NOVEL COMPUTATIONAL AND ANALYTIC TECHNIQUES FOR NONLINEAR SYSTEMS APPLIED TO STRUCTURAL AND CELESTIAL MECHANICS

A Dissertation

by

TAREK ADEL ABDELSALAM ELGOHARY

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	John L. Junkins
Co-Chair of Committee,	James D. Turner
Committee Members,	Rao S. Vadali
	Shankar P. Bhattacharyya
Head of Department,	Rodney Bowersox

May 2015

Major Subject: Aerospace Engineering

ABSTRACT

In this Dissertation, computational and analytic methods are presented to address nonlinear systems with applications in structural and celestial mechanics. Scalar Homotopy Methods (SHM) are first introduced for the solution of general systems of nonlinear algebraic equations. The methods are applied to the solution of post-buckling and limit load problems of solids and structures as exemplified by simple plane elastic frames, considering only geometrical nonlinearities. In many problems, instead of simply adopting a root solving method, it is useful to study the particular problem in more detail in order to establish an especially efficient and robust method. Such a problem arises in satellite geodesy coordinate transformation where a new highly efficient solution, providing global accuracy with a non-iterative sequence of calculations, is developed. Simulation results are presented to compare the solution accuracy and algorithm performance for applications spanning the LEO-to-GEO range of missions. Analytic methods are introduced to address problems in structural mechanics and astrodynamics. Analytic transfer functions are developed to address the frequency domain control problem of flexible rotating aerospace structures. The transfer functions are used to design a Lyapunov stable controller that drives the spacecraft to a target position while suppressing vibrations in the flexible appendages. In astrodynamics, a Taylor series based analytic continuation technique is developed to address the classical two-body problem. A key algorithmic innovation for the trajectory propagation is that the classical averaged approximation strategy is replaced with a rigorous series based solution for exactly computing the acceleration derivatives. Evidence is provided to demonstrate that high precision solutions are easily obtained with the analytic continuation approach. For general nonlinear initial value problems (IVPs), the method of Radial Basis Functions time domain collocation (*RBF-Coll*) is used to address strongly nonlinear dynamical systems and to analyze short as well as long-term responses. The algorithm is compared against, the second order central difference, the classical Runge-Kutta, the adaptive Runge-Kutta-Fehlberg, the Newmark-β, the Hilber-Hughes-Taylor and the modified Chebyshev-Picard iteration methods in terms of accuracy and computational cost for three types of problems; (1) the unforced highly nonlinear Duffing oscillator, (2) the Duffing oscillator with impact loading and (3) a nonlinear three degrees of freedom (3-DOF) dynamical system. The *RBF-Coll*method is further extended for time domain inverse problems addressing fixed time optimal control problems and Lamberts orbital transfer problem. It is shown that this method is very simple, efficient and very accurate in obtaining the solutions. The proposed algorithm is advantageous and has promising applications in solving general nonlinear dynamical systems, optimal control problems and high accuracy orbit propagation in celestial mechanics.

DEDICATION

To my parents, my wife Heba, and my sons Omar & Ali

ACKNOWLEDGEMENTS

I would like to thank God for giving me the strength and the faith to keep going and trust that everything will always work out for the best.

My parents, for their unconditional emotional and financial support throughout my graduate studies. Your love, care, encouragement and unwavering faith made me get this far. All that I am and all that I will ever be, I owe to you.

My wife Heba, you have gone through a lot with me, never complained, never doubted. You were always there for me, you took care of everything and everyone in my continuous absence. For that, words can not describe how grateful I am and how blessed I feel to have you by my side.

Dr. John L. Junkins, working with you is an absolute honor and pleasure. You are the source of so many great ideas and so much invaluable advice. You are always supportive, understanding and encouraging. It is an honor and a privilege to be your student

Dr. James D. Turner, your technical contributions to this dissertation is quite evident. Throughout all our interactions, you were extremely helpful and insightful. I learned a lot from you both technically and in life.

Dr. Srinivas R. Vadali and Dr. Shankar P. Bhattacharyya for serving on my committee, for being amazing teachers and for showing incredible flexibility and understanding when needed the most.

Ms. Karen Knabe and Ms. Lisa Jordan, your administrative support is the best. You have always been so flexible and extremely helpful in everything needed to get to this point. I thank you for your support and dedication, making administrative requirements such a pleasant experience.

I would like to dedicate many special thanks to Professor Satya N. Atluri from the University of California, Irvine whose contributions to this work are invaluable. I learned a lot from your vast technical experience. Without your contributions, this dissertation would not have come to life in its current form. For that I am extremely thankful and forever grateful. I would also like to thank Dr. Leiting Dong from Hohai University, throughout our collaboration at the University of California, Irvine, your technical insights and direct contributions had a huge impact on the successful completion of this dissertation.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

In this Dissertation, several computational and analytic methods are presented to address nonlinear systems with applications in structural and celestial mechanics. This dissertation generally addresses the following main points:

1. Solution of nonlinear algebraic equations with Jacobian inverse-free methods and perturbation methods derived from the physics of the problem.
2. Dynamics and control of aerospace structures with analytic transfer functions.
3. Computational and analytic methods for general nonlinear initial value problems.
4. Computational techniques for optimal control and two-point boundary value problems.

In chapter two, the solution of systems of nonlinear algebraic equations is addressed. Numerical Scalar Homotopy Methods applied to the solution of post-buckling and limit load problems of solids and structures as exemplified by simple plane elastic frames, considering only geometrical nonlinearities are developed. Explicitly derived tangent stiffness matrices and nodal forces of large-deformation planar beam elements, with two translational and one rotational degrees of freedom at each node, are adopted. By using the Scalar Homotopy Methods, the displacements of the equilibrium state are iteratively solved for, without inverting the Jacobian (tangent stiffness) matrix. It is well-known that, the simple Newton's method (and the closely related Newton-Raphson iteration method) that is widely used in nonlinear structural mechanics, which necessitates the inversion of the Jacobian matrix, fails to pass the limit load as the Jacobian matrix becomes singular. Although the so called arc-length method can resolve this problem by limiting both the incremental displacements and forces, it is quite complex for implementation. Moreover, inverting the Jacobian matrix generally consumes the majority of the computational burden especially for large-scale problems. On the contrary, by using the Scalar Homotopy Methods, the convergence can be easily achieved, without inverting the tangent stiffness matrix and without using complex arc-length methods. On the other hand, using physical insight, the problem of Cartesian to Geodetic coordinate transformation is addressed with a singularity-free perturbation solution. Geocentric latitude is used to model the satellite ground track position vector. A natural geometric perturbation variable is identified as the ratio of the major and minor Earth ellipse radii minus one. A rapidly converging perturbation solution is developed by expanding the satellite height above the Earth and the geocentric latitude as a perturbation power series in the geometric perturbation variable. The solution avoids the classical problem encountered of having to deal with highly nonlinear solutions for quartic equations. Simulation results are presented to compare the solution accuracy and algorithm performance for applications spanning the LEO-to-GEO range of missions.

In chapter three, analytic methods are introduced for the dynamics and control of flexible spacecraft structure and the orbit propagation of the two-body problem in celestial mechanics. First, a symmetric flexible rotating spacecraft is modeled as a distributed parameter system of a rigid hub attached to two flexible appendages with tip masses. Hamilton's extended principle is utilized to present a general treatment for deriving the dynamics of multi-body dynamical systems to establish the hybrid system of integro-partial differential equations. A Generalized State Space (GSS) system of equations is constructed in the frequency domain to obtain analytic transfer functions for the rotating spacecraft. The frequency response of the generally modeled spacecraft and a special case with no tip masses are presented. Numerical results for the system frequency response obtained from the analytic transfer functions are presented and compared against the classical assumed modes numerical method. The truncation-error-free analytic results are shown to agree well with the classical numerical solutions without any truncation errors. Fundamentally, the rigorous transfer function, without introduction of spatial discretization, can be directly used in control law design. The frequency response of the system is used in a classical control problem where a Lyapunov stable controller is derived and tested for gain selection. The correlation between the controller design in the frequency domain utilizing the analytic transfer functions and the system response is analyzed and verified. The derived analytic transfer functions are shown to provide a powerful tool to test various control schemes in the frequency domain and a validation platform for existing numerical methods for distributed parameters models. Second, high accuracy orbit propagation for the classical two-body problem is presented with Taylor series based approximation. The success of this strategy is intimately tied to the availability of exact derivative models for the system acceleration. In all cases, a Leibniz product rule provides the enabling analytical machinery for recursively generating series solutions for all higher order time derivatives. All two-body and nonlinear perturbation terms are easily handled. A key algorithmic innovation for the trajectory propagation is that the classical averaged approximation strategy is replaced with a rigorous series based solution for exactly computing the acceleration derivatives. Of course, when many terms are retained in a series approximation it is natural to raise the question: can numerical precision be lost because of many operations involving products and sums of both large and small numbers. The resolution for this question will remain a research topic for future studies; nevertheless, evidence is provided to demonstrate that high precision solutions are easily obtained with the analytic continuation approach.

In the fourth chapter, the Initial Value Problems (IVPs) for strongly nonlinear dynamical systems are studied to analyze short as well as long-term responses. Dynamical systems characterized by a system of second-order nonlinear ordinary differential equations (ODEs) are recast into a system of nonlinear first order ODEs in mixed variables of positions as well as velocities. For each discrete-time interval Radial Basis Functions (RBFs) are assumed as trial functions for the mixed variables in the time domain. A simple collocation method is developed in the time-domain, with Legendre-Gauss-Lobatto nodes as RBF source points as well as collocation points. Numerical examples are provided to compare the present algorithm with explicit as well implicit methods in terms of accuracy, required size of time-interval (or step) and computational cost. The present algorithm is compared against, the second order central difference method, the classical Runge-Kutta method, the adaptive Runge-Kutta-Fehlberg method, the Newmark-β method, the Hilber-Hughes-Taylor method and the Modified Chebyshev-Picard Iterations (MCPI) method. First the highly nonlinear Duffing oscillator is analyzed and the solutions obtained from all algorithms are compared against the analytical solution for free oscillation at long times. A Duffing oscillator with impact forcing function is next solved. Solutions are compared against numerical solutions from state of the art *ODE45* numerical integrator for long times. Finally, a nonlinear 3-DOF system is presented and results from all algorithms are compared against *ODE45*. It is shown that the present *RBF-Coll* algorithm is very simple, efficient and very accurate in obtaining the solution for the nonlinear IVP. Since other presented methods require a smaller step size and usually higher computational cost, the proposed algorithm is advantageous and has promising applications in solving general nonlinear dynamical systems.

In chapter five, ill-posed time-domain inverse problems for dynamical systems with split boundary conditions and unknown controllers are considered. The simple collocation method, with Radial Basis Function (RBF) as trial solutions, and Legendre-Gauss-Lobatto nodes as RBF source points as well as collocation points, is applied. The duffing optimal control problem with various prescribed initial and final conditions, as well as the orbital transfer Lambert's problem are solved by the proposed RBF collocation method as examples. It is shown that this method is very simple, efficient and very accurate in obtaining the solutions, with an arbitrary initial guess. Since other methods such as the shooting method and the pseudo-spectral method can be unstable, the proposed method is advantageous and has promising applications in optimal control and celestial mechanics.

Finally, in the last chapter, concluding remarks are presented regarding each of the developed methods, their applicability to other types of problems and future research thrusts in those areas.

2. METHODS FOR NONLINEAR ALGEBRAIC EQUATIONS*

2.1 Introduction

A large number of problems in engineering and applied sciences, such as large deformation solid mechanics, fluid dynamics, post-buckling of structural frames, plates, and shells, etc, are characterized by nonlinear differential equations that will lead to a system of nonlinear algebraic equations (NAEs) after discretization:
$F_i(x_1, \ldots, x_n) = 0, \; i = 1, \ldots, n$, or in their vector-form:

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0} \tag{2.1}$$

where $\boldsymbol{F}(\boldsymbol{x}) \in \mathbb{R}^n$ is a given vector function. We mention that $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$ may have zero roots, multiple roots, and the simplest ideal case of a single unique solution. Furthermore, it is frequently the case that we have some starting approximate root $\boldsymbol{x}_0$ and we are motivated to consider iterative refinements of this approximate root. To find the unknown vector $\boldsymbol{x} \in \mathbb{R}^n$, the famous Newton's method is widely used to iteratively solve Eq. (2.1):

$$\begin{aligned} &\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \boldsymbol{B}_k^{-1}\boldsymbol{F}_k \\ &\boldsymbol{x}_0 = \boldsymbol{a} \end{aligned} \tag{2.2}$$

where, $\boldsymbol{a}$ represents the initial guess of the solution, $\boldsymbol{B}$ is the Jacobian (tangent stiffness) matrix given by $\boldsymbol{B} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}}$, and k denotes the number iteration index.

*Part of this chapter is reprinted with permission from "Solution of Post-Buckling & Limit Load Problems, Without Inverting the Tangent Stiffness Matrix & Without Using Arc-Length Methods" by Elgohary, T. A., Dong, L., Junkins, J. L. and Atluri, S. N., 2014. *CMES: Computer Modeling in Engineering & Sciences*, Vol. 98, No. 6, pp. 543–563, Copyright [2014] by Tech Science Press. And from "A Simple Perturbation Algorithm for Inverting the Cartesian to Geodetic Transformation" by Turner, J. D. and Elgohary, T. A., 2013. *Mathematical Problems in Engineering*, Vol. 2013 Copyright [2013] by Hindawi Publishing.

As motivated by Eq. (2.2), the continuous Newton's method is introduced in [1] as:

$$\begin{aligned}\dot{\boldsymbol{x}} &= -\boldsymbol{B}^{-1}(\boldsymbol{x})\boldsymbol{F}(\boldsymbol{x}), \quad t > 0 \\ \boldsymbol{x}(0) &= \boldsymbol{a}\end{aligned} \tag{2.3}$$

From a parameter optimization perspective, see [2–4], Eq. (2.1) can be solved by defining a quadratic cost function to be minimized as:

$$J = \frac{1}{2}\boldsymbol{F}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{F}(\boldsymbol{x}) \tag{2.4}$$

Clearly, for the case of a single root of $\boldsymbol{F}(\boldsymbol{x}) = \mathbf{0}$, J has its global minimum of zeros at the root. Also note, for the case that there are no exact roots, minimizing J will seek a point that locally most nearly satisfies $\boldsymbol{F}(\boldsymbol{x}) = \mathbf{0}$. By locally linearizing $\boldsymbol{F}(\boldsymbol{x})$ about the estimate $\boldsymbol{x}_k$, we establish the *nonlinear least-squares* or *Gaussian least-squares differential correction*, and the iterative process is

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \Delta\boldsymbol{x} \\ \Delta\boldsymbol{x} &= -(\boldsymbol{B}_k^{\mathrm{T}}\boldsymbol{B}_k)^{-1}\boldsymbol{B}_k^{\mathrm{T}}\boldsymbol{F}_k\end{aligned} \tag{2.5}$$

This algorithm is ideally suited to the over-determined case that the vector function $\boldsymbol{F}(\boldsymbol{x})$ has dimension $m > n$, where n is the dimension of $\boldsymbol{x}$. Note for the $n \times n$ and B_k of full rank, that the least squares algorithm reduces identically to the Newton iteration of Eq. (2.2)

As with the case of the classical and continuous Newton's methods, nonlinear least squares convergence can become problematic if the initial guess is too far from the solution or the Jacobian matrix is either ill-conditioned or rank deficient. The *method of gradients* or *method of steepest descent* tries to overcome these problems by finding the solution along the direction of steepest descent along the negative gradient of J. It can be applied to solve Eq. (2.1) as:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha^k \nabla \boldsymbol{F}_k \tag{2.6}$$

where the scalar α^k is varied to improve convergence and not to overshoot the function minimum, [2]. In general, gradient methods show very good convergence towards the solution in the first few iterations. However, convergence is very poor near the solution and the number of iterations to reach the final solution is generally unbounded.

Homotopy methods, as firstly introduced by Davidenko in 1953, [5,6], represents one of the best methods to enhance the convergence from a local convergence to a global convergence. Previously, all the homotopy methods were based on the construction of a vector homotopy function $\boldsymbol{H}(\boldsymbol{x}, t)$, which serves the objective of continuously transforming a function $\boldsymbol{G}(\boldsymbol{x})$ into $\boldsymbol{F}(\boldsymbol{x})$ by introducing a homotopy parameter t $(0 \leq t \leq 1)$. The homotopy parameter t can be treated as a time-like fictitious variable, and the homotopy function can be any continuous function such that $\boldsymbol{H}(\boldsymbol{x}, 0) = 0 \Leftrightarrow \boldsymbol{G}(\boldsymbol{x}) = 0$ and $\boldsymbol{H}(\boldsymbol{x}, 1) = 0 \Leftrightarrow \boldsymbol{F}(\boldsymbol{x}) = 0$.

However, in order to find the solution, both the iterative and the continuous Newton's methods require the the inversion of the Jacobian matrix. On one hand, inverting the Jacobian matrix in each iteration is computationally very expensive. On the other hand, for complex problems where the Jacobian matrix may be singular at limit points, as shown in Fig. 2.1, finding the solution with Newton's methods can not be achieved. Various variants of the arc-length methods have been widely used for marching through the limit-points such as those presented in [7–10] for post-buckling analyses. These methods generally involve complex procedures by appending various constraints, and monitoring the eigenvalues of the Jacobian matrix.

It will be advantageous to have a method to find the solutions for systems of NAEs without inverting the Jacobian matrix, without using the arc-length method, and without worrying about initial guesses for the Newton's methods.

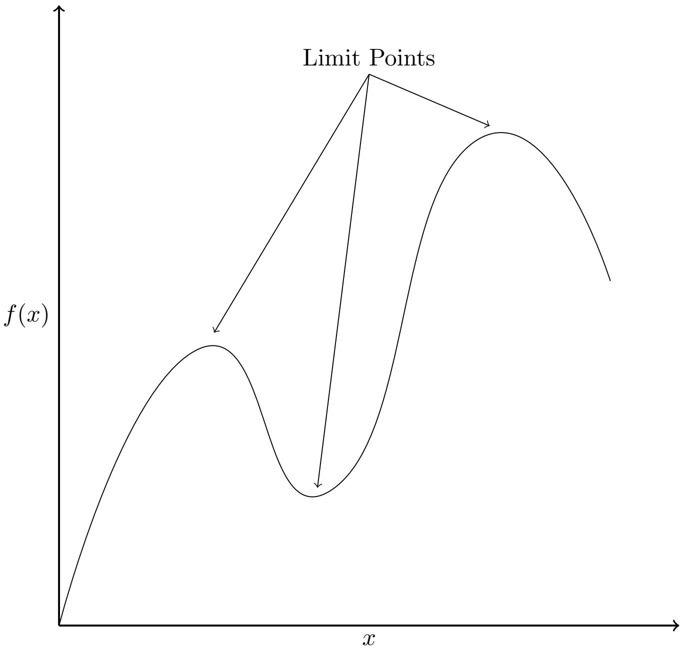

Figure 2.1: Newton's Method & Limit Points

2.2 Solution of NAEs without Inverting the Jacobian

As introduced in the previous sections, the *method steepest descent* can be considered as an early attempt to avoid inverting the Jacobian matrix for solving NAEs. It is known to converge rapidly in the first few iteration but has very poor convergence in the proximity of the solution. The *Davidenko* method still require a matrix inversion operation at each iteration step. Quasi-Newton's methods have been developed in order to eliminate the need to compute the Jacobian and consequently its inverse at each iteration of the solution. In general, at each iteration the Hessian matrix and/or its inverse are computed from the previous step following an update equation to solve the secant equation which represents the Taylor series of the gradient/Jacobian. Several variations of the method exist and are implemented in most optimization toolboxes, [11]. The most famous of which is introduced by Broyden in several works, [12–15]. A discussion of the derivation of the method and its application in optimization problem is presented in [16].

In order to eliminate the need for inverting the Jacobian matrix in the Newton's iteration procedure, an alternate first-order system of nonlinear ordinary differential equations (ODEs) is proposed in [17]. The solution of the set of NAEs, $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$, in Eq. (2.1), can be obtained by postulating an evolutionary equation for $\boldsymbol{x}$, thus:

$$\dot{\boldsymbol{x}} = \frac{-\nu}{q(t)}\boldsymbol{F}(\boldsymbol{x}), \quad t > 0 \tag{2.7}$$

where ν is a nonzero constant and $q(t)$ may in general be a monotonically increasing function of t. In that approach, the term $\nu/q(t)$ plays the major role of being a stabilizing controller to help obtain the solution even for a bad initial guess, and speeds up the convergence. This *Fictitious Time Integration Method* was successfully applied to the solution of various engineering problems in [17–19]. In spite of its success, the *Fictitious Time Integration Method* was postulated only based on an engineering intuition, does not involve the Jacobian matrix at all, and was shown to only have local convergence.

Two of the most popular vector homotopy functions are the Fixed-point Homotopy Function:

$$\boldsymbol{H}_F(\boldsymbol{x}, t) = t\boldsymbol{F}(\boldsymbol{x}) + (1-t)(\boldsymbol{x} - \boldsymbol{x}_0) = 0, \quad 0 \leq t \leq 1 \tag{2.8}$$

and the Newton Homotopy Function:

$$\boldsymbol{H}_N(\boldsymbol{x}, t) = t\boldsymbol{F}(\boldsymbol{x}) + (1-t)\left[\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}(\boldsymbol{x}_0)\right] = 0, \quad 0 \leq t \leq 1 \tag{2.9}$$

By using the vector homotopy method, the solution of the NAEs can be obtained by numerically integrating:

$$\dot{\boldsymbol{x}} = -\left(\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}}\right)^{-1} \frac{\partial \boldsymbol{H}}{\partial t}, \quad 0 \leq t \leq 1 \tag{2.10}$$

As can be seen in Eq. (2.10), the implementation of the Vector Homotopy Method necessitates the inversion of the matrix $\frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}}$ at each iteration. This is a known problem in homotopy method, [20, 21], where the homotopy path to convergence would generally have turning points, see Fig. 2.2. This necessitates the use of arc-lengths along the curve, $ds^2 = dx^2 + dt^2$, as the independent variable and solve the homotopy differential equation in Eq. (2.10) for the augmented space, $\boldsymbol{z} = [\boldsymbol{x} \quad t]^{\mathrm{T}}$ as in the Chow-Yorke homotopy method developed in [21].

In order to remedy the shortcoming of the Vector Homotopy Method, the system of NAEs is solved by constructing a Scalar Homotopy Function $\boldsymbol{h}(\boldsymbol{x}, t)$, such that $\boldsymbol{h}(\boldsymbol{x}, 0) = 0 \Leftrightarrow \|\boldsymbol{G}(\boldsymbol{x})\| = 0$ and $\boldsymbol{h}(\boldsymbol{x}, 1) = 0 \Leftrightarrow \|\boldsymbol{F}(\boldsymbol{x})\| = 0$, , [22]. As an example,

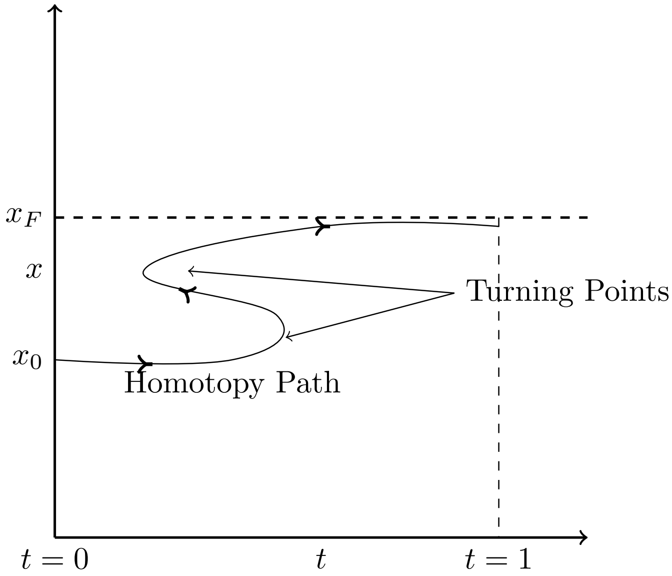

Figure 2.2: Homotopy Path Turning Points

the following Scalar Fixed-point Homotopy Function is introduced in [22]:

$$h(\boldsymbol{x}, t) = \frac{1}{2}[t\|\boldsymbol{F}(\mathbf{x})\|^2 - (1-t)\|\boldsymbol{x} - \boldsymbol{x}_0\|^2], \quad 0 \leq t \leq 1 \tag{2.11}$$

However, it is more convenient to to define homotopy functions with $t \in [0, \infty]$ instead of $t \in [0, 1]$, and use a positive and monotonically increasing function $Q(t)$ to enhance the convergence speed. Two new scalar homotopy functions are introduced. For both of these two methods, the inversion of the Jacobian matrix is not involved. The two methods are based on the homotopy functions in Eq. (2.8) and Eq. (2.9) with the Fixed-point Homotopy Function better suited for scalar problems than the Newton Homotopy Function. The two homotopy functions are introduced as denoted by the *Scalar Fixed-point Homotopy Function*:

$$h_f(\boldsymbol{x}, t) = \frac{1}{2}\|\boldsymbol{F}(\boldsymbol{x})\|^2 + \frac{1}{2Q(t)}\|\boldsymbol{x} - \boldsymbol{x}_0\|^2, \quad t \geq 0 \tag{2.12}$$

and the *Scalar Newton Homotopy function*:

$$h_n(\boldsymbol{x},t) = \frac{1}{2}\|\boldsymbol{F}(\boldsymbol{x})\|^2 + \frac{1}{2Q(t)}\|\boldsymbol{F}(\boldsymbol{x}_0)\|^2, \quad t \geq 0 \tag{2.13}$$

By selecting a driving vector $\boldsymbol{u}$ so that the evolution of $\dot{\boldsymbol{x}}$ is parallel to $\boldsymbol{u}$, the system of NAEs can be solved by numerically integrating:

$$\dot{\boldsymbol{x}} = -\frac{\frac{\partial h}{\partial t}}{\left(\frac{\partial h}{\partial \boldsymbol{x}}\right)\cdot \boldsymbol{u}}\boldsymbol{u}, \quad t \geq 0 \tag{2.14}$$

With different scalar homotopy functions $h(\boldsymbol{x},t)$, different $Q(t)$, and different driving vectors, $\boldsymbol{u}$, Eq. (2.14) leads to different variants of scalar homotopy methods, see [22–28] for a variety of selection of the driving vector $\boldsymbol{u}$ with various applications in structural mechanics problems. We select $\boldsymbol{u}$ such that $\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}}$. Thus, if h_f is to be used, Eq. (2.14) leads to:

$$\dot{\boldsymbol{x}} = -\frac{1}{2}\frac{\dot{Q}\|\boldsymbol{F}\|^2}{\|Q\boldsymbol{B}^T\boldsymbol{F} + \boldsymbol{x} - \boldsymbol{x}_0\|^2}\left(Q\boldsymbol{B}^T\boldsymbol{F} + \boldsymbol{x} - \boldsymbol{x}_0\right), \quad t \geq 0 \tag{2.15}$$

and if h_n is to be used, we have:

$$\dot{\boldsymbol{x}} = -\frac{1}{2}\frac{\dot{Q}\|\boldsymbol{F}\|^2}{Q\|\boldsymbol{B}^T\boldsymbol{F}\|^2}\boldsymbol{B}^T\boldsymbol{F}, \quad t \geq 0 \tag{2.16}$$

$Q(t) = e^t$ is used here for simplicity, while various possible choices can be found in [28].

In the next sections, two solution methodologies are introduced for structural mechanics and geodesy problems, [29, 30]. In the first study, the *Scalar Homotopy Methods* are applied to the solution of post-buckling and limit load problems of solids and structures as exemplified by simple plane elastic frames, considering only geometrical nonlinearities. Explicitly derived tangent stiffness matrices and nodal forces of large-deformation planar beam elements, with two translational and one rotational degrees of freedom at each node, are adopted following the work of Kondoh and Atluri in [31]. By using the *Scalar Homotopy Methods*, the displacements of the equilibrium state are iteratively solved for, without inverting the Jacobian (tangent stiffness) matrix. The simple Newton's method (and the Newton-Raphson iteration method that is widely used in nonlinear structural mechanics), which necessitates the inversion of the Jacobian matrix, fails to pass the limit load as the Jacobian matrix becomes singular. Although arc-length method can resolve this problem by limiting both the incremental displacements and forces, it is quite complex for implementation. Moreover, inverting the Jacobian matrix generally consumes the majority of the computational burden especially for large-scale problems. On the contrary, by using the *Scalar Homotopy Methods*, the convergence can be easily achieved, without inverting the tangent stiffness matrix and without using complex arc-length methods. The study thus opens a promising path for conducting post-buckling and limit-load analyses of nonlinear structures as applied to the classical Williams' toggle problem, [32].

The second study introduces a singularity-free perturbation solution for inverting the Cartesian to Geodetic transformation. Geocentric latitude is used to model the satellite ground track position vector. A natural geometric perturbation variable is identified as the ratio of the major and minor Earth ellipse radii minus one. A rapidly converging perturbation solution is developed by expanding the satellite height above the Earth and the geocentric latitude as a perturbation power series in the geometric perturbation variable. The solution avoids the classical problem encountered of having to deal with highly nonlinear solutions for quartic equations. Simulation results are presented that compare the solution accuracy and algorithm performance for applications spanning the Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO) range of missions.

2.3 Solution of Post-Buckling & Limit Load Problems, Without Inverting the Jacobian & Without Using Arc-Length Methods

In computational solid mechanics, the trend over the past 30 – 40 years has been to directly derive the tangent stiffness matrix, $\boldsymbol{B}$, (rather than forming the nonlinear equations, $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$) through incremental finite element methods, [7–9, 31]. Recently, however, the system of equations, $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$, is directly derived for a von Kármán plate theory using Galerkin method, [28].

As discussed earlier, as the Jacobian matrix becomes singular, such as the limit-load points for geometrically nonlinear frames or in elastic-plastic solids, the iterative and the continuous Newton's methods become problematic, as shown in Fig. 2.3. Various variants of the arc-length methods have been widely used for marching through the limit-points, in post-buckling analyses, such as those presented in [7–10]. These methods generally involve complex procedures by appending various constraints, and monitoring the eigenvalues of the Jacobian matrices. The present method finds the solutions for post-buckling problems of structures without inverting the Jacobian matrix, without using the arc-length method, and without worrying about initial guesses for the Newton's methods.

The classical toggle problem, as introduced in [32], comprises of two rigidly jointed equal members of length l and angle β with respect to the horizontal axis and subjected to an externally applied vertical load W at the apex, as shown in Fig. 2.4.

The structure deforms in a symmetrical mode as shown in Fig. 2.5 with the deflected position of the neutral axis of member rs denoted by $r's'$, Following the same assumptions and nomenclature in [32], the externally applied load W can be expressed in terms of the deformation at the apex, δ, through the following series of

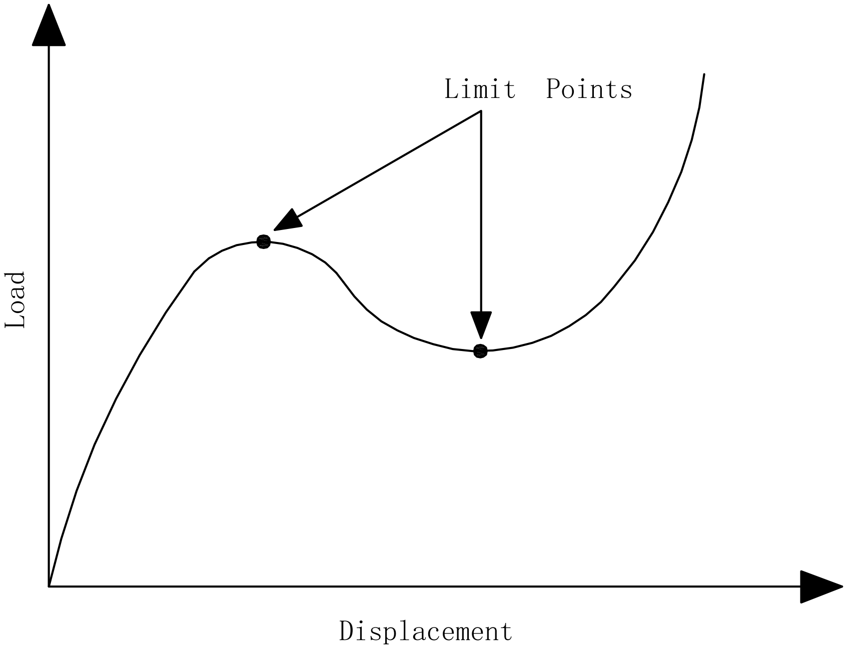

Figure 2.3: Newton's Method Fails at Limit Points, [29]

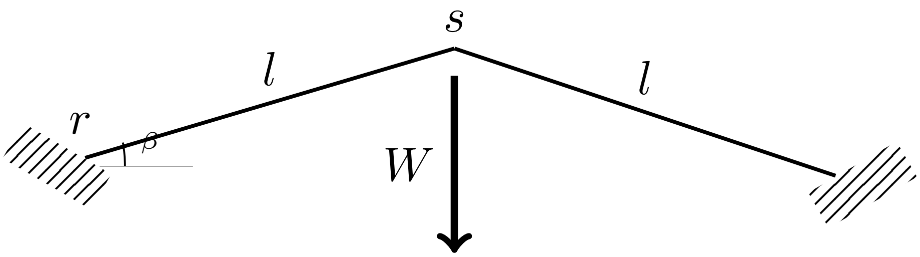

Figure 2.4: The Classical Williams' Toggle, [29]

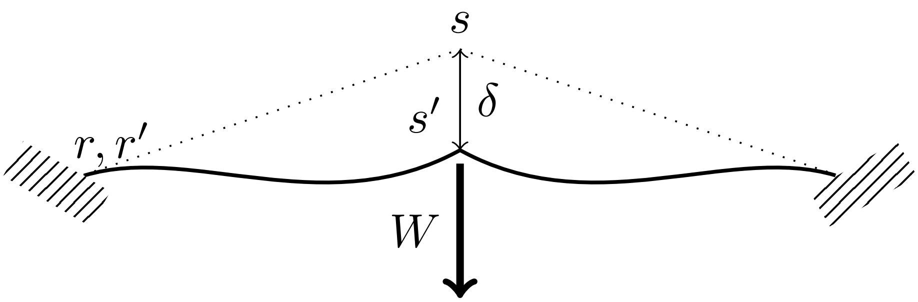

Figure 2.5: Symmetrical Deformation of Toggle, [29]

equations,

$$\frac{W}{2} \approx F + P \sin\beta \tag{2.17}$$

where, F is the component of the reaction force perpendicular to the undeflected position of the neutral axis denoted here by rs and P is the component of the force at the end of the member parallel to rs. P is expressed in terms of δ as:

$$P = \frac{AE}{l}\left(\delta \sin\beta - 0.6\frac{\delta^2}{l}\right) \tag{2.18}$$

where, AE is the extensional rigidity of the member. F is then expressed in terms of δ using nonlinear elastic stability theory as:

$$F = \frac{6EI}{l^2} d_5 \frac{\delta}{l} \tag{2.19}$$

where d_5 can be obtained by the following relations:

$$\begin{aligned}
d_5 &= 2d_4 w(\rho) \\
d_4 &= \frac{d_3}{3} \\
d_3 &= d_1 + d_2 \\
d_2 &= d_1 - w(\rho) \\
d_1 &= \frac{1}{2}\left[\frac{\pi^2\rho}{4\left(1 - w(\rho)\right)} + w(\rho)\right] \\
w(\rho) &= \frac{\pi}{2}\sqrt{\rho}\cot\frac{\pi}{2}\sqrt{\rho} \\
\rho &= \frac{\pi^2 EI}{l^2}
\end{aligned} \tag{2.20}$$

Combining Eq. (2.17) through Eq. (2.20), for a given load W, the vertical displacement, δ, of the Williams' toggle can be found by solving the following scalar nonlinear algebraic equation:

$$\frac{3AE\sin\beta}{5l^2}\delta^2+\left(\frac{6EId_5}{l^3}+\frac{AE\sin\beta^2}{l}\right)\delta-\frac{W}{2}=0 \tag{2.21}$$

In this study, the set of parameters l, EI, AE are considered to be the same as those presented in [32], and are given in Table 2.1. By changing the height of the apex of the toggle, three cases of interest are generated, as shown in Fig. 2.6. The first case takes $l\sin\beta = 0.32$, which represents the original plot in [32]. The second and third cases, $l\sin\beta = 0.38$ and $l\sin\beta = 0.44$, respectively, show the effect of raising the apex on the load-deflection curve of the toggle, and introduce limit points in Eq. (2.21) at which the Jacobian is singular.

Table 2.1: Parameters set in [32], [29]

Parameter	**Value**	**Units**
l	12.94	in
EI	9.27×10^3	lb/in^2
EA	1.885×10^6	lb

2.3.1 Solving Williams' Equation with a Scalar Homotopy Method

In order to characterize the deflection δ resulting from a specific external load, the scalar NAE of Eq. (2.21) must be solved. To better understand the limitations on solving Eq. (2.21) utilizing the classical Newton's method, the behavior of the Jacobian derived analytically from Eq. (2.21) is shown in Fig. 2.7 for the three values of $l\sin\beta$ introduced earlier.

Limit-points are those where the Jacobian becomes close to zero and thus classical Newton's method will fail. For this end, the previously introduced Scalar Homotopy

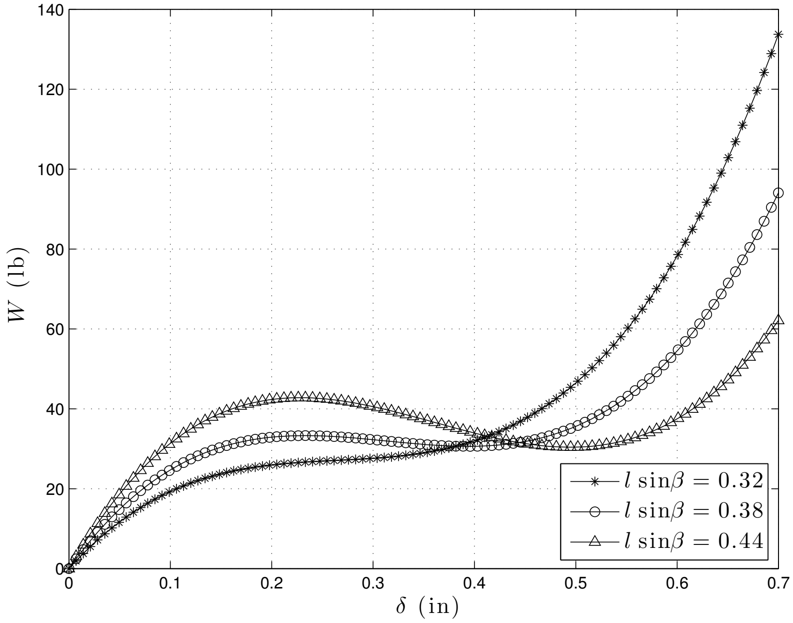

Figure 2.6: Three Cases of Load-Deflection Curves for Williams' Toggle, [29]

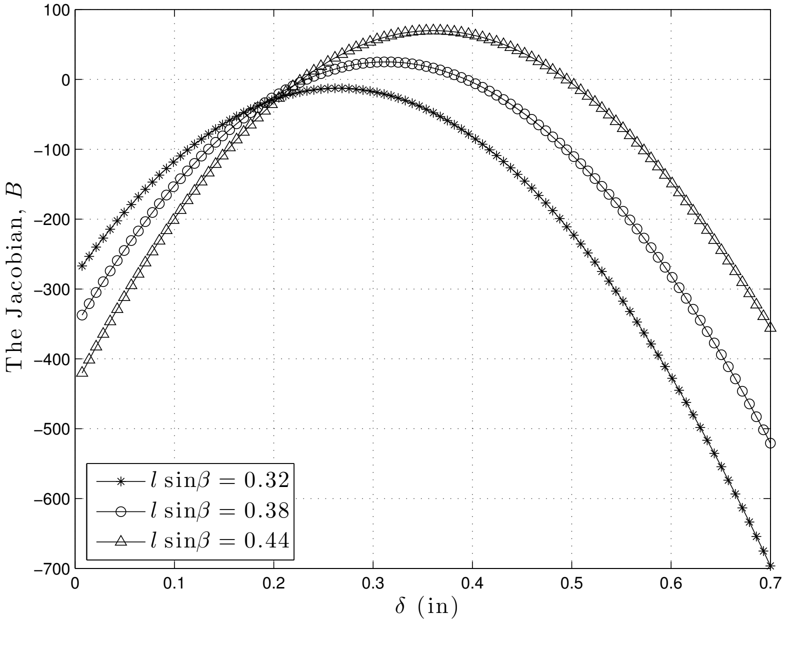

Figure 2.7: Scalar Jacobian Evaluation of Williams' Scalar NAE, [29]

Methods are used to avoid inverting the Jacobian for the solution of the NAE. The Scalar Newton Homotopy Method in Eq. (2.16) only works for a system of NAEs. Thus, the Scalar Fixed-point Homotopy Method in Eq. (2.15) is adopted here for the solution of the scalar NAE of the Williams' toggle.

Setting the tolerance to 10^{-6} for the original Williams' toggle with $l \sin\beta = 0.32$, the Williams' equation can be solved for an arbitrary load, here chosen as $W = 27.11$ lb. A comparison between Newton's Method and the Scalar Fixed-point Homotopy Method is shown in Table 2.2

Table 2.2: Solution of original Williams' equation with no limit points, [29]

Method	**No. Iteration, N**	**Achieved Accuracy**
Newton's Method	$N = 9$	6.4×10^{-7}
Scalar Fixed-point Homotopy	$N = 30$	8.9×10^{-7}

The comparison shows the fast convergence speed of Newton's method achieving the required accuracy in just 9 iterations whereas it took the Scalar Fixed-point Homotopy Method 30 iterations to achieve similar accuracy results. This case of $l \sin\beta = 0.32$ as shown in Fig. 2.7 has no singularities in the Jacobian, thus the superior performance of Newton's method is expected. Figure 2.8 and Fig. 2.9 show the evolution of the solution and the fast convergence of Newton's Method as compared to the Scalar Fixed-point Homotopy Method.

A second case is considered for $l \sin\beta = 0.44$ where the Williams' equation is solved for an externally applied load selected near the limit point, $W = 43.79$ lb. Both Newton's Method and the Scalar Fixed-point Homotopy Method are utilized, and the results are shown in Table 2.3

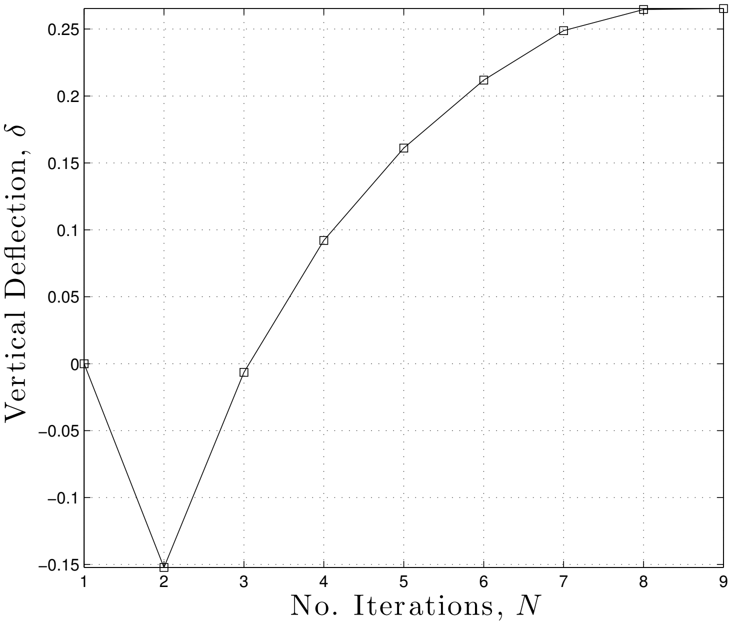

Figure 2.8: Vertical Deflection vs. No. Iterations, $l \sin \beta = 0.32$, Newton's Method, [29]

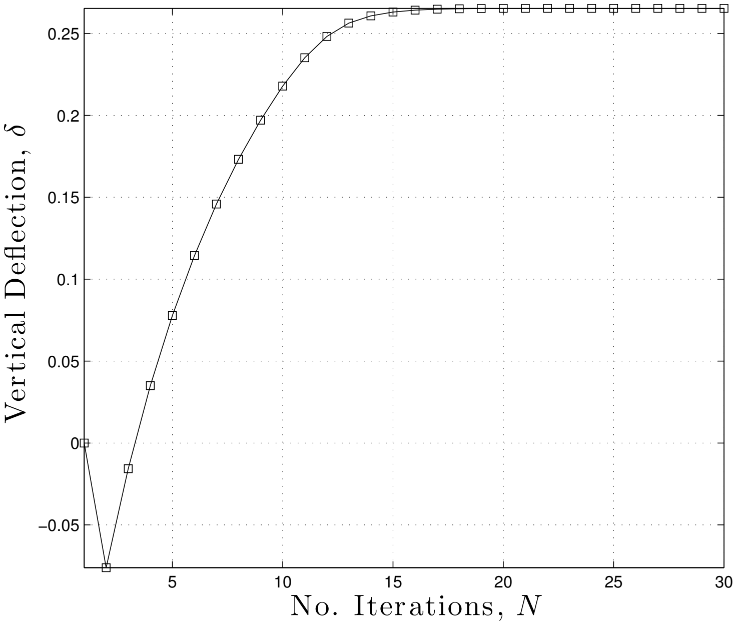

Figure 2.9: Vertical Deflection vs. No. Iterations, $l \sin \beta = 0.32$, Scalar Fixed-point Homotopy Method, [29]

Table 2.3: Solution of Williams' Equation for Loading near Limit Point, [29]

Method	**No. Iteration, N**	**Achieved Accuracy**
Newton's Method	$N = 1000$	0.044
Scalar Fixed-point Homotopy	$N = 345$	7.6×10^{-7}

After 1000 iterations Newton's method did not converge to the solution whereas the Scalar Fixed-point Homotopy Method achieved the required accuracy in 345 iterations. Figure 2.10 and Fig. 2.11 show a comparison between the evolution of the solution for the two methods. It is shown that Newton's method will keep fluctuating about the solution and not converge to achieve the required accuracy, whereas the Scalar Fixed-point Homotopy Method converges to the solution with the required high accuracy.

