∞^s)	U0-V0 Optimal	$\rho_\infty^{min} = \rho_\infty^s$

2.1.3 Equation of Motion Time Level

The concept of time level at which the equation motion needs to be computed is extremely fundamental and important, especially in the case of extension of linear to nolinear dynamic problems. Incorrect computation of time level in the equation of motion will lead to decrease in time accuracy of the algorithm; also including failure to preserve certain quantities in algorithm design for energy and momentum conservation. Fundamentally, computational structural dynamics problems comprise of the strong form of the partial differential equations of motion. The semi-discrete weak form of the equation of motion is obtained after spatial discretization, which yields a system of ordinary differential equations that can be solved computationally. The ordinary differential equations from the semi-discretized problem can then be discretized in time, which leads to various forms. In the linear dynamics cases, this fully discretized equation of motion is written as:

$$\mathbf{M}\tilde{a}(t) + \mathbf{C}\tilde{v}(t) + \mathbf{K}\tilde{u}(t) = \tilde{\boldsymbol{F}}(t) \tag{2.7}$$

This equation must be valid at any time t. From the set of given initial conditions $u(t_0)$, $v(t_0)$, and matrices $\mathbf{M}$, $\mathbf{C}$, $\mathbf{K}$, and vector $\mathbf{F}$ known from the spatially discretized mesh, the solution is then straightforward. Given the initial values, one solves the equation of motion at some certain time point: $\tilde{t}$, where $t_n \leq \tilde{t} \leq t_{n+W_1}$. This is the time point at which the equation of motion must be satisfied precisely.

It was shown specifically that the GSSSS family of algorithms satisfies the equation of motion at the time level t_{n+W_1} for time dependent variables v, u, and F. The value of a seems to be inconsistently calculated not at time t_{n+W_1}, but instead at time $t_{n+\Lambda_6 W_1}$. In [34] this seemingly inconsistent time level evaluation was explained, and it was also explained that the acceleration

is in fact also calculated at time t_{n+W_1}. This special time level must be satisfied for *any* algorithm that is designed from the second order accurate framework of the GSSSS family of algorithms. This concept was essential in this research; particularly for the extensions to nonlinear problems and differential-algebraic equations.

2.1.4 Extensions to Nonlinear Equations

Both U0 and V0 families of algorithms satisfy the equation of motion at a consistent time level t_{n+W_1}. The importance of this time level for proper extension to nonlinear equations is required. For the linear dynamic problems, if the proper time level is not satisfied, it will cause order reduction in time accuracy from second order accuracy to first order accuracy. Nonlinear dynamic applications require further attention be paid to this time level. The algorithmic designs and developments in linear dynamic cases help us to understand the nonlinear dynamics applications carefully. The literature has not been very careful or consistent in drawing the subtle, but important distinctions in enacting these extensions. With consideration, for nonlinear dynamic applications, the details of this were carefully expressed and analyzed in [34]-[36]. The energy-momentum based algorithms are the focus of this research due to their reputation for providing stable solutions for multibody dynamical simulation. Recall that for the general case of nonlinear dynamic problems, the equation of motion takes the following form.

$$\mathbf{M}(\tilde{q})\tilde{a}(t) + \mathbf{C}(\tilde{q})\tilde{v}(t) + \mathbf{K}(\tilde{q})\tilde{u}(t) = \tilde{\boldsymbol{F}}(t) \tag{2.8}$$

The updated parameter $\tilde{q}$ can be a function of time dependent variable: displacement $\tilde{u}$, velocity $\tilde{v}$, and acceleration $\tilde{a}$. To be consistent with displacements, velocities, and accelerations (shown to be at time level t_{n+W_1}), all updated variables need to be also approximated at the same time level t_{n+W_1}. In this way, the order reduction in time accuracy can be avoided. Literature has shown that incorrect time level in nonlinear truss element simulation. This will be shown in

chapter 3 for the implementation of proper time level issue in formulating constrained equations. Improper time level not only loses accuracy, but also causes numerical failure for come cases.

2.1.5 Sympletic-Momentum Methodology

The sympletic-momentum methodology, described in detail in [37], advances the parent linear dynamic GSSSS family of algorithms via that termed as the displacement based normalized time weight residual approach to nonlinear dynamic applications such that : (i) for case of no physical damping ($\mathbf{C} = \mathbf{0}$) and for the case of numerically non-dissipative algorithms, they enable a class of algorithms and designs (in the sense of the single-field form) that are sympletic-momentum conserving, and (ii) one can embed controllable numerical dissipation as well, and when the controllable numerical dissipation is turned off, they readily recover the original sympletic-momentum family of conserving schemes for unconstrained dynamic systems. From computational and implementation aspects, employing the normalized time weighted residual methodology to the constant terms in the equation of motion described above, one can readily derived the a-, v- and d-form representations as highlighted next. We employ the Newton-Raphson method to solve the nonlinear structural dynamic equation iteratively within each time step. Assume the internal force can be expressed as:

$$\mathbf{F}_{int} = \mathbf{K}(u)\mathbf{u} \tag{2.9}$$

Construct the effective structural dynamic equation employing the displacement based normalized time weighted residual approach given by

$$\mathbf{M}\tilde{a} + \mathbf{C}\tilde{v} + \tilde{\boldsymbol{F}}_{int} = \tilde{\boldsymbol{F}}_{ext} \tag{2.10}$$

where

$$\tilde{\boldsymbol{F}}_{int} = F_{int}(\tilde{u}) \tag{2.11}$$

Employ the Newton-Raphson method to iteratively solve for the nonlinear effective structural dynamic equation above: At the beginning of time step, predict the state vectors

$$\begin{aligned}
u_{n+1}^{k} &= \chi_{Pu}^{(1)} u_n + \chi_{Pu}^{(2)} v_n + \chi_{Pu}^{(3)} a_n \\
\tilde{u}_{n+1}^{k} &= \chi_{P\tilde{u}}^{(1)} u_n + \chi_{P\tilde{u}}^{(2)} v_n + \chi_{P\tilde{u}}^{(3)} a_n \\
\tilde{v}_{n+1}^{k} &= \chi_{P\tilde{v}}^{(1)} u_n + \chi_{P\tilde{v}}^{(2)} v_n + \chi_{P\tilde{v}}^{(3)} a_n \\
\tilde{a}_{n+1}^{k} &= \chi_{P\tilde{a}}^{(1)} u_n + \chi_{P\tilde{a}}^{(2)} v_n + \chi_{P\tilde{a}}^{(3)} a_n
\end{aligned} \tag{2.12}$$

Start nonlinear iteration. Solving for $\Delta\delta_{n+1}^{k+1}$ from

$$\begin{aligned}
&\left[\chi_{Ca}\mathbf{M} + \chi_{Cv}\mathbf{C} + \chi_{Cd}\mathbf{K}^{k}\right]\Delta\delta_{n+1}^{k+1} = \\
&-\left(\mathbf{M}\tilde{a}_{n+1}^{k} + \mathbf{C}\tilde{v}_{n+1}^{k} + \mathbf{K}(\tilde{u}_{n+1}^{k})\tilde{u}_{n+1}^{k} - \tilde{\boldsymbol{F}}_{n+1}^{k}\right)
\end{aligned} \tag{2.13}$$

Then correct the primary variables as follows:

$$\begin{aligned}
u_{n+1}^{k+1} &= \chi_{Pu}^{(1)} u_n + \chi_{Cu}\Delta\delta_{n+1}^{k+1} \\
\tilde{u}_{n+1}^{k+1} &= \chi_{P\tilde{u}}^{(1)} u_n + \chi_{C\tilde{u}}\Delta\delta_{n+1}^{k+1} \\
\tilde{v}_{n+1}^{k+1} &= \chi_{P\tilde{v}}^{(1)} u_n + \chi_{C\tilde{v}}\Delta\delta_{n+1}^{k+1} \\
\tilde{a}_{n+1}^{k+1} &= \chi_{P\tilde{a}}^{(1)} u_n + \chi_{C\tilde{a}}\Delta\delta_{n+1}^{k+1}
\end{aligned} \tag{2.14}$$

Once the solution converges, update the primary variables at the end of the time step as follows.

$$
\begin{aligned}
a_{n+1} &= a_n + (\tilde{a}_{n+1}^{k+1} - a_n)/\Lambda_6 W_1 \\
v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 (a_{n+1} - a_n)\Delta t \\
u_{n+1} &= u_n + \lambda_1 v \Delta t + \lambda_2 a_n \Delta t + \lambda_3 (a_{n+1} - a_n)\Delta t^2
\end{aligned}
\tag{2.15}
$$

The predictor-corrector coefficients χ above in the corresponding a, v, and d-form are listed in Table 2.2.

2.1.6 Energy-Momentum Methodology

The energy-momentum methodology, described in detail in [38], involves a hybrid procedure where the strain is approximated independently from the displacement. For the U0 family of algorithms it defines the strain to be a linear approximation between t_n and t_{n+1}. The approximations for $\ddot{u}$ and u comes directly from the parent linear dynamic algorithms. The importance of understanding the EOM time level is evident from the strain term. Without knowledge of the EOM time level, simply stating that we wish to approximate strain by a trapezoidal rule representation between t_n and t_{n+1} provides no basis for *where* to choose the time point between t_n and t_{n+1}. Considering the point collocation at time level t_{n+W_1}, we now have a justification for the time point at which to approximate the strain. It has been shown extensively in [36] that correctly evaluating the strain at time level t_{n+W_1} results in algorithms which are second order accurate in all variables, whereas any shift from this correct time level destroys the accuracy of the algorithms as well as the energy conserving properties.

The key outcome of the previous research via energy and momentum algorithms is the absolute necessity of time level collocation of any additional terms. Here, that was manifest in the development of a method where the strain was independently approximated from the other

primary variables. Such an approximation has no basis without this time level concept and any algorithm which deviates from this time level has been shown to suffer from loss of accuracy and loss of conservation properties. In the following chapter, when this method is extended to include constrained dynamics, this concept is essential. The generic algorithm for the energy-momentum based method for a special case of a truss element example is shown below. Notice that the strain term, a displacement level in time term, is approximated in exactly the same way as the unknown displacement: at time t_{n+W_1}.

At the beginning of time step, predict the state vectors

$$\begin{aligned} u^k_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi^{(2)}_{Pu} v_n + \chi^{(3)}_{Pu} a_n \\ \tilde{u}^k_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi^{(2)}_{P\tilde{u}} v_n + \chi^{(3)}_{P\tilde{u}} a_n \\ \tilde{v}^k_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi^{(2)}_{P\tilde{v}} v_n + \chi^{(3)}_{P\tilde{v}} a_n \\ \tilde{a}^k_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi^{(2)}_{P\tilde{a}} v_n + \chi^{(3)}_{P\tilde{a}} a_n \end{aligned} \tag{2.16}$$

Start nonlinear iteration. Solving for $\Delta\delta^{k+1}_{n+1}$ from

$$\begin{aligned} &\left[\chi_{Ca}\mathbf{M} + \chi_{Cv}\mathbf{C} + \chi_{Cd}\mathbf{K}^k\right]\Delta\delta^{k+1}_{n+1} = \\ &-(\mathbf{M}\tilde{a}^k_{n+1} + \mathbf{C}\tilde{v}^k_{n+1} + \mathbf{K}_1\tilde{u}^k_{n+1} + \mathbf{K}_2\tilde{\epsilon}_n + 1 + \mathbf{K}_3\tilde{u}^k_{n+1}\epsilon_{n+1} - \tilde{\boldsymbol{F}}^k_{n+1}) \\ &\mathbf{K}^k_{n+1} = \mathbf{K}_1 + \mathbf{K}_2(b + \frac{4}{l_0^2}Au^k_{n+1}) + \mathbf{K}_3\left[\tilde{u}^k_{n+1}(b + \frac{4}{l_0^2}Au^k_{n+1}) + \tilde{\epsilon}_{n+1}\right] \end{aligned} \tag{2.17}$$

where the element based EOM time level strain $\tilde{\epsilon}_{n+1}$ is evaluated as :

$$\tilde{\epsilon} = \chi^{(1)}_{P\tilde{u}}\epsilon_n + \chi^{(2)}_{P\tilde{u}}\dot{\epsilon}_n + \chi^{(3)}_{P\tilde{u}}\ddot{\epsilon}_n\chi_{C\tilde{u}}\left(\epsilon(u^k_{n+1}) - \epsilon(u_n)\right) \tag{2.18}$$

Then correct the primary variables as follows:

$$
\begin{aligned}
u_{n+1}^{k+1} &= \chi_{Pu}^{(1)} u_n + \chi_{Cu} \Delta\delta_{n+1}^{k+1} \\
\tilde{u}_{n+1}^{k+1} &= \chi_{P\tilde{u}}^{(1)} u_n + \chi_{C\tilde{u}} \Delta\delta_{n+1}^{k+1} \\
\tilde{v}_{n+1}^{k+1} &= \chi_{P\tilde{v}}^{(1)} u_n + \chi_{C\tilde{v}} \Delta\delta_{n+1}^{k+1} \\
\tilde{a}_{n+1}^{k+1} &= \chi_{P\tilde{a}}^{(1)} u_n + \chi_{C\tilde{a}} \Delta\delta_{n+1}^{k+1}
\end{aligned}
\tag{2.19}
$$

Once the solution converges, update the primary variables at the end of the time step as follows.

$$
\begin{aligned}
a_{n+1} &= a_n + (\tilde{a}_{n+1}^{k+1} - a_n)/\Lambda_6 W_1 \\
v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 (a_{n+1} - a_n) \Delta t \\
u_{n+1} &= u_n + \lambda_1 v \Delta t + \lambda_2 a_n \Delta t + \lambda_3 (a_{n+1} - a_n) \Delta t^2
\end{aligned}
\tag{2.20}
$$

The predictor-corrector coefficients χ above in the corresponding a, v, and d-form are listed in Table 2.2. It is useful to note that this methodology describes a generic algorithmic framework which includes energy and momentum conserving algorithms, but also includes algorithms with controllable numerical dissipation. In addition, it produces an entire family of algorithms which conserve energy and momentum. Any algorithm of the V0 family with the values $\rho_\infty^{min} = \rho_\infty^{max} = 1$ will result in an exact energy and momentum preserving algorithm using the above framework. The fundamental reason for this is that any algorithm with the V0 family with $\rho_\infty^{min} = \rho_\infty^{max} = 1$ satisfies the equation of motion exactly at $t_{n+\frac{1}{2}}$. Gonzalez [38] provides a theoretical basis for the two-field form, which essentially says that for any linear material model evaluation of the equation of motion at $t_{n+\frac{1}{2}}$ will result in energy conservation. The independent approximation of strain described above is required to obtain energy conservation in this case due to the necessity of a nonlinear strain definition resulting from very large rotations. A linear strain definition would allow simple evaluation of the internal force term directly from the midpoint displacement. Any algorithm not resulting in consistent evaluation of all terms at the

midpoint will no longer display the conservative properties. It is worth noting that for nonlinear materials (those which are not simply quadratic in their potential energy) simply evaluating the internal force at the midpoint is no longer sufficient to obtain energy conservation. Techniques for dealing with such nonlinearities are given in [39] and [40].

Table 2.2: Predictor multi-corrector coefficients for the incremental a-, v-, and d-form representations

	a-form	v-form	d-form
$\chi_{Pu}^{(1)}$	1	1	1
$\chi_{Pu}^{(2)}$	$\lambda_1\Delta t$	$\lambda_1\Delta t$	0
$\chi_{Pu}^{(3)}$	$\lambda_2\Delta t^2$	$(\lambda_2 - \frac{\lambda_3\lambda_4}{\lambda_5})\Delta t$	0
$\chi_{P\tilde{u}}^{(1)}$	1	1	1
$\chi_{P\tilde{u}}^{(2)}$	$\Lambda_1 W_1\Delta t$	$\Lambda_1 W_1\Delta t$	$(\Lambda_1 W_1 - \frac{\Lambda_3 W_3\lambda_1}{\lambda_3})\Delta t$
$\chi_{P\tilde{u}}^{(3)}$	$\Lambda_2 W_2\Delta t^2$	$(\Lambda_2 W_2 - \frac{\Lambda_3 W_3\lambda_3}{\lambda_5})\Delta t^2$	$(\Lambda_2 W_2 - \frac{\Lambda_3 W_3\lambda_2}{\lambda_3})\Delta t^2$
$\chi_{P\tilde{v}}^{(1)}$	0	0	0
$\chi_{P\tilde{v}}^{(2)}$	1	1	$1 - \frac{\Lambda_5 W_2\lambda_1}{\lambda_3}$
$\chi_{P\tilde{v}}^{(3)}$	$\Lambda_4 W_1\Delta t$	$(\Lambda_4 W_1 - \frac{\Lambda_5 W_2\lambda_4}{\lambda_5})\Delta t$	$(\Lambda_4 W_1 - \frac{\Lambda_5 W_2\lambda_2}{\lambda_3})\Delta t$
$\chi_{P\tilde{a}}^{(1)}$	0	0	0
$\chi_{P\tilde{a}}^{(2)}$	0	0	$-\frac{\Lambda_6 W_1\lambda_1}{\lambda_3\Delta t}$
$\chi_{P\tilde{a}}^{(3)}$	1	$1 - \frac{\Lambda_6 W_1\lambda_4}{\lambda_5}$	$1 - \frac{\Lambda_6 W_1\lambda_2}{\lambda_3}$
χ_{Pu}	$\lambda_3\Delta t^2$	$\frac{\lambda_3}{\lambda_5}\Delta t$	1
$\chi_{P\tilde{u}}$	$\Lambda_3 W_3\Delta t^2$	$\frac{\Lambda_3 W_3}{\lambda_5}\Delta t$	$\frac{\Lambda_3 W_3}{\lambda_3}$
$\chi_{P\tilde{v}}$	$\Lambda_5 W_2\Delta t$	$\frac{\Lambda_5 W_2}{\lambda_5}\Delta t$	$\frac{\Lambda_5 W_2}{\lambda_3\Delta t}$
$\chi_{P\tilde{a}}$	$\Lambda_6 W_1$	$\frac{\Lambda_6 W_1}{\lambda_5\Delta t}$	$\frac{\Lambda_6 W_1}{\lambda_3\Delta t^2}$

Chapter 3

Multibody Dynamics

3.1 Motivation

The basic difference between structural dynamics and constrained dynamics is the presence of the constrained equation, or algebraic equation within the mathematical model. In constrained dynamics, constrained equations which serve as algebraic equations are coupled with the unconstrained equation of motion. The coupled equations can be solved through various techniques such as penalty method, Lagrange multipliers, and augmented Lagrange method. The Lagrange multiplier is the most commonly used and it will be introduced in this research. This coupling of the constrained equations to the ordinary differential equations transforms the system to a case of differential-algebraic equations (DAEs). The study of numerical solutions to differential-algebraic equations is well known over the last 25 years or so. Research in the area is becoming more and more popular since DAEs are encountered frequently in different research fields such as multibody dynamics, molecular dynamics, and electrical engineering, etc.

3.2 Difficulties in DAEs from MBD

Rigid/Flexible multibody dynamics is the subject concerned with computer modeling and analysis of constrained undeformable/deformable bodies that undergo large displacements including large rotations. The large displacements includes rigid body motions as well as elastic deformations. A flexible multibody system may consist of elastic and rigid components which are connected by joint and/or force elements such as springs, dampers, and actuator.

Numerical solutions to differential-algebraic equations from multibody dynamics causes additional difficulties. The major difficulties include: 1) Nonlinearility in equations of motion, and 2) Index reduction. The former comes from different approaches employed in the equation of motion to derive multibody dynamics formulations, and the latter comes from the incorrect satisfaction of the constraint equations.

3.3 Selection of Coordinate System

The configuration of a multibody system can be described using displacements, velocities, and accelerations, measured with respect to a specific ”frame of reference” or ”coordinate system.” In general, there are various methods developed to help transform multibody dynamical problem to the semi-discretized second order system. Generally, the most popular solutions in MBD can be classified as two groups: 1) Inertial reference frame approach (IRF), and 2) Floating reference frame (FRF) approach based on the reference frame used. The differences between inertial frame of reference and floating frame of reference has been discussed in [65]. Furthermore, the number of coordinates required to obtain a reasonable mathematical model for flexible multibody systems can be very large as compared to the number of coordinates used in the analysis of rigid body system. Because of the nonlinearity and large dimensionality, the use

of modern digital computers in the analysis of flexible multibody system becomes a necessity.

3.3.1 Floating Frame of Reference Formulation

The floating frame of reference formulation is currently one of the most widely used methods in the computational simulation of flexible multibody systems. It is implemented in several commercial as well as research general purpose multibody computer programs.

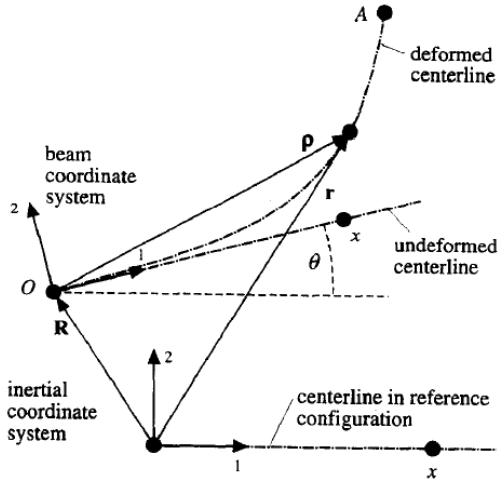


Figure 3.1: Representation of motion in terms of inertial reference frame and local (body) reference frame from [41].

In the floating frame of reference formulations, two sets of coordinates are used to describe the configuration of the deformable bodies as shown in Fig. 3.1. One set describes the location and orientation of a selected body coordinate system, while the second set describes the deformation of the body with respect to its coordinates. Using this description, the global position vector of an arbitrary point on the deformable body *i* can be written as:

$$\mathbf{r}^i = \mathbf{R}^i + \mathbf{A}^i \rho^i \tag{3.1}$$

where all vectors that appear in this equation are shown in Fig. 3.1, and A^i is the transformation matrix that defines the orientation of the body coordinate system with respect to the global coordinate system. ρ is the local deformation respect to body reference coordinates. One of the fundamental problems which arises when using the the floating frame of reference formulation is the selection of the deformable body coordinate system as shown in Fig. 3.2. The shape of deformation of the body is defined in its coordinate system and hence different body references result in different deformation results obviously.

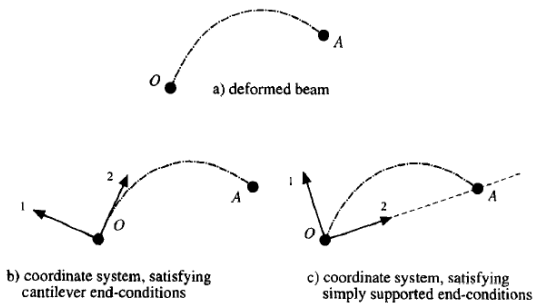


Figure 3.2: Two class of local coordinate from [42].

One of the significant difference between FRF and IRF is the nonlinearility in equation of motion. The inertia forces via the FRF involve nonlinear centrifugal, Coriolis, and tangential terms because the accelerations are measured with respect to a rotating frame (the floating frame). On the other hand, the internal forces are linear for small strains and slow rotational velocities. The linear part of the stiffness matrix is the same as that used in classical linear finite element method.

Besides the selection of body coordinates and nonlinearity in the mass matrix, hinge joint and

general constraint formulated from the floating frame of reference require the addition of algebraic constraint equations. As a result, the influence of nonlinearity in the mass matrix and constraint formulation is also a numerical issue. In this research, the formulation of a hinge joint will be shown to compare the basic differences between different approach.

Finally, although the floating frame formulations are often employed for most multibody simulation commercial codes, it is not suitable for multibody problems which involve large deformation due to the assumption of body coordinate and rigid body motion [42].

3.3.2 Inertial Frame of Reference Approach

Inertial reference frame approach was first know to be used in nonlinear, large deformation FEM since the beginning of 1970s. It was first applied to modeling beam type flexible multibody systems in Simo and Vu-Quoc [43]-[44]. The equation of motion in multibody system are written with respect to the global inertial frame. It has its origins in the nonlinear finite element method and continuum mechanics principles. These techniques were applied to the dynamic analysis of continuum bodies undergoing large rotations and large deformations (including both large strains and large deflections). Since all elements and bodies are described in terms of global inertial frame, there is no selection of body reference as shown in Fig. 3.3.

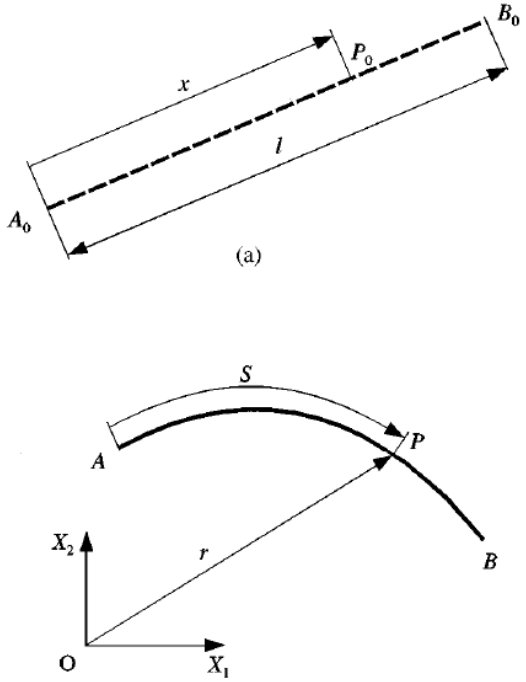


Figure 3.3: Representation of motion in terms of inertial reference frame from [41].

Comparing with the formulation of floating frame, the inertia forces are the product of the mass matrix and the vector of nodal accelerations with respect to the inertial frame as shown in Fig. 3.3. Therefore, the mass matrix is constant since effects such as flexible and rigid body motion, centrifugal and coriolis acceleration are not present. However, the internal forces are nonlinear even for small strains because they are expressed in terms of nonlinear finite strain and stress measures.

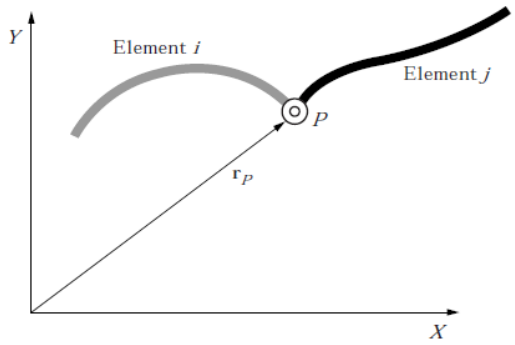


Figure 3.4: Three dimensional revolute joint [41].

Constraints from IRF approach result in different forms compared to FRF. For example, hinge (revolute) joints do not need an extra algebraic equation and can be modelled by letting two bodies share a node as shown in Fig. 3.4.

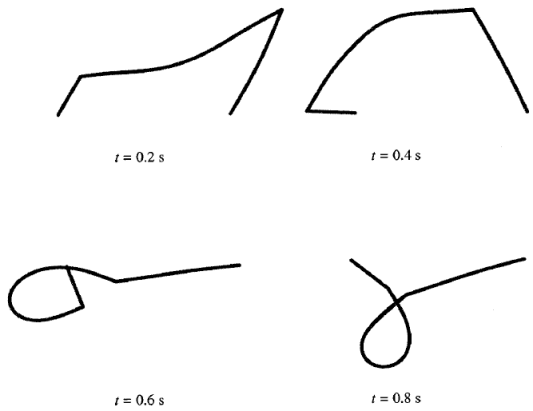


Figure 3.5: Large deformations of the four-bar mechanism [42].

As mentioned, inertial frame formulation can handle flexible MBD undergoing large deflection as shown in Fig. 3.5. However, for MBD involving small deformation and slow rotational speeds, the solution time is generally an order of magnitude greater than that of typical methods based on FRF.

3.4 Index Reduction

Numerical solutions to differential algebraic equations pose additional difficulties not found in the solution of their ODE counterparts, but also inherit many of the same properties and problems. In order to discuss these additional difficulties and the current state for overcoming them one needs first to understand how DAEs are classified. The most important property of a DAE is its so-called "Index". Petzold [18] defines the Index of the DAE as:

the minimum number of times that all or part of [the DAE] must be differentiated with respect to t in order to determine y' as a continuous function of y, t, is the Index of the DAE.

$y(t)$ is the solution to some general DAE $F(t, y, y') = 0$. That is, the Index is the number of times you need to differentiate the DAE in order to recover a system which can be written as a standard ODE. Within the area of multibody dynamics (MBD) the equation of motion coupled to the constraint equations naturally give rise to a set of Index 3 DAEs, due to the constraint equations being constraints on the displacement of the system (as opposed to velocity or acceleration constraints). This is the most common form of constraint and includes revolute joints, sliding contact, rolling contact, and many other common joints between two structural bodies. Though these Index 3 DAEs are quite natural to MBD systems, solving them directly in this form is not very common in the literature. The majority of the codes used in solving MBD problems reformulate the natural Index 3 equations into either Index 2 or Index 1 formulations. It is commonly noted that the higher the Index of the DAE, the more difficult it is to obtain a stable numerical solution. Many authors cite stiffness, instability, and order reduction as reasons to avoid solving the Index 3 problem directly. Though this seems to be a generally accepted norm, it is rare to find an explanation of the source of these issues. That said, multibody systems of any practical interest almost certainly involve large rotation or nonlinear constraints (resulting

in nonlinear system equations), making any sort of analytic proof nearly impossible.

Examples of simulations run directly on Index 3 DAEs which ”blow up” preface almost every paper introducing a new Index 2 or Index 1 algorithm. Even the most fundamental MBD problem, namely the dynamic response of a simple pendulum, when run using existing methods fails. That is not to say there are not specific Index 3 DAE solvers that can solve the problem, but these solvers either use generalized coordinates (not Cartesian coordinates as required by most standard finite element codes) or use controllable numerical dissipation which artificially removes energy from the system. More often, this unbounded oscillation is the basis for developing algorithms in Index 2 or 1 which are able to avoid the problem. Borri [47] demonstrates this failure for the simple pendulum by showing that the numerical solution to the Index 3 problem (without dissipation), though able to produce smooth and stable displacements and velocities, will develop oscillations in the accelerations (and therefore Lagrange multipliers) as seen in Fig. 3.6. Also, it is noted that the order of accuracy of the Lagrange multipliers will suffer a reduction with respect to the displacements and velocities of the system.

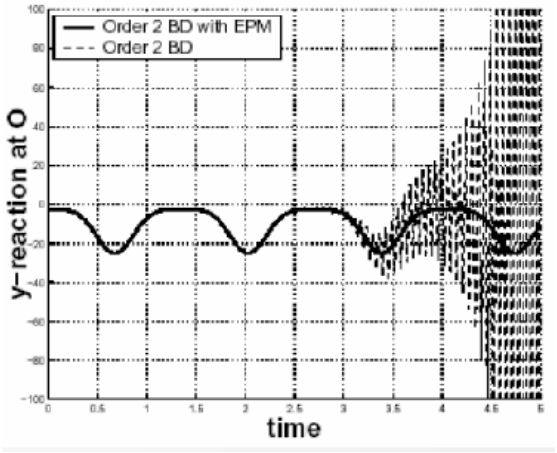


Figure 3.6: Simple Pendulum: Constraint Force Oscillations from [47].

Another example, this time of a double pendulum, is discussed at length by Geradin in [12]. Using the (numerically non-dissipative) Newmark method it is again shown that the accelerations and Lagrange multipliers exhibit unbounded oscillatory behaviour, see Fig. 3.7. These unbounded oscillations eventually lead to convergence problems in the nonlinear iterations and the solutions is unable to proceed. In an attempt to remedy these unstable oscillation it is later shown that using HHT-α method removes the unwanted (and non-physical) oscillations, but at the price of dissipating energy from what should be a conservative system.

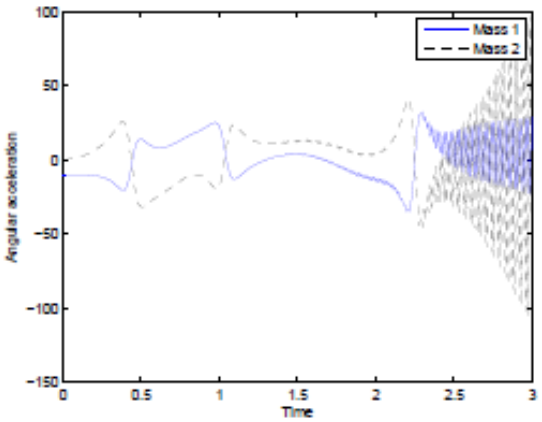


Figure 3.7: Newmark Method: Double Pendulum Acceleration blow-up from [12].

The most basic idea to lower the Index of a DAE is to simply differentiate the constraint equation with respect to time. Assuming we start with a constraint equation on the displacements, one such differentiation will result in an Index 2 DAE where the constraint equation now involves the time derivative of the displacement, the velocity. One more differentiation will result in an Index 1 DAE where the constraint equation involves the accelerations of the system. Upon implementation of standard BDF or RK time integration algorithms these Index 2 and Index 1 systems no longer exhibit the unbounded oscillations which occur in the Index 3 system. Unfortunately, while correcting one issue, the Index reduction introduces a new issue: constraint drift. This problem is, among a variety of other places, described by Harier in [1]. The fundamental

problem is that when the constraint equation occurs at either the velocity or acceleration (Index 2 or Index 1 respectively) the original position level constraint ”drift” away from its true solution as shown in Fig. 3.8. That is, the original Index 3 system which defines the mathematical model for the physical system under consideration is no longer being satisfied to the nonlinear tolerance.

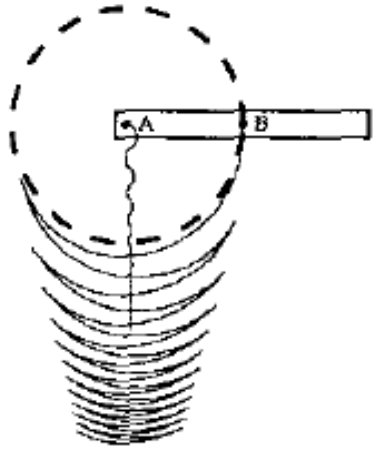


Figure 3.8: Drift effect of a Swinging pendulum [53].

To deal with the problem of constraint drift, a wide variety of techniques have been developed. Arguably the most famous and widely known is termed Baumgarte’s stabilization [21]. Baumgarte proposed a method which combines the position, velocity, and acceleration constraint into a single equation which was originally used in control theory. This method includes two parameters which are used to control the combined constraint equation. Though optimal values of these parameters which guarantee solution stability often do exist, it is required that the user is somehow able to determine them. With no robust mathematical method to determine these parameters the user is likely to simply use a guess and check method to find what works empirically, and is often not an efficient process.

Another common way of dealing with constraint drift are the so-called ”projection methods.” First made popular by Lubich [48], these methods work by first solving the reduced Index system and then projecting that solution back onto the solution manifold of the original Index 3 system from which it has drifted away. Extensions of the original method are abound in the literature. Popular is the well known Index 1 DAE code DASSL [49]. These projection methods are again, in general, able to overcome the instability seen in the Index methods. In general, they are able to overcome the instability seen in the Index 3 solution but at the cost of: a) reformulation by Index reduction, and b) additional computational cost of the projection.

The final method commonly used to overcome drift of the constraint equation within a reduced Index formulation is again to take the derivative of the constraint equations with respect to time, but instead of using only the first or second (velocity or acceleration level) derivative equation, the original position constraint is used as well. That is, the original equations of motion are coupled with not only the position level constraint, but also with one or possibly both of the velocity and acceleration level constraints. In this case, of course, there are now more equations than unknowns which is referred to as an overdetermined system. This formulation has garnered significant support after being introduced by Gear in [50]. Recently both Lunk [51] and Negrut [52] have used this method in conjunction with popular time integration algorithms from structural dynamics with limited success.

All of these methods were developed to overcome the initial problem of instability in the natural Index 3 formulation. Though capable of removing the problem of acceleration instability, they introduce error in the position level (Index 3) constraint which then requires additional effort to remove. Position constraint satisfaction guarantees the energy conserving issue. Without satisfaction of physical constraint (displacement, and lower level constraint velocity and

acceleration), if there were an algorithm which could eliminate the instability found in current methods to solve Index 3 DAEs, none of these Index reduction methods would be necessary or desirable. If constraint satisfaction and numerical stability are top priorities, then Index 1 numerical simulation with post processing (projection method) is desirable since all quantity are properly computed.

3.5 Implementation

We have seen in the previous chapter that the method of extending the parent linear time integration algorithm to nonlinear problems can drastically change the performance of the algorithm, as well as provide certain properties such as phase space preservation or energy conservation. To properly assess the extension of the GSSSS family of algorithms to nonlinear differential-algebraic equations, it is necessary to show the total process of derivation from mathematical model to semi-discrete equations of motion, and then to the fully-discretized equations.

The semi-discrete equations of motion for constrained dynamics may be developed through IRF/FRF approaches, and with the addition of the Lagrange equation to incorporate the constraint equation into the Lagrangian. In structural dynamics, we define $L = T - V$, where T is the kinetic energy and V is the potential energy of a conservative system. For constrained dynamics, we include the product of the Lagrange multiplier and their corresponding constraints when minimizing the variation of the L. We employ the descriptive function, $L = T - V + \Phi_i \lambda_i$. Thus the semi-discrete equation of motion is defined by the standard form as:

$$\frac{d}{dt}\left(\frac{\partial \mathbf{L}}{\partial \dot{u}_i}\right) - \frac{\partial \mathbf{L}}{\partial u_i} = 0 \tag{3.2}$$

Now include the product of the vector of Lagrange multiplier and the Jacobian of the constraint

matrix. Define $\frac{\partial \Phi}{\partial u} = C(u)$ the resulting semi-discrete equation of motion subject to position constraint (Index 3) equations can be written as:

$$\begin{aligned} &\mathbf{M}\ddot{u} + \mathbf{F}_{int} + \mathbf{C}^T\lambda = \mathbf{F}_{ext} \\ &\mathbf{\Phi}(u,t) = 0 \end{aligned} \tag{3.3}$$

Similarly, rewritten as in velocity form (Index 2) DAE we have:

$$\begin{aligned} &\mathbf{M}\ddot{u} + \mathbf{F}_{int} + \mathbf{C}^T\lambda = \mathbf{F}_{ext} \\ &\mathbf{\Phi}_u(u,t)\dot{u} + \mathbf{\Phi}_t(u,t) + 2\mathbf{\Phi}_{qt}(u,t)\dot{u} + \mathbf{\Phi}_{tt}(u,t) = 0 \end{aligned} \tag{3.4}$$

and finally in acceleration (Index 1) form:

$$\begin{aligned} &\mathbf{M}\ddot{u} + \mathbf{F}_{int} + \mathbf{C}^T\lambda = \mathbf{F}_{ext} \\ &\mathbf{\Phi}_u(u,t)\ddot{u}(\mathbf{\Phi}_u(u,t))_u\dot{u} + 2\mathbf{\Phi}_{ut}(u,t)\ddot{u} + \mathbf{\Phi}_{tt}(u,t) = 0 \end{aligned} \tag{3.5}$$

In this research, three approaches will be illustrated from IRF/FRF to derive the mathematical model of the equation of motion. Traditionally, research originated for the inertial reference formulation to multibody dynamics to solve large deformation problems. In addition, an improved version of first method was introduced via the so-called absolute nodal coordinate formulation (ANCF) by Shabana [17]. The last is the basic floating reference frame approach. These methods are well-known for a long time in the MBD research field; thus it is essential to point out some relevant details for using these methods. Each of them is discussed next. In this research, three kinds of elements will be shown: 1) elastic bar, 2) Euler-Bernoulli beam and 3) Timoshenko beam. The process for each kind of element is very similar, and the reader can follow step by step to learn how to properly establish the correct model. More further, this process can also apply to other kinds of elements.

3.5.1 Inertial Reference Frame Method (IRF)

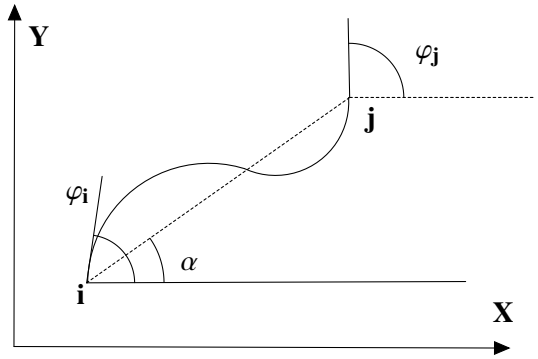


Figure 3.9: Inertial reference formulation: single element with two nodes.

Inertial Reference Frame Method (IRF) is the basic approximation for structural dynamics and multibody dynamics. All nodal displacements are defined (in two dimension) as three quantities: x- and y-direction global displacements and slope subject to global X coordinates as shown in Fig. 3.9. As a result, the global displacement vector can then be written as:

$$\begin{aligned} q &= [q_i \quad q_j]^T \\ q_i &= [x_i \quad y_i \quad \varphi_i]^T \\ q_j &= [x_j \quad y_j \quad \varphi_j]^T \end{aligned} \tag{3.6}$$

where x_i and y_i are nodes defined as the global position of ith node, and φ is the global slope measured respect to global X coordinate. Then the longitudinal strain ϵ_l and local slope change at each node ϵ_i, ϵ_j can be defined as

$$\begin{aligned} \epsilon_l &= \frac{d-L}{L} = (\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} - L)/L \\ \epsilon_i &= \varphi_i - \alpha \\ \epsilon_j &= \varphi_j - \alpha \end{aligned} \tag{3.7}$$

where d is nodal distance and L is undeformed length of element and α is the orientation angle which can be written as

$$\alpha(q) = cos^{-1}\left[\frac{x_j - x_i}{L}\right] = sin^{-1}\left[\frac{x_j - x_i}{L}\right] \tag{3.8}$$

With such a model, it is possible to deal with flexible bodies for two dimensional dynamic response.

[a] Elastic Bar Model

The most basic element is the bar element. It does not contain any rotation degree of freedom which implies

$$\varphi_i = \varphi_j = \alpha \tag{3.9}$$

As a result, in the elastic bar model, one can reduce the position vector to

$$\begin{aligned} q &= [q_i \quad q_j]^T \\ q_i &= [x_i \quad y_i]^T \\ q_j &= [x_j \quad y_j]^T \end{aligned} \tag{3.10}$$

The element longitudinal displacements $u(x)$ and x is local coordinate of element. In finite element analysis, it is assumed to be a linear function subject to two boundary conditions:

$$\begin{aligned} u(x) &\simeq a_0 + a_1 x \\ u(0) &= 0 \\ u(L) &= L\epsilon_l \end{aligned} \tag{3.11}$$

so that the local displacement $u(x)$ can be evaluate by longitudinal strain ϵ_l:

$$u(x) \simeq \epsilon_l x \tag{3.12}$$

Since there is no rotation degrees of freedom in the elastic bar element, ϵ_i and ϵ_j are both zero.

Mass Matrix The kinetic energy T of an element is calculated as:

$$\mathbf{T} = \frac{1}{2}\int_V \rho \dot{r}^T \dot{r} dV = \frac{1}{2}\int_0^L \rho \dot{r}^T \dot{r} A dx \tag{3.13}$$

where L is the length, A is the cross-sectional area, and ρ is the density of the element. In this equation, $\dot{r}$ is the global velocity vector, which can also be defined through the interpolation functions (or called shape function):

$$\mathbf{M} = \frac{1}{2}\int_0^L \rho A \mathbf{S}^T \mathbf{S} dx \tag{3.14}$$

which is constant as shown in Eq. (A.2)

Stiffness Matrix It is obvious in finite elements that internal energy U of an elastic bar element can be written as

$$\mathbf{U} = \frac{1}{2}\int_V E(u'(x))dV = \frac{1}{2}\int_0^L EA\epsilon^2 dx \tag{3.15}$$

and

$$\mathbf{U} = \frac{EA}{2}\epsilon_l^2 \tag{3.16}$$

The vector of internal force $\mathbf{F}_{int}$ of the generalized elastic forces due to the longitudinal deformation can be defined as

$$\mathbf{F}_{int} = \left(\frac{\partial \mathbf{U}}{\partial q}\right)^T \tag{3.17}$$

The stiffness matrix can be obtained by

$$\mathbf{K} = \left(\frac{\partial \mathbf{F}_{int}}{\partial q}\right) \tag{3.18}$$

as shown in Eq. (A.3).

External Force The work done by external force can be defined as:

$$\mathbf{W}_{ext} = \int_0^L r(x)\cdot f^{ext}dx = q^T\left(\int_0^L \mathbf{S}^T f^{ext}dx\right) \tag{3.19}$$

where f^{ext} is the external load. The equivalent external force can them be defined as

$$\mathbf{F}_{ext} = \frac{\partial \mathbf{W}_{ext}}{dq} = \int_0^L \mathbf{S}^T f^{ext} dx \tag{3.20}$$

For the constant gravity g, the equivalent load is shown in Eq. (A.5). Notice the shape functions for both mass matrix and external force satisfy

$$\begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \mathbf{S} \begin{Bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{Bmatrix} \tag{3.21}$$

where r_1 and r_2 are global $X-$ and $Y-$coordinate and this relationship is exactly the same shown in the ANCF formulation, which yields the shape function as:

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & s_2 & 0 \\ 0 & s_1 & 0 & s_2 \end{bmatrix} \tag{3.22}$$

The functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - \xi, \qquad s_2 = \xi \tag{3.23}$$

and $\xi = x/L$.

[b] Euler-Bernoulli Beam

In the beam model, one cannot eliminate the rotation degrees of freedom. The derivation of Euler-Bernoulli beam follows:

$$
\begin{aligned}
w(x) &\simeq a_0 + a_1 x + a_2 x^2 + a_3 x^3 \\
w(0) &= w(L) = 0 \\
w'(0) &= \epsilon_i \\
w'(L) &= \epsilon_j
\end{aligned}
\tag{3.24}
$$

and result in the local displacement function:

$$
w(x) \simeq \epsilon_i x - \frac{2\epsilon_i + \epsilon_j}{L} x^2 + \frac{\epsilon_i + \epsilon_j}{L^2} x^3 \tag{3.25}
$$

Mass Matrix The mass matrix, like elastic bar, follows the same procedure to obtain the expression, which the kinetic energy $\mathbf{T}$ of an element is calculated as:

$$
\mathbf{T} = \frac{1}{2} \int_V \rho \dot{r}^T \dot{r} dV = \frac{1}{2} \int_0^L \rho \dot{r}^T \dot{r} A dx \tag{3.26}
$$

where L is the length, A is the cross-sectional area, and ρ is the density. In this equation, $\dot{r}$ is the global velocity vector, which can also be defined through the interpolation functions (or shape functions):

$$
\mathbf{M} = \frac{1}{2} \int_0^L \rho A \mathbf{S}^T \mathbf{S} dx \tag{3.27}
$$

which is constant as shown in Eq. (A.53)

Stiffness Matrix For the beam element, the deformation configuration is caused by two parts: longitudinal deformation and bending deformation. As a result, except the longitudinal strain energy we discussed in elastic bar, the bending strain energy is defined as

$$\mathbf{U}_b = \frac{EI}{2}\int_0^L (w'')^2 dx = \frac{2EI}{L}(\epsilon_i^2 + \epsilon_i\epsilon_j + \epsilon_j^2) \tag{3.28}$$

and the total strain energy can be defined as

$$\mathbf{U} = \mathbf{U}_l + \mathbf{U}_b = \frac{EA}{2}\epsilon_l^2 + \frac{2EI}{L}(\epsilon_i^2 + \epsilon_i\epsilon_j + \epsilon_j^2) \tag{3.29}$$

The internal force and stiffness matrix can then be evaluated as

$$\mathbf{F}_{int} = \left(\frac{\partial \mathbf{U}}{\partial q}\right)^T \tag{3.30}$$

as shown in Eq. (A.54), and the stiffness matrix can be obtained by

$$\mathbf{K} = \left(\frac{\partial \mathbf{F}_{int}}{\partial q}\right) \tag{3.31}$$

External Force The work done by external force can also follow the same routine such as we did for the elastic bar

$$\mathbf{W}_{ext} = \int_0^L r(x)\cdot f_{ext} dx = q^T\left(\int_0^L \mathbf{S}^T f^{ext} dx\right) \tag{3.32}$$

where f^{ext} is the external load. The equivalent external force can them be defined as

$$\mathbf{F}_{ext} = \frac{\partial \mathbf{W}_{ext}}{dq} = \int_0^L \mathbf{S}^T f^{ext} dx \tag{3.33}$$

For constant gravity force, the external force is shown in Eq. (A.57).

[c] Timoshenko Beam

Shear deformation can be introduced by relaxing the assumption in the classical Euler-Bernoulli beam theory, that plane sections initially normal to the neutral axis remain plane and normal to that axis during deformation. The total rotation of the centerline of the beam arising from both bending and shear deformation can, therefore, be written as

$$\frac{dw(x)}{dx} = \phi(x) + \gamma(x) \tag{3.34}$$

where $w(x)$ represents the transverse displacement in element. $\gamma(x)$ is a measure of the shear strain at points along the centerline, and $\phi(x)$ is the rotation of a line element along the centerline due to the bending effect only. In considering the shear effect, the assumption that a plane section originally normal to the neutral axis remains plane will be retained; however, because of shear this section will not remain normal to the neutral axis. The basic assumption for Euler-Bernoulli beam is the same; however, the transverse displacement now becomes as in [61]:

$$w(x) = \left(\frac{\epsilon_i + \epsilon_j}{L^2(1+\phi)}\right)x^3 + \left(\frac{(4+\phi)\epsilon_i + (2-\phi)\epsilon_j}{L(1+\phi)}\right)x^2 + \left(\frac{(1+\frac{\phi}{2})\epsilon_i - \frac{1}{2}\epsilon_j}{1+\phi}\right)x \tag{3.35}$$

where ϕ is the shear deformation parameter, given by

$$\phi = \frac{12EI}{\kappa GAL^2} \tag{3.36}$$

One can obviously tell that if ϕ is set equal to zero, the transverse displacement function will be identical to Euler-Bernoulli beam. That is to say, for a long, slender beam, the Timoshenko beam reduces to the Euler-Bernoulli beam. The mass matrix, stiffness matrix, and external force follows the same process shown for the Euler-Bernoulli beam for the inertial reference frame approach. In this research, all terms are shown in Eq. (A.113) to Eq. (A.128).

3.5.2 Simplified Absolute Nodal Coordinate Formulation (ANCF-S)

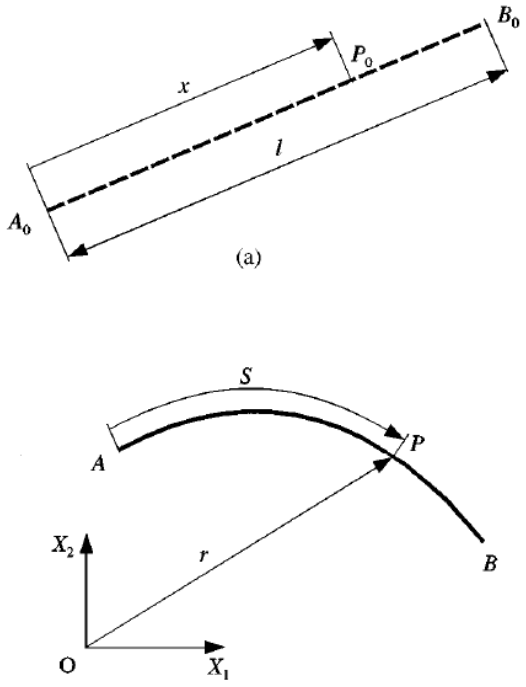


Figure 3.10: Absolute nodal coordinate approach from [41].

In the absolute nodal coordinate formulation, the nodal coordinates of the elements are defined in a fixed inertial coordinate system, and consequently the use of body frame is eliminated. The element nodal coordinates represent global displacements and slopes, and no infinitesimal or finite rotations are used as nodal coordinates. Furthermore, no assumption on the magnitude of the element rotations is made. In the absolute nodal coordinate formulation, elements such as beams and plates considered in the finite element literature as non-isoparametric, can be treated as isoparametric elements. The global position vector $\mathbf{r}$ of an arbitrary point P on the neutral axis of a two dimensional element is defined in terms of the nodal coordinates and the element shape function, as shown in Fig. 3.10 as

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \mathbf{Sq} \tag{3.37}$$

where **S** is the global shape function which has a complete set of rigid-body modes, and **q** is the vector of element local coordinates. Notice that for different kinds of element, the number of nodal quantities will be different. Also, the traditional absolute nodal coordinate formulation results in high nonlinearity in stiffness matrix. This research uses a simplified version of ANCF to simulate flexible body dynamics (ANCF-S).

[a] Elastic Bar Model

Notice in traditional finite element formulation, the displacement of an elastic bar is linear, which can be written as:

$$u(x) = a_0 + a_1 x \tag{3.38}$$

where x is local element coordinate system. Here, we also follow the same assumption for the global displacement, which for a bar element is:

$$\mathbf{r}(x) = \begin{bmatrix} r_1(x) \\ r_2(x) \end{bmatrix} = \begin{bmatrix} a_0 + a_1 x \\ b_0 + b_1 x \end{bmatrix} \tag{3.39}$$

subject to four nodal quantities q_1, q_2, q_3 and q_4 which represent the global displacements in each direction (X_1, X_2) at each node. In this way, the displacement field for an elastic bar element can be expressed in terms of the shape functions:

$$\mathbf{r}(x) = \begin{bmatrix} r_1(x) \\ r_2(x) \end{bmatrix} = \mathbf{S}q = \begin{bmatrix} s_1 & 0 & s_2 & 0 \\ 0 & s_1 & 0 & s_2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{3.40}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - \xi, \qquad s_2 = \xi \tag{3.41}$$

and $\xi = x/L$. It can be shown that the preceding shape function contains a complete set of rigid-body modes that can describe arbitrary rigid-body translational and rotational displacements. Since the displacement field has been well constructed, the mass matrix, stiffness matrix, and external force can now be formulated through the usage of the shape functions.

Mass Matrix The kinetic energy $\mathbf{T}$ of an element is calculated as:

$$\mathbf{T} = \frac{1}{2} \int_V \rho \dot{r}^T \dot{r} dV = \frac{1}{2} \int_0^L \rho \dot{r}^T \dot{r} A dx \tag{3.42}$$

where L is the length, A is the cross-sectional area, and ρ is the density of the element. In this equation, $\dot{r}$ is the time derivative of the global position vector, and the substitution of $r = \mathbf{S}q$ yields

$$\mathbf{T} = \dot{q}^T \left[\frac{1}{2} \int_0^L \rho A \mathbf{S}'^T \mathbf{S}' dx \right] \dot{q} \tag{3.43}$$

or

$$\mathbf{T} = q^T \mathbf{M} q \tag{3.44}$$

and the mass matrix can then be expressed as

$$\mathbf{M} = \frac{1}{2} \int_0^L \rho A \mathbf{S}^T \mathbf{S} dx \tag{3.45}$$

which is constant as shown in Eq. (A.17).

Stiffness Matrix It is obvious that internal energy U of an elastic bar element can be written as

$$\mathbf{U} = \frac{1}{2} \int_V E \epsilon_l^2 dV = \frac{1}{2} \int_0^L EA \epsilon_l^2 dx \tag{3.46}$$

where ϵ_l is the longitudinal strain since there is only longitudinal deformation in the local coordinate x. For a bar element, it is easily written as:

$$\epsilon_l = \frac{d - L}{L} \tag{3.47}$$

where d is the current distance between the nodes of the elements defined as

$$d = \sqrt{(q_3 - q_1)^2 + (q_4 - q_2)^2} \tag{3.48}$$

The vector of internal force $\mathbf{F}_{int}$ of the generalized elastic forces due to the longitudinal deformation can be defined as

$$\mathbf{F}_{int} = \left(\frac{\partial \mathbf{U}}{\partial q} \right)^T \tag{3.49}$$

and the stiffness matrix can be obtained by

$$\mathbf{K} = \left(\frac{\partial \mathbf{F}_{int}}{\partial q} \right) \tag{3.50}$$

Notice that either IRF or ANCF approach leads to nonlinear stiffness matrix shown in Eq. A.18 to Eq. A.21.

External Force The work done by external force is defined as

$$\mathbf{W}_{ext} = \int_0^L r(x) \cdot f_{ext} dx = q^T \left(\int_0^L \mathbf{S}^T f_{ext} dx \right) \tag{3.51}$$

where f^{ext} is the external load. Then, the equivalent external force can then be defined as

$$\mathbf{F}_{ext} = \frac{\partial \mathbf{W}_{ext}}{dq} = \int_0^L \mathbf{S}^T f^{ext} dx \tag{3.52}$$

For constant external force case, see Eq. A.22.

[b] Euler-Bernoulli Beam

The derivation of Euler-Bernoulli beam theory for flexible multibody dynamics follows . We assume the displacement vector to be a cubic function as

$$\mathbf{r}(x) = \begin{bmatrix} r_1(x) \\ r_2(x) \end{bmatrix} = \begin{bmatrix} a_0 + a_1 x + a_2 x^2 + a_3 x^3 \\ b_0 + b_1 x + b_2 x^2 + b_3 x^3 \end{bmatrix} \tag{3.53}$$

with eight global quantities

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \tag{3.54}$$

This vector of absolute nodal coordinates includes the global displacements

$$q_1 = r_1|_{x=0}, \qquad q_2 = r_2|_{x=0}, \qquad q_5 = r_1|_{x=L}, \qquad q_6 = r_2|_{x=L} \tag{3.55}$$

and the global slopes of the element nodes are defined as

$$q_3 = \frac{\partial r_1}{\partial x}|_{x=0}, \qquad q_4 = \frac{\partial r_2}{\partial x}|_{x=0}, \qquad q_7 = \frac{\partial r_1}{\partial x}|_{x=L}, \qquad q_8 = \frac{\partial r_2}{\partial x}|_{x=L} \tag{3.56}$$

Here, x is the coordinate of an arbitrary point on the element in the undeformed configuration, and L is the original length of the beam element (at point A, $x = 0$; while at point B, $x = L$ as shown in Fig. 3.10). A cubic polynomial is employed to describe both components of the displacements. Therefore, the global shape function $\mathbf{S}$ can be written as

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 & 0 \\ 0 & s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 \end{bmatrix} \tag{3.57}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - 3\xi^2 + 2\xi^3, \qquad s_2 = L(\xi - 2\xi^2 + \xi^3), \qquad s_3 = 3\xi^2 - 2\xi^3, \qquad s_4 = L(\xi^3 - \xi^2) \tag{3.58}$$

and $\xi = x/L$. It can be shown that the preceding shape function contains a complete set of rigid-body modes that can describe arbitrary rigid-body translational and rotational displacements.

Mass Matrix The mass matrix for Euler Bernoulli beam follows the same derivation as shown in the elastic bar element. It can be written as

$$\mathbf{M} = \frac{1}{2}\int_0^L \rho A \mathbf{S}^T \mathbf{S} dx \tag{3.59}$$

As we discussed in the previous section, the mass matrix remains constant due to the fact that displacement vector is automatically measured in the inertial reference frame.

Stiffness Matrix The beam element cross-sections are assumed to remain plane and perpendicular to the beam center line. The configuration of a beam element at time t can be defined using the parametric equation

$$\mathbf{r} = \mathbf{r}(x), \quad 0 \leq x \leq L \tag{3.60}$$

where the vector $\mathbf{r}$ defines the coordinates of an arbitrary point on the beam axis, and x is considered as a parameter that represents the coordinate of the point in the undeformed configuration. The infinitesimal arc length is

$$ds = \sqrt{\mathbf{r}'^T \mathbf{r}'} dx \tag{3.61}$$

where

$$\mathbf{r}' = \frac{d\mathbf{r}}{dx} \tag{3.62}$$

It follows that the length of the beam after deformation is given by

$$\int_L ds = \int_0^L \sqrt{\mathbf{r}'^T \mathbf{r}'} dx \tag{3.63}$$

The quantity $\sqrt{\mathbf{r}'^T \mathbf{r}'}$ represents the deformation gradient f for longitudinal deformations. This deformation gradient can be considered as the Cauchy-Green longitudinal strain which is defined here as

$$f = \frac{ds}{dx} \tag{3.64}$$

Adopting a Lagrangian viewpoint, the longitudinal strain ϵ_l is defined by a relationship between the current and the original length of an infinitesimal segment of the beam as follows:

$$ds^2 - dx^2 = 2dx\epsilon_l dx \tag{3.65}$$

which implies that

$$\epsilon_l = \frac{1}{2}(f^2 - 1) = \frac{1}{2}(\mathbf{r}'^T \mathbf{r}' - 1) \tag{3.66}$$

For isotropic materials, the strain energy due to the longitudinal deformation can be written as

$$\mathbf{U}_l = \frac{1}{2} \int_0^L EA\epsilon_l^2 dx \tag{3.67}$$

where E is Young's modulus. Note that in this development the problem of objectivity is solved by referring all the quantities to the undeformed configuration. The effect of bending can be introduced using the equation

$$M = EI\kappa \tag{3.68}$$

where I is the second moment of area. The Serret-Frenet formulas give the following result for curvature κ of a curve described in the parametric form:

$$\kappa = \left\| \frac{d^2\mathbf{r}}{ds^2} \right\| \tag{3.69}$$

In this case, the strain energy $\mathbf{U}_b$, due to bending can be written as

$$\mathbf{U}_b = \frac{1}{2} \int_0^L EI\kappa^2 dx \tag{3.70}$$

Therefore, the expression for the strain energy that accounts for both axial and bending effects is

$$\mathbf{U} = \mathbf{U}_l + \mathbf{U}_b = \frac{1}{2} \int_0^L [EA\epsilon_l^2 + EI\kappa^2] dx \tag{3.71}$$

It is obvious that the introduction of $\mathbf{r} = \mathbf{S}q$ brings a lot of benefit for deriving the stiffness matrix. The total energy can then be written as

$$\mathbf{U} = \frac{1}{2} \int_0^L [EA(q^T \mathbf{S}'^T \mathbf{S}' q - 1) + EI\mathbf{S}''^T \mathbf{S}'' q)^2] dx \tag{3.72}$$

and follows

$$\mathbf{F}_{int} = \left(\frac{\partial \mathbf{U}}{\partial q} \right)^T \tag{3.73}$$

The stiffness matrix can be obtained by

$$\mathbf{K} = \left(\frac{\partial \mathbf{F}_{int}}{\partial q} \right) \tag{3.74}$$

as shown in Eq. (A.71) to Eq. (A.82).

External Force The work done by external force can also be computed through

$$\mathbf{W}_{ext} = \int_0^L f^{ext} r(x) dx = \left(\int_0^L f^{ext} \mathbf{S} dx \right) q \tag{3.75}$$

where f^{ext} is the external load. The equivalent external force can them be defined as

$$\mathbf{F}_{ext} = \frac{\partial \mathbf{W}_{ext}}{dq} = \int_0^L f^{ext} \mathbf{S} dx \tag{3.76}$$

which for constant gravity, the equivalent external force is shown in Eq. (A.83).

[c] Timoshenko Beam

The basic concepts and background has been previously introduced in the section on Inertial reference frame approach. Researchers can follow the same steps to obtain the mass matrix, stiffness matrix and external force. The only two differences are the expression for the shape function and the stiffness matrix due to shear deformation. The shape function for a Timoshenko beam is shown in [60] and [61] as:

$$\mathbf{S}(\xi, \eta) = \frac{1}{1+\varphi} \begin{bmatrix} s_1 & t_1 & s_2 & t_2 & s_3 & t_3 & s_4 & t_4 \\ t_1 & s_1 & t_2 & s_2 & t_3 & s_3 & t_4 & s_4 \end{bmatrix} \tag{3.77}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$
\begin{aligned}
s_1 &= 1 - 3\xi^2 + 2\xi^3 + (1-\xi)\phi, \qquad s_2 = L[\xi - 2\xi^2 + \xi^3 + \frac{1}{2}(\xi - \xi^2)\phi] \\
s_3 &= 3\xi^2 - 2\xi^3 + \xi\phi, \qquad s_4 = L[\xi^3 - \xi^2 - \frac{1}{2}(\xi - \xi^2)\phi] \\
t_1 &= 6(\xi - \xi^2)\eta, \qquad t_2 = L[-1 + 4\xi - 3\xi^2 - (1-\xi)\phi]\eta \\
t_3 &= 6(-\xi + \xi^2)\eta, \qquad t_4 = L[2\xi - 3\xi^2 - \xi\phi]\eta
\end{aligned}
\tag{3.78}
$$

and ϕ is the shear deformation parameter, given by

$$
\phi = \frac{12EI}{\kappa GAL^2} \tag{3.79}
$$

and $\xi = x/L$, $\eta = y/L$ where y denotes the local transverse direction coordinate. It is also obvious that if the beam is long and slender, the shape functions will reduce to the shape function for the Euler-Bernoulli beam. Also, [59] shows the basic continuum mechanics approach for deriving the stiffness matrix by setting η is equal to zero. The mass matrix and external force can be formulated through the same procedure as shown in Eq. A.129 to Eq. A.143.

Mass Matrix The mass matrix for Timoshenko beam follows the same steps. It can be written as

$$
\mathbf{M} = \frac{1}{2}\int_0^L \rho A \mathbf{S}^T \mathbf{S} dx \tag{3.80}
$$

Stiffness Matrix In the formulation of stiffness matrix, the total internal energy can be separated into longitudinal strain energy, bending strain energy, and shear strain energy as:

$$
\mathbf{U} = \mathbf{U}_l + \mathbf{U}_b \tag{3.81}
$$

Then strain energy due to tension, and bending are defined as:

$$\begin{aligned} \mathbf{U}_l &= \frac{1}{2}\int_0^L EA\epsilon_l^2 dx \\ \mathbf{U}_b &= \frac{1}{2}\int_0^L EI\kappa^2 dx \end{aligned} \tag{3.82}$$

where ϵ_l and κ can then be defined as

$$\begin{aligned} \epsilon_l &= \frac{1}{2}(q^T \mathbf{S}_{,\xi}^T \mathbf{S}_{,\xi} q - 1) \\ \kappa &= \left\| \frac{d^2\mathbf{r}}{ds^2} \right\| \end{aligned} \tag{3.83}$$

External Force The work done by external force can also be computed through

$$\mathbf{W}_{ext} = \int_0^L r(x) \cdot f^{ext} dx = q^T \left(\int_0^L f^{ext} \mathbf{S}^T dx \right) \tag{3.84}$$

where f^{ext} is the external load. The equivalent external force can be defined as

$$\mathbf{F}_{ext} = \frac{\partial \mathbf{W}_{ext}}{dq} = \int_0^L f^{ext} \mathbf{S}^T dx \tag{3.85}$$

3.5.3 Floating Reference Frame Approach

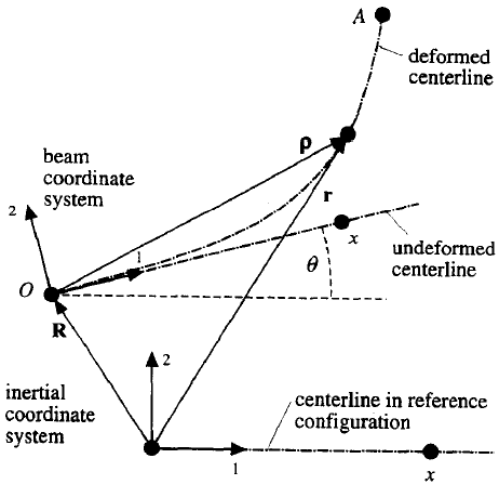


Figure 3.11: Floating reference frame approach from [41].

In the floating reference frame approach, a coordinate system is assigned to each deformable body. The large rigid body displacement is described using a set of absolute Cartesian and orientation coordinates. The deformation of the body is described using a set of local shape functions and a set of deformation coordinates defined in the body coordinate. In this approach, it is assumed that there is no rigid body motion between the body and its coordinate system.

Consider a deformable body in Fig. 3.11. the introduction of rotation tensor $\mathbf{A} = \mathbf{A}(t)$ couples the absolute coordinate and local coordinate. The position vector are now follow as

$$\mathbf{r}(x,t) = \mathbf{R}(x,t) + \mathbf{A}(t)p(x,t) \tag{3.86}$$

where $p(x,t)$ describes the deformation, which can be defined by local shape function and local quantities as:

$$p(x,t) = \mathbf{S}_l(x) q_f(t) \tag{3.87}$$

where $q_f(t)$ is the vector of time dependent deformation coordinates. As the kinematic description of equation is used, it is assumed that there is no rigid body motion between the body and its coordinates system. As a consequence, it is required that the local shape function matrix $\mathbf{S}_l(x)$ contains no rigid body modes. The motion of the flexible body can be described using the floating frame of reference formulation as:

$$\mathbf{r} = \mathbf{R} + \mathbf{A}\mathbf{S}_l q_f \tag{3.88}$$

where the vector q_f describes the deformation, and the vector

$$q_r = \begin{bmatrix} \mathbf{R} \\ \theta \end{bmatrix} \tag{3.89}$$

describes the reference motion. The expression of local shape function is introduced in the last subsection for different elements: elastic bar, Euler-Bernoulli beam, and Timoshenko beam. Also, one can understand how to formulate the local shape function from the global shape function used in the ANCF-S. It is worth noting that the formulation of the floating reference frame approach is highly systematic since the formulation of mass matrix, stiffness matrix, and external force all follow the same derivation. The systematic formulation of floating reference frame approach gives rise to its popularity in the simulation software. Most multibody dynamic simulation programs are governed by the FRF approach. In the case of planar motion, it can be shown that the nonlinear mass matrix and the Coriolis and centrifugal forces of any deformable body or finite element can be expressed in terms of the following constant elementary shape integrals [41][11]:

$$\bar{S} = \int_V \rho \mathbf{S}_l dV, \qquad \mathbf{m}_{ff} = \int_V \rho \mathbf{S}_l^T \mathbf{S}_l dV, \qquad \tilde{S} = \int_V \rho \mathbf{S}_l^T \tilde{I} \mathbf{S}_l dV \tag{3.90}$$

where ρ and V are the mas density and volume of the body, and

$$\tilde{I} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{3.91}$$

Mass Matrix The kinetic energy of an element can be defined as

$$\mathbf{T} = \frac{1}{2} \int_V \rho \dot{r}^T \dot{r} dV \tag{3.92}$$

where $\dot{r}$ is the global velocity vector, which can be evaluated by taking the first derivative with respect to time of the position vector:

$$\dot{r} = \frac{d\mathbf{r}}{dt} = \dot{R} + \frac{\partial \mathbf{A}\mathbf{S}_l q_f}{\partial t} = \begin{bmatrix} \mathbf{I} & \mathbf{B} & \mathbf{A}\mathbf{S}_l \end{bmatrix} \begin{bmatrix} \dot{R} \\ \dot{\theta} \\ \dot{q_f} \end{bmatrix} \tag{3.93}$$

where I is identity matrix and $\mathbf{B}$ is defined as

$$\frac{\partial \mathbf{A}\mathbf{S}_l}{\partial t} = \frac{\partial \mathbf{A}\mathbf{S}_l}{\partial \theta} \frac{\partial \theta}{\partial t} = \mathbf{A}_\theta \mathbf{S}_l \dot{\theta} = \mathbf{B}\dot{\theta} \tag{3.94}$$

For planar dynamic, transformation matrix $\mathbf{A}$ and it's derivative $\mathbf{A}_\theta$ are:

$$\mathbf{A} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}, \qquad \mathbf{A}_\theta = \begin{bmatrix} -\sin(\theta) & -\cos(\theta) \\ \cos(\theta) & -\sin(\theta) \end{bmatrix} \tag{3.95}$$

In short, the global velocity vector can be expressed as

$$\dot{r} = \mathbf{L}\dot{q}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{I} & \mathbf{B} & \mathbf{A}\mathbf{S}_l \end{bmatrix}, \quad \dot{q} = \begin{bmatrix} \dot{R} \\ \dot{\theta} \\ \dot{q}_f \end{bmatrix} \tag{3.96}$$

and the substitution of $\dot{r}$ in kinetic energy yields

$$\mathbf{T} = \frac{1}{2}\int_V \rho \dot{r}^T \dot{r} dV = \frac{1}{2}\dot{q}^T \left[\int_V \rho \mathbf{L}^T \mathbf{L} dV \right] \dot{q} \tag{3.97}$$

The mass matrix is

$$\begin{aligned} \mathbf{M} &= \int_V \rho \mathbf{L}^T \mathbf{L} dV \\ &= \int_V \rho \begin{bmatrix} \mathbf{I} \\ \mathbf{B} \\ \mathbf{A}\mathbf{S}_l \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{B} & \mathbf{A}\mathbf{S}_l \end{bmatrix} dV \\ &= \int_V \begin{bmatrix} \mathbf{I} & \mathbf{B} & \mathbf{A}\mathbf{S}_l \\ & \mathbf{B}^T\mathbf{B} & \mathbf{B}^T\mathbf{A}\mathbf{S}_l \\ sym & & \mathbf{S}_l^T\mathbf{S}_l \end{bmatrix} dV \end{aligned} \tag{3.98}$$

which leads to a nonlinear mass matrix

$$\mathbf{M} = \begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \bar{S} q_f & \mathbf{A}\bar{S} \\ & q_f^T \mathbf{m}_{ff} q_f & q_f^T \tilde{S} \\ sym & & \mathbf{m}_{ff} \end{bmatrix} \tag{3.99}$$

which is nonlinear compared to the inertial reference formulation or absolute nodal coordinate formulation.

Stiffness Matrix It is obvious that the position vector can be separated into two coupled coordinates, namely, the global reference (rigid body motion) and local reference (flexible body deformation). Also, the internal energy can be written as

$$\mathbf{U} = \int_V \sigma \epsilon dV \tag{3.100}$$

while the strain and stress can be represented through the constitutive equation as

$$\begin{aligned} \epsilon &= \mathbf{DS}_l q_f \\ \sigma &= \mathbf{E}\epsilon \end{aligned} \tag{3.101}$$

where **D** is a differential operator and the above equation yields

$$\sigma = \mathbf{ES}_l q_f \tag{3.102}$$

The stress vector is written in terms of the elastic generalized coordinates of the flexible body, such that the internal energy yields

$$\mathbf{U} = q_f^T \left[\int_V (\mathbf{DS}_l)^T \mathbf{E} (\mathbf{DS}_l) \right] q_f^T \tag{3.103}$$

The stiffness matrix is

$$\mathbf{K} = \int_V (\mathbf{DS}_l)^T \mathbf{E} (\mathbf{DS}_l) \tag{3.104}$$

or for convenience, it can be rewritten in global system as

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \mathbf{K}_{ff} \end{bmatrix} \tag{3.105}$$

which is linear compared to the stiffness matrix in IRF or ANCF-S.

Quadratic Velocity Vector For a body in a multibody system, Lagrange's equation takes the form

$$\frac{d}{dt}\left(\frac{\partial \mathbf{T}}{\partial \dot{q}}\right) - \frac{\partial \mathbf{T}}{\partial q} + \mathbf{K}q + \mathbf{C}^T\lambda = \mathbf{F}_{ext} \tag{3.106}$$

where $\mathbf{T}$ is the kinetic energy. Using the general expression of the kinetic energy, we can write the first two terms in the left-hand side as

$$\frac{d}{dt}\left(\frac{\partial \mathbf{T}}{\partial \dot{q}}\right) - \frac{\partial \mathbf{T}}{\partial q} = \mathbf{M}\ddot{q} + \dot{M}\dot{q} - \left[\frac{\partial}{\partial q}\left(\frac{1}{2}\dot{q}^T M\dot{q}\right)\right] \tag{3.107}$$

We may define $\mathbf{Q}_v$ to be

$$\mathbf{Q}_v = -\dot{M}\dot{q} + \left[\frac{\partial}{\partial q}\left(\frac{1}{2}\dot{q}^T M\dot{q}\right)\right] \tag{3.108}$$

The quadratic velocity vector is a result from the differentiation of the kinetic energy with respect to time and respect to the body coordinates. This quadratic velocity vector contains the gyroscopic and Coriolis force components. It can be eliminated as

$$
\begin{aligned}
(\mathbf{Q}_v)_R &= (\dot{\theta})^2\mathbf{A}(\bar{S}\,q_f + \mathbf{I}_1) - 2\theta\mathbf{A}_\theta\bar{S}\,q_f \\
(\mathbf{Q}_v)_\theta &= -2\dot{\theta}\dot{q}_f^T(\mathbf{m}_{ff}q_f + \mathbf{I}_0) \\
(\mathbf{Q}_v)_f &= (\dot{\theta})^2(\mathbf{m}_{ff}q_f + \mathbf{I}_0) + 2\dot{\theta}\bar{S}\,\dot{q}_f
\end{aligned}
\tag{3.109}
$$

and the quadratic velocity vector can be shown as

$$
\mathbf{Q}_v = [(\mathbf{Q}_v)_R \quad (\mathbf{Q}_v)_\theta \quad (\mathbf{Q}_v)_f]^T \tag{3.110}
$$

External Force Because of the the use of body coordinates, it is obvious that the external force on the flexible/rigid body has to be modified in order to obtain a consistent load vector. The virtual work done by external force can be defined as

$$
\delta\mathbf{W}_{ext} = \mathbf{F}_{ext}\delta\mathbf{r}_c \tag{3.111}
$$

where $\mathbf{r}_c$ is the global position vector at the center of mass, which can be written as the sum of two vectors, namely,

$$
\mathbf{r}_c = \mathbf{r}_0 + \mathbf{r}_f(\xi = 0.5) \tag{3.112}
$$

where $\mathbf{r}_0$ and $\mathbf{r}_f(\xi = 0.5)$ are the undeformed position of mass center and deformed vector as shown

$$
\begin{aligned}
\mathbf{r}_0 &= [L/2 \quad 0]^T \\
\mathbf{r}_f &= \mathbf{S}_l(\xi = 0.5)q_f
\end{aligned}
\tag{3.113}
$$

and therefore the virtual displacement is given by

$$\mathbf{r}_c = \begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{r}_c & \mathbf{A}\mathbf{S}_l(\xi = 0.5) \end{bmatrix} \begin{bmatrix} \delta R \\ \delta\theta \\ \delta q_f \end{bmatrix} \tag{3.114}$$

the external force for each mode in the generalized coordinate are respectively written as

$$\begin{aligned} (\mathbf{Q}_e)_R &= f_{ext}^T \mathbf{I} \\ (\mathbf{Q}_e)_\theta &= f_{ext}^T \mathbf{A}_\theta \mathbf{r}_c \\ (\mathbf{Q}_e)_f &= f_{ext}^T \mathbf{A}\mathbf{S}_l(\xi = 0.5) \end{aligned} \tag{3.115}$$

and the external force vector can be shown as

$$\mathbf{Q}_e = [(\mathbf{Q}_e)_R \quad (\mathbf{Q}_e)_\theta \quad (\mathbf{Q}_e)_f]^T \tag{3.116}$$

The above derivation works for all kinds of elements since the reader only has to substitute the corresponding shape function.

[a] Elastic Bar Model

The global shape function from ANCF-S for an elastic bar is

$$\mathbf{S} = \begin{bmatrix} 1-\xi & 0 & \xi & 0 \\ 0 & 1-\xi & 0 & \xi \end{bmatrix} \tag{3.117}$$

corresponding to four global generalized coordinates (see subsection ANCF-S for details)

$$q = [q_1 \quad q_2 \quad q_3 \quad q_4]^T \tag{3.118}$$

Notice that global shape function already contains three rigid body mode ($x - y$ translation and rotation). As a result, the local shape function will only allow one variable left, which is

$$\mathbf{S}_l = \begin{bmatrix} \xi \\ 0 \end{bmatrix} \tag{3.119}$$

subject to also four global coordinates

$$q = [R_1 \quad R_2 \quad \theta \quad q_f]^T \tag{3.120}$$

[b] Euler-Bernoulli Beam

The global shape function from ANCF-S for an Euler-Bernoulli beam is

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 & 0 \\ 0 & s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 \end{bmatrix} \tag{3.121}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - 3\xi^2 + 2\xi^3, \qquad s_2 = L(\xi - 2\xi^2 + \xi^3), \qquad s_3 = 3\xi^2 - 2\xi^3, \qquad s_4 = L(\xi^3 - \xi^2) \tag{3.122}$$

and $\xi = x/L$. This shape function correspond to eight global generalized coordinates (see subsection ANCF for details)

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \tag{3.123}$$

and the local shape function is reduced to

$$\mathbf{S}_l = \begin{bmatrix} s_2 & s_3 & 0 & s_4 & 0 \\ 0 & 0 & s_3 & 0 & s_4 \end{bmatrix} \tag{3.124}$$

also subject to eight global quantities

$$q = [R_1 \quad R_2 \quad \theta \quad q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5]^T \tag{3.125}$$

[c] Timoshenko Beam

The global shape function from ANCF-S for a Timoshenko beam is

$$\mathbf{S}(\xi, \eta) = \frac{1}{1+\varphi} \begin{bmatrix} s_1 & t_1 & s_2 & t_2 & s_3 & t_3 & s_4 & t_4 \\ t_1 & s_1 & t_2 & s_2 & t_3 & s_3 & t_4 & s_4 \end{bmatrix} \tag{3.126}$$

where the functions $s_i = s_i(\xi, \eta)$, and $t_i = t_i(\xi, \eta)$ are defined as

$$\begin{aligned} s_1 &= 1 - 3\xi^2 + 2\xi^3 + (1-\xi)\phi, \qquad s_2 = L[\xi - 2\xi^2 + \xi^3 + \frac{1}{2}(\xi - \xi^2)\phi] \\ s_3 &= 3\xi^2 - 2\xi^3 + \xi\phi, \qquad s_4 = L[\xi^3 - \xi^2 - \frac{1}{2}(\xi - \xi^2)\phi] \\ t_1 &= 6(\xi - \xi^2)\eta, \qquad t_2 = L[-1 + 4\xi - 3\xi^2 - (1-\xi)\phi]\eta \\ t_3 &= 6(-\xi + \xi^2)\eta, \qquad t_4 = L[2\xi - 3\xi^2 - \xi\phi]\eta \end{aligned} \tag{3.127}$$

and $\xi = x/L$, $\eta = y/L$. This shape function corresponds to eight global generalized coordinates

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \tag{3.128}$$

The local shape function is reduced to

$$\mathbf{S}_l(\xi, \eta) = \frac{1}{1+\phi} \begin{bmatrix} s_2 & s_3 & t_3 & s_4 & t_4 \\ t_2 & t_3 & s_3 & t_4 & s_4 \end{bmatrix} \tag{3.129}$$

and is also subject to eight global quantities

$$q = [R_1 \quad R_2 \quad \theta \quad q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5]^T \tag{3.130}$$

ϕ is defined as

$$\phi = \frac{12EI}{\kappa GAL^2} \tag{3.131}$$

3.6 GSSSS for Constrained Dynamics for Index 3/2/1

Since the equations of motion and constraint equations are formulated through IRF/ANCF-S/FRF, solving a DAE system by employing the GSSSS family of algorithms for constrained dynamics will be introduced next. Also, notice that in ODE, the equations of motion at the algorithm time level (t_{n+W_1} for GSSSS) are solved. However, in DAE system, two coupled equations are solved simultaneously (EOM and constraints), and the time level issues are different for these two equations. Ref. [55] notes that both are computed at the algorithm time level for providing robustness and stability. However, in physics, constraint satisfaction is required at time level t_{n+1}. That is to say, the equations of motion are still computed at t_{n+W_1}; the constraint

equations, however, are computed at time level t_{n+1} in order to satisfy constraint exactly. For each Index 3/2/1, the GSSSS family of algorithms for constrained dynamics is as follows.

3.6.1 Index 3 Form: Unified Predictor Multi-Corrector Representation

The effective DAE employing the normalized time weighted residual approach is given by

$$\begin{aligned} &\mathbf{M}\tilde{a} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} = \tilde{\boldsymbol{F}}_{ext} \\ &\chi_{Cu}\boldsymbol{\Phi}(u_{n+1}) = 0 \end{aligned} \tag{3.132}$$

Employ the Newton-Raphson method to iteratively solve for the nonlinear effective DAE above: At the beginning of time step, predict the state vectors

$$\begin{aligned} u_{n+1}^k &= \chi_{Pu}^{(1)}u_n + \chi_{Pu}^{(2)}v_n + \chi_{Pu}^{(3)}a_n \\ \tilde{u}_{n+1}^k &= \chi_{P\tilde{u}}^{(1)}u_n + \chi_{P\tilde{u}}^{(2)}v_n + \chi_{P\tilde{u}}^{(3)}a_n \\ \tilde{v}_{n+1}^k &= \chi_{P\tilde{v}}^{(1)}u_n + \chi_{P\tilde{v}}^{(2)}v_n + \chi_{P\tilde{v}}^{(3)}a_n \\ \tilde{a}_{n+1}^k &= \chi_{P\tilde{a}}^{(1)}u_n + \chi_{P\tilde{a}}^{(2)}v_n + \chi_{P\tilde{a}}^{(3)}a_n \\ \tilde{\lambda}_{n+1}^k &= \lambda_n \end{aligned} \tag{3.133}$$

Start nonlinear iteration. Solve for $\Delta\delta_{n+1}^{k+1}$ and $\Delta\lambda_{n+1}^{k+1}$ from

$$\mathbf{J}\begin{bmatrix} \Delta\delta_{n+1}^{k+1} \\ \Delta\lambda_{n+1}^{k+1} \end{bmatrix} = -\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \tag{3.134}$$

where

$$
\begin{aligned}
R_1 &= \mathbf{M}\tilde{a} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} - \tilde{\boldsymbol{F}}_{ext} \\
R_2 &= \chi_{Cu}\boldsymbol{\Phi}(u_{n+1}) = 0
\end{aligned}
\tag{3.135}
$$

and

$$
\mathbf{J} = \begin{bmatrix} \frac{\partial R_1}{\partial q^*_{n+1}} & \frac{\partial R_1}{\partial \lambda_{n+1}} \\ \mathbf{C}(u_{n+1}) & 0 \end{bmatrix}
\tag{3.136}
$$

with $q^*_{n+1} = u_{n+1}, v_{n+1}, a_{n+1}$ for d-, v-, or a-form, respectively.

Then correct the primary variables as follows:

$$
\begin{aligned}
u^{k+1}_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi_{Cu}\Delta\delta^{k+1}_{n+1} \\
\tilde{u}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi_{C\tilde{u}}\Delta\delta^{k+1}_{n+1} \\
\tilde{v}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi_{C\tilde{v}}\Delta\delta^{k+1}_{n+1} \\
\tilde{a}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi_{C\tilde{a}}\Delta\delta^{k+1}_{n+1} \\
\lambda^{k+1}_{n+1} &= \lambda^{k}_{n+1} + \Delta\lambda^{k+1}_{n+1}
\end{aligned}
\tag{3.137}
$$

Repeat nonlinear iterations until the solution converges. Once a converged solution is found, update remaining variables as:

$$
\begin{aligned}
a_{n+1} &= a_n + (\tilde{a}^{k+1}_{n+1} - a_n)/\Lambda_6 W_1 \\
v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 (a_{n+1} - a_n)\Delta t \\
u_{n+1} &= u_n + \lambda_1 v \Delta t + \lambda_2 a_n \Delta t + \lambda_3 (a_{n+1} - a_n)\Delta t^2
\end{aligned}
\tag{3.138}
$$

3.6.2 Index 2 Form: Unified Predictor Multi-Corrector Representation

The effective DAE employing the normalized time weighted residual approach is given by

$$\begin{aligned} &\mathbf{M}\tilde{\ddot{u}} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} = \tilde{\boldsymbol{F}}_{ext} \\ &\chi_{Cv}\dot{\boldsymbol{\Phi}}(u_{n+1}, v_{n+1}) = 0 \end{aligned} \tag{3.139}$$

Employ the Newton-Raphson method to iteratively solve for the nonlinear effective DAE above: At the beginning of time step, predict the state vectors

$$\begin{aligned} u^k_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi^{(2)}_{Pu} v_n + \chi^{(3)}_{Pu} a_n \\ \tilde{u}^k_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi^{(2)}_{P\tilde{u}} v_n + \chi^{(3)}_{P\tilde{u}} a_n \\ \tilde{v}^k_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi^{(2)}_{P\tilde{v}} v_n + \chi^{(3)}_{P\tilde{v}} a_n \\ \tilde{a}^k_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi^{(2)}_{P\tilde{a}} v_n + \chi^{(3)}_{P\tilde{a}} a_n \\ \tilde{\lambda}^k_{n+1} &= \lambda_n \end{aligned} \tag{3.140}$$

Start nonlinear iteration. Solve for $\Delta\delta^{k+1}_{n+1}$ and $\Delta\lambda^{k+1}_{n+1}$ from

$$\mathbf{J}\begin{bmatrix} \Delta\delta^{k+1}_{n+1} \\ \Delta\lambda^{k+1}_{n+1} \end{bmatrix} = -\begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \tag{3.141}$$

where

$$\begin{aligned} R_1 &= \mathbf{M}\tilde{a} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} - \tilde{\boldsymbol{F}}_{ext} \\ R_2 &= \chi_{Cv}\dot{\boldsymbol{\Phi}}(u_{n+1}, v_{n+1}) \end{aligned} \tag{3.142}$$

and

$$\mathbf{J} = \begin{bmatrix} \frac{\partial R_1}{\partial q^*_{n+1}} & \frac{\partial R_1}{\partial \lambda_{n+1}} \\ \boldsymbol{C}(u_{n+1}) & 0 \end{bmatrix} \tag{3.143}$$

with $q^*_{n+1} = u_{n+1}, v_{n+1}, a_{n+1}$ for d-, v-, or a-form, respectively.

Then correct the primary variables as follows:

$$\begin{aligned} u^{k+1}_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi_{Cu} \Delta\delta^{k+1}_{n+1} \\ \tilde{u}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi_{C\tilde{u}} \Delta\delta^{k+1}_{n+1} \\ \tilde{v}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi_{C\tilde{v}} \Delta\delta^{k+1}_{n+1} \\ \tilde{a}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi_{C\tilde{a}} \Delta\delta^{k+1}_{n+1} \\ \lambda^{k+1}_{n+1} &= \lambda^{k}_{n+1} + \Delta\lambda^{k+1}_{n+1} \end{aligned} \tag{3.144}$$

Repeat nonlinear iterations until the solution converges. Once a converged solution is found, update remaining variables as:

$$\begin{aligned} a_{n+1} &= a_n + (\tilde{a}^{k+1}_{n+1} - a_n)/\Lambda_6 W_1 \\ v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 (a_{n+1} - a_n)\Delta t \\ u_{n+1} &= u_n + \lambda_1 v \Delta t + \lambda_2 a_n \Delta t + \lambda_3 (a_{n+1} - a_n)\Delta t^2 \end{aligned} \tag{3.145}$$

3.6.3 Index 1 Form: Unified Predictor Multi-Corrector Representation

The effective DAE employing the normalized time weighted residual approach is given by

$$\begin{aligned} \mathbf{M}\tilde{a} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} &= \tilde{\boldsymbol{F}}_{ext} \\ \chi_{Ca}\ddot{\boldsymbol{\Phi}}(u_{n+1}, v_{n+1}, a^*_{n+1}) &= 0 \end{aligned} \tag{3.146}$$

It is worth noting that the "true acceleration time level a^*_{n+1}" is not the traditional acceleration time level $\ddot{u}^*_{n+1-\gamma}$ we used in the past; instead, one has to shift the acceleration from $t_{n+1-\gamma}$ to t_{n+1} at time step n by

$$\begin{aligned} a^*_{n+1} &= a_0 + \frac{1}{(1-\gamma)}(a_1 - a_0), \quad for \quad n = 0 \\ a^*_{n+1} &= a_n + (1+\gamma)(a_{n+1} - a_n), \quad for \quad n \geq 1 \end{aligned} \tag{3.147}$$

where, the shift γ is defined by

$$\gamma = \Lambda_6 W_1 - W_1 \tag{3.148}$$

In this way, not only constraint satisfaction at time level t_{n+1} is satisfied, but also one can achieve second order time accuracy. Employ the Newton-Raphson method to iteratively solve for the nonlinear effective DAE above:

At the beginning of time step, predict the state vectors

$$\begin{aligned} u^k_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi^{(2)}_{Pu} \dot{u}_n + \chi^{(3)}_{Pu} \ddot{u}_n \\ \tilde{u}^k_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi^{(2)}_{P\tilde{u}} v_n + \chi^{(3)}_{P\tilde{u}} a_n \\ \tilde{v}^k_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi^{(2)}_{P\tilde{v}} v_n + \chi^{(3)}_{P\tilde{v}} a_n \\ \tilde{a}^k_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi^{(2)}_{P\tilde{a}} v_n + \chi^{(3)}_{P\tilde{a}} a_n \\ \tilde{\lambda}^k_{n+1} &= \lambda_n \end{aligned} \tag{3.149}$$

Start nonlinear iteration. Solve for $\Delta\delta^{k+1}_{n+1}$ and $\Delta\lambda^{k+1}_{n+1}$ from

$$\mathbf{J} \begin{bmatrix} \Delta\delta^{k+1}_{n+1} \\ \Delta\lambda^{k+1}_{n+1} \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \tag{3.150}$$

where

$$
\begin{aligned}
R_1 &= \mathbf{M}\tilde{a} + \mathbf{F}_{int}(\tilde{u}) + \mathbf{C}^T(\tilde{u})\tilde{\lambda} - \tilde{\boldsymbol{F}}_{ext} \\
R_2 &= \chi_{Ca}\dot{\boldsymbol{\Phi}}(u_{n+1}, v_{n+1}, a^*_{n+1})
\end{aligned}
\tag{3.151}
$$

Due to the shift for the true acceleration a^*_{n+1}, the Jacobian has to be modified at first time step ($n = 0$) as

$$
\mathbf{J} = \begin{bmatrix} \frac{\partial R_1}{\partial q^*_{n+1}} & \frac{\partial R_1}{\partial \lambda_{n+1}} \\ \frac{1}{1-\gamma}\boldsymbol{C}(u_{n+1}) & 0 \end{bmatrix}
\tag{3.152}
$$

and time step $n \geq 1$ as

$$
\mathbf{J} = \begin{bmatrix} \frac{\partial R_1}{\partial q^*_{n+1}} & \frac{\partial R_1}{\partial \lambda_{n+1}} \\ (1+\gamma)\boldsymbol{C}(u_{n+1}) & 0 \end{bmatrix}
\tag{3.153}
$$

with $q^*_{n+1} = u_{n+1}, \dot{u}_{n+1}, \ddot{u}_{n+1}$ for d-, v-, or a-form, respectively.

Then correct the primary variables as follows:

$$
\begin{aligned}
u^{k+1}_{n+1} &= \chi^{(1)}_{Pu} u_n + \chi_{Cu}\Delta\delta^{k+1}_{n+1} \\
\tilde{u}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{u}} u_n + \chi_{C\tilde{u}}\Delta\delta^{k+1}_{n+1} \\
\tilde{v}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{v}} u_n + \chi_{C\tilde{v}}\Delta\delta^{k+1}_{n+1} \\
\tilde{a}^{k+1}_{n+1} &= \chi^{(1)}_{P\tilde{a}} u_n + \chi_{C\tilde{a}}\Delta\delta^{k+1}_{n+1} \\
\lambda^{k+1}_{n+1} &= \lambda^{k}_{n+1} + \Delta\lambda^{k+1}_{n+1}
\end{aligned}
\tag{3.154}
$$

Repeat nonlinear iterations until the solution converges. Once a converged solution is found, update remaining variables as:

$$
\begin{aligned}
a_{n+1} &= a_n + (\tilde{a}_{n+1}^{k+1} - a_n)/\Lambda_6 W_1 \\
v_{n+1} &= v_n + \lambda_4 a_n \Delta t + \lambda_5 (a_{n+1} - a_n)\Delta t \\
u_{n+1} &= u_n + \lambda_1 v \Delta t + \lambda_2 a_n \Delta t + \lambda_3 (a_{n+1} - a_n)\Delta t^2
\end{aligned}
\tag{3.155}
$$

The predictor-corrector coefficients χ above in the corresponding a, v, and d-form are listed in Table 2.2.

3.6.4 Time Level Correction of Lagrange Multiplier

Notice that the scalar λ_i values in the updates are the algorithm specific parameters resulting from the GSSSS framework and are in no way associated with the bold vector of Lagrange multiplier $\tilde{\lambda}_{n+1}^{k+1}$.

As a result of previous work, it is known that the value of acceleration that results from the algorithm is not always at time t_{n+1}. To obtain second order accurate accelerations the time level of the acceleration must be accounted for. Similarly, we do the same for the time level of the computed Lagrange multipliers in the algorithm is t_{n+W_1}. In order to obtain a consistent convergence plot, the Lagrange multipliers at the final time step may be interpolated forward in time to obtain values at t_{n+1} as follows:

$$
\lambda_{n+1} = (-1 + W_1)\lambda_n + (2 - W_1)\tilde{\lambda}_{n+1}^{k+1} \tag{3.156}
$$

3.7 Stabilization Techniques

During the integration process, the Lagrangian multiplier formulation yields a solution set of u, $\dot{u}$, and $\ddot{u}$ that does not enforce each individual constraint to be exactly equal to zero. In [24], Burgermeister discussed several ways to extend the linear-implicit method to a non-iterative method for constrained system and proved that the error in the constraints $\Phi(u)$ is bounded by $\|\Phi(u)\| \leq Ch^r$ with $r \geq 2$ and h is time step size, Constant C is independent of the length of the time interval.

3.7.1 Baumgarte Stabilization

The first method for stabilization of the constraints was proposed by Baumgarte [21]; he substituted the constraints on acceleration level by a linear combination of each level constraint (position, velocity, acceleration constraint):

$$0 = \ddot{\mathbf{\Phi}} + 2\alpha\dot{\mathbf{\Phi}} + \beta\mathbf{\Phi} \tag{3.157}$$

The scalar constants α and β are called *Baumgarte parameters*. These parameters have to be chosen so that the scalar differential equation $0 = \ddot{y} + 2\alpha\dot{y} + \beta y$ has an asymptotically stable solution $y(t) = c_1 e^{\tau_1 t} + c_2 e^{\tau_2 t}$.

The quality of the stabilization depends on the parameters α and β. If we choose them too small, the stabilization is not sufficient. Too large parameters introduce additional stiffness to the system and make the numerical integration inefficient and sensitive to round-off errors [62]. To find the optimal parameters, [24] shows the process for the constraint and make use of truncation errors analysis. In general to avoid problems with the additional stiffness, the Baumgarte

parameters are usually chosen smaller: $0 < \alpha < 1/h$ and $0 < \beta < 1/h^2$ [24]. As a result, the DAEs system under GSSSS family of algorithms can be written as

$$\begin{aligned} \mathbf{M}a + \mathbf{F}_{int} + \mathbf{C}^T\lambda &= \mathbf{F}_{ext} \\ \ddot{\mathbf{\Phi}} + 2\alpha\dot{\mathbf{\Phi}} + \beta\mathbf{\Phi} &= 0 \end{aligned} \tag{3.158}$$

Applying the GSSSS framework, the discretization of equation of motion and constraint equation are

$$\begin{aligned} \mathbf{M}\tilde{a} + \tilde{\boldsymbol{F}}_{int} + \mathbf{C}(\tilde{u})\tilde{\lambda} &= \tilde{\boldsymbol{F}}_{ext} \\ \chi_{Ca}\ddot{\mathbf{\Phi}} + 2\alpha^*\chi_{Cv}\dot{\mathbf{\Phi}} + \beta^*\chi_{Cu}\mathbf{\Phi} \end{aligned} \tag{3.159}$$

This way, new optimal Baumgarte stabilization parameters are chosen to be: $0 < \alpha^* < 1$ and $0 < \beta^* < 1$ for convenience.

3.7.2 Mass-Orthogonal Projection Method

In order to achieve exact constraint satisfaction at each time step we propose a mass-orthogonal projection that takes the solution to the constrain manifold where $\mathbf{\Phi}$, $\dot{\mathbf{\Phi}}$, $\ddot{\mathbf{\Phi}}$ are all equal to zero. In essence, our objective is to force the set of DAE to meet the underlying ODE at each time step. It is important to note that in doing so, one eliminates the need of further stabilization such as the method of Baumgarte.

Projection in position

During the time integration process the numerical integration scheme yields a set of coordinates u^* that does not completely satisfy the constraint coordinates $\Phi = 0$. In order to satisfy the constraints, we perform a mass-orthogonal projection of the solution to the displacement constraint manifold and obtain a new set of position u that satisfy $\Phi = 0$. This can be enforced by the solution of the following constrained minimization problem

$$\begin{aligned} min_u \mathbf{V} &= \frac{1}{2}(u - u^*)\mathbf{M}(u - u^*) \\ \mathbf{\Phi}(u, t) &= 0 \end{aligned} \tag{3.160}$$

It can be seen how the mass matrix $\mathbf{M}$ has been intentionally added to the objective function with the idea that the resulting equations will have the same leading matrix as that used in the equation of motion. In what follows and for the sake of clarity we will assume that the mass matrix is constant. The ensuing procedure is perfectly valid for a non-constant mass matrix.

In order to solve the problem posed by Eq. (3.160), we again use the Lagrange multiplier method and minimize the following equation

$$\mathbf{V}^* = \frac{1}{2}(u - u^*)^T\mathbf{M}(u - u^*) + \mathbf{\Phi}^T\sigma \tag{3.161}$$

where σ is a new set of Lagrange multiplier to satisfy minimization of constraint. We now differentiate $\mathbf{V}^*$ with respect to u and equal to zero, yielding

$$\mathbf{H}(u, t) = \frac{\partial \mathbf{V}^*}{\partial u} = \mathbf{M}(u - u^*) + \mathbf{C}^T\sigma = 0 \tag{3.162}$$

where $\mathbf{H}(u,t)$ is the vector of n nonlinear functions, and Eq. 3.160 constitutes a system of nonlinear algebraic equations that can be also solved by Newton-Raphson iteration as

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}^T \\ \mathbf{C}^T & 0 \end{bmatrix} \begin{Bmatrix} \Delta u \\ \Delta\sigma \end{Bmatrix}^{k+1} = - \begin{Bmatrix} \mathbf{M}(u-u^*) + \mathbf{C}^T\sigma \\ \mathbf{\Phi}(u,t) \end{Bmatrix}^{k} \tag{3.163}$$

and update the displacement and Lagrange multiplier with

$$\begin{aligned} u^{k+1} &= u^k + \Delta u^{k+1} \\ \sigma^{k+1} &= \sigma^k + \Delta\sigma^{k+1} \end{aligned} \tag{3.164}$$

where the recursion is started by setting

$$\begin{aligned} u^k &= u^* \\ \sigma^k &= \mathbf{\Phi}^k \end{aligned} \tag{3.165}$$

Projection in velocity

Similarly during the time integration process the numerical integration scheme yields a set of coordinates v^* that does not completely satisfy the constraint coordinates $\dot{\mathbf{\Phi}} = 0$. Again, we perform a mass-orthogonal projection of the solution to the velocity constrain manifold and obtain a new set of velocity v that satisfy $\dot{\Phi} = 0$. This can be enforced by the solution of the following constrained minimization problem

$$\begin{aligned} min_v \mathbf{V} &= \frac{1}{2}(v-v^*)^T\mathbf{M}(v-v^*) \\ \dot{\mathbf{\Phi}}(u,v,t) &= 0 \end{aligned} \tag{3.166}$$

In order to solve the problem posed by equation Eq. (3.166), we again use Lagrange multiplier method and minimize the following equation

$$\mathbf{V} = \frac{1}{2}(v - v^*)\mathbf{M}(v - v^*) + \dot{\mathbf{\Phi}}^T \sigma \tag{3.167}$$

where σ is a new set of Lagrange multiplier to satisfy minimization of constraint. We now differentiate $\mathbf{V}^*$ with respect to u and equal to zero, yielding

$$\mathbf{H}(u, v, t) = \frac{\partial \mathbf{V}^*}{\partial v} = \mathbf{M}(v - v^*) + \mathbf{C}^T \sigma = 0 \tag{3.168}$$

and it can also be solved by Newton-Raphson iteration as:

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}^T \\ \mathbf{C}^T & 0 \end{bmatrix} \begin{Bmatrix} \Delta v \\ \Delta \sigma \end{Bmatrix}^{k+1} = - \begin{Bmatrix} \mathbf{M}(v - v^*) + \mathbf{C}^T \sigma \\ \dot{\mathbf{\Phi}}(u, v, t) \end{Bmatrix}^k \tag{3.169}$$

and update the displacement and Lagrange multiplier with

$$\begin{aligned} v^{k+1} &= v^k + \Delta v^{k+1} \\ \sigma^{k+1} &= \sigma^k + \Delta \sigma^{k+1} \end{aligned} \tag{3.170}$$

where the recursion is started by setting

$$\begin{aligned} v^k &= v^* \\ \sigma^k &= \dot{\mathbf{\Phi}}^k \end{aligned} \tag{3.171}$$

Since we are solving a set of nonlinear algebraic equations, the Newton-Raphson converges rapidly.

Projection in acceleration

Following the same procedure as in the velocity analysis, the projection of the generalized accelerations onto the acceleration constraint manifold can be obtained through the following constrained minimization problem:

$$\begin{aligned} min_a \mathbf{V} &= \frac{1}{2}(a - a^*)^T \mathbf{M}(a - a^*) \\ \ddot{\mathbf{\Phi}}(u, v, a, t) &= 0 \end{aligned} \tag{3.172}$$

Again, it has to be emphasized that the acceleration term a is "true acceleration" at time level t_{n+1}. The Lagrange multiplier formulation of the above minimization problem is established by defining a modified objective function:

$$\mathbf{V} = \frac{1}{2}(a - a^*)\mathbf{M}(a - a^*) + \ddot{\mathbf{\Phi}}^T \sigma \tag{3.173}$$

where σ is a new set of Lagrange multiplier to satisfy minimization of constraint. We now differentiate $\mathbf{V}^*$ with respect to generalized accelerations to obtain

$$\mathbf{H}(u, a, t) = \frac{\partial \mathbf{V}^*}{\partial a} = \mathbf{M}(a - a^*) + \mathbf{C}^T \sigma = 0 \tag{3.174}$$

and it can also be solved by Newton-Raphson iteration as:

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}^T \\ \mathbf{C}^T & 0 \end{bmatrix} \begin{Bmatrix} \Delta a \\ \Delta \sigma \end{Bmatrix}^{k+1} = - \begin{Bmatrix} \mathbf{M}(a - a^*) + \mathbf{C}^T \sigma \\ \ddot{\mathbf{\Phi}}(u, v, a, t) \end{Bmatrix}^k \tag{3.175}$$

and update the displacement and Lagrange multiplier with

$$\begin{aligned} a^{k+1} &= a^k + \Delta a^{k+1} \\ \sigma^{k+1} &= \sigma^k + \Delta\sigma^{k+1} \end{aligned} \tag{3.176}$$

where the recursion is started by setting

$$\begin{aligned} a^k &= a^* \\ \sigma^k &= \ddot{\mathbf{\Phi}}^k \end{aligned} \tag{3.177}$$

Eq. (3.175) shows a system of linear algebraic equations in a that are solved repeatedly until convergence in the generalized accelerations is achieved.

3.8 Numerical Procedure of Multibody Dynamics

In this section, the numerical algorithms for the solution of coupled set of differential and algebraic constraint equations that described the dynamics of constrained mechanical systems are discussed. This will help beginners in this field to learn and implement multibody dynamic simulation in proper manner. Below two flow diagrams are shown for multibody simulations with and without projection method. Notice that the flow diagram for Baumgarte's method is exactly the same as the flow diagram without the projection method.

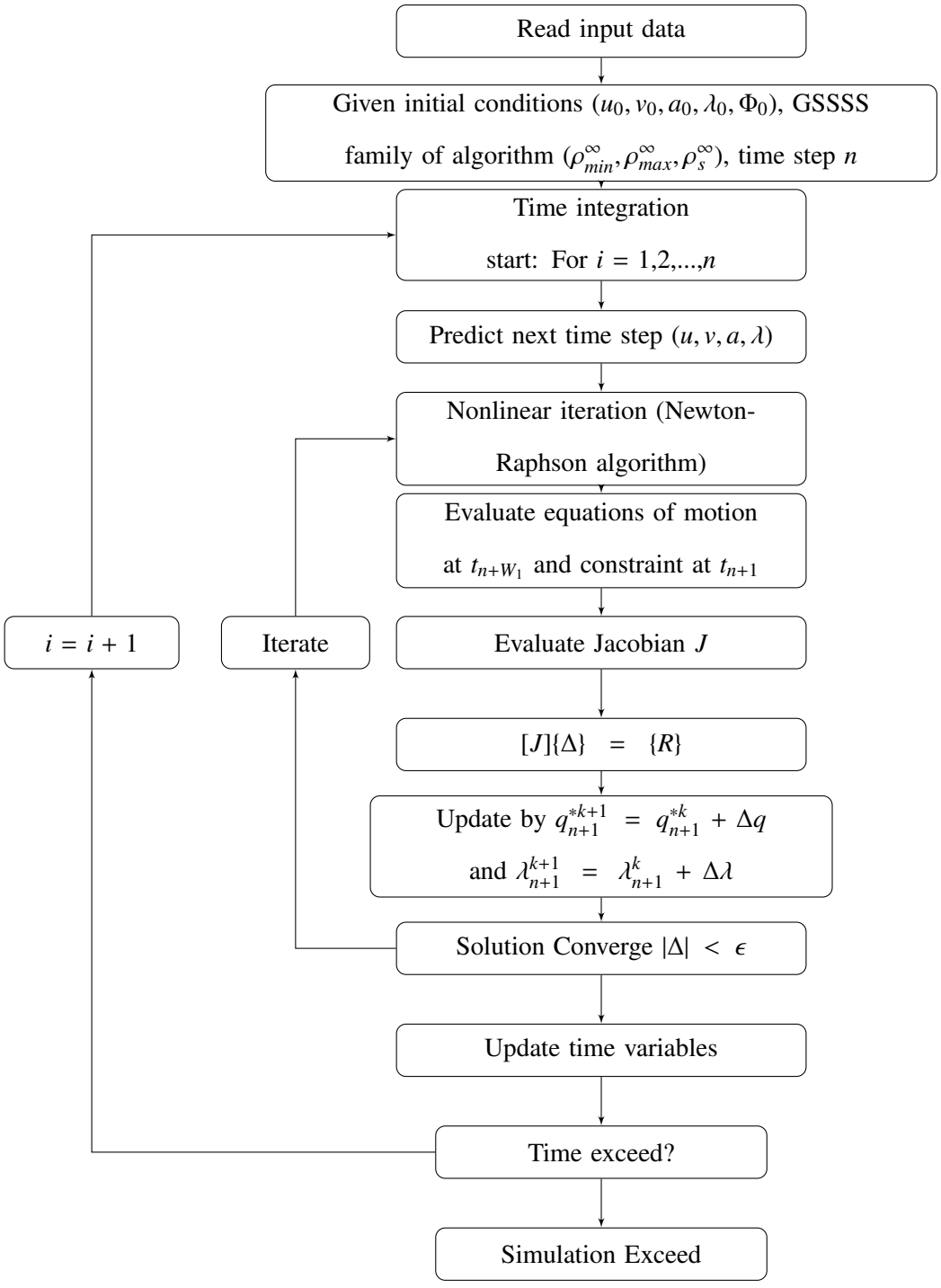


Figure 3.12: Flow diagram of multibody system without projection method.

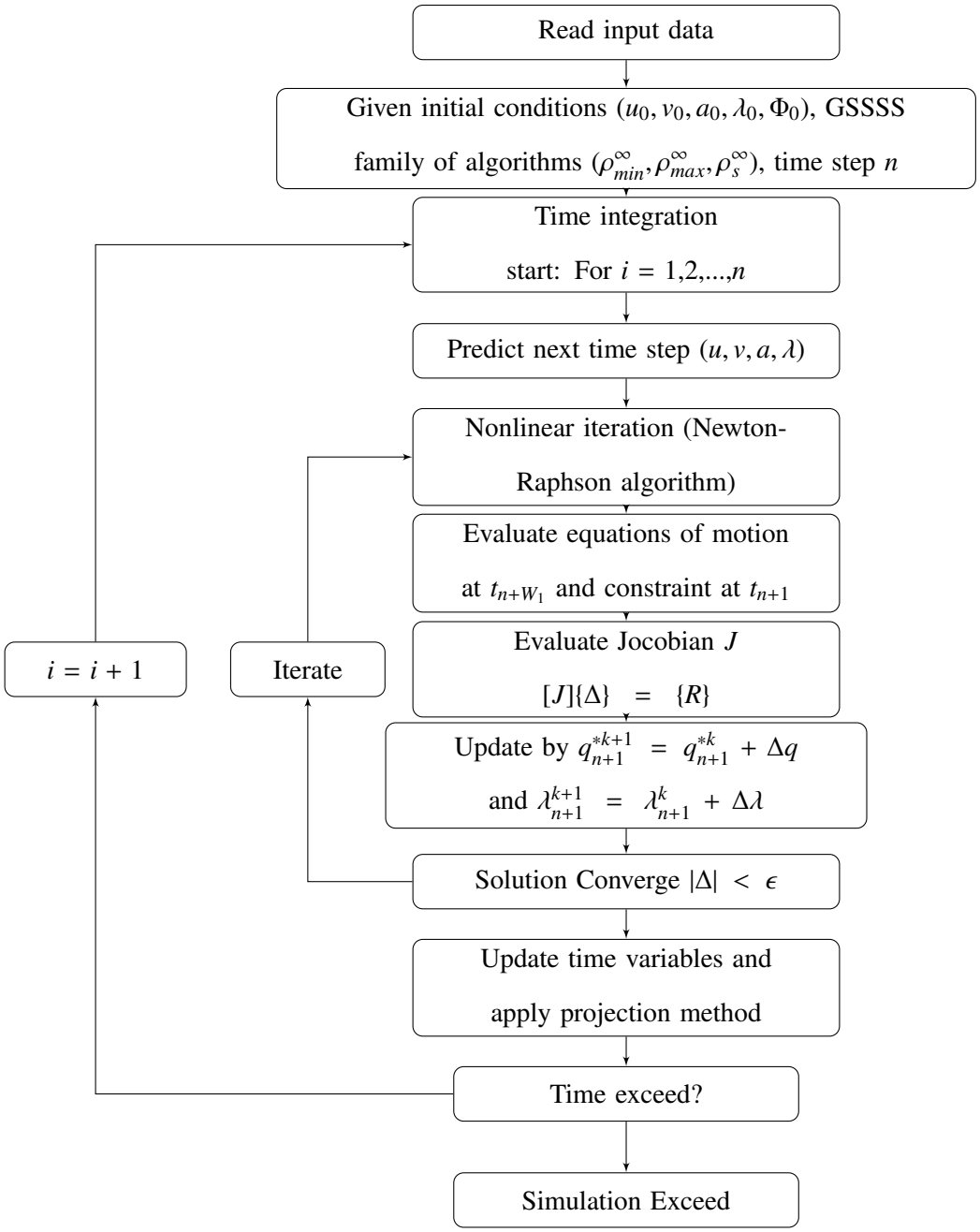


Figure 3.13: Flow diagram of multibody system with projection method.

Chapter 4

Numerical Examples

In this research, numerical examples for both rigid and rigid/flexible body dynamic will be shown. Dynamic responses with and without stabilization for three non-dissipative schemes, namely, the Newmark U0(1,1,0), Midpoint rule with end point acceleration U0V0(1,1,1), and Midpoint rule with midpoint acceleration V0(1,1,0).

4.1 Rigid Body Dynamics

4.1.1 Single Pendulum

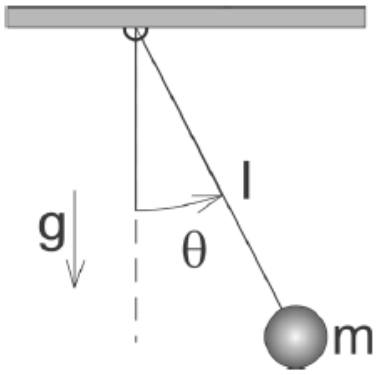


Figure 4.1: Single pendulum problem from[53].

Demonstration of the rigid body dynamics with and without stabilization techniques is first shown for the rigid single pendulum shown in Fig. 4.1. The model consists of one planar rigid bar, with a point mass at its end, pinned to the ground at origin, Gravity is the only external force acting on the system. Redundant coordinates (x, y, θ) are used for simplicity to track the location of each point mass during the simulation. This problem is discussed in great detail in [56], with pertinent information repeated below.

In general the problem is of the form:

$$\begin{aligned} \mathbf{M}a + \mathbf{C}^T(u)\lambda &= \mathbf{F} \\ \mathbf{\Phi}(u) &= 0 \end{aligned} \tag{4.1}$$

The problem consists of three degrees of freedom and two Lagrange multiplier. That is, the unknowns are:

$$
\begin{aligned}
u &= [x_1 \quad y_1 \quad \theta_1]^T \\
v &= [\dot{x}_1 \quad \ddot{y}_1 \quad \dot{\theta}_1]^T \\
a &= [\ddot{x}_1 \quad \ddot{y}_1 \quad \ddot{\theta}_1]^T \\
\lambda &= [\lambda_1 \quad \lambda_2]^T
\end{aligned}
\tag{4.2}
$$

Two constraint equations serve to keep the length of bar constant (geometry constraint):

$$
\mathbf{\Phi}(u) = \begin{bmatrix} x_1 - L_1 sin(\theta_1) \\ y_1 + L_1 cos(\theta_1) \end{bmatrix}
\tag{4.3}
$$

The first derivative of geometry constraint yields velocity constraint:

$$
\dot{\mathbf{\Phi}}(u, v) = \begin{bmatrix} \dot{x}_1 - L_1 cos(\theta_1)\dot{\theta}_1 \\ \dot{y}_1 - L_1 sin(\theta_1)\dot{\theta}_1 \end{bmatrix}
\tag{4.4}
$$

and the acceleration constraint is:

$$
\ddot{\mathbf{\Phi}}(u, v, a) = \begin{bmatrix} \ddot{x}_1 + L_1 sin(\theta_1)\dot{\theta}_1^2 - L_1 cos(\theta_1)\ddot{\theta}_1 \\ \ddot{y}_1 - L_1 cos(\theta_1)\dot{\theta}_1^2 - L_1 sin(\theta_1)\ddot{\theta}_1 \end{bmatrix}
\tag{4.5}
$$

The corresponding constraint matrix C follows as:

$$
\mathbf{C}(u) = \frac{\partial \mathbf{\Phi}(u)}{\partial u} = \begin{bmatrix} 1 & 0 & -L_1 cos(\theta_1) \\ 0 & 1 & -L_1 sin(\theta_1) \end{bmatrix}
\tag{4.6}
$$

As the θ_1 degree of freedom is redundant and used simply to avoid respective calls to tan^{-1}, the potential energy (PE), kinetic energy (KE) the mass matrix and external force use only the Cartesian coordinates in x and y as below.

$$\begin{aligned}
\mathbf{KE} &= \frac{1}{2} v^T \mathbf{M} v \\
\mathbf{PE} &= \mathbf{F}^T u \\
\mathbf{M} &= diag([M_1 \quad M_1 \quad 0]) \\
\mathbf{F} &= [0 \quad -M_1 g \quad 0]
\end{aligned} \tag{4.7}$$

In all simulations below the problem parameters are taken as $M_1 = 1$, $L_1 = 1$, and the gravitational constant is rounded to $g = 9.81$. The initial conditions are taken as follows:

$$\begin{aligned}
u_0 &= [L_1 \quad 0 \quad \frac{\pi}{2}]^T \\
v_0 &= [0 \quad 0 \quad 0]^T
\end{aligned} \tag{4.8}$$

The Jacobian matrix under Index 1, 2, and 3 for the single field form follow as:

Index 3:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.9}$$

Index 2:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.10}$$

Index 1:

for first time step ($n = 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \frac{1}{1-\gamma} \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.11}$$

and the next every other time step ($n > 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ (1+\gamma)\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.12}$$

where γ is the time shift:

$$\gamma = \Lambda_6 W_1 - W_1 \tag{4.13}$$

and $\mathbf{J}_{C\lambda}(u, \lambda)$ is a 3×3 matrix of zeros with entries:

$$\mathbf{J}_{C\lambda}(3,3) = L_1 \sin\theta_1 \lambda_1 - L_1 \cos\theta_1 \lambda_2 \tag{4.14}$$

Post Processing The stabilization techniques (Baumgarte method and projection method) will be applied to satisfy the constraints. All simulation parameters are kept identical to the single field form simulation above. The nonlinear Newton-Raphson residual vector and Jacobian matrices in a-form representation of the single field form, from Eq. (3.159) are:

$$\begin{aligned} R_1 &= \mathbf{M}\tilde{a} + \tilde{\boldsymbol{F}}_{int} + \mathbf{C}(\tilde{u})\tilde{\lambda} - \tilde{\boldsymbol{F}}_{ext} \\ R_2 &= \chi_{Ca}\ddot{\Phi} + 2\alpha^* \chi_{Cv}\dot{\Phi} + \beta^* \chi_{Cu}\Phi \end{aligned} \tag{4.15}$$

and in the nonlinear Newton-Raphson iteration, it is solved iteratively by

$$\mathbf{J} \begin{bmatrix} \Delta\ddot{u}_{n+1}^{k+1} \\ \Delta\lambda_{n+1}^{k+1} \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \tag{4.16}$$

and the modified Jacobian matrix follows:

for first time step ($n = 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (1 + \alpha + 2\beta) \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & (1 + \alpha + 2\beta) \mathbf{C}(\tilde{u})^T \\ (\alpha + 2\beta + \frac{1}{1-\gamma}) \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.17}$$

and the next every other time steps ($n > 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (1 + \alpha + 2\beta) \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & (1 + \alpha + 2\beta) \mathbf{C}(\tilde{u})^T \\ (\alpha + 2\beta + (1 + \gamma)) \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.18}$$

where γ is the time shift defined by

$$\gamma = \Lambda_6 W_1 - W_1 \tag{4.19}$$

Note that the Jabobian matrix almost remains the same except for $\mathbf{C}^T(u_{n+1})$. For the optimal quality of constraint satisfaction, α^* and β^* are chosen to be 1.

Although the projection method in each level constraint has to be computed repeatedly until converge, practical experience shows, that for real-time applications only one step of the Newton iteration for the projection of the position coordinates is sufficient for stabilization. The rigorous proof that this strategy guarantees that the error in the constraints remains bounded for arbitrary time intervals appears in [24]. To calculate the propagation of the error in the position constraints we have to examine the following steps:

$$\begin{aligned} (u_n, v_n, a_n) &\xrightarrow{integrator} (u^*_{n+1}, v^*_{n+1}, a^*_{n+1}) \xrightarrow{u-projection} (u_{n+1}, v^*_{n+1}, a^*_{n+1}) \\ &\xrightarrow{v-projection} (u_{n+1}, v_{n+1}, a^*_{n+1}) \xrightarrow{a-projection} (u_{n+1}, v_{n+1}, a_{n+1}) \end{aligned} \tag{4.20}$$

Any inverse way of projection can only satisfy the new constraint manifold. In the nonlinear Newton-Raphson iteration, it is solved iteratively by

$$\begin{bmatrix} \mathbf{M}(q^*_{n+1}) & C^T(q_{n+1}) \\ C^T(q_{n+1}) & 0 \end{bmatrix} \begin{Bmatrix} \Delta q^{k+1}_{n+1} \\ \Delta\sigma^{k+1} \end{Bmatrix}^{k+1} = - \begin{Bmatrix} \mathbf{M}(q_{n+1} - q^*_{n+1}) + C^T(q_{n+1})\sigma \\ \mathbf{\Phi}^*(q_{n+1}, t) \end{Bmatrix}^k \tag{4.21}$$

and update the displacement and Lagrange multiplier with

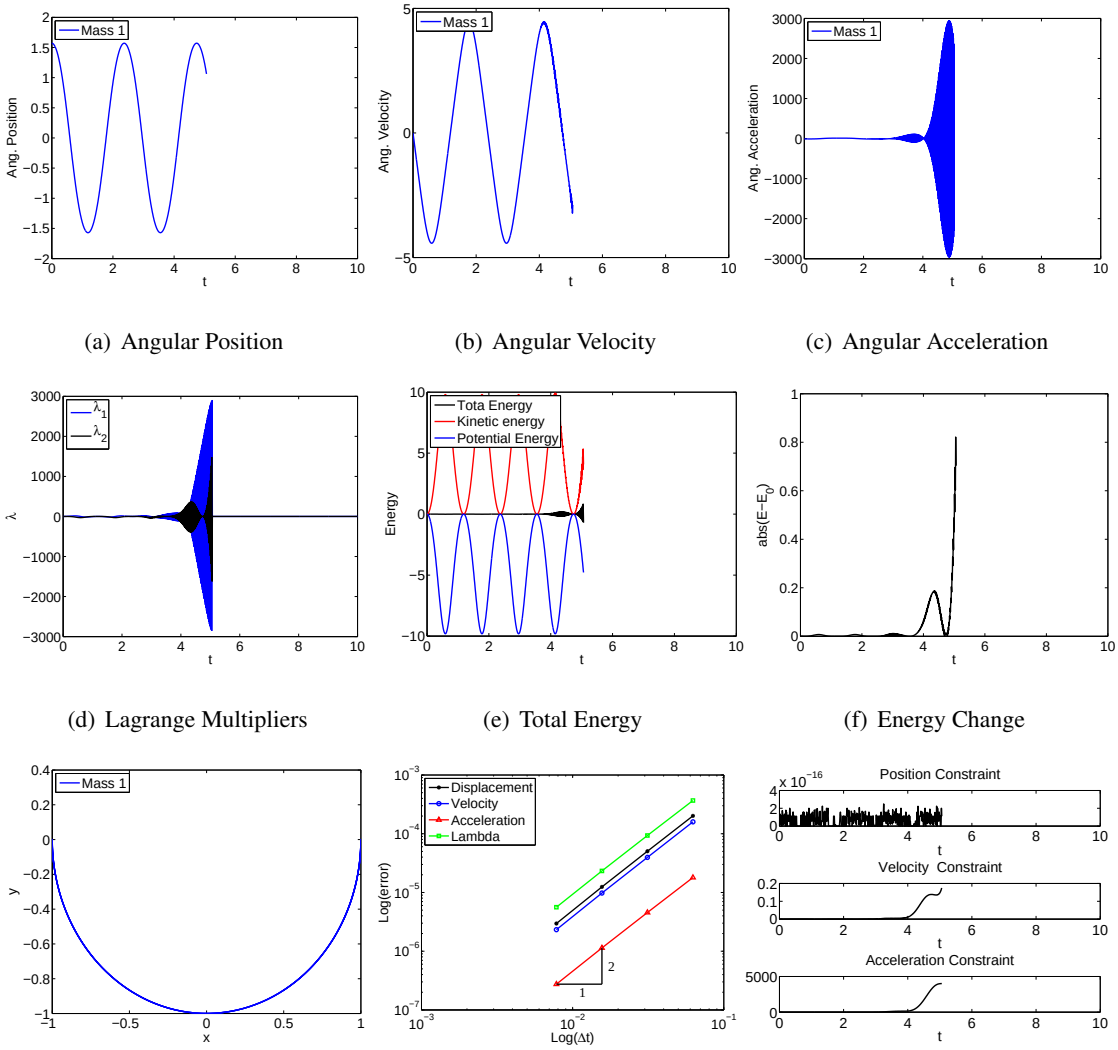
$$\begin{aligned} q^{k+1}_{n+1} &= q^k_{n+1} + \Delta q^{k+1}_{n+1} \\ \sigma^{k+1} &= \sigma^k + \Delta\sigma^{k+1} \end{aligned} \tag{4.22}$$

where the recursion is started by setting

$$\begin{aligned} q^k_{n+1} &= q^*_{n+1} \\ \sigma^k &= \mathbf{\Phi}^k \end{aligned} \tag{4.23}$$

where q could be u, v, a respectively in each constraint projection.

The problem was run using a-form representation of the single field form algorithm with and without projection methods. The time step used in all simulations was $\Delta t = 0.01$, the nonlinear iteration tolerance 10^{-8}, and the number of iterations limited to 100 per time step. The simulation duration was first set to 10 seconds, through not all cases were able to provide a solution. A discussion of the results follows the figures below.

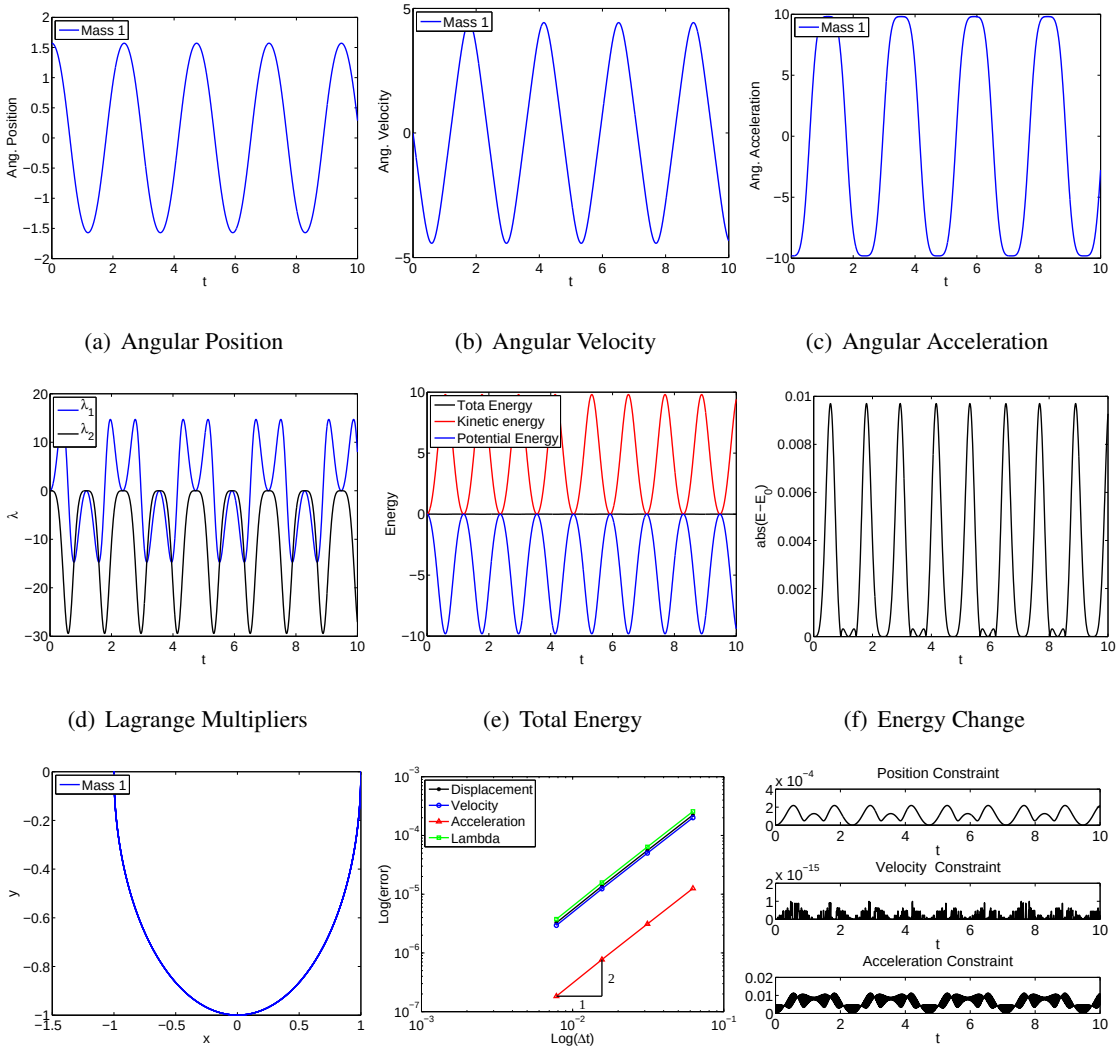


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.2: Single rigid pendulum: U0(1,1,0) - Index 3.

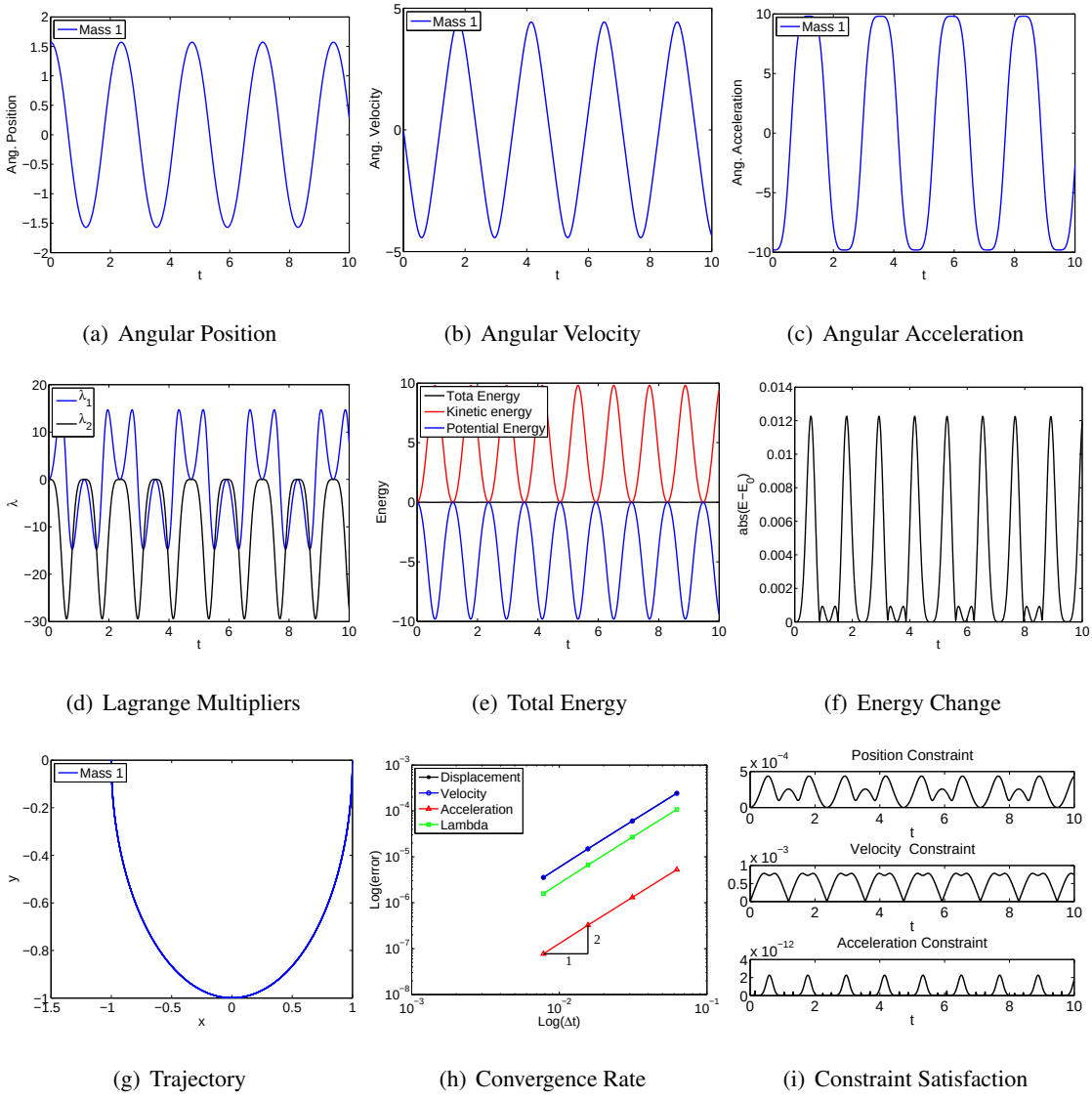


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.3: Single rigid pendulum: U0(1,1,0) - Index 2.

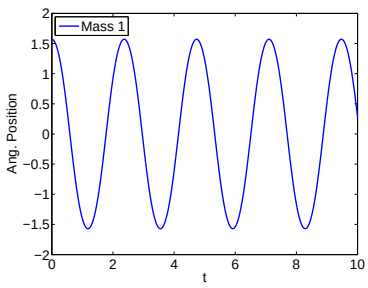


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

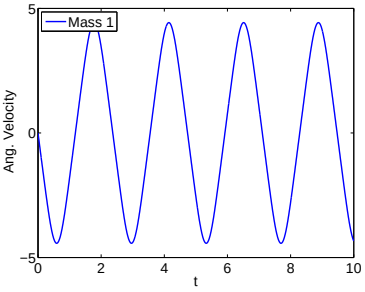
(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

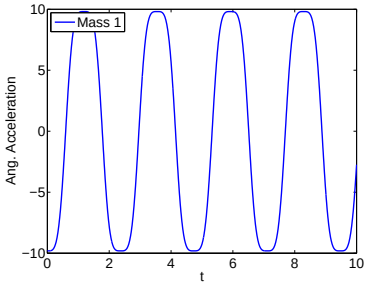
Figure 4.4: Single rigid pendulum: U0(1,1,0) - Index 1.



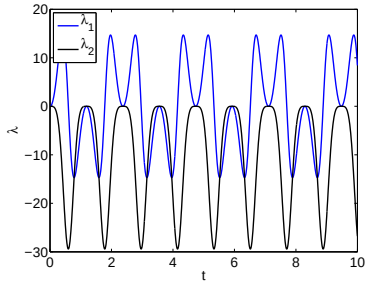
(a) Angular Position



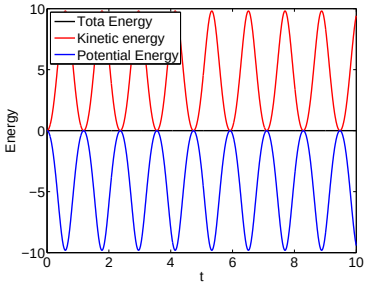
(b) Angular Velocity



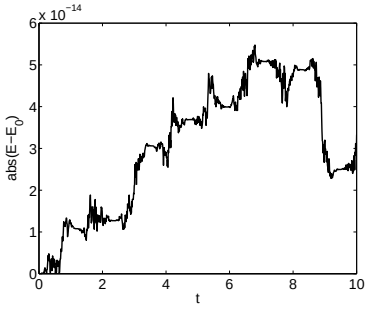
(c) Angular Acceleration



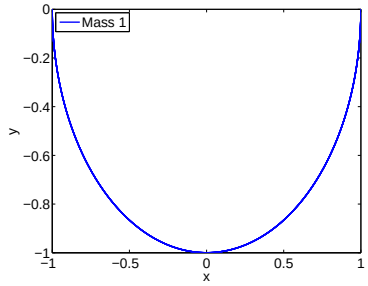
(d) Lagrange Multipliers



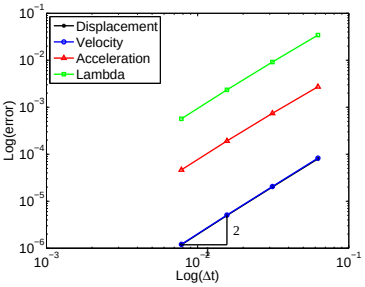
(e) Total Energy



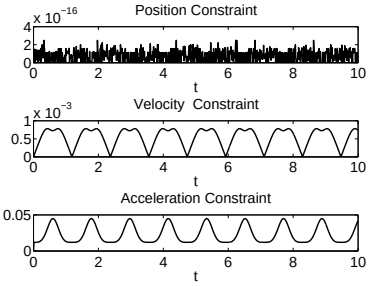
(f) Energy Change



(g) Trajectory

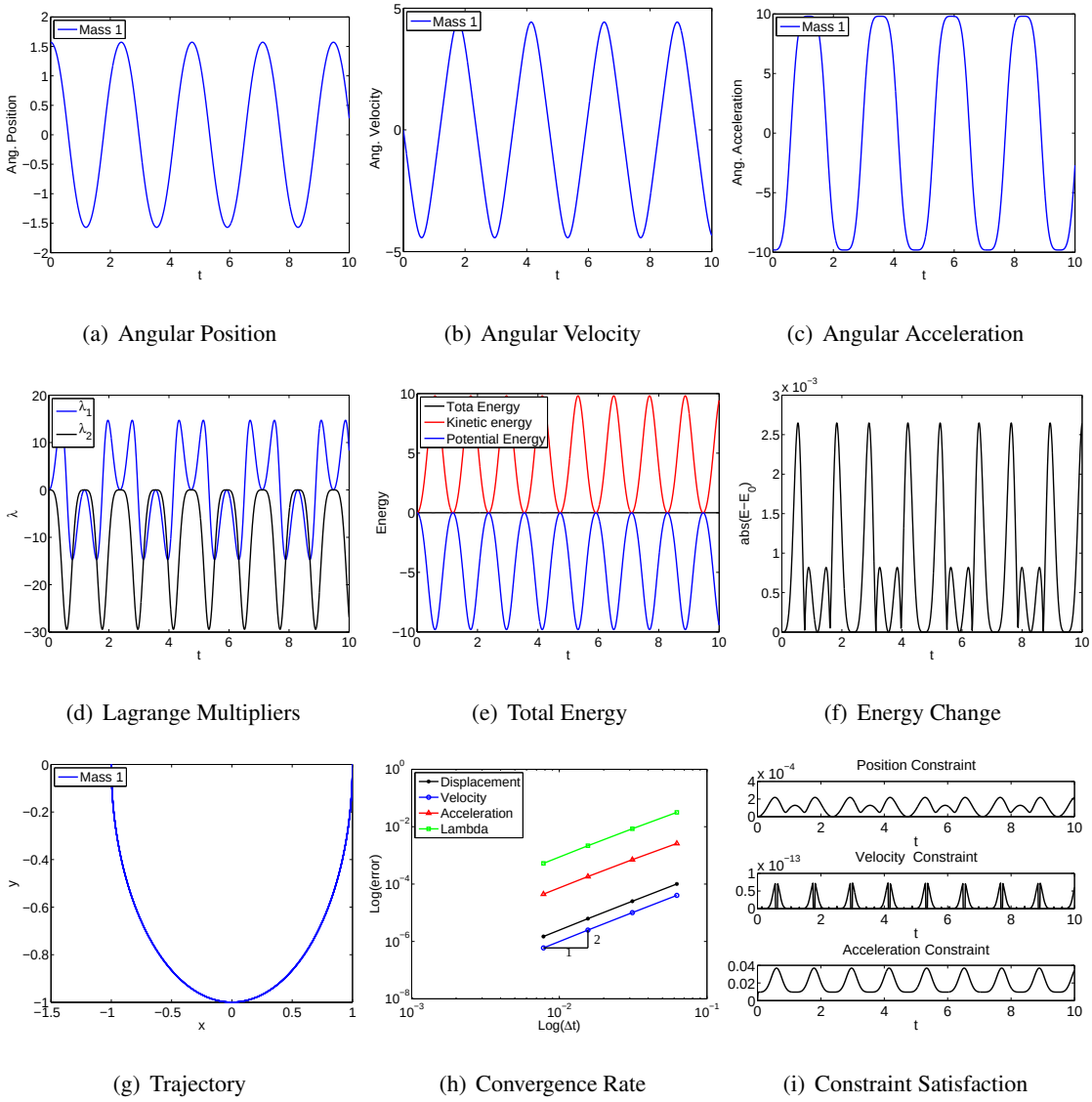


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.5: Single rigid pendulum: V0(1,1,0) - Index 3.

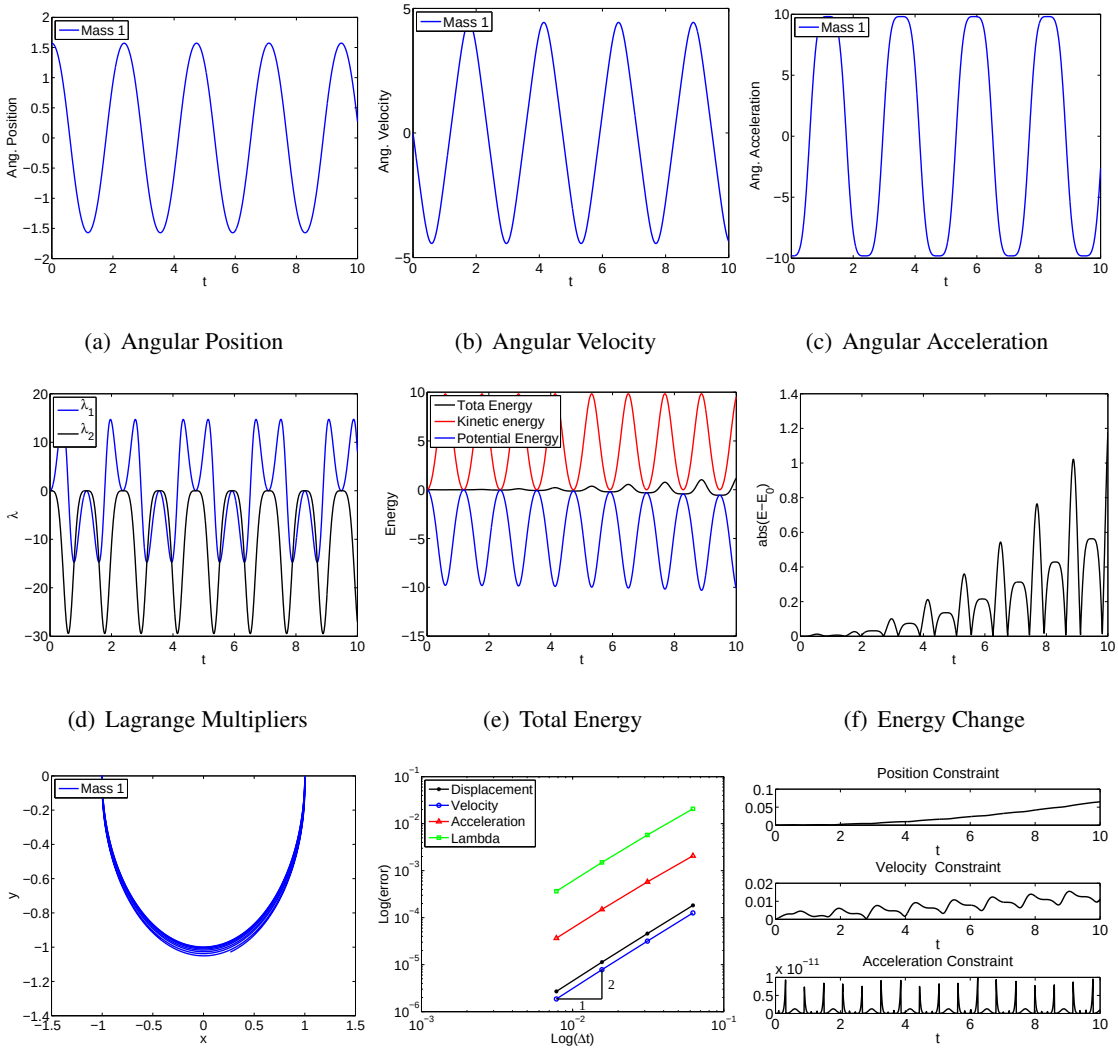


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.6: Single rigid pendulum: V0(1,1,0) - Index 2.

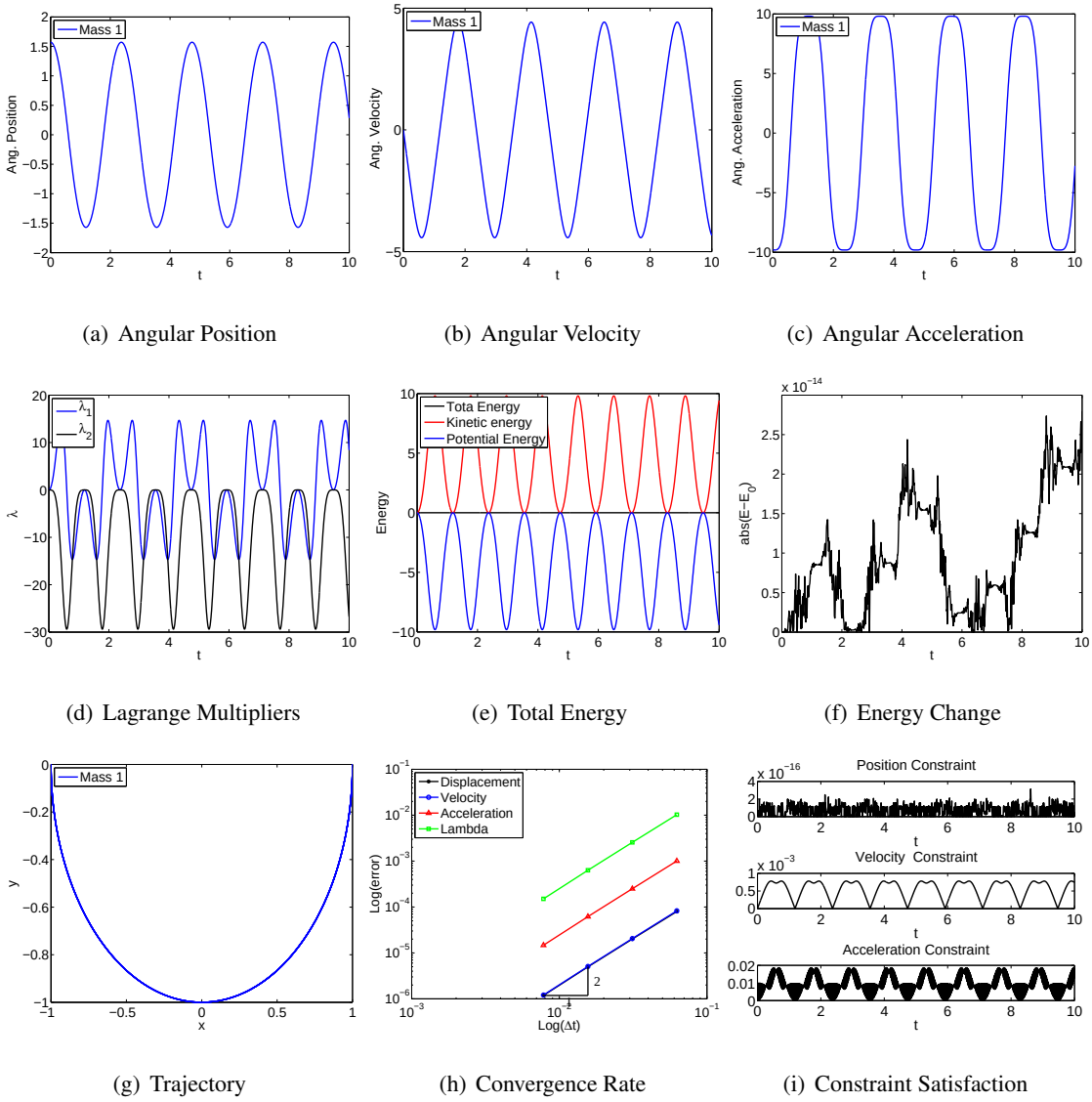


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.7: Single rigid pendulum: V0(1,1,0) - Index 1.

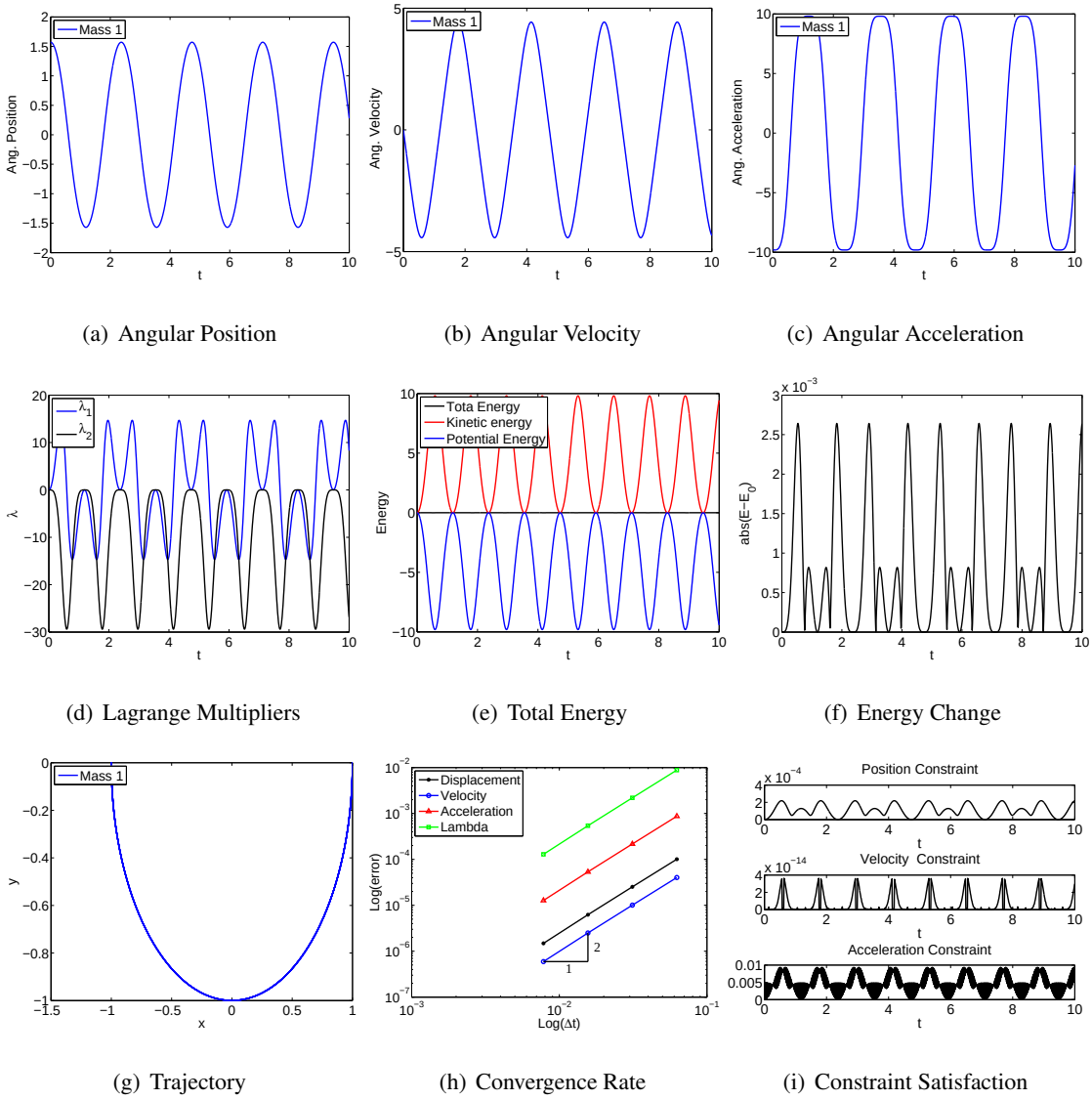


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.8: Single rigid pendulum: U0V0(1,1,1) - Index 3.

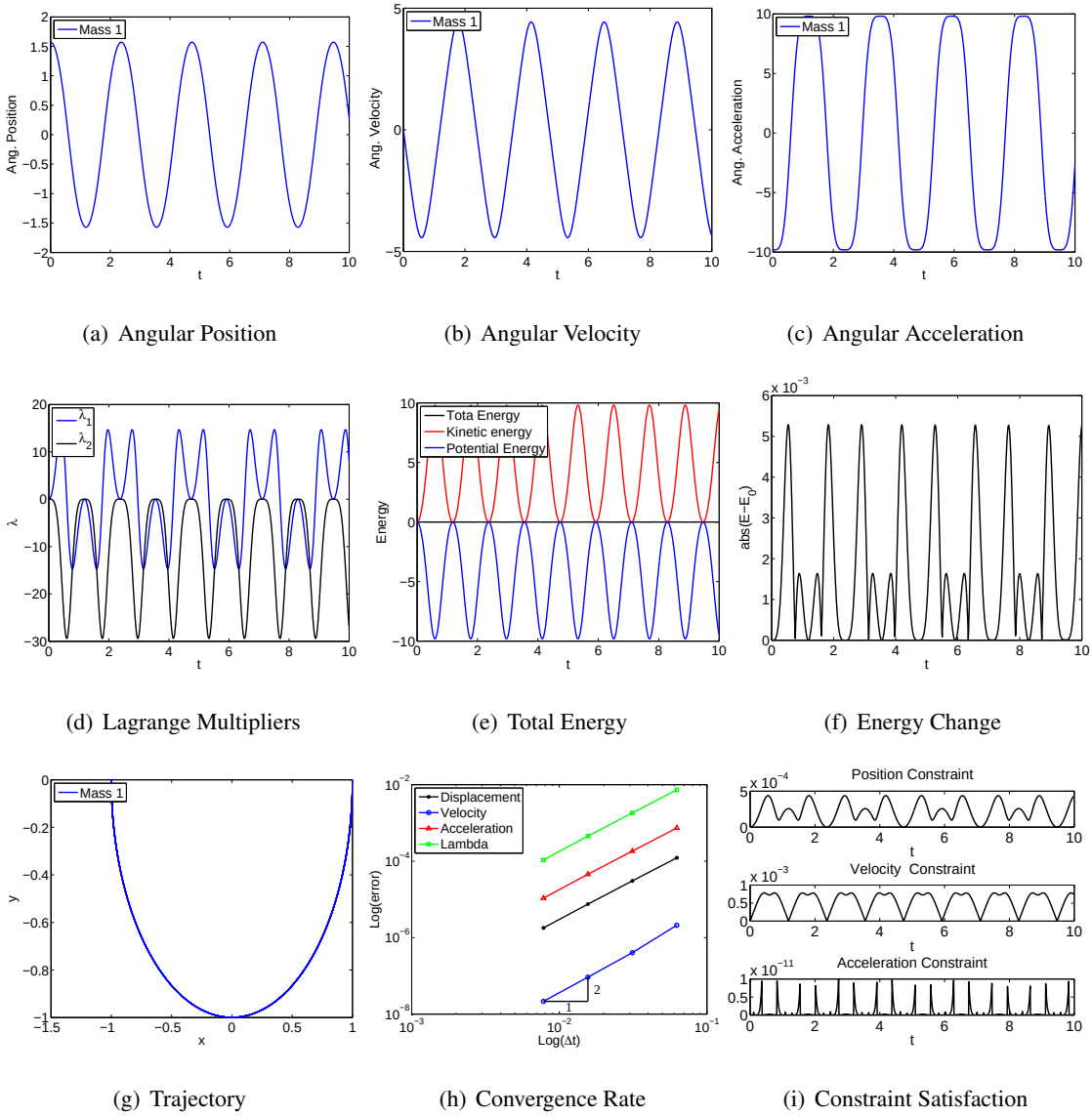


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.9: Single rigid pendulum: U0V0(1,1,1) - Index 2.



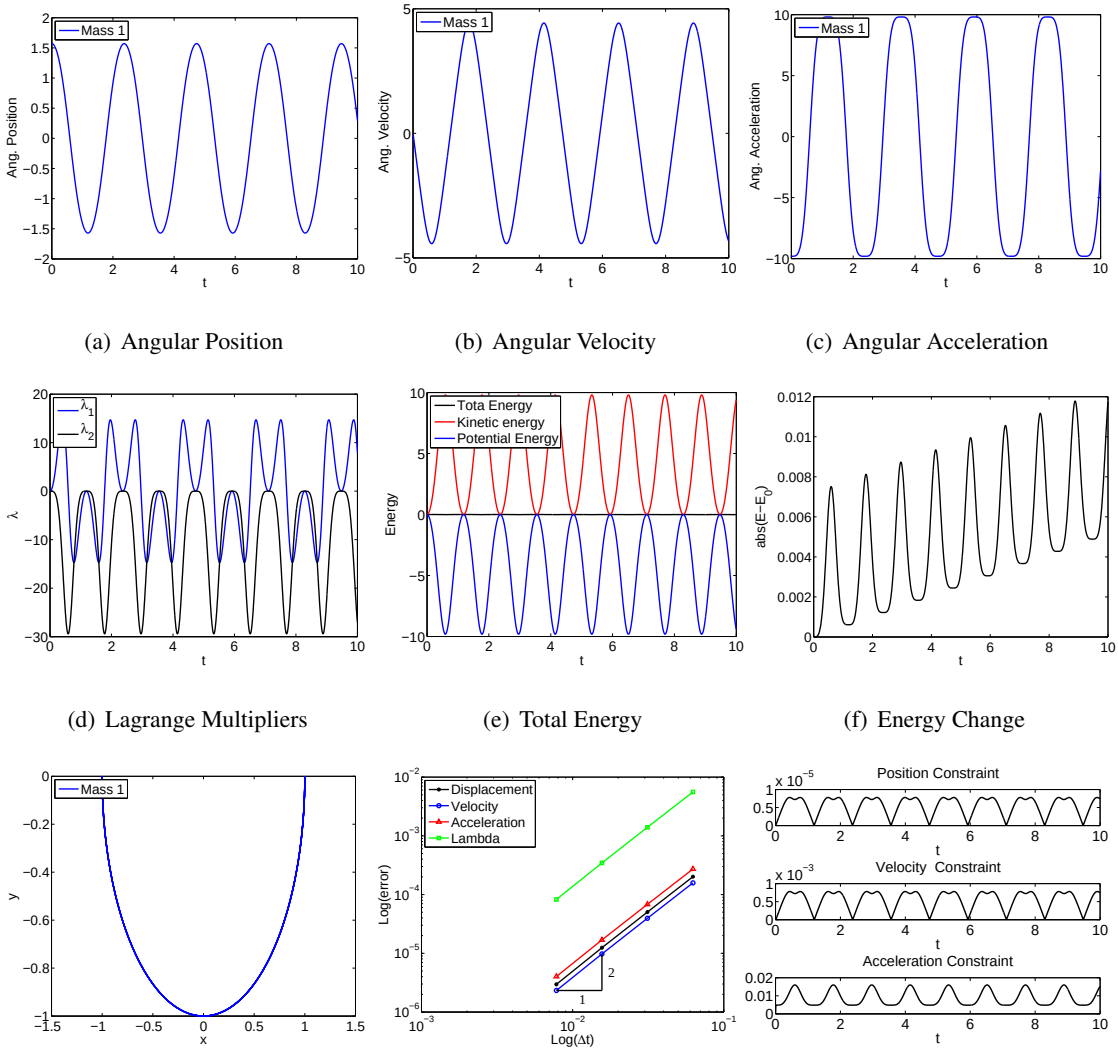
(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.10: Single rigid pendulum: U0V0(1,1,1) - Index 1.

Baumgarte's Method

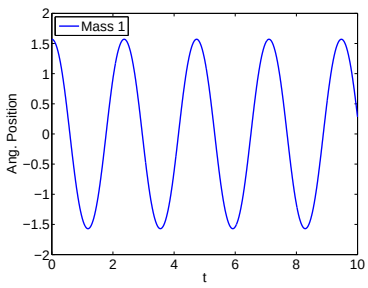


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

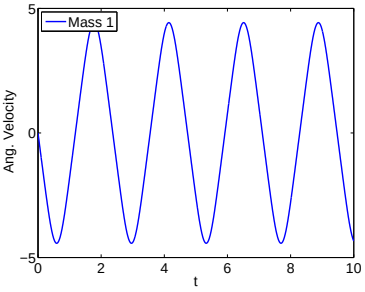
(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

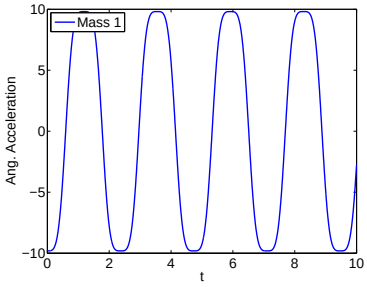
Figure 4.11: Single rigid pendulum with Baumgarte method: U0(1,1,0).



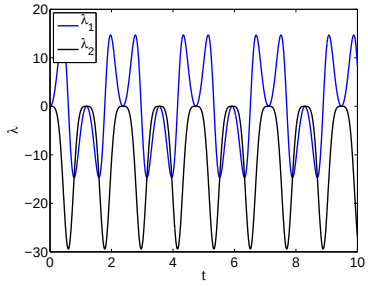
(a) Angular Position



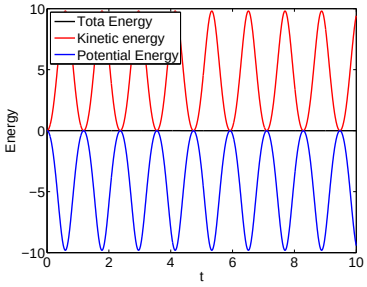
(b) Angular Velocity



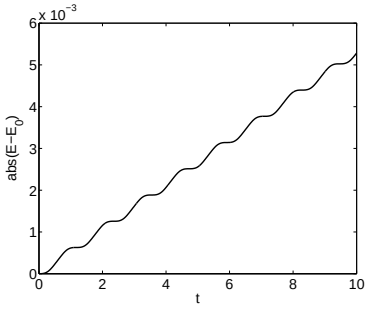
(c) Angular Acceleration



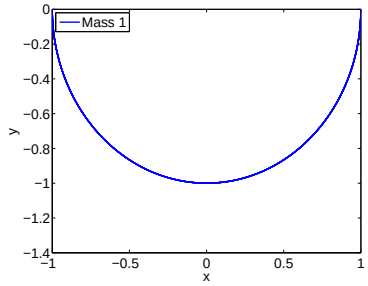
(d) Lagrange Multipliers



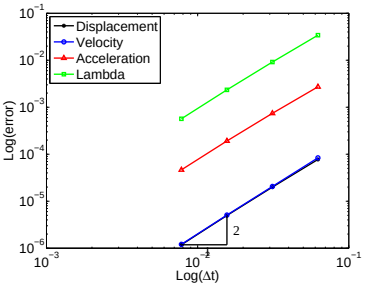
(e) Total Energy



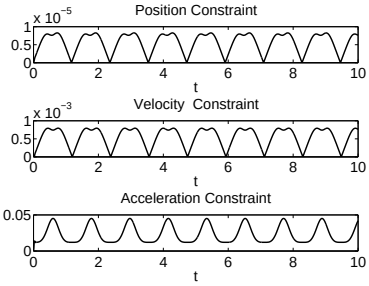
(f) Energy Change



(g) Trajectory

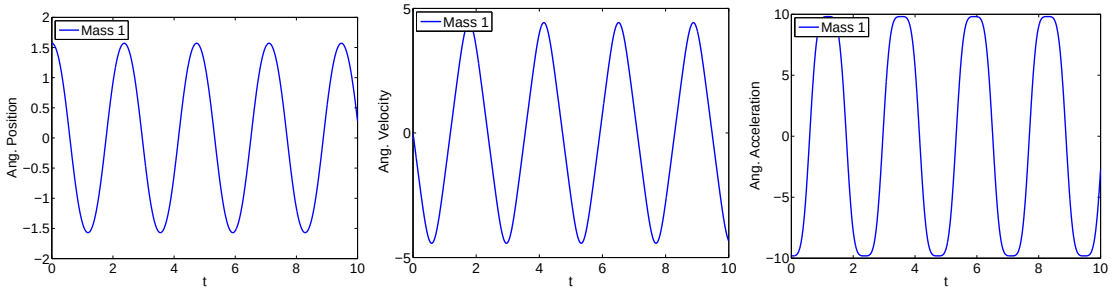


(h) Convergence Rate



(i) Constraint Satisfaction

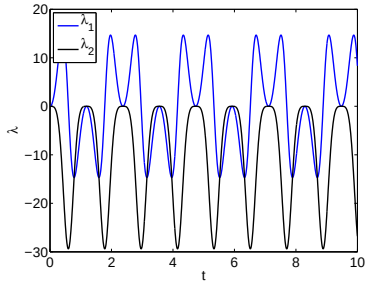
Figure 4.12: Single rigid pendulum with Baumgarte method: V0(1,1,0).



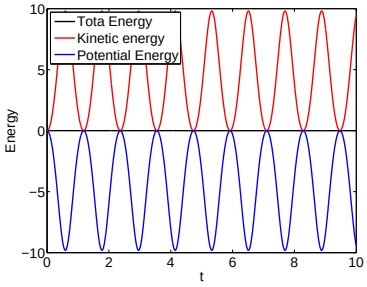
(a) Angular Position

(b) Angular Velocity

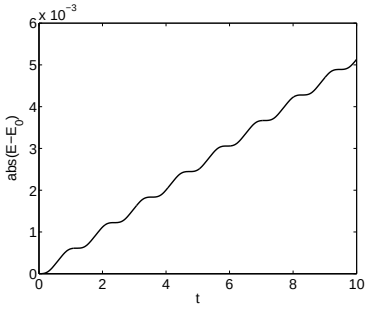
(c) Angular Acceleration



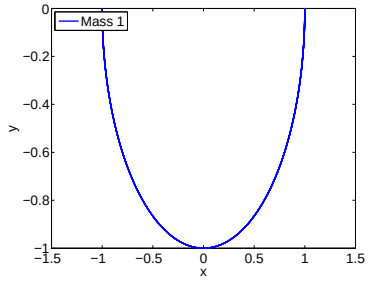
(d) Lagrange Multipliers



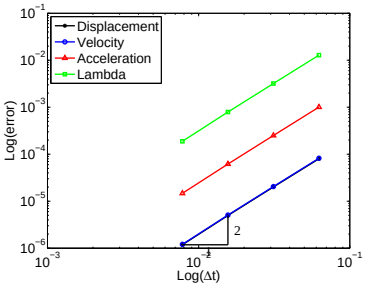
(e) Total Energy



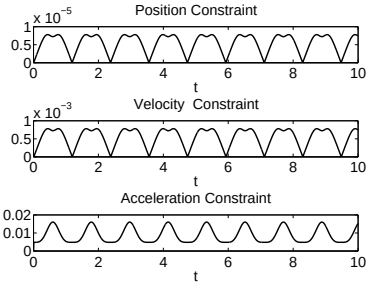
(f) Energy Change



(g) Trajectory



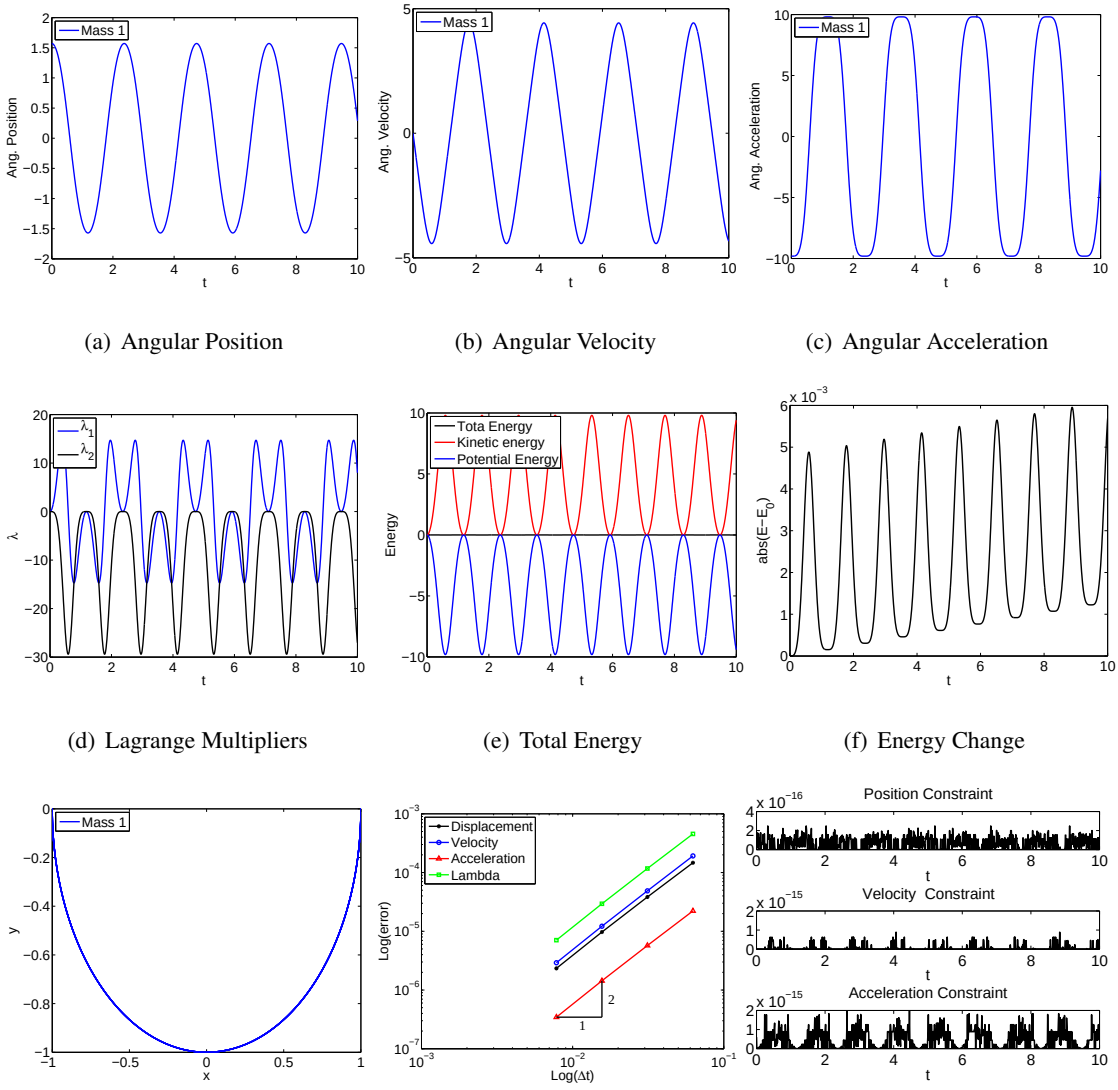
(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.13: Single rigid pendulum with Baumgarte method: U0V0(1,1,1).

Projection Method

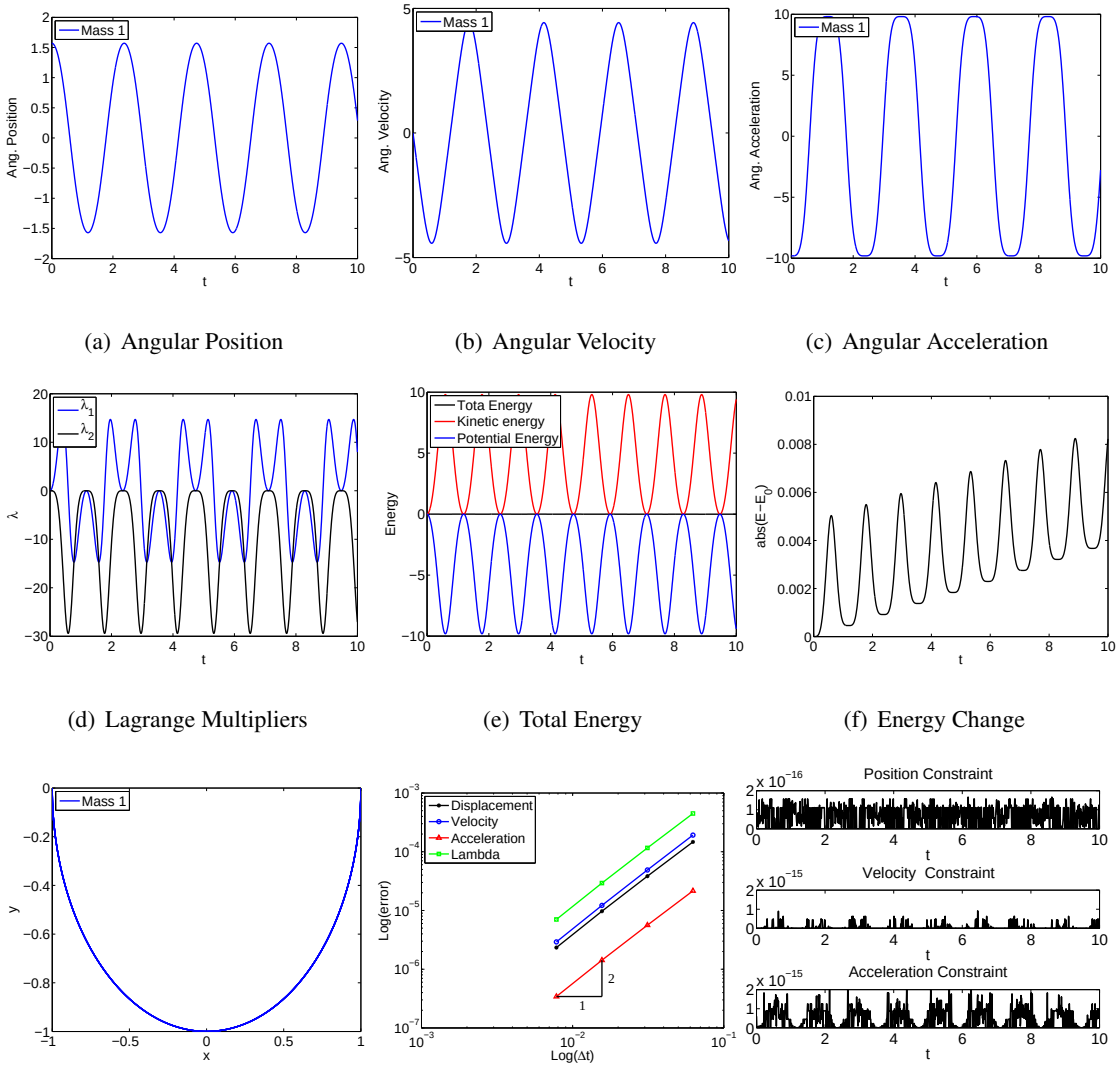


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.14: Single rigid pendulum with projection method: U0(1,1,0) - Index 3.

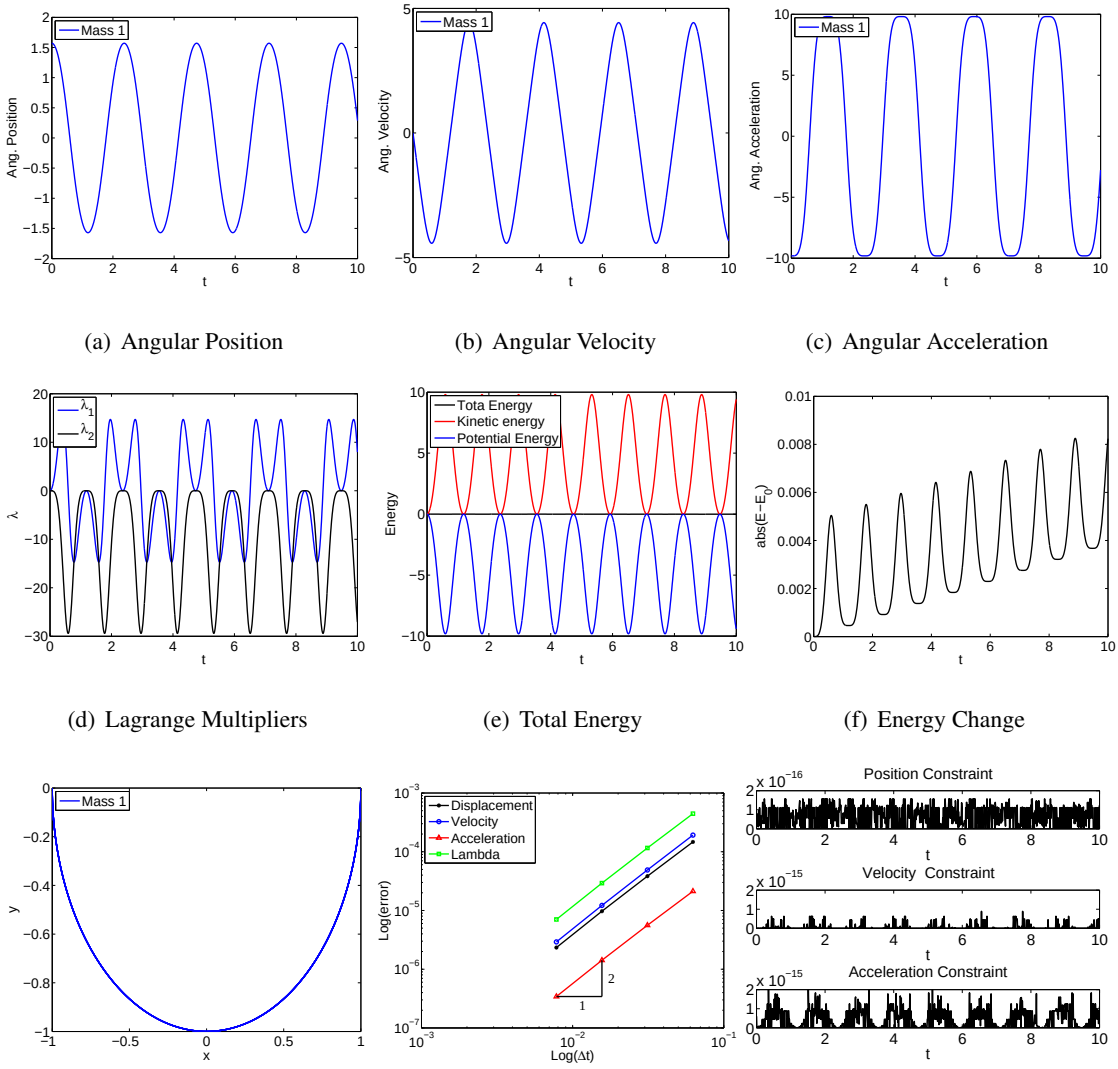


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.15: Single rigid pendulum with projection method: U0(1,1,0) - Index 2.

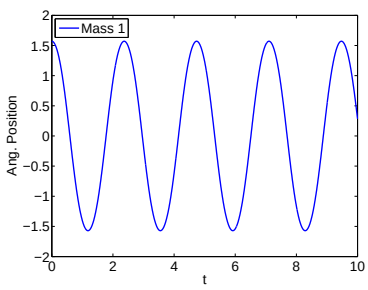


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

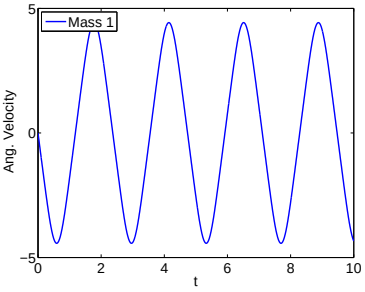
(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

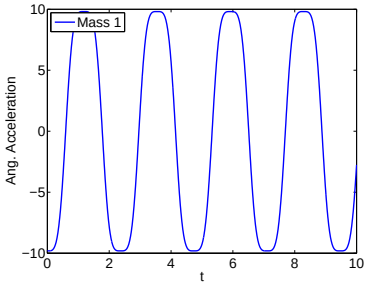
Figure 4.16: Single rigid pendulum with projection method: U0(1,1,0) - Index 1.



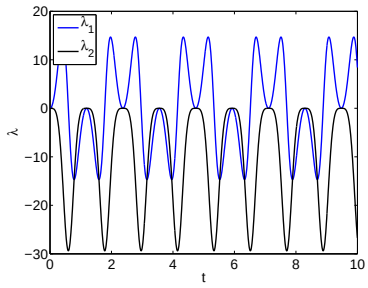
(a) Angular Position



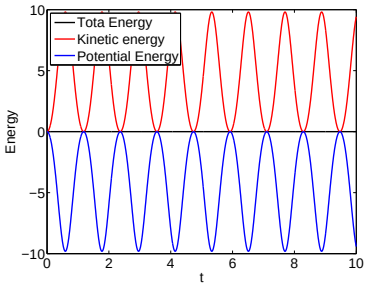
(b) Angular Velocity



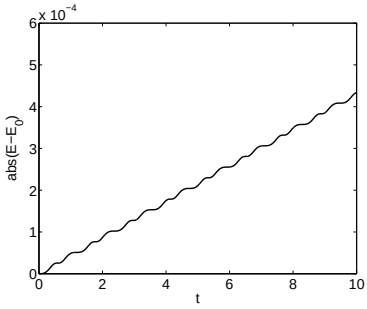
(c) Angular Acceleration



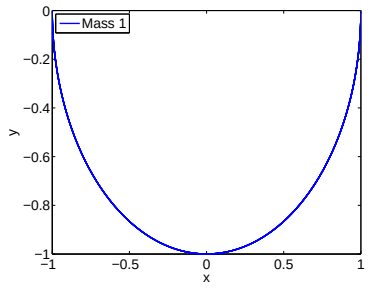
(d) Lagrange Multipliers



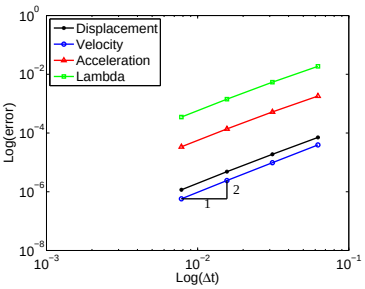
(e) Total Energy



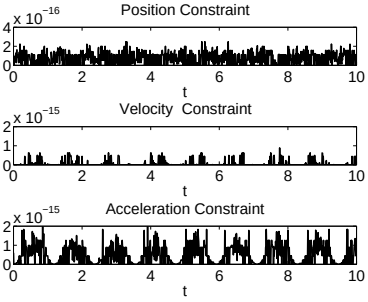
(f) Energy Change



(g) Trajectory

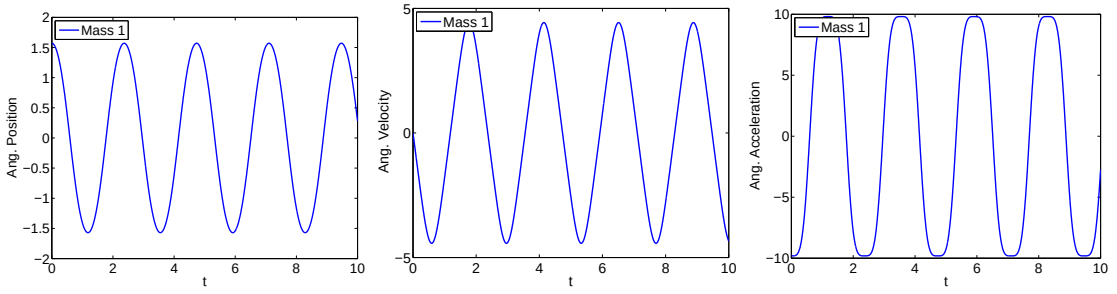


(h) Convergence Rate



(i) Constraint Satisfaction

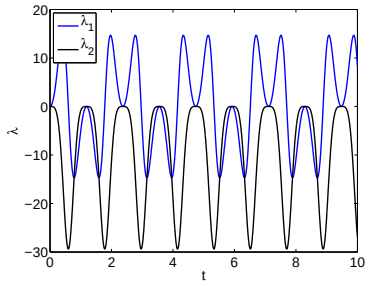
Figure 4.17: Single rigid pendulum with projection method: V0(1,1,0) - Index 3.



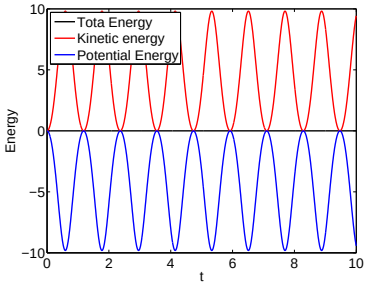
(a) Angular Position

(b) Angular Velocity

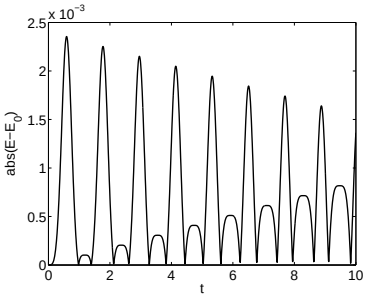
(c) Angular Acceleration



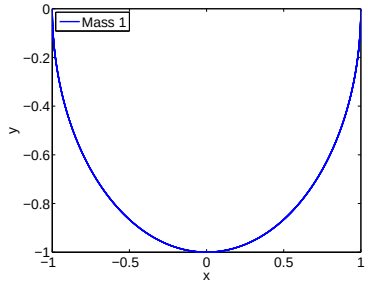
(d) Lagrange Multipliers



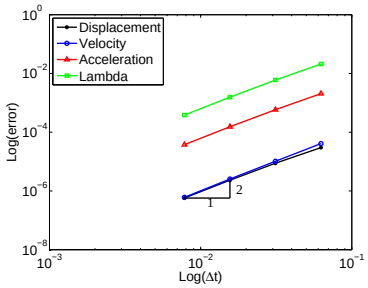
(e) Total Energy



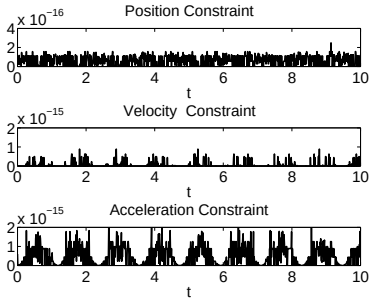
(f) Energy Change



(g) Trajectory

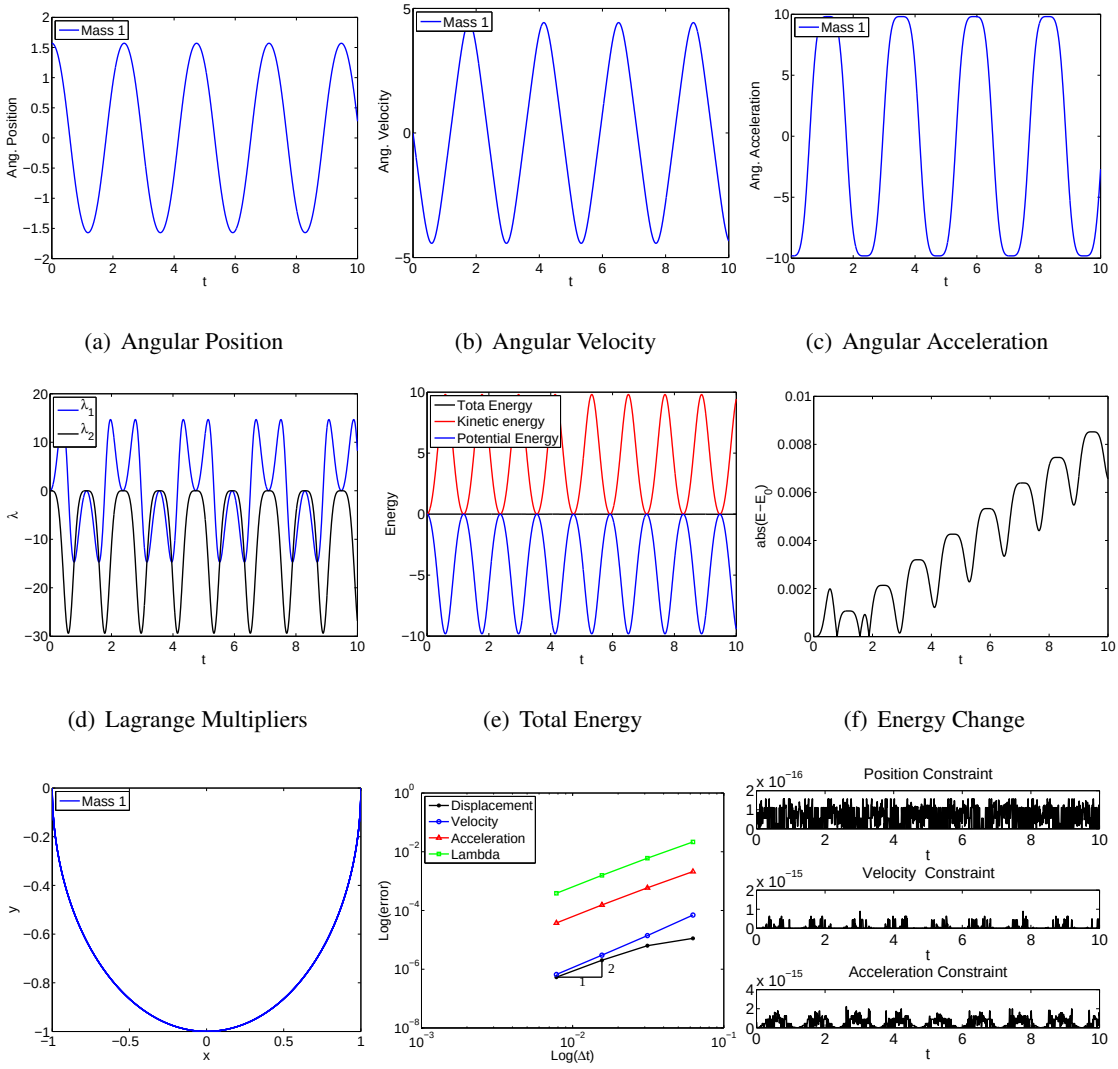


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.18: Single rigid pendulum with projection method: V0(1,1,0) - Index 2.

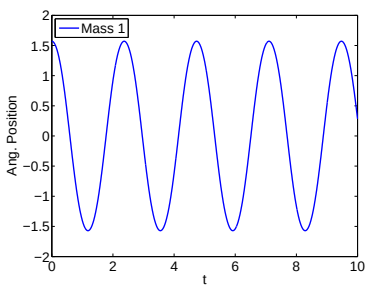


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

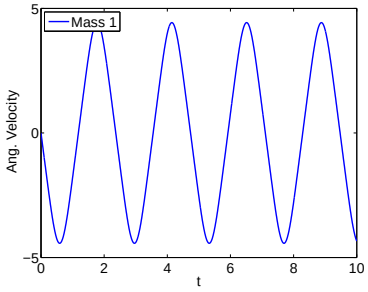
(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

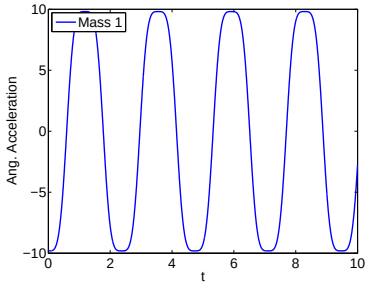
Figure 4.19: Single rigid pendulum with projection method: V0(1,1,0) - Index 1.



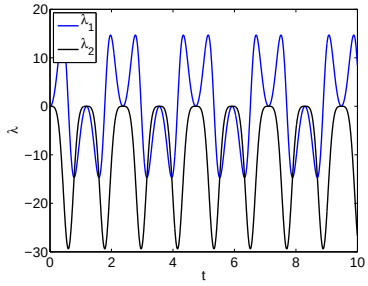
(a) Angular Position



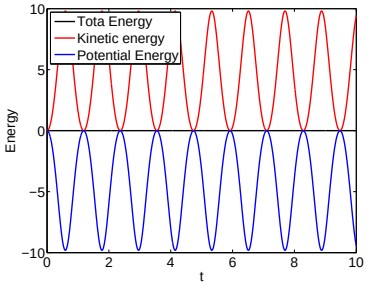
(b) Angular Velocity



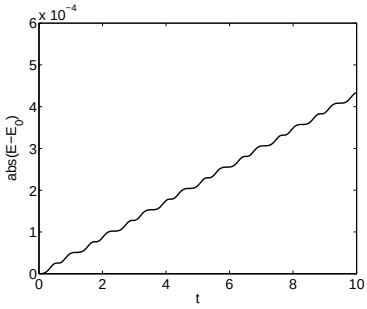
(c) Angular Acceleration



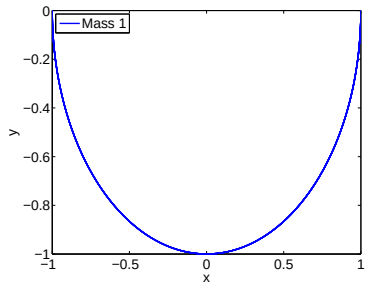
(d) Lagrange Multipliers



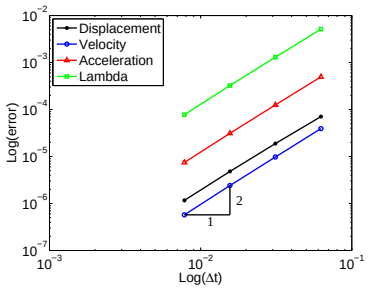
(e) Total Energy



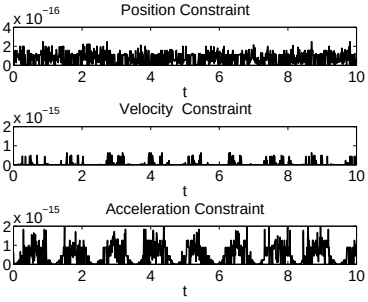
(f) Energy Change



(g) Trajectory

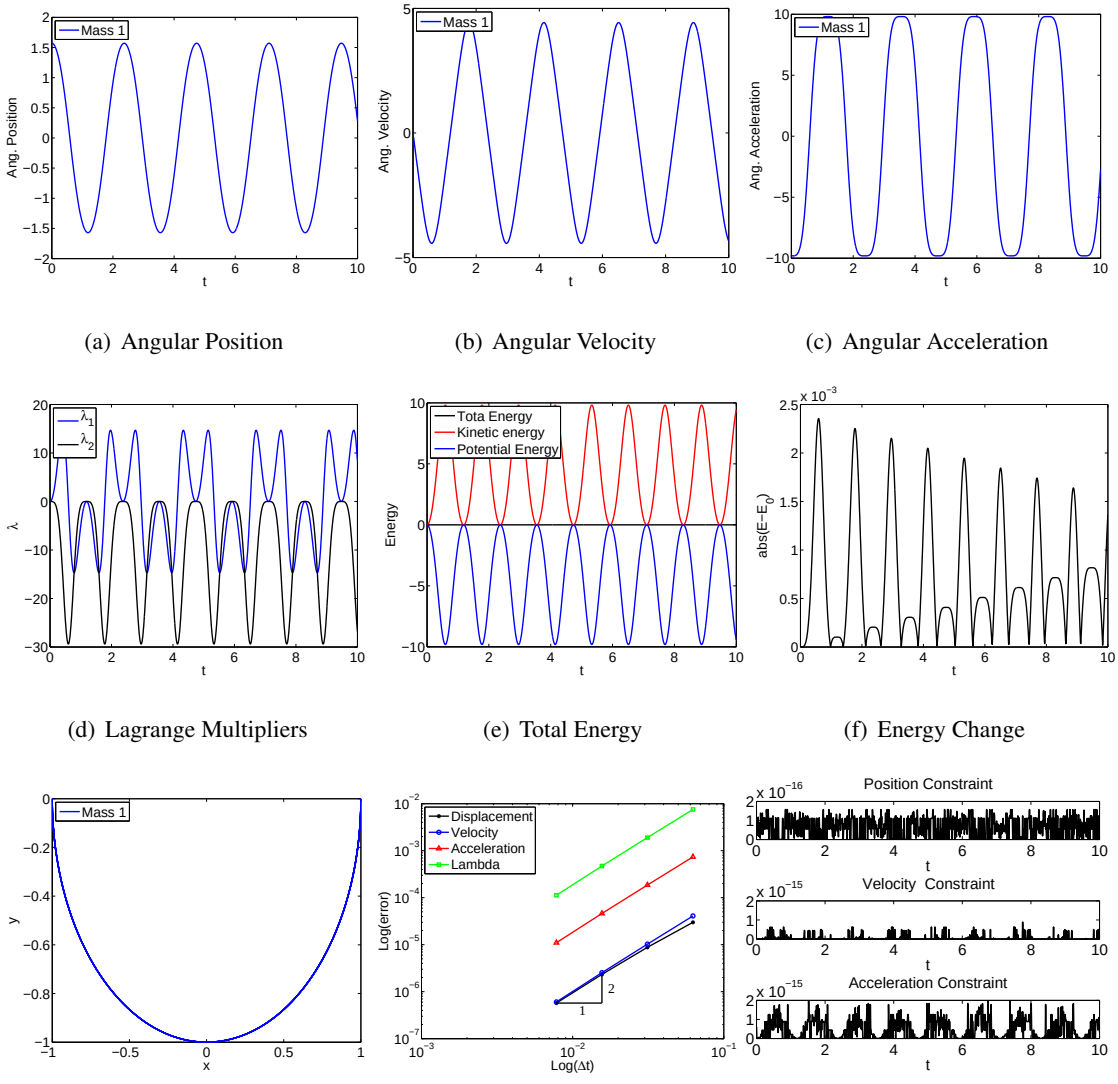


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.20: Single rigid pendulum with projection method: U0V0(1,1,1) - Index 3.

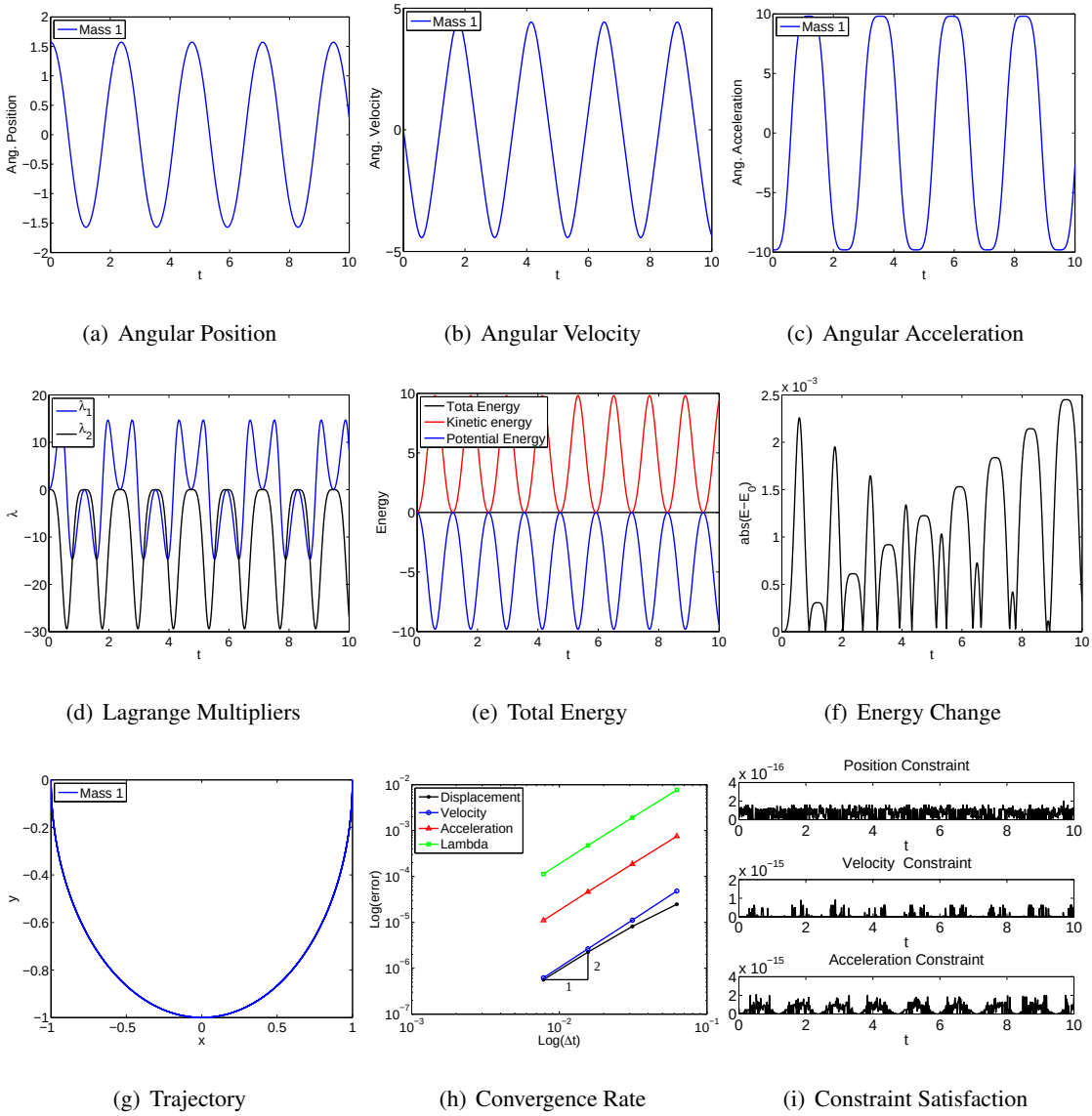


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.21: Single rigid pendulum with projection method: U0V0(1,1,1) - Index 2.



(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.22: Single rigid pendulum with projection method: U0V0(1,1,1) - Index 1.

4.1.2 Double Pendulum

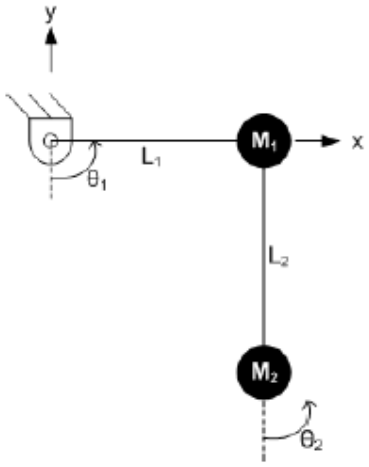


Figure 4.23: Double pendulum problem from[12].

Demonstration of the rigid body dynamics with and without stabilization techniques is shown for the rigid double pendulum shown in Fig. 4.23. The model consists of two planar rigid bar, with two point masses at both ends, pinned to the ground at origin and to the hinge connecting both bar, Gravity is the only external force acting on the system. Redundant coordinates (x, y, θ) are used for simplicity to track the location of each point mass during the simulation. This problem is discussed in great detail in [12], with pertinent information repeated below.
In general the problem is of the form:

$$\begin{aligned} \mathbf{M}a + \mathbf{C}^T(u)\lambda &= \mathbf{F} \\ \mathbf{\Phi}(u) &= 0 \end{aligned} \tag{4.24}$$

The problem consists of six degrees of freedom and four Lagrange multiplier. That is, the unknowns are:

$$
\begin{aligned}
u &= [x_1 \quad y_1 \quad \theta_1 \quad x_2 \quad y_2 \quad \theta_2]^T \\
v &= [\dot{x}_1 \quad \ddot{y}_1 \quad \dot{\theta}_1 \quad \dot{x}_2 \quad \ddot{y}_2 \quad \dot{\theta}_2]^T \\
a &= [\ddot{x}_1 \quad \ddot{y}_1 \quad \ddot{\theta}_1 \quad \ddot{x}_2 \quad \ddot{y}_2 \quad \ddot{\theta}_2]^T \\
\lambda &= [\lambda_1 \quad \lambda_2 \quad \lambda_3 \quad \lambda_4]^T
\end{aligned}
\tag{4.25}
$$

Two constraint equations serve to keep the length of bar constant (geometry constraint):

$$
\mathbf{\Phi}(u) = \begin{bmatrix} x_1 - L_1 sin(\theta_1) \\ y_1 + L_1 cos(\theta_1) \\ x_2 - x_1 - L_2 sin(\theta_2) \\ y_2 - y_1 + L_2 cos(\theta_2) \end{bmatrix}
\tag{4.26}
$$

The first derivative of geometry constraint yield velocity constraint:

$$
\dot{\mathbf{\Phi}}(u, v) = \begin{bmatrix} \dot{x}_1 - L_1 cos(\theta_1)\dot{\theta}_1 \\ \dot{y}_1 - L_1 sin(\theta_1)\dot{\theta}_1 \\ \dot{x}_2 - \dot{x}_1 - L_2 cos(\theta_2)\dot{\theta}_2 \\ \dot{y}_2 - \dot{y}_1 - L_2 sin(\theta_2)\dot{\theta}_2 \end{bmatrix}
\tag{4.27}
$$

where Index 1 form are:

$$
\ddot{\mathbf{\Phi}}(u, v, a) = \begin{bmatrix} \ddot{x}_1 + L_1 sin(\theta_1)\dot{\theta}_1^2 - L_1 cos(\theta_1)\ddot{\theta}_1 \\ \ddot{y}_1 - L_1 cos(\theta_1)\dot{\theta}_1^2 - L_1 sin(\theta_1)\ddot{\theta}_1 \\ \ddot{x}_2 - \ddot{x}_1 + L_2 sin(\theta_2)\dot{\theta}_2^2 - L_2 cos(\theta_2)\ddot{\theta}_2 \\ \ddot{y}_2 - \ddot{y}_1 - L_2 cos(\theta_2)\dot{\theta}_2^2 - L_2 sin(\theta_2)\ddot{\theta}_2 \end{bmatrix}
\tag{4.28}
$$

the corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \frac{\partial \mathbf{\Phi}(u)}{\partial u} = \begin{bmatrix} 1 & 0 & -L_1 cos(\theta_1) & 0 & 0 & 0 \\ 0 & 1 & -L_1 sin(\theta_1) & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & -L_2 cos(\theta_2) \\ 0 & -1 & 0 & 0 & 1 & -L_2 sin(\theta_2) \end{bmatrix} \tag{4.29}$$

As the θ degree of freedom is redundant and used simply to avoid respective calls to tan^{-1}, the potential energy (PE), kinetic energy (KE) the mass matrix and external force use only the Cartesian coordinates in x and y as below.

$$\begin{aligned} \mathbf{KE} &= \frac{1}{2} v^T \mathbf{M} v \\ \mathbf{PE} &= \mathbf{F}^T u \\ \mathbf{M} &= diag([M_1 \quad M_1 \quad 0 \quad M_2 \quad M_2 \quad 0]) \\ \mathbf{F} &= [0 \quad -M_1 g \quad 0 \quad 0 - M_2 g \quad 0] \end{aligned} \tag{4.30}$$

In all simulations below the problem parameters are taken as $M_1 = M_2 = 1$, $L_1 = L_2 = 1$, and the gravitational constant is rounded to $g = 9.81$. The initial conditions are taken as follows:

$$\begin{aligned} u_0 &= [L_1 \quad \frac{\pi}{2} \quad L_1 \quad -L_2 \quad 0]^T \\ v_0 &= [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \end{aligned} \tag{4.31}$$

The Jacobian matrix under Index 1, 2, and 3 for the single field form follow as:

Index 3:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.32}$$

Index 2:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.33}$$

Index 1:

for first time step ($n = 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ \frac{1}{1-\gamma}\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.34}$$

and the next every other time step ($n > 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 \mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) & \mathbf{C}(\tilde{u})^T \\ (1+\gamma)\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.35}$$

where γ is the time shift:

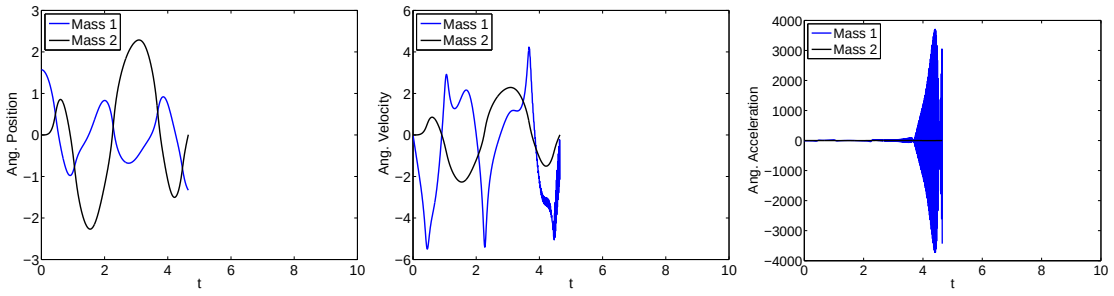
$$\gamma = \Lambda_6 W_1 - W_1 \tag{4.36}$$

and $\mathbf{J}_{C\lambda}(u, \lambda)$ is a 6×6 matrix of zeros with entries:

$$\begin{aligned} \mathbf{J}_{C\lambda}(3,3) &= L_1 \sin\theta_1 \lambda_1 - L_1 \cos\theta_1 \lambda_2 \\ \mathbf{J}_{C\lambda}(6,6) &= L_2 \sin\theta_2 \lambda_3 - L_2 \cos\theta_2 \lambda_4 \end{aligned} \tag{4.37}$$

The problem was ran using a-form representation of the single field form algorithm with and without post processing (Baumgarte method and projection method). The time step used in all simulations was $\Delta t = 0.01$, the nonlinear iteration tolerance 10^{-8}, and the number of iterations limited to 100 per time step. The simulation duration was first set to 10 seconds, through not

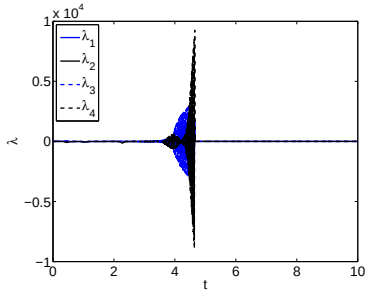
all cases were able to provide a solution. Those which did yield a stable solution were then tested and ran to 100 seconds to determine long duration stability. Similarly, the computational processes runs were conducted for stabilization methods for the problem and all simulation parameters are identical to the single field form simulation above. A discussion of the results follows the figures below.



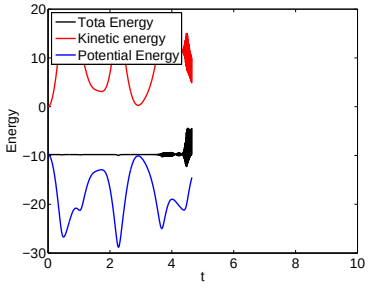
(a) Angular Position

(b) Angular Velocity

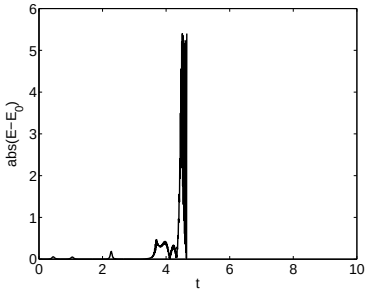
(c) Angular Acceleration



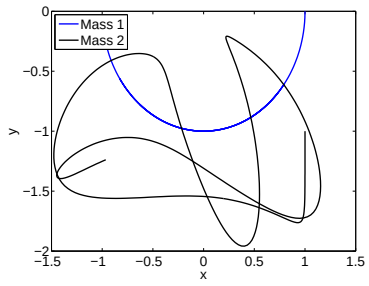
(d) Lagrange Multipliers



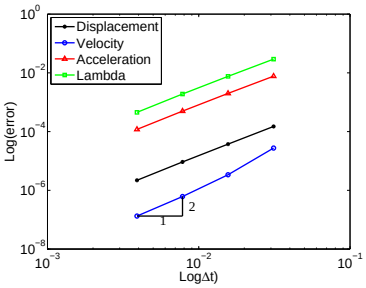
(e) Total Energy



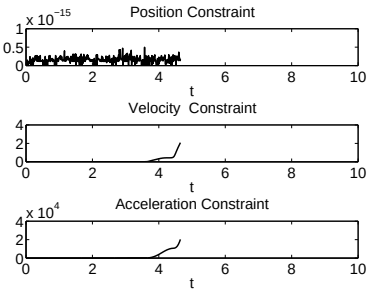
(f) Energy Change



(g) Trajectory

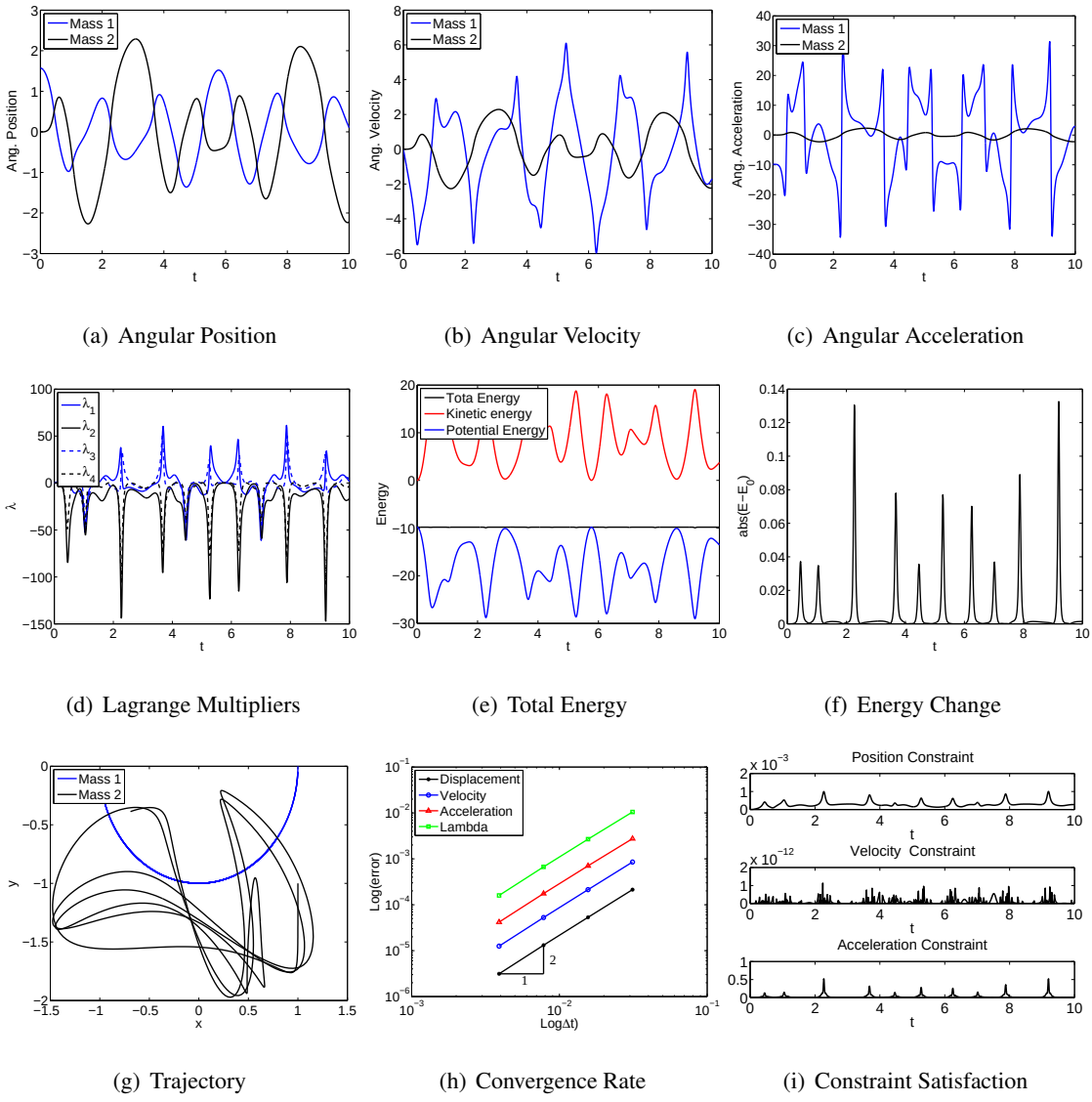


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.24: Double rigid pendulum: U0(1,1,0) - Index 3.

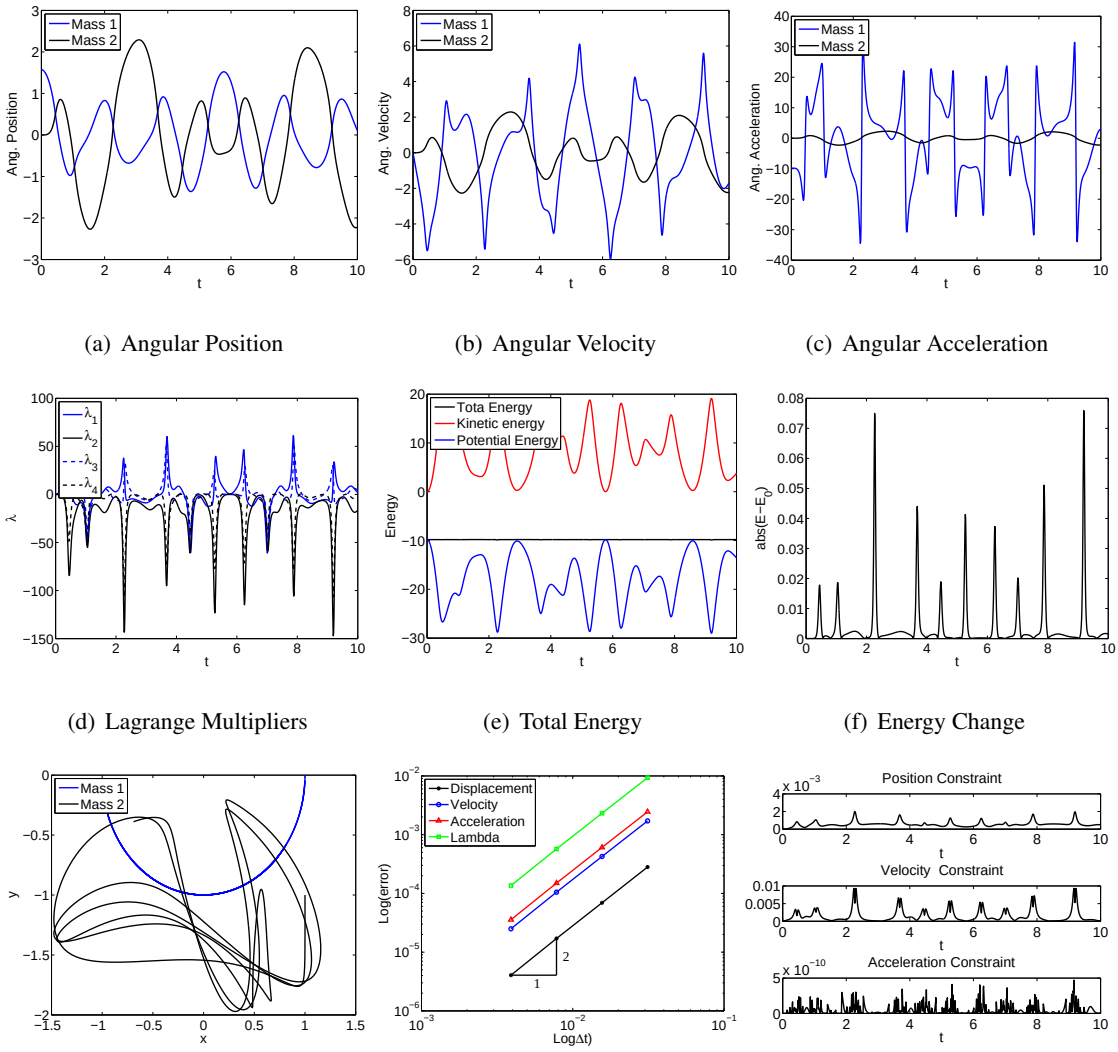


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.25: Double rigid pendulum: U0(1,1,0) - Index 2.

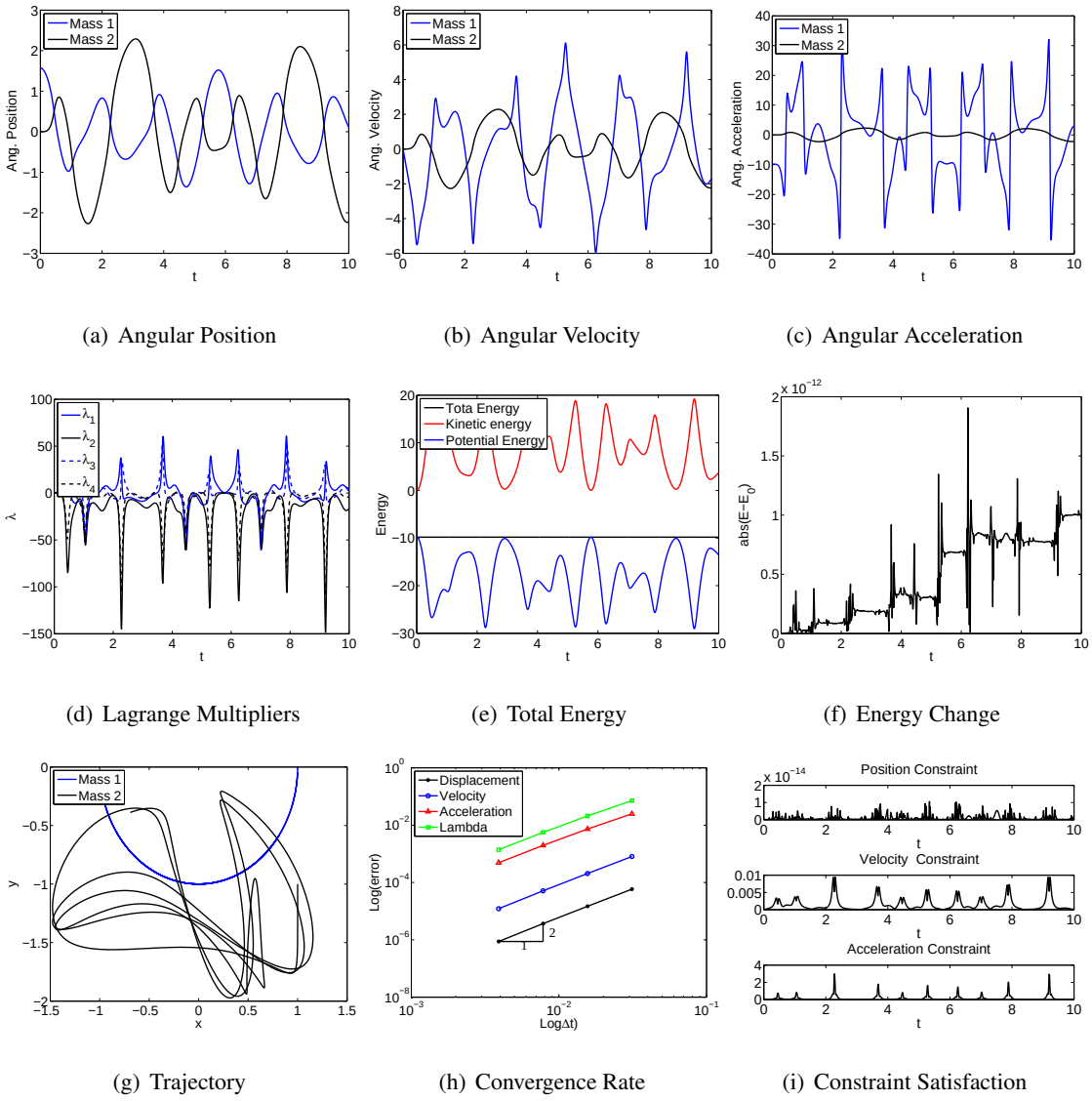


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.26: Double rigid pendulum: U0(1,1,0) - Index 1.

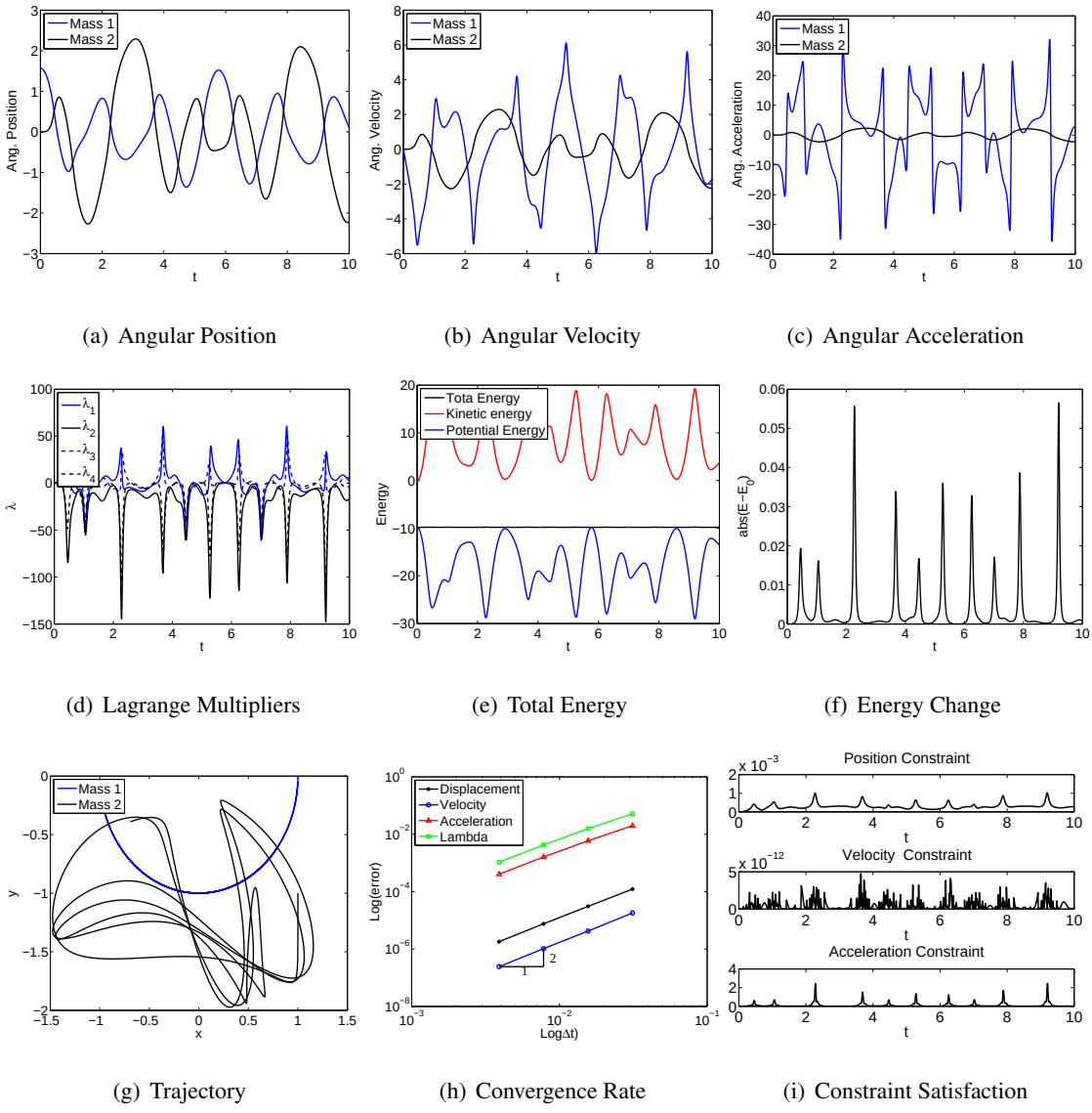


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.27: Double rigid pendulum: V0(1,1,0) - Index 3.

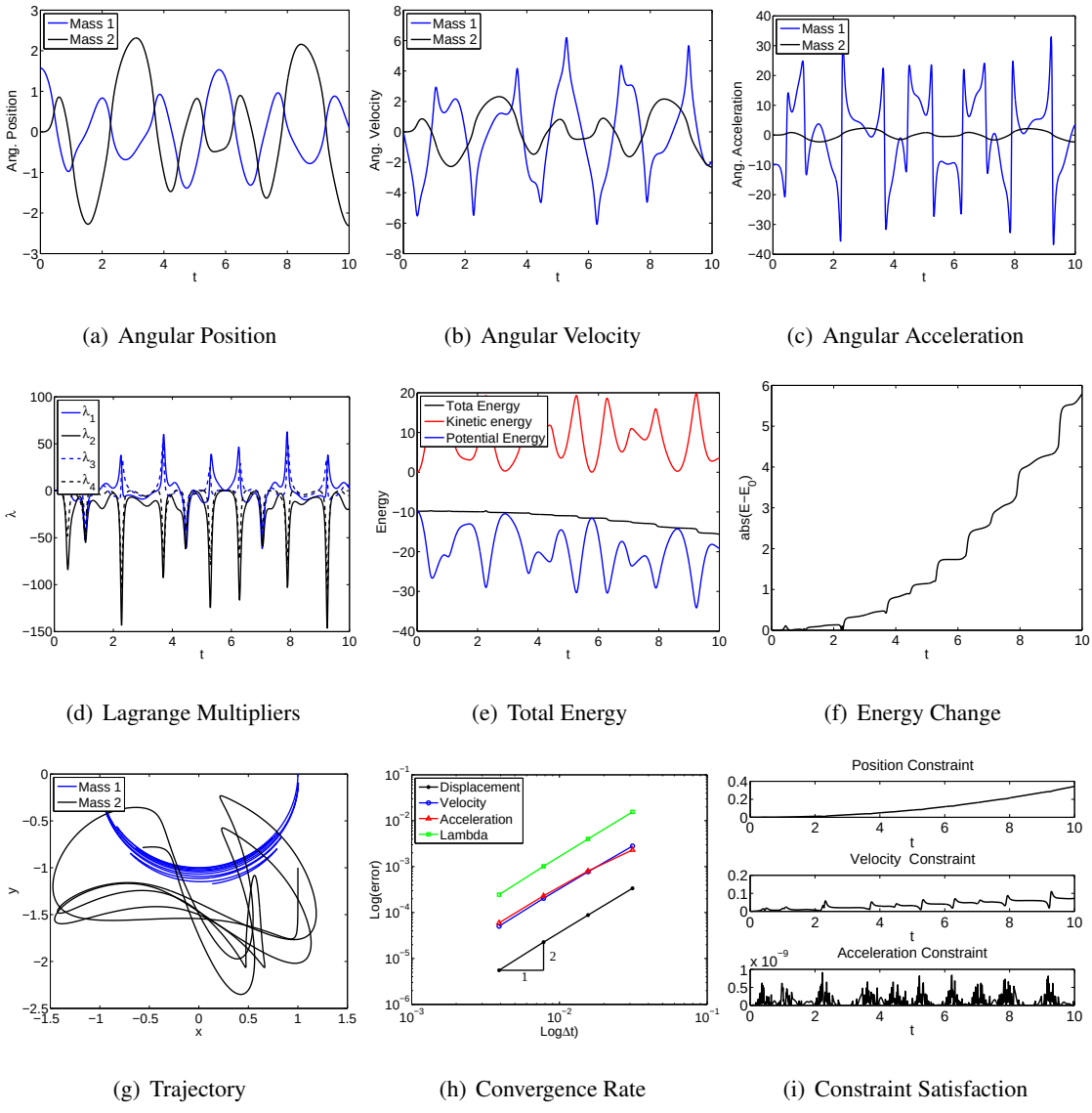


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.28: Double rigid pendulum: V0(1,1,0) - Index 2.

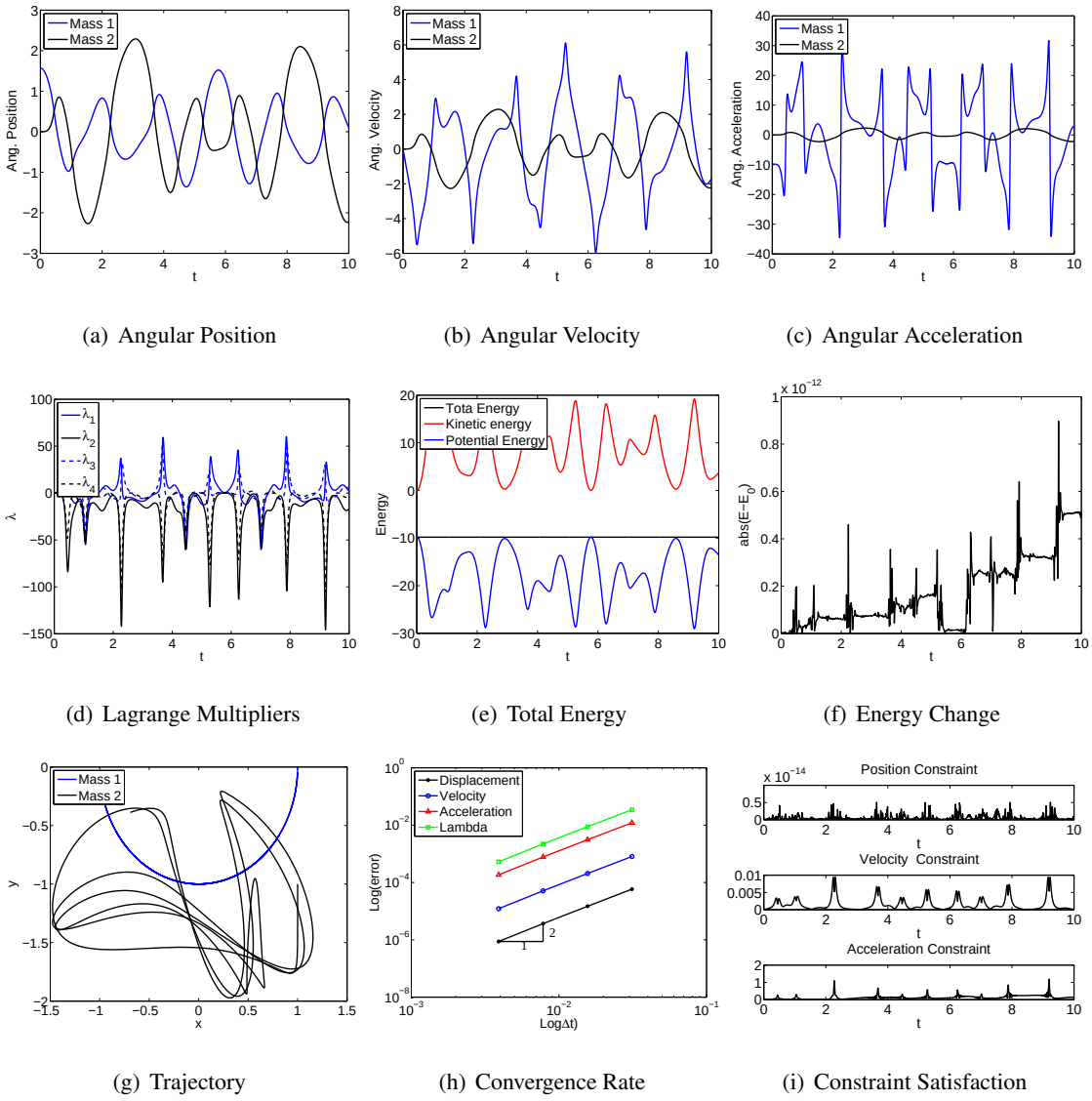


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.29: Double rigid pendulum: V0(1,1,0) - Index 1.

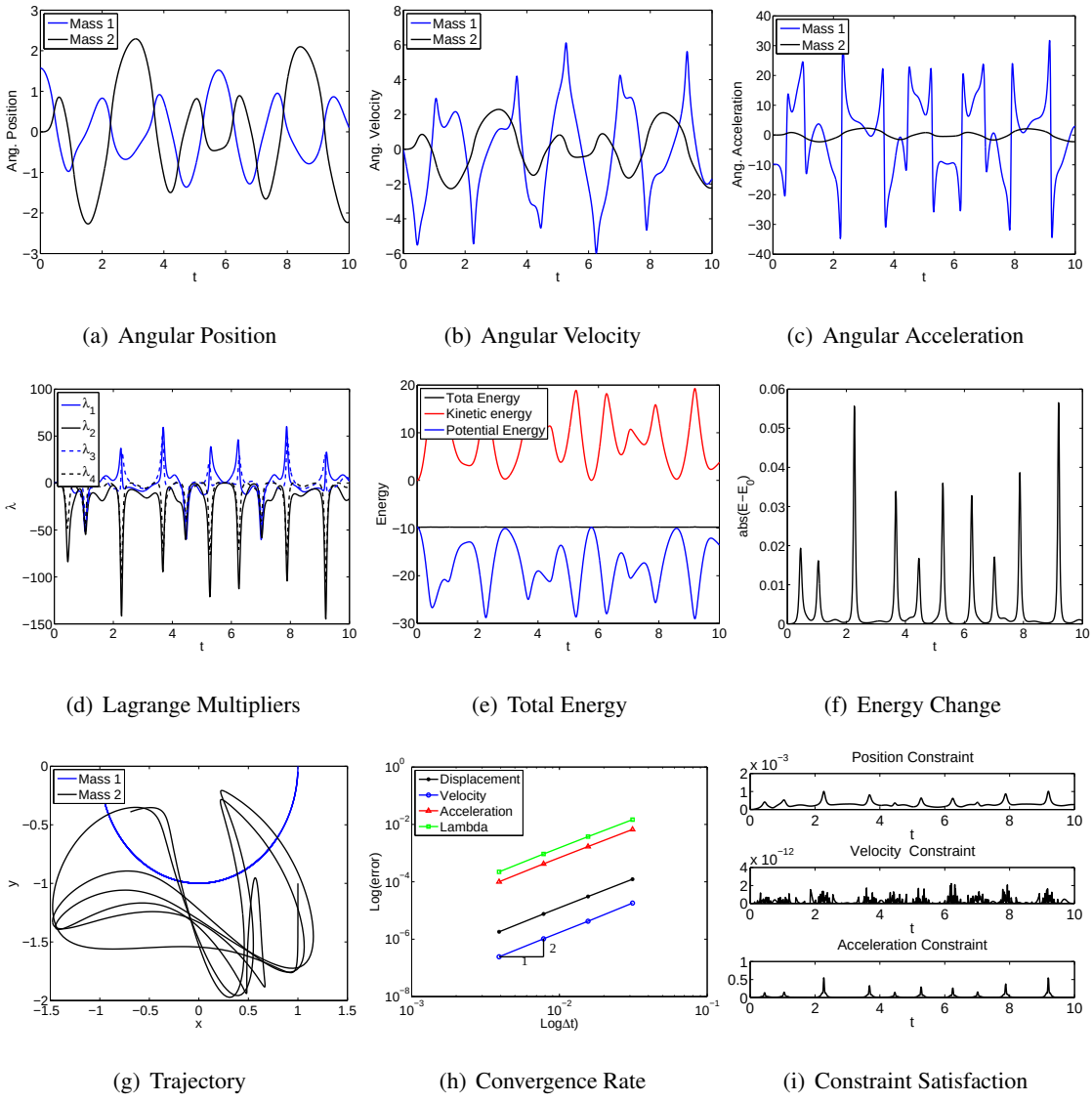


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.30: Double rigid pendulum: U0V0(1,1,1) - Index 3.

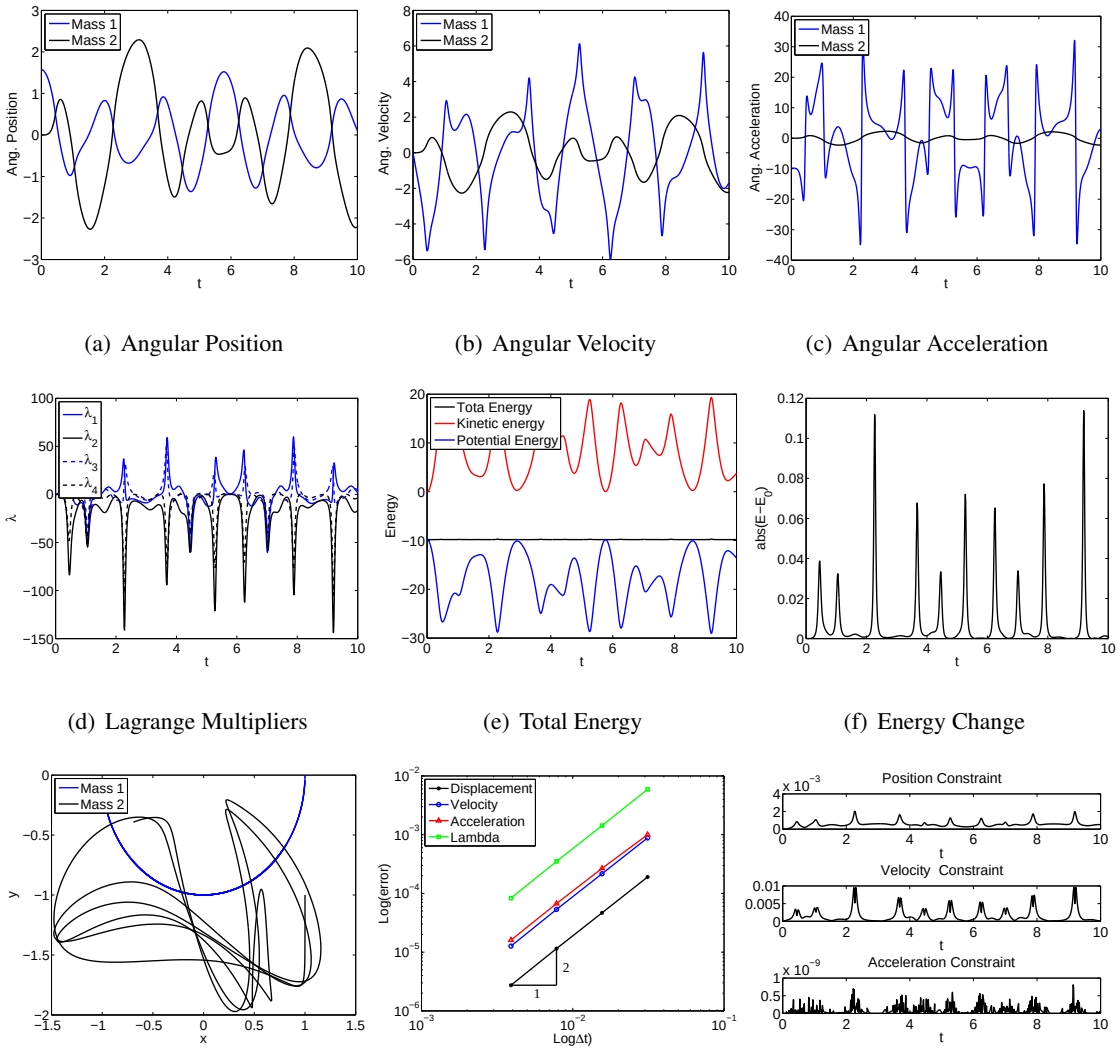


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.31: Double rigid pendulum: U0V0(1,1,1) - Index 2.



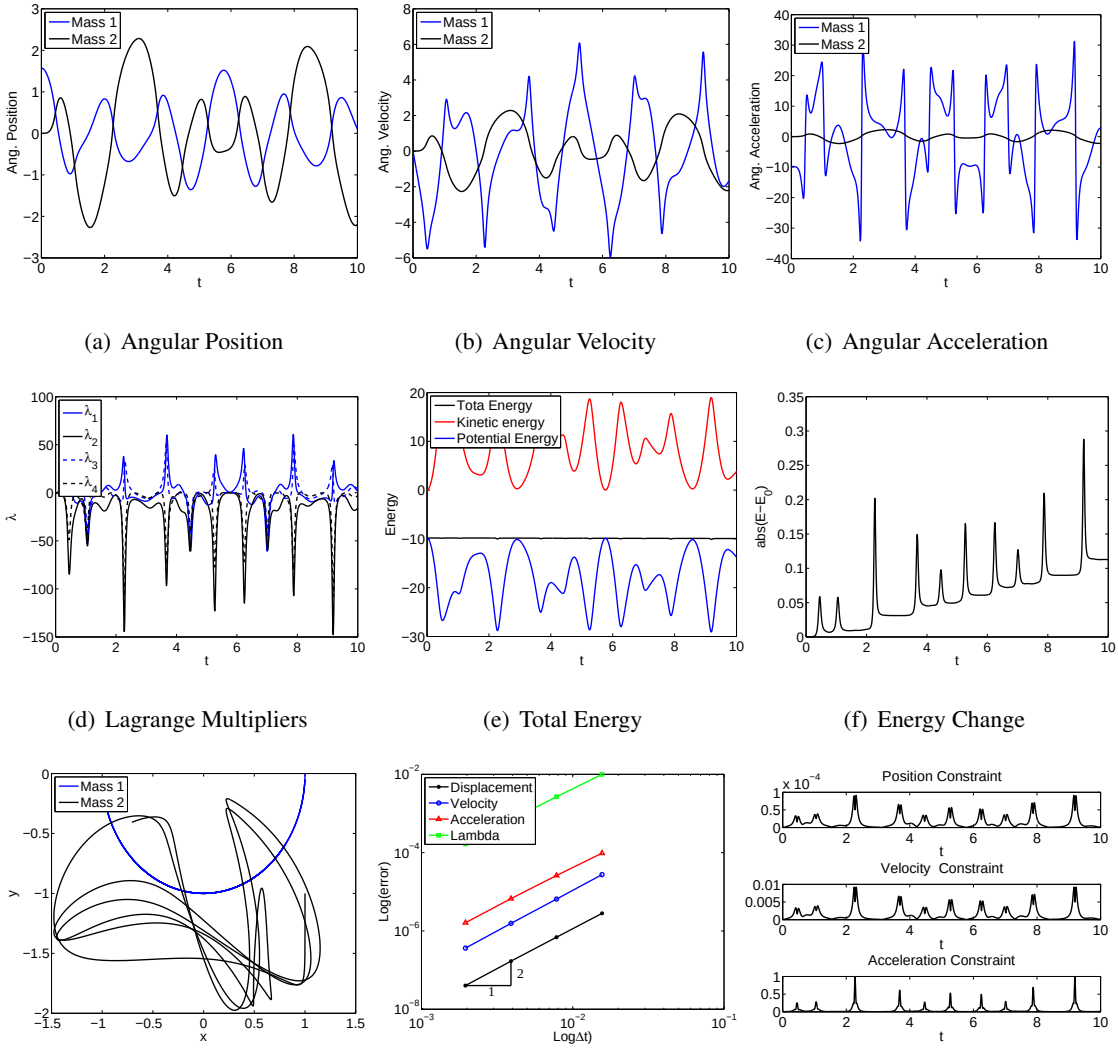
(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.32: Double rigid pendulum: U0V0(1,1,1) - Index 1.

Baumgarte's Method

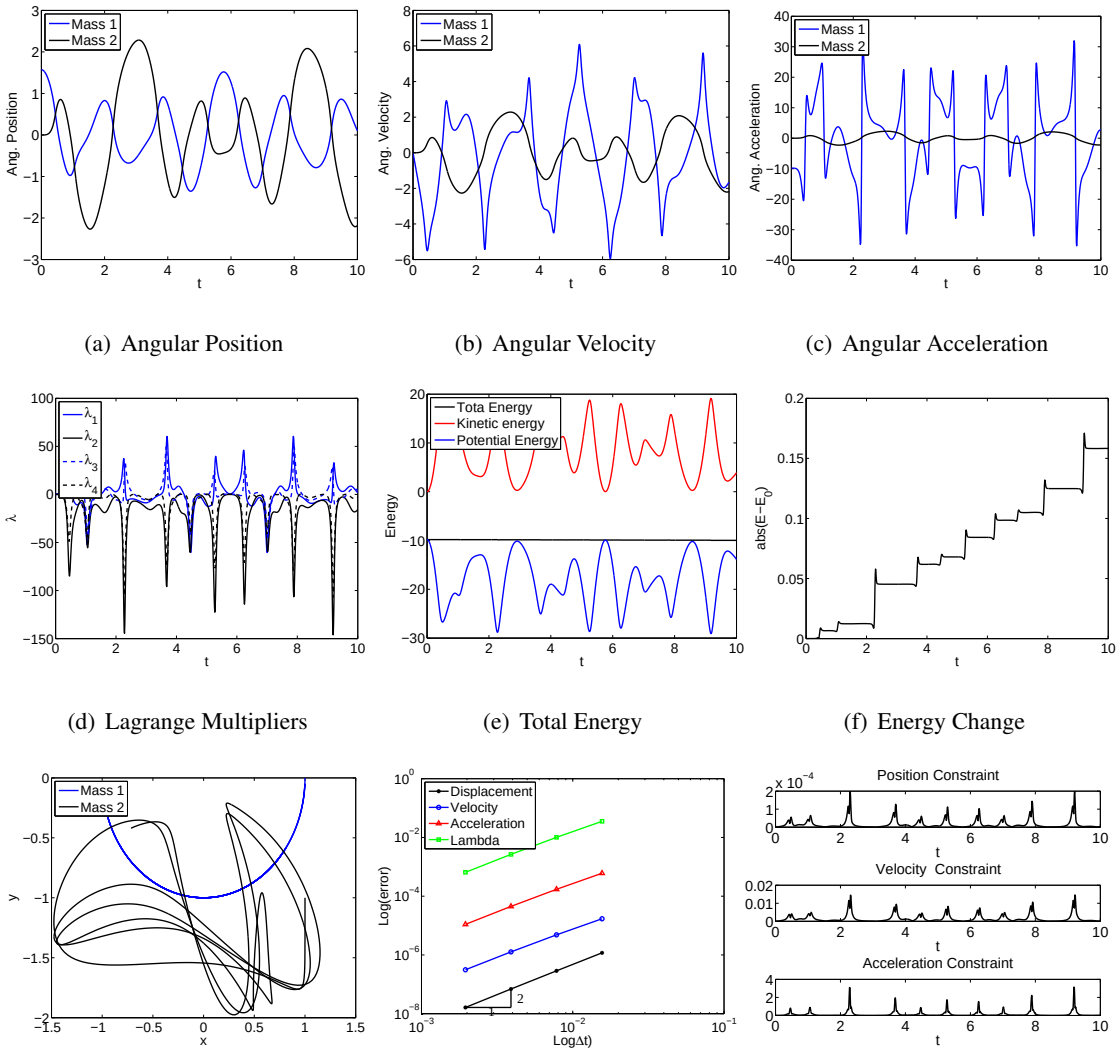


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.33: Double rigid pendulum with Baumgarte's method: U0(1,1,0).

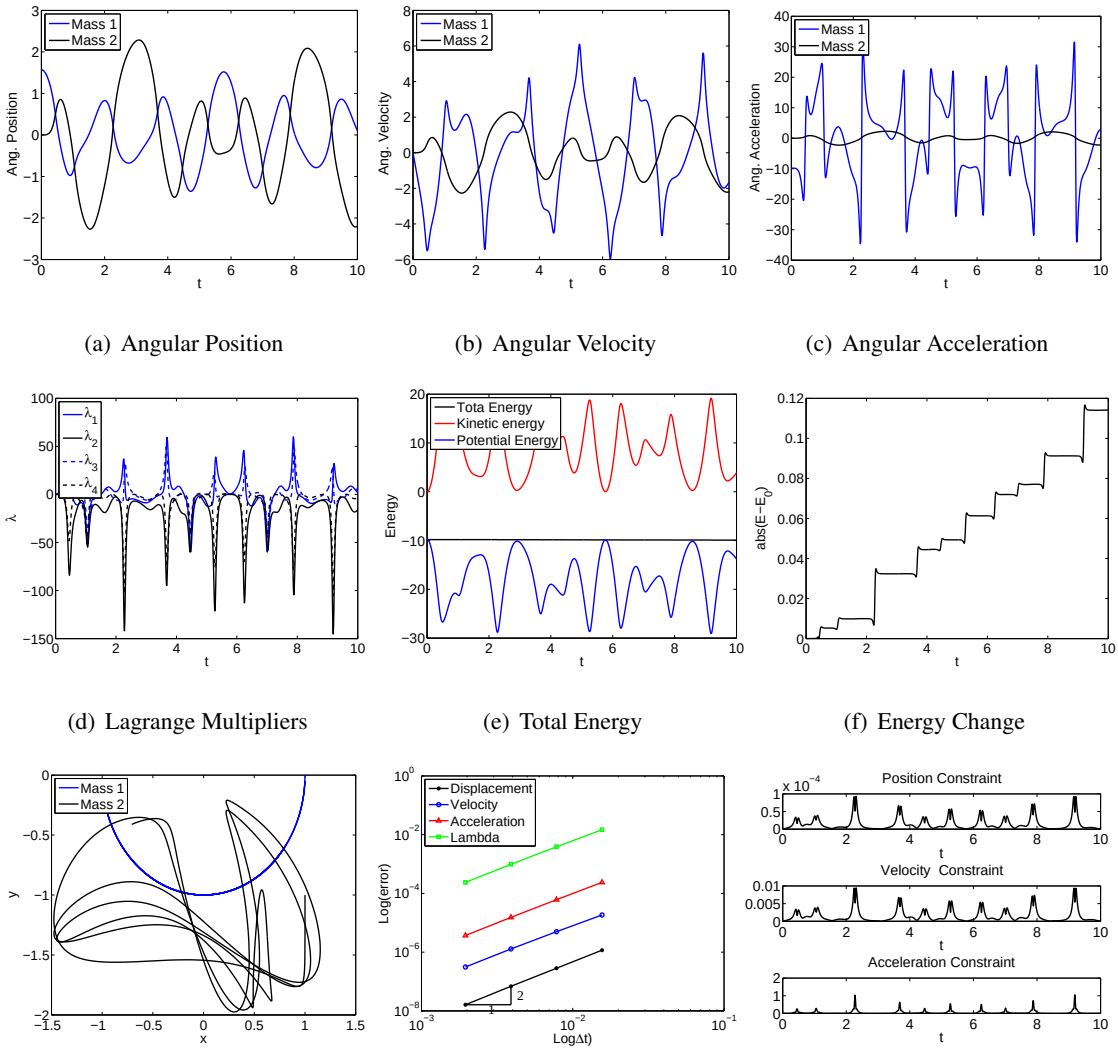


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.34: Double rigid pendulum with Baumgarte's method: V0(1,1,0).



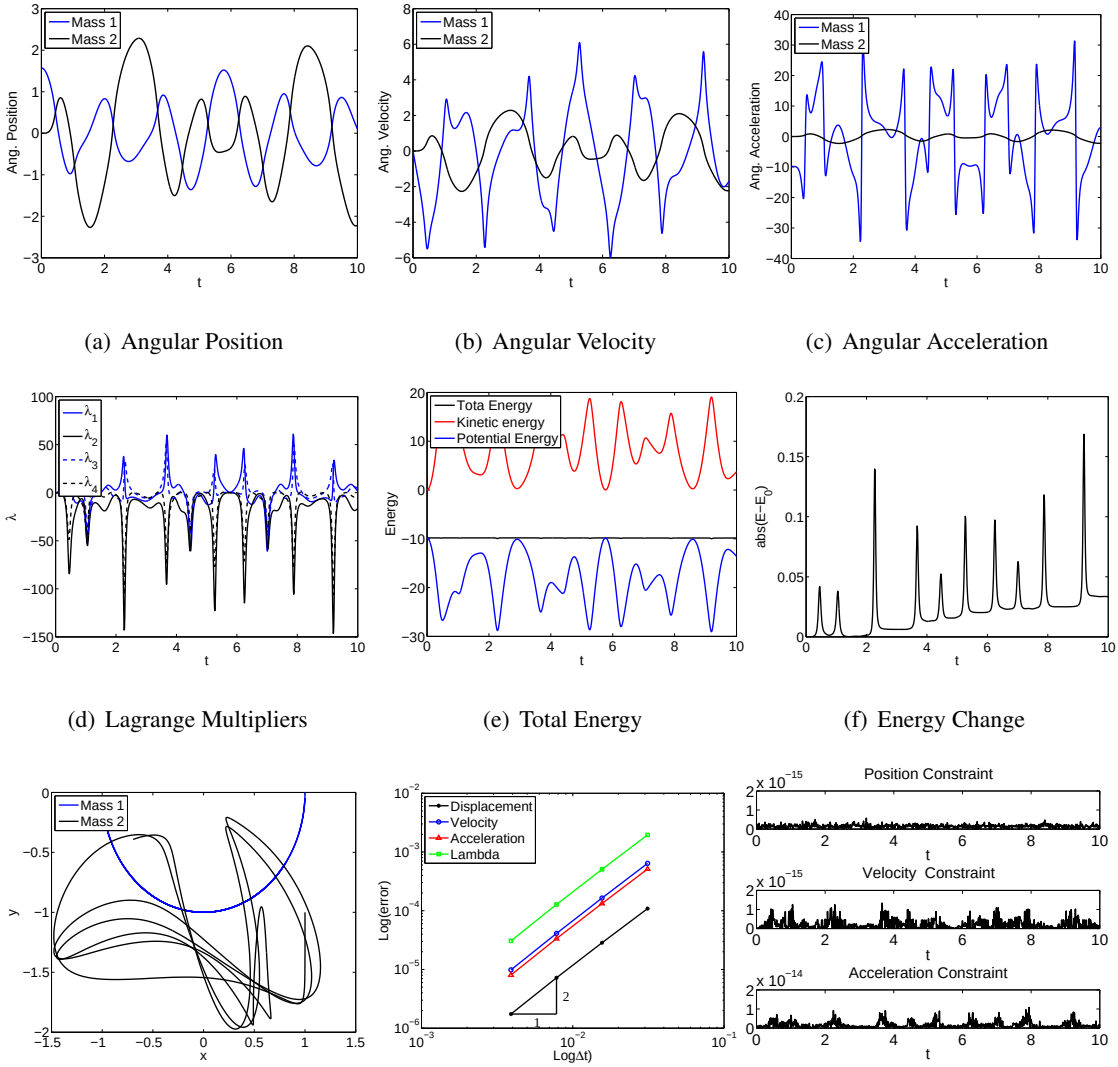
(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.35: Double rigid pendulum with Baumgarte's method: U0V0(1,1,1).

Projection Method

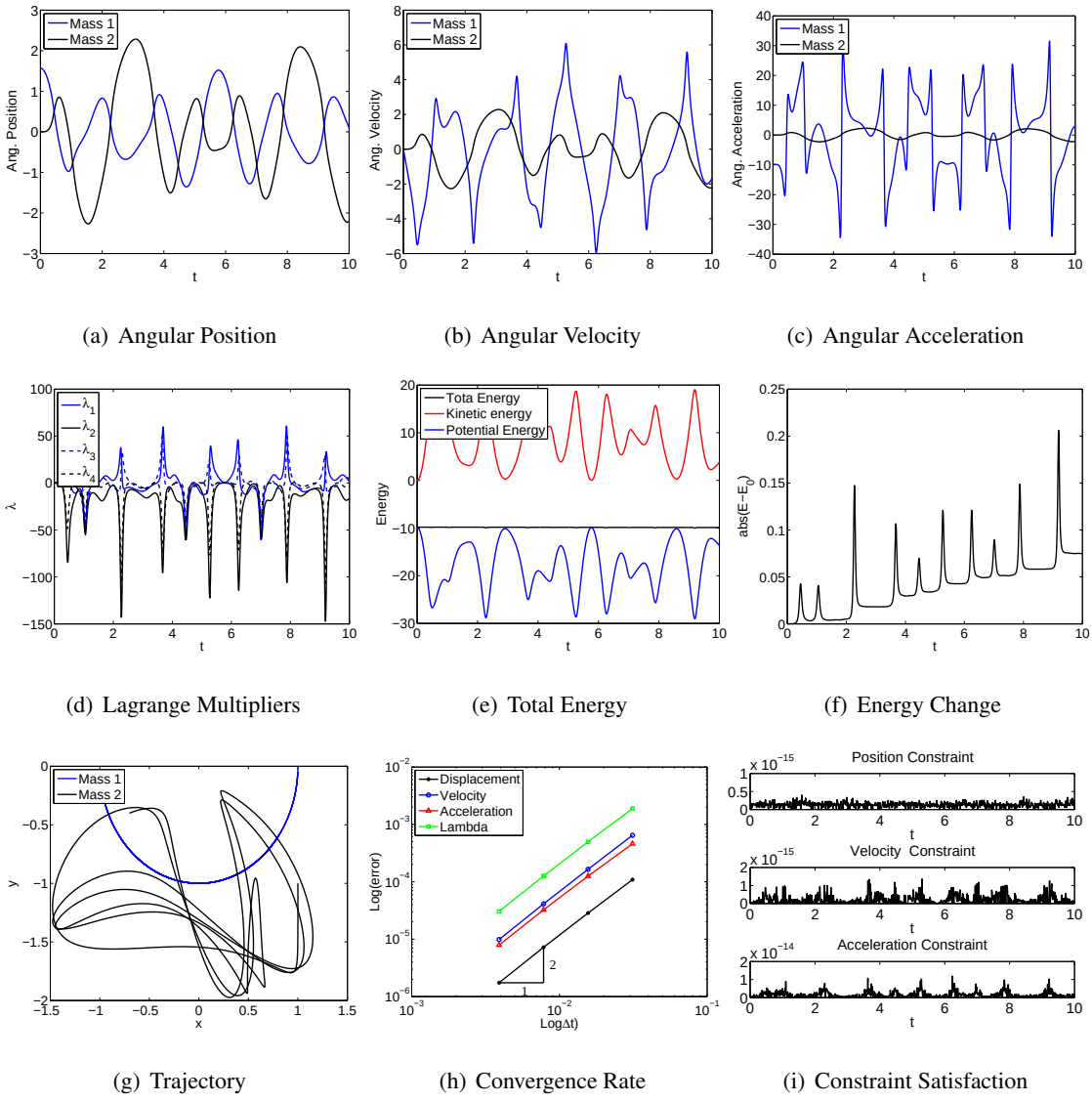


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.36: Double rigid pendulum with Projection method: U0(1,1,0) - Index 3.

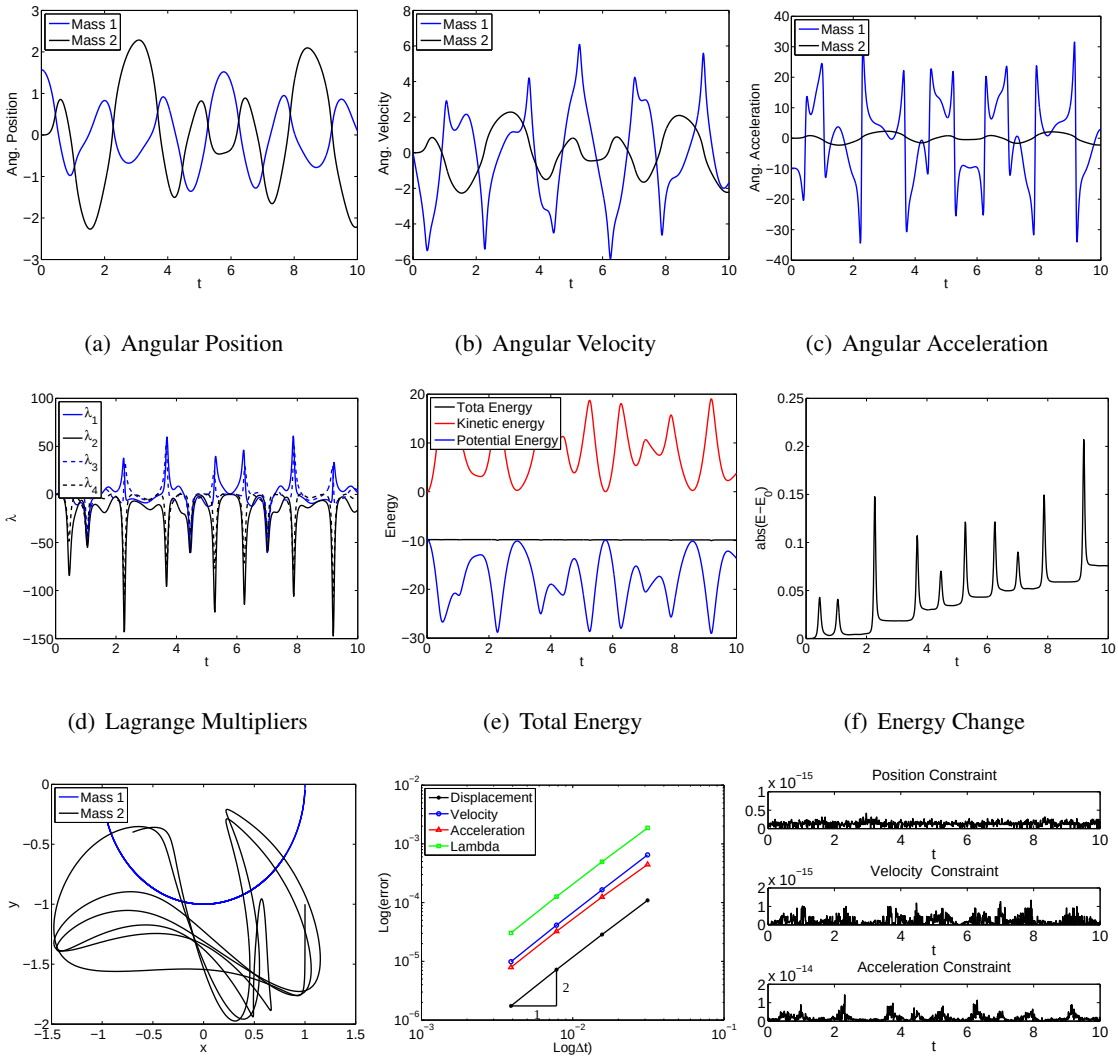


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.37: Double rigid pendulum with Projection method: U0(1,1,0) - Index 2.

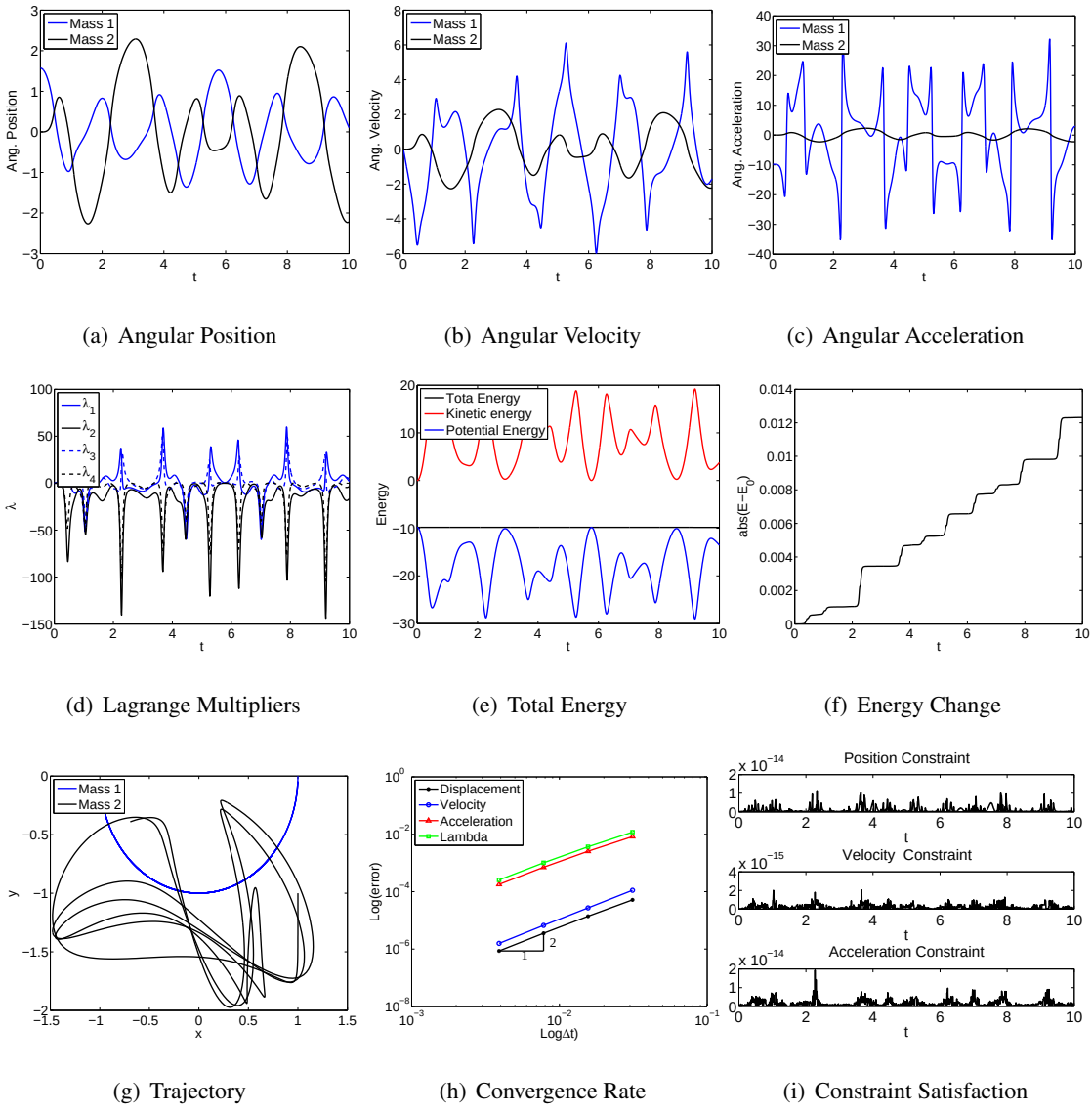


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.38: Double rigid pendulum with Projection method: U0(1,1,0) - Index 1.

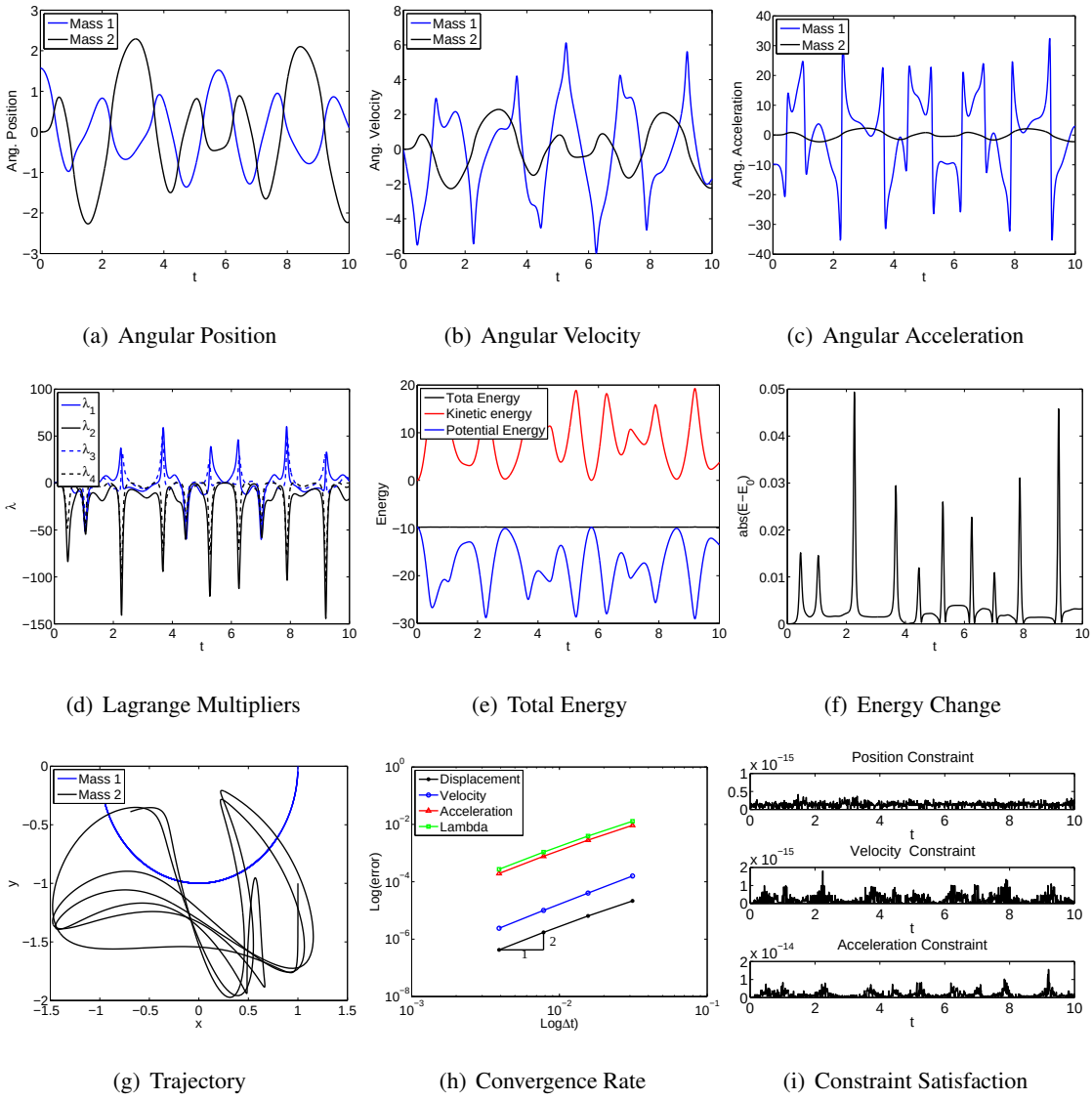


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.39: Double rigid pendulum with Projection method: V0(1,1,0) - Index 3.

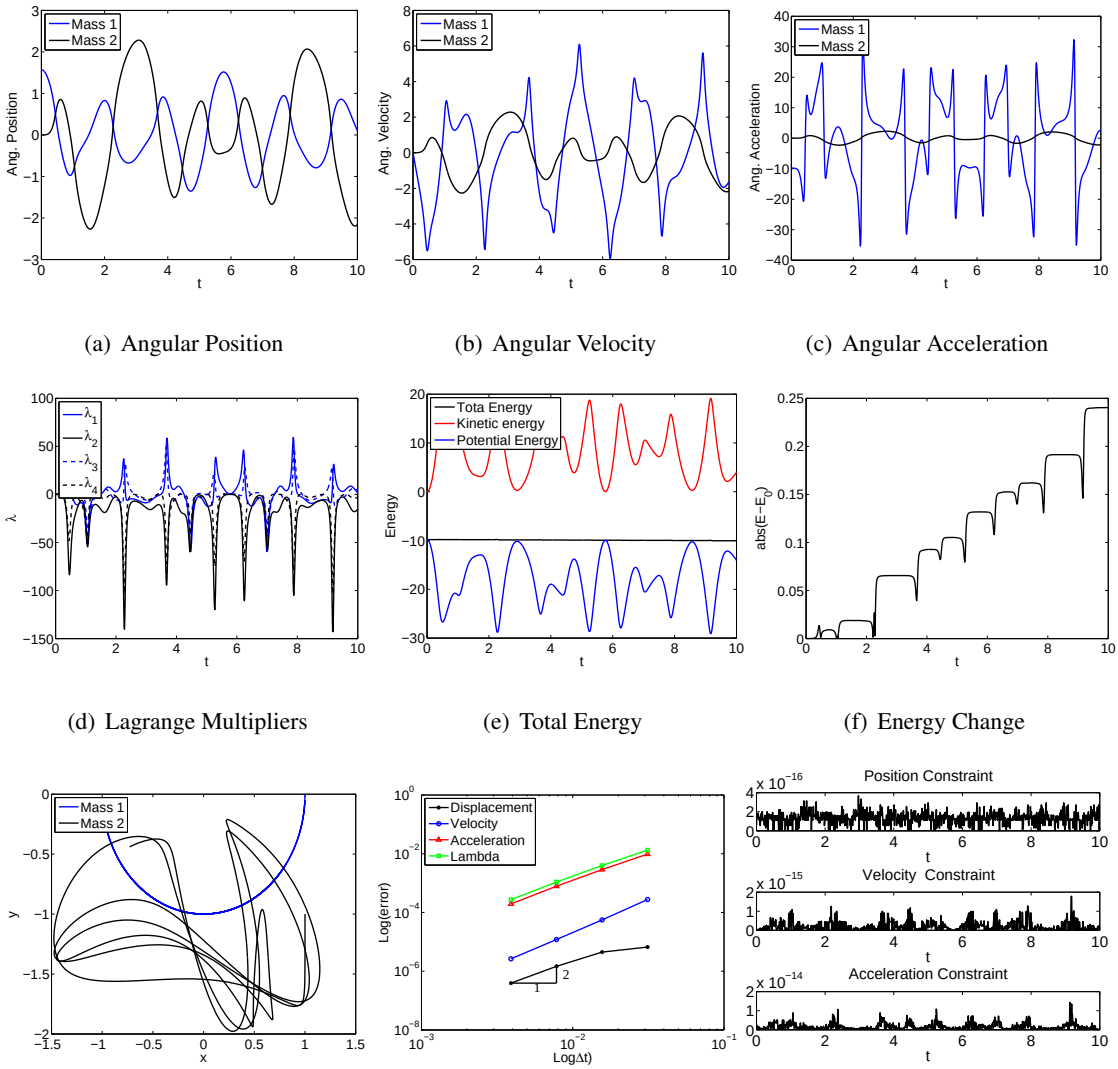


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.40: Double rigid pendulum with Projection method: V0(1,1,0) - Index 2.

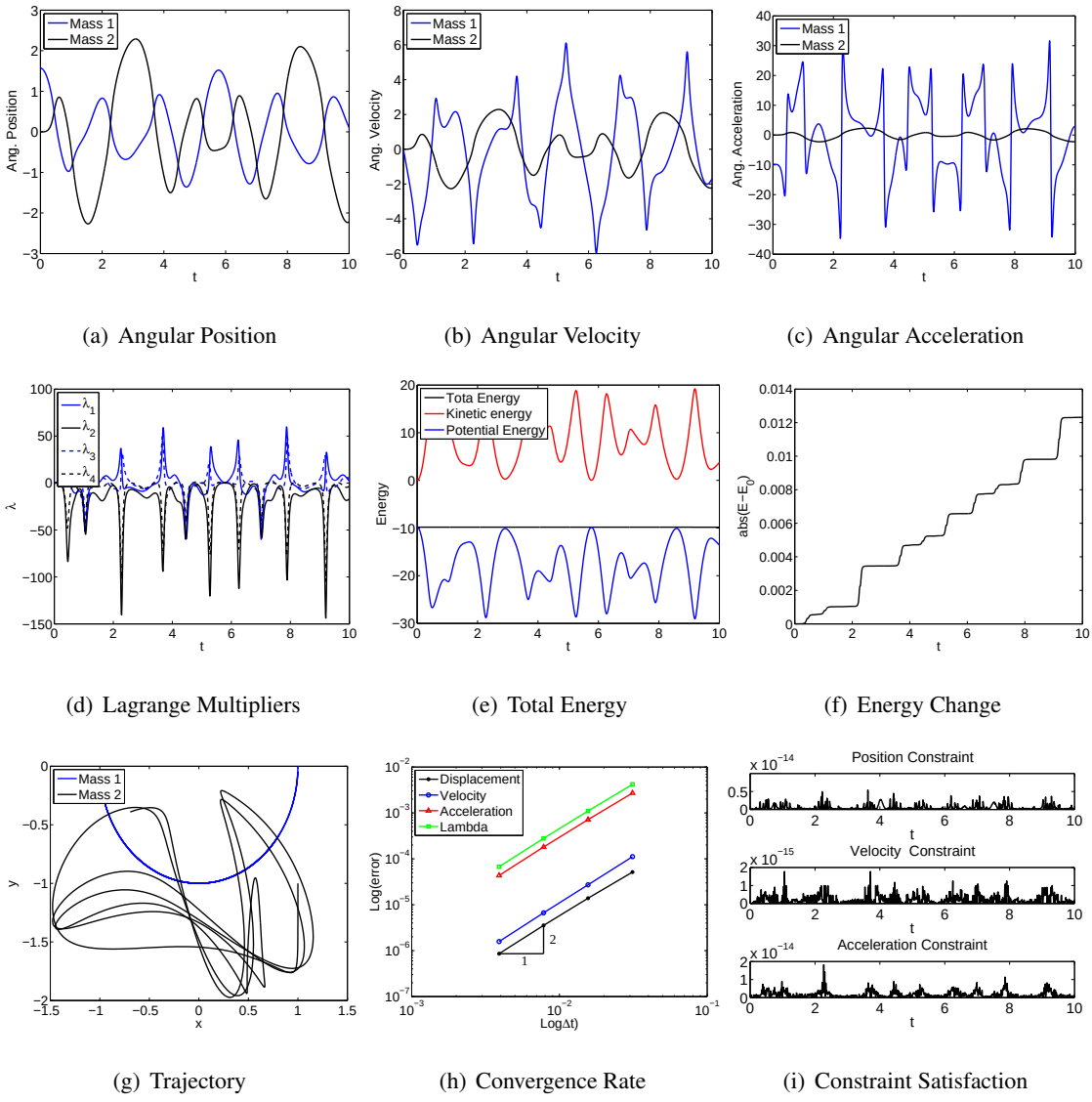


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.41: Double rigid pendulum with Projection method: V0(1,1,0) - Index 1.

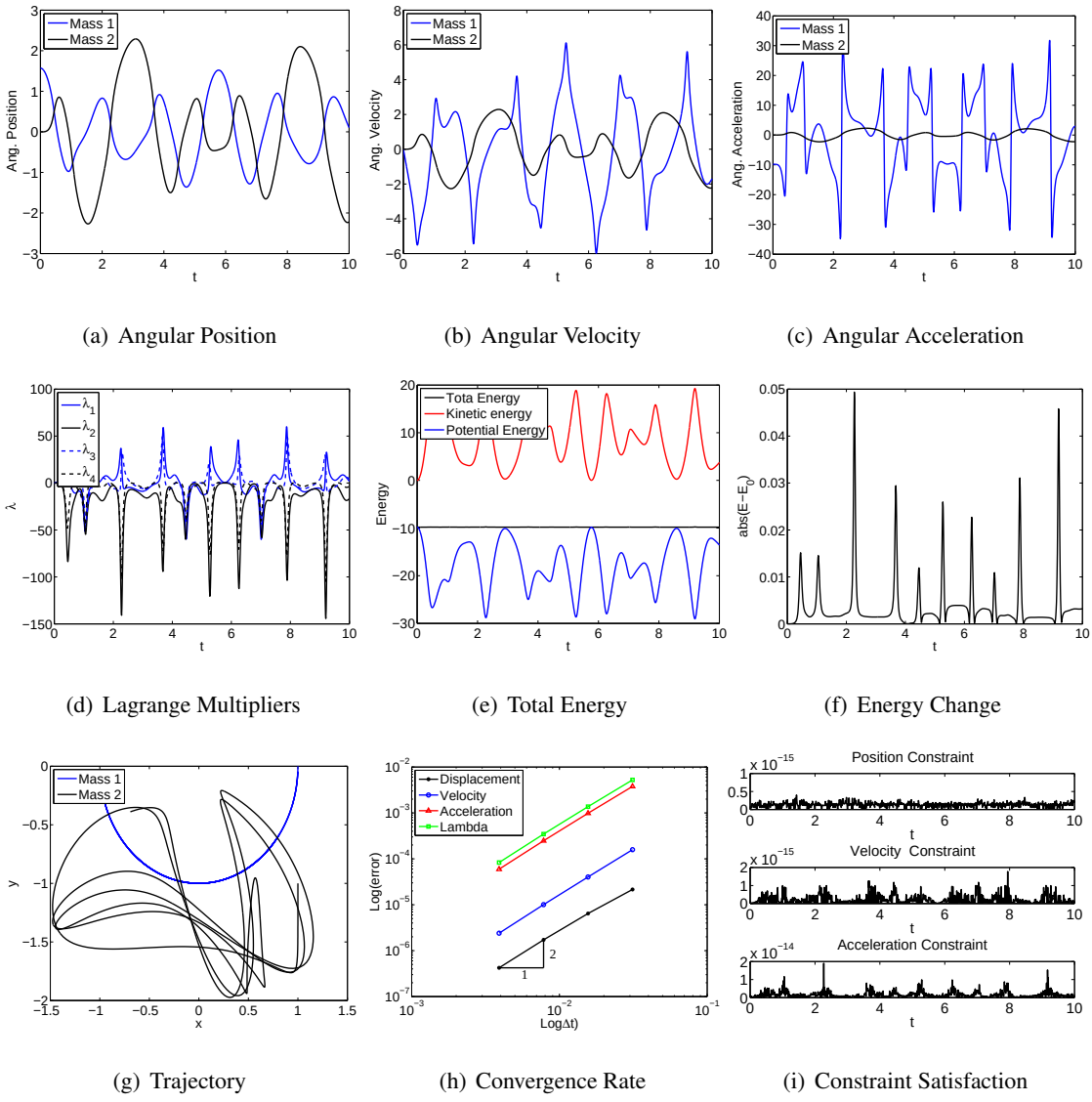


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.42: Double rigid pendulum with Projection method: U0V0(1,1,1) - Index 3.

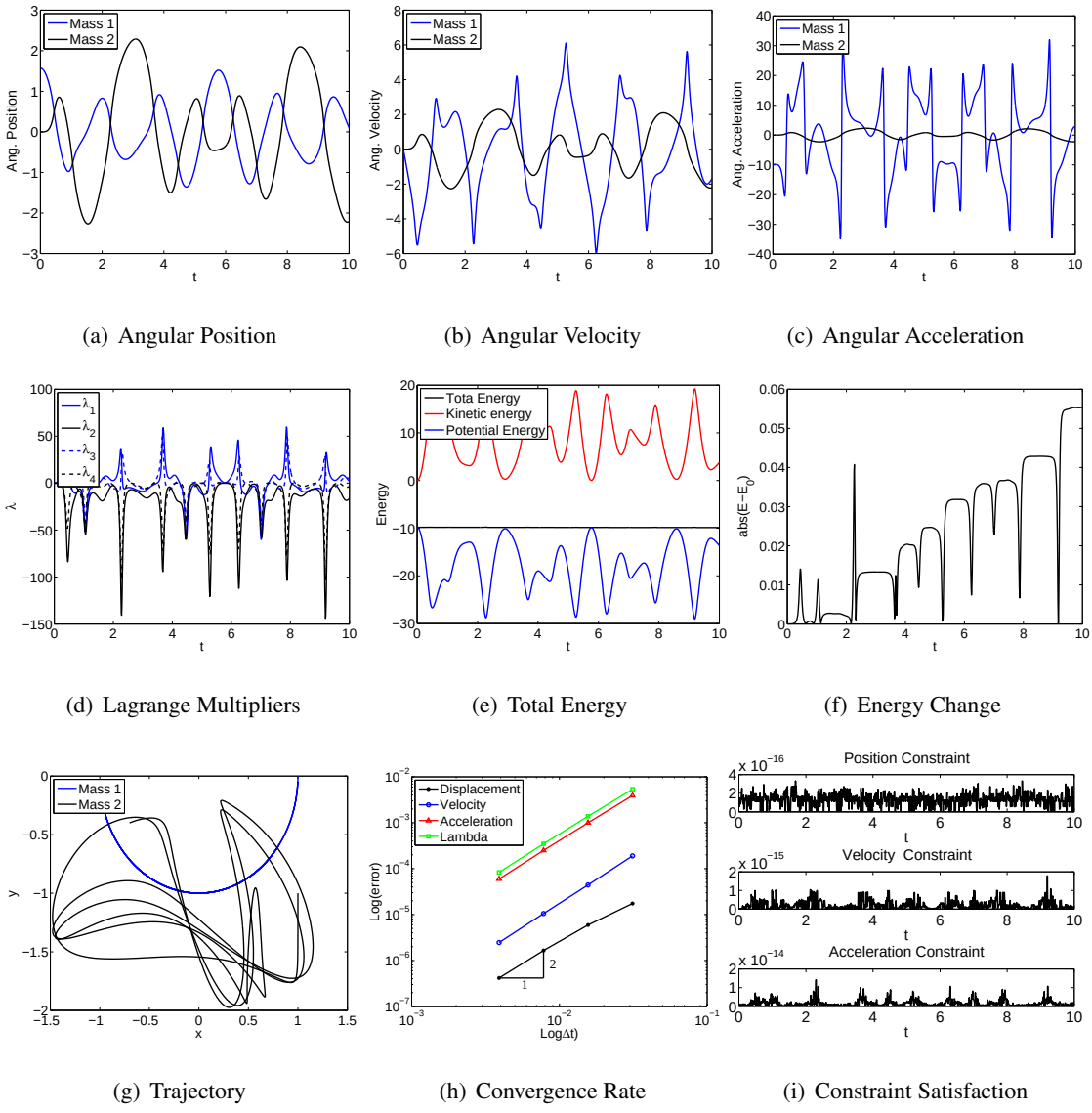


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.43: Double rigid pendulum with Projection method: U0V0(1,1,1) - Index 2.



(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.44: Double rigid pendulum with Projection method: U0V0(1,1,1) - Index 1.

4.1.3 Discussion

It is more meaningful to discuss the double pendulum.

Table 4.1: Comparison of double pendulum with $\Delta t = 0.01$

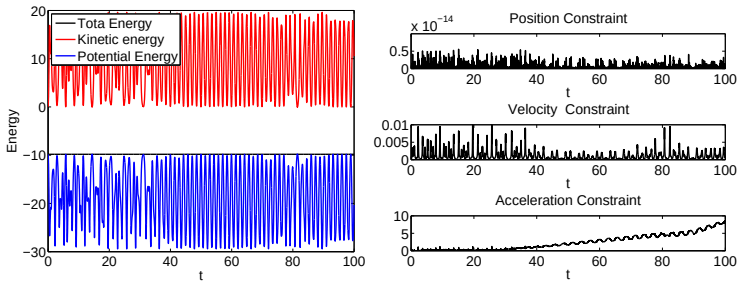
Index	Algorithm	Original	Baumgarte's methods	Projection methods
3	U0(1,1,0)	Oscillate	Stable	Stable
	V0(1,1,0)	Stable	Stable	Stable
	U0V0(1,1,1)	Stable	Stable	Stable
2	U0(1,1,0)	Stable	Stable	Stable
	V0(1,1,0)	Stable	Stable	Stable
	U0V0(1,1,1)	Stable	Stable	Stable
1	U0(1,1,0)	Stable	Stable	Stable
	V0(1,1,0)	Oscillate	Stable	Stable
	U0V0(1,1,1)	Stable	Stable	Stable

Table 4.2: Computation cost of double pendulum (iterations/Δt)

Index	Algorithm	Original	Baumgarte's methods	Projection methods
3	U0(1,1,0)	X	6.49	5.99
	V0(1,1,0)	6.22	7.44	6.19
	U0V0(1,1,1)	6.24	7.21	6.16
2	U0(1,1,0)	6.23	6.49	6.22
	V0(1,1,0)	6.85	7.44	6.85
	U0V0(1,1,1)	6.85	7.21	6.85
1	U0(1,1,0)	6.858	6.49	6.86

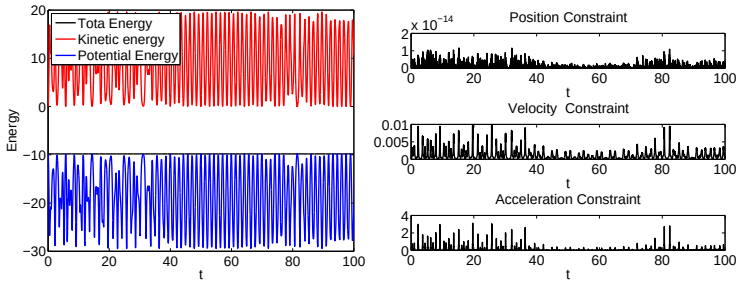
	V0(1,1,0)	7.77	7.44	7.77
	U0V0(1,1,1)	7.39	7.21	7.38

From the numerical result of the simple pendulum and double pendulum under 10 second simulation under Index 3, it can be seen that there are two fundamental sources of instability in simulation: the accelerations and the Lagrange multipliers. These values are seen to oscillate with increasing magnitude until the nonlinear iterations are no longer able to converge. Notice that though the acceleration and Lagrange multipliers oscillate violently the velocities and displacement are smooth and match exactly the results of stable solutions under Index 3. To reduce the instability, one needs stabilization techniques or reduce the Index. Although U0/V0(1,1,1) (mid point rule with end point acceleration) yields stable solution and exact energy conservation as well as reasonable dynamic response. However, under long term computation, acceleration and Lagrange multiplier under U0/V0(1,1,1) still oscillate and drift effect therefore occurs in acceleration constraint. In fact, only V0(1,1,0) yields stable solution for long term simulation under Index 3 DAE as shown in Fig. 4.45 and 4.45.



(a) Total Energy (b) Constraint Satisfaction

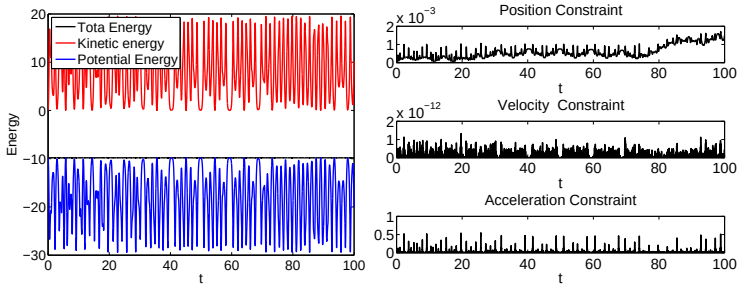
Figure 4.45: Double rigid pendulum with long time computation: U0V0(1,1,1) - Index 3



(a) Total Energy (b) Constraint Satisfaction

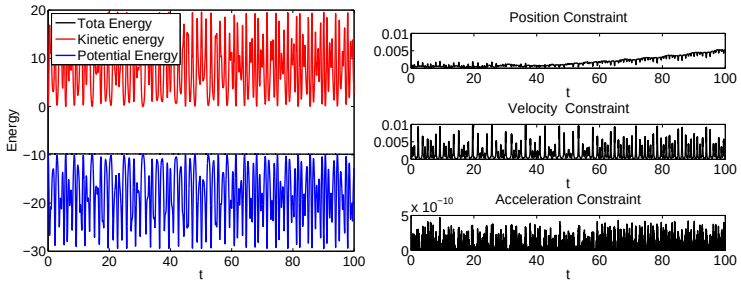
Figure 4.46: Double rigid pendulum with long time computation: V0(1,1,0) - Index 3

From the result of each scheme under index 2 or index 1 DAEs with each scheme, solution for each scheme are much stable except V0(1,1,0). However, for long term simulation, drift effect still occurs for Index 1 simulation as shown in Fig. 4.47 and Fig. 4.48



(a) Total Energy (b) Constraint Satisfaction

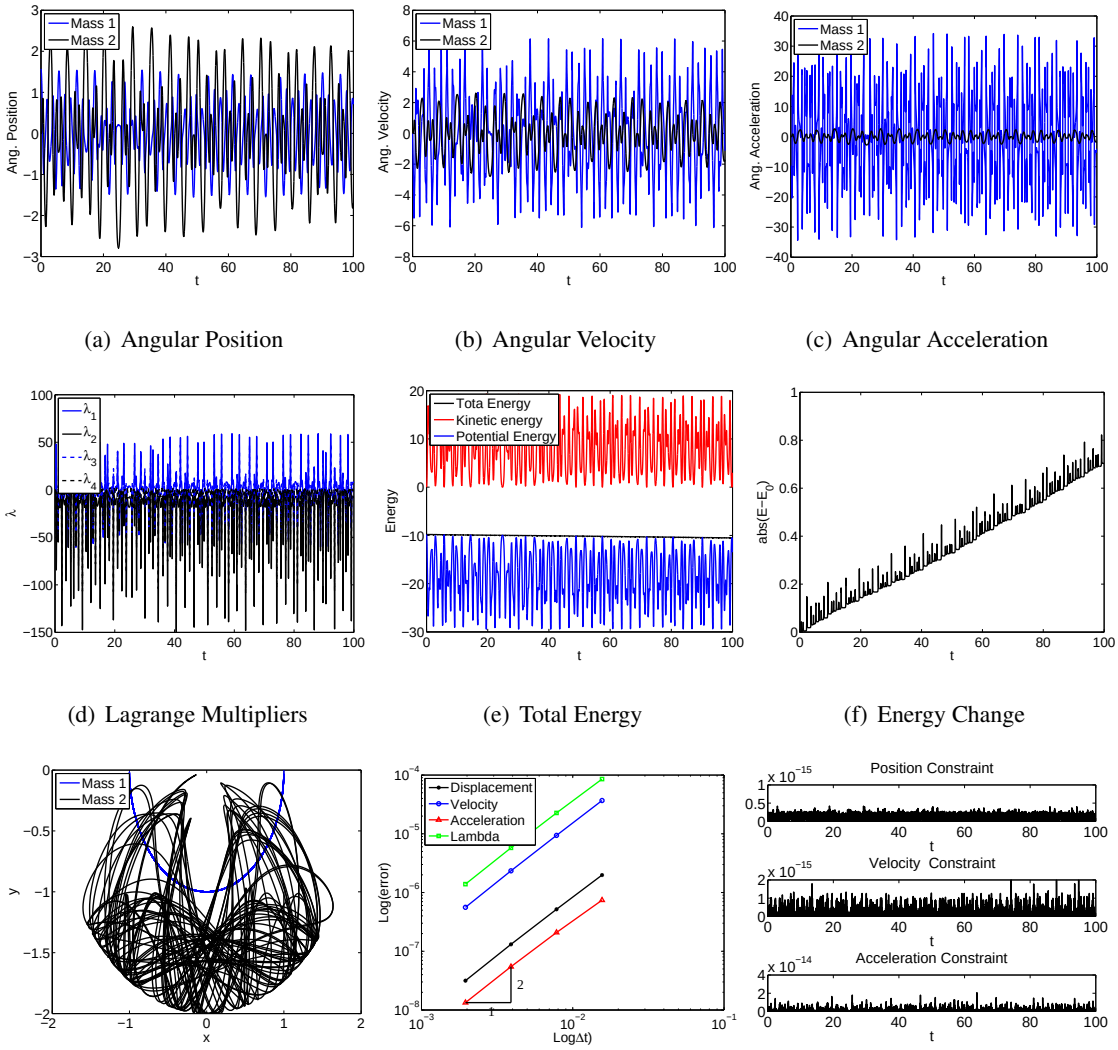
Figure 4.47: Double rigid pendulum with long time computation: U0(1,1,0) - Index 2



(a) Total Energy (b) Constraint Satisfaction

Figure 4.48: Double rigid pendulum with long time computation: U0(1,1,0) - Index 1

That is, for rigid body dynamics, except V0(1,1,0) in index 3 case, stabilization techniques (constraint satisfaction) are required to eliminate the drift effect as well as to ensure stability. From Fig. 4.33 to Fig. 4.44, one can see that although both methods satisfy all constraints with good quality, only projection method satisfies the constraint exactly with machine precision. Baumgarte's methods satisfy the constraint only up to 10^{-4} in position level and also from Table. 4.2, the cost for Baumgarte's is greater than projection method generally. Also, selecting the Baumgarte's parameters is based on experience. As a result, projection method is better for constraint satisfaction. One issue for projection method is that although we satisfied the constraint exactly, exact energy conservation does not exist since we project the original position, velocity and acceleration onto a new constraint manifold. This causes the original equation of motion to be not exactly satisfied and therefore the we cannot conserve energy exactly.

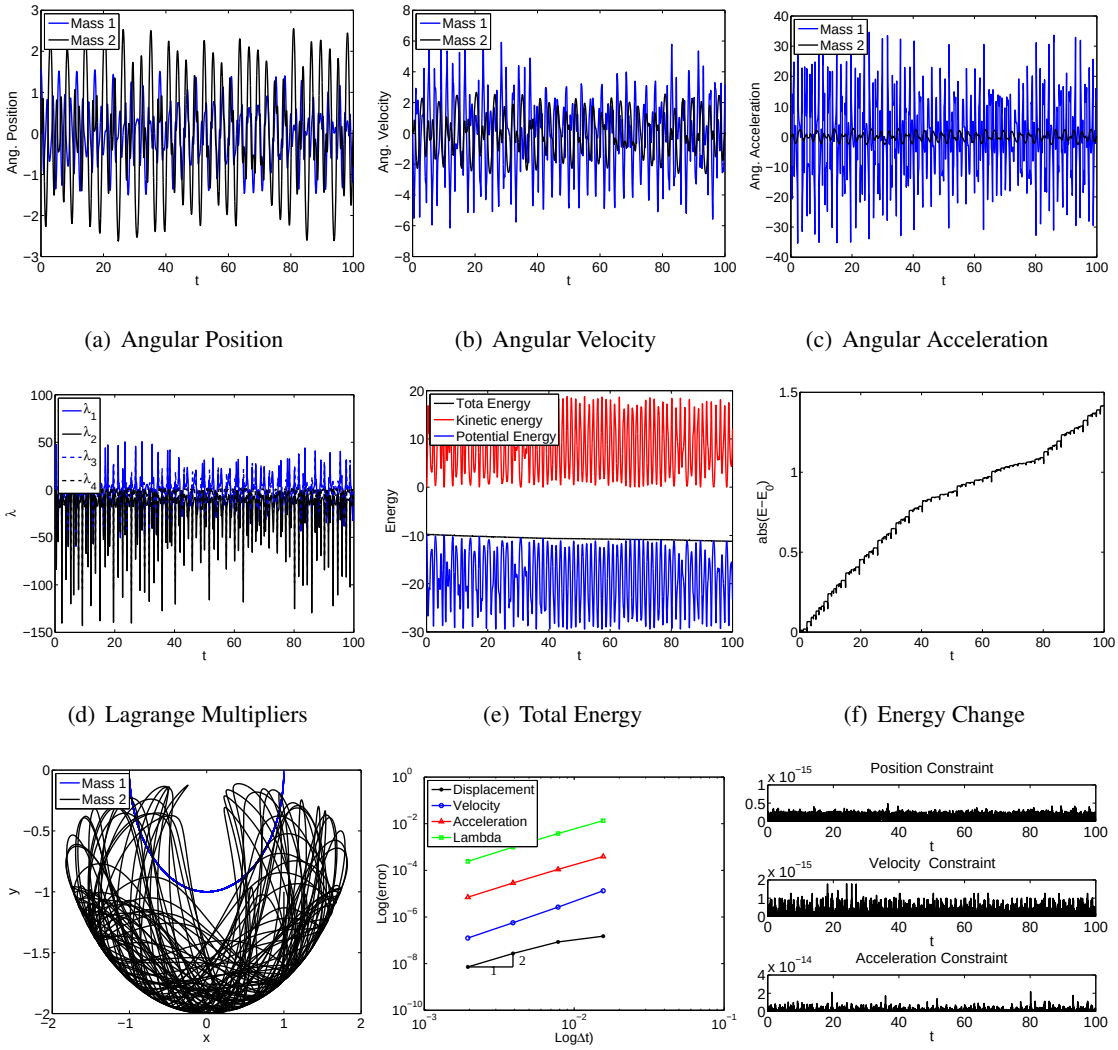


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.49: Double rigid pendulum with Projection method: U0(1,1,0) - Index 1 - Long time simulation

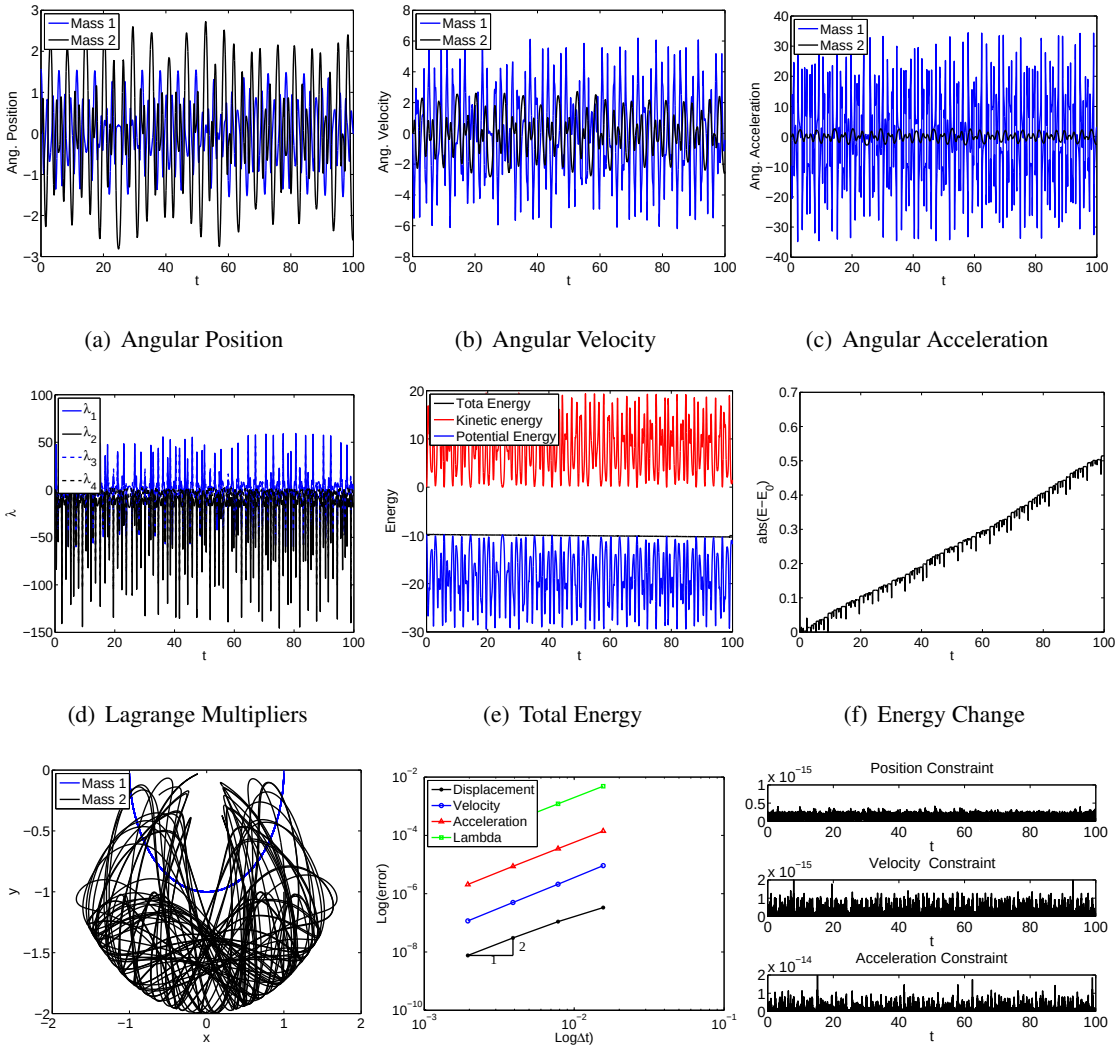


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.50: Double rigid pendulum with Projection method: V0(1,1,0) - Index 1 - Long time simulation



(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.51: Double rigid pendulum with Projection method: U0V0(1,1,1) - Index 1 - Long time simulation

Finally, convergence plots are required to prove that stabilization techniques are processing correctly. For Baumgarte methods, the shift coefficient $\gamma = \Lambda_6 W_1 - W_1$ ensures the second order time accuracy. For projection method, the post processing is nothing to do with time integration. However, to ensure the satisfaction of equation of motion under machine precision it is required to obtain Lagrange multiplier λ_{n+1} by:

$$\lambda_{n+1} = (\mathbf{C}(u_{n+1})\mathbf{C}^T(u_{n+1}))^{-1}(\mathbf{C}(u_{n+1})(\mathbf{F}_{n+1} - \mathbf{M}a_{n+1})) \tag{4.38}$$

and λ_{n+1} has to be shifted back to correct time level. Without this step, an order reduction will happen on time level accuracy for λ.

4.2 Flexible Body Dynamics

Next these methods were tested on systems which include rigid/flexible elements; the most practical and general type of system in real-world applications. The same solution space of each algorithms and stabilization techniques was investigated for IRF/ANCF-S/FRF approach in different flexible element type (Elastic bar, Euler-Bernoulli beam, and Timoshenko beam). Two examples of single and double pendulum are studied.

4.2.1 Single Pendulum

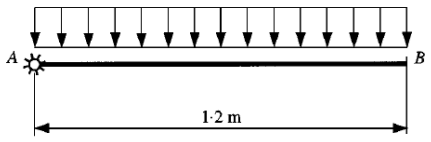


Figure 4.52: Single flexible pendulum problem from[58].

In the demonstration of single pendulum, unlike the rigid bar example, the bar can be segmented into finite elements by IRF/ANCF-S/FRF as well as the constraint formulation. As a result, the coordinates for different formulation are different (see Eq. (3.86) to Eq. (3.130)). Also, nonlinearity in each matrix and each element are also different based on the formulation. This problem is discussed in great detail in [58], with pertinent information repeated below.