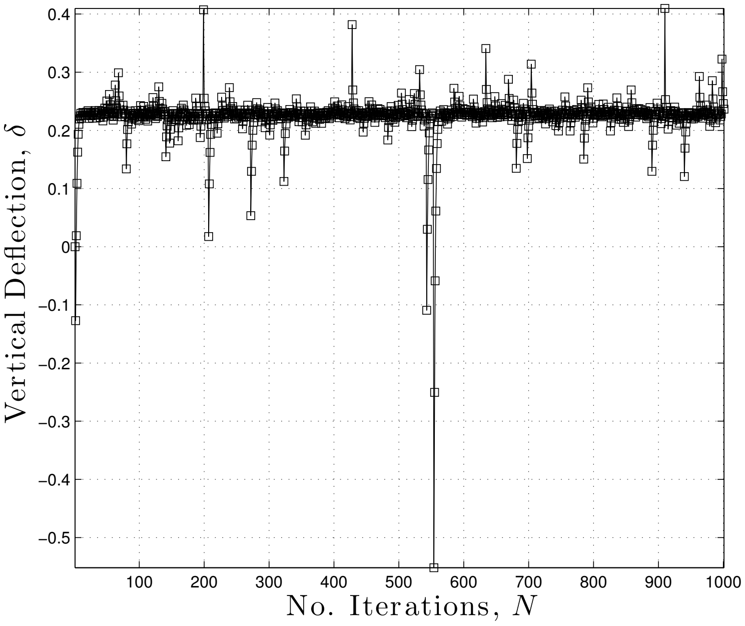

Figure 2.10: Vertical Deflection vs. No. Iterations, $l \sin \beta = 0.44$, Newton's Method, [29]

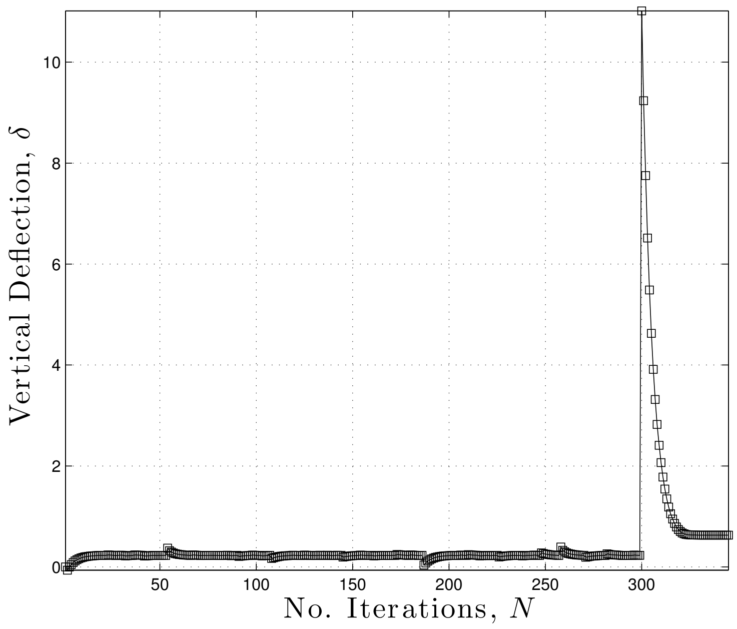

Figure 2.11: Vertical Deflection vs. No. Iterations, $l \sin \beta = 0.44$, Scalar Fixed-point Homotopy Method, [29]

2.3.2 A Generalized Finite Element Model for Frame Structures

The currently adapted Scalar Homotopy Methods can be easily combined with general purpose nonlinear finite element programs, by taking the directly derived tangent stiffness matrix at each iteration as the Jacobian matrix, and taking the difference between generalized internal force vector and the external force vector (the residual) as $\boldsymbol{F}(\boldsymbol{x})$. In this work, explicitly derived tangent stiffness matrices and nodal forces of large-deformation beam-column members are adopted following [31]. The basic derivations in [31] are briefly reviewed here.

First, the nomenclature and the sign convention used in the derivation for a general beam column member are shown in Fig. 2.12. The functions $w(z)$ and $u(z)$ describe the displacement at the centroidal axis of the element along the z and the

x axes, respectively. The angles θ_1^* and θ_2^* are the angles between the tangent to the deformed centroidal axis and the line joining the two nodes of the deformed element at nodes 1 and 2, respectively. M_1 and M_2 are the bending moments at the two nodes and N is the axial force in the beam member. The total rotation of the beam member is then given by,

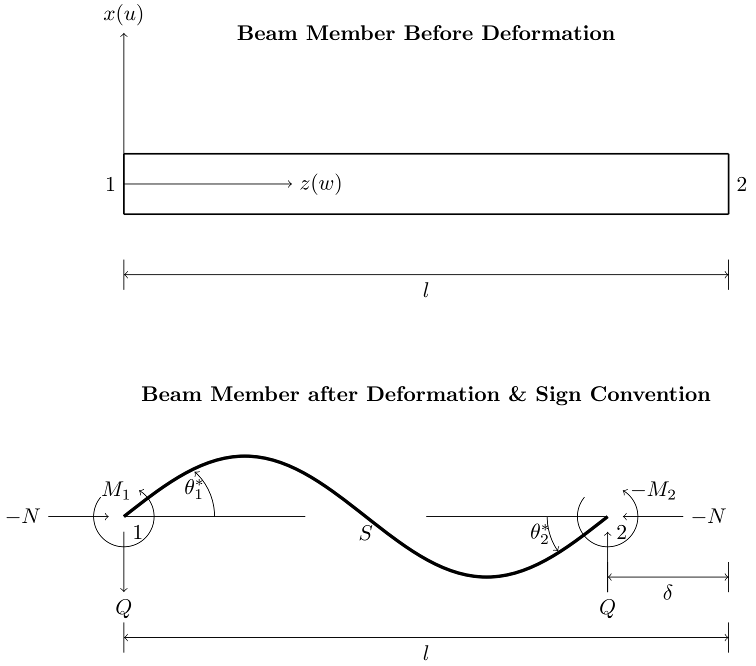

Figure 2.12: Kinematics & Nomenclature for a Beam Member, [29]

$$\theta = \tilde{\theta} + \theta^* \tag{2.22}$$

where, $\tilde{\theta}$ describes the rigid rotation of the beam member and is measured between the line joining the two nodes of the deformed beam and the z-axis. $\tilde{\theta}$ is expressed in terms of the nodal displacements as,

$$\tilde{\theta} = \tan^{-1}\left(\frac{\tilde{u}}{l + \tilde{w}}\right) \tag{2.23}$$

where, $\tilde{u} = u_2 - u_1$ and $\tilde{w} = w_2 - w_1$. From Eq. (2.22) and Eq. (2.23), the non-rigid rotations at the two nodes, θ_1^* and θ_2^*, are given by,

$$\begin{aligned} \theta_1^* &= \theta_1 - \tan^{-1}\left(\frac{\tilde{u}}{l+\tilde{w}}\right) \\ \theta_2^* &= \theta_2 - \tan^{-1}\left(\frac{\tilde{u}}{l+\tilde{w}}\right) \end{aligned} \tag{2.24}$$

The total stretch/deformation of the beam member is then expressed in terms of the displacements at the two nodes as,

$$\delta = \left[(l+\tilde{w})^2 + \tilde{u}^2\right]^{1/2} - l \tag{2.25}$$

The axial force and bending moment are non-dimensionalized through,

$$n = \frac{Nl^2}{EI}, \qquad m = \frac{MI}{EI} \tag{2.26}$$

The non-rigid rotation and the non-dimensional bending moment are decomposed into symmetric and anti-symmetric parts given by,

$$\begin{aligned} \theta_a^* &= \tfrac{1}{2}\left(\theta_1^* + \theta_2^*\right), & \theta_s^* &= \tfrac{1}{2}\left(\theta_1^* - \theta_2^*\right) \\ m_a &= \left(m_1 - m_2\right), & m_s &= \left(m_1 + m_2\right) \end{aligned} \tag{2.27}$$

The relation between the generalized displacements and forces at the nodes of the beam member is given by,

$$\begin{aligned} &\theta_a^* = h_a m_a, \qquad \theta_s^* = h_s m_s \\ &\frac{\delta}{l} = \frac{1}{2}\left(\frac{dh_a}{dn}\right)\frac{\theta_a^{*2}}{h_a^2} + \frac{1}{2}\left(\frac{dh_s}{dn}\right)\frac{\theta_s^{*2}}{h_s^2} + \frac{N}{EA} \end{aligned} \tag{2.28}$$

where, h_a and h_s are given by,

$$\begin{aligned}
&\text{for} \quad n \le 0 \\
&h_a = \frac{1}{-n} - \frac{1}{2(-n)^{1/2}} \cot\left(\frac{(-n)^{1/2}}{2}\right), \quad h_s = \frac{1}{2(-n)^{1/2}} \tan\left(\frac{(-n)^{1/2}}{2}\right) \\
&\text{for} \quad n > 0 \\
&h_a = -\frac{1}{n} - \frac{1}{2(n)^{1/2}} \coth\left(\frac{(n)^{1/2}}{2}\right), \qquad h_s = \frac{1}{2(n)^{1/2}} \tanh\left(\frac{(n)^{1/2}}{2}\right)
\end{aligned} \tag{2.29}$$

The kinematics variables can then be expressed in a vector form for a beam member m as,

$$\{d^m\} = \begin{bmatrix} w_1 & w_2 & u_1 & u_2 & \theta_1 & \theta_2 \end{bmatrix}^{\mathrm{T}} \tag{2.30}$$

The increment of the internal energy of a beam member is then expressed in terms of the increment of kinematics variables vector, $\{d^m\}$, the tangent stiffness matrix $[K^m]$ and the internal force vector $\{R^m\}$ as,

$$\Delta\pi = \frac{1}{2} \{\Delta d^m\}^{\mathrm{T}} [K^m] \{\Delta d^m\} + \{\Delta d^m\}^{\mathrm{T}} \{R^m\} \tag{2.31}$$

The tangent stiffness matrix, $[K^m]$, and the internal force vector, $\{R^m\}$, for the member m are given by,

$$[K^m] = [A_{dd}] - \frac{1}{A_{nn}} \{A_{nd}\} \{A_{nd}\}^{\mathrm{T}} \tag{2.32}$$

$$\{R^m\} = \{B_d\} - \frac{1}{A_{nn}} \{A_{nd}\} \tag{2.33}$$

where the elements constructing Eq. (2.32) and Eq. (2.33) are given by,

$$[A_{dd}] = \begin{bmatrix} \begin{matrix} \Big(N\frac{\partial^2\delta}{\partial\tilde{w}^2} + M_a\frac{\partial^2\theta_a^*}{\partial\tilde{w}^2} \\ +\frac{EI}{l}\frac{1}{h_a}\left(\frac{\partial\theta_a^*}{\partial\tilde{w}}\right)^2\Big) \end{matrix} [E] & \begin{matrix} \Big(N\frac{\partial^2\delta}{\partial\tilde{w}\partial\tilde{u}} + M_a\frac{\partial^2\theta_a^*}{\partial\tilde{w}\partial\tilde{u}} \\ +\frac{EI}{l}\frac{1}{h_a}\left(\frac{\partial\theta_a^*}{\partial\tilde{w}}\frac{\partial\theta_a^*}{\partial\tilde{u}}\right)\Big) \end{matrix} [E] & \frac{EI}{2lh_a}\frac{\partial\theta_a^*}{\partial\tilde{w}}\{I\} & \frac{EI}{2lh_a}\frac{\partial\theta_a^*}{\partial\tilde{w}}\{I\} \\ & \begin{matrix} \Big(N\frac{\partial^2\delta}{\partial\tilde{w}^2} + M_a\frac{\partial^2\theta_a^*}{\partial\tilde{w}^2} \\ +\frac{EI}{l}\frac{1}{h_a}\left(\frac{\partial\theta_a^*}{\partial\tilde{w}}\right)^2\Big) \end{matrix} [E] & \frac{EI}{2lh_a}\frac{\partial\theta_a^*}{\partial\tilde{u}}\{I\} & \frac{EI}{2lh_a}\frac{\partial\theta_a^*}{\partial\tilde{u}}\{I\} \\ & & \frac{EI}{4l}\left(\frac{1}{h_a}+\frac{1}{h_s}\right) & \frac{EI}{4l}\left(\frac{1}{h_a}-\frac{1}{h_s}\right) \\ \text{Symmetric} & & & \frac{EI}{4l}\left(\frac{1}{h_a}+\frac{1}{h_s}\right) \end{bmatrix} \tag{2.34}$$

$$\{A_{nd}\} = \begin{Bmatrix} \left(\frac{\partial\delta}{\partial\tilde{w}} + \frac{d}{dn}\left(\frac{1}{h_a}\right)\frac{\partial\theta_a^*}{\partial\tilde{w}}\theta_a^*\right)\{I\} \\ \left(\frac{\partial\delta}{\partial\tilde{u}} + \frac{d}{dn}\left(\frac{1}{h_a}\right)\frac{\partial\theta_a^*}{\partial\tilde{u}}\theta_a^*\right)\{I\} \\ \frac{1}{2}\left(\frac{d}{dn}\left(\frac{1}{h_a}\right)\theta_a^* + \frac{d}{dn}\left(\frac{1}{h_s}\right)\theta_s^*\right) \\ \frac{1}{2}\left(\frac{d}{dn}\left(\frac{1}{h_a}\right)\theta_a^* - \frac{d}{dn}\left(\frac{1}{h_s}\right)\theta_s^*\right) \end{Bmatrix} \tag{2.35}$$

$$A_{nn} = \frac{l^3}{2EI}\left(\frac{d^2}{dn^2}\left(\frac{1}{h_a}\right)\theta_a^{*2} + \frac{d^2}{dn^2}\left(\frac{1}{h_s}\right)\theta_s^{*2}\right) - \frac{l}{EA} \tag{2.36}$$

$$\{B_d\} = \begin{Bmatrix} \left(N\frac{\partial\delta}{\partial\tilde{w}} + M_a\frac{\partial\theta_a^*}{\partial\tilde{w}}\right)\{I\} \\ \left(N\frac{\partial\delta}{\partial\tilde{u}} + M_a\frac{\partial\theta_a^*}{\partial\tilde{u}}\right)\{I\} \\ \frac{1}{2}\left(M_a + M_s\right) \\ \frac{1}{2}\left(M_s - M_a\right) \end{Bmatrix} \tag{2.37}$$

$$B_n = \delta + \frac{1}{2}\frac{d}{dn}\left(\frac{1}{h_a}\right)\theta_a^{*2}l + \frac{1}{2}\frac{d}{dn}\left(\frac{1}{h_s}\right)\theta_s^{*2}l - \frac{Nl}{EA} \tag{2.38}$$

$$\{I\} = \begin{Bmatrix} -1 \\ 1 \end{Bmatrix} \qquad [E] = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{2.39}$$

The load-deflection curve generated using the finite element model in Eq. (2.31) is compared against the original Williams' problem with $l\sin\beta = 0.32$ in Fig. 2.13. Other cases with $l\sin\beta = 0.38$ and $l\sin\beta = 0.44$ are shown in Fig. 2.14 and Fig. 2.15, respectively. The Scalar Fixed Point Homotopy Method, Eq. (2.12), is used to generate the load-deflection curves for the finite element model for all three cases. As shown, the finite element model accurately describe the load-deflection characteristics of the Williams' toggle as it agrees well with the solutions of the scalar NAE presented in [[32]] and summarized in Eqs. 2.17 – 2.21. The Scalar Fixed-point Homotopy Method successfully solved the FEM equations capturing the load-deflection relation around the limit points, at which the Newton's method fails to find the solution, as shown next.

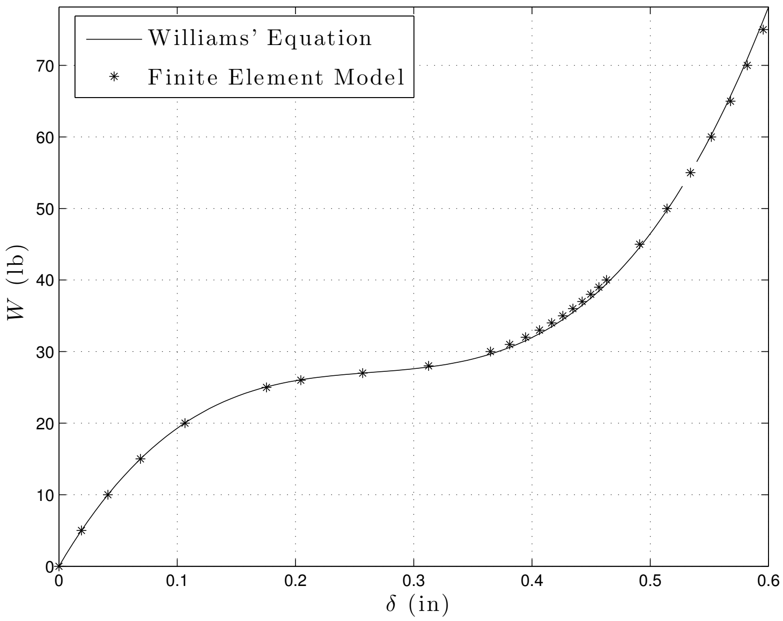

Figure 2.13: Load-Deflection, Williams' Equation & Finite Element, $l \sin\beta = 0.32$, [29]

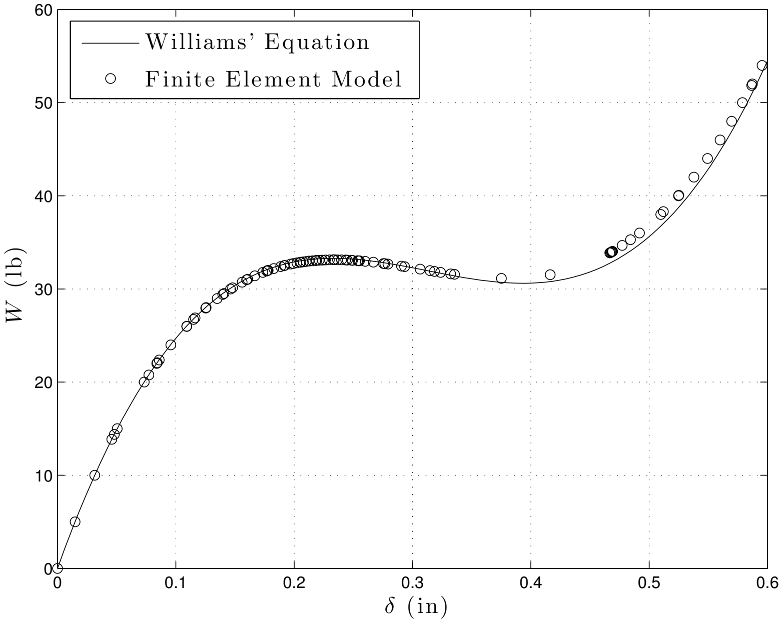

Figure 2.14: Load-Deflection, Williams' Equation & Finite Element, $l \sin \beta = 0.38$, [29]

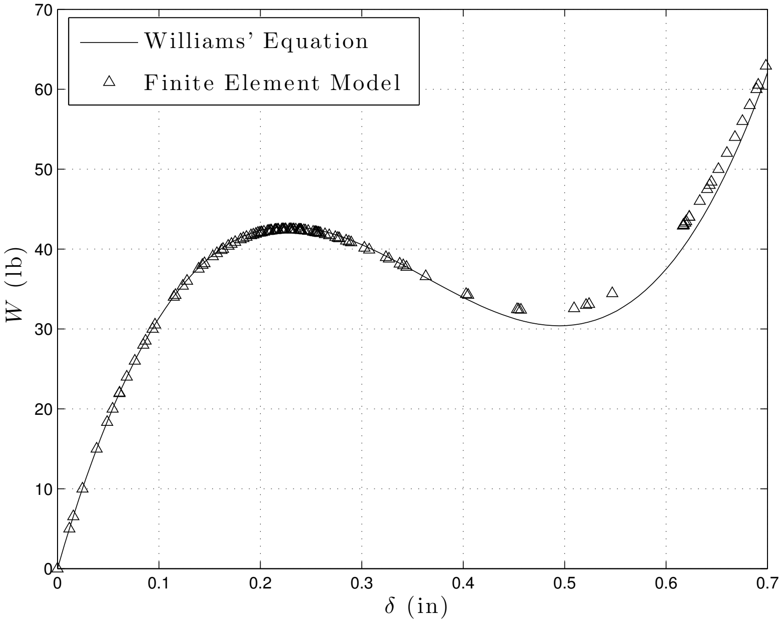

Figure 2.15: Load-Deflection, Williams' Equation & Finite Element, $l \sin \beta = 0.44$, [29]

2.3.3 Solution of the Finite Element Model Using Scalar Homotopy Methods

The Scalar Fixed-point Homotopy Method, Eq. (2.15), and the Scalar Newton Homotopy Method, Eq. (2.16), are both applied to the finite element model to solve for the deflection given a specific load. As with the scalar example, the case of $l \sin \beta = 0.44$ is examined with the same value of the load applied near the limit point. Setting the tolerance for the relative residual error to be 10^{-6}, the two methods are compared with Newton's method and the results are shown in Table 2.4.

Table 2.4: Solution of finite element model for loading near limit point, [29]

Method	**No. Iteration, N**	**Achieved Accuracy**
Newton's Method	$N = 1000$	4.2894
Scalar Fixed-point Homotopy	$N = 160$	6.7×10^{-8}
Scalar Newton Homotopy	$N = 500$	2.4×10^{-8}

Both Scalar Homotopy Methods proved superior to the Newton's method, as both converged to the solution with the required accuracy whereas the Newton's method failed to find the solution after 1000 iterations. A zoomed in plot is shown in Fig. 2.16 to illustrate the oscillating behavior of Newton's method and its failure to find the solution. It must be noted that due to the quadratic convergence nature of Newton's method, divergence can generally be shown after $10 - 15$ iterations. In the comparisons presented here the max number of iterations is set to 1000 in order to accommodate for the Scalar Homotopy Methods convergence rate and to compare them with Newton's method. The Scalar Fixed-point Homotopy Method converged in 160 iterations, Fig. 2.17), which is about one third the number of iterations required by the Scalar Newton Homotopy Method, Fig. 2.18. This makes the Scalar Fixed-point Homotopy Method more suitable for solving the problem of

Williams' toggle, whereas the Scalar Newton Homotopy Method provides a valid alternative to obtain the solution. The Scalar Homotopy Methods developed in this work and in previous works are suitable to solve general nonlinear finite element models with very high accuracy, without inverting the tangent stiffness matrix, and without having to use the computationally expensive arc-length methods.

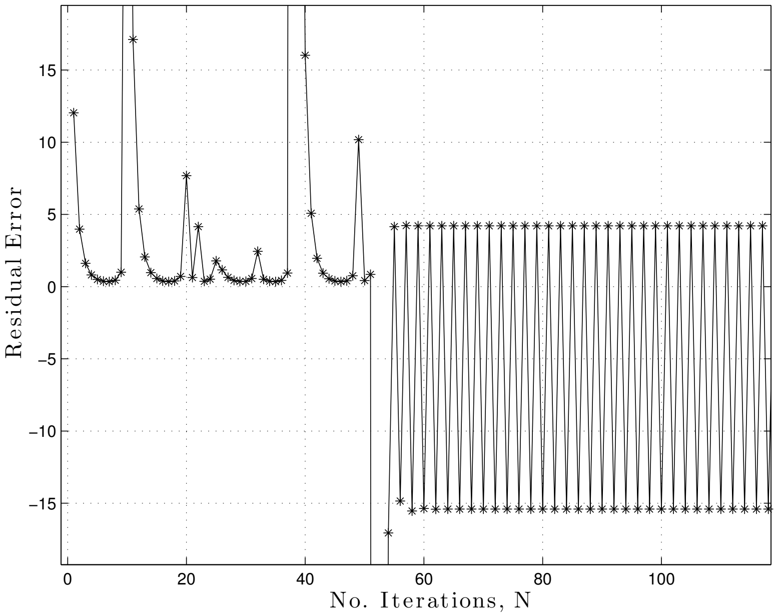

Figure 2.16: Residual Error in Newton's Method, [29]

In order to illustrate the efficiency of the scalar homotopy methods the tolerance for the relative residual error is relaxed to match existing finite element solvers (0.1%). For this case an external load of 44 lb. is applied and the results are shown in Table 2.5. The two methods achieved the required accuracy within 14 iterations, which demonstrates the power of the scalar homotopy methods in solving engineering problems and the fast convergence that can be achieved when addressing such

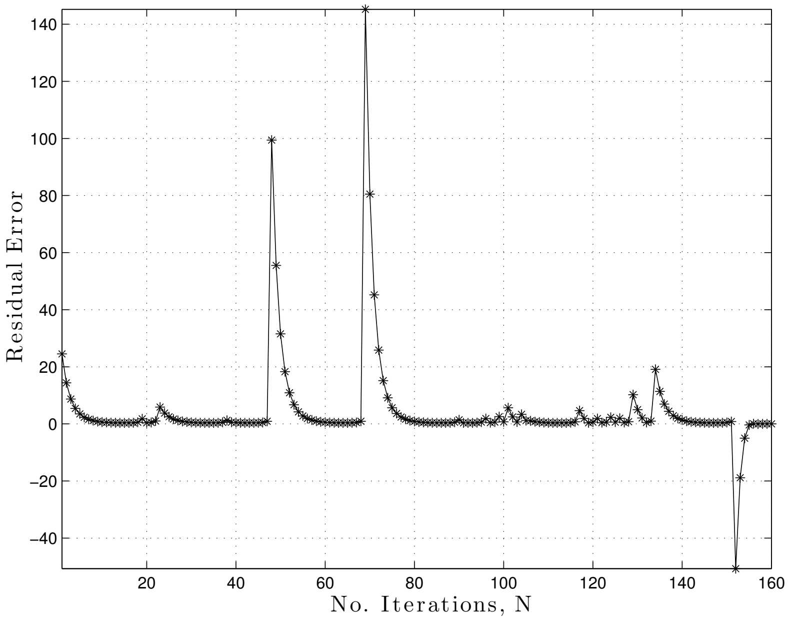

Figure 2.17: Residual Error in Scalar Fixed-point Homotopy Method, [29]

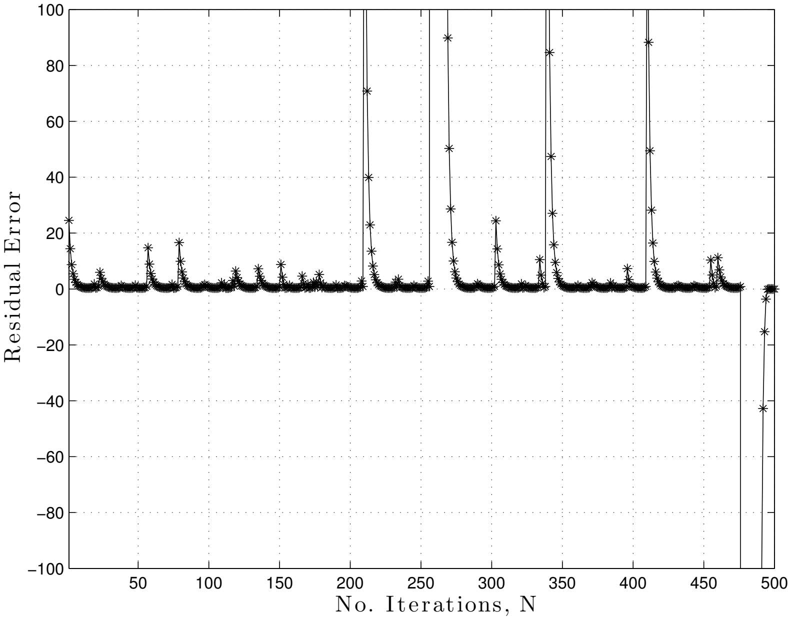

Figure 2.18: Residual Error in Scalar Newton Homotopy Method, [29]

problems. The Newton's method diverged with the same oscillatory non-convergent behavior shown in Fig. 2.16 for up to 1000 iterations. Figs. 2.19-2.20 show the path to convergence of the Scalar Fixed-point Homotopy Method and the Scalar Newton Homotopy Method, respectively.

Table 2.5: solution of finite element model for loading near limit point, [29]

Method	**No. Iteration,** N	**Achieved Accuracy**
Newton's Method	$N = 1000$	0.3042
Fixed-point Homotopy	$N = 14$	3.9×10^{-4}
Newton Homotopy	$N = 14$	4.02×10^{-4}

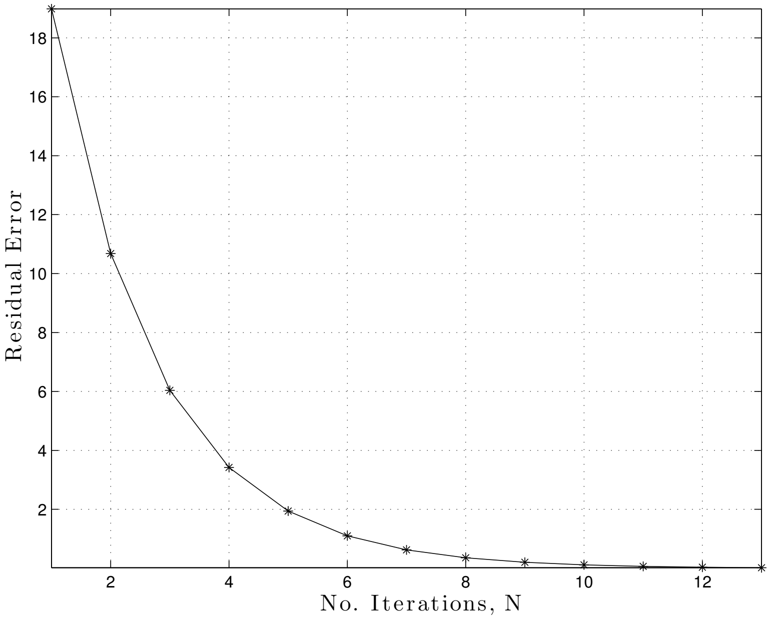

Figure 2.19: Residual Error in Scalar Fixed-point Homotopy Method, [29]

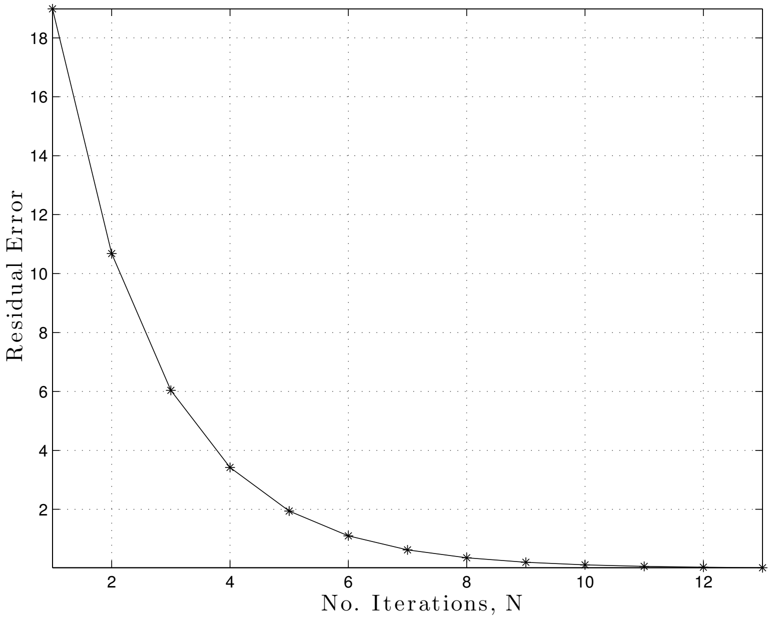

Figure 2.20: Residual Error in Scalar Newton Homotopy Method, [29]

The Scalar Homotopy Method is applied to the solution of post-buckling and limit load problems of plane frames considering geometrical nonlinearities. Explicitly derived tangent stiffness matrices and nodal forces of large-deformation beam-column members are adopted following the work in, [31]. By using the Scalar Homotopy Method, nodal displacements of the equilibrium state are iteratively solved for, without inverting the Jacobian (tangent stiffness) matrix and without using complex arc-Length methods. This simple method thus saves computational time and avoids the problematic behavior of the Newton's method when the Jacobian matrix is singular. The simple Williams' toggle is presented in this work, however, the method is applicable to general finite element analyses of space frames, plates, shells and elastic-plastic solids.

2.4 Geodetic Coordinate Transformation

Frequently, we encounter especially important NAEs, and instead of simply adopting a standard root solving or homotopy method, it is useful to study the particular problem in more detail in order to establish an especially efficient and robust method. Such a problem arises in satellite geodesy and has been the subject of significant historical studies. Here we revisit this problem and establish a new highly efficient solution that provides global accuracy with a non-iterative sequence of calculations.

A frequent calculation for satellites in Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO) involves inverting transformations between 3D satellite Cartesian Earth centered coordinates and geodetic coordinates. The geodetic coordinates consist of λ_g, ϕ_g, and h, which denote the geodetic longitude of the satellite sub-point g, the geodetic latitude of the satellite, and the height of the satellite above the reference Earth elliptical surface along the surface normal from the geodetic ellipsoid to the satellite position. Referring to Fig. 2.21, the transformation from geodetic coordinates to Cartesian (x_s, y_s, z_s) coordinates is given in [33] by:

$$\begin{aligned} x_s &= \left(N\left(\phi_g\right)+h\right)\cos\phi_g\cos\lambda_g \\ y_s &= \left(N\left(\phi_g\right)+h\right)\cos\phi_g\sin\lambda_g \\ z_s &= \big(N\left(\phi_g\right)\left(1-e^2\right)+h\big)\sin\phi_g \end{aligned} \tag{2.40}$$

where, $N(\phi_g) = a/\sqrt{1-e^2\sin^2\phi_g}$ denotes the ellipsoid radius of curvature in the prime vertical plane defined by vectors $\hat{\boldsymbol{n}}$ (ellipsoid outward normal) and $\hat{\boldsymbol{\tau}}$ (local east), h is assumed to lie along $\hat{\boldsymbol{n}}$, a denotes the semi-major axis, b denotes the semi-minor axis and e denotes the eccentricity of the Earth reference ellipsoid. The geodetic longitude is trivially obtained from Eq. (2.40) as,

$$\lambda_g = \tan^{-1}\left(\frac{y_s}{x_s}\right) \tag{2.41}$$

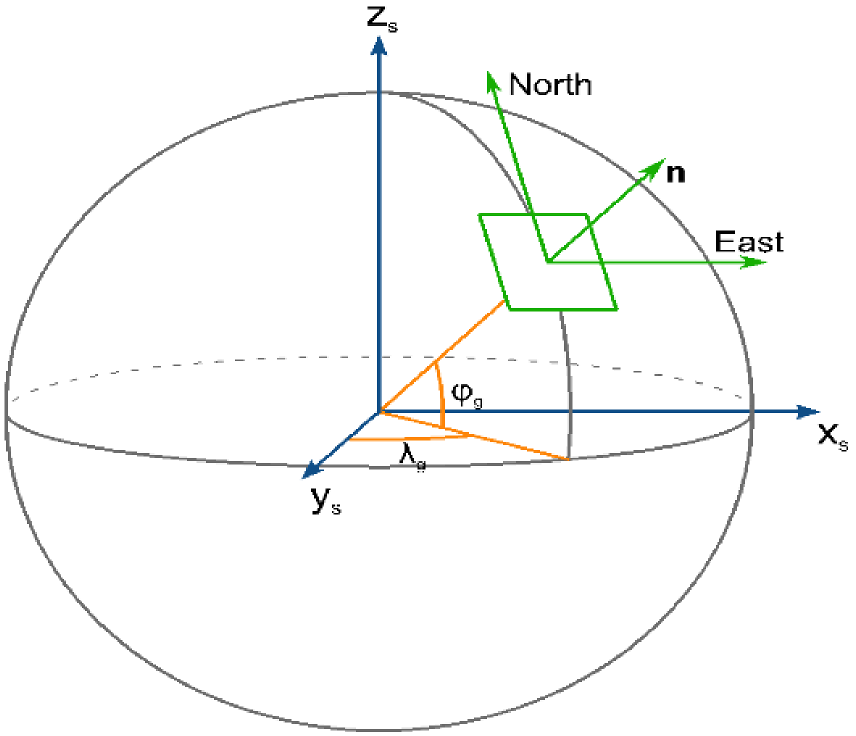

Figure 2.21: Geodetic & Cartesian Coordinates, [30]

Many methods have been proposed for implementing the inverse of the transformation presented in Eq. (2.40). The nonlinear Cartesian-to-Geodetic transformation problem is challenging, as geometrical singularities plague many published solution strategies. The for the geodetic longitude is elementary and non-iterative. The most common problem encountered is the need for handling sensitive quartic polynomial solutions, [34–37]. The analytic complexity of the problem arises because the geodetic latitude and satellite height solution algorithms are coupled and highly nonlinear. Three classes of methods have been proposed: (i) closed-form solutions involving cubic and quartic polynomials, (ii) approximate perturbation methods, and (iii) successive approximation algorithms similar to those in the previous sections. The closed-form class of solution algorithms typically introduce sequences of trigonometric transformations that exploit identities to simplify the governing equation. Important examples of this approach include the following: (i) the well-known solution in [38], where the reduced latitude is iterated in Newton's method; (ii) a closed-form solution for a high-order algebraic equation, [39]; (iii) introducing the geodetic height of the satellite to develop an elliptic integral-based arc-length solution, [40]; (iv) development of an approximate closed-form solution, [41]; and (v) introduction of complicated algebraic transformations to develop a series solution, [42]. The closed-form solutions are generally highly accurate, but they are computationally more expensive to perform than a single iteration of most published algorithms.

Many iterative techniques have been proposed that exploit the special structure of the problem at hand. Early examples of this approach include the work in [33] which influenced the GPS-based need for the geodetic transformation methods developed in [43–45]. Several innovative problem formulations have been proposed, such as in [34, 35, 46, 47]. Geometric singularities plague many of these iterative strategies. To avoid troublesome singularities, several authors have investigated vector methods, as presented in [37, 48]. In [49], an elegant optimization-based strategy is presented. Accelerated convergence techniques are considered in [50], presenting a third-order version of Newton's method, known as Halleys method. In [51], Turner has presented a very fast singularity-free second-order perturbation solution that introduces an artificial perturbation variable to transform the classical quartic solution problem into a singularity-free non-iterative quadratic equation problem. In a more recent addition to iterative methods, [52], the projection of a point on the reference ellipsoid is used to solve a system of nonlinear equations using second and third order Newton's method. The results presented by the authors show millimeter accuracy in height and 10^{-8} accuracy in latitude with the third-order approach.

2.4.1 A Simple Perturbation Solution

The problem is formulated by introducing a local coordinate system that tracks the local $x-y$ axis motion of the satellite. In the local coordinate system, a simplified perturbation solution is developed in the $\tau - z$ plane by defining a vector constraint of the form

$$\boldsymbol{r} - \boldsymbol{r}_g = h\hat{\boldsymbol{n}} = \boldsymbol{0} \tag{2.42}$$

where, $\boldsymbol{r} = (r_{xy}, z)$ denotes the satellite position vector, with $r_{xy} = \sqrt{x^2 + y^2}$, $\boldsymbol{r}_g = [a\cos\phi_c, b\sin\phi_c]^{\mathrm{T}}$ the ground track point, ϕ_c the geocentric latitude, h the height of the satellite above the Earth's surface, and $\hat{\boldsymbol{n}}$ the unit vector normal to the Earth's surface pointing to the satellite is given by,

$$\hat{\boldsymbol{n}} = \frac{\left[\frac{\cos\phi_c}{a}, \frac{\sin\phi_c}{b}\right]^{\mathrm{T}}}{\sqrt{\left(\frac{\cos\phi_c}{a}\right)^2 + \left(\frac{\sin\phi_c}{b}\right)^2}} \tag{2.43}$$

See Fig. 2.22 for an illustration. Expanding Eq. (2.42) provides the two necessary

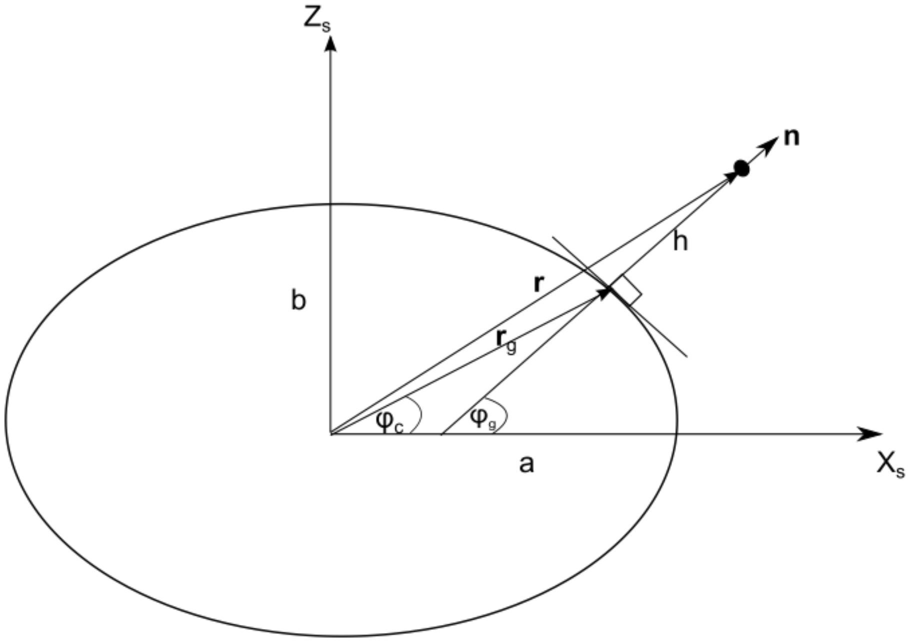

Figure 2.22: Geodetic Vs. Geocentric Latitude, [30]

conditions in the two unknowns, h, ϕ_c:

$$\begin{aligned} r_{xy} - a\cos\phi_c - \frac{h\cos\phi_c}{a\sqrt{\left(\frac{\cos\phi_c}{a}\right)^2 + \left(\frac{\sin\phi_c}{b}\right)^2}} = 0 \\ z - b\sin\phi_c - \frac{h\sin\phi_c}{b\sqrt{\left(\frac{\cos\phi_c}{a}\right)^2 + \left(\frac{\sin\phi_c}{b}\right)^2}} = 0 \end{aligned} \tag{2.44}$$

The equations are clearly highly nonlinear. To begin the simplification process, we replace a in Eq. (2.44) with,

$$a = b(1+p), \qquad p \approx 0.00314 \tag{2.45}$$

which exploits the natural parameter of the problem and transforms Eq. (2.44) into,

$$\begin{aligned} r_{xy} - b(1+p)\cos\phi_c - \frac{h\cos\phi_c}{\sqrt{(\cos\phi_c)^2 + ((1+p)\sin\phi_c)^2}} &= 0 \\ z - b\sin\phi_c - \frac{h\sin\phi_c}{\sqrt{\left(\frac{\cos\phi_c}{(1+p)}\right)^2 + (\sin\phi_c)^2}} &= 0 \end{aligned} \tag{2.46}$$

An approximate solution is recovered by assuming that the geocentric latitude and the satellite height are expanded in the power series representations as:

$$\begin{aligned} \phi_c &= \phi_0 + p\phi_1 + p^2\phi_2 + \dots \\ h &= h_0 + ph_1 + p^2h_2 + \dots \end{aligned} \tag{2.47}$$

Substituting Eq. (2.47) into Eq. (2.46) and collecting terms in powers of p the expansion is carried up to fourth order in the powers of p. Hence, for p^0,

$$\begin{aligned} h_0 &= \sqrt{r_{xy}^2 + z^2} - b \\ \phi_0 &= 2\tan^{-1}\left(\frac{\sqrt{r_{xy}^2 + z^2}}{z} - \frac{r_{xy}}{z}\right) \end{aligned} \tag{2.48}$$

for p^1,

$$\begin{aligned} h_1 &= -b\cos^2\phi_0 \\ \phi_1 &= \frac{b-h_0}{2(b+h_0)}\sin(2\phi_c) \end{aligned} \tag{2.49}$$

for p^2,

$$\begin{aligned} h_2 &= \frac{b(3b-h_0)}{8(b+h_0)}\sin^2(2\phi_0) \\ \phi_2 &= \frac{h_0^2 - 4bh_0 + 3b^2}{8(b+h_0)^2}\sin(4\phi_0) + \frac{\sin(2\phi_0)}{4} \end{aligned} \tag{2.50}$$

for p^3

$$\begin{aligned} h_3 &= \frac{b\sin^2(2\phi_0)}{8(b+h_0)^2}\left[(h_0-3b)^2\cos^2\phi_0 + 4b(h_0-b)\right] \\ \phi_3 &= \frac{\sin(2\phi_0)}{6(b+h_0)^3}\left[C_1\cos^4\phi_0 + C_2\cos^2\phi_0 + C_3\right] \end{aligned} \tag{2.51}$$

and finally for p^4

$$\begin{aligned} h_4 &= \frac{b\sin^2(2\phi_0)}{32(b+h_0)^3}\left[C_4\cos^4\phi_0 + C_5\cos^2\phi_0 + C_6\right] \\ \phi_4 &= \frac{\sin(2\phi_0)}{4(b+h_0)^4}\left[C_7\cos^6\phi_0 + C_8\cos^4\phi_0 + C_9\cos^2\phi_0 + C_{10}\right] \end{aligned} \tag{2.52}$$

where, the set of coefficients $C_1, \ldots, C_{10}$ in the third and fourth order expansions are shown in Table 2.6 These analytic results are very compact for a fourth-order perturbation expansion. The conversion from the geocentric to the geodetic latitude is given by,

Table 2.6: Set of coefficients in third & fourth order expansions, [30]

Coeffecient	**Expression**
C_1	$-4h_0^3 + 37b^3 - 66b^2h_0 + 33bh_0^2$
C_2	$h_0^3 - 31b^3 + 75b^2h_0 - 33bh_0^2$
C_3	$3b\,(b^2 + 4h_0^2 - 6bh_0)$
C_4	$139b^3 - 5h_0^3 + 49bh_0^2 - 143b^2h_0$
C_5	$h_0^3 - 127b^3 + 163b^2h_0 - 45bh_0^2$
C_6	$4b\,(-110bh_0 + h_0^2 + 5b^2)$
C_7	$4h_0^4 + 118b^4 + 198b^2h_0^2 - 266b^3h_0 - 54bh_0^3$
C_8	$-155b^4 = 2h_0^4 + 67bh_0^3 + 431b^3h_0 - 315b^2h_0^2$
C_9	$49b^4 - 15bh_0^3 - 185b^3h_0 + 135b^3h_0^2$
C_{10}	$2b^2\,(10bh_0 - 5h_0^2 - b^2)$

$$\phi_g = \tan^{-1}\left(\frac{\cos\phi_c}{a/b\sin\phi_c}\right) \tag{2.53}$$

2.4.2 LEO to GEO Satellite Coordinates Transformation

The perturbation expansion method is used to carry out the coordinate transformation for cases of LEO-to- GEO orbits. Using the WGS84, the forward transformation is carried out first, then the perturbation solution is applied, and the results are compared against the original values, which represent the true values for the inverse solution. The error is simply calculated as,

$$\begin{aligned} \Delta\phi_g &= \phi_g - \hat{\phi}_g \\ \Delta h &= h - \hat{h} \end{aligned} \tag{2.54}$$

where, $\hat{*}$ represents the perturbation solution. For the sake of demonstration a longitude angle of 30° is utilized. The geodetic latitude, ϕ_g, is swept for angles from -90 to 90 degrees and the height is swept from 200 KM (LEO) to $35,000$ KM (GEO). We have found that the accuracy is invariant with respect to longitude. First, the expansion is carried to second order, and the errors in latitude and height are plotted as functions of the true latitudes and heights as shown in Fig. 2.23 and Fig. 2.24, respectively. The expansion is then carried out to third order, and the errors in latitude and height are shown in Fig. 2.25 and Fig. 2.26, respectively. Finally, the fourth order expansion is used, and the error results are shown in Fig. 2.27 and Fig. 2.28.

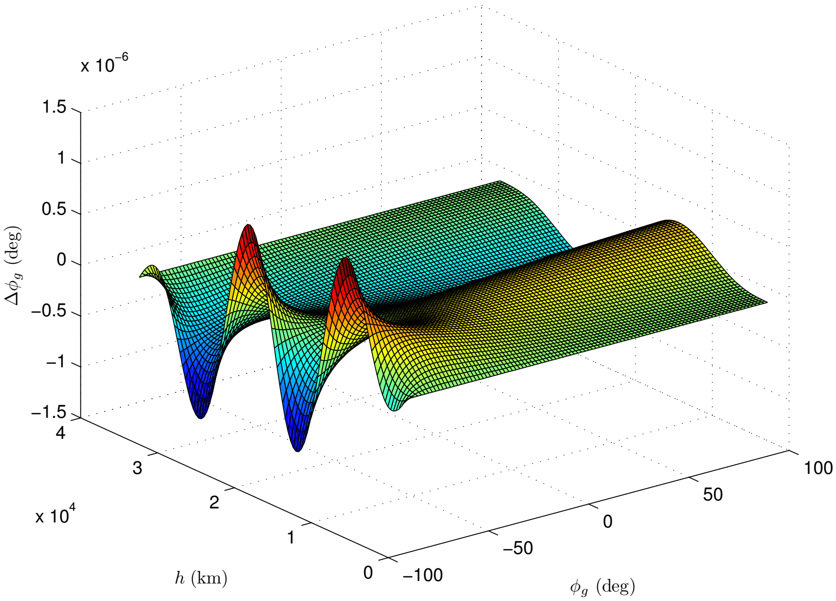

Figure 2.23: Errors in Latitude, Second Order Expansion, [30]

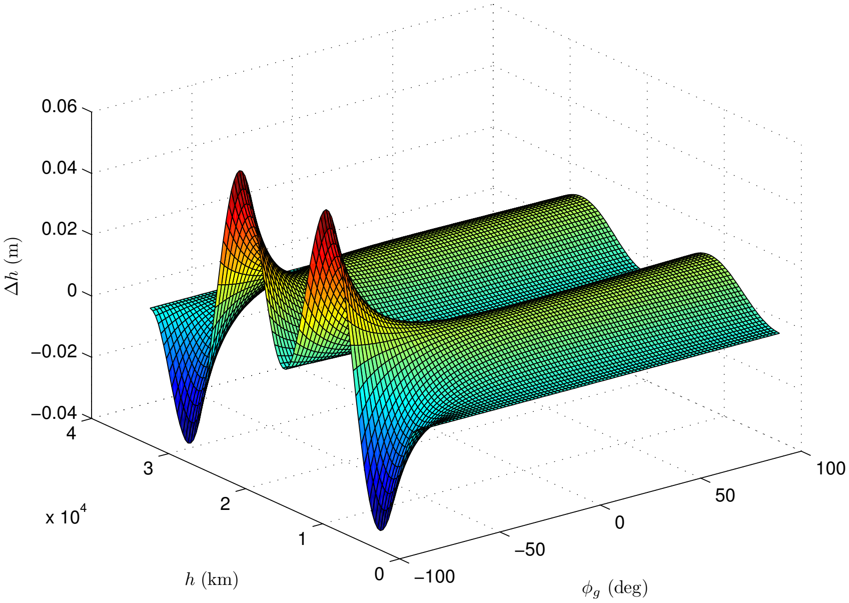

Figure 2.24: Errors in Height, Second Order Expansion, [30]

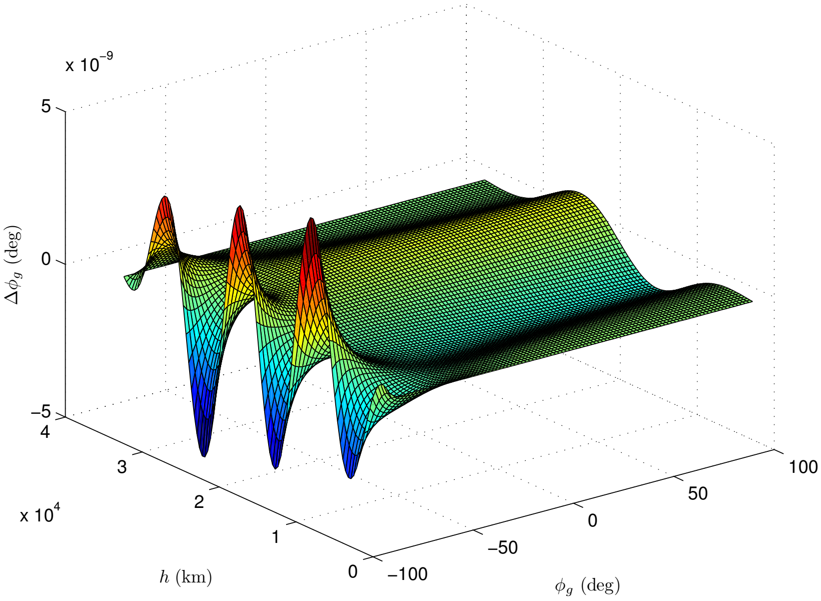

Figure 2.25: Errors in Latitude, Third Order Expansion, [30]

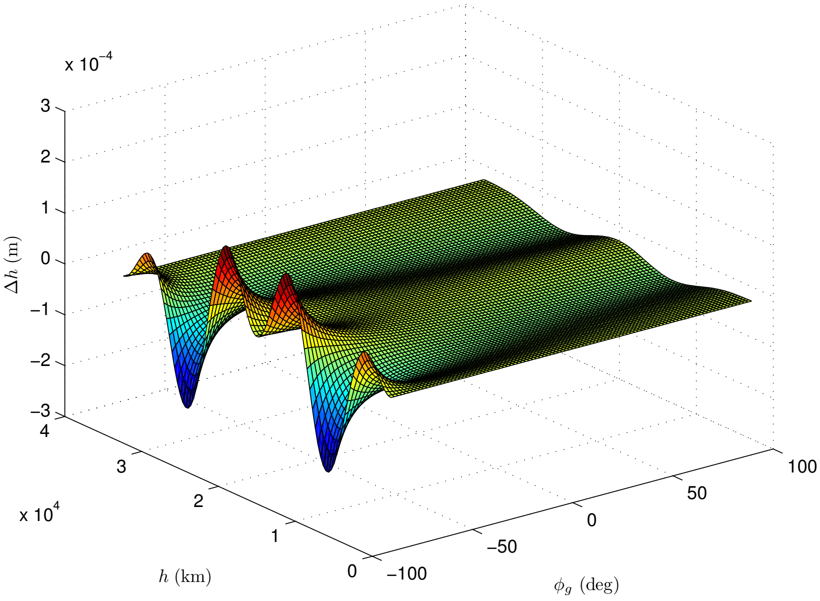

Figure 2.26: Errors in Height, Third Order Expansion, [30]

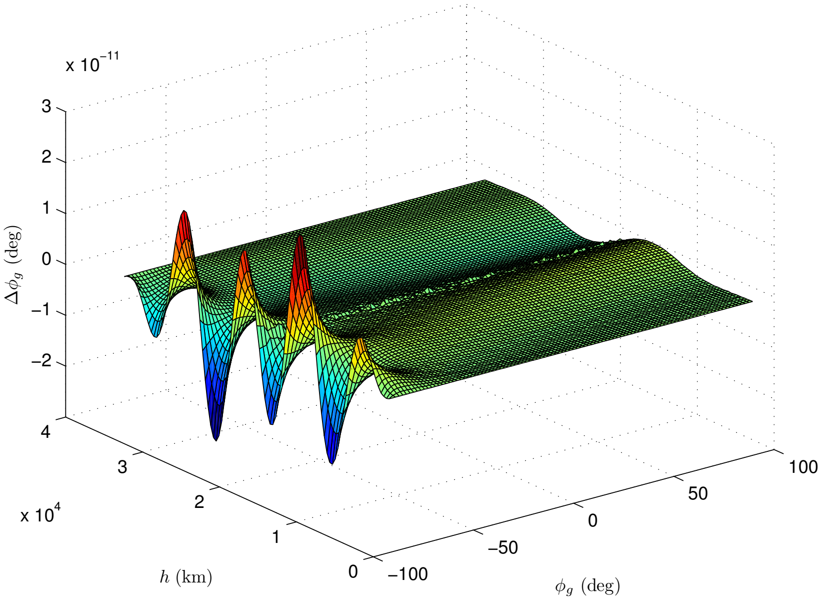

Figure 2.27: Errors in Latitude, Fourth Order Expansion, [30]

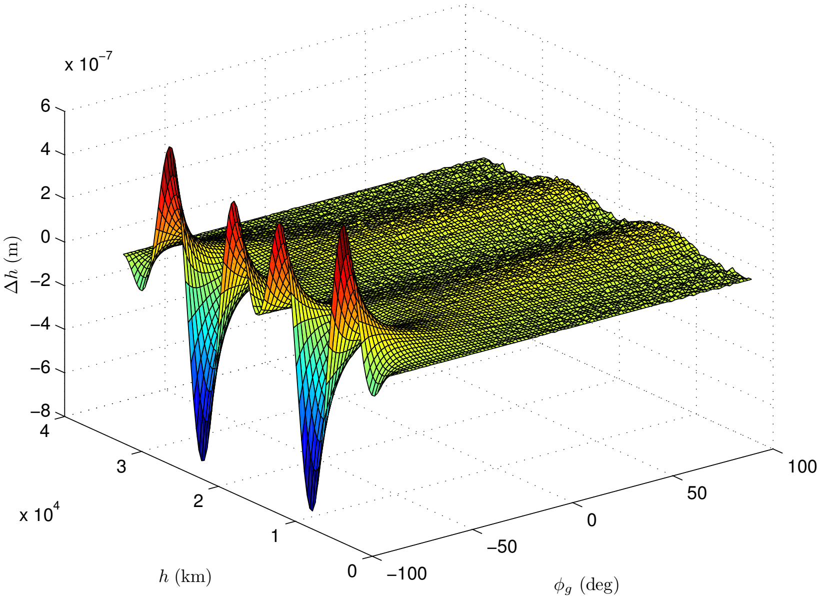

Figure 2.28: Errors in Height, Fourth Order Expansion, [30]

The improvement of accuracy is quite obvious as the order of expansion is increased. Two orders of magnitude improvement is achieved by adding the third order terms to each of the coordinates. Another two orders of magnitude improvement is achieved with the fourth order terms. In height, millimeter accuracy is achieved at the fourth order expansion level. As is evident, the fourth order expansion produces essentially negligible error globally and can be considered an explicit non-iterative solution. This shows the fast convergence nature and the accuracy of the perturbation solution. The results also demonstrate that higher order approximations are not needed as they will not provide additional useful information for the inversion process.

Earth-Centered Earth-Fixed (ECEF) to geodetic coordinate transformation has been examined with several numerical and analytical approaches throughout the literature. A non-iterative expansion based approach inspired by the Earth's perturbed geometry is introduced, where the expansion parameter is nothing but the ratio of the Earth semi-major axis and semi-minor axis subtracted from 1. The expansion is carried out to second, third, and fourth orders. A numerical example is introduced to compare the accuracy at each order of expansion. Significant improvement in accuracy is demonstrated as the order of expansion is increased, and at fourth order, millimeter accuracy is achieved in height and 10^{-11} degree error in latitude. Those errors at such low orders of the expansion are proof of the effectiveness of the method and its potential in solving such a highly nonlinear transformation non-iteratively. The method can be further streamlined for timing studies, but in general it is a clean straightforward approach to the coordinate transformation problem that utilizes a physical perturbation parameter and that proved to be globally accurate and efficient.

3. ANALYTIC METHODS IN STRUCTURAL MECHANICS AND ASTRODYNAMICS*

In this chapter, two analytic approaches are introduced to address problems in structural mechanics and astrodynamics. First, the problems of a symmetric flexible rotating spacecraft modeled as a distributed parameter system of a rigid hub attached to two flexible appendages with tip masses is addressed. Traditional methods typically initiate with a spatial discretization process such as the finite element method to obtain a high order, but now truncated system of ordinary differential equations before initiating controller design. The effects of truncation error therefore corrupts to some degree the model of the controlled system. We present here an approach which, for a class of systems, avoids spatial truncation altogether in designing the control law. Hamilton's extended principle is utilized to present a general treatment for deriving the dynamics of multi-body dynamical systems to establish a hybrid system of integro-partial differential equations that model the evolution of the system in space and time. A Generalized State Space (GSS) system of equations is constructed in the frequency domain to obtain analytic transfer functions for the rotating spacecraft. The frequency response of the generally modeled spacecraft and a special case with no tip masses are presented. Numerical results for the system frequency response obtained from the analytic transfer functions are presented and compared against the classical assumed modes numerical method. The truncation-error-free analytic results are shown to agree well with the classical numerical solutions, thus validating the truncated model. Fundamentally, we show that the rigorous transfer

*Part of this chapter is reprinted with permission from "Dynamics and Controls of a Generalized Frequency Domain Model Flexible Rotating Spacecraft" by Elgohary, T. A., Turner, J. D. and Junkins, J. L., 2014. *AIAA SpaceOps Conference*, Copyright [2014] by AIAA.

function, without introduction of spatial discretization, can be directly used in control law design. The frequency response of the system is used in a classical control problem where a Lyapunov stable controller is derived and tested for gain selection. The correlation between the controller design in the frequency domain utilizing the analytic transfer functions and the system response is analyzed and verified. The derived analytic transfer functions provide a powerful tool to test various control schemes in the frequency domain and a validation platform for existing numerical methods for distributed parameters models.

The second analytic method addresses the orbit propagation of the two-body problem. Arbitrary order Taylor expansions are developed for propagating the trajectories of space objects subject to two-body gravitational potential. Arbitrary order time derivative models are made possible by introducing a nonlinear change of variables, where the new variables and the transformed equations of motion are analytically generated by Leibniz product rule. The analytic continuation algorithm is developed and numerical examples are introduced for a Low Earth Orbit (LEO) and a High Eccentricity Orbit (HEO) cases. Numerical simulations and timing comparison results are presented for comparing the performance of the series solution with other state of the art integrators.

3.1 Analytic Transfer Functions for the Dynamics and Control of Flexible Rotating Spacecraft Performing Large Angle Maneuvers

A maneuvering flexible spacecraft is often modeled as coupled rigid hub with attached flexible beam-like sub-structures. A widely used model describing such systems is shown in Fig. 3.1 where a flexible rotating spacecraft is modeled as two symmetric flexible appendages with identical tip masses attached to a rotating rigid hub. Such models are described by coupled systems of Integro-Partial Differential Equations (IPDEs), [53–59]. Solution techniques presented in these works are mainly numerical based on spatial discretization approaches that apply the finite element method and/or the assumed modes technique. Numerical solutions in general are approximate and the accuracy is a function of the number of elements/modes chosen, which can impose high computational cost, and there is always the issue of truncation errors associated with the spatial discretization. Also in the case of the assumed modes technique the number of accurate modes can be limited by the numerical errors introduced by the matrix operations, [58]. As a natural extension for such techniques the control problem is addressed in several works for optimality and/or robustness, [58,60–64]. This work presents a new approach based on deriving analytic transfer functions for the hybrid system that has been recently introduced by Elgohary and Turner, [65–67]. In those works, results from the analytic solution have been successfully compared and verified against the classical assumed modes method.

The control problem of a single axis rotating flexible spacecraft has been addressed extensively utilizing several controls and modeling schemes. The optimal control problem of a rotating hub with symmetric four flexible appendages is presented as a numerical example, [68]. A set of admissible functions that meet both physical and geometrical boundary conditions is chosen. The effectiveness of the minimization rely on the number of modes retained in the series of the chosen admissible function. Finite elements techniques besides the assumed modes approach are used to solve similar problems, [58]. The natural frequencies of the system are calculated and the two methods are compared in terms of accuracy and the required number of elements/modes. Several other flexible structures examples are similarly addressed, [60]. The optimal control problem is again addressed for various control schemes and penalty functions including free final time, free final angle and control rate penalty methods. Large angle maneuvers for a flexible spacecraft are addressed for a hub-four flexible appendages model, [61]. Two-point boundary value problems and kinematic nonlinearities are also addressed, [62]. In a more recent work, [69], the adaptive control problem is addressed for a similar rigid hub flexible appendage model. Because this work builds on the analytic solution for the integro-partial differential equation of motion, the proposed control scheme is independent of the truncation generated from the flexible modes admissible function. No series approximations are introduced. Several other works address similar problems with emphasis on optimality, [63, 64, 70–73], and/or robustness, [63, 64]. A comprehensive literature survey, [74], covers the modeling and control of flexible appendage in the controls community.

The existence of the exact transfer functions, [65–67], for the rigid hub flexible beam problem is utilized in developing frequency domain control schemes for the spacecraft, [75, 76]. The frequency response developed is used to obtain the gains required to implement a controller that drives the system from its initial state to a target state while controlling the vibrations of the flexible appendages. First, a general framework for the derivation of multi-body dynamics is presented. Following that technique, the dynamical equations for the model in Fig. 3.1 are derived. Next, the concept of the Generalized State Space (GSS) is presented with the steps to obtain its closed form solution and the associated analytic transfer functions. Numerical results of the frequency response for the general system in Fig. 3.1 and the no tip-mass special case are then presented. Finally, the control problem in the frequency domain for the derived analytic transfer functions is presented with numerical results for the rigid and flexible coordinates response.

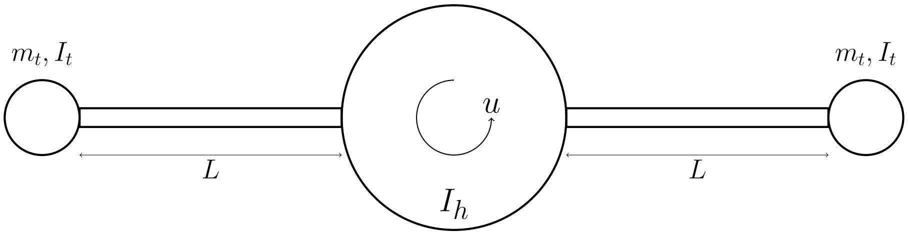

Figure 3.1: Symmetric Rotating Spacecraft, [76]

3.1.1 Dynamics of Multi-Body Hybrid Coordinate Systems

A hybrid coordinates dynamical system is described by m generalized coordinates describing the rigid body motion, denoted by $q_i = q_i(t), i = 1, \dots, m$ and n elastic coordinates, $w_j = w_j(\boldsymbol{P}, t), j = 1, \dots, n$, describing the relative elastic motion of a spatial position $\boldsymbol{P}$, [58]. Hence, $\boldsymbol{q} = [q_1, \dots, q_m]^{\mathrm{T}}$ and $\boldsymbol{w} = [w_1, \dots, w_n]^{\mathrm{T}}$.

For a general multi-body system of n beam-like flexible bodies and one spatial independent variable x_i, the kinetic and potential energy are assumed to have the general structure,

$$T = T_D(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \sum_{i=1}^{n} \int_{l_{0i}}^{l_i} \hat{T}_i(\mathrm{arg})\mathrm{d}x_i + T_B(\mathrm{arg}_\mathrm{B}) \tag{3.1}$$

$$V = V_D(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \sum_{i=1}^{n} \int_{l_{0i}}^{l_i} \hat{V}_i(\mathrm{arg})\mathrm{d}x_i + V_B(\mathrm{arg_B}) \tag{3.2}$$

where, $(*)_D$ denotes the energy of the rigid body, $(\hat{*})$ denotes the energy in the elastic domain, $(*)_B$ denotes the energy at the boundaries of the elastic domain, $\mathrm{arg} = \{\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{w}_i, \dot{\boldsymbol{w}}_i, \boldsymbol{w}_i, \boldsymbol{w}_i', \boldsymbol{w}_i'', \dot{\boldsymbol{w}}(l), \boldsymbol{w}(l), \boldsymbol{w}'(l), \dot{\boldsymbol{w}}'(l)\}$ and $\mathrm{arg_B} = \{\boldsymbol{q}, \dot{\boldsymbol{q}}, \dot{\boldsymbol{w}}(l), \boldsymbol{w}(l), \boldsymbol{w}'(l), \dot{\boldsymbol{w}}'(l)\}$.

The Lagrangian can then be expressed as,

$$\begin{aligned} \mathcal{L} &= T - V \\ &= L_D + \sum_{i=1}^{n} \int_{l_{0i}}^{l_i} \hat{L}_i \mathrm{d}x_i + L_B \end{aligned} \tag{3.3}$$

where, $\mathrm{arg}, \mathrm{arg_B}$ are dropped for brevity, $L_D \equiv T_D - V_D$, $\hat{L}_i = \hat{T}_i - \hat{V}_i$, and $L_B = T_B - V_B$. The non-conservative virtual work can then be expressed as,

$$\delta W_{nc} = Q^{\mathrm{T}} \delta \boldsymbol{q} + \sum_{i=1}^{n} \left\{ \int_{l_{0i}}^{l_i} \hat{\boldsymbol{f}}_i^{\mathrm{T}}(x_i) \delta \boldsymbol{w}_i \mathrm{d}x_i + \boldsymbol{f}_i^{\mathrm{T}} \delta \boldsymbol{w}_i(l_i) + \boldsymbol{g}_i^{\mathrm{T}} \delta \boldsymbol{w}_i'(l_i) \right\} \tag{3.4}$$

where, Q is the non-conservative force associated with the rigid body coordinates $\boldsymbol{q}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ are the non-conservative force and torque, respectively, applied at the boundary, $x_i = l_i$. Hamilton's extended principle, Eq. (3.5), is then applied to obtain the set of coupled hybrid ordinary and partial differential equations and the associated boundary conditions, Eq. (3.6) through Eq. (3.9)

$$\int_{t_1}^{t_2} (\delta \mathcal{L} + \delta W_{nc}) = 0 \qquad \delta \boldsymbol{q} = \delta \boldsymbol{w}_i = 0 \quad \text{at} \quad t = t_1, t_2 \tag{3.5}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} = Q^{\mathrm{T}} \tag{3.6}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \hat{L}_i}{\partial \dot{\boldsymbol{w}}_i}\right) - \frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i} + \frac{\partial}{\partial x_i}\left(\frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i'}\right) - \frac{\partial^2}{\partial x_i^2}\left(\frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i''}\right) = \hat{\boldsymbol{f}}_i^{\mathrm{T}} \tag{3.7}$$

$$\begin{aligned}\left\{\frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i'} - \frac{\partial}{\partial x_i}\left(\frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i''}\right)\right\}\delta \boldsymbol{w}_i\Bigg|_{l_{0i}}^{l_i} &+ \left\{\frac{\partial \mathcal{L}_B}{\partial \boldsymbol{w}_i(l_i)} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}_B}{\partial \dot{\boldsymbol{w}}_i(l_i)}\right)\right\}\delta \boldsymbol{w}_i(l_i) \\ &+ \boldsymbol{f}_i^{\mathrm{T}}\delta \boldsymbol{w}_i(l_i) = 0\end{aligned} \tag{3.8}$$

$$\frac{\partial \hat{L}_i}{\partial \boldsymbol{w}_i''}\delta \boldsymbol{w}_i'\Bigg|_{l_{0i}}^{l_i} + \left\{\frac{\partial \mathcal{L}_B}{\partial \boldsymbol{w}_i'(l_i)} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}_B}{\partial \dot{\boldsymbol{w}}_i'(l_i)}\right)\right\}\delta \boldsymbol{w}_i'(l_i) + \boldsymbol{g}_i^{\mathrm{T}}\delta \boldsymbol{w}_i'(l_i) = 0 \tag{3.9}$$

where, $\mathcal{L}_B \equiv L_B + \sum_{i=1}^{n}\int_{l_{0i}}^{l_i} \hat{L}_i \mathrm{d}x_i$. Eq. (3.6) through Eq. (3.9) are used to derive the dynamics of the hybrid model presented in Fig. 3.1.

Considering the deformation and the coordinate system presented in Fig. 3.2, the inertial position and velocity of a point on the i-th flexible appendage for the multi-body system is given by,

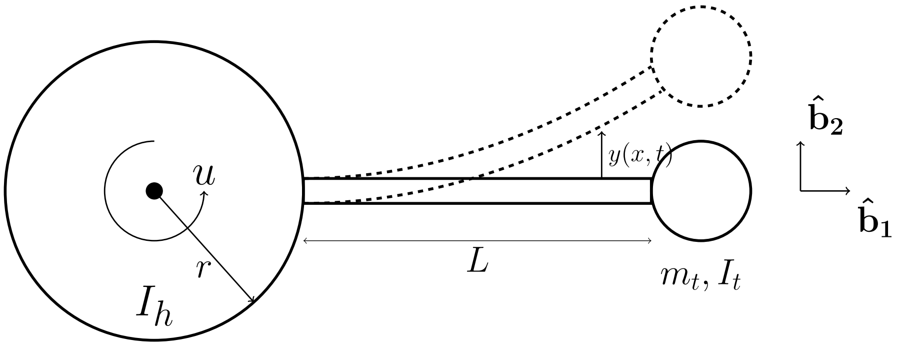

Figure 3.2: Deformation & Axis of Flexible Appendage, [76]

$$\boldsymbol{p}_i = (x_i + r)\,\hat{\mathbf{b}}_1 + y_i\hat{\mathbf{b}}_2 \tag{3.10}$$

$$\boldsymbol{v}_i = \dot{y}_i\hat{\mathbf{b}}_2 + \dot{\theta}\hat{\mathbf{b}}_3 \times \Big[(x_i + r)\,\hat{\mathbf{b}}_1 + y_i\hat{\mathbf{b}}_2\Big] \tag{3.11}$$

where, r is the rotating hub radius, L the length of the flexible appendage, $x \in [0, L]$ the position on the flexible appendage and y the transverse deflection of the flexible appendage. Neglecting the $y\dot{\theta}$ term in the velocity and assuming that the two appendages have the same deflection profiles, $y_1(x,t) = y_2(x,t)$, the kinetic and the potential energy for the model in Fig. 3.1 can be expressed as,

$$\begin{aligned} T &= T_{\text{hub}} + 2T_{\text{appendage}} + 2T_{\text{tip}} \\ T &= \frac{1}{2}I_{\text{hub}}\dot{\theta}^2 + \int_0^L \rho\left(\dot{y} + (x+r)\dot{\theta}\right)^2 \mathrm{d}x \\ &\quad + m_{\text{tip}}\left((r+L)\dot{\theta} + \dot{y}(L)\right)^2 + I_{\text{tip}}\left(\dot{\theta} + \dot{y}'(L)\right)^2 \end{aligned} \tag{3.12}$$

$$V = \int_0^L EI\left(y''\right)^2 \mathrm{d}x \tag{3.13}$$

The Lagrangian can then be constructed as,

$$\begin{aligned} \mathcal{L} = {} & \frac{1}{2}I_{\text{hub}}\dot{\theta}^2 + \int_0^L \rho\left(\dot{y} + (x+r)\dot{\theta}\right)^2 \mathrm{d}x - \int_0^L EI\left(y''\right)^2 \mathrm{d}x \\ & + m_{\text{tip}}\left((r+L)\dot{\theta} + \dot{y}(L)\right)^2 + I_{\text{tip}}\left(\dot{\theta} + \dot{y}'(L)\right)^2 \end{aligned} \tag{3.14}$$

where from Eq. (3.6) and Eq. (3.7) we have, $L_D = \frac{1}{2}I_{\text{hub}}\dot{\theta}^2$, $\hat{L} = \rho\left(\dot{y} + (x+r)\dot{\theta}\right)^2 - EI\left(y''\right)^2$, $L_B = m_{\text{tip}}\left((r+L)\dot{\theta} + \dot{y}(L)\right)^2 + I_{\text{tip}}\left(\dot{\theta} + \dot{y}'(L)\right)^2$ and $\mathcal{L}_B = L_B + \int_0^L \hat{L}\mathrm{d}x$. The equations of motion and the boundary conditions are then derived from Eq. (3.6) through Eq. (3.9) as,

$$\begin{aligned} & I_{\text{hub}}\ddot{\theta} + 2\int_0^L \rho(x+r)\left(\ddot{y} + (x+r)\ddot{\theta}\right)\mathrm{d}x \\ & \qquad + 2m_{\text{tip}}(L+r)\left((L+r)\ddot{\theta} + \ddot{y}(L)\right) + 2I_{\text{tip}}\left(\ddot{\theta} + \ddot{y}'(L)\right) = u \\ & \rho\left(\ddot{y} + (x+r)\ddot{\theta}\right) + EIy^{IV} = 0 \end{aligned} \tag{3.15}$$

$$
\begin{aligned}
&\text{at}\, x = 0 : y = 0, \quad y' = 0 \\
&\text{at}\, x = L : EI \frac{\partial^3 y}{\partial x^3}\bigg|_L = m_{\text{tip}} \left((L + r)\ddot{\theta} + \ddot{y}(L) \right), \\
&\qquad\quad EI \frac{\partial^2 y}{\partial x^2}\bigg|_L = -I_{\text{tip}} \left(\ddot{\theta} + \ddot{y}'(L) \right)
\end{aligned}
\tag{3.16}
$$

It is noted that by setting $m_{\text{tip}} = I_{\text{tip}} = 0$ a simpler no-tip-mass model dynamics and boundary conditions are obtained as,

$$
\begin{aligned}
I_{\text{hub}}\ddot{\theta} + 2\int_0^L \rho(x + r)\left(\ddot{y} + (x + r)\ddot{\theta} \right) \mathrm{d}x = u \\
\rho \left(\ddot{y} + (x + r)\ddot{\theta} \right) + EIy^{IV} = 0
\end{aligned}
\tag{3.17}
$$

$$
\begin{aligned}
&\text{at}\, x = 0 : y = 0, \quad y' = 0 \\
&\text{at}\, x = L : EI \frac{\partial^3 y}{\partial x^3}\bigg|_L = 0, \quad EI \frac{\partial^2 y}{\partial x^2}\bigg|_L = 0
\end{aligned}
\tag{3.18}
$$

3.1.2 The Generalized State Space Model

A generalized state space (GSS) model is developed by taking the Laplace transform, Eq. (3.19), of Eq. (3.15) and performing integration by parts to remove the spatial dependency from the integral, [66, 67]:

$$
F(s) = \int_0^\infty e^{-st} f(t)\mathrm{d}t \tag{3.19}
$$

$$
\begin{aligned}
s^2 J\bar{\theta} + 2s^2\rho \int_0^L (r + x)\bar{y}\mathrm{d}x + 2s^2 m_{\text{tip}}(r + L)\bar{y}(L) + 2s^2 I_{\text{tip}}\bar{y}'(L) = \bar{u} \\
s^2\rho \left(\bar{y} + (x + r)\bar{\theta} \right) + EI\bar{y}^{IV} = 0
\end{aligned}
\tag{3.20}
$$

where J is the total inertia of the model and is given by$J \equiv I_{\text{hub}} + 2m_{\text{tip}}(r+L)^2 + 2I_{\text{tip}} + 2\int_0^L \rho(r + x)^2\mathrm{d}x$. Integration by parts is then utilized to decouple the deformation parameter $\bar{y}$ from the spatial variable x such that,

$$\int_0^L (r+x)\bar{y}\mathrm{d}x = (r+x)\int_0^L \bar{y}\mathrm{d}x - \iint \bar{y}\mathrm{d}x\mathrm{d}x' \tag{3.21}$$

Plugging Eq. (3.21) into Eq. (3.20) yields the generalized integral equation,

$$\begin{aligned} s^2J\bar{\theta} + 2s^2\rho\left((x+r)\int_0^L \bar{y}\mathrm{d}x - \iint \bar{y}\mathrm{d}x\mathrm{d}x'\right) + 2s^2m_{\mathrm{tip}}(r+L)\bar{y}(L) + 2s^2I_{\mathrm{tip}}\bar{y}'(L) &= \bar{u} \\ \frac{s^2\rho}{EI}\left(\bar{y} + (x+r)\bar{\theta}\right) + \bar{y}^{IV} &= 0 \end{aligned} \tag{3.22}$$

Similar to the equations of motion, the boundary conditions are expressed in the Laplace/frequency domain as,

$$\begin{aligned} &\text{at}\, x = 0 : \bar{y} = 0, \quad \bar{y}' = 0 \\ &\text{at}\, x = L : \bar{y}''' = \frac{s^2m_{\mathrm{tip}}}{EI}\left[(r+L)\bar{\theta} + \bar{y}(L)\right], \quad \bar{y}'' = -\frac{s^2I_{\mathrm{tip}}}{EI}\left[\bar{y}'(L) + \bar{\theta}\right] \end{aligned} \tag{3.23}$$

which leads to the definition of the state space system,

$$\begin{aligned} z_1 &= \textstyle\iint \bar{y}\mathrm{d}x\mathrm{d}x' & z_1' &= z_2 \\ z_2 &= \textstyle\int \bar{y}\mathrm{d}x & z_2' &= z_3 \\ z_3 &= \bar{y} & z_3' &= z_4 \\ z_4 &= \bar{y}' & z_4' &= z_5 \\ z_5 &= \bar{y}'' & z_5' &= z_6 \\ z_6 &= \bar{y}''' & z_6' &= -\beta\left(z_3 + (r+x)\bar{\theta}\right) \end{aligned} \qquad \text{where, } \beta \equiv \frac{s^2\rho}{EI} \tag{3.24}$$

In state space representation Eq. (3.24) can be simply expressed as,

$$\begin{Bmatrix} z_1' \\ z_2' \\ z_3' \\ z_4' \\ z_5' \\ z_6' \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\beta & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\beta(r+x)\bar{\theta} \end{Bmatrix} \tag{3.25}$$

or in the more compact form,

$$\{Z'\} = A\{Z\} + \{b\} \tag{3.26}$$

The state space is generalized in the sense that the states consist of a distributed parameter variable, spatial partial derivatives, and first and second order integrals, which mix solutions at points in the flexible body domain with global response variables. The generalized state space model system of equations is solved by first developing the homogeneous and the forced solutions for the linear state space model in Eq. (3.26),

$$\{Z(x)\} = \underbrace{\exp[Ax]\{Z(0)\}}_{Z_H} + \underbrace{\int_0^x \exp\left[A(x-\tau)\right]\{b(\tau)\}\,\mathrm{d}\tau}_{Z_F} \tag{3.27}$$

where the homogeneous solution Z_H is given by,

$$\{Z_H\} = \left\{ \begin{array}{c} (1-f)z_5/\beta - (\beta x + f''')z_6/\beta^2 \\ -f'z_5/\beta + (1-f)z_6/\beta \\ -f''z_5/\beta - f'z_6/\beta \\ -f'''z_5/\beta - f''z_6/\beta \\ fz_5 - f'''z_6/\beta \\ f'z_5 + fz_6 \end{array} \right\} \tag{3.28}$$

and the forced part is evaluated from,

$$\begin{aligned} \{Z_F\} &= \int_0^x \exp\left[A(x-\tau)\right]\{b(\tau)\}\,\mathrm{d}\tau \\ &= -\beta\bar{\theta}\int_0^x (r+\tau) \left\{ \begin{array}{c} \left(\beta(x-\tau) + f'''(x-\tau)\right)/\beta^2 \\ \left(1 - f(x-\tau)\right)/\beta \\ -f'(x-\tau)/\beta \\ -f''(x-\tau)/\beta \\ -f'''(x-\tau)/\beta \\ f(x-\tau) \end{array} \right\} \mathrm{d}\tau \end{aligned} \tag{3.29}$$

The function f that represents the elements of the solution is derived from the matrix exponential solution of the flexible appendage sub-problem and is given by, [66, 67]

$$f(x) = \cos\left(\frac{\beta^{1/4}x}{\sqrt{2}}\right)\cosh\left(\frac{\beta^{1/4}x}{\sqrt{2}}\right) \tag{3.30}$$

Observing that the function f represents the real part of the complex function

$$f = Re\left\{\cos(\sigma x)\right\}, \quad \text{where, } \sigma \equiv \sqrt{i\sqrt{\beta}} \tag{3.31}$$

the homogeneous solution in Eq. (3.28) follows,

$$\{Z_H\} = \left\{ \begin{array}{c} (1-\cos(\sigma x))z_5/\beta + (\beta x + \sigma^3 \sin(\sigma x))z_6/\beta^2 \\ \sigma \sin(\sigma x) z_5/\beta + (1-\cos(\sigma x))z_6/\beta \\ \sigma^2 \cos(\sigma x) z_5/\beta + \sigma \sin(\sigma x) z_6/\beta \\ -\sigma^3 \sin(\sigma x) z_5/\beta + \sigma^2 \cos(\sigma x) z_6/\beta \\ \cos(\sigma x) z_5 - \sigma^3 \sin(\sigma x) z_6/\beta \\ -\sigma \sin(\sigma x) z_5 + \cos(\sigma x) z_6 \end{array} \right\} \tag{3.32}$$

Similarly, the forced part of the solution, Eq. (3.29), is obtained as,

$$\begin{aligned} \{Z_F\} &= \left\{ \begin{array}{c} \frac{1}{6\beta}\left(-3\beta r x^2 - \beta x^3 + 6\sigma^2 r \cos(\sigma x) + 6\sigma \sin(\sigma x) - 6\sigma^2 r - 6\sigma^2 x\right)\bar{\theta} \\ \frac{1}{2\sigma^2}\left(-2\sigma^2 r x - \sigma^2 x^2 + 2\sigma r \sin(\sigma x) - 2\cos(\sigma x) + 2\right)\bar{\theta} \\ \frac{1}{\sigma}\left(\sigma r \cos(\sigma x) + \sin(\sigma x) - \sigma r - \sigma x\right)\bar{\theta} \\ \left(-\sigma r \sin(\sigma x) + \cos(\sigma x) - 1\right)\bar{\theta} \\ -\sigma\left(\sigma r \cos(\sigma x) + \sin(\sigma x) - \sigma r - \sigma x\right)\bar{\theta} \\ -\frac{1}{\sigma^2}\left(\beta\left(\sigma r \sin(\sigma x) - \cos(\sigma x) + 1\right)\right)\bar{\theta} \end{array} \right\} \\ &= \left\{ \begin{array}{c} I_1(x) \\ I_1(x) \\ I_3(x) \\ I_4(x) \\ I_5(x) \\ I_6(x) \end{array} \right\} \bar{\theta} \end{aligned} \tag{3.33}$$

Eq. (3.32) and Eq. (3.33) are combined to produce the full GSS solution as a function of the GSS variables z_5 and z_6.

$$\{Z(x)\} = \left\{ \begin{array}{c} (1-\cos(\sigma x))z_5/\beta + (\beta x + \sigma^3 \sin(\sigma x))z_6/\beta^2 + I_1(x)\bar{\theta} \\ \sigma \sin(\sigma x)z_5/\beta + (1-\cos(\sigma x))z_6/\beta + I_2(x)\bar{\theta} \\ \sigma^2 \cos(\sigma x)z_5/\beta + \sigma \sin(\sigma x)z_6/\beta + I_3(x)\bar{\theta} \\ -\sigma^3 \sin(\sigma x)z_5/\beta + \sigma^2 \cos(\sigma x)z_6/\beta + I_4(x)\bar{\theta} \\ \cos(\sigma x)z_5 - \sigma^3 \sin(\sigma x)z_6/\beta + I_5(x)\bar{\theta} \\ -\sigma \sin(\sigma x)z_5 + \cos(\sigma x)z_6 + I_6(x)\bar{\theta} \end{array} \right\} \tag{3.34}$$

The expressions obtained can then be expressed in the general compact form as,

$$\{Z(x)\} = \left\{ \begin{array}{c} g_1(x) \\ g_2(x) \\ g_3(x) \\ g_4(x) \\ g_5(x) \\ g_6(x) \end{array} \right\} \bar{\theta} \tag{3.35}$$

The solution of the GSS model in Eq. (3.34) is obviously invariant to many aspects of the modeling assumptions, and holds for all infinity of model parameters (e.g., EI, ρ, L, r), and clearly admits a variety of boundary conditions. By applying the specific model boundary conditions and solving for the unknown GSS variables, z_5, z_6, the solution is complete in terms of the known system parameters. This makes the GSS solution capable of handling a wide range of distributed parameters problems as the need to apply the model specific boundary conditions arises at the last step of the solution.