In general the problem is of the form:

$$\begin{aligned} \mathbf{M}a + \mathbf{F}_{int} + \mathbf{C}^T(u)\lambda &= \mathbf{F} \\ \mathbf{\Phi}(u) &= 0 \end{aligned} \tag{4.39}$$

The problem consists of different degrees of freedom and which element is employed and two Lagrange multipliers (see Appendix). Because there is only one planar hinge in this system, the constraint equation for all formulations is the same as follows:

$$\mathbf{\Phi}(u) = \begin{bmatrix} r_1 \\ \\ r_2 \end{bmatrix} \tag{4.40}$$

where r_1 and r_2 represent the displacement in global x and y direction at the node connecting to the hinge. Hence, the first derivative of geometry constraint yields the velocity constraint:

$$\dot{\mathbf{\Phi}}(v) = \begin{bmatrix} \dot{r}_1 \\ \\ \dot{r}_2 \end{bmatrix} \tag{4.41}$$

and the acceleration constraint is:

$$\ddot{\mathbf{\Phi}}(a) = \begin{bmatrix} \ddot{r}_1 \\ \\ \ddot{r}_2 \end{bmatrix} \tag{4.42}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \frac{\partial \mathbf{\Phi}(u)}{\partial u} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 0 & \cdots \end{bmatrix} \tag{4.43}$$

where the number of zero components extends to the number of the coordinates. In all simulations below the problem parameters are taken as:

- Length of the body: $L = 1.2$ m
- Cross-section area: $A = 0.0018$ m^2
- Density: $\rho = 5540$ Kg/m^3
- Elastic Modulus: $E = 0.7 \times 10^6$ Pa
- Second moment of area: $I = 1.215 \times 10^{-8}$ m^4
- Gravity: 9.81 m/s^2

The mass matrix, stiffness matrix, and external force for different elements are shown in the Appendix. Also, the initial condition for this problem are set that the body is horizontal and the origin is constrained by a hinge, subjected to body force (gravity force) as shown in Fig. 4.52. The initial velocity are all zeros.

The Jacobian matrix under Index 1, 2, and 3 for the single field form follow as:

Index 3:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.44}$$

Index 2:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.45}$$

Index 1:

for first time step ($n = 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \frac{1}{1-\gamma}\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.46}$$

and next every time step ($n > 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ (1+\gamma)\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.47}$$

where γ is the time shift:

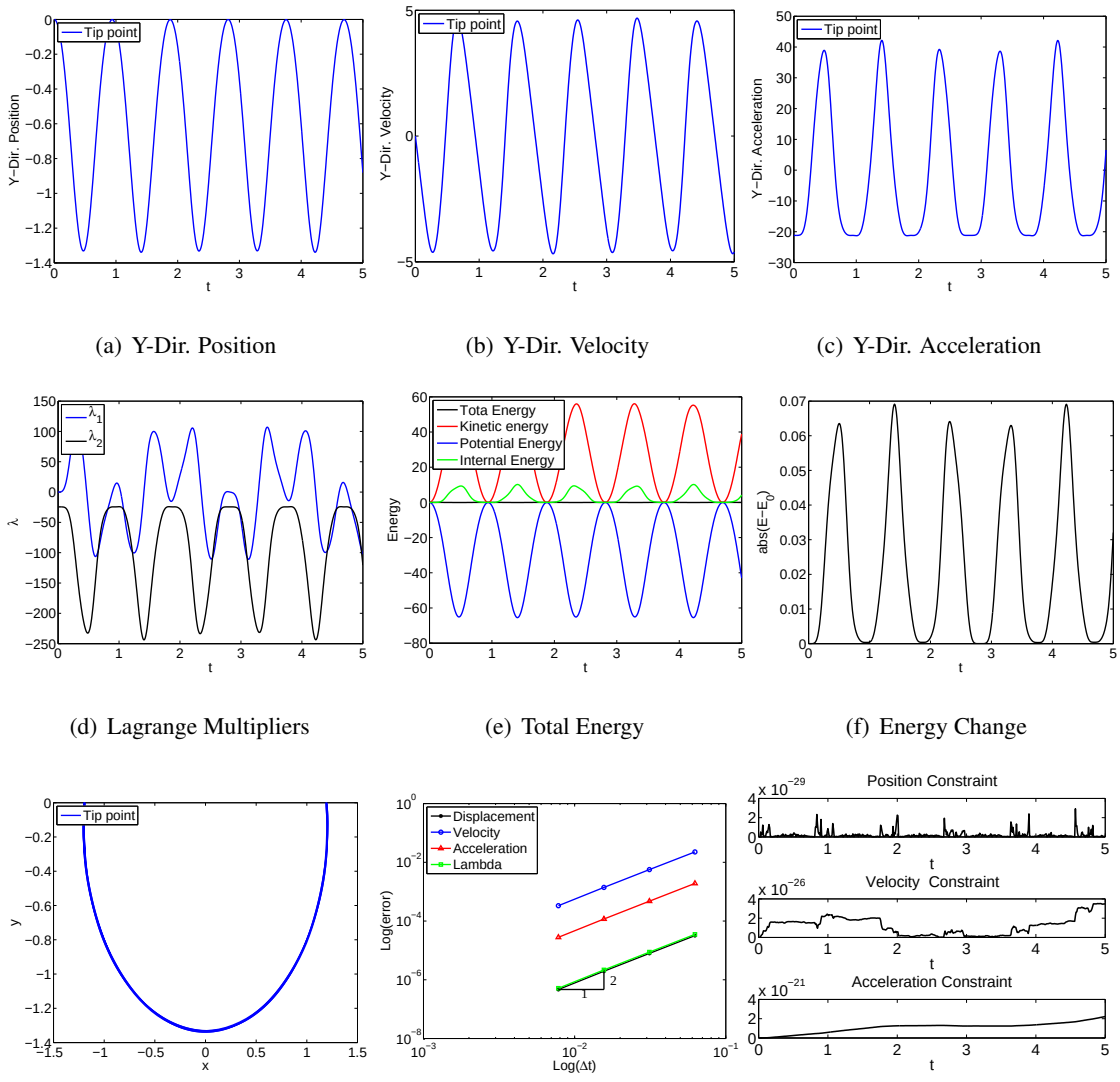
$$\gamma = \Lambda_6 W_1 - W_1 \tag{4.48}$$

and $\mathbf{J}_{C\lambda}(u, \lambda)$ is a zero matrix since the constraint equation is linear. Also, $\mathbf{K}_T$ and $\mathbf{M}_T$ are the tangent stiffness matrix and mass matrix as follows:

$$\begin{aligned} \mathbf{M}_T &= \frac{\partial \mathbf{M}(u)a}{\partial u} \\ \mathbf{K}_T &= \frac{\partial \mathbf{F}_{int}}{\partial u} \end{aligned} \tag{4.49}$$

The problem was run using a-form representation of the single field form algorithm. The time step used in all simulations was $\Delta t = 0.001$, the nonlinear iteration tolerance is 10^{-8}. The simulation duration was first set to 5 seconds, through not all cases were able to provide a stable solution. Also, three kinds of elements will be analyzed: Elastic Bar element(Bar), Euler-Bernoulli Beam(EB), and Timoshenko Beam(TB).

Elastic Bar

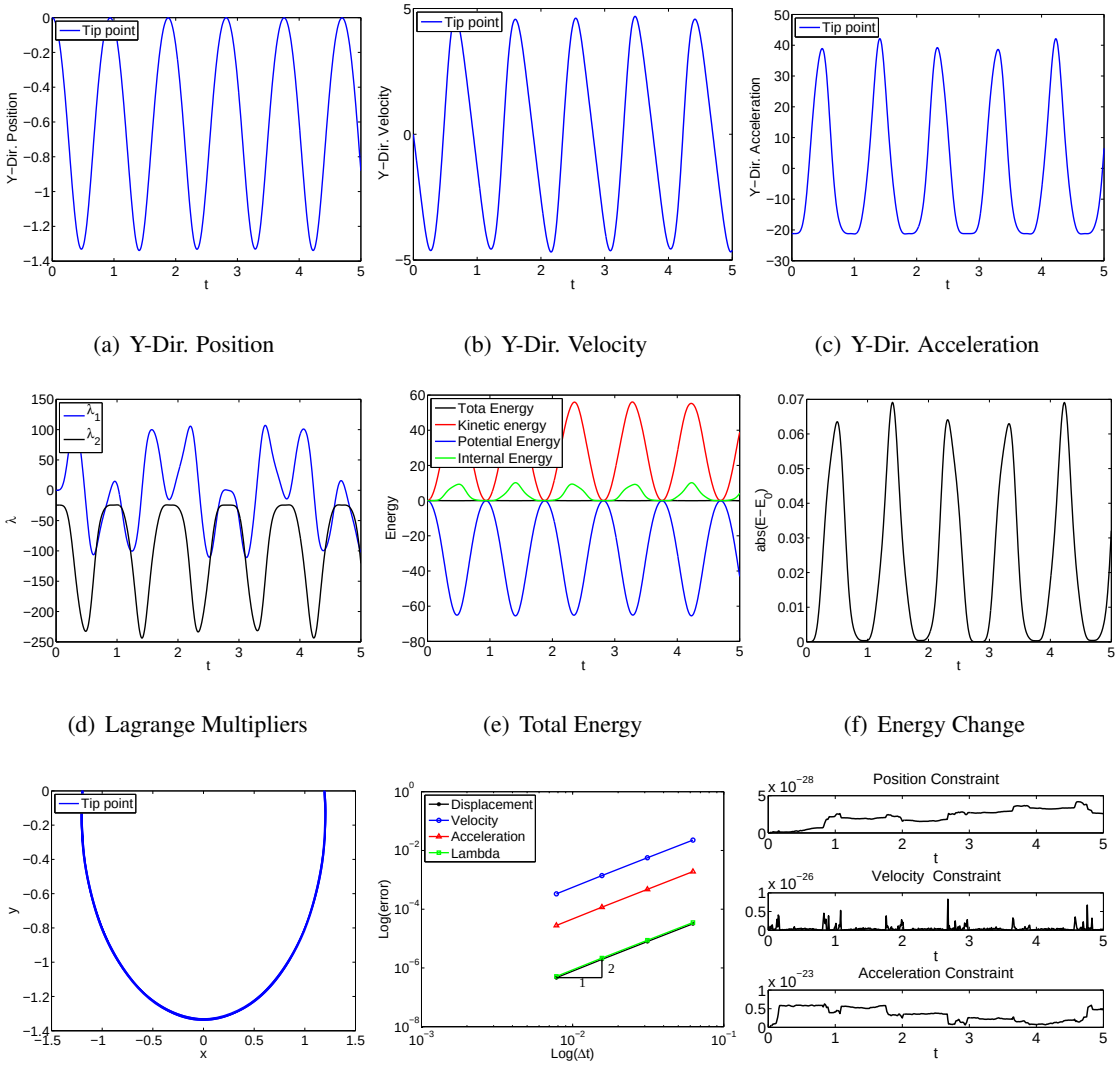


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.53: Single pendulum with bar element in IRF: U0(1,1,0) - Index 3.

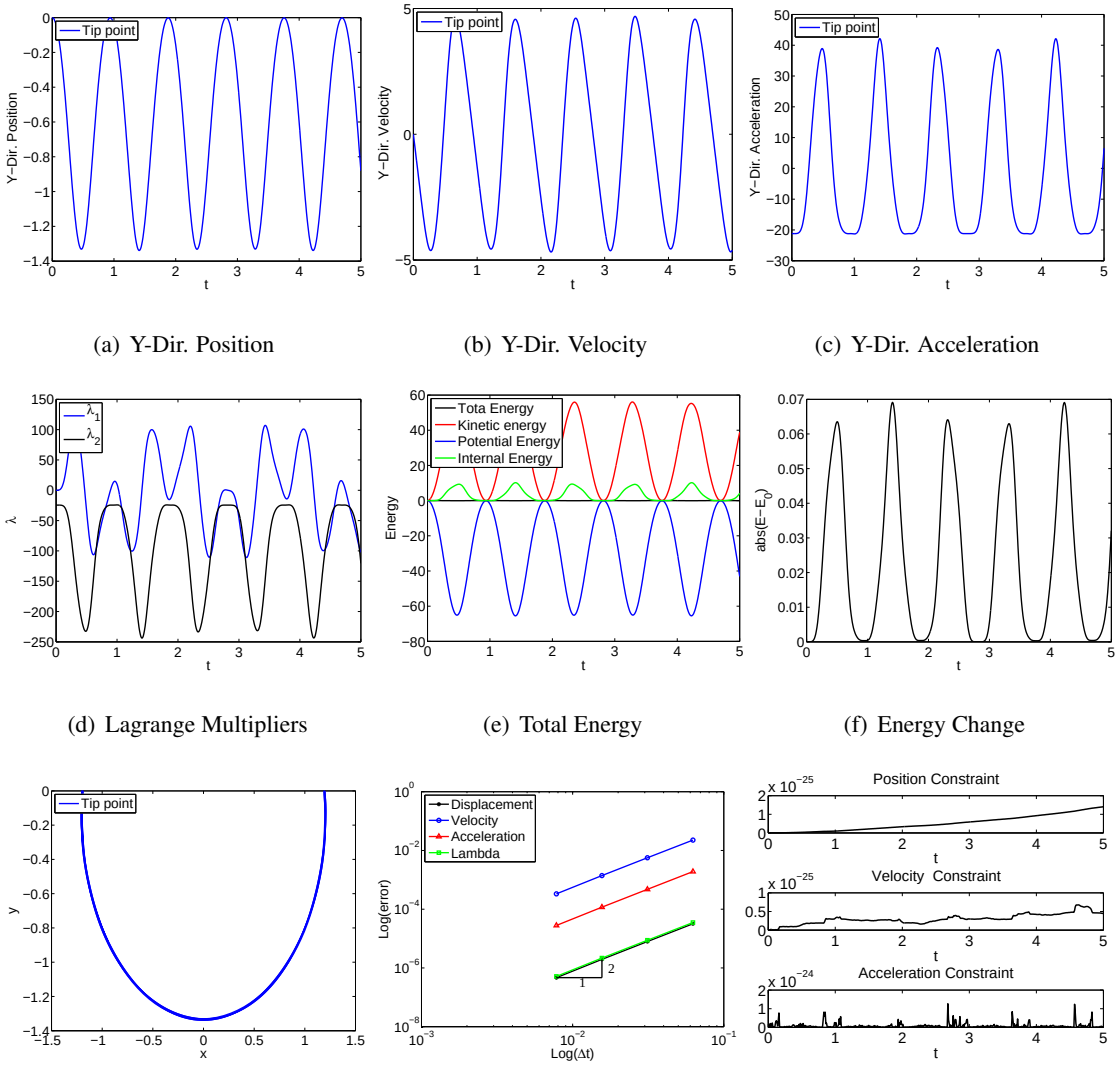


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.54: Single pendulum with bar element in IRF: U0(1,1,0) - Index 2.

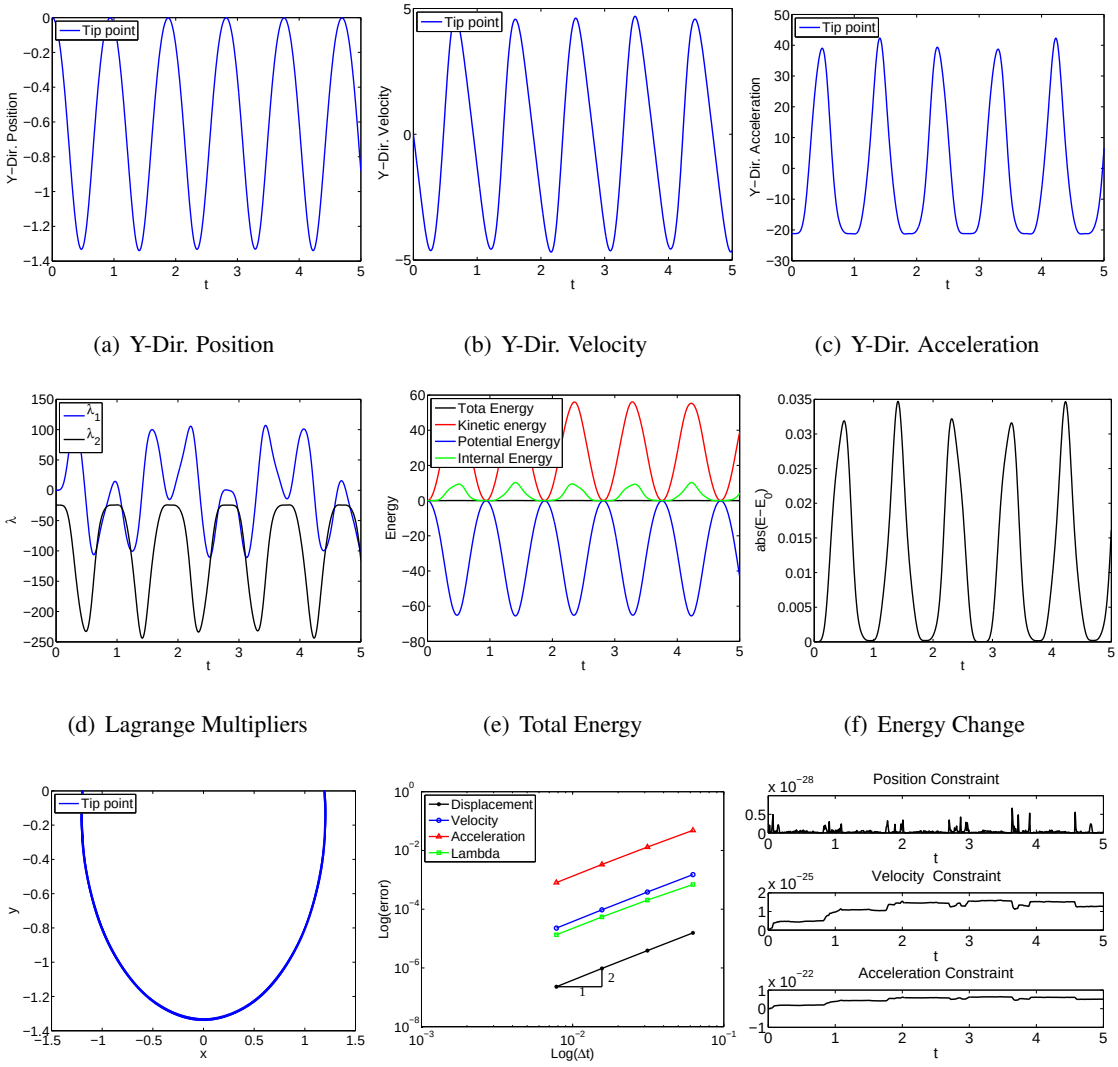


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.55: Single pendulum with bar element in IRF: U0(1,1,0) - Index 1.

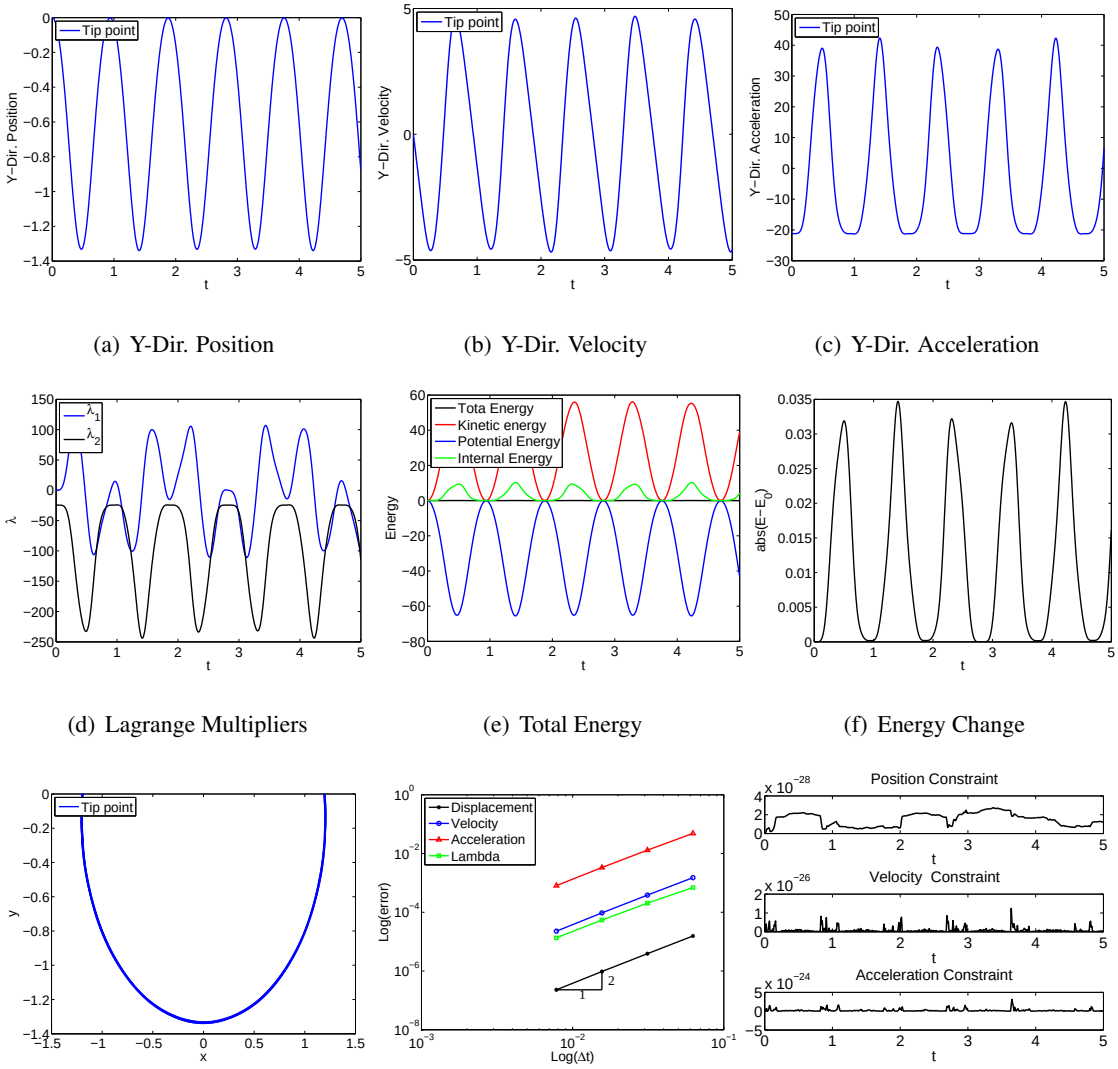


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.56: Single pendulum with bar element in IRF: V0(1,1,0) - Index 3.

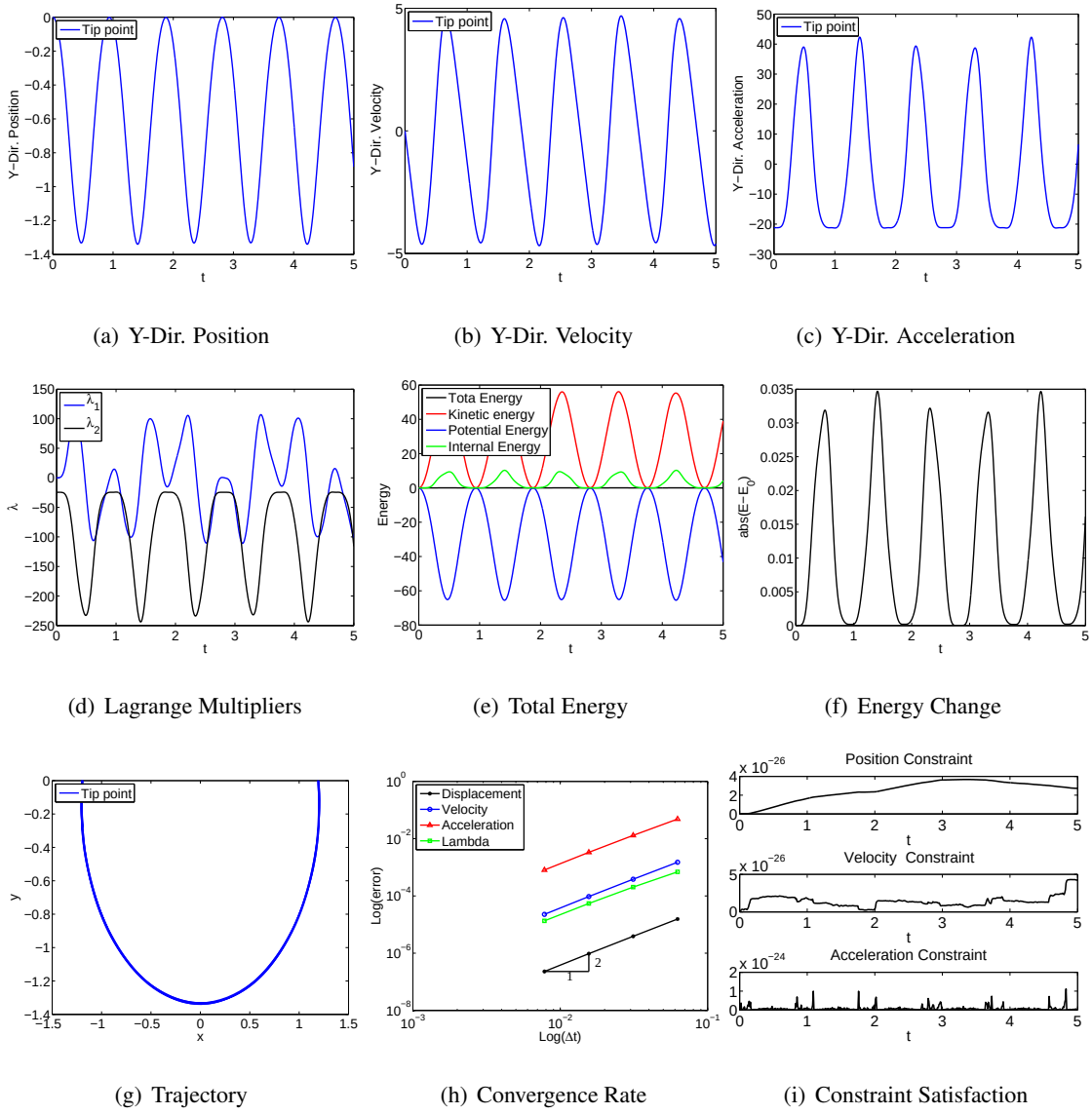


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.57: Single pendulum with bar element in IRF: V0(1,1,0) - Index 2.

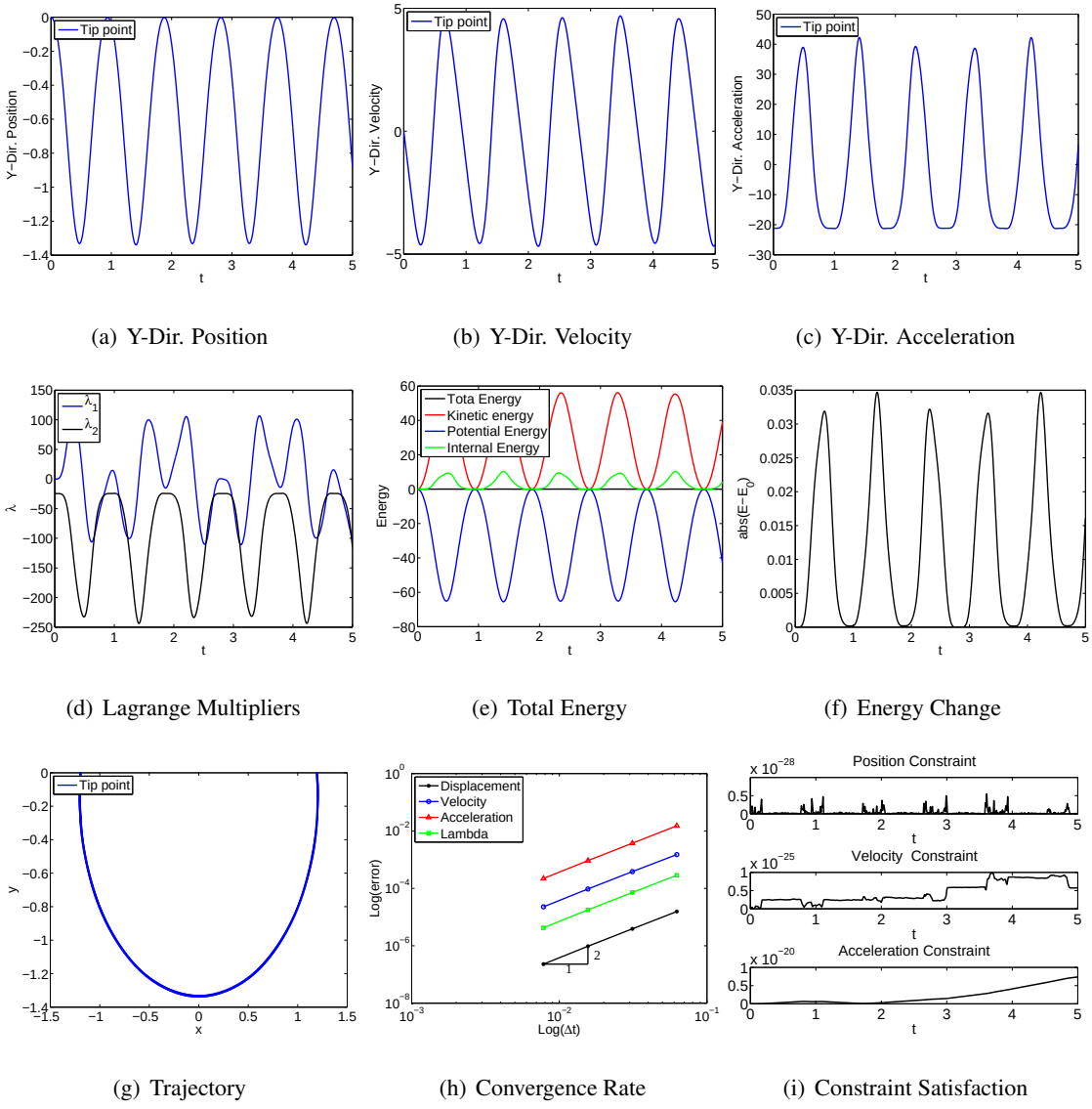


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.58: Single pendulum with bar element in IRF: V0(1,1,0) - Index 1.

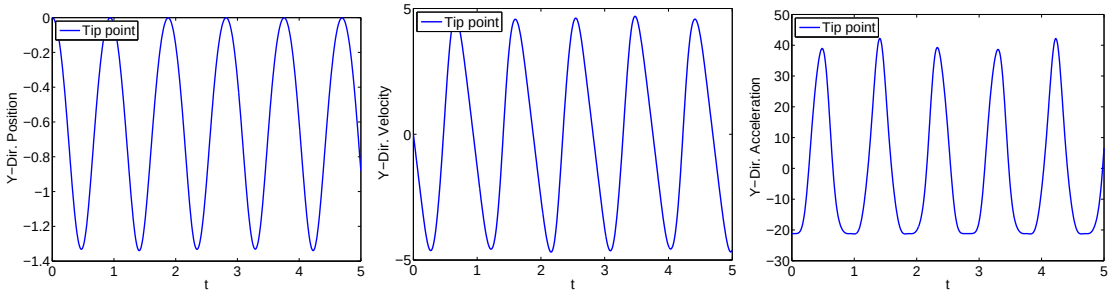


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

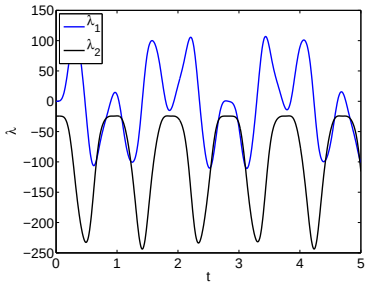
Figure 4.59: Single pendulum with bar element in IRF: U0V0(1,1,1) - Index 3.



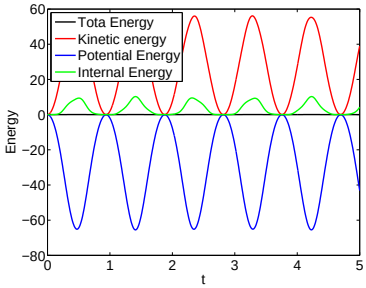
(a) Y-Dir. Position

(b) Y-Dir. Velocity

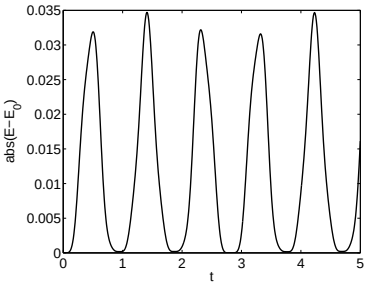
(c) Y-Dir. Acceleration



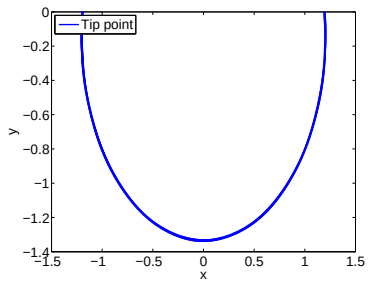
(d) Lagrange Multipliers



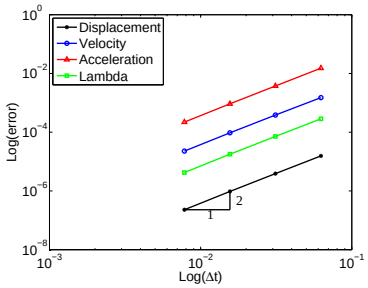
(e) Total Energy



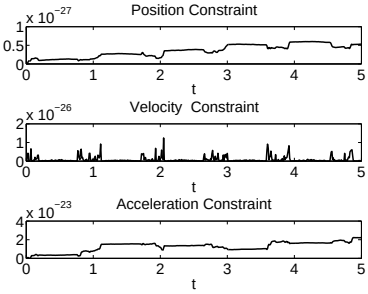
(f) Energy Change



(g) Trajectory

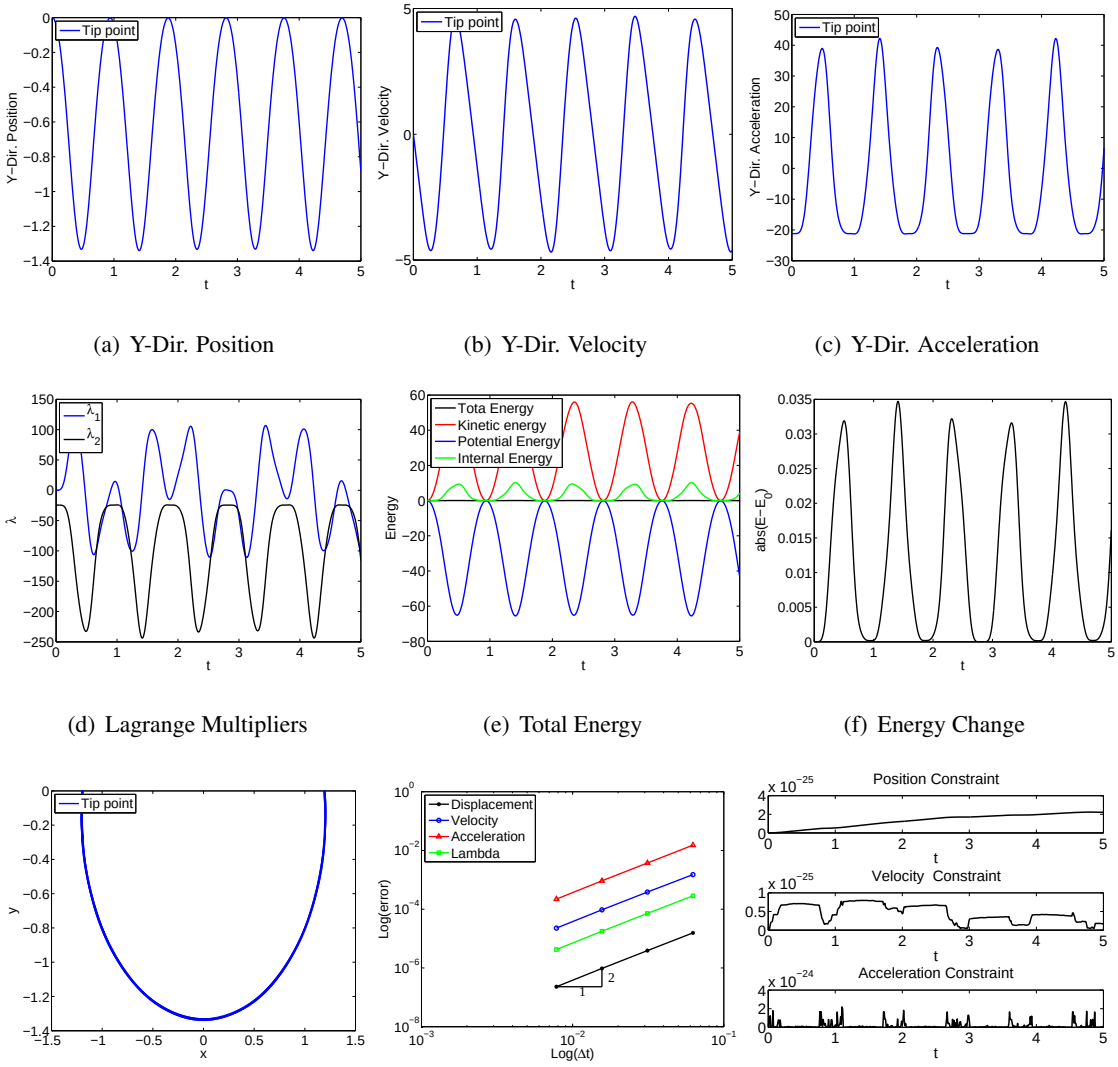


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.60: Single pendulum with bar element in IRF: U0V0(1,1,1) - Index 2.

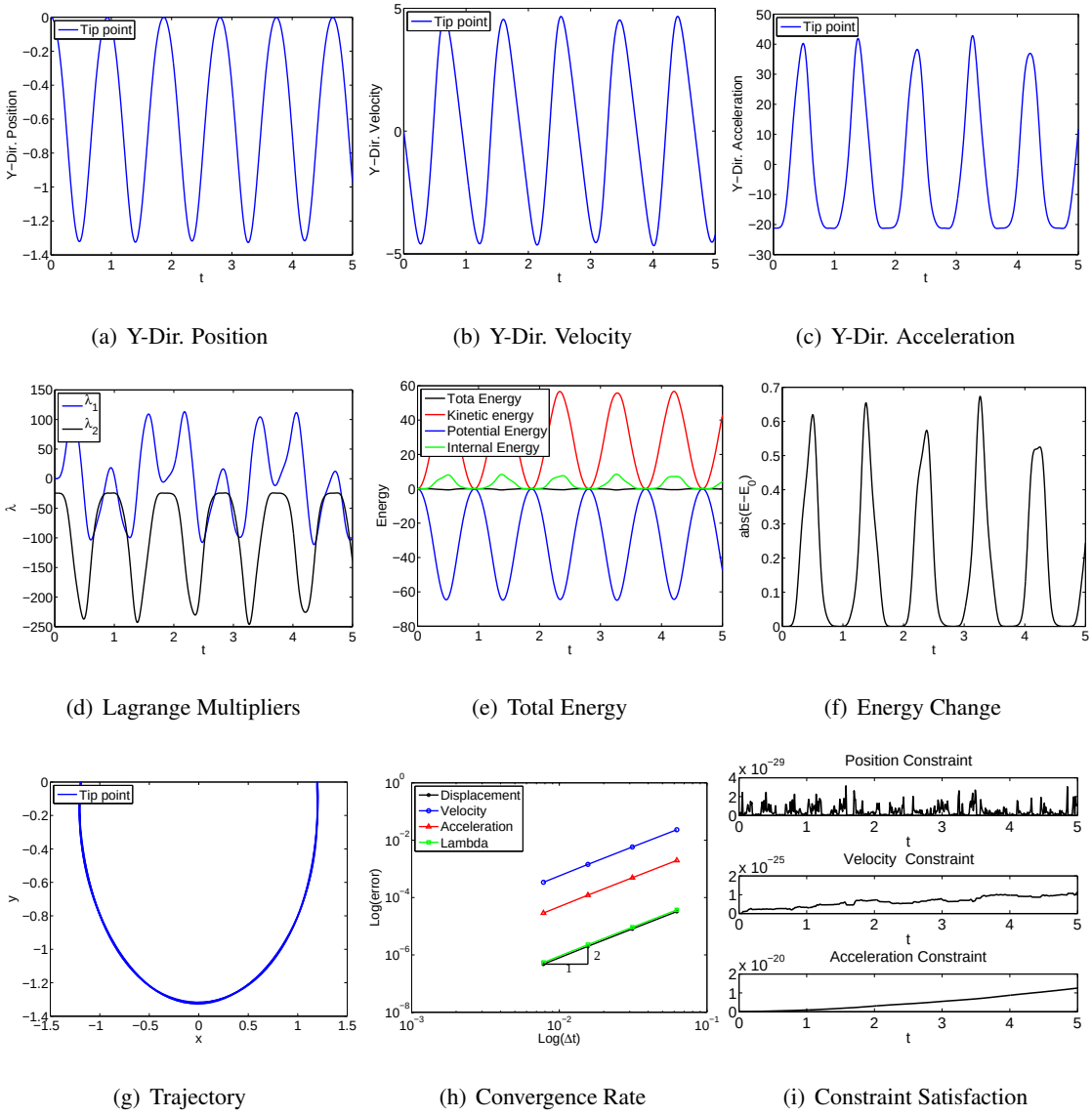


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.61: Single pendulum with bar element in IRF: U0V0(1,1,1) - Index 1.

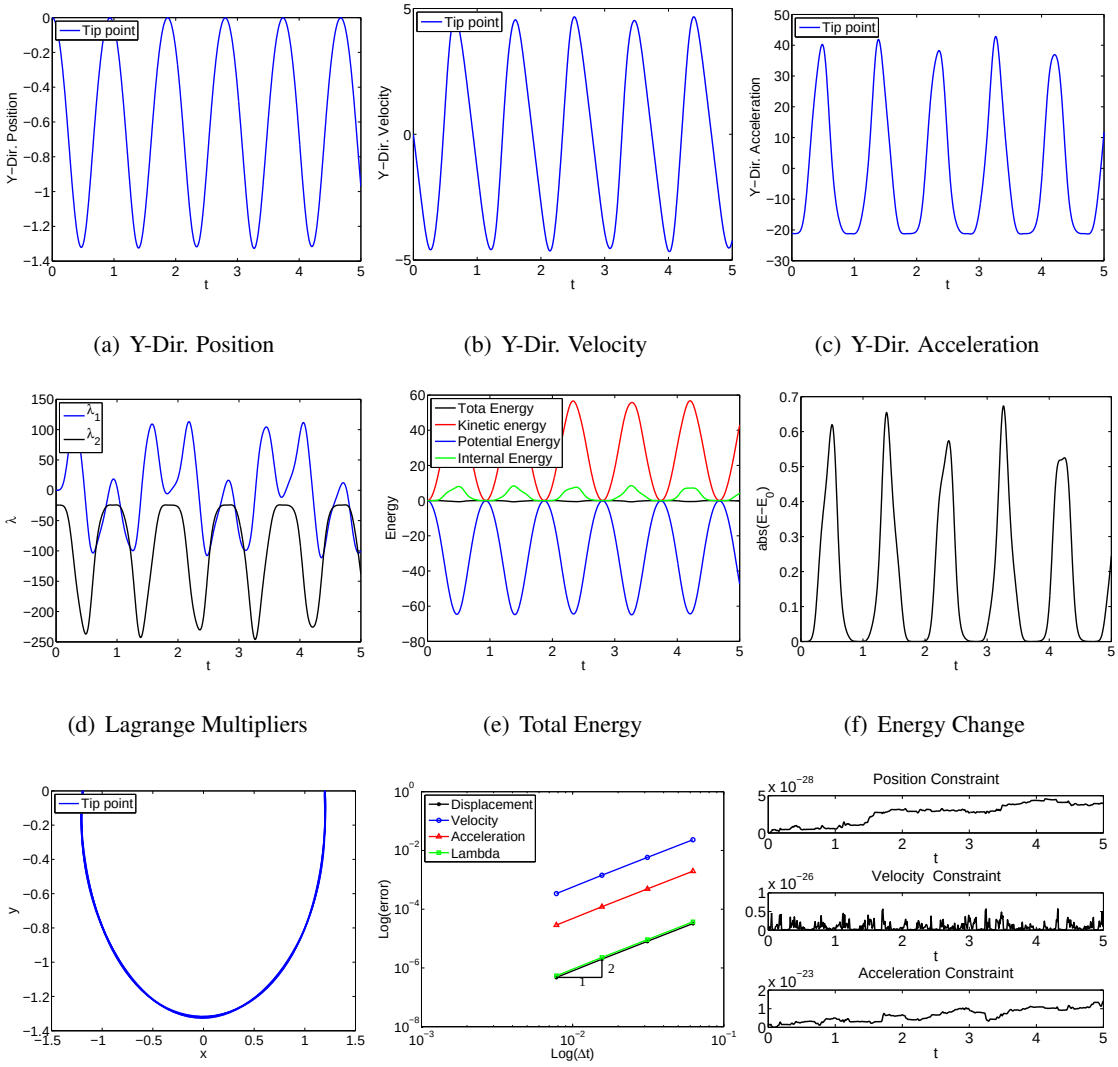


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.62: Single pendulum with bar element in ANCF-S: U0(1,1,0) - Index 3.

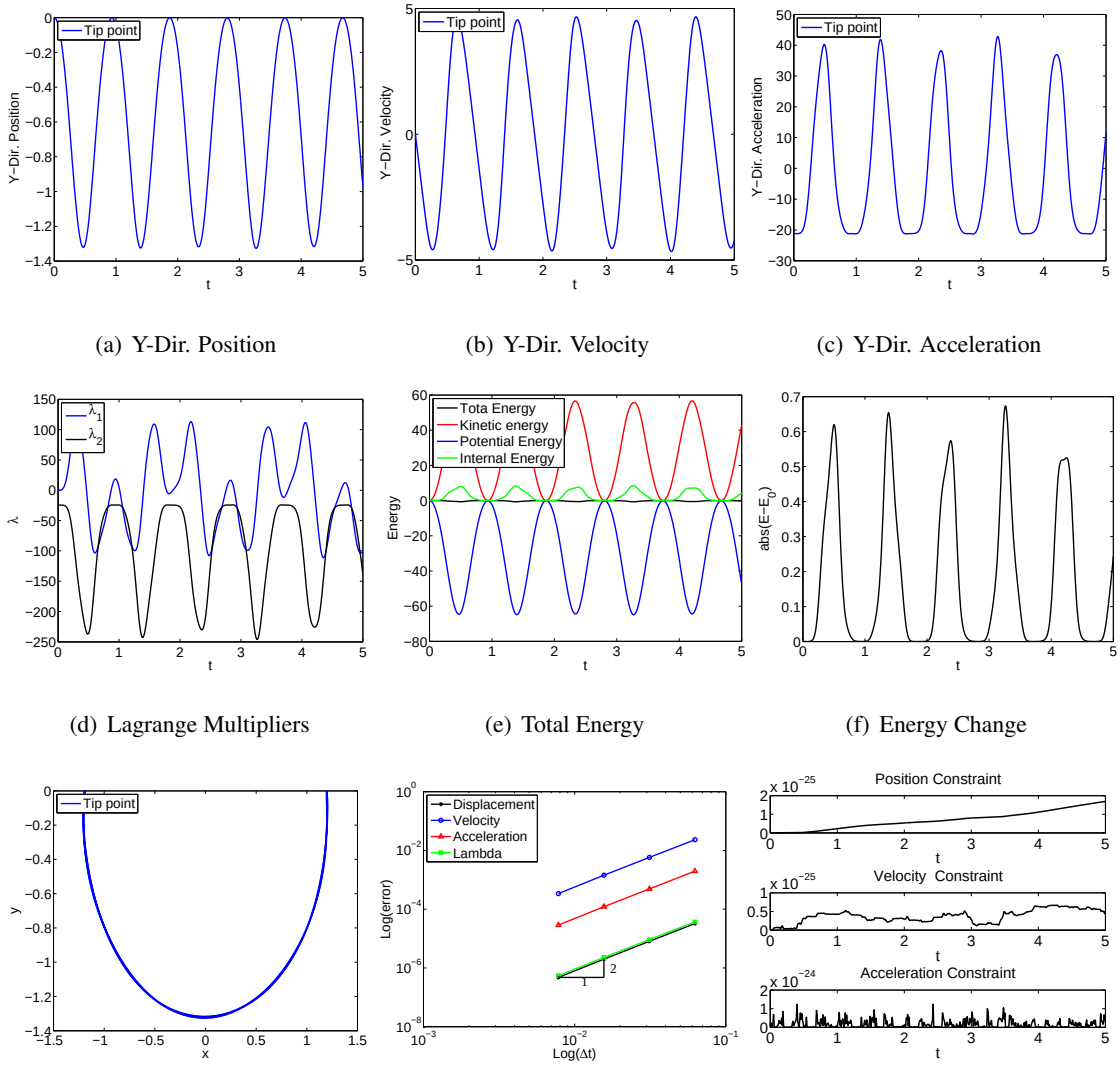


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.63: Single pendulum with bar element in ANCF-S: U0(1,1,0) - Index 2.

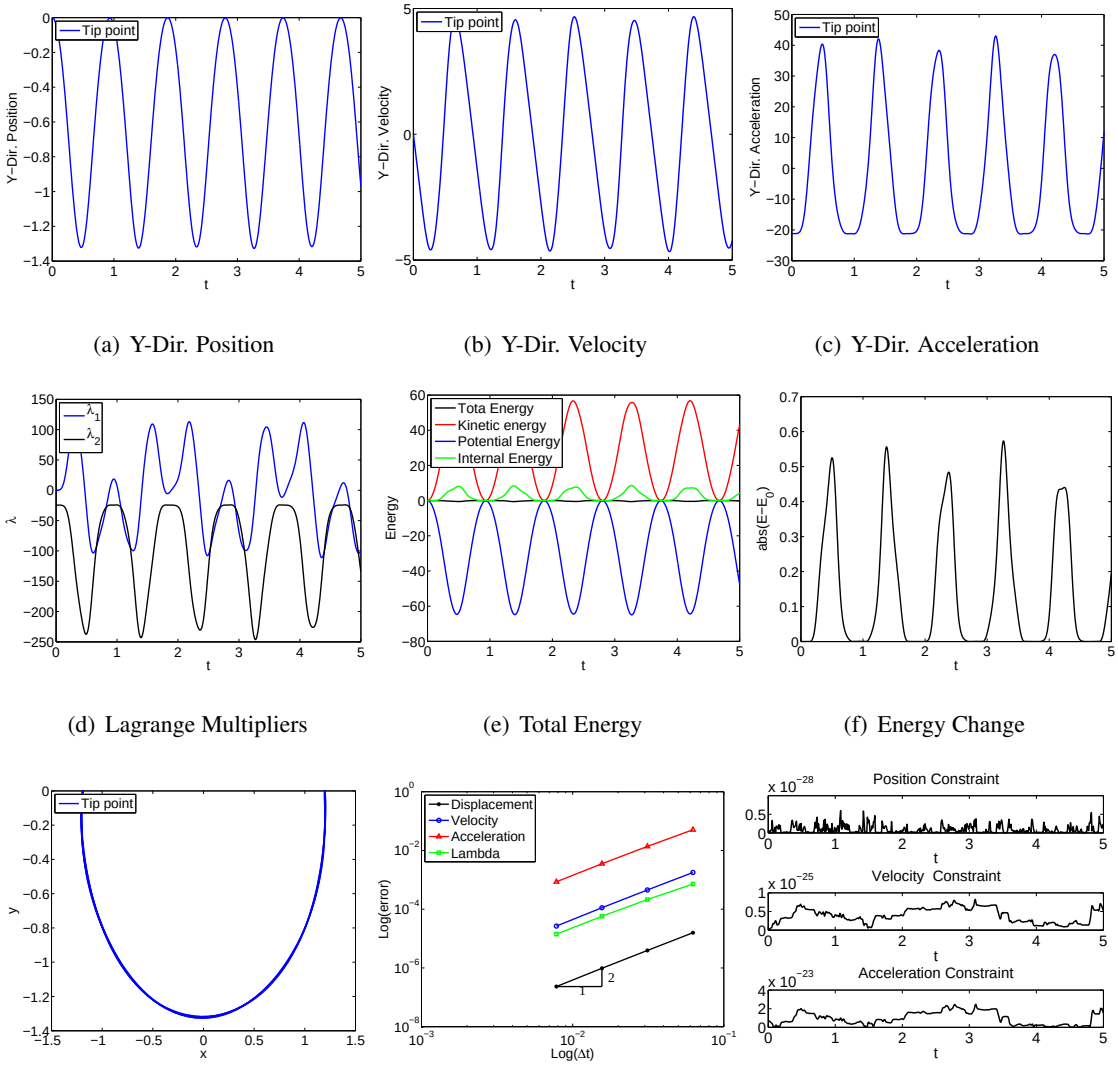


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.64: Single pendulum with bar element in ANCF-S: U0(1,1,0) - Index 1.

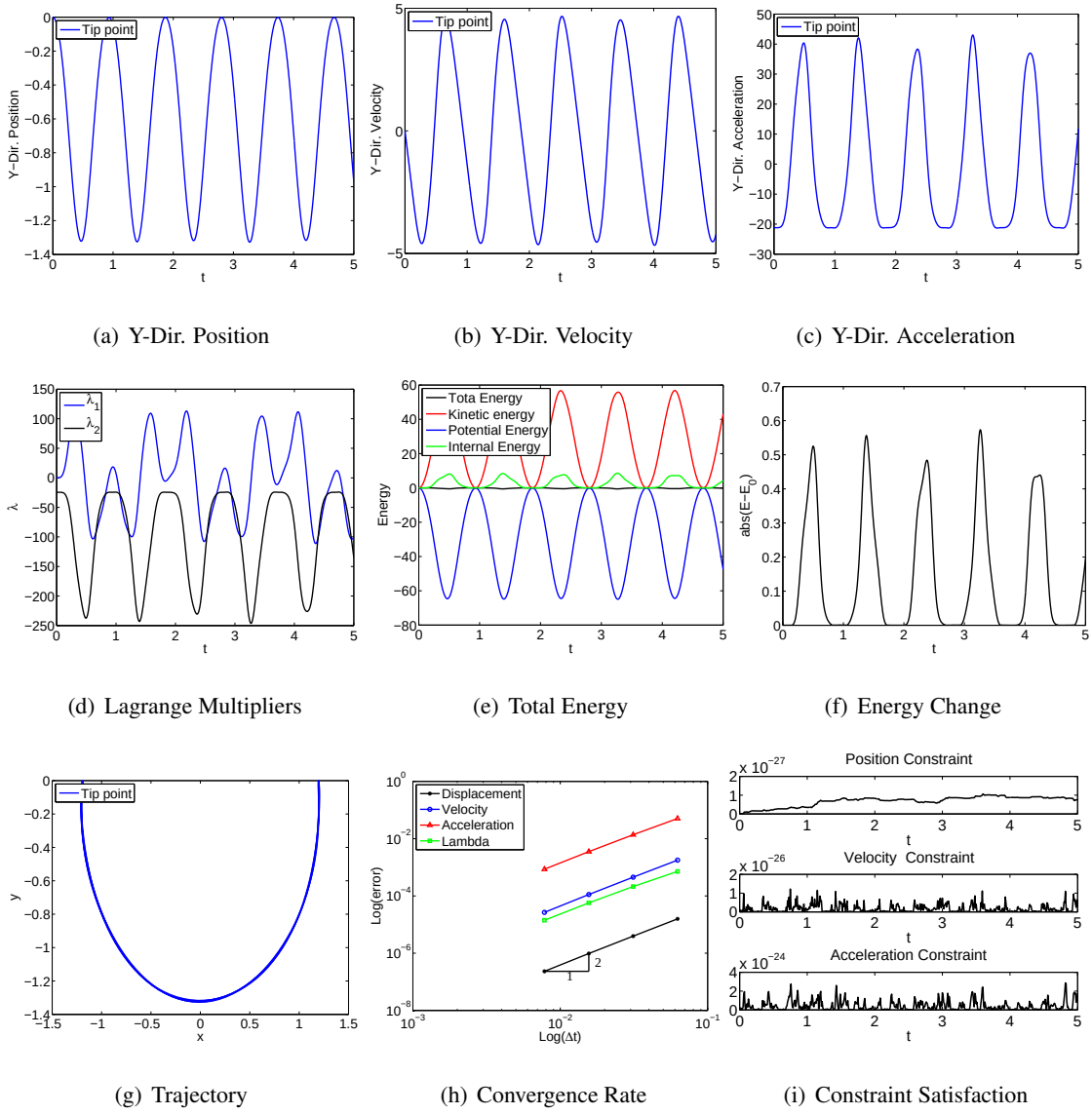


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.65: Single pendulum with bar element in ANCF-S: V0(1,1,0) - Index 3.

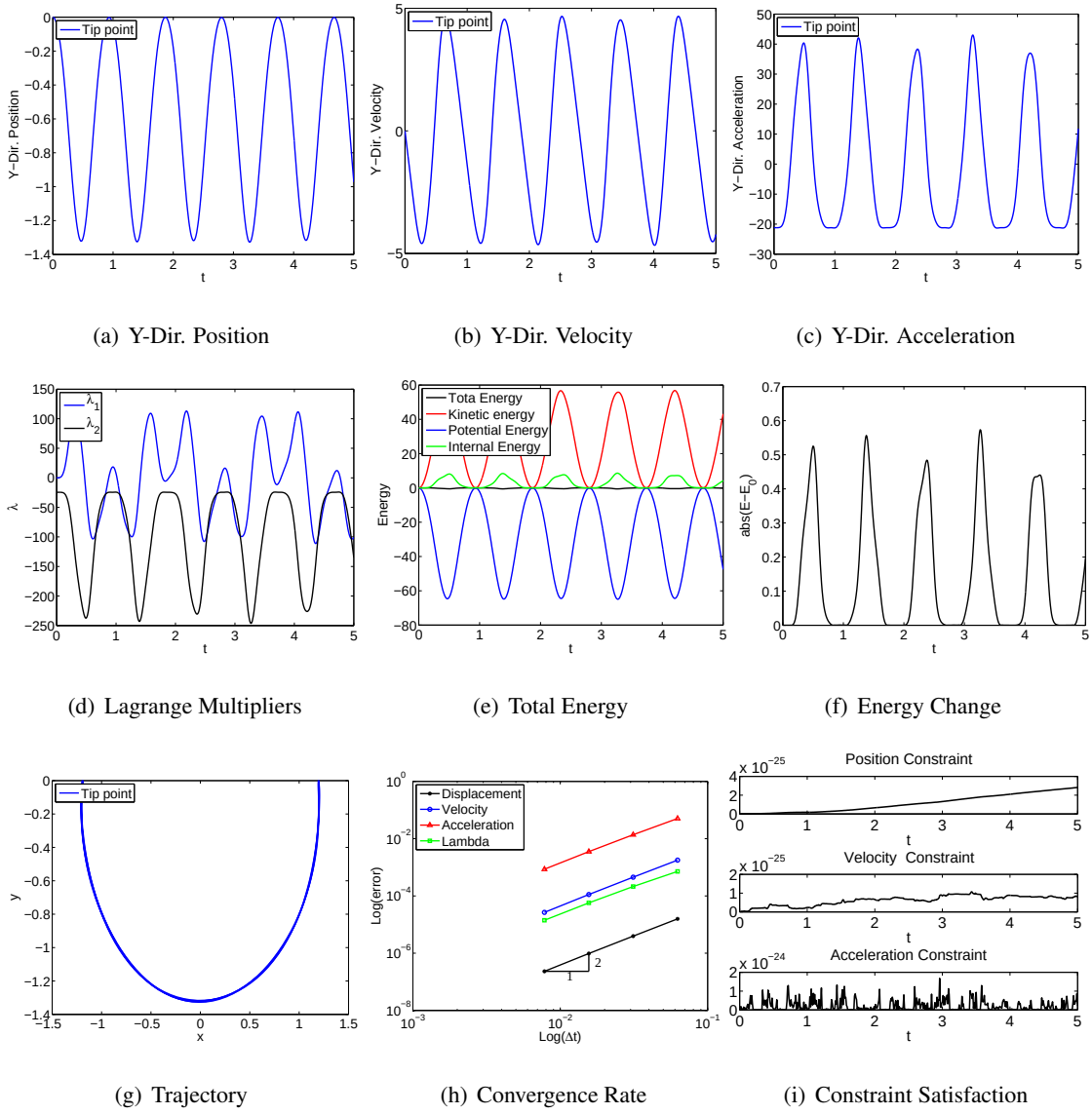


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.66: Single pendulum with bar element in ANCF-S: V0(1,1,0) - Index 2.

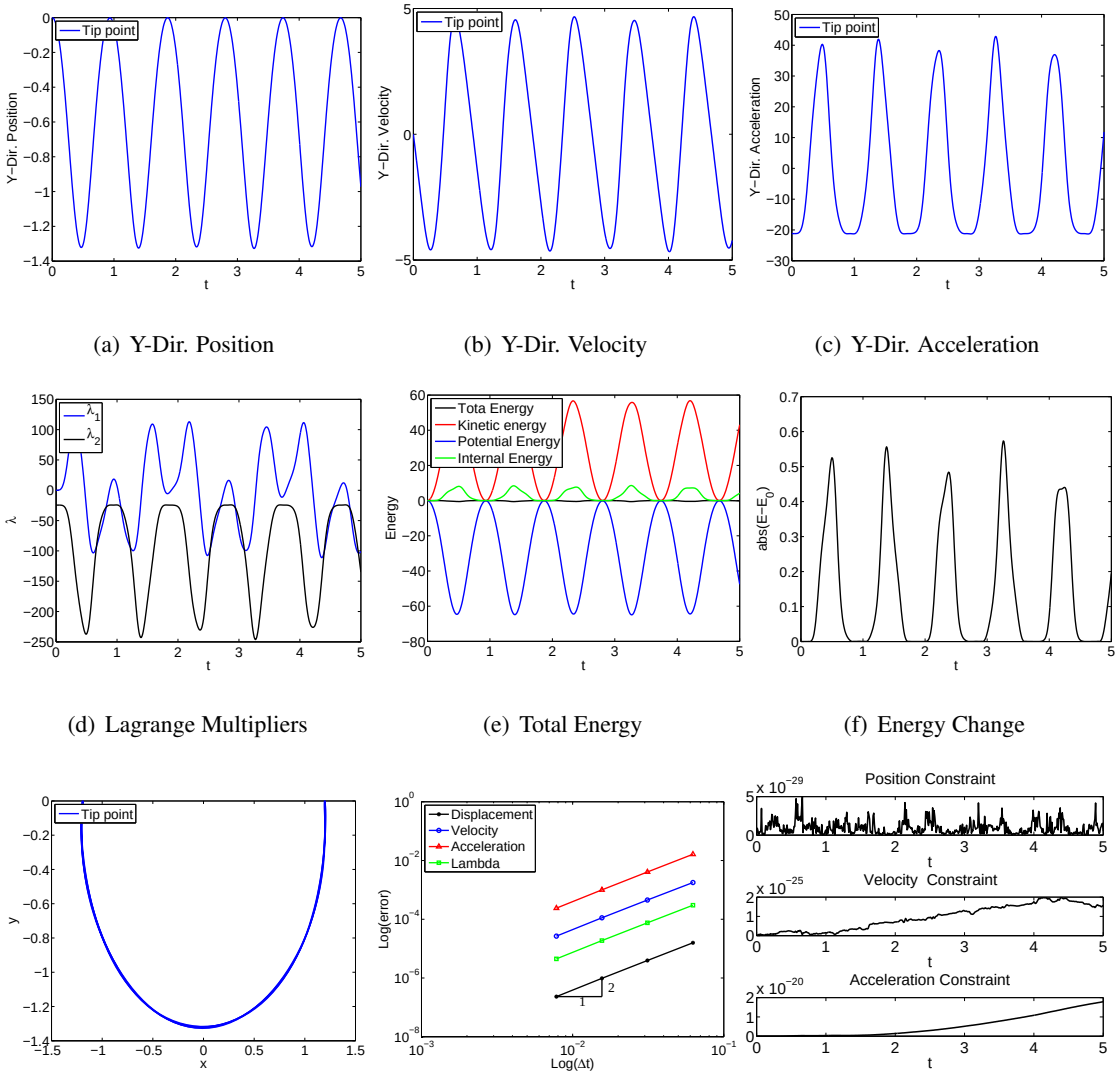


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.67: Single pendulum with bar element in ANCF-S: V0(1,1,0) - Index 1.

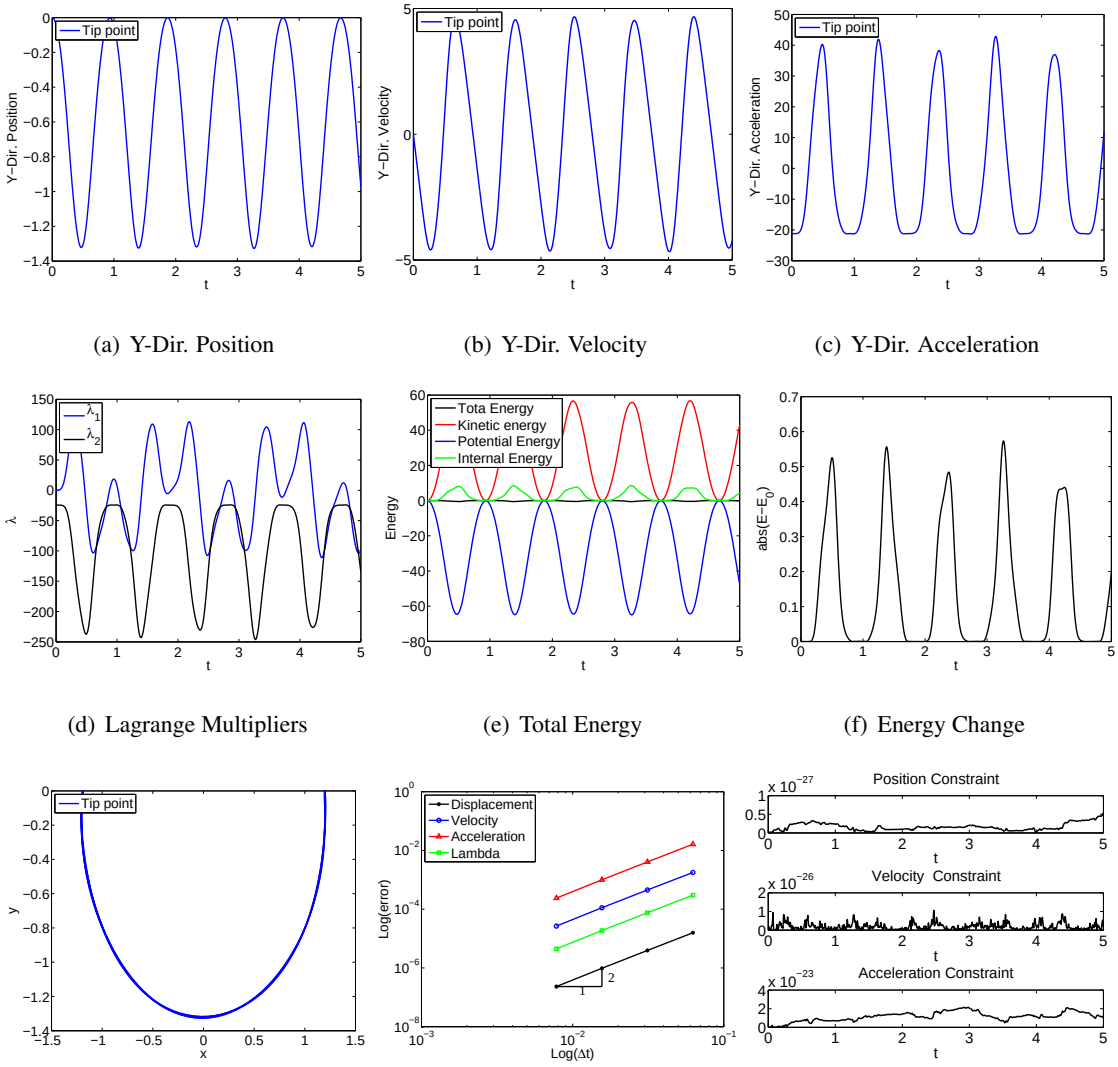


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.68: Single pendulum with bar element in ANCF-S: U0V0(1,1,1) - Index 3.

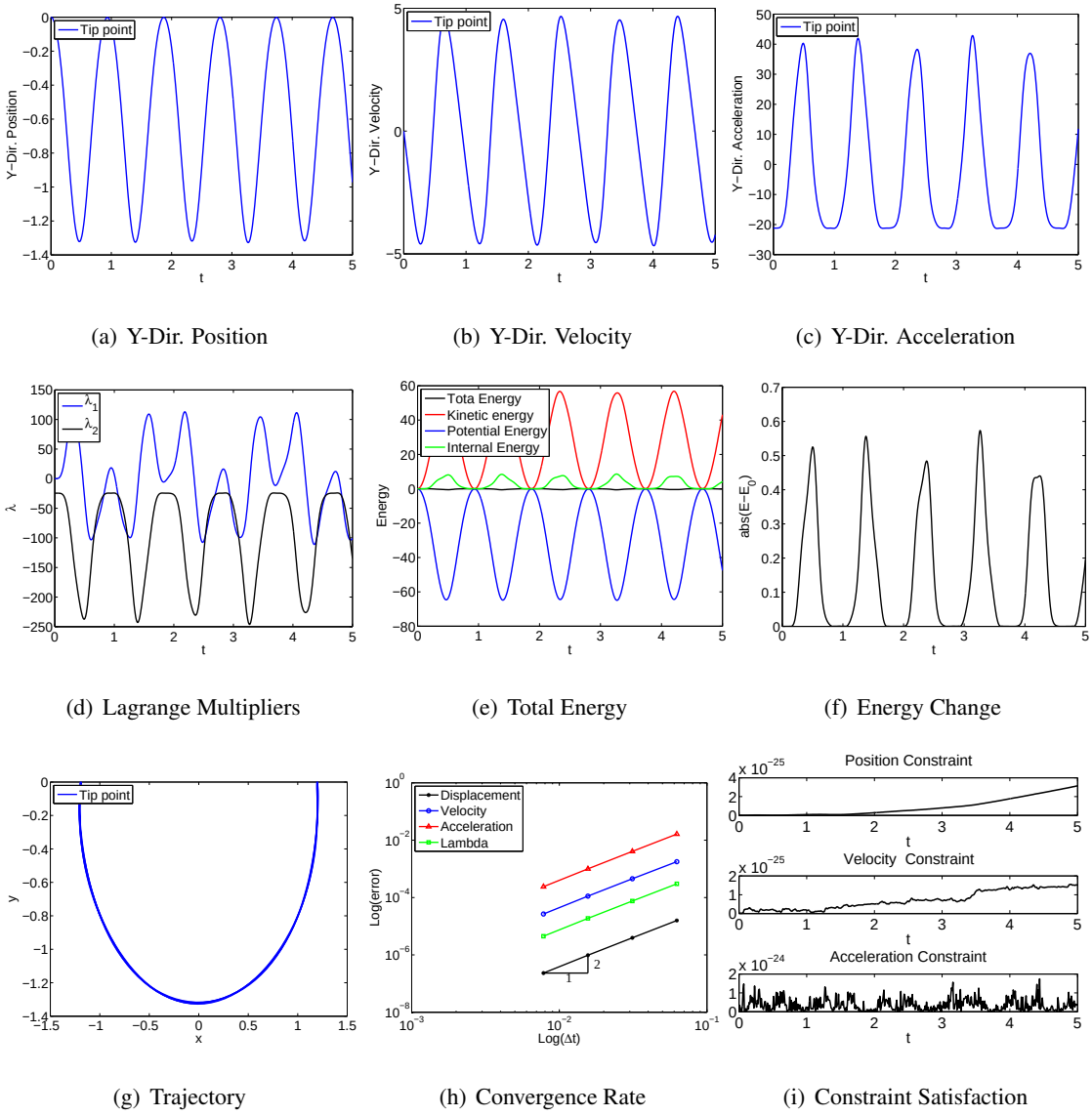


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.69: Single pendulum with bar element in ANCF-S: U0V0(1,1,1) - Index 2.

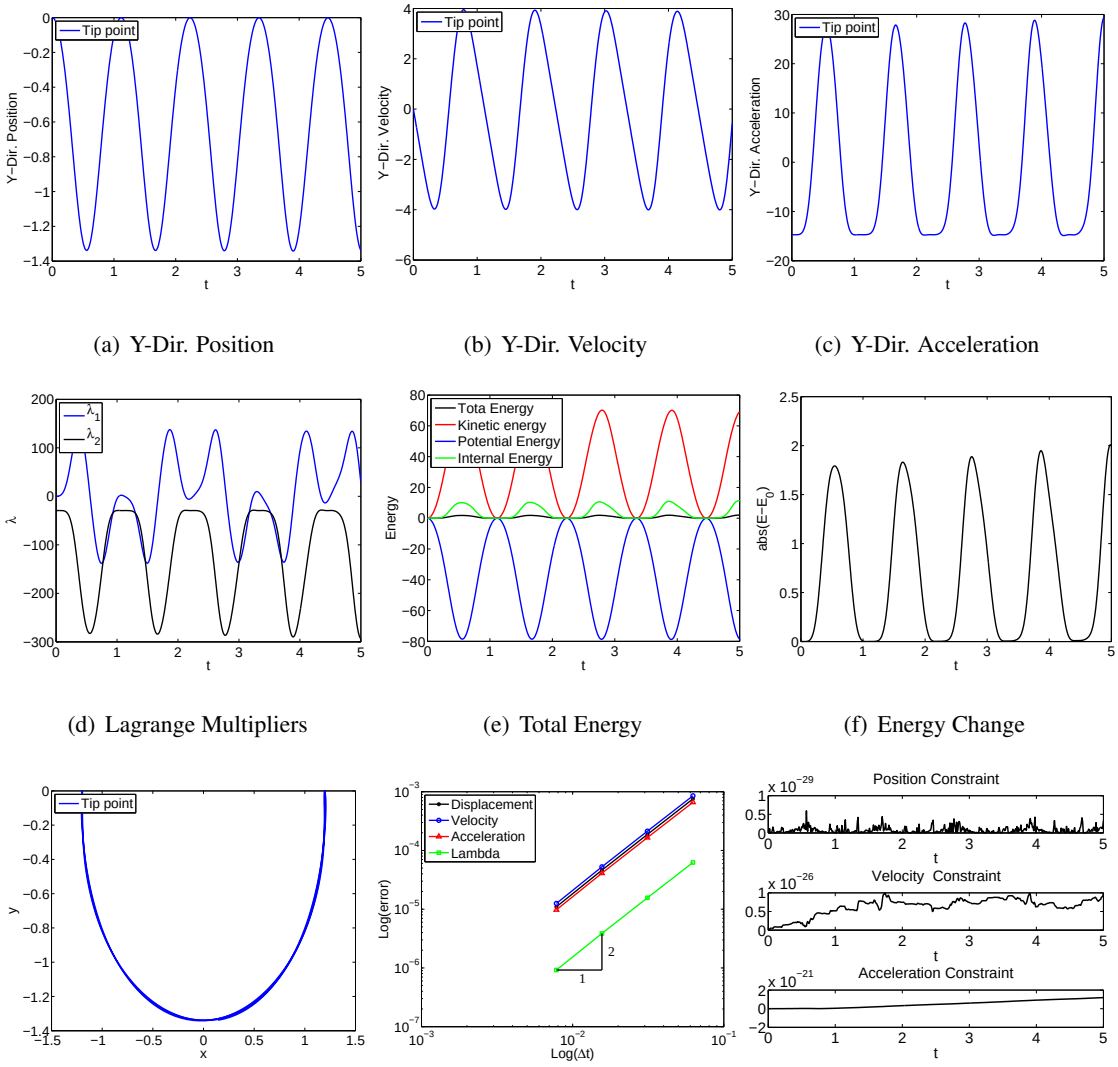


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.70: Single pendulum with bar element in ANCF-S: U0V0(1,1,1) - Index 1.

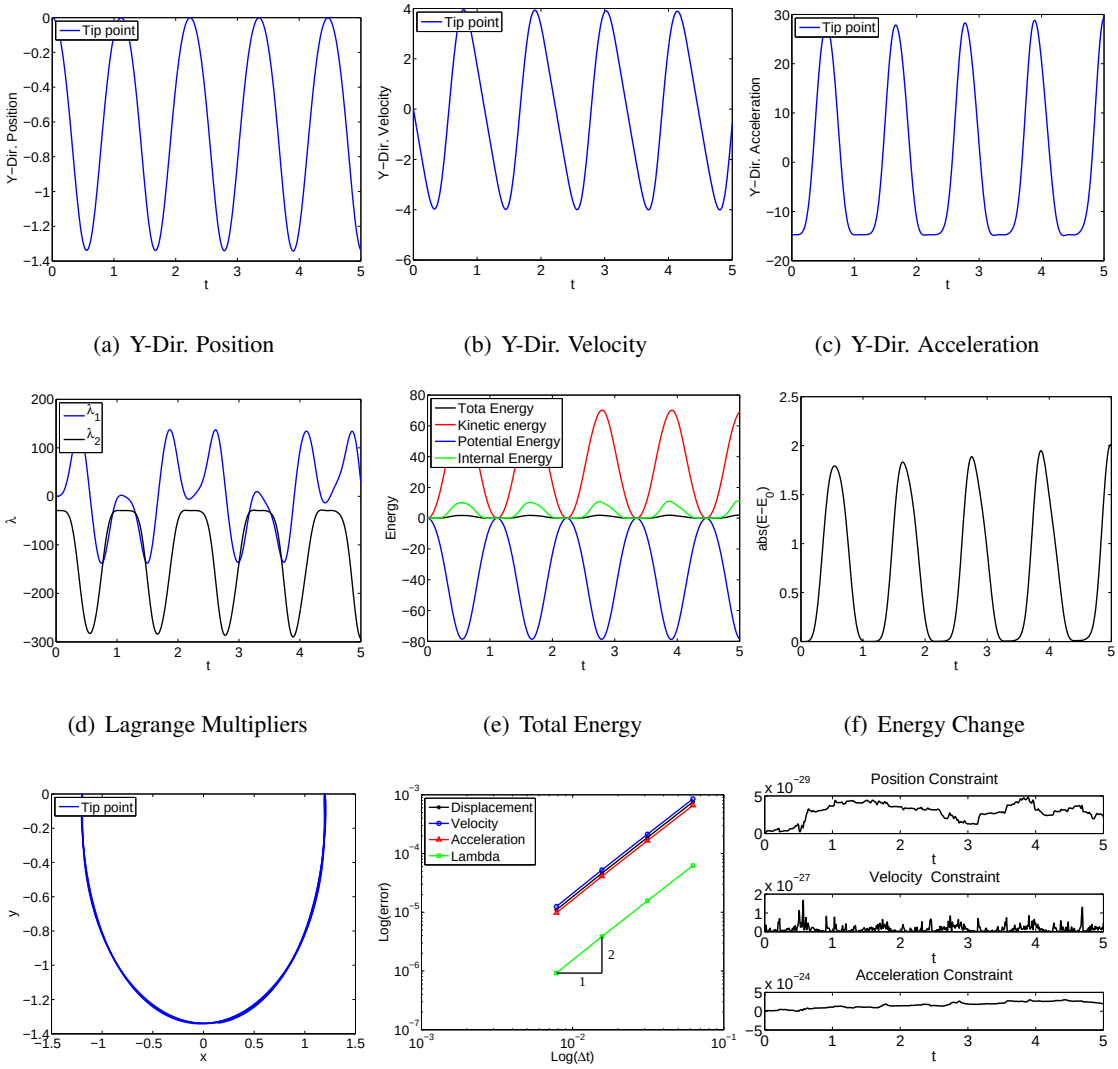


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.71: Single pendulum with bar element in FRF: U0(1,1,0) - Index 3.

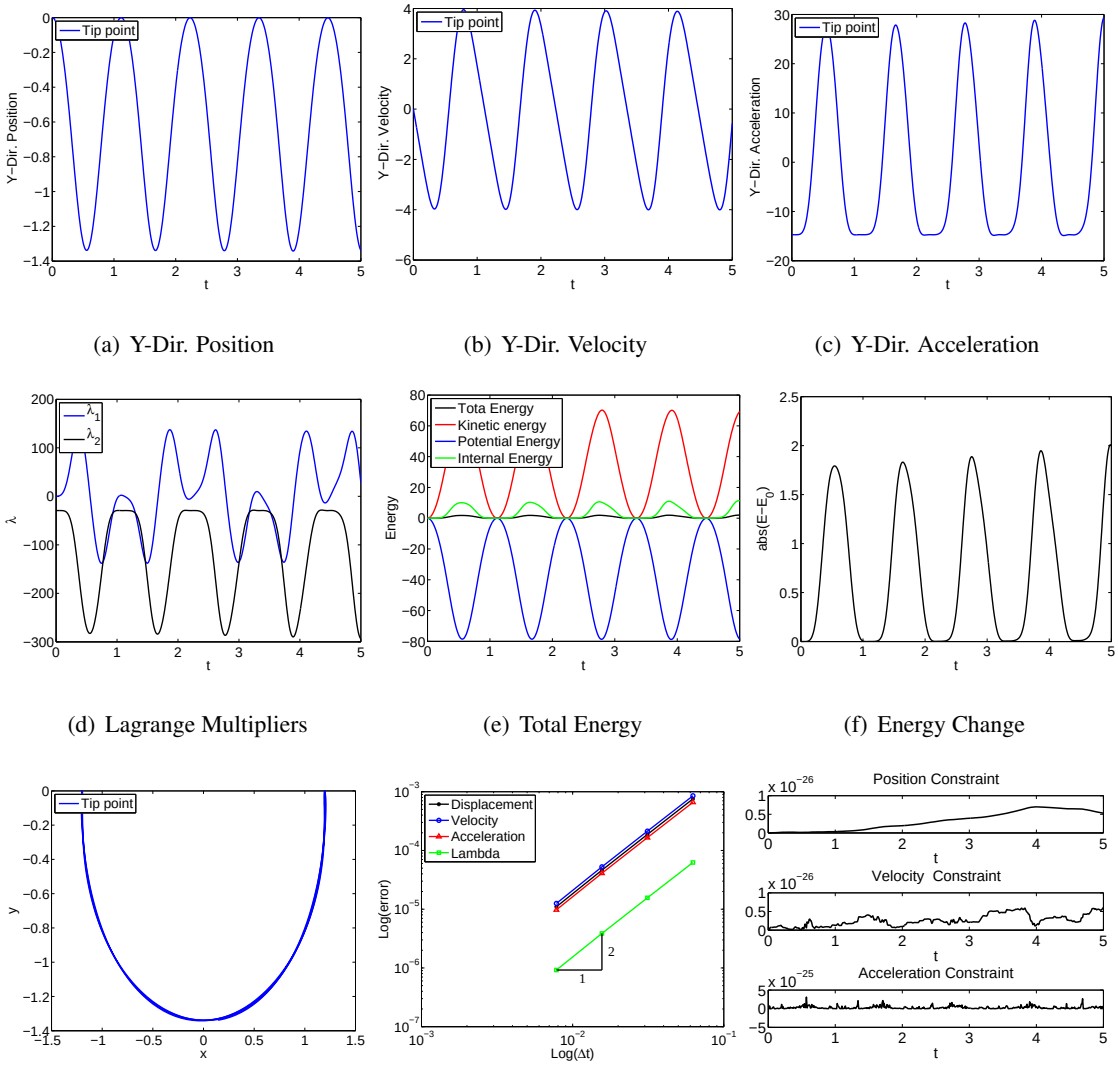


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.72: Single pendulum with bar element in FRF: U0(1,1,0) - Index 2.

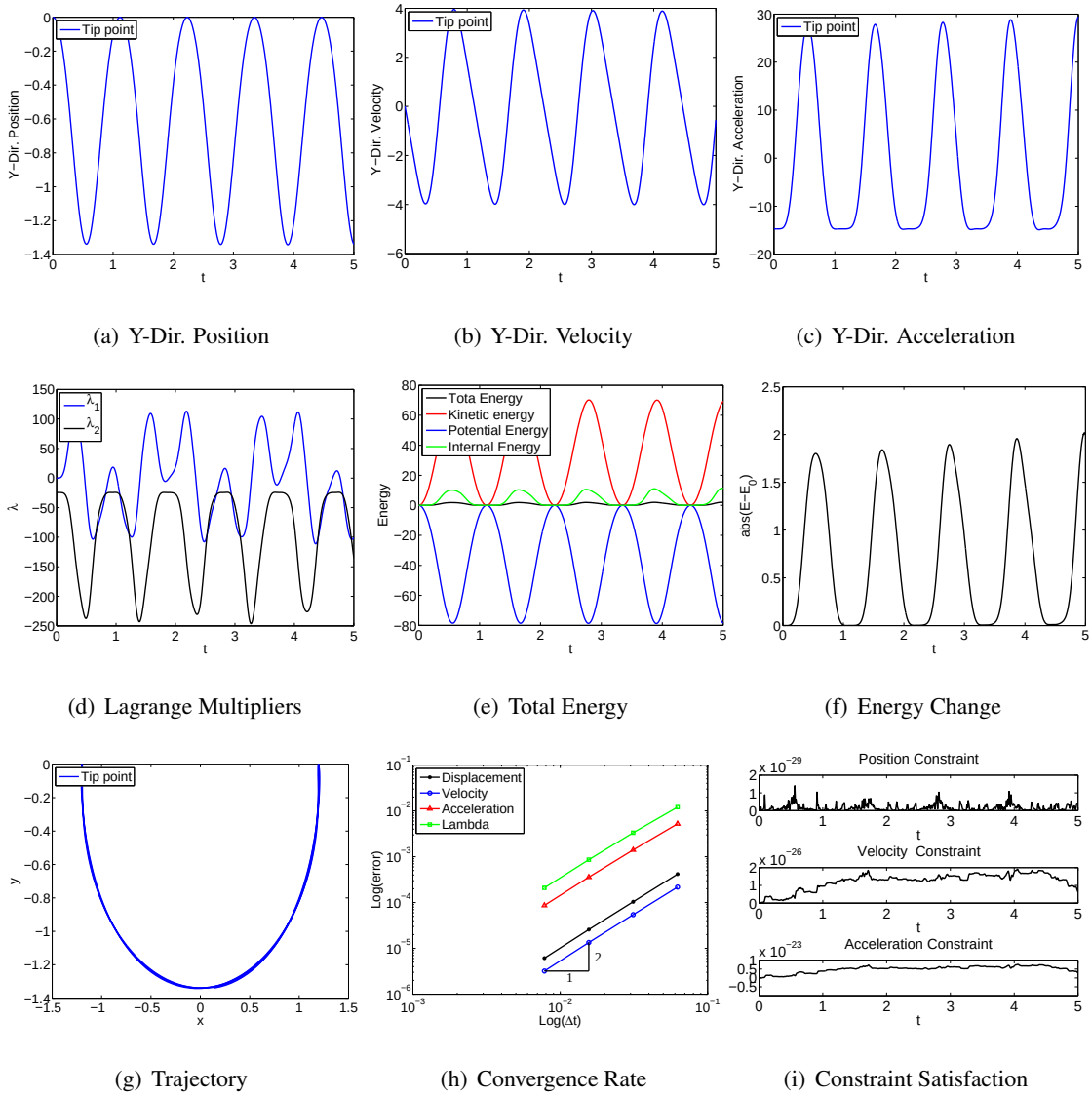


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.73: Single pendulum with bar element in FRF: U0(1,1,0) - Index 1.

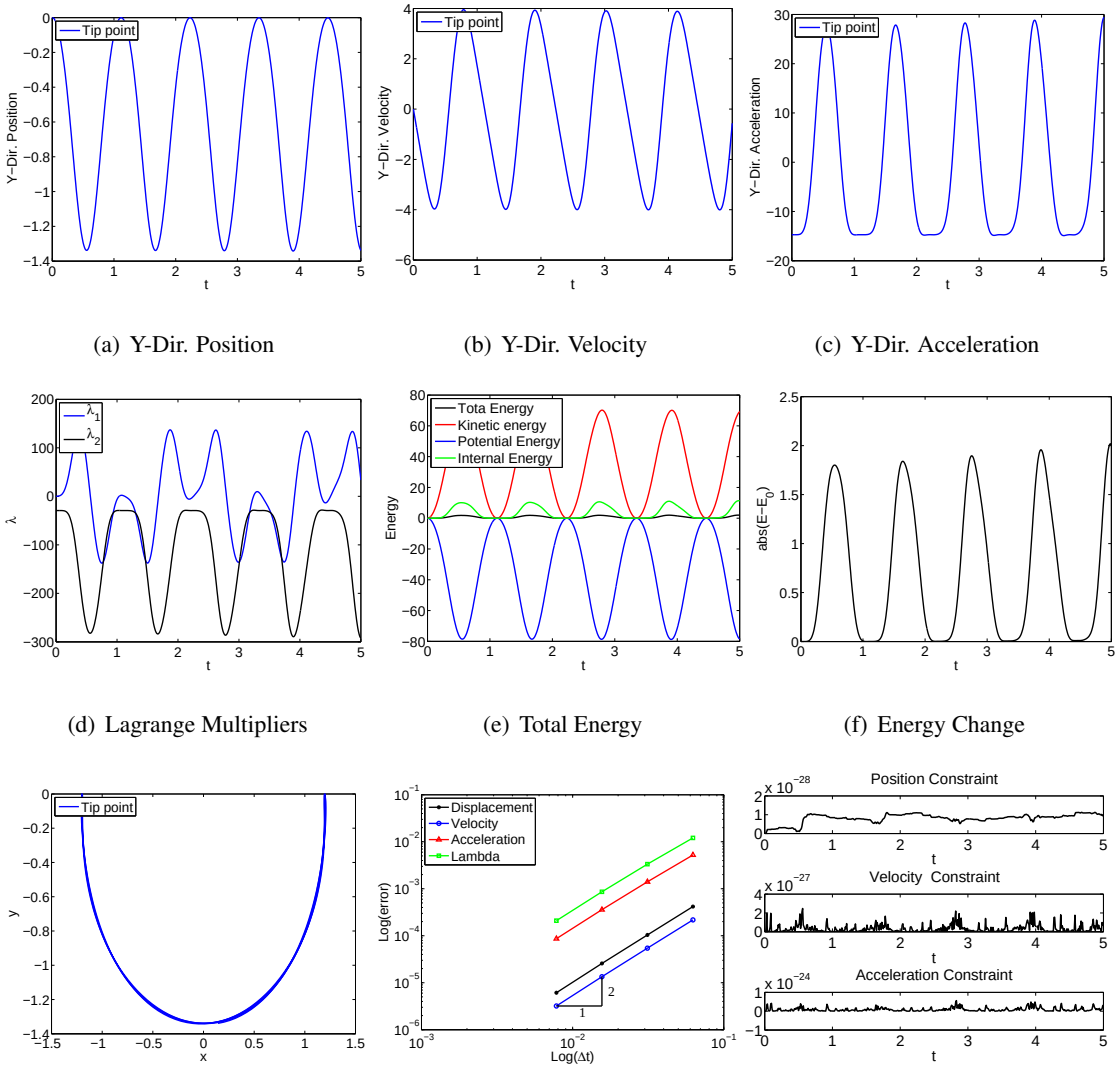


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.74: Single pendulum with bar element in FRF: V0(1,1,0) - Index 3.

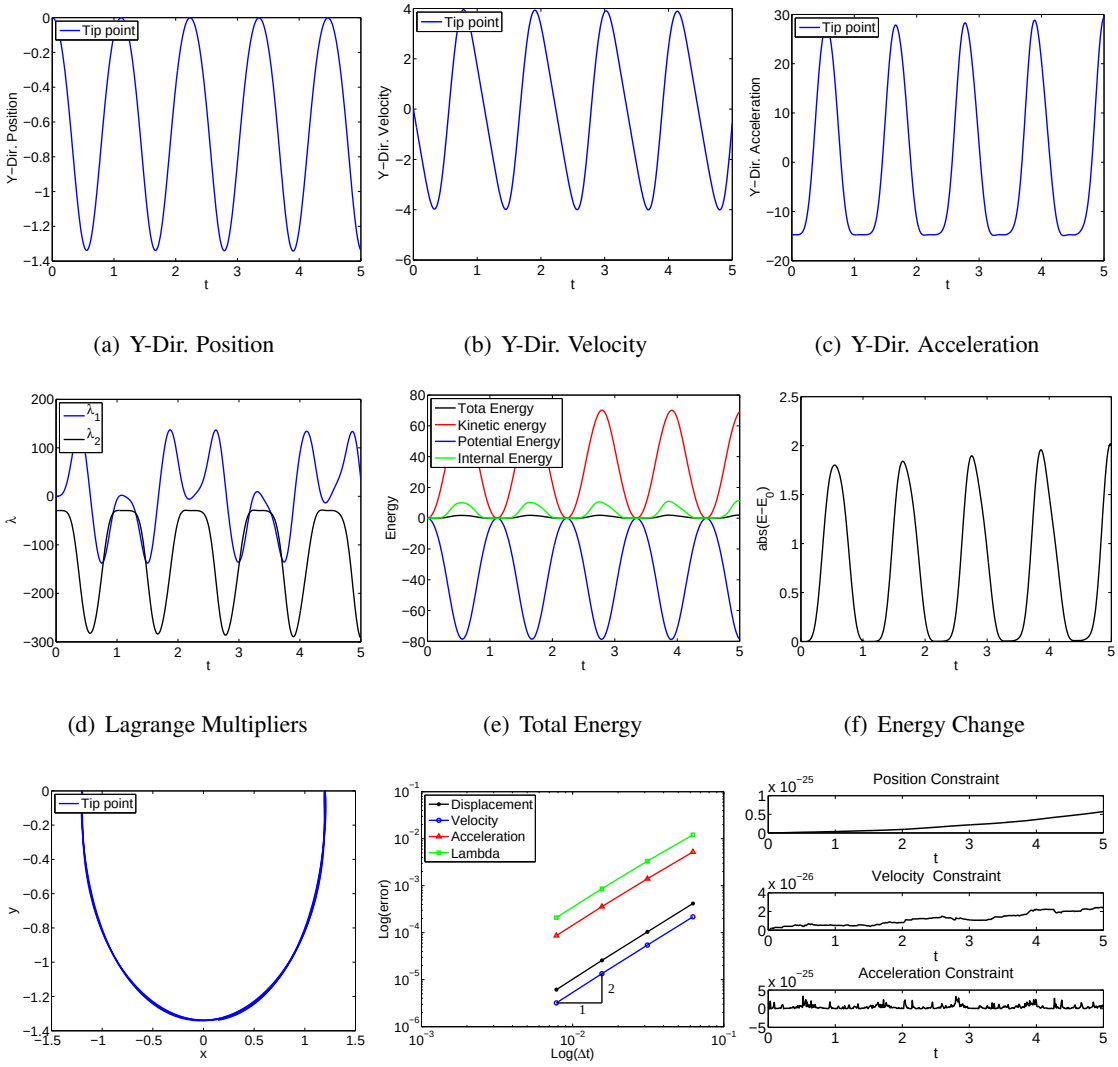


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.75: Single pendulum with bar element in FRF: V0(1,1,0) - Index 2.

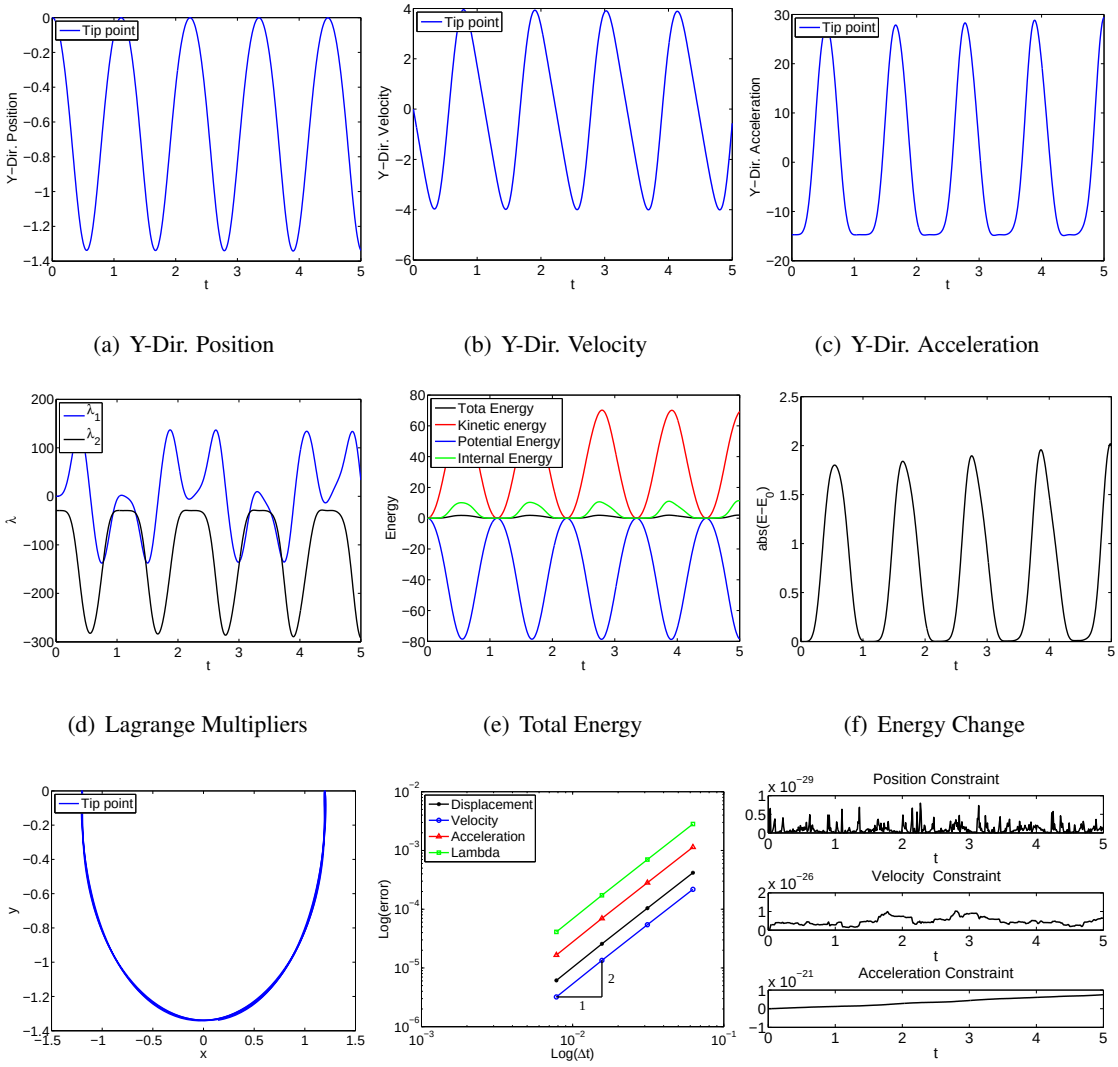


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.76: Single pendulum with bar element in FRF: V0(1,1,0) - Index 1.

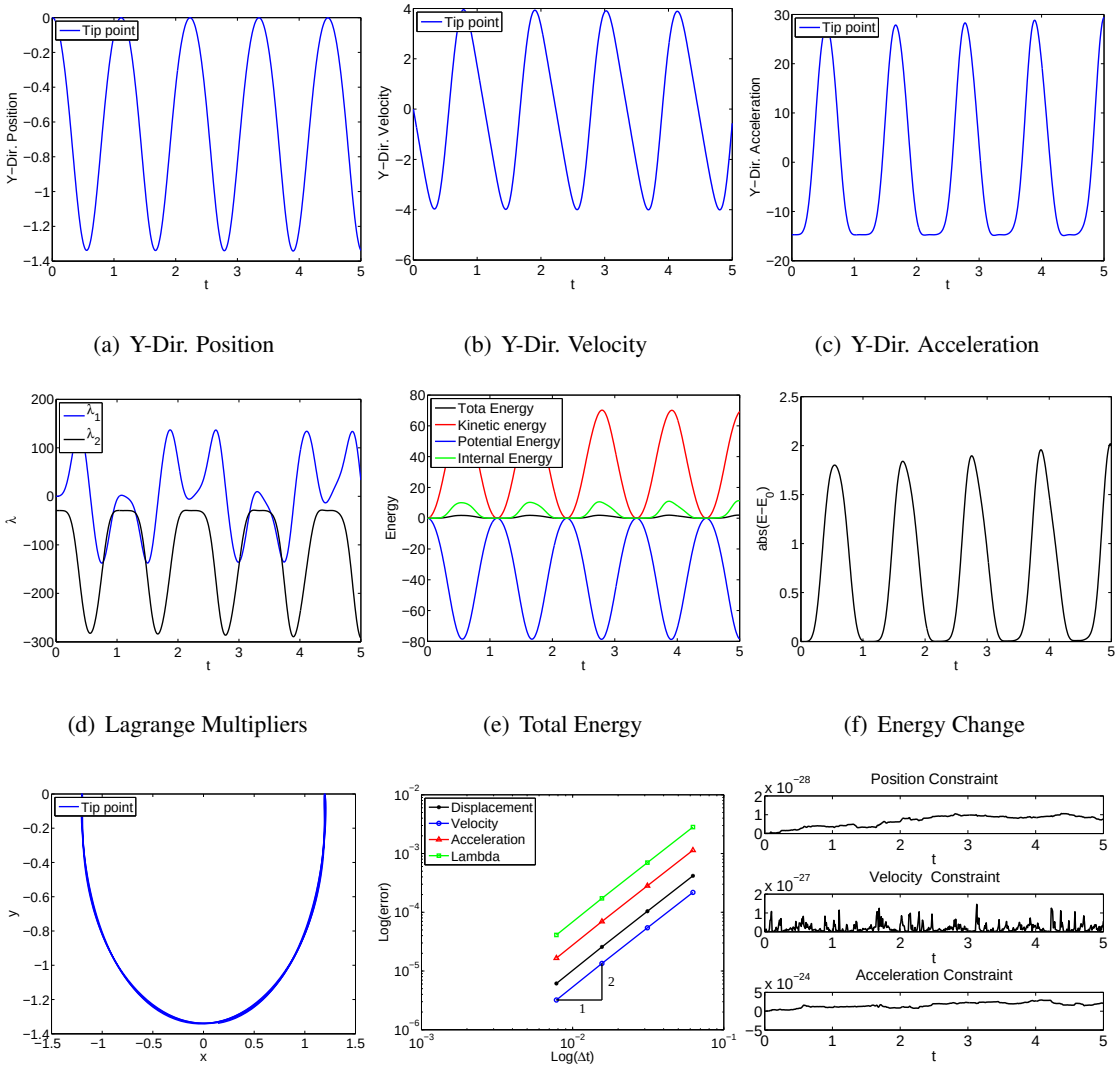


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.77: Single pendulum with bar element in FRF: U0V0(1,1,1) - Index 3.

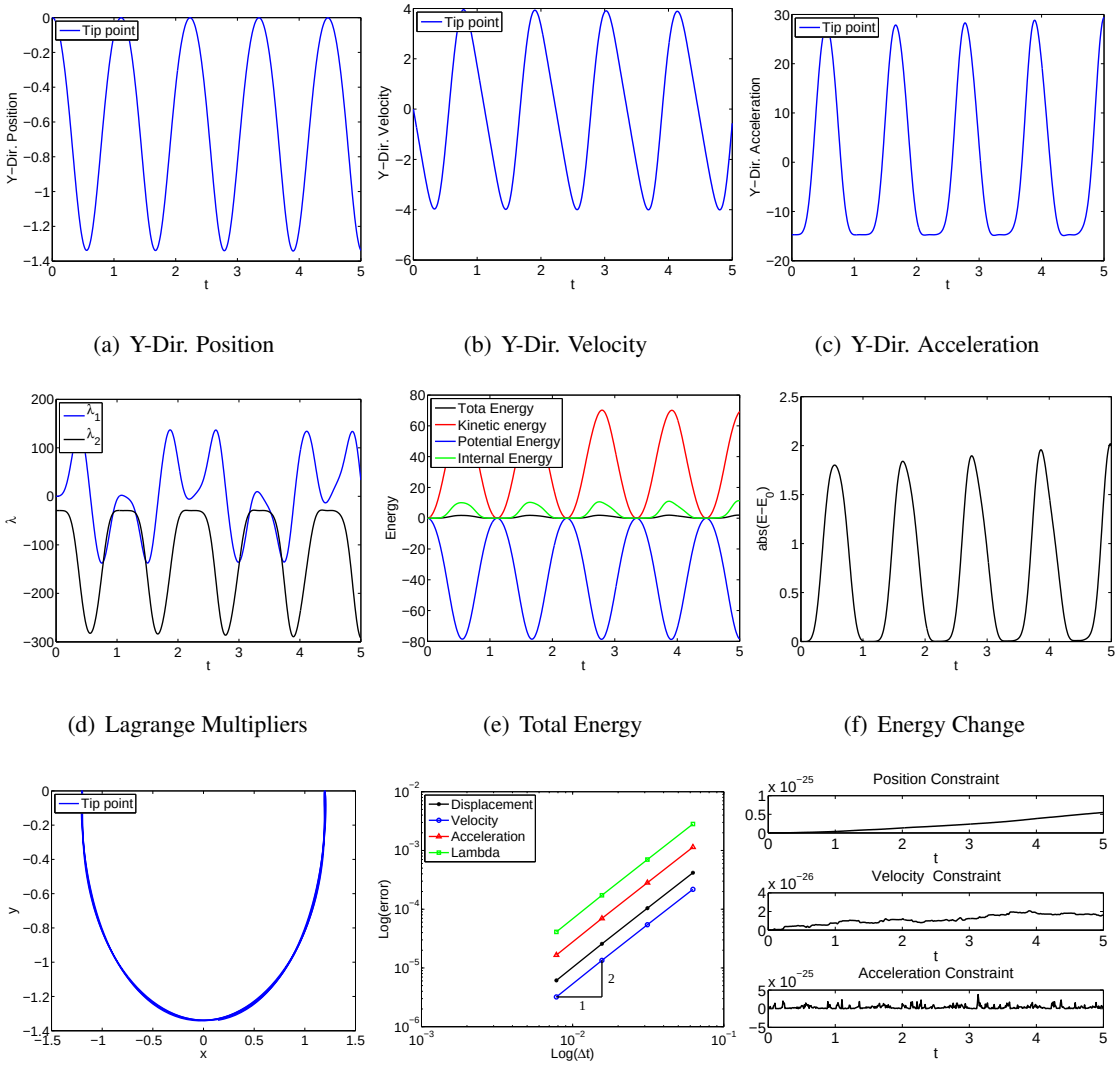


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.78: Single pendulum with bar element in FRF: U0V0(1,1,1) - Index 2.



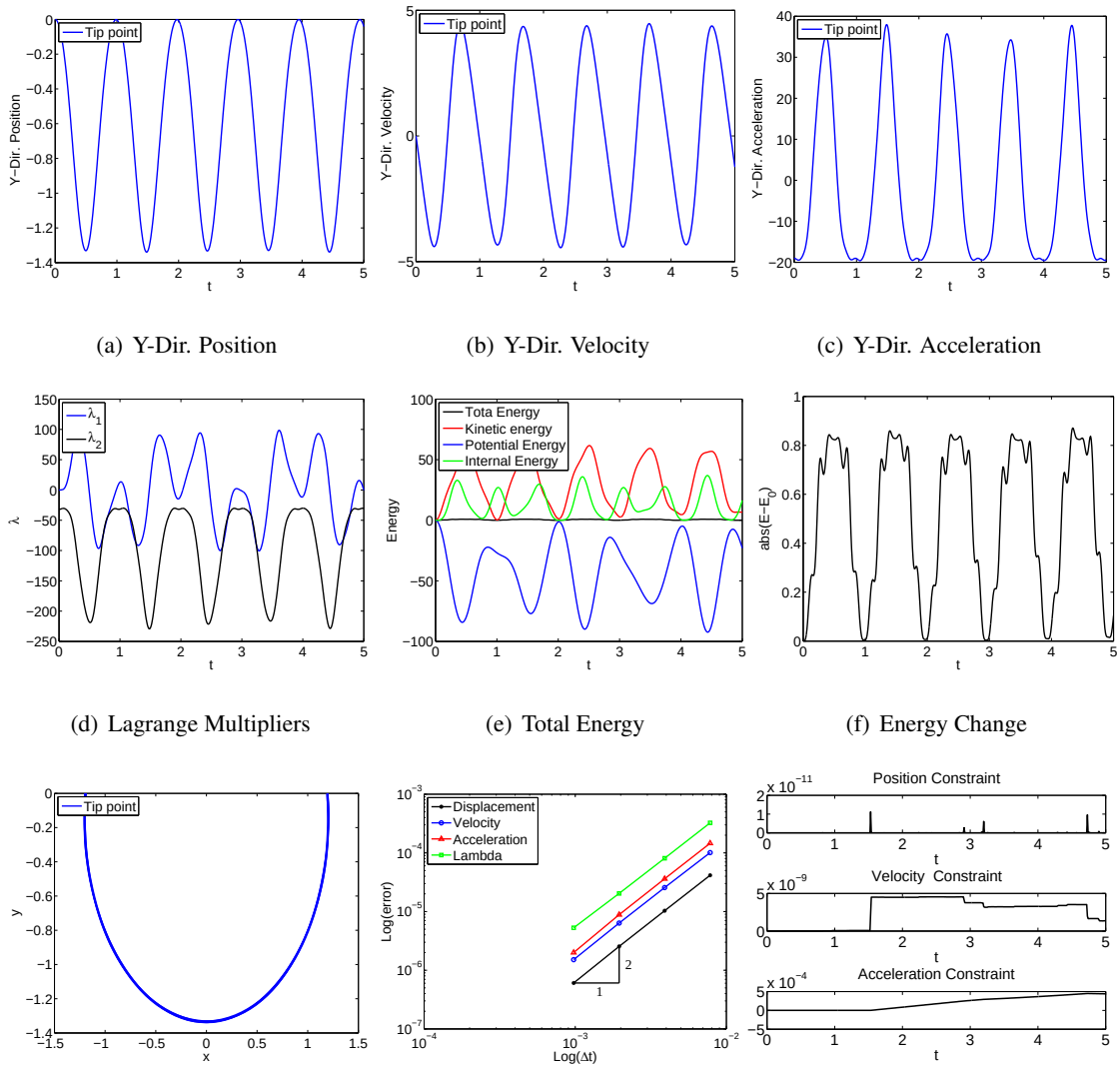
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.79: Single pendulum with bar element in FRF: U0V0(1,1,1) - Index 1.

Euler-Bernoulli Beam

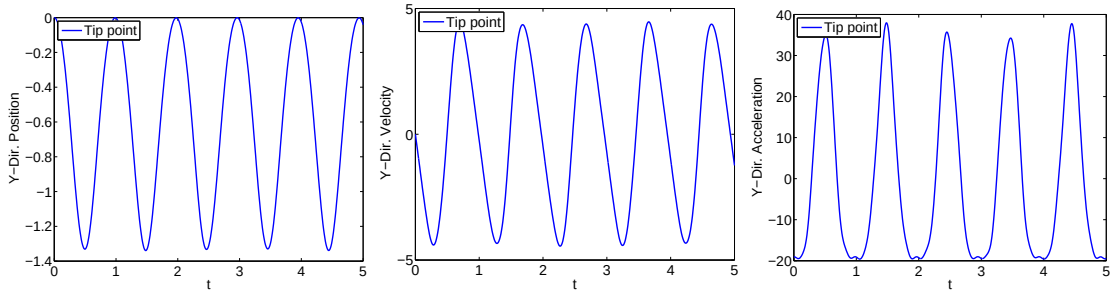


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

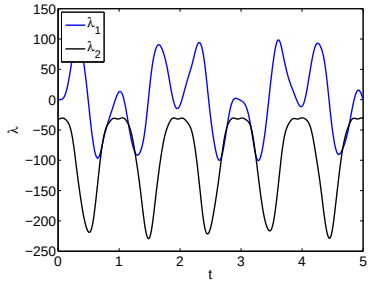
Figure 4.80: Single pendulum with EB element in IRF: U0(1,1,0) - Index 3.



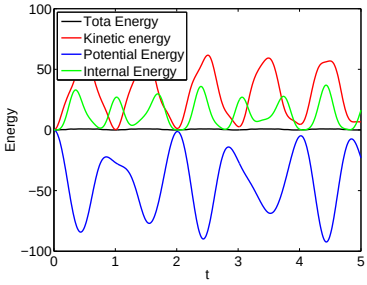
(a) Y-Dir. Position

(b) Y-Dir. Velocity

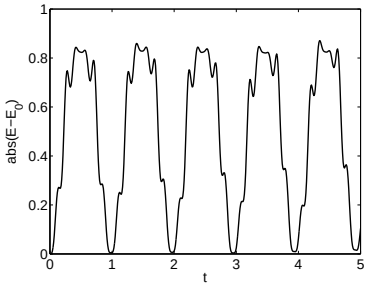
(c) Y-Dir. Acceleration



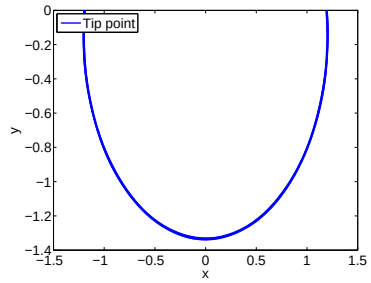
(d) Lagrange Multipliers



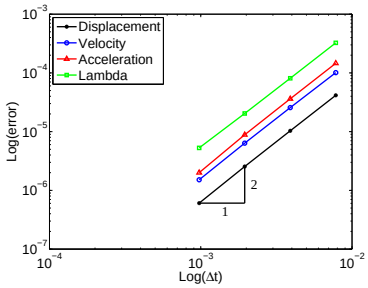
(e) Total Energy



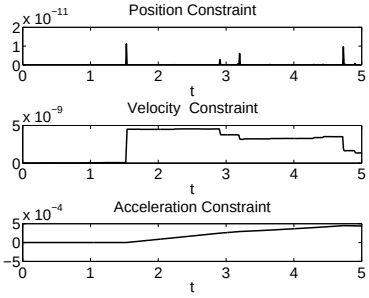
(f) Energy Change



(g) Trajectory

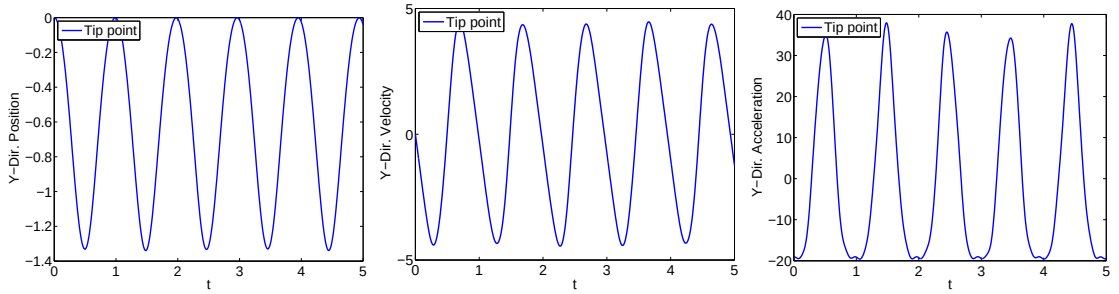


(h) Convergence Rate



(i) Constraint Satisfaction

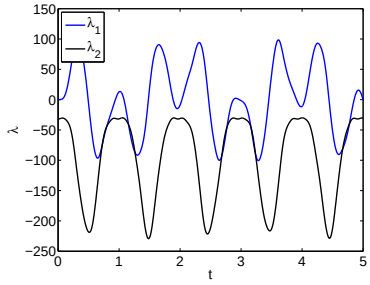
Figure 4.81: Single pendulum with EB element in IRF: U0(1,1,0) - Index 2.



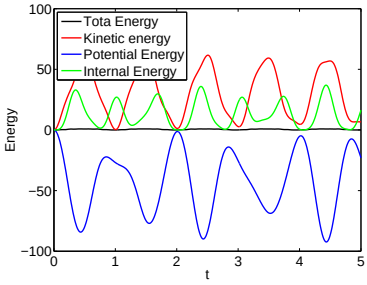
(a) Y-Dir. Position

(b) Y-Dir. Velocity

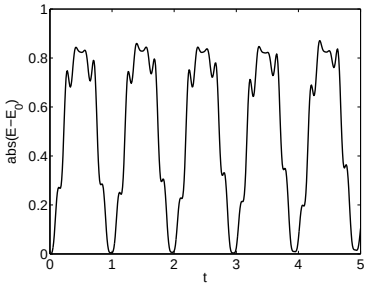
(c) Y-Dir. Acceleration



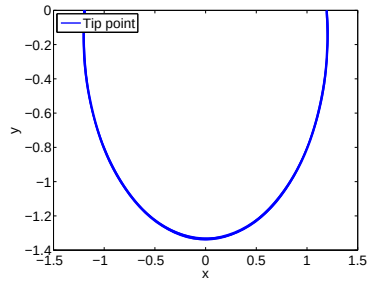
(d) Lagrange Multipliers



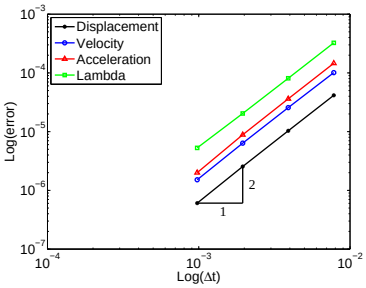
(e) Total Energy



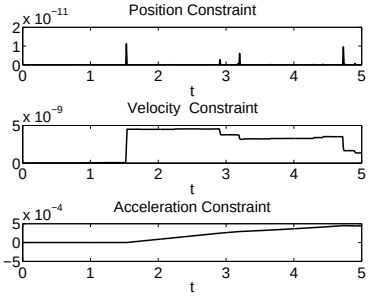
(f) Energy Change



(g) Trajectory

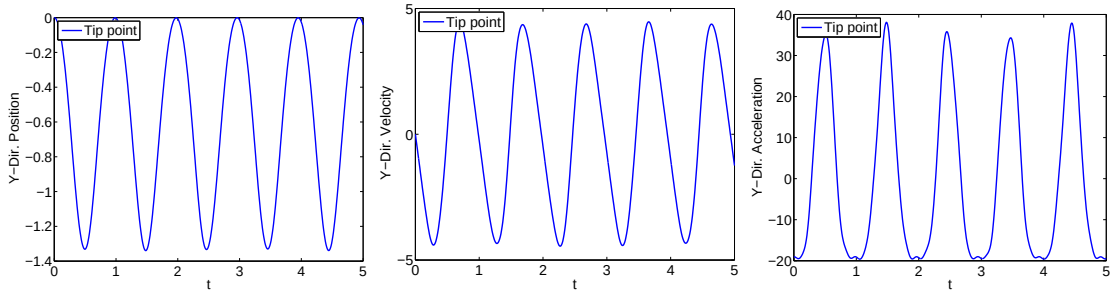


(h) Convergence Rate



(i) Constraint Satisfaction

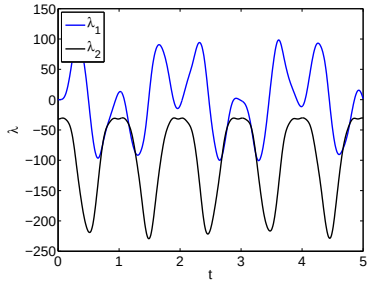
Figure 4.82: Single pendulum with EB element in IRF: U0(1,1,0) - Index 1.



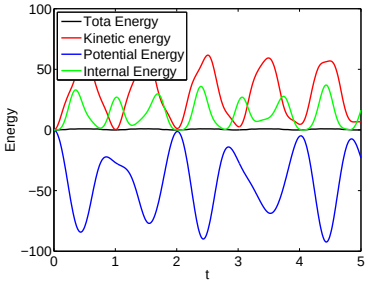
(a) Y-Dir. Position

(b) Y-Dir. Velocity

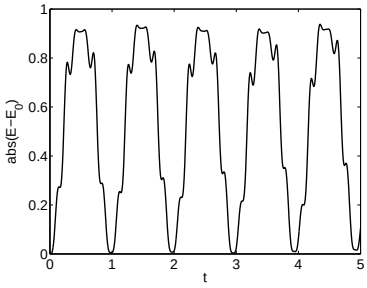
(c) Y-Dir. Acceleration



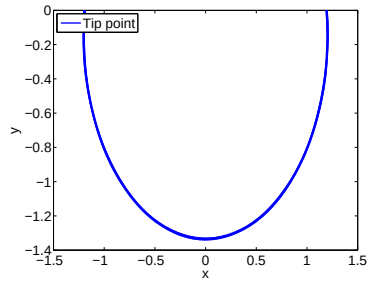
(d) Lagrange Multipliers



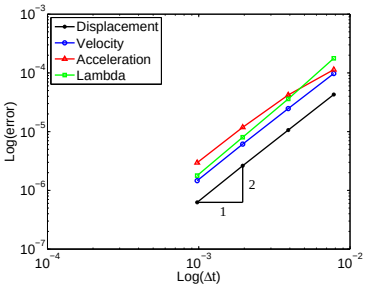
(e) Total Energy



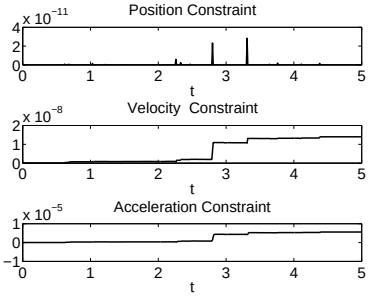
(f) Energy Change



(g) Trajectory

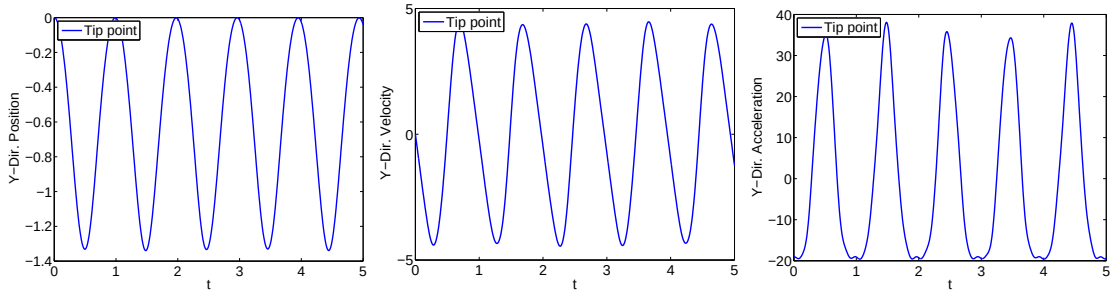


(h) Convergence Rate



(i) Constraint Satisfaction

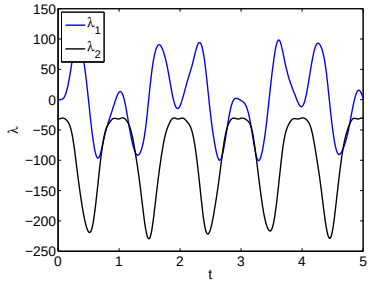
Figure 4.83: Single pendulum with EB element in IRF: V0(1,1,0) - Index 3.



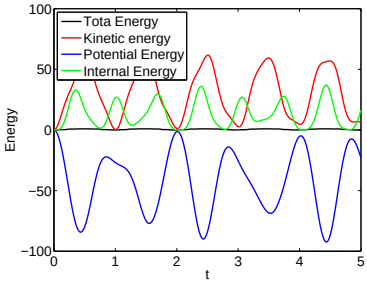
(a) Y-Dir. Position

(b) Y-Dir. Velocity

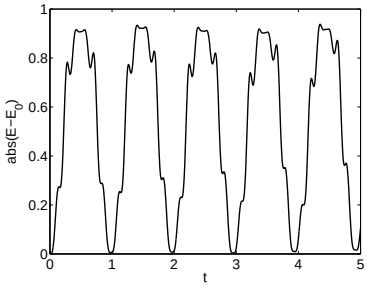
(c) Y-Dir. Acceleration



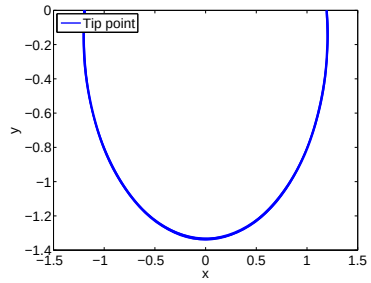
(d) Lagrange Multipliers



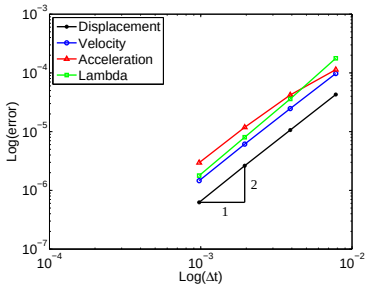
(e) Total Energy



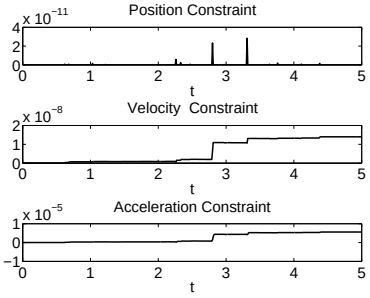
(f) Energy Change



(g) Trajectory

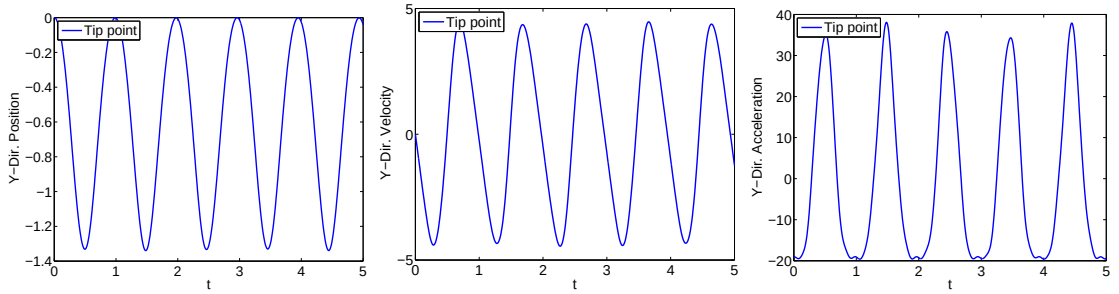


(h) Convergence Rate



(i) Constraint Satisfaction

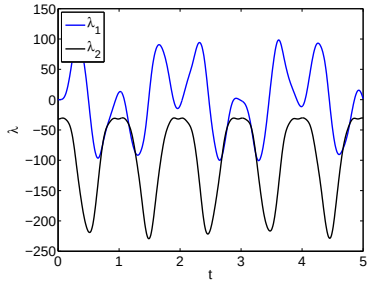
Figure 4.84: Single pendulum with EB element in IRF: V0(1,1,0) - Index 2.



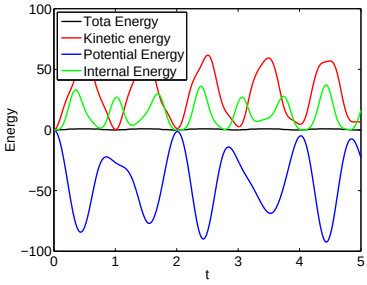
(a) Y-Dir. Position

(b) Y-Dir. Velocity

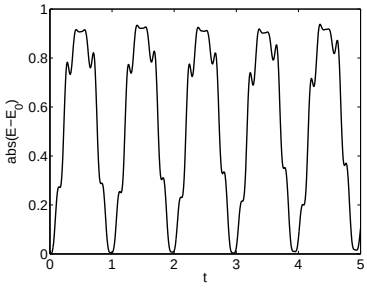
(c) Y-Dir. Acceleration



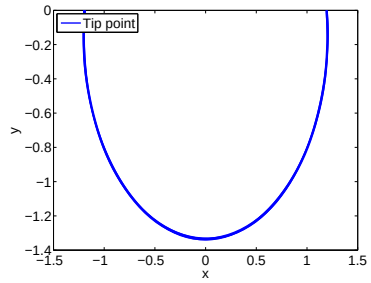
(d) Lagrange Multipliers



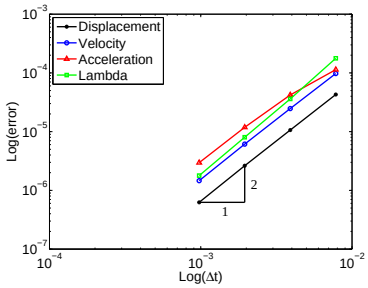
(e) Total Energy



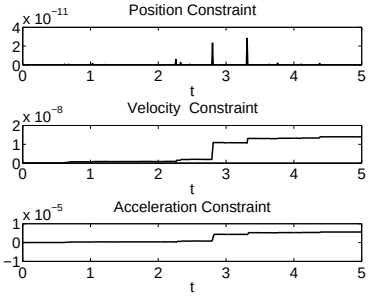
(f) Energy Change



(g) Trajectory

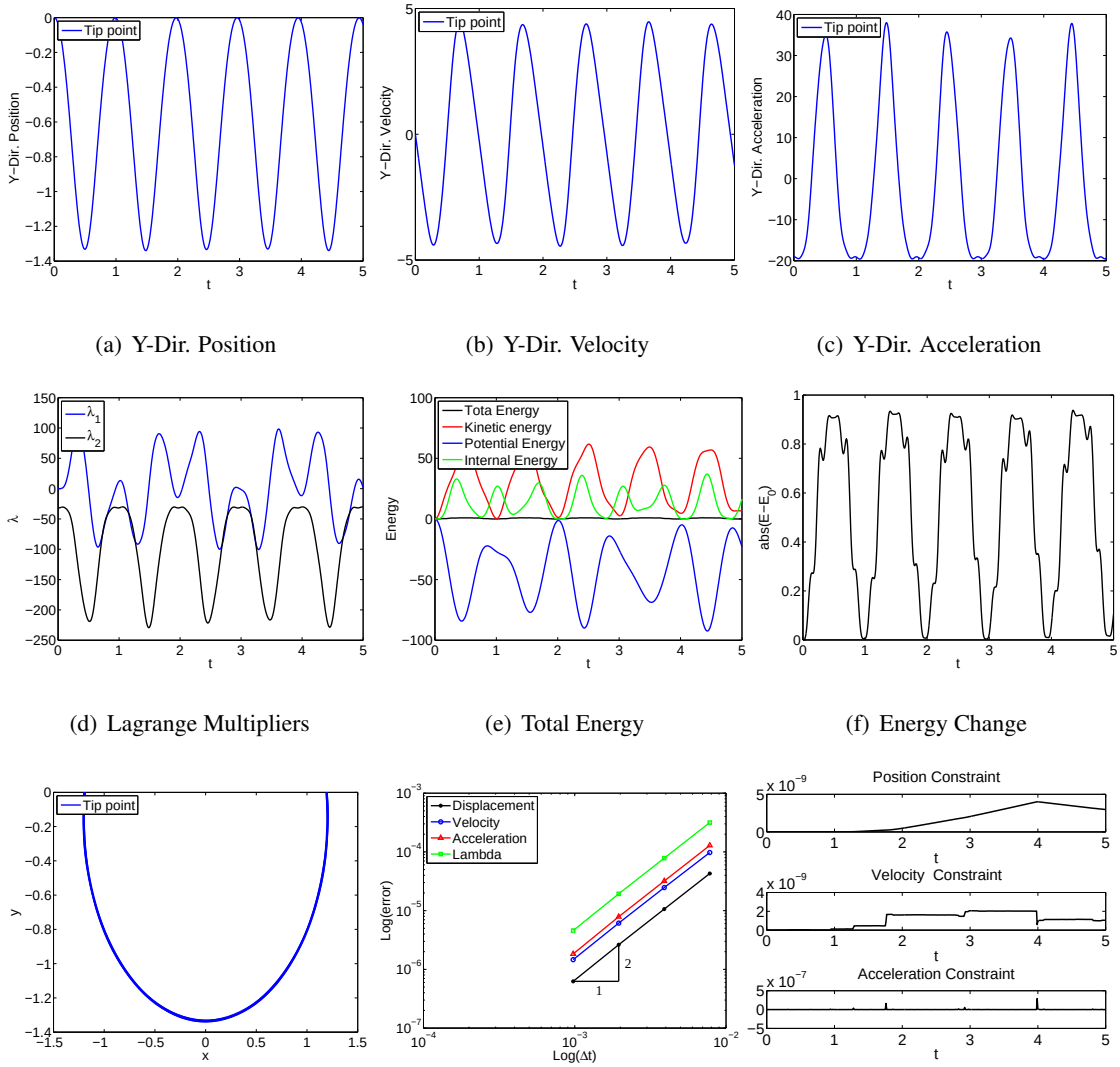


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.85: Single pendulum with EB element in IRF: V0(1,1,0) - Index 1.

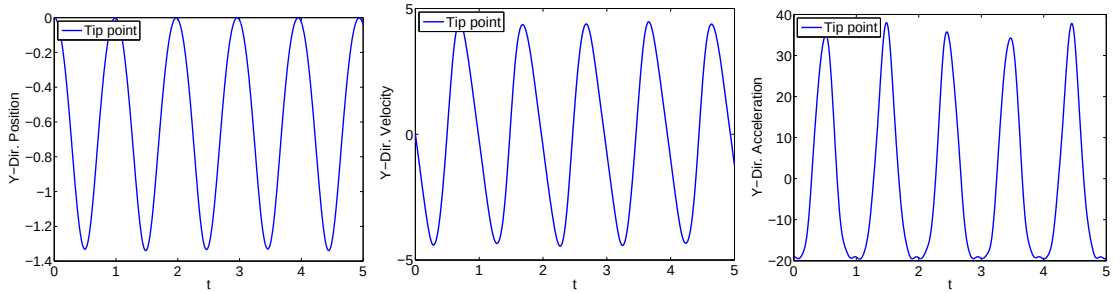


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

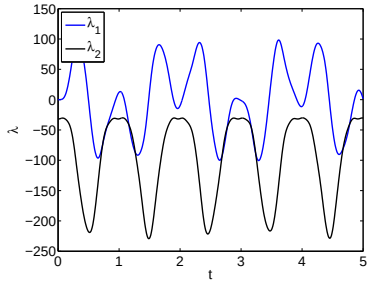
Figure 4.86: Single pendulum with EB element in IRF: U0V0(1,1,1) - Index 3.



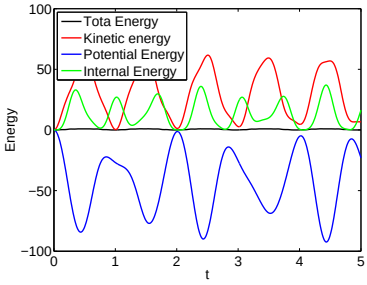
(a) Y-Dir. Position

(b) Y-Dir. Velocity

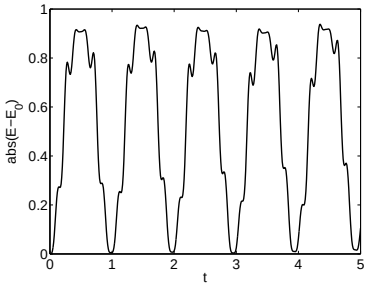
(c) Y-Dir. Acceleration



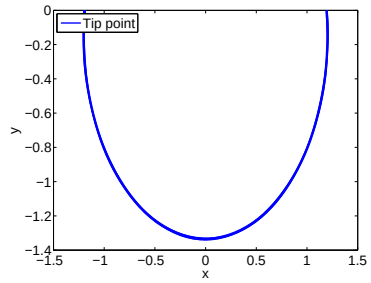
(d) Lagrange Multipliers



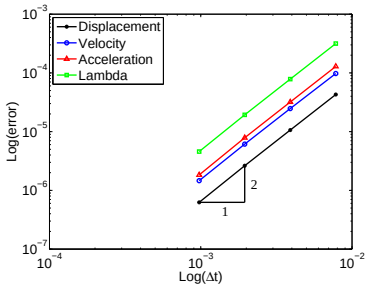
(e) Total Energy



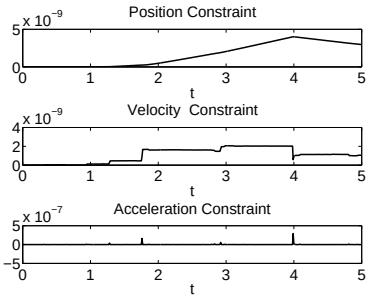
(f) Energy Change



(g) Trajectory

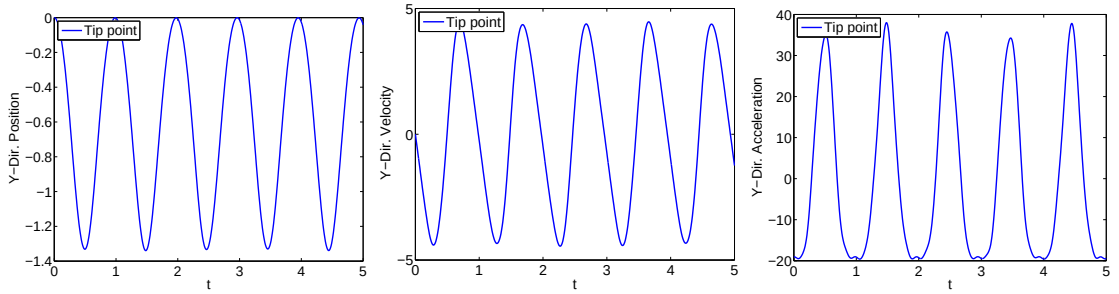


(h) Convergence Rate



(i) Constraint Satisfaction

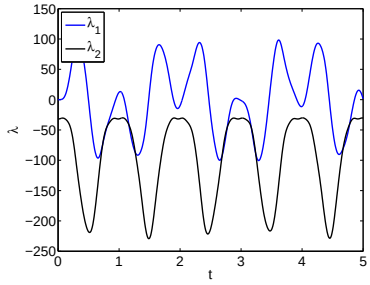
Figure 4.87: Single pendulum with EB element in IRF: U0V0(1,1,1) - Index 2.



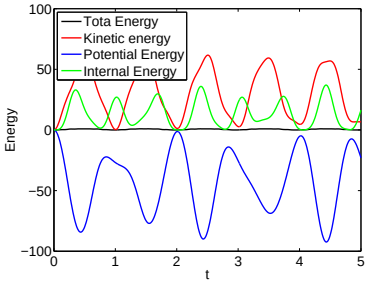
(a) Y-Dir. Position

(b) Y-Dir. Velocity

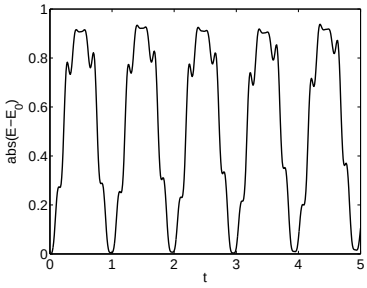
(c) Y-Dir. Acceleration



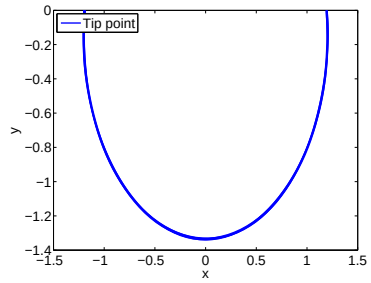
(d) Lagrange Multipliers



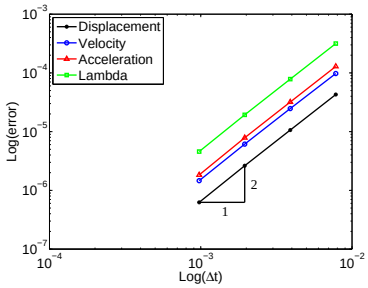
(e) Total Energy



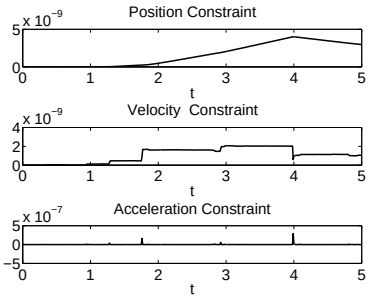
(f) Energy Change



(g) Trajectory

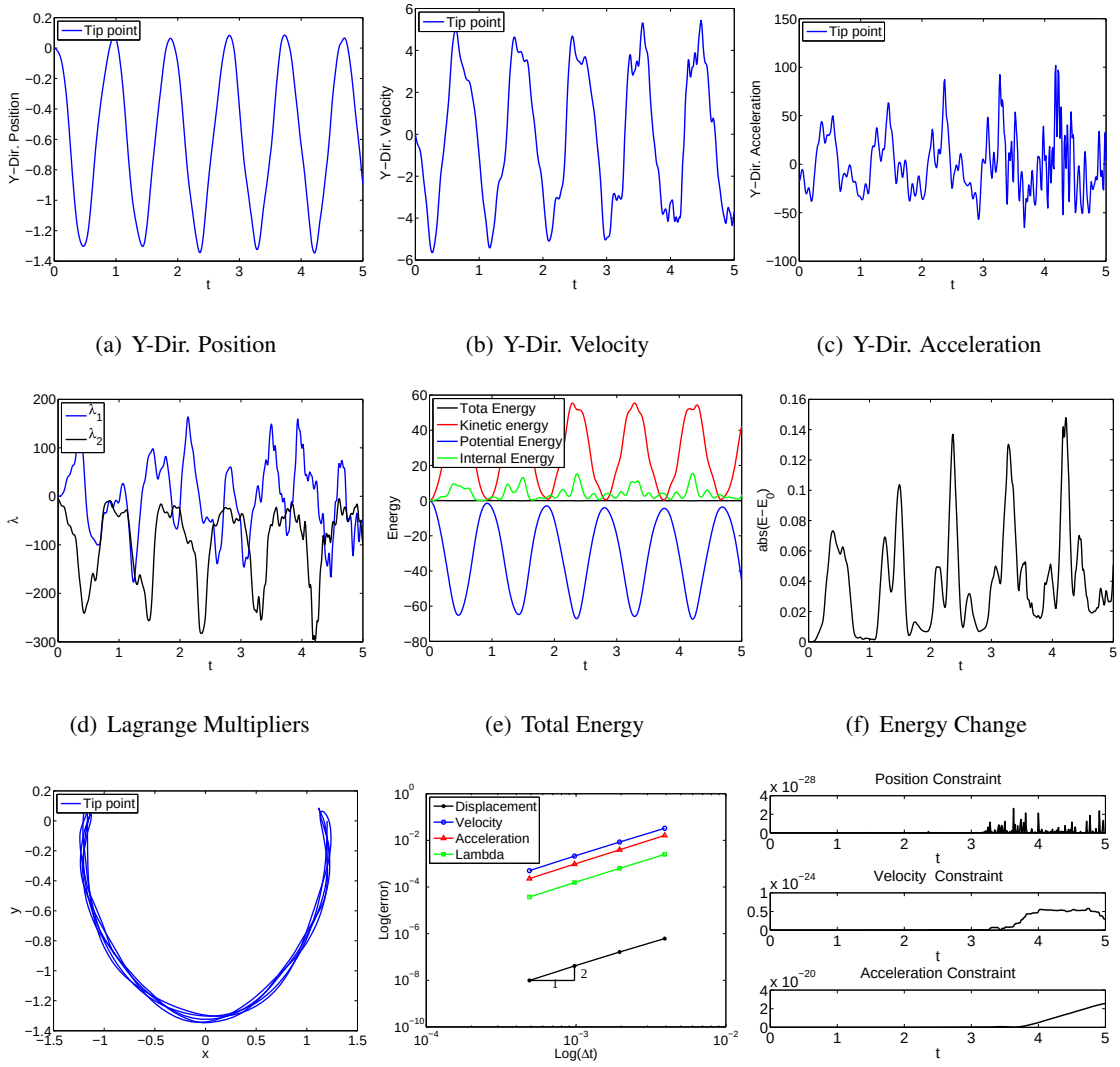


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.88: Single pendulum with EB element in IRF: U0V0(1,1,1) - Index 1.

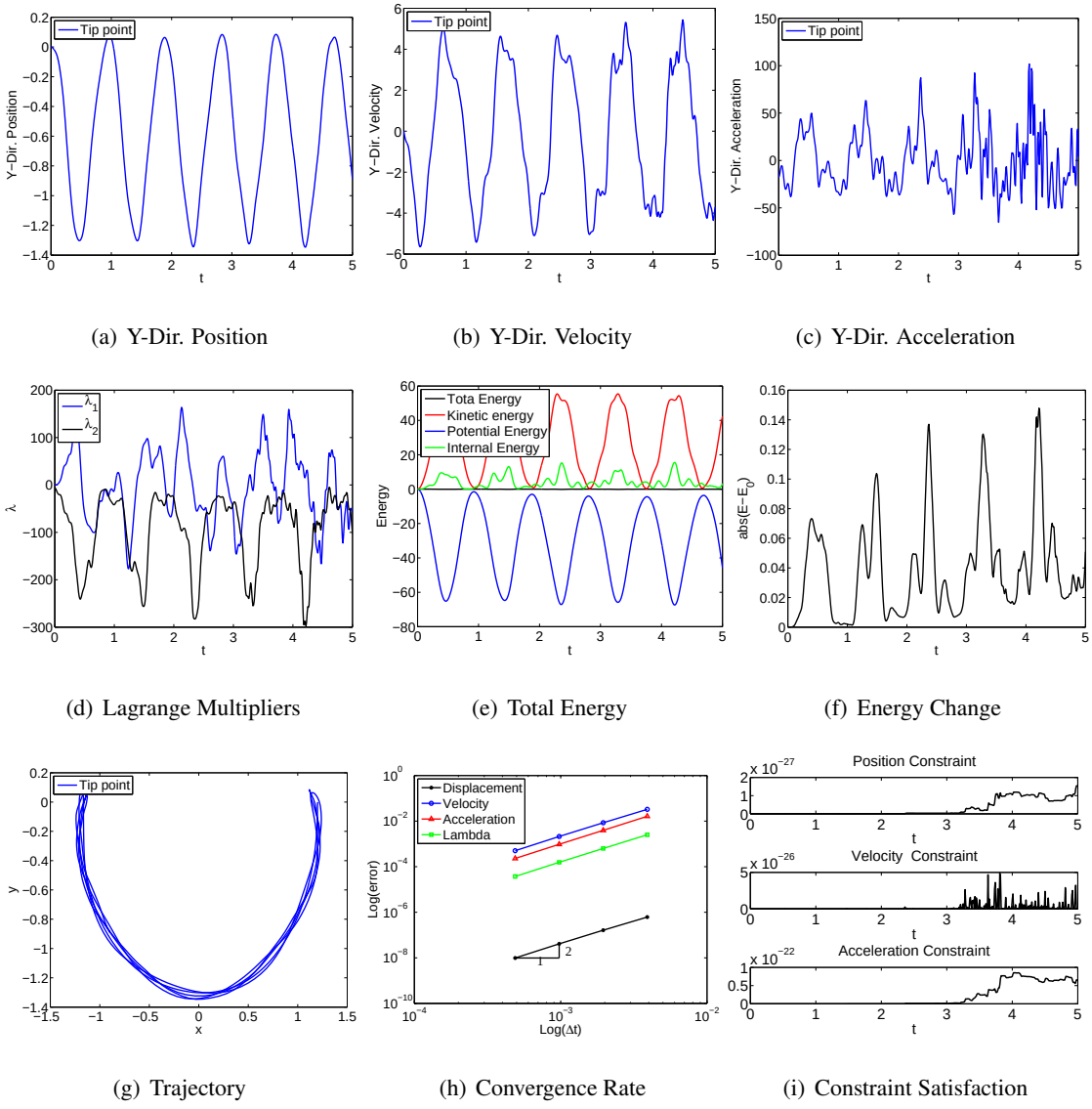


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.89: Single pendulum with EB element in ANCF-S: U0(1,1,0) - Index 3.

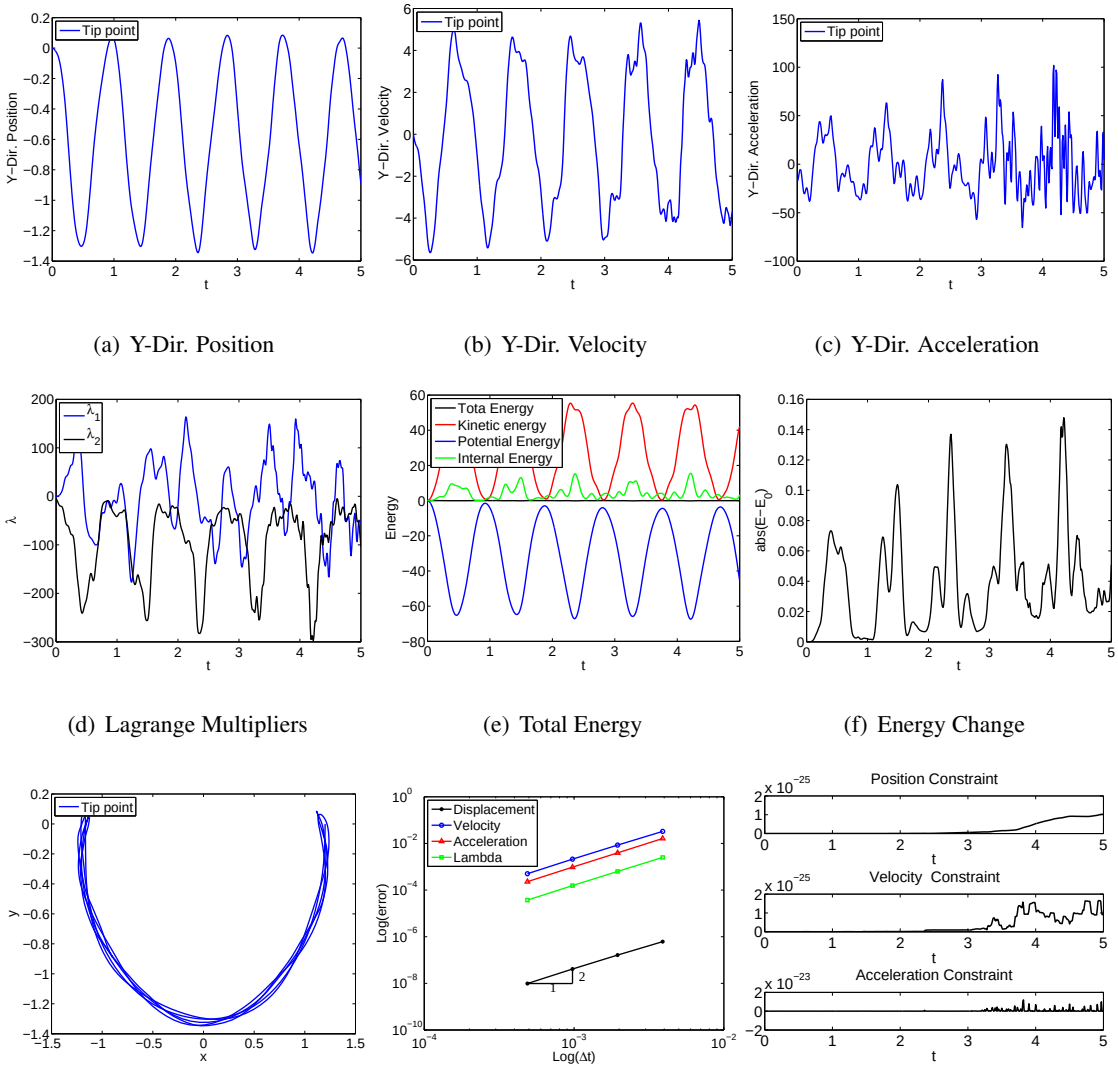


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.90: Single pendulum with EB element in ANCF-S: U0(1,1,0) - Index 2.

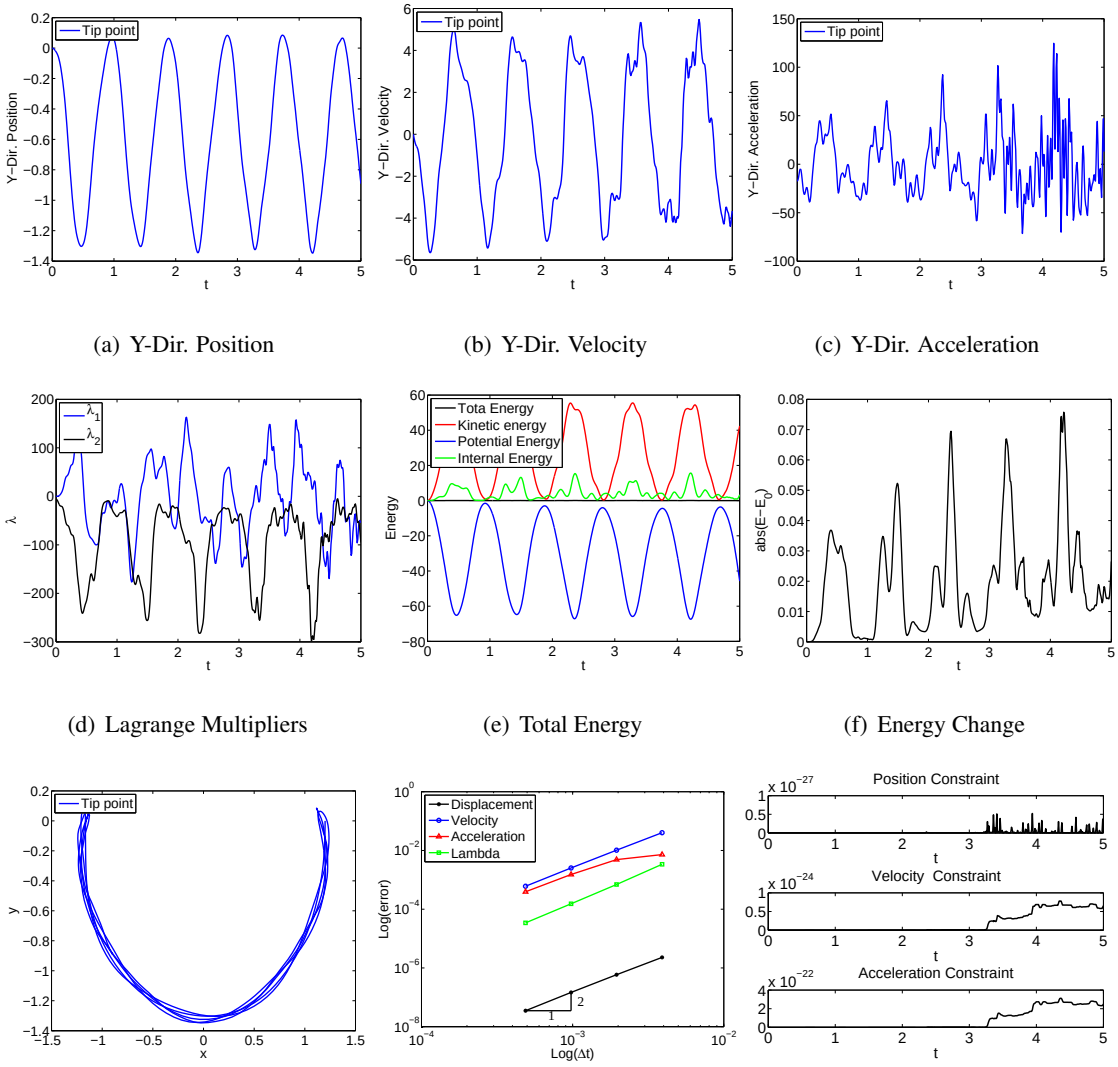


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.91: Single pendulum with EB element in ANCF-S: U0(1,1,0) - Index 1.