By setting the inertia and the tip mass to 0 and by applying the model dependent boundary conditions for the no-tip-mass model, Eq. (3.36), z_5, z_6 are completely solved for as shwon in Eq. (3.37)

$$\begin{aligned} &\text{at}\, x = 0 : \{Z\} = \begin{bmatrix} 0 & 0 & 0 & 0 & z_5 & z_6 \end{bmatrix}^{\mathrm{T}} \\ &\text{at}\, x = L : z_5(L) = 0, \quad z_6(L) = 0 \end{aligned} \tag{3.36}$$

$$\begin{Bmatrix} z_5 \\ z_6 \end{Bmatrix} = \frac{1}{\sigma^4 \sin(\sigma L)/\beta - \cos(\sigma L)^2} \begin{bmatrix} \cos(\sigma L) & \sigma^3 \sin(\sigma L)/\beta \\ \sigma \sin(\sigma L) & \cos(\sigma L) \end{bmatrix} \begin{Bmatrix} I_5(L) \\ I_6(L) \end{Bmatrix} \bar{\theta} \tag{3.37}$$

For the no-tip-mass model, from Eq. (3.17) the rotation angle of the rigid hub is associated with the control torque by the transfer function,

$$\begin{aligned} &s^2 \left[J_1 + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right] \bar{\theta} = \bar{u} \\ &\bar{\theta} = \frac{\bar{u}}{s^2 \left[J_1 + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]} \end{aligned} \tag{3.38}$$

and from the GSS model, Eq. (3.24), the beam deformation is given by,

$$\bar{y} = g_3(x)\bar{\theta} = \frac{g_3(x)}{s^2 \left[J_1 + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]} \bar{u} \tag{3.39}$$

where, $J_1 \equiv I_{\text{hub}} + 2\int_0^L \rho(r+x)^2 \mathrm{d}x$ is the total inertia for the model.

For the general model with tip mass shown in Fig. 3.1, the boundary conditions are expressed in terms of the GSS variables as,

$$\begin{aligned}
&\text{at}\, x = 0 : \{Z\} = \begin{bmatrix} 0 & 0 & 0 & 0 & z_5 & z_6 \end{bmatrix}^{\mathrm{T}} \\
&\text{at}\, x = L : z_5(L) = -\alpha\left[z_4(L) + \bar{\theta}\right], \quad z_6(L) = \gamma\left[z_3(L) + (r+L)\bar{\theta}\right] \\
&\text{where, } \alpha \equiv \frac{s^2 I_{\text{tip}}}{EI} \quad \text{and} \quad \gamma \equiv \frac{s^2 m_{\text{tip}}}{EI}
\end{aligned} \tag{3.40}$$

where the unknown z_5, z_6 are obtained from,

$$\begin{aligned}
\begin{Bmatrix} z_5 \\ z_6 \end{Bmatrix} = \frac{1}{\sigma^4 \sin(\sigma L)^2/\beta - \cos(\sigma L)^2} &\begin{bmatrix} -\alpha\cos(\sigma L) & \gamma\sigma^3\sin(\sigma L)/\beta \\ -\alpha\sigma\sin(\sigma L) & \gamma\cos(\sigma L) \end{bmatrix} \begin{Bmatrix} z_4(L) \\ z_3(L) \end{Bmatrix} \\
+ \begin{bmatrix} \sigma^3\left(\gamma(r+L) - I_6(L)\right)/\beta & (-\alpha - I_5(L)) \\ \sigma\left(-\alpha - I_5(L)\right) & \gamma(r+L) - I_6(L) \end{bmatrix} &\begin{Bmatrix} \sin(\sigma L) \\ \cos(\sigma L) \end{Bmatrix} \bar{\theta}
\end{aligned}$$

$$\begin{Bmatrix} z_4(L) \\ z_3(L) \end{Bmatrix} = \frac{1}{\beta}\begin{bmatrix} -\sigma^3\sin(\sigma L) & \sigma^2\cos(\sigma L) \\ \sigma^2\cos(\sigma L) & \sigma\sin(\sigma L) \end{bmatrix} \begin{Bmatrix} z_5 \\ z_6 \end{Bmatrix} + \begin{Bmatrix} I_4(L) \\ I_3(L) \end{Bmatrix} \bar{\theta} \tag{3.41}$$

The beam deformation $\bar{y}$ is represented as a function of the input torque $\bar{u}$ as,

$$\begin{aligned}
&s^2\left[J_2 + 2m_{\text{tip}}(r+L)g_3(L) + 2I_{\text{tip}}g_4(L) + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]\bar{\theta} = \bar{u} \\
&\bar{\theta} = \frac{\bar{u}}{s^2\left[J_2 + 2m_{\text{tip}}(r+L)g_3(L) + 2I_{\text{tip}}g_4(L) + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]} \\
&\bar{y} = \frac{g_3(x)}{s^2\left[J_2 + 2m_{\text{tip}}(r+L)g_3(L) + 2I_{\text{tip}}g_4(L) + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]}\bar{u}
\end{aligned} \tag{3.42}$$

where the total inertia in that case is $J_2 \equiv I_{\text{hub}} + 2m_{\text{tip}}(r+L)^2 + 2I_{\text{tip}} + 2\int_0^L \rho(r+x)^2 dx$. The analytic transfer functions obtained in Eq. (3.38), Eq. (3.39) and Eq. (3.42) are utilized to accurately obtain the frequency response of the hybrid system and in control of the flexible modes when performing large angle maneuvers.

3.1.3 Frequency Response Numerical Results

From Eq. (3.39) and Eq. (3.42), the transfer function of the no-tip-mass model, $G_1(s,x)$, and the tip-mass model $G_2(s,x)$ are expressed as,

$$\begin{aligned} G_1(s,x) &= \frac{g_3(x)}{s^2\left[J_1 + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]} \\ G_2(s,x) &= \frac{g_3(x)}{s^2\left[J_2 + 2m_{\text{tip}}(r+L)g_3(L) + 2I_{\text{tip}}g_4(L) + 2\rho\left((r+x)g_2(x) - g_1(x)\right)\right]} \end{aligned} \tag{3.43}$$

For the purpose of numerical comparison the classical assumed modes solution, [58], is utilized. The method assumes a decoupled spatial and time dependent beam response expressed with the series,

$$y(x,t) = \sum_{i=1}^{N} q_i(t)\phi_i(x) \tag{3.44}$$

The spatial function $\phi_i(x)$ describes the i-th spatial assumed mode shape function of the flexible structure and is designed to meet the physical and the geometrical boundary conditions of the beam. A widely used admissible function that satisfies the boundary conditions is given in [58, 62] as,

$$\begin{aligned} &\phi_i(x) = 1 - \frac{\cos(i\pi x)}{L} + \frac{1}{2}(-1)^{i+1}\left(\frac{i\pi x}{L}\right)^2 \\ &\text{where, } 0 \leq x \leq L \end{aligned} \tag{3.45}$$

Using Eq. (3.44) and Eq. (3.45) with Eq. (3.12) and Eq. (3.13) and following the Lagrangian approach,

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\boldsymbol{x}}}\right) - \frac{\partial T}{\partial \boldsymbol{x}} + \frac{\partial V}{\partial \boldsymbol{x}} = \boldsymbol{F} \tag{3.46}$$

the system of equations of motion for the tip mass model is represented in the matrix form,

$$\begin{bmatrix} J_2 & M_{\theta q}^{\mathrm{T}} \\ M_{\theta q} & M_{qq} \end{bmatrix} \ddot{x} + \begin{bmatrix} 0 & 0 \\ 0 & K_{qq} \end{bmatrix} \boldsymbol{x} = \begin{Bmatrix} u \\ 0 \end{Bmatrix} \tag{3.47}$$

where the elements of the mass and the stiffness matrices are defined as,

$$\begin{aligned} J_2 &= I_{\text{hub}} + 2m_{\text{tip}}(r+L)^2 + 2I_{\text{tip}} + 2\int_0^L \rho(r+x)^2 \mathrm{d}x \\ [M_{\theta q}]_i &= 2\left[\rho \int_0^L (r+x)\phi_i(x)\mathrm{d}x + m_{\text{tip}}(r+L)\phi_i(L) + I_{\text{tip}}\phi_i'(L)\right] \\ [K_{qq}]_{ij} &= 2\int_0^L \phi_i''(x)\phi_j''(x)\mathrm{d}x \end{aligned} \tag{3.48}$$

In order to obtain the no-tip-mass model equations, one can set $m_{\text{tip}} = I_{\text{tip}} = 0$ in Eq. (3.48) to obtain the mass and stiffness matrices elements for the model. A comparison between the frequency response of the analytical GSS and the numerical assumed modes method, assuming 10 modes, is presented in Fig. 3.3 and Fig. 3.4 for the no-tip-mass model at $x = 2$ ft. and $x = 4$ ft., respectively. The set of parameters values used in this comparison are extracted from the physical model presented in [58], and are shown in Table 3.1. Similar results are obtained for the tip-mass model and are shown in Fig. 3.5 and Fig. 3.6

It is shown that the numerical results obtained from the analytic transfer functions are in agreement with the classical numerical solutions. It is also noted that in Fig. 3.3 and Fig. 3.5 a slight truncation error can be observed in the numerical solution.

In this case, this assumed model has previously been tuned to match experimental results, so it is not a huge surprise that the distributed parameter model, with zero truncation error, was in good agreement with the known to be reasonably well

Table 3.1: System Parameters Values

Parameter	**Value**
I_{hub}	8 slug-ft^2
ρ	0.0271875 slug/ft
E	0.1584×10^{10} lb/ft^2
L	4 ft
r	1 ft
I	0.47095×10^{-7} ft^4
m	0.1569 slug
I_{tip}	0.0018 slug-ft^2

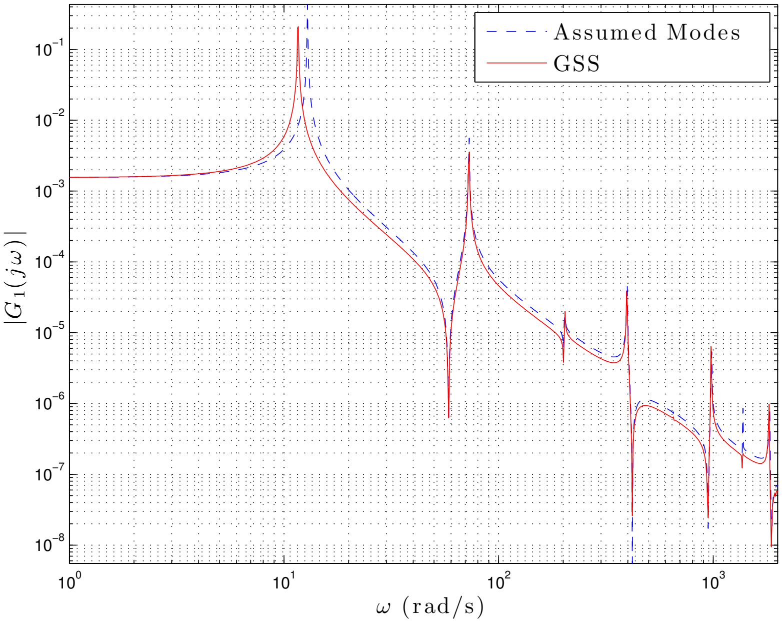

Figure 3.3: Frequency Response Comparison No Tip Mass Model at $x = 2$ ft, [76]

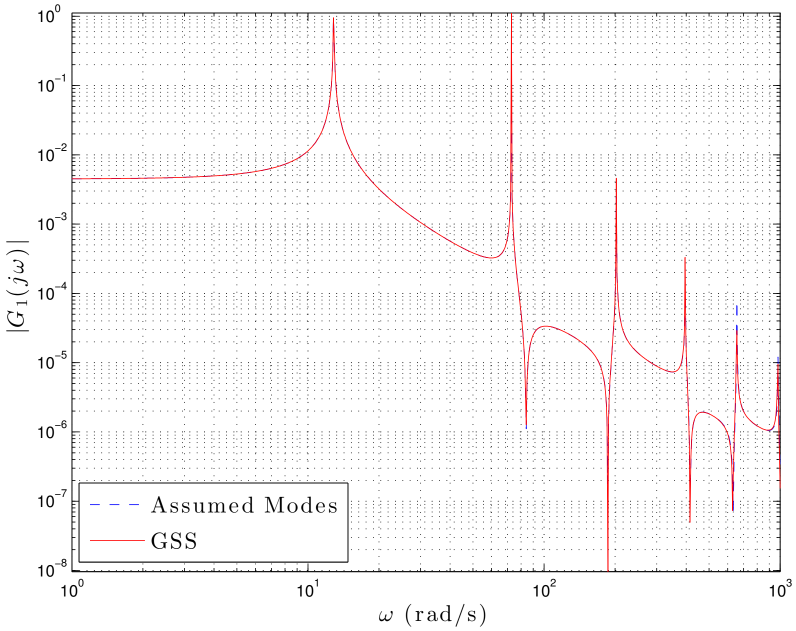

Figure 3.4: Frequency Response Comparison No Tip Mass Model at $x = 4$ ft, [76]

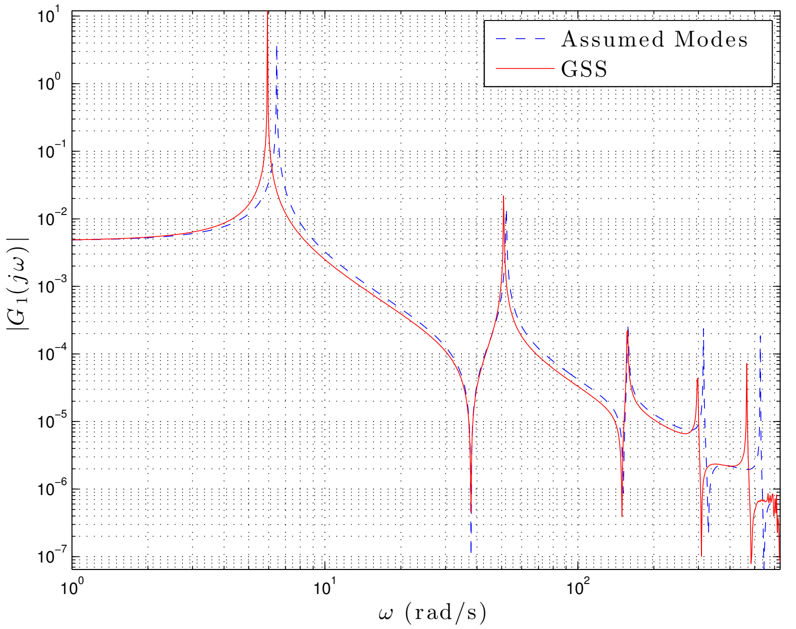

Figure 3.5: Frequency Response Comparison Tip Mass Model at $x = 2$ ft, [76]

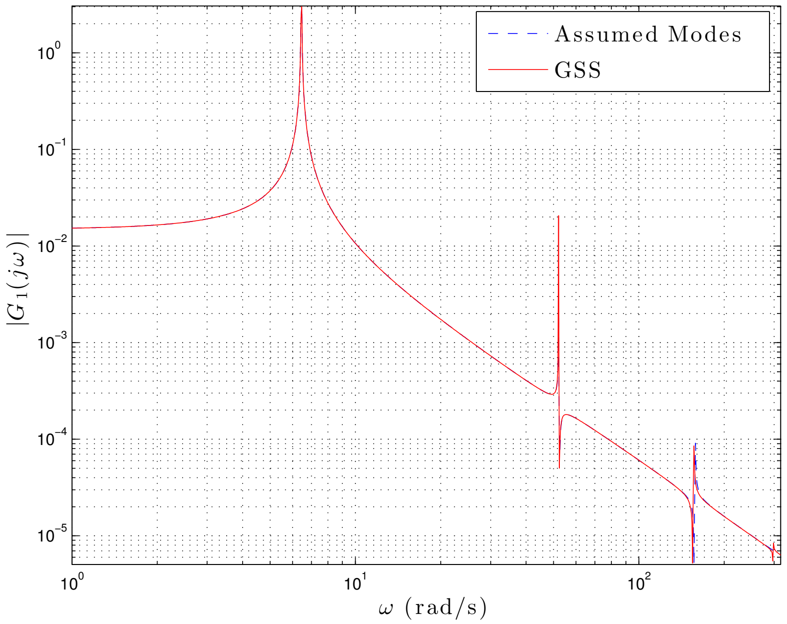

Figure 3.6: Frequency Response Comparison Tip Mass Model at $x = 4$ ft, [76]

converged discretized model. More generally however, it is important to note that the distributed parameter approach affords a rigorous means to address the issue of whether or not the discrete model has unacceptable truncation errors, and to design controllers, as shown next, which are free of truncation errors.

3.1.4 The Control Problem

Building on the analytical solution obtained for the more general tip mass model, the control problem is analyzed. To gain some insight on the system behavior, the unit step input Bode plots are generated at the midpoint of the appendage $x = 2$ ft. for both the responses of the rigid body, $\bar{\theta}(j\omega)$, and the flexible appendage, $\bar{y}(j\omega)$, Fig. 3.7 and Fig. 3.8 respectively. The resonant behavior of the system previously obtained from the GSS tranfer function, Eq. (3.43), is clearly present in this analysis with the phase angle shifting between $+90°$ and $-90°$ at those frequencies. For

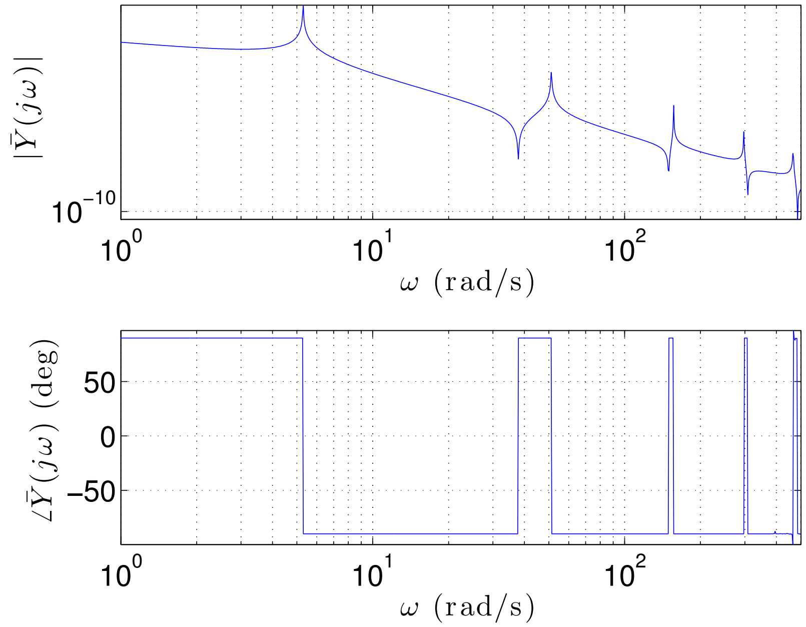

Figure 3.7: Bode Plot $\bar{Y}$ at $x = 2$ ft, [76]

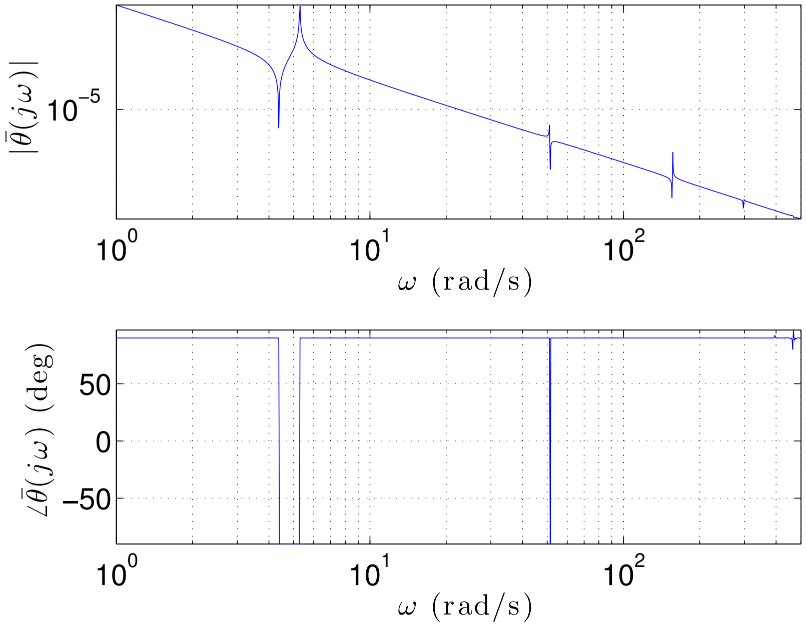

Figure 3.8: Bode Plot $\bar{\theta}$ at $x = 4$ ft, [76]

further insights, the assumed modes method is used to generate the model time response for a unit step input for $\theta(t)$, $\dot{\theta(t)}$, Fig. 3.9 and Fig. 3.10, and for $y(x,t)$, $\dot{y}(x,t)$, Fig. 3.11 and Fig. 3.12. A case study is constructed for the more general tip

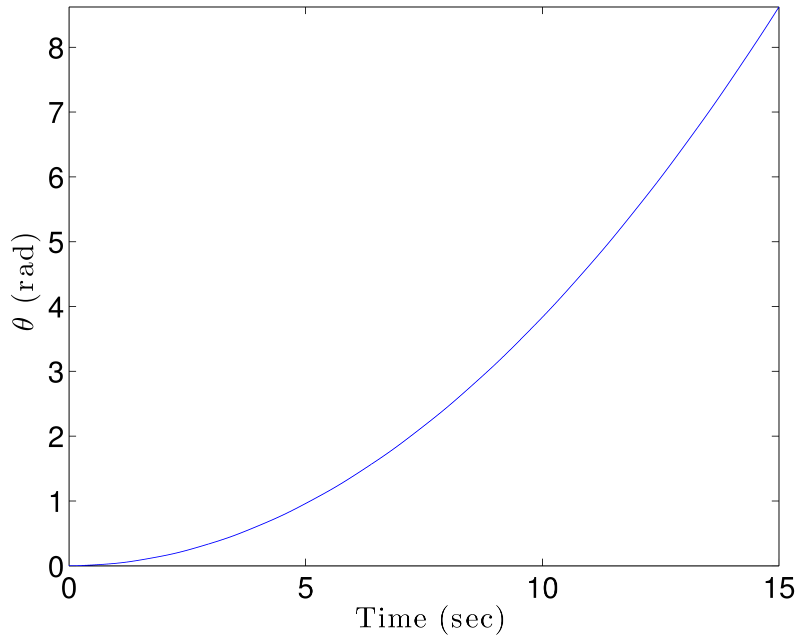

Figure 3.9: Step Input Response $\theta(t)$, [76]

mass model applying a Lyapunov stable controller, [58]. First Eq. (3.15) is rewritten as,

$$\begin{aligned}
&I_{\text{hub}}\ddot{\theta} = u + (M_0 - rS_0) \\
&-(M_0 - rS_0) = \int_0^L \rho(x+r)\left(\ddot{y} + (x+r)\ddot{\theta}\right)\mathrm{d}x + m_{\text{tip}}(L+r)\left((L+r)\ddot{\theta} + \ddot{y}(L)\right) \\
&\rho\left(\ddot{y} + (x+r)\ddot{\theta}\right) + EIy^{IV} = 0
\end{aligned} \tag{3.49}$$

where, (M_0, S_0) represent the bending moment and shear force at the root of the beam. The effect of the tip mass inertia is left out for simplification and can be con-

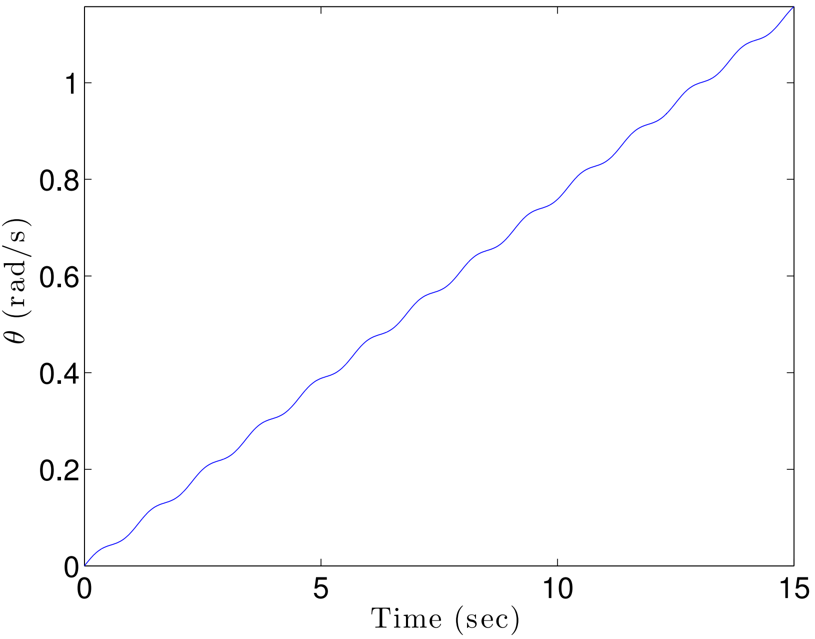

Figure 3.10: Step Input Response $\dot{\theta}(t)$, [76]

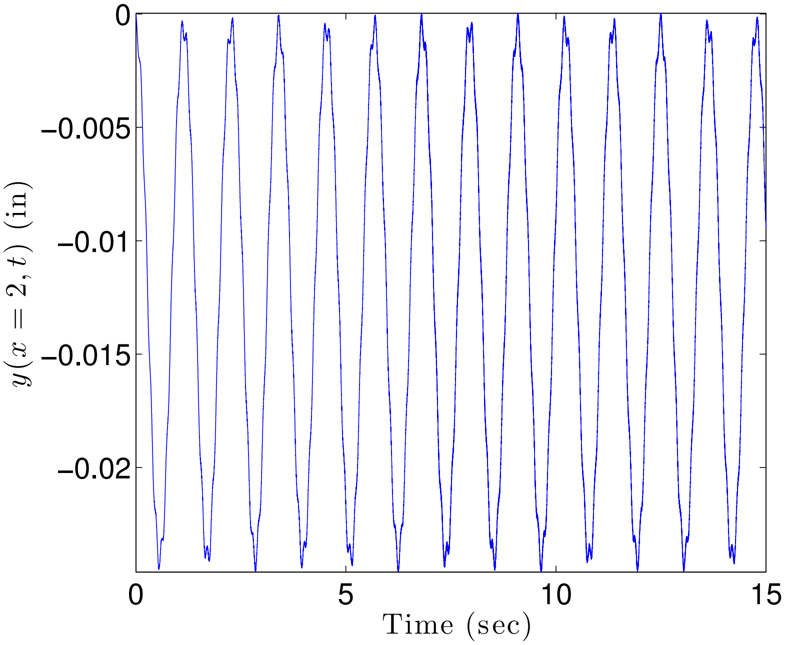

Figure 3.11: Step Input Response $y(x = 2, t)$, [76]

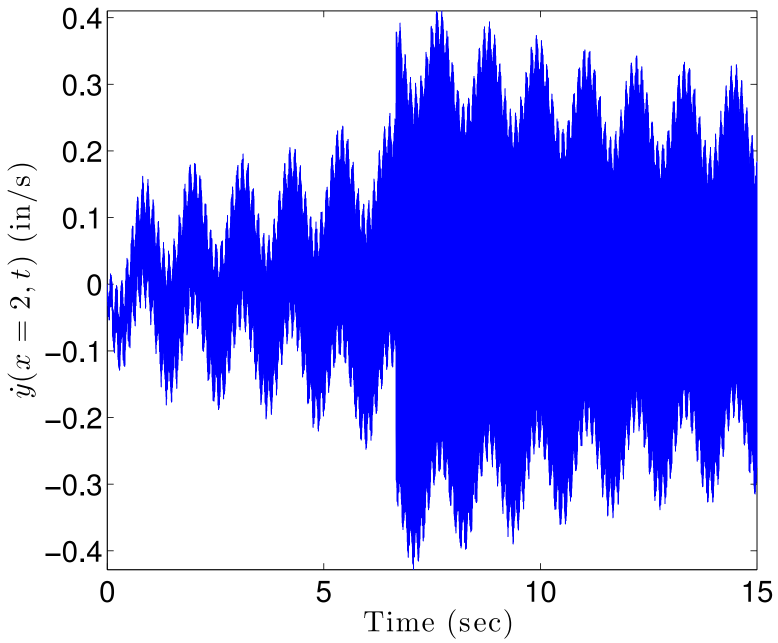

Figure 3.12: Step Input Response $\dot{y}(x = 2, t)$, [76]

sidered as a disturbance error or as part of the model uncertainty the controller needs to overcome. The set of boundary conditions in Eq. (3.16) can then be simplified as,

$$\begin{aligned} &\text{at}\, x = 0 : y = 0, \quad y' = 0 \\ &\text{at}\, x = L : \, EI\frac{\partial^3 y}{\partial x^3}\bigg|_L = m_{\text{tip}} \left((L + r)\ddot{\theta} + \ddot{y}(L) \right), \quad EI\frac{\partial^2 y}{\partial x^2}\bigg|_L = 0 \end{aligned} \tag{3.50}$$

We are interested in large angle maneuvers with a target final state given by,

$$\left[\theta, \dot{\theta}, y(x,t), \dot{y}(x,t)\right]_{\text{Target}} = [\theta_f, 0, 0, 0] \tag{3.51}$$

A weighted Lyapunov function is then given by,

$$\begin{aligned} 2V &= w_1 I_{\text{hub}}\dot{\theta}^2 + w_2\left(\theta - \theta_f\right)^2 \\ &+ w_3\left[\int_0^L \rho\left(\dot{y} + (x+r)\dot{\theta}\right)^2 \mathrm{d}x + m_{\text{tip}}\left((r+L)\dot{\theta} + \dot{y}(L)\right)^2 + \int_0^L EI\left(y''\right)^2 \mathrm{d}x\right] \end{aligned} \tag{3.52}$$

where an extra term that includes a penalty for the current state versus the target state, $(\theta - \theta_f)$ is added to achieve the required maneuver. By differentiating the Lyapunov function, Eq. (3.52) w.r.t. time and substituting the dynamics, Eq. (3.49), and the boundary conditions, Eq. (3.50), $\dot{V}$ can be expressed as,

$$\begin{aligned} \dot{V} &= w_1 u\dot{\theta} + w_2\left(\theta - \theta_f\right)\dot{\theta} \\ &+ \left(w_3 - w_1\right)\left[\int_0^L \rho(x+r)\left(\ddot{y} + (x+r)\ddot{\theta}\right)\mathrm{d}x + m_{\text{tip}}(L+r)\left((L+r)\ddot{\theta} + \ddot{y}(L)\right)\right]\dot{\theta} \\ &= \left[w_1 u + w_2\left(\theta - \theta_f\right) + \left(w_3 - w_1\right)\left(rS_0 - M_0\right)\right]\dot{\theta} \end{aligned} \tag{3.53}$$

In order to ensure stability, $\dot{V}$ should meet the condition $\dot{V} \leq 0$ and the control law is chosen as,

$$u = \frac{-1}{w_1}\left[w_2\left(\theta - \theta_f\right) + \left(w_3 - w_1\right)\left(rS_0 - M_0\right) + w_4\dot{\theta}\right] \tag{3.54}$$

By substituting Eq. (3.54) into Eq. (3.53), the negative semi-definite expression, $\dot{V} = -w_4\dot{\theta}^2 \leq 0$ is obtained. In order to simplify the gain choices associated with the control law, Eq. (3.54) can be re-written as,

$$u = -\left[k_1\left(\theta - \theta_f\right) + k_2\left(rS_0 - M_0\right) + k_3\dot{\theta}\right] \tag{3.55}$$

where, $k_1 \equiv \frac{w_2}{w_1} \geq 0$, $k_2 \equiv \frac{w_3 - w_1}{w_1} > -1$ and $k_3 \equiv \frac{w_4}{w_1} \geq 0$. The sign and value of k_2 will determine whether the beam vibration energy, $w_3 - w_1 > 0$, or the hub motion energy, $w_1 > w_3$ is dissipated. To investigate the frequency domain response applying the control law, the Laplace transformation of Eq. (3.55) is expressed as,

$$\begin{aligned}\bar{u} = &-k_1\left(\bar{\theta} - \frac{\theta_f}{s}\right) \\ &- k_2 s^2\left(\rho\int_0^L (x+r)\left(\bar{y} + (x+r)\bar{\theta}\right)\mathrm{d}x + m_{\text{tip}}(L+r)\left((L+r)\bar{\theta} + \bar{y}(L)\right)\right) \\ &- k_3 s\bar{\theta}\end{aligned} \tag{3.56}$$

Utilizing integration by parts the transformed control law is expressed in terms of the GSS state variables as,

$$\begin{aligned}\bar{u} = &\left[-k_1 - k_2\rho s^2\int_0^L (r+x)^2\mathrm{d}x - k_2 m_{\text{tip}} s^2 (L+r)^2 - k_3 s\right]\bar{\theta} \\ &+ k_1\frac{\theta_f}{s} - k_2\rho s^2\left((x+r)g_2(x) - g_1(x)\right) - k_2 m_{\text{tip}} s^2 (L+r) g_3(L)\end{aligned} \tag{3.57}$$

Substituting Eq. (3.57) into the transfer function, Eq. (3.42), and collecting variables produced the transfer function for the hub angle $\bar{\theta}$ as,

$$\begin{aligned}\bar{\theta} = &\left[k_1\frac{\theta_f}{s} - k_2\rho s^2\left((x+r)g_2(x) - g_1(x)\right) - k_2 m_{\text{tip}} s^2 (L+r) g_3(L)\right] / \\ &s^2\left[\frac{k_1}{s^2} + k_2\rho\int_0^L (r+x)^2\mathrm{d}x + k_2 m_{\text{tip}}(L+r)^2 + \frac{k_3}{s} + J_2\right. \\ &\left. + m_{\text{tip}}(r+L)g_3(L) + I_{\text{tip}} g_4(L) + \rho\left((r+x)g_2(x) - g_1(x)\right)\right]\end{aligned} \tag{3.58}$$

The deformation transfer function can then be expressed in terms of $\bar{\theta}$ as,

$$\bar{y} = g_3(x)\bar{\theta} \tag{3.59}$$

After some trial and error the controller gains are adjusted to obtain the most stable response. To illustrate the effect of gain changes on the system frequency response the gains are first set to $k_1 = 1, k_2 = 1, k_3 = 1$. Figure 3.13 and Fig. 3.14 show the amplitude and phase plots associated with the two transfer functions, Eq. (3.58) and Eq. (3.59), respectively, for unit gains. Clearly, the frequency response highlights

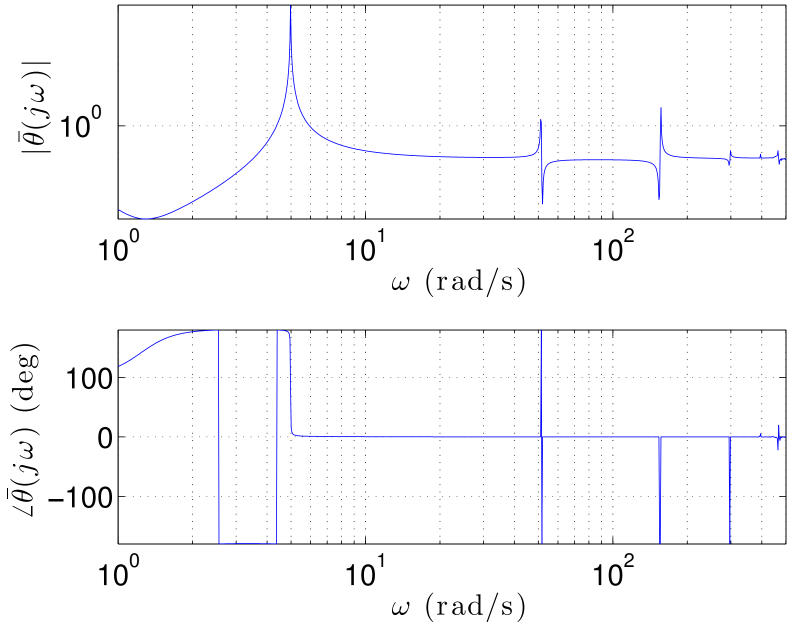

Figure 3.13: Bode Plot $\bar{\Theta}$, Unit Gains, [76]

potential system instability with order of magnitude gain amplifications and a −180° phase angle. The gains are then adjusted to $k_1 = 12, k_2 = 0, k_3 = 16$. The amplitude and phase plots associated with the new set of gains are shown in Fig. 3.15 and Fig. 3.16.

With no significant lags or high amplitude amplification, the chosen set of parameters can be suitable for a controller to drive the rigid hub to its target final angle while mitigating the vibrations effect of the flexible appendages. The time response plots for the system are shown for $\theta(t)$, $\dot{\theta}(t)$, in Fig. 3.17 and Fig. 3.18, and

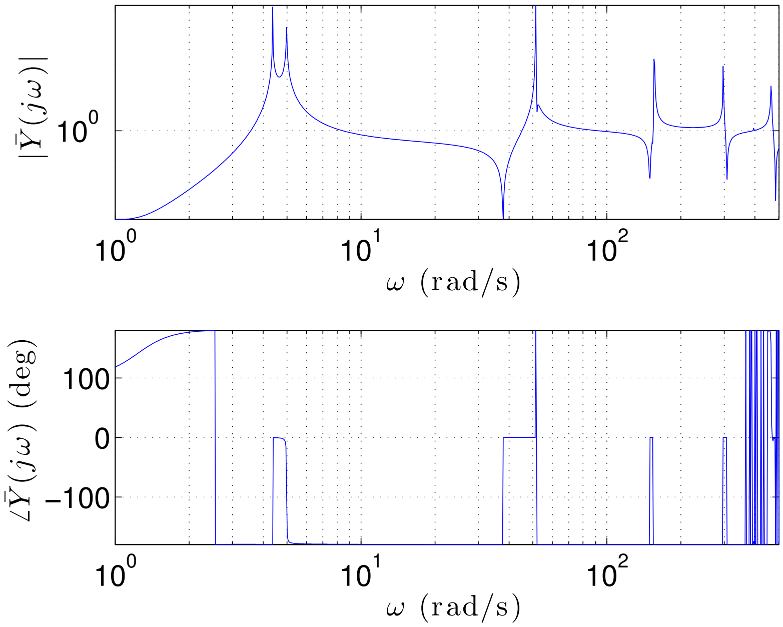

Figure 3.14: Bode Plot $\bar{Y}$ Unit Gains, [76]

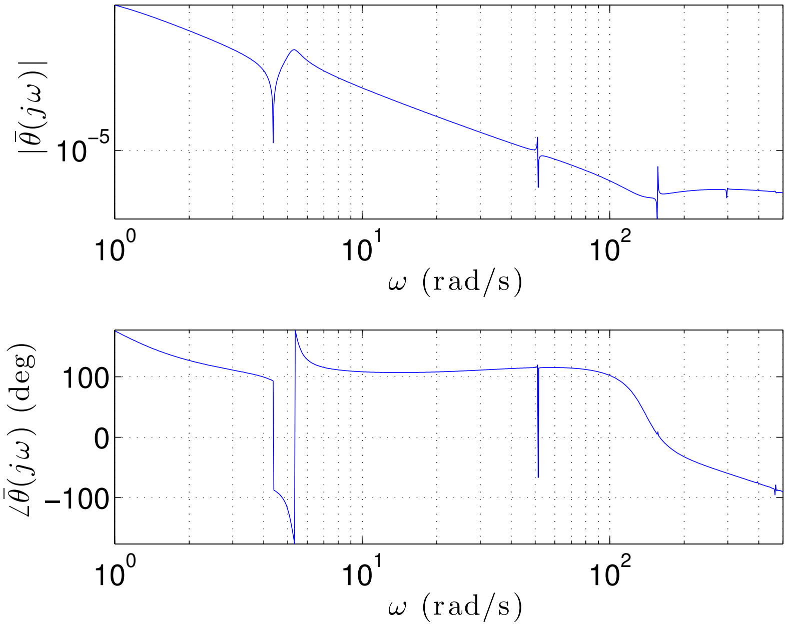

Figure 3.15: Bode Plot $\bar{\Theta}$, Designed Gains, [76]

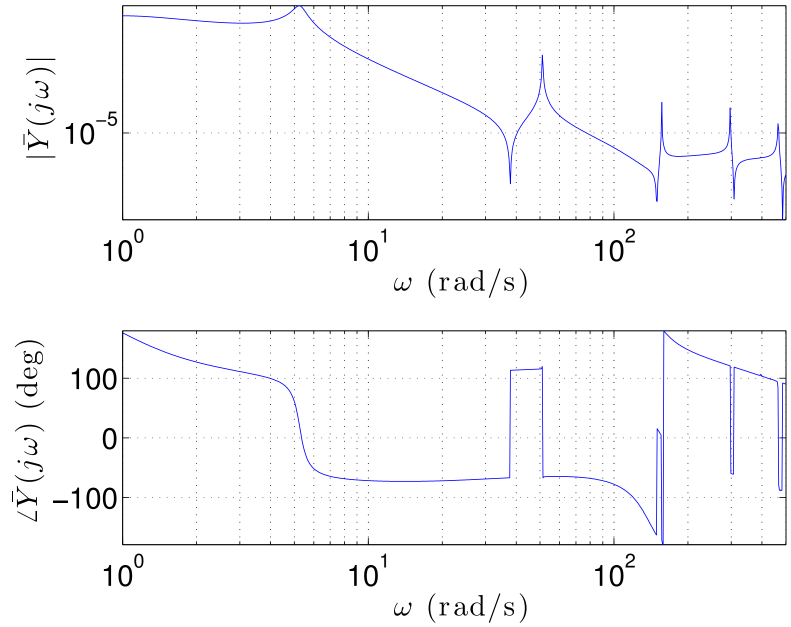

Figure 3.16: Bode Plot $\bar{Y}$ Designed Gains, [76]

for $y(x = 2, t)$, $\dot{y}(x = 2, t)$ in Fig. 3.19 and Fig. 3.20.

The results show achieving the target state for θ and $\dot{\theta}$ while reducing the vibrations in, $y(x, t)$ and $\dot{y}(x, t)$, It has to be noted that no controls are applied to the flexible appendage and the control torque is solely driving the hub while achieving acceptable results on the vibrations control. Several works discussed techniques of controlling the flexible structure by applying controls to the flexible appendages, [77,78].

The generalized state space approach provides analytic transfer functions for the system frequency response for both the tip-mass and the no-tip-mass models, without introducing spatial discretization. The fact that nontrivial problems can be solved by these methods, using distributed parameter models, does not appear to be widely appreciated. These special case models and control design methods serve important roles in evaluating the applicability and validity of the approximations implicit in

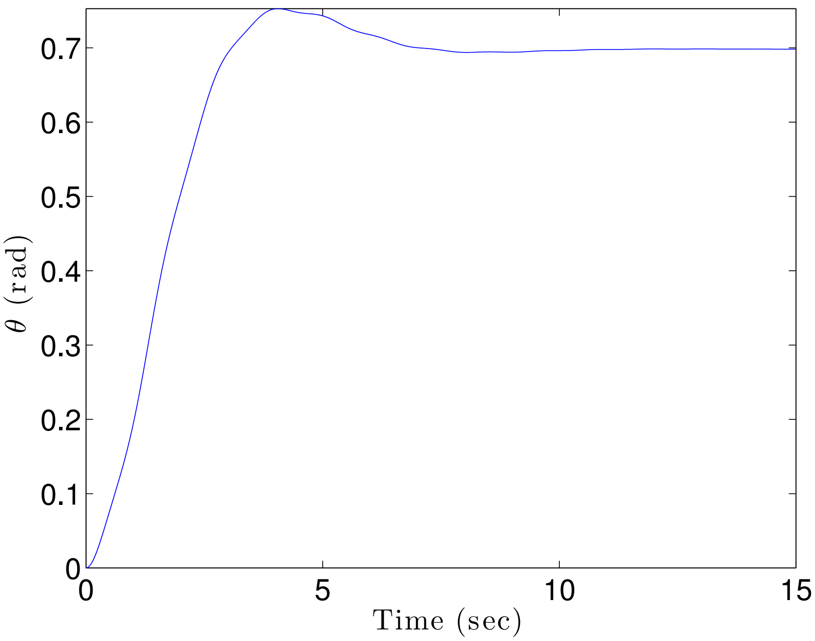

Figure 3.17: $\theta(t)$ Designed Gains, [76]

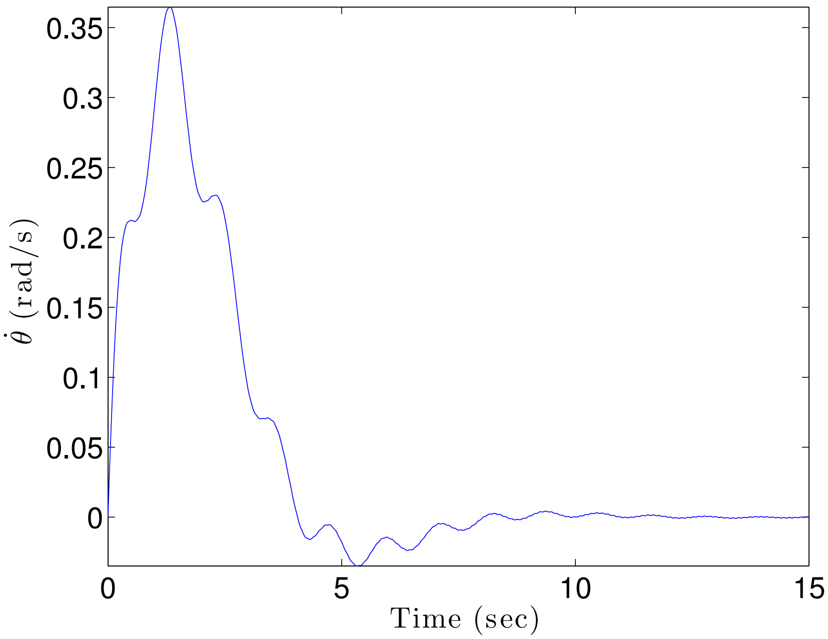

Figure 3.18: $\dot{\theta}(t)$ Designed Gains, [76]

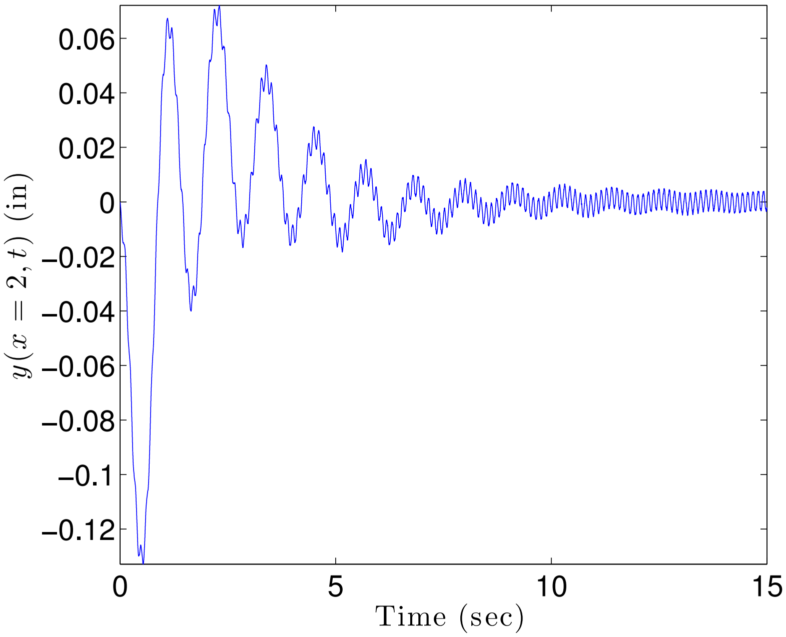

Figure 3.19: $y(x = 2, t)$ Designed Gains, [76]

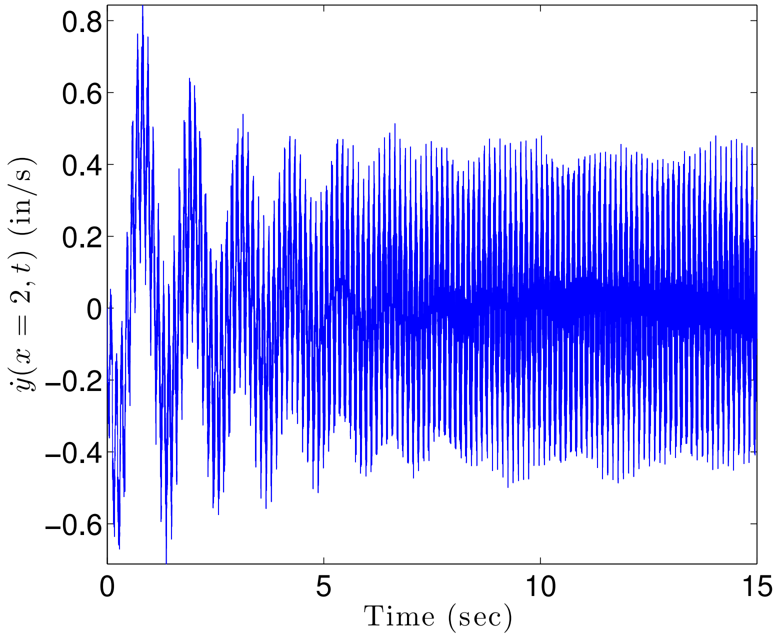

Figure 3.20: $\dot{y}(x = 2, t)$ Designed Gains, [76]

more generally applicable spatial discretization methods. Several other boundary conditions and constitutive assumptions can be applied and the analogous steps to those presented here can be followed in order to obtain the analytical distributed parameter solution. By utilizing the full transfer function solution provided by the GSS approach any control problem design in the frequency domain can be addressed. A case study is constructed for the gain selection of a Lyapunov stable control law. By looking at the frequency response and changing the gains an acceptable performance was achieved driving the structure from a stationary initial state to a target state while suppressing the beam vibrations. The GSS approach can be considered a platform through which distributed parameters models can be addressed. Most recently, the more complex model following the Timoshenko beam theory is addressed with possible extensions to a variety of boundary conditions and control problems, [79].

The presented control problem has potential for several extensions. Optimization was not considered in this work whereas several techniques exists for optimization in the frequency domain based on Parseval's theorem. The approach can also be extended to address the tracking problem where the control law can be designed to track a specified rigid-body motion while suppressing vibrations of the flexible appendage following the work presented in [70]. In general the GSS solution provides a general framework for any control scheme in the frequency domain. This is shown to be a very powerful tool. When it is possible to use discretization and truncation-free distributed parameter model transfer function solutions provided by the GSS approach, any control problem design in the frequency domain can be addressed rigorously. The set of distributed parameter systems for which analogous exact solutions for the response and closed loop frequencies can be established, provide important tools for validation of codes for more general applications where spatial discretization is required.

3.2 Analytic Power Series Solutions for the Two-Body Problem

Several methods exist for integrating the Keplerian motion of two bodies. By introducing two scalar Lagrange-like invariants, it is possible to integrate the two-body and two-body problem using a recursive formulation for developing an analytic continuation-based power series that overcomes the limitations of standard integration methods. Numerical comparisons with *RKN12(10)*, and other state of the art integration methods indicate significant performance improvements, while maintaining millimeter accuracy for the orbit predictions. The proposed mathematical models are expected to be broadly useful for celestial mechanic applications for optimization, uncertainty propagation, and nonlinear estimation theory, [80–82].

The equation of motion for the relative perturbed two-body problem is given by,

$$\ddot{r} = -\frac{\mu}{r^3}\boldsymbol{r} + \boldsymbol{a}_d \tag{3.60}$$

where, $\boldsymbol{r} = [x, y, z]^{\mathrm{T}}$ is the inertial relative position, μ the gravitational parameter, $r = \sqrt{\boldsymbol{r} \cdot \boldsymbol{r}}$ and $\boldsymbol{a}_d$ refers to the perturbation acceleration. For the unperturbed/classical two-body problem, $\boldsymbol{a}_d = \boldsymbol{0}$, Eq. (3.60) has an analytical solution extracted from the the conservation of angular momentum and the fundamental orbit integrals, [83, 84]. Similarly, the Lagrange/Gibbs, *F&G*, solution exploits the conservation of angular momentum to express the future state vector as a linear combination of the initial conditions, [83, 84]. The recursion of the equation produced by successive differentiation has also been exploited to produce a power series based solution with Lagrange Fundamental Invariants, [83].

Several numerical techniques exist to handle the solution of the nonlinear initial value problem (IVP) in Eq. (3.60). The Runge-Kutta, *RK*, family of methods can be considered as the most widely used explicit methods for numerical integration.

The classical or the 4th-order *RK* method is the most commonly used among various *RK* methods. Adaptive step-size 4th-order *RK* methods are developed and are known as the Runge-Kutta-Fehlberg, *RKF*, methods, [85]. Higher order adaptive *RK* methods are then developed for high accuracy requirement applications. Adaptive Runge-Kutta-Nyström, *RKN*, methods with order 8(7), 9(8), 10(9) and 11(10) are developed to solve general second-order ordinary differential equations, [86]. For orbit propagation problems, the Gauss-Jackson method is studied extensively and compared against other numerical techniques, e.g. *RK4*, *RKN* and Taylor series expansion, [87]. It is a predictor-corrector finite difference method designed specifically for solving second order differential equations, [88]. The *RKN12(10)* and *RKN8(6)* methods are introduced for general dynamical systems, [89]. The method is then compared against several Nyström methods and recursive power series solutions for orbit propagation problems, [90, 91]. Furthermore, the accuracy of several of the above mentioned numerical integrators are tested in solving different N celestial bodies problems such as, Sun, Jupiter, Saturn, Uranus, and Neptune and nine planets problems, [92]. Most of the comparisons performed in the literature addressed the issue of the integration step-size. The need for a high accuracy solution in many cases necessitates a smaller time-step. More recently, Modified Chebyshev-Picard Iteration (MCPI) method has been developed for orbit propagation and general initial value problems, [93,94]. The method combines orthogonal basis function, Chebyshev polynomials, with Picard iterations to solve the initial value problem. It is used in long-term orbit propagation problem and showed significant improvement over the *RKN12(10)* in terms of computational speed, [93]. Parallelization of MCPI is then explored and showed a substantial improvement in computational cost over *Matlab ode45* for several initial value problems including a near circular orbit for the classical two-body problem, [94].

In this study an analytic power series continuation algorithm is developed to address the solution of the unperturbed/classical two-body problem. The algorithm exploits the recursive nature of the two-body problem and introduces an accurate and efficient solution for various types of orbits as will be shown next.

3.2.1 Analytic Continuation for the Two-Body Problem

Building on the idea of Lagrange invariants and the recursive nature of the classical two-body problem, arbitrary order time derivatives are recursively computed. The recursive algorithm is made by introducing a nonlinear change of variables. Two scalar variables are defined, whose higher order time derivatives are recursively generated from applying Leibniz product rule. The first variable involves a quadratic measure of distance and Leibniz rule provides an explicit derivative calculation. The second variable is defined as a constraint equation involving the first variable. A first-order differential equation is developed for the constraint equation and manipulated to eliminate division operations. The resulting differential equation is differentiated using Leibniz product rule to provide an implicit differential equation for the highest time derivative of the second variable. The trajectory calculations are evaluated by introducing the second variable into the equation of motion to yield a quadratic product involving a position vector and the second variable, which can be differentiated by Leibniz product rule as an explicit equation. By linking the recursive calculations for the two scalar variables and the two-body acceleration, one obtains a recursive algorithm for generating arbitrary order time derivatives for the trajectory motion. After computing all the vector trajectory time derivatives, one easily generates Taylor expansions for the position and velocity.

The first scalar variable is defined as,

$$f = \boldsymbol{r} \cdot \boldsymbol{r} \tag{3.61}$$

From Leibniz rule, arbitrary order derivatives of Eq. (3.61) can be computed from,

$$f^{(n)} = \sum_{m=0}^{n} \left(\begin{array}{c} n \\ \\ m \end{array} \right) \boldsymbol{r}^{(m)} \cdot \boldsymbol{r}^{(n-m)} \tag{3.62}$$

where, the expression with the superscript $f^{(n)}$ denotes the n-th order time derivative of f and $\left(\begin{array}{c} n \\ \\ m \end{array} \right)$ is the binomial coefficient. The second scalar variable is then defined as,

$$g_p = f^{-p/2} \tag{3.63}$$

where, $p = 3$ corresponds to two-body interactions and larger values of p correspond to higher-order gravity perturbations. Efforts to generate arbitrary order time derivative models for Eq. (3.63), using classical methods quickly lead to very complicated vector-valued differential expressions. To eliminate the analytical complexity, this work develops a first-order differential equation for Eq. (3.63), which is cleared of fractions, leading to the differential constraint equation,

$$f\dot{g}_p + \frac{p}{2} g_p \dot{f} = 0 \tag{3.64}$$

Two observations are important: (1) higher order time derivatives of g_p are implicitly defined, and (2) all expressions involve bilinear products which are ideally suited for an application of Leibniz product rule. By applying Leibniz product rule to each term one can solve for the implicitly defined highest order of g_p. For example, computing the nth order time derivative of Eq. (3.64), leads to the implicit rate equation,

$$\sum_{m=0}^{n} \left(\begin{array}{c} n \\ \\ m \end{array} \right) f^{(m)} g_p^{(n-m+1)} + \frac{p}{2} \sum_{m=0}^{n} \left(\begin{array}{c} n \\ \\ m \end{array} \right) g_p^{(m)} f^{(n-m+1)} = 0 \tag{3.65}$$

The highest time derivative of g_p is contained in the first term when $m = 0$. Factoring this term out leads to the highest time derivative of g_p given by,

$$g_p^{(n+1)} = - \left[\frac{p}{2} f^{(1)} g^{(n)} + \sum_{m=1}^{n} \left(\begin{array}{c} n \\ \\ m \end{array} \right) \left(\frac{p}{2} f^{(m+1)} g^{(n-m)} + f^{(m)} g^{(n-m+1)} \right) \right] / f \tag{3.66}$$

The resulting equation is remarkably simple compared with the alternative that involves solving Eq. (3.64) for $\dot{g}_p$ and generating the higher order derivatives. From Eq. (3.60) the two-body acceleration can be expressed in terms of g_3 as,

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3} \boldsymbol{r} = -\mu g_3 \boldsymbol{r} \tag{3.67}$$

and the higher order time derivatives can be computed via Leibniz rule as,

$$\boldsymbol{r}^{(n+2)} = -\mu \sum_{m=0}^{n} \left(\begin{array}{c} n \\ \\ m \end{array} \right) \boldsymbol{r}^{(m)} g^{(n-m)} \tag{3.68}$$

Combining Eq. (3.61), Eq. (3.66) and Eq. (3.68) yields a coupled recursive algorithm for generating the time derivatives of the trajectory motion. The algorithm is shown in Fig. 3.21 and is implemented for solving cases of orbit propagation problems as demonstrated next.

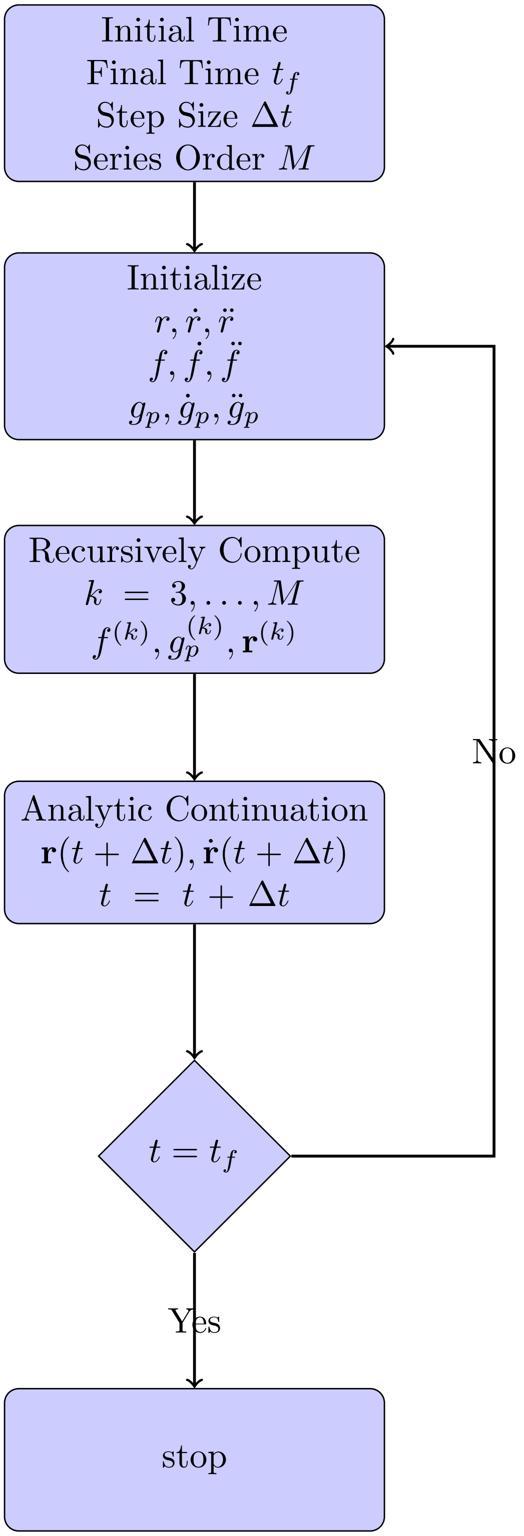

Figure 3.21: The Analytic Continuation Algorithm

3.2.2 Numerical Results

Two examples are presented for the classical/unperturbed two-body problem; (1) a Low Earth Orbit (LEO) with an eccentricity of $e = 0.1$ and (2) a High Eccentricity Orbit (HEO) with $e = 0.7$. Each set of initial conditions is propagated for 20 complete orbits and the results are compared against the analytical Lagrange/Gibbs *F&G* solution. The results are also compared against *Matlab* adaptive Runge-Kutta numerical integrator, *ode45* and the *RKN12(10)* algorithm, [89]. For measuring the accuracy the conservation of energy of the orbit is used and the errors are calculated from,

$$\epsilon_k = \left| \frac{E_k - E_0}{E_0} \right| \tag{3.69}$$

Where, E_0 is the total energy evaluated at the initial conditions and E_k is the total energy at each time step,

$$E_k = \frac{\dot{\boldsymbol{r}}_k \cdot \dot{\boldsymbol{r}}_k}{2} - \frac{\mu}{r_k} \tag{3.70}$$

The initial conditions for the Low Earth Orbit (LEO) are

$$\begin{aligned} \boldsymbol{r} &= \begin{bmatrix} 1.702547136867679 & 6.353992417071098 & 0 \end{bmatrix}^{\mathrm{T}} \times 10^6 \,\mathrm{m} \\ \dot{\boldsymbol{r}} &= \begin{bmatrix} -7.886014053829254 & 2.113051097224035 & 0 \end{bmatrix}^{\mathrm{T}} \times 10^3 \,\mathrm{m/s} \end{aligned} \tag{3.71}$$

The orbit elements are then calculated as,

$$\begin{aligned} a &= 7.30904 \times 10^6 \,\mathrm{m} \\ e &= 0.1 \\ i &= \Omega = \omega = 0^\circ \\ T_p &= 6.21872 \times 10^3 \,\mathrm{sec} \end{aligned} \tag{3.72}$$

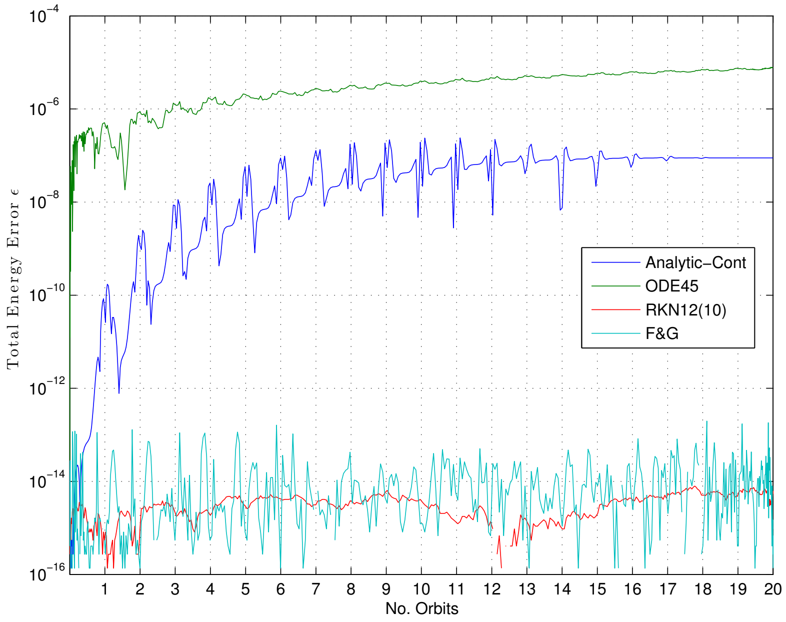

Figure 3.22: Total Energy Error, LEO

where a is the semi-major axis, e is the eccentricity, i is the inclination angle, Ω is the longitude of the ascending node, ω is the argument of periapsis, and T_p is the orbit period. Eq. (3.60) is numerically integrated with the present *Analytic-Cont* algorithm, *ode45* and *RKN12(10)* for 20 complete orbits. The total energy errors are shown in Fig. 3.22 for *RBF-Coll*, *ode45* and the *F&G* analytical solution.

Table 3.2 shows a comparison between *Analytic-Cont*, *RKN12(10)* and *ode45* with the *F&G* solution as the reference in terms of accuracy, calculated from the norm of the total energy error vector in Eq. (3.69) and the simulations time in seconds.

It is shown from Table 3.2 that the *Analytic-Cont* algorithm is more accurate with a lower computational cost when compared to the more optimized *ode45*. *RKN12(10)* almost achieved the same accuracy as the analytical *F&G* but with a significant hit

Table 3.2: Numerical Results Comparison, LEO

Method	**No. Steps**	**Deriv Order**	$\|\epsilon\|$	Sim. Time (sec)
F&G, Orbits = 5	125	N/A	3.6×10^{-13}	N/A
ode45, Orbits = 5	Varies	N/A	1.15×10^{-5}	0.23
RKN12(10), Orbits = 5	Varies	N/A	1.5×10^{-14}	1.06
Analytic-Cont, Orbits = 5	125	10	4.45×10^{-7}	0.044
F&G, Orbits = 10	250	N/A	3.6×10^{-13}	N/A
ode45, Orbits = 10	Varies	N/A	3.43×10^{-5}	0.3
RKN12(10), Orbits = 10	Varies	N/A	1.21×10^{-13}	1.76
Analytic-Cont, Orbits = 10	250	10	7.38×10^{-7}	0.051
F&G, Orbits = 20	500	N/A	6.3×10^{-13}	N/A
ode45, Orbits = 20	Varies	N/A	1.01×10^{-4}	0.4
RKN12(10), Orbits = 20	Varies	N/A	8.57×10^{-14}	3.99
Analytic-Cont, Orbits = 20	500	10	1.55×10^{-6}	0.09

on the computational cost. On average the *Analytic-Cont* is $\approx 35X$ faster than *RKN12(10)* and $\approx 5X$ faster than *ODE45*. This can be considered a significant improvement in the computational cost of the propagator that can have several applications in on-board orbit calculation algorithms.

The initial conditions for the High Eccentricity Orbit (HEO) are given by

$$\begin{aligned} \boldsymbol{r} &= \begin{bmatrix} 2.096434265330419 & 7.823999192941453 & 0 \end{bmatrix}^{\mathrm{T}} \times 10^{6}\,\mathrm{m} \\ \dot{\boldsymbol{r}} &= \begin{bmatrix} -8.834757074967362 & 2.367266023562654 & 0 \end{bmatrix}^{\mathrm{T}} \times 10^{3}\,\mathrm{m/s} \end{aligned} \tag{3.73}$$

With the classical orbit elements,

$$\begin{aligned} a &= 2.7 \times 10^7\,\mathrm{m} \\ e &= 0.7 \\ i &= 0^\circ \\ \Omega &= 45^\circ \\ \omega &= 30^\circ \\ T_p &= 4.41526 \times 10^4\,\mathrm{sec} \end{aligned} \tag{3.74}$$

As was done with the LEO case, Eq. (3.60) is numerically integrated with the present *Analytic-Cont* algorithm, *ode45* and *RKN12(10)* for 20 orbits. The total energy errors are shown in Fig. 3.23 for *Analytic-Cont*, *ode45*, *RKN12(10)* and the *F&G* analytical solution.

Table 3.3 shows the comparison between *Analytic-Cont*, *ode45*, *RKN12(10)* and the *F&G* solution. The comparison is again shown for 5, 10 and 20 orbits.

Table 3.3: Numerical Results Comparison, HEO

Method	**No. Steps**	**Deriv Order**	$\|\epsilon\|$	Sim. Time (sec)
F&G, Orbits = 5	500	N/A	5.23×10^{-12}	N/A
ode45, Orbits = 5	Varies	N/A	1.65×10^{-4}	0.28
RKN12(10), Orbits = 5	Varies	N/A	3.3×10^{-13}	2.99
Analytic-Cont, Orbits = 5	500	30	3.73×10^{-8}	0.34
F&G, Orbits = 10	1000	N/A	7.47×10^{-12}	N/A
ode45, Orbits = 10	Varies	N/A	5.18×10^{-4}	0.4
RKN12(10), Orbits = 10	Varies	N/A	2.06×10^{-13}	6.47
Analytic-Cont, Orbits = 10	1000	30	1.21×10^{-7}	0.57
F&G, Orbits = 20	2000	N/A	6.3×10^{-13}	N/A
ode45, Orbits = 20	Varies	N/A	1.5×10^{-3}	0.5
RKN12(10), Orbits = 20	Varies	N/A	7.35×10^{-13}	11.23
Analytic-Cont, Orbits = 20	2000	30	2.27×10^{-7}	1.001

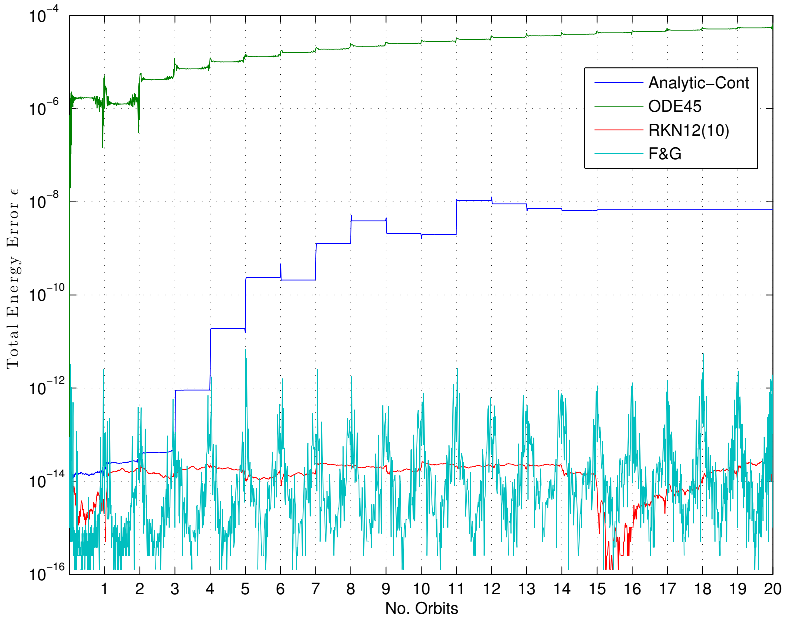

Figure 3.23: Total Energy Error, HEO

As shown from Table 3.3, the optimized adaptive nature of *ode45* proved useful when it comes to the computational cost comparison for 20 orbits. The *Analytic-Cont* algorithm however, is still more accurate across all experiments and with acceptable computational cost with $\approx 10X$ speed up over *RKN12(10)*. Essentially, the Analytic-Cont algorithm is a fixed step size numerical integrator. The step size had to be decreased and the derivative order had to be increased to account for the very high eccentricity of the orbit to maintain high solution accuracy. By developing an adaptive *Analytic-Cont* algorithm the trade-off between the accuracy, the derivative order and the step size can be explored to further improve the overall computational cost of the algorithm.

The present *Analytic-Cont* algorithm is shown to be highly accurate, fast and very simple to implement for long-term orbit propagation of the classical two-body problem. The comparison against *Matlab ODE45* and *RKN12(10)* shows the advantages of the *Analytic-Cont* algorithm that enables larger step-size with relatively high solution accuracy while maintaining low computational cost. The case of 20 full HEO orbits shows the potential of the algorithm for adaptive step size schemes. Gravitational perturbations can also be easily handled with the *Analytic-Cont* algorithm. Areas of studies that will be explored in future works.

4. RADIAL BASIS FUNCTIONS DIRECT TIME-DOMAIN COLLOCATION APPLIED TO THE INITIAL VALUE PROBLEM*

In this chapter, we consider Initial Value Problems (IVPs) for strongly nonlinear dynamical systems, and study numerical methods to analyze short as well as long-term responses. Dynamical systems characterized by a system of second-order nonlinear ordinary differential equations (ODEs) are recast into a system of nonlinear first order ODEs in mixed variables of positions as well as velocities. For each discrete-time interval Radial Basis Functions (RBFs) are assumed as trial functions for the mixed variables in the time domain. A simple collocation method is developed in the time-domain, with Legendre-Gauss-Lobatto nodes as RBF source points as well as collocation points. Numerical examples are provided to compare the present algorithm with explicit as well as implicit methods in terms of accuracy, required size of time-interval (or step) and computational cost. The algorithm developed in this chapter is compared against, the second order central difference method, the classical Runge-Kutta method, the adaptive Runge-Kutta-Fehlberg method, the Newmark-β and the Hilber-Hughes-Taylor methods. First the highly nonlinear Duffing oscillator is analyzed and the solutions obtained from all algorithms are compared against the analytical solution for free oscillation at long times. A Duffing oscillator with impact forcing function is next solved. Solutions are compared against numerical solutions from state of the art *ODE45* numerical integrator for long times. Finally, a nonlinear 3-DOF system is presented and results from all algorithms are compared against *ODE45*. It is shown that the present *RBF-Coll* algorithm is very simple, ef-

[*] Part of this chapter is reprinted with permission from "A Simple, Fast, and Accurate Time-Integrator for Strongly Nonlinear Dynamical Systems" by Elgohary, T. A., Dong, L., Junkins, J. L. and Atluri, S. N., 2014. *CMES: Computer Modeling in Engineering & Sciences*, Vol. 100, No. 3, pp. 240–275, Copyright [2014] by Tech Science Press.

ficient and very accurate in obtaining the solution for the nonlinear IVP. Since other presented methods require a much smaller step size and higher computational cost, the proposed algorithm is advantageous and has promising applications in solving nonlinear dynamical systems, [95].

4.1 Methods for Solving the IVP

A second-order nonlinear dynamic system can be generally recast into a system of first-order ODEs as:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{g}_1(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{f}, t) \equiv \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = \boldsymbol{g}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{f}, t) \end{cases} , \quad t_0 \leq t \leq t_F \tag{4.1}$$

which can be simply rewritten as:

$$\dot{\boldsymbol{x}} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t), \quad t_0 \leq t \leq t_F \tag{4.2}$$

where $\boldsymbol{x}$ is the vector of mixed variables, $\boldsymbol{x} \equiv [\boldsymbol{x}_1, \boldsymbol{x}_2]^{\mathrm{T}}$, $\dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2$, $\boldsymbol{f}$ is the force applied to the system. For a specified set of initial conditions $\boldsymbol{x}_0$ at $t = t_0$, and being given the force function $\boldsymbol{f}(t)$, the initial value problem (IVP) of Eq. (4.2) can be numerically integrated and solved by various explicit and implicit methods of numerical integration.

In explicit methods, the future unknown state is directly expressed in terms of the currently-known system state with an explicit formula. The simplest explicit method is the forward Euler-method:

$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}(t) + \Delta t \boldsymbol{g}\left(\boldsymbol{x}(t), \boldsymbol{f}(t), t\right) \tag{4.3}$$

which is a first-order Taylor series expansion in the time domain.

Another explicit method, the second order central difference method presented in [96–98], is widely used for transient finite element analyses of large scale nonlinear structures, such as crash simulation of automobiles. In this method, the velocity $\boldsymbol{x}_2$ is firstly evaluated at $t + \frac{\Delta t}{2}$, and then $\boldsymbol{x}_1$ is obtained at $t + \Delta t$:

$$\begin{aligned} \boldsymbol{x}_2(t + \frac{\Delta t}{2}) &= \boldsymbol{x}_2(t - \frac{\Delta t}{2}) + \Delta t \boldsymbol{g}_2(\boldsymbol{x}(t), \boldsymbol{f}(t), t) \\ \boldsymbol{x}_1(t + \Delta t) &= \boldsymbol{x}_1(t) + \Delta t \boldsymbol{x}_2(t + \frac{\Delta t}{2}) \end{aligned} \tag{4.4}$$

The Runge-Kutta, *RK*, family of methods can be considered as the most widely used explicit methods for numerical integration of general dynamical systems. The first order *RK* method is simply the forward Euler-method given in Eq. (4.3). The classical or the 4th-order *RK* method, which evaluates the solution in 4 steps, is the most commonly used among various *RK* methods. In [85], adaptive step-size 4th-order *RK* methods are developed and are now known as the Runge-Kutta-Fehlberg, *RKF*, methods. Several higher order adaptive *RKF* methods, [86], are widely used for very-high accuracy applications such as orbit propagation problems, see [90, 92].

Implicit methods put the currently-known state and the unknown future state in a set of linear or nonlinear algebraic equations, by solving which the future state can be obtained. Backward Euler-Method is an illustration of this concept:

$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}(t) + \Delta t \boldsymbol{g}(\boldsymbol{x}(t + \Delta t), \boldsymbol{f}(t + \Delta t), t + \Delta t) \tag{4.5}$$

In [99], Newmark introduced the Newmark-β method based on the extended mean value theorem, which is among the most widely-used implicit methods for the numerically evaluating the dynamical response of engineering structures,

$$
\begin{aligned}
\boldsymbol{x}_1(t+\Delta t) &= \boldsymbol{x}(t) + \Delta t \boldsymbol{x}_2(t) + \frac{1}{2}\Delta t^2 \left[(1-2\beta)\, \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{f}(t), t)\right. \\
&\qquad \left. + 2\beta \boldsymbol{g}(\boldsymbol{x}(t+\Delta t), \boldsymbol{f}(t+\Delta t), t+\Delta t)\right] \\
\boldsymbol{x}_2(t+\Delta t) &= \boldsymbol{x}_2(t) + \Delta t \left[(1-\gamma) \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{f}(t), t)\right. \\
&\qquad \left. + \gamma \boldsymbol{g}(\boldsymbol{x}(t+\Delta t), \boldsymbol{f}(t+\Delta t), t+\Delta t)\right]
\end{aligned}
\tag{4.6}
$$

Typical values for γ and β are $\gamma = 1/2$ and $\beta = 1/4$.

What is considered as a generalization of the Newmark-β method is introduced in the Hilber-Hughes-Taylor or *HHT*-α method, [100]:

$$
\begin{aligned}
&\boldsymbol{x}_1(t+\Delta t) = \boldsymbol{x}(t) + \Delta t \boldsymbol{x}_2(t) + \frac{1}{2}\Delta t^2 \left[(1-2\beta)\, \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{f}(t), t) + 2\beta \boldsymbol{a}(\alpha)\right] \\
&\boldsymbol{x}_2(t+\Delta t) = \boldsymbol{x}_2(t) + \Delta t \left[(1-\gamma) \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{f}(t), t) + \gamma \boldsymbol{a}(\alpha)\right] \\
&\boldsymbol{a}(\alpha) = (1+\alpha) \boldsymbol{g}(\boldsymbol{x}(t+\Delta t), \boldsymbol{f}(t+\Delta t), t+\Delta t) - \alpha \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{f}(t), t)
\end{aligned}
\tag{4.7}
$$

where, $\gamma = \frac{1-2\alpha}{2}$, $\beta = \left(\frac{1-\alpha}{2}\right)^2$ and $\alpha \in [-\frac{1}{3}, 0]$.

For all the above mentioned explicit and implicit methods, the size of time-steps plays an important role for the accuracy of computational results. Generally speaking, numerical stability is not guaranteed for explicit methods, thus a much smaller time step is necessary for explicit methods to obtain an accurate solution. On the other hand, because an implicit method requires the solution of set of linear/nonlinear algebra equations in each time step, the computational burden/time of implicit methods in each time step is much higher than explicit methods. A comprehensive review of such methods with applications in computational structural dynamics is presented in [101, 102].

Besides all of the above-mentioned widely-used numerical integrators, Eq. (4.2) as a set of first order ODEs can be numerically solved by a wide spectrum of computational methods, see [103]. These methods in the time domain, such as collocation, finite volume, Galerkin, MLPG, are all essentially branches from the same tree, using the concept of weighted-residual weak-forms, and with different trial and test functions, see [104]. Among these methods, collocation is one of the simplest and the most efficient ones. In [105], a collocation method with harmonic trial functions was developed for studying the periodic responses of nonlinear Duffing oscillators and aeroelastic systems. Modified Chebyshev-Picard Iteration, *MCPI*, methods have been recently introduced for orbit propagation and general initial value problems in [93, 94, 106]. The algorithm uses Chebyshev polynomials as basis functions and uses Picard iteration to solve the set of NAEs produced after after discretization. The method is extensively implemented for orbit propagation problems, [107, 108], handling higher degrees of perturbations and low to high eccentricity orbits. The algorithm as presented in [109] is shown in Fig. 4.1.

In this study, the *REB-Coll* algorithm is further recast as a general time-domain step-wise numerical integrator, to numerically integrate the IVP in Eq. (4.2) for arbitrary nonlinear systems. The algorithm is compared against the central difference *Explicit* method, the classical Runge-Kutta, *RK4*, method, the *Newmwark-β* method, the *HHT-α* method and the *MCPI* method in terms of accuracy, step size and computational time, using various free-vibration, forced vibration, impact load problems of single-DOF as well as coupled multi-DOF Duffing oscillators. These numerical examples clearly show the advantages of the present *RBF-Coll* algorithm which enables a much larger time step, and produces high solution accuracy while maintaining a relatively low computational cost.

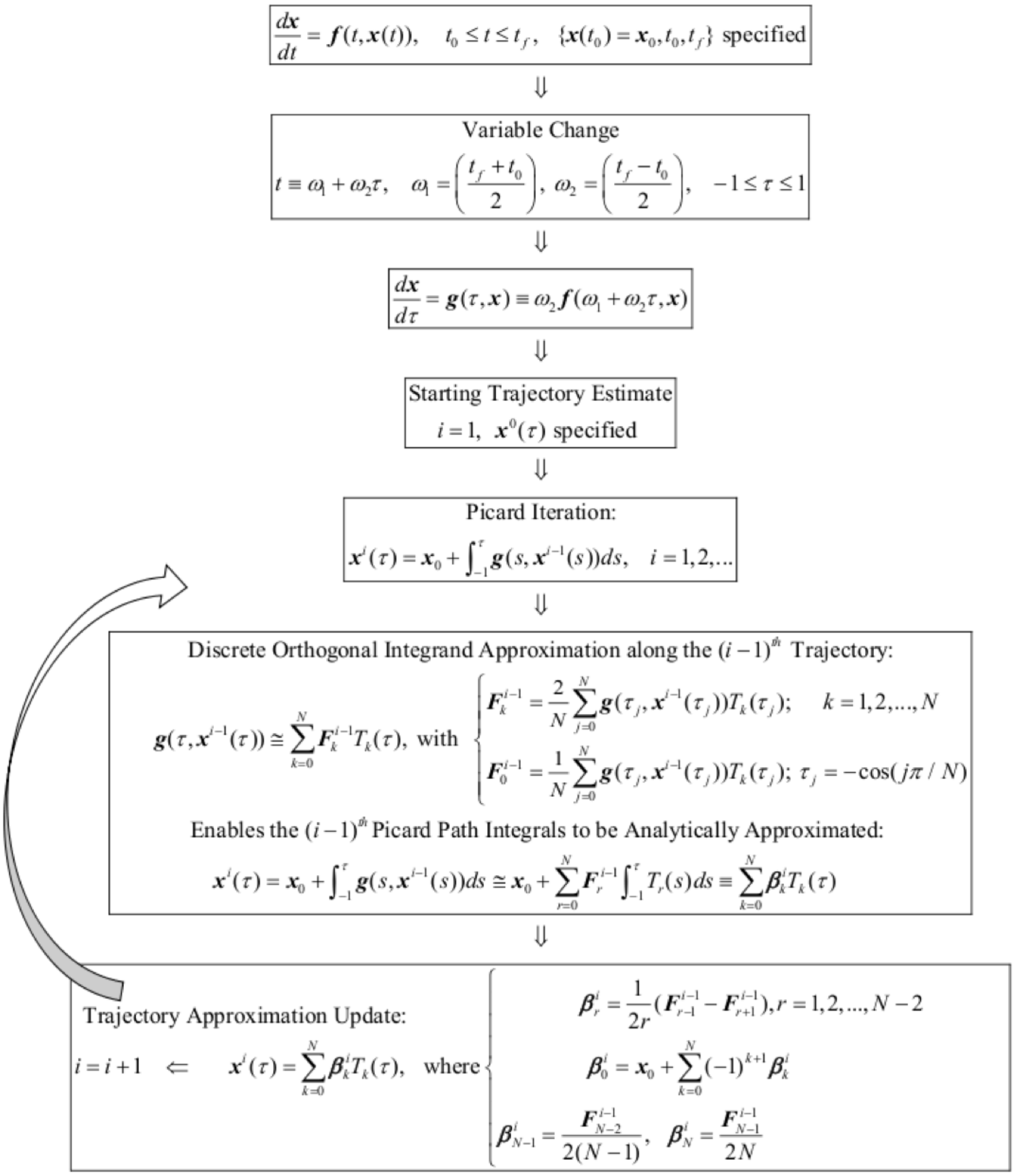
$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}(t,\boldsymbol{x}(t)), \quad t_0 \le t \le t_f, \quad \{\boldsymbol{x}(t_0) = \boldsymbol{x}_0, t_0, t_f\}$ specified
⇓
Variable Change
$t \equiv \omega_1 + \omega_2\tau, \quad \omega_1 = \left(\frac{t_f + t_0}{2}\right), \ \omega_2 = \left(\frac{t_f - t_0}{2}\right), \quad -1 \le \tau \le 1$
⇓
$\frac{d\boldsymbol{x}}{d\tau} = \boldsymbol{g}(\tau,\boldsymbol{x}) \equiv \omega_2 \boldsymbol{f}(\omega_1 + \omega_2\tau, \boldsymbol{x})$
⇓
Starting Trajectory Estimate
$i = 1, \ \boldsymbol{x}^0(\tau)$ specified
⇓
Picard Iteration:
$\boldsymbol{x}^i(\tau) = \boldsymbol{x}_0 + \int_{-1}^{\tau} \boldsymbol{g}(s, \boldsymbol{x}^{i-1}(s))ds, \quad i = 1,2,...$
⇓
Discrete Orthogonal Integrand Approximation along the $(i-1)^{th}$ Trajectory:
$\boldsymbol{g}(\tau, \boldsymbol{x}^{i-1}(\tau)) \cong \sum_{k=0}^{N} \boldsymbol{F}_k^{i-1} T_k(\tau)$, with $\begin{cases} \boldsymbol{F}_k^{i-1} = \frac{2}{N}\sum_{j=0}^{N} \boldsymbol{g}(\tau_j, \boldsymbol{x}^{i-1}(\tau_j))T_k(\tau_j); \quad k = 1,2,...,N \\ \boldsymbol{F}_0^{i-1} = \frac{1}{N}\sum_{j=0}^{N} \boldsymbol{g}(\tau_j, \boldsymbol{x}^{i-1}(\tau_j))T_k(\tau_j); \ \tau_j = -\cos(j\pi / N) \end{cases}$
Enables the $(i-1)^{th}$ Picard Path Integrals to be Analytically Approximated:
$\boldsymbol{x}^i(\tau) = \boldsymbol{x}_0 + \int_{-1}^{\tau} \boldsymbol{g}(s, \boldsymbol{x}^{i-1}(s))ds \cong \boldsymbol{x}_0 + \sum_{r=0}^{N} \boldsymbol{F}_r^{i-1} \int_{-1}^{\tau} T_r(s)ds \equiv \sum_{k=0}^{N} \boldsymbol{\beta}_k^i T_k(\tau)$
⇓
Trajectory Approximation Update:
$i = i+1 \quad \Leftarrow \quad \boldsymbol{x}^i(\tau) = \sum_{k=0}^{N} \boldsymbol{\beta}_k^i T_k(\tau)$, where $\begin{cases} \boldsymbol{\beta}_r^i = \frac{1}{2r}(\boldsymbol{F}_{r-1}^{i-1} - \boldsymbol{F}_{r+1}^{i-1}), r = 1,2,...,N-2 \\ \boldsymbol{\beta}_0^i = \boldsymbol{x}_0 + \sum_{k=0}^{N}(-1)^{k+1}\boldsymbol{\beta}_k^i \\ \boldsymbol{\beta}_{N-1}^i = \frac{\boldsymbol{F}_{N-2}^{i-1}}{2(N-1)}, \ \boldsymbol{\beta}_N^i = \frac{\boldsymbol{F}_{N-1}^{i-1}}{2N} \end{cases}$

Figure 4.1: *MCPI* Algorithm for Solving the IVP

4.2 Radial Basis Functions & Collocation Time Integrator

Radial Basis Functions (RBFs) are real-valued functions with values depending on the distance from a source point, $\phi(\boldsymbol{x}, \boldsymbol{x}_c) = \phi(\|\boldsymbol{x} - \boldsymbol{x}_c\|) = \phi(r)$. Some of the commonly used types for RBFs are as follows, [110]:

$$\begin{aligned}
&\phi(r) = e^{-(cr)^2} \quad \text{Gaussian} \\
&\phi(r) = \frac{1}{1+(cr)^2} \quad \text{Inverse quadratic} \\
&\phi(r) = \sqrt{1+(cr)^2} \quad \text{Multiquadric} \\
&\phi(r) = \frac{1}{\sqrt{1+(cr)^2}} \quad \text{Inverse multiquadric}
\end{aligned} \tag{4.8}$$

where $c > 0$ is a shaping parameter.