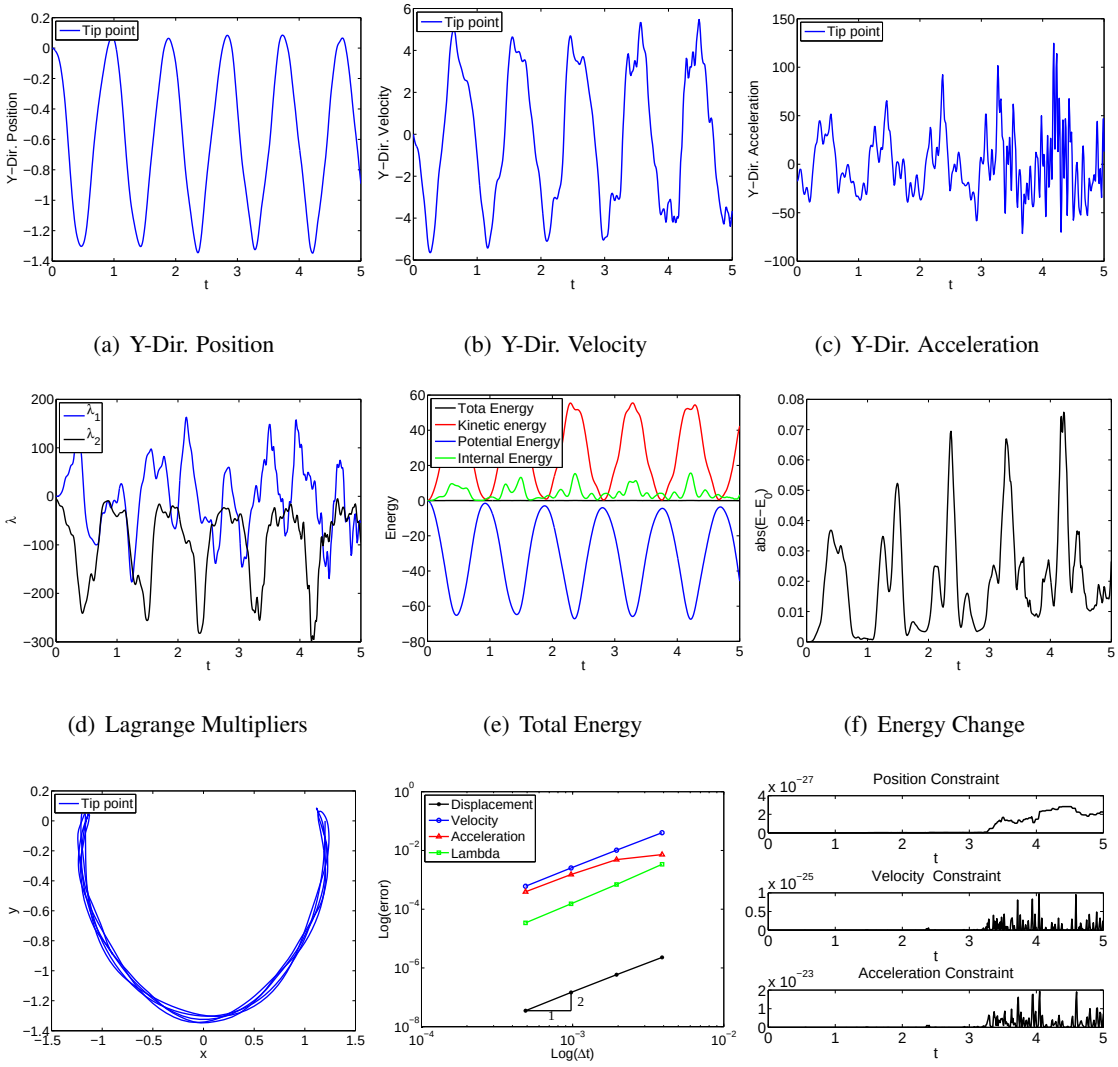


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.92: Single pendulum with EB element in ANCF-S: V0(1,1,0) - Index 3.

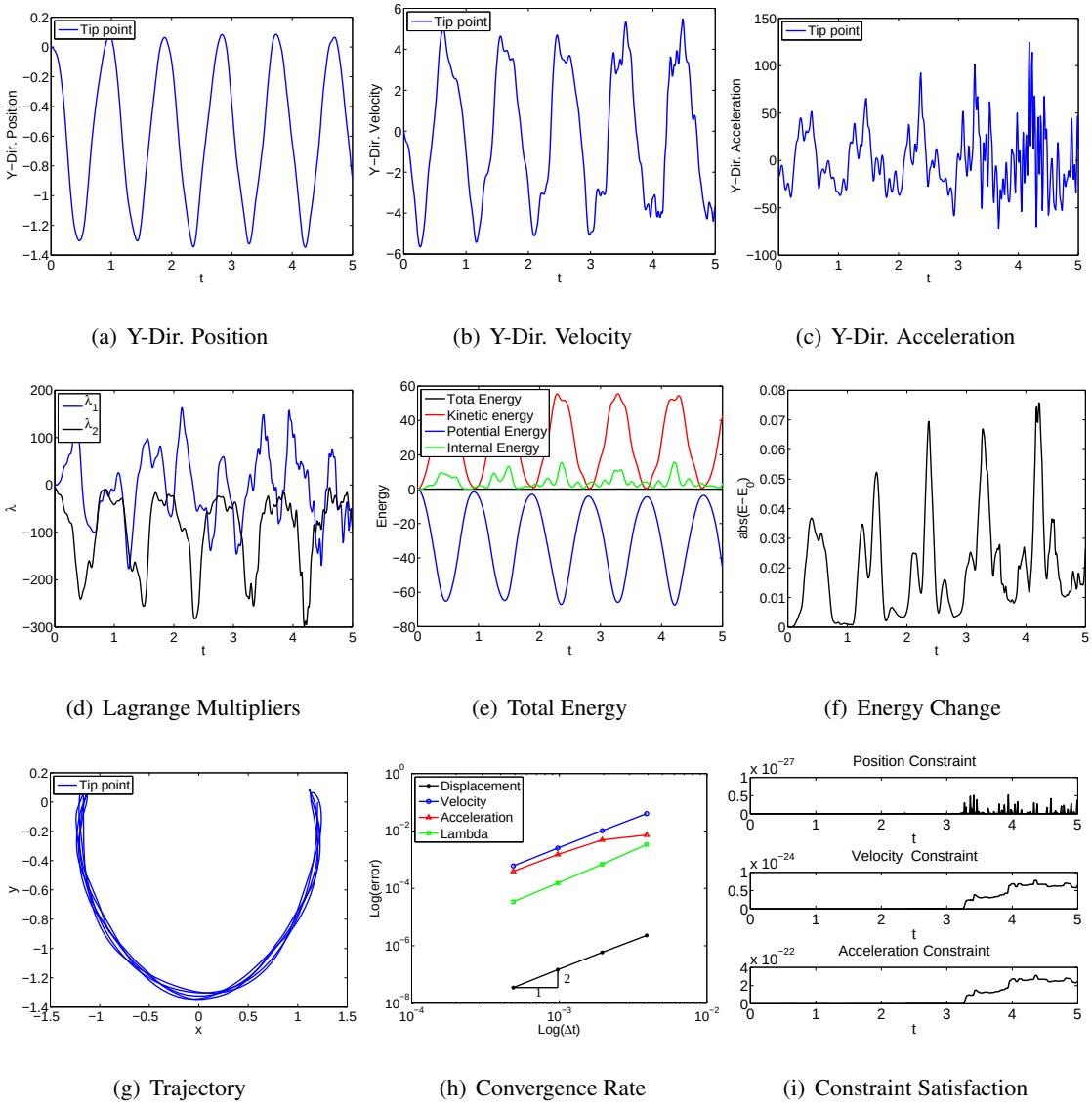


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.93: Single pendulum with EB element in ANCF-S: V0(1,1,0) - Index 2.

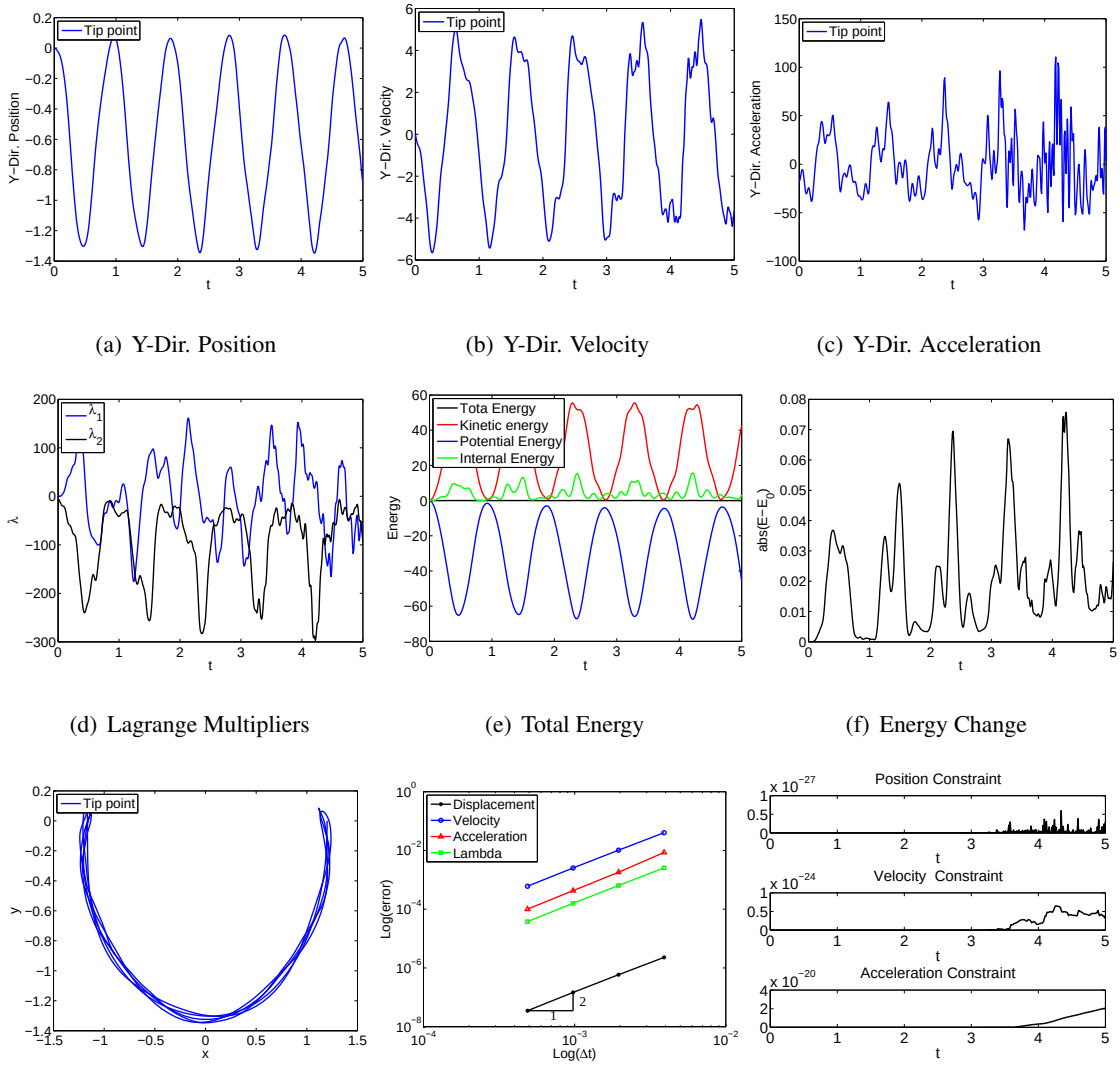


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.94: Single pendulum with EB element in ANCF-S: V0(1,1,0) - Index 1.

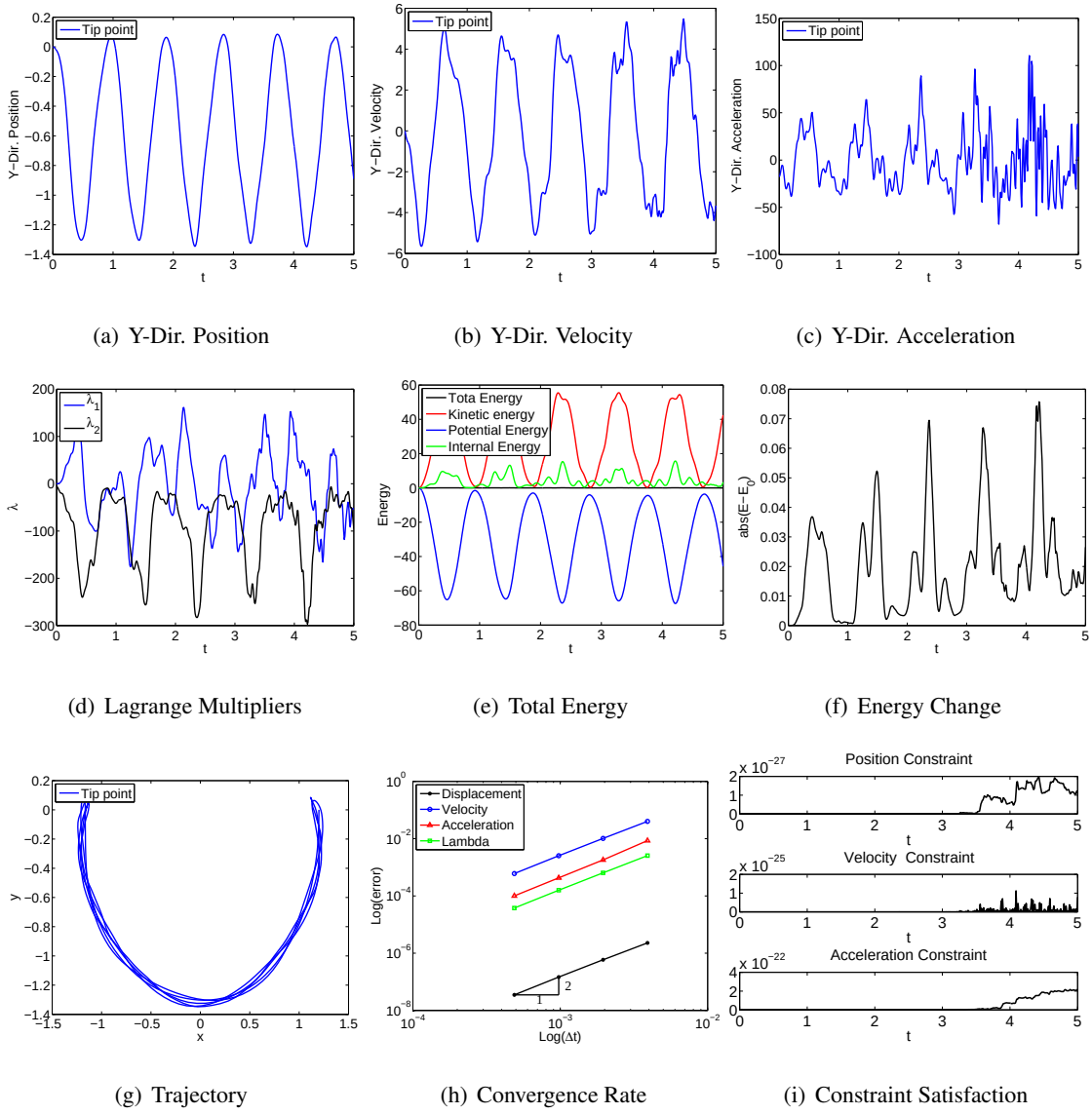


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.95: Single pendulum with EB element in ANCF-S: U0V0(1,1,1) - Index 3.

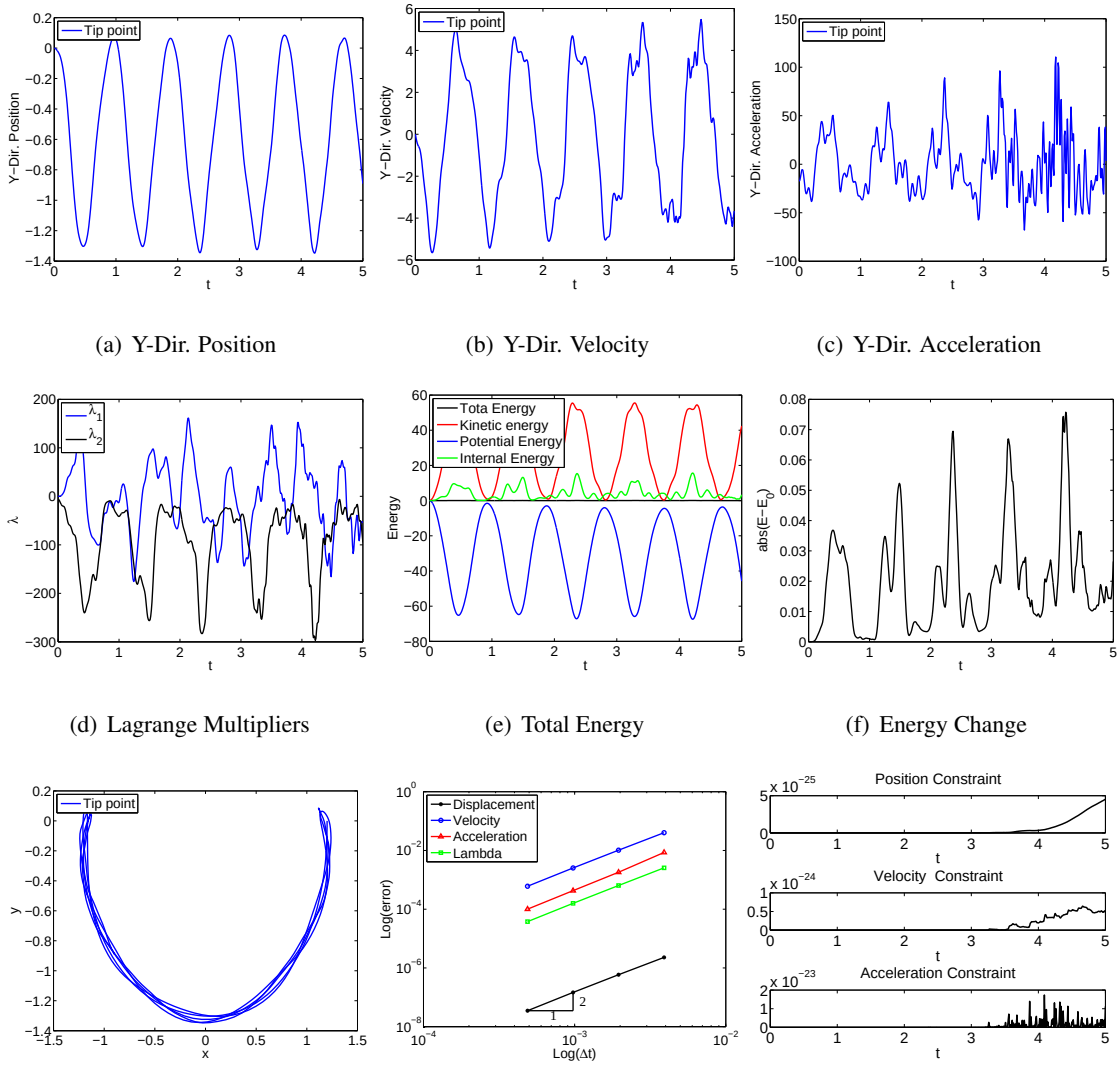


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.96: Single pendulum with EB element in ANCF-S: U0V0(1,1,1) - Index 2.

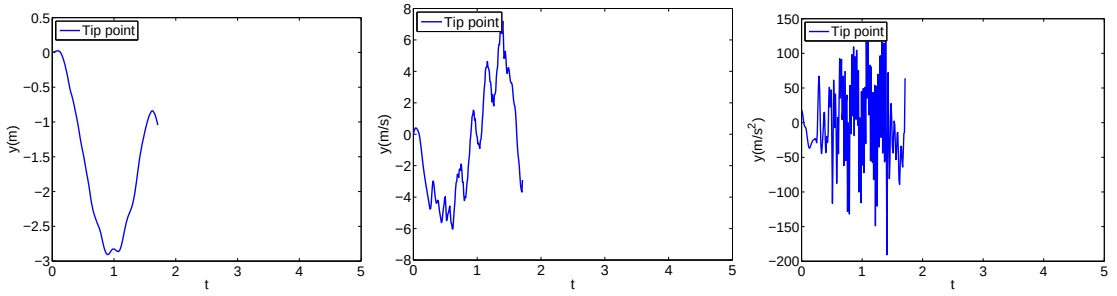


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

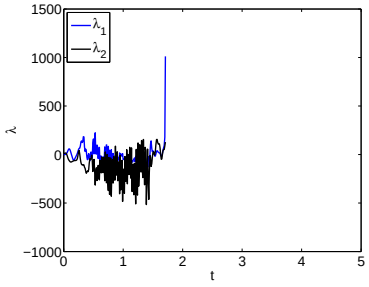
Figure 4.97: Single pendulum with EB element in ANCF-S: U0V0(1,1,1) - Index 1.



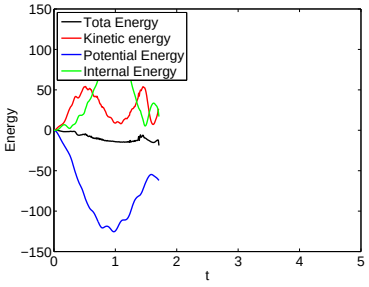
(a) Y-Dir. Position

(b) Y-Dir. Velocity

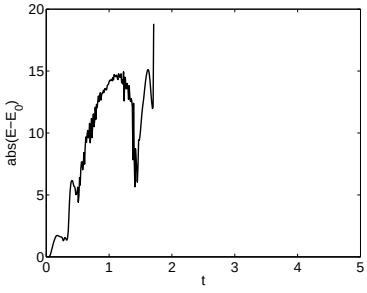
(c) Y-Dir. Acceleration



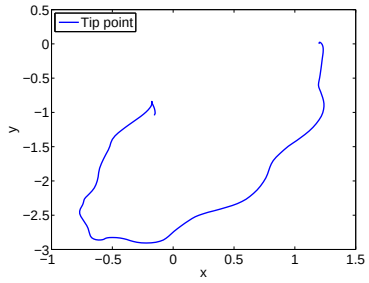
(d) Lagrange Multipliers



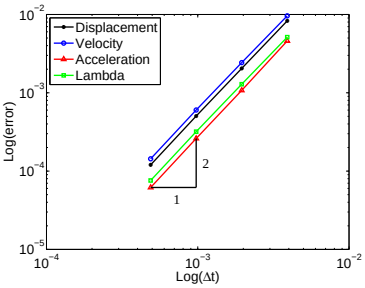
(e) Total Energy



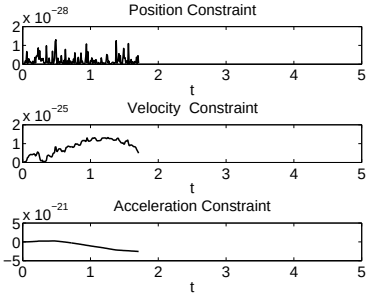
(f) Energy Change



(g) Trajectory

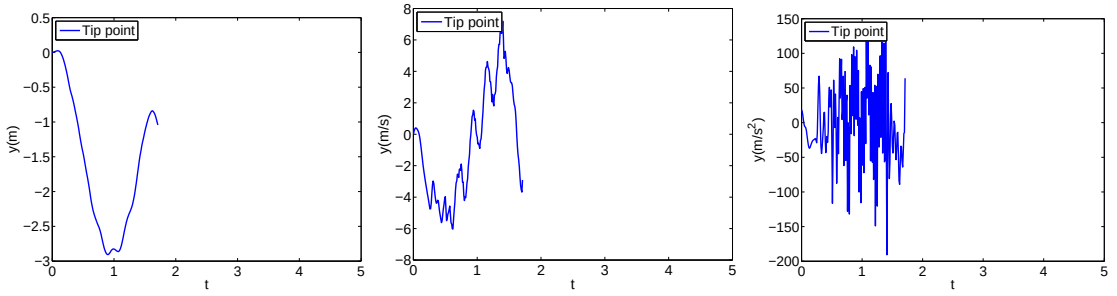


(h) Convergence Rate



(i) Constraint Satisfaction

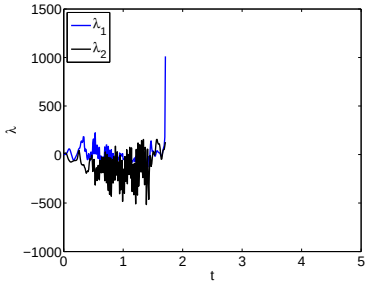
Figure 4.98: Single pendulum with EB element in FRF: U0(1,1,0) - Index 3.



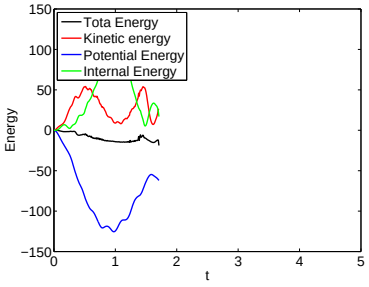
(a) Y-Dir. Position

(b) Y-Dir. Velocity

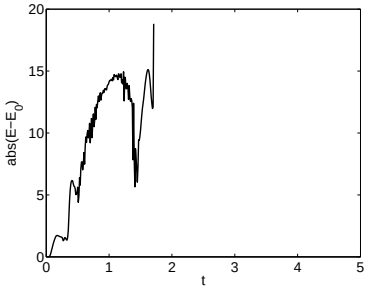
(c) Y-Dir. Acceleration



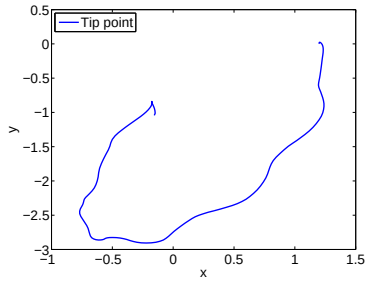
(d) Lagrange Multipliers



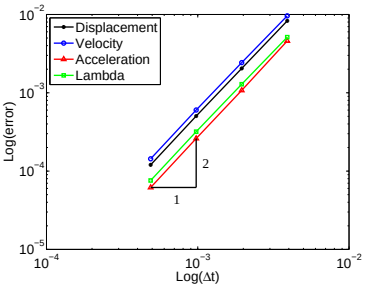
(e) Total Energy



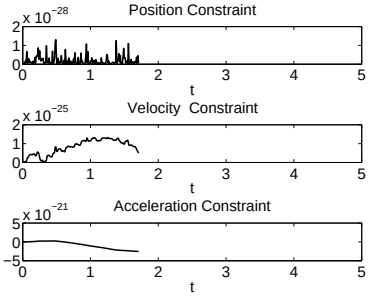
(f) Energy Change



(g) Trajectory

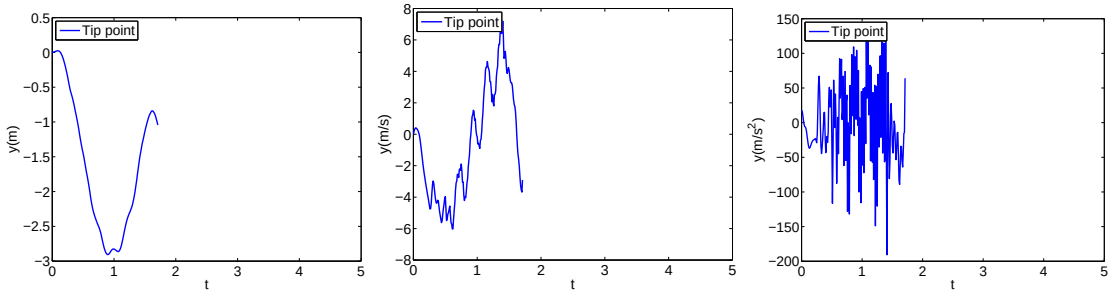


(h) Convergence Rate



(i) Constraint Satisfaction

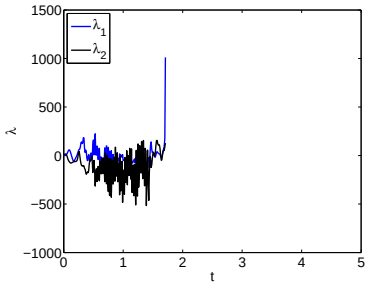
Figure 4.99: Single pendulum with EB element in FRF: U0(1,1,0) - Index 2.



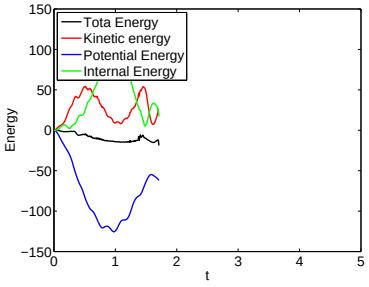
(a) Y-Dir. Position

(b) Y-Dir. Velocity

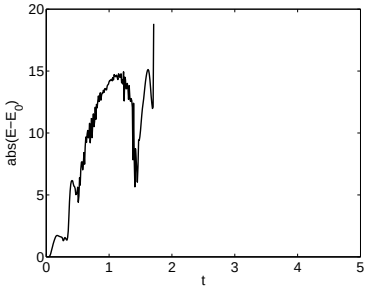
(c) Y-Dir. Acceleration



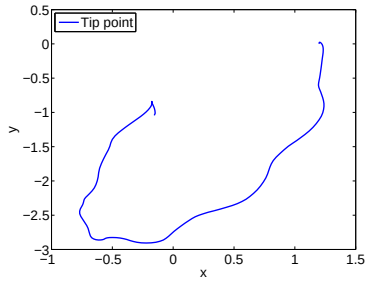
(d) Lagrange Multipliers



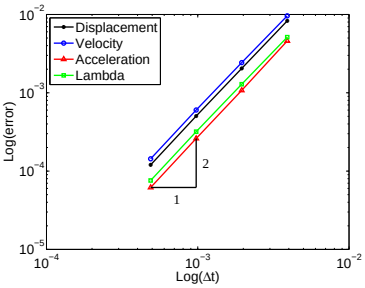
(e) Total Energy



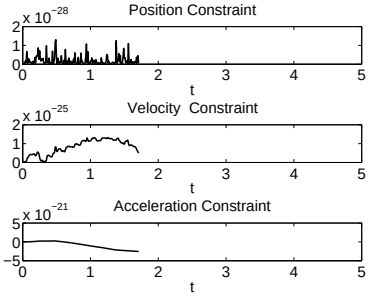
(f) Energy Change



(g) Trajectory

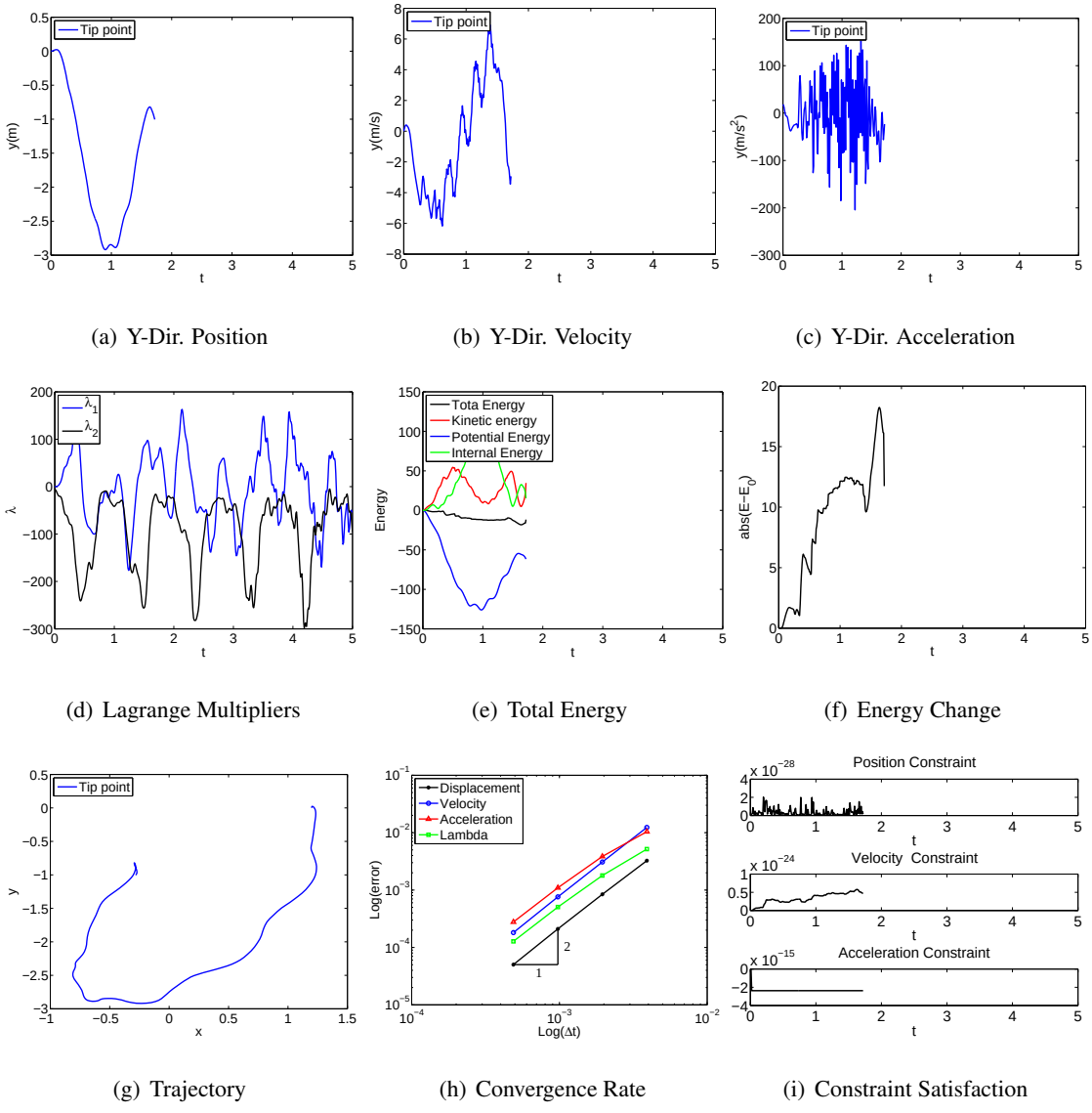


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.100: Single pendulum with EB element in FRF: U0(1,1,0) - Index 1.

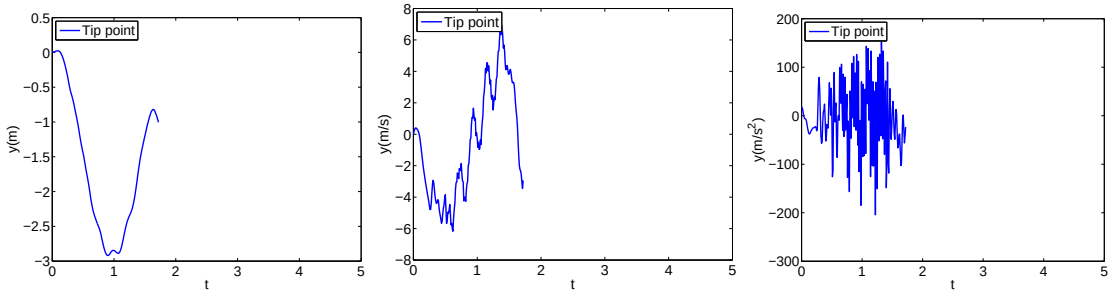


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

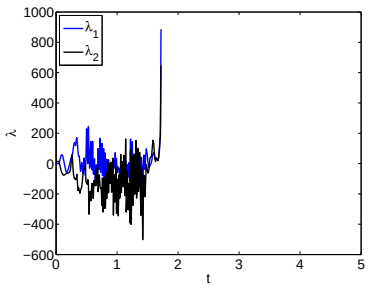
Figure 4.101: Single pendulum with EB element in FRF: V0(1,1,0) - Index 3.



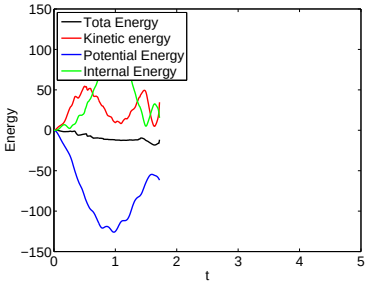
(a) Y-Dir. Position

(b) Y-Dir. Velocity

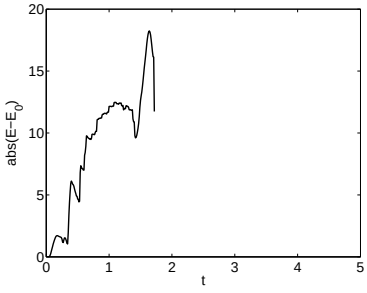
(c) Y-Dir. Acceleration



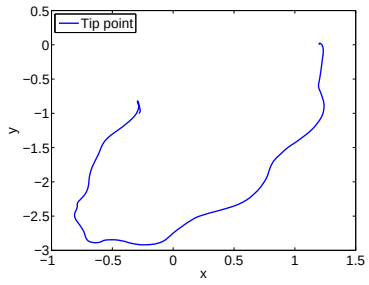
(d) Lagrange Multipliers



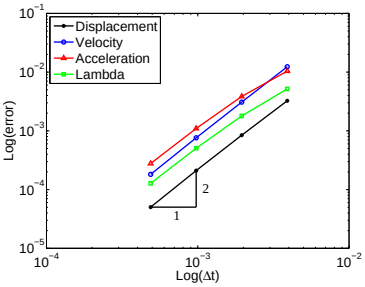
(e) Total Energy



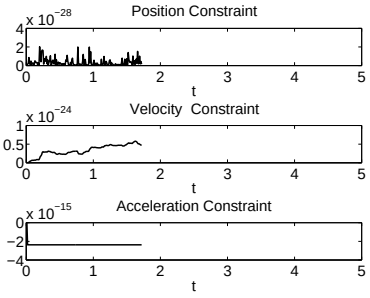
(f) Energy Change



(g) Trajectory

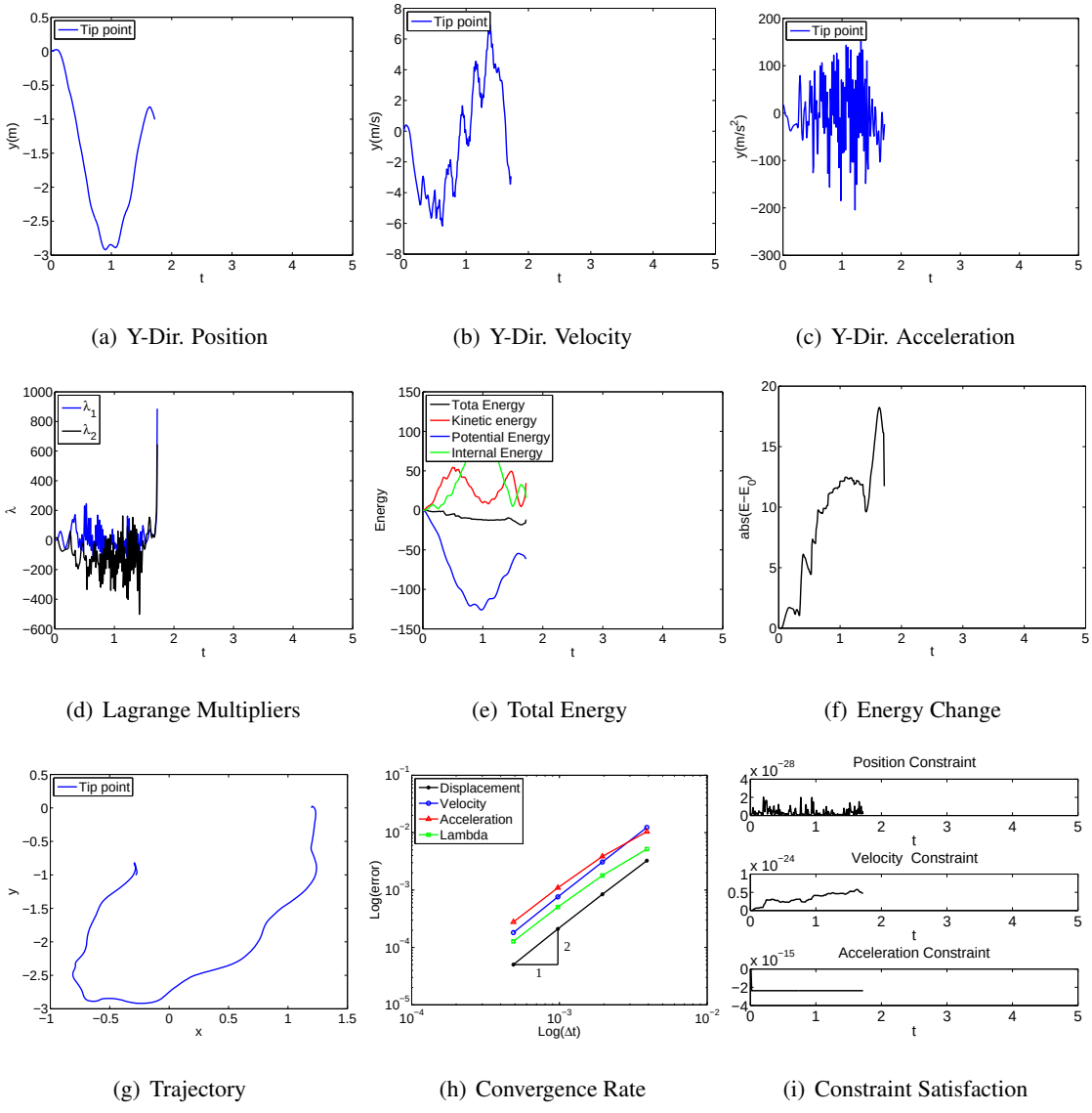


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.102: Single pendulum with EB element in FRF: V0(1,1,0) - Index 2.

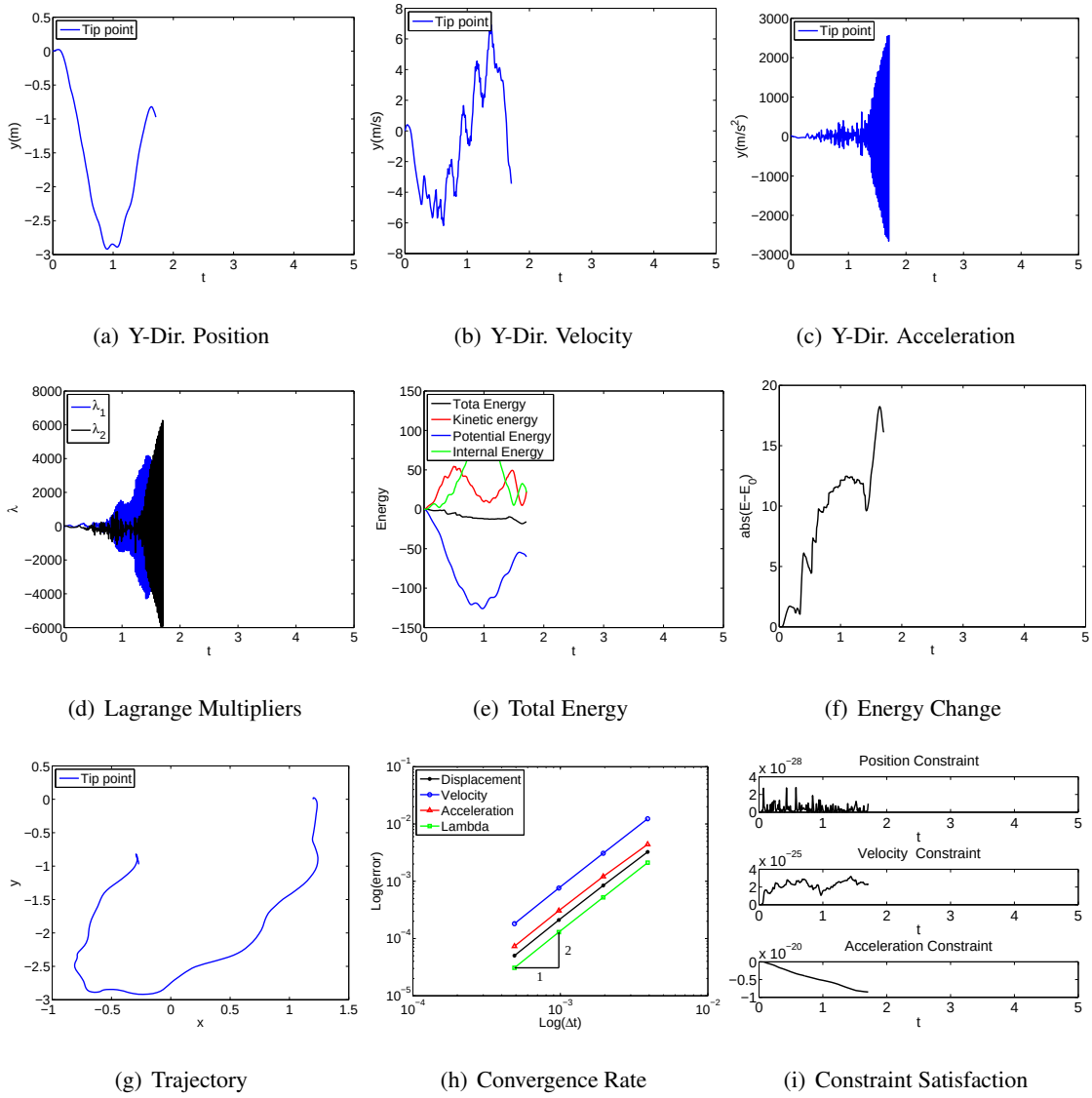


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.103: Single pendulum with EB element in FRF: V0(1,1,0) - Index 1.

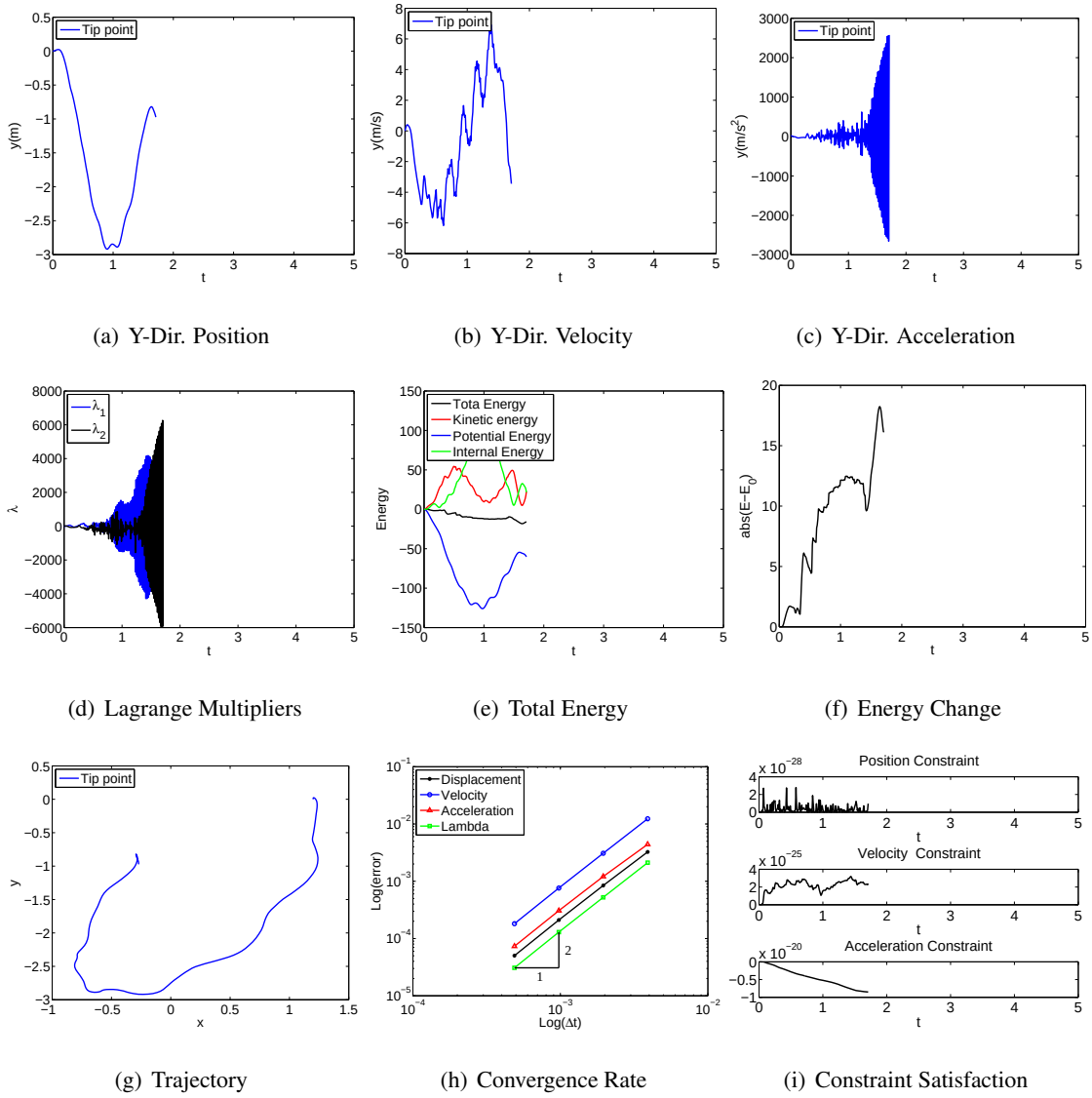


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.104: Single pendulum with EB element in FRF: U0V0(1,1,1) - Index 3.

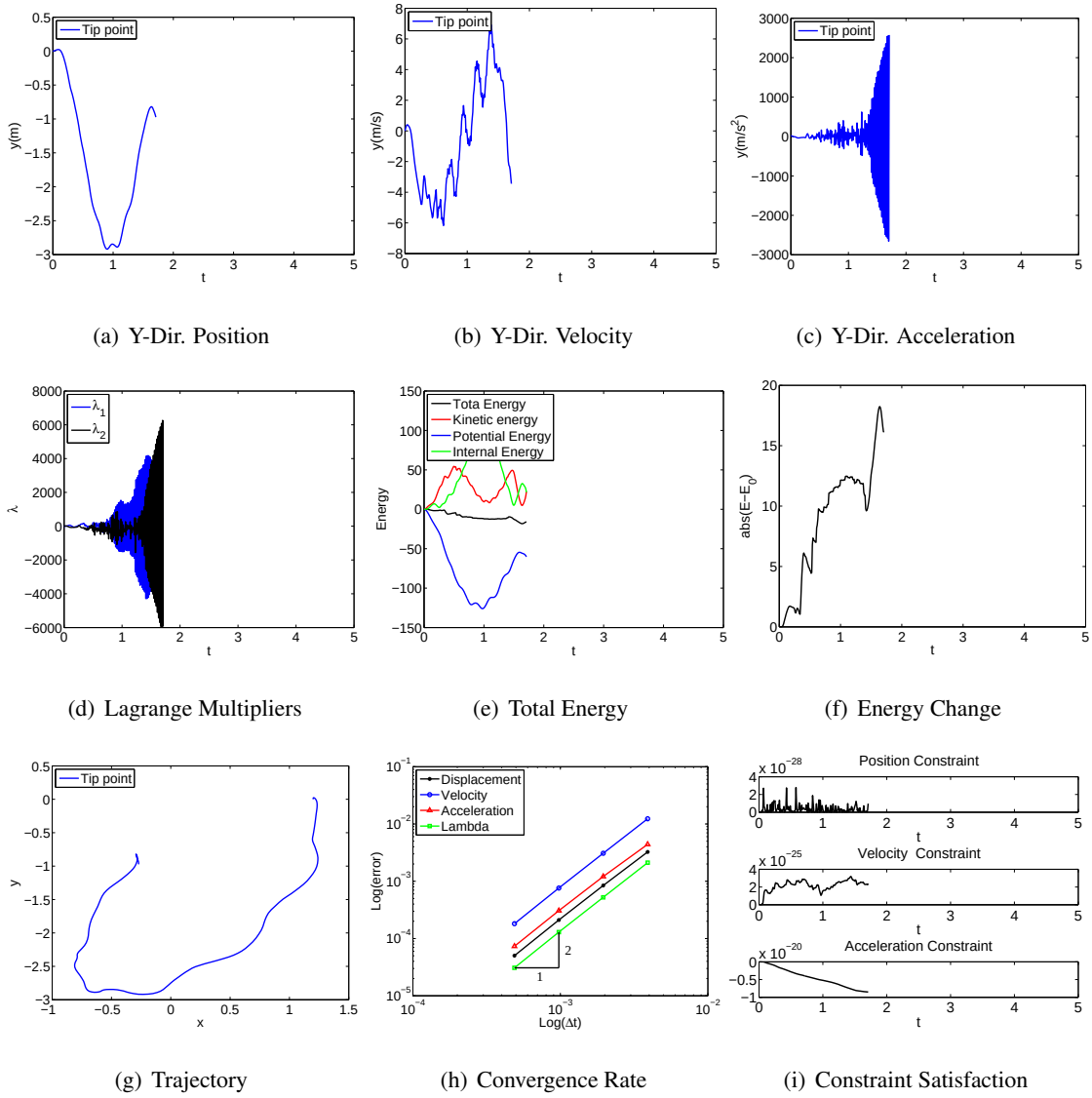


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.105: Single pendulum with EB element in FRF: U0V0(1,1,1) - Index 2.



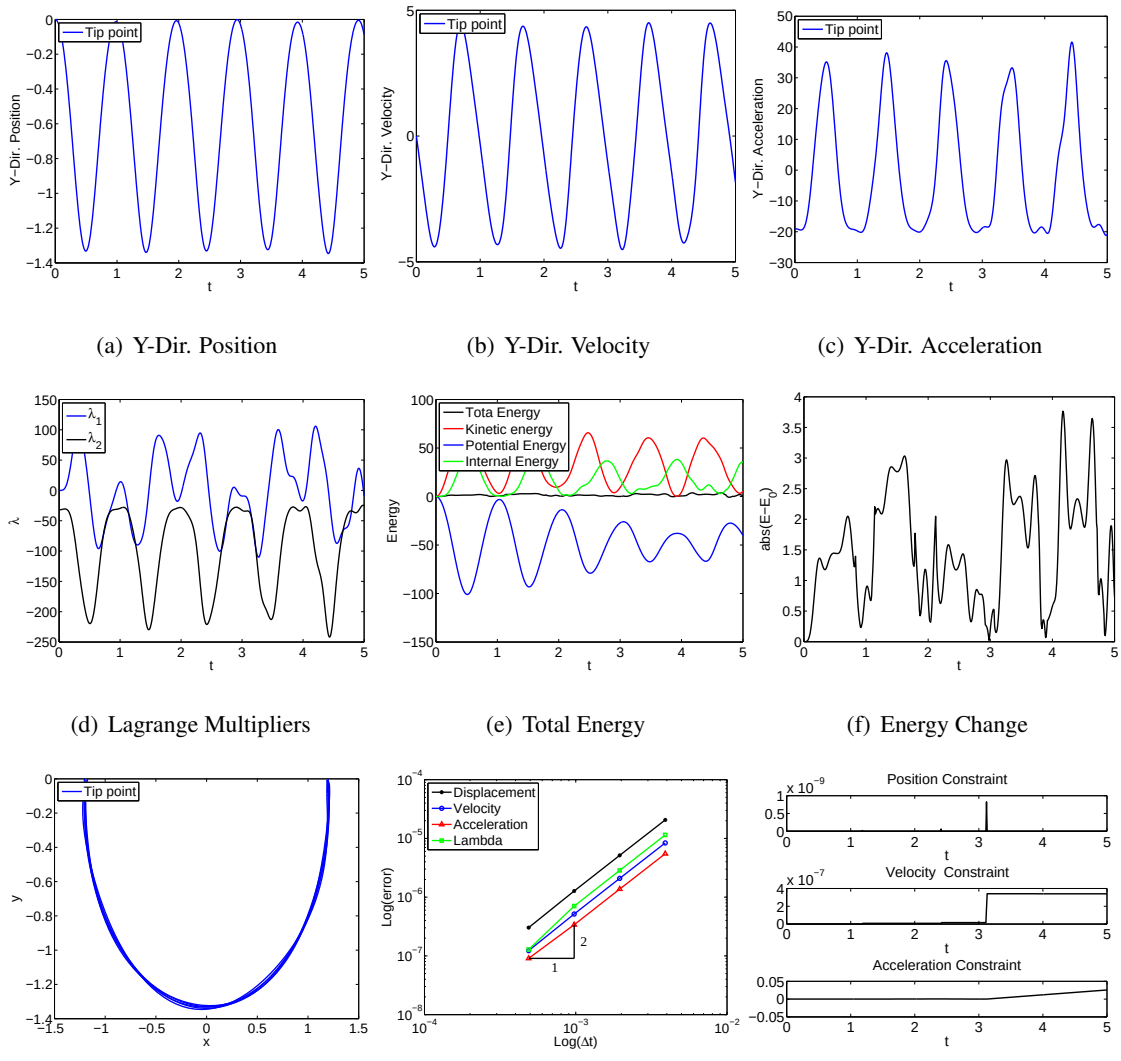
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.106: Single pendulum with EB element in FRF: U0V0(1,1,1) - Index 1.

Timoshenko Beam

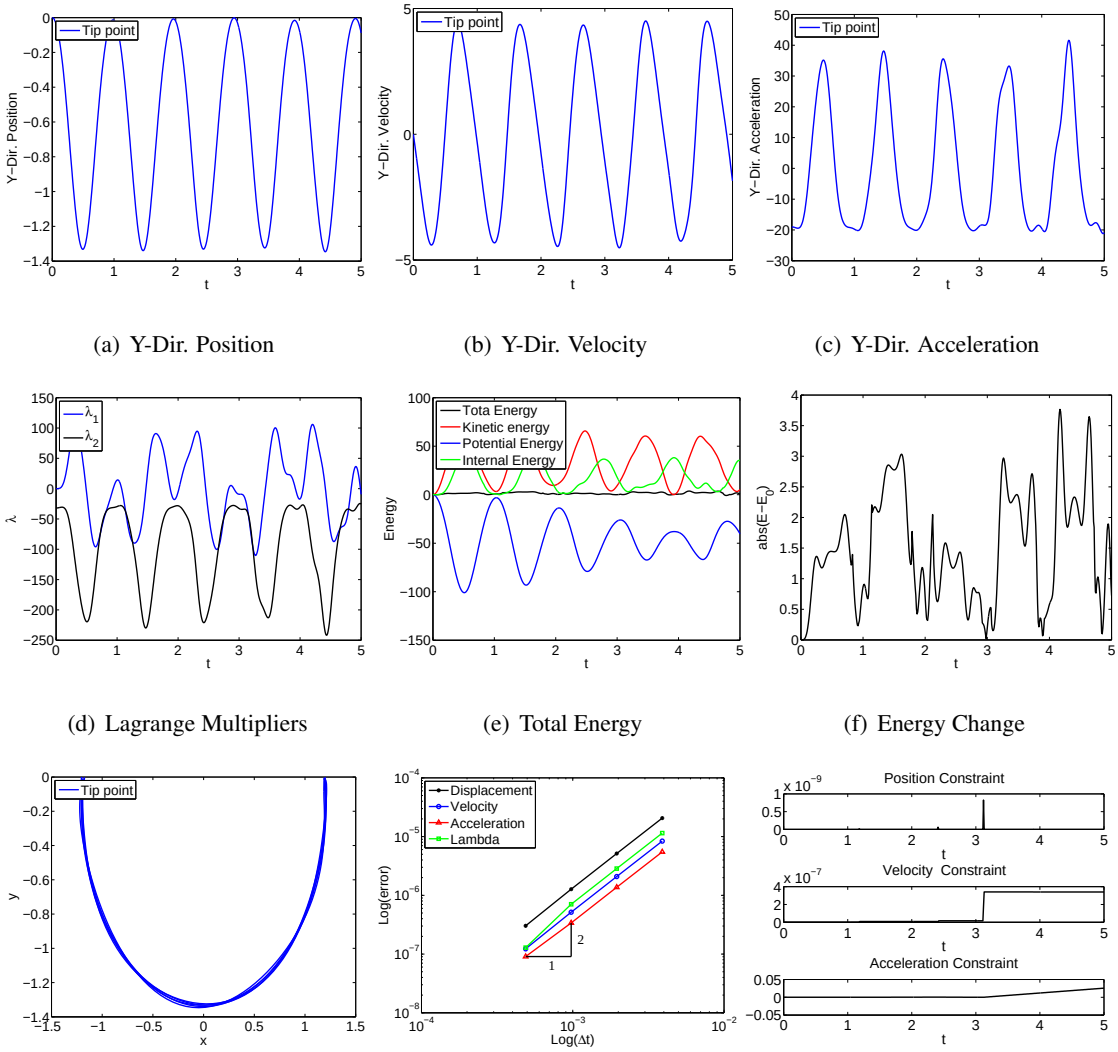


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.107: Single pendulum with TB element in IRF: U0(1,1,0) - Index 3.

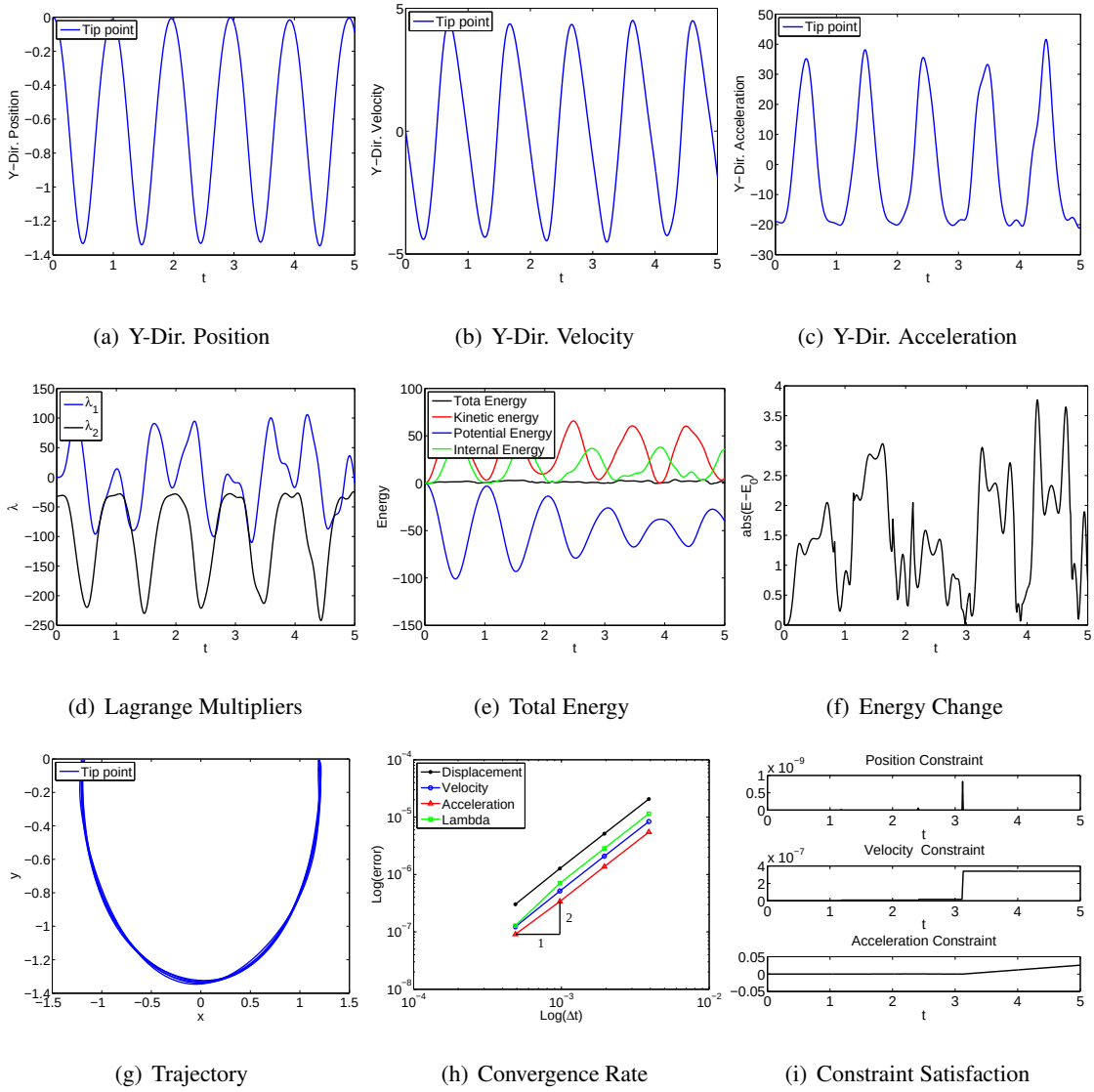


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.108: Single pendulum with TB element in IRF: U0(1,1,0) - Index 2.

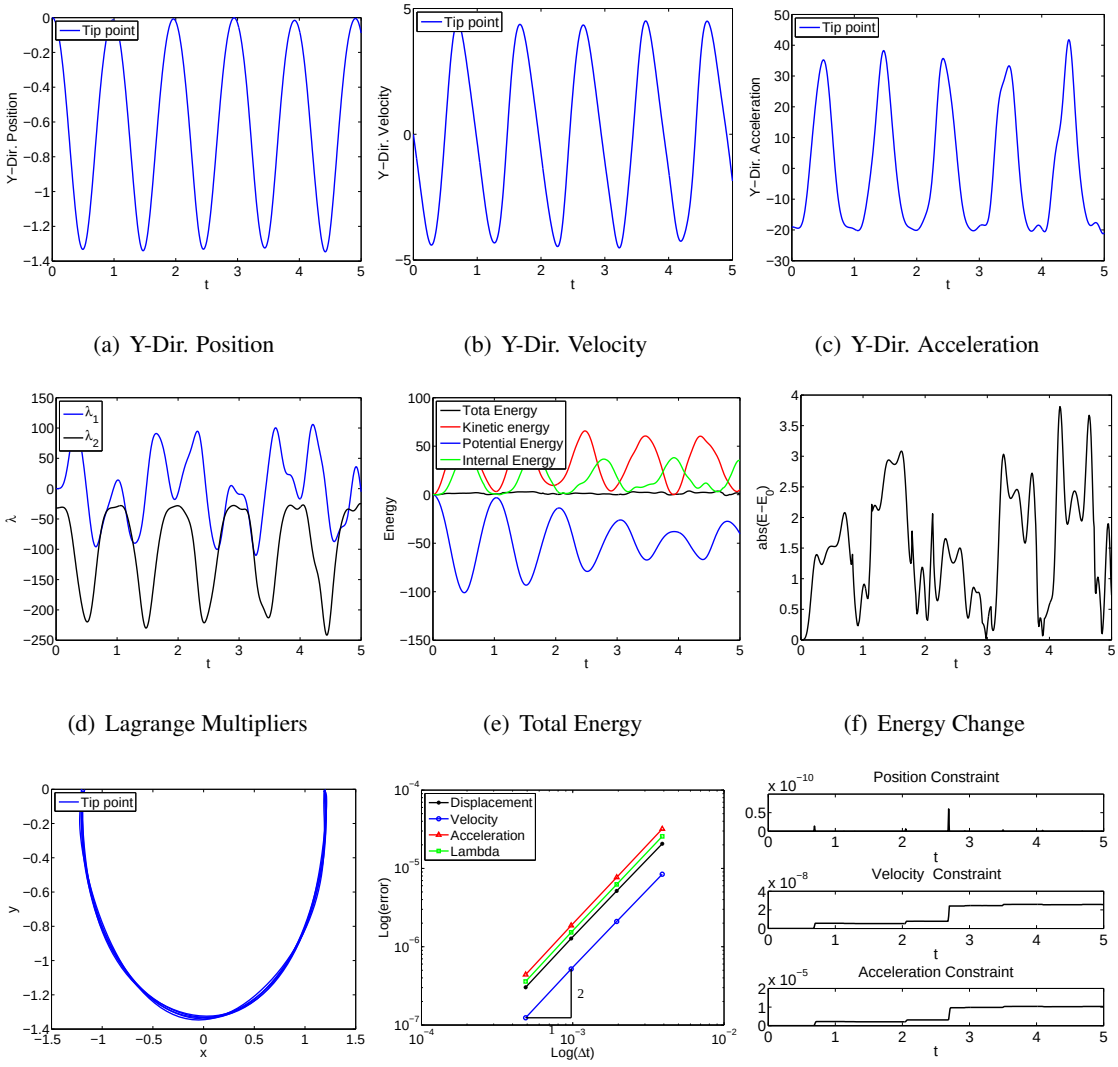


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.109: Single pendulum with TB element in IRF: U0(1,1,0) - Index 1.

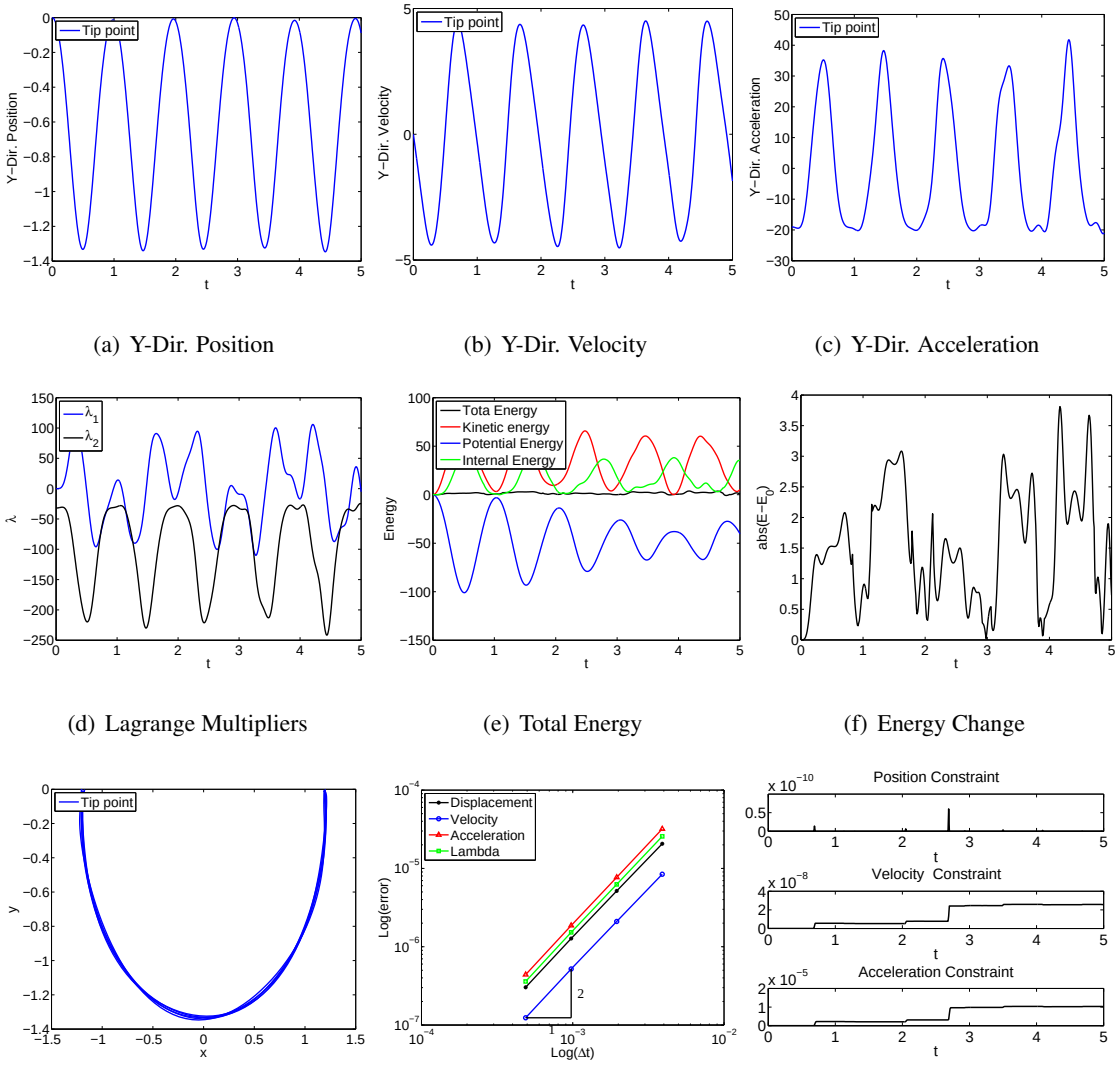


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.110: Single pendulum with TB element in IRF: V0(1,1,0) - Index 3.

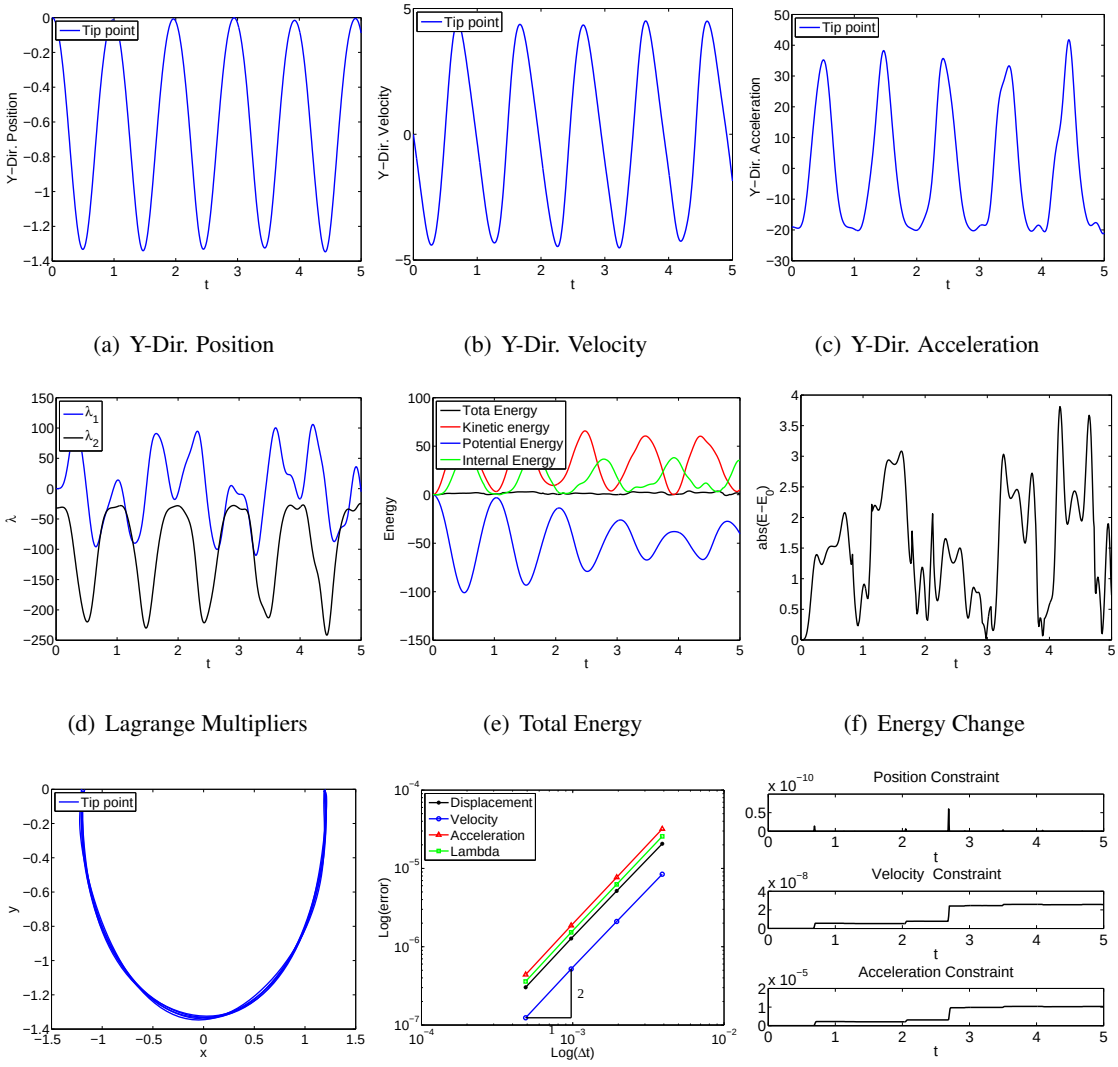


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.111: Single pendulum with TB element in IRF: V0(1,1,0) - Index 2.

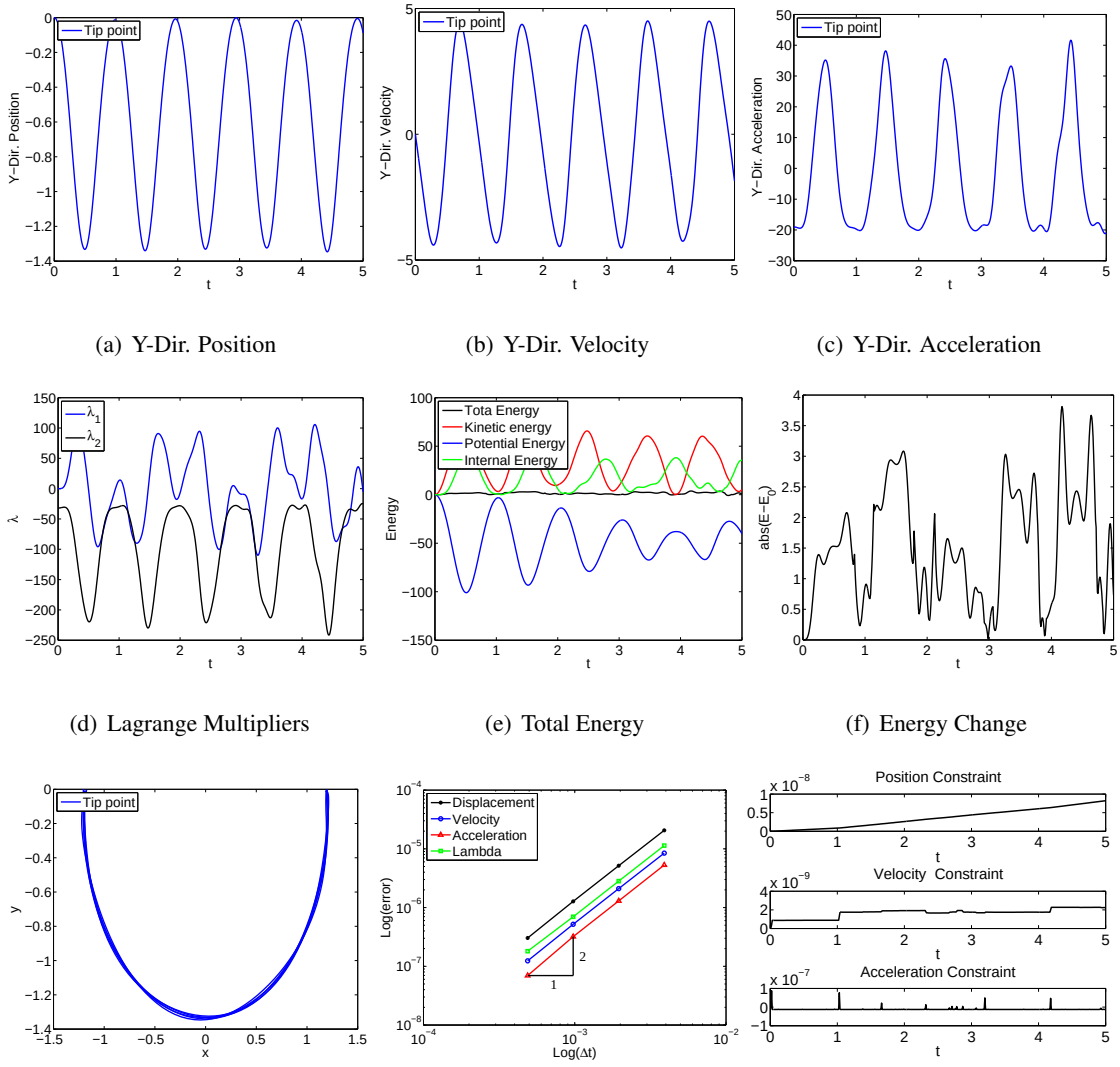


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.112: Single pendulum with TB element in IRF: V0(1,1,0) - Index 1.

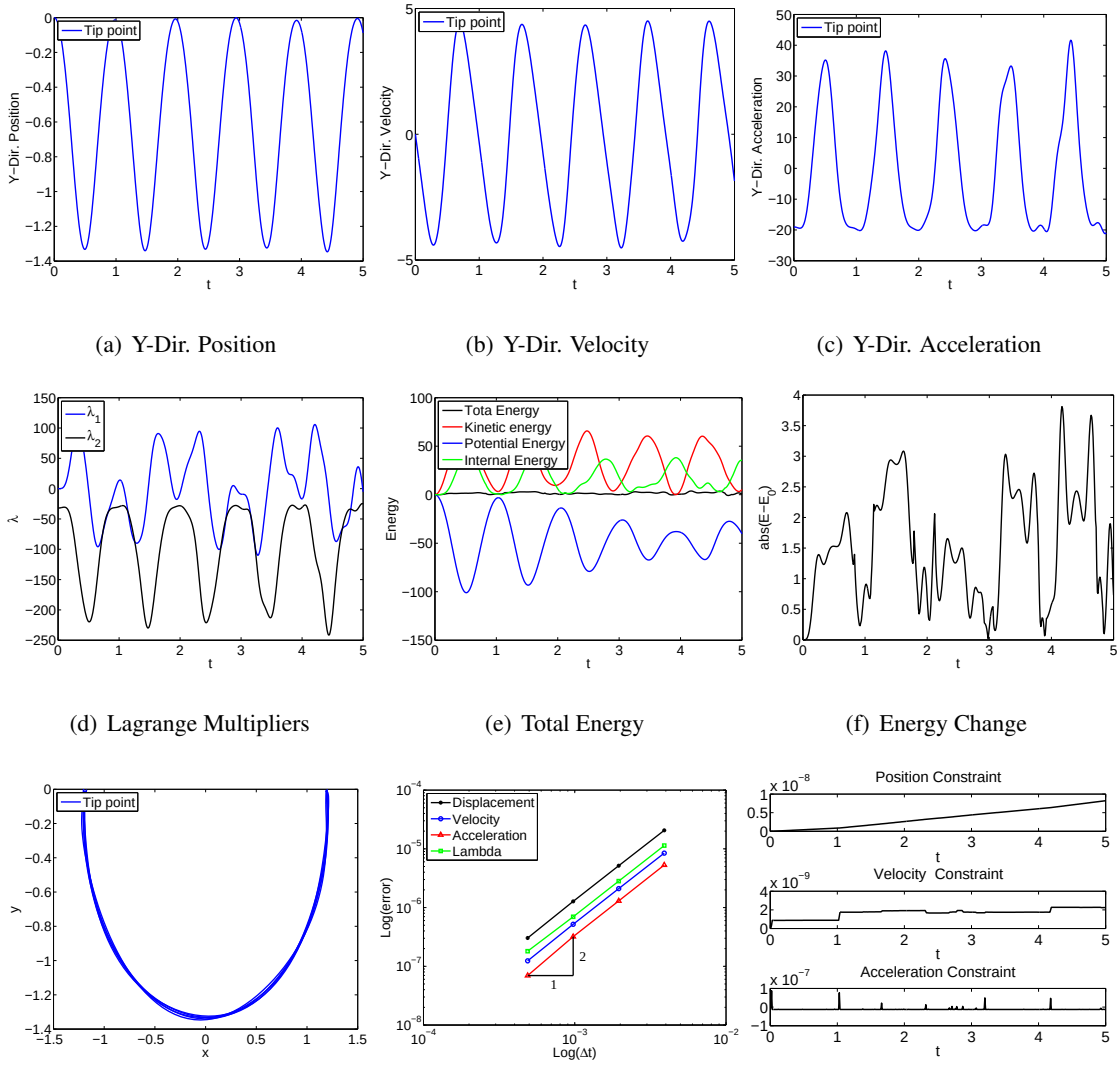


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.113: Single pendulum with TB element in IRF: U0V0(1,1,1) - Index 3.

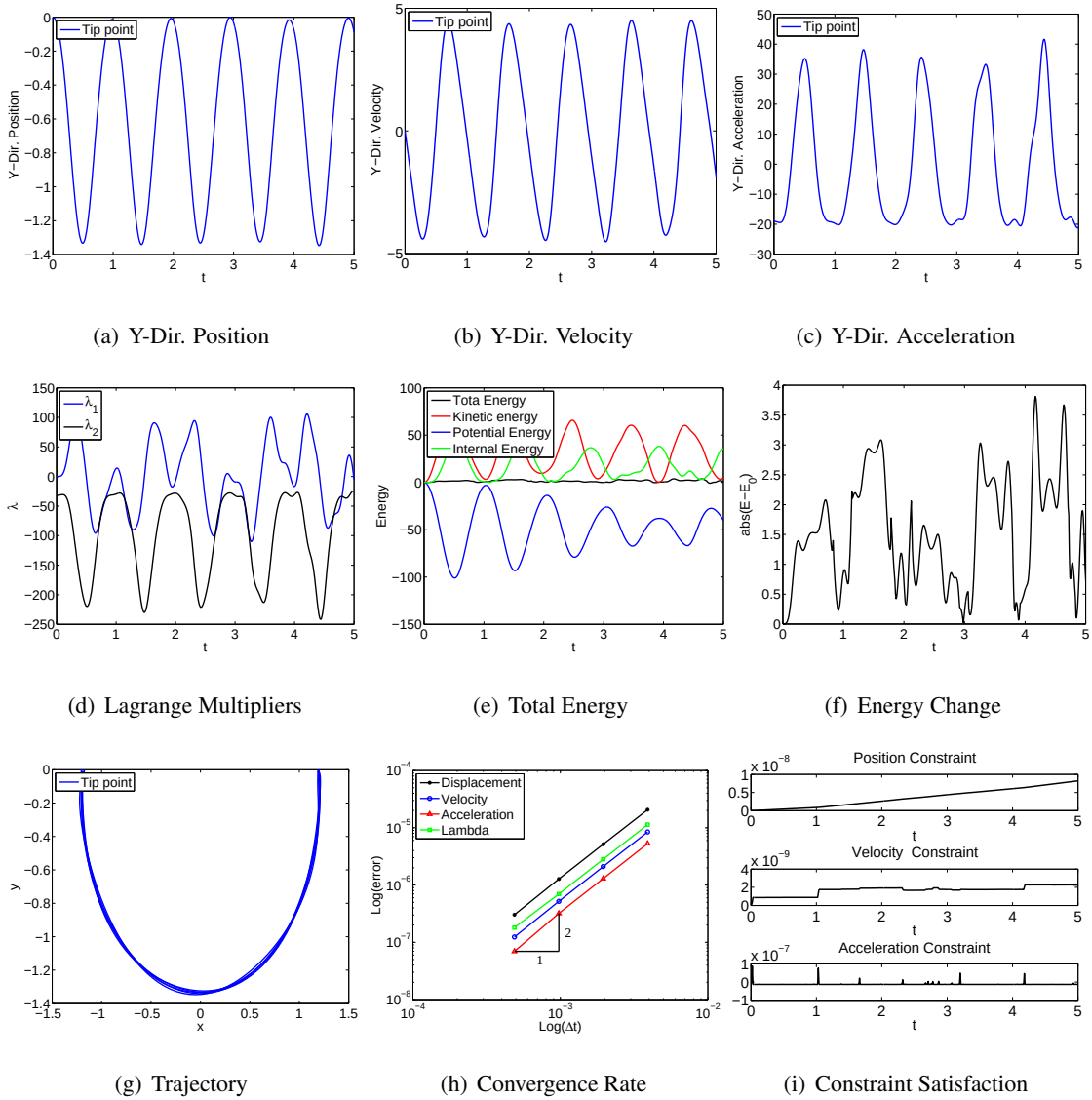


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.114: Single pendulum with TB element in IRF: U0V0(1,1,1) - Index 2.

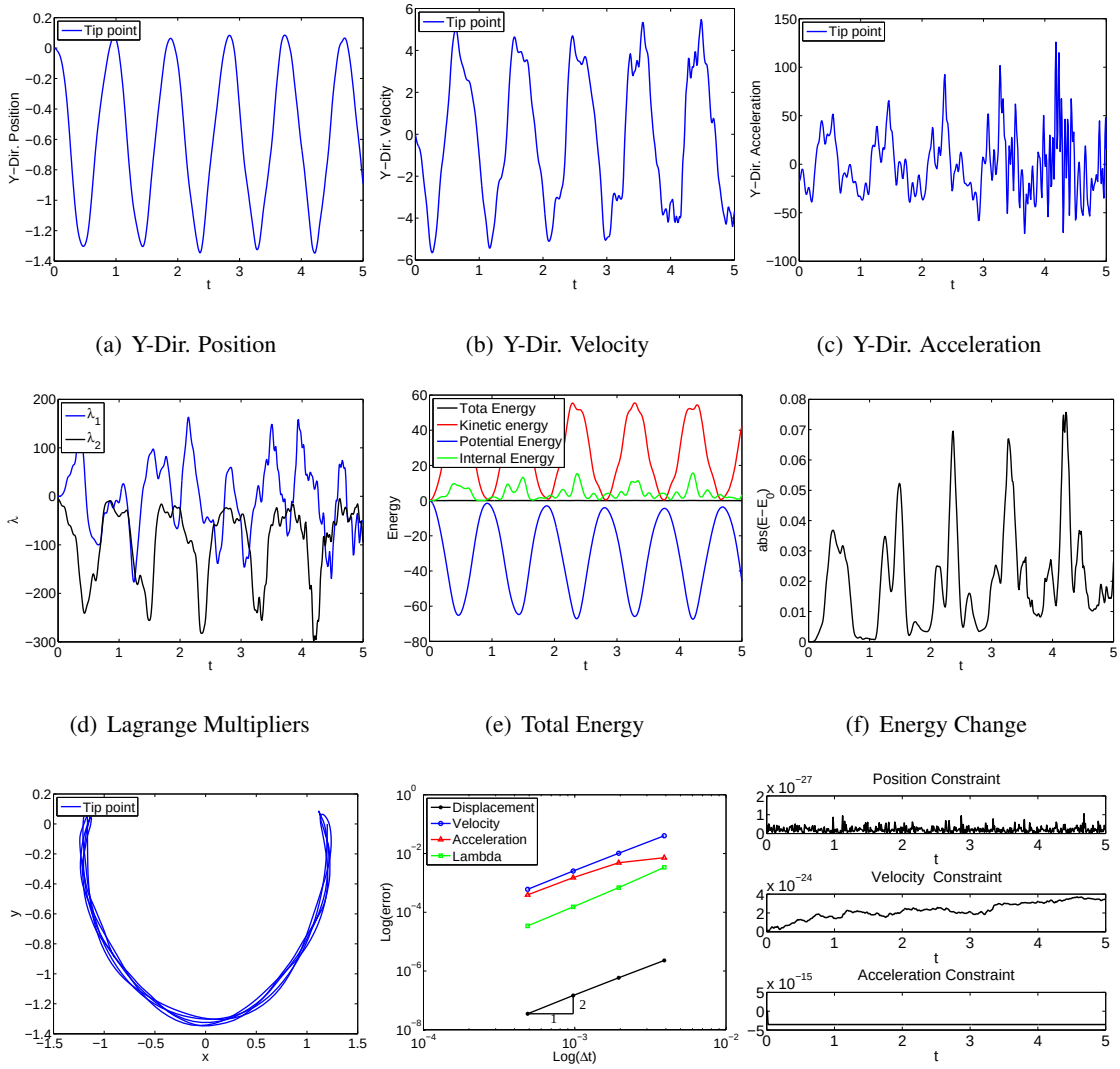


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.115: Single pendulum with TB element in IRF: U0V0(1,1,1) - Index 1.

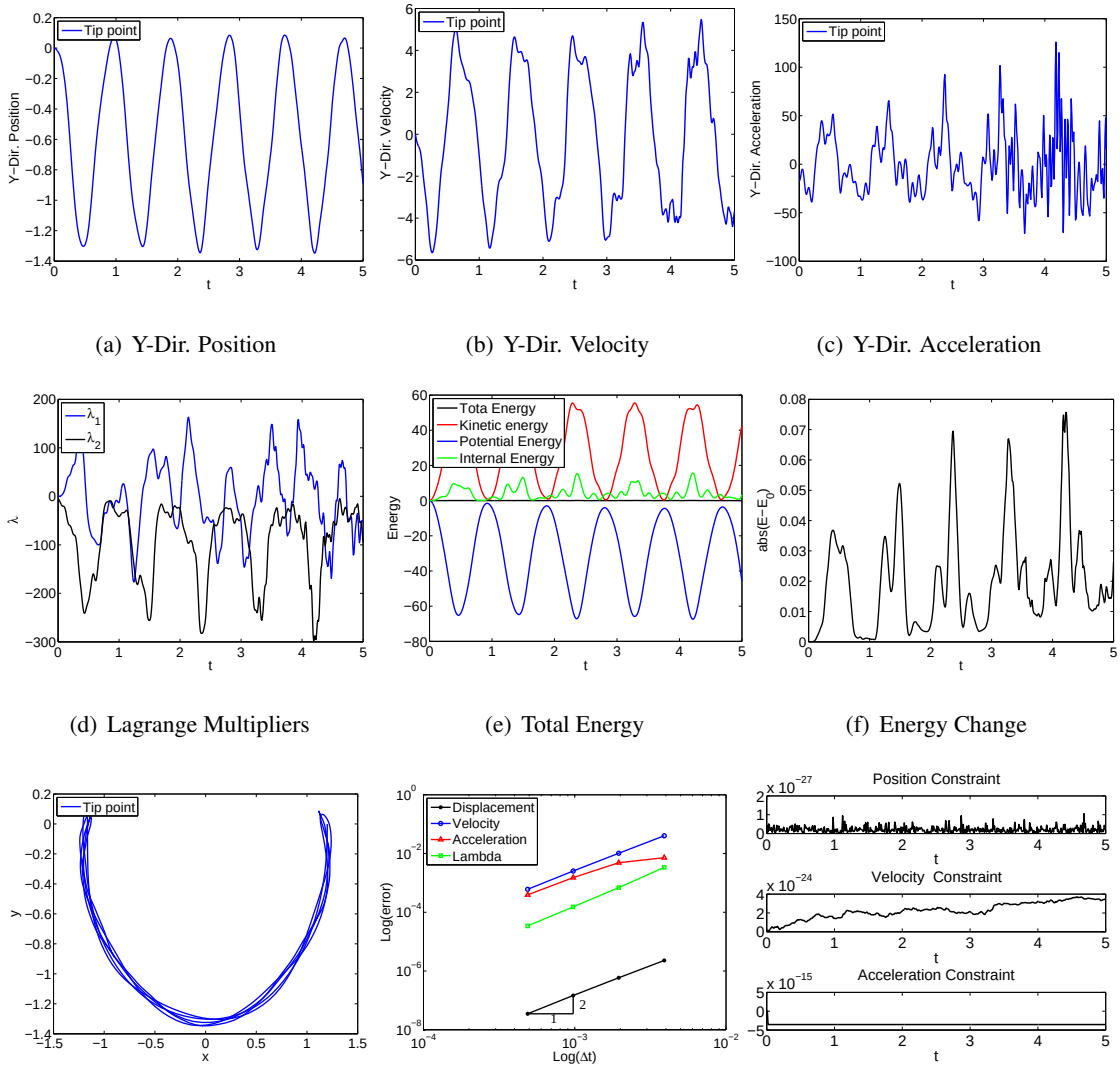


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.116: Single pendulum with TB element in ANCF-S: U0(1,1,0) - Index 3.

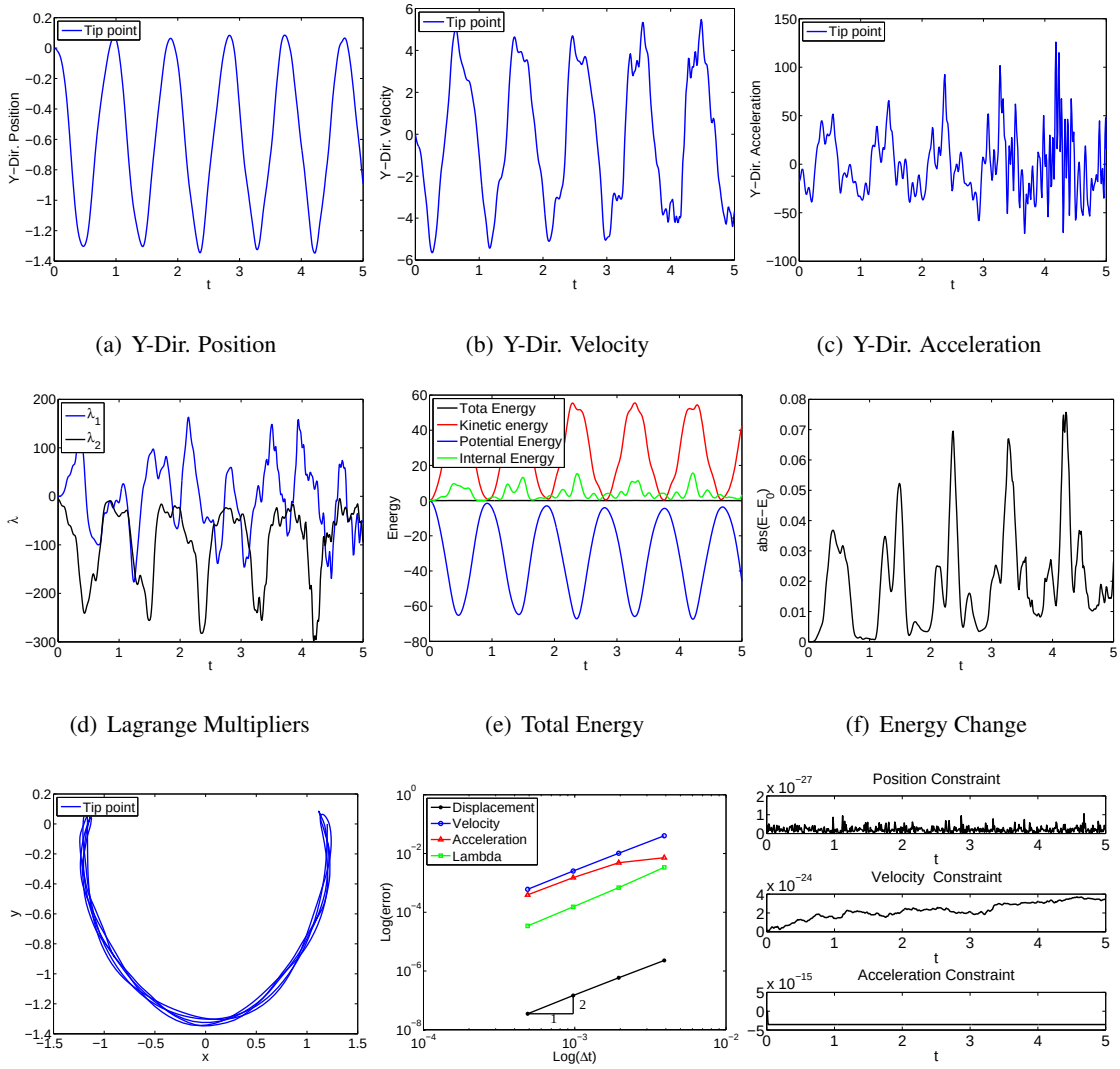


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.117: Single pendulum with TB element in ANCF-S: U0(1,1,0) - Index 2.

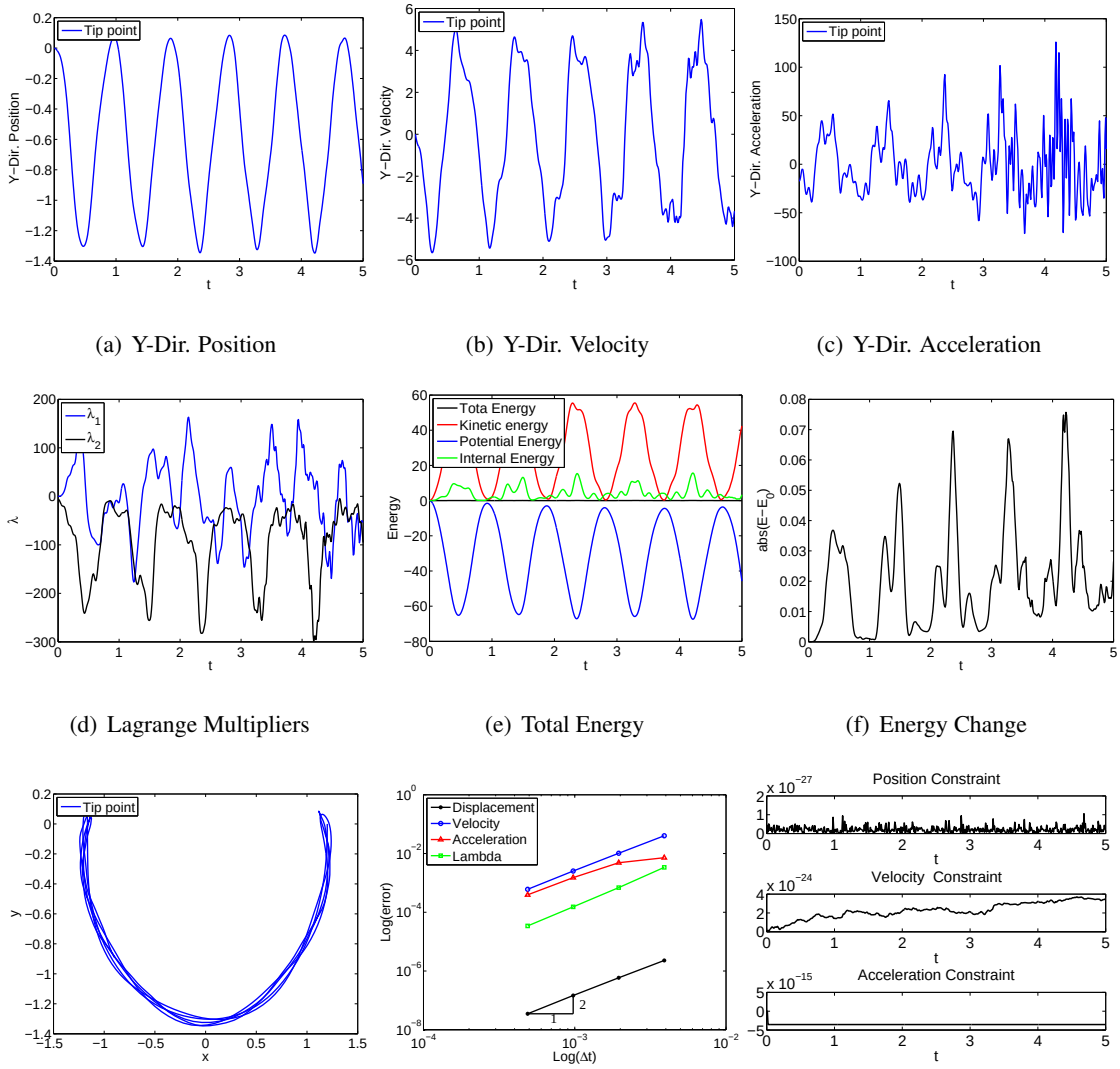


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.118: Single pendulum with TB element in ANCF-S: U0(1,1,0) - Index 1.

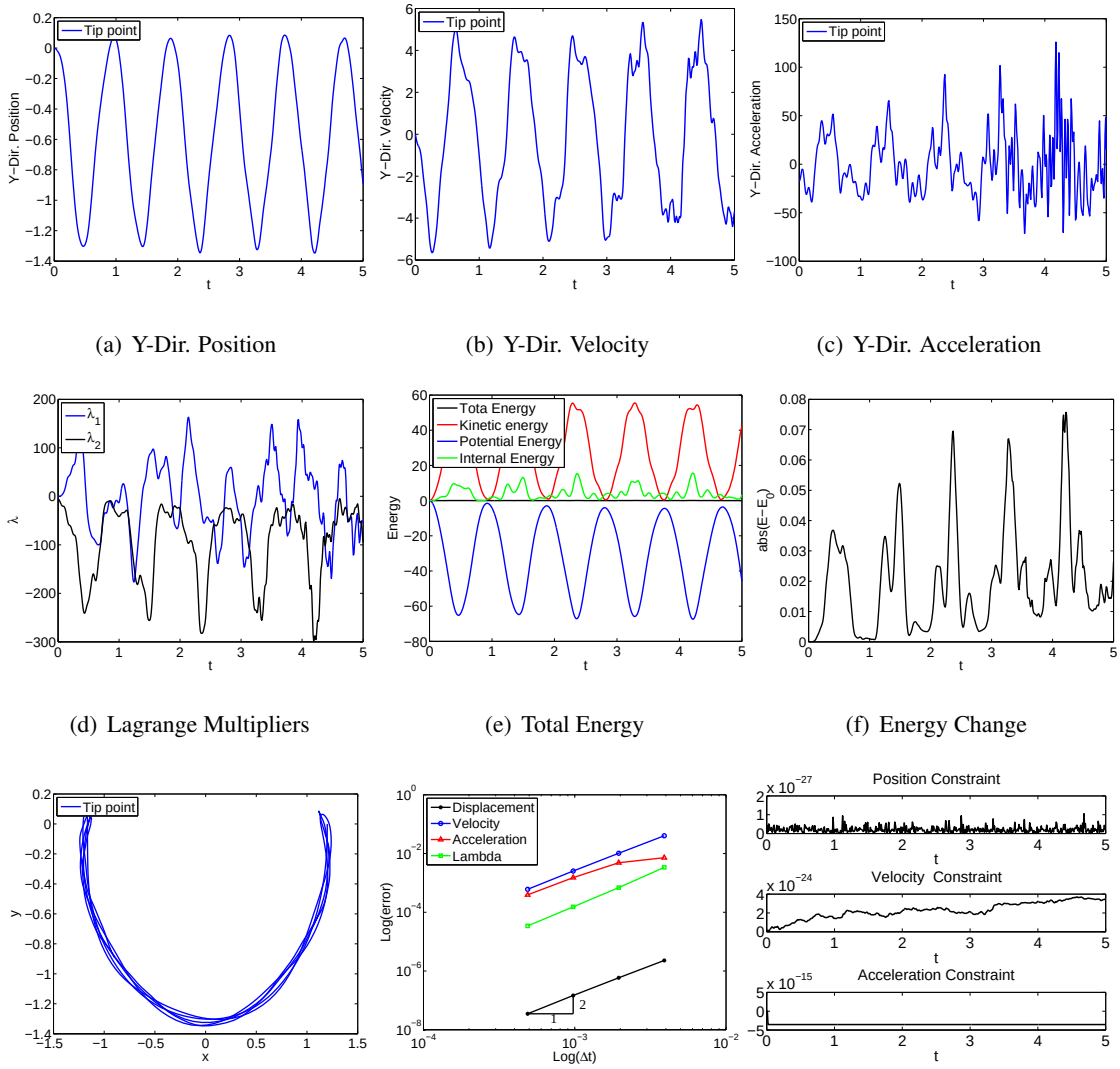


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.119: Single pendulum with TB element in ANCF-S: V0(1,1,0) - Index 3.

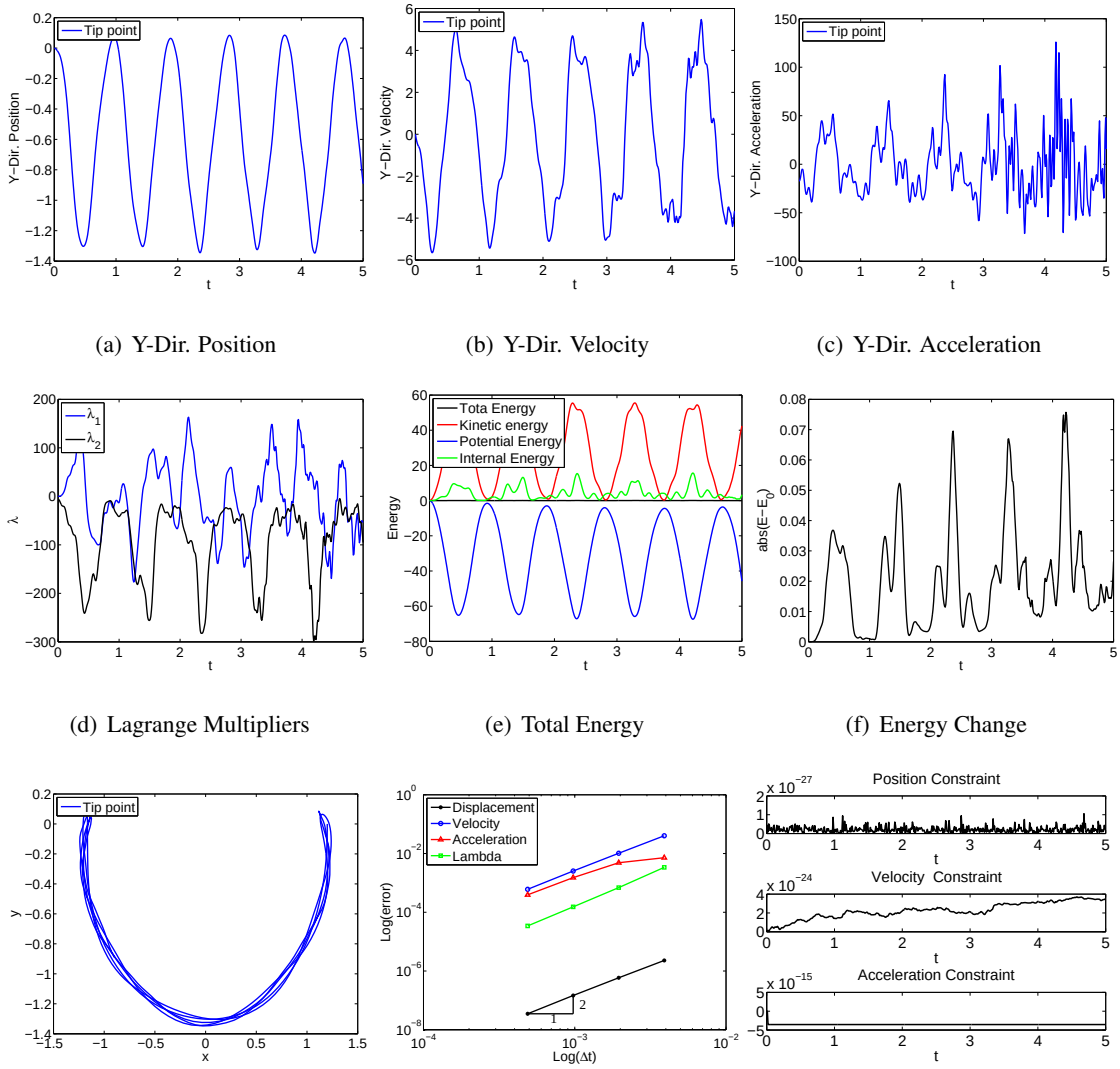


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.120: Single pendulum with TB element in ANCF-S: V0(1,1,0) - Index 2.

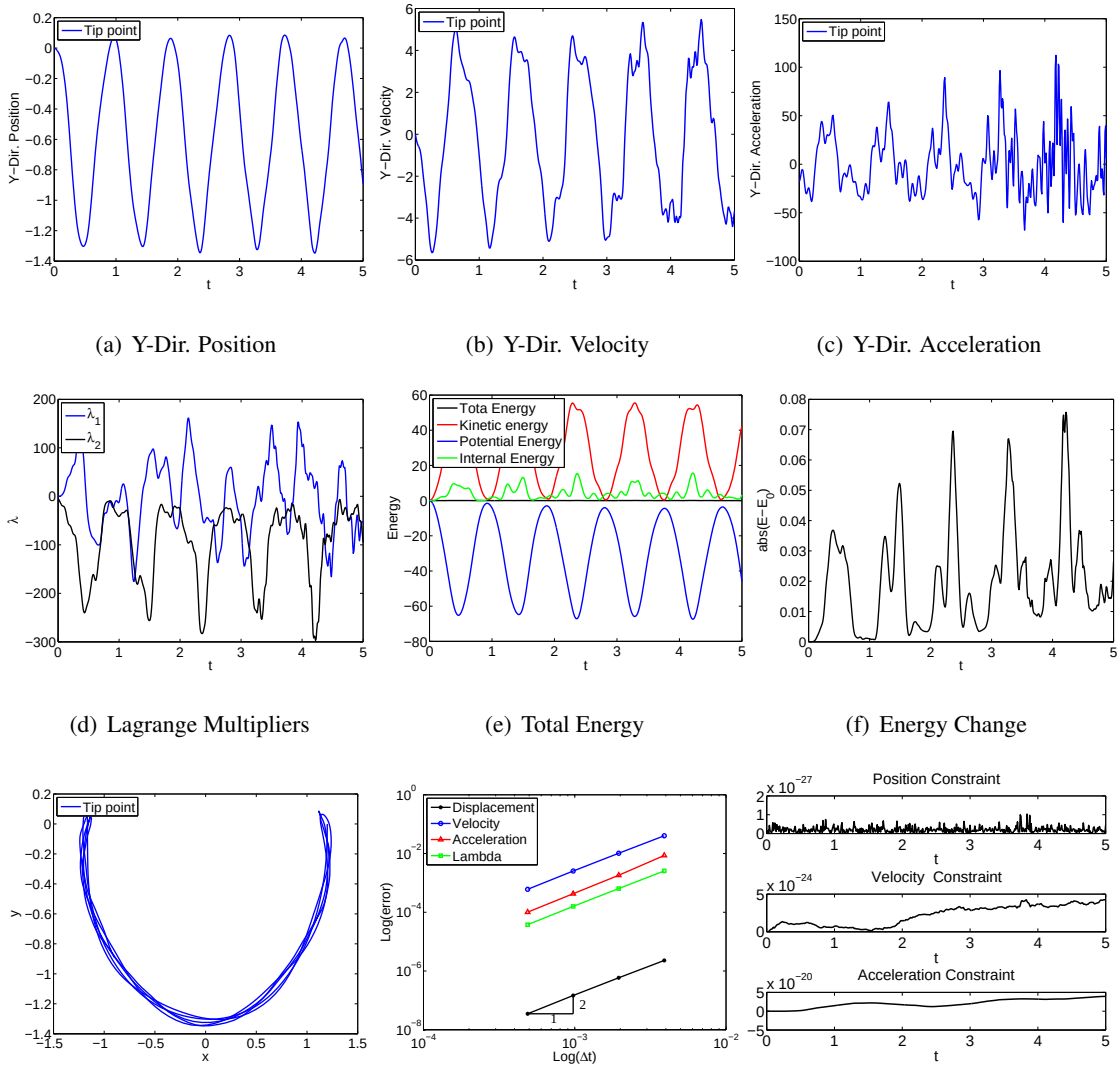


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.121: Single pendulum with TB element in ANCF-S: V0(1,1,0) - Index 1.

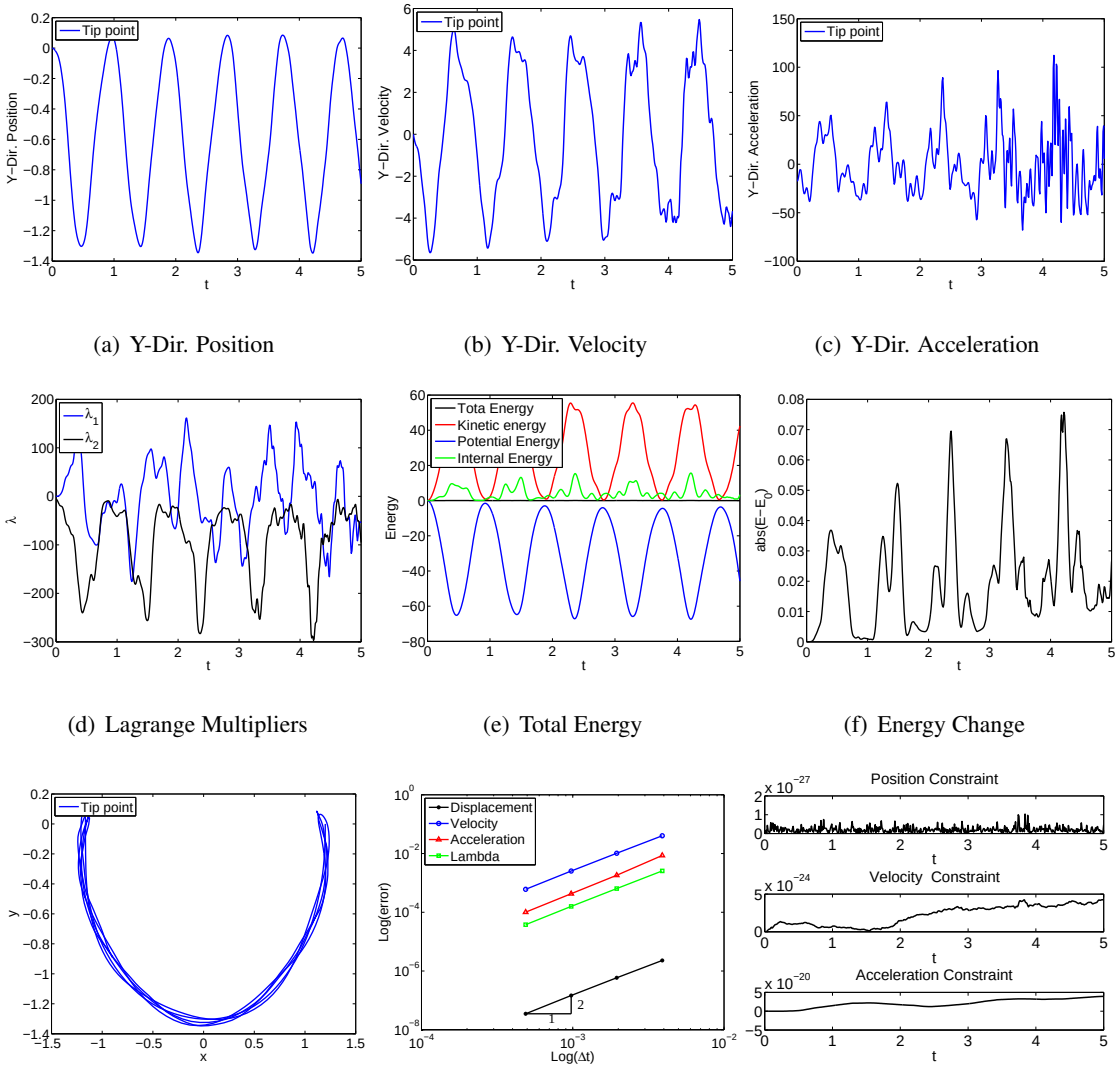


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.122: Single pendulum with TB element in ANCF-S: U0V0(1,1,1) - Index 3.

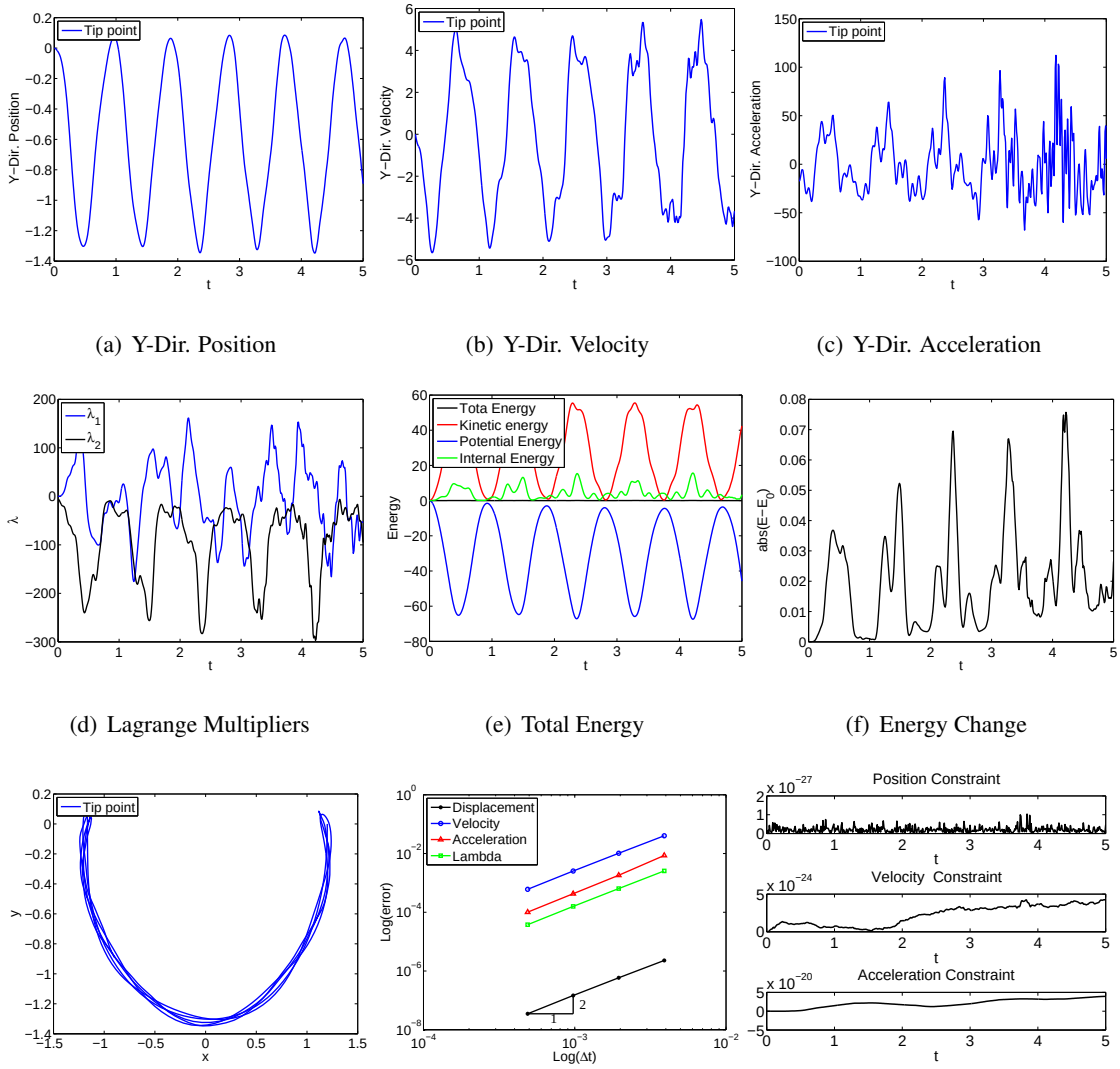


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.123: Single pendulum with TB element in ANCF-S: U0V0(1,1,1) - Index 2.

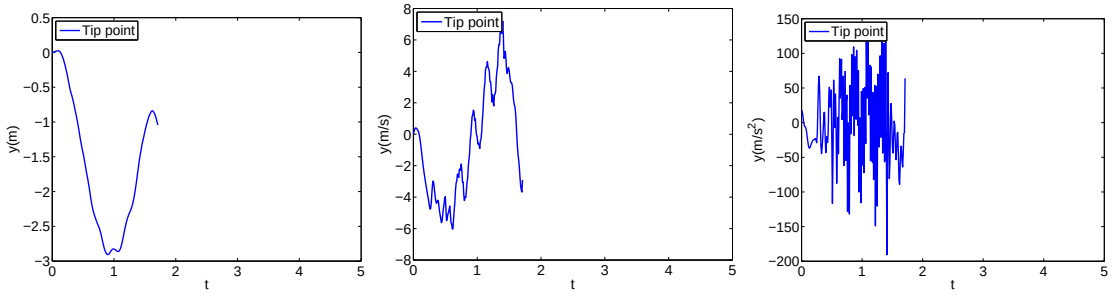


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

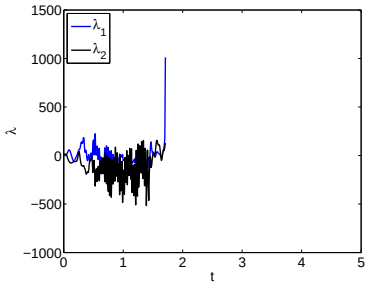
Figure 4.124: Single pendulum with TB element in ANCF-S: U0V0(1,1,1) - Index 1.



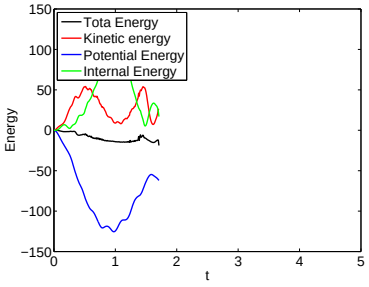
(a) Y-Dir. Position

(b) Y-Dir. Velocity

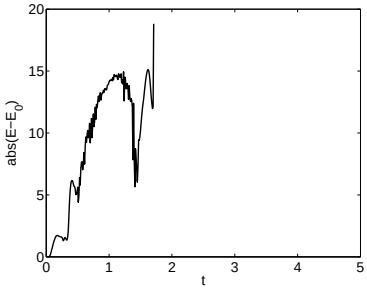
(c) Y-Dir. Acceleration



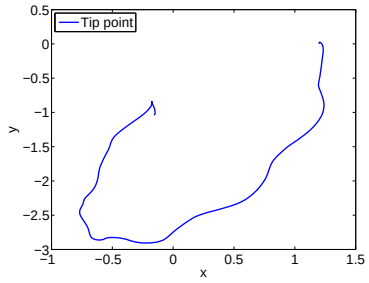
(d) Lagrange Multipliers



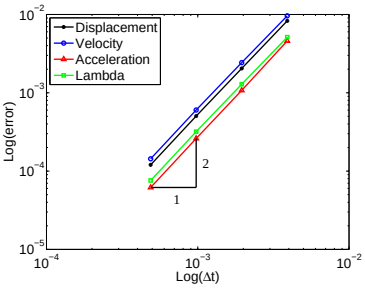
(e) Total Energy



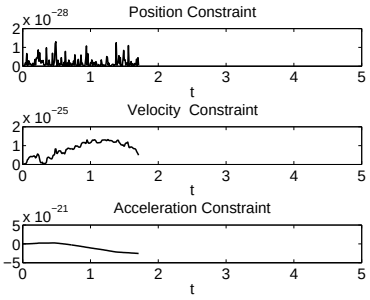
(f) Energy Change



(g) Trajectory

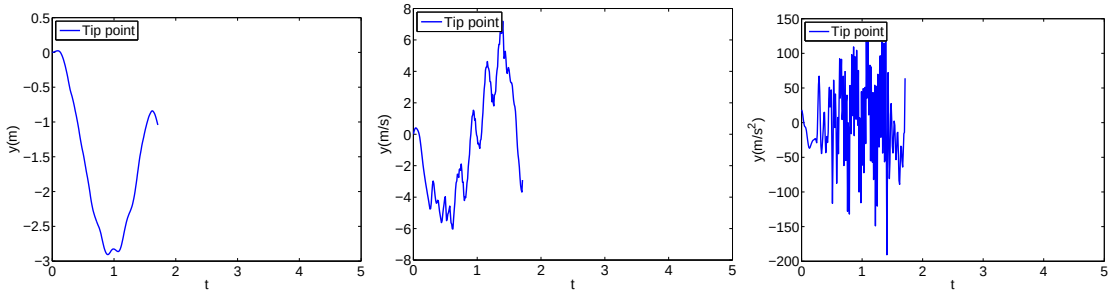


(h) Convergence Rate



(i) Constraint Satisfaction

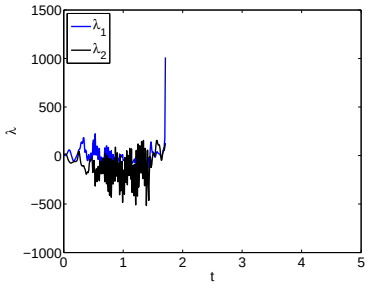
Figure 4.125: Single pendulum with TB element in FRF: U0(1,1,0) - Index 3.



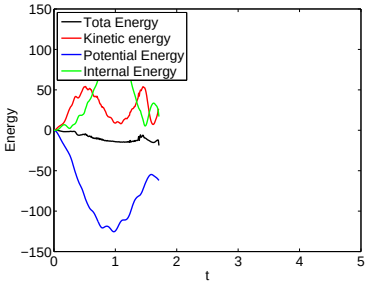
(a) Y-Dir. Position

(b) Y-Dir. Velocity

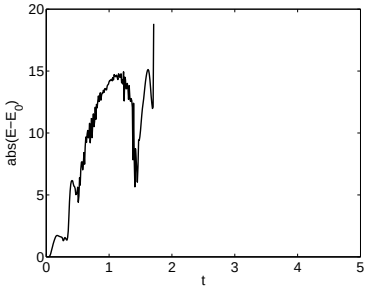
(c) Y-Dir. Acceleration



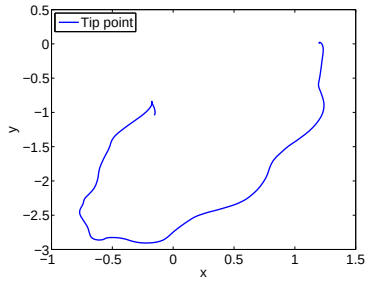
(d) Lagrange Multipliers



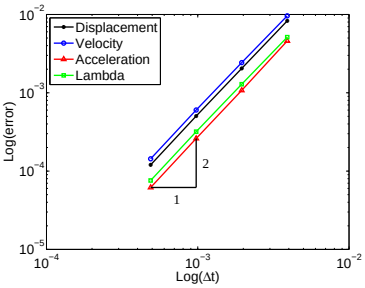
(e) Total Energy



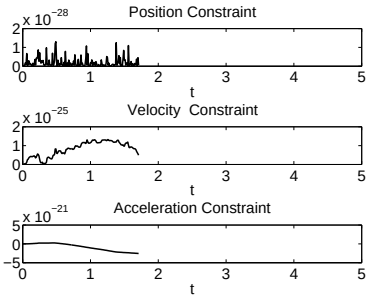
(f) Energy Change



(g) Trajectory

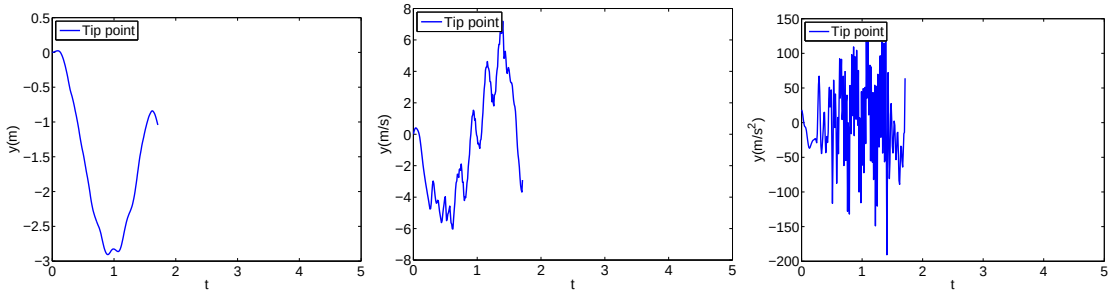


(h) Convergence Rate



(i) Constraint Satisfaction

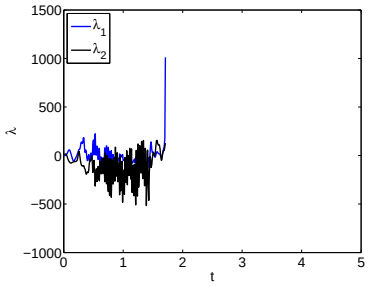
Figure 4.126: Single pendulum with TB element in FRF: U0(1,1,0) - Index 2.



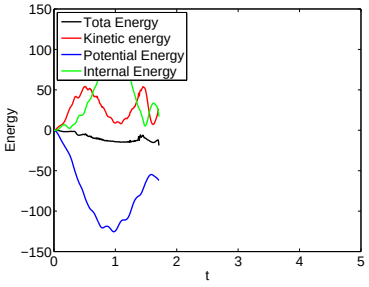
(a) Y-Dir. Position

(b) Y-Dir. Velocity

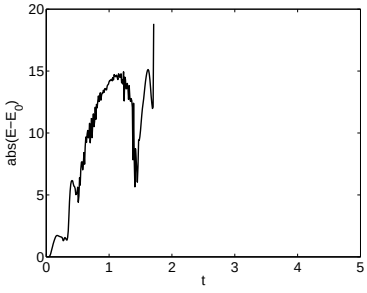
(c) Y-Dir. Acceleration



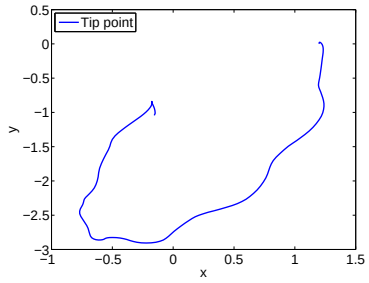
(d) Lagrange Multipliers



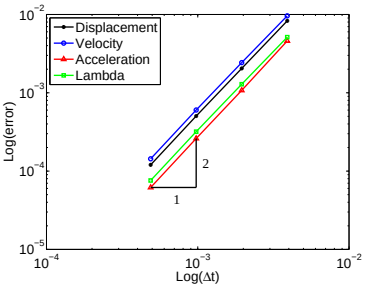
(e) Total Energy



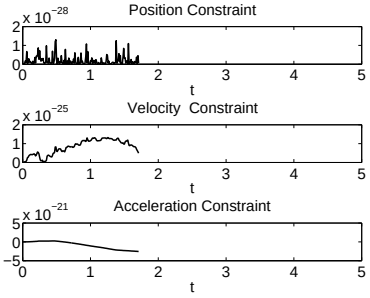
(f) Energy Change



(g) Trajectory

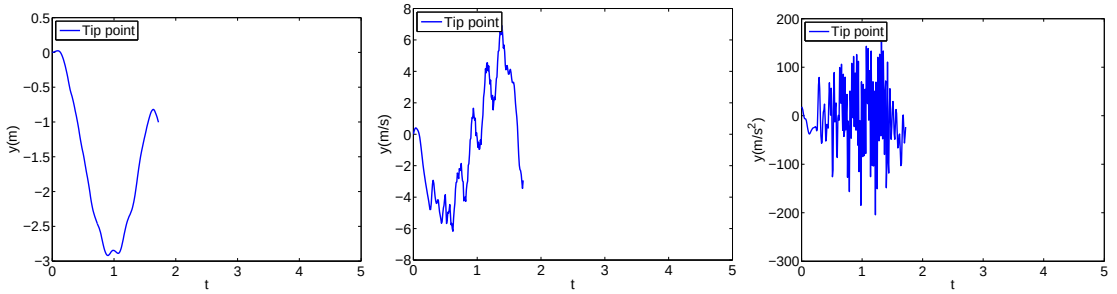


(h) Convergence Rate



(i) Constraint Satisfaction

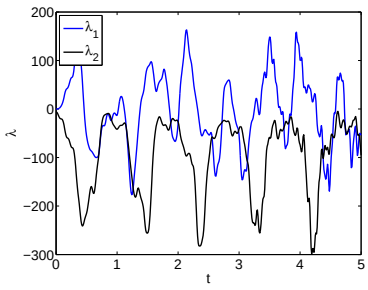
Figure 4.127: Single pendulum with TB element in FRF: U0(1,1,0) - Index 1.



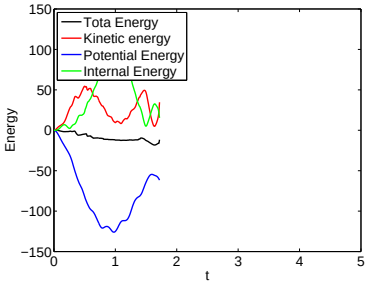
(a) Y-Dir. Position

(b) Y-Dir. Velocity

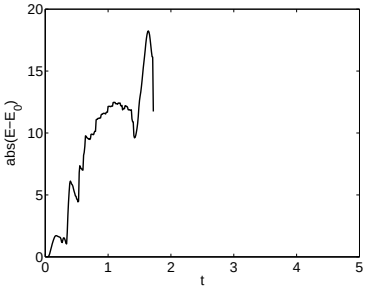
(c) Y-Dir. Acceleration



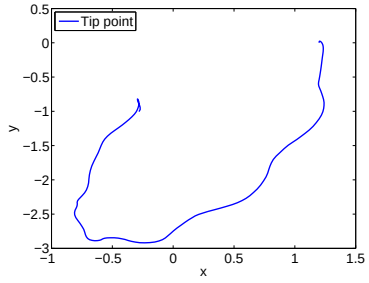
(d) Lagrange Multipliers



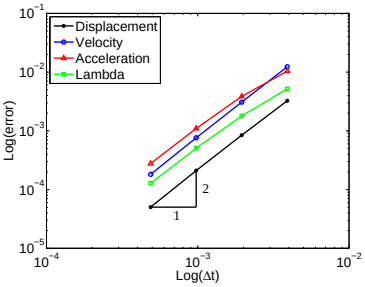
(e) Total Energy



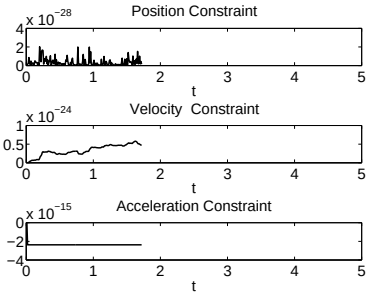
(f) Energy Change



(g) Trajectory

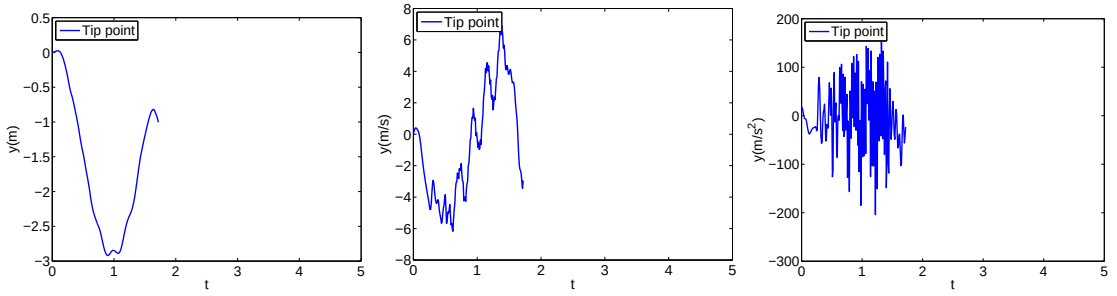


(h) Convergence Rate



(i) Constraint Satisfaction

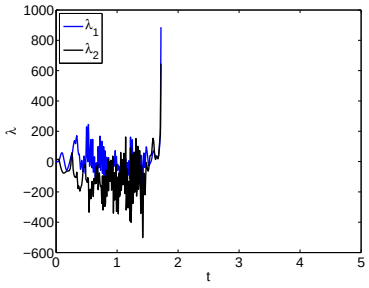
Figure 4.128: Single pendulum with TB element in FRF: V0(1,1,0) - Index 3.



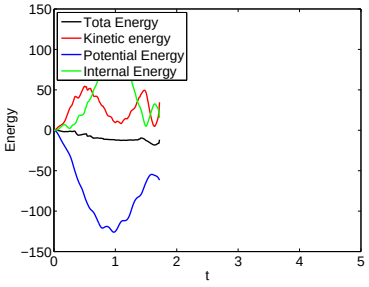
(a) Y-Dir. Position

(b) Y-Dir. Velocity

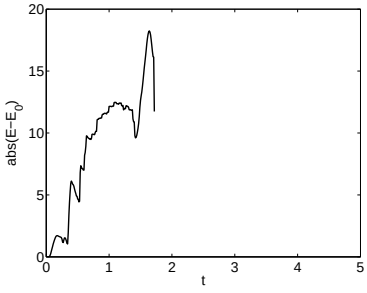
(c) Y-Dir. Acceleration



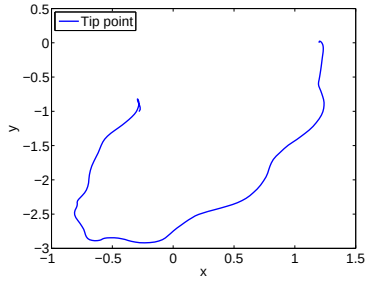
(d) Lagrange Multipliers



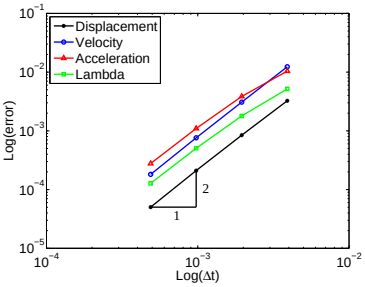
(e) Total Energy



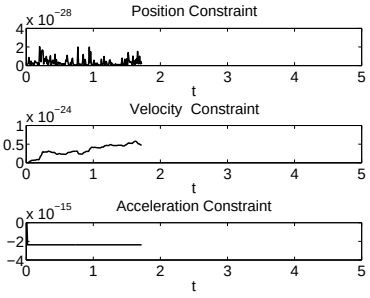
(f) Energy Change



(g) Trajectory

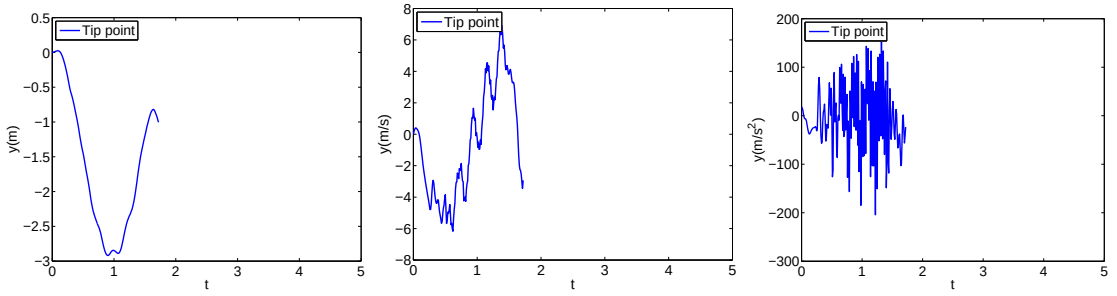


(h) Convergence Rate



(i) Constraint Satisfaction

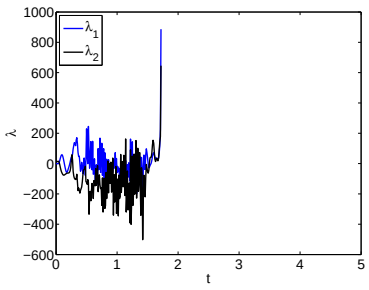
Figure 4.129: Single pendulum with TB element in FRF: V0(1,1,0) - Index 2.



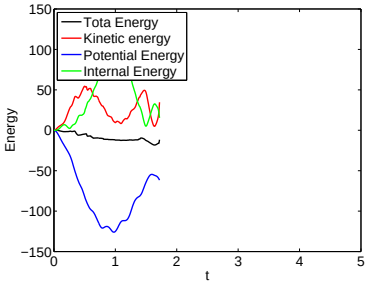
(a) Y-Dir. Position

(b) Y-Dir. Velocity

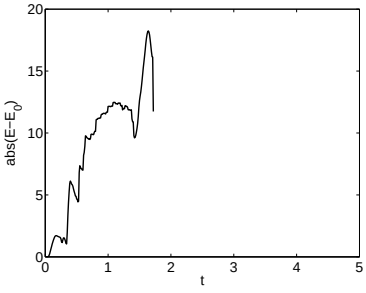
(c) Y-Dir. Acceleration



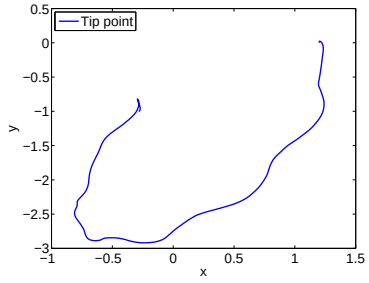
(d) Lagrange Multipliers



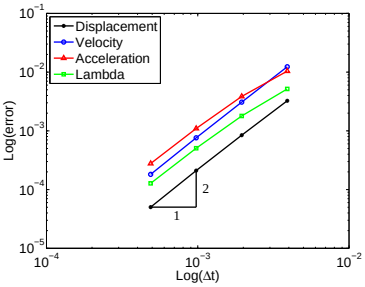
(e) Total Energy



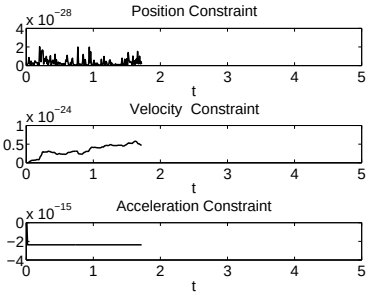
(f) Energy Change



(g) Trajectory

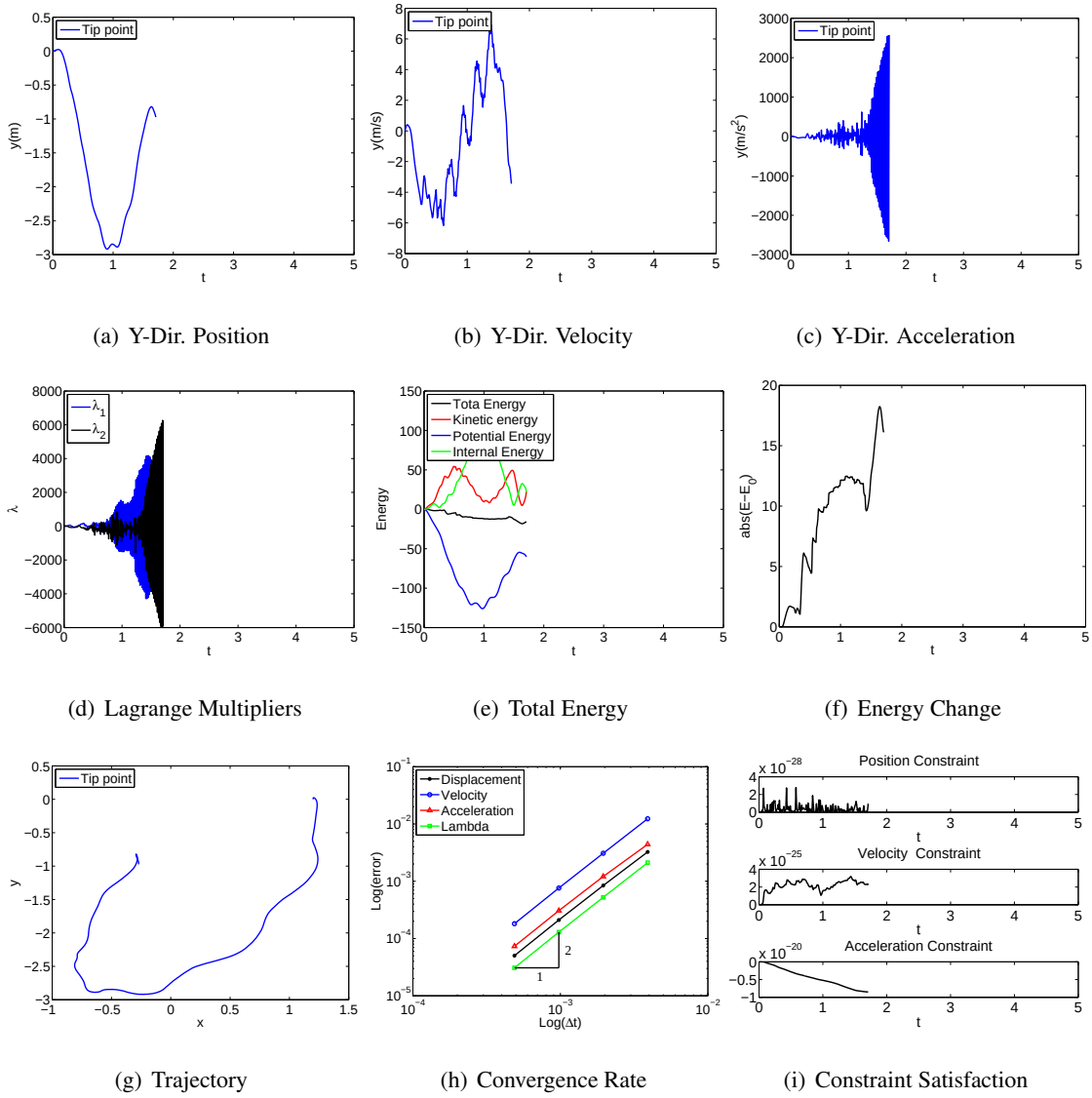


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.130: Single pendulum with TB element in FRF: V0(1,1,0) - Index 1.

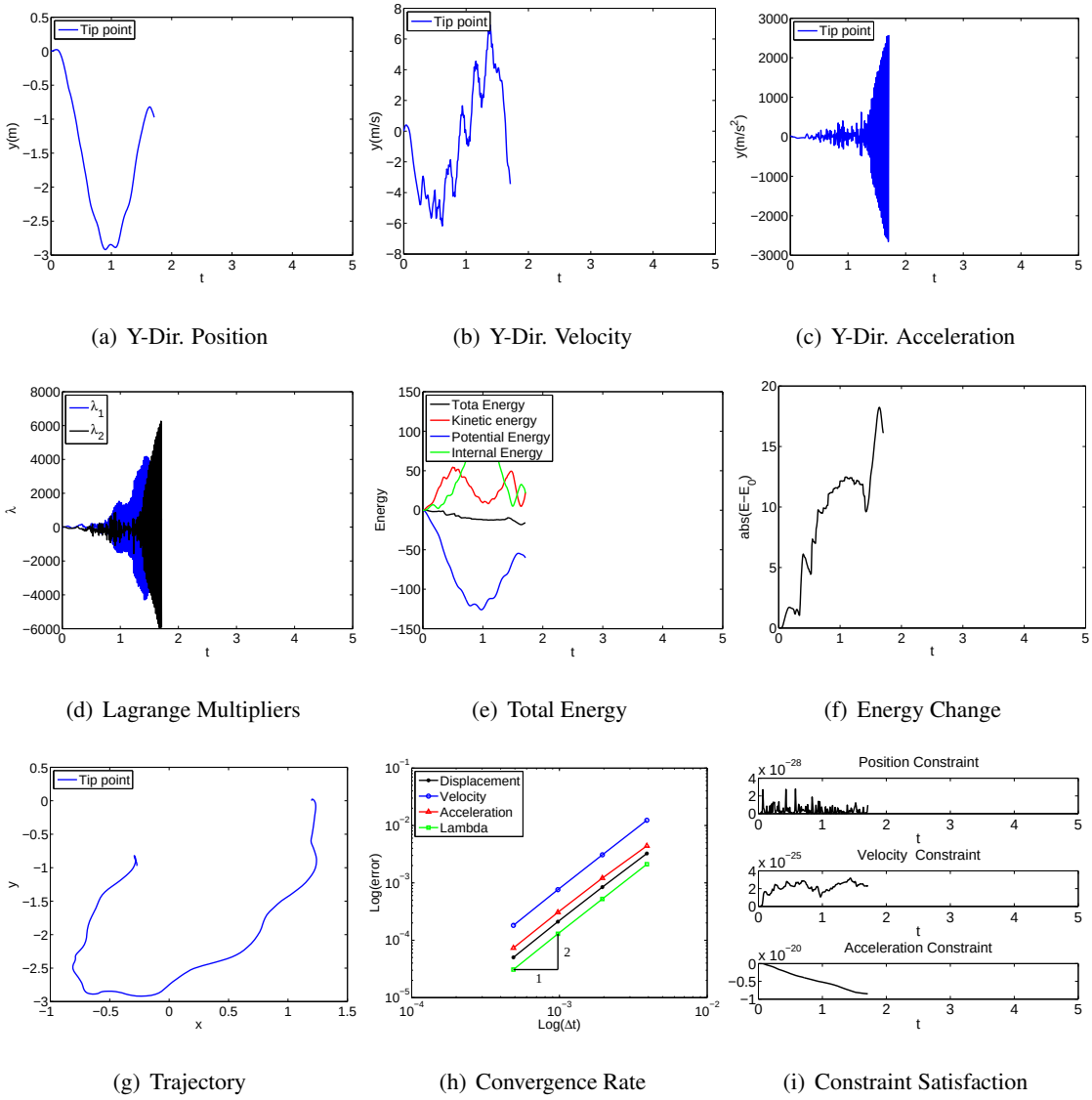


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.131: Single pendulum with TB element in FRF: U0V0(1,1,1) - Index 3.

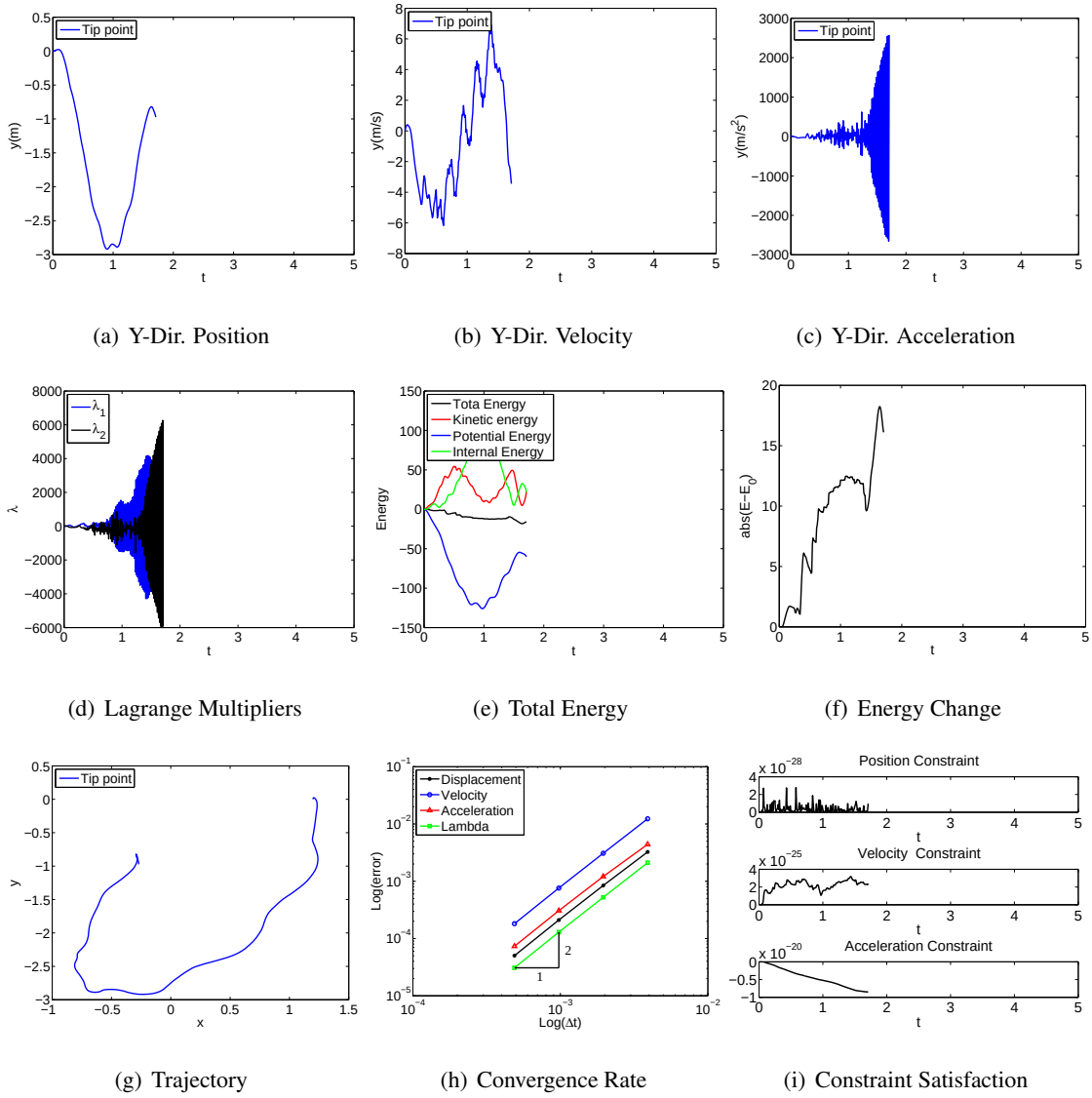


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.132: Single pendulum with TB element in FRF: U0V0(1,1,1) - Index 2.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.133: Single pendulum with TB element in FRF: U0V0(1,1,1) - Index 1.

From the simulation results for the simple pendulum, one can easily see that ANCF/IRF based results are identical in Index 3, Index 2, and Index 1. In fact, as long as the constraint equations are in linear form, the DAEs are identical to ODEs and this way we can eliminate the coupling equation. Hence, for the next example, this research only shows the result of ANCF and IRF in Index 3 since the constraint equation for these two formulation are linear. For FRF, the constraint is nonlinear and the result for Index 3, Index 2, and Index 1 is therefore different.