In this work, the trial functions are expressed as a linear combination of Gaussian functions, with N Legendre-Gauss-Lobatto (LGL) nodes ($t_j, j = 1, \ldots, N$) as the source points. We adopt these nodes because they concentrate nodal density near the end of the approximation interval in a way that is known to reduce Runge's phenomenon effect of large oscillatory errors near the boundary. The results we present here do not require us to restrict attention only to the LGL nodes. The LGL nodes can be obtained from solving the differential equation,

$$\left(1 - \tau_j^2\right) \dot{p}_N(\tau_j) = 0 \tag{4.9}$$

Where, $p_N(\tau)$ are the well known Legendre polynomials orthogonal to the weight function $w(\tau) = 1$ on the interval $\tau \in [-1, 1]$ and satisfy the recursion,

$$\begin{aligned} p_0(\tau) &= 1 \\ p_1(\tau) &= x \\ p_{i+1} &= \left(\frac{2i+1}{i+1}\right) \tau p_i(\tau) - \left(\frac{i}{i+1}\right) p_{i-1}(\tau), \qquad i = 1, 2, \ldots \end{aligned} \tag{4.10}$$

The solution of (4.9) produces the node distribution $-1 = \tau_0 < \tau_1 < \ldots < \tau_N = 1$. The solution is generally obtained by numerical algorithms. [111]. By the simple mapping in (4.11), $\tau = [-1, \ldots, 1]$ is transformed into $t = [t_0, \ldots, t_F]$ to obtain the LGL nodes for an arbitrary time interval:

$$t = \frac{t_F - t_0}{2}\tau + \frac{t_F + t_0}{2} \tag{4.11}$$

In the currently-developed *RBF-Coll* numerical integrator, the time domain of interest, for long-time responses, is divided into a set of time steps [or time intervals] with $t_0, t_1, t_2, \ldots, t_F$, with $t_i = t_{i-1} + \Delta t$. For each time step or interval $t_{i-1} \le t \le t_i$, the trial functions in the time-domain are expressed as a liner combination of Radial Basis Functions. In this study, Gaussian functions are used because of its simplicity. The source points of RBFs, for one attractive choice, are located at N Legendre-Gauss-Lobatto (LGL) nodes within each time step, i.e. $t_i^1 = t_{i-1}, t_i^2, t_i^3 \ldots, t_i^N = t_i$, where the subscript denotes the time step, and the superscript denotes the number of LGL nodes.

The state of the dynamical system at each LGL node can therefore be expressed in terms of the undetermined coefficients of RBF basis functions :

$$\boldsymbol{x}(t_i^j) = \sum_{k=1}^{N} \phi(t_i^j, t_i^k)\boldsymbol{a}_k, \quad i = 1, \ldots, N \tag{4.12}$$

In matrix-vector form Eq. (4.12) can be rewritten as,

$$\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{A} \tag{4.13}$$

where $\boldsymbol{\Phi}$ represents the matrix of basis functions, $\boldsymbol{A}$ is the vector of undetermined coefficients, and $\boldsymbol{X}$ is the vector of unknown states at each LGL node. The time-differentiation of Eq. (4.13) can be then expressed as,

$$\dot{\boldsymbol{X}} = \dot{\boldsymbol{\Phi}}\boldsymbol{A} \tag{4.14}$$

Hence, combining Eq. (4.13) and Eq. (4.14), $\dot{\boldsymbol{X}}$ is related to $\boldsymbol{X}$ by,

$$\dot{\boldsymbol{X}} = D\boldsymbol{X}, \quad D \equiv \dot{\boldsymbol{\Phi}}\boldsymbol{\Phi}^{-1} \tag{4.15}$$

where, D is called the derivative matrix, which is numerically evaluated from the RBF basis functions and their time-derivatives evaluated at the LGL nodes.

With the derivative matrix being defined, and with $\boldsymbol{x}_{i-1} = \hat{\boldsymbol{x}}_{i-1}$ being known from the previous time step, we collocate Eq. (4.2) at $t_i^2, t_i^3, \ldots, t_i^N$, and collocate the initial condition at t_i^1, leading to the following set of discretized equations:

$$\begin{aligned} &\boldsymbol{x}_i^1 - \hat{\boldsymbol{x}}_{i-1} = 0 \\ &D\boldsymbol{x}_j - \boldsymbol{g}(\boldsymbol{x}_j, \boldsymbol{f}, t_j) = 0, \quad j = 2, 3, \ldots, N \end{aligned} \tag{4.16}$$

The algorithm as presented above is very simple and easy to implement. The set of nonlinear algebraic equations is solved with the classical Newton's method in this study, whereas other Jacobian-inverse-free methods can also be applied as in [22, 29, 112]. By solving Eq. (4.16), the unknown states at each LGL node, as well as the unknown state at the end of this time interval or step ($\boldsymbol{x}_i$) are obtained. In this way, the states at each time step within the entire time history, i.e. $t_0, t_1, t_2, \ldots, t_F$ can be numerically evaluated by applying the above numerical algorithm in a sequential procedure.

4.3 Optimal Selection of the Shape Parameter

The free shape parameter c appearing in most radial basis functions, Eq. (4.8), has a significant effect on the conditioning of the matrix of basis functions, $\mathbf{\Phi}$. The best convergence accuracy is obtained when c is very small, $c \to 0$ and the nodal density is separated by about $c/2$. However, this leads, for large N, to an ill-conditioned matrix and consequently corruption of the accuracy by arithmetic errors. To overcome such poor conditioning problem, Fornberg et al. introduced two methods to stably compute solutions as $c \to 0$; (1) the Contour-Padé method and (2) the RBF-QR method, [113–115]. Those methods are shown to successfully overcome the ill-conditioning of the basis functions matrix for relatively large number of basis functions and to accurately compute the solution at very small values of c. Nonetheless, for finite precision arithmetic and a particular inversion algorithm, it is still shown that an optimal value of c exists that would result in the most accurate solution.

An algorithm for finding the optimal value of the shape parameter was introduced based on the leave-one-out cross validation (LOOCV) algorithm, [116]. The method minimizes a cost function defined by the norm of the error vector evaluated from the difference between the data point and the interpolant to a reduced data set obtained by removing the point and the corresponding data value. Hence, for a system given by,

$$A\boldsymbol{x} = \boldsymbol{b} \tag{4.17}$$

The error vector to be minimized is given by,

$$e_i = \frac{x_i}{A_{ii}^{-1}} \tag{4.18}$$

The method is extended to encompass radial basis functions used in pseudo-spectral methods applied to partial differential equations. [117] An algorithm based on the LOOCV is developed and implemented to obtain the optimal shape parameter for the best conditioning of the derivative matrix D with direct implementation in *Matlab*. [117, 118] The derivative matrix expression in Eq. (4.15), can be rewritten as,

$$\mathbf{\Phi} D^{\mathrm{T}} = \dot{\mathbf{\Phi}}^{\mathrm{T}} \tag{4.19}$$

which has the same form of the general linear system in Eq. (4.17) leading to the evaluation of the error matrix as,

$$E_{ij} = \frac{D_{ij}^{\mathrm{T}}}{\mathbf{\Phi}_{ii}^{-1}} \tag{4.20}$$

Following the work of Fasshauer, the LOOCV algorithm is used to find the optimal value of the shape parameter to evaluate the derivative matrix D at each time interval. The algorithm is very simple and can be implemented as follows, [116, 117].

The minimization presented in the algorithm can be then simply performed via *Matlab* built-in function *fminbnd*, [117, 118]. It is noted that for the best results one could combine the Contour-Padé method or the RBF-QR method with the shape parameter optimization algorithm. This approach will be pursued in future studies.

Algorithm 1 Algorithm for finding optimal shape parameter c

```
Initial guess of $c$
Evaluate $\mathbf{\Phi}$
Evaluate $\dot{\Phi}$
$D = \dot{\mathbf{\Phi}}\mathbf{\Phi}^{-1}$
Error Matrix $= D^{\mathrm{T}}/\mathrm{Diag}[\mathbf{\Phi}^{-1}]$
Find $c$ to Min $\|\text{Error Matrix}\|$
```

4.4 Numerical Results

In this section numerical experiments are presented for three cases of nonlinear dynamical systems, namely: the highly nonlinear unforced single Duffing oscillator, a single Duffing oscillator subjected to an impact load, and a 3-Degree-of-Freedom (3-DOF) coupled nonlinear Duffing oscillator system subjected to a harmonic load. For each case, the present RBF-Collocation, *RBF-Coll*, algorithm is compared against the 4th order Runge-Kutta (*RK4*), the *Explicit* method, [96, 98], the *Newmark-β* method, [99] and the *HHT-α* method, [100]. A table is presented to show the time step Δt , the Root-Mean-Square (RMS) error of the position state and the computation time for each case. Plots of the error time history normalized by the response amplitude, Eq. (4.21), are also presented for each case to demonstrate the accuracy of each algorithm. Furthermore, the analytical solution and the solutions obtained by various numerical methods are plotted to highlight the magnitudes of the errors for each algorithm.

$$\Delta x(t) \equiv \left\| \frac{x(t)_{\mathrm{Ana}} - x(t)_{\mathrm{Num}}}{\mathrm{Amp}\left[x(t)_{\mathrm{Ana}}\right]} \right\| \tag{4.21}$$

For each method, two numerical cases are presented. The first, denoted by the superscript 1, uses the same Δt as the *RBF-Coll* algorithm, for all the other algorithms.

This establishes a baseline for comparing solution accuracy and computational time of all algorithms. The second, denoted by the superscript 2, *explores the ability of each algorithm to achieve/maintain higher accuracy by decreasing* Δt. As expected, and also shown, all methods will require reducing the step size to achieve a higher solution accuracy except for the *RBF-Coll* method, which can maintain the solution accuracy by increasing the number of collocation points without decreasing the step size.

4.4.1 Free Vibration of Highly Nonlinear Duffing Oscillator

The unforced Duffing oscillator is described by,

$$\begin{aligned} &\ddot{x} + \omega_n^2 x + \eta x^3 = 0 \qquad 0 \leq t \leq t_F \\ &x(0) = x_0 \quad \dot{x}(0) = \dot{x}_0 \end{aligned} \tag{4.22}$$

For a high hardening nonlinearity $\eta \geq 1$, Jacobi-elliptic functions provide an analytical solution for Eq. (4.22), [119], as,

$$x(t) = X \mathrm{cn}\left(\omega t + \theta, k^2\right) \tag{4.23}$$

where, cn is the the Jacobi-elliptic function, ω the frequency, k the modulus, X the amplitude, and θ the phase angle of the function. The reader is referred to [119] for the detailed expression of each parameter in Eq. (4.23).

For the parameters values shown in Table 4.1, the analytical solution is compared against MATLAB *ODE45*, *Explicit* method, *RK4*, *Newmark-*β, *HHT-*α, *MCPI* and the present *RBF-Coll* algorithm. For *ODE45* ^{1} and *ODE45* ^{2} the tolerances are set to 10^{-6} and 10^{-9}, respectively. For *MCPI* ^{1} and *MCPI* ^{2} the order and the number of sample points are 8 and 16, respectively. For *RBF-Coll* ^{1}, the number of collocation points $N = 7$ and shaping parameter $c = 0.77$. For *RBF-Coll* ^{2}, $N = 35$ and $c = (N - 1)/4\Delta t$. Table 4.2 shows the step size, the positions state, $x_1(t)$, RMS error and the computation time for all the tested methods. Figure 4.2 through Fig. 4.13 show the normalized state error time history for each method, Fig. 4.14 and Fig. 4.15 show the normalized state error time history for the present *RBF-Coll* algorithm and finally Fig. 4.16 show the comparison of computations results by various methods in the last period of the freely vibrating duffing oscillator.

Table 4.1: Parameters for the Freely-Vibrating Duffing System, [95]

Parameter	**Value**
t_0	0
t_F	500
$x_1(0)$	1
$x_2(0)$	0
ω_n	1
η	1
Period, P	4.77

Table 4.2: Comparison of Numerical Methods, Freely Vibrating Highly Nonlinear Duffing Oscillator

Method	$\boldsymbol{\Delta} t$ (sec)	**Position RMS Error**	**Simulation Time (sec)**
ODE45 [1]	Variable	1.18×10^{-4}	0.48
ODE45 [2]	Variable	1.14×10^{-6}	1.37
Explicit [1]	$P/10$	0.1	0.006
Explicit [2]	$P/500$	1.5×10^{-3}	0.2
RK4 [1]	$P/10$	0.86	0.03
RK4 [2]	$P/250$	5×10^{-6}	0.56
Newmark-β [1]	$P/10$	0.998	3.62
Newmark-β [2]	$P/500$	3.1×10^{-3}	147
HHT-α [1]	$P/10$	0.841	3.53
HHT-α [2]	$P/500$	4.4×10^{-3}	149.1
MCPI [1]	$P/10$	2.1×10^{-8}	0.16
MCPI [2]	$P/2$	8.5×10^{-8}	0.08
RBF-Coll [1]	$P/10$	3.5×10^{-6}	0.32
RBF-Coll [2]	P	1.8×10^{-6}	0.19

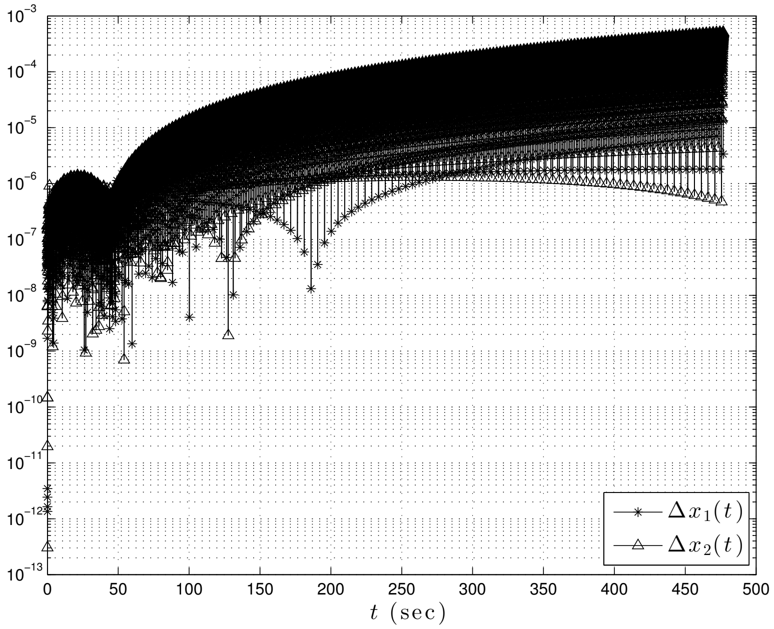

Figure 4.2: Normalized State Error Time History, *ODE45* [1], [95]

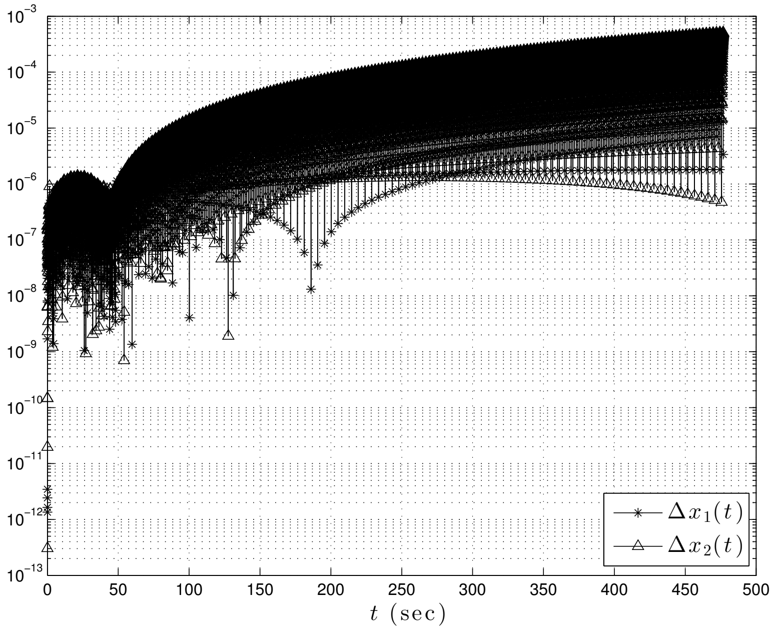

Figure 4.3: Normalized State Error Time History, *ODE45* [2], [95]

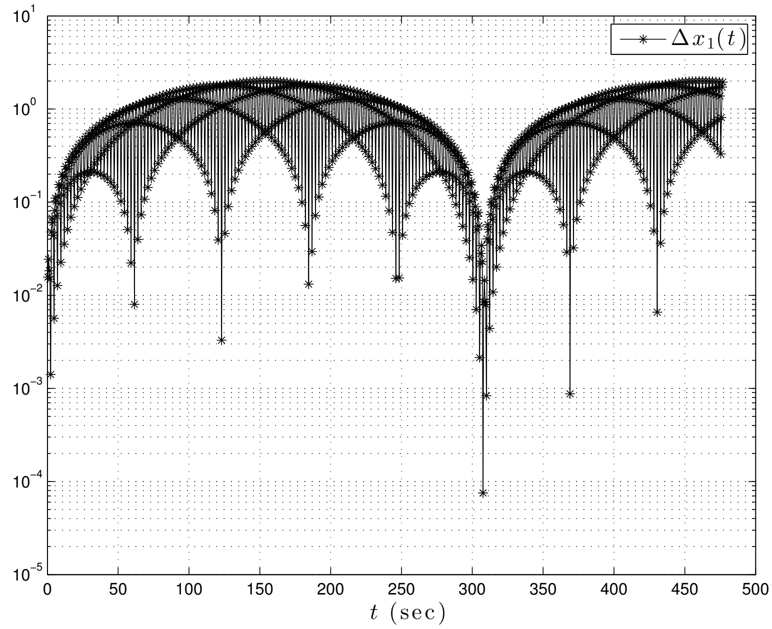

Figure 4.4: Normalized Position Error Time History, *Explicit* Method [1], [95]

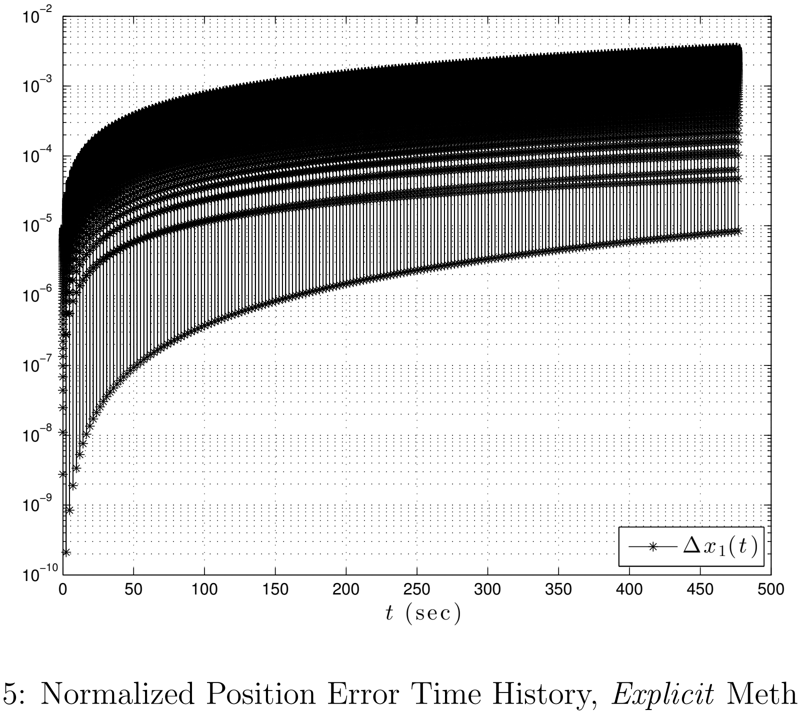

Figure 4.5: Normalized Position Error Time History, *Explicit* Method [2], [95]

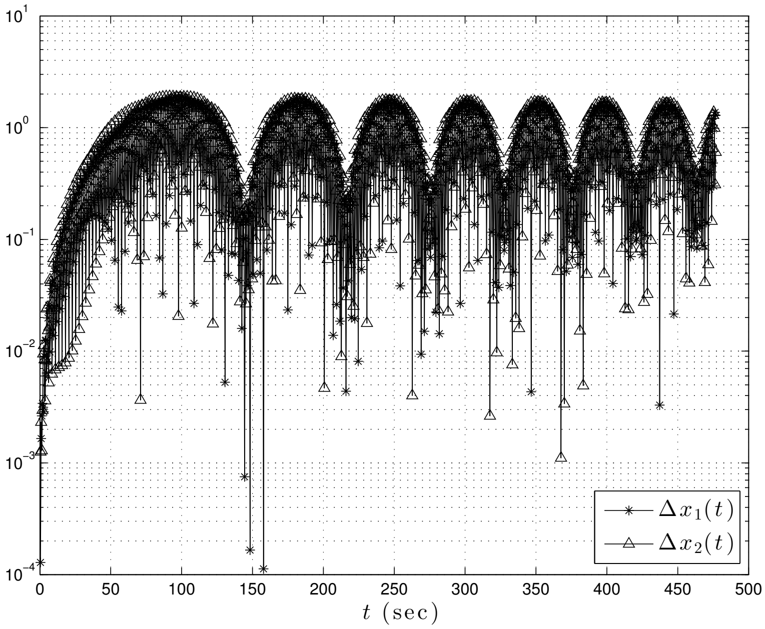

Figure 4.6: Normalized State Error Time History, *RK4* [1], [95]

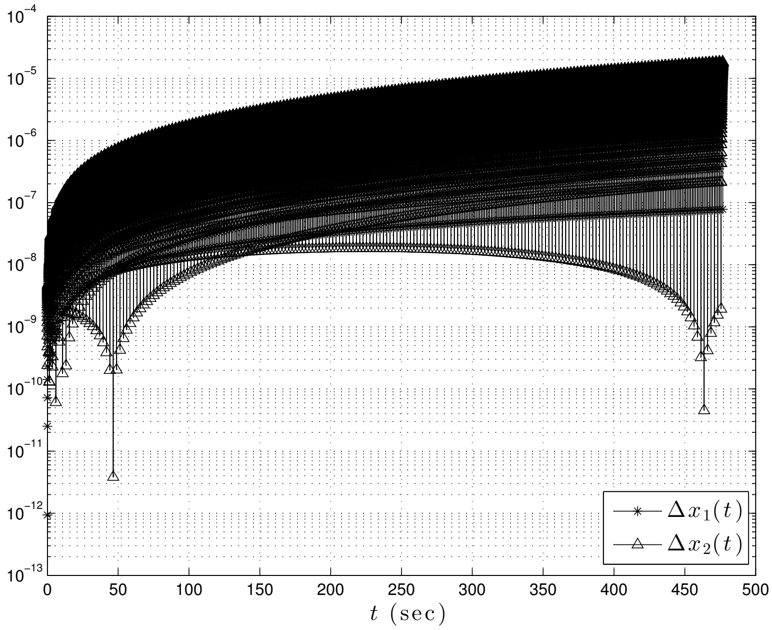

Figure 4.7: Normalized State Error Time History, *RK4* [2], [95]

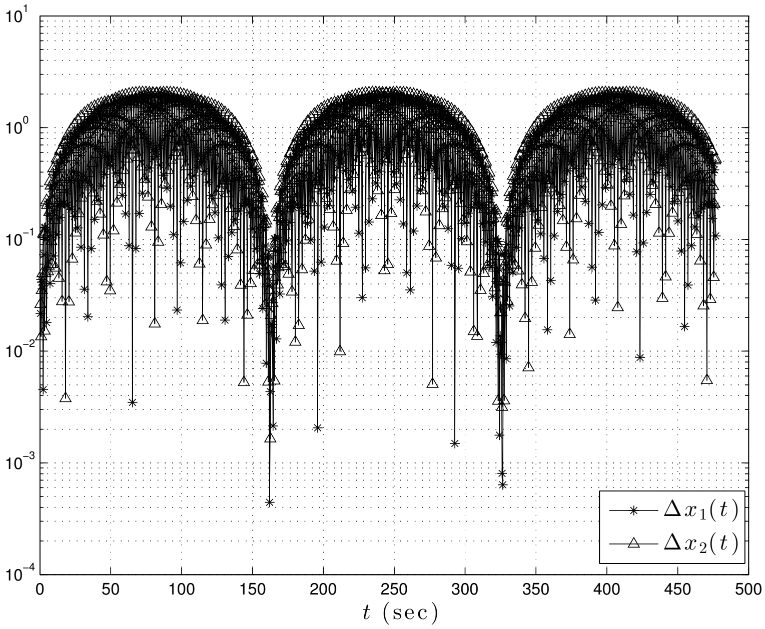

Figure 4.8: Normalized State Error Time History, *Newmark-*β^{1}, [95]

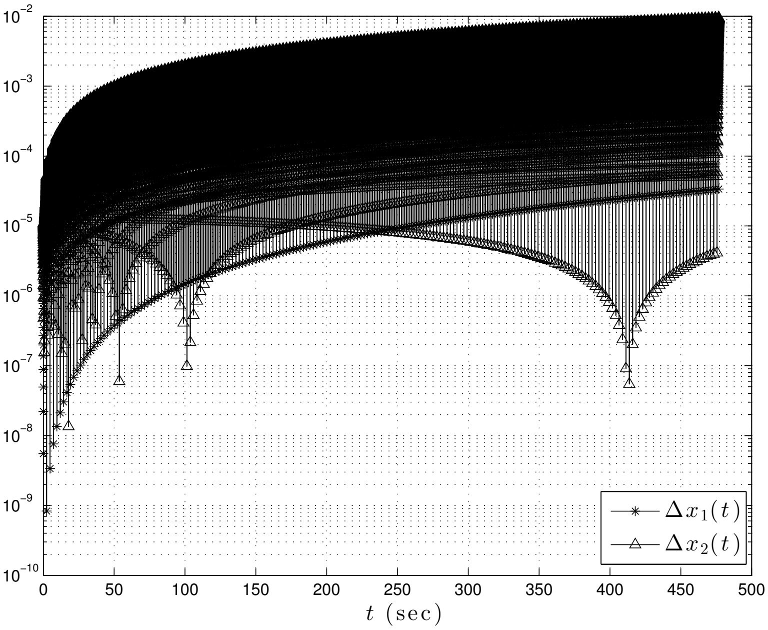

Figure 4.9: Normalized State Error Time History, *Newmark-*β^{2}, [95]

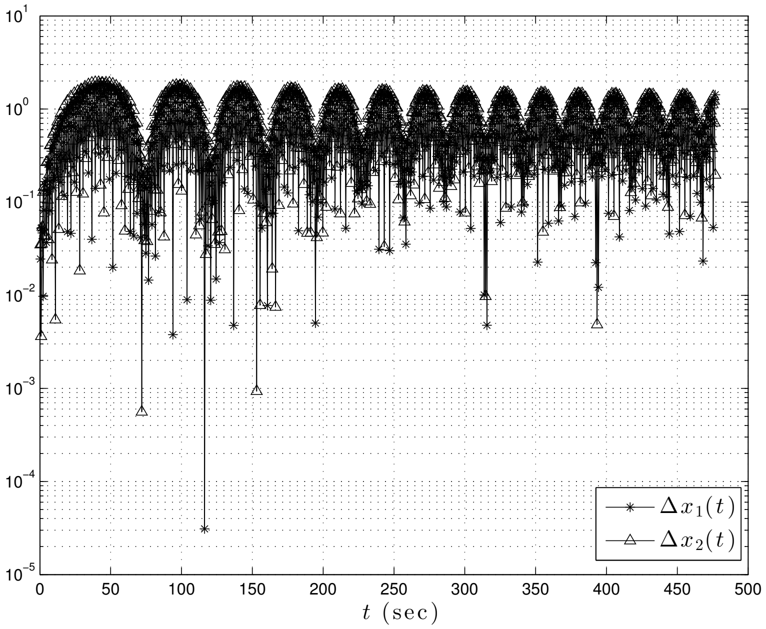

Figure 4.10: Normalized State Error Time History, $HHT\alpha^1$, [95]

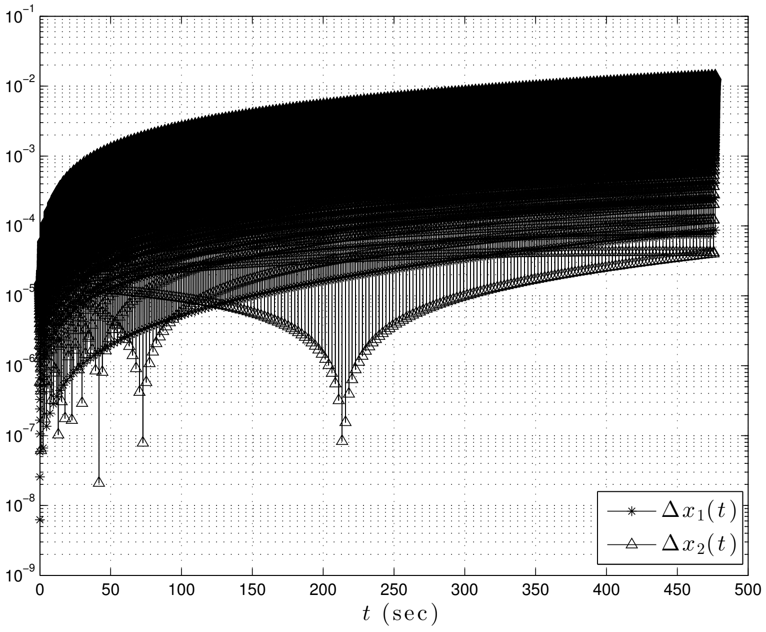

Figure 4.11: Normalized State Error Time History, $HHT\alpha^2$, [95]

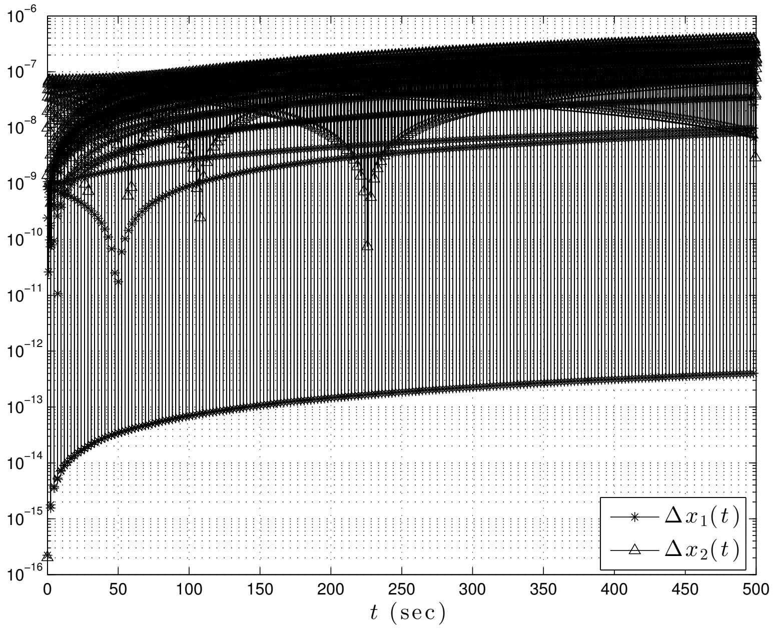

Figure 4.12: Normalized State Error Time History, *MCPI* [1]

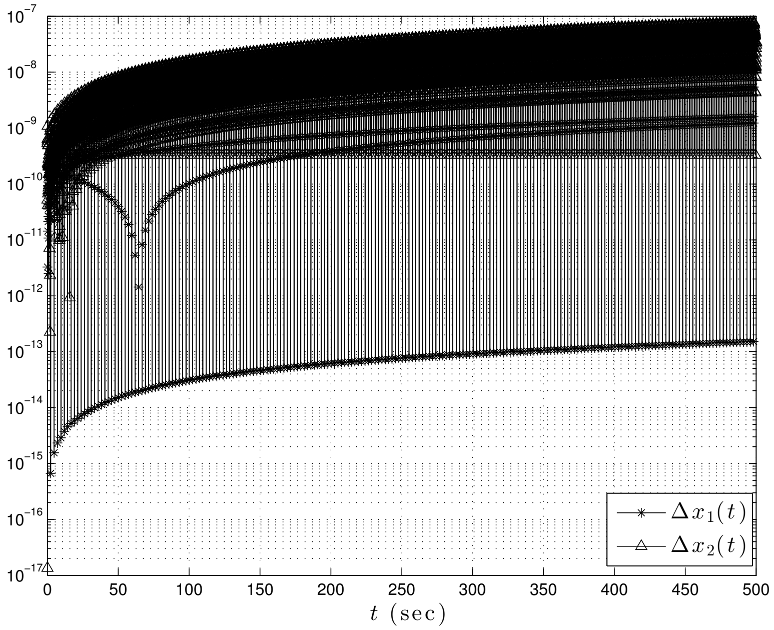

Figure 4.13: Normalized State Error Time History, *MCPI* [2]

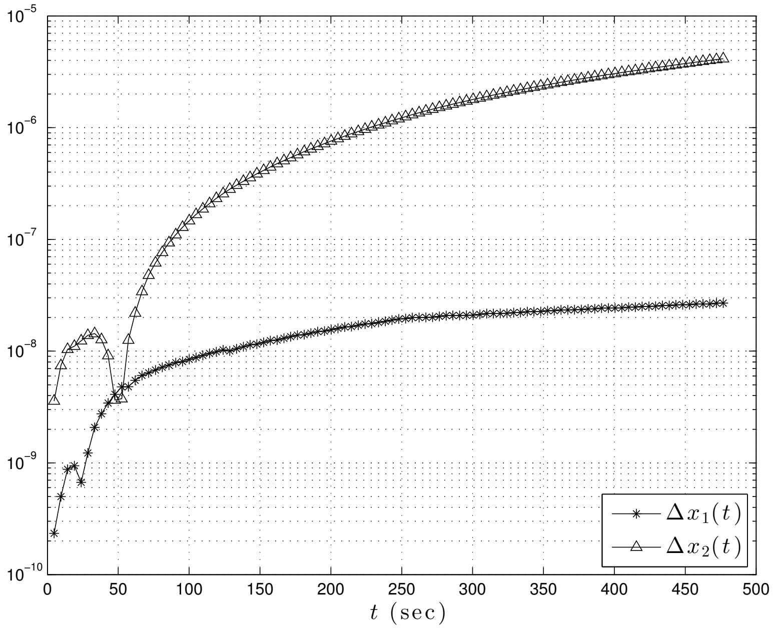

Figure 4.14: Normalized State Error Time History, *RBF-Coll* [1], [95]

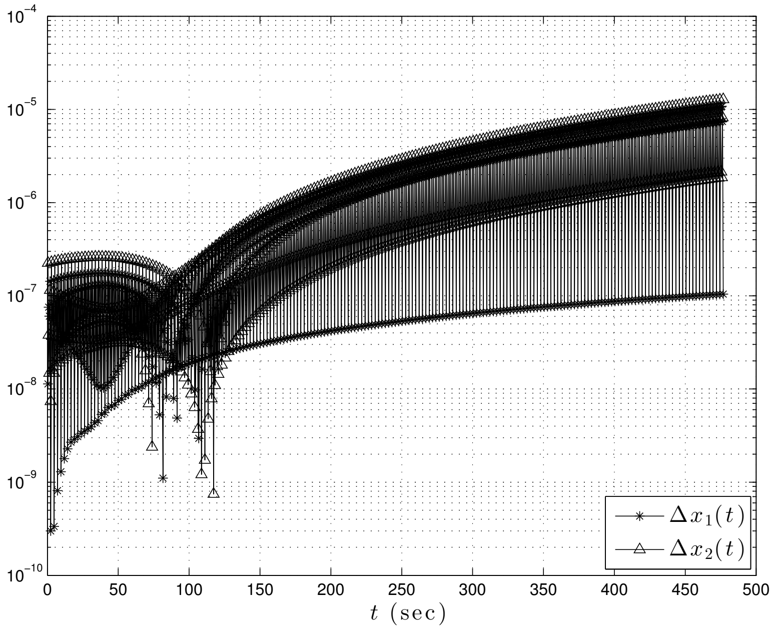

Figure 4.15: Normalized State Error Time History, *RBF-Coll* [2], [95]

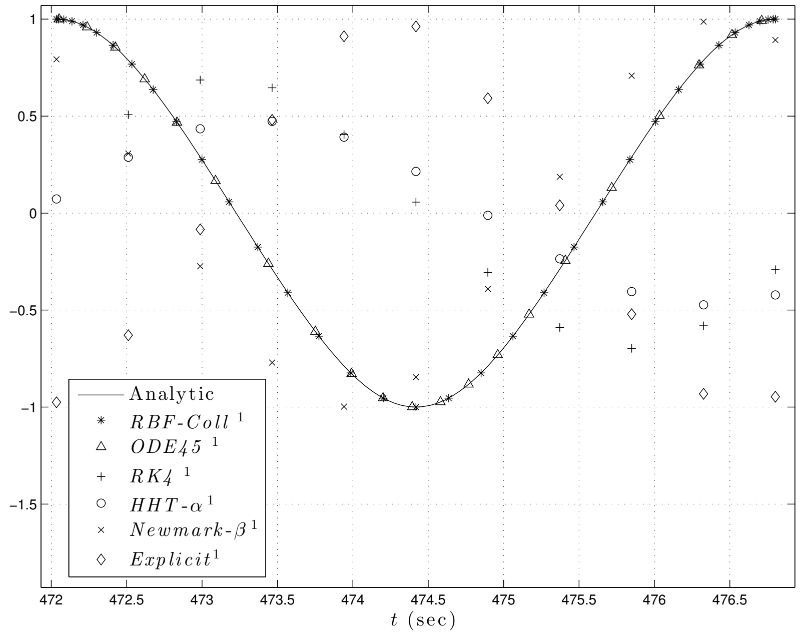

Figure 4.16: Solution Comparison Last Period of Integration, [95]

Given the nonlinear oscillator period shown in Table 4.2, the numerical integration is performed for slightly above a total of 100 cycles. Clearly, all the other presented numerical methods requires a much smaller step size to achieve an acceptable accuracy, whereas the *RBF-Coll* algorithm can maintain high accuracy with a very large step size as demonstrated in Table 4.2, Fig. 4.14 and Fig. 4.14. It should also be noted that *Explicit*, *Newmark-β*, and *HHT-α* methods perform reasonably well for a short term, but accuracy is gradually lost in the long term even if a very tiny time step is used. The *MCPI* method also shows very high accuracy while maintaining a relatively low computational cost. However, numerical experiments reveal a limitation on increasing the time-step as the solution diverges for values of $\Delta t > P/2$ even with the order and the number of sample points are increased to 500.

4.4.2 Highly Nonlinear Duffing Oscillator with Impact Loading

An impact triangular forcing function is applied to the Duffing oscillator in Eq. (4.22) as,

$$\begin{aligned}
&\ddot{x} + \omega_n^2 x + \eta x^3 = f(t) \qquad 0 \leq t \leq t_F \\
&x(0) = x_0 \quad \dot{x}(0) = \dot{x}_0 \\
&f(t) = \begin{cases} a\,t & 0 \leq t \leq t_1/2 \\ a(1-t) & t_1/2 < t \leq t_1 \end{cases}
\end{aligned} \tag{4.24}$$

where t_1 defines the time interval of the applied impact force.

For the parameters shown in Table 4.3, MATLAB *ODE45* is used to obtain the numerical solution which, in the absence of an analytical solution, is treated as the reference solution for Eq. (4.24).

Table 4.3: Forced Duffing System Paramaters Values, [95]

Parameter	**Value**
t_0	0
t_1	1
t_F	500
$x_1(0)$	0
$x_2(0)$	0
ω_n	1
η	1
a	2

In order to capture the effect of the impact forcing function, the dynamical system in Eq. (4.24) is solved in two intervals defined by $t_0 \leq t \leq t_1$ and $t_1 \leq t \leq t_F$. The *ODE45* (reference) solution is obtained by setting the tolerances in the numerical solver to the lowest possible values, 10^{-13}. For *RBF-Coll* [1] the number of collocation points $N = 7$ and the shaping parameter $c = 2.5$. And for *RBF-Coll* ^{2}, $N = 35$ and $c = (N-1)/4\Delta t$. Table 4.4 shows the comparison of step size, accuracy and computation time for various numerical methods. Figure 4.17 and Fig. 4.18 show the normalized state error time history for *Explicit* ^{1} and *Explicit* ^{2}, respectively. Errors associated with *RK4* ^{1,2} are shown in Fig. 4.19 and Fig. 4.20. *Newmark-β* ^{1,2} results are presented in Fig. 4.21 and Fig. 4.22. *HHT-α* ^{1,2} results are presented in Fig. 4.23 and Fig. 4.24. Results for *MCPI* ^{1} and *MCPI* ^{2} are shown in Fig. 4.25 and Fig. 4.26, respectively with the order and the number of sample points equal 6 for *MCPI* ^{1} and 10 for *MCPI* ^{2}. Figure 4.27 and Fig. 4.28 show the normalized state error time history for the present *RBF-Coll* algorithm and finally, Fig. 4.29 shows the reference solution and solutions from *Method* ^{1} for the last 5 sec. of integration.

Similar to the results obtained in the previous section, the present *RBF-Coll* algorithm is shown to maintain the high solution accuracy and the low computational cost when compared to the other existing numerical methods by varying the step size and the number of collocation points. By comparing the results from *Method* ^{1} and *Method* ^{2}, it can be seen that the step size is the major contributor to the computational cost associated with each method. In that sense *RBF-Coll* is superior to all other methods with the exception of *MCPI* as there is no need to take smaller steps in order to achieve higher solution accuracy. Thus, *RBF-Coll* may be a useful method to study a periodic and chaotic responses in nonlinear dynamical systems.

Table 4.4: Comparison of Numerical Methods, Forced Duffing Oscillator

Method	$\boldsymbol{\Delta} t_1$ (sec)	$\boldsymbol{\Delta} t_2$ (sec)	**Position RMS Error**	**Simulation Time (sec)**
ODE45	Variable	Variable	N/A	4.85
Explicit [1]	0.1	0.5	0.41	0.01
Explicit [2]	0.005	0.006	0.0025	0.18
RK4 [1]	0.1	0.5	0.399	0.03
RK4 [2]	0.05	0.03	2.88×10^{-6}	0.32
Newmark-β [1]	0.1	0.5	0.457	3.6
Newmark-β [2]	0.05	0.03	1.46×10^{-2}	49.25
HHT-α [1]	0.1	0.5	0.405	3.51
HHT-α [2]	0.05	0.03	2.41×10^{-2}	49.2
MCPI [1]	0.1	0.5	1.5×10^{-6}	0.18
MCPI [2]	0.5	2	2.5×10^{-6}	0.071
RBF-Coll [1]	0.1	0.5	1.19×10^{-6}	0.31
RBF-Coll [2]	0.5	6	7.98×10^{-7}	0.14

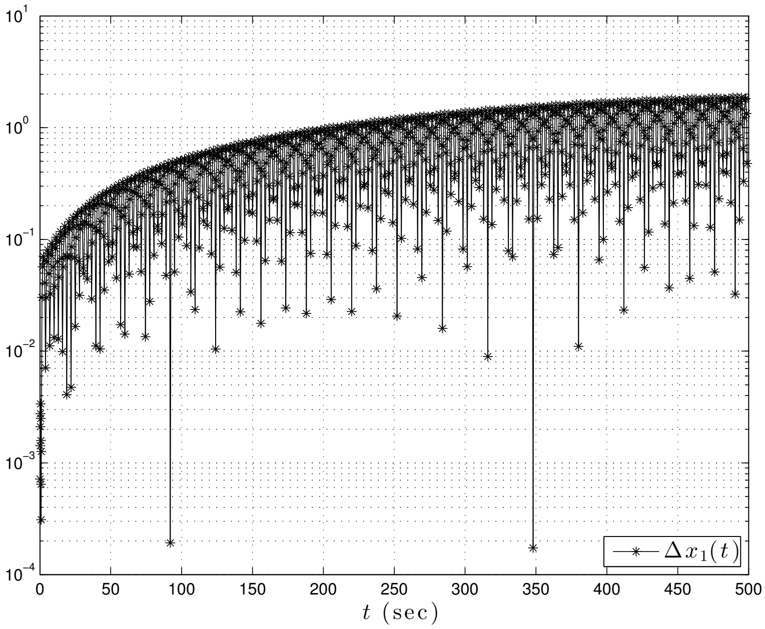

Figure 4.17: Normalized Position Error Time History, *Explicit* Method ^{1}, [95]

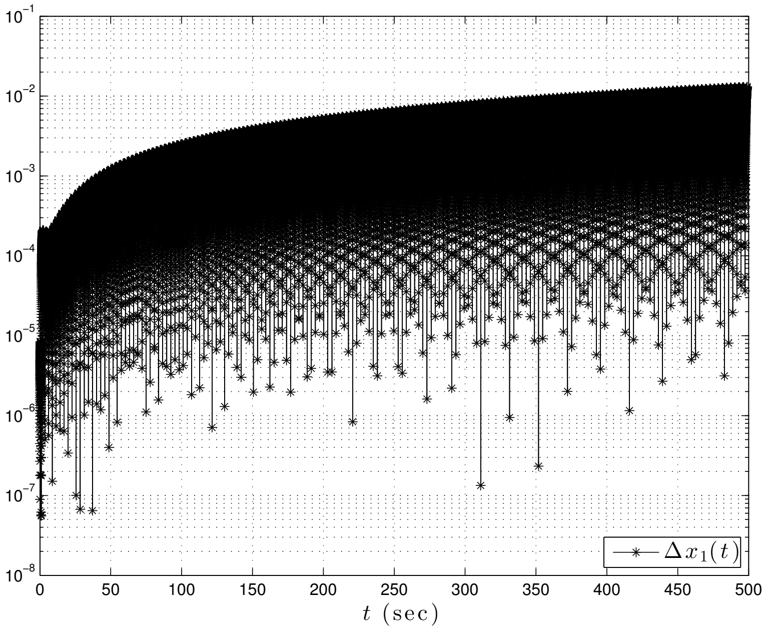

Figure 4.18: Normalized Position Error Time History, *Explicit* Method ^{2}, [95]

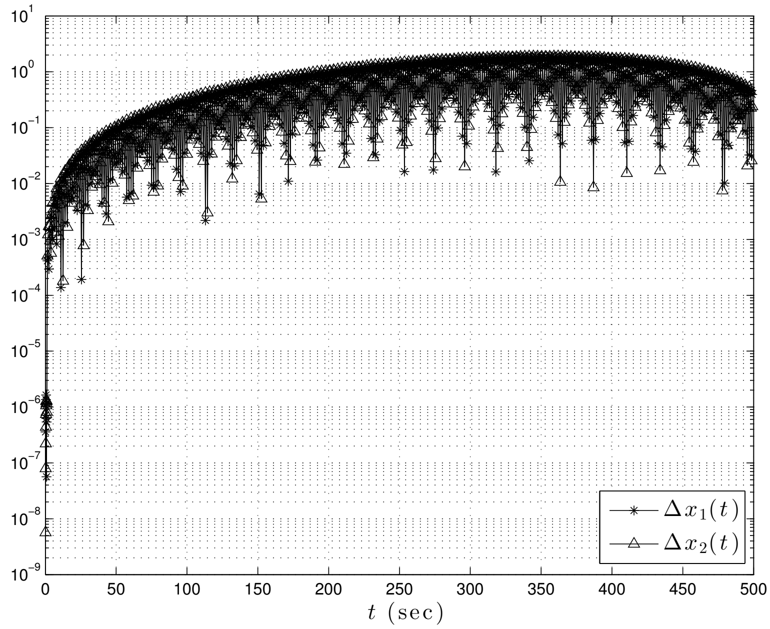

Figure 4.19: Normalized State Error Time History, *RK4* ^{1}, [95]

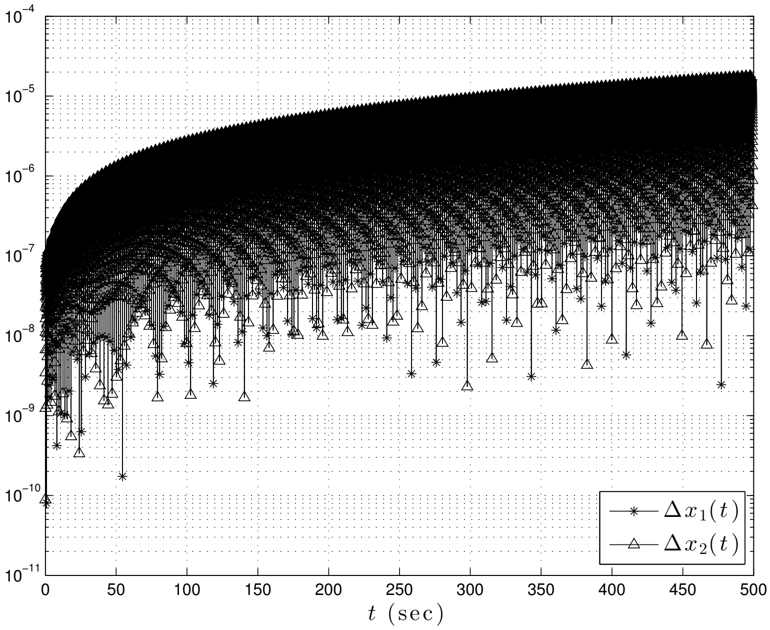

Figure 4.20: Normalized State Error Time History, *RK4* ^{2}, [95]

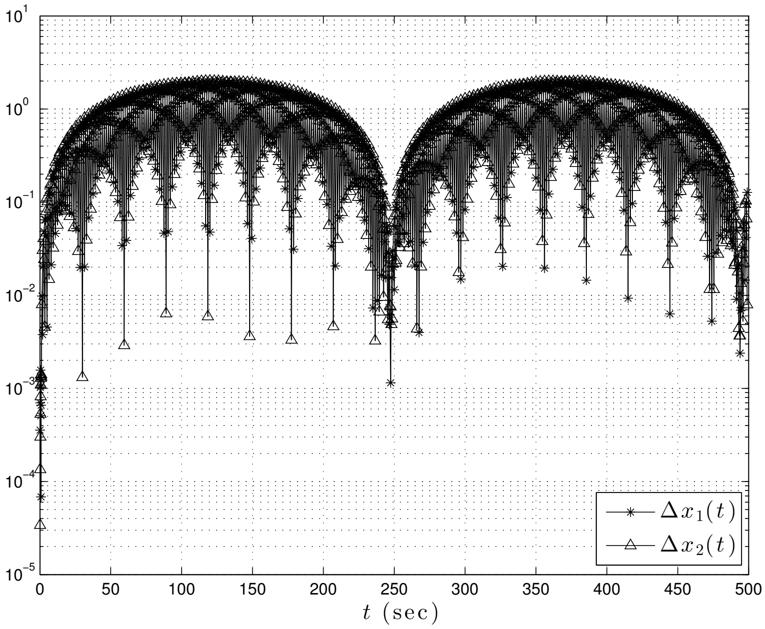

Figure 4.21: Normalized State Error Time History, *Newmark-β* ^{1}, [95]

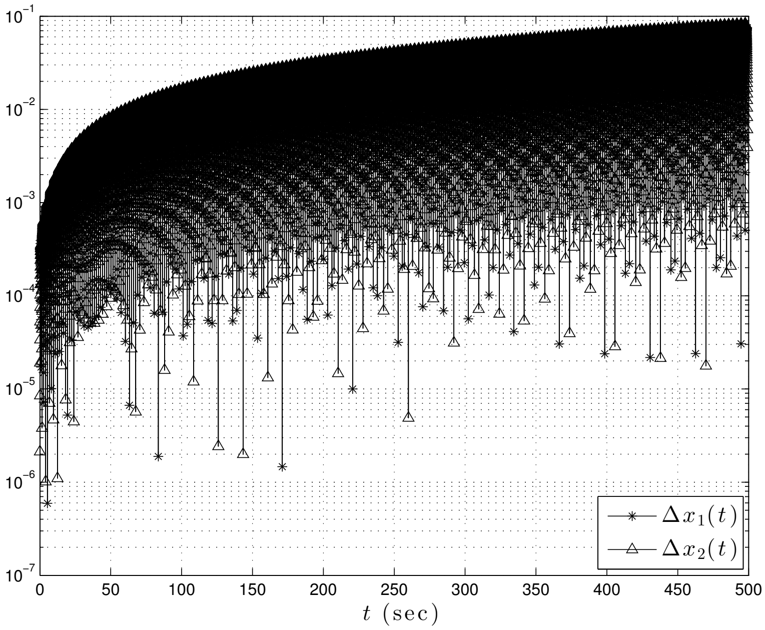

Figure 4.22: Normalized State Error Time History, *Newmark-β* ^{2}, [95]

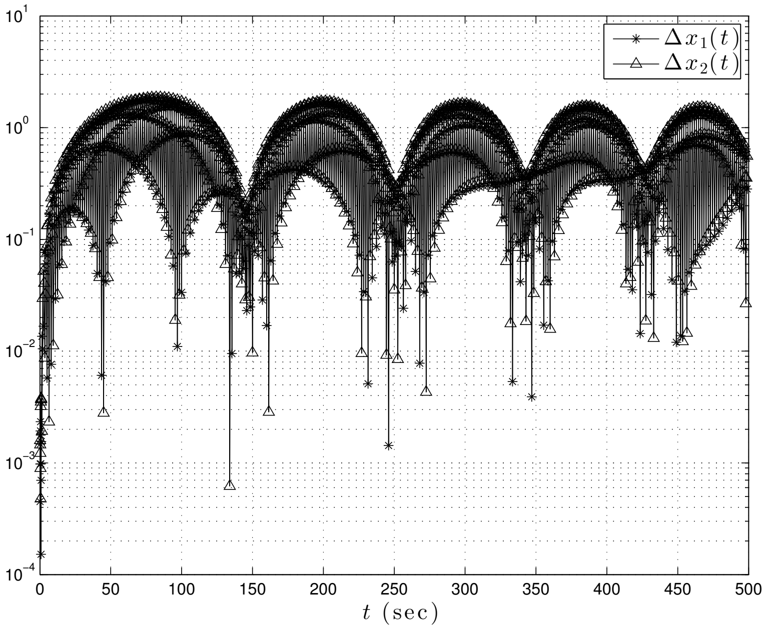

Figure 4.23: Normalized State Error Time History, *HHT*-α^1, [95]

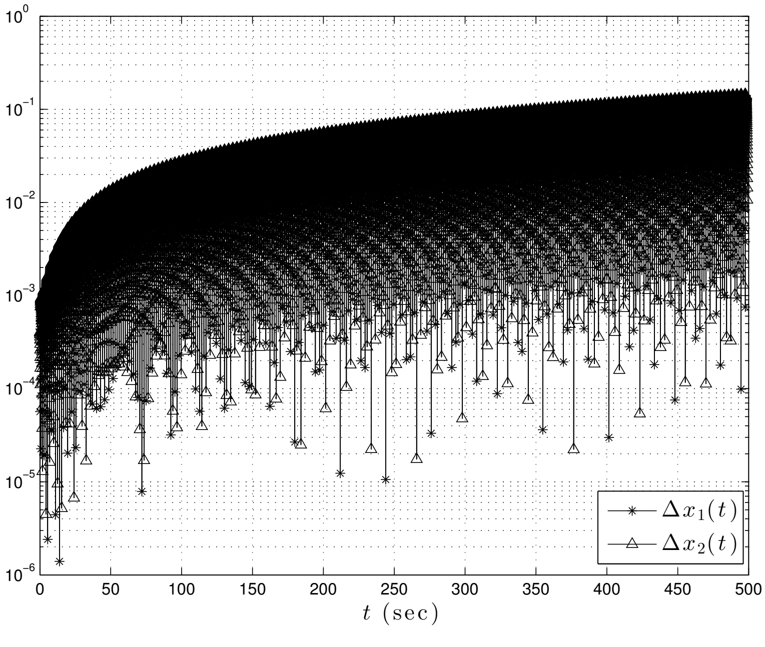

Figure 4.24: Normalized State Error Time History, *HHT*-α^2, [95]

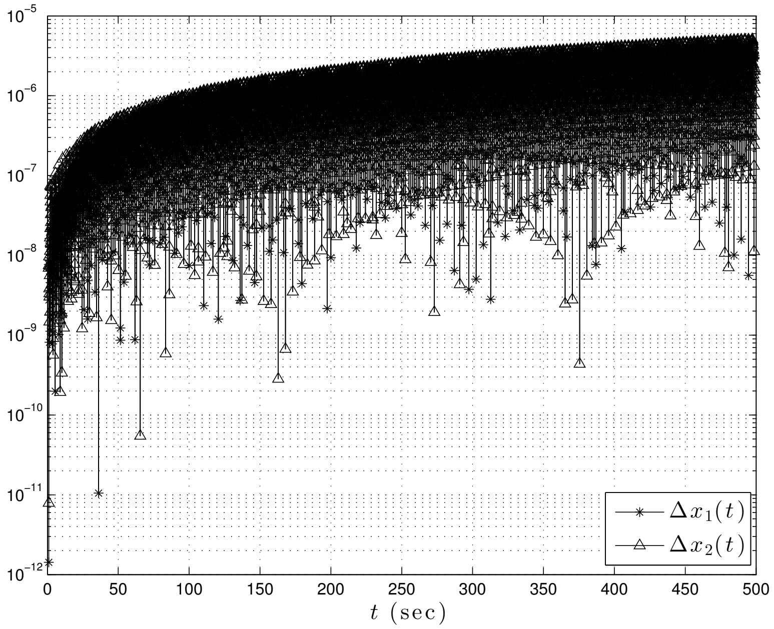

Figure 4.25: Normalized State Error Time History, *MCPI* ^{1}

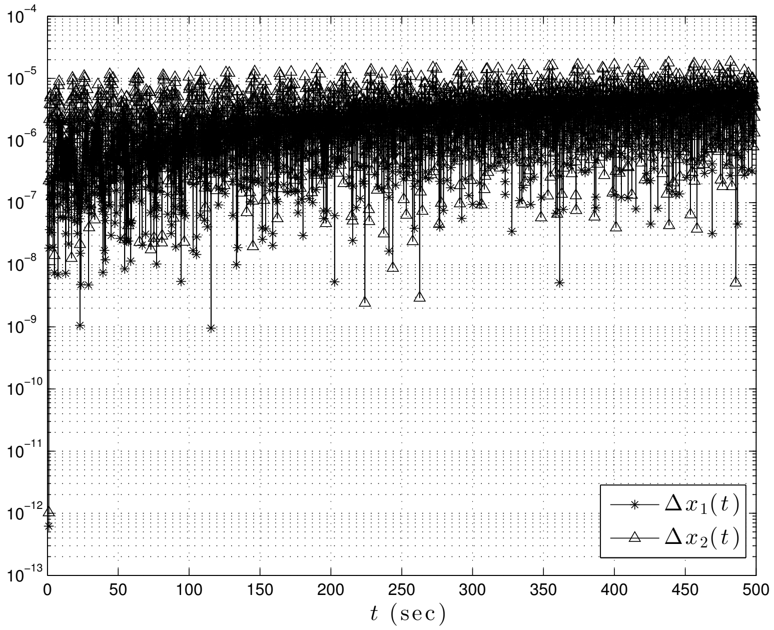

Figure 4.26: Normalized State Error Time History, *MCPI* ^{2}

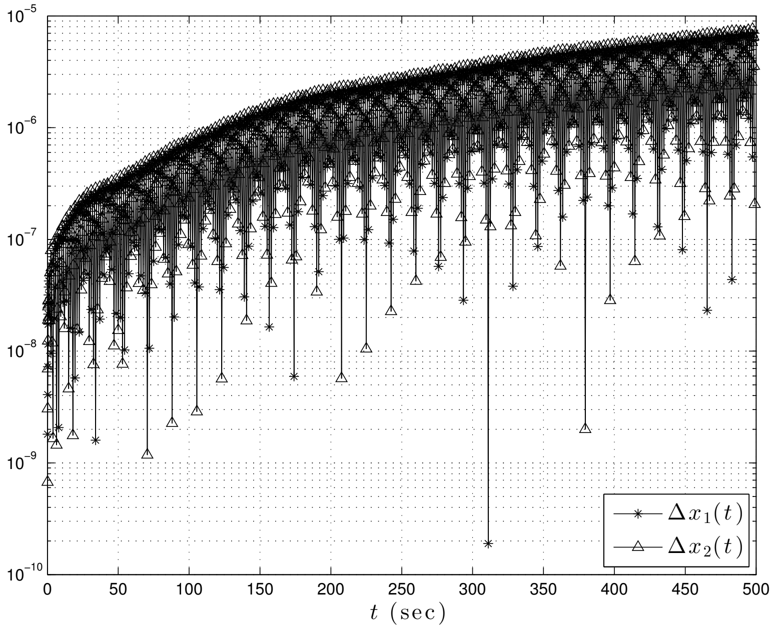

Figure 4.27: Normalized State Error Time History, *RBF-Coll* [1], [95]

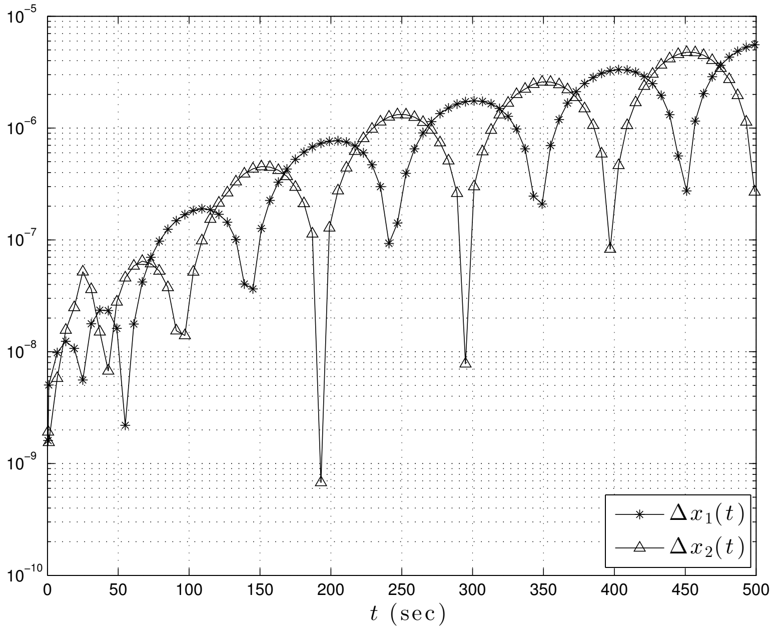

Figure 4.28: Normalized State Error Time History, *RBF-Coll* [2], [95]

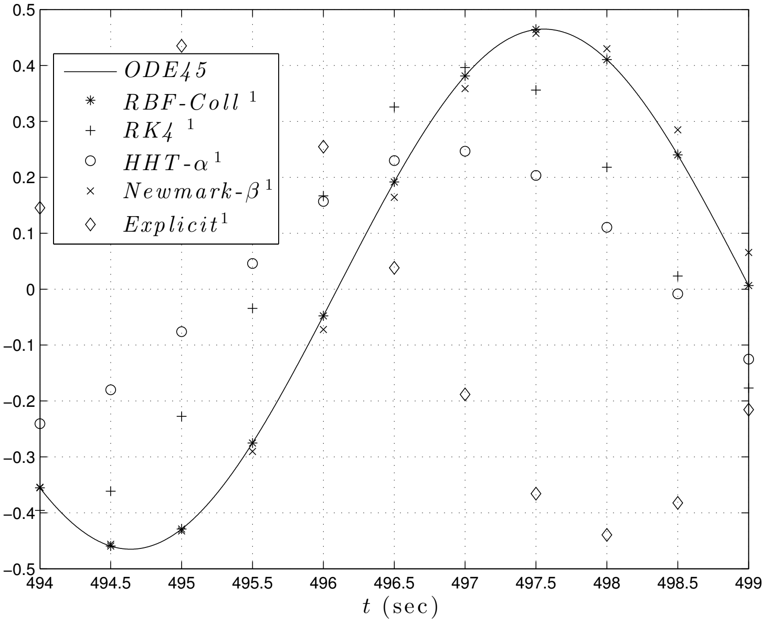

Figure 4.29: Solution Comparison, Last 5 sec. of Integration, [95]

4.4.3 Multi-DOF Highly Nonlinear Duffing Oscillator

The 3-DOF coupled nonlinear system shown in Fig. 4.30 is analyzed in this section.

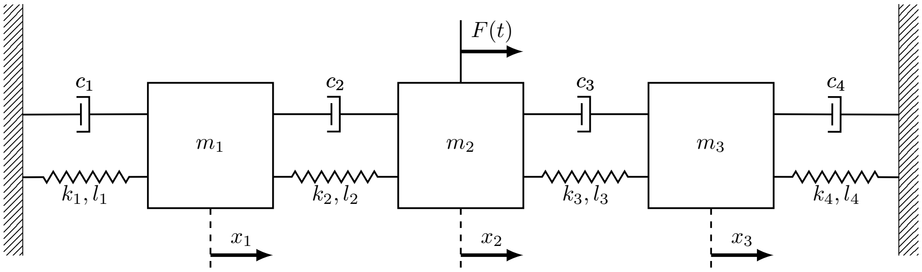

Figure 4.30: 3-DOF Nonlinear System, [95]

With $F(t) = F \cos \Omega t$, the set of 3 coupled ODEs for this nonlinear system are,

$$\begin{aligned} &m_1\ddot{x}_1 + c_1x_1 + k_1x_1 + l_1x_1^3 + k_2(x_1 - x_2) + c_2(\dot{x}_1 - \dot{x}_2) + l_2(x_1 - x_2)^3 = 0 \\ &m_2\ddot{x}_2 - c_2(\dot{x}_2 - \dot{x}_2) - k_2(x_1 - x_2) - l_2(x_1 - x_2)^3 + c_3(\dot{x}_2 - \dot{x}_3) \\ &\qquad + k_3(x_2 - x_3) + l_3(x_2 - x_3)^3 = F \cos \Omega t \\ &m_3\ddot{x}_3 - c_3(\dot{x}_2 - \dot{x}_3) - k_3(x_2 - x_3) - l_3(x_2 - x_3)^3 + c_4\dot{x}_3 + k_4x_3 + l_4x_3^3 = 0 \end{aligned} \tag{4.25}$$

The parameters selected for the 3 DOF system are given in Table 4.5. The reference solution for this coupled nonlinear system, as in the previous section, is obtained with MATLAB *ODE45* and shwon in Fig. 4.31. The comparison between various numerical integrators is given in Table 4.6. For *RBF-Coll* ^{1}, the number of collocation points $N = 7$ and the shaping parameter $c = 0.35$ and for *RBF-Coll* ^{2}, $N = 35$ and $c = 0.39$. Figure 4.32 through Fig. 4.49 show the normalized state error time history for *Explicit* ^{1,2}, *RK4* ^{1,2}, *Newmark-*β ^{1,2}, *HHT-*α ^{1,2} and *MCPI* ^{1,2}. For *MCPI* ^{1} the

Table 4.5: Paramaters for the 3-DOF Coupled System, [95]

Parameter	**Value**
t_0	0
t_F	200
$\boldsymbol{x}_0$	**0**
m_1	2
m_2	1
m_3	0.5
c_1	0
c_2	0.05
c_3	0
c_4	0
k_1	2
k_2	1
k_3	0.5
k_4	0
l_1	0.2
l_2	0
l_3	0
l_4	0

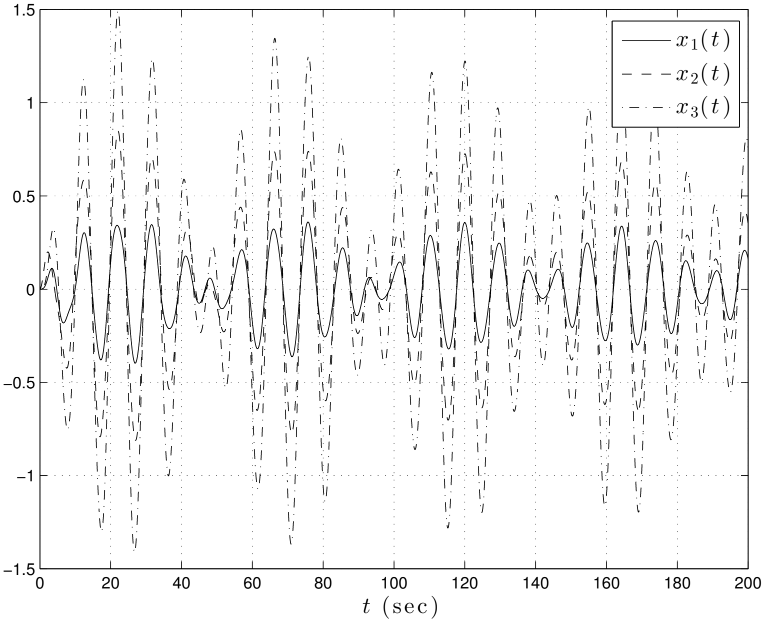

Figure 4.31: Dynamical response of the 3-DOF System, [95]

order and the number of sample points used are both 6 and for *MCPI* [2] 12. The *RBF-Coll* normalized state error time history is shown in Fig. 4.50 through Fig. 4.53. Finally, Fig. 4.54, Fig. 4.55 and Fig. 4.56 show the reference solution and *Method* [1] solutions for the last 10 sec. of integration for the three position states, $x_1(t), x_2(t)$ and $x_3(t)$, respectively.

Table 4.6: Comparison of Numerical Methods, 3-DOF System

Method	$\boldsymbol{\Delta}t$ (sec)	$\boldsymbol{x}_1(t)$ **RMS Error**	**Simulation Time (sec)**
ODE45	Variable	N/A	1.64
Explicit [1]	0.625	0.131	0.034
Explicit [2]	0.00625	1.44×10^{-5}	2.49
RK4 [1]	0.625	4.6×10^{-3}	0.017
RK4 [2]	0.0625	5.1×10^{-7}	0.1
Newmark-β [1]	0.625	6.64×10^{-2}	1.42
Newmark-β [2]	0.0625	1.2×10^{-3}	12.76
HHT-α [1]	0.625	7.6×10^{-2}	1.42
HHT-α [2]	0.0625	1.5×10^{-3}	13.15
MCPI [1]	0.625	5.8×10^{-9}	0.18
MCPI [2]	4	2.75×10^{-9}	0.082
RBF-Coll [1]	0.625	2.01×10^{-8}	0.282
RBF-Coll [2]	20	4.5×10^{-9}	0.095

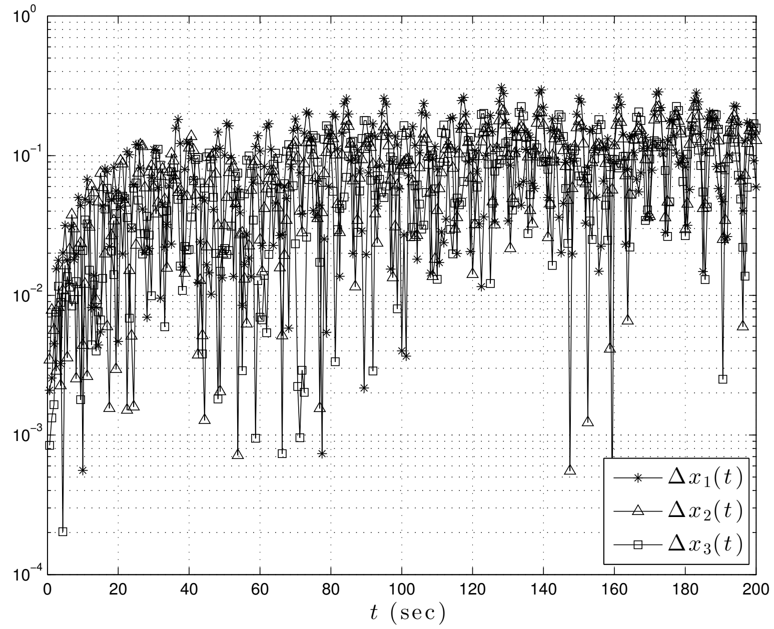

Figure 4.32: Normalized Position Error Time History, *Explicit* Method [1], [95]

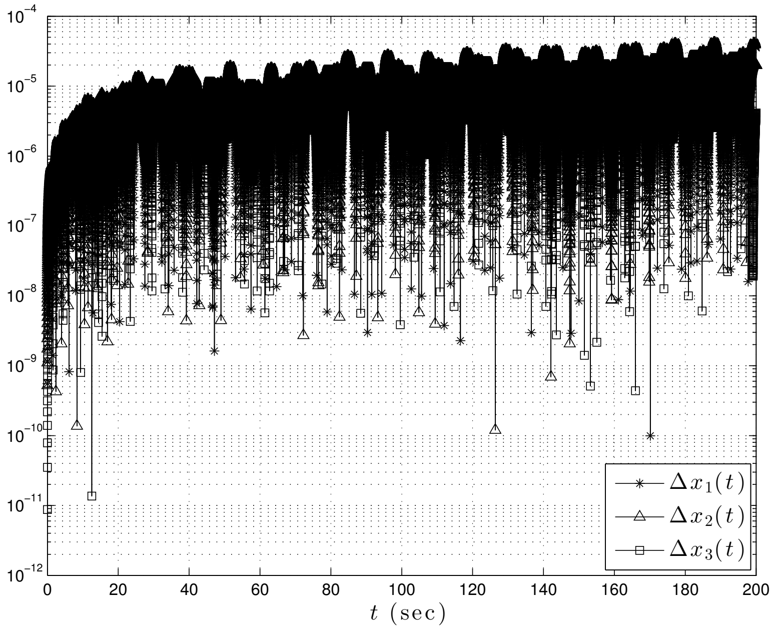

Figure 4.33: Normalized Position Error Time History, *Explicit* Method [2], [95]

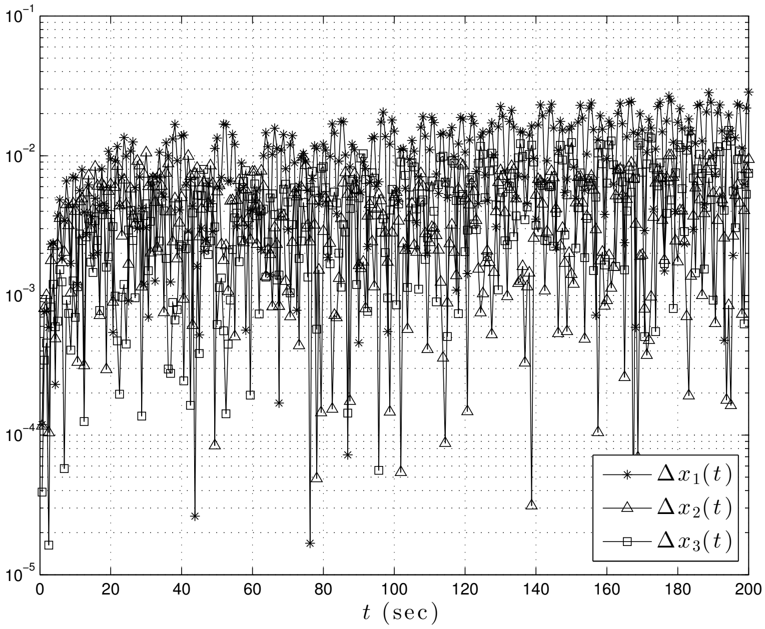

Figure 4.34: Normalized Position Error Time History, *RK4* [1], [95]

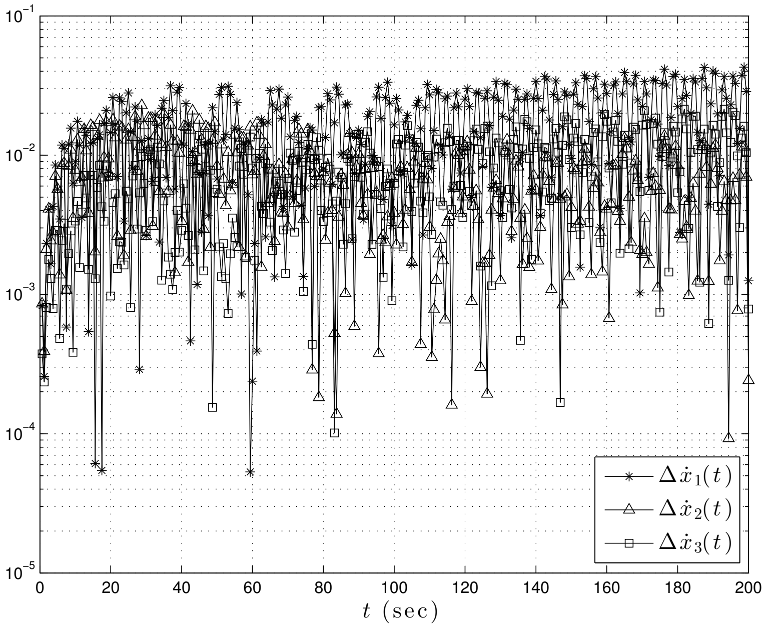

Figure 4.35: Normalized Velocity Error Time History, *RK4* [1], [95]

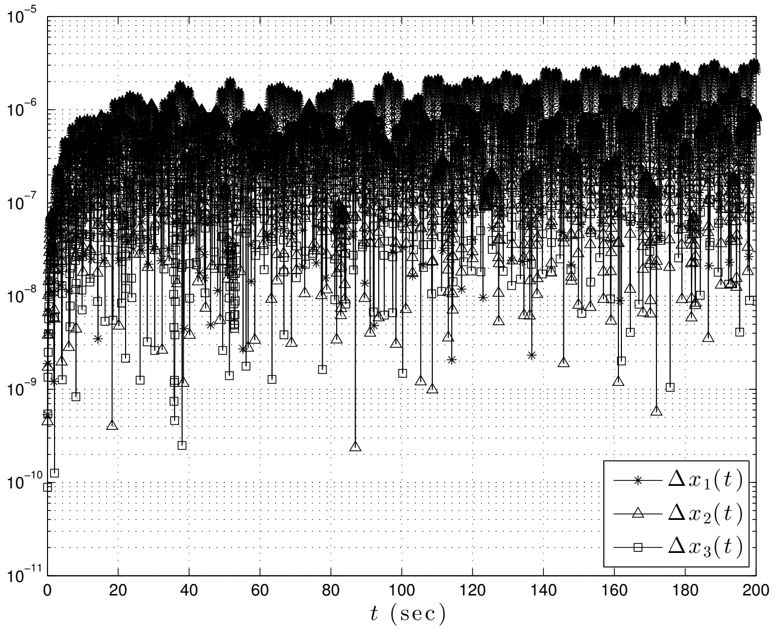

Figure 4.36: Normalized Position Error Time History, *RK4* ^{2}, [95]

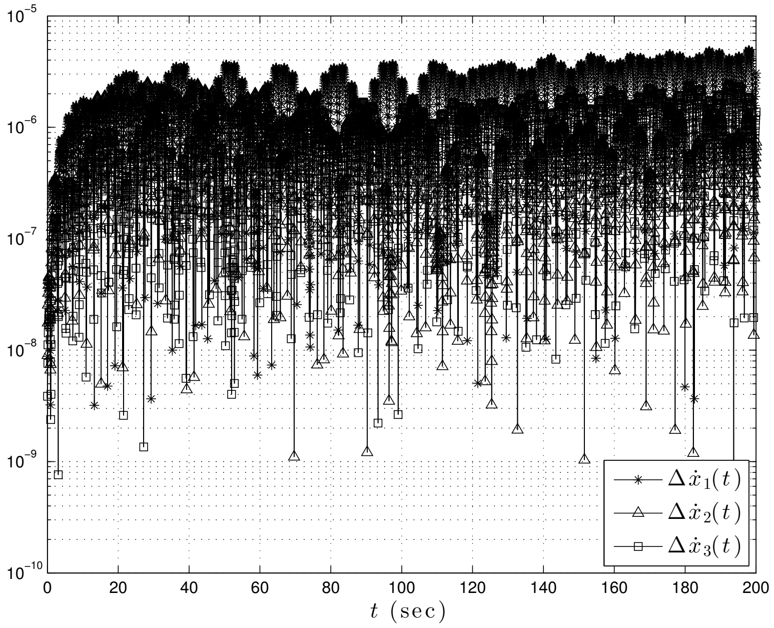

Figure 4.37: Normalized Velocity Error Time History, *RK4* ^{2}, [95]

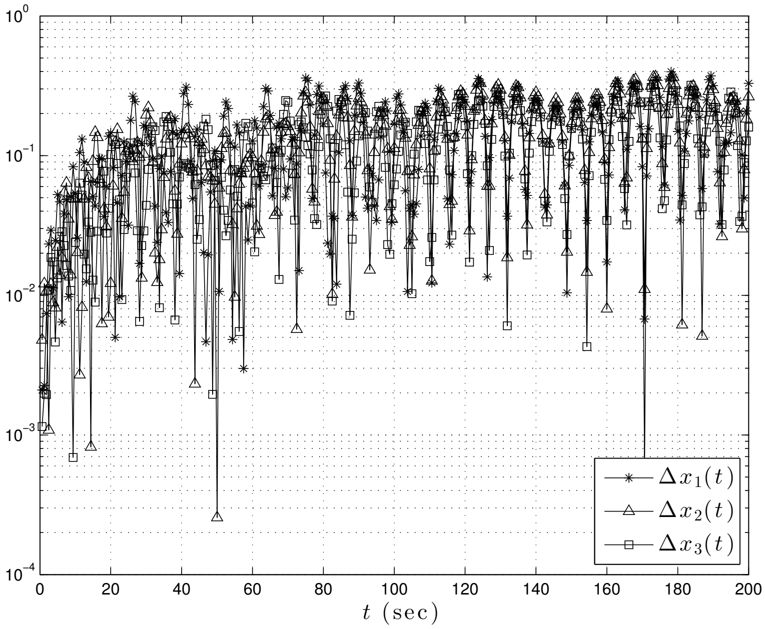

Figure 4.38: Normalized Position Error Time History, *Newmark-*β [1], [95]

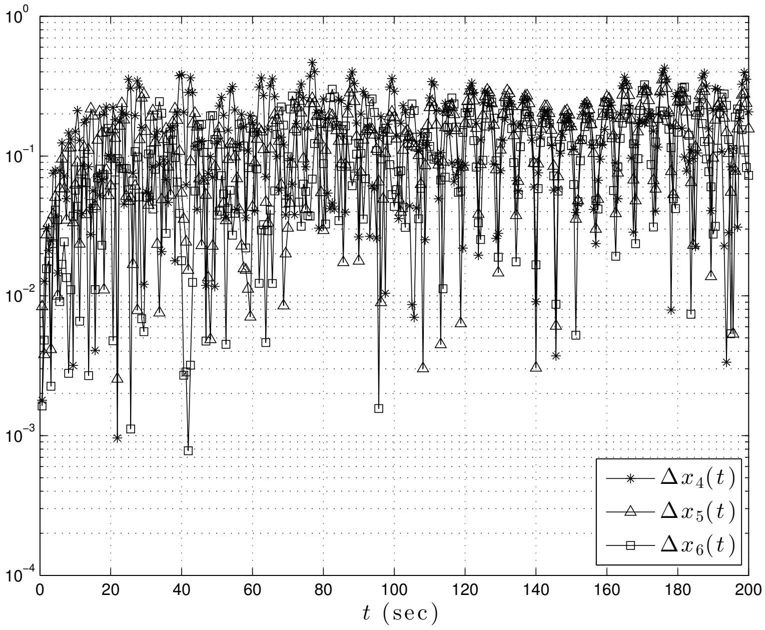

Figure 4.39: Normalized Velocity Error Time History, *Newmark-*β [1], [95]

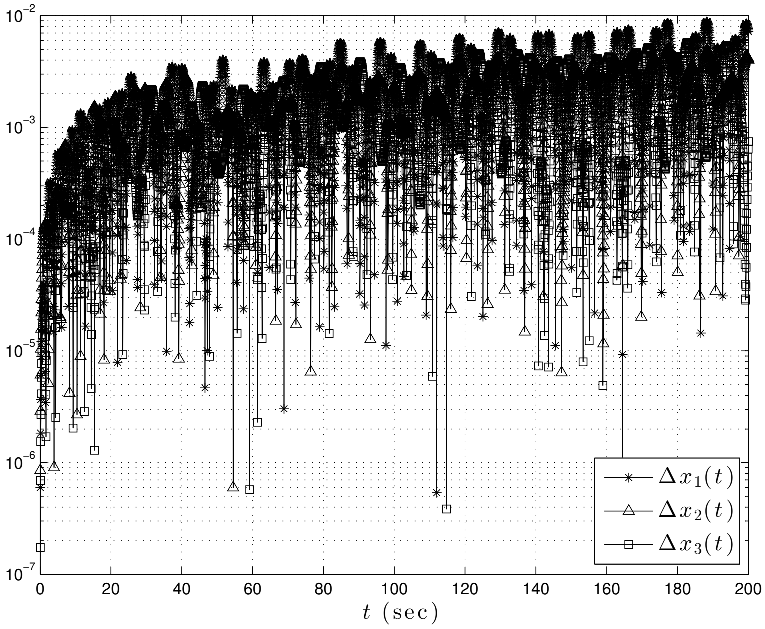

Figure 4.40: Normalized Position Error Time History, *Newmark*-β^2, [95]

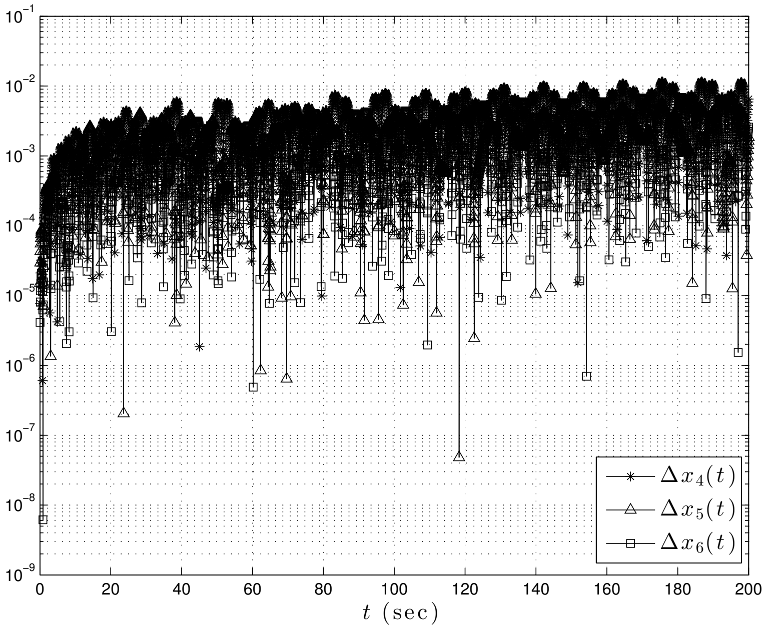

Figure 4.41: Normalized Velocity Error Time History, *Newmark*-β^2, [95]

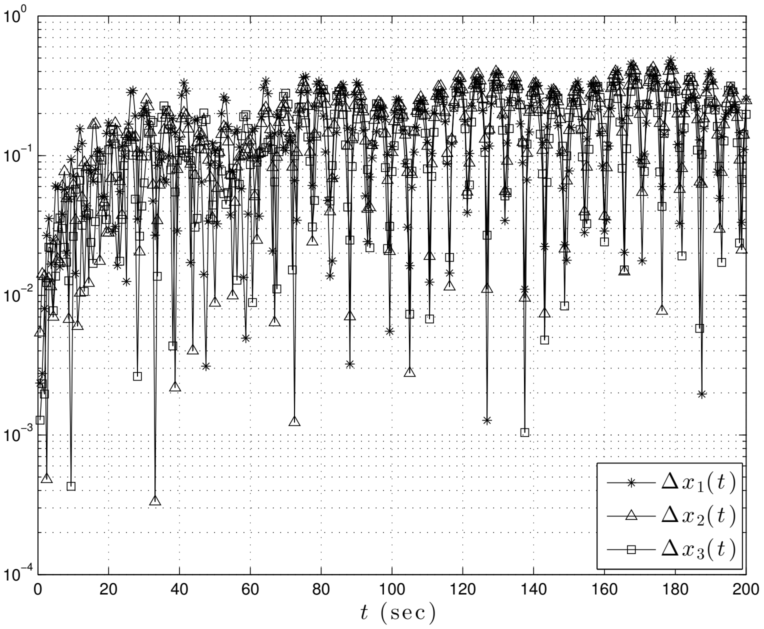

Figure 4.42: Normalized Position Error Time History, HHT-α^1, [95]