4.2.2 Double Pendulum

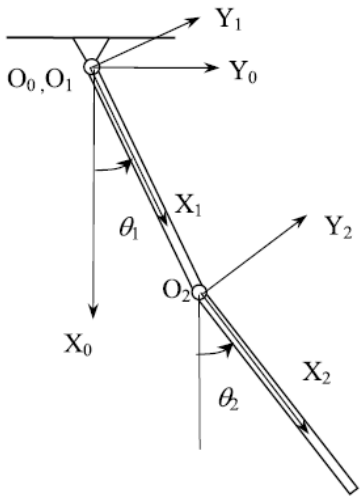


Figure 4.134: Double flexible pendulum problem from[63].

In the demonstration of the double pendulum as shown in Fig. 4.134, unlike the rigid bar example, the bar can be segmented into finite elements by IRF/FRF/ANCF-S as well as the constraint formulation. As a result, the coordinates for different formulations are different. Also, nonlinearity in each matrix and each element are also different. This problem is discussed in great detail in [63], with pertinent information repeated below.

In general the problem is of the form:

$$\begin{aligned} \mathbf{M}a + \mathbf{F}_{int} + \mathbf{C}^T(u)\lambda &= \mathbf{F} \\ \mathbf{\Phi}(u) &= 0 \end{aligned} \tag{4.50}$$

The problem consists of different degrees of freedom and depends on which element is employed and four Lagrange multipliers (see Appendix). Because there is only one planar hinge in this system, the constraint equation for all formulations is the same:

$$\mathbf{\Phi} = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^1_{2x} - r^2_{1x} \\ r^1_{2y} - r^2_{1y} \end{bmatrix} \tag{4.51}$$

where r^i_{jx} indicate global x-direction displacement of the jth node on the ith body. Here, r^1_{1x} and r^1_{1y} represent the displacement in global x and y direction at the node connecting to the hinge at the origin, and $r^1_{2x} = r^2_{1x}$ and $r^1_{2y} = r^2_{1y}$ indicate the hinge connect body 1 and body 2. Hence, the first derivative of geometry constraint yields the velocity constraint:

$$\dot{\mathbf{\Phi}} = \begin{bmatrix} \dot{r}_1 \\ \dot{r}_2 \\ \dot{r}^1_{2x} - \dot{r}^2_{1x} \\ \dot{r}^1_{2y} - \dot{r}^2_{1y} \end{bmatrix} \tag{4.52}$$

and the acceleration constraint is:

$$\ddot{\mathbf{\Phi}} = \begin{bmatrix} \ddot{r}_1 \\ \ddot{r}_2 \\ \ddot{r}^1_{2x} - \ddot{r}^2_{1x} \\ \ddot{r}^1_{2y} - \ddot{r}^2_{1y} \end{bmatrix} \tag{4.53}$$

the corresponding constraint matrix $\boldsymbol{C}$ for different formulations results in different results. Generally, using the inertial reference frame type approach, a simple joint (hinge) results in sparse matrix and some components are equal to one. For the floating frame approach it is nonlinear as shown in Appendix. In all simulations below the problem parameters are taken as:

- Length of the body: $L_1 = L_2 = 1.8$ m
- Cross-section area: $A_1 = A_2 = 2.5 \times 10^{-4}$ m^2
- Density: $\rho_1 = \rho_2 = 2767$ Kg/m^3
- Elastic Modulus: $E_1 = E_2 = 6.895 \times 10^{10}$ Pa
- Second moment of area: $I_1 = I_2 = 1.3 \times 10^{-5}$ m^4
- Gravity: 9.81 m/s^2

The mass matrix, stiffness matrix, and external force for different element are shown in Appendix. Also, the initial condition for this problem are set that the body is horizontal and original is constrained by a hinge, subjected to body force (gravity force). The initial velocity are all zeros.

The Jacobian matrix under Index 1, 2, and 3 for the single field form follow as:

Index 3:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.54}$$

Index 2:

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.55}$$

Index 1:

for first time step ($n = 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ \frac{1}{1-\gamma}\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.56}$$

and next every time step ($n > 1$)

$$\mathbf{J} = \begin{bmatrix} \Lambda_6 W_1 \mathbf{M} + \Lambda_3 W_3 \Delta t^2 (\mathbf{J}_{C\lambda}(\tilde{u}, \lambda_{n+1}) + \mathbf{K}_T(u) + \mathbf{M}_T(u, a)) & \mathbf{C}(\tilde{u})^T \\ (1+\gamma)\mathbf{C}(u_{n+1})^T & 0 \end{bmatrix} \tag{4.57}$$

where γ is the time shift:

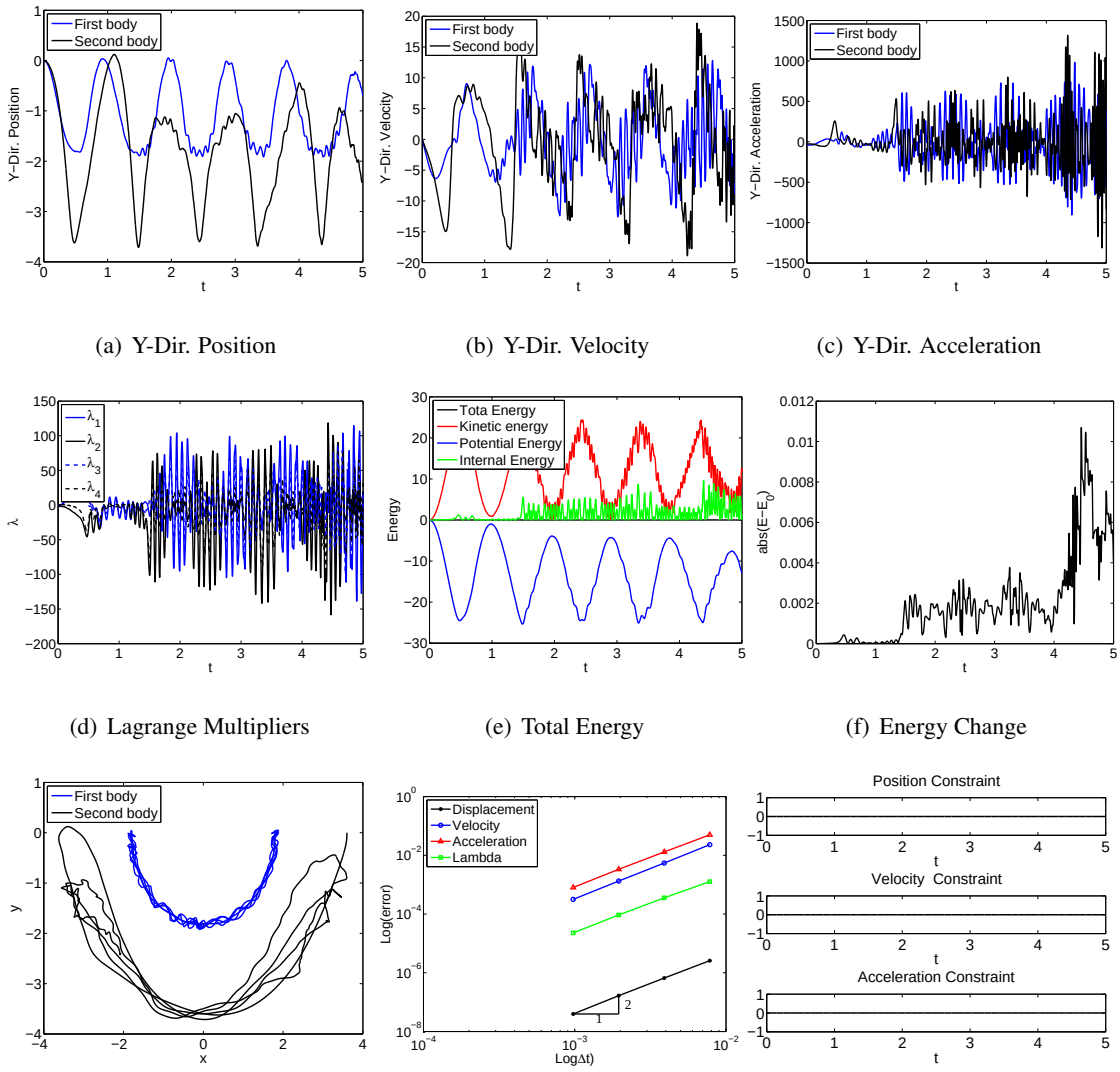
$$\gamma = \Lambda_6 W_1 - W_1 \tag{4.58}$$

and $\mathbf{J}_{C\lambda}(u, \lambda)$ is a zero matrix since the constraint equation is linear. Also, $\mathbf{K}_T$ and $\mathbf{M}_T$ are the tangent stiffness matrix and mass matrix as follows:

$$\begin{aligned} \mathbf{M}_T &= \frac{\partial \mathbf{M}(u)a}{\partial u} \\ \mathbf{K}_T &= \frac{\partial \mathbf{F}_{int}}{\partial u} \end{aligned} \tag{4.59}$$

For different frames of approach, the components for each tangent matrix is different (if matrix is linear, the corresponding tangent matrix is zero). The problem was run using a-form representation of the single field form algorithm. The time step used in all simulations was $\Delta t = 0.001$ to ensure the convergence for all cases (IRF/FRF/ANCF), and the nonlinear iteration tolerance is 10^{-8}, and the number of iterations limited to 100 per time step. The simulation duration was first set to 5 seconds, through not all cases were able to provide a stable solution. Also, for each case, the simulation were run first for a low Young's modulus (flexible case $E = 10^6$) and then for high Youngs modulus (quasi-rigid case $E = 10^{10}$).

Elastic Bar

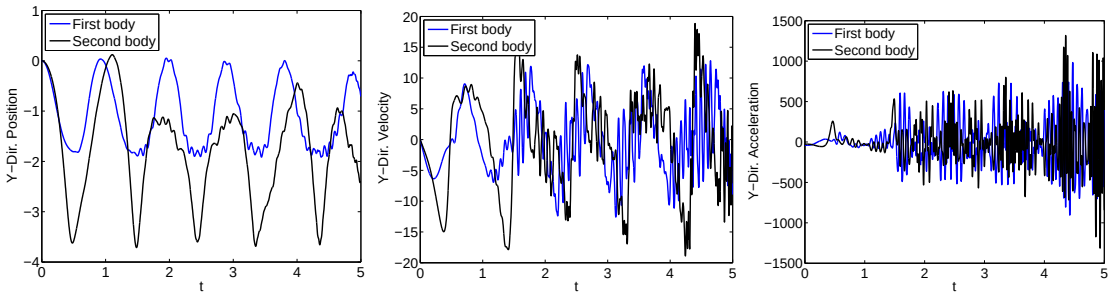


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

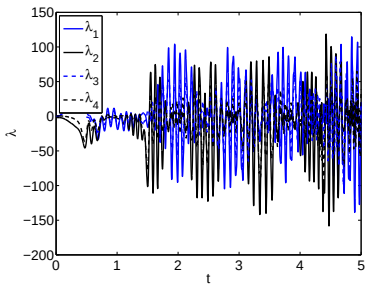
Figure 4.135: Double flexible pendulum with bar element in IRF: U0(1,1,0) - Index 3.



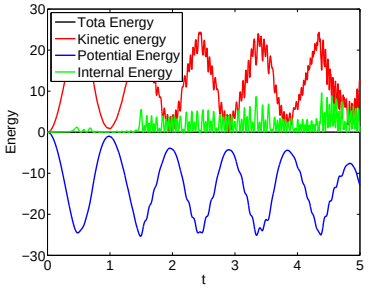
(a) Y-Dir. Position

(b) Y-Dir. Velocity

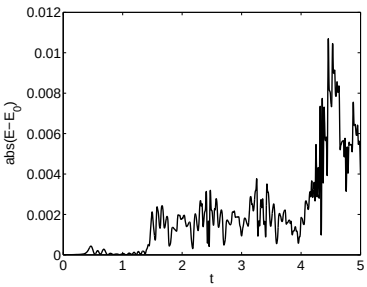
(c) Y-Dir. Acceleration



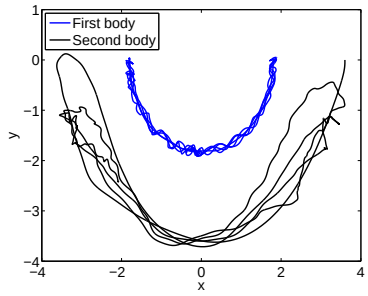
(d) Lagrange Multipliers



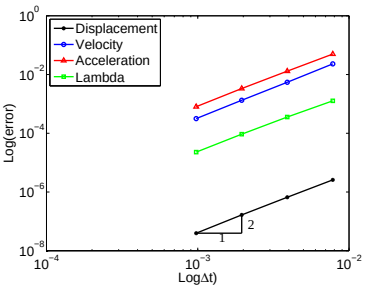
(e) Total Energy



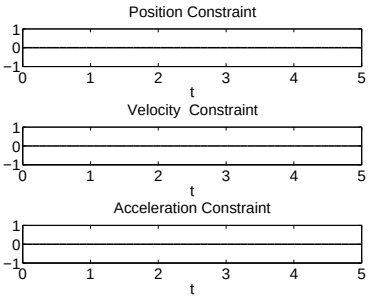
(f) Energy Change



(g) Trajectory

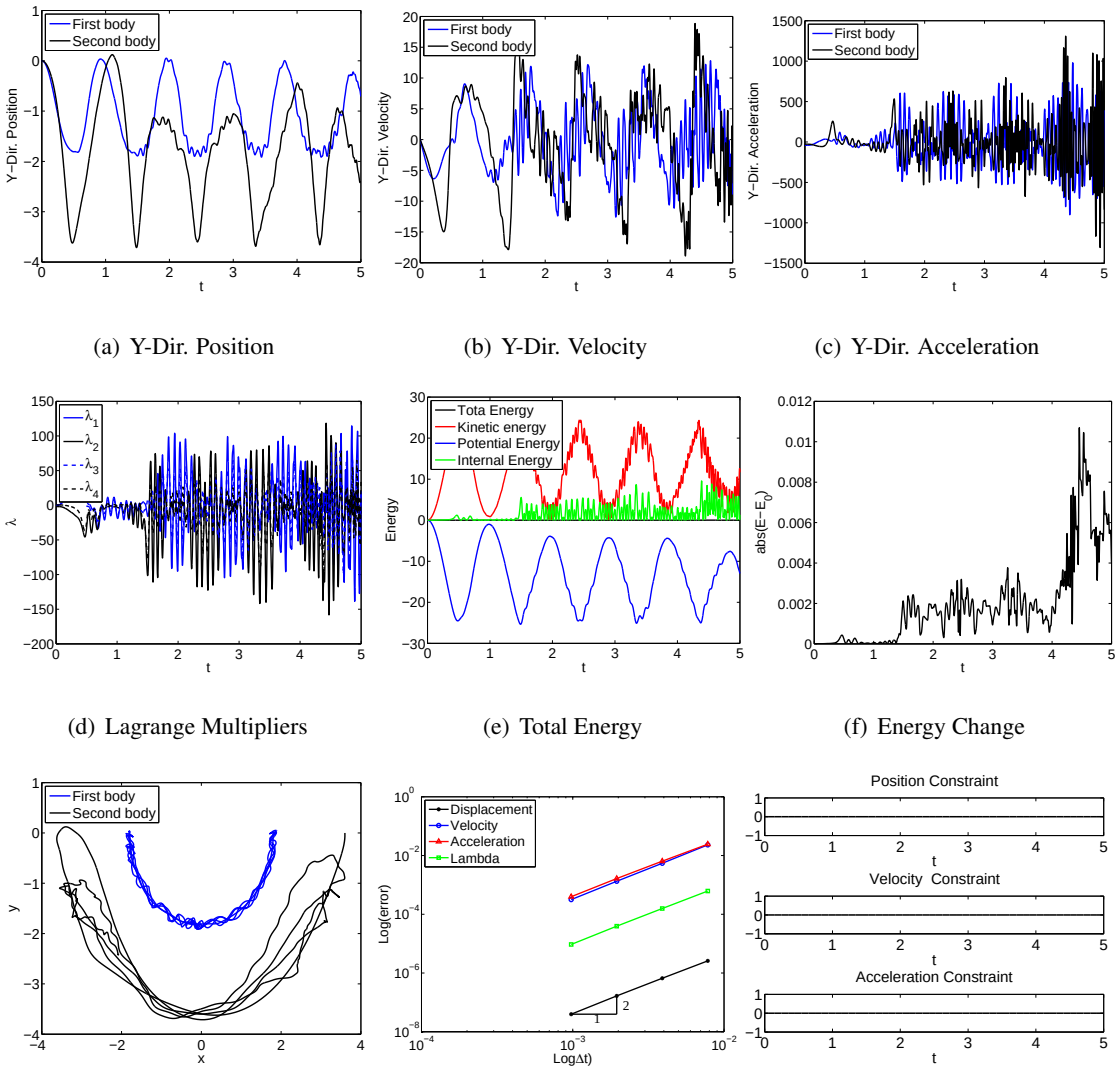


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.136: Double flexible pendulum with bar element in IRF: V0(1,1,0) - Index 3.

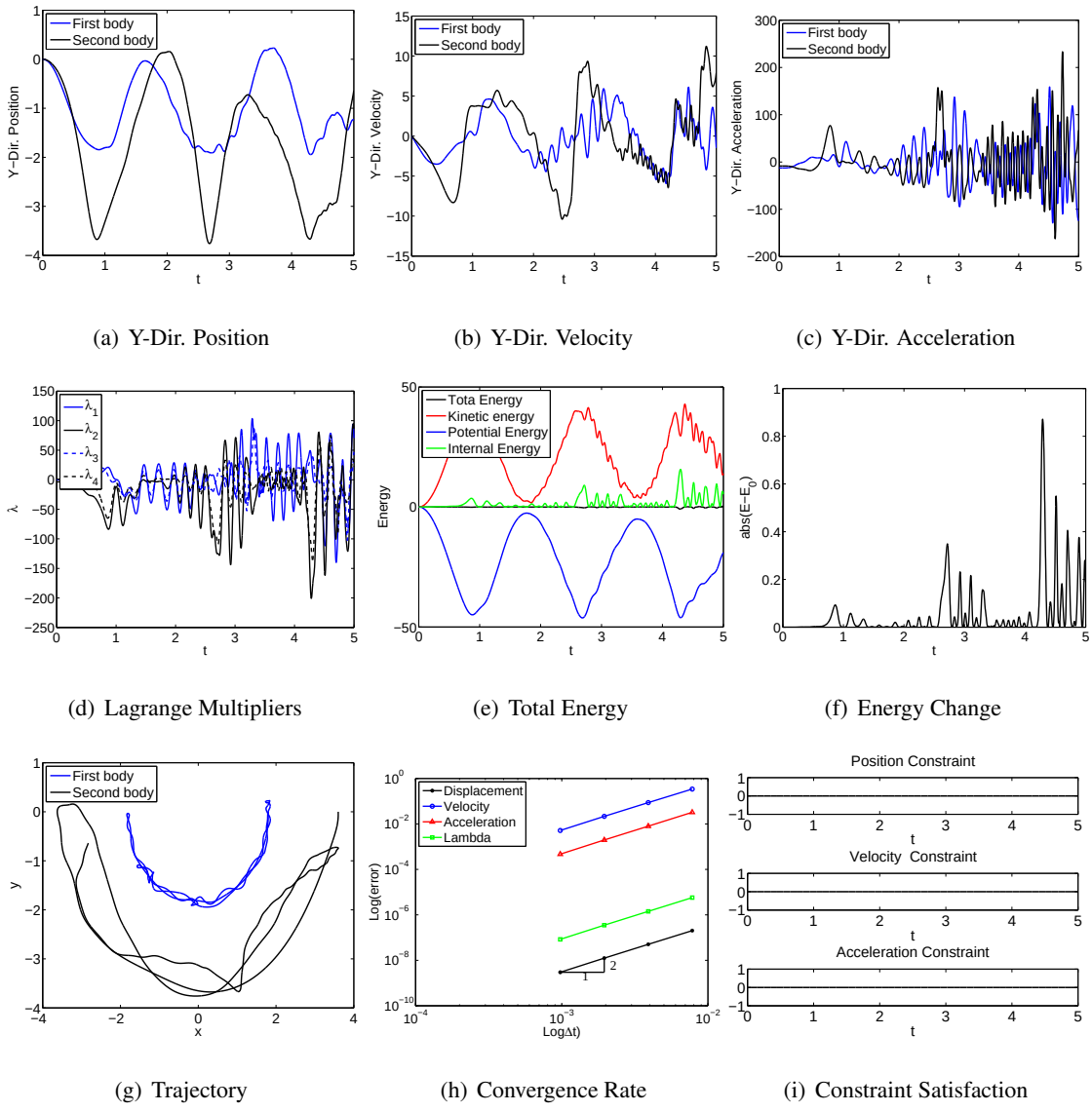


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.137: Double flexible pendulum with bar element in IRF: U0V0(1,1,1) - Index 3.

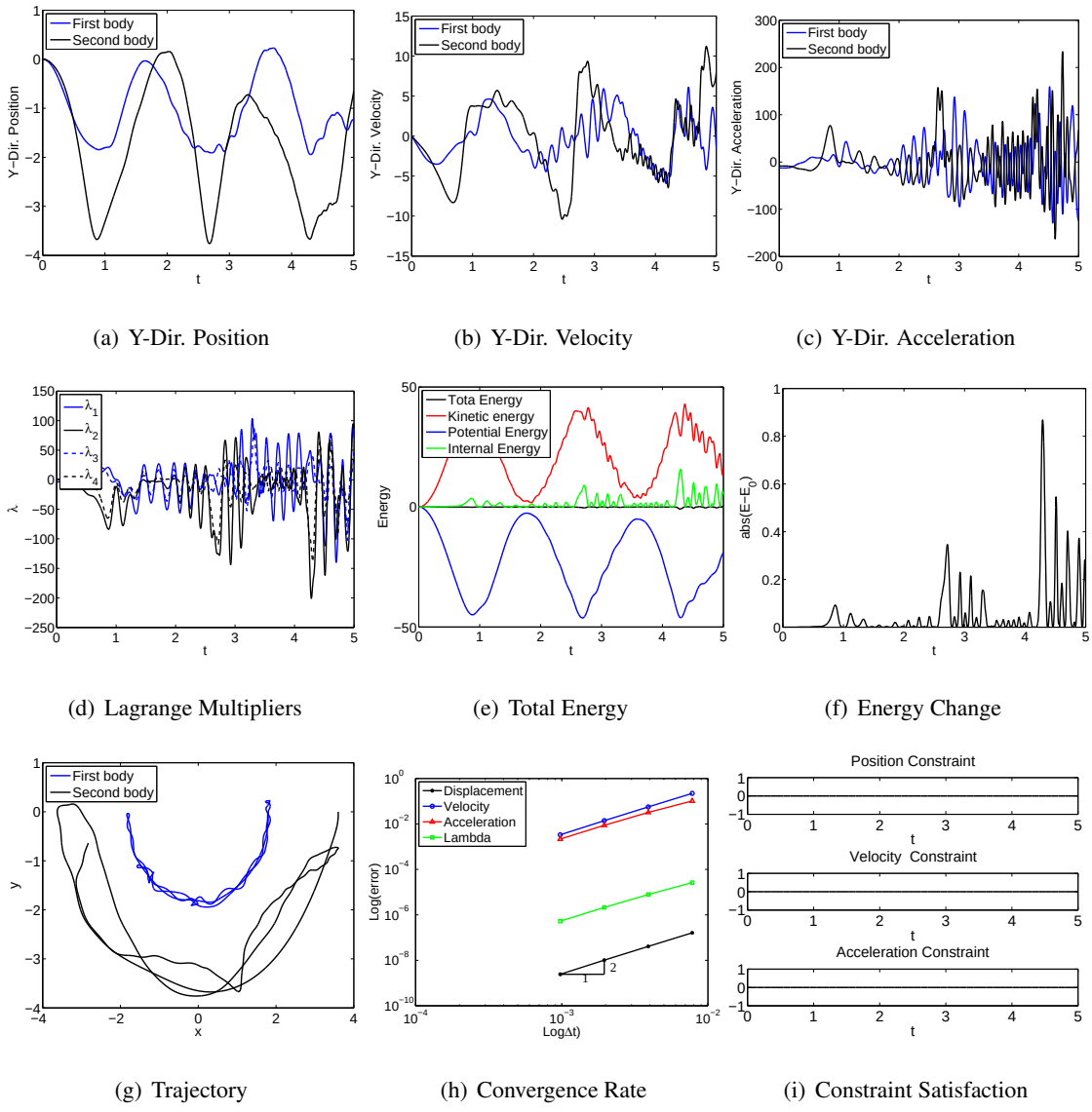


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.138: Double flexible pendulum with bar element in ANCF-S: U0(1,1,0) - Index 3.

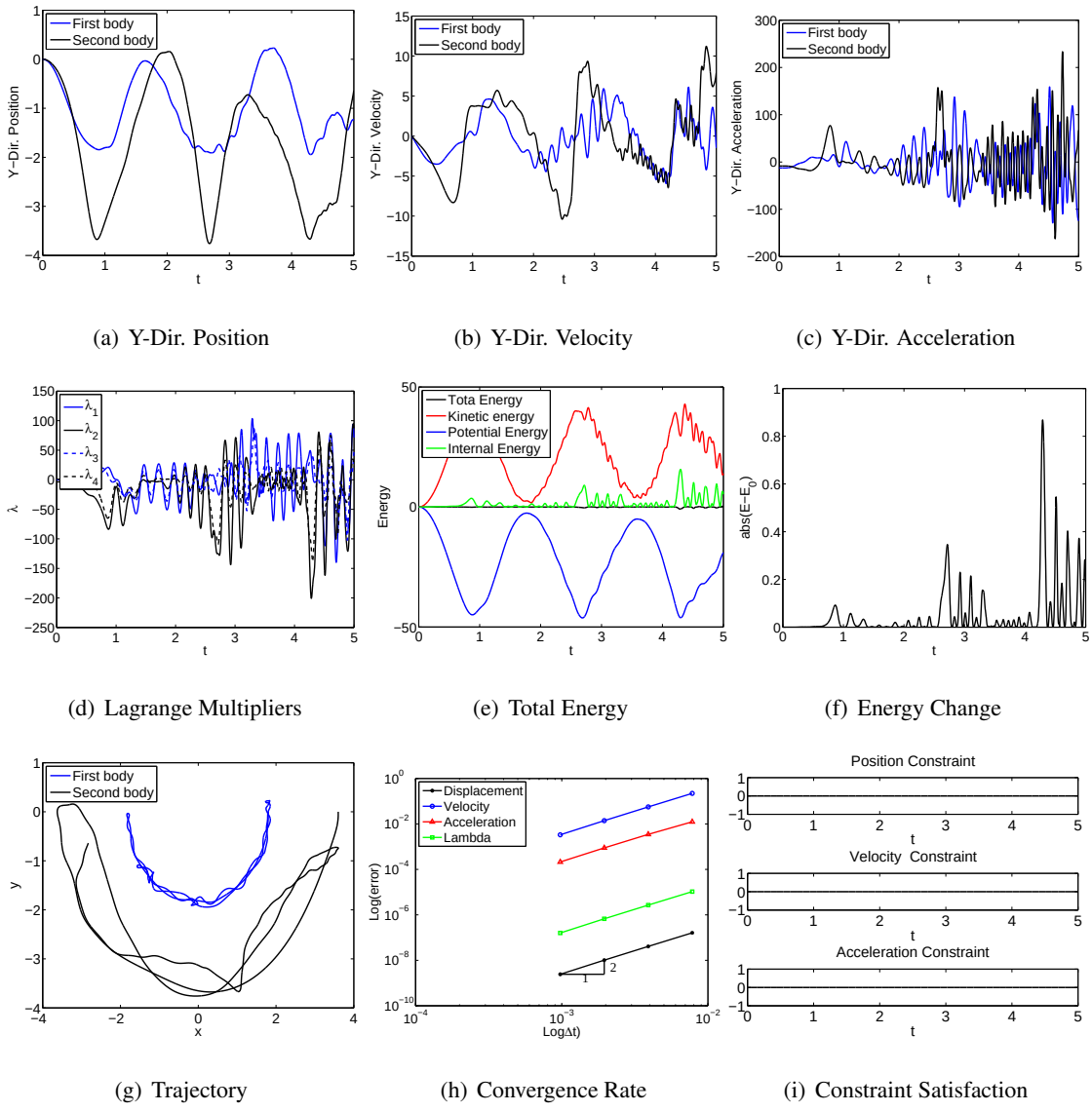


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.139: Double flexible pendulum with bar element in ANCF-S: V0(1,1,0) - Index 3.

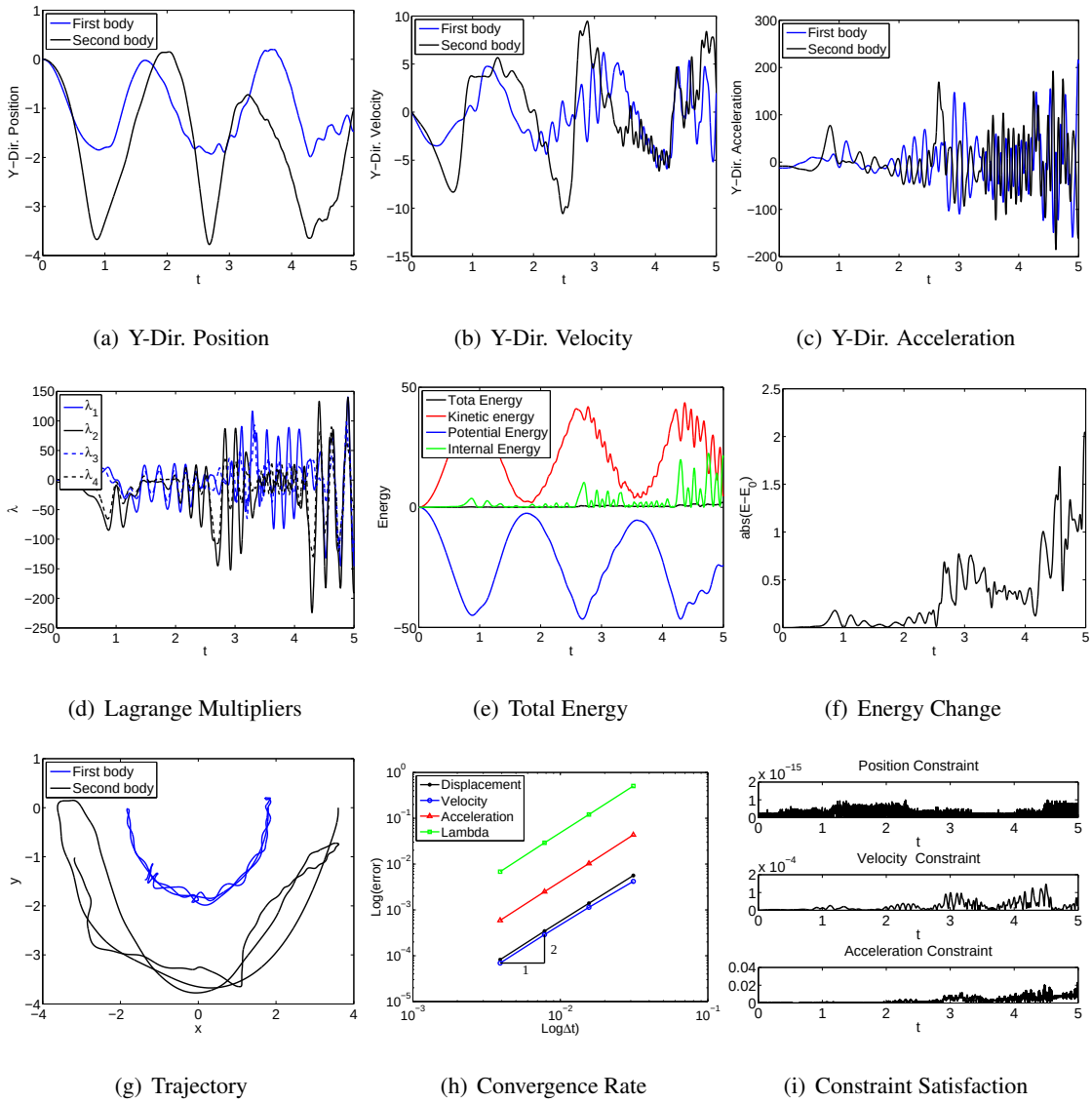


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.140: Double flexible pendulum with bar element in ANCF-S: U0V0(1,1,1) - Index 3.

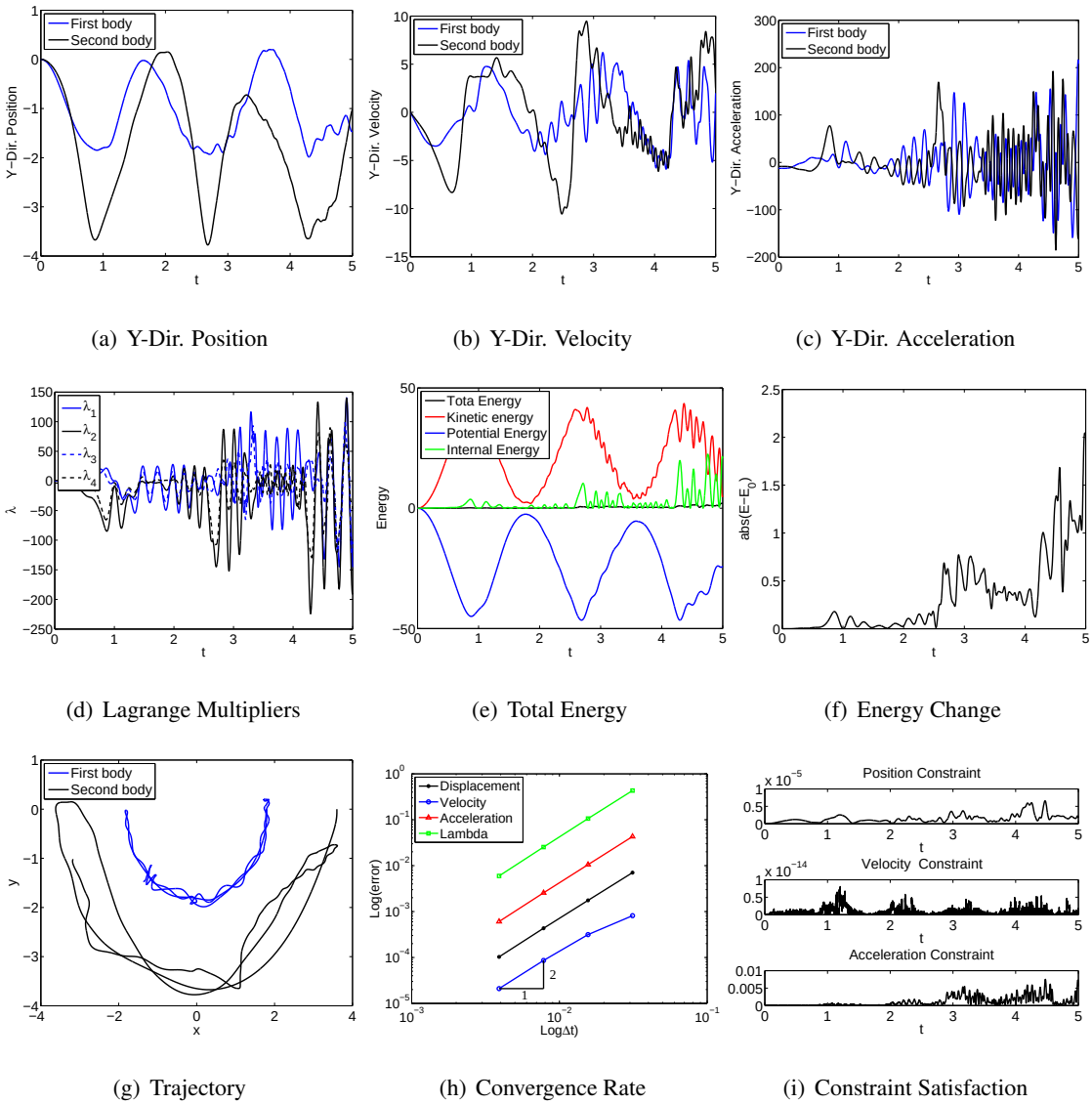


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.141: Double flexible pendulum with bar element in FRF: U0(1,1,0) - Index 3.

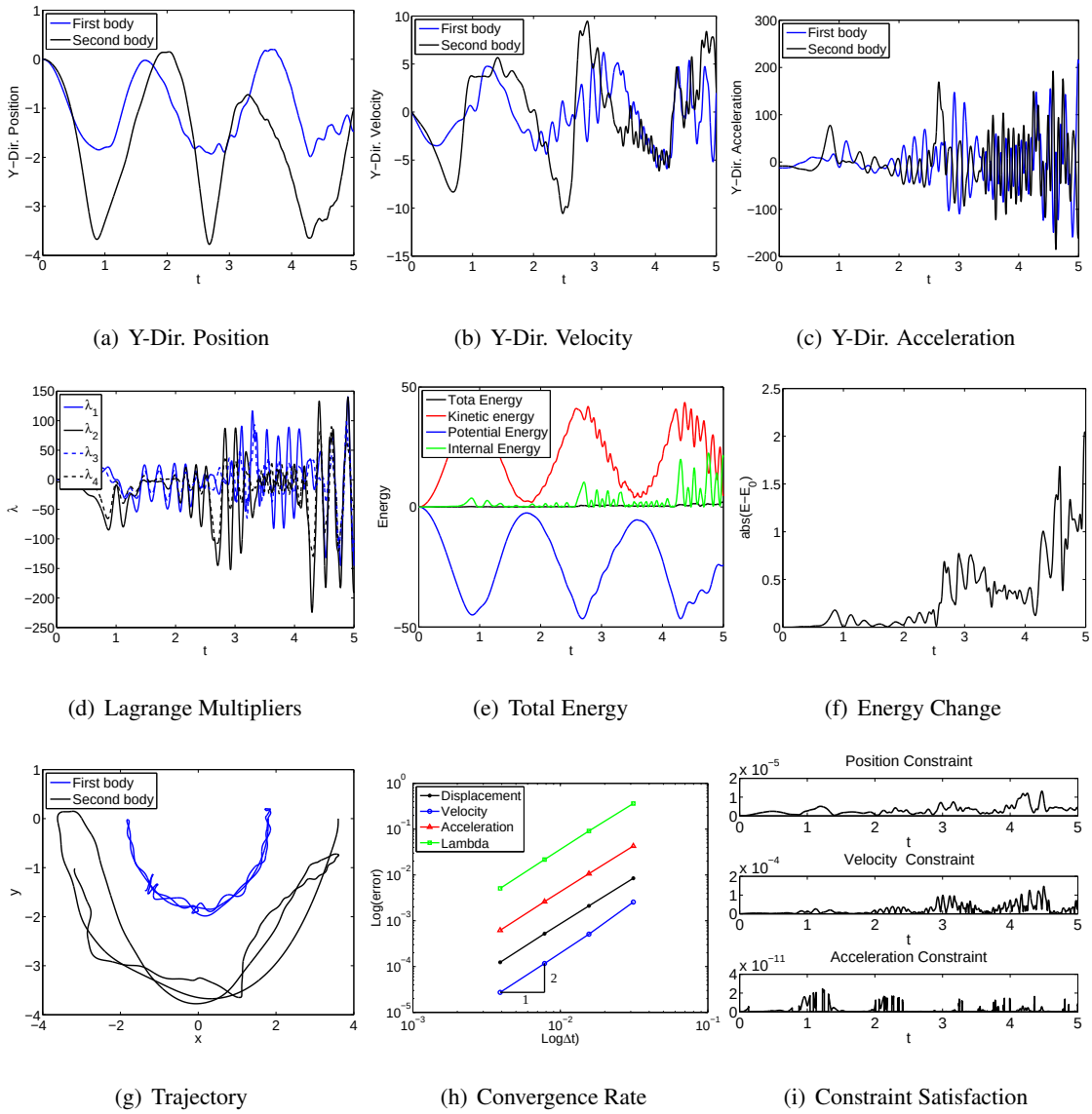


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.142: Double flexible pendulum with bar element in FRF: U0(1,1,0) - Index 2.

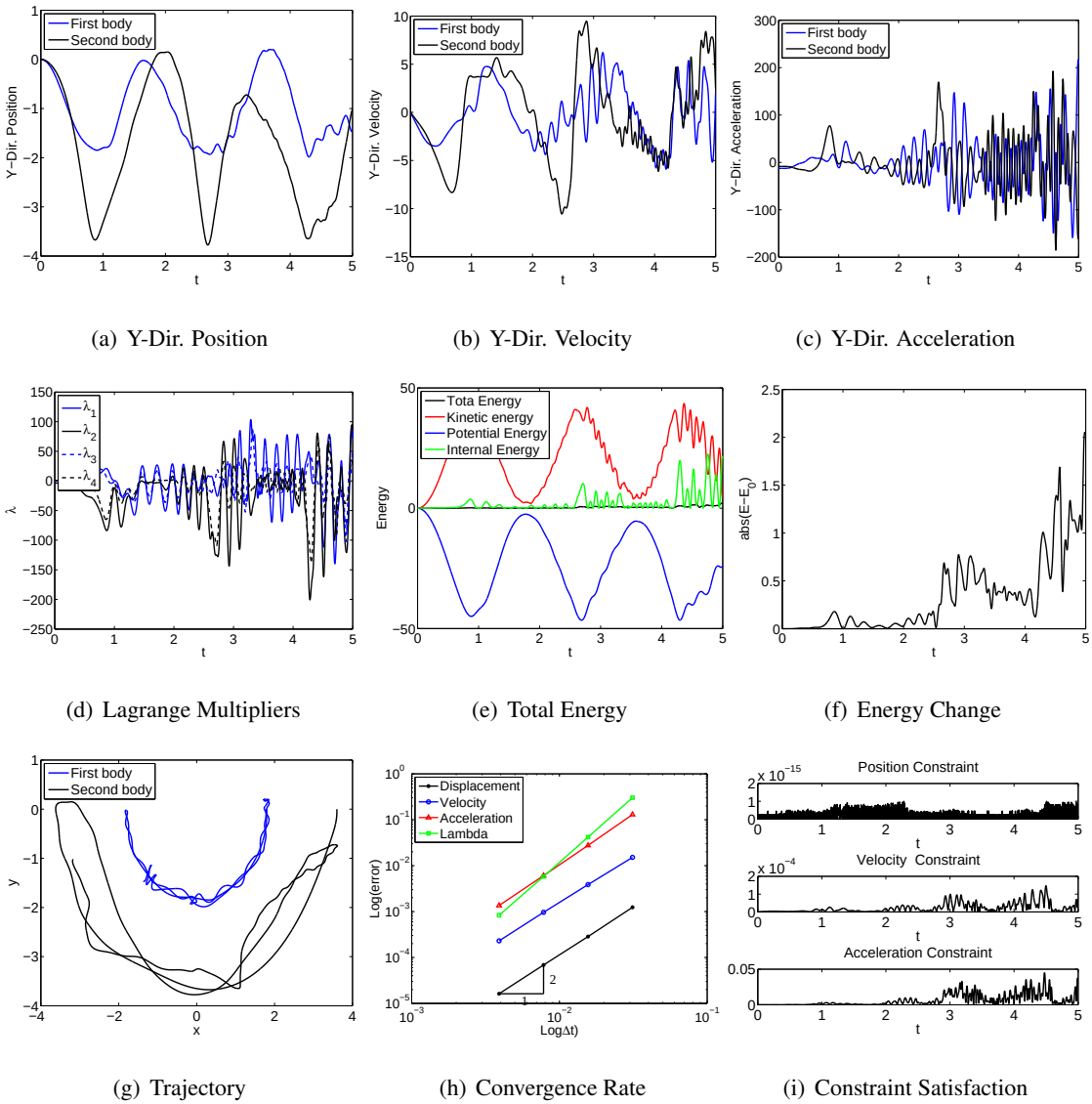


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.143: Double flexible pendulum with with bar element in FRF: U0(1,1,0) - Index 1.

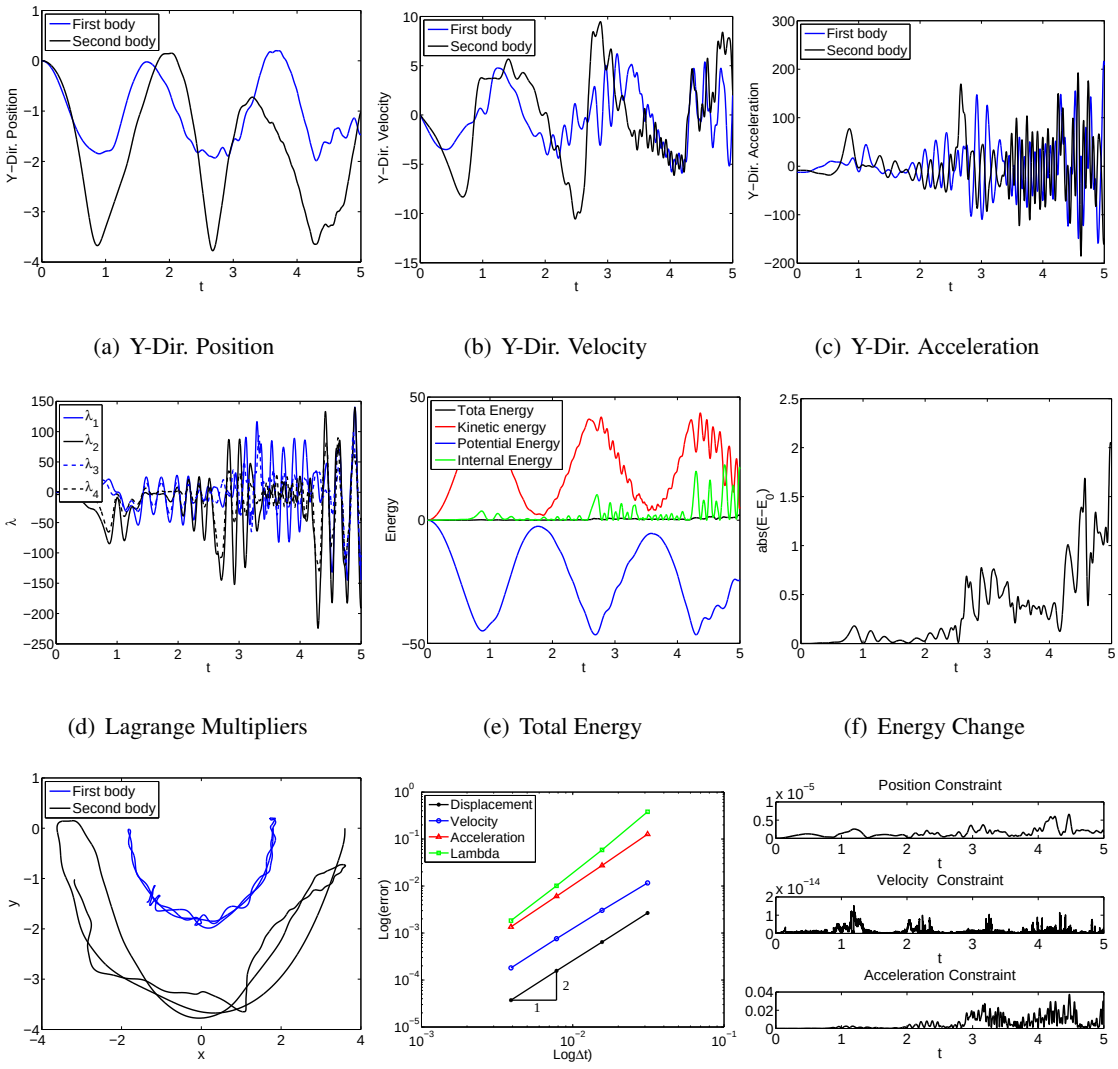


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.144: Double flexible pendulum with with bar element in FRF: V0(1,1,0) - Index 3.

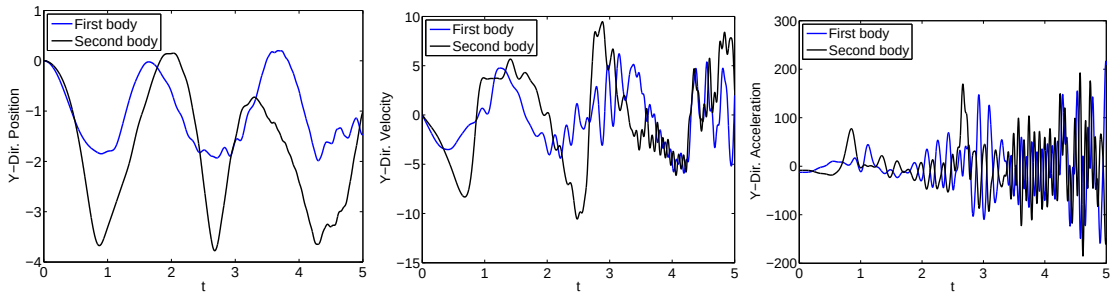


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

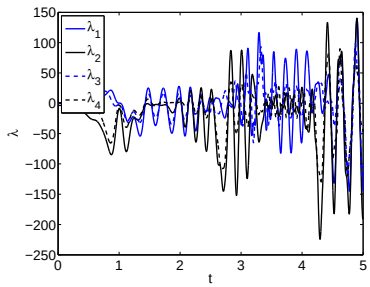
Figure 4.145: Double flexible pendulum with with bar element in FRF: V0(1,1,0) - Index 2.



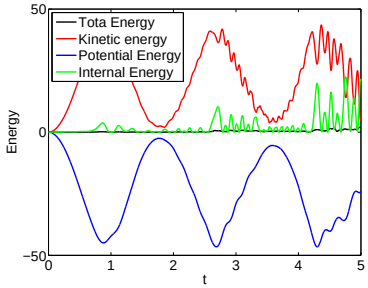
(a) Y-Dir. Position

(b) Y-Dir. Velocity

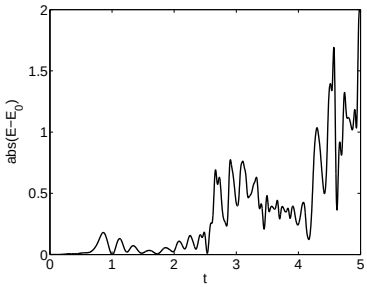
(c) Y-Dir. Acceleration



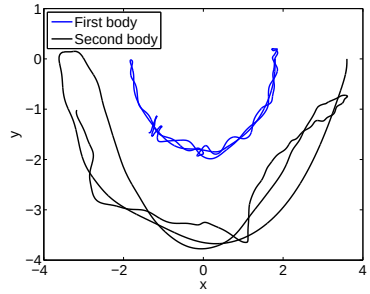
(d) Lagrange Multipliers



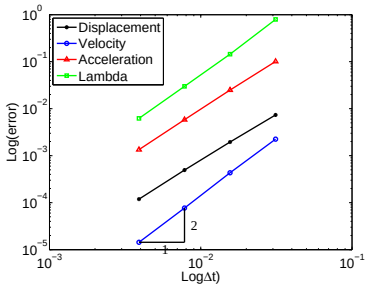
(e) Total Energy



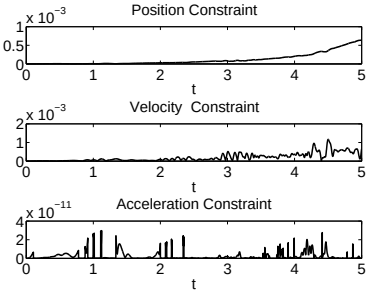
(f) Energy Change



(g) Trajectory

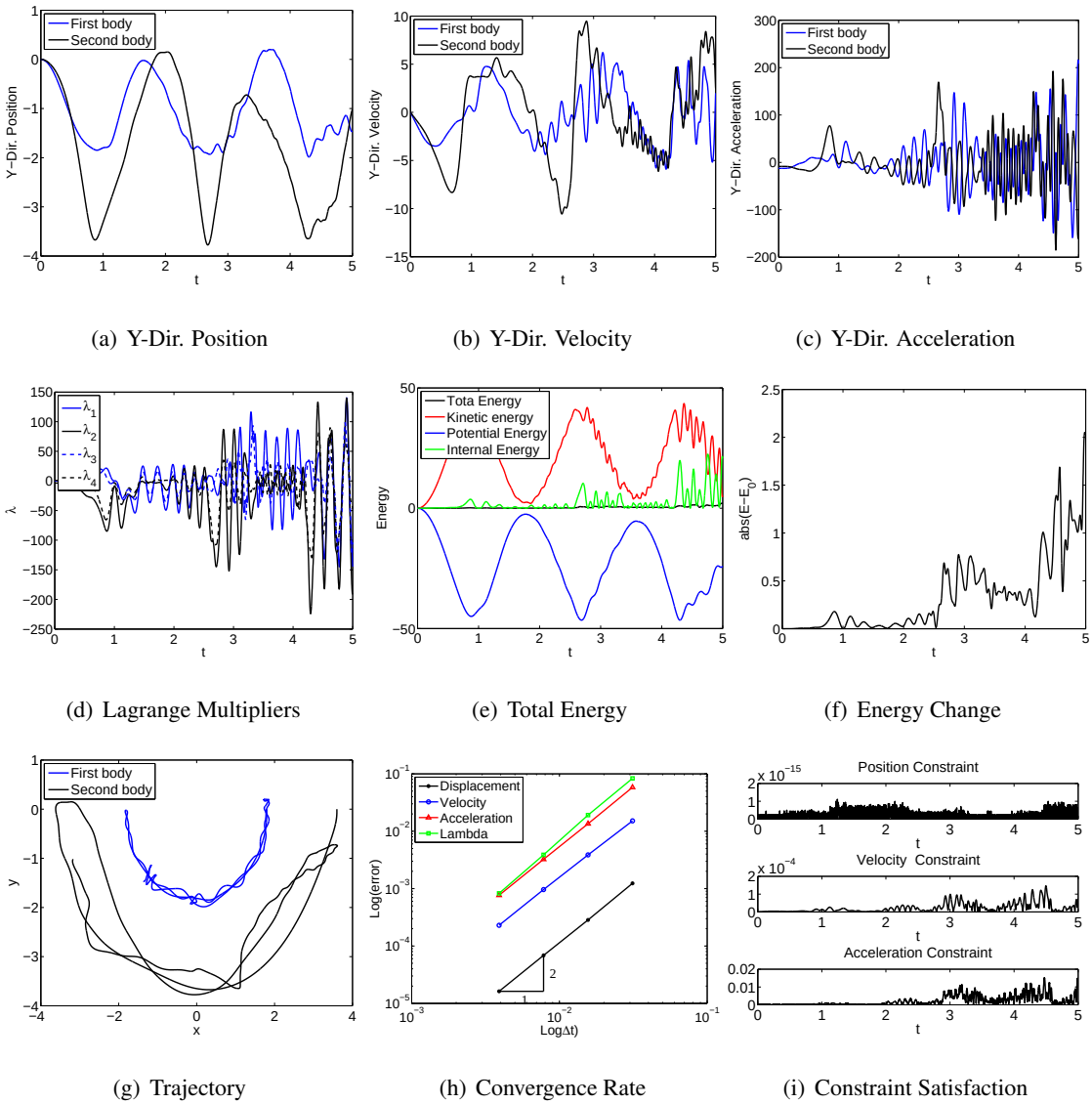


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.146: Double flexible pendulum with with bar element in FRF: V0(1,1,0) - Index 1.

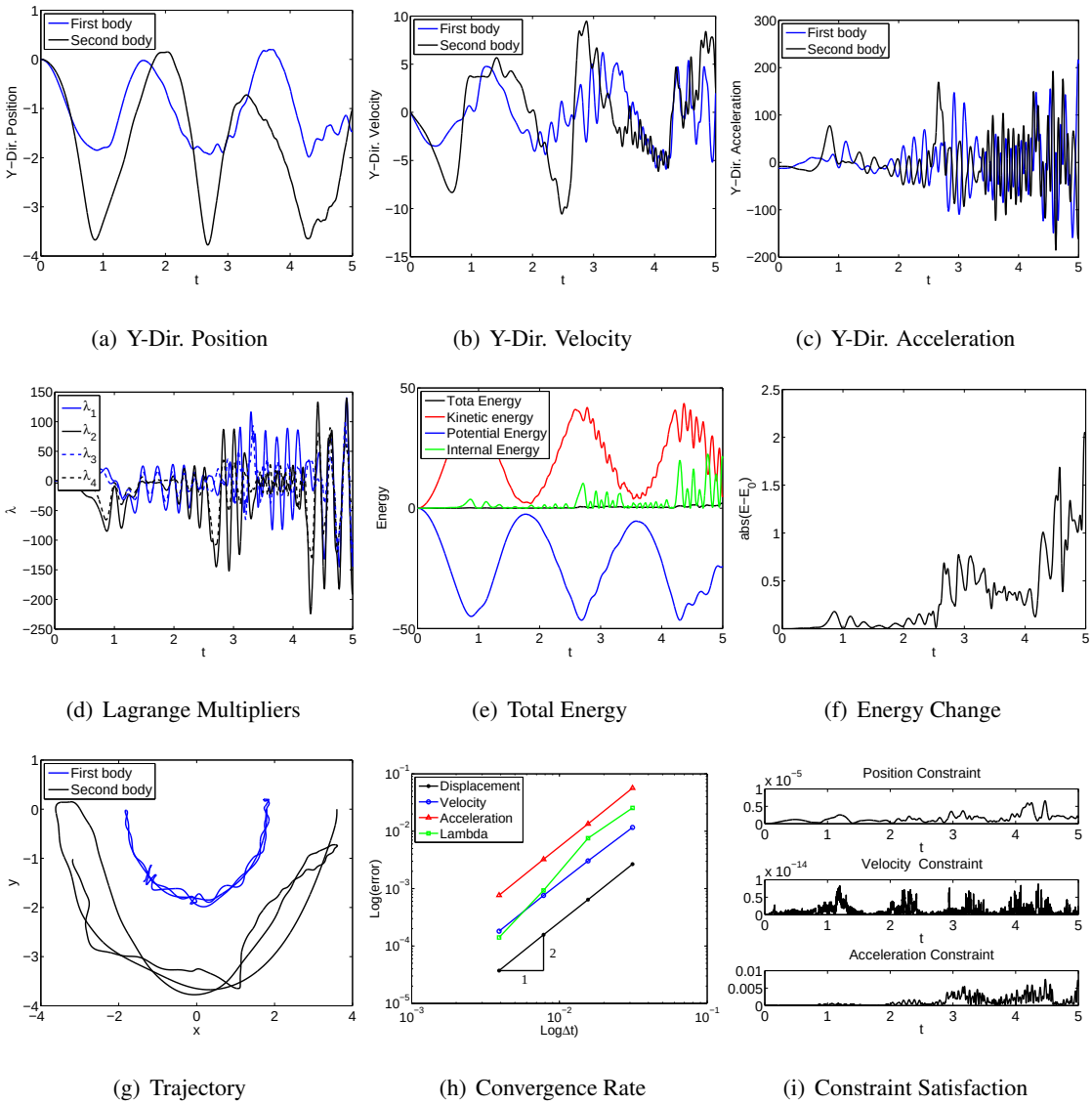


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.147: Double flexible pendulum with bar element in FRF: U0V0(1,1,1) - Index 3.

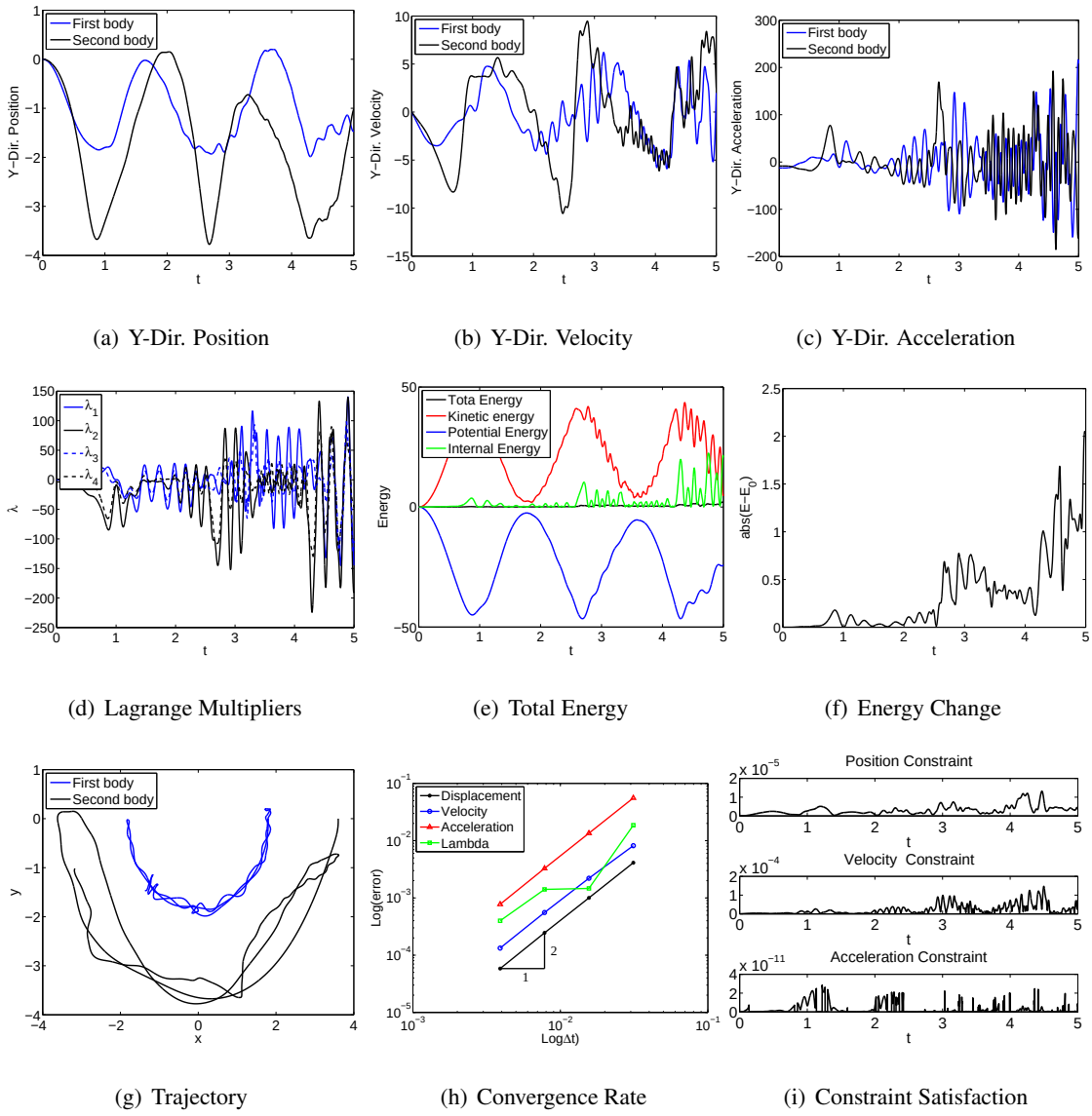


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.148: Double flexible pendulum with bar element in FRF: U0V0(1,1,1) - Index 2.



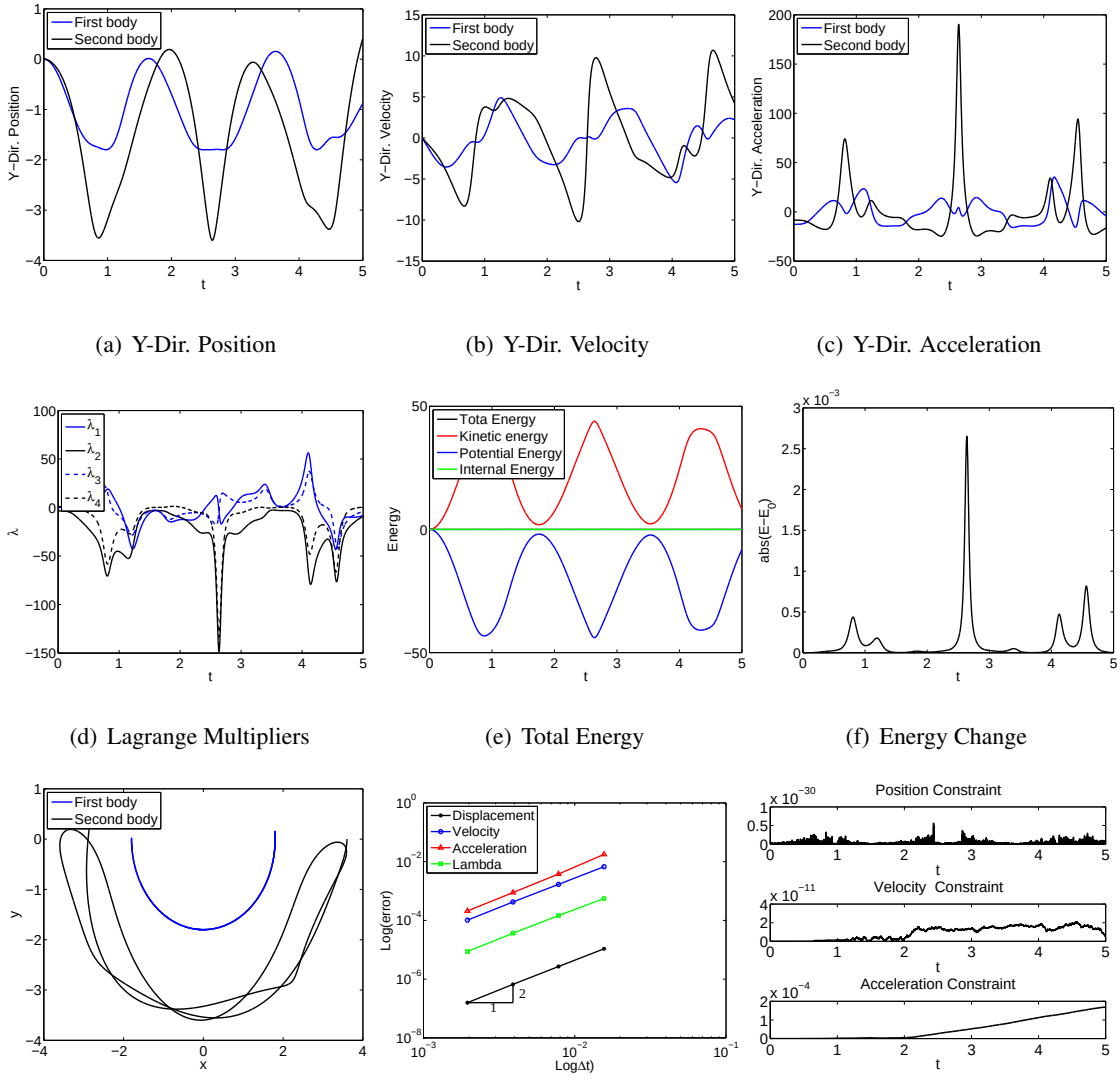
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.149: Double flexible pendulum with bar element in FRF: U0V0(1,1,1) - Index 1.

Elastic bar: Quasi-Rigid Case

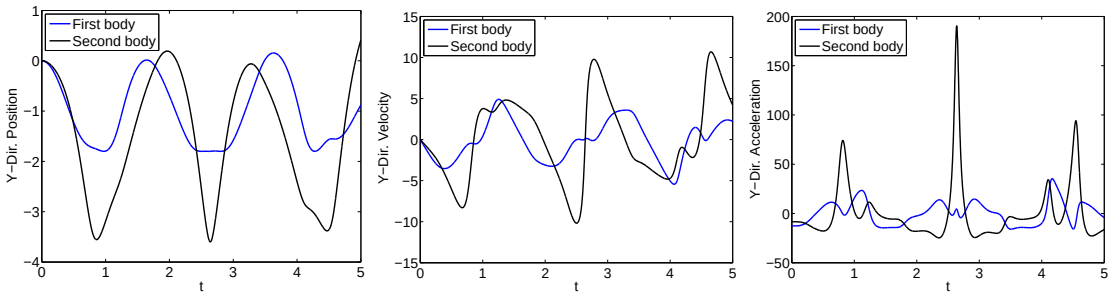


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

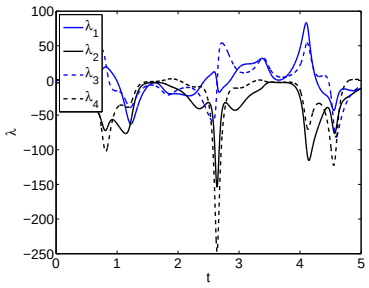
Figure 4.150: Double quasi-rigid pendulum with bar element in ANCF-S: U0(1,1,0) - Index 3.



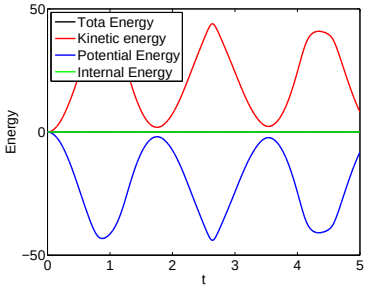
(a) Y-Dir. Position

(b) Y-Dir. Velocity

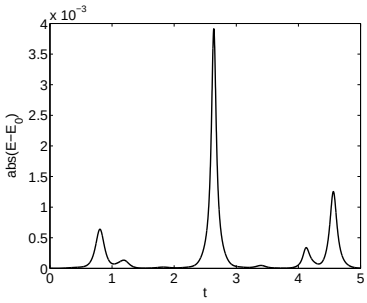
(c) Y-Dir. Acceleration



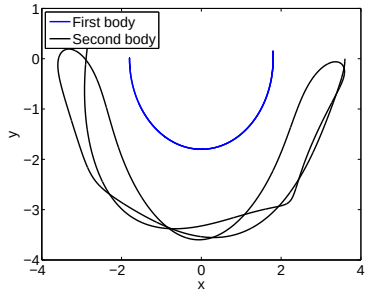
(d) Lagrange Multipliers



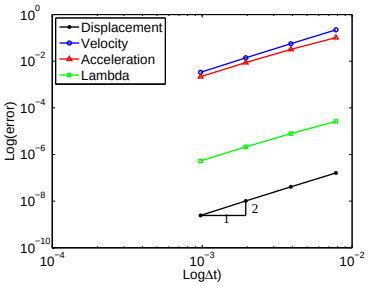
(e) Total Energy



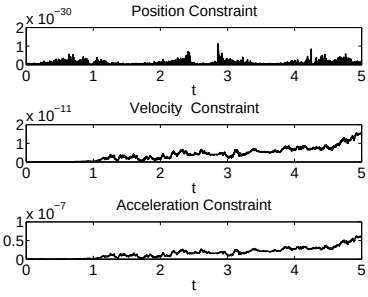
(f) Energy Change



(g) Trajectory

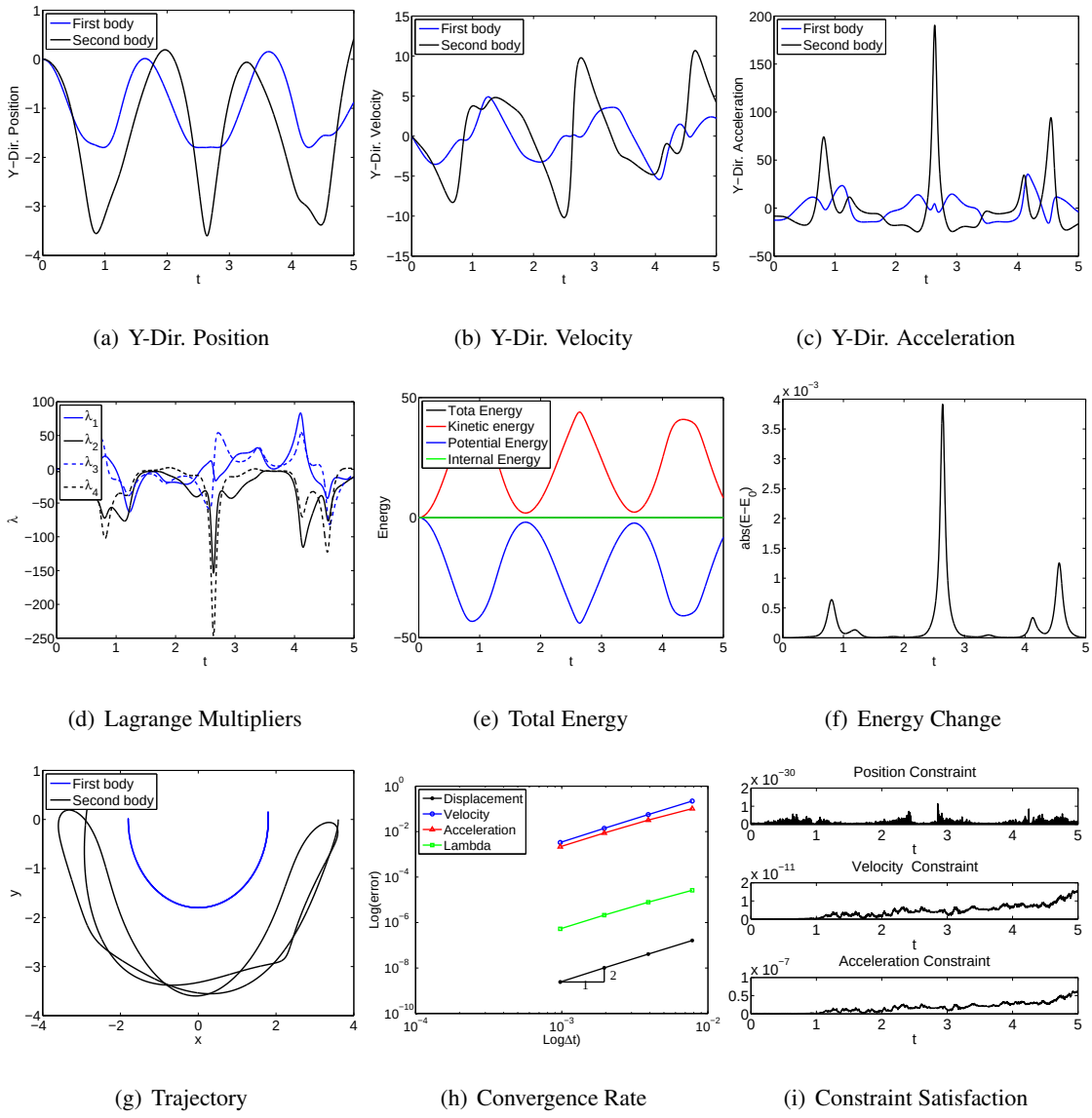


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.151: Double quasi-rigid pendulum with bar element in ANCF-S: V0(1,1,0) - Index 3.

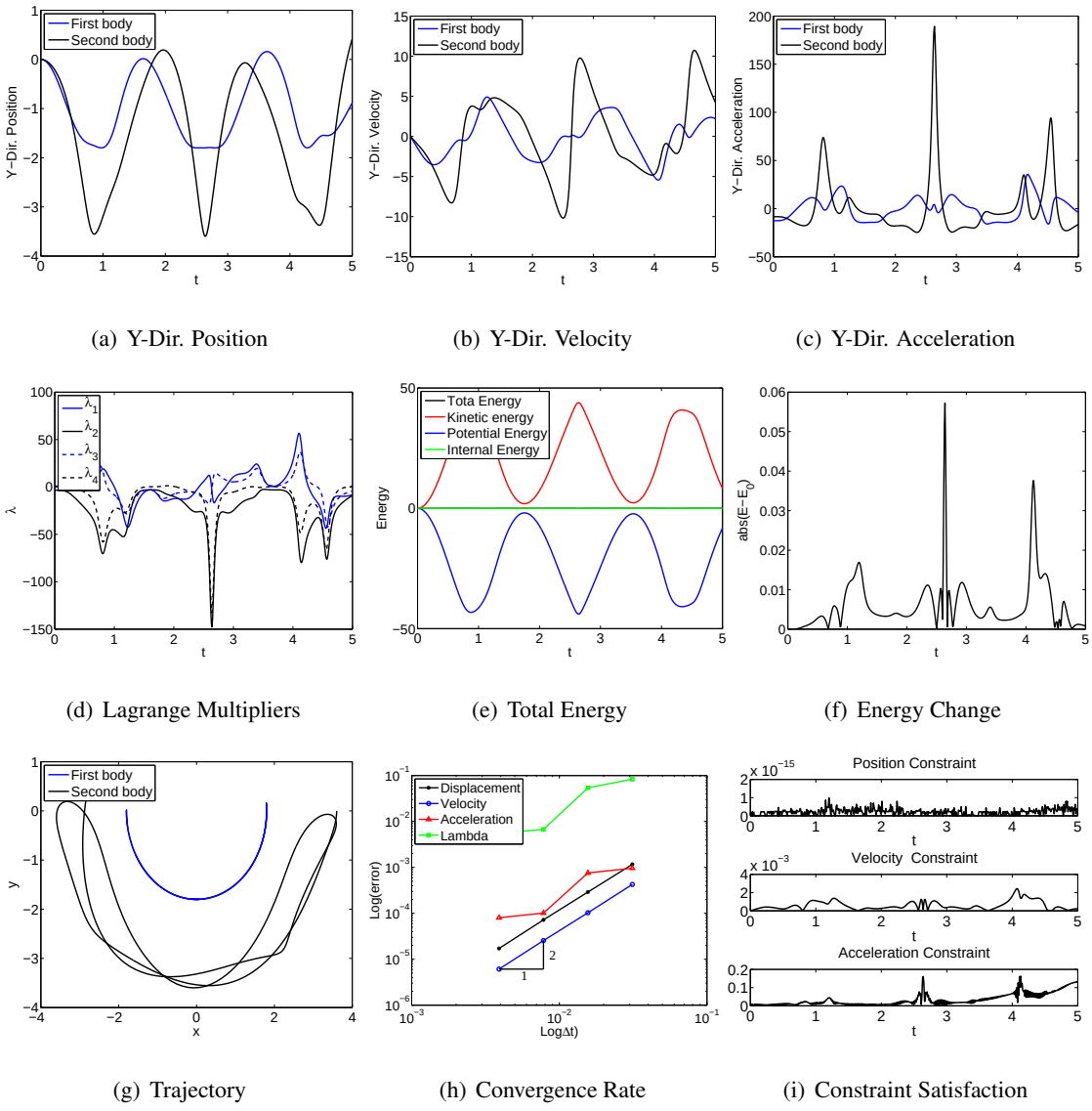


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.152: Double quasi-rigid pendulum with bar element in ANCF-S: U0V0(1,1,1) - Index 3.

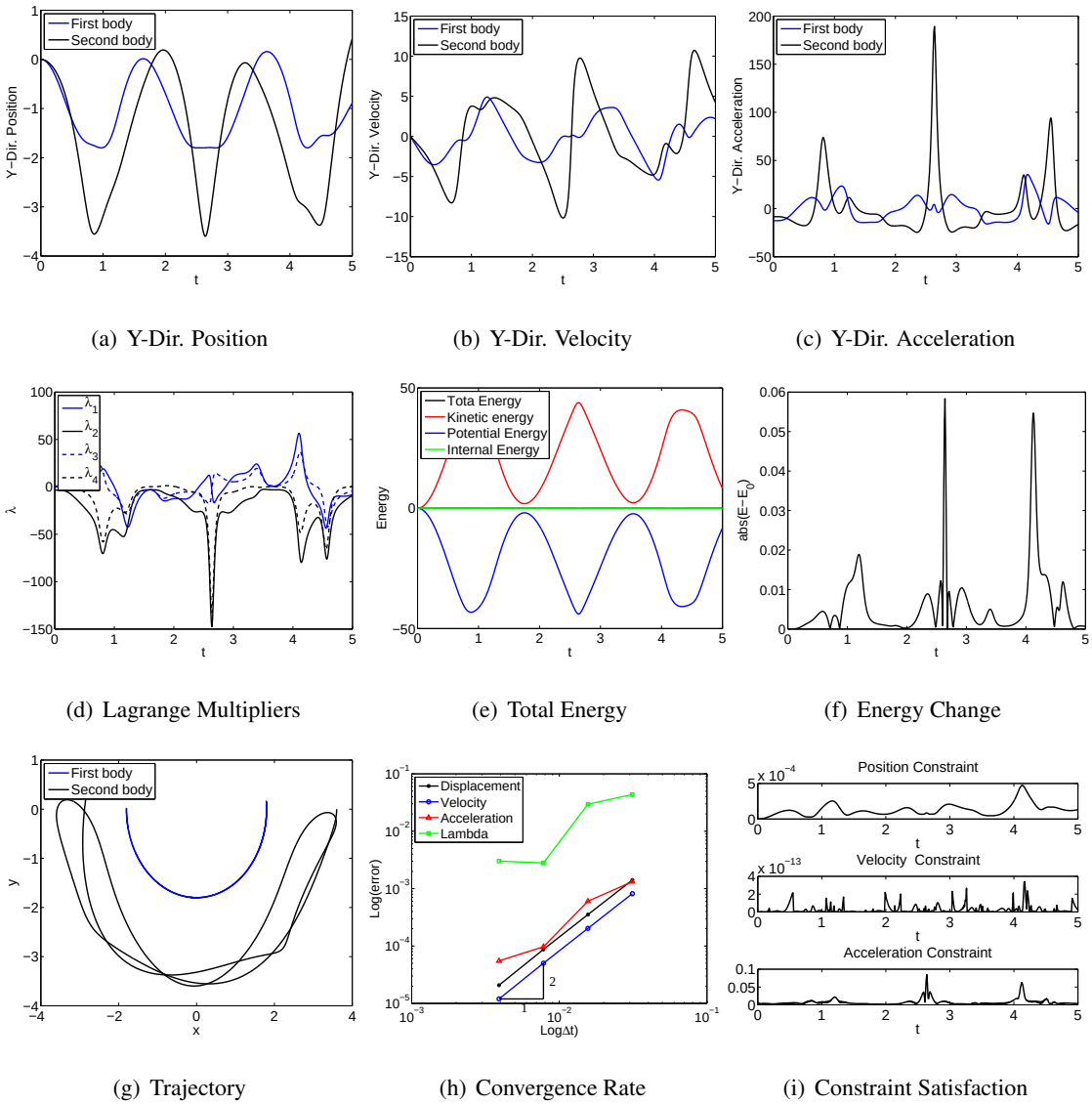


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.153: Double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 3.

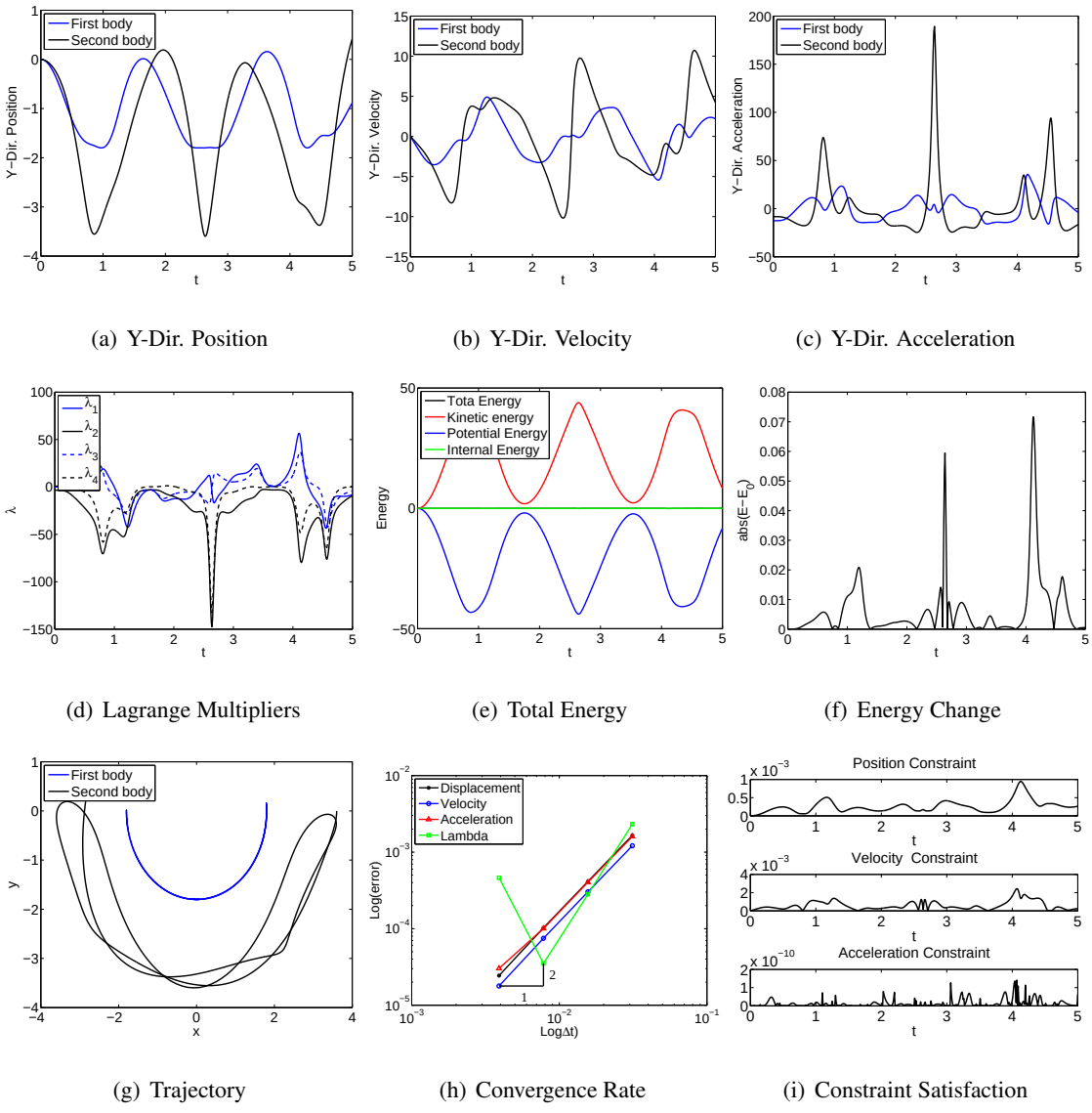


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.154: Double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 2.

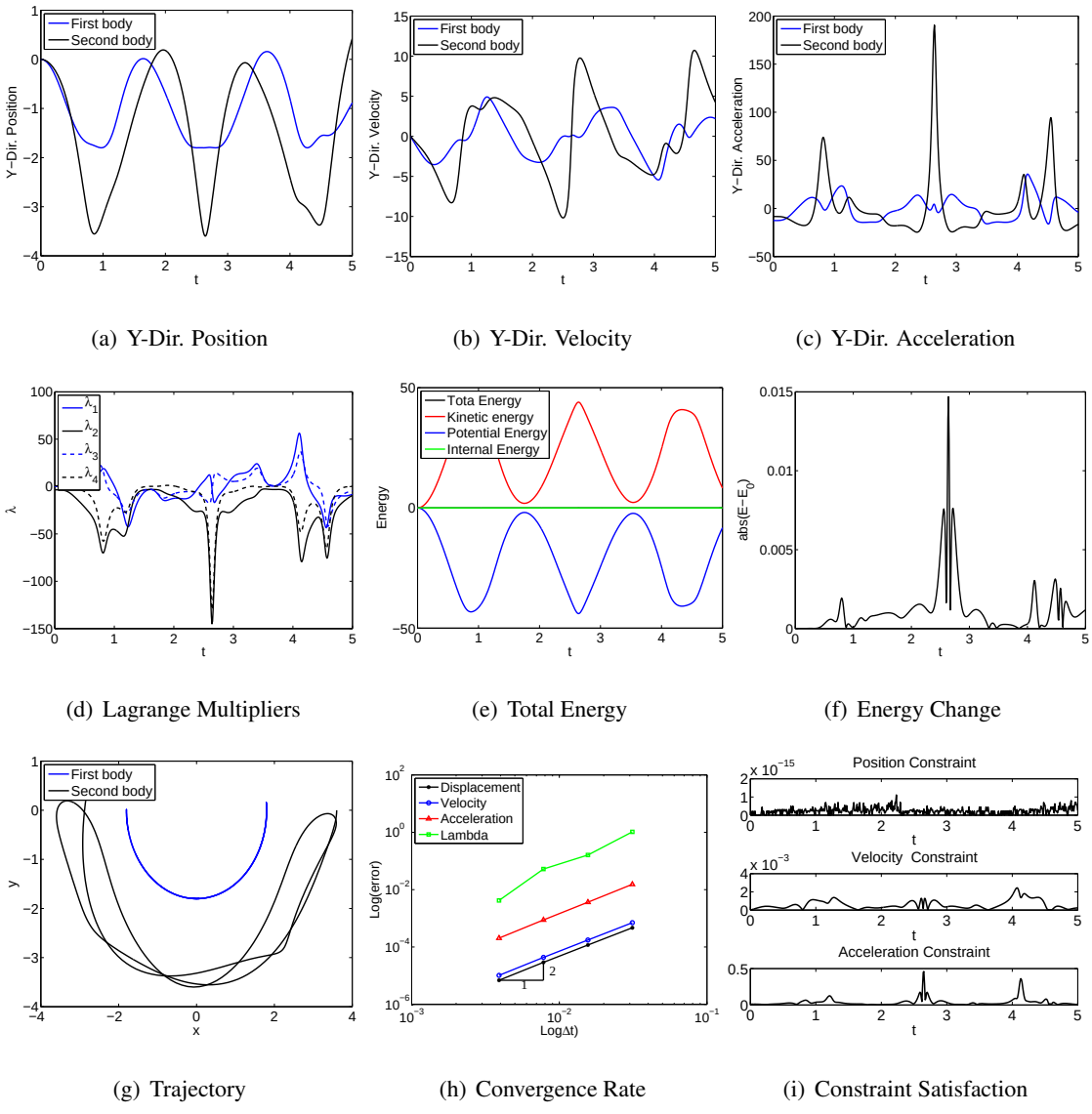


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.155: Double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 1.

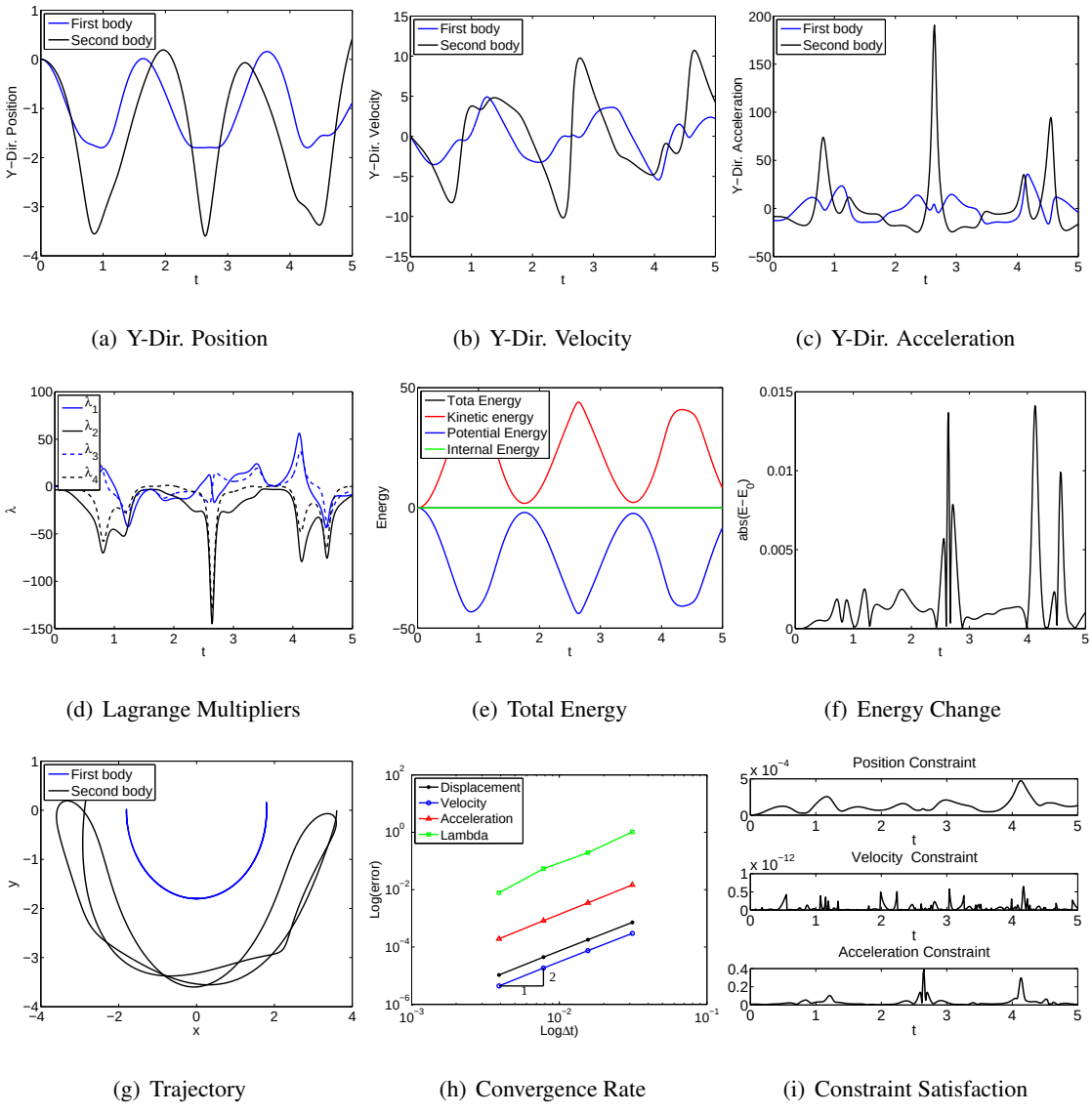


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.156: Double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 3.

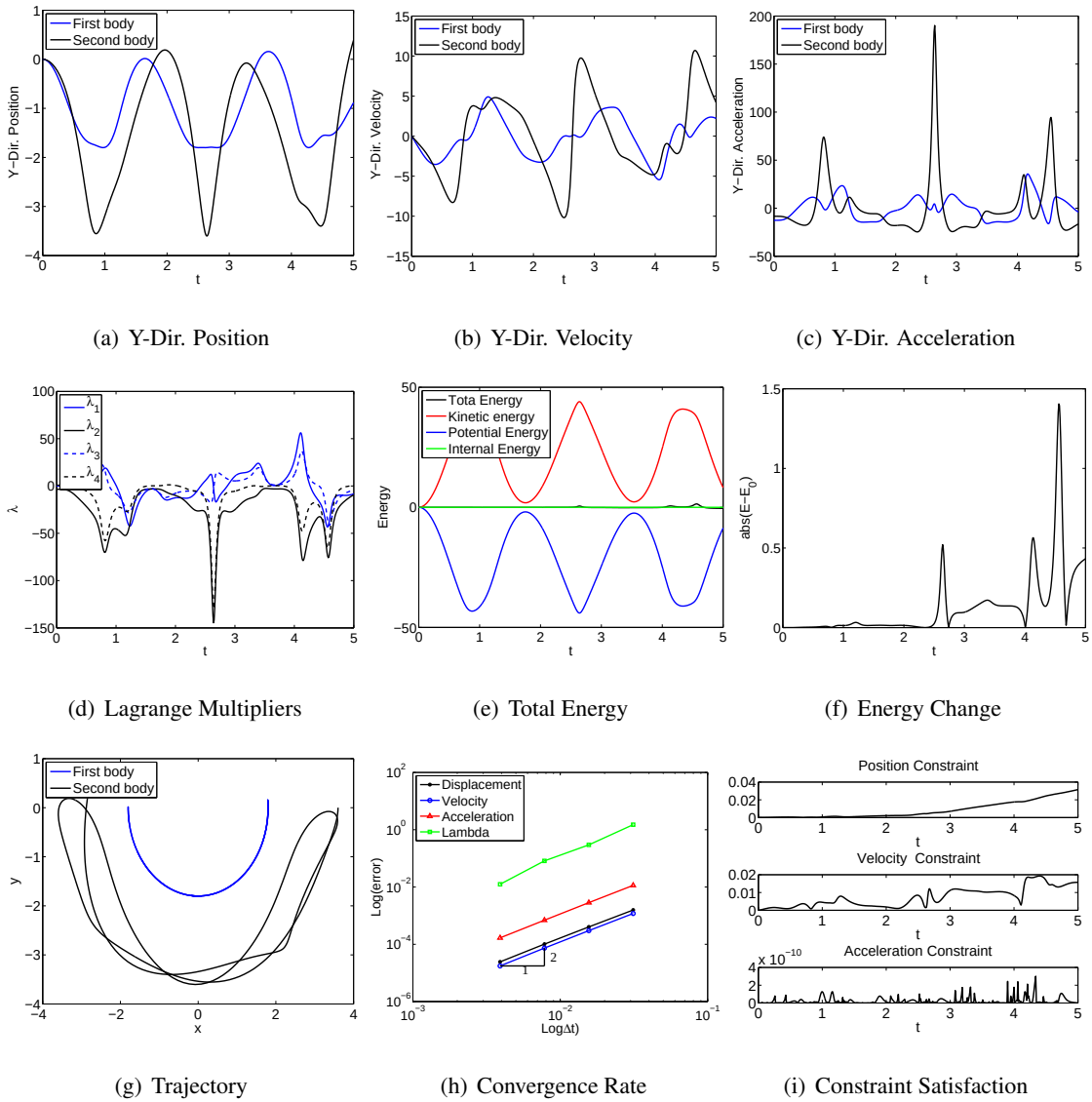


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.157: Double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 2.

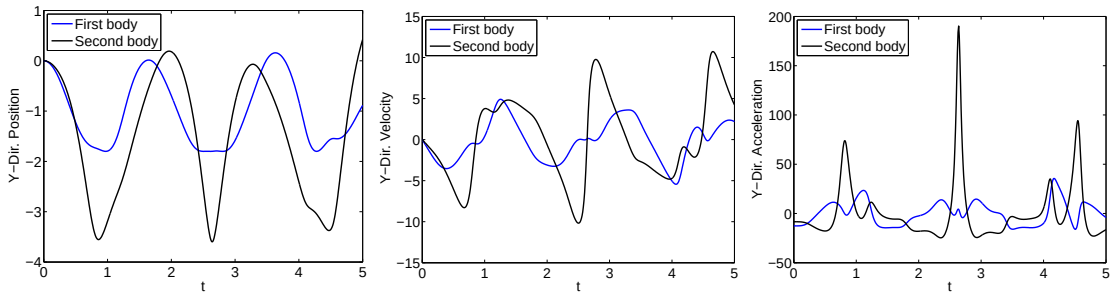


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

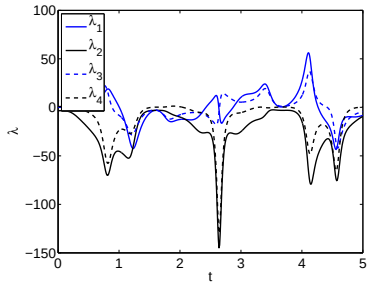
Figure 4.158: Double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 1.



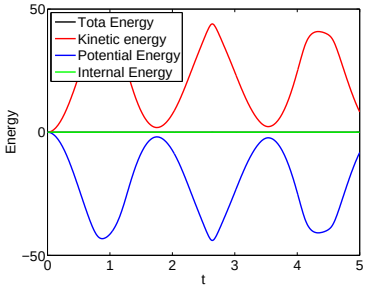
(a) Y-Dir. Position

(b) Y-Dir. Velocity

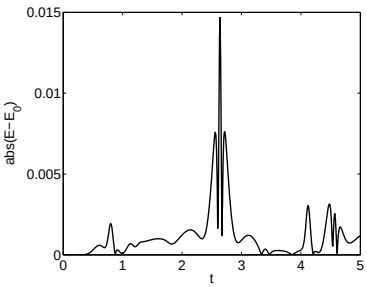
(c) Y-Dir. Acceleration



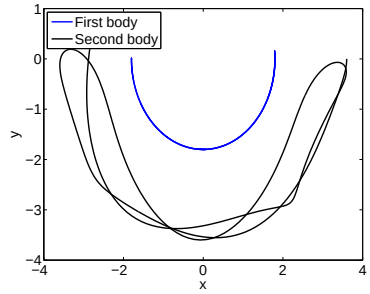
(d) Lagrange Multipliers



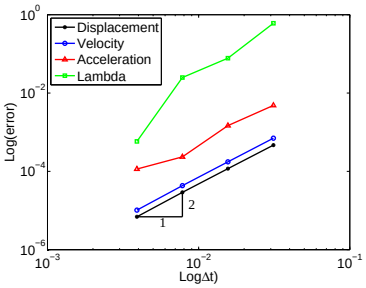
(e) Total Energy



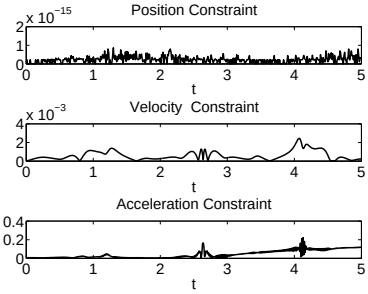
(f) Energy Change



(g) Trajectory

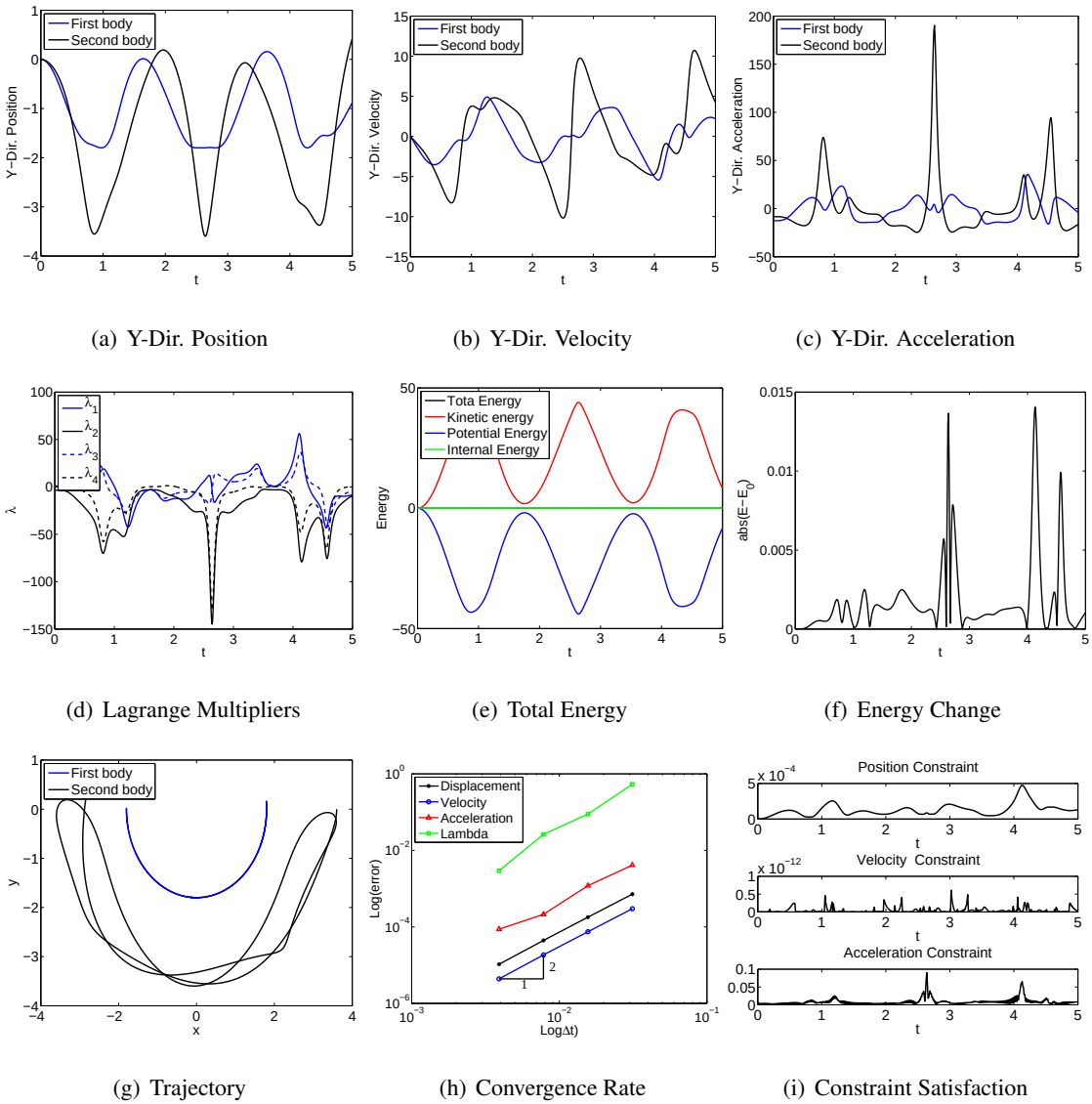


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.159: Double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 3.

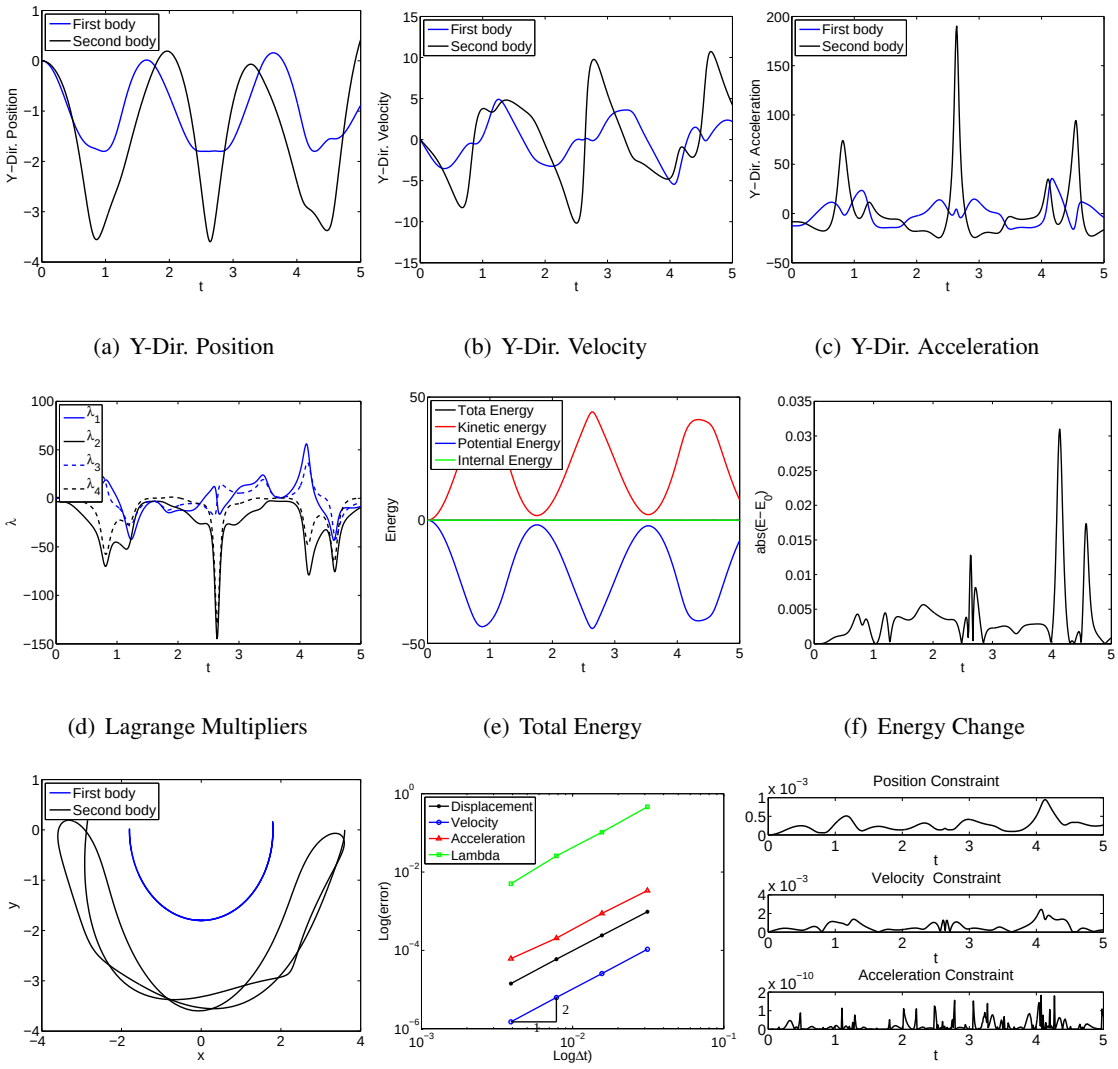


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.160: Double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 2.



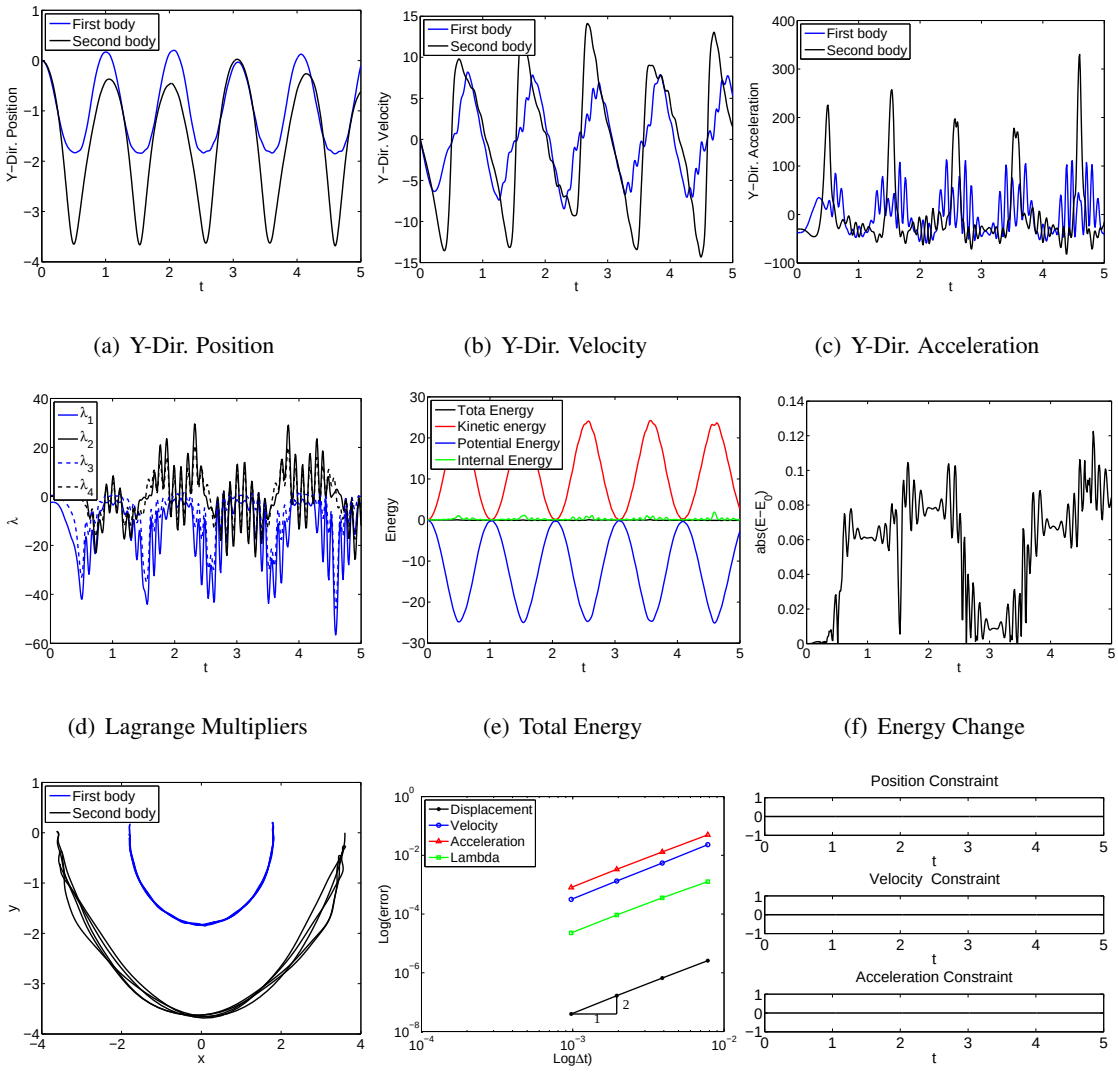
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.161: Double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 1.

Euler-Bernoulli Beam

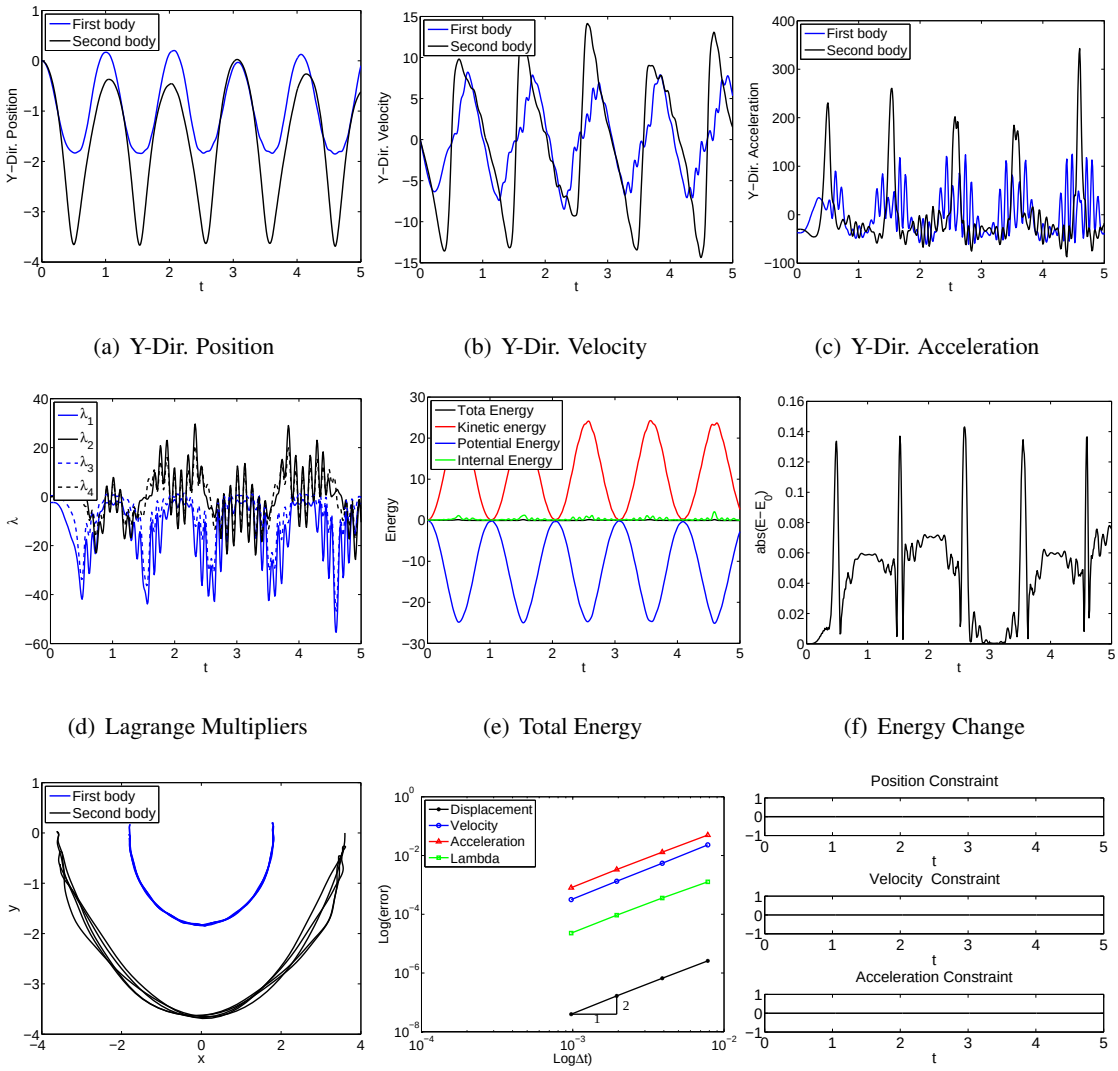


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.162: Double flexible pendulum with EB element in IRF: U0(1,1,0) - Index 3.

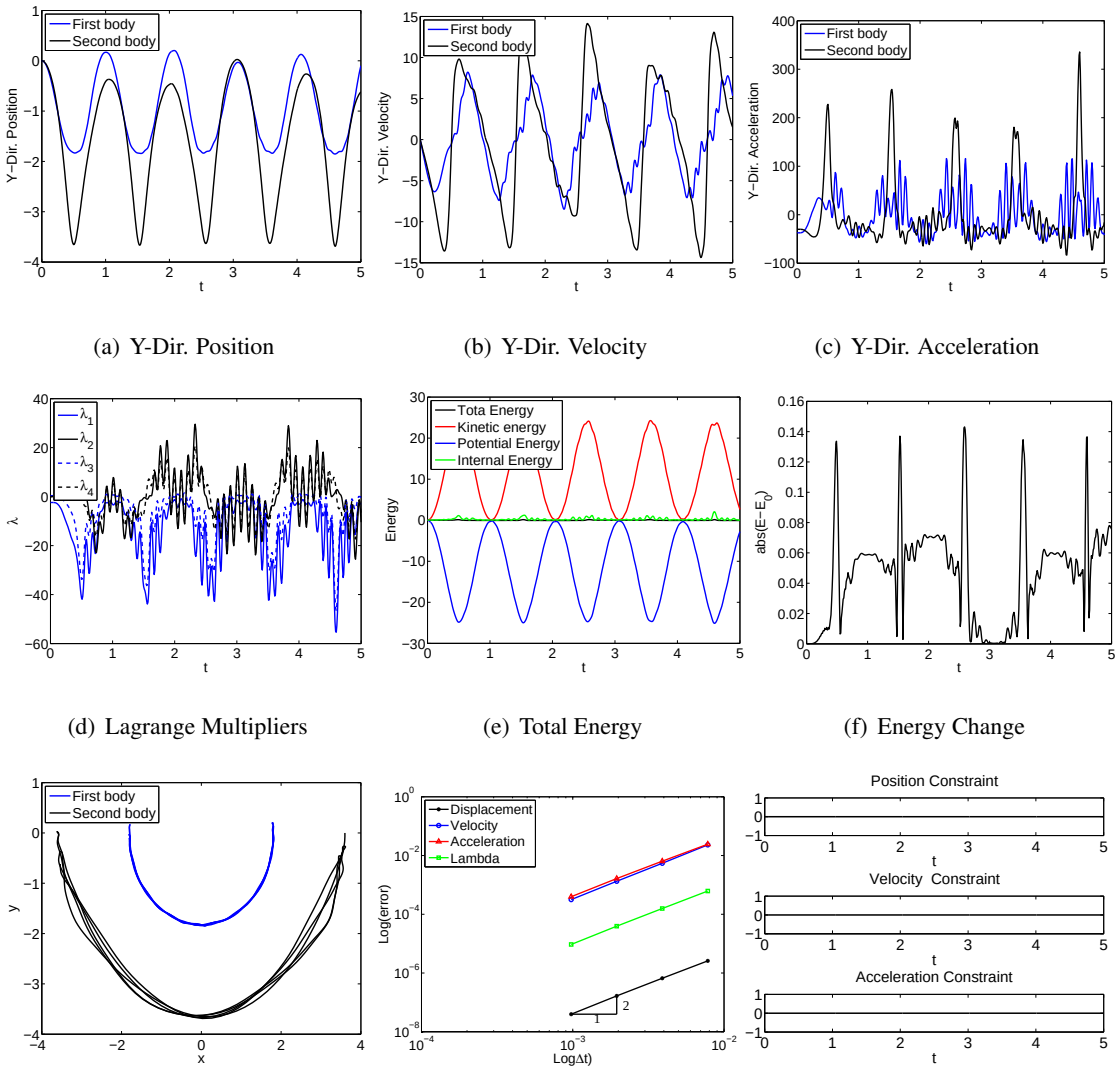


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.163: Double flexible pendulum with EB element in IRF: V0(1,1,0) - Index 3.

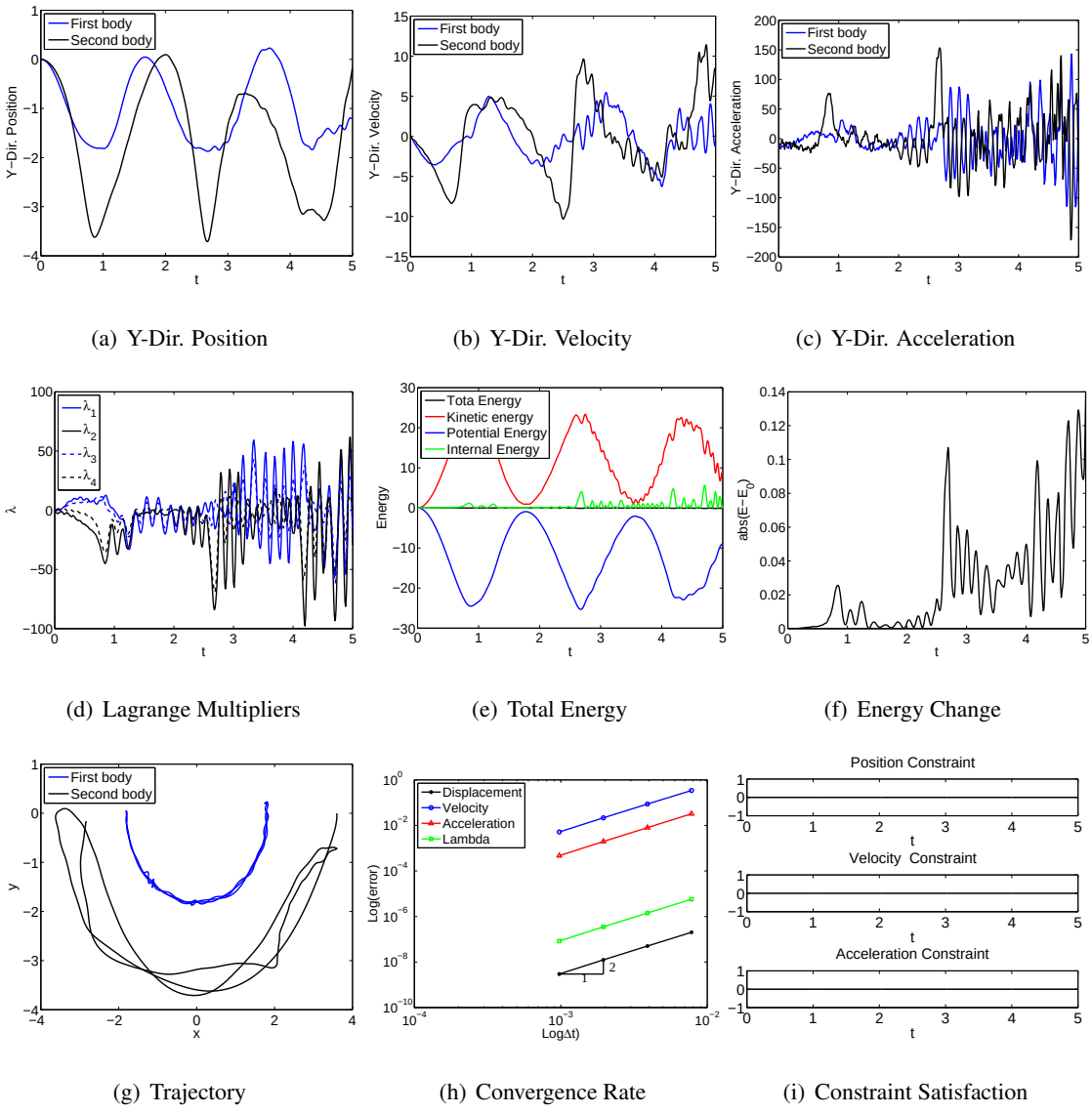


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.164: Double flexible pendulum with EB element in IRF: U0V0(1,1,1) - Index 3.

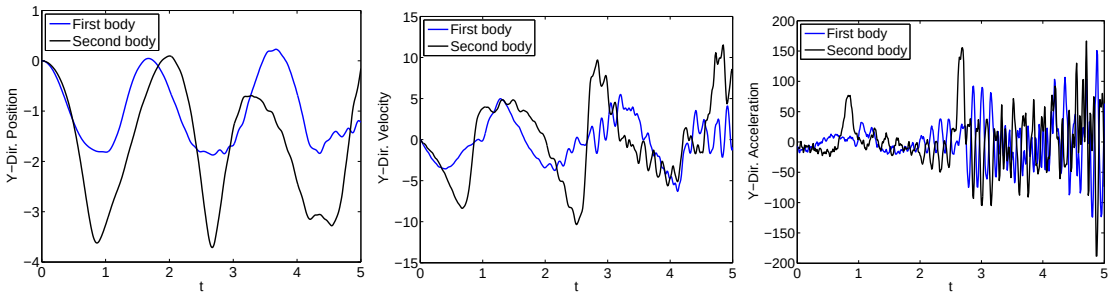


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

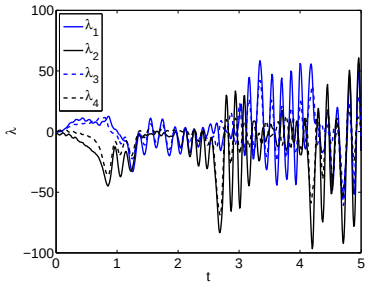
Figure 4.165: Double flexible pendulum with EB element in ANCF-S: U0(1,1,0) - Index 3.



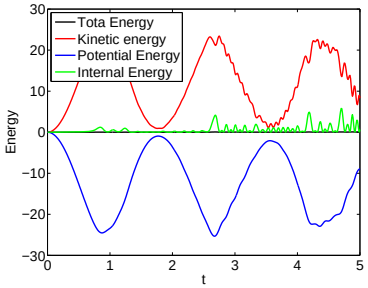
(a) Y-Dir. Position

(b) Y-Dir. Velocity

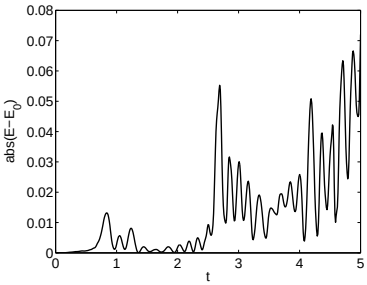
(c) Y-Dir. Acceleration



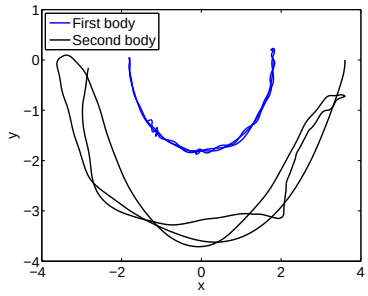
(d) Lagrange Multipliers



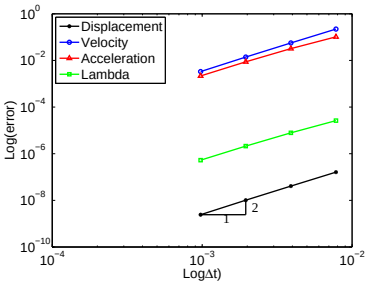
(e) Total Energy



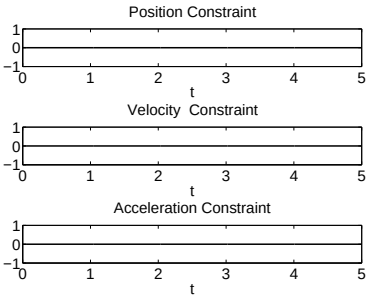
(f) Energy Change



(g) Trajectory

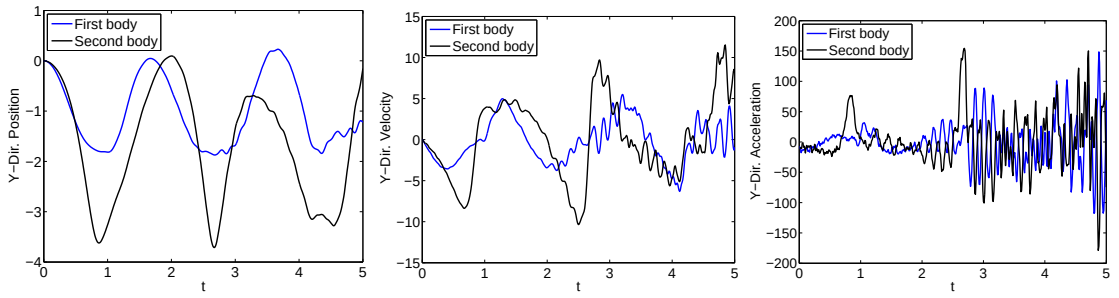


(h) Convergence Rate



(i) Constraint Satisfaction

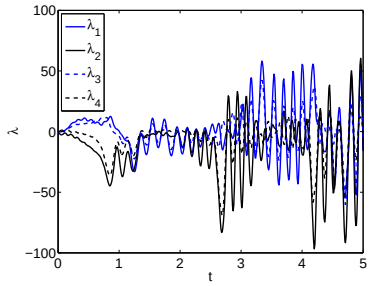
Figure 4.166: Double flexible pendulum with EB element in ANCF-S: V0(1,1,0) - Index 3.



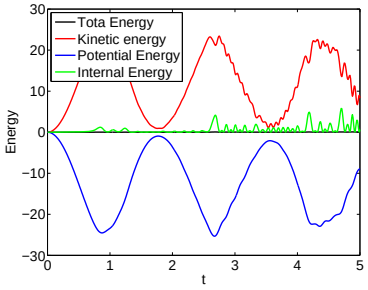
(a) Y-Dir. Position

(b) Y-Dir. Velocity

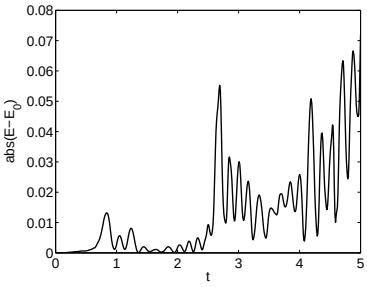
(c) Y-Dir. Acceleration



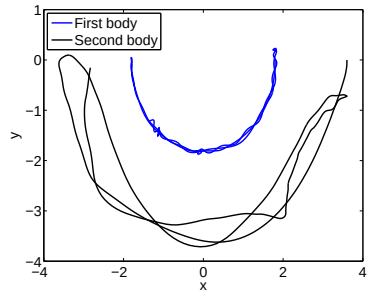
(d) Lagrange Multipliers



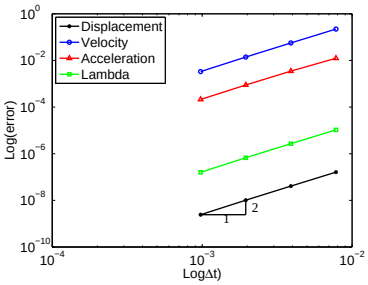
(e) Total Energy



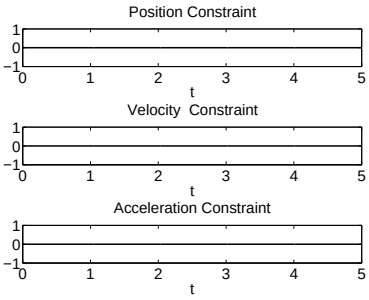
(f) Energy Change



(g) Trajectory

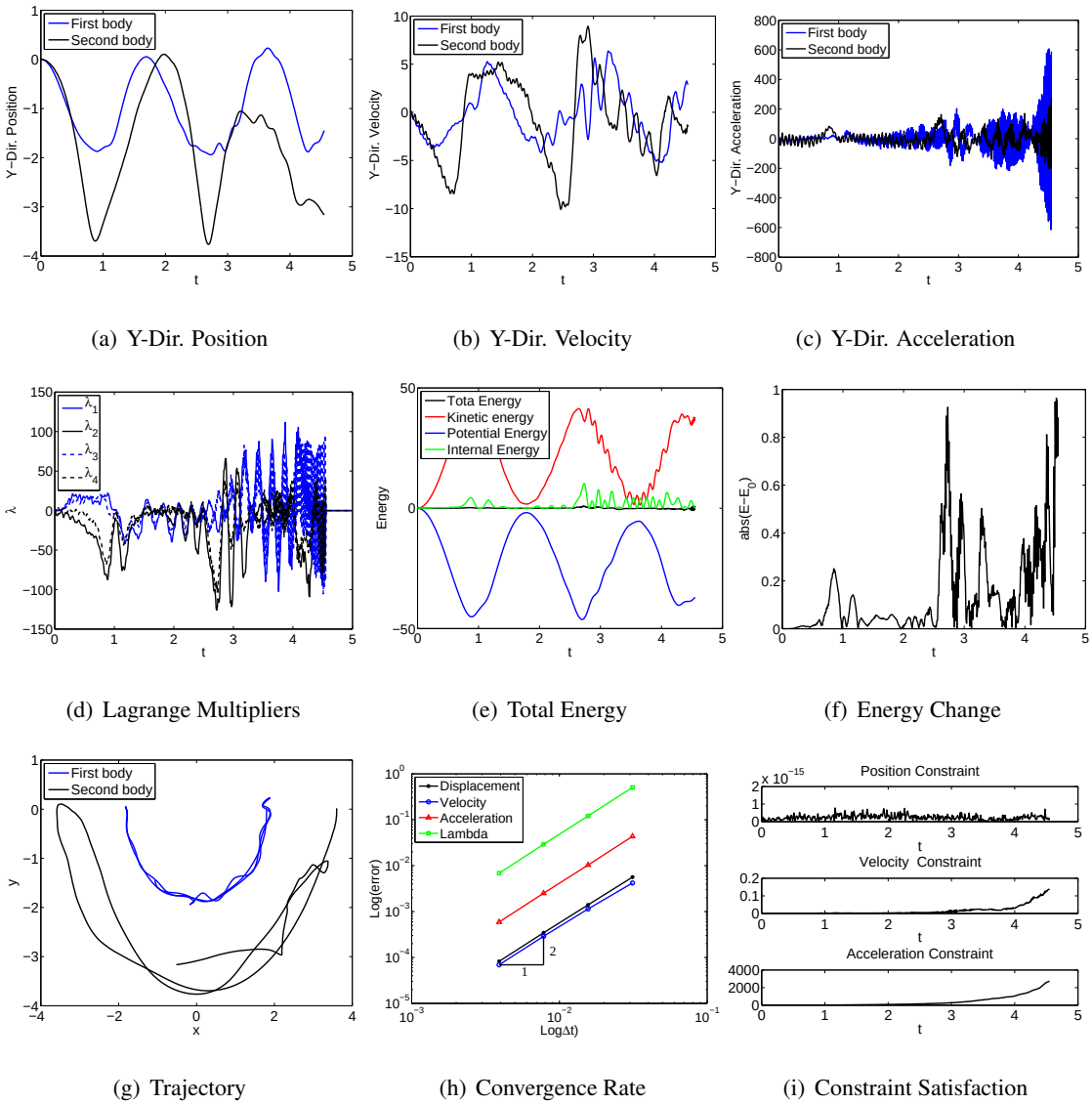


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.167: Double flexible pendulum with EB element in ANCF-S: U0V0(1,1,1) - Index 3.

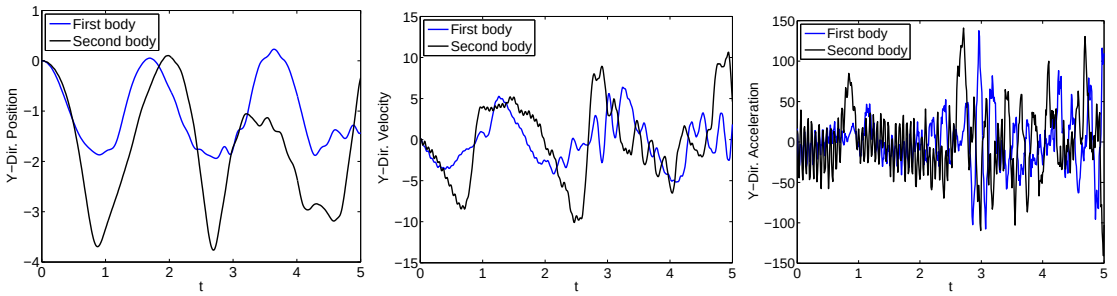


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

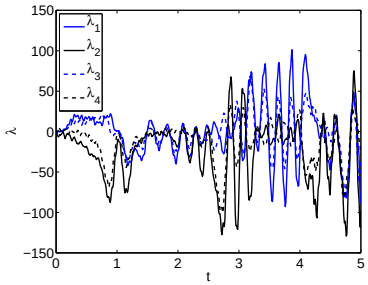
Figure 4.168: Double flexible pendulum with EB element in FRF: U0(1,1,0) - Index 3.



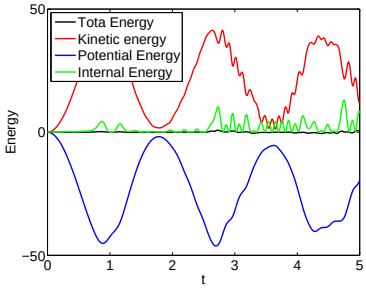
(a) Y-Dir. Position

(b) Y-Dir. Velocity

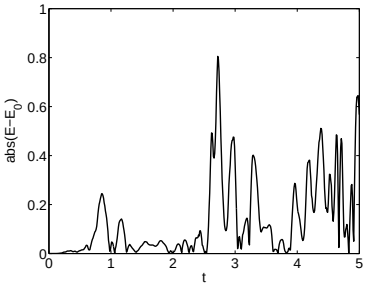
(c) Y-Dir. Acceleration



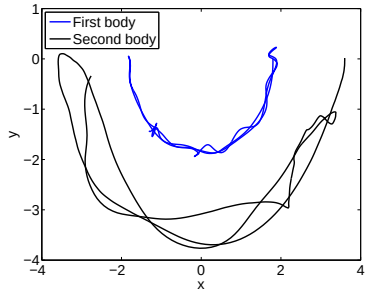
(d) Lagrange Multipliers



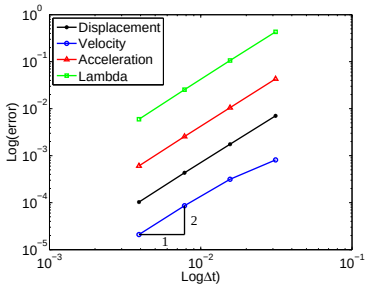
(e) Total Energy



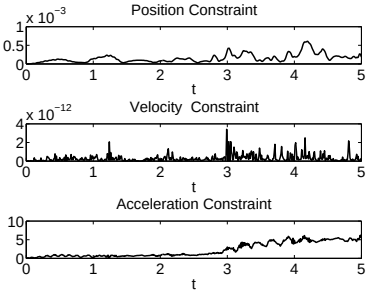
(f) Energy Change



(g) Trajectory

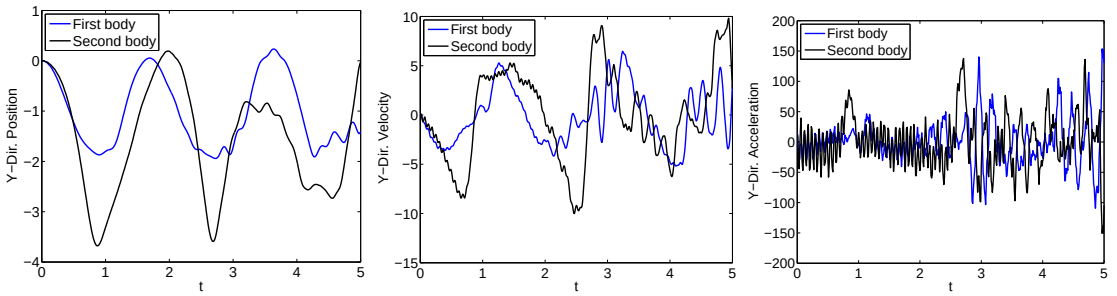


(h) Convergence Rate



(i) Constraint Satisfaction

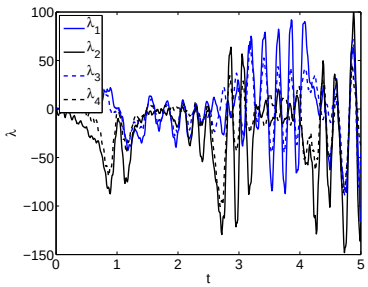
Figure 4.169: Double flexible pendulum with EB element in FRF: U0(1,1,0) - Index 2.



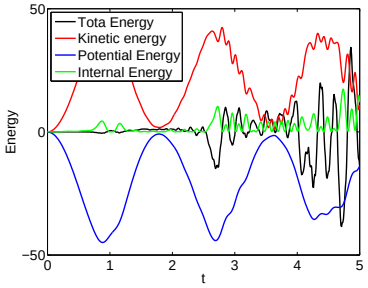
(a) Y-Dir. Position

(b) Y-Dir. Velocity

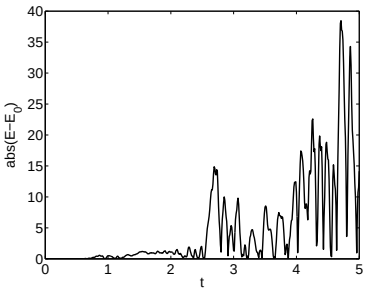
(c) Y-Dir. Acceleration



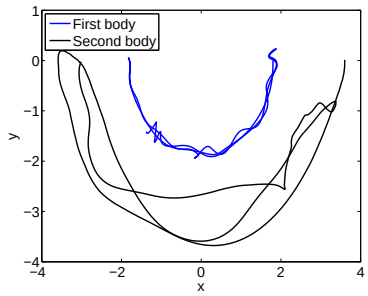
(d) Lagrange Multipliers



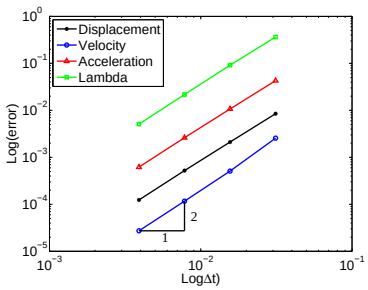
(e) Total Energy



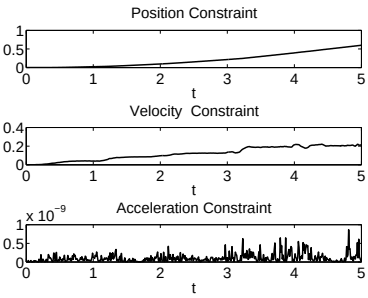
(f) Energy Change



(g) Trajectory

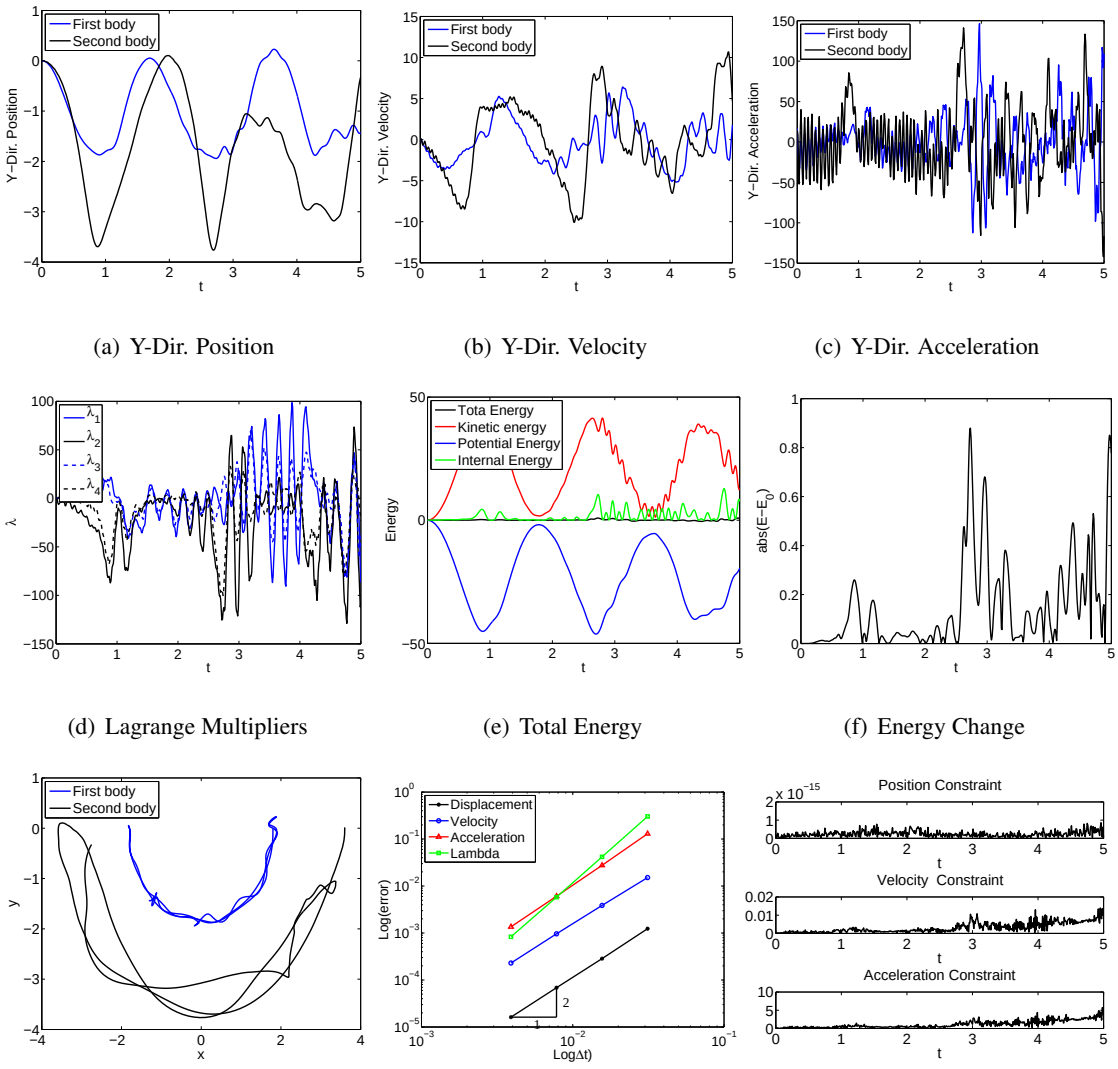


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.170: Double flexible pendulum with EB element in FRF: U0(1,1,0) - Index 1.

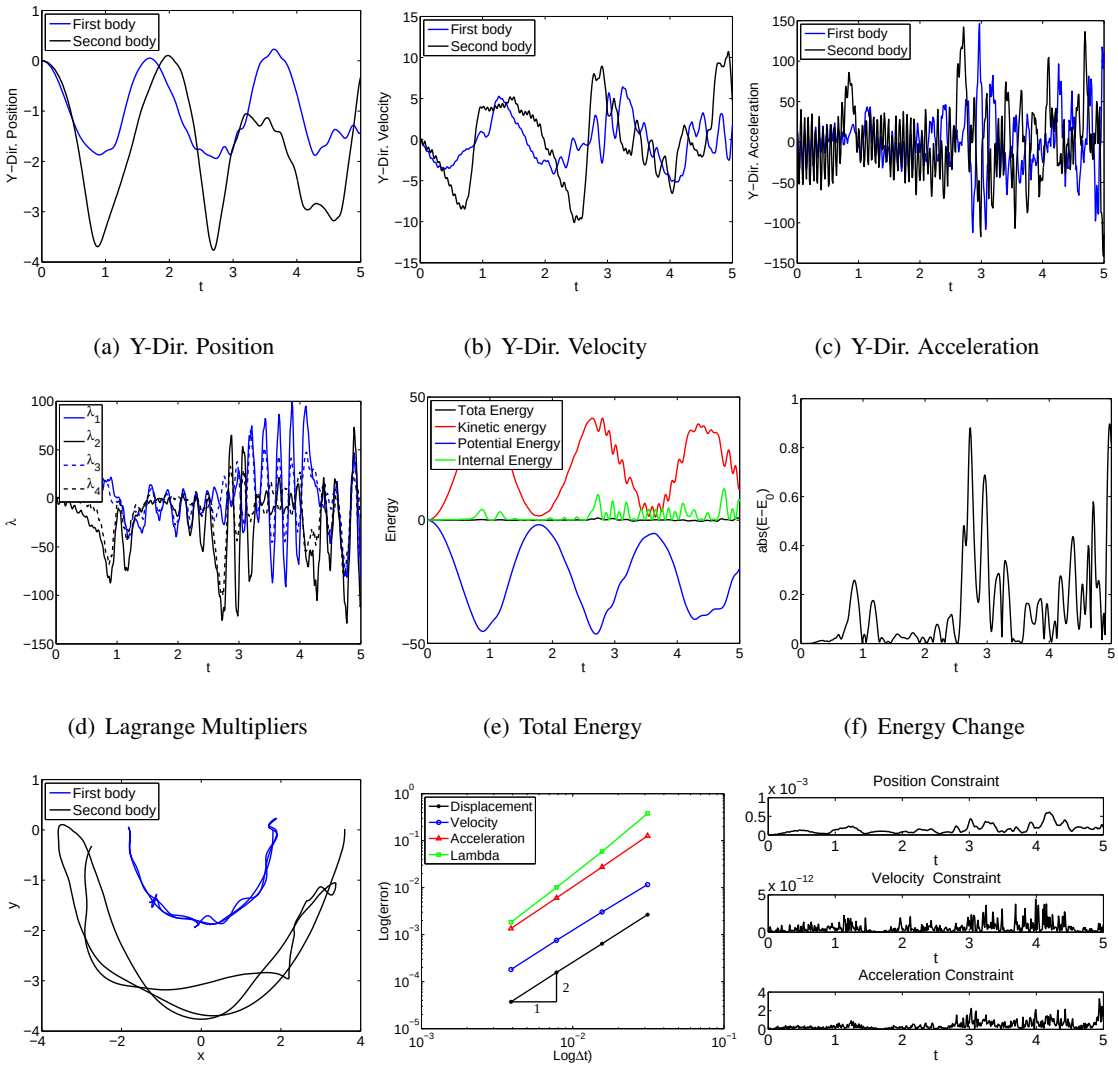


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.171: Double flexible pendulum with EB element in FRF: V0(1,1,0) - Index 3.

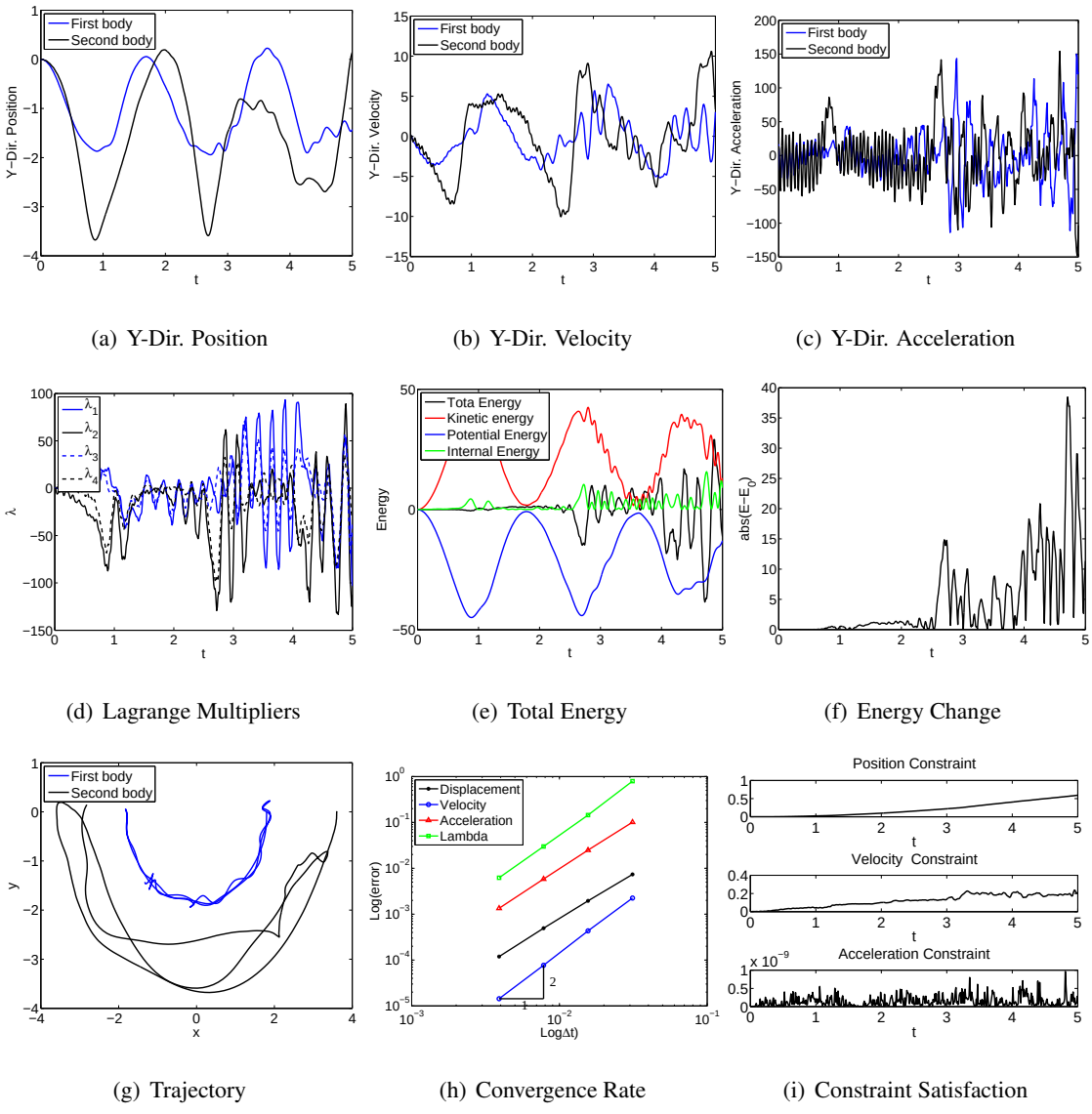


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.172: Double flexible pendulum with EB element in FRF: V0(1,1,0) - Index 2.

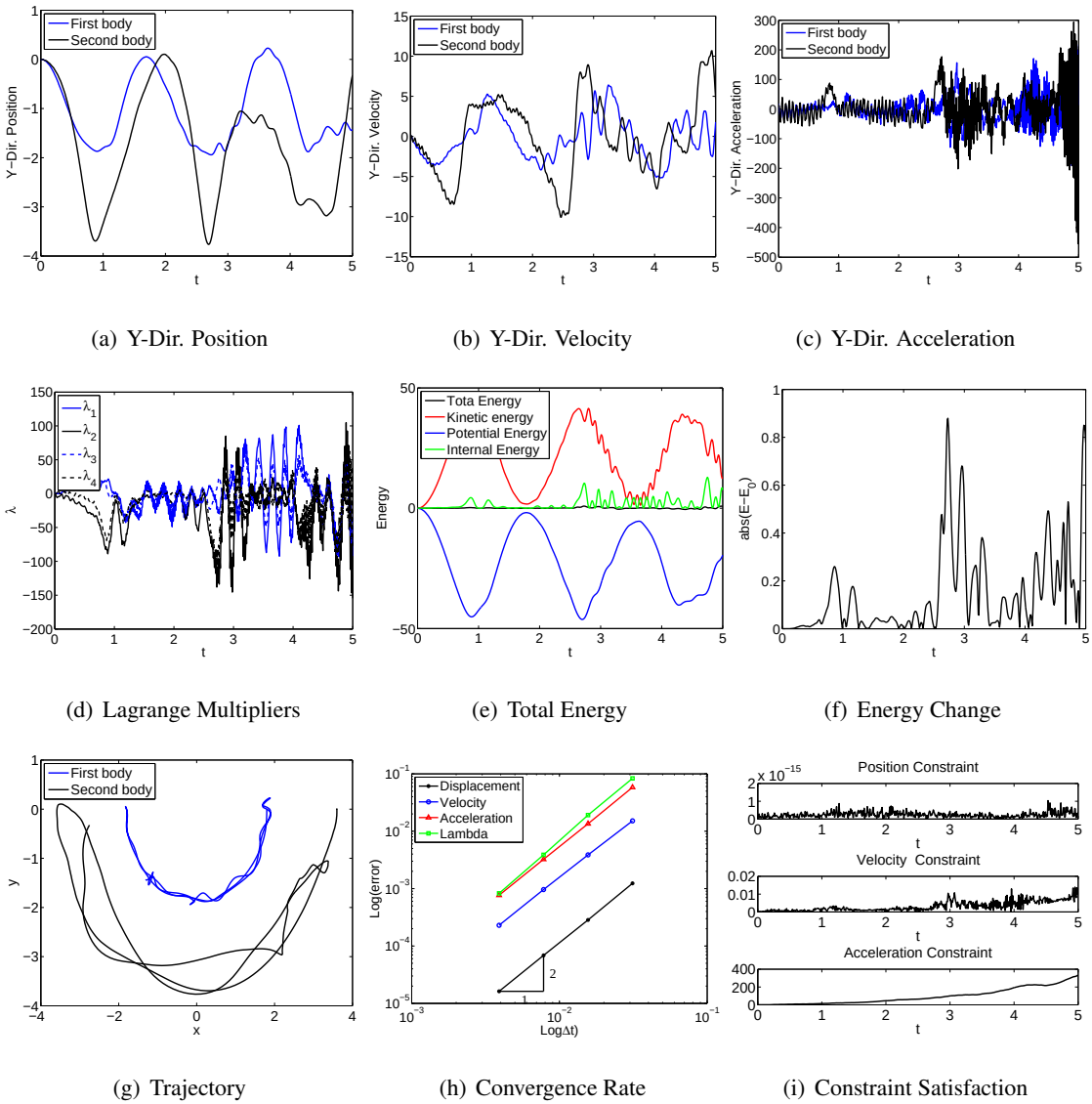


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.173: Double flexible pendulum with EB element in FRF: V0(1,1,0) - Index 1.

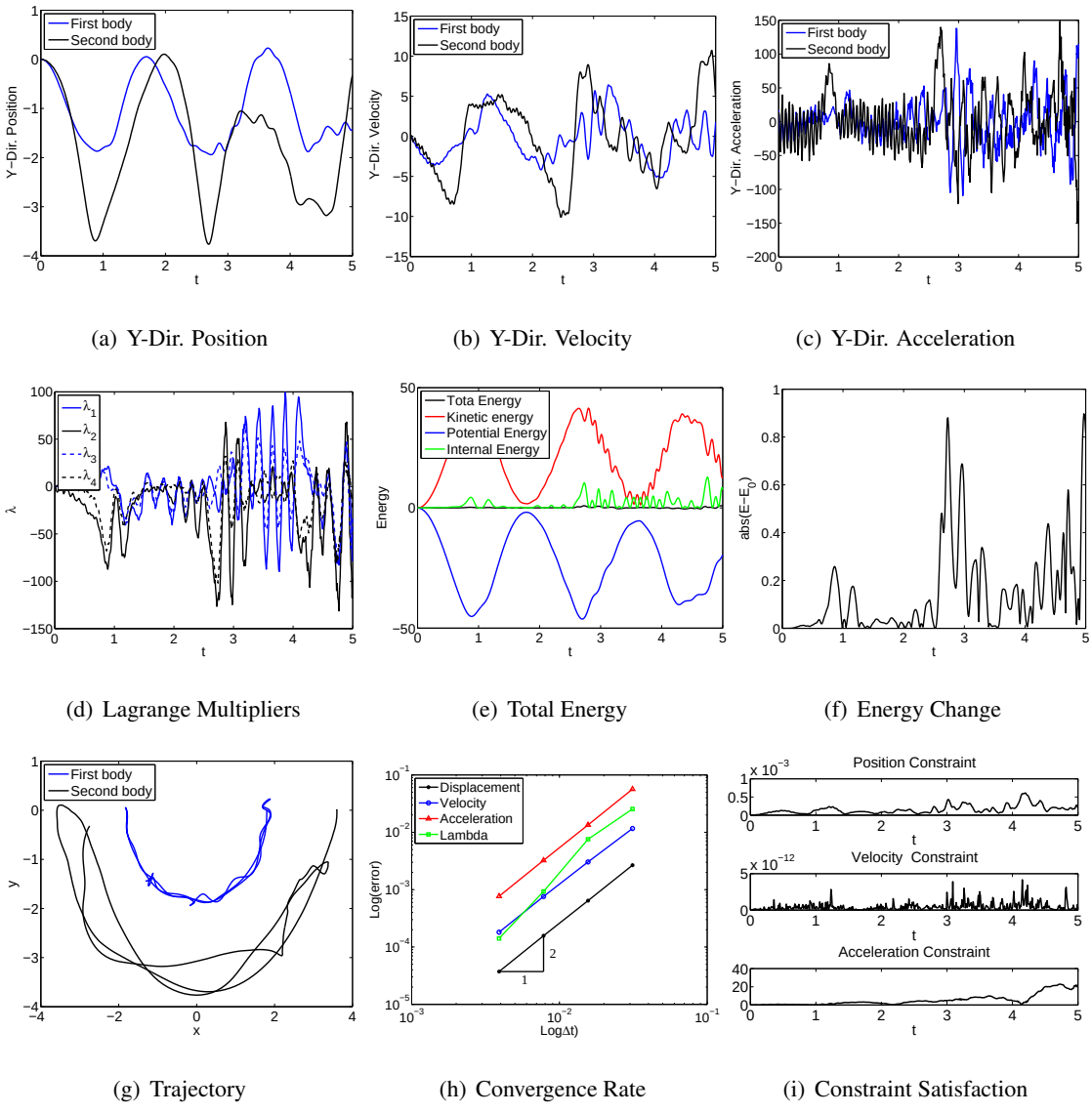


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.174: Double flexible pendulum with EB element in FRF: U0V0(1,1,1) - Index 3.

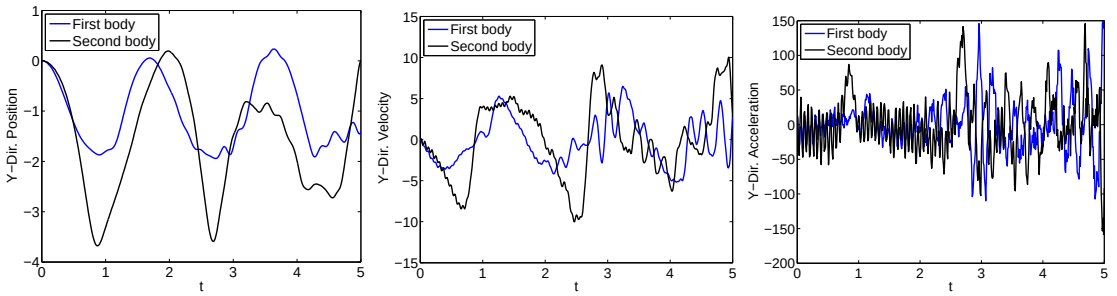


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

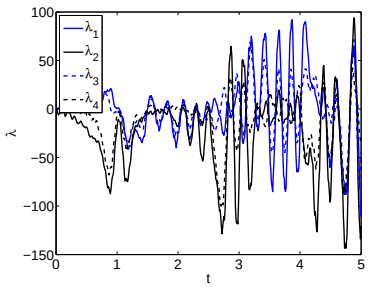
Figure 4.175: Double flexible pendulum with EB element in FRF: U0V0(1,1,1) - Index 2.



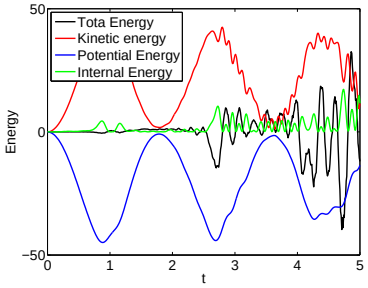
(a) Y-Dir. Position

(b) Y-Dir. Velocity

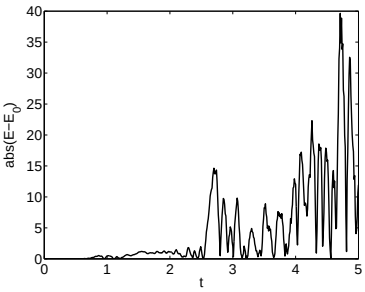
(c) Y-Dir. Acceleration



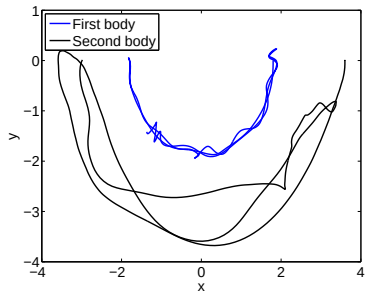
(d) Lagrange Multipliers



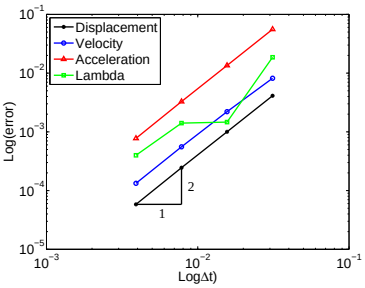
(e) Total Energy



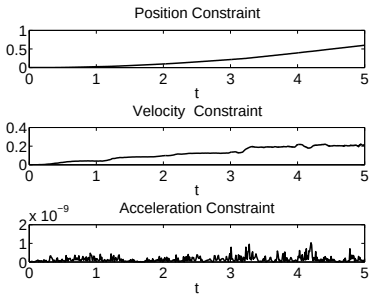
(f) Energy Change



(g) Trajectory



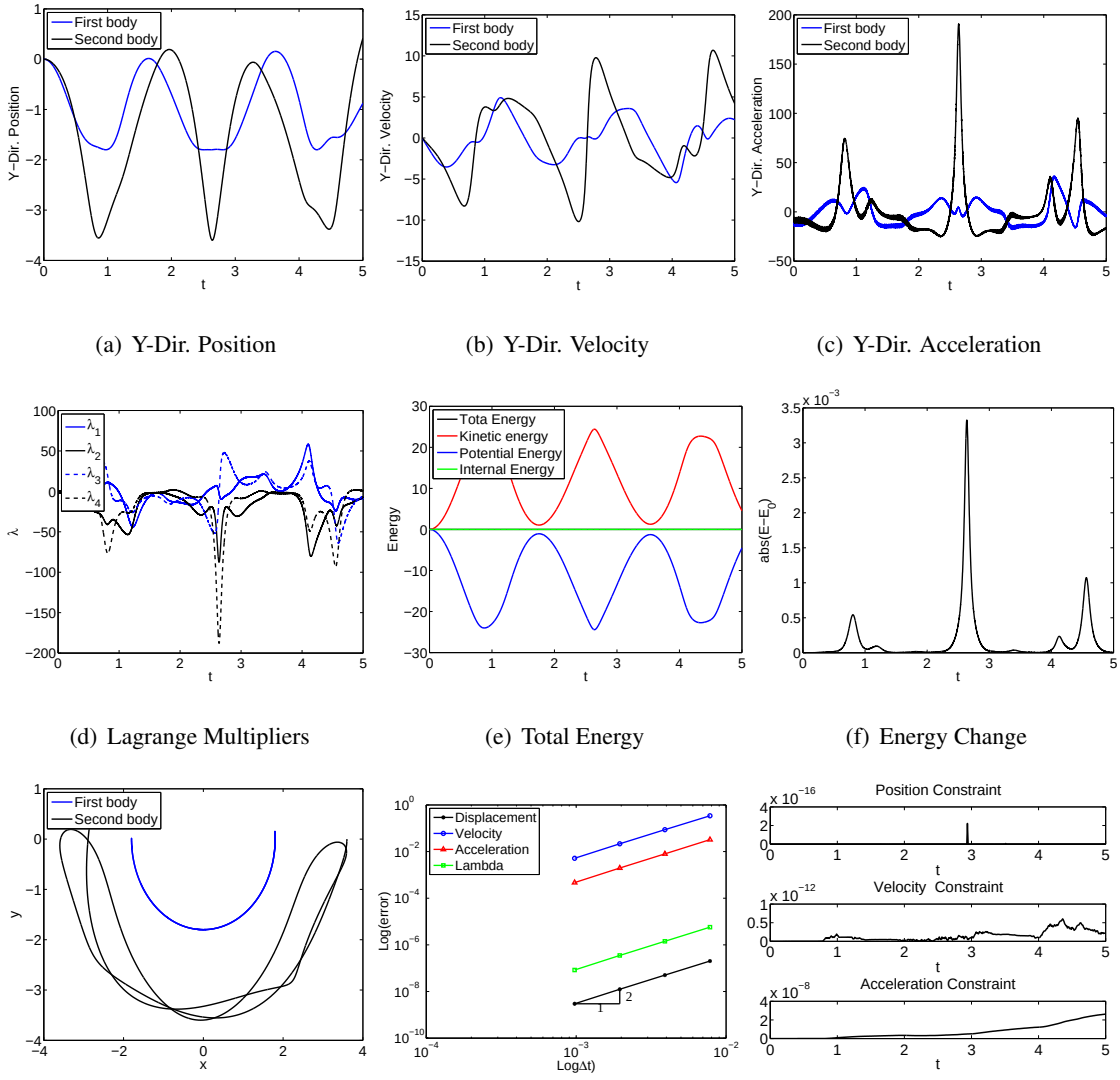
(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.176: Double flexible pendulum with EB element in FRF: U0V0(1,1,1) - Index 1.

Euler-Bernoulli Beam: Quasi-Rigid Case

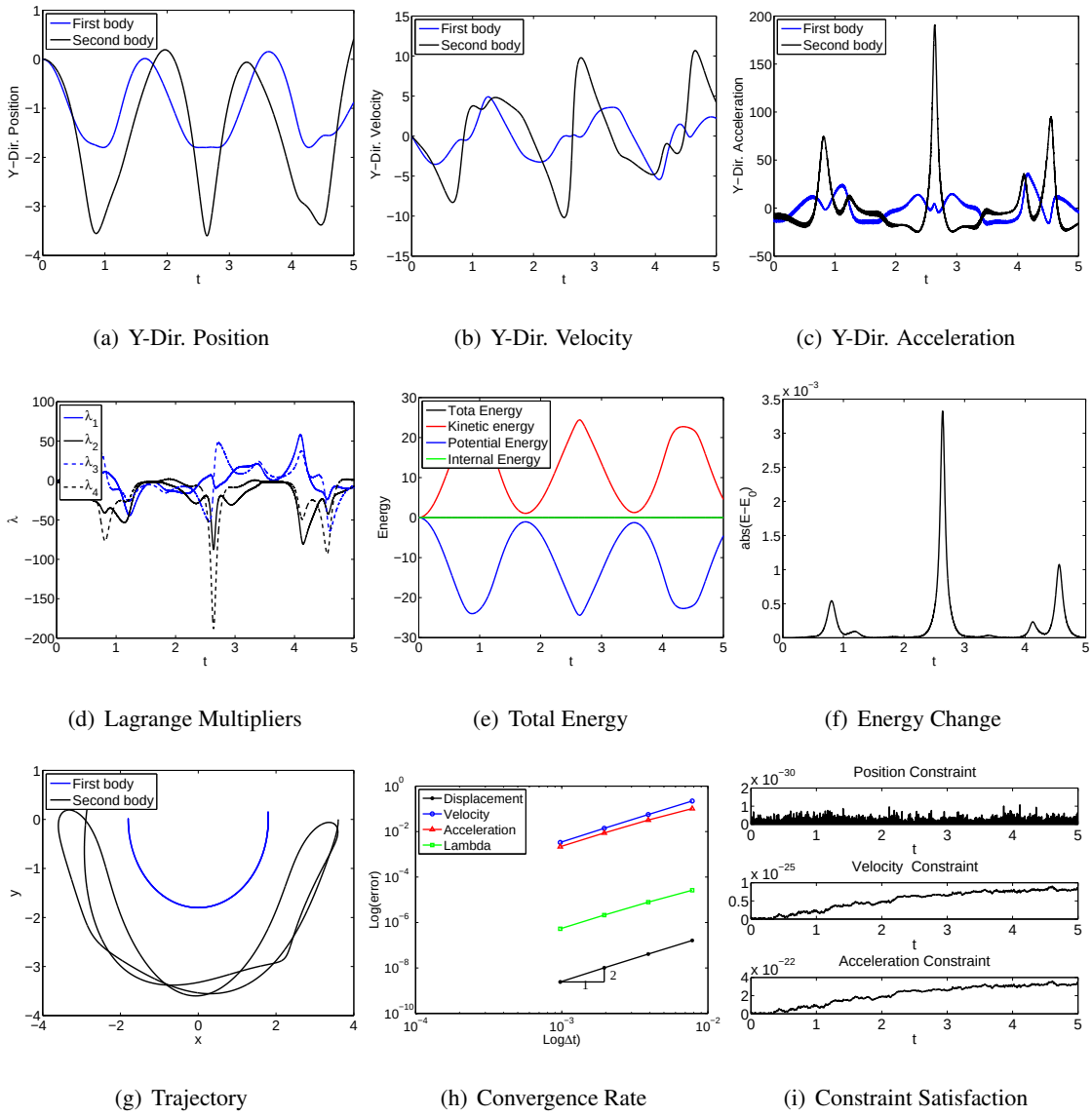


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.177: Double quasi-rigid pendulum with EB element in ANCF-S: U0(1,1,0) - Index 3.

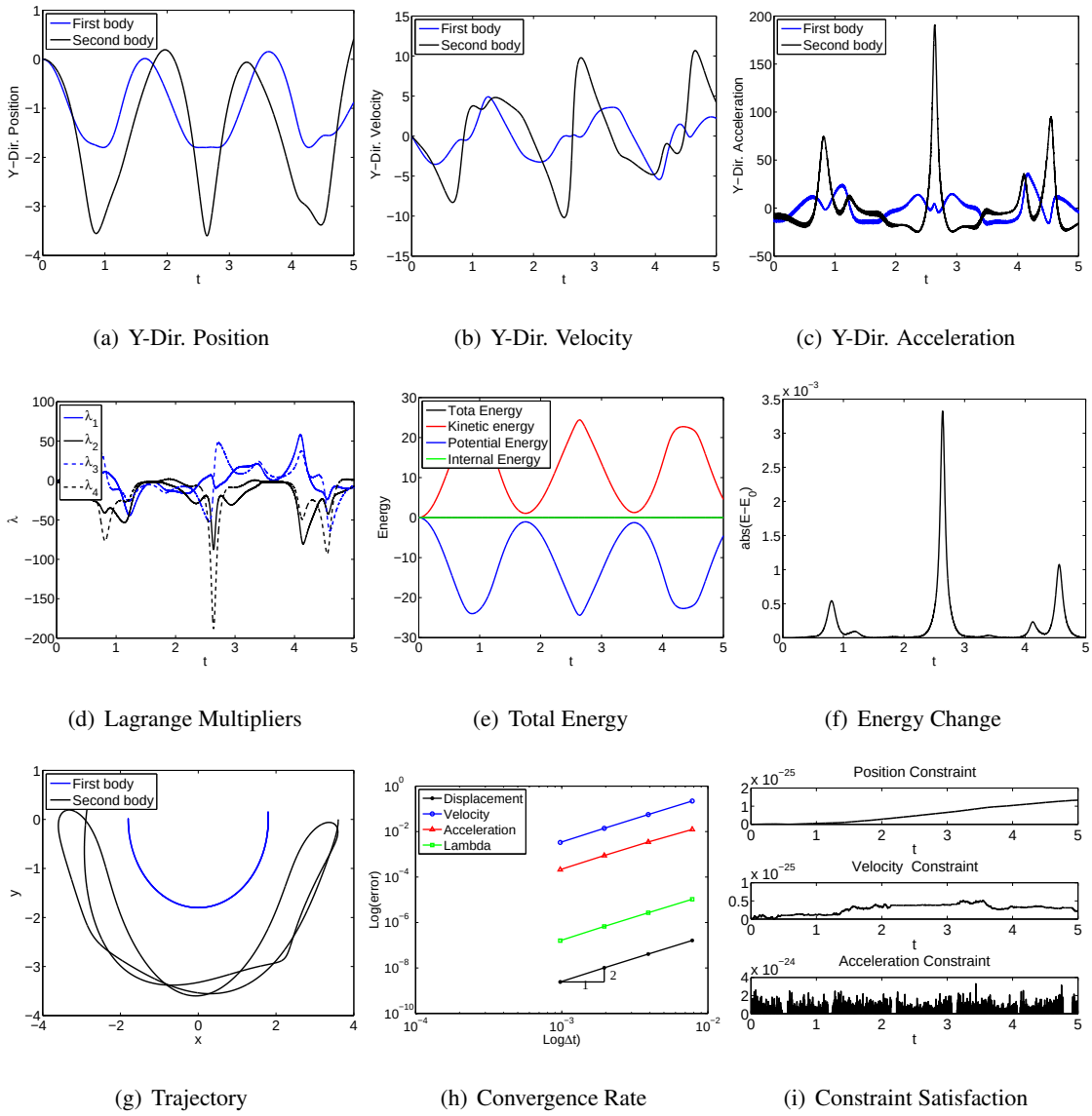


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.178: Double quasi-rigid pendulum with EB element in ANCF-S: V0(1,1,0) - Index 3.

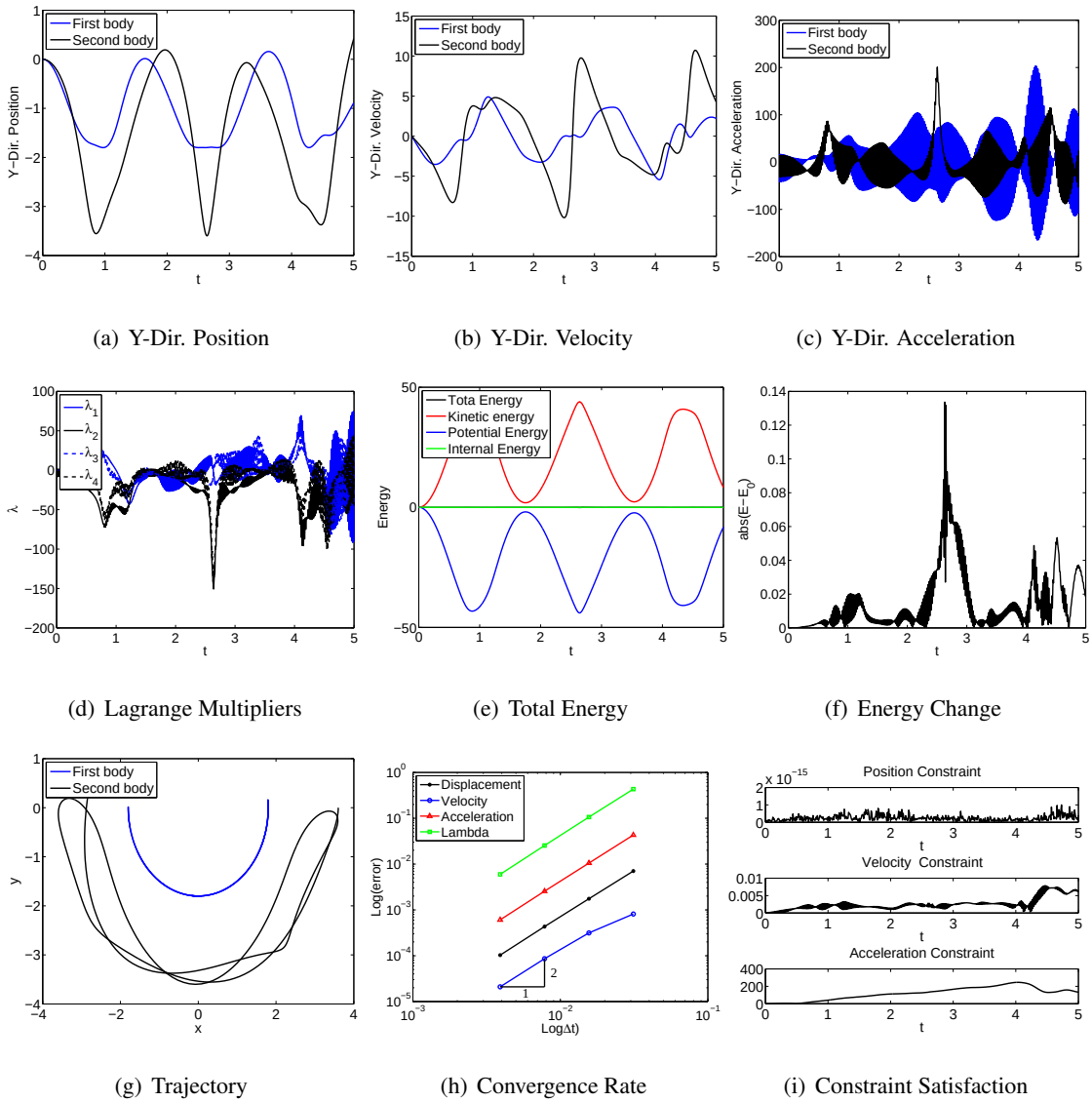


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.179: Double quasi-rigid pendulum with EB element in ANCF-S: U0V0(1,1,1) - Index 3.

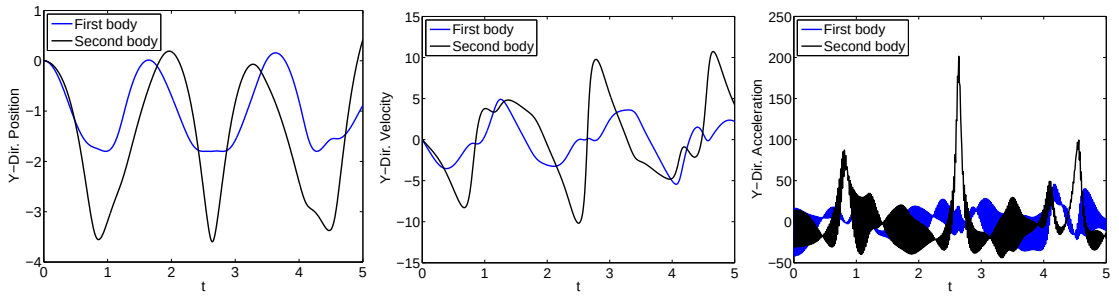


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

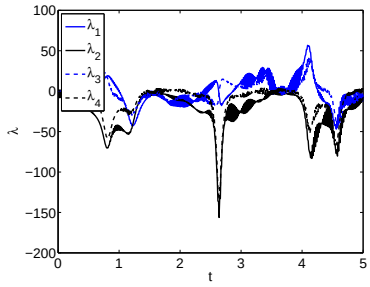
Figure 4.180: Double quasi-rigid pendulum with EB element in FRF: U0(1,1,0) - Index 3.



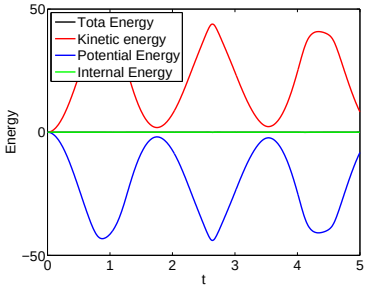
(a) Y-Dir. Position

(b) Y-Dir. Velocity

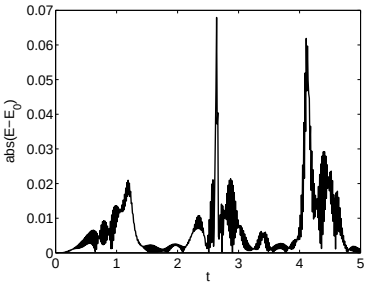
(c) Y-Dir. Acceleration



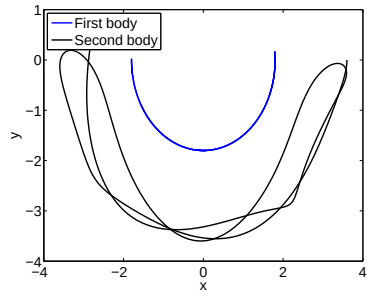
(d) Lagrange Multipliers



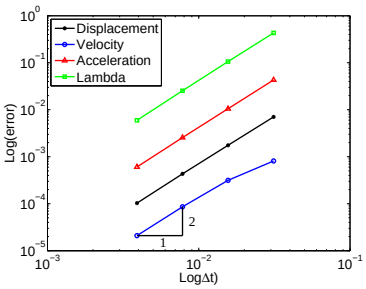
(e) Total Energy



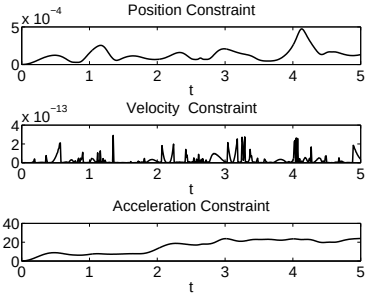
(f) Energy Change



(g) Trajectory

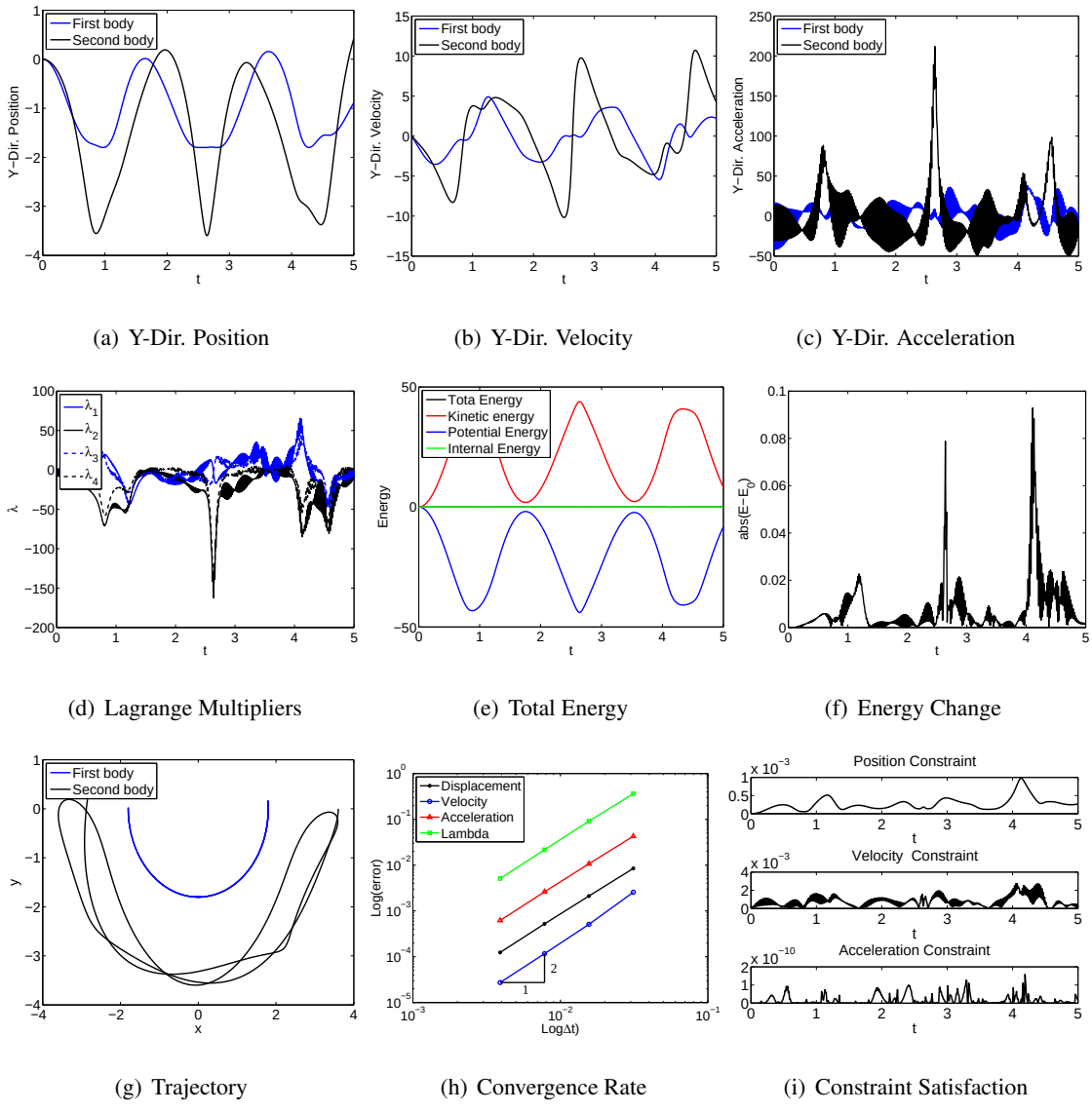


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.181: Double quasi-rigid pendulum with EB element in FRF: U0(1,1,0) - Index 2.

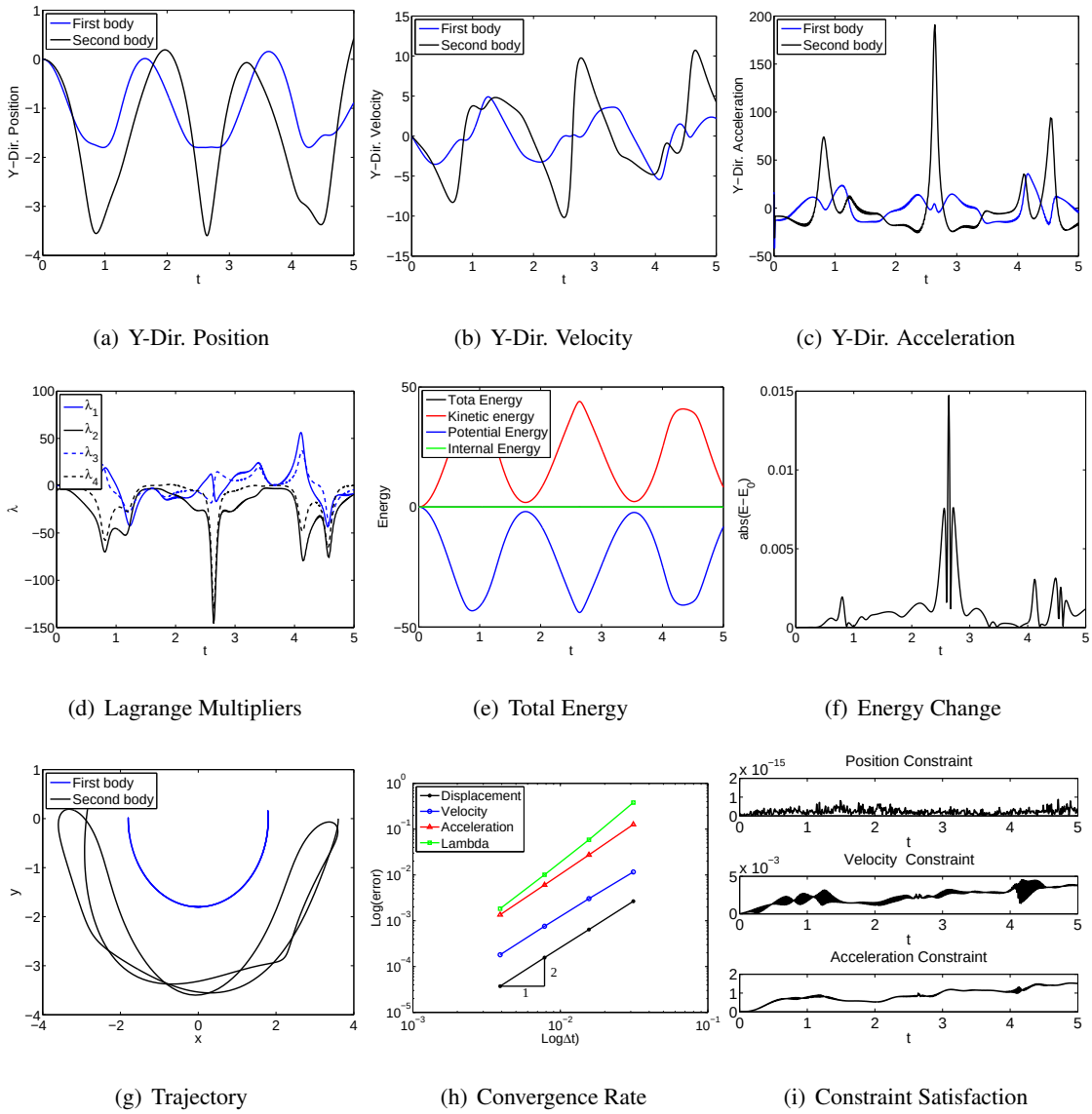


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.182: Double quasi-rigid pendulum with EB element in FRF: U0(1,1,0) - Index 1.

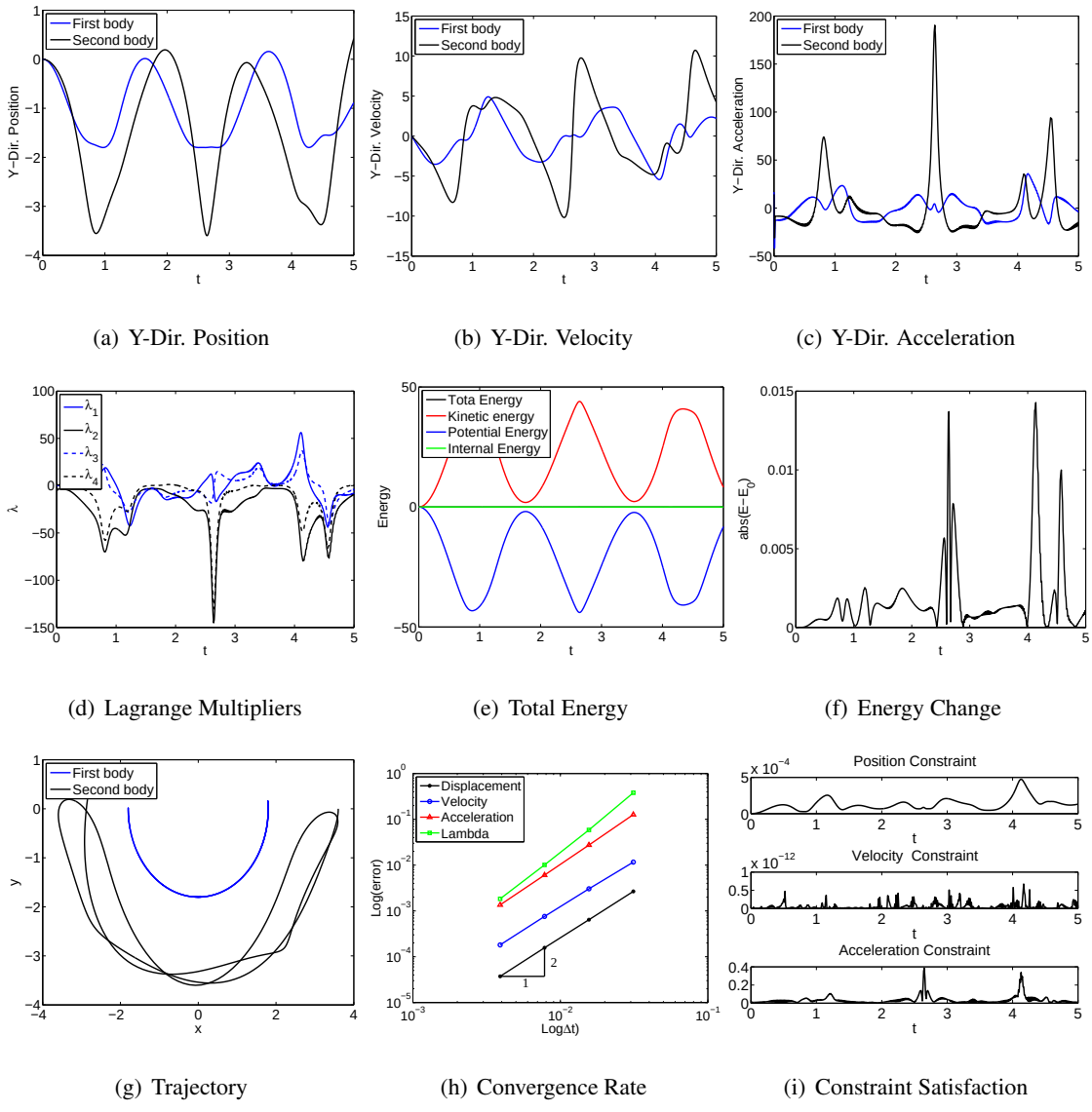


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.183: Double quasi-rigid pendulum with EB element in FRF: V0(1,1,0) - Index 3.

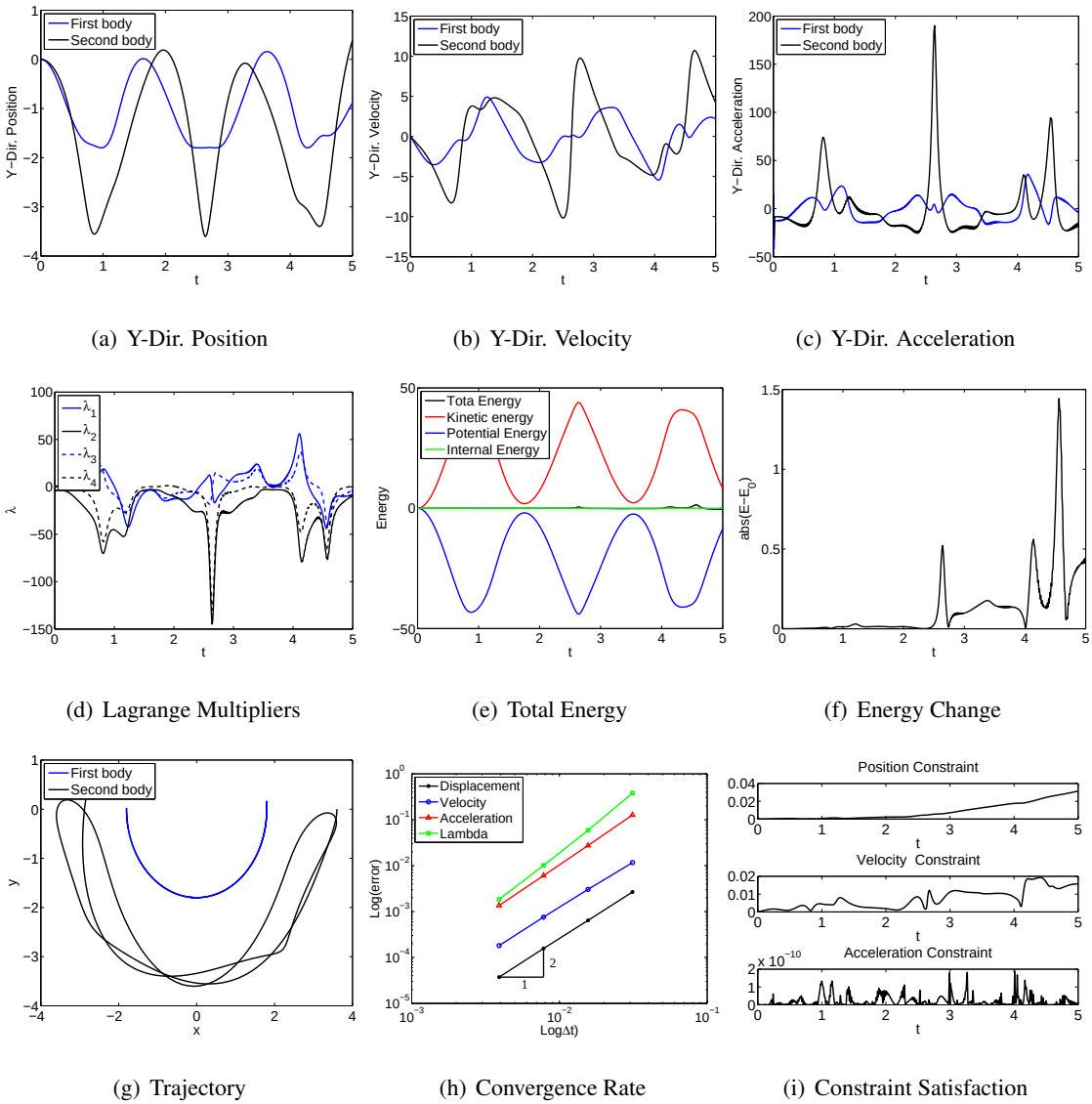


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.184: Double quasi-rigid pendulum with EB element in FRF: V0(1,1,0) - Index 2.

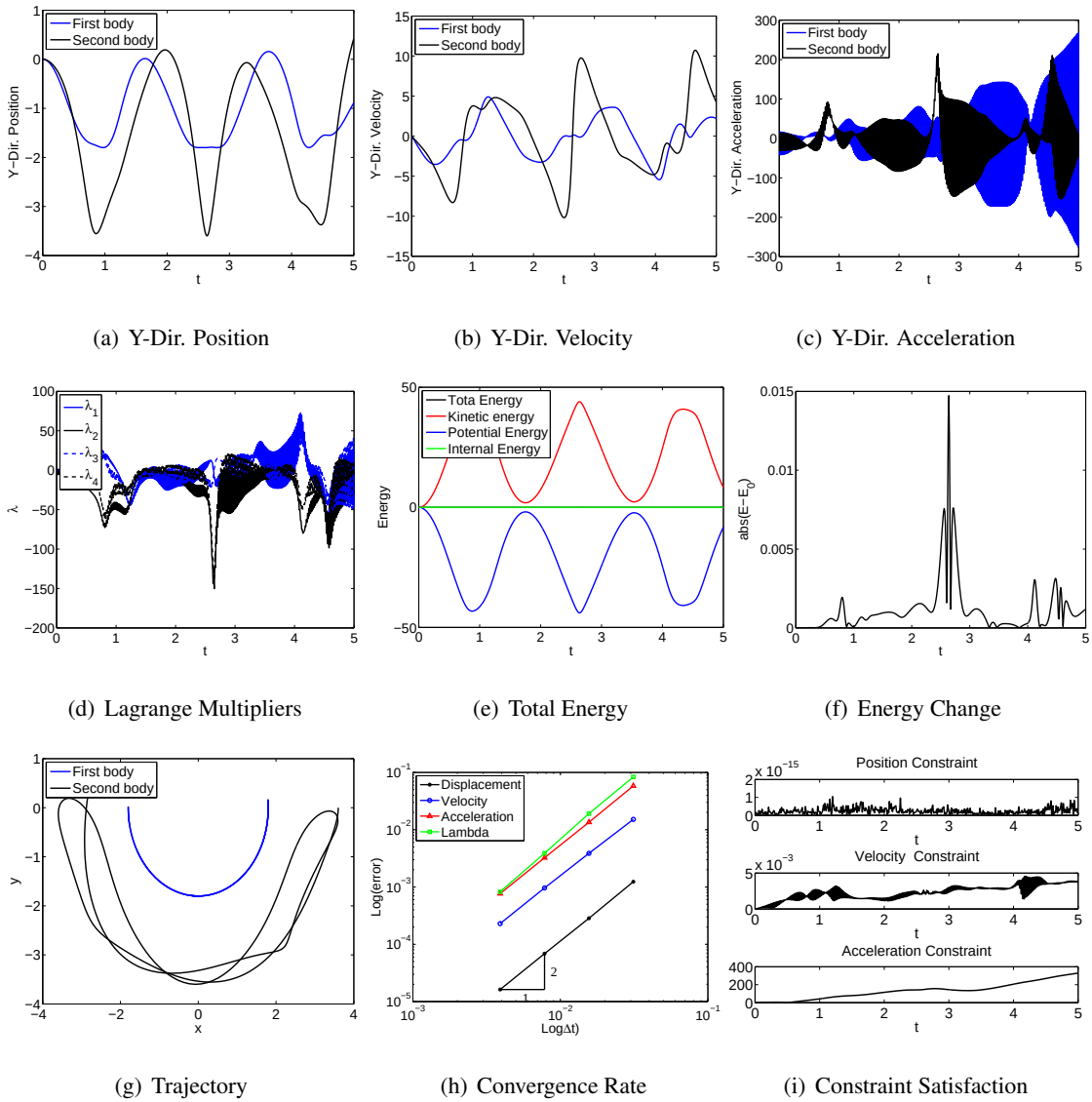


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.185: Double quasi-rigid pendulum with EB element in FRF: V0(1,1,0) - Index 1.

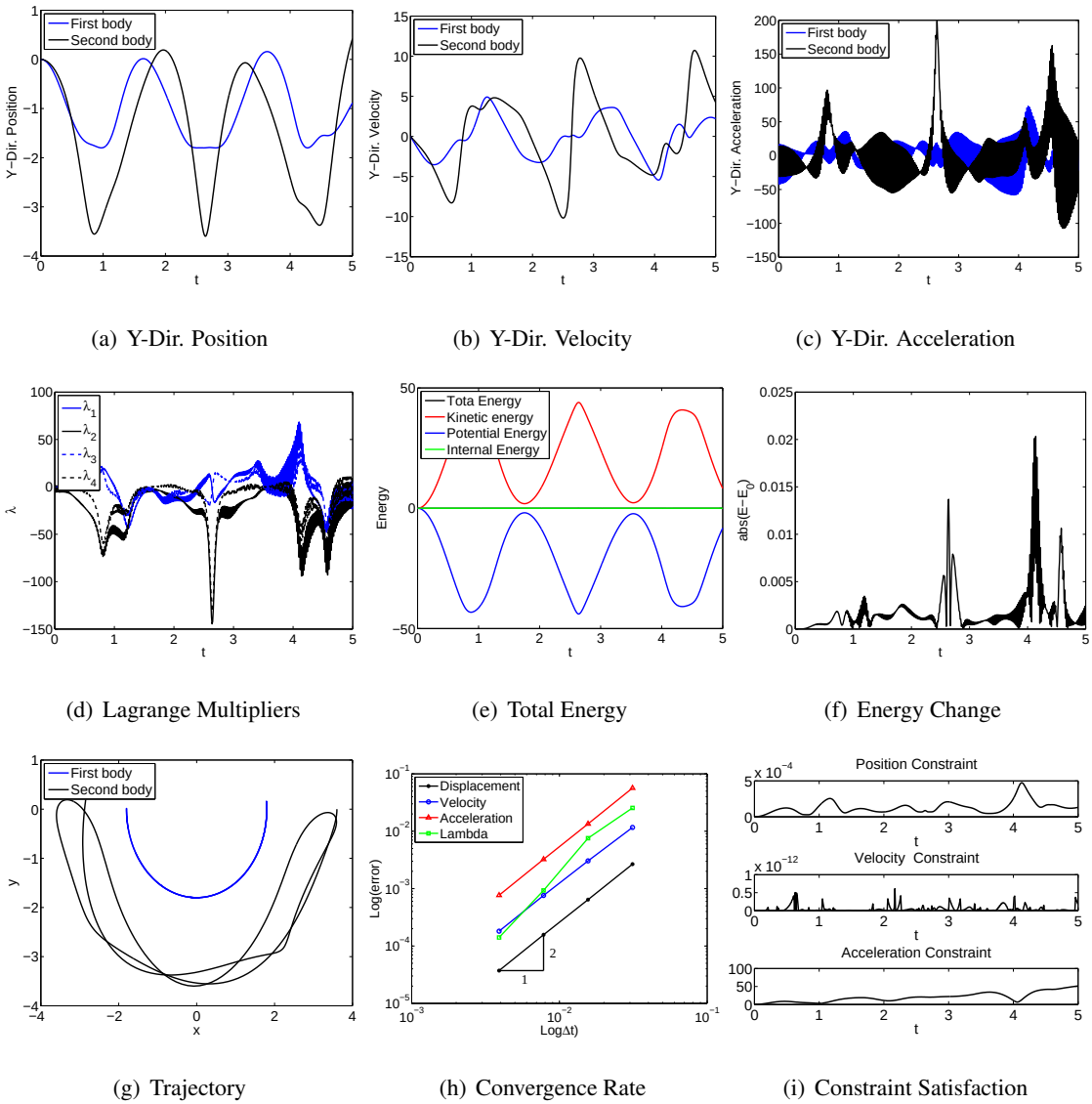


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.186: Double quasi-rigid pendulum with EB element in FRF: U0V0(1,1,1) - Index 3.

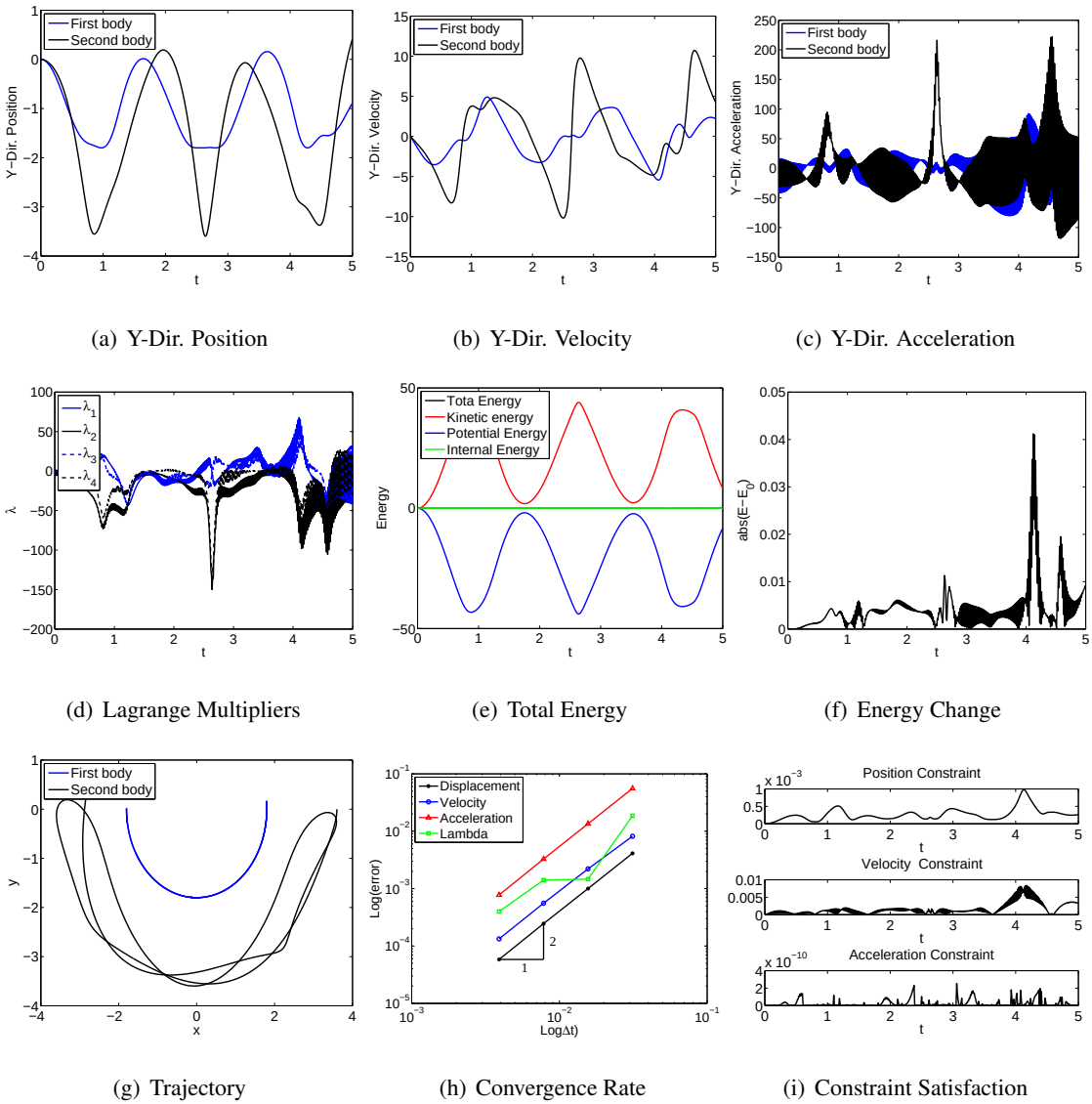


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.187: Double quasi-rigid pendulum with EB element in FRF: U0V0(1,1,1) - Index 2.



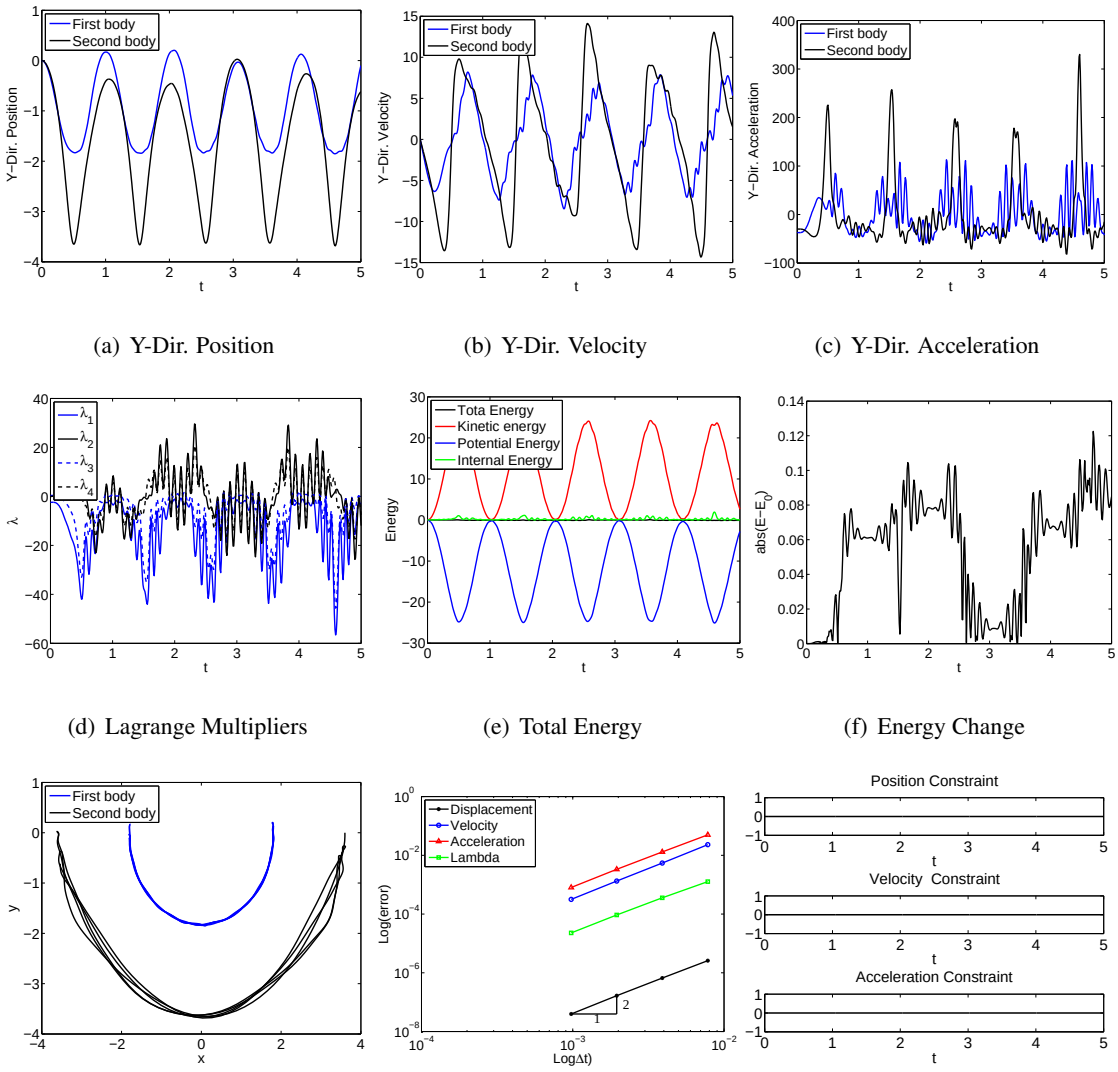
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.188: Double quasi-rigid pendulum with EB element in FRF: U0V0(1,1,1) - Index 1.

Timoshenko Beam

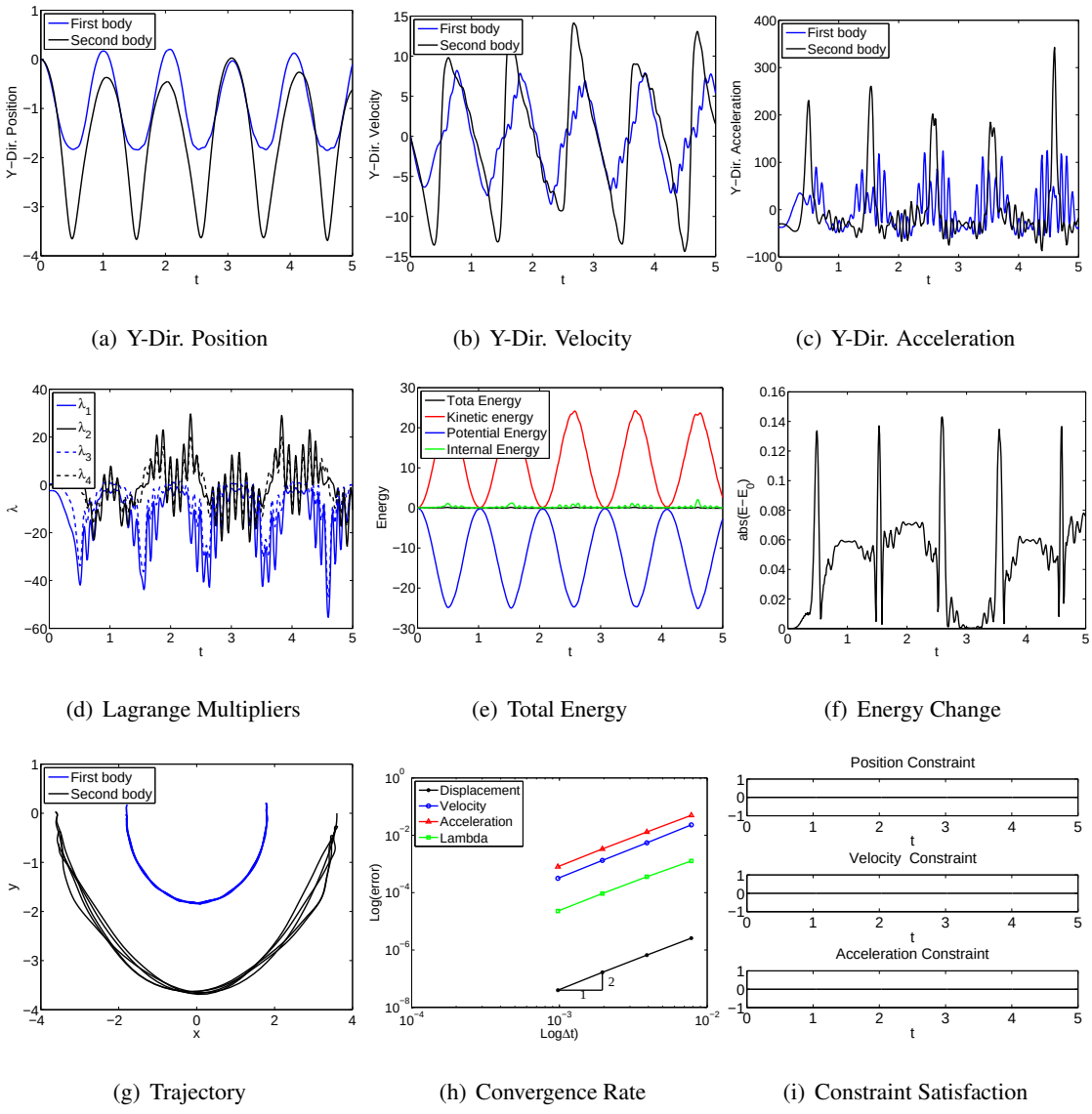


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.189: Double flexible pendulum with TB element in IRF: U0(1,1,0) - Index 3.

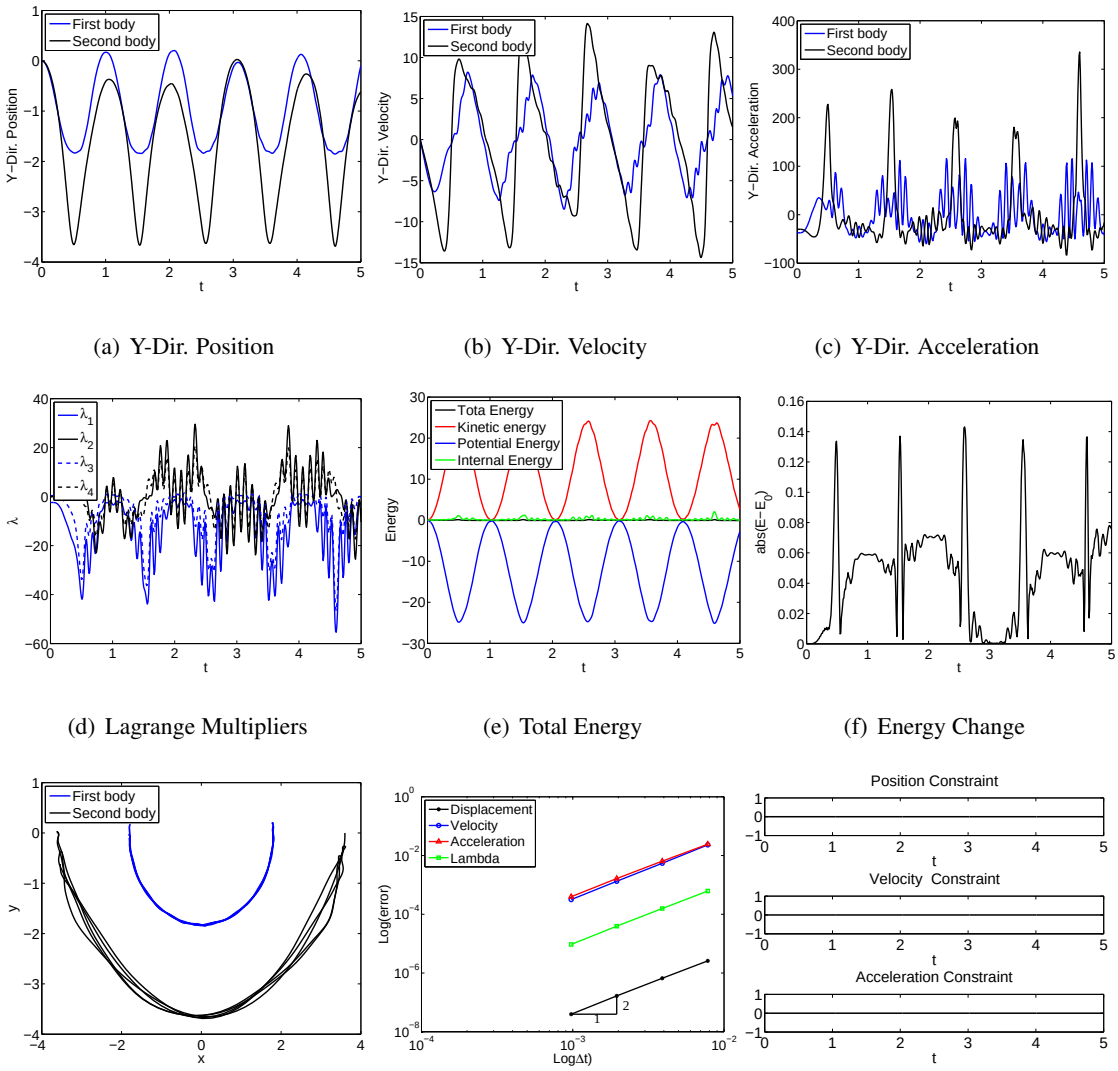


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.190: Double flexible pendulum with TB element in IRF: V0(1,1,0) - Index 3.

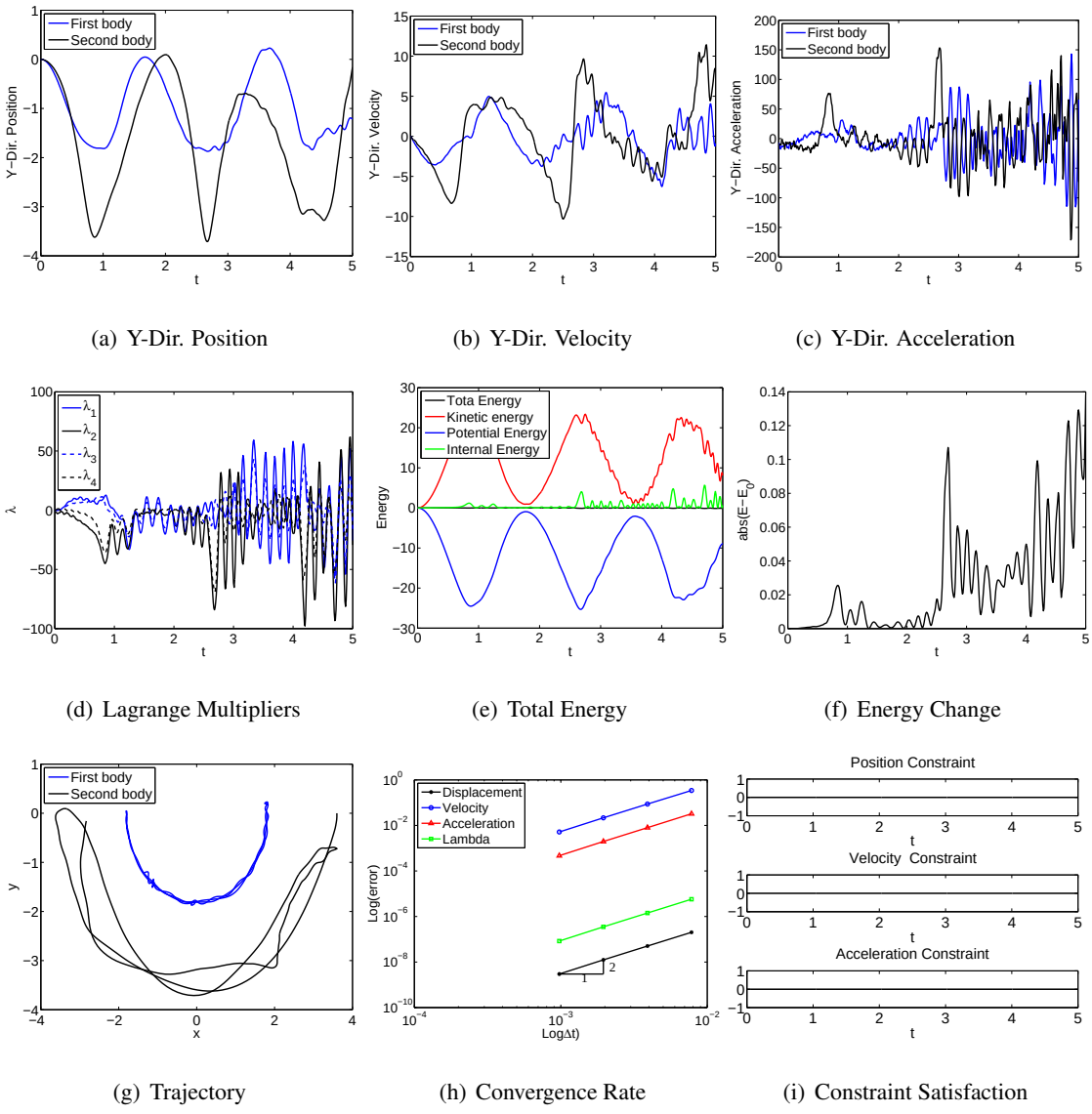


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.191: Double flexible pendulum with TB element in IRF: U0V0(1,1,1) - Index 3.

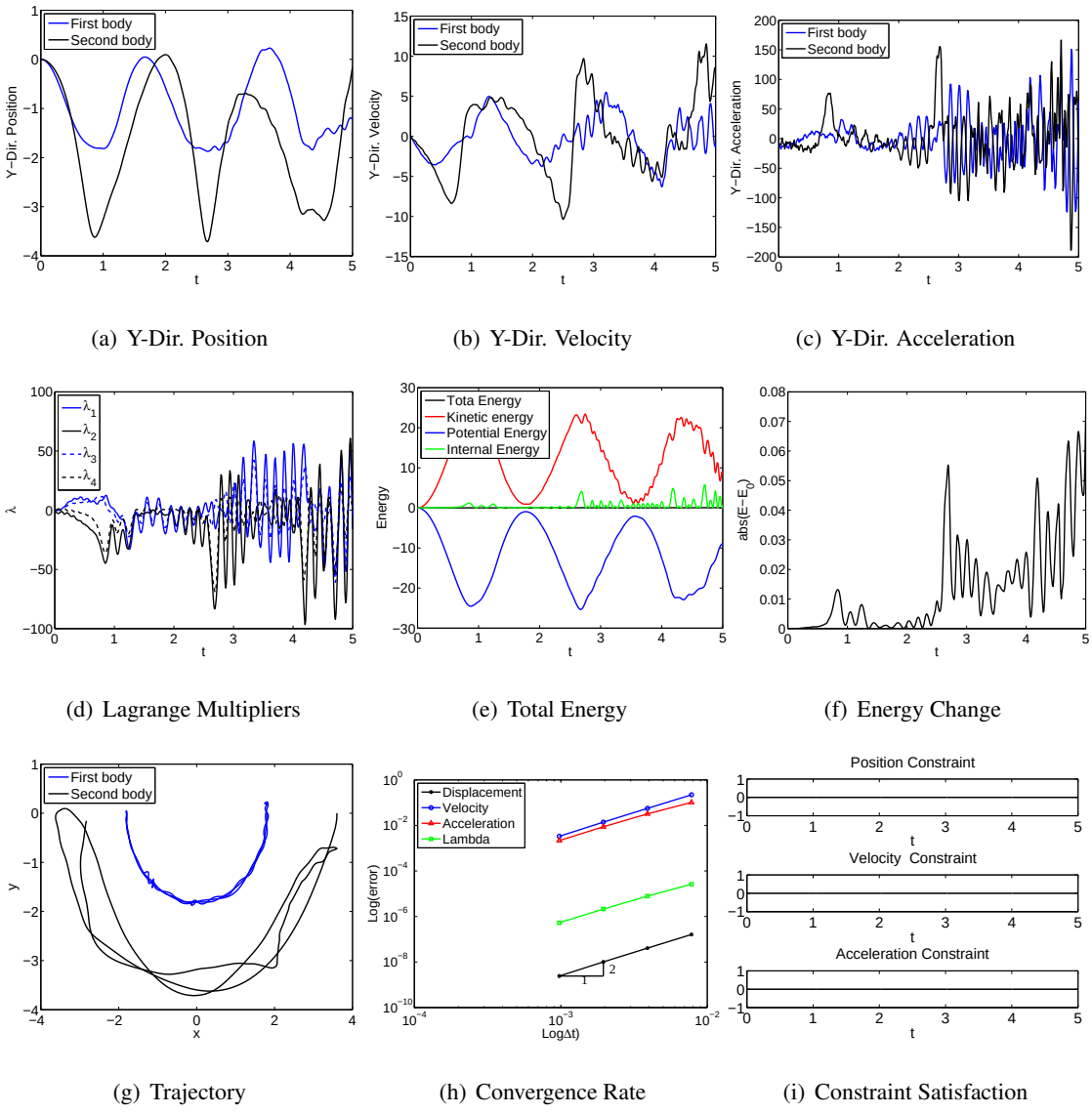


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.192: Double flexible pendulum with TB element in ANCF-S: U0(1,1,0) - Index 3.

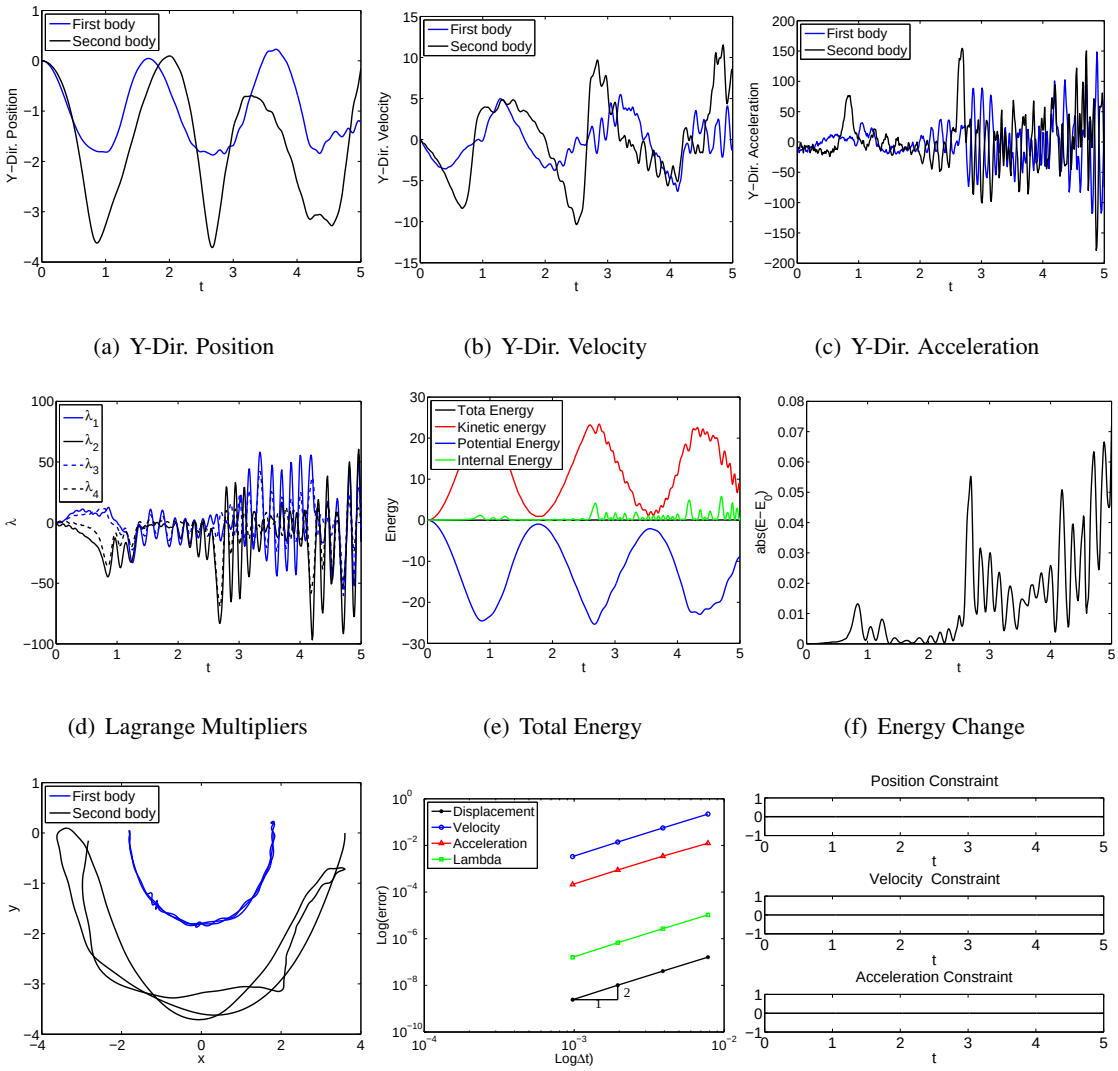


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.193: Double flexible pendulum with TB element in ANCF-S: V0(1,1,0) - Index 3.

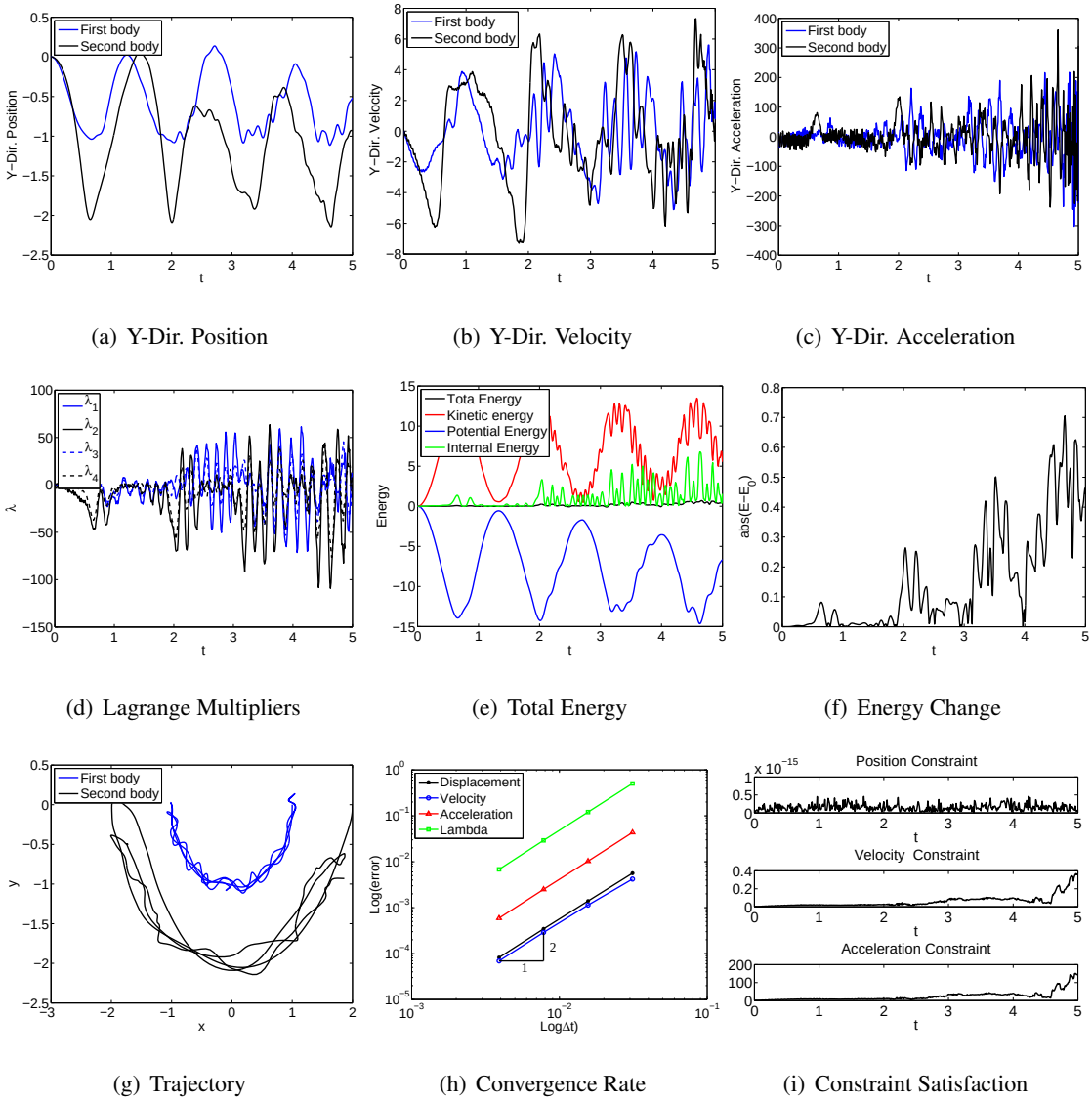


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.194: Double flexible pendulum with TB element in ANCF-S: U0V0(1,1,1) - Index 3.

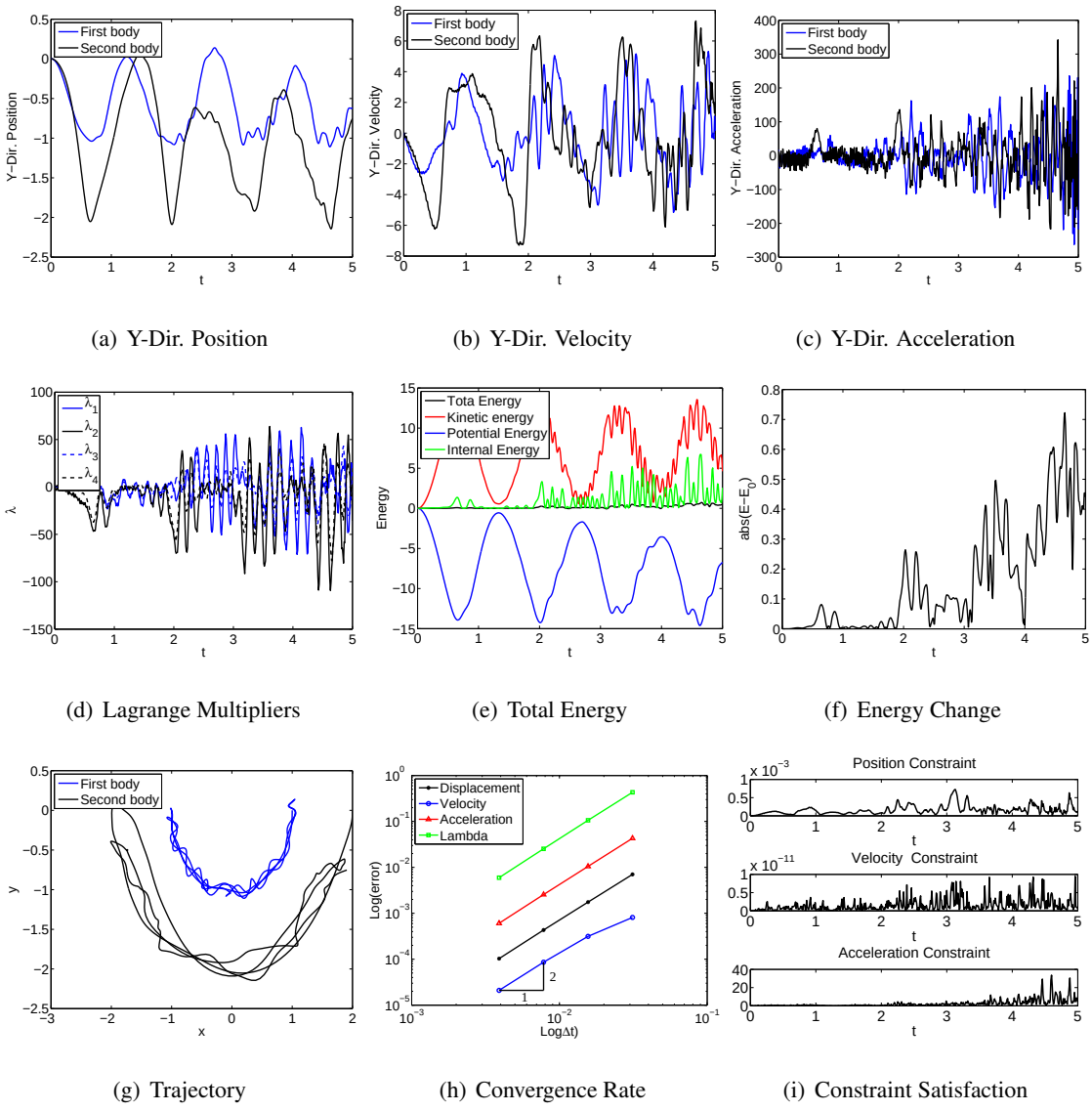


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.195: Double flexible pendulum with TB element in FRF: U0(1,1,0) - Index 3.

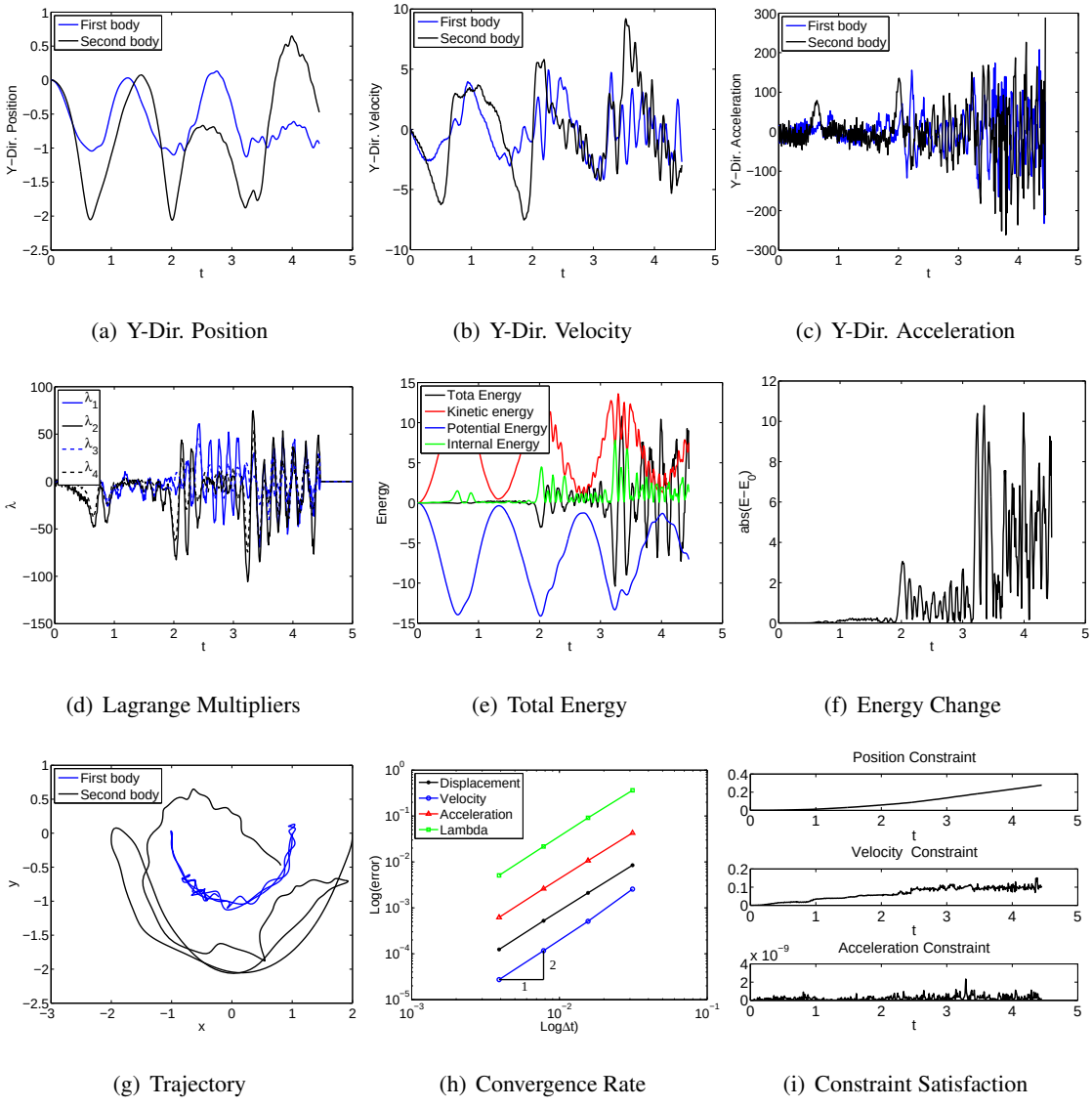


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.196: Double flexible pendulum with TB element in FRF: U0(1,1,0) - Index 2.

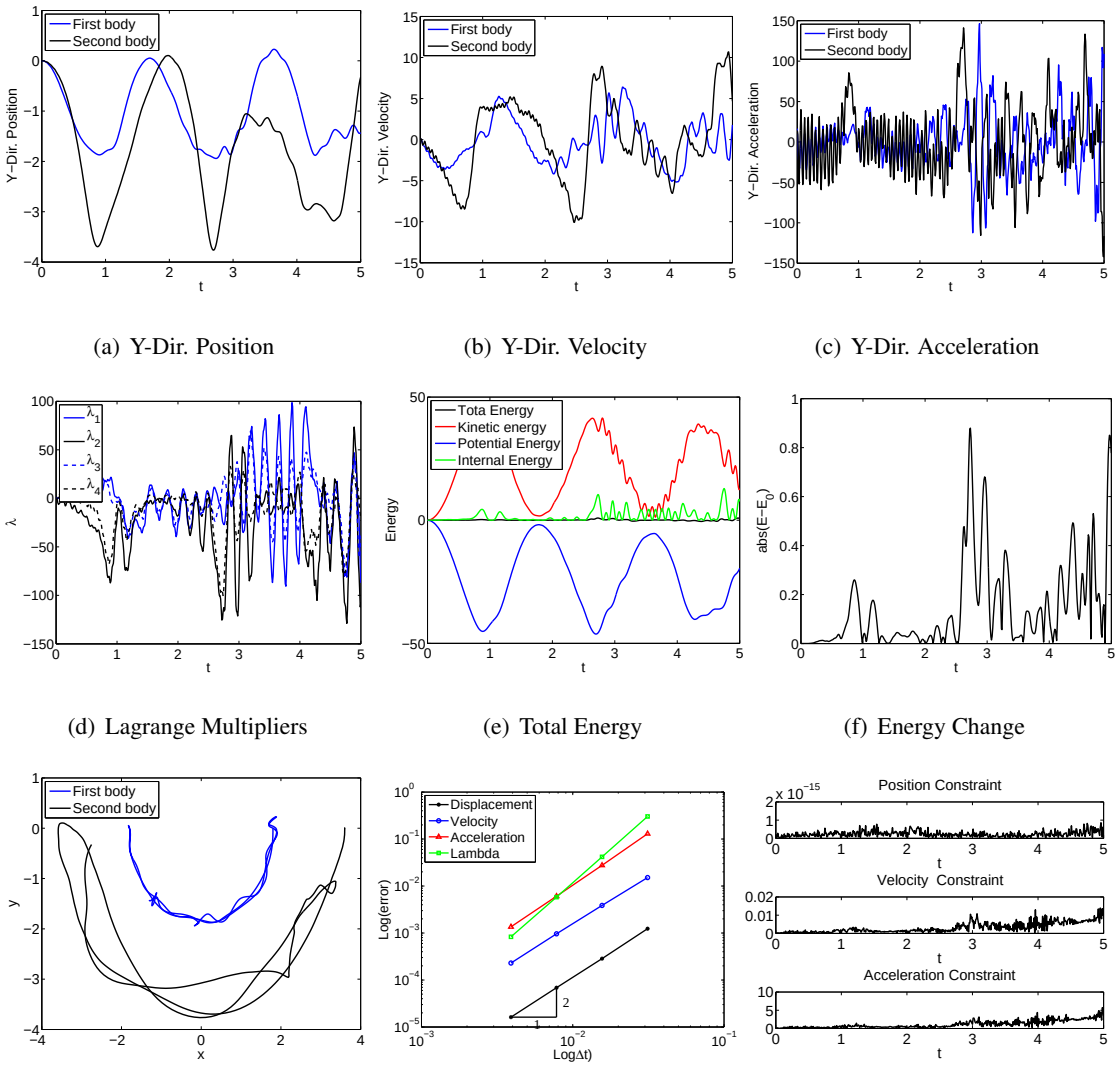


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.197: Double flexible pendulum with TB element in FRF: U0(1,1,0) - Index 1.

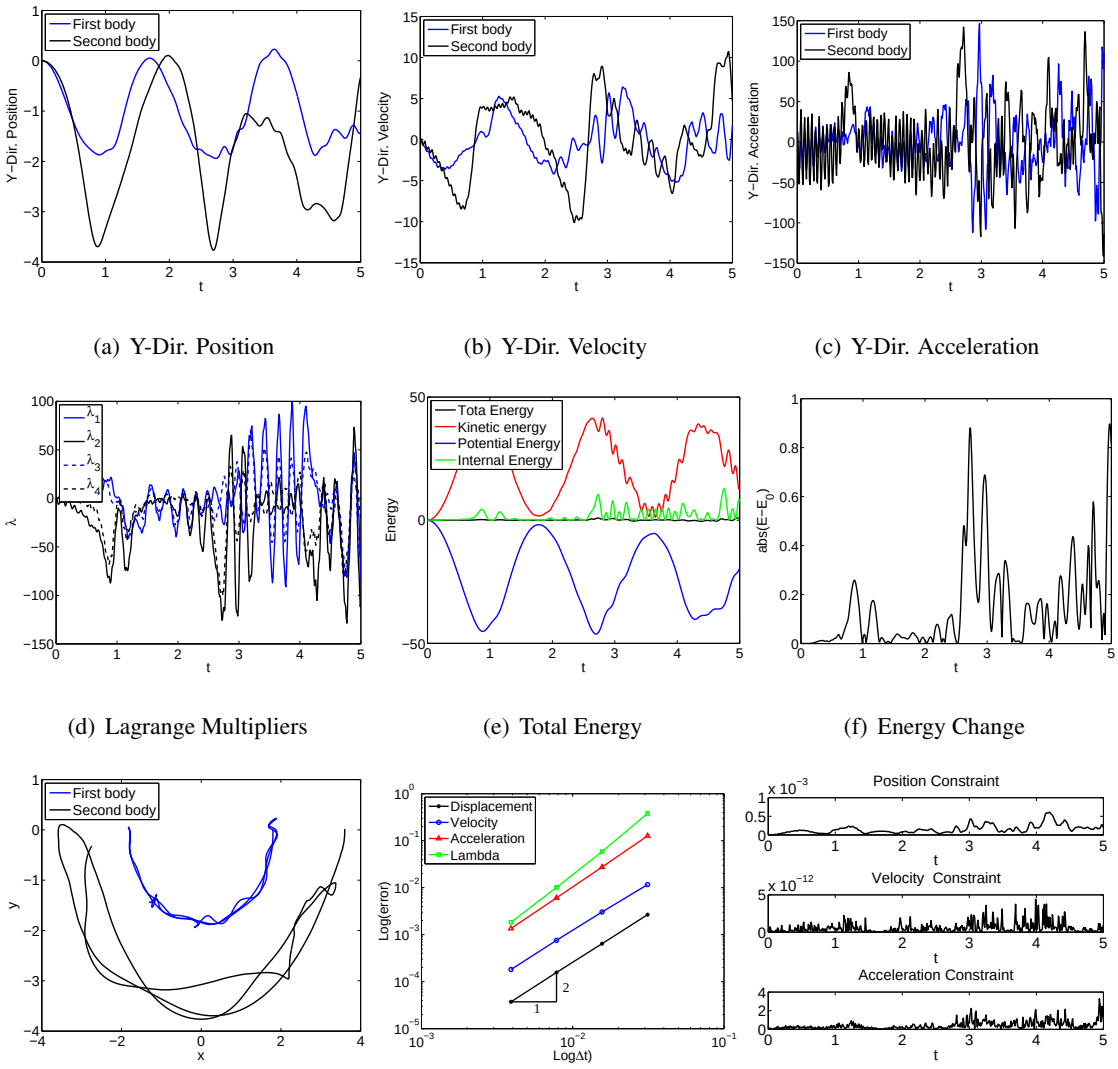


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.198: Double flexible pendulum with TB element in FRF: V0(1,1,0) - Index 3.

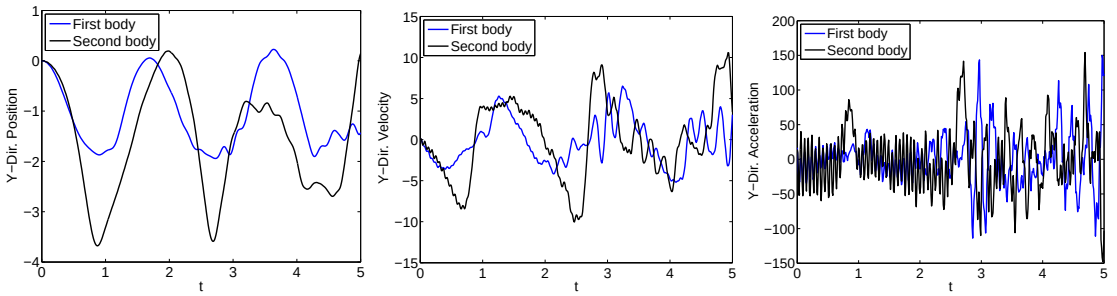


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

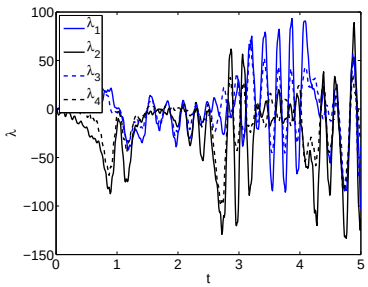
Figure 4.199: Double flexible pendulum with TB element in FRF: V0(1,1,0) - Index 2.



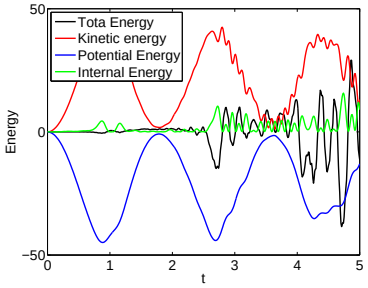
(a) Y-Dir. Position

(b) Y-Dir. Velocity

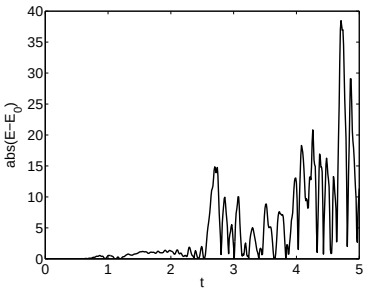
(c) Y-Dir. Acceleration



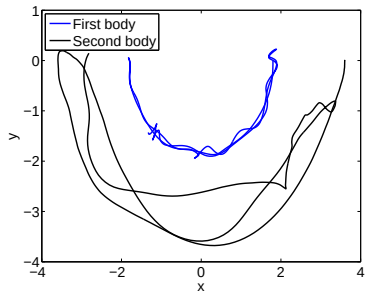
(d) Lagrange Multipliers



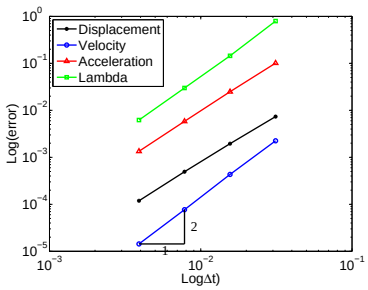
(e) Total Energy



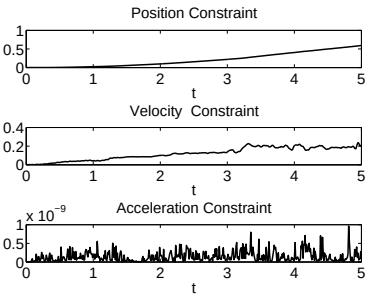
(f) Energy Change



(g) Trajectory

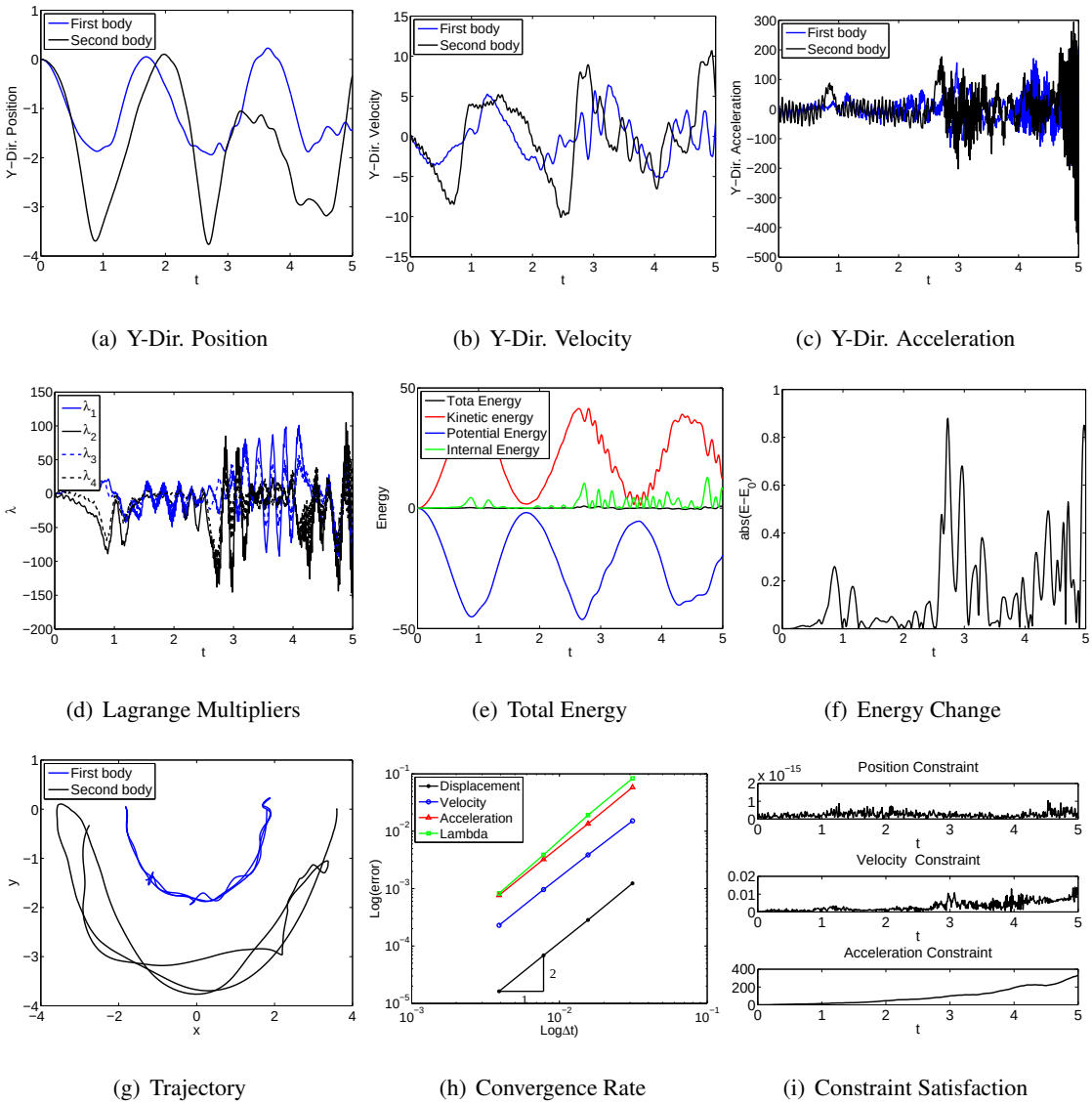


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.200: Double flexible pendulum with TB element in FRF: V0(1,1,0) - Index 1.

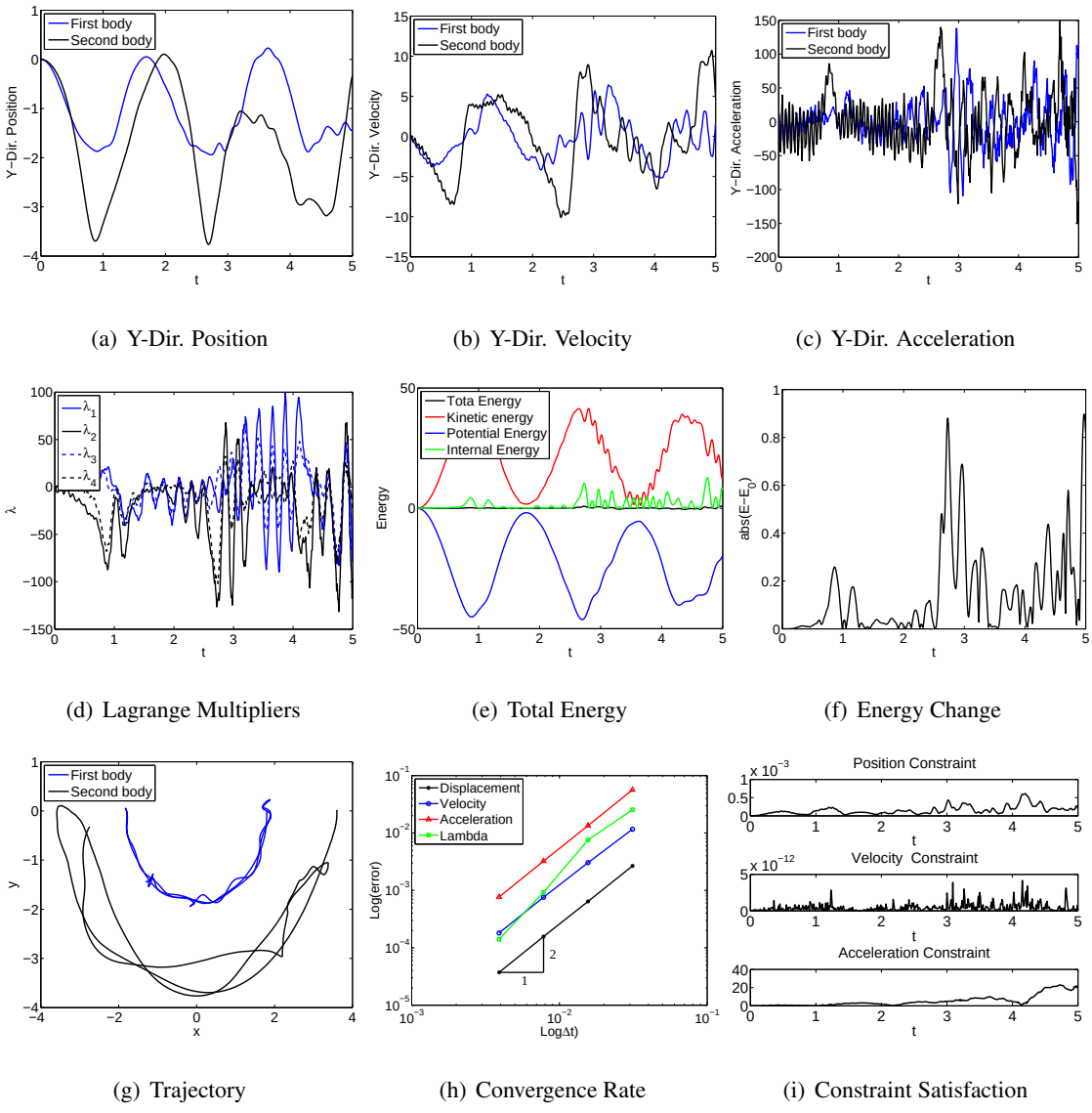


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.201: Double flexible pendulum with TB element in FRF: U0V0(1,1,1) - Index 3.

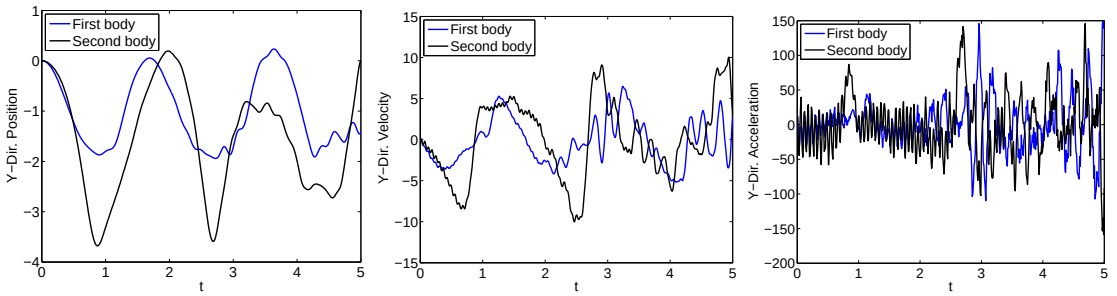


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

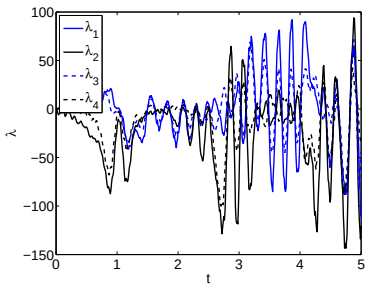
Figure 4.202: Double flexible pendulum with TB element in FRF: U0V0(1,1,1) - Index 2.



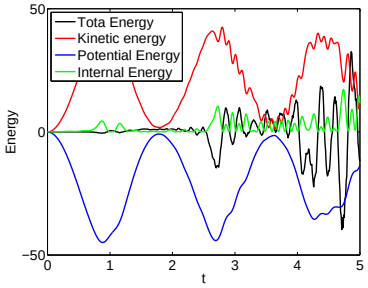
(a) Y-Dir. Position

(b) Y-Dir. Velocity

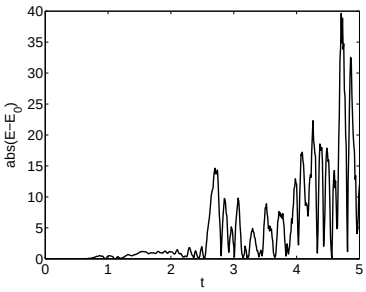
(c) Y-Dir. Acceleration



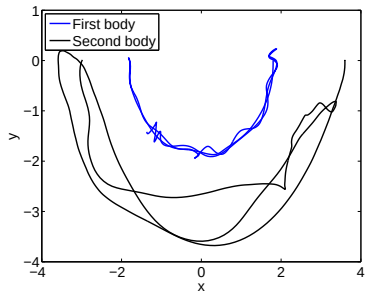
(d) Lagrange Multipliers



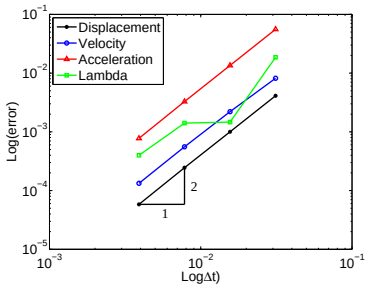
(e) Total Energy



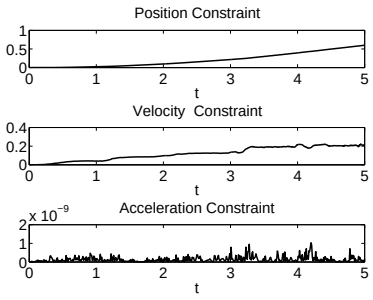
(f) Energy Change



(g) Trajectory



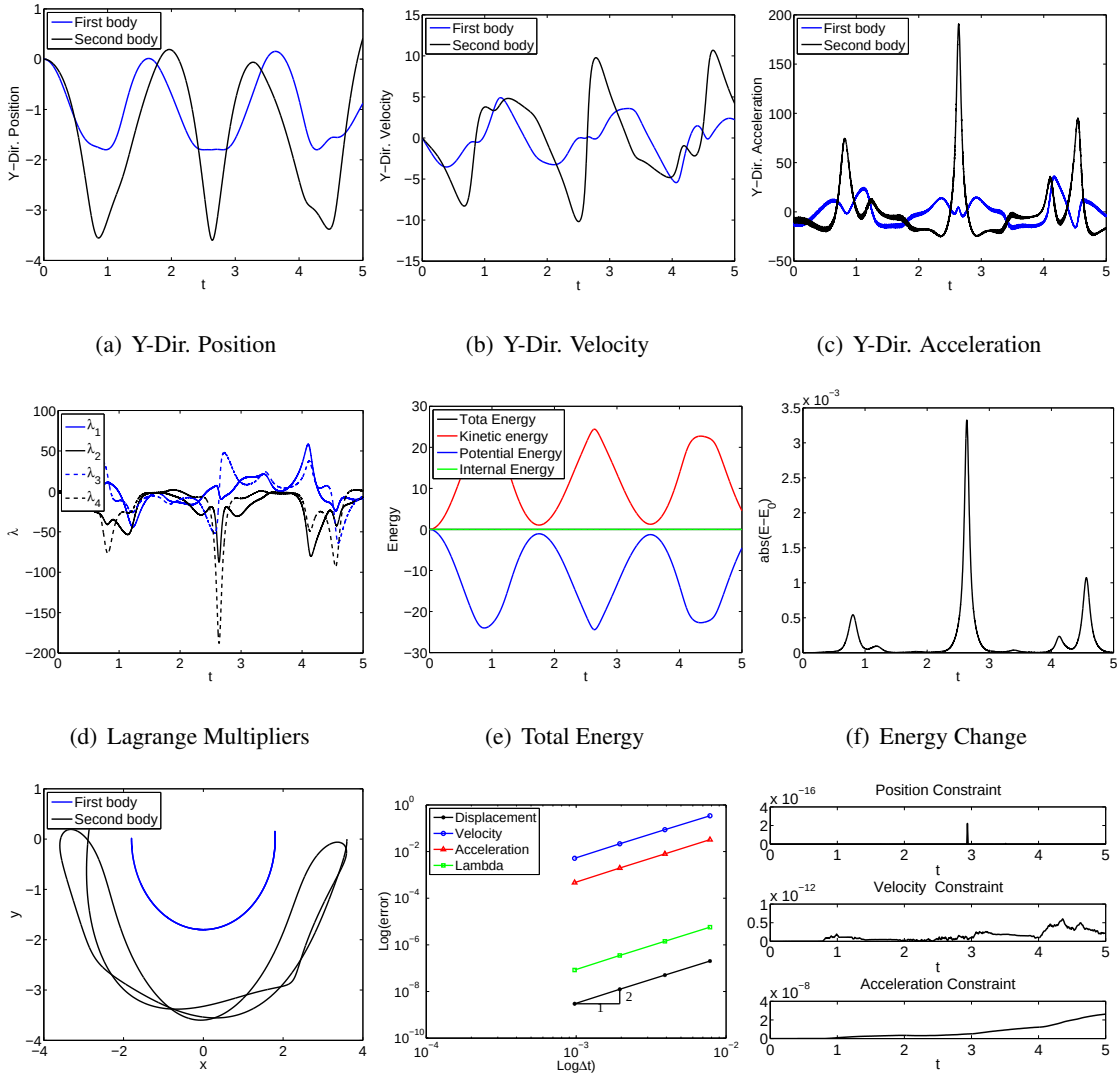
(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.203: Double flexible pendulum with TB element in FRF: U0V0(1,1,1) - Index 1.

Timoshenko Beam: Quasi-Rigid Case

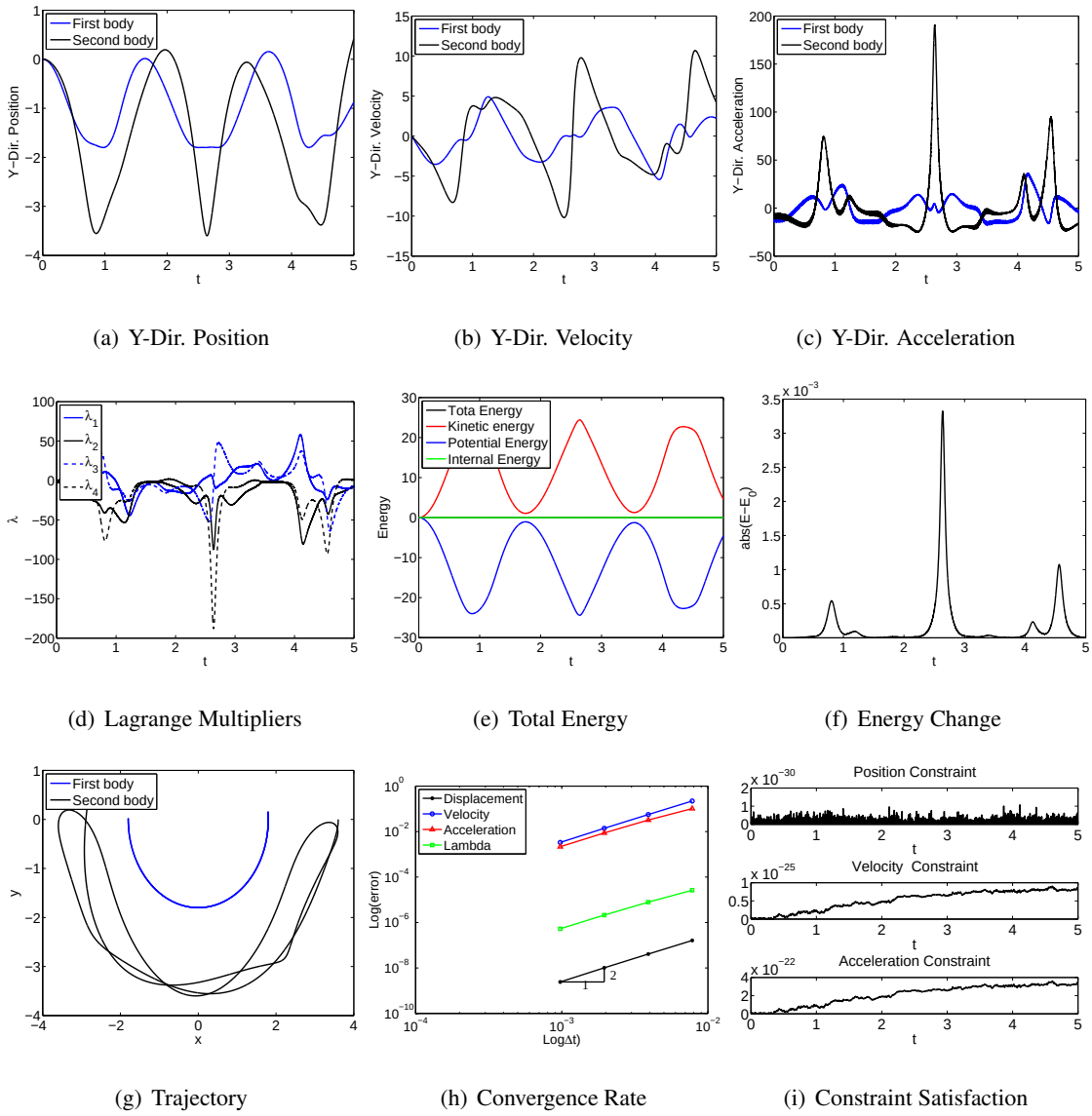


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.204: Double quasi-rigid pendulum with TB element in ANCF-S: U0(1,1,0) - Index 3.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.205: Double quasi-rigid pendulum with TB element in ANCF-S: V0(1,1,0) - Index 3.

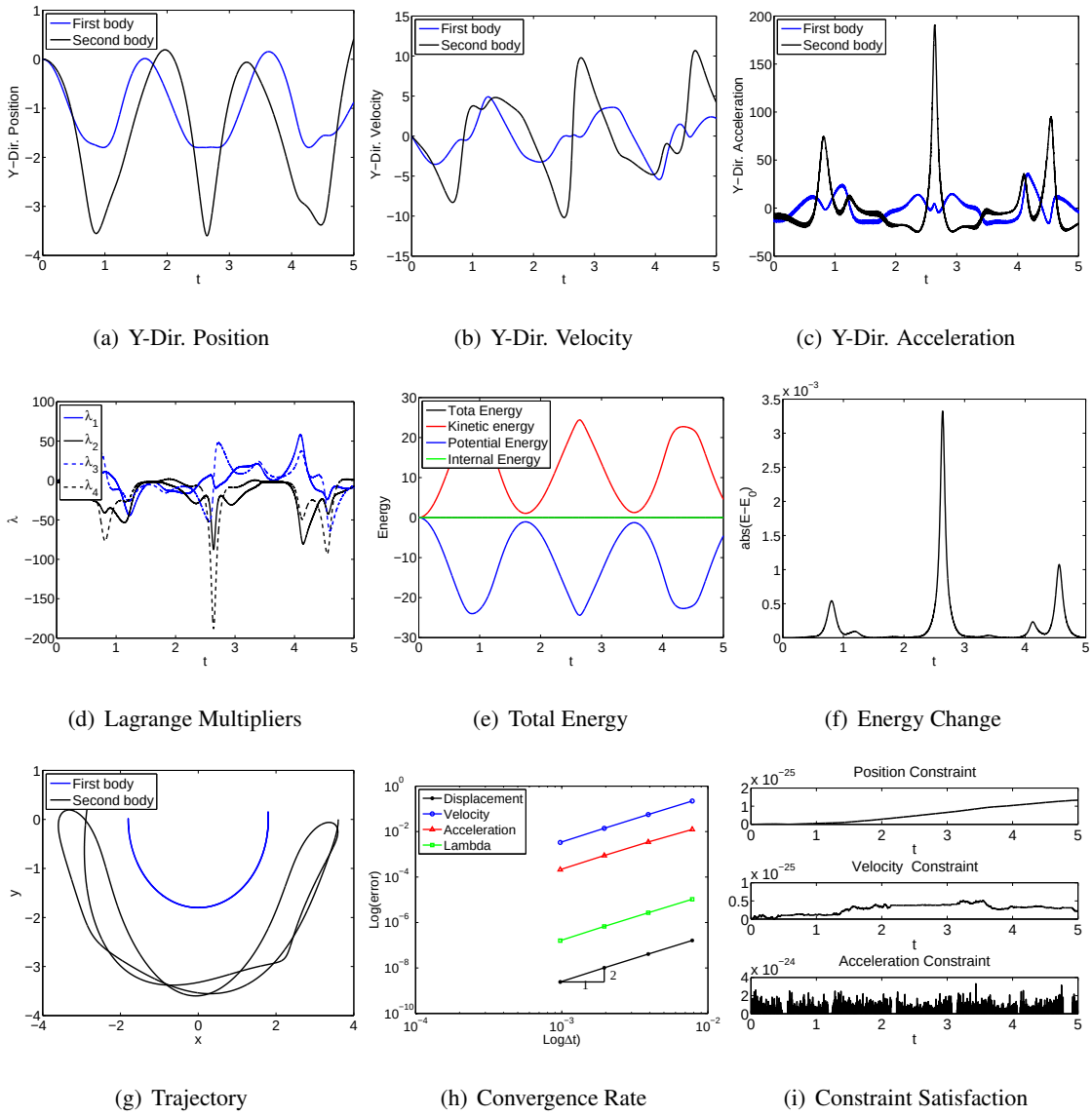
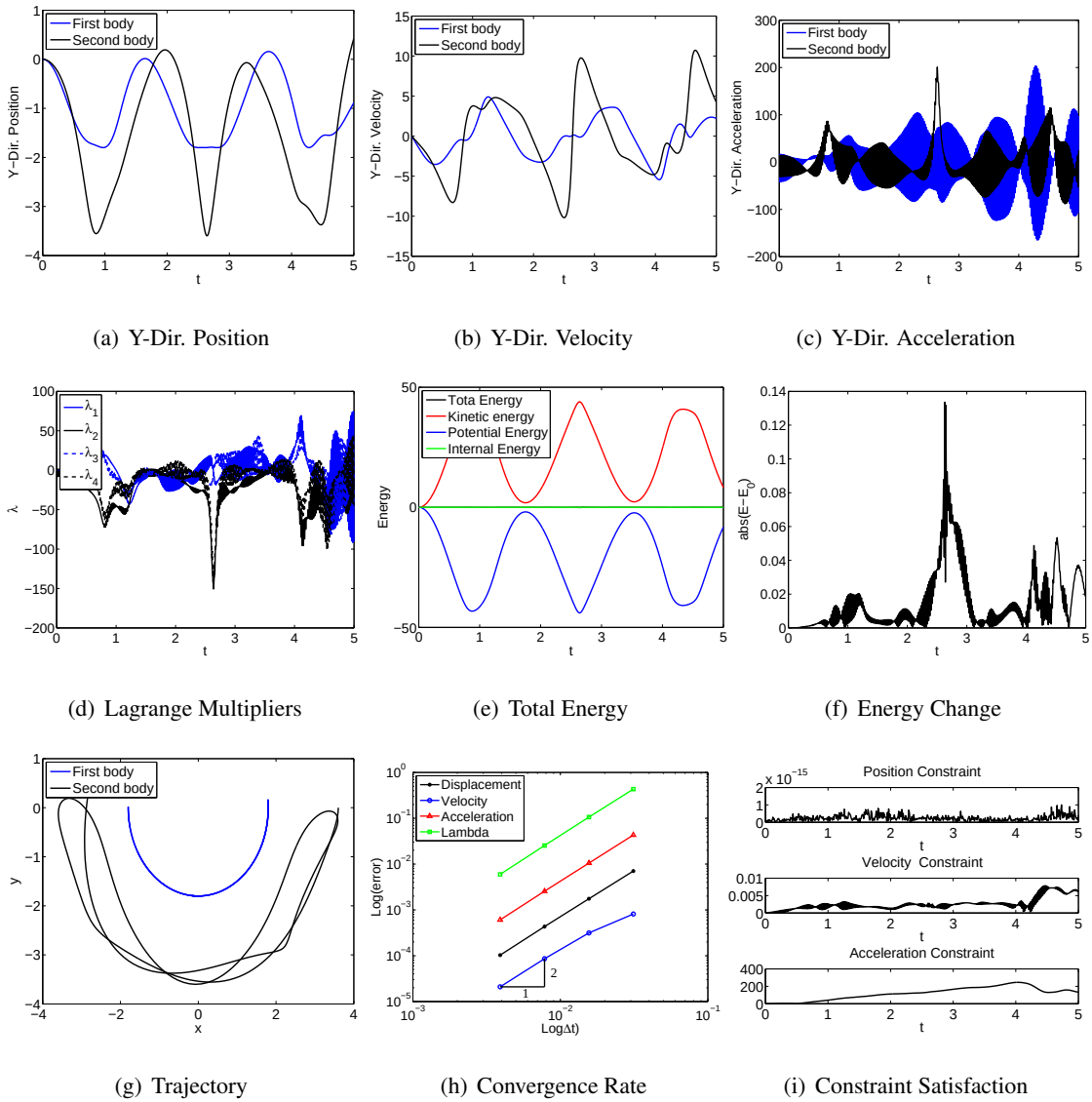


Figure 4.206: Double quasi-rigid pendulum with TB element in ANCF-S: U0V0(1,1,1) - Index 3.

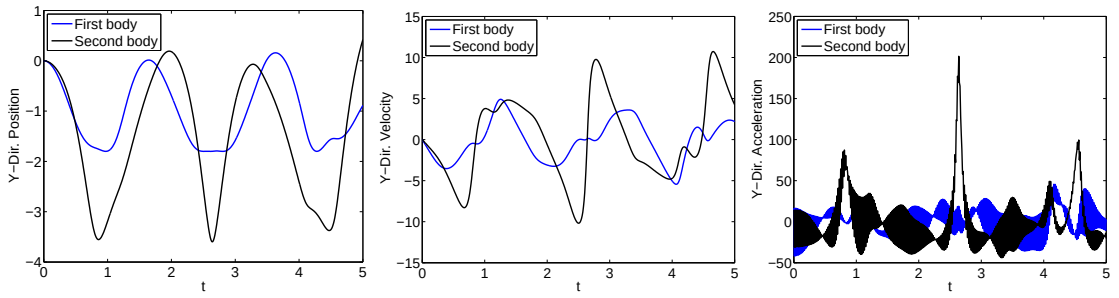


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

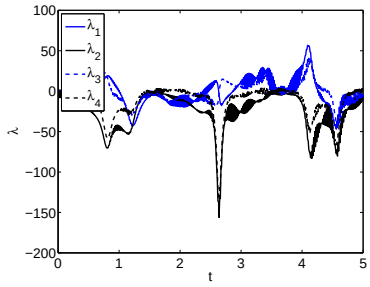
Figure 4.207: Double quasi-rigid pendulum with TB element in FRF: U0(1,1,0) - Index 3.



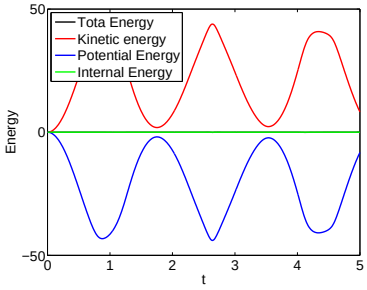
(a) Y-Dir. Position

(b) Y-Dir. Velocity

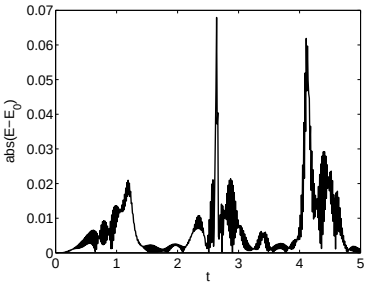
(c) Y-Dir. Acceleration



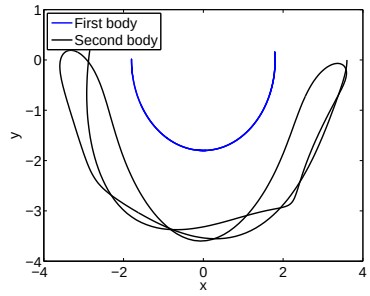
(d) Lagrange Multipliers



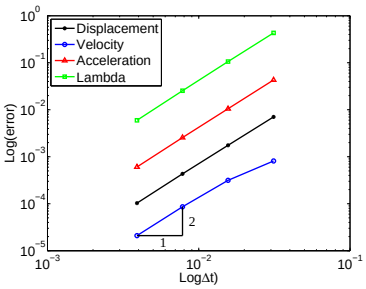
(e) Total Energy



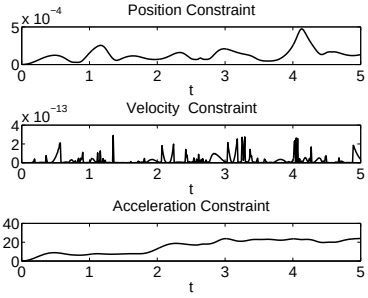
(f) Energy Change



(g) Trajectory

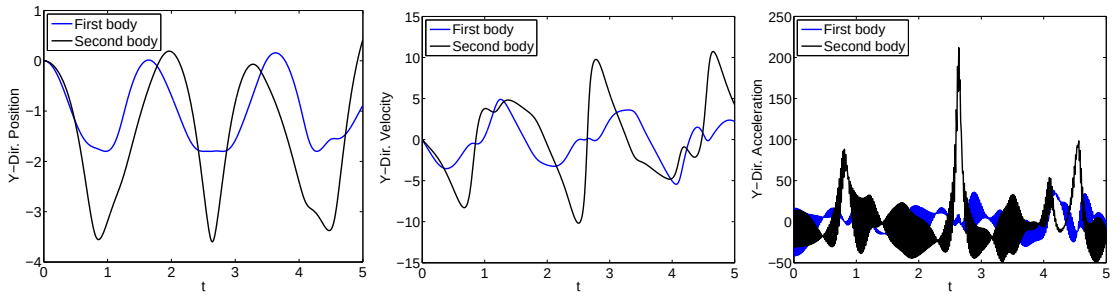


(h) Convergence Rate



(i) Constraint Satisfaction

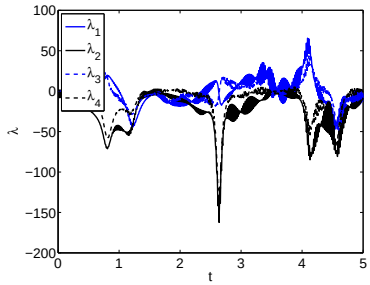
Figure 4.208: Double quasi-rigid pendulum with TB element in FRF: U0(1,1,0) - Index 2.



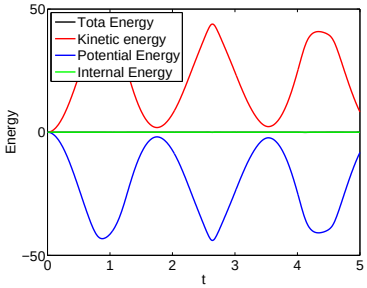
(a) Y-Dir. Position

(b) Y-Dir. Velocity

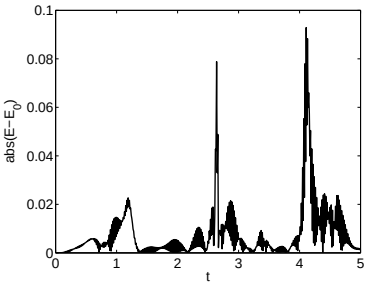
(c) Y-Dir. Acceleration



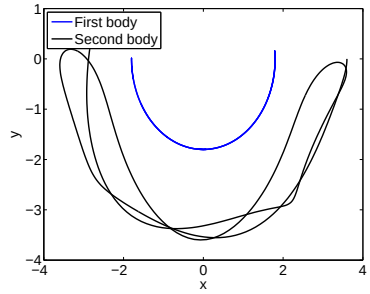
(d) Lagrange Multipliers



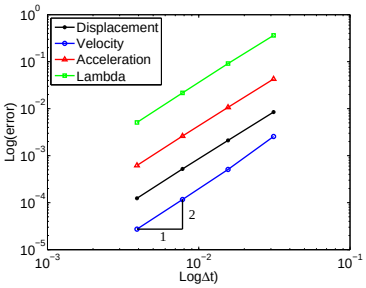
(e) Total Energy



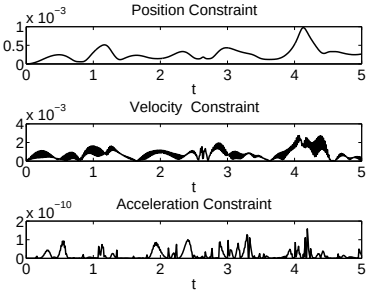
(f) Energy Change



(g) Trajectory

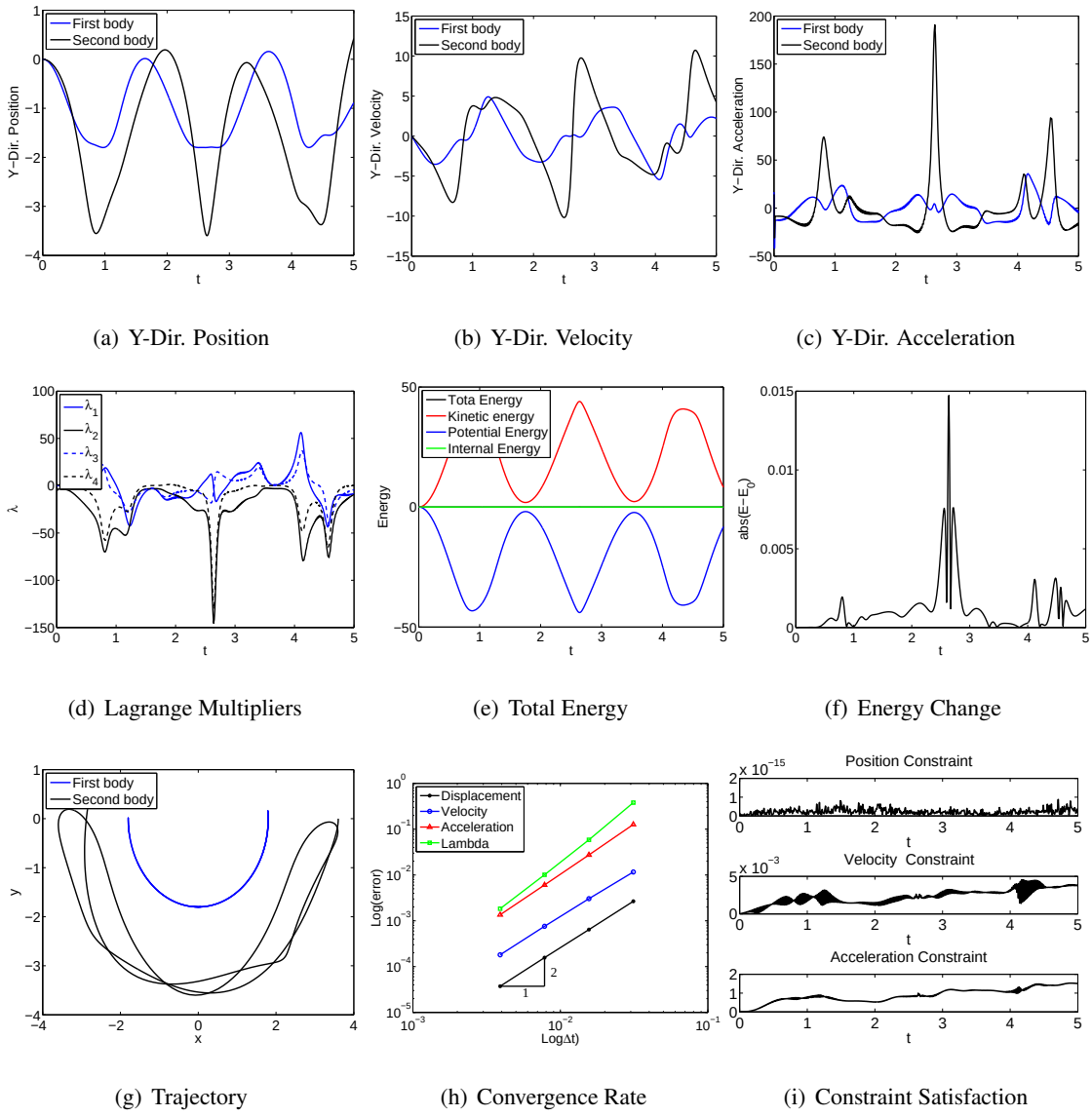


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.209: Double quasi-rigid pendulum with TB element in FRF: U0(1,1,0) - Index 1.

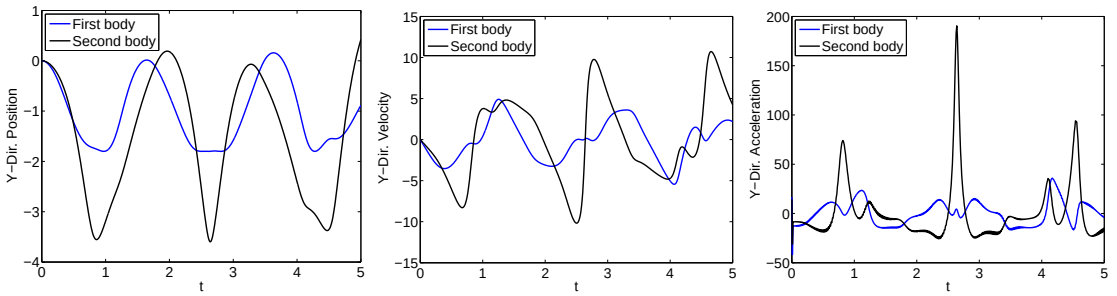


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

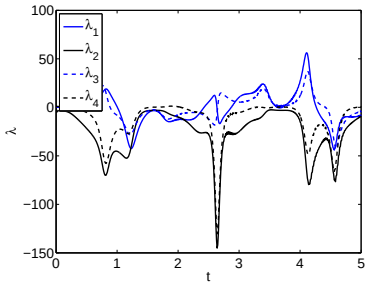
Figure 4.210: Double quasi-rigid pendulum with TB element in FRF: V0(1,1,0) - Index 3.



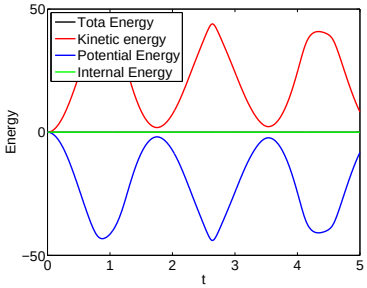
(a) Y-Dir. Position

(b) Y-Dir. Velocity

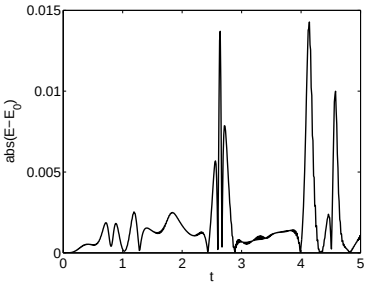
(c) Y-Dir. Acceleration



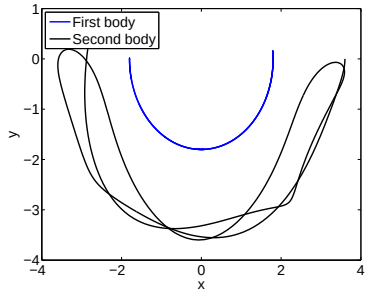
(d) Lagrange Multipliers



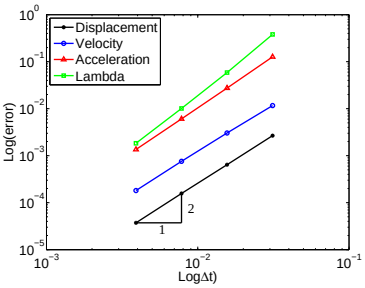
(e) Total Energy



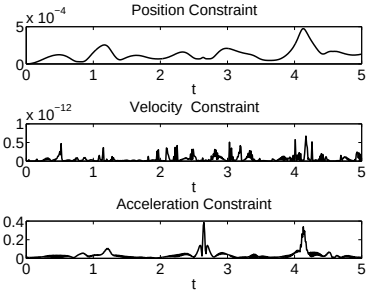
(f) Energy Change



(g) Trajectory

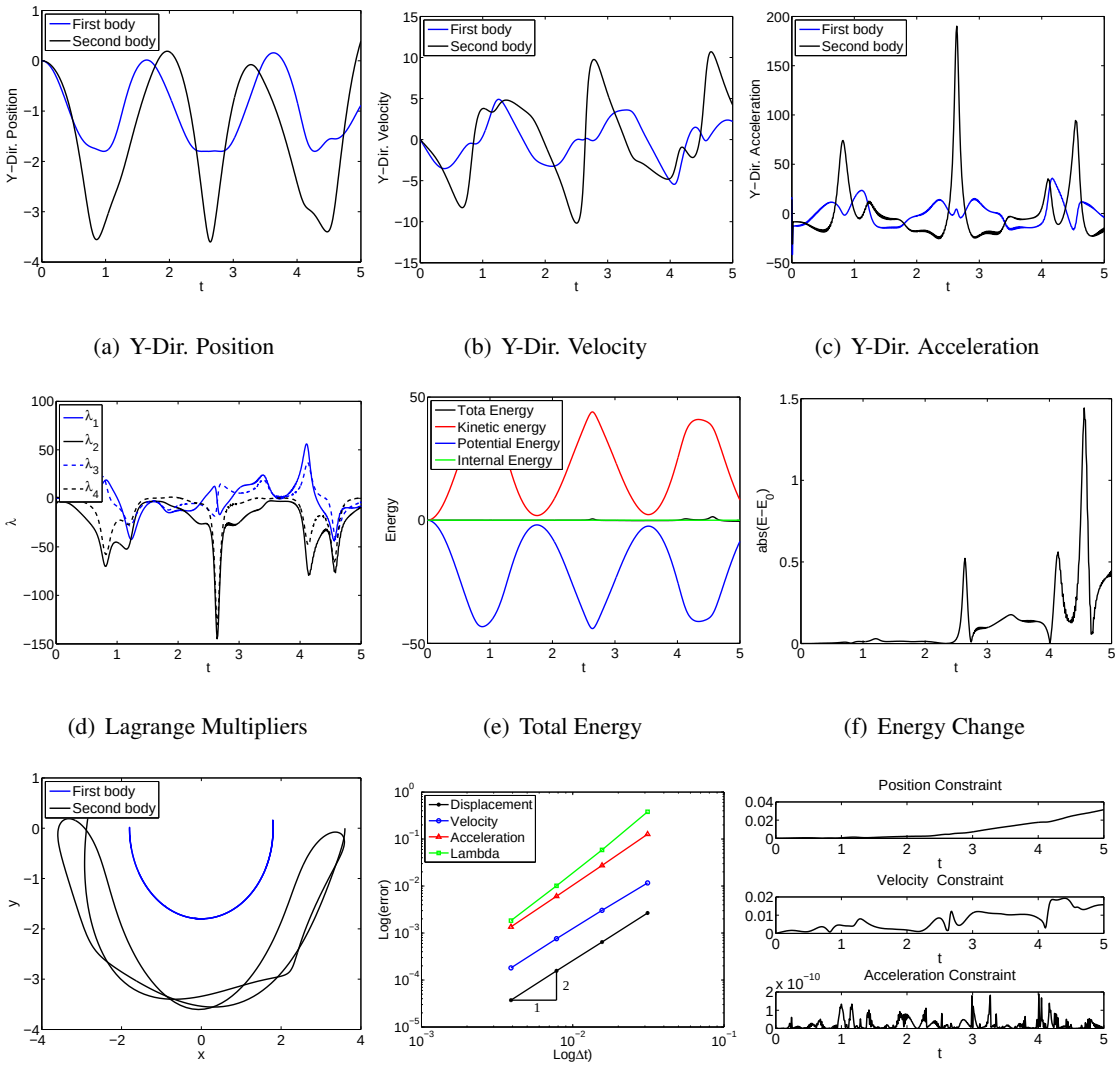


(h) Convergence Rate



(i) Constraint Satisfaction

Figure 4.211: Double quasi-rigid pendulum with TB element in FRF: V0(1,1,0) - Index 2.

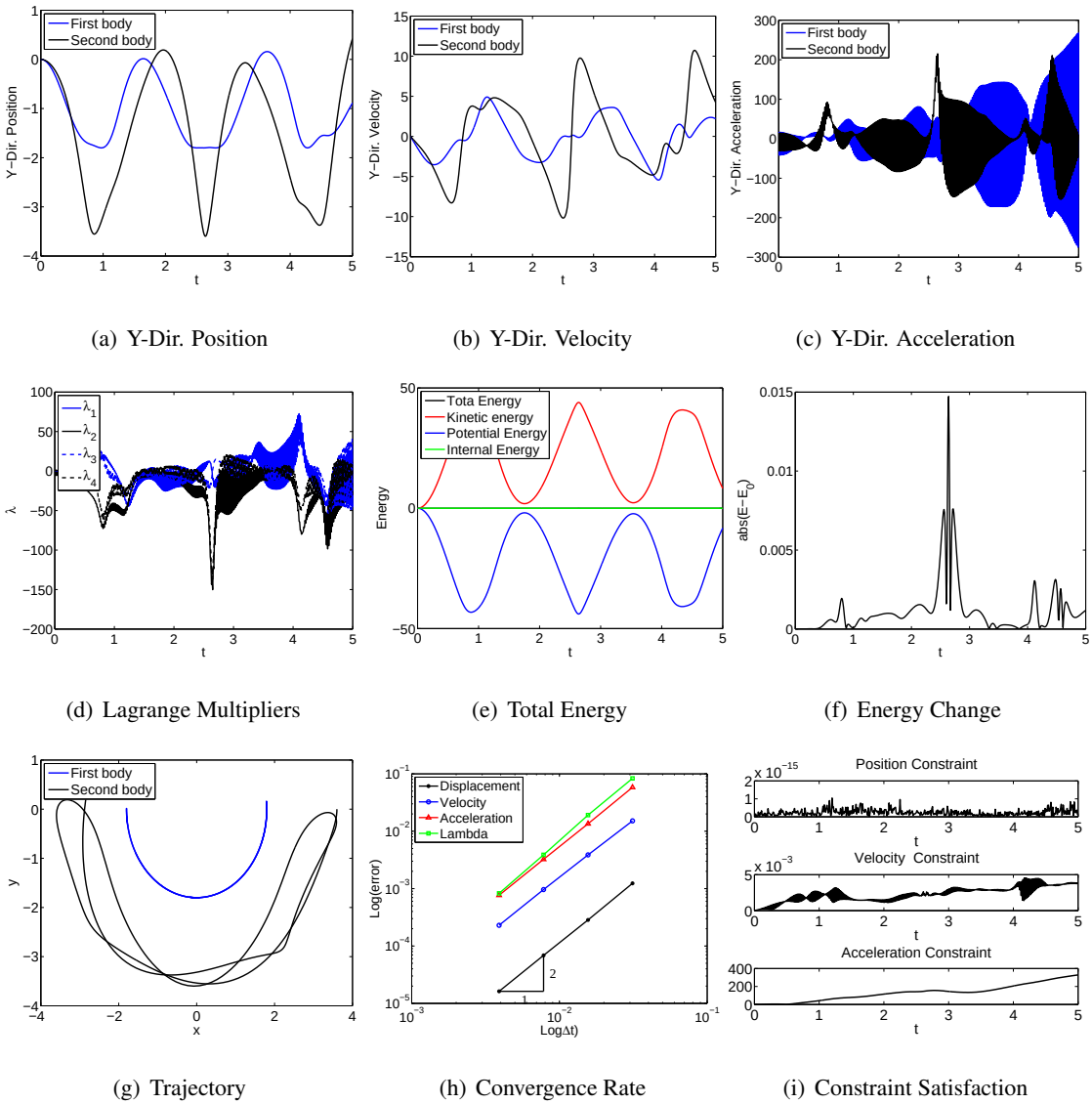


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.212: Double quasi-rigid pendulum with TB element in FRF: V0(1,1,0) - Index 1.

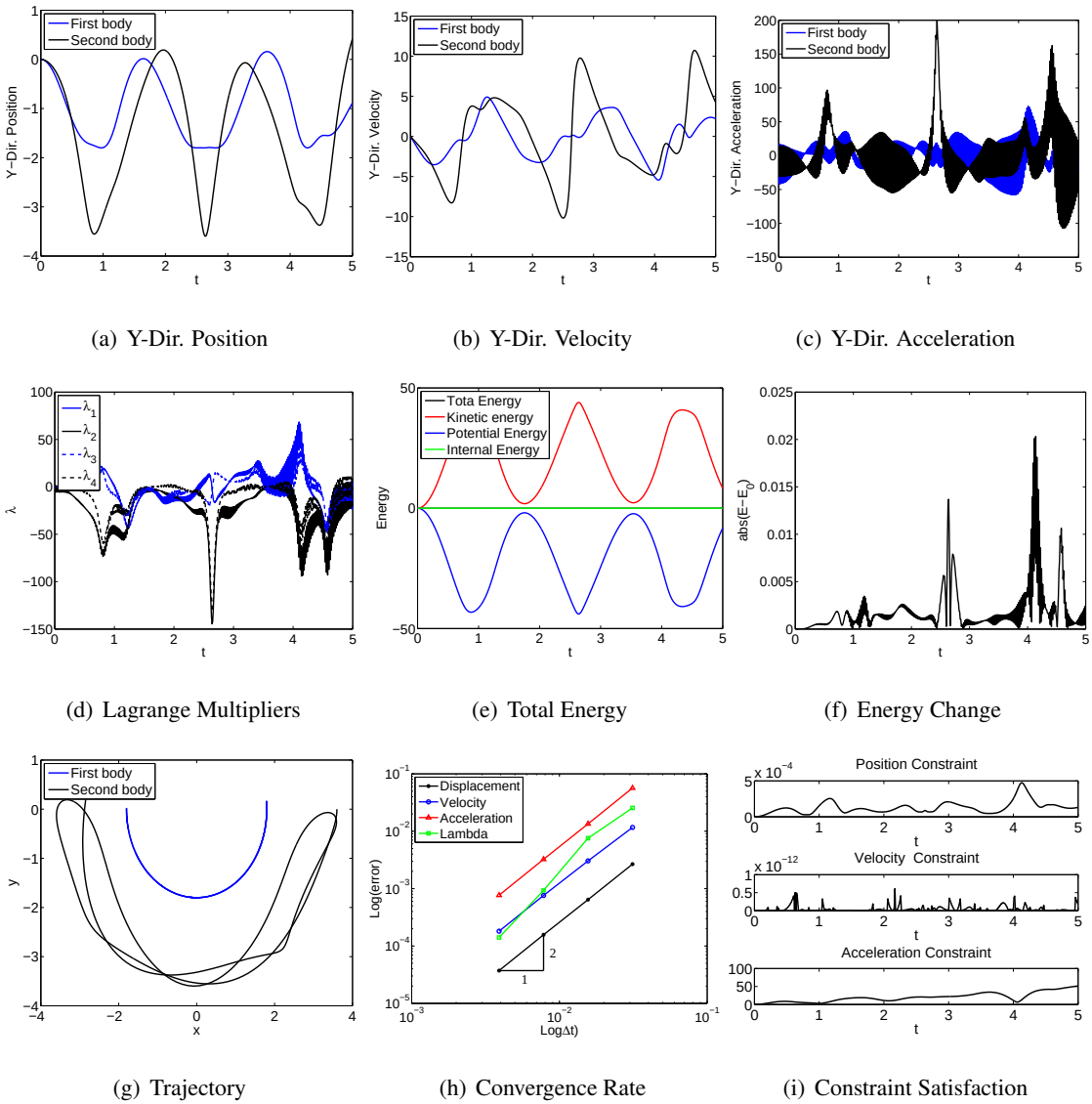


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.213: Double quasi-rigid pendulum with TB element in FRF: U0V0(1,1,1) - Index 3.