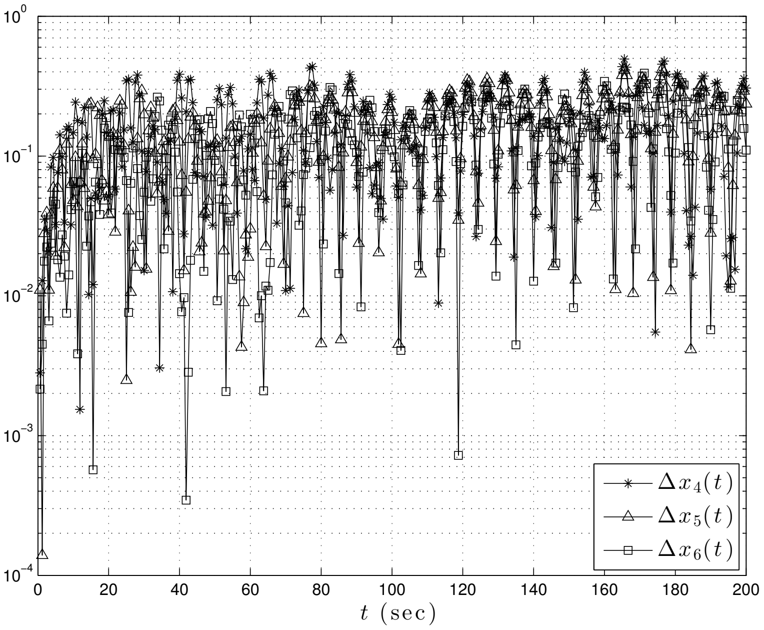

Figure 4.43: Normalized Velocity Error Time History, HHT-α^1, [95]

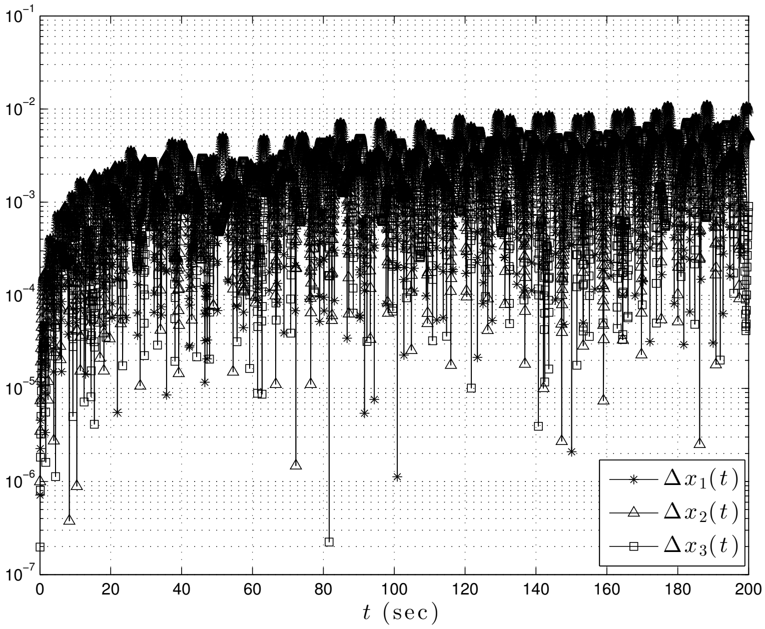

Figure 4.44: Normalized Position Error Time History, HHT-α^2, [95]

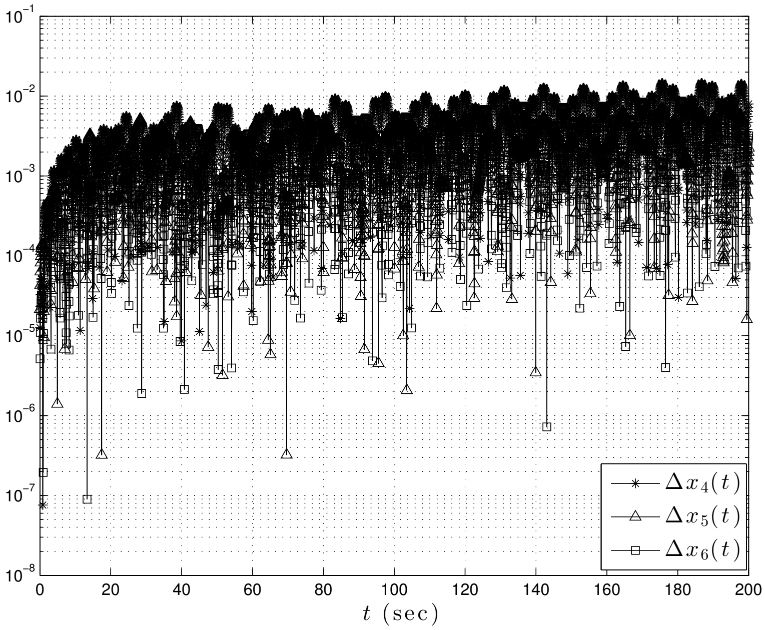

Figure 4.45: Normalized Velocity Error Time History, HHT-α^2, [95]

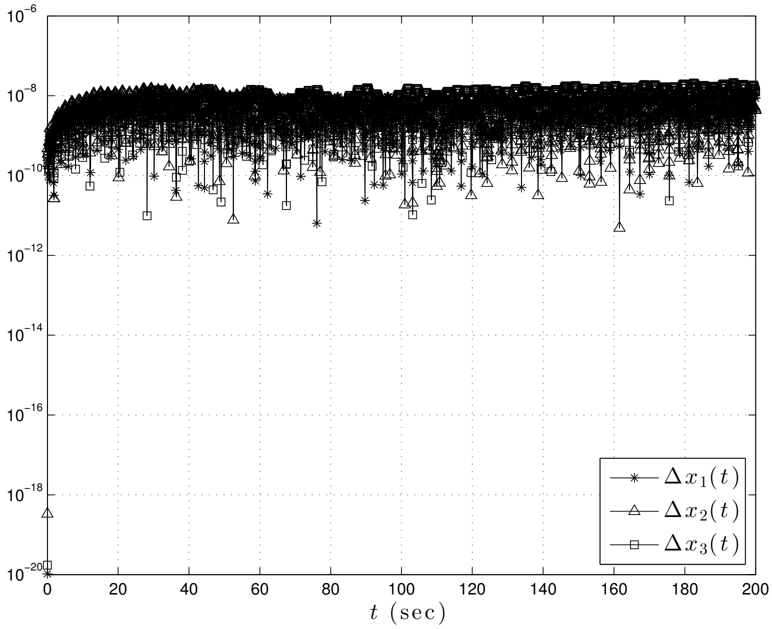

Figure 4.46: Normalized Position Error Time History, *MCPI* [1]

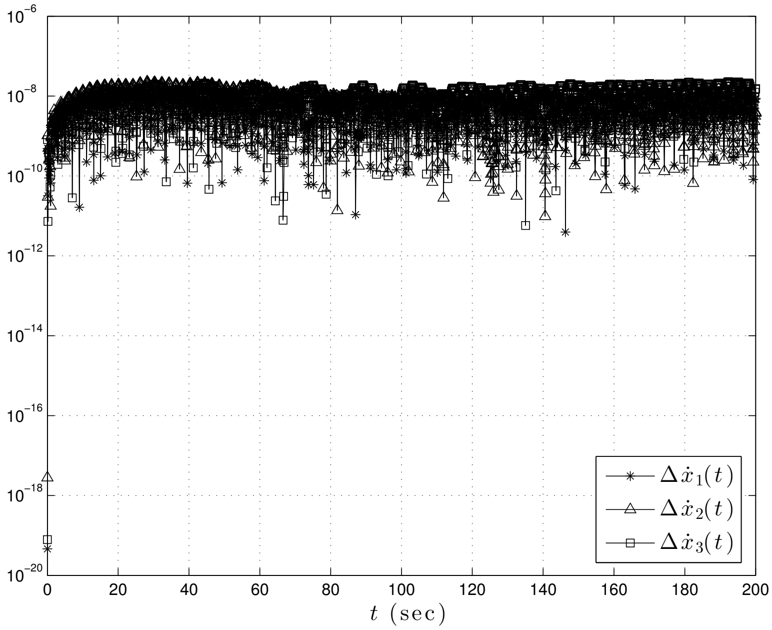

Figure 4.47: Normalized Velocity Error Time History, *MCPI* [1]

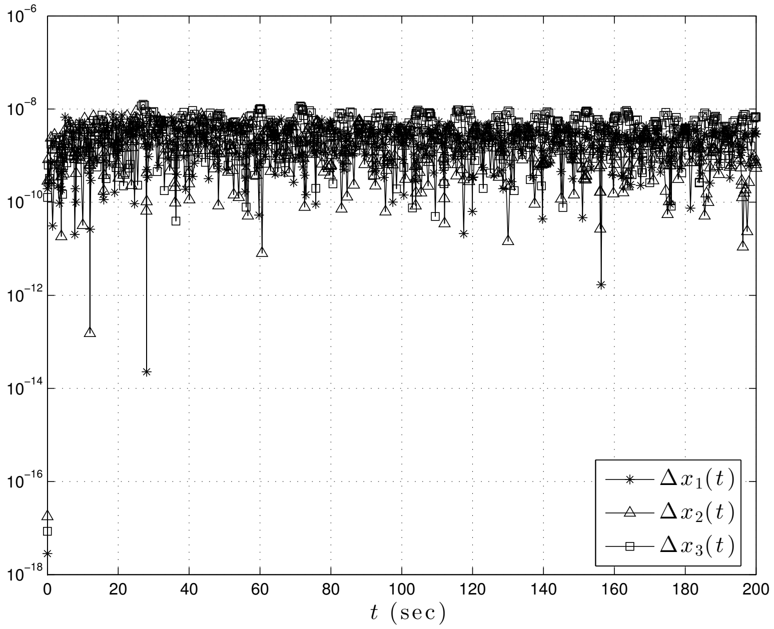

Figure 4.48: Normalized Position Error Time History, *MCPI* ^{2}

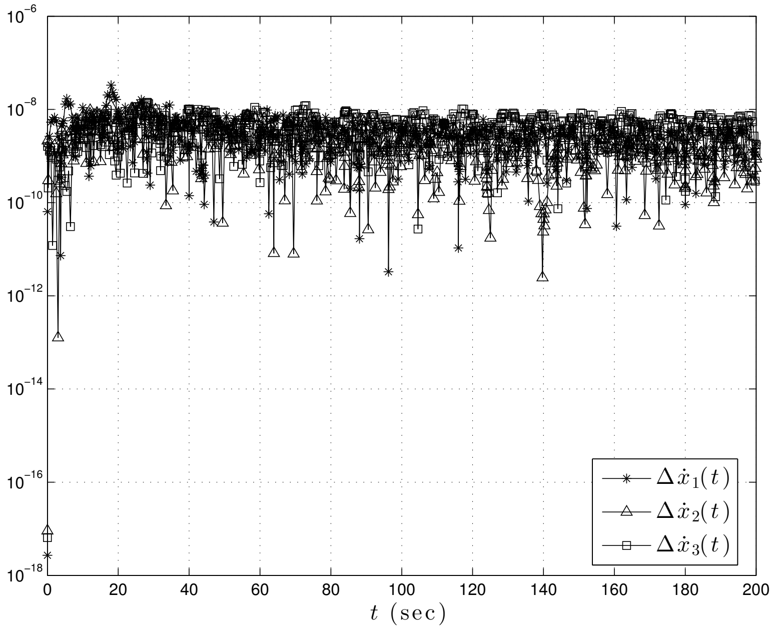

Figure 4.49: Normalized Velocity Error Time History, *MCPI* ^{2}

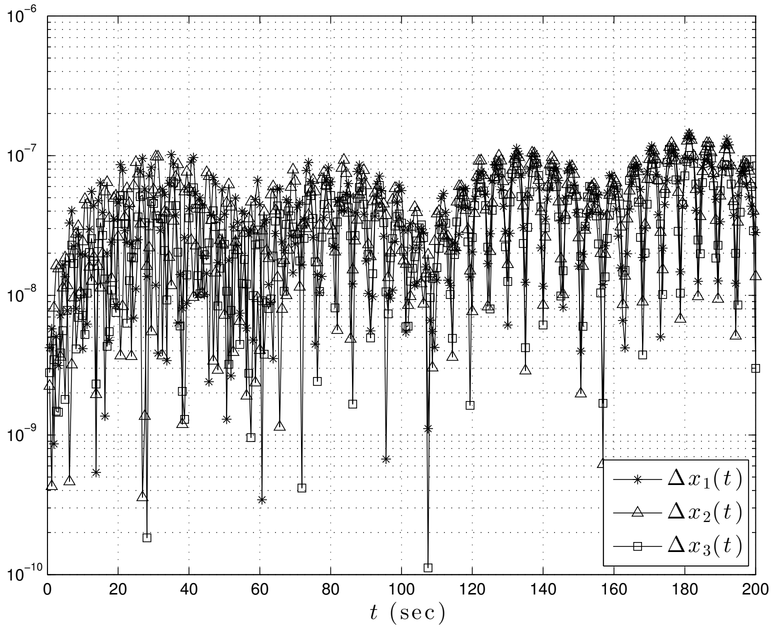

Figure 4.50: Normalized Position Error Time History, *RBF-Coll* [1], [95]

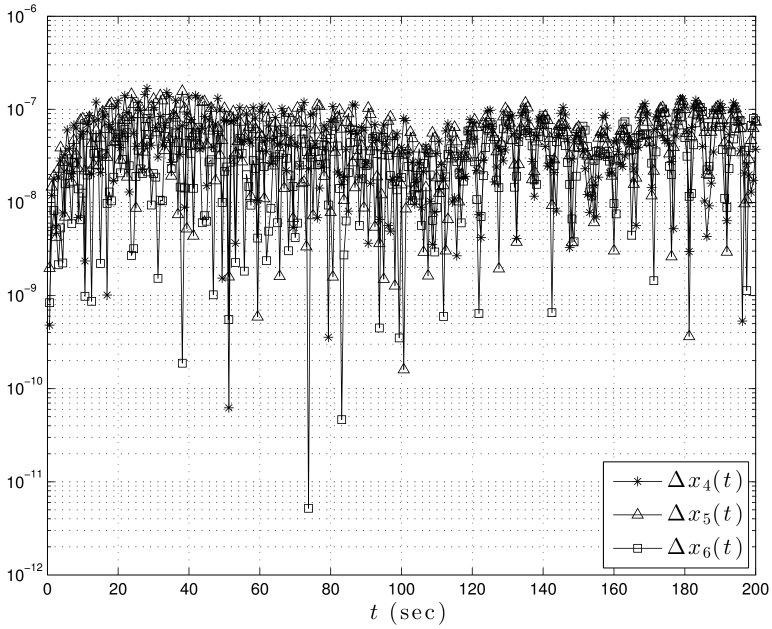

Figure 4.51: Normalized Velocity Error Time History, *RBF-Coll* [1], [95]

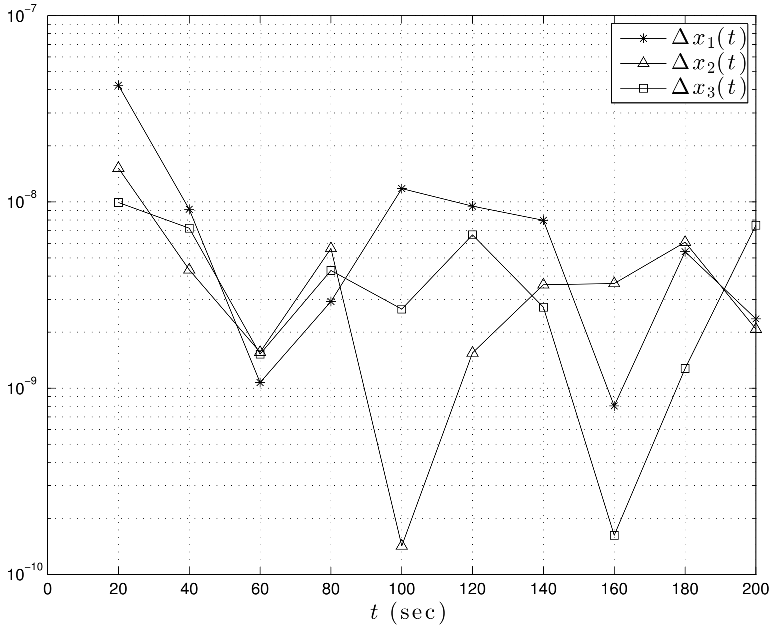

Figure 4.52: Normalized Position Error Time History, *RBF-Coll* ^{2}, [95]

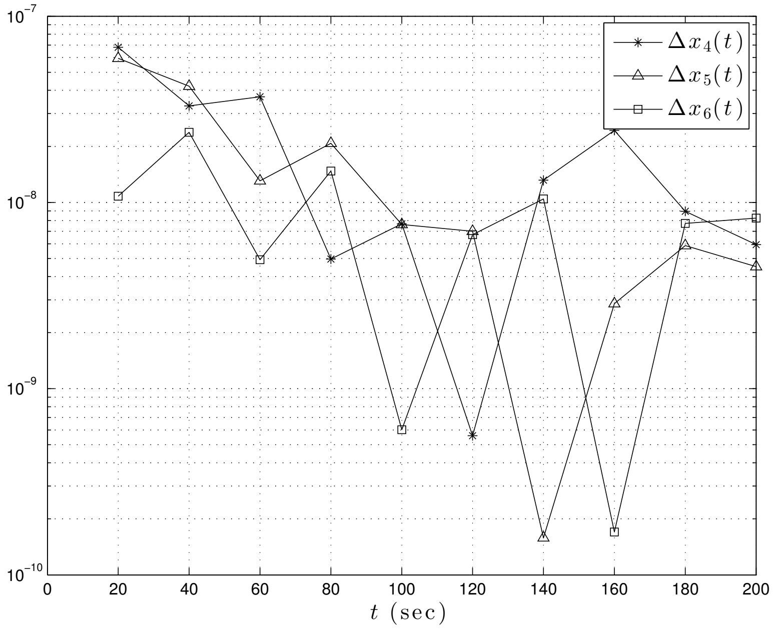

Figure 4.53: Normalized Velocity Error Time History, *RBF-Coll* ^{2}, [95]

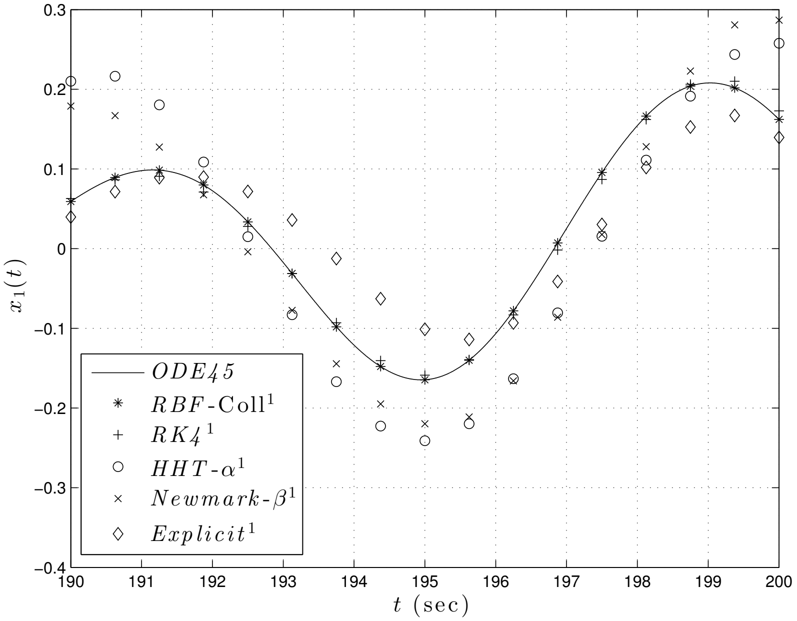

Figure 4.54: Solution Comparison for $x_1(t)$, $190 \le t \le 200$, [95]

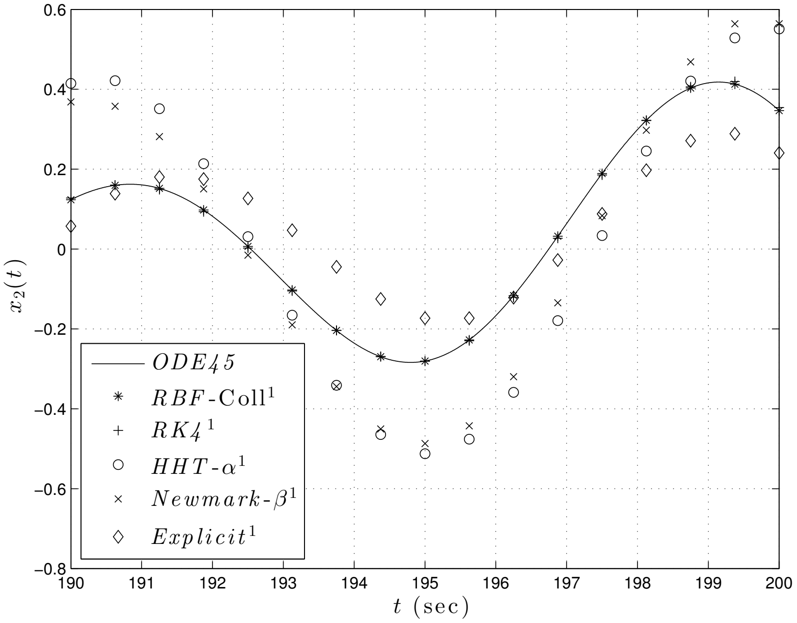

Figure 4.55: Solution Comparison for $x_2(t)$, $190 \le t \le 200$, [95]

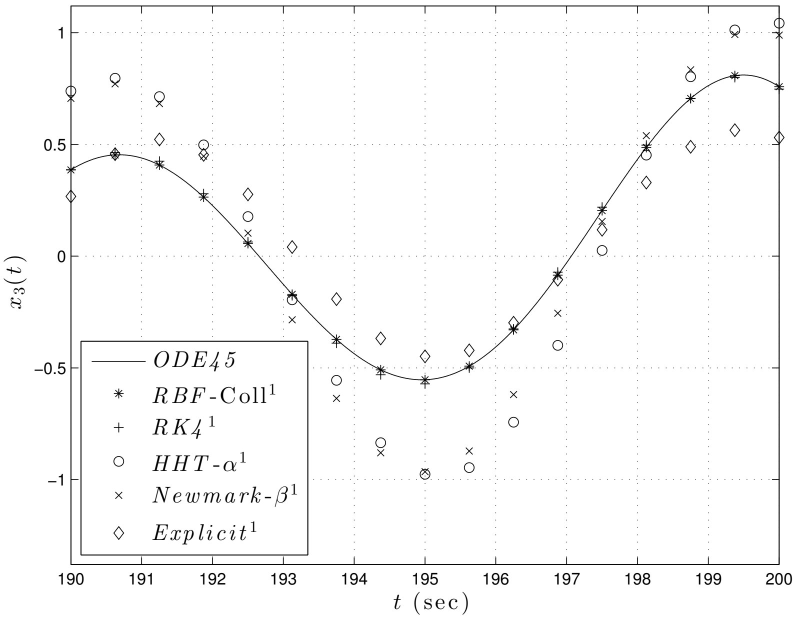

Figure 4.56: Solution Comparison for $x_3(t)$, $190 \leq t \leq 200$, [95]

As in the previous two sections the present *RBF-Coll* algorithm has the best combination of accuracy and computational cost which allows the user to take a larger time step and maintain a very high solution accuracy. *MCPI* algorithm shows very comparable results to *RBF-Coll*, however, the conducted numerical experiments are not able to increase the step size beyond the shown value for *MCPI* [2]. In general, the computational cost of *RBF-Coll* is very comparable to the fastest, and as a consequence least accurate methods, which gives the algorithm a significant competitive advantage among various numerical integrators of general dynamical systems.

4.5 Summary

The present *RBF-Coll* algorithm is shown to be highly accurate, fast and very simple to implement for various types of dynamical systems. Comparing the algorithm versus several implicit and explicit numerical integration methods clearly shows the advantages of such an algorithm that enables larger time step, high solution accuracy while maintaining a relatively low computational cost. *RBF-Coll* combination of accuracy and computational cost is superior to all presented classical algorithms. Furthermore, when compared with *MCPI*, *RBF-Coll* is achieving comparable accuracy at almost the same computational cost. The *RBF-Coll* algorithm is generally better suited for adaptation schemes as there is no need to meet orthogonality conditions which relax the requirements on the node distribution. The algorithm is shown to accurately and simply handle short and long periods of time integration, single and multi degrees of freedom system and periodic and finally transient and periodic solutions. *RBF-Coll* algorithm can have a significant potential in handling various types of dynamical systems governed by second or higher order differential equations. Applications of algorithm are, but not limited to, orbit propagation in celestial mechanics, dynamic buckling problems and optimal control and two-point boundary value problems. Areas of study that will be explored in future works.

It must be noted that the *RBF-Coll* algorithm can be considered as an implicit Runge-Kutta algorithm where radial basis functions are used instead of Legendre or Chebyshev polynomials as developed in [120–122]. The Legendre and/or Chebyshev basis functions limit the usage of the nodal distribution to Gauss-quadrature distribution to meet orthogonality conditions for the basis functions and avoid a rank deficient basis functions matrix. Radial basis functions do not need to meet orthogonality conditions and while LGL node distribution has been used in this work the selection of the number of nodes, their concentration and distribution is a topic to be explored in future studies to develop an adaptive self-tuning algorithm to handle general nonlinear dynamical systems.

5. RADIAL BASIS FUNCTIONS DIRECT TIME-DOMAIN COLLOCATION APPLIED TO ILL-POSED PROBLEMS[*]

In this chapter, the *RBF-Coll* algorithm used earlier for the Initial Value Problem (IVP) is used to address ill-posed time-domain inverse problems for dynamical systems with split boundary conditions and unknown controllers. The duffing optimal control problem with various prescribed initial and final conditions, as well as the orbital transfer Lambert's problem are solved by the proposed *RBF-Coll* method as examples. It is shown that this method is very simple, efficient and very accurate in obtaining the solutions, with an arbitrary solution as the initial guess. Since other methods such as the shooting method and the pseudo-spectral method can be unstable and requires a close initial guess, the proposed method is advantageous and have promising applications in optimal control and celestial mechanics, [123].

5.1 Introduction

As previously introduced, a second-order system of nonlinear ordinary differential equations (ODEs) can, in general, be recast as a system of first-order ODEs as,

$$\dot{\boldsymbol{x}} = \boldsymbol{g}\left(\boldsymbol{x}, \boldsymbol{f}, t\right), \qquad t_0 \leq t \leq t_F \tag{5.1}$$

where $\boldsymbol{x}$ is the vector of mixed variables, $\boldsymbol{x} \equiv [\boldsymbol{x}_1, \boldsymbol{x}_2]^{\mathrm{T}}$, $\dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2$, $\boldsymbol{f}$ is the force applied to the system and t is the time with t_0 and t_F the initial and final time, respectively. For a specified set of initial conditions $\boldsymbol{x}_0$ and being given the force function $\boldsymbol{f}(t)$, the initial value problem (IVP) of Eq. (5.1) is well-posed and the

[*] Part of this chapter is reprinted with permission from "Time Domain Inverse Problems in Nonlinear Systems Using Collocation & Radial Basis Functions" by Elgohary, T. A., Dong, L., Junkins, J. L. and Atluri, S. N., 2014. *CMES: Computer Modeling in Engineering & Sciences*, Vol. 100, No. 1, pp. 59–84, Copyright [2014] by Tech Science Press.

solution methodologies are well understood, [95]. On the other hand, the inverse problem requiring the solution of the unknown initial conditions, given a prescribed final state, is considered ill-posed. In that sense, Eq. (5.1) is ill-posed in case of split boundary conditions when both $\boldsymbol{x}_1(t_0)$ and $\boldsymbol{x}_1(t_F)$ are prescribed; or when $\boldsymbol{x}_1(t_0)$ and $\boldsymbol{x}_2(t_F)$ are both prescribed; or when $\boldsymbol{x}_2(t_0)$ and $\boldsymbol{x}_1(t_F)$ are both prescribed; or when $\boldsymbol{x}_2(t_0)$ and $\boldsymbol{x}_2(t_F)$ are both prescribed; or when $\boldsymbol{f}(t)$ is an unknown function to be solved for.

For the majority of applications, $\boldsymbol{f}(t)$ can not be arbitrary and a minimization scheme is introduced to meet a set of engineering requirements. Among those class of problems is the given-time-interval optimal control problem found in several classical works, [2, 124]. The problem is generally given as,

$$\begin{aligned} &\text{Min:} \quad J = \phi(\boldsymbol{x}(t_0), \boldsymbol{x}(t_F), t_0, t_F) + \int_{t_0}^{t_F} L(\boldsymbol{x}, \boldsymbol{f}, t)\mathrm{d}t \\ &\text{Subject to:} \quad \dot{\boldsymbol{x}} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t), \qquad t_0 \le t \le t_F \end{aligned} \tag{5.2}$$

where the objective is to minimize a prescribed performance index J along the trajectory of the system dynamics given by $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t)$. The methodology to obtain the solution for the optimal control force in Eq. (5.2) is presented in several text books, see [2], [124] and/or [125], based on the calculus of variations, by using Lagrange multipliers to obtain the augmented performance index, J_a:

$$J_a = \phi(\boldsymbol{x}(t_0), \boldsymbol{x}(t_F), t_0, t_F) + \int_{t_0}^{t_F} \left\{ L(\boldsymbol{x}, \boldsymbol{f}, t) + \boldsymbol{\lambda}^{\mathrm{T}} \left[\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t) - \dot{\boldsymbol{x}} \right] \right\} \mathrm{d}t \tag{5.3}$$

The scalar Hamiltonian function is then defined as,

$$H = L(\boldsymbol{x}, \boldsymbol{f}, t) + \boldsymbol{\lambda}^{\mathrm{T}} \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t) \tag{5.4}$$

and the augmented performance index can then be re-written as follows after integration by parts,

$$J_a = \phi(\boldsymbol{x}(t_0), \boldsymbol{x}(t_F), t_0, t_F) - \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{x}(t_F) + \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{x}(t_0) + \int_{t_0}^{t_F} \left(H + \dot{\boldsymbol{\lambda}}^{\mathrm{T}}\boldsymbol{x}\right)\,\mathrm{d}t \tag{5.5}$$

The variation of the augmented performance index, δJ_a, is then expressed in terms of the variations of $\boldsymbol{x}$, $\boldsymbol{\lambda}$, and $\boldsymbol{f}$:

$$\begin{aligned}
\delta J_a &= (\phi_{\boldsymbol{x}} - \boldsymbol{\lambda}^{\mathrm{T}})\,\delta\boldsymbol{x}(t_F) + (\phi_{\boldsymbol{x}} + \boldsymbol{\lambda}^{\mathrm{T}})\,\delta\boldsymbol{x}(t_0) - \boldsymbol{x}(t_F)^{\mathrm{T}}\delta\boldsymbol{\lambda} + \boldsymbol{x}(t_0)^{\mathrm{T}}\ \delta\boldsymbol{\lambda} \\
&\quad + \int_{t_0}^{t_F} \left[\left(H_{\boldsymbol{x}} + \dot{\boldsymbol{\lambda}}^{\mathrm{T}}\right)\delta\boldsymbol{x} + \left(H_{\boldsymbol{\lambda}}\delta\boldsymbol{\lambda} + \boldsymbol{x}^{\mathrm{T}}\delta\dot{\boldsymbol{\lambda}}\right) + H_{\boldsymbol{f}}\delta\boldsymbol{f}\right] \\
&= (\phi_{\boldsymbol{x}} - \boldsymbol{\lambda}^{\mathrm{T}})\,\delta\boldsymbol{x}(t_F) + (\phi_{\boldsymbol{x}} + \boldsymbol{\lambda}^{\mathrm{T}})\,\delta\boldsymbol{x}(t_0) \\
&\quad + \int_{t_0}^{t_F} \left[\left(H_{\boldsymbol{x}} + \dot{\boldsymbol{\lambda}}^{\mathrm{T}}\right)\delta\boldsymbol{x} + (H_{\boldsymbol{\lambda}} - \dot{\boldsymbol{x}}^{\mathrm{T}})\,\delta\boldsymbol{\lambda} + H_{\boldsymbol{f}}\delta\boldsymbol{f}\right]\mathrm{d}t
\end{aligned} \tag{5.6}$$

where $(\,)_* = \frac{\partial()}{\partial *}$. The stationarity of Eq. (5.5) necessitates vanishing of $\int_{t_0}^{t_F}[(H_{\boldsymbol{x}} + \dot{\boldsymbol{\lambda}}^{\mathrm{T}})\delta\boldsymbol{x} + (H_{\boldsymbol{\lambda}} - \dot{\boldsymbol{x}}^{\mathrm{T}})\delta\boldsymbol{\lambda} + H_{\boldsymbol{f}}\delta\boldsymbol{f}]\mathrm{d}t$, leading to the following 3 Euler-Lagrange equations:

$$\begin{aligned}
\dot{\boldsymbol{x}} &= \frac{\partial H}{\partial \boldsymbol{\lambda}^{\mathrm{T}}} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{f}, t) \\
-\dot{\boldsymbol{\lambda}} &= \frac{\partial H}{\partial \boldsymbol{x}} = \left[\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right]^{\mathrm{T}}\boldsymbol{\lambda} + \frac{\partial L}{\partial \boldsymbol{x}} \\
\boldsymbol{0} &= \frac{\partial H}{\partial \boldsymbol{f}} = \frac{\partial L}{\partial \boldsymbol{f}} + \left[\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{f}}\right]^{\mathrm{T}}\boldsymbol{\lambda}
\end{aligned} \tag{5.7}$$

And depending on what are prescribed for the states of $\boldsymbol{x}(t_0)$ and $\boldsymbol{x}(t_F)$, some complementary boundary conditions at t_0 and t_F can be obtained from the vanishing of $(\phi_{\boldsymbol{x}} - \boldsymbol{\lambda}^{\mathrm{T}})\delta\boldsymbol{x}(t_F) + \boldsymbol{\lambda}^{\mathrm{T}}\delta\boldsymbol{x}(t_0)$, as is listed in detail in Table 5.1.

Several solution techniques exist for such problems. The shooting method, see [126], is one of the most widely used approaches in the optimal control literature. Starting with an initial guess for the unknown initial conditions, the system of equa-

Table 5.1: Boundary Conditions for Various Cases of Ill-posed Problems, [123]

Prescribed boundary conditions	**Complementary boundary conditions**
Full final state prescribed, $\boldsymbol{x}(t_F)$	$\boldsymbol{\lambda}(t_0) = \mathbf{0}$
$\boldsymbol{x}_1(t_0)$ and $\boldsymbol{x}_1(t_F)$ prescribed	$-\boldsymbol{\lambda}_2^{\mathrm{T}}(t_0) = \phi_{\boldsymbol{x}_2}\|_{t_0}$, $\boldsymbol{\lambda}_2^{\mathrm{T}}(t_F) = \phi_{\boldsymbol{x}_2}\|_{t_F}$
$\boldsymbol{x}_1(t_0)$ and $\boldsymbol{x}_2(t_F)$ prescribed	$-\boldsymbol{\lambda}_2^{\mathrm{T}}(t_0) = \phi_{\boldsymbol{x}_2}\|_{t_0}$, $\boldsymbol{\lambda}_1^{\mathrm{T}}(t_F) = \phi_{\boldsymbol{x}_1}\|_{t_F}$
$\boldsymbol{x}_2(t_0)$ and $\boldsymbol{x}_1(t_F)$ prescribed	$-\boldsymbol{\lambda}_1^{\mathrm{T}}(t_0) = \phi_{\boldsymbol{x}_1}\|_{t_0}$, $\boldsymbol{\lambda}_2^{\mathrm{T}}(t_F) = \phi_{\boldsymbol{x}_2}\|_{t_F}$
$\boldsymbol{x}_2(t_0)$ and $\boldsymbol{x}_2(t_F)$ prescribed	$-\boldsymbol{\lambda}_1^{\mathrm{T}}(t_0) = \phi_{\boldsymbol{x}_1}\|_{t_0}$, $\boldsymbol{\lambda}_1^{\mathrm{T}}(t_F) = \phi_{\boldsymbol{x}_1}\|_{t_F}$

tions is integrated and matched with the terminal conditions. By examining the sensitivity, the initial guess is iteratively updated until an acceptable tolerance is achieved at the terminal boundary. The main disadvantage of the shooting method is that a good initial guess is generally required to achieve convergence, which in turn requires the user to have a deep insight of the physical and the mathematical properties of the problem, [127]. More recently, Modified Chebyshev Picard Iteration (MCPI) has been used to address two-point boundary value problems without resorting to the shooting method, [106, 128].

By using different trial and test functions, Eq. (5.7) lends itself to a wide spectrum of solution methodologies, [103], such as collocation, finite volume, Galerkin, MLPG, etc. In [104] a comprehensive review of various computational methods is presented and used to solve well-posed and ill-posed problems of a fourth order ODE describing a beam on an elastic foundation. In [28, 105], a collocation method with harmonic trial function was developed for studying the nonlinear responses of aeroelastic system. In this study, a simple collocation method is developed, with radial basis functions as trial functions, to tackle various time-domain inverse problems in nonlinear systems. Detailed formulations and numerical examples are presented next.

5.2 Direct Collocation and Radial Basis Functions

One direct collocation method in optimal control is the pseudo-spectral method. It transforms the set of nonlinear ODEs into a nonlinear programming problem (NLP) by using global polynomials and collocating at Gauss quadrature nodes. The methods were successfully implemented in NASA missions for the International Space Station (ISS), [129], and the space telescope TRACE, [130]. Legendre or Chebyshev polynomials were used as trial functions in [131–134]. Three types of collocations points in the time domain are mostly used: Legendre-Gauss (LG) points, Legendre-Gauss-Radau (LGR) points and Legendre-Gauss-Lobatto (LGL) points. In [135], the effect of collocation points on the accuracy of solutions was tested based on a first-order dynamical system. In [136], radial basis functions (RBFs) were also used as trial functions in a pseudo-spectral frame work. But the solution of the NLP turned out to be very sensitive to the initial guess, and it generally requires one to analytically solve a low order system and provide the solution as the initial guess for the NLP.

In this work, the two-point boundary value problem for an optimal controller is solved using direct collocation and RBFs without resorting to the pseudo-spectral methods. Using radial basis functions as the trial functions, direct collocation at the LGL nodes lead to a system of nonlinear algebraic equations (NAEs), which are solved using the classical Newton's method. Based on a large number of numerical examples for various time-domain inverse problems, it is shown that the proposed method is very simple, very accurate and insensitive to the initial guess of the unknown states. The detailed formulations are given as follows.

5.2.1 The Legendre-Gauss-Lobatto (LGL) Nodes

The well known Legendre polynomials are orthogonal to the weight function $w(\tau) = 1$ on the interval $\tau \in [-1, 1]$ and satisfy the recursion,

$$\begin{aligned} p_0(\tau) &= 1 \\ p_1(\tau) &= x \\ p_{i+1} &= \left(\frac{2i+1}{i+1}\right) \tau p_i(\tau) - \left(\frac{i}{i+1}\right) p_{i-1}(\tau), \qquad i = 1, 2, \ldots \end{aligned} \tag{5.8}$$

The LGL nodes can then be obtained from solving the differential equation,

$$\left(1 - \tau_j^2\right) \dot{p}_N(\tau_j) = 0 \tag{5.9}$$

which produces the node distribution $-1 = \tau_0 < \tau_1 < \ldots < \tau_N = 1$. The solution of the LGL nodes is generally obtained by numerical algorithms such as in [111]. By a simple mapping in Eq. (5.10), $\tau = [-1, \ldots, 1]$ is transformed into $t = [t_0, \ldots, t_F]$ to obtain the LGL nodes for an arbitrary time interval:

$$t = \frac{t_F - t_0}{2} \tau + \frac{t_F + t_0}{2} \tag{5.10}$$

5.2.2 Radial Basis Functions and Collocation

Radial Basis Functions (RBFs) are real-valued functions with values depending on the distance from a source point, $\phi(\boldsymbol{x}, \boldsymbol{x}_c) = \phi(\|\boldsymbol{x} - \boldsymbol{x}_c\|) = \phi(r)$. Some of the commonly used types for RBFs are as follows, [110]:

$$\begin{array}{cc} \phi(r) = e^{-(cr)^2} & \text{Guassian} \\ \phi(r) = \frac{1}{1+(cr)^2} & \text{Inverse quadratic} \\ \phi(r) = \sqrt{1+(cr)^2} & \text{Multiquadric} \\ \phi(r) = \frac{1}{\sqrt{1+(cr)^2}} & \text{Inverse multiquadric} \end{array} \tag{5.11}$$

where $c > 0$ is a shaping parameter.

In this study, the trial functions are expressed as a liner combination of Gaussian functions, with N LGL nodes $(t_j, j = 1, \dots, N)$ as the source points. And the collocation is also performed at the N LGL nodes, leading to:

$$\boldsymbol{x}(t_i) = \sum_{j=0}^{N} \phi(t_i, t_j)\boldsymbol{a}_j, \quad i = 1, \dots, N \tag{5.12}$$

In matrix-vector form Eq. (5.12) can be rewritten as,

$$\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{A} \tag{5.13}$$

where $\boldsymbol{\Phi}$ represents the matrix of basis functions, $\boldsymbol{A}$ is the vector of undetermined coefficients, and $\boldsymbol{X}$ is the vector of unknown states at each LGL node. The time-differentiation of Eq. (5.13) can be then expressed as,

$$\dot{\boldsymbol{X}} = \dot{\boldsymbol{\Phi}}\boldsymbol{A} \tag{5.14}$$

Hence, combining Eq. (5.13) and Eq. (5.14), $\dot{\boldsymbol{X}}$ is related to $\boldsymbol{X}$ by,

$$\dot{\boldsymbol{X}} = D\boldsymbol{X}, \quad D \equiv \dot{\boldsymbol{\Phi}}\boldsymbol{\Phi}^{-1} \tag{5.15}$$

where, D is called the derivative matrix, which is generated from the RBFs and their time-derivatives evaluated at the LGL nodes.

Similarly, the co-state functions and its time-derivatives can also be expressed using the same Gaussian functions, leading to:

$$\dot{\mathbf{\Lambda}} = D\mathbf{\Lambda}, \tag{5.16}$$

where $\mathbf{\Lambda}$ represents the unknown co-states at at each of the LGL nodes.

Using this formulation the state/co-state equations in Eq. (5.7) and the boundary conditions in Table 5.1 are discretized and transformed into a set of NAEs that can be handled by classical numerical iterative solvers such as Newton's method or by recently-developed Jacobian inverse free methods, [29, 112, 137]. In this study the classical Newton's method is utilized and other Jacobian inverse free NAEs solvers are to be explored in future studies.

5.3 The Duffing Optimal Control Problem

The duffing equation has been in the literature for almost a century, [119], with a wide range of applications in science and engineering from a nonlinear spring-mass system in mechanics to fault signal detection, [138], and structures design, [139]. The control of a duffing oscillator has a seminal significance to the control problems of nonlinear dynamic responses of structures such as beams, plates, and shells. The duffing oscillator is governed by the following second-order nonlinear ODE:

$$\ddot{x} + \omega_n^2 x + \beta x^3 = f, \quad 0 \leq t \leq T \tag{5.17}$$

which can be re-written as a system of 2 first-order ODE equations:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 + f\end{aligned} \tag{5.18}$$

where ω_n is the natural frequency and β describes the nonlinearity of the system. With $x_1(0), x_2(0)$ and $f(t)$ being given the problem is well-posed, whereas the ill-posed problem arises from the prescribed: (1) $x_1(T), x_2(T)$; (2) $x_1(0), x_1(T)$; (3) $x_1(0), x_2(T)$; (4) $x_2(0), x_1(T)$; (5) $x_2(0), x_2(T)$, with the unknown forcing function f. In order to satisfy those boundary conditions the force function f is to be obtained subject to a simple performance index :

$$\begin{aligned}\phi(\boldsymbol{x}(t_F), t_F) &= \frac{1}{2}\left[\boldsymbol{x} - \boldsymbol{x}_F\right]^{\mathrm{T}} S \left[\boldsymbol{x} - \boldsymbol{x}_F\right] \\ L(\boldsymbol{x}, \boldsymbol{f}, t) &= \frac{1}{2}\int_0^T f^2 \mathrm{d}t\end{aligned} \tag{5.19}$$

Hence,

$$J = \frac{1}{2}\left[\boldsymbol{x} - \boldsymbol{x}_F\right]^{\mathrm{T}} S \left[\boldsymbol{x} - \boldsymbol{x}_F\right] + \frac{1}{2}\int_0^T f^2 \mathrm{d}t \tag{5.20}$$

where, $\boldsymbol{x} = [x_1, x_2]^{\mathrm{T}}$, $S > 0$ is