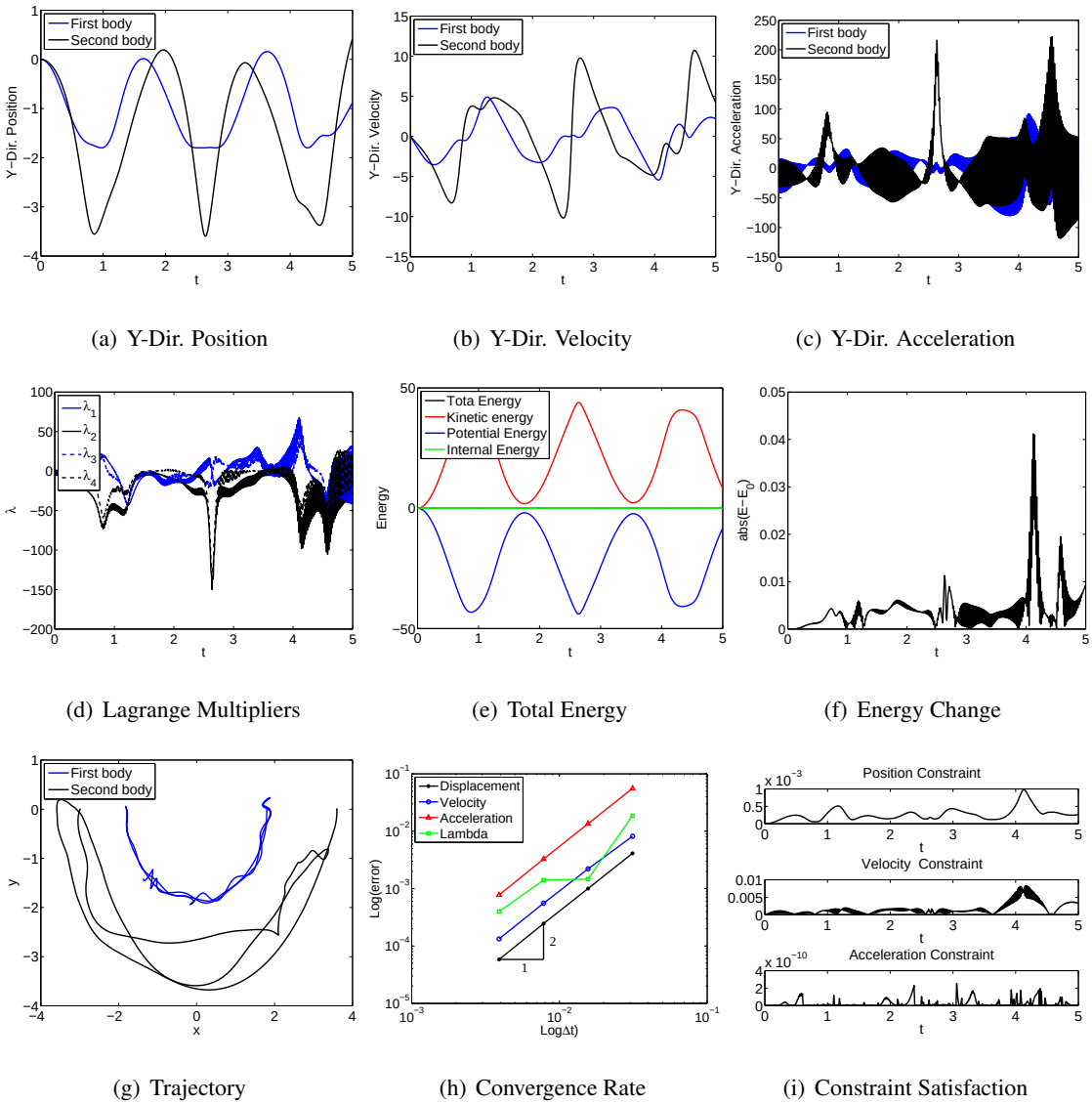


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.214: Double quasi-rigid pendulum with TB element in FRF: U0V0(1,1,1) - Index 2.



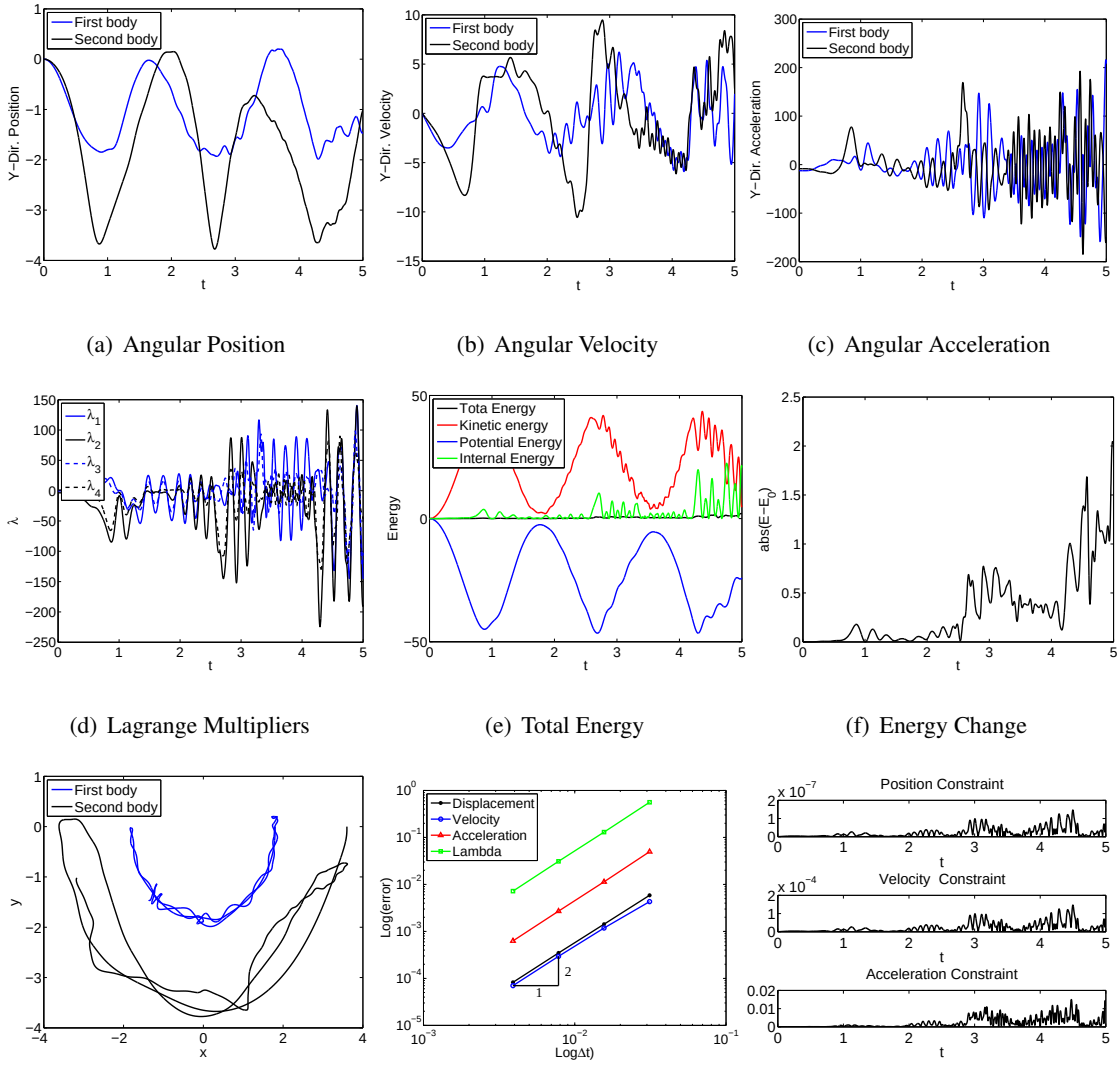
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.215: Double quasi-rigid pendulum with TB element in FRF: U0V0(1,1,1) - Index 1.

Baumgarte's method

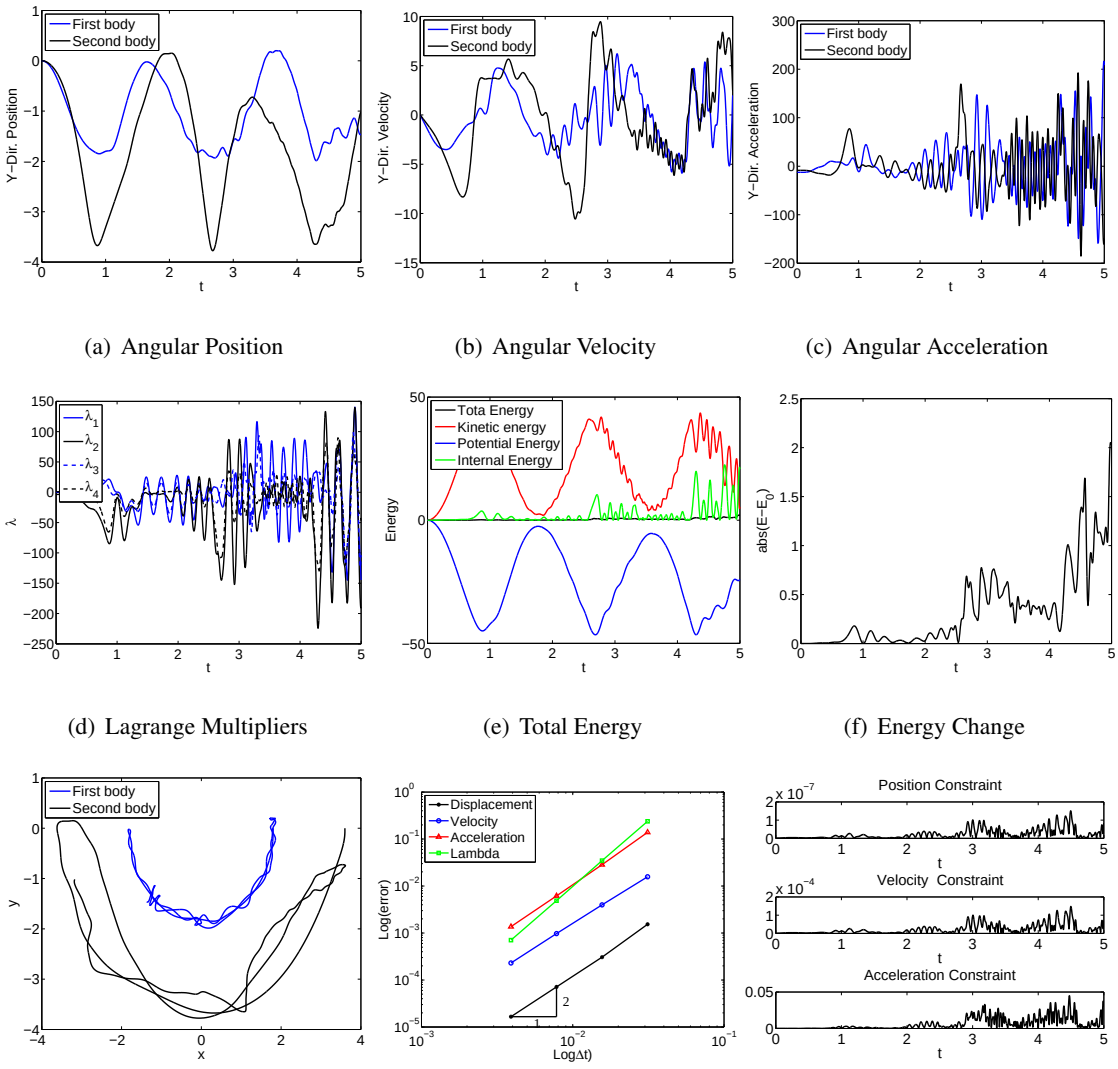


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.216: Double bar pendulum with Baumgarte's method: U0(1,1,0).

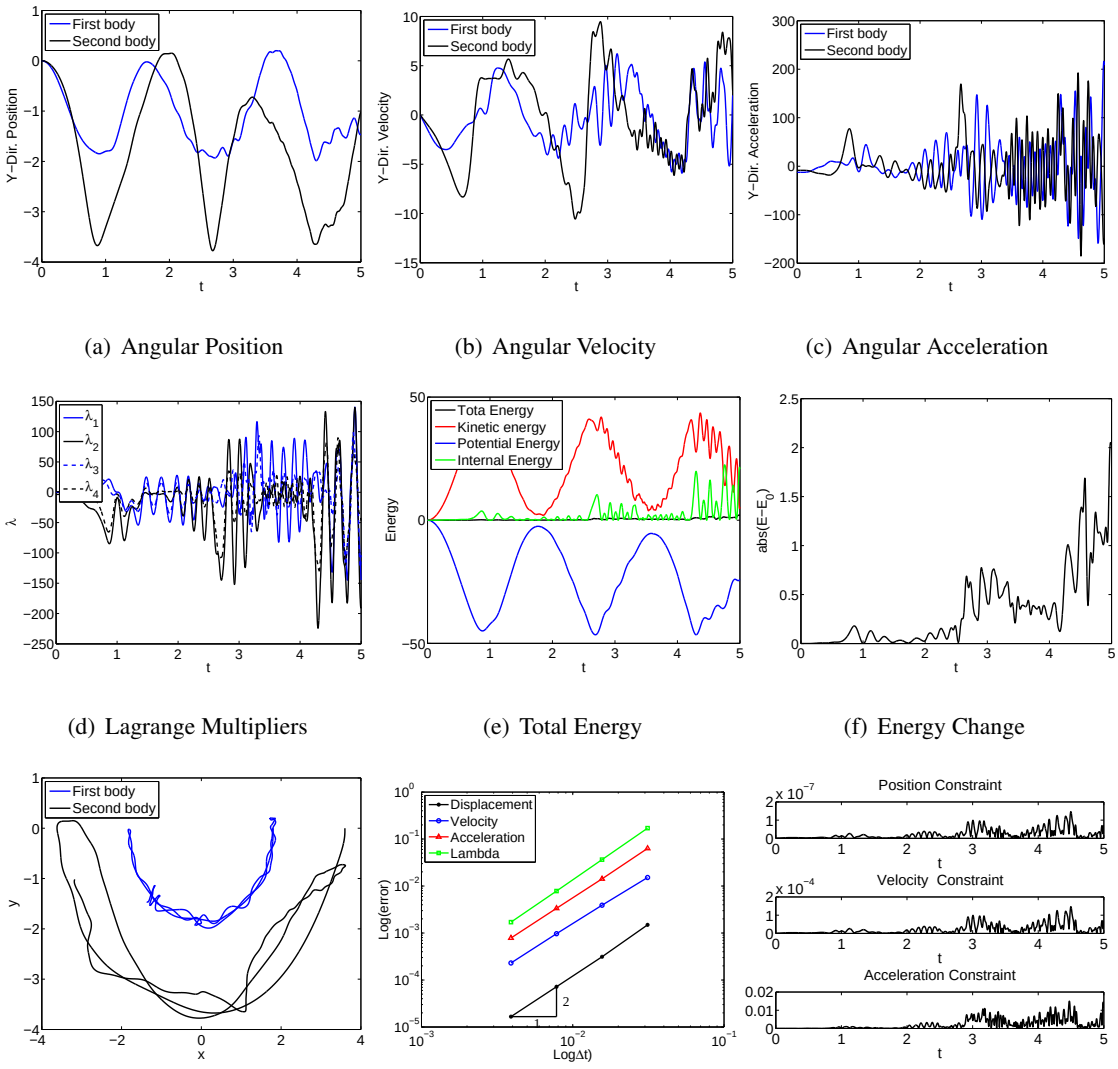


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.217: Double bar pendulum with Baumgarte's method: V0(1,1,0).

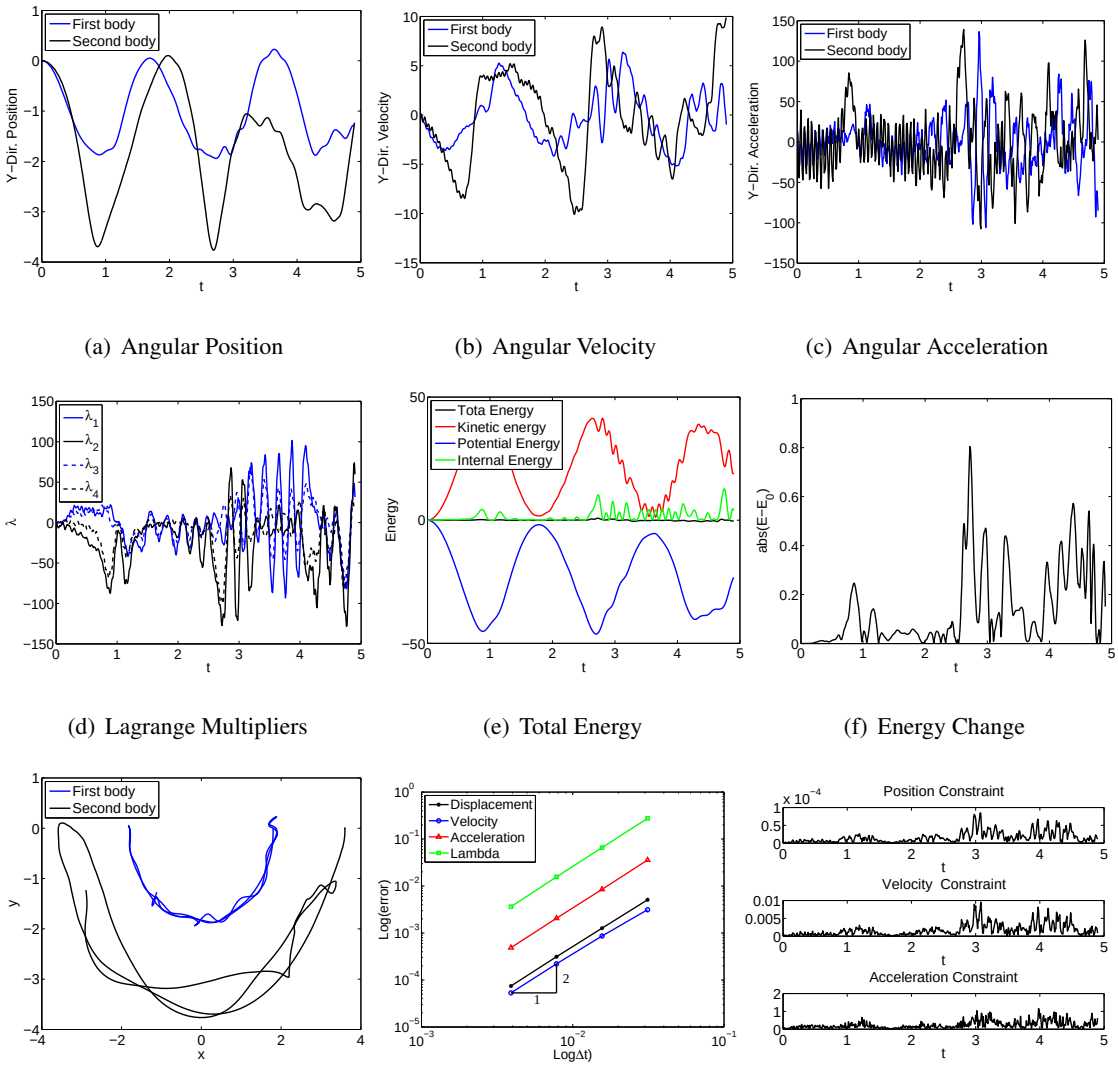


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.218: Double bar pendulum with Baumgarte's method: U0V0(1,1,1).

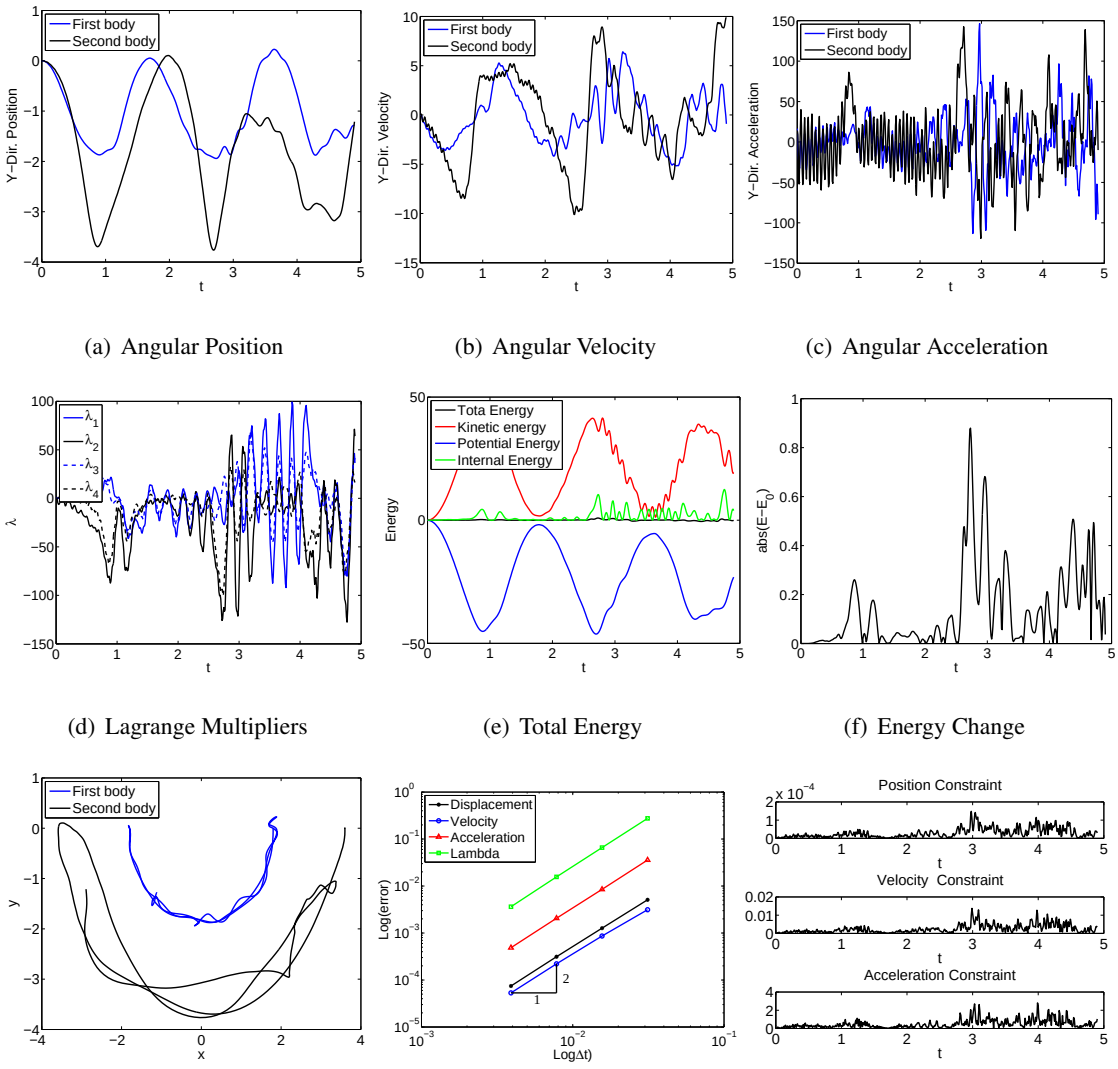


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.219: Double EB pendulum with Baumgarte's method: U0(1,1,0).

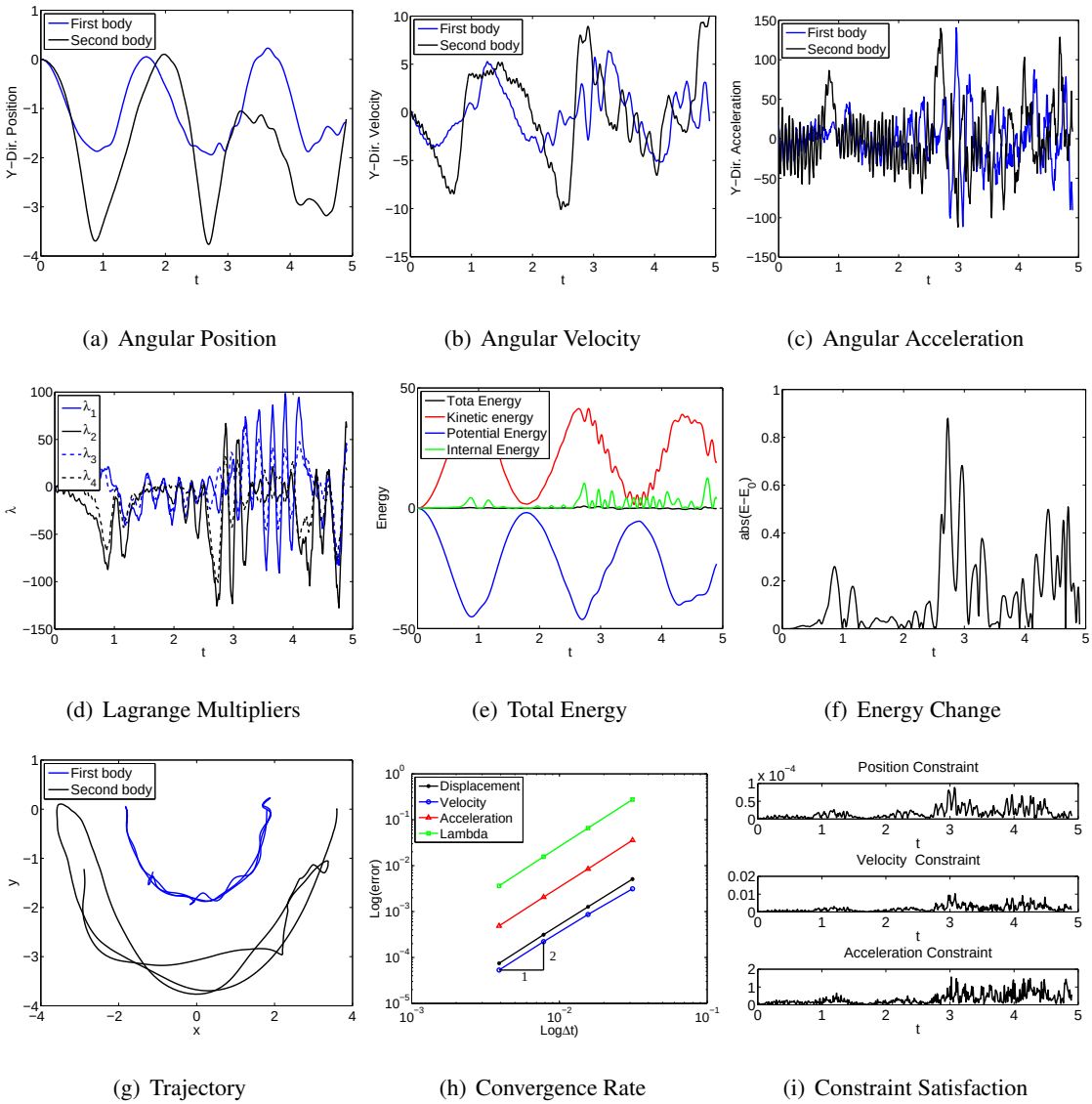


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.220: Double EB pendulum with Baumgarte's method: V0(1,1,0).



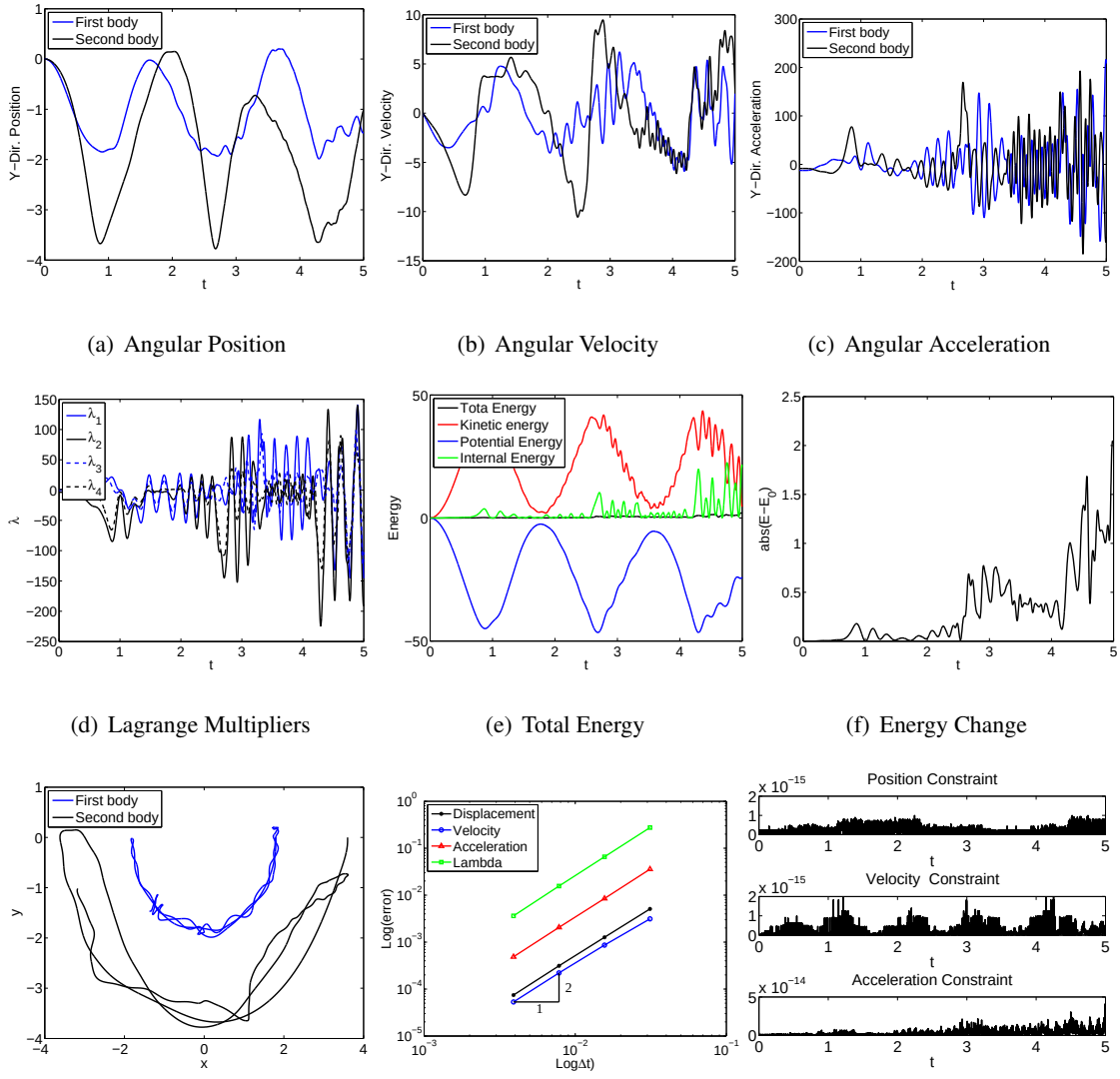
(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.221: Double EB pendulum with Baumgarte's method: U0V0(1,1,1).

Projection method

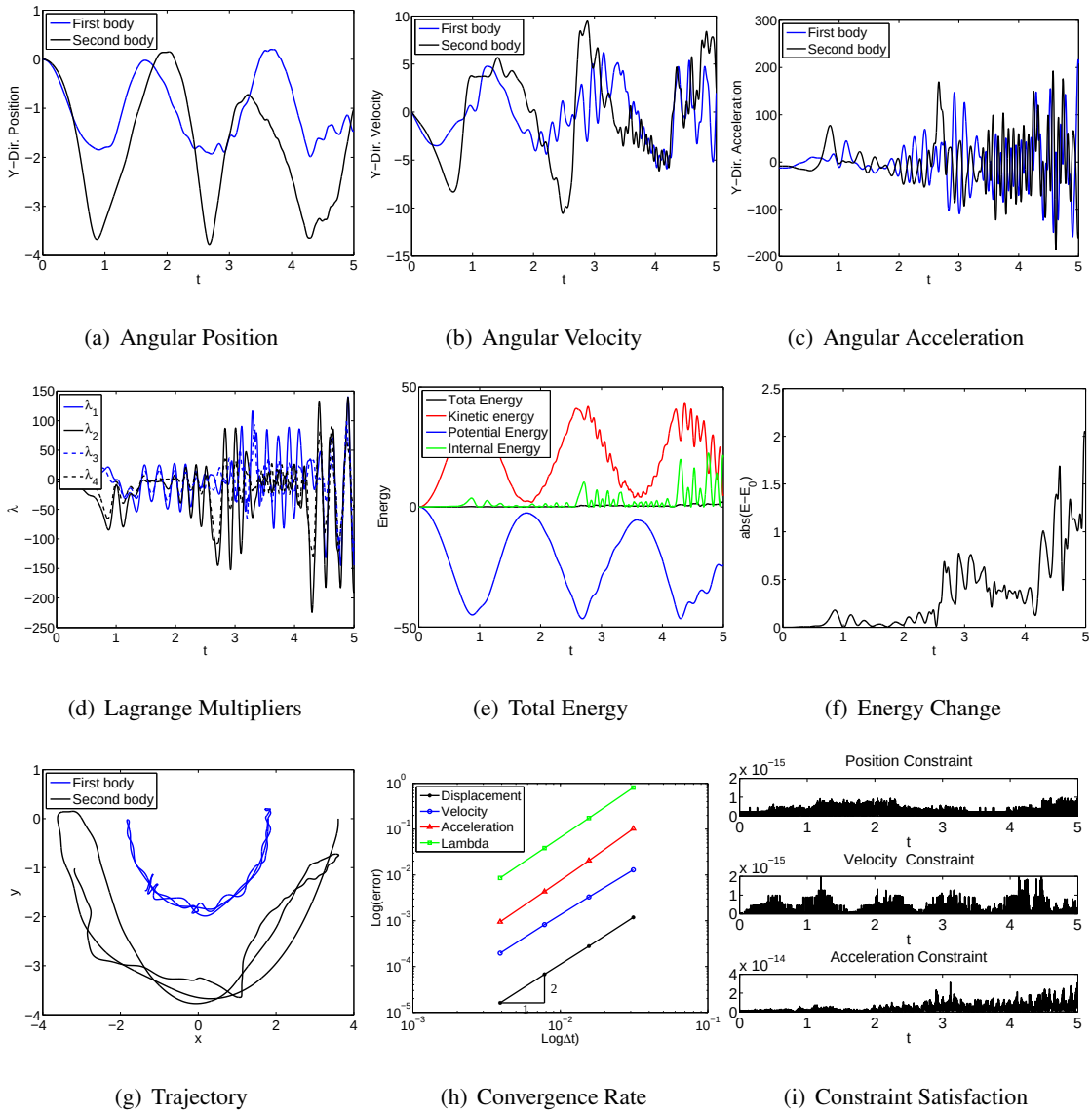


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.222: Double bar pendulum with Projection method: U0(1,1,0).

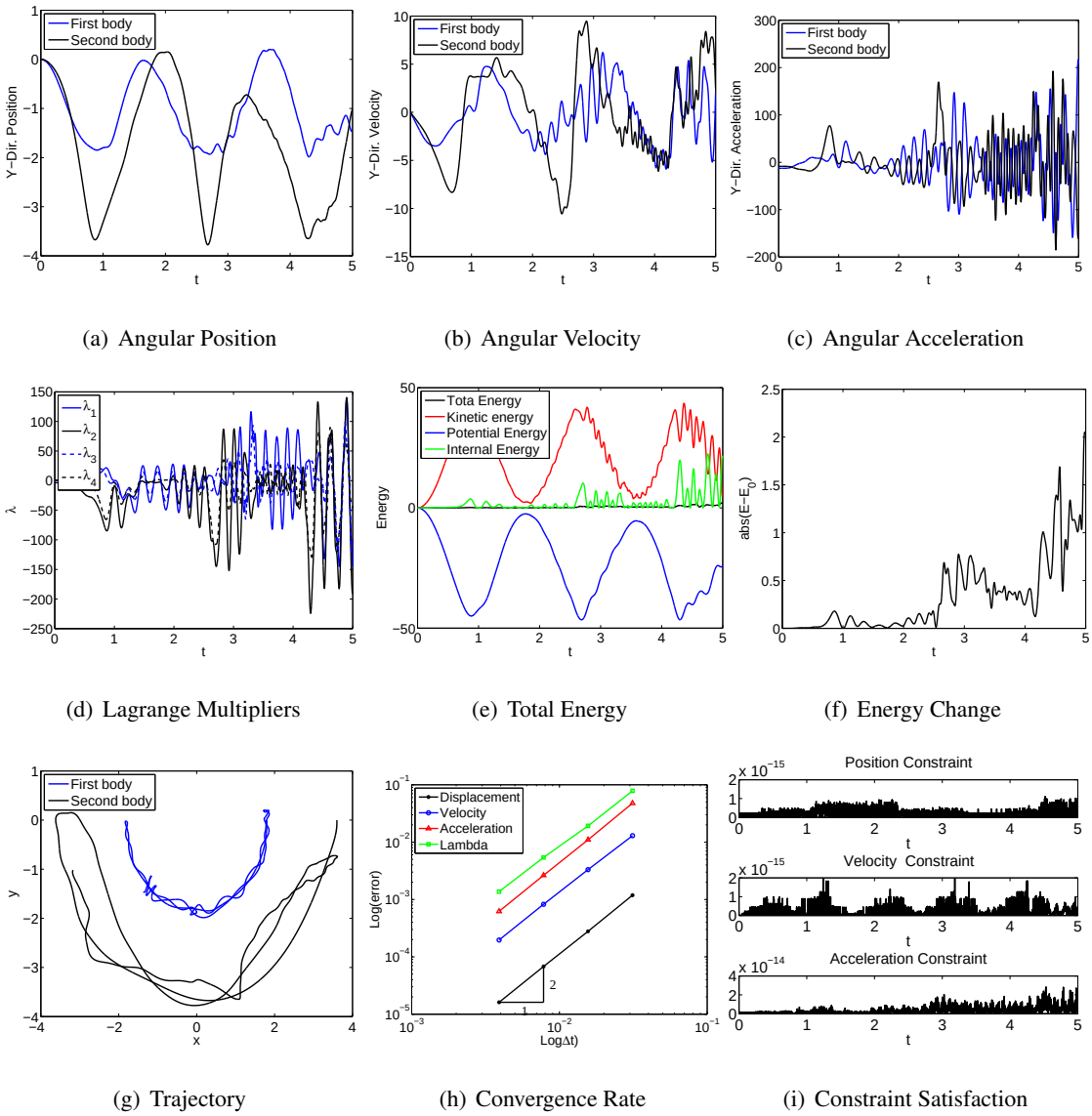


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.223: Double bar pendulum with Projection method: V0(1,1,0).

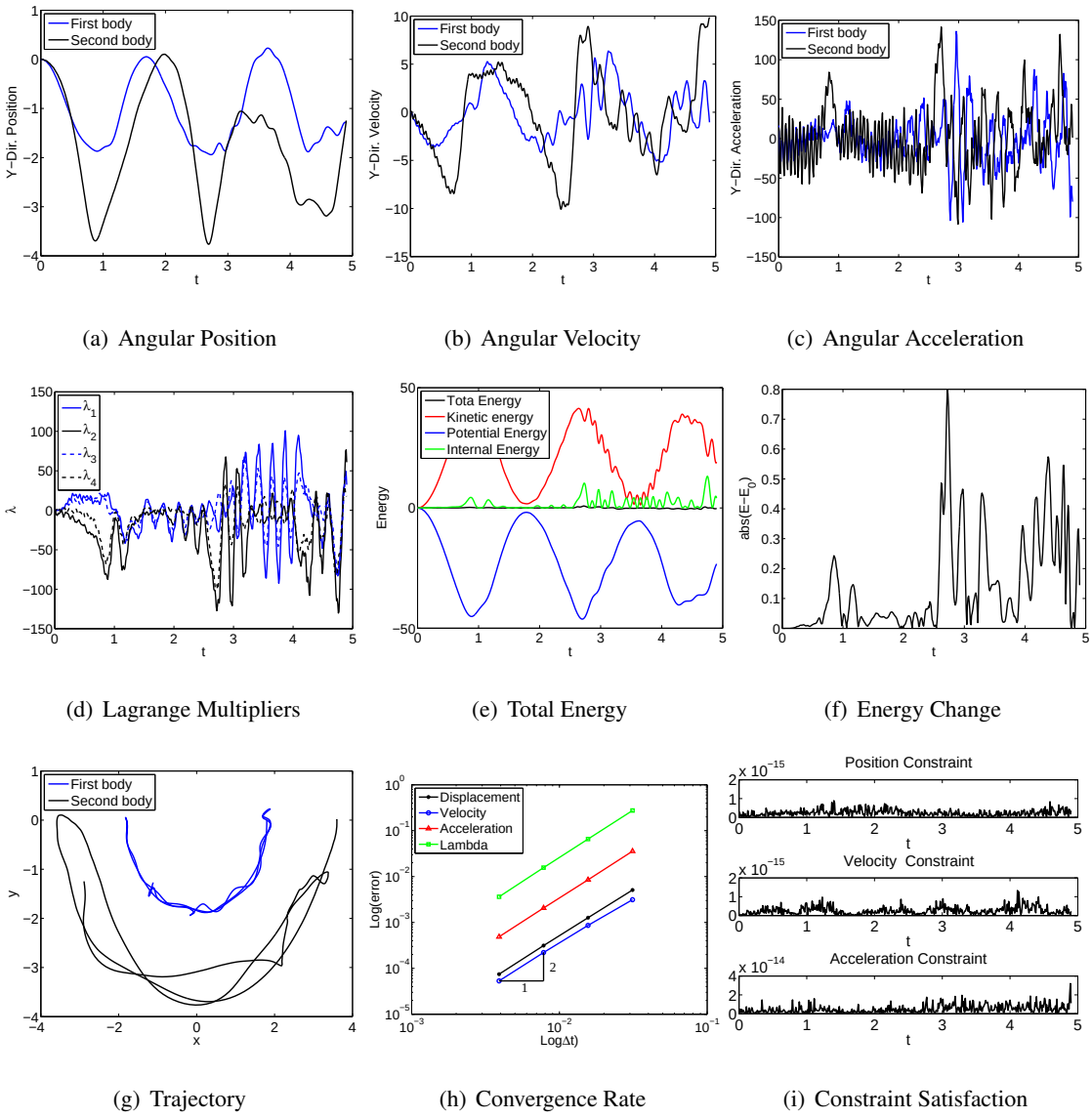


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.224: Double bar pendulum with Projection method: U0V0(1,1,1).

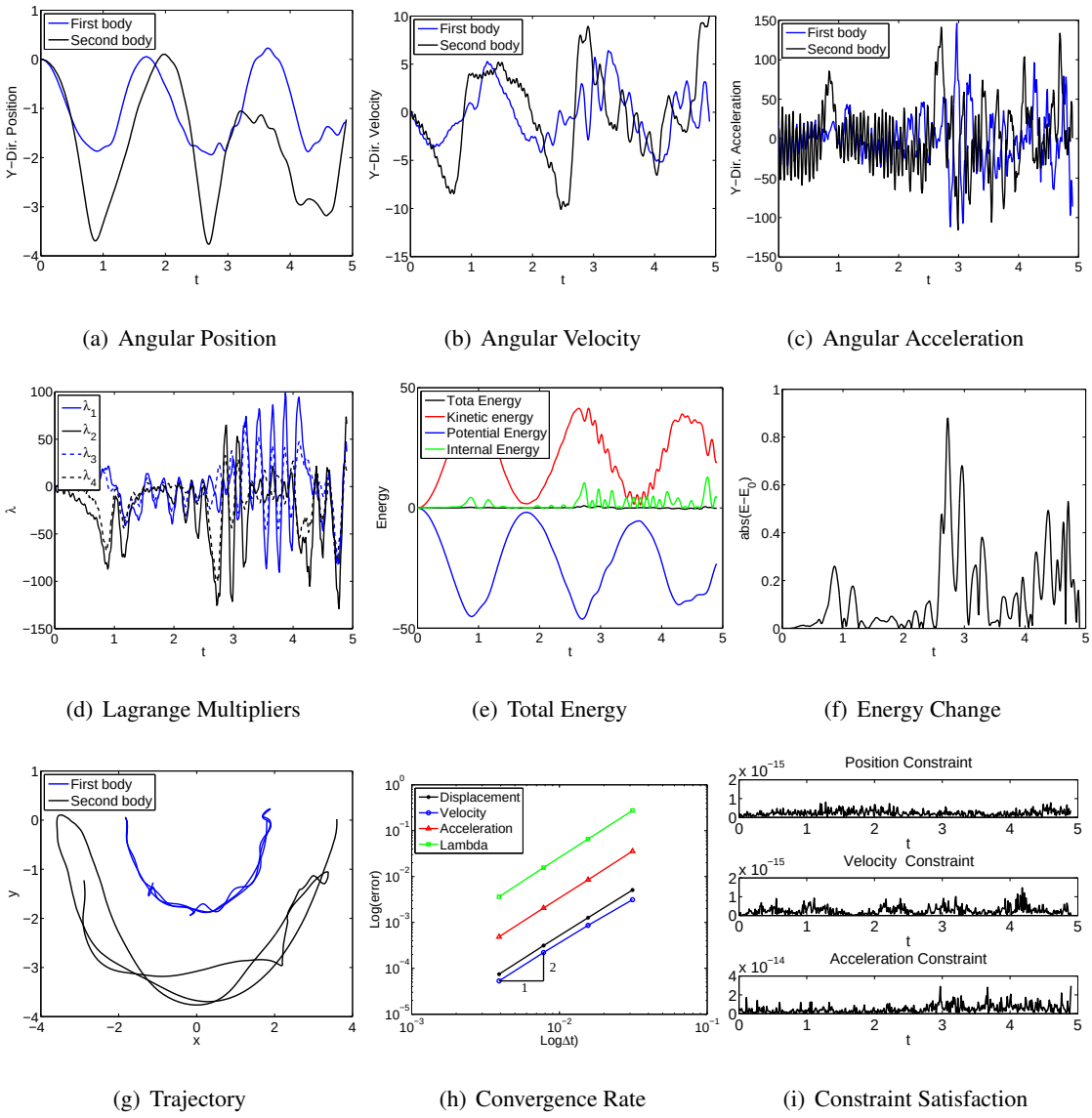


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.225: Double EB pendulum with Projection method: U0(1,1,0).

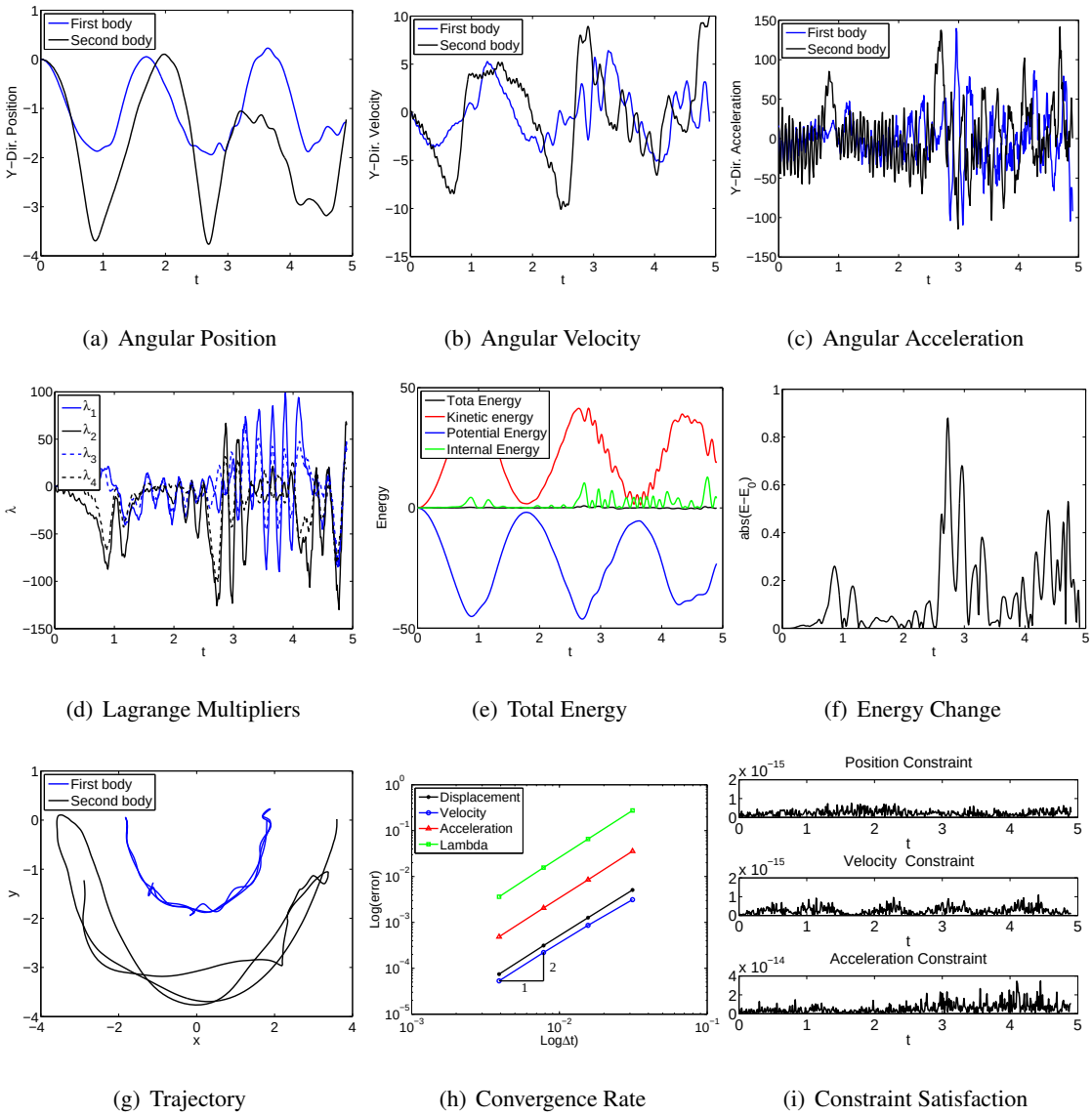


(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.226: Double EB pendulum with Projection method: V0(1,1,0).



(a) Angular Position (b) Angular Velocity (c) Angular Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.227: Single EB pendulum with Projection method: U0V0(1,1,1).

4.2.3 Disscussion

Table 4.3: Computational cost of quasi-rigid double pendulum with bar element (iterations/Δt)

Index	Algorithm	IRF	ANCF-S	FRF
3	U0(1,1,0)	Fail	67.43	4.91*
	V0(1,1,0)	Fail	67.41	4.91
	U0V0(1,1,1)	Fail	64.86	4.82
2	U0(1,1,0)	Fail	67.43	4.93
	V0(1,1,0)	Fail	67.41	4.93
	U0V0(1,1,1)	Fail	64.86	4.88
1	U0(1,1,0)	Fail	67.39	4.96
	V0(1,1,0)	Fail	67.40	4.98*
	U0V0(1,1,1)	Fail	64.85	4.92

Table 4.4: Computational cost of flexible double pendulum with bar element (iterations/Δt)

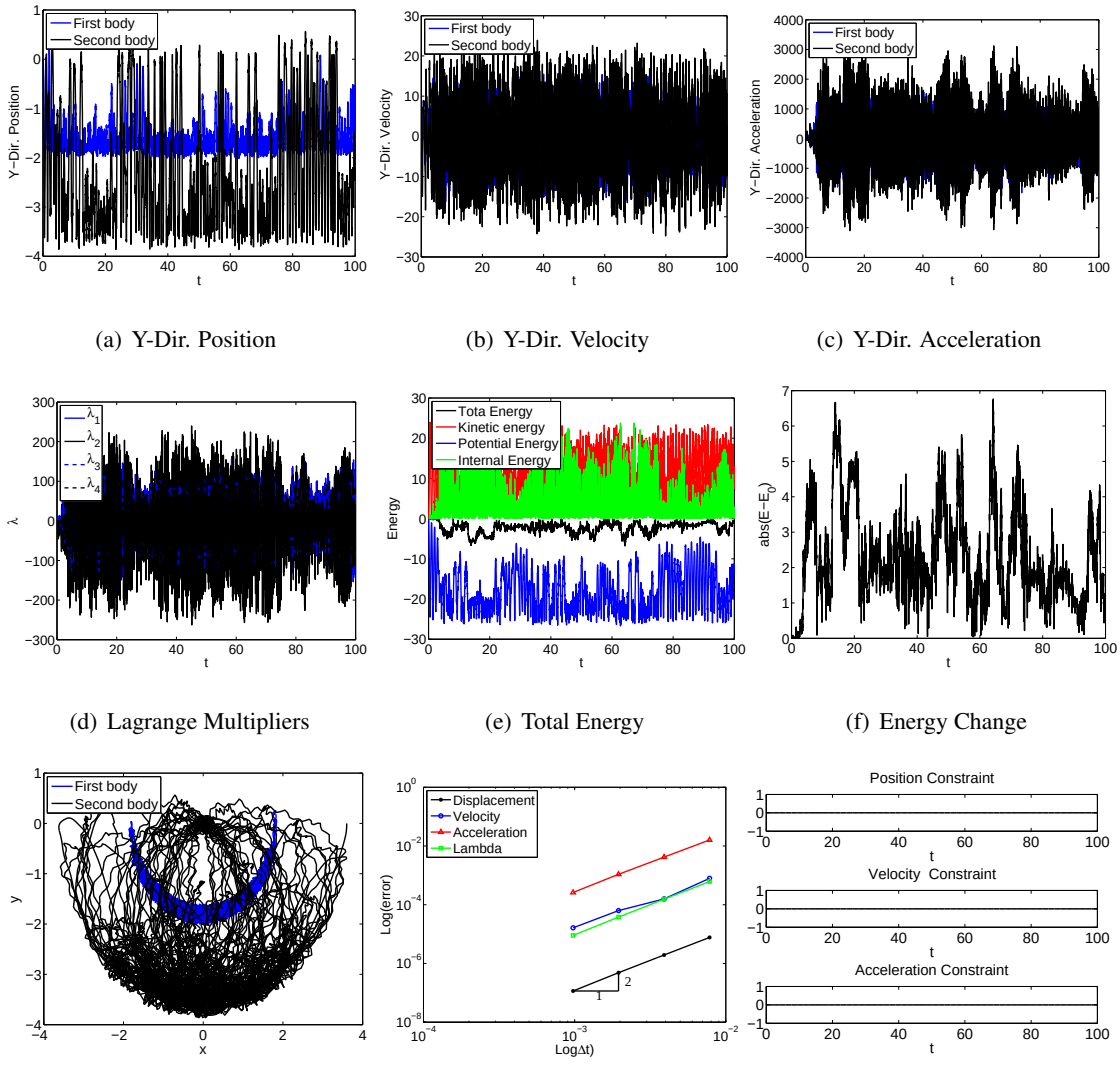
Index	Algorithm	IRF	ANCF-S	FRF
3	U0(1,1,0)	6.02	5.62	5.43*
	V0(1,1,0)	6.02	5.59	5.43
	U0V0(1,1,1)	5.88	5.60	5.34
2	U0(1,1,0)	6.02	5.60	5.48
	V0(1,1,0)	6.02	5.59	5.50
	U0V0(1,1,1)	5.88	5.54	5.40
1	U0(1,1,0)	6.02	5.61	5.53
	V0(1,1,0)	6.02	5.60	5.61*

	U0V0(1,1,1)	5.88	5.49	5.56

Note:* denotes drift effect occurs within 5 seconds and unstable phenomenon.

From the results of the single pendulum with bar element, one can see that there is no big difference between IRF/FRF/ANCF-S approaches from the dynamic response. However, from the energy point point of view, ANCF-S and IRF show better energy conservation features (energy is bounded). Also, there are differences in the dynamic response in single pendulum with beam element. Since the bending rigidity EI is extremely small, large deformation simulation causes FRF to not converge. On the other hand, ANCF-S and IRF still work very well. This results is consistent with the theory in that FRF is suitable for small deformation simulation. Also, simulation results from single formulation methods between Index 3, Index 2 and Index 1 are identical to each other (eg. Fig. 4.53 to Fig. 4.55). Although we formulated constraints in DAE form, it shows that linear formulation of constraint equation implies exact constraint satisfaction for position, velocity, acceleration constraint; and further constraint stabilizing technique are not necessary. In the simulation of double pendulum, we compare with both rigidity cases and seek to compare IRF, ANCF-S, and FRF from the result. From Table. 4.3, one can easily find that in the bar element, IRF fails to converge and the cost of ANCF is too expensive in the quasi-rigid bar case with bar element. Only the simulation under FRF shows a good efficiency for 5 iterations. For the general flexible case, from Table. 4.4, the cost for each formulation method (IRF/ANCF-S/FRF) is very close to each other. However, the formulation of constraint equation from FRF causes nonlinearity, and this results in non-satisfaction in each constraint level in each Index. As a result, projection method or Baumgarte's method is required. In the

simulation of rigid body dynamics, only V0(1,1,0) results in long time stable solution without drift. All cases in long time simulation for the double pendulum with bar element without projection are shown below:

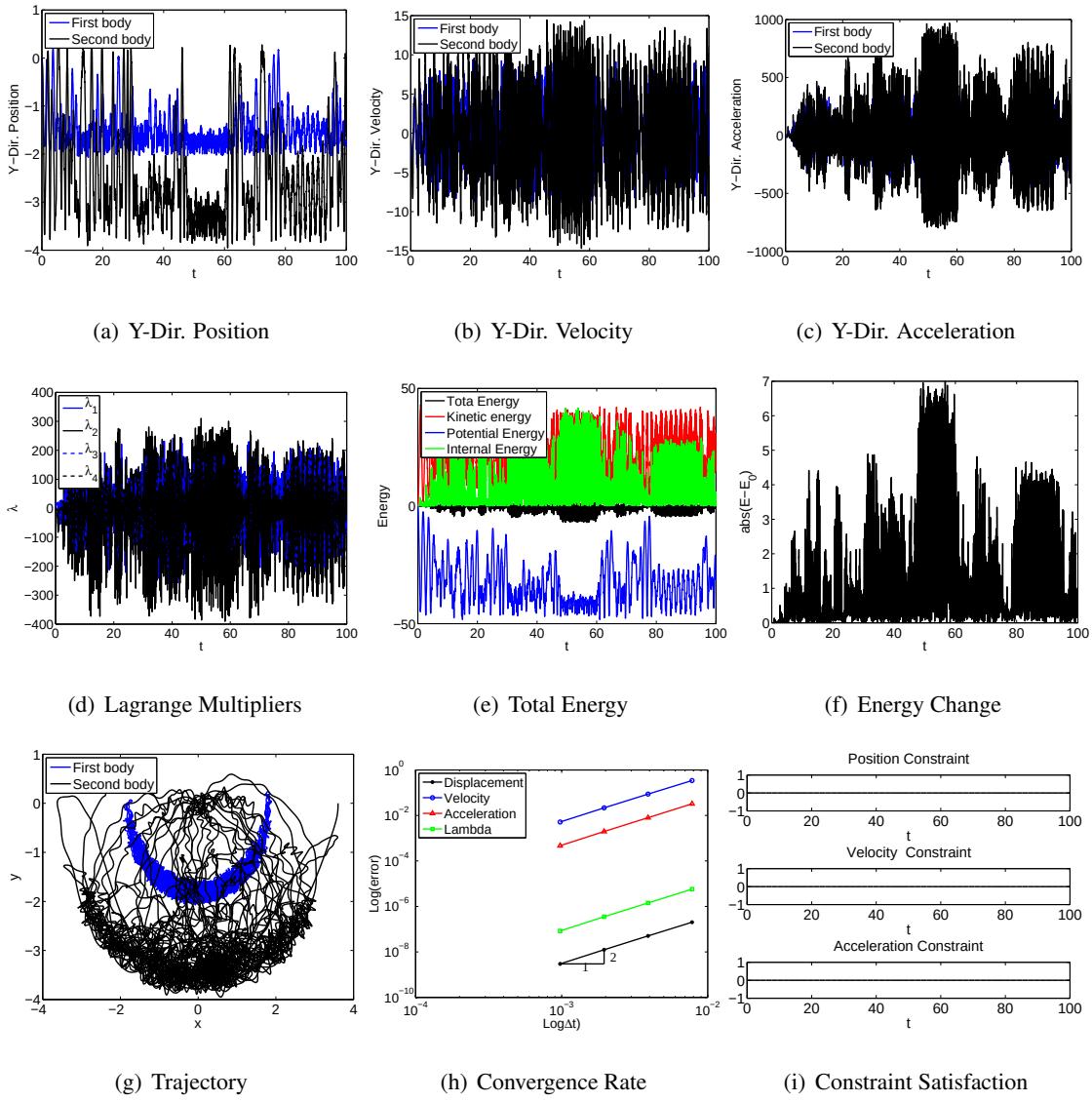


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.228: Long time simulation for double flexible pendulum with bar element in IRF: U0(1,1,0) - Index 3.

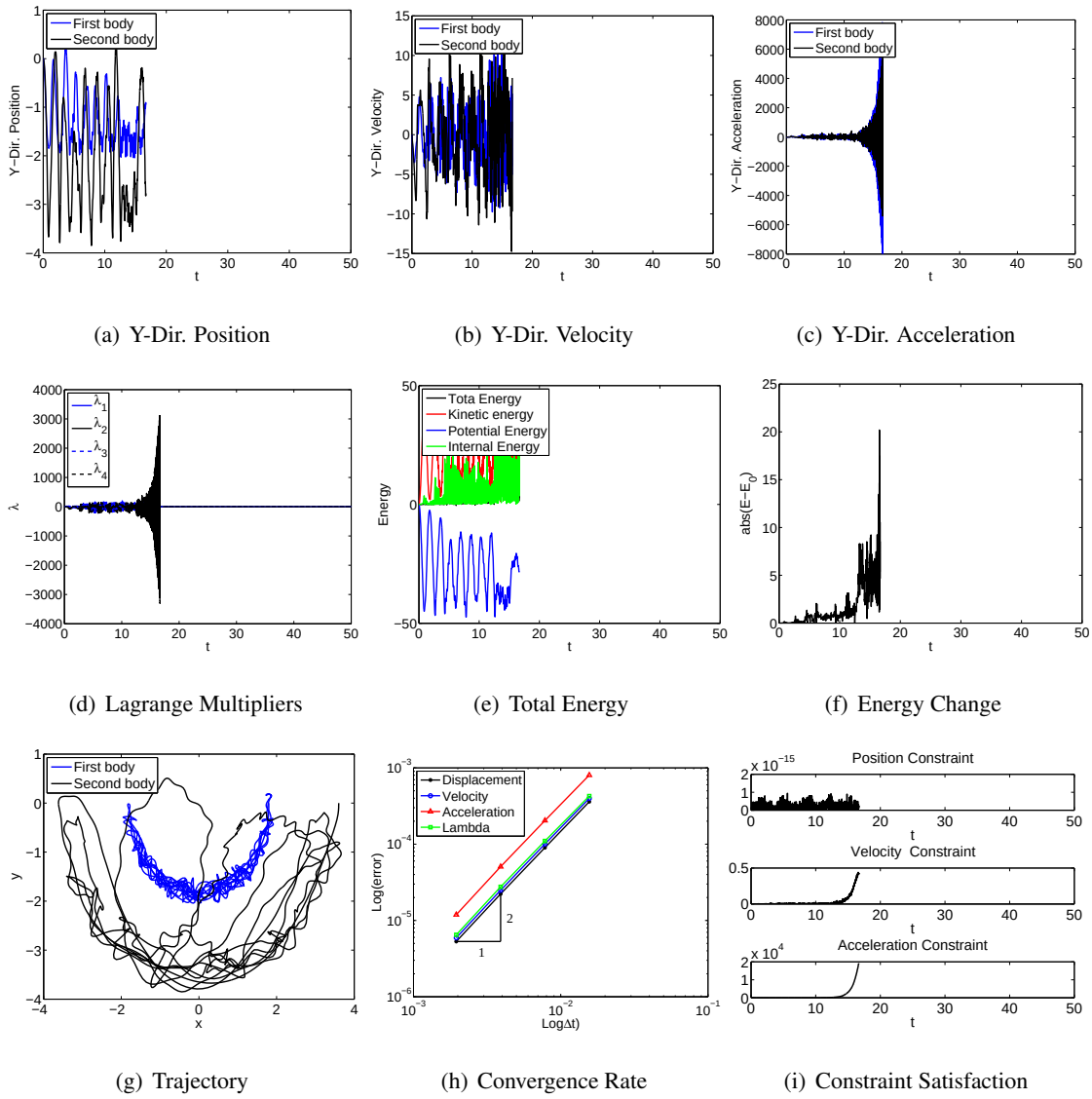


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.229: Long time simulation for double flexible pendulum with bar element in ANCF: U0(1,1,0) - Index 3.

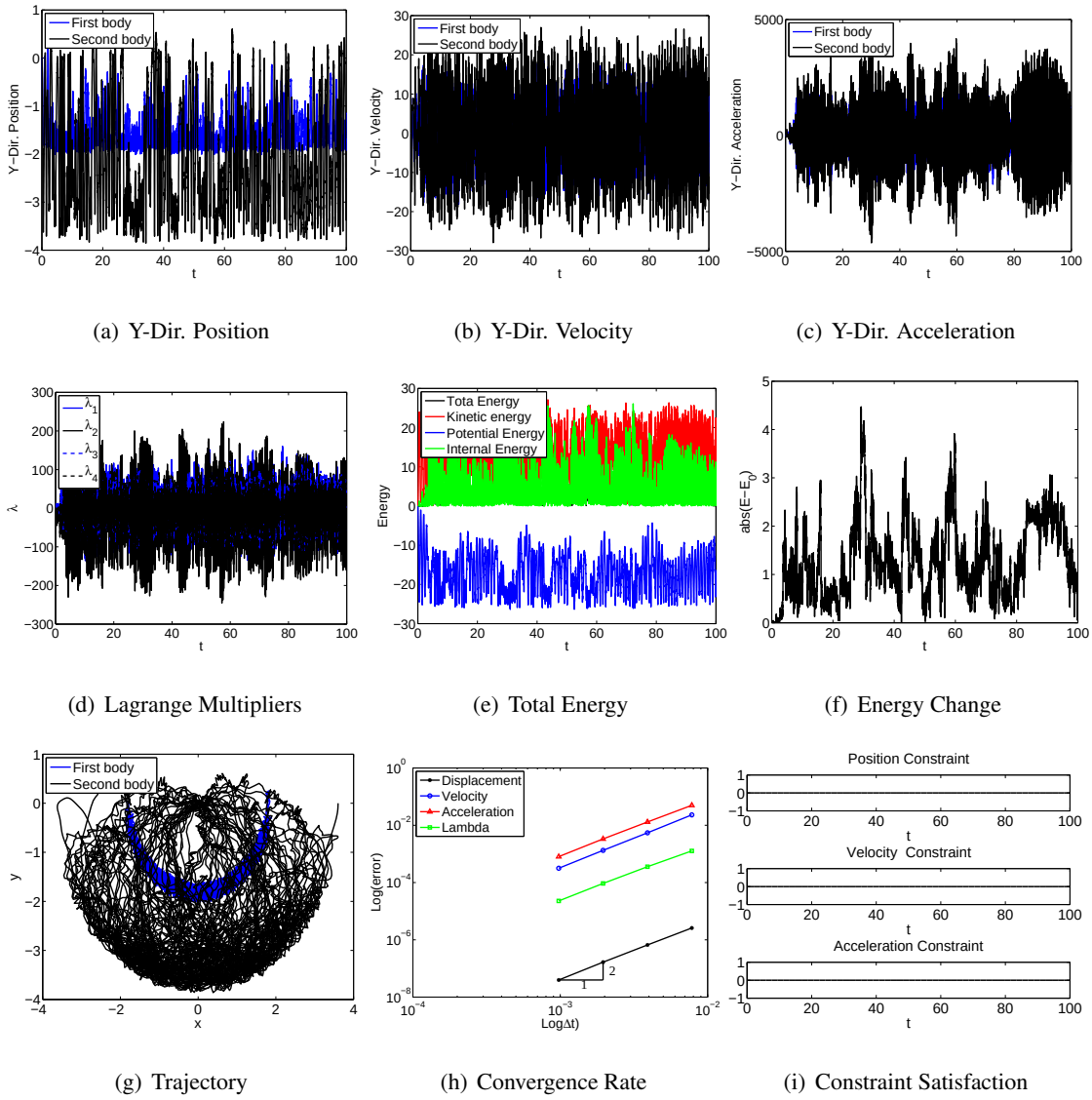


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.230: Long time simulation for double flexible pendulum with bar element in FRF: U0(1,1,0) - Index 3.

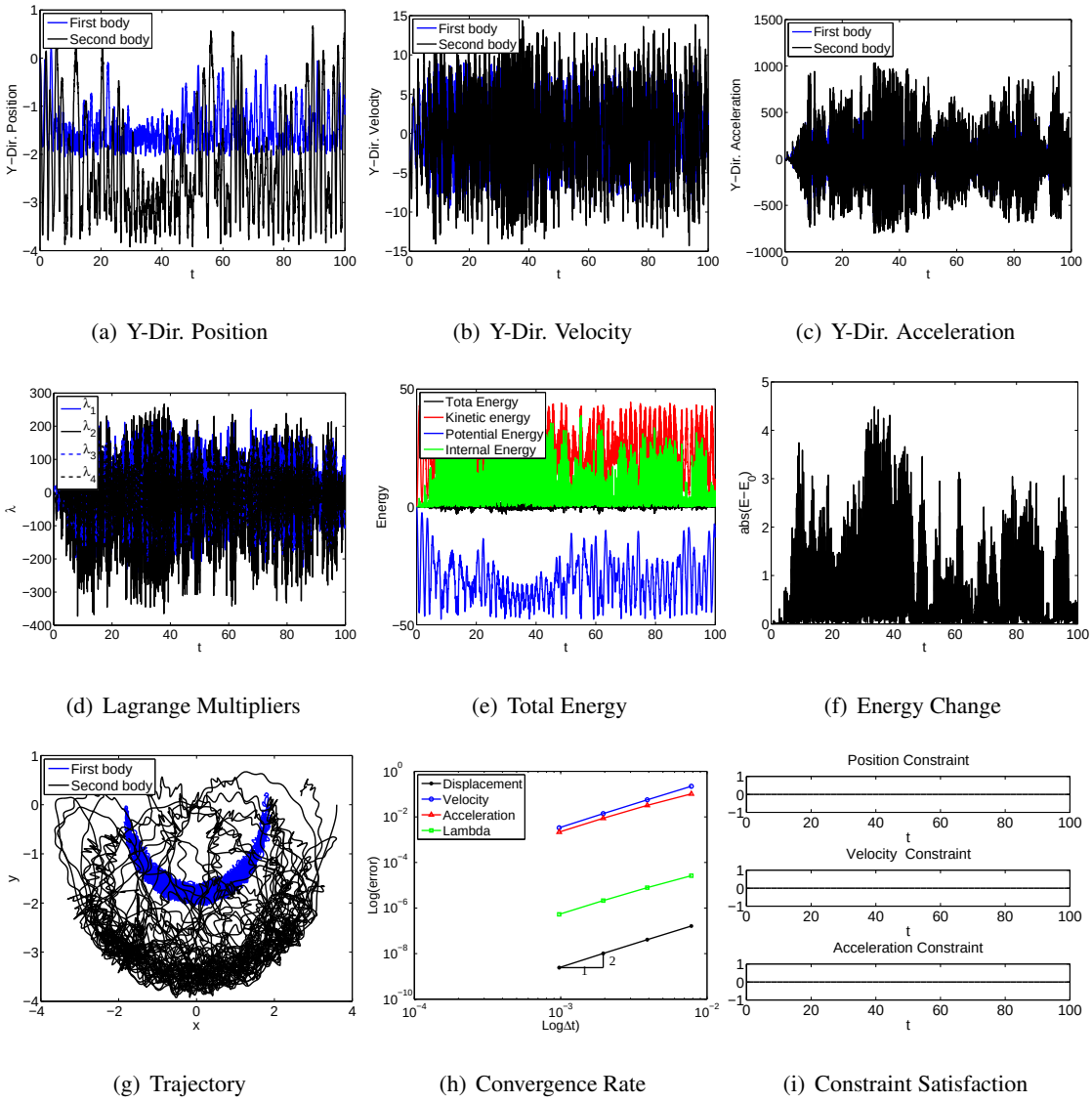


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.231: Long time simulation for double flexible pendulum with bar element in IRF: V0(1,1,0) - Index 3.

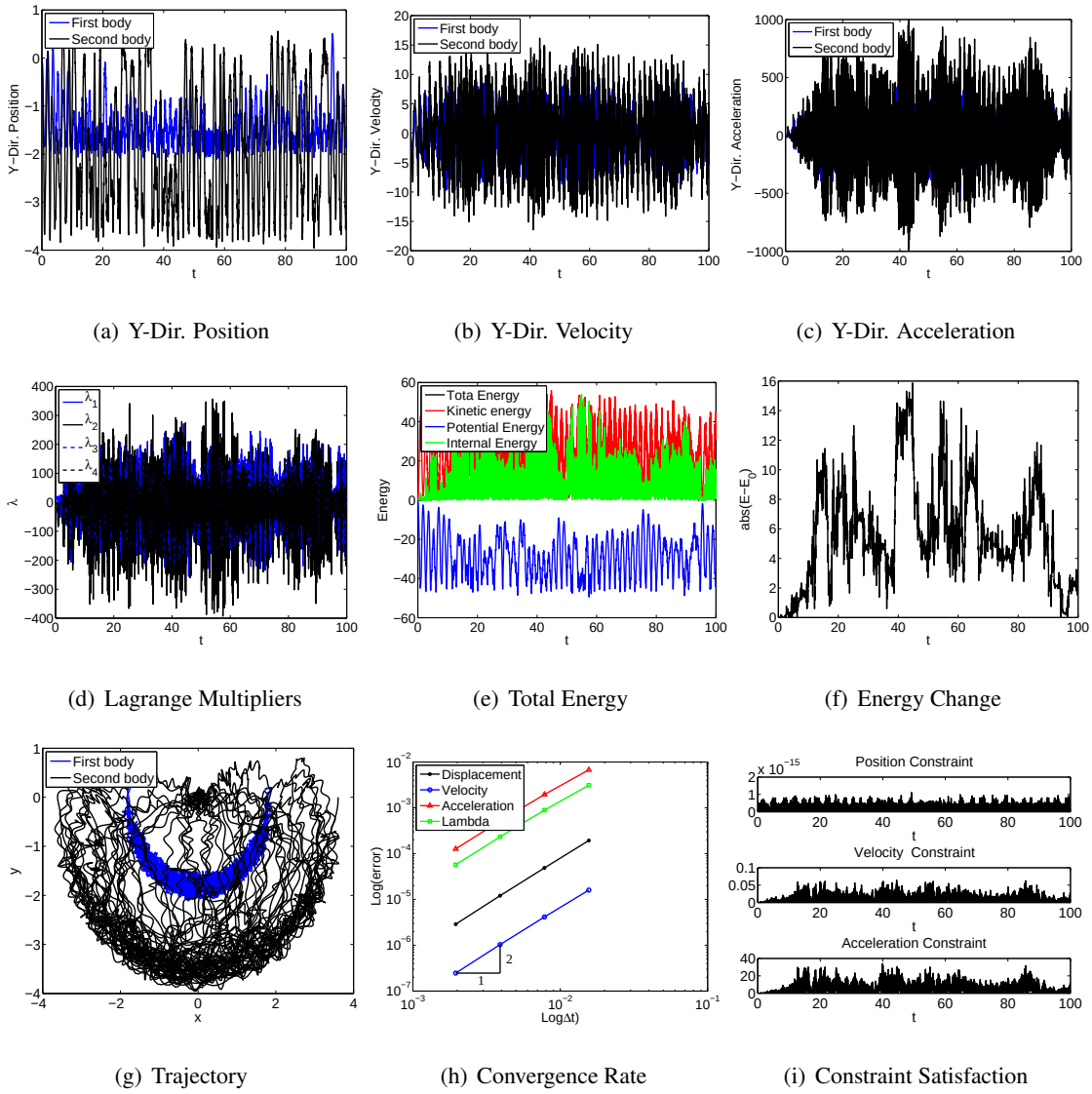


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.232: Long time simulation for double flexible pendulum with bar element in ANCF: V0(1,1,0) - Index 3.

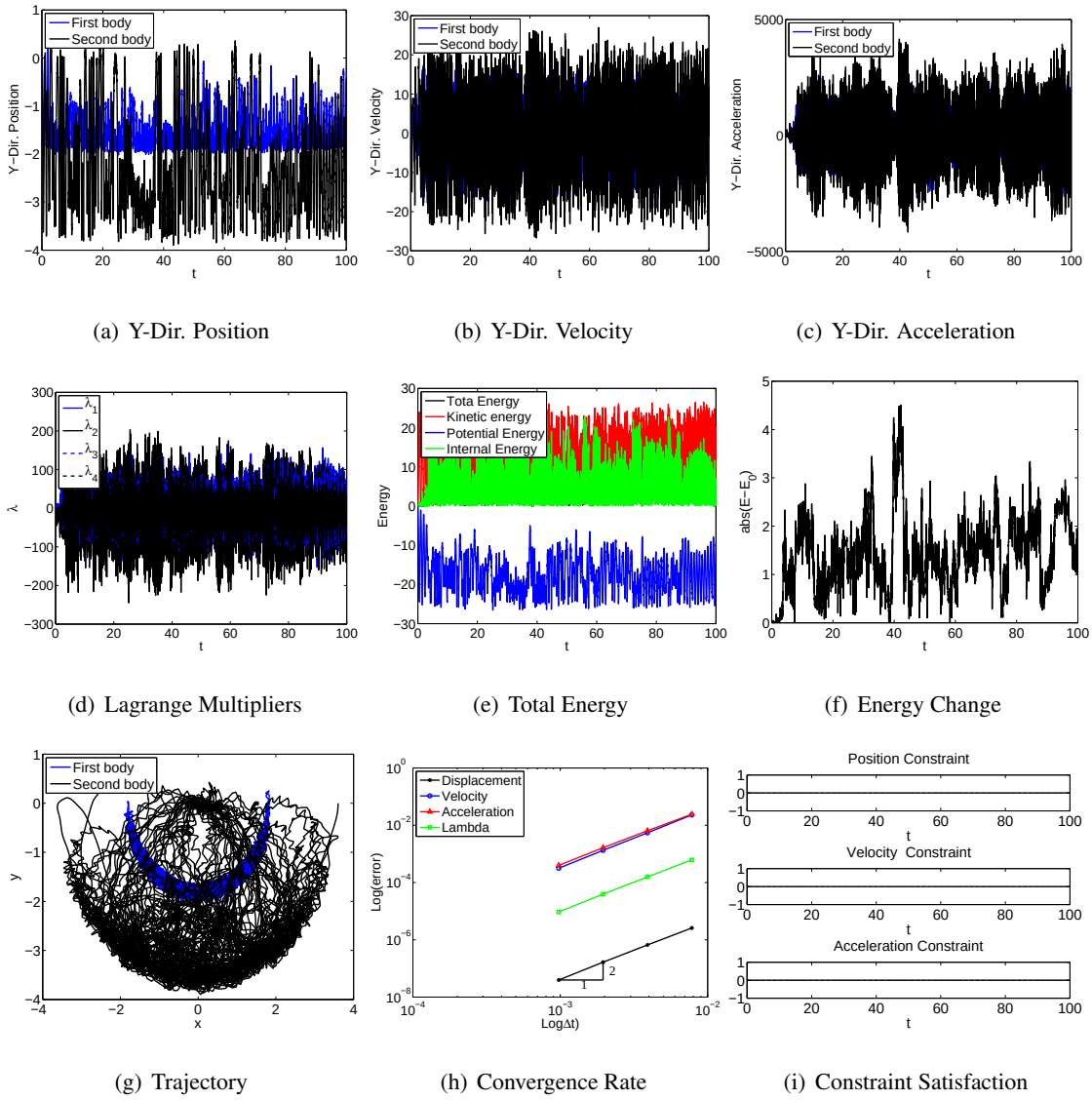


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.233: Long time simulation for double flexible pendulum with bar element in FRF: V0(1,1,0) - Index 3.

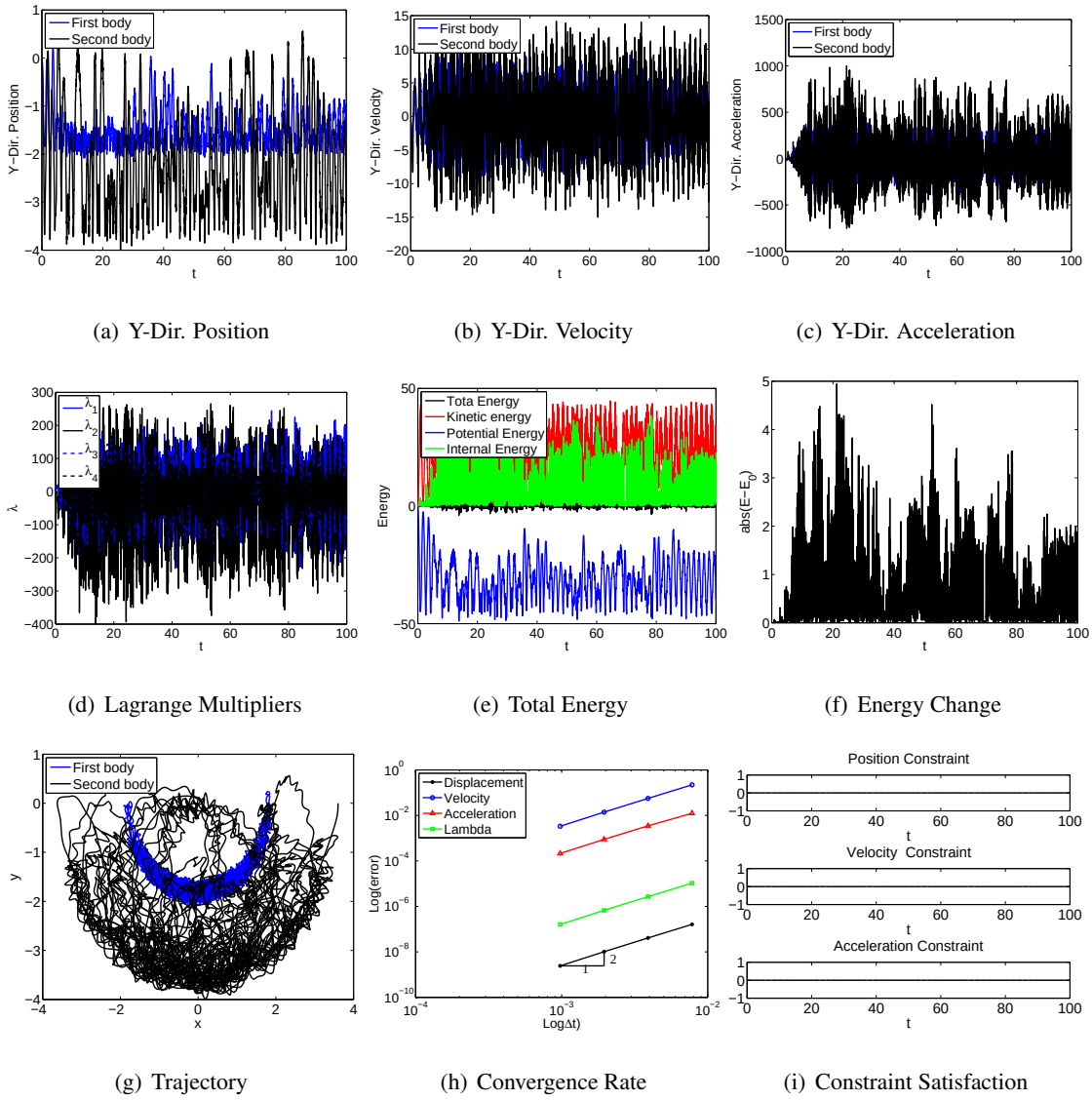


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.234: Long time simulation for double flexible pendulum with bar element in IRF: U0V0(1,1,1) - Index 3.

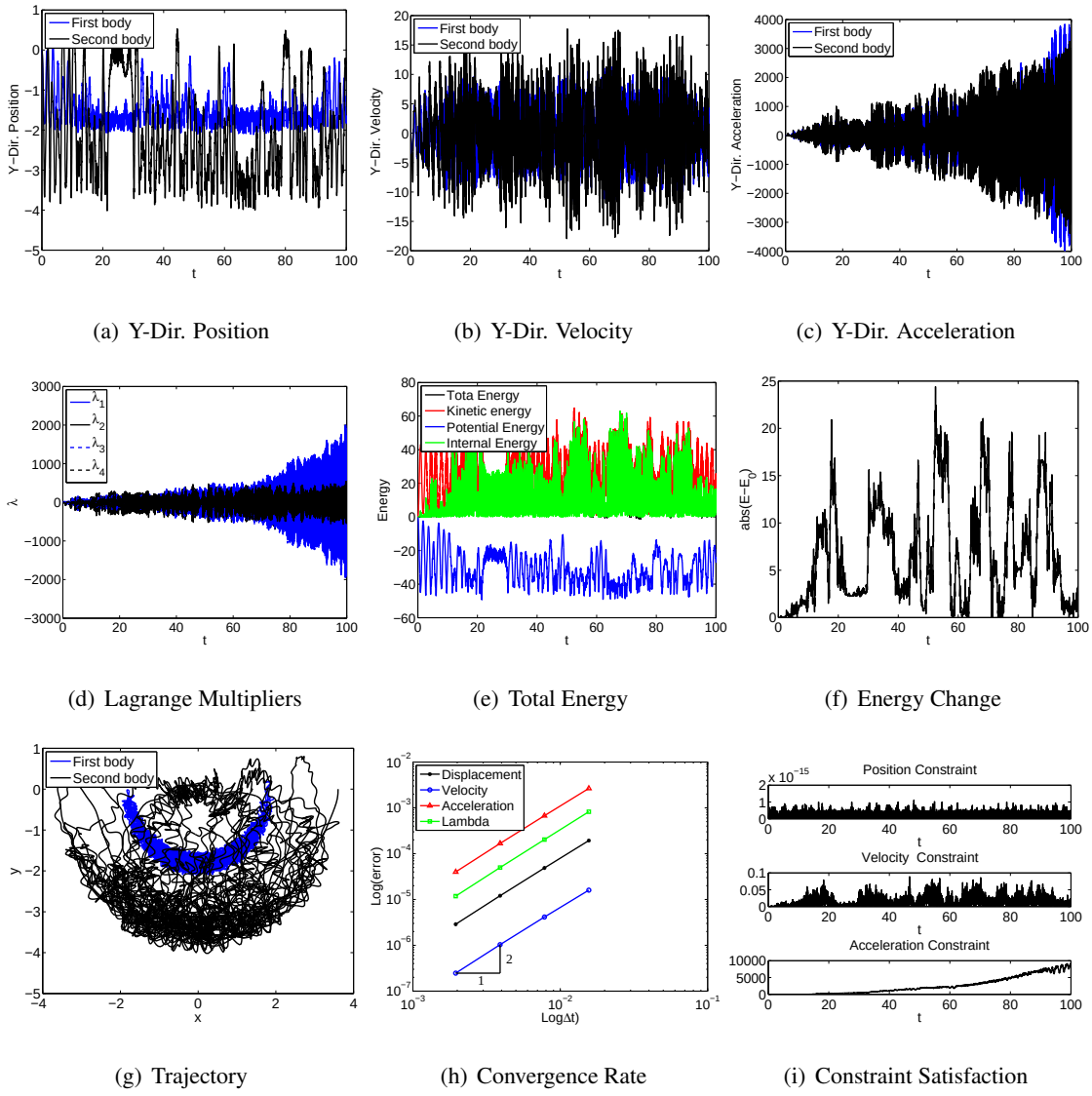


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.235: Long time simulation for double flexible pendulum with bar element in ANCF: U0V0(1,1,1) - Index 3.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

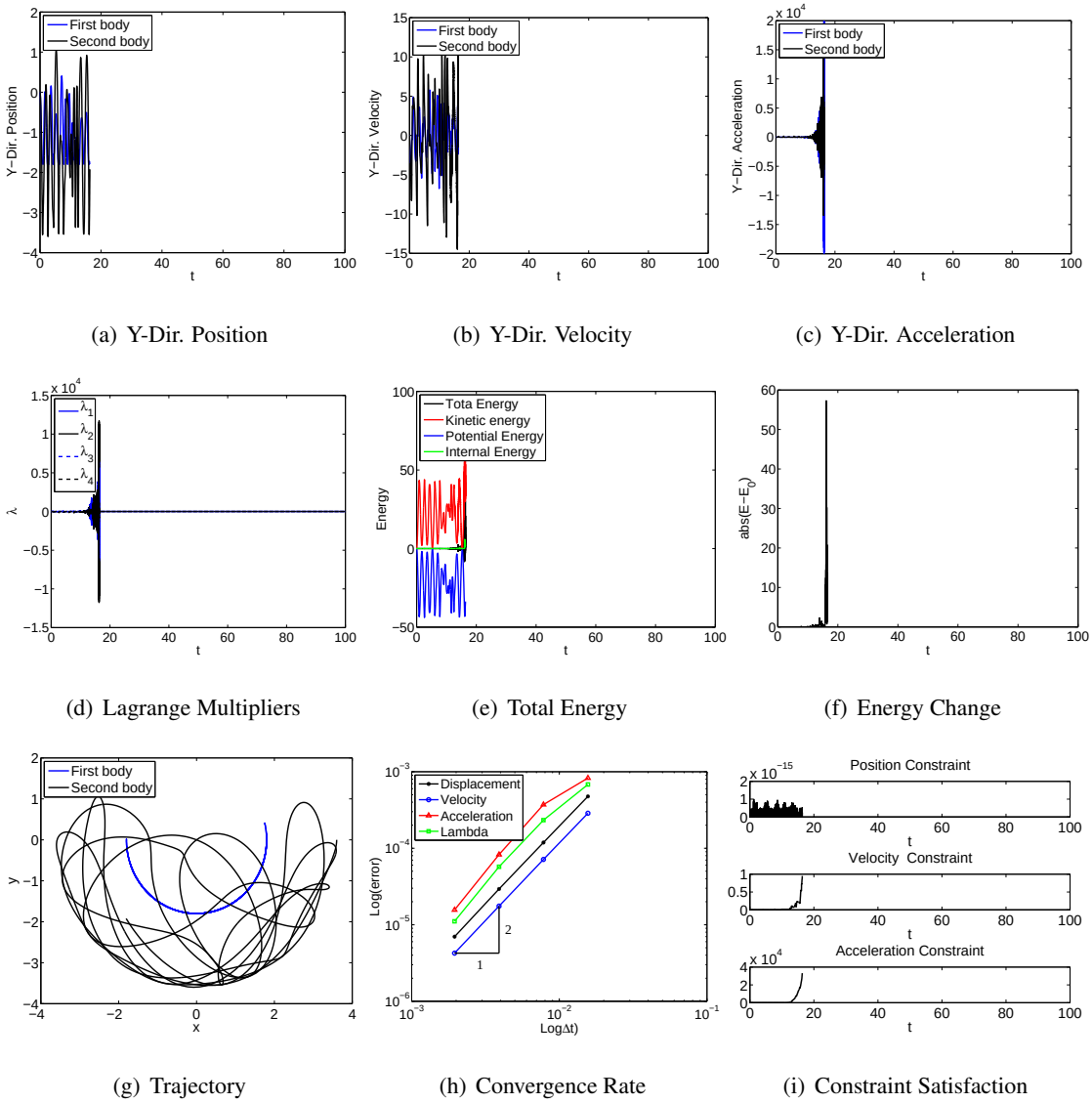
(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.236: Long time simulation for double flexible pendulum with bar element in FRF: U0V0(1,1,1) - Index 3.

Table 4.5: Long time computational cost of flexible double pendulum with bar element (iterations/Δt)

Index	Algorithm	IRF	ANCF-S	FRF
3	U0(1,1,0)	26.78	19.37	Non-Converge
3	V0(1,1,0)	27.94	19.17	8.78
3	U0V0(1,1,1)	27.62	18.34	8.64

For long time simulation without projection, V0(1,1,0) in Index 3 from simulation of rigid body dynamics shows long term stability as well as good feature of energy conservation. Here, we employ it for flexible double pendulum for bar element and compare with each formulation. From Fig. 4.231 to Fig. 4.233, one can easily found that although all of the method can finish long term simulation and result in stable solution with drift effect in constraint. From energy point of view, only IRF/ANCF-S formulation can conserve energy within 5 and energy is well bounded. Then, compare with computational cost in Table. 4.5, ANCF-S is more efficient than IRF approach. As a result, for general flexible body dynamics ANCF seems much better than the other two formulation. Then for quasi-rigid case, FRF is the only option we are able to do since it is less expensive and the good energy conserving feature, the result for quasi-rigid double pendulum in bar element is shown below:

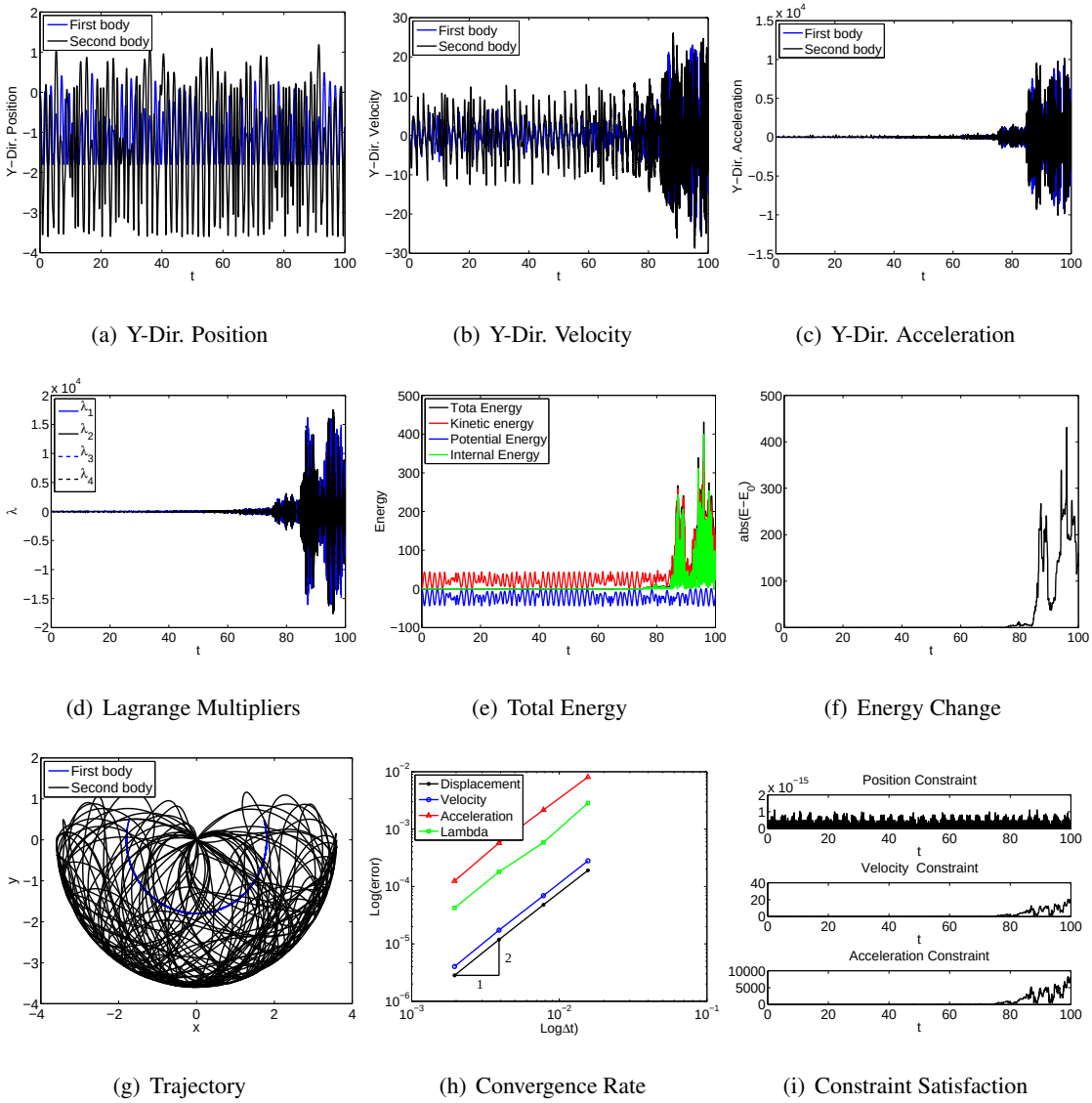


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.237: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 3.

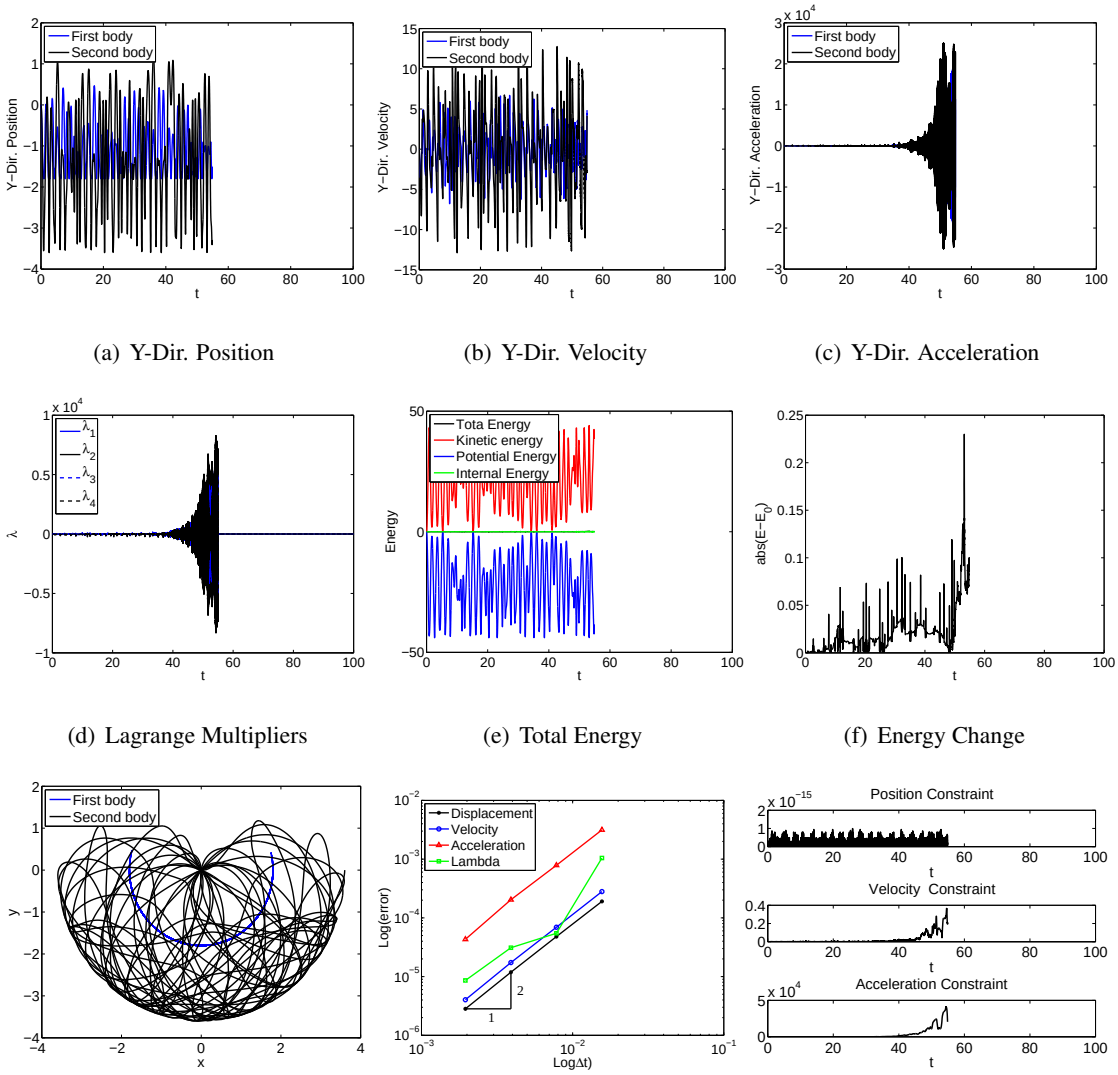


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.238: Long time simulation for double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 3.

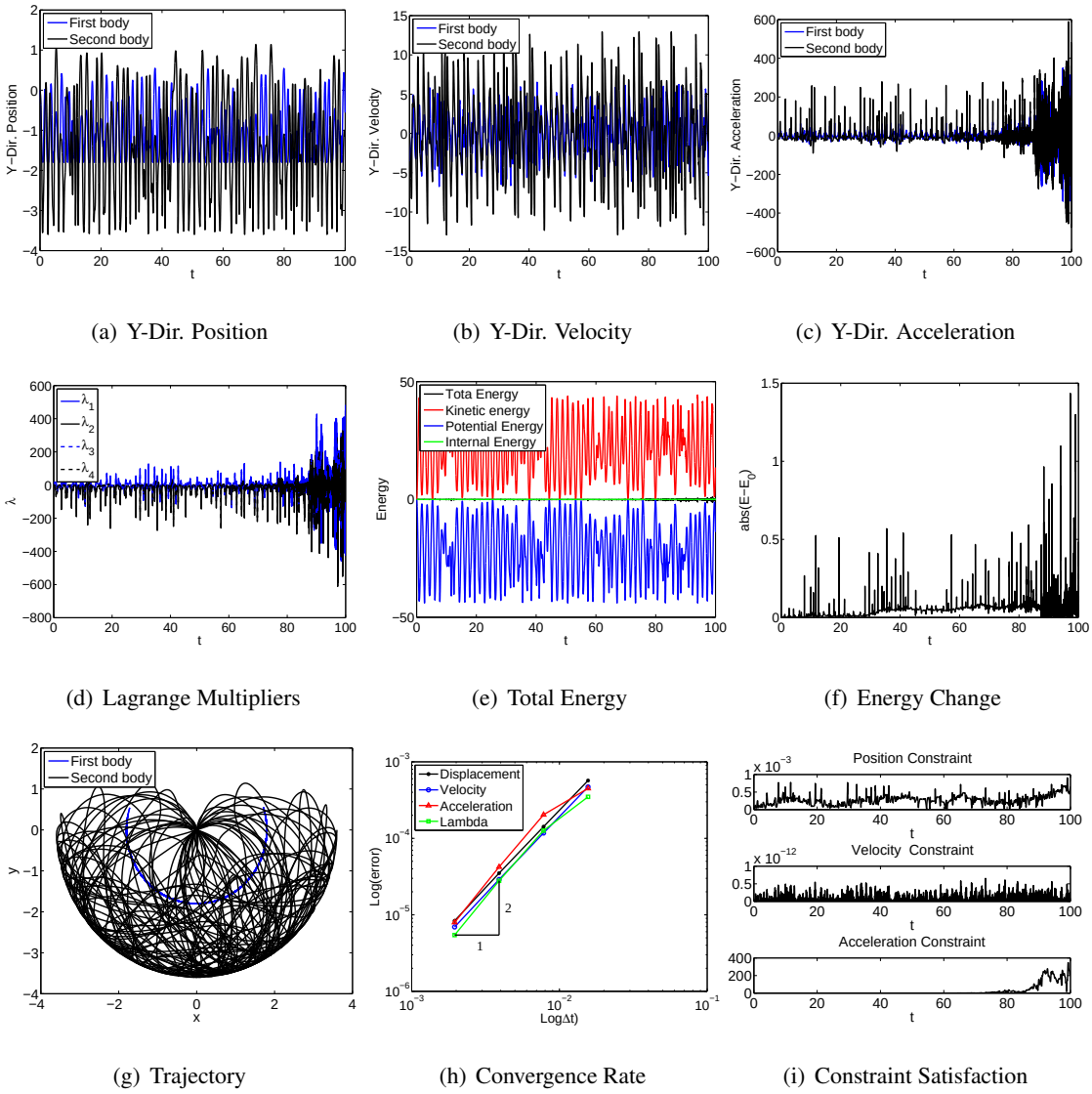


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.239: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 3.

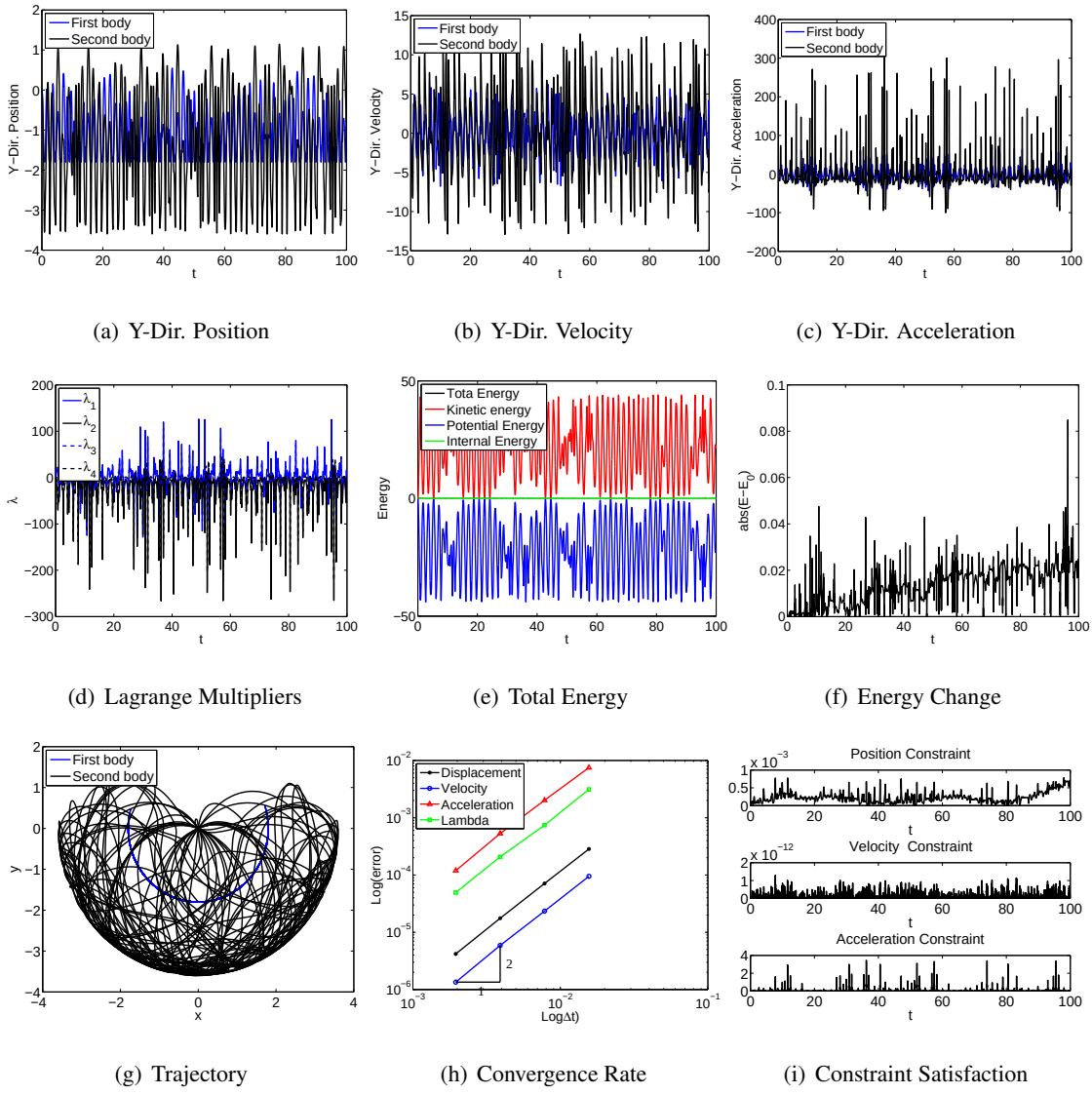


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.240: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 2.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.241: Long time simulation for double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 2.

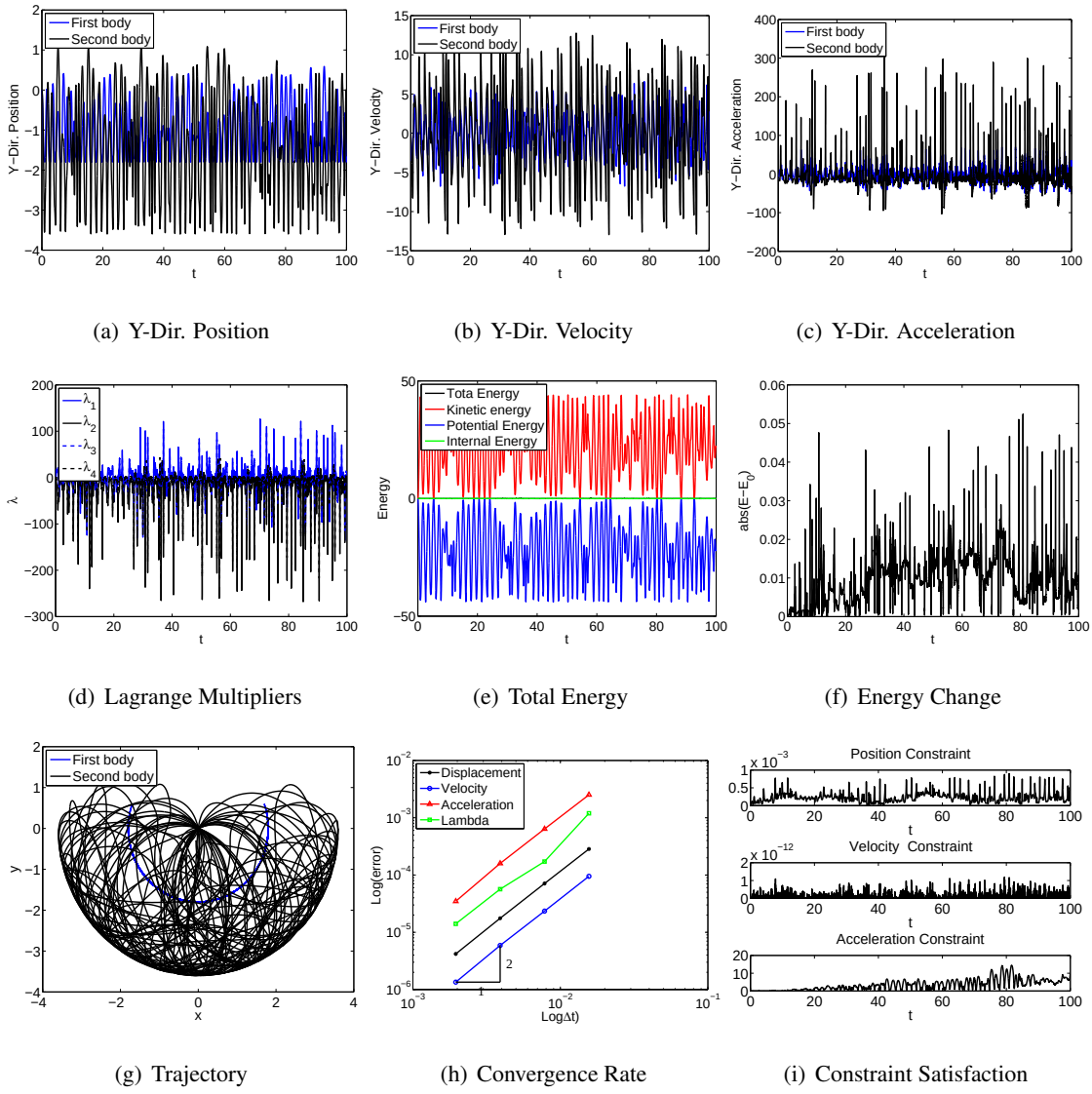
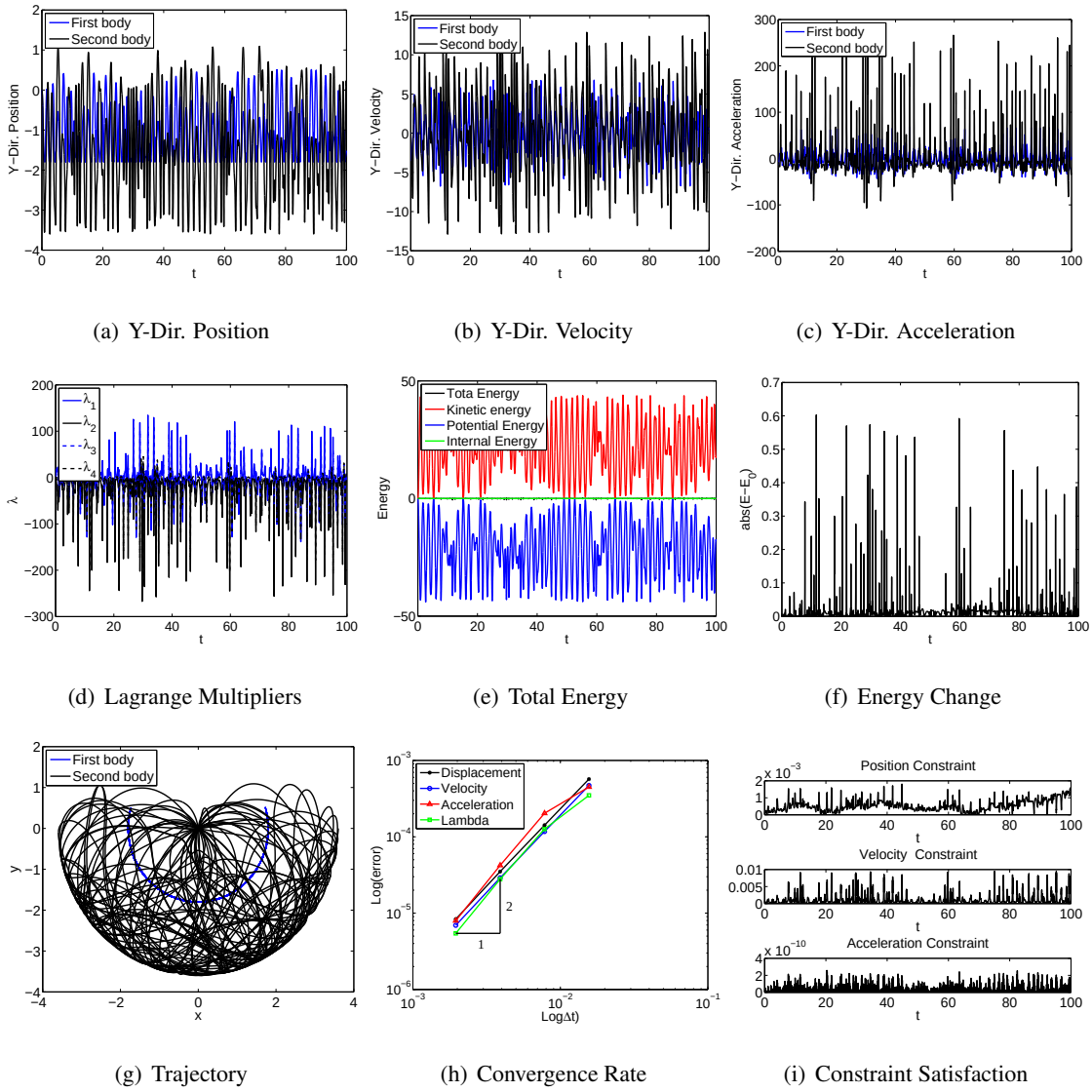


Figure 4.242: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 2.

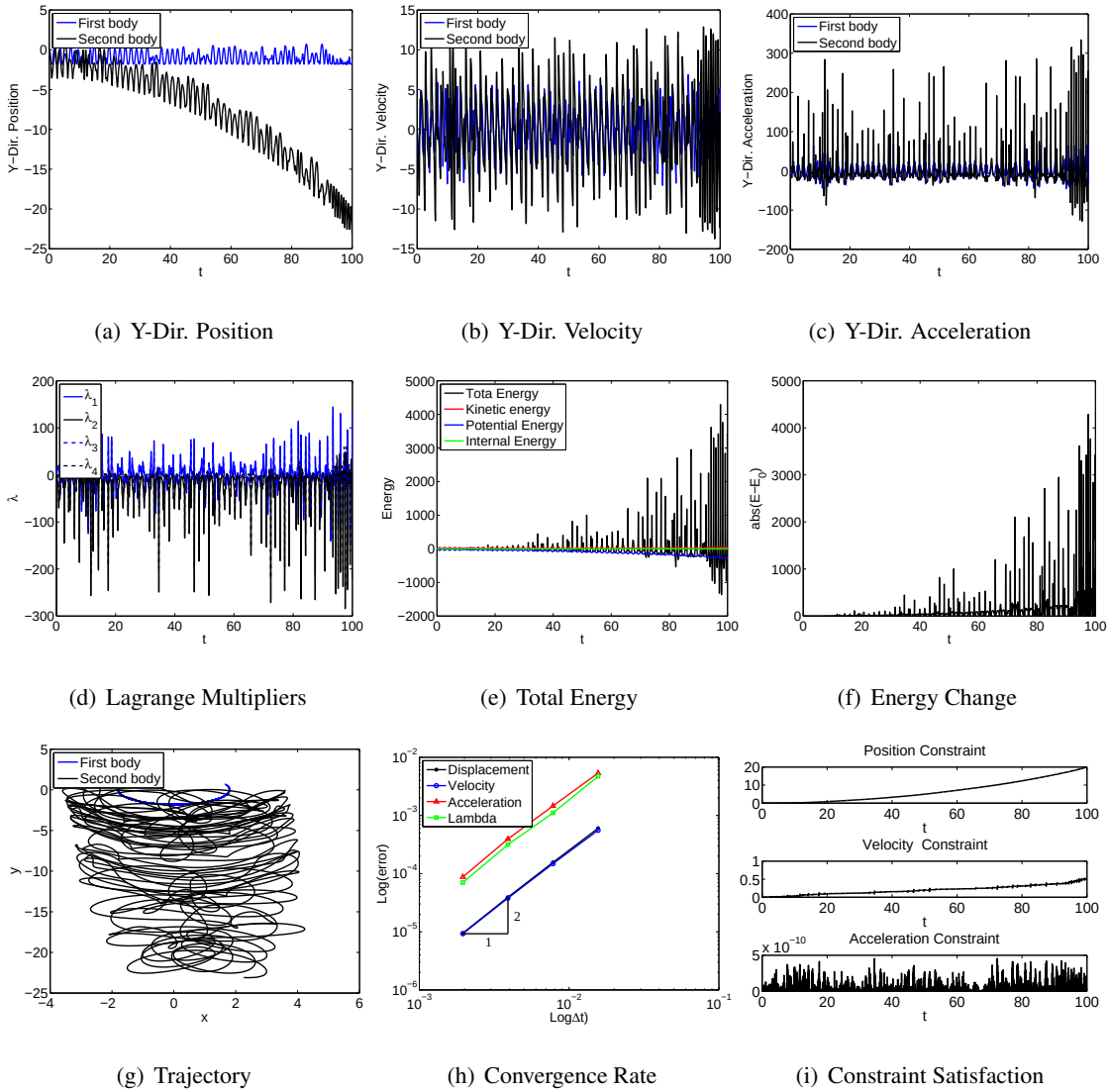


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.243: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 1.

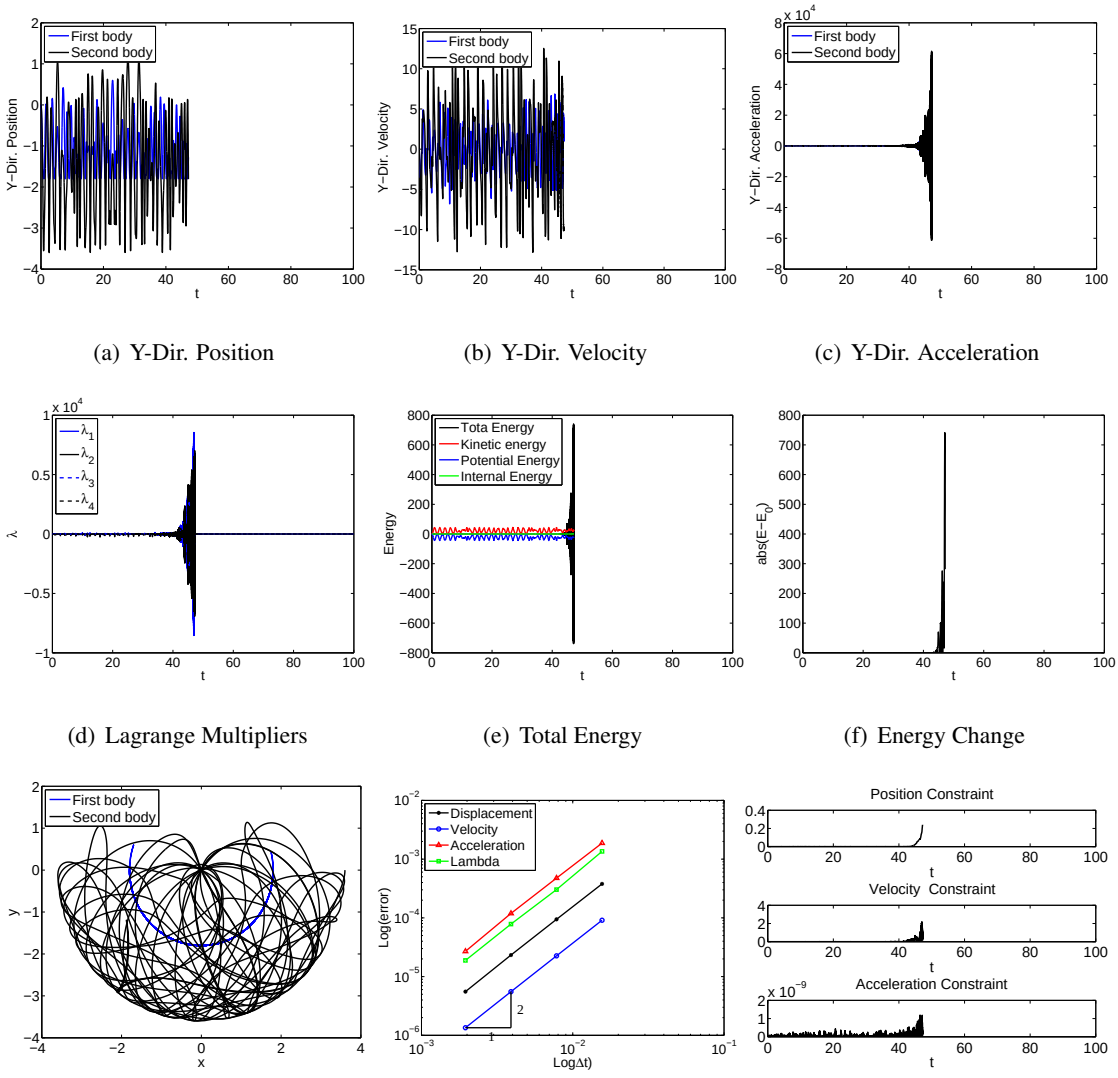


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.244: Long time simulation for double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 1.



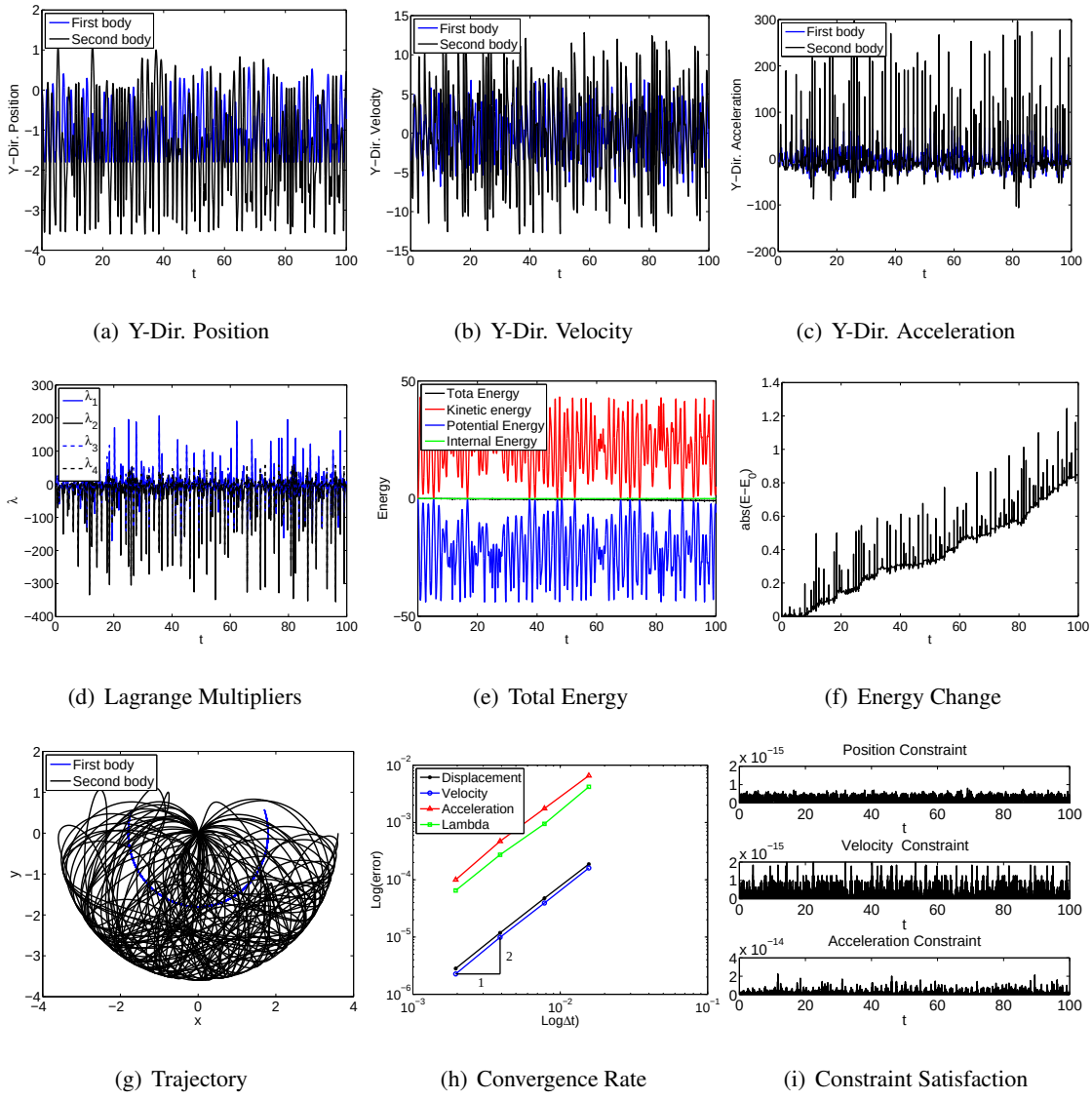
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.245: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 1.

From the results, it is obvious that Index reduction techniques stabilize the system without the projection method. Comparing Fig. 4.240, Fig. 4.241, and Fig. 4.242, without projection, Index 2 simulation is much more robust than Index 1 and Index 3 generally since one obtain stability and also satisfy position constraint upto 10^{-4}. As a result, for general cases, it is recommended to do Index 2 without projection method than the other two. However, projection method for exact constraint satisfaction is essential and the results are shown below:

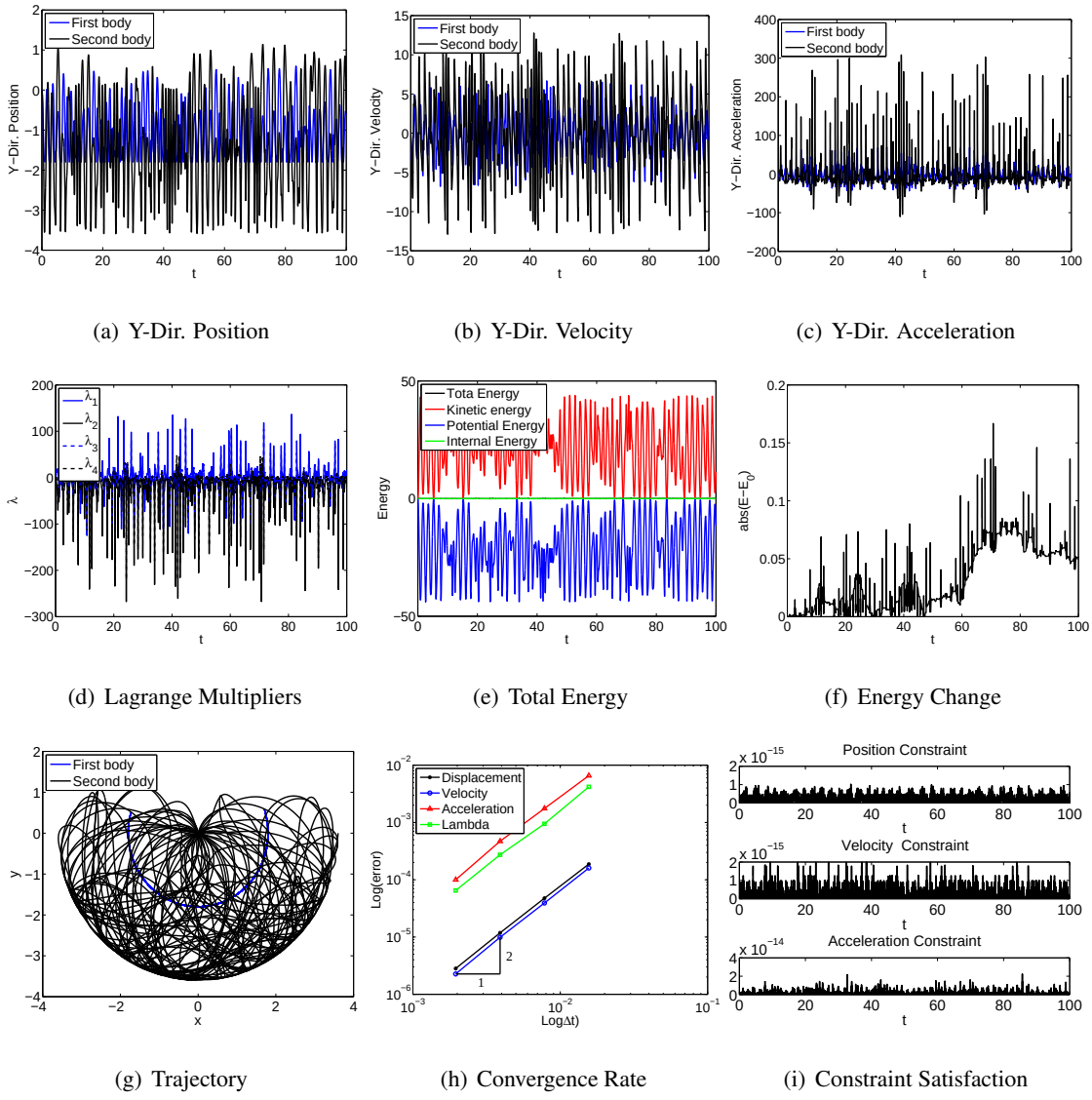


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.246: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0(1,1,0) - Index 1 - with projection method.

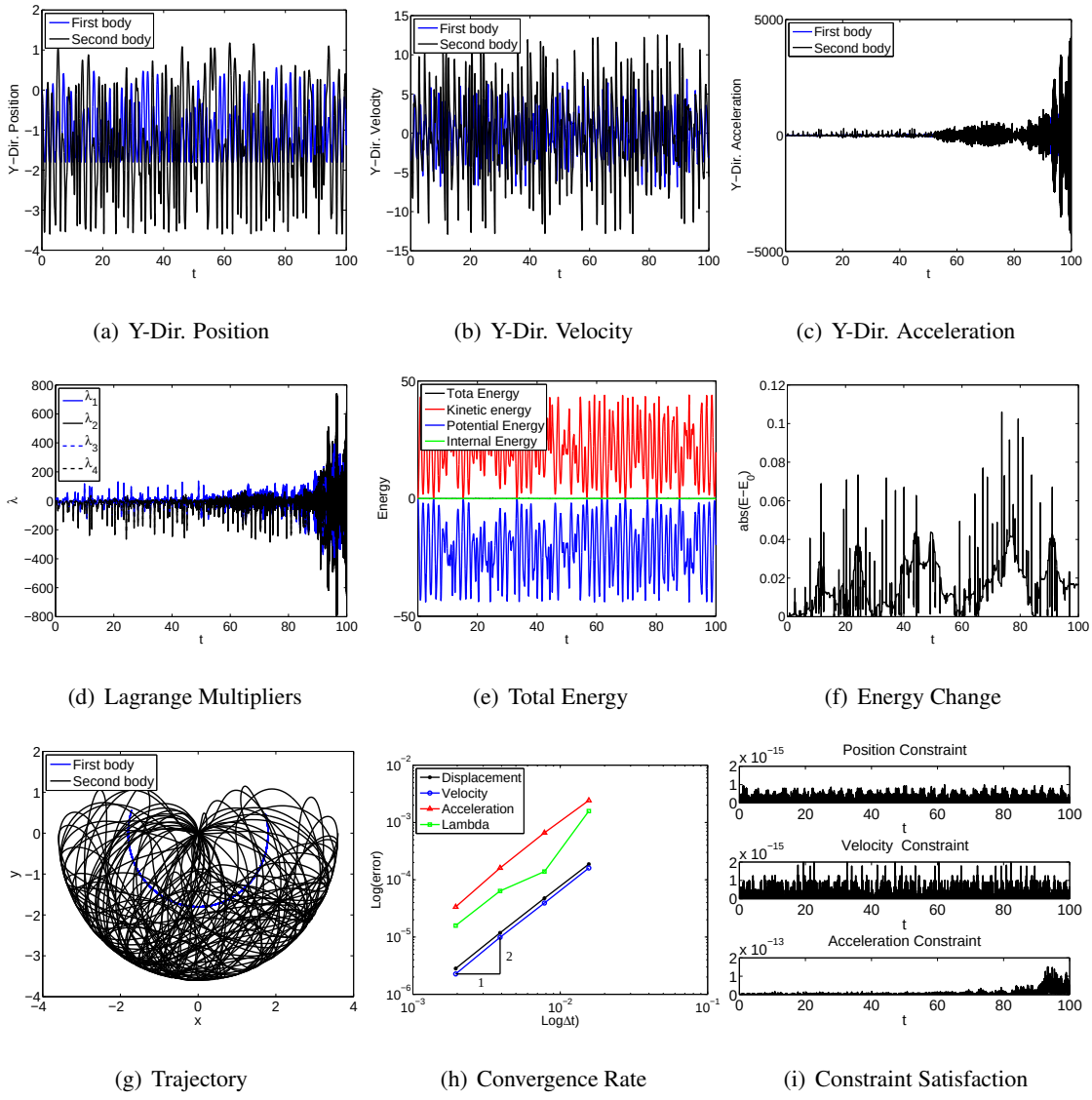


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.247: Long time simulation for double quasi-rigid pendulum with bar element in FRF: V0(1,1,0) - Index 1 - with projection method.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.248: Long time simulation for double quasi-rigid pendulum with bar element in FRF: U0V0(1,1,1) - Index 1 - with projection method.

From the results of Fig. 4.246 to Fig. 4.248, one can easily observe that only V0(1,1,0) simulates successful. With the projection method, the U0(1,1,0) energy is not bounded exactly and U0V0(1,1,1) causes oscillation although the energy is bounded. That is to say: for quasi-rigid case with bar element, V0(1,1,0) with FRF and projection method are much robust than the other two schemes. For general flexible bar element case: V0(1,1,0) with ANCF-S is recommended than the other two formulations (IRF and FRF). For the beam element, the participation of rotation degrees of freedom causes different results for high rigidity case. See Table. 4.6 and Table. 4.7.

Table 4.6: Computational cost of quasi-rigid double pendulum with beam element (iterations/Δt)

Index	Algorithm	IRF	ANCF-S	FRF
3	U0(1,1,0)	Fail	60.21	7.37*
	V0(1,1,0)	Fail	60.80	7.07
	U0V0(1,1,1)	Fail	60.12	7.30
2	U0(1,1,0)	Fail	60.20	7.48
	V0(1,1,0)	Fail	60.80	7.43
	U0V0(1,1,1)	Fail	60.10	7.53
1	U0(1,1,0)	Fail	60.32	7.83
	V0(1,1,0)	Fail	60.82	7.96
	U0V0(1,1,1)	Fail	60.11	7.70

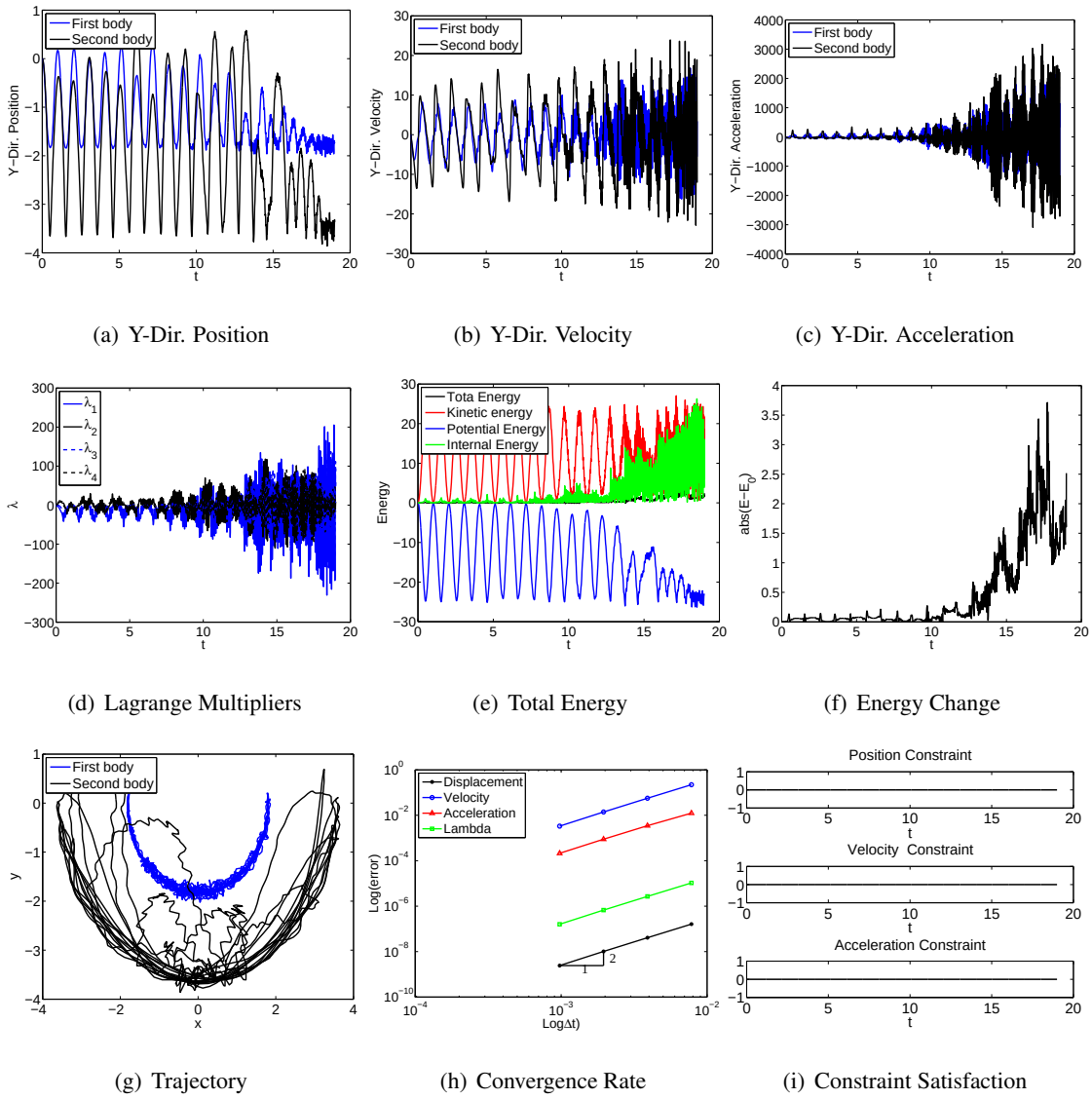
Table 4.7: Computational cost of flexible double pendulum with beam element (iterations/Δt)

Index	Algorithm	IRF	ANCF-S	FRF

3	U0(1,1,0)	6.48	4.04	10.23*
	V0(1,1,0)	6.48	4.03	9.67
	U0V0(1,1,1)	6.48	4.00	10.23*
2	U0(1,1,0)	6.50	4.04	10.72
	V0(1,1,0)	6.51	4.05	9.93
	U0V0(1,1,1)	6.49	4.00	10.20
1	U0(1,1,0)	6.51	4.04	11.01*
	V0(1,1,0)	6.51	4.07	10.17*
	U0V0(1,1,1)	6.50	4.00	10.27*

Note:* denotes drift effect occurs within 5 second and unstable phenomenon.

The same situation occurs here for IRF since it cannot converge due to the high nonlinearity in stiffness. Also, the cost for ANCF-S is very expensive and Table. 4.7 shows the same feature for computational cost for both cases. Therefore, for quasi-rigid body dynamics, it seems that FRF is much more preferable. The long time simulation for flexible beam cases is shown:

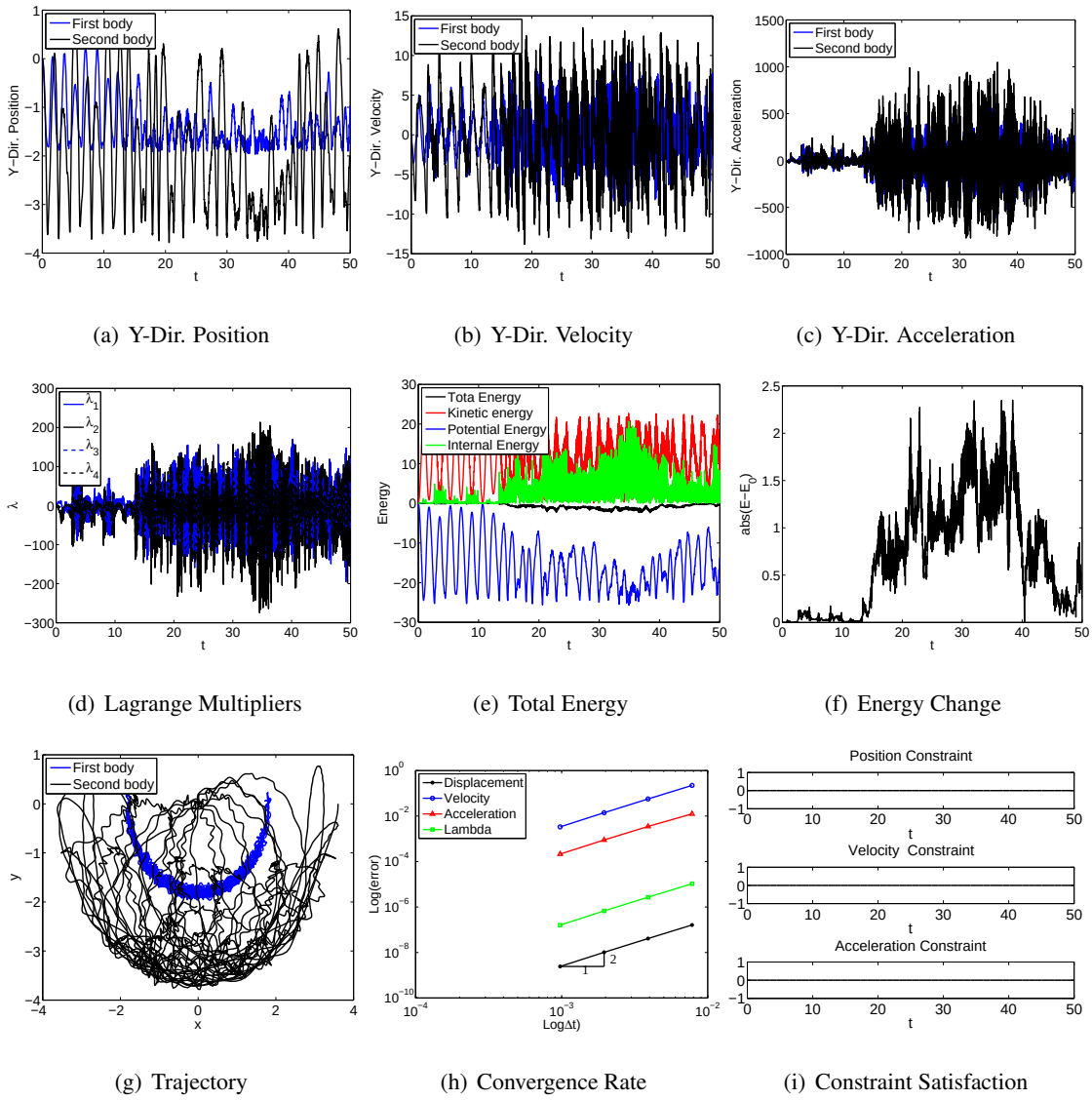


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.249: Long time simulation for double flexible pendulum with EB/TB beam element in IRF: U0(1,1,0) - Index 3.

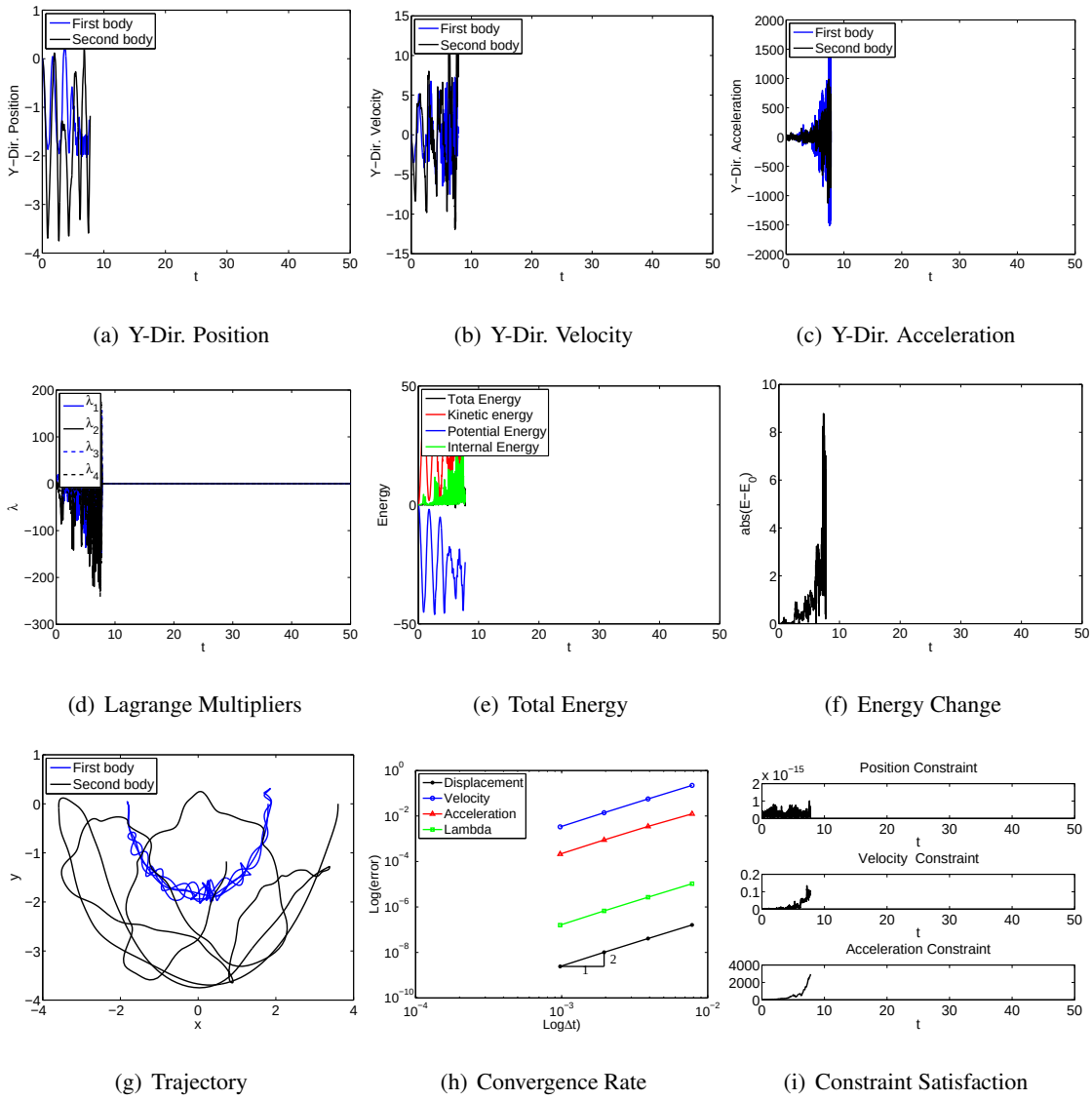


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.250: Long time simulation for double flexible pendulum with EB/TB element in ANCF: U0(1,1,0) - Index 3.

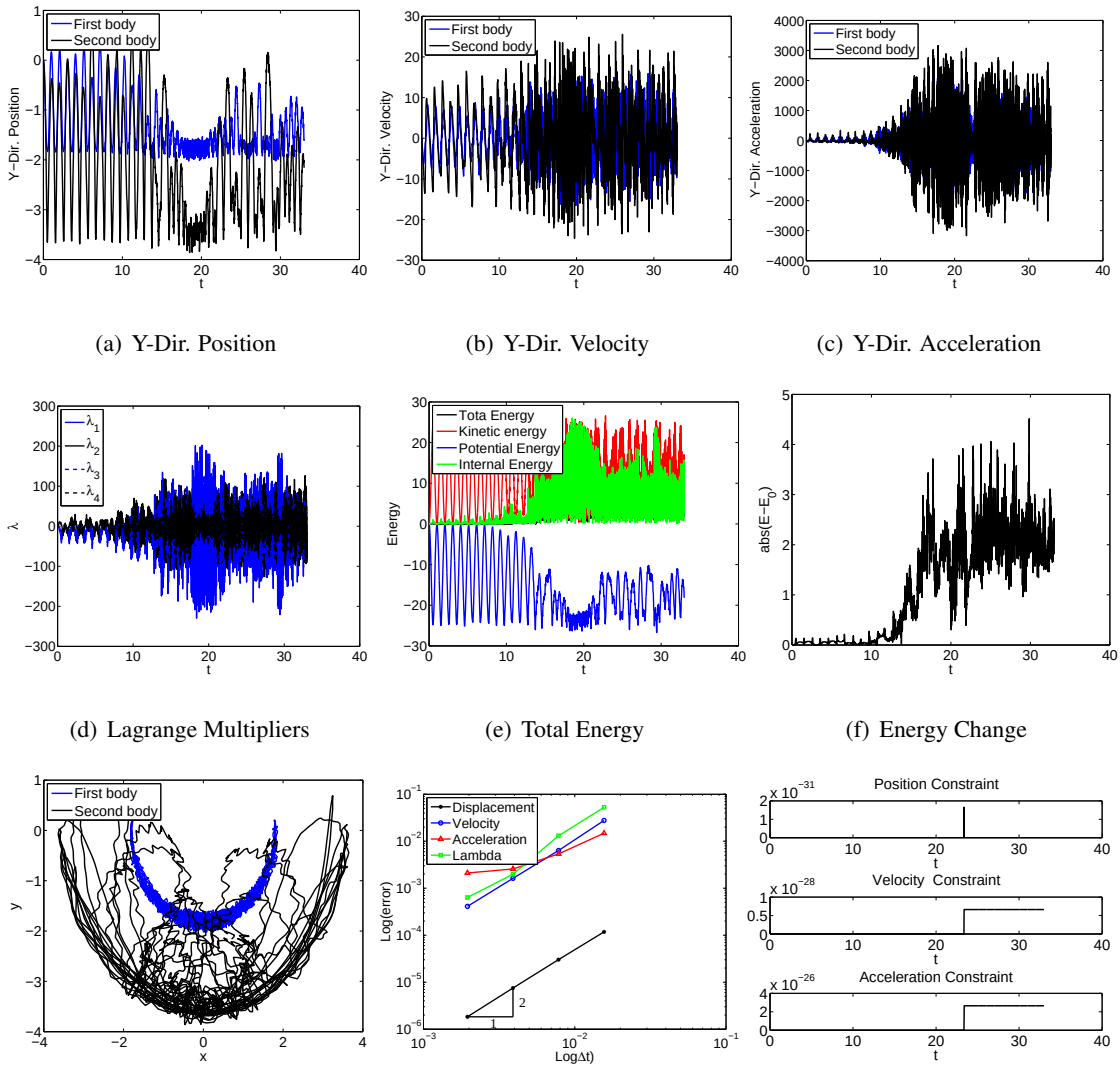


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.251: Long time simulation for double flexible pendulum with EB element in FRF: U0(1,1,0) - Index 3.

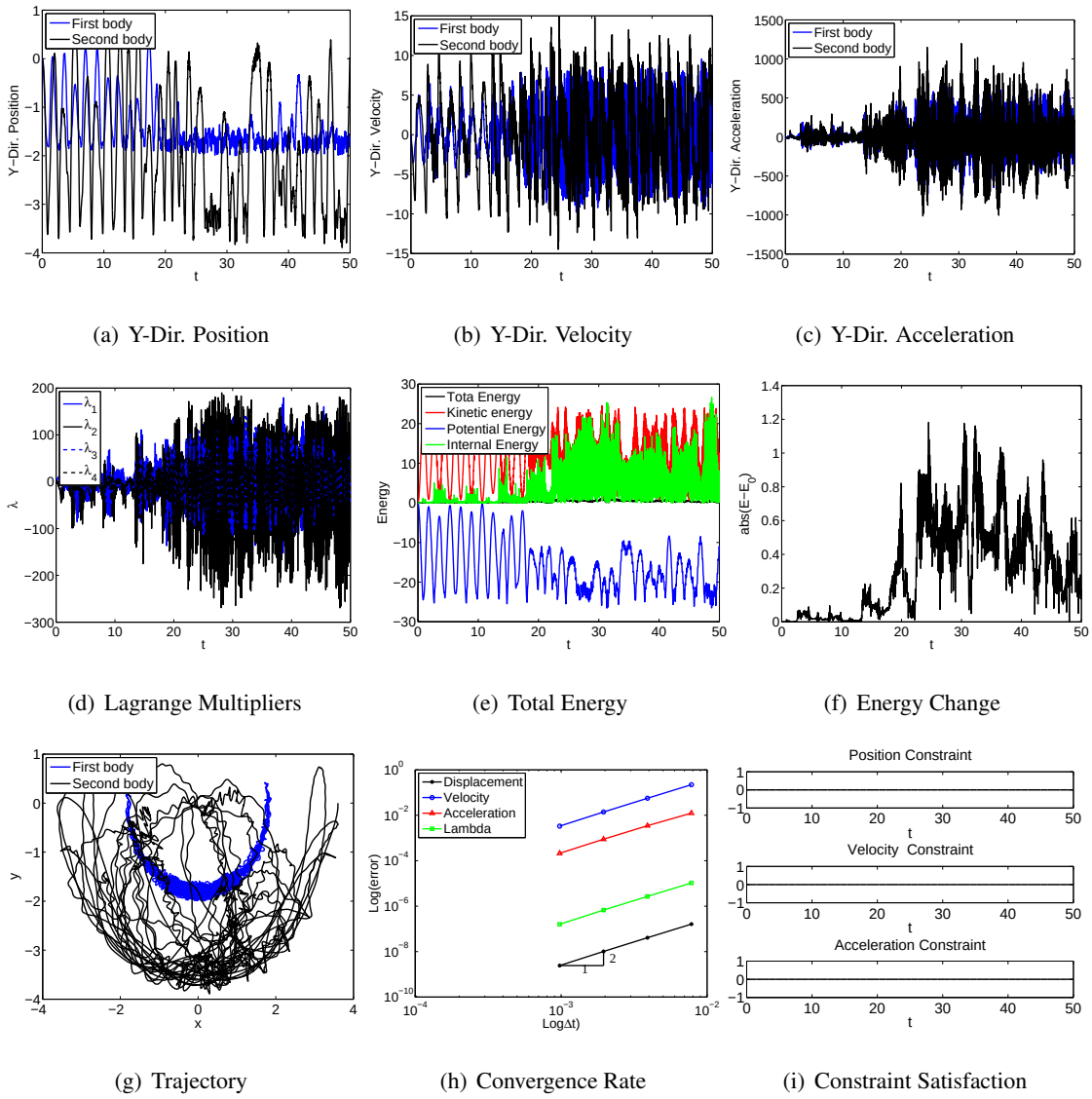


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.252: Long time simulation for double flexible pendulum with EB/TB beam element in IRF: V0(1,1,0) - Index 3.

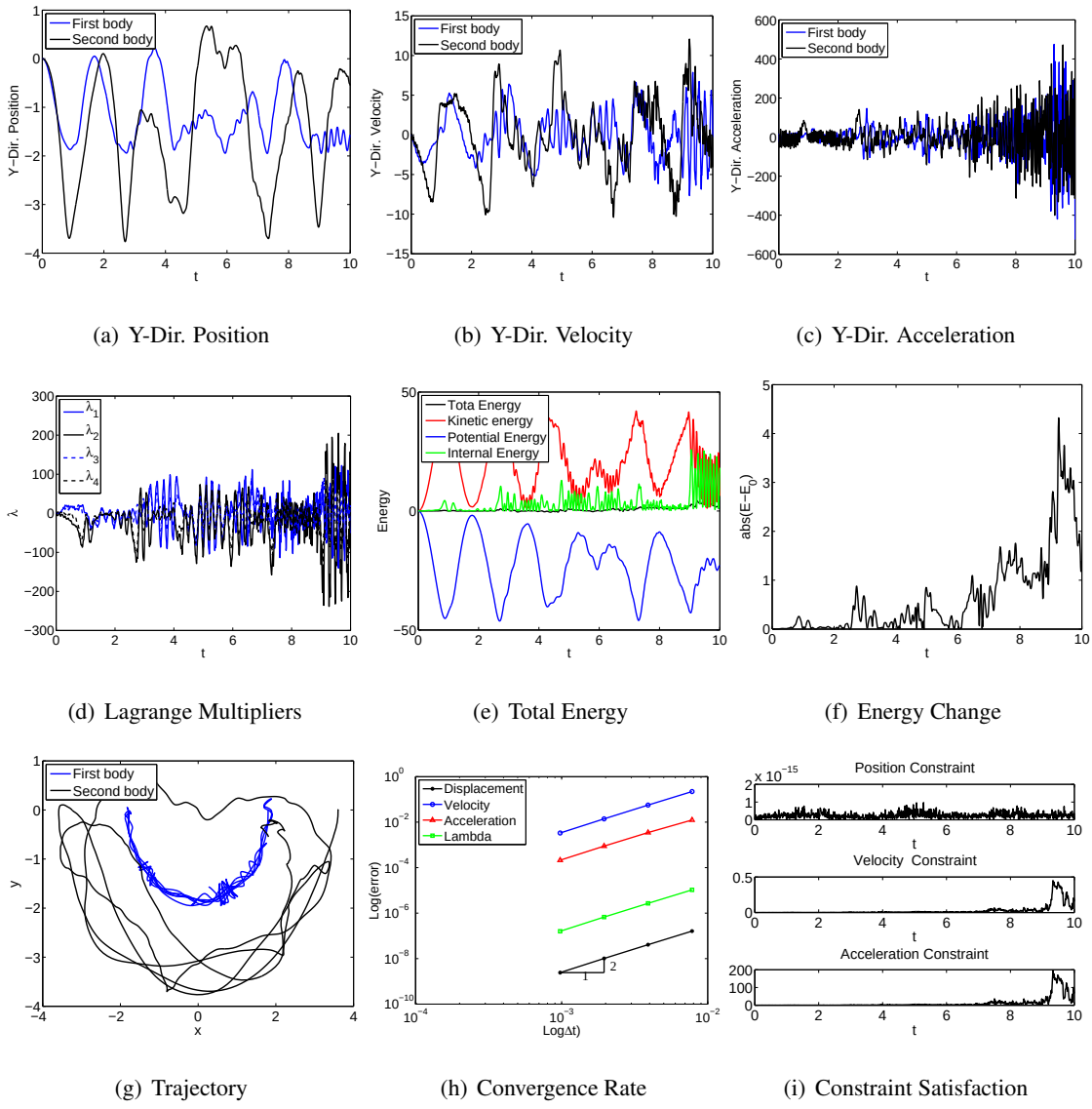


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.253: Long time simulation for double flexible pendulum with EB/TB element in ANCF: V0(1,1,0) - Index 3.

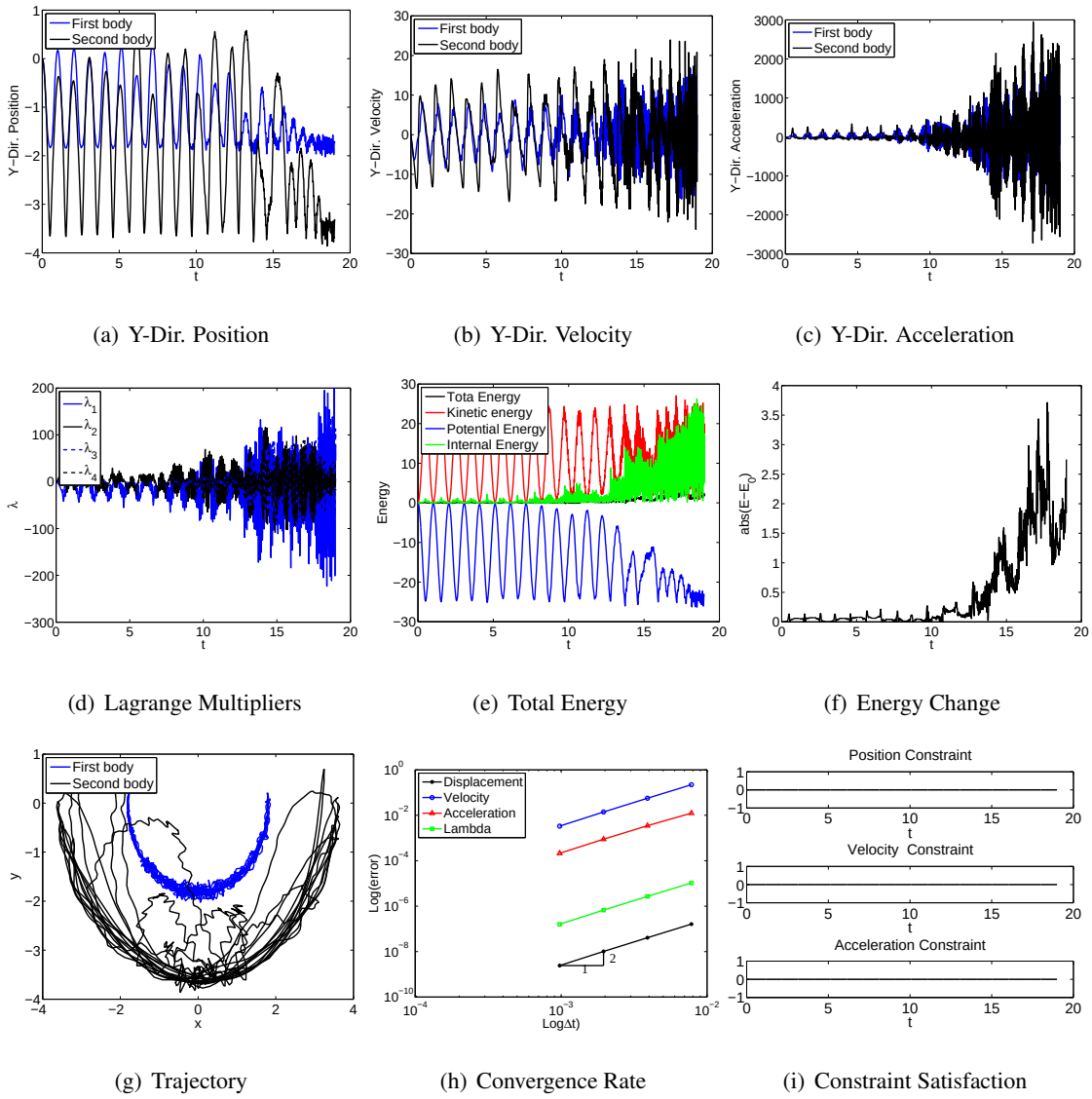


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.254: Long time simulation for double flexible pendulum with EB element in FRF: V0(1,1,0) - Index 3.

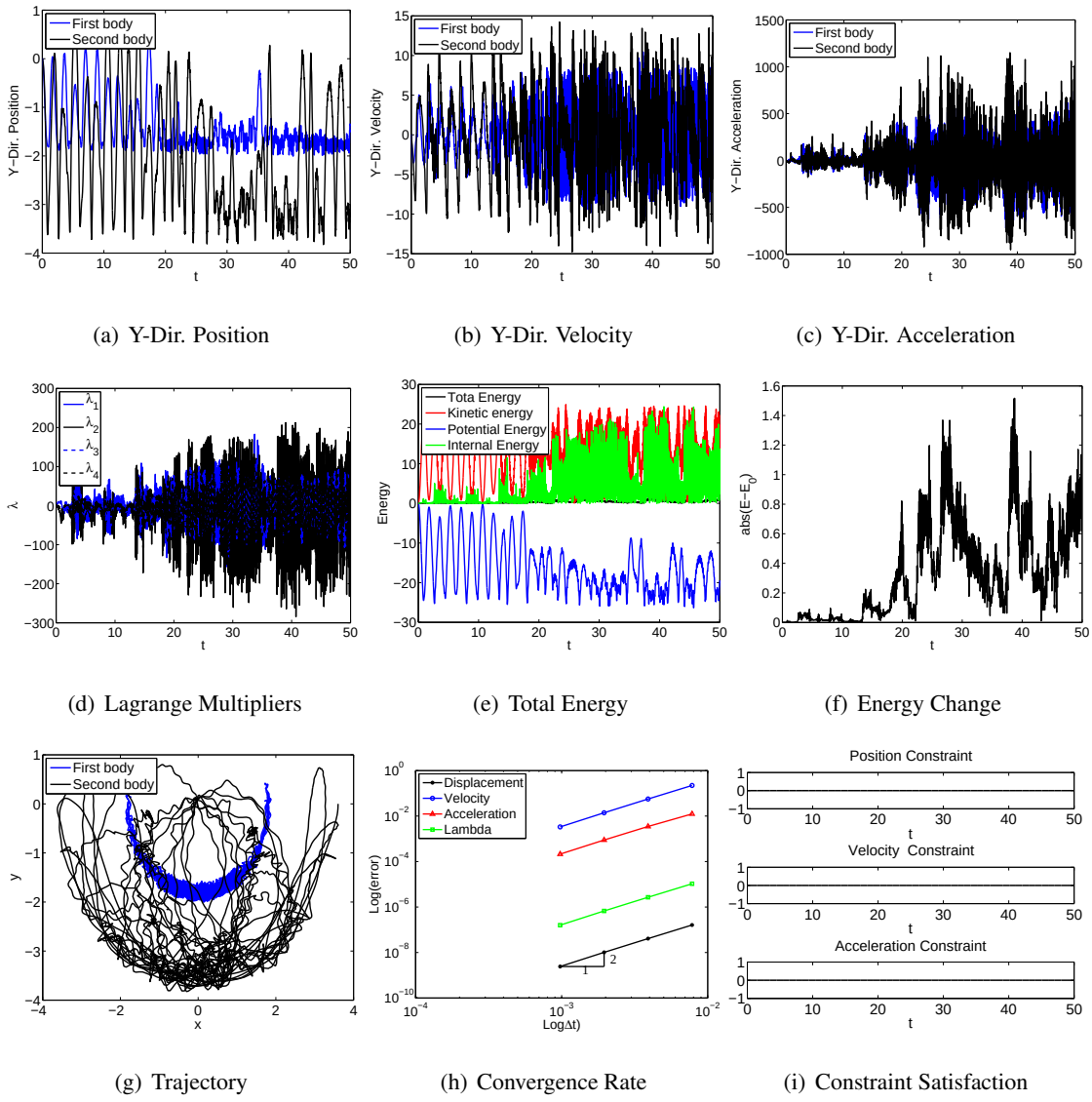


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.255: Long time simulation for double flexible pendulum with EB/TB beam element in IRF: U0V0(1,1,1) - Index 3.

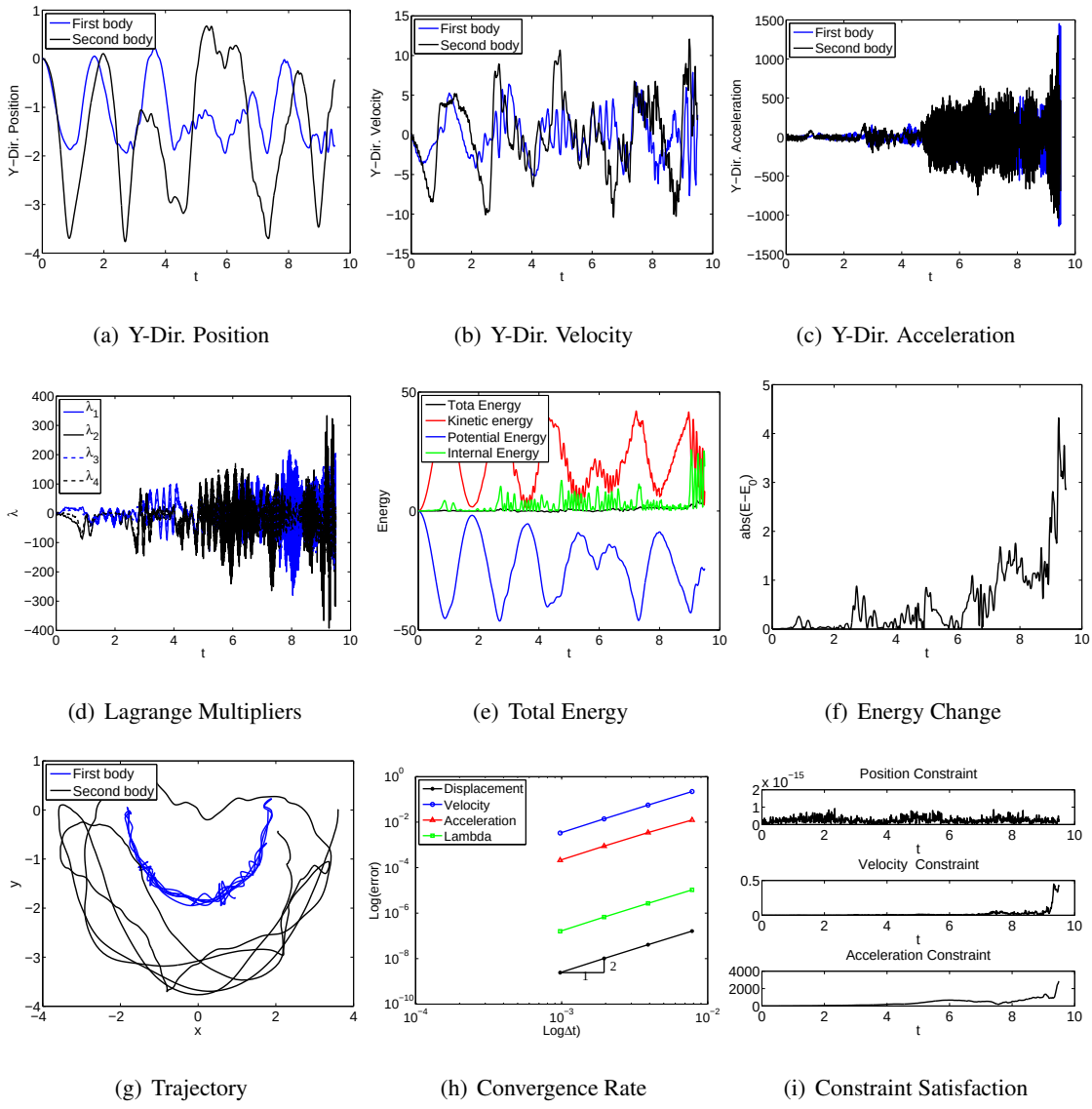


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.256: Long time simulation for double flexible pendulum with EB/TB element in ANCF: U0V0(1,1,1) - Index 3.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

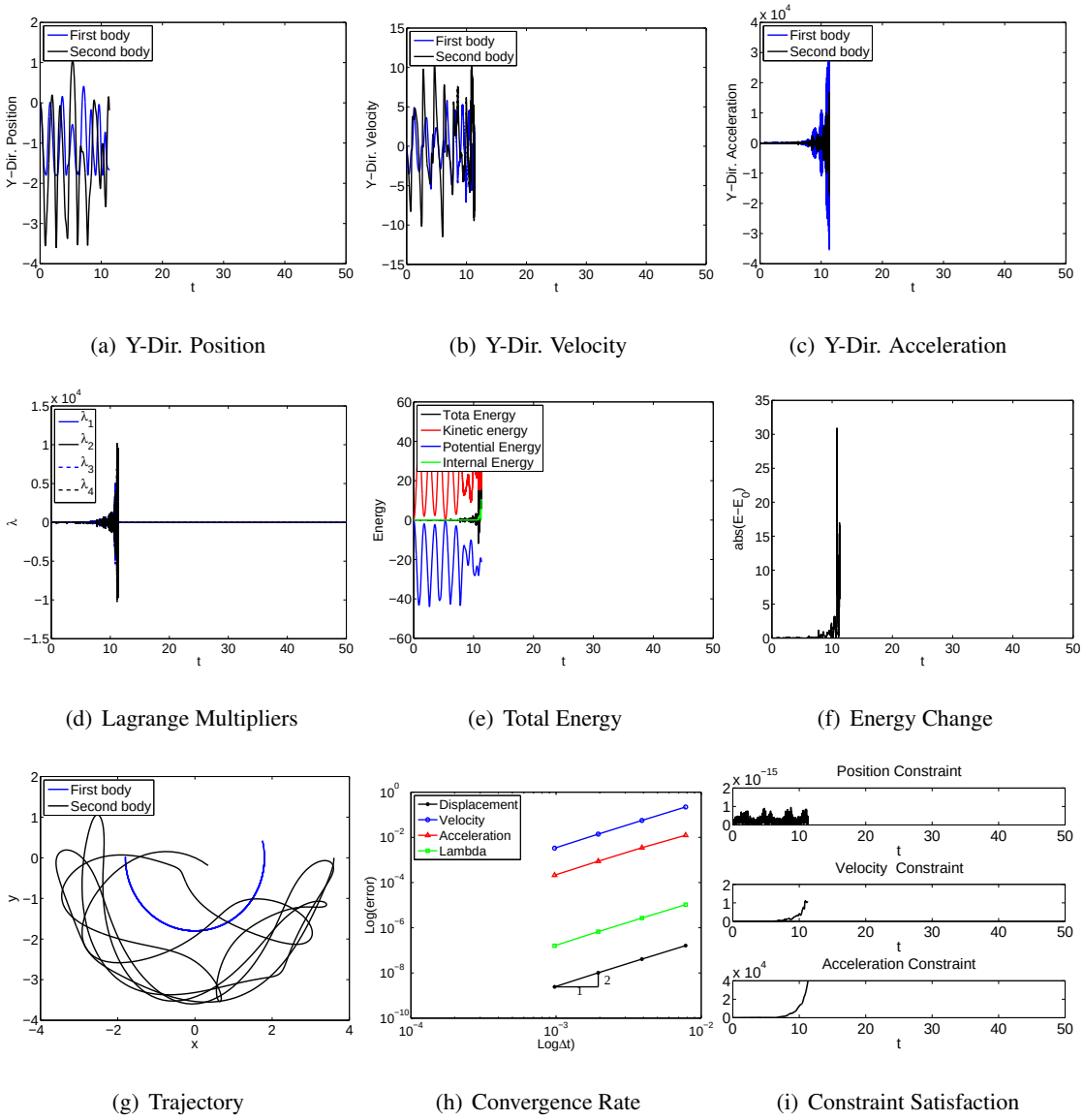
(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.257: Long time simulation for double flexible pendulum with EB element in FRF: U0V0(1,1,1) - Index 3.

Table 4.8: Long time computational cost of flexible double pendulum with beam element (iterations/Δt)

Index	Algorithm	IRF	ANCF-S	FRF
3	U0(1,1,0)	Fail	5.67	Fail
3	V0(1,1,0)	Fail	5.77	Fail
3	U0V0(1,1,1)	Fail	5.31	Fail

From Fig. 4.252 to Fig. 4.257, one may see that for long time simulation, FRF fail to converge within 10 second and so does IRF for 33 second. However, from the dynamic response of ANCF-S approach, one can easily discover the energy blows up and thus system become unstable. Although ANCF-S and IRF perform well in the bar element. It seems that the basic ANCF-S or IRF approach didn't result in good response. However, compare to failure simulation, at least ANCF-S shows the result in long time simulation. On the other hand, for quasi-rigid dynamics, the floating reference approach for U0(1,1,0), V0(1,1,0) and U0V0(1,1,1) are applied again for beam element in this case:

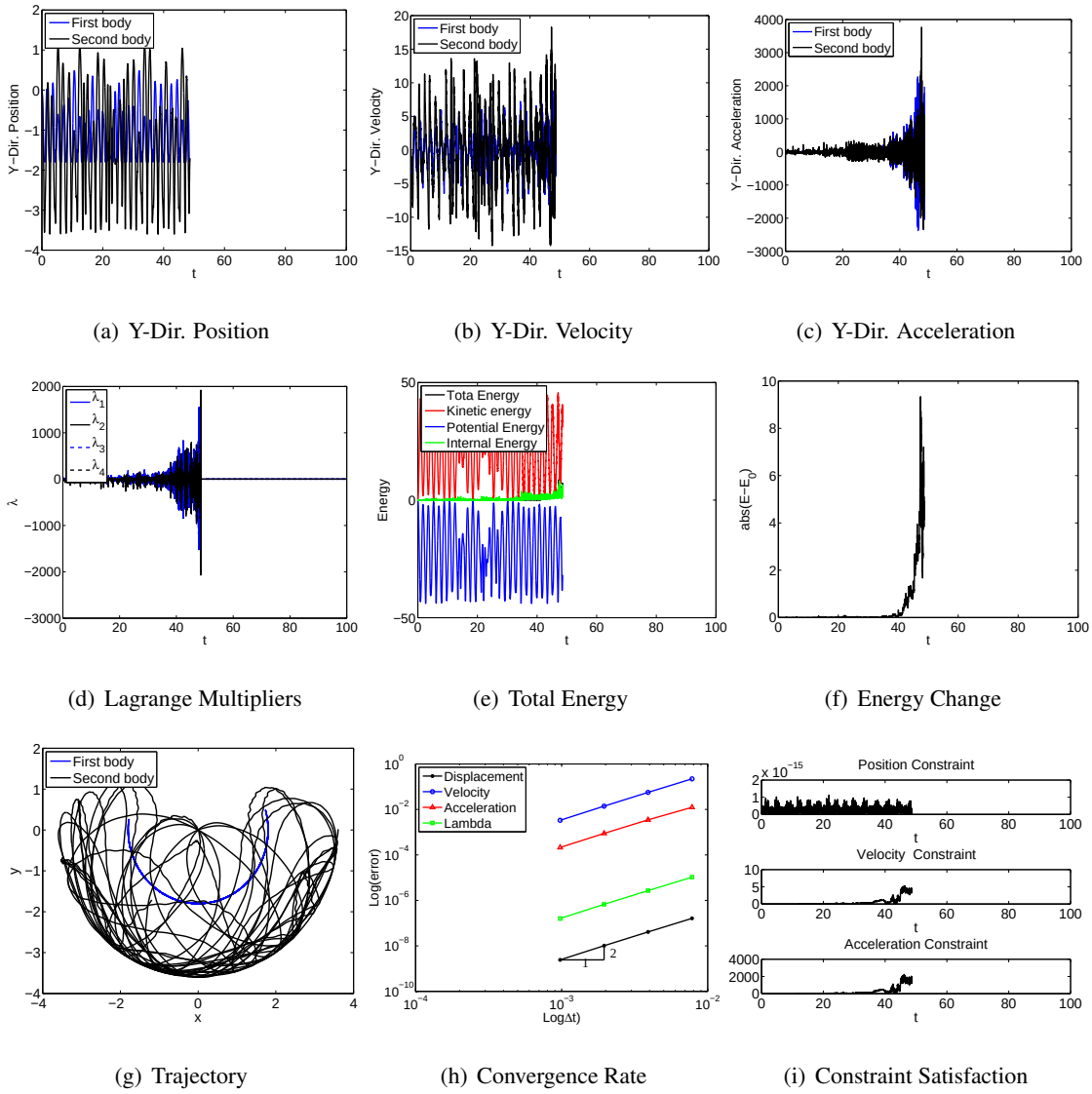


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.258: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0(1,1,0) - Index 3.

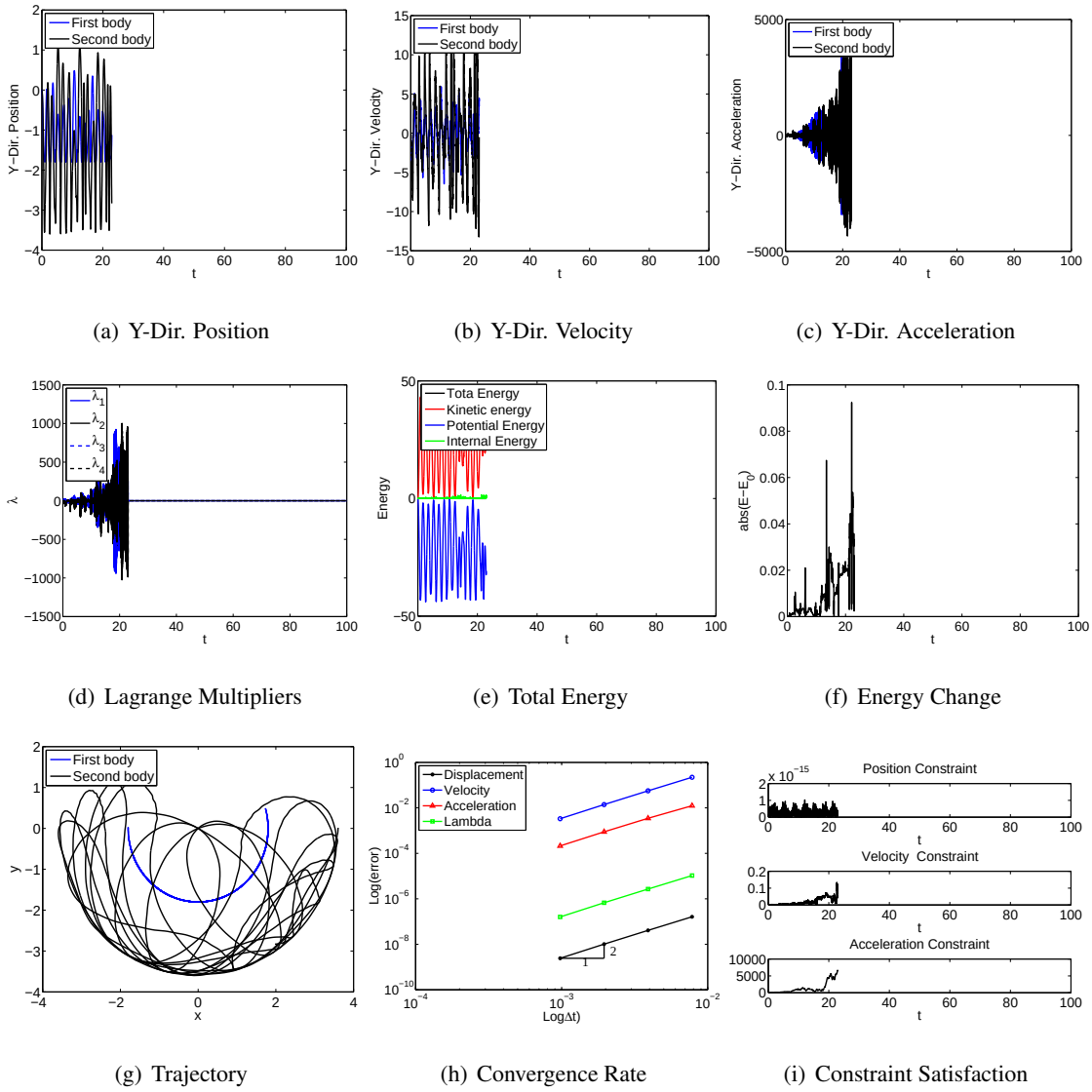


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.259: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: V0(1,1,0) - Index 3.

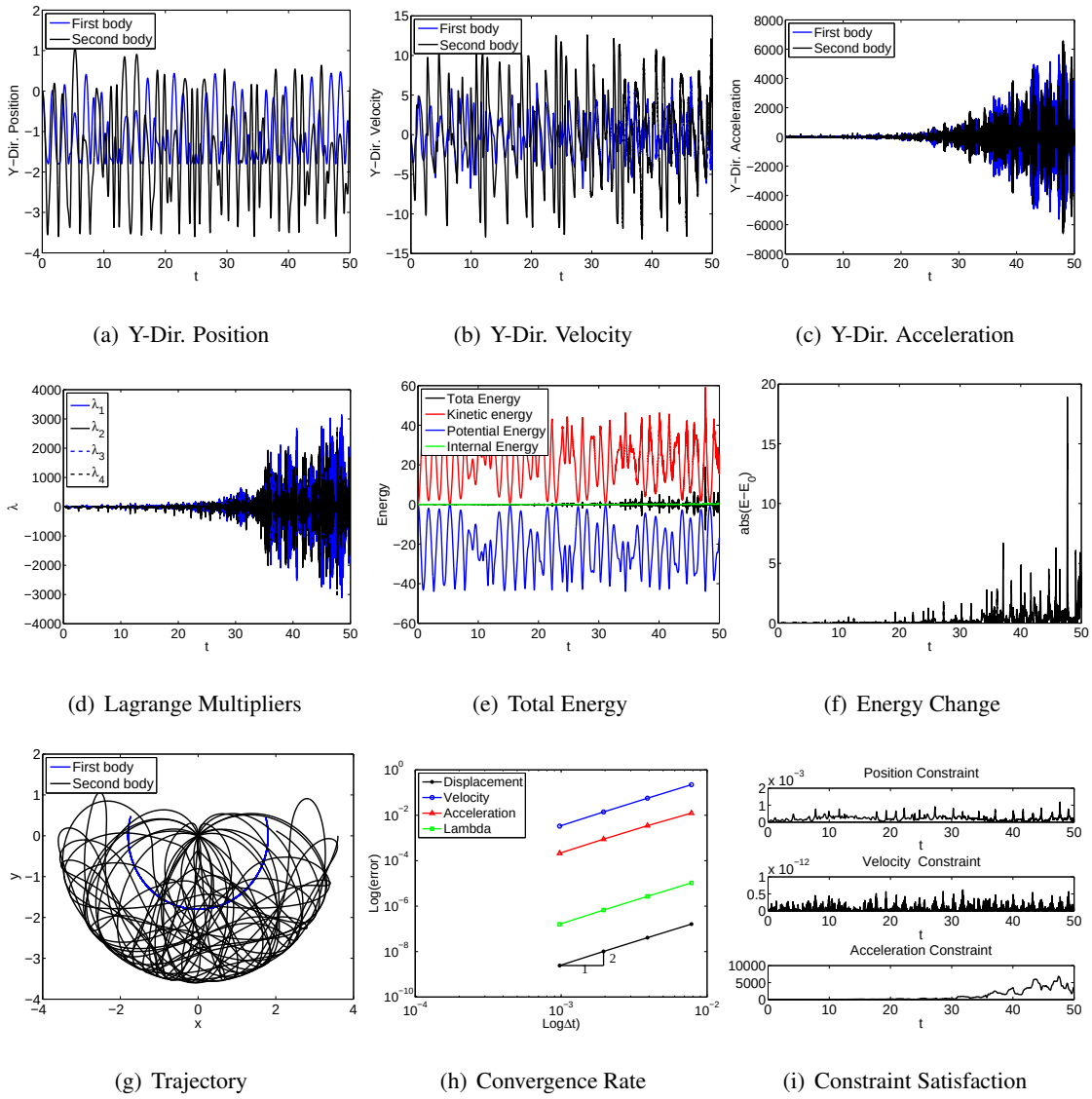


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.260: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0V0(1,1,1) - Index 3.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.261: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0(1,1,0) - Index 2.

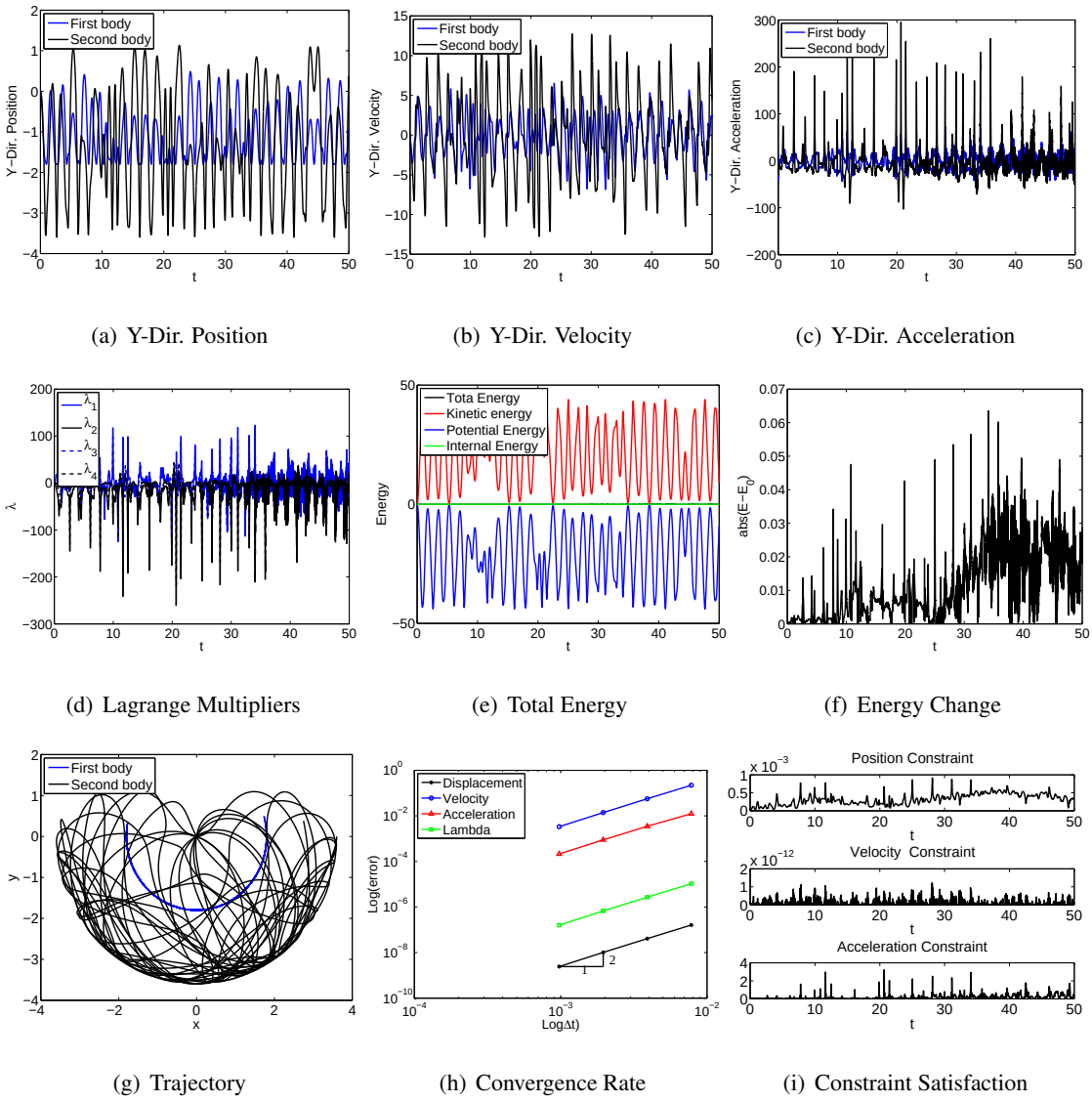
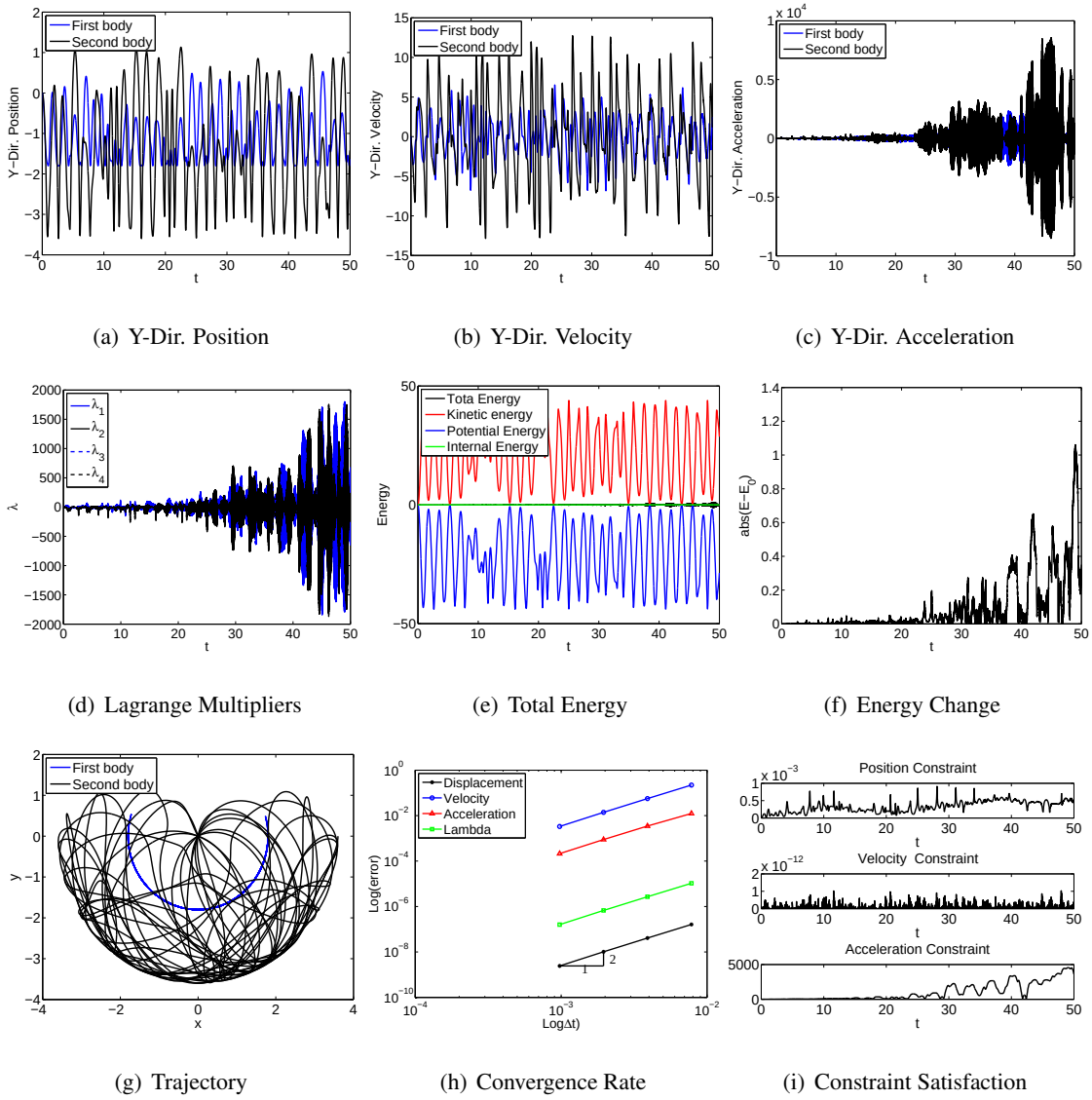


Figure 4.262: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: V0(1,1,0) - Index 2.

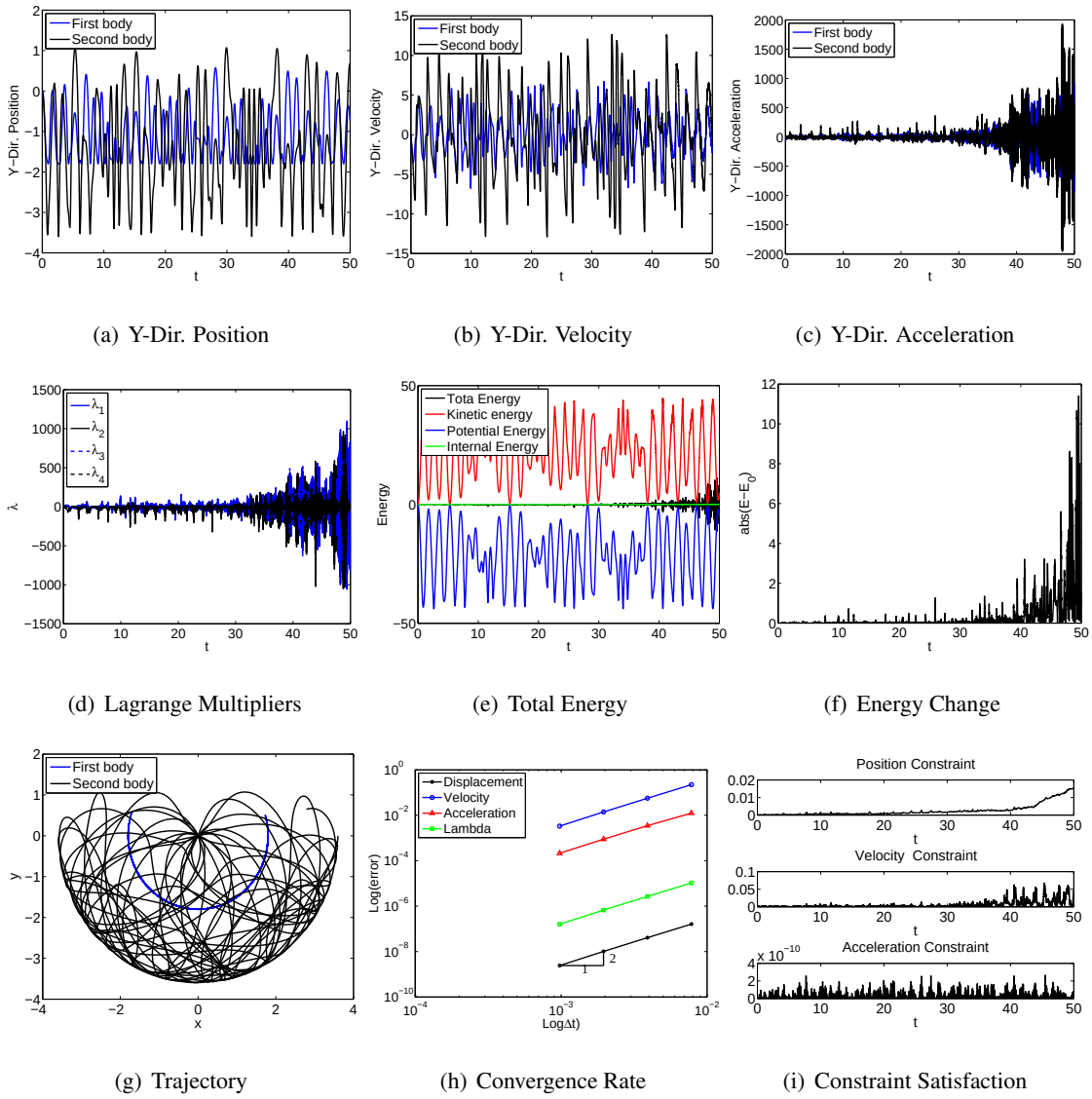


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.263: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0V0(1,1,1) - Index 2.

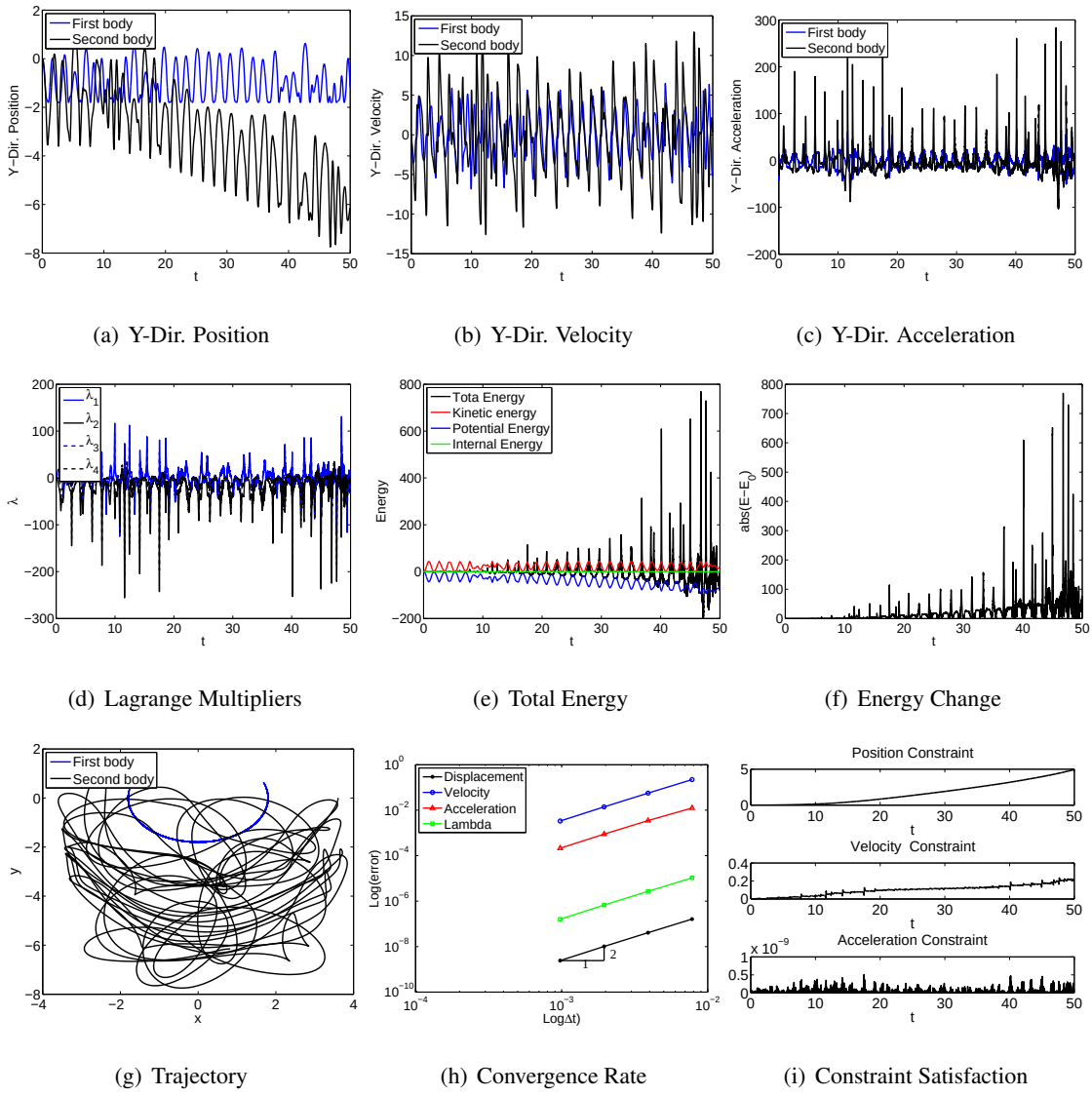


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.264: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0(1,1,0) - Index 1.

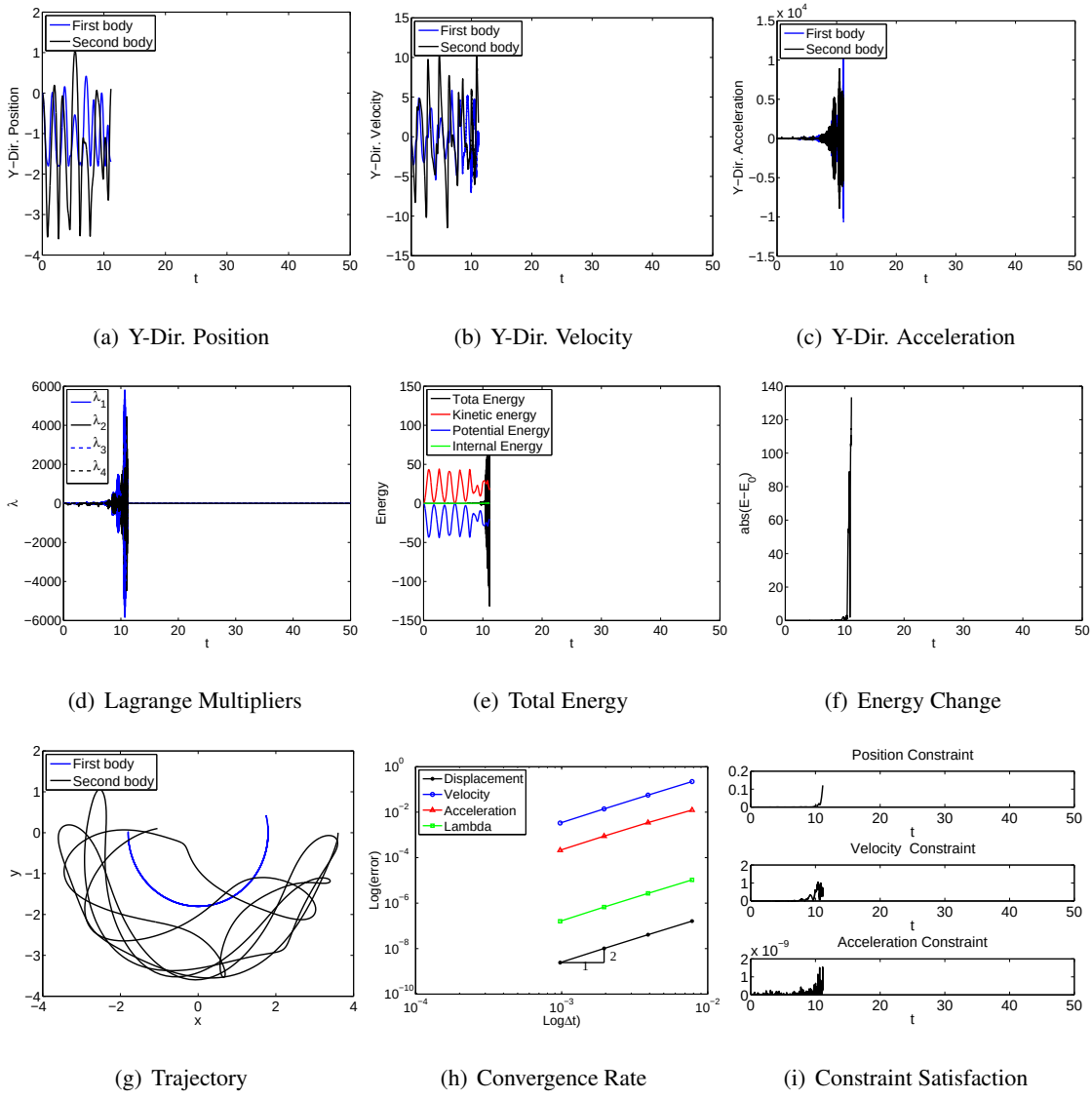


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.265: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: V0(1,1,0) - Index 1.



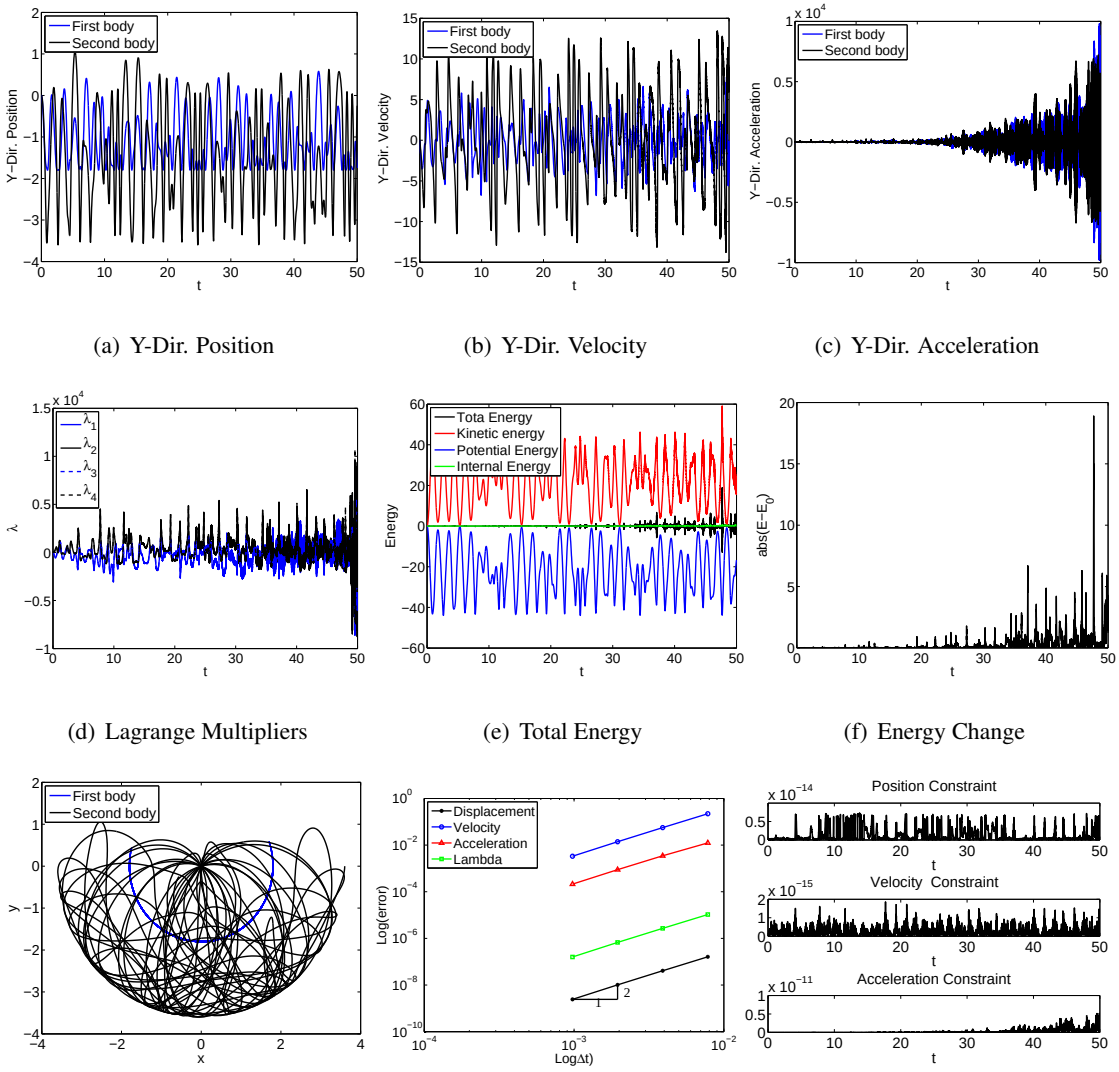
(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.266: Long time simulation for double quasi-rigid pendulum with EB/TB element in FRF: U0V0(1,1,1) - Index 1.

As in the case of the bar element, obviously the results for Index 2 are much robust than Index 1 and Index 3. Hence either the bar or the beam element show that without any stabilization technique, index 2 simulation is preferred if one only pursues a stable solution with exact constraint satisfaction. The results for projection method are shown below and show the exact constraint satisfaction.

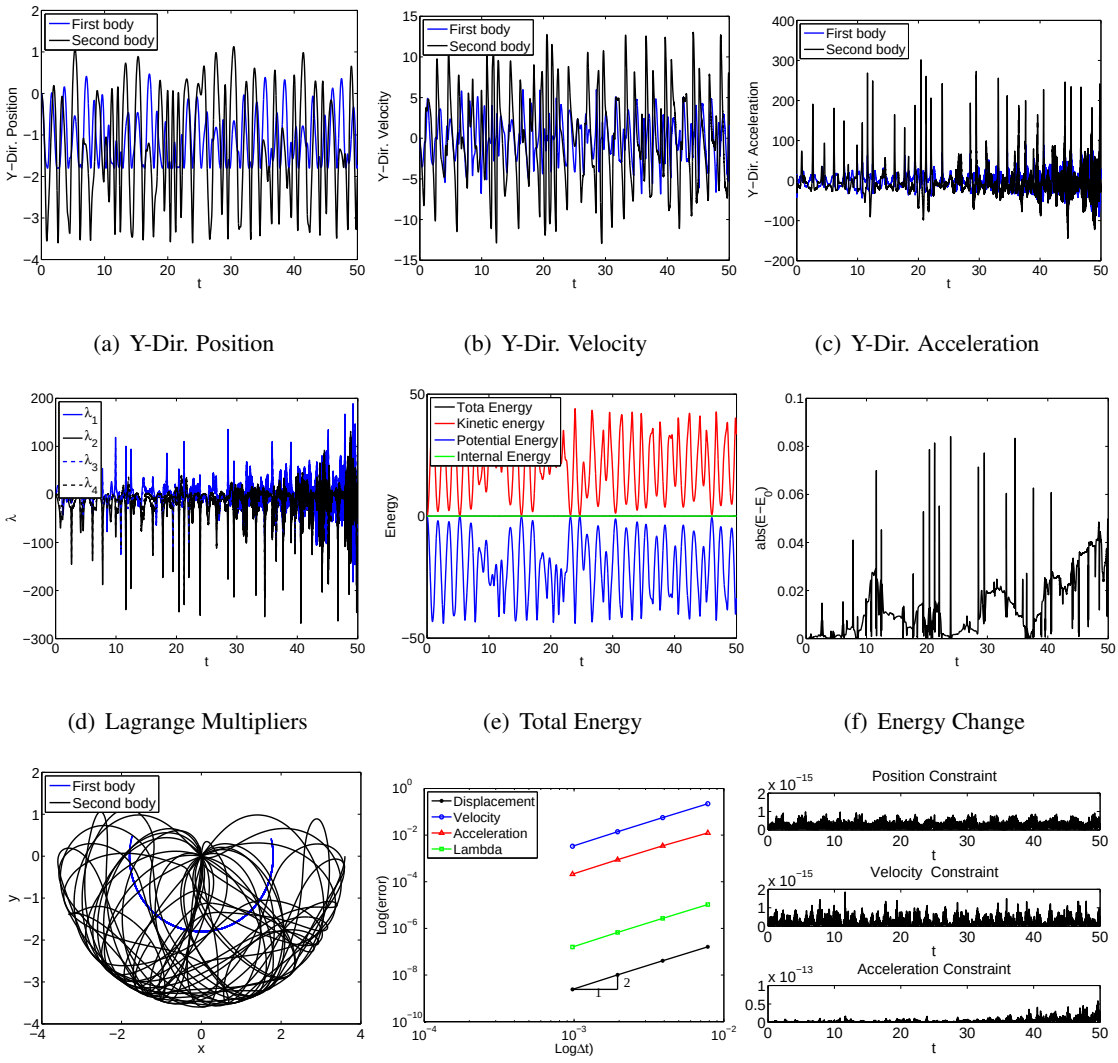


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.267: Long time simulation for double quasi-rigid pendulum with EB/TB beam element in FRF: U0(1,1,0) - Index 1 - with projection method.

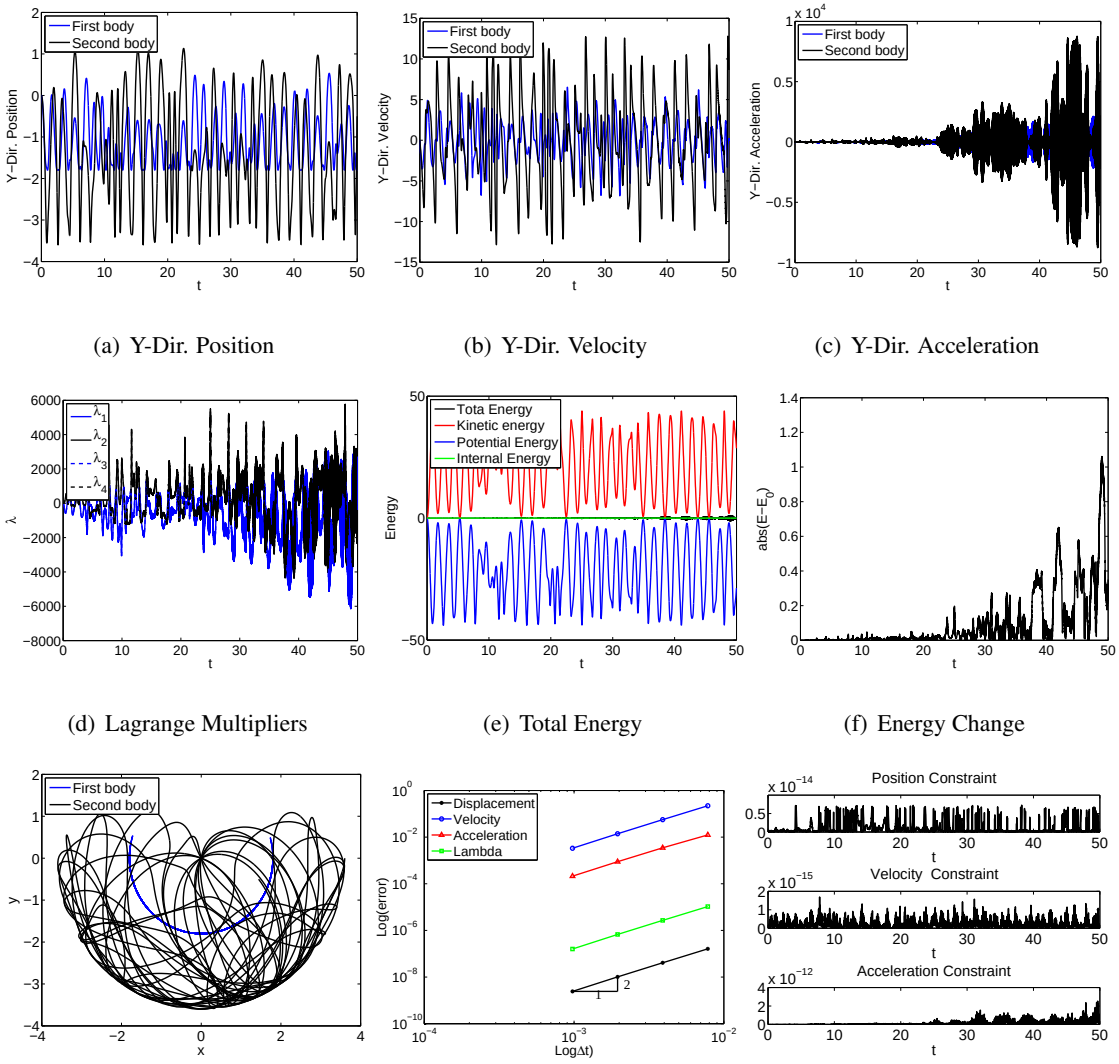


(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.268: Long time simulation for double quasi-rigid pendulum with EB/TB beam element in FRF: V0(1,1,0) - Index 1 - with projection method.



(a) Y-Dir. Position (b) Y-Dir. Velocity (c) Y-Dir. Acceleration

(d) Lagrange Multipliers (e) Total Energy (f) Energy Change

(g) Trajectory (h) Convergence Rate (i) Constraint Satisfaction

Figure 4.269: Long time simulation for double quasi-rigid pendulum with EB/TB beam element in FRF: U0V0(1,1,1) - Index 1 - with projection method.

From the results of Fig. 4.267 to Fig. 4.269, one can observe that with the projection method, the result for V0(1,1,0) is the most robust. Even with the projection method. U0(1,1,0) and U0V0(1,1,1) cannot preserve the energy exactly as well as it causes oscillations. Therefore, for the double pendulum for beam or bar element case, if the material is close to rigid (high Young's modulus), FRF in V0(1,1,0) with projection method is recommended due to it's stability and energy bounding feature. For the flexible case, ANCF-S is recommended compared to the IRF approach for the bar and beam element. The long time results indicate that although Index reduction can obtain optimal stability, however the drift effect will increase correspondingly especially for Index 1. As a result, from Index point of view, it shows that Index 2 simulation without projection is the most robust (if projection method is used, EOM is violated). If Index 1 simulation is employed for stability point of view, one must use projection method.

Chapter 5

Conclusions and Discussions

5.1 Analysis

We attempted to investigate and answer the following three problems in the field of multibody simulation:

1) What are the numerical advantages and disadvantages for using FRF or IRF or other such as ANCF?
2) Which Index is preferable? Under which condition?
3) How to simulate the DAEs for MBD problem properly (satisfaction of constraint, second order time accuracy, and numerical stability)?

The previous chapters employ the GSSSS family of algorithms as well as three formulation methods. Within this research space lies three numerically non-dissipative algorithms: the Newmark method U0(1,1,0), the Midpoint rule with endpoint acceleration U0V0(1,1,1), and

the Midpoint rule with end midpoint acceleration V0(1,1,0). Previous work on the application of these methods to nonlinear structural dynamics have provided two core concepts: the equation of motion time level and the normalized time weighted residual method. Utilizing these concepts three Index level techniques were developed which extend the parent linear algorithms, designed to integrate ordinary differential equations, to the realm of nonlinear differential-algebraic equations.

Another search space lies within the three basic formulation approaches in the field of multibody dynamic: Inertial reference formulation (IRF), Simplified absolute nodal coordinates formulation (ANCF-S), and Floating reference formulation (FRF) which are now regarded as common basis for approaches in this field. The principle of these formulation is the usage of the global inertial frames and local body coordinates. The former two approaches are based on the inertial reference frame only and employ the concept of continuum mechanics to derive the equation of motion in semi-discretized form for the equation of motion. While the last approach of FRF relies on both inertial reference frame and local body coordinates, this formulation yields nonlinear mass matrix and additional quadratic velocity vector.

Finally, due to the numerical instability issue (drift effect or numerical oscillation), one would choose to do Index reduction to ensure stability but sacrifice the exact position constraint satisfaction and then turn to employ additional constraint stabilization techniques such as Baumgarte's method and projection method discussed in this research. To determine the effect of Index reduction and post processing techniques, four examples of rigid body and flexible body dynamics were demonstrated. As shown in Chapter 4, the double pendulum for rigid body only is demonstrated and the result with and without post processing are given. From the results of double rigid pendulum, only V0(1,1,0) can provide long term stable simulation without the help of projection method or Baumgarte's method.

Table 5.1: Comparison of double pendulum without post processing

Index	Algorithm	short time simulation	Long time simulation
3	U0(1,1,0)	Fail	Fail
	V0(1,1,0)	Stable	Stable
	U0V0(1,1,1)	Stable	λ/a blows up
2	U0(1,1,0)	Stable	Drift
	V0(1,1,0)	Stable	Drift
	U0V0(1,1,1)	Stable	Drift
1	U0(1,1,0)	Stable	Drift
	V0(1,1,0)	Drift	Fail
	U0V0(1,1,1)	Stable	Drift

From Table. 5.1, Index reduction increases stability since few hidden constraints are needed to satisfy. However, position constraint satisfaction is violated eventually for long time simulation. That is to say, for rigid body dynamics simulations such as double pendulum case, for short time simulation, it is fine to do Index reduction to guarantee simulation stability. Also, Index reduction can help stability and satisfy the constraints up to 10^{-3} level. Based on this analysis, for short time simulation, it is allowable to do Index reduction to increase stability. Sometimes Index 2 is preferable for cases without projection method, it can still preserve position constraint. On the other hand, for long time simulation in Index 1, drift occurs and post processing is required to stabilize the system.

After post processing such as Baumgarte's methods and projection method, all three algorithms can handle long time simulation with stable solution. However, the projected solution does not

satisfy the equation of motion exactly and therefore one sacrifices the exact energy conservation for V0(1,1,0) and U0V0(1,1,1) case. Also the constraint satisfaction level can follow Table. 5.2:

Table 5.2: Comparison of Bumagarte's method and projection method

Constraint Level	Algorithm	Baumgarte's method	Projection Method
Position Constraint	U0(1,1,0)	10^{-4}	10^{-14}
Velocity Constraint	V0(1,1,0)	10^{-2}	10^{-14}
Acceleration Constraint	U0V0(1,1,1)	10^{0}	10^{-14}

The table implies that projection method satisfies the constraint much better than Baumgarte's stabilization. Also, from Table. 4.2, the computational cost is very close to each other. As a result, among the two post processing techniques, the projection method is recommended for: 1) satisfying all constraint exactly, and 2) low computational cost.

For the flexible body dynamics, among three formulation techniques, result in different nonlinearility in the equation of motion as in Table. 5.3

Table 5.3: Comparison of three methods

	IRF	ANCF-S	FRF
Usage of body coordinates	No	No	Yes
Mass matrix	Linear	Linear	Nonlinear
Stiffness matrix	Nonlinear	Nonlinear	Linear
External force	Linear	Linear	Nonlinear
Constraint equation	Linear	Linear	Nonlinear

From the results of the single and double flexible pendulum, it is easy to observe that FRF is not suitable for large deformation (very flexible body) simulation and it is consistent with the theory since FRF was originally introduced for small deformation simulations. For IRF and ANCF-S, they are developed for large deformation simulations; however, the resulting highly nonlinear stiffness matrix in the equation of motion often causes stiff problem. For the traditional inertial reference approach (IRF), the performance is good for the case in general short time simulation. However, from the results of single and double pendulum, it fails to converge in the quasi-rigid case and for long time flexible pendulum simulation. It is also unable to capture the physics of rotation degree of freedom in good manner. Hence researchers developed the concept of absolute nodal coordinate to overcome this situation even though it still causes high nonlinearity in the stiffness matrix. A modified ANCF was developed and it is a simple version to avoid the nonlinear effect. ANCF-S, as a simplified version of ANCF, can preserve energy in a bounded and long time simulation and is stable. For quasi-rigid case, or small deformable body, FRF seems to be more robust than the other two methods from the view point of energy-preserving feature and oscillations in position/velocity/acceleration/Lagrange multiplier point of view. Also, from the computational cost of each formulation in quasi-rigid case, Table. 4.6, indicates that its cheaper to use FRF in a quasi-rigid beam element. From the theory, it is also easier to understand that when deformation is really small, it is hard to capture the exact value of longitudinal deformation ϵ_l since it is very close to zero. Therefore it is hard to capture the physics. From Fig. 4.267 to Fig. 4.269, we observe that only V0(1,1,0) from FRF with projection method provides a long time stable solution without any oscillation and energy is well bounded.

Table 5.4: Comparison of double quasi-rigid bar pendulum with FRF approach

Index	Algorithm	Short time	Long time	Long time with projection method
3	U0(1,1,0)	Oscillation	Fail	Stable
	V0(1,1,0)	Stable	Oscillation	Stable
	U0V0(1,1,1)	Stable	Fail	Oscillation
2	U0(1,1,0)	Stable	Oscillation	Stable
	V0(1,1,0)	Stable	Stable	Stable
	U0V0(1,1,1)	Stable	Stable	Oscillation
1	U0(1,1,0)	Stable	Stable	Stable
	V0(1,1,0)	Drift	Drift	Stable
	U0V0(1,1,1)	Stable	Fail	Oscillation

Table 5.5: Comparison of double quasi-rigid beam pendulum with FRF approach

Index	Algorithm	Short time	Long time	Long time with projection method
3	U0(1,1,0)	Oscillation	Fail	Oscillation
	V0(1,1,0)	Stable	Fail	Stable
	U0V0(1,1,1)	Oscillation	Fail	Oscillation
2	U0(1,1,0)	Oscillation	Oscillation	Oscillation
	V0(1,1,0)	Stable	Stable	Stable
	U0V0(1,1,1)	Oscillation	Oscillation	Oscillation
1	U0(1,1,0)	Oscillation	Oscillation	Oscillation
	V0(1,1,0)	Drift	Drift	Stable
	U0V0(1,1,1)	Oscillation	Fail	Oscillation

Note: Fail indicates that the solution cannot converge as simulation time is longer. Oscillation means that although solution converges, some quantities (especially Lagrange multiplier and acceleration) have numerical oscillation. Drift indicate that drift effect occurs in position constraint. Stable means solution converges without any oscillation.

From the Table. 5.4 and Table. 5.5, it is very clear that Index 1 simulation generally provides a stable solution even without projection method for bar case. That is why many researchers resort to Index reduction for better solution. This is the reason to turn to Index 2 or Index 1 for stable simulation. However, for quasi-rigid beam element, many of the algorithm in Index 3, Index 2, and Index 1 fail to converge or many oscillate dramatically in long time simulation. Based on this situation, the projection method is required to avoid the drift effect and oscillation.

5.2 Conclusions

The primary goal of this research was to investigate the existing methodology among the field of multibody dynamics and provide a proper path and clarify some subtle details in this field. Currently the literature is filled with methods for solving these systems with Index 3 system with numerically dissipative schemes or by Index reduction and methods to overcome the additional problems introduced by drift. For the formulation of equation of motion, literature shows two main classes of approach: inertial reference frame approach and floating reference frame approach, which provide big source of research data for either large or small deformation simulation. The methodology developed herein is able to give pros and cons for each issue above.

This research was essential for the GSSSS family of algorithms and the variety of methods and

their extension to nonlinear differential-algebraic equations. Three formulation (IRF/ANCF-S/FRF) with three numerically non-dissipative schemes were simulated with Index 3, Index 2, and Index 1. Also, the use of stabilization techniques such as Buamgarte's method and projection method were described. The significant result of this investigation follows in Table. 5.6 to Table. 5.9:

Table 5.6: Comparison of Index with three schemes without projection (Rigid body dynamics)

Index	U0(1,1,0)	V0(1,1,0)	U0V0(1,1,1)
3	Fail	Stable	Stable (Numerical Oscillation)
2	Stable (Drift)	Stable(Drift)	Stable (Drift)
1	Stable (Drift)	Drift(Fail)	Stable(Drift)

Note: () in the bracket indicate the numerical phenomenon in long term simulation

Table 5.7: Constraint satisfaction for two methods

Constraint Level	Baumgarte's method	Projection Method
Position Constraint	10^{-4}	10^{-14}
Velocity Constraint	10^{-2}	10^{-14}
Acceleration Constraint	10^{0}	10^{-14}

Table 5.8: Comparison of three formulation in double pendulum case

	IRF	ANCF-S	FRF
Usage of body coordinates	No	No	Yes

Mass matrix	Constant	Constant	Nonlinear
Stiffness matrix	Nonlinear	Nonlinear	Linear
External force	Linear	Linear	Nonlinear
Constraint equation	Linear	Linear	Nonlinear

Table 5.9: Comparison of different formulation

Formulation	Qausi-rigid bar	Flexible bar	Qausi-rigid beam	Flexible beam
IRF	Fail	Stable	Fail	Fail
ANCF-S	Stable (Expensive)	Stable	Stable (Expensive)	Stable
FRF	Stable	Fail	Stable	Fail

Note: () in the bracket indicate the side effect although the simulation is stable.

The optimal features for simulation in multibody dynamics follows next:

- Energy conservation
- Ability for long time simulation without dissipation
- Stable, non-oscillations in accelerations and Lagrange multipliers
- Satisfaction in position, velocity and acceleration constraint
- Second order time accuracy in displacement, velocity, acceleration, and Lagrange multipliers

To the knowledge of author, no algorithm with specific formulation can provide long time stable solution with all the features. As a result, it is essential to provides a design space for researchers and readers to follow and guide. Also, the design space guidelines are helpful for beginners in this field. Especially for those who wish achieve successful multibody simulations without violating the position constraint too much, and also to preserve stability and without the techniques of post processing, and ensure energy is bounded, one may follow Table. 5.10

Table 5.10: Recommendation table for multibody dynamics:(General)

Recommendation	Rigid	Rigid-Flexible	Qausi-rigid-Flexible	Flexible
GSSSS option	Symplecitc	Symplecitc	Symplecitc	Symplecitc
Algorithm	No difference	V0(1,1,0)	V0(1,1,0)	V0(1,1,0)
Formulation	No difference	ANCF	FRF(quasi-rigid)-ANCF(flexible)	ANCF
Index	2	2	2	2
Projection	Yes	Yes	Yes	Yes

Note: If projection method is applied, the original equations of motion will not be exactly satisfied, and hence the exact energy conservation does not exist.

If exact energy conservation is the first priority, one may follow Table. 5.11.

Table 5.11: Recommendation table for multibody dynamics:(Energy conservation)

Recommendation	Rigid	Rigid-Flexible	Qausi-rigid-Flexible	Flexible
GSSSS option	EMM	EMM	EMM	EMM
Algorithm	V0(1,1,0)	V0(1,1,0)	V0(1,1,0)	V0(1,1,0)
Formulation	No difference	ANCF	FRF(quasi-rigid)-ANCF(flexible)	ANCF

Index	3	3	3	3
Projection	Yes	Yes	Yes	Yes

Note: If projection method is applied, the original equations of motion won't be exactly satisfied, and hence the exact energy conservation doesn't exist.

If stability is the top priority, one may use Table. 5.12.

Table 5.12: Recommendation table for multibody dynamics:(Stability)

Recommendation	Rigid	Rigid-Flexible	Qausi-rigid-Flexible	Flexible
GSSSS option	Symplectic	Symplectic	Symplectic	Symplectic
Recommendation	Rigid	Rigid-Flexible	QausiRigid-Flexible	Flexible
Algorithm	U0(1,1,0)	U0(1,1,0)	U0(1,1,0)	U0(1,1,0)
Formulation	No difference	ANCF	FRF(quasi-rigid)-ANCF(flexible)	ANCF
Index	1	1	1	1
Projection	Yes	Yes	Yes	Yes

Note: If projection method is applied, the original equations of motion will not be exactly satisfied, and hence the exact energy conservation does not exist.

From Table. 5.10 to Table. 5.12, this research provides a way to improve and advance the current understanding to solve rigid/flexible multibody dynamics, thus greatly increasing efficiency of

multibody system design. Future efforts should focus upon studying in-depth the pros and cons of EMM (as opposed to Symplectic algorithms in this thesis), sub-domain methods, reduction methods, adaptive time stepping, large scale and parallel computing, and multi-scale aspects.

Bibliography

[1] E. Hairer, S. P. Norsett, and G. Wanner. Solving ordinary differential equations. *Springer-Verlag*, 2006.

[2] R. McLachlan. Six lectures on the geometric integration. *Foundations of Computational Mathematics*, (2001): 155-210.

[3] E. Noether. Invariante variationsprobleme. *In Nachr. d. König. Gesellsch. d. Wiss. zu Göttingen*, pages 235-257, Math-Phys. Klasse, 1918.

[4] A. A. Shabana. Flexible multibody dynamics: review of past and recent developments. *Multibody Syst. Dyn*, 1.2 (1997): 189-222.

[5] H. Bermer. On the dynamics of elastic multibody systems. *Applied Mechanics Reviews*, 52.9 (1999): 275-303.

[6] W. Schiehlen. Computational dynamics: theory and applications of multibody systems. *European Journal of Mechanics-A/Solids*, 25.4 (2006): 566-594.

[7] R. L. Huston. Multibody danumics since 1990. *Applied Mechanics Reviews*, 49.10S (1996): S35-S40.

[8] B. Simeon. On Lagrange multipliers in flexible multibody dynamics. *Computer methods in applied mechanics and engineering*, 195.50 (2006): 6993-7005.

[9] S. Leyendecker, P. Betsch, and P. Steinmann. Energy-conserving integration of constrained Hamiltonian systems - a comparison of approaches. *Computational Mechanics*, 33.3 (2004): 174-185.

[10] P. Betsch, and P. Steinmann. Constrained dynamics of geometrically exact beams. *Computational Mechanics*, 31.1-2 (2003): 49-59.

[11] A. A. Shabana. Dynamics of multibody systems, *Cambridge university press*, 2013.

[12] M. Geradin and A. Cardona. Flexible multibody dynamics: a finite element approach. *Wiley*, 2001.

[13] P. E. Gaultier and W. L. Cleghorn. Modeling of flexible manipulator dynamics: a literature survey. *Proc of 1st Natl App Mech and Robotics Conf*, 1989.

[14] P. W. Likins. Modal method for analysis of free rotations of spacecraft. *AIAA Journal*, 5.7 (1967): 1304-1308.

[15] J. T. Oden. Finite elements of nonlinear continua, *Courier Dover Publications*, 2006.

[16] K. J. Bathe, E. Ramm, and E. L. Wilson. Finite element formulations for large deformation dynamic analysis. *International Journal for Numerical Methods in Engineering*, 9.2 (1975): 353-386.

[17] A. A. Shabana. Definition of the slopes and the finite element absolute nodal coordinate formulation. *Multibody System Dynamics*, 1.3 (1997): 339-348.

[18] U. M. Ascher and L. R. Petzold. *Computer methods for ordinary differential equations and differential-algebraic equations*, Vol. 61. Siam, 1998.

[19] E, Edda, C. Fuehrer, B. Leimkuhler, and S. Reich. Stabilization and projection methods for multibody dynamics, *Unknown*, 1 (1990).

[20] C. Fuhrerand and B. J. Leimkuhler. Numerical solution of differential-algebraic equations for constrained mechanical motion. *Numerische Mathematik*, 59.1 (1991): 55-69.

[21] J. Baumgarte. Stabilization of constraints and integrals of motion in dynamical systems. *Computer methods in applied mechanics and engineering*, 1.1 (1972): 1-16.

[22] E. Bayo, and R. Ledesma. Augmented Lagrangian and mass-orthogonal projection methods for constrained multibody dynamics. *Nonlinear Dynamics*, 9.1-2 (1996): 113-130.

[23] J. Cuadrado, J. Cardenal, and E. Bayo. Modeling and solution methods for efficient real-time simulation of multibody dynamics. *Multibody System Dynamics*, 1.3 (1997): 259-280.

[24] 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/Zeitschrift fur Angewandte Mathematik und Mechanik*, 86.10 (2006): 759-771.

[25] K. K. Tamma, X. Zhou, and D. Sha. The time dimension: a theory towards the evolution, classification, characterization and design of computational algorithms for transient/dynamic applications. *Archives in Computational Mechanics*, 7.2 (2000): 67-290.

[26] K. K. Tamma, X. Zhou, and D. Sha. A theory of development and design of generalized integration operators for computational structural dynamics. *International Journal of Numerical Methods in Engineering*, 50.7 (2001): 1619-1664.

[27] X. Zhou and K. K. Tamma. Design, analysis, and synthesis of generalized single step single solve and optimal algorithms for structural dynamics. *International Journal for Numerical Methods in Engineering*, 59.5 (2004): 597-668.

[28] X. Zhou and K. K. Tamma. A new unified theory underlying time dependent linear

firstorder systems: a prelude to algorithms by design. *International Journal for Numerical Methods in Engineering*, 60.10 (2004): 1699-1740.

[29] X. Zhou and K. K. Tamma. Algorithms by design with illustrations to solid and structural mechanics/dynamics. *International Journal for Numerical Methods in Engineering*, 66.11 (2006): 1738-1790.

[30] K. K. Tamma and R. R. Namburu. Applicability and evaluation of an implicit self-starting unconditionally stable methodology for the dynamics of structures. *Computers and Structures*, 34.6 (1990): 835-842.

[31] N. M. Newmark. A method of computation for structural dynamics. *Journal for American Society of Civil Engineers*, Vol. 85. No. 3. 1959.

[32] W. L. Wood, M. Bossak, and O. C. Zienkiewicz. An alpha modification of Newmark's method. *International Journal for Numerical Methods in Engineering*, 15.10 (1980): 1562-1566.

[33] 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*, 5.3 (1977): 283-292.

[34] A. Hoitink, S. Masuri, X. Zhou, and K. K. Tamma. Algorithms by design: part ion the hidden point collocation within LMS methods and implications for nonlinear dynamics applications. *International Journal for Computational Methods in Engineering Science and Mechanics*, 9.6 (2008): 383-407.

[35] S. U. Masuri, A. Hoitink, X. Zhou, and K. K. Tamma. Algorithms by design: Part iii: A novel normalized time weighted residual methodology and design of optimal

symplectic-momentum based controllable numerical dissipative algorithms for nonlinear structural dynamics, *International Journal for Computational Methods in Engineering Science and Mechanics*, 10.1 (2009): 57-90.

[36] A. Hoitink. Investigations encompassing the equations of motion and proper and accurate treatment of algorithmic variables: computational structural dynamics and stiff systems, *Masters thesis dissertation*, Department of Mechanical Engineering, The University of Minnesota, 2009.

[37] J. Har and K. K. Tamma. Advances in Computational Dynamics, *Wiley*, ISBN 978-0-470-74980-7, 2012.

[38] O. Gonzalez. Exact energy and momentum conserving algorithms for general models in nonlinear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 190.13 (2000): 1763-1783.

[39] R. A. Labudde and D. Greenspan. Energy and momentum conserving methods of arbitrary order for the numerical integration of equations of motion. *Numerisch Mathematik*, 25.4 (1975): 323-346.

[40] J. C. Simo, N. Tarnow, and K. K. Wong. Exact energy-momentum conserving algorithms and symplectic schemes for nonlinear dynamics. *Computer Methods in Applied Mechanics and Engineering*, 100.1 (1992): 63-116.

[41] A. A. Shabana and R. Schwertassek. Equivalence of the floating frame of reference approach and finite element formulations, *International journal of non-linear mechanics*, 33.3 (1998): 417-432.

[42] A. A. Shabana, H. A. Hussien, and J. L. Escalona. Application of the absolute nodal coordinate formulation to large rotation and large deformation problems. *Journal of mechanical design*, 120.2 (1998): 188-195.

[43] J. C. Simo and L. Vu-Quoc. On the dynamics in space of rods undergoing large motionsa geometrically exact approach. *Computer methods in applied mechanics and engineering*, 66.2 (1988): 125-161.

[44] J. C. Simo and L. Vu-Quoc. On the dynamics of flexible beams under large overall motionsThe plane case: Part II. *Journal of Applied Mechanics*, 53.4 (1986): 855-863.

[45] P. Betsch and P. Steinmann. A DAE approach to flexible multibody Dynamics. *Multibody System Dynamics*, 8.3 (2002): 365-389.

[46] M. Berzeri, M. Campanelli, and A. A. Shabana. Definition of the elastic forces in the finite-element absolute nodal coordinate formulation and the floating frame of reference formulation. *Multibody System Dynamics*, 5.1 (2001): 21-54.

[47] M. Borri, L. Trainelli, and A. Croce. The embedded projection method: A general index reduction procedure for constrained system dynamics. *Computer Methods in Applied Mechanics and Engineering*, 195.50 (2006): 6974-6992.

[48] C. Lubich. On projected runge-kutta methods for differential algebraic equations. *BIT Numerical Mathematics*, 31.3 (1991): 545-550.

[49] L. R. Petzold. A description of dassl: A differential/algebraic system solver. In *International Conference on Scientific Computing*, 1983.

[50] C. W. Gear, B. Leimkuhler, and G. K. Gupta. Automatic integration of euler-lagrange equations with constraints. *Journal of Computational and Applied Mathematics*, 12 (1985): 77-90.

[51] C. Lunk and B. Simeon. Solving constrained mechanical systems by the family of newmark and α-methods. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik*, 86.10 (2006): 772-784.

[52] L. O. Jay and D. Negrut. Extensions of the hht-α method to differential-algebraic equations in mechanics. *Electronic Transactions on Numerical Analysis*, 26 (2007): 190-208.

[53] M. B. Cline and D. K. Pai. Post-stabilization for rigid body simulation with contact and constraints. *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*. IEEE, 2003. p. 3744-3751.

[54] D. Negrut and L. O. Jay, and N Khude. A discussion of low-order numerical integration formulas for rigid and flexible multibody dynamics. *Journal of Computational and Nonlinear Dynamics*, 4.2 (2009): 021008.

[55] J. Yen, L. Petzold, and S. Raha. A time integration algorithm for flexible mechanism dynamics: The DAE α-method. *Computer methods in applied mechanics and engineering*, 158.3 (1998): 341-355.

[56] E. V. Lens, A. Cardona, and M. Geradin. Energy preserving time integration for constrained multibody systems. *Multibody System Dynamics*, 11.1 (2004): 41-61.

[57] E. Eich. Convergence results for a coordinate projection method applied to mechanical systems with algebraic constraints. *SIAM Journal on Numerical Analysis*, 30.5 (1993): 1467-1482.

[58] M. Berzeri and A. A. Shabana. Development of simple models for the elastic forces in the absolute nodal co-ordinate formulation. *Journal of Sound and Vibration*, 235.4 (2000): 539-565.

[59] B. A. Hussein, H. Sugiyama, and A. A. Shabana. Coupled deformation modes in the finite element absolute nodal coordinate formulation. *12th IFToMM World Congress, Besancon, France, June*, 2007.

[60] E. M. Bakr and A. A. Shabana. Timoshenko beams and flexible multibody system dynamics. *Journal of sound and vibration*, 116.1 (1987): 89-107.

[61] D. Logan. A first course in the finite element method. *Cengage Learning*, page 174-179, 2011.

[62] H. Hahn and B. Simeon. "Separation principle of mechanical system models including stabilized constraint relations. *Archive of Applied Mechanics*, 64.3 (1994): 147-153.

[63] Zhu-Yong Liu, Jia-Zhen Hong, and Jin-Yang Liu. Finite element formulation for dynamics of planar flexible multi-beam system.*Multibody System Dynamics*, 22.1 (2009): 1-26.

[64] M. Arnold and O Brüls. Convergence of the generalized- scheme for constrained mechanical systems. *Multibody System Dynamics*, 18.2 (2007): 185-202.

[65] T. M. Wasfy and A. K. Noor. Computational strategies for flexible multibody systems. *Applied Mechanics Reviews* 56.6 (2003): 553-613.

Appendix A

Appendix

Each term in the equation of motion as well as constraint equation of rigid body dynamics are shown in the numerical example. As a result, for abbreviation, the mass matrix, stiffness matrix, load vector and constraint equation for flexible element under each formulation IRF/ANCF/FRF will be shown in this appendix.

A.1 Elastic Bar

A.1.1 Inertial Reference Frame Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \tag{A.1}$$

In the inertia reference formulation, the mass matrix is defined as

$$\mathbf{M} = m \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{3} \end{bmatrix} \tag{A.2}$$

where m is the mass of the element ρA and the internal force can be defined as

$$\mathbf{F}_{int} = EA\epsilon_l \begin{bmatrix} -\frac{q_3 - q_1}{d} \\ -\frac{q_4 - q_2}{d} \\ \frac{q_3 - q_1}{d} \\ \frac{q_4 - q_2}{d} \end{bmatrix} \tag{A.3}$$

where longitudinal strain ϵ_l and nodal distance d can be defined as

$$\epsilon_l = \frac{d - L}{L} \quad d = \sqrt{(q_3 - q_1)^2 + (q_4 - q_2)^2} \tag{A.4}$$

and the external force by gravity is evaluated as

$$\mathbf{F}_{ext} = \rho A g [0 \quad -\frac{1}{2} \quad 0 \quad -\frac{1}{2}]^T \tag{A.5}$$

Constraint formulation follows that two bodies share a node which for a simple pendulum, the constraint can be defined as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.6}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \\ \dot{q}_2 \end{bmatrix} \tag{A.7}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \\ \ddot{q}_2 \end{bmatrix} \tag{A.8}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.9}$$

For double pendulum case, it then becomes

$$\Phi(u) = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ q^{(2)}_1 - q^{(1)}_3 \\ q^{(2)}_2 - q^{(1)}_4 \end{bmatrix} \tag{A.10}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}^{(1)}_1 \\ \dot{q}^{(1)}_2 \\ \dot{q}^{(2)}_1 - \dot{q}^{(1)}_3 \\ \dot{q}^{(2)}_2 - \dot{q}^{(1)}_4 \end{bmatrix} \tag{A.11}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1^{(1)} \\ \ddot{q}_2^{(1)} \\ \ddot{q}_1^{(2)} - \ddot{q}_3^{(1)} \\ \ddot{q}_2^{(2)} - \ddot{q}_4^{(1)} \end{bmatrix} \tag{A.12}$$

The corresponding constraint matrix C is

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.13}$$

A.1.2 Simplified Absolute Nodal Coordinate Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \tag{A.14}$$

The global shape function of elastic bar element is

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & s_2 & 0 \\ 0 & s_1 & 0 & s_2 \end{bmatrix} \tag{A.15}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - \xi, \qquad s_2 = \xi \tag{A.16}$$

Mass matrix follows

$$\mathbf{M} = \rho A \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{3} \end{bmatrix} \tag{A.17}$$

and the stiffness matrix can be evaluated by defining the longitudinal strain along the element as

$$\epsilon_l = \frac{d - L}{L} \tag{A.18}$$

where d is the distance between the nodes of the elements and is defined as

$$d = \sqrt{(q_3 - q_1)^2 + (q_4 - q_2)^2} \tag{A.19}$$

Using the constant strain assumption, the stiffness matrix K can be written explicitly as

$$\mathbf{K} = \frac{EA}{L}\epsilon_l \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \tag{A.20}$$

and the internal force is the defined as

$$\mathbf{F}_{int} = \mathbf{K}q \tag{A.21}$$

and the external force by gravity is defined as

$$\mathbf{F}_{ext} = \rho A g [0 \quad -\frac{1}{2} \quad 0 \quad -\frac{1}{2}]^T \tag{A.22}$$

Constraint formulation follows that two bodies share a node which for simple pendulum, the constraint can be defined as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.23}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.24}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.25}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.26}$$

For double pendulum case, it then becomes

$$\Phi(u) = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ q^{(2)}_1 - q^{(1)}_3 \\ q^{(2)}_2 - q^{(1)}_4 \end{bmatrix} \tag{A.27}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1^{(1)} \\ \dot{q}_2^{(1)} \\ \dot{q}_1^{(2)} - \dot{q}_3^{(1)} \\ \dot{q}_2^{(2)} - \dot{q}_4^{(1)} \end{bmatrix} \tag{A.28}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1^{(1)} \\ \ddot{q}_2^{(1)} \\ \ddot{q}_1^{(2)} - \ddot{q}_3^{(1)} \\ \ddot{q}_2^{(2)} - \ddot{q}_4^{(1)} \end{bmatrix} \tag{A.29}$$

The corresponding constraint matrix C is

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.30}$$

A.1.3 Floating Reference Formulation

The position vector in floating reference formulation is

$$
\begin{aligned}
q &= \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \\
&= \begin{bmatrix} R_1 & R_2 & \theta & \hat{q}_1 \end{bmatrix}^T \\
&= \begin{bmatrix} q_r & q_f \end{bmatrix}^T
\end{aligned}
\tag{A.31}
$$

where $q_r = q(1:3)$ and $q_f = q(4)$. The local shape function for an elastic bar element is

$$
\mathbf{S}_l = \begin{bmatrix} \xi \\ 0 \end{bmatrix}
\tag{A.32}
$$

with the shape integral

$$
\begin{aligned}
\bar{S} &= \frac{m}{2} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
\mathbf{m}_{ff} &= \frac{m}{3} \\
\tilde{S} &= 0
\end{aligned}
\tag{A.33}
$$

and rotation matrix $\mathbf{A}$ and $\mathbf{A}_\theta$

$$
\mathbf{A} = \begin{bmatrix} \cos(q_3) & -\sin(q_3) \\ \sin(q_3) & \cos(q_3) \end{bmatrix}, \quad \mathbf{A}_\theta = \begin{bmatrix} -\sin(q_3) & -\cos(q_3) \\ \cos(q_3) & -\sin(q_3) \end{bmatrix}
\tag{A.34}
$$

In the floating reference formulation, the mass matrix is defined as

$$
\mathbf{M} = \begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \bar{S} q_f & \mathbf{A}\bar{S} \\ & q_f^T \mathbf{m}_{ff} q_f & q_f^T \tilde{S} \\ sym & & \mathbf{m}_{ff} \end{bmatrix} = \begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \begin{bmatrix} \frac{m}{2} \\ 0 \end{bmatrix} q_f & \mathbf{A} \begin{bmatrix} \frac{m}{2} \\ 0 \end{bmatrix} \\ & q_f^T \frac{m}{3} q_f & 0 \\ sym & & \frac{m}{3} \end{bmatrix}
\tag{A.35}
$$

where $m = \rho A$ and the stiffness matrix is then evaluated by

$$\mathbf{K}_{ff} = \frac{EA}{L}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.36}$$

The external force can be defined as two parts: external load $\mathbf{Q}_e$, and quadratic velocity vector $\mathbf{Q}_v$

$$\begin{aligned} (\mathbf{Q}_v)_R &= (\dot{\theta})^2\mathbf{A}(\bar{S}\,q_f + \mathbf{I}_1) - 2\theta\mathbf{A}_\theta\bar{S}\,q_f \\ (\mathbf{Q}_v)_\theta &= -2\dot{\theta}\dot{q}_f^T(\mathbf{m}_{ff}q_f + \mathbf{I}_0) \\ (\mathbf{Q}_v)_f &= (\dot{\theta})^2(\mathbf{m}_{ff}q_f + \mathbf{I}_0) + 2\dot{\theta}\bar{S}\,\dot{q}_f \end{aligned} \tag{A.37}$$

and yields

$$\mathbf{Q}_v = \begin{bmatrix} \dot{q}_3^2 * \mathbf{A}(\bar{S} * q_f + \begin{bmatrix} \frac{\rho AL^2}{2} \\ 0 \end{bmatrix}) - 2\dot{q}_3^2\mathbf{A}_\theta\bar{S}\,\dot{q}_f \\ -2\dot{q}_3\dot{q}_f^T(\mathbf{m}_{ff}q_f + \frac{\rho AL^2}{4}) \\ \dot{q}_3^2(\mathbf{m}_{ff}q_f + \frac{\rho AL^2}{4})) + 2\dot{q}_3\tilde{S}\,\dot{q}_f \end{bmatrix} \tag{A.38}$$

The external force for each mode in the generalized coordinate are respectively written as

$$\begin{aligned} (\mathbf{Q}_e)_R &= f_{ext}^T\mathbf{I} \\ (\mathbf{Q}_e)_\theta &= f_{ext}^T\mathbf{A}_\theta\mathbf{r}_c \\ (\mathbf{Q}_e)_f &= f_{ext}^T\mathbf{A}\mathbf{S}_l(\xi = 0.5) \end{aligned} \tag{A.39}$$

and f_{ext} is the gravity and yields

$$\mathbf{Q}_e = \begin{bmatrix} \begin{bmatrix} 0 & -\rho ALg \end{bmatrix}^T \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A}_\theta \left(\begin{bmatrix} L/2 \\ 0 \end{bmatrix} + \mathbf{S}_l(\xi = 0.5) q_f \right) \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A} \mathbf{S}_l(\xi = 0.5) \end{bmatrix} \tag{A.40}$$

The total external load is

$$\mathbf{F}_{ext}(q, \dot{q}) = \mathbf{Q}_v + \mathbf{Q}_e \tag{A.41}$$

For the simple pendulum, the position constraint for hinge joint connected to the ground can be written as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.42}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.43}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.44}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.45}$$

For the double pendulum case, the formulation of hinge joint connecting to ground remains the same. For the hinge connecting two bodies, one has to use the relation:

$$\mathbf{r} = \mathbf{R} + \mathbf{A}\mathbf{S}_l q_f \tag{A.46}$$

and the position constraint becomes

$$\Phi = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ \begin{bmatrix} q^{(2)}_1 \\ q^{(2)}_1 \end{bmatrix} - \mathbf{A}\mathbf{S}_l(\xi = 1) q_f \end{bmatrix} \tag{A.47}$$

and velocity constraint is

$$\dot{\Phi} = \begin{bmatrix} \dot{q}^{(1)}_1 \\ \dot{q}^{(1)}_2 \\ \begin{bmatrix} \dot{q}^{(2)}_1 \\ \dot{q}^{(2)}_2 \end{bmatrix} - \left(\begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l(\xi = 1) q^{(1)}_f & \mathbf{A}\mathbf{S}_l(\xi = 1) \end{bmatrix} \dot{q}^{(1)} \right) \end{bmatrix} \tag{A.48}$$

where $\mathbf{I}$ is identity matrix, and final the acceleration constraint is:

$$\ddot{\Phi} = \begin{bmatrix} \ddot{q}^{(1)}_1 \\ \ddot{q}^{(1)}_2 \\ \begin{bmatrix} \ddot{q}^{(2)}_1 \\ \ddot{q}^{(2)}_2 \end{bmatrix} - \left(\begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{A}_\theta \mathbf{S}_l \dot{q}^{(1)}_f & \mathbf{A}\mathbf{S}_l \dot{q}^{(1)}_3 \end{bmatrix} \dot{q}^{(1)} + \begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l q^{(1)}_f & \mathbf{A}\mathbf{S}_l \end{bmatrix} \ddot{q}^{(1)} \right) \end{bmatrix} \tag{A.49}$$

where

$$\mathbf{S}_l = \mathbf{S}_l(\xi = 1) \tag{A.50}$$

The corresponding constraint matrix C is

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & (L^{(1)} + q_4)\sin(q_3^{(1)}) & -\cos(q_3^{(1)}) & 1 & 0 & 0 & 0 \\ 0 & -1 & -(L^{(1)} + q_4)\cos(q_3^{(1)}) & -\sin(q_3^{(1)}) & 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.51}$$

A.2 Euler-Bernoulli Beam

A.2.1 Inertia Reference Frame Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 \end{bmatrix}^T \tag{A.52}$$

In the inertial reference formulation, the mass matrix is defined as

$$\mathbf{M} = m \begin{bmatrix} \frac{13}{35} & 0 & 0 & \frac{9}{70} & 0 & 0 \\ & \frac{13}{35} & 0 & 0 & \frac{9}{70} & 0 \\ & & \frac{L^2}{105} & 0 & 0 & -\frac{L^2}{140} \\ & & & \frac{13}{35} & 0 & 0 \\ & & & & \frac{13}{35} & 0 \\ sym & & & & & \frac{L^2}{105} \end{bmatrix} \tag{A.53}$$

where m is the mass of the bar ρAL, and the internal force can be defined as

$$\mathbf{F}_{int} = EA\epsilon_l \begin{bmatrix} -\frac{q_3-q_1}{d} \\ -\frac{q_4-q_2}{d} \\ 0 \\ \frac{q_3-q_1}{d} \\ \frac{q_4-q_2}{d} \\ 0 \end{bmatrix} + \frac{6EI}{Ld}(\epsilon_i + \epsilon_j) \begin{bmatrix} -\frac{q_5-q_2}{d} \\ -\frac{q_4-q_1}{d} \\ 0 \\ \frac{q_5-q_2}{d} \\ \frac{q_4-q_1}{d} \\ 0 \end{bmatrix} + \frac{2EI}{L} \begin{bmatrix} 0 \\ 0 \\ 2\epsilon_i + \epsilon_j \\ 0 \\ 0 \\ \epsilon_i + 2\epsilon_j \end{bmatrix} \tag{A.54}$$

where the longitudinal strain ϵ_l and nodal distance d can be defined as

$$\begin{aligned} \epsilon_l &= \frac{d-L}{L} \\ \epsilon_i &= q_3 - \alpha \\ \epsilon_j &= q_6 - \alpha \end{aligned} \tag{A.55}$$

with the following change

$$\begin{aligned} d &= \sqrt{(q_4-q_1)^2 + (q_5-q_2)^2} \\ \alpha &= cos^{-1}\left[\frac{q_4-q_1}{d}\right] = sin^{-1}\left[\frac{q_5-q_2}{d}\right] \end{aligned} \tag{A.56}$$

and the external force by gravity is evaluated as

$$\mathbf{F}_{ext} = \rho Ag[0 \quad -\frac{1}{2} \quad \frac{L}{12} \quad 0 \quad -\frac{1}{2} \quad -\frac{L}{12}]^T \tag{A.57}$$

Constraint formulation follows that two bodies share a node which for the simple pendulum, the constraint can be defined as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ \\ q_2 \end{bmatrix} \tag{A.58}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \\ \dot{q}_2 \end{bmatrix} \tag{A.59}$$

and finally the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \\ \ddot{q}_2 \end{bmatrix} \tag{A.60}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.61}$$

For double pendulum case, it then become

$$\Phi(u) = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ q^{(2)}_1 - q^{(1)}_4 \\ q^{(2)}_2 - q^{(1)}_5 \end{bmatrix} \tag{A.62}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1^{(1)} \\ \dot{q}_2^{(1)} \\ \dot{q}_1^{(2)} - \dot{q}_4^{(1)} \\ \dot{q}_2^{(2)} - \dot{q}_5^{(1)} \end{bmatrix} \tag{A.63}$$

and, finally the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1^{(1)} \\ \ddot{q}_2^{(1)} \\ \ddot{q}_1^{(2)} - \ddot{q}_4^{(1)} \\ \ddot{q}_2^{(2)} - \ddot{q}_5^{(1)} \end{bmatrix} \tag{A.64}$$

The corresponding constraint matrix C is 4×12 sparse zeros matrix with entries as:

$$\begin{aligned} &\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,4) = -1 \\ &\mathbf{C}(7,9) = -1, \quad \mathbf{C}(4,5) = 1, \quad \mathbf{C}(4,8) = 1 \end{aligned} \tag{A.65}$$

A.2.2 Simplified Absolute Nodal Coordinate Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \tag{A.66}$$

The global shape function of Euler-Bernoulli beam element is

$$\mathbf{S} = \begin{bmatrix} s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 & 0 \\ 0 & s_1 & 0 & s_2 & 0 & s_3 & 0 & s_4 \end{bmatrix} \tag{A.67}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$s_1 = 1 - 3\xi^2 + 2\xi^3, \qquad s_2 = L(\xi - 2\xi^2 + \xi^3), \qquad s_3 = 3\xi^2 - 2\xi^3, \qquad s_4 = L(\xi^3 - \xi^2) \quad \text{(A.68)}$$

and $\xi = x/L$. Integral element $\mathbf{S}_l$ is

$$\mathbf{S}_l = \frac{1}{L^2}\mathbf{S}_{,\xi}^T\mathbf{S}_{,\xi}, \quad \bar{S}_l = \int_0^1 \mathbf{S}_l d\xi \tag{A.69}$$

and the mass matrix $\mathbf{M}$ yields

$$\mathbf{M} = \rho A g \begin{bmatrix} \frac{13}{35} & 0 & \frac{11L}{210} & 0 & \frac{9}{70} & 0 & -\frac{13L}{420} & 0 \\ & \frac{13}{35} & 0 & \frac{11L}{210} & 0 & \frac{9}{70} & 0 & -\frac{13L}{420} \\ & & \frac{L^2}{105} & 0 & \frac{13L}{420} & 0 & -\frac{L^2}{140} & 0 \\ & & & \frac{L^2}{105} & 0 & \frac{13L}{420} & 0 & -\frac{L^2}{140} \\ & & & & \frac{13}{35} & 0 & \frac{11L}{210} & 0 \\ & & & & & \frac{13}{35} & 0 & \frac{11L}{210} \\ & & & & & & \frac{L^2}{105} & 0 \\ sym & & & & & & & \frac{L^2}{105} \end{bmatrix} \tag{A.70}$$

Recall that the stiffness matrix $\mathbf{K}$ can be separated into two parts: longitudinal deformation $\mathbf{K}_l$ and transverse deformation $\mathbf{K}_t$. Note that the position vector q of nodal coordinates is written as the sum of two vectors,

$$q = q_r + q_f \tag{A.71}$$

where q_r represents an arbitrary rigid-body displacement,

$$q_r = [x \quad y \quad c \quad s \quad x + Lc \quad y + Ls \quad c \quad s]^T \tag{A.72}$$

and q_f is the result of the difference $q - q_r$. in equation (), x and y are arbitrary rigid body translations, and c and s stand for $cos\theta$ and $sin\theta$, respectively, where θ represents an arbitrary rigid body rotation. One can easily show that

$$q_r^T \mathbf{S}_l q_r = 1 \tag{A.73}$$

The longitudinal strain becomes

$$\epsilon_l = \frac{1}{2}(q - q_r)^T \mathbf{S}_l (q + q_r) \tag{A.74}$$

and the stiffness matrix K_l from Eq. (3.50), assuming that E and A are constant, is

$$\mathbf{K}_l = EAL \int_0^1 \epsilon_l \mathbf{S}_l d\xi \tag{A.75}$$

Since q_r is arbitrary, one may conveniently choose $q_r = [0 \quad 0 \quad 1 \quad 0 \quad L \quad 0 \quad 1 \quad 0]^T$. Using this vector, the stiffness matrix can be explicitly written as

$$\mathbf{K}_l = \frac{EA}{L}\begin{bmatrix} \mathbb{A} & 0 & \mathbb{B} & 0 & -\mathbb{A} & 0 & \mathbb{C} & 0 \\ & \mathbb{A} & 0 & \mathbb{B} & 0 & -\mathbb{A} & 0 & \mathbb{C} \\ & & \mathbb{D} & 0 & -\mathbb{B} & 0 & \mathbb{E} & 0 \\ & & & \mathbb{D} & 0 & -\mathbb{B} & 0 & \mathbb{E} \\ & & & & \mathbb{A} & 0 & -\mathbb{C} & 0 \\ & & & & & \mathbb{A} & 0 & -\mathbb{C} \\ & & & & & & \mathbb{F} & 0 \\ sym & & & & & & & \mathbb{F} \end{bmatrix} \tag{A.76}$$

which contains six independent elements $\mathbb{A}$, $\mathbb{B}$, $\mathbb{C}$, $\mathbb{D}$, $\mathbb{E}$, $\mathbb{F}$. In order to write the expression of these six components, it is convenient to introduce the quantities

$$d_x = q_5 - q_1, \quad d_y = q_6 - q_2 \tag{A.77}$$

where $d = \sqrt{d_x^2 + d_y^2}$ is also defined, and

$$\begin{aligned} a_x &= Lq_3, \quad a_y = Lq_4, \quad a = \sqrt{a_x^2 + a_y^2} \\ b_x &= Lq_7, \quad b_y = Lq_8, \quad b = \sqrt{b_x^2 + b_y^2} \end{aligned} \tag{A.78}$$

Using these definitions, the six independent elements can be written as

$$\begin{aligned} \mathbb{A} &= \frac{3}{70L^2}(a^2 + b^2 - 14L^2 - 6a_xd_x - 6b_xd_x - 6a_yd_y - 6b_yd_y + 24d^2) \\ \mathbb{B} &= \frac{1}{280L^2}(b^2 - a^2 + 2a_xb_x + 2a_yb_y - 14L^2 - 24a_xd_x - 24a_yd_y + 36d^2) \\ \mathbb{C} &= \frac{1}{280L^2}(a^2 - b^2 + 2a_xb_x + 2a_yb_y - 14L^2 - 24b_xd_x - 24b_yd_y + 36d^2) \end{aligned} \tag{A.79}$$

$$\mathbb{D} = \frac{1}{420}(12a^2 + b^2 - 3a_x b_x - 3a_y b_y - 28L^2 + 3a_x d_x - 3b_x d_x + 3a_y d_y - 3b_y d_y + 18d^2)$$
$$\mathbb{E} = \frac{-1}{840}(3a^2 + 3b^2 - 4a_x b_x - 4a_y b_y - 14L^2 + 6a_x d_x + 6b_x d_x + 6a_y d_y + 6b_y d_y)$$
$$\mathbb{F} = \frac{1}{420}(a^2 + 12b^2 - 3a_x b_x - 3a_y b_y - 28L^2 - 3a_x d_x + 3b_x d_x - 3a_y d_y + 3b_y d_y + 18d^2) \tag{A.80}$$

For the simplified expression of the stiffness matrix due to transverse deformation $\mathbf{K}_t$, it can be obtained, assuming **E** and **I** are constant, as

$$\mathbf{K}_t = \frac{EI}{L^3}\begin{bmatrix} 12 & 0 & 6L & 0 & -12 & 0 & 6L & 0 \\ & 12 & 0 & 6L & 0 & -12 & 0 & 6L \\ & & 4L^2 & 0 & -6L & 0 & 2L^2 & 0 \\ & & & 4L^2 & 0 & -6L & 0 & 2L^2 \\ & & & & 12 & 0 & -6L & 0 \\ & & & & & 12 & 0 & -6L \\ & & & & & & 4L^2 & 0 \\ sym & & & & & & & 4L^2 \end{bmatrix} \tag{A.81}$$

This constant matrix is similar to the stiffness matrix used in linear structural dynamics. In sum, the simplified stiffness matrix for Euler-Bernoulli beam is then

$$\begin{aligned} \mathbf{K} &= \mathbf{K}_l + \mathbf{K}_t \\ \mathbf{F}_{int} &= \mathbf{K}q \end{aligned} \tag{A.82}$$

The external load vector due to gravity, assuming constant ρ and **A**, is then defined as

$$\mathbf{F}_{ext} = \rho A g[0 \quad -\frac{1}{2} \quad 0 \quad \frac{L}{12} \quad 0 \quad -\frac{1}{2} \quad 0 \quad -\frac{L}{12}]^T \tag{A.83}$$

Finally, the constraint equation formulation is then simply two element share one node, which for simple pendulum case the position constraint is:

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.84}$$

and velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{r}_1 \\ \dot{r}_2 \end{bmatrix} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.85}$$

and finally the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{r}_1 \\ \ddot{r}_2 \end{bmatrix} = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.86}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A.87}$$

For double pendulum case, it then becomes

$$\Phi(u) = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ q^{(2)}_1 - q^{(1)}_5 \\ q^{(2)}_2 - q^{(1)}_6 \end{bmatrix} \tag{A.88}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1^{(1)} \\ \dot{q}_2^{(1)} \\ \dot{q}_1^{(2)} - \dot{q}_5^{(1)} \\ \dot{q}_2^{(2)} - \dot{q}_6^{(1)} \end{bmatrix} \tag{A.89}$$

and, finally the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1^{(1)} \\ \ddot{q}_2^{(1)} \\ \ddot{q}_1^{(2)} - \ddot{q}_5^{(1)} \\ \ddot{q}_2^{(2)} - \ddot{q}_6^{(1)} \end{bmatrix} \tag{A.90}$$

The corresponding constraint matrix C is 4×16 sparse zeros matrix with entries as follows:

$$\begin{aligned} &\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,5) = -1 \\ &\mathbf{C}(3,9) = -1, \quad \mathbf{C}(4,6) = 1, \quad \mathbf{C}(4,10) = 1 \end{aligned} \tag{A.91}$$

A.2.3 Floating Reference Formulation

The position vector of the Euler-Bernoulli beam can defined as

$$\begin{aligned} q &= \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \\ &= \begin{bmatrix} R_1 & R_2 & \theta & \hat{q}_1 & \hat{q}_2 & \hat{q}_3 & \hat{q}_4 & \hat{q}_5 \end{bmatrix}^T \\ &= \begin{bmatrix} q_r & q_f \end{bmatrix}^T \end{aligned} \tag{A.92}$$

where $q_r = q(1:3)$ and $q_f = q(4:8)$. The local shape function for Euler-Bernoulli beam is

$$\mathbf{S}_l = \begin{bmatrix} L(\xi - 2\xi^2 + \xi^3) & 3\xi^2 - 2\xi^3 & 0 & L(\xi^3 - \xi^2) & 0 \\ 0 & 0 & 3\xi^2 - 2\xi^3 & 0 & L(\xi^3 - \xi^2) \end{bmatrix} \tag{A.93}$$

and the shape integral that appears in the floating frame formulation are

$$\begin{aligned} \bar{S} &= \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 \\ 0 & 0 & \frac{m}{2} & 0 & -\frac{mL}{12} \end{bmatrix} \\ \mathbf{m}_{ff} &= m \begin{bmatrix} \frac{L^2}{105} & \frac{13L}{420} & 0 & -\frac{L^2}{140} & 0 \\ & \frac{13}{35} & 0 & -\frac{11L}{210} & 0 \\ & & \frac{13}{35} & 0 & -\frac{11L}{210} \\ & & & \frac{L^2}{105} & 0 \\ & \textit{symmetric} & & & \frac{L^2}{105} \end{bmatrix} \\ \tilde{S} &= \begin{bmatrix} 0 & 0 & \frac{13L}{420} & 0 & -\frac{L^2}{140} \\ 0 & 0 & \frac{13}{35} & 0 & -\frac{11L}{210} \\ \frac{13L}{420} & -\frac{13}{35} & 0 & \frac{11L}{210} & 0 \\ 0 & 0 & -\frac{11L}{210} & 0 & \frac{L^2}{105} \\ \frac{L^2}{140} & \frac{11L}{210} & 0 & -\frac{L^2}{105} & 0 \end{bmatrix} \end{aligned} \tag{A.94}$$

with rotation matrix $\mathbf{A}$ and $\mathbf{A}_\theta$:

$$\mathbf{A} = \begin{bmatrix} \cos q_3 & -\sin q_3 \\ \sin q_3 & \cos q_3 \end{bmatrix}, \quad \mathbf{A}_\theta = \begin{bmatrix} -\sin q_3 & -\cos q_3 \\ \cos q_3 & -\sin q_3 \end{bmatrix} \tag{A.95}$$

In the floating reference formulation, the mass matrix is defined as

$$\mathbf{M} = \begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \bar{S} q_f & \mathbf{A}\bar{S} \\ & q_f^T \mathbf{m}_{ff} q_f & q_f^T \tilde{S} \\ sym & & \mathbf{m}_{ff} \end{bmatrix} =$$

$$\begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 & \\ 0 & 0 & \frac{m}{2} & 0 & & -\frac{mL}{12} \end{bmatrix} q_f & \mathbf{A} \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 & \\ 0 & 0 & \frac{m}{2} & 0 & & -\frac{mL}{12} \end{bmatrix} \\ & mq_f^T \begin{bmatrix} \frac{L^2}{105} & \frac{13L}{420} & 0 & -\frac{L^2}{140} & 0 \\ & \frac{13}{35} & 0 & -\frac{11L}{210} & 0 \\ & & \frac{13}{35} & 0 & -\frac{11L}{210} \\ & & & \frac{L^2}{105} & 0 \\ sym & & & & \frac{L^2}{105} \end{bmatrix} q_f & mq_f^T \begin{bmatrix} 0 & 0 & \frac{13L}{420} & 0 & -\frac{L^2}{140} \\ 0 & 0 & \frac{13}{35} & 0 & -\frac{11L}{210} \\ \frac{13L}{420} & -\frac{13}{35} & 0 & \frac{11L}{210} & 0 \\ 0 & 0 & -\frac{11L}{210} & 0 & \frac{L^2}{105} \\ \frac{L^2}{140} & \frac{11L}{210} & 0 & -\frac{L^2}{105} & 0 \end{bmatrix} \\ & & m \begin{bmatrix} \frac{L^2}{105} & \frac{13L}{420} & 0 & -\frac{L^2}{140} & 0 \\ & \frac{13}{35} & 0 & -\frac{11L}{210} & 0 \\ & & \frac{13}{35} & 0 & -\frac{11L}{210} \\ & & & \frac{L^2}{105} & 0 \\ & symmetric & & & \frac{L^2}{105} \end{bmatrix} \end{bmatrix}$$

(A.96)

And the stiffness matrix is then evaluated as

$$\mathbf{K}_{ff} = \frac{EI}{L^3}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4L^2 & -6L & 0 & 2L^2 & 0 \\ 0 & 0 & 0 & -6L & 12 & 0 & -6L & 0 \\ 0 & 0 & 0 & 0 & 0 & 12 & 0 & -6L \\ 0 & 0 & 0 & 2L^2 & -6L & 0 & 4L^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -6L & 0 & 4L^2 \end{bmatrix} \tag{A.97}$$

The external force can be defined as two parts: external load $\mathbf{Q}_e$, and quadratic velocity vector $\mathbf{Q}_v$ as

$$\begin{aligned} (\mathbf{Q}_v)_R &= (\dot{\theta})^2 \mathbf{A}(\bar{S} q_f + \mathbf{I}_1) - 2\theta \mathbf{A}_\theta \bar{S} q_f \\ (\mathbf{Q}_v)_\theta &= -2\dot{\theta} \dot{q}_f^T (\mathbf{m}_{ff} q_f + \mathbf{I}_0) \\ (\mathbf{Q}_v)_f &= (\dot{\theta})^2 (\mathbf{m}_{ff} q_f + \mathbf{I}_0) + 2\dot{\theta} \bar{S} \dot{q}_f \end{aligned} \tag{A.98}$$

and yields

$$\mathbf{Q}_v = \begin{bmatrix} \dot{q}_3^2 * \mathbf{A}(\bar{S} * q_f + \begin{bmatrix} \frac{\rho AL^2}{2} \\ 0 \end{bmatrix}) - 2\dot{q}_3^2 \mathbf{A}_\theta \bar{S}\, \dot{q}_f \\ -2\dot{q}_3 \dot{q}_f^T (\mathbf{m}_{ff} q_f + \begin{bmatrix} \frac{\rho AL^3}{24} \\ \frac{\rho AL^2}{4} \\ 0 \\ -\frac{\rho AL^3}{24} \\ 0 \end{bmatrix} \\ \dot{q}_3^2 (\mathbf{m}_{ff} q_f + \begin{bmatrix} \frac{\rho AL^3}{24} \\ \frac{\rho AL^2}{4} \\ 0 \\ -\frac{\rho AL^3}{24} \\ 0 \end{bmatrix}) + 2\dot{q}_3 \tilde{S}\, \dot{q}_f \end{bmatrix} \tag{A.99}$$

The external force for each mode in the generalized coordinate are respectively written as

$$\begin{aligned} (\mathbf{Q}_e)_R &= f_{ext}^T \mathbf{I} \\ (\mathbf{Q}_e)_\theta &= f_{ext}^T \mathbf{A}_\theta \mathbf{r}_c \\ (\mathbf{Q}_e)_f &= f_{ext}^T \mathbf{A} \mathbf{S}_l(\xi = 0.5) \end{aligned} \tag{A.100}$$

For f_{ext} as gravity, yields

$$\mathbf{Q}_e = \begin{bmatrix} \begin{bmatrix} 0 & -\rho ALg \end{bmatrix}^T \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A}_\theta (\begin{bmatrix} L/2 \\ 0 \end{bmatrix} + \mathbf{S}_l(\xi = 0.5) q_f) \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A} \mathbf{S}_l(\xi = 0.5) \end{bmatrix} \tag{A.101}$$

The total external load is

$$\mathbf{F}_{ext}(q, \dot{q}) = \mathbf{Q}_v + \mathbf{Q}_e \tag{A.102}$$

The position constraint for hinge joint connect to ground can be written as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.103}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.104}$$

and, finally the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.105}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.106}$$

For double pendulum case, the formulation of hinge joint connecting to the ground remains the same. For the hinge connecting two bodies, one has to use the relation:

$$\mathbf{r} = \mathbf{R} + \mathbf{A}\mathbf{S}_l q_f \tag{A.107}$$

and the position constraint becomes

$$\Phi = \begin{bmatrix} r_{1x}^1 \\ r_{1y}^1 \\ r_{1x}^2 - r_{2x}^1 \\ r_{1y}^2 - r_{2y}^1 \end{bmatrix} = \begin{bmatrix} q_1^{(1)} \\ q_2^{(1)} \\ \begin{bmatrix} q_1^{(2)} \\ q_1^{(2)} \end{bmatrix} - \mathbf{A}\mathbf{S}_l(\xi = 1)q_f \end{bmatrix} \tag{A.108}$$

and velocity constraint is

$$\dot{\Phi} = \begin{bmatrix} \dot{q}_1^{(1)} \\ \dot{q}_2^{(1)} \\ \begin{bmatrix} \dot{q}_1^{(2)} \\ \dot{q}_2^{(2)} \end{bmatrix} - \left(\begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l(\xi = 1)q_f^{(1)} & \mathbf{A}\mathbf{S}_l(\xi = 1) \end{bmatrix} \dot{q}^{(1)} \right) \end{bmatrix} \tag{A.109}$$

where $\mathbf{I}$ is identity matrix, and final the acceleration constraint is:

$$\ddot{\Phi} = \begin{bmatrix} \ddot{q}_1^{(1)} \\ \ddot{q}_2^{(1)} \\ \begin{bmatrix} \ddot{q}_1^{(2)} \\ \ddot{q}_2^{(2)} \end{bmatrix} - \left(\begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{A}_\theta \mathbf{S}_l \dot{q}_f^{(1)} & \mathbf{A}\mathbf{S}_l \dot{q}_3^{(1)} \end{bmatrix} \dot{q}^{(1)} + \begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l q_f^{(1)} & \mathbf{A}\mathbf{S}_l \end{bmatrix} \ddot{q}^{(1)} \right) \end{bmatrix} \tag{A.110}$$

where

$$\mathbf{S}_l = \mathbf{S}_l(\xi = 1) \tag{A.111}$$

The corresponding constraint matrix C is a 4×16 zeros matrix with non-zero entries:

$$\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,1) = -1$$

$$\mathbf{C}(3,9) = -1, \quad \mathbf{C}(4,2) = -1, \quad \mathbf{C}(4,10) = -1$$

$$\mathbf{C}(3,3) = (L^{(1)} + q_5^{(1)})\sin(q_3^{(1)}) - (-q_6^{(1)}\cos(q_3^{(1)})), \quad \mathbf{C}(3,5) = \cos(q_3^{(1)}), \quad \mathbf{C}(3,6) = \sin(q_3^{(1)})$$

$$\mathbf{C}(4,3) = (L^{(1)} + q_5^{(1)})\cos(q_3^{(1)}) + (q_6^{(1)}\sin(q_3^{(1)})), \quad \mathbf{C}(4,5) = -\sin(q_3^{(1)}), \quad \mathbf{C}(3,6) = -\cos(q_3^{(1)}) \tag{A.112}$$

A.3 Timoshenko beam

A.3.1 Inertia Reference Frame Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 \end{bmatrix}^T \tag{A.113}$$

In the inertial reference formulation, the mass matrix is defined as

$$\mathbf{M} = \frac{m}{(1+\phi)^2} \begin{bmatrix} \mathbb{M}_1 & 0 & 0 & \mathbb{M}_3 & 0 & 0 \\ & \mathbb{M}_1 & 0 & 0 & \mathbb{M}_3 & 0 \\ & & \mathbb{M}_5 & 0 & 0 & \mathbb{M}_6 \\ & & & \mathbb{M}_1 & 0 & 0 \\ & & & & \mathbb{M}_1 & 0 \\ sym & & & & & \mathbb{M}_5 \end{bmatrix} \tag{A.114}$$

where $\mathbb{M}_1$ to $\mathbb{M}_6$ can be explicitly expressed as:

$$\begin{aligned}
\mathbb{M}_1 &= \left(\frac{13}{35} + \frac{7\phi}{10} + \frac{\phi^2}{3}\right), \quad \mathbb{M}_2 = L\left(\frac{11}{210} + \frac{11\phi}{120} + \frac{\phi^2}{24}\right) \\
\mathbb{M}_3 &= \left(\frac{9}{70} + \frac{30\phi}{10} + \frac{\phi^2}{6}\right), \quad \mathbb{M}_4 = -L\left(\frac{13}{420} + \frac{3\phi}{40} + \frac{\phi^2}{24}\right) \\
\mathbb{M}_5 &= \left(\frac{1}{105} + \frac{\phi}{60} + \frac{\phi^2}{120}\right), \quad \mathbb{M}_6 = -L\left(\frac{1}{140} + \frac{\phi}{60} + \frac{\phi^2}{120}\right)
\end{aligned} \tag{A.115}$$

and the internal force can be defined as

$$\mathbf{F}_{int} = EA\epsilon_l \begin{bmatrix} -\frac{q_3-q_1}{d} \\ -\frac{q_4-q_2}{d} \\ 0 \\ \frac{q_3-q_1}{d} \\ \frac{q_4-q_2}{d} \\ 0 \end{bmatrix} + EI(\mathbf{J}_{\epsilon_i}) \begin{bmatrix} -\frac{q_5-q_2}{d} \\ -\frac{q_4-q_1}{d} \\ 0 \\ \frac{q_5-q_2}{d} \\ \frac{q_4-q_1}{d} \\ 0 \end{bmatrix} + \frac{EI}{2} \begin{bmatrix} 0 \\ 0 \\ \mathbf{J}_{\epsilon} \\ 0 \\ 0 \\ \mathbf{J}_{\epsilon} \end{bmatrix} \tag{A.116}$$

where $\mathbf{J}_{\epsilon}$ can be defined as

$$\mathbf{J}_{\epsilon} = \frac{24(\epsilon_i + \epsilon_j)}{L^4(1+\phi)^2} - \frac{12((\frac{\phi}{2}+2)\epsilon_i + (\frac{\phi}{2}-1)\epsilon_j)}{L^3(1+\phi)^2} - \frac{12(\epsilon_i+\epsilon_j)(\frac{\phi}{2}+2)}{L^3(1+\phi)^2} + \frac{8(\frac{\phi}{2}+2)((\frac{\phi}{2}+2)\epsilon_i + (\frac{\phi}{2}-1)\epsilon_j)}{L^2(1+\phi)^2} \tag{A.117}$$

The longitudinal strain ϵ_l and nodal distance d can be defined as

$$\begin{aligned}
\epsilon_l &= \frac{d-L}{L} \\
\epsilon_i &= q_3 - \alpha \\
\epsilon_j &= q_6 - \alpha
\end{aligned} \tag{A.118}$$

with following change

$$
\begin{aligned}
d &= \sqrt{(q_4 - q_1)^2 + (q_5 - q_2)^2} \\
\alpha &= cos^{-1}\left[\frac{q_4 - q_1}{d}\right] = sin^{-1}\left[\frac{q_5 - q_2}{d}\right]
\end{aligned}
\tag{A.119}
$$

and the external force by gravity is evaluated as

$$
\mathbf{F}_{ext} = \rho A g[0 \quad -\frac{1}{2} \quad \frac{L}{12} \quad 0 \quad -\frac{1}{2} \quad -\frac{L}{12}]^T \tag{A.120}
$$

The constraint formulation follows that two bodies share a node, which for the simple pendulum, the constraint can be defined as

$$
\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.121}
$$

and velocity constraint is

$$
\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.122}
$$

and the acceleration constraint is:

$$
\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.123}
$$

The corresponding constraint matrix C follows as:

$$
\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.124}
$$

For the double pendulum case, it is then become

$$\Phi(u) = \begin{bmatrix} r_{1x}^{1} \\ r_{1y}^{1} \\ r_{1x}^{2} - r_{2x}^{1} \\ r_{1y}^{2} - r_{2y}^{1} \end{bmatrix} = \begin{bmatrix} q_{1}^{(1)} \\ q_{2}^{(1)} \\ q_{1}^{(2)} - q_{4}^{(1)} \\ q_{2}^{(2)} - q_{5}^{(1)} \end{bmatrix} \tag{A.125}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_{1}^{(1)} \\ \dot{q}_{2}^{(1)} \\ \dot{q}_{1}^{(2)} - \dot{q}_{4}^{(1)} \\ \dot{q}_{2}^{(2)} - \dot{q}_{5}^{(1)} \end{bmatrix} \tag{A.126}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_{1}^{(1)} \\ \ddot{q}_{2}^{(1)} \\ \ddot{q}_{1}^{(2)} - \ddot{q}_{4}^{(1)} \\ \ddot{q}_{2}^{(2)} - \ddot{q}_{5}^{(1)} \end{bmatrix} \tag{A.127}$$

The corresponding constraint matrix C is 4×12 sparse zeros matrix will following entries follows as:

$$\begin{aligned} &\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,4) = -1 \\ &\mathbf{C}(7,9) = -1, \quad \mathbf{C}(4,5) = 1, \quad \mathbf{C}(4,8) = 1 \end{aligned} \tag{A.128}$$

A.3.2 Simplified Absolute Nodal Coordinate Formulation

Assume the position vector is defined as:

$$q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \tag{A.129}$$

The global shape function of Timoshenko beam element is

$$\mathbf{S}(\xi,\eta) = \frac{1}{1+\varphi}\begin{bmatrix} s_1 & t_1 & s_2 & t_2 & s_3 & t_3 & s_4 & t_4 \\ t_1 & s_1 & t_2 & s_2 & t_3 & s_3 & t_4 & s_4 \end{bmatrix} \tag{A.130}$$

where the functions $s_i = s_i(\xi)$ are defined as

$$\begin{aligned} &s_1 = 1 - 3\xi^2 + 2\xi^3 + (1-\xi)\varphi, \qquad s_2 = L[\xi - 2\xi^2 + \xi^3 + \frac{1}{2}(\xi - \xi^2)\varphi] \\ &s_3 = 3\xi^2 - 2\xi^3 + \xi\varphi, \qquad s_4 = L[\xi^3 - \xi^2 - \frac{1}{2}(\xi - \xi^2)\varphi] \\ &t_1 = 6(\xi - \xi^2)\eta, \qquad t_2 = L[-1 + 4\xi - 3\xi^2 - (1-\xi)\varphi]\eta \\ &t_3 = 6(-\xi + \xi^2)\eta, \qquad t_4 = L[2\xi - 3\xi^2 - \xi\varphi]\eta \end{aligned} \tag{A.131}$$

and $\xi = x/L$ and $\eta = y/L$. In this research, we use elastic center line approach which means $\nu = 0$. Integral element $\mathbf{S}_l$ is

$$\mathbf{S}_l = \frac{1}{L^2}\mathbf{S}_{,\xi}^T\mathbf{S}_{,\xi}, \quad \bar{S}_l = \int_0^1 \mathbf{S}_l d\xi \tag{A.132}$$

and the mass matrix $\mathbf{M}$ yields

$$\mathbf{M} = \frac{m}{(1+\phi)^2}\begin{bmatrix} \mathbb{M}_1 & 0 & \mathbb{M}_2 & 0 & \mathbb{M}_3 & 0 & \mathbb{M}_4 & 0 \\ & \mathbb{M}_1 & 0 & \mathbb{M}_2 & 0 & \mathbb{M}_3 & 0 & \mathbb{M}_4 \\ & & \mathbb{M}_5 & 0 & -\mathbb{M}_4 & 0 & \mathbb{M}_6 & 0 \\ & & & \mathbb{M}_5 & 0 & -\mathbb{M}_4 & 0 & \mathbb{M}_6 \\ & & & & \mathbb{M}_1 & 0 & -\mathbb{M}_2 & 0 \\ & & & & & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ & & & & & & \mathbb{M}_5 & 0 \\ sym & & & & & & & \mathbb{M}_5 \end{bmatrix} \tag{A.133}$$

where $\mathbb{M}_1$ to $\mathbb{M}_6$ can be explicitly expressed as:

$$\begin{aligned} \mathbb{M}_1 &= \left(\frac{13}{35} + \frac{7\phi}{10} + \frac{\phi^2}{3}\right), \quad \mathbb{M}_2 = L\left(\frac{11}{210} + \frac{11\phi}{120} + \frac{\phi^2}{24}\right) \\ \mathbb{M}_3 &= \left(\frac{9}{70} + \frac{30\phi}{10} + \frac{\phi^2}{6}\right), \quad \mathbb{M}_4 = -L\left(\frac{13}{420} + \frac{3\phi}{40} + \frac{\phi^2}{24}\right) \\ \mathbb{M}_5 &= \left(\frac{1}{105} + \frac{\phi}{60} + \frac{\phi^2}{120}\right), \quad \mathbb{M}_6 = -L\left(\frac{1}{140} + \frac{\phi}{60} + \frac{\phi^2}{120}\right) \end{aligned} \tag{A.134}$$

Recall that the stiffness matrix $\mathbf{K}$ can be separated into two parts: longitudinal deformation $\mathbf{K}_l$ and transverse deformation $\mathbf{K}_t$. Note that the position vector q of nodal coordinates is written as the sum of two vectors,

$$q = q_r + q_f \tag{A.135}$$

where q_r represents an arbitrary rigid-body displacement,

$$q_r = [x \quad y \quad c \quad s \quad x + Lc \quad y + Ls \quad c \quad s]^T \tag{A.136}$$

and q_f is the result of the difference $q - q_r$. x and y are arbitrary rigid body translations, and c and s stand for $cos\theta$ and $sin\theta$, respectively, where θ represents an arbitrary rigid body rotation. One can easily show that

$$q_r^T \mathbf{S}_l q_r = 1 \tag{A.137}$$

The longitudinal strain becomes

$$\epsilon_l = \frac{1}{2}(q - q_r)^T \mathbf{S}_l (q + q_r) \tag{A.138}$$

and the stiffness matrix K_l, assuming that E and A are constant, is

$$\mathbf{K}_l = EAL \int_0^1 \epsilon_l \mathbf{S}_l d\xi \tag{A.139}$$

For the simplified expression of stiffness matrix due to transverse deformation $\mathbf{K}_t$, it can be obtained, assuming $\mathbf{E}$ and $\mathbf{I}$ are constant, as

$$\mathbf{K}_t = \frac{EI}{L^3} \begin{bmatrix} \mathbb{K}_1 & 0 & \mathbb{K}_2 & 0 & -\mathbb{K}_1 & 0 & \mathbb{K}_2 & 0 \\ & \mathbb{K}_1 & 0 & \mathbb{K}_2 & 0 & -\mathbb{K}_1 & 0 & \mathbb{K}_2 \\ & & \mathbb{K}_3 & 0 & -\mathbb{K}_2 & 0 & -\mathbb{K}_4 & 0 \\ & & & \mathbb{K}_3 & 0 & -\mathbb{K}_2 & 0 & -\mathbb{K}_4 \\ & & & & \mathbb{K}_1 & 0 & -\mathbb{K}_2 & 0 \\ & & & & & \mathbb{K}_1 & 0 & -\mathbb{K}_2 \\ & & & & & & \mathbb{K}_3 & 0 \\ sym & & & & & & & \mathbb{K}_3 \end{bmatrix} \tag{A.140}$$

where $\mathbb{K}_1$ to $\mathbb{K}_4$ can be explicitly expressed as:

$$\begin{aligned} \mathbb{K}_1 &= \frac{12EI}{L^3(1+\phi)}, \quad \mathbb{K}_2 = \frac{6EI}{L^2(1+\phi)} \\ \mathbb{K}_3 &= \frac{(4+\phi)EI}{L(1+\phi)}, \quad \mathbb{K}_2 = \frac{(2-\phi)EI}{L(1+\phi)} \end{aligned} \tag{A.141}$$

This constant matrix is similar to the stiffness matrix used in linear structural dynamics. In sum, the simplified stiffness matrix for Timoshenko beam is then

$$\begin{aligned} \mathbf{K} &= \mathbf{K}_l + \mathbf{K}_t \\ \mathbf{F}_{int} &= \mathbf{K}q \end{aligned} \tag{A.142}$$

The external load vector due to gravity, assuming constant ρ and $\mathbf{A}$, is then defined as

$$\mathbf{F}_{ext} = \rho Ag[0 \quad -\frac{1}{2} \quad 0 \quad \frac{L}{12} \quad 0 \quad -\frac{1}{2} \quad 0 \quad -\frac{L}{12}]^T \tag{A.143}$$

Finally, the constraint equation formulation is that two elements share one node, which for simple pendulum case the position constraint is:

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ \\ q_2 \end{bmatrix} \tag{A.144}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{r}_1 \\ \\ \dot{r}_2 \end{bmatrix} = \begin{bmatrix} \dot{q}_1 \\ \\ \dot{q}_2 \end{bmatrix} \tag{A.145}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{r}_1 \\ \\ \ddot{r}_2 \end{bmatrix} = \begin{bmatrix} \ddot{q}_1 \\ \\ \ddot{q}_2 \end{bmatrix} \tag{A.146}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A.147}$$

For the double pendulum case, it then becomes

$$\Phi(u) = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ q^{(2)}_1 - q^{(1)}_5 \\ q^{(2)}_2 - q^{(1)}_6 \end{bmatrix} \tag{A.148}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}^{(1)}_1 \\ \dot{q}^{(1)}_2 \\ \dot{q}^{(2)}_1 - \dot{q}^{(1)}_5 \\ \dot{q}^{(2)}_2 - \dot{q}^{(1)}_6 \end{bmatrix} \tag{A.149}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}^{(1)}_1 \\ \ddot{q}^{(1)}_2 \\ \ddot{q}^{(2)}_1 - \ddot{q}^{(1)}_5 \\ \ddot{q}^{(2)}_2 - \ddot{q}^{(1)}_6 \end{bmatrix} \tag{A.150}$$

The corresponding constraint matrix C is 4×16 sparse zeros matrix with entries follows as:

$$\begin{aligned} &\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,5) = -1 \\ &\mathbf{C}(3,9) = -1, \quad \mathbf{C}(4,6) = 1, \quad \mathbf{C}(4,10) = 1 \end{aligned} \tag{A.151}$$

A.3.3 Floating Reference Formulation

The position vector of the Euler-Bernoulli beam can defined as

$$\begin{aligned} q &= \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 \end{bmatrix}^T \\ &= \begin{bmatrix} R_1 & R_2 & \theta & \hat{q}_1 & \hat{q}_2 & \hat{q}_3 & \hat{q}_4 & \hat{q}_5 \end{bmatrix}^T \\ &= \begin{bmatrix} q_r & q_f \end{bmatrix}^T \end{aligned} \tag{A.152}$$

where $q_r = q(1:3)$ and $q_f = q(4:8)$. The local shape function for Timoshenko beam is

$$\mathbf{S}_l(\xi,\eta) = \frac{1}{1+\varphi} \begin{bmatrix} s_2 & s_3 & t_3 & s_4 & t_4 \\ t_2 & t_3 & s_3 & t_4 & s_4 \end{bmatrix} \tag{A.153}$$

$$\begin{aligned} &s_1 = 1 - 3\xi^2 + 2\xi^3 + (1-\xi)\varphi, \qquad s_2 = L[\xi - 2\xi^2 + \xi^3 + \frac{1}{2}(\xi - \xi^2)\varphi] \\ &s_3 = 3\xi^2 - 2\xi^3 + \xi\varphi, \qquad s_4 = L[\xi^3 - \xi^2 - \frac{1}{2}(\xi - \xi^2)\varphi] \\ &t_1 = 6(\xi - \xi^2)\eta, \qquad t_2 = L[-1 + 4\xi - 3\xi^2 - (1-\xi)\varphi]\eta \\ &t_3 = 6(-\xi + \xi^2)\eta, \qquad t_4 = L[2\xi - 3\xi^2 - \xi\varphi]\eta \end{aligned} \tag{A.154}$$

where $\xi = x/L$, $\eta = y/L$. The shape integral that appears in the floating frame formulation are

$$
\begin{aligned}
\bar{S} &= \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 \\ 0 & 0 & \frac{m}{2} & 0 & -\frac{mL}{12} \end{bmatrix} \\
\mathbf{m}_{ff} &= \frac{m}{(1+\phi)^2} \begin{bmatrix} \mathbb{M}_5 & -\mathbb{M}_4 & 0 & -\mathbb{M}_6 & 0 \\ & \mathbb{M}_1 & 0 & -\mathbb{M}_2 & 0 \\ & & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ & & & \mathbb{M}_5 & 0 \\ sym & & & & \mathbb{M}_5 \end{bmatrix} \\
\tilde{S} &= \frac{1}{(1+\phi)^2} \begin{bmatrix} 0 & 0 & -\mathbb{M}_4 & 0 & -\mathbb{M}_6 \\ 0 & 0 & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ \mathbb{M}_4 & -\mathbb{M}_1 & 0 & \mathbb{M}_2 & 0 \\ 0 & 0 & \mathbb{M}_2 & 0 & \mathbb{M}_5 \\ \mathbb{M}_6 & \mathbb{M}_2 & 0 & -\mathbb{M}_5 & 0 \end{bmatrix}
\end{aligned}
\tag{A.155}
$$

where $\mathbb{M}_1$ to $\mathbb{M}_6$ can be explicitly expressed as:

$$
\begin{aligned}
\mathbb{M}_1 &= \left(\frac{13}{35} + \frac{7\phi}{10} + \frac{\phi^2}{3}\right), \quad \mathbb{M}_2 = L\left(\frac{11}{210} + \frac{11\phi}{120} + \frac{\phi^2}{24}\right) \\
\mathbb{M}_3 &= \left(\frac{9}{70} + \frac{30\phi}{10} + \frac{\phi^2}{6}\right), \quad \mathbb{M}_4 = -L\left(\frac{13}{420} + \frac{3\phi}{40} + \frac{\phi^2}{24}\right) \\
\mathbb{M}_5 &= \left(\frac{1}{105} + \frac{\phi}{60} + \frac{\phi^2}{120}\right), \quad \mathbb{M}_6 = -L\left(\frac{1}{140} + \frac{\phi}{60} + \frac{\phi^2}{120}\right)
\end{aligned}
\tag{A.156}
$$

The rotation matrix $\mathbf{A}$ and $\mathbf{A}_\theta$:

$$
\mathbf{A} = \begin{bmatrix} \cos q_3 & -\sin q_3 \\ \sin q_3 & \cos q_3 \end{bmatrix}, \quad \mathbf{A}_\theta = \begin{bmatrix} -\sin q_3 & -\cos q_3 \\ \cos q_3 & -\sin q_3 \end{bmatrix} \tag{A.157}
$$

In the floating reference formulation, the mass matrix is defined as

$$\mathbf{M} = \begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \bar{S} q_f & \mathbf{A}\bar{S} \\ & q_f^T \mathbf{m}_{ff} q_f & q_f^T \tilde{S} \\ sym & & \mathbf{m}_{ff} \end{bmatrix} =$$

$$\begin{bmatrix} m\mathbf{I} & \mathbf{A}_\theta \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 & \\ 0 & 0 & \frac{m}{2} & 0 & & -\frac{mL}{12} \end{bmatrix} q_f & \mathbf{A} \begin{bmatrix} \frac{mL}{12} & \frac{m}{2} & 0 & \frac{mL}{12} & 0 & \\ 0 & 0 & \frac{m}{2} & 0 & & -\frac{mL}{12} \end{bmatrix} \\ & q_f^T \frac{m}{(1+\phi)^2} \begin{bmatrix} \mathbb{M}_5 & -\mathbb{M}_4 & 0 & -\mathbb{M}_6 & 0 \\ & \mathbb{M}_1 & 0 & -\mathbb{M}_2 & 0 \\ & & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ & & & \mathbb{M}_5 & 0 \\ sym & & & & \mathbb{M}_5 \end{bmatrix} q_f & q_f^T \frac{1}{(1+\phi)^2} \begin{bmatrix} 0 & 0 & -\mathbb{M}_4 & 0 & -\mathbb{M}_6 \\ 0 & 0 & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ \mathbb{M}_4 & -\mathbb{M}_1 & 0 & \mathbb{M}_2 & 0 \\ 0 & 0 & \mathbb{M}_2 & 0 & \mathbb{M}_5 \\ \mathbb{M}_6 & \mathbb{M}_2 & 0 & -\mathbb{M}_5 & 0 \end{bmatrix} \\ & & \begin{bmatrix} \mathbb{M}_5 & -\mathbb{M}_4 & 0 & -\mathbb{M}_6 & 0 \\ & \mathbb{M}_1 & 0 & -\mathbb{M}_2 & 0 \\ & & \mathbb{M}_1 & 0 & -\mathbb{M}_2 \\ & & & \mathbb{M}_5 & 0 \\ sym & & & & \mathbb{M}_5 \end{bmatrix} \end{bmatrix} \tag{A.158}$$

where $\mathbb{M}_1$ to $\mathbb{M}_6$ can be explicitly expressed as:

$$\begin{aligned} \mathbb{M}_1 &= \left(\frac{13}{35} + \frac{7\phi}{10} + \frac{\phi^2}{3}\right), \quad \mathbb{M}_2 = L\left(\frac{11}{210} + \frac{11\phi}{120} + \frac{\phi^2}{24}\right) \\ \mathbb{M}_3 &= \left(\frac{9}{70} + \frac{30\phi}{10} + \frac{\phi^2}{6}\right), \quad \mathbb{M}_4 = -L\left(\frac{13}{420} + \frac{3\phi}{40} + \frac{\phi^2}{24}\right) \\ \mathbb{M}_5 &= \left(\frac{1}{105} + \frac{\phi}{60} + \frac{\phi^2}{120}\right), \quad \mathbb{M}_6 = -L\left(\frac{1}{140} + \frac{\phi}{60} + \frac{\phi^2}{120}\right) \end{aligned} \tag{A.159}$$

And the stiffness matrix is then evaluated as

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbb{K}_3 & -\mathbb{K}_2 & 0 & -\mathbb{K}_4 & 0 \\ 0 & 0 & 0 & -\mathbb{K}_2 & \mathbb{K}_1 & 0 & -\mathbb{K}_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbb{K}_1 & 0 & -\mathbb{K}_2 \\ 0 & 0 & 0 & -\mathbb{K}_4 & -\mathbb{K}_2 & 0 & \mathbb{K}_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\mathbb{K}_2 & 0 & \mathbb{K}_3 \end{bmatrix} \tag{A.160}$$

where $\mathbb{K}_1$ to $\mathbb{K}_4$ can be explicitly expressed as:

$$\begin{aligned} \mathbb{K}_1 &= \frac{12EI}{L^3(1+\phi)}, \quad \mathbb{K}_2 = \frac{6EI}{L^2(1+\phi)} \\ \mathbb{K}_3 &= \frac{(4+\phi)EI}{L(1+\phi)}, \quad \mathbb{K}_2 = \frac{(2-\phi)EI}{L(1+\phi)} \end{aligned} \tag{A.161}$$

The external force can be defined as two parts: external load $\mathbf{Q}_e$, and quadratic velocity vector $\mathbf{Q}_v$

$$\begin{aligned} (\mathbf{Q}_v)_R &= (\dot{\theta})^2 \mathbf{A}(\bar{S} q_f + \mathbf{I}_1) - 2\theta \mathbf{A}_\theta \bar{S} q_f \\ (\mathbf{Q}_v)_\theta &= -2\dot{\theta} \dot{q}_f^T (\mathbf{m}_{ff} q_f + \mathbf{I}_0) \\ (\mathbf{Q}_v)_f &= (\dot{\theta})^2 (\mathbf{m}_{ff} q_f + \mathbf{I}_0) + 2\dot{\theta} \bar{S} \dot{q}_f \end{aligned} \tag{A.162}$$

and yield

$$\mathbf{Q}_v = \begin{bmatrix} \dot{q}_3^2 * \mathbf{A}(\bar{S} * q_f + \begin{bmatrix} \frac{\rho AL^2}{2} \\ 0 \end{bmatrix}) - 2\dot{q}_3^2 \mathbf{A}_\theta \bar{S}\, \dot{q}_f \\ -2\dot{q}_3 \dot{q}_f^T (\mathbf{m}_{ff} q_f + \begin{bmatrix} \frac{\rho AL^3}{24} \\ \frac{\rho AL^2}{4} \\ 0 \\ -\frac{\rho AL^3}{24} \\ 0 \end{bmatrix} \\ \dot{q}_3^2 (\mathbf{m}_{ff} q_f + \begin{bmatrix} \frac{\rho AL^3}{24} \\ \frac{\rho AL^2}{4} \\ 0 \\ -\frac{\rho AL^3}{24} \\ 0 \end{bmatrix}) + 2\dot{q}_3 \tilde{S}\, \dot{q}_f \end{bmatrix} \tag{A.163}$$

The external force for each mode in the generalized coordinate are respectively written as

$$\begin{aligned} (\mathbf{Q}_e)_R &= f_{ext}^T \mathbf{I} \\ (\mathbf{Q}_e)_\theta &= f_{ext}^T \mathbf{A}_\theta \mathbf{r}_c \\ (\mathbf{Q}_e)_f &= f_{ext}^T \mathbf{A} \mathbf{S}_l(\xi = 0.5) \end{aligned} \tag{A.164}$$

for f_{ext} as gravity, it yields

$$\mathbf{Q}_e = \begin{bmatrix} \begin{bmatrix} 0 & -\rho ALg \end{bmatrix}^T \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A}_\theta (\begin{bmatrix} L/2 \\ 0 \end{bmatrix} + \mathbf{S}_l(\xi = 0.5) q_f) \\ \begin{bmatrix} 0 & -\rho ALg \end{bmatrix} \mathbf{A} \mathbf{S}_l(\xi = 0.5) \end{bmatrix} \tag{A.165}$$

The total external load is

$$\mathbf{F}_{ext}(q, \dot{q}) = \mathbf{Q}_v + \mathbf{Q}_e \tag{A.166}$$

The formulation of constraint equation for the simple pendulum case, the position constraint for hinge joint connect to ground can be written as

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{A.167}$$

and the velocity constraint is

$$\dot{\Phi}(\dot{u}) = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{A.168}$$

and the acceleration constraint is:

$$\ddot{\Phi}(\ddot{u}) = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \tag{A.169}$$

The corresponding constraint matrix C follows as:

$$\mathbf{C}(u) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{A.170}$$

For the double pendulum case, the formulation of hinge joint connecting to ground remains the same. For the hinge connecting two body, one has to use the relation:

$$\mathbf{r} = \mathbf{R} + \mathbf{A}\mathbf{S}_l q_f \tag{A.171}$$

and the position constraint becomes

$$\Phi = \begin{bmatrix} r^1_{1x} \\ r^1_{1y} \\ r^2_{1x} - r^1_{2x} \\ r^2_{1y} - r^1_{2y} \end{bmatrix} = \begin{bmatrix} q^{(1)}_1 \\ q^{(1)}_2 \\ \begin{bmatrix} q^{(2)}_1 \\ q^{(2)}_1 \end{bmatrix} - \mathbf{A}\mathbf{S}_l(\xi = 1) q_f \end{bmatrix} \tag{A.172}$$

and the velocity constraint is

$$\dot{\Phi} = \begin{bmatrix} \dot{q}^{(1)}_1 \\ \dot{q}^{(1)}_2 \\ \begin{bmatrix} \dot{q}^{(2)}_1 \\ \dot{q}^{(2)}_2 \end{bmatrix} - \left(\begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l(\xi = 1) q^{(1)}_f & \mathbf{A}\mathbf{S}_l(\xi = 1) \end{bmatrix} \dot{q}^{(1)} \right) \end{bmatrix} \tag{A.173}$$

where $\mathbf{I}$ is identity matrix; and the acceleration constraint is:

$$\ddot{\Phi} = \begin{bmatrix} \ddot{q}^{(1)}_1 \\ \ddot{q}^{(1)}_2 \\ \begin{bmatrix} \ddot{q}^{(2)}_1 \\ \ddot{q}^{(2)}_2 \end{bmatrix} - \left(\begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{A}_\theta \mathbf{S}_l(\xi = 1) \dot{q}^{(1)}_f & \mathbf{A}\mathbf{S}_l(\xi = 1) \dot{q}^{(1)}_3 \end{bmatrix} \dot{q}^{(1)} + \begin{bmatrix} \mathbf{I} & \mathbf{A}_\theta \mathbf{S}_l(\xi = 1) q^{(1)}_f & \mathbf{A}\mathbf{S}_l(\xi = 1) \end{bmatrix} \ddot{q}^{(1)} \right) \end{bmatrix} \tag{A.174}$$

The corresponding constraint matrix C is a 4×16 matrix of zeros with non-zero entries:

$$\mathbf{C}(1,1) = 1, \quad \mathbf{C}(2,1) = 1, \quad \mathbf{C}(3,1) = -1$$

$$\mathbf{C}(3,9) = -1, \quad \mathbf{C}(4,2) = -1, \quad \mathbf{C}(4,10) = -1$$

$$\mathbf{C}(3,3) = (L^{(1)} + q_5^{(1)})\sin(q_3^{(1)}) - (-q_6^{(1)}\cos(q_3^{(1)})), \quad \mathbf{C}(3,5) = \cos(q_3^{(1)}), \quad \mathbf{C}(3,6) = \sin(q_3^{(1)})$$

$$\mathbf{C}(4,3) = (L^{(1)} + q_5^{(1)})\cos(q_3^{(1)}) + (q_6^{(1)}\sin(q_3^{(1)})), \quad \mathbf{C}(4,5) = -\sin(q_3^{(1)}), \quad \mathbf{C}(3,6) = -\cos(q_3^{(1)})$$

(A.175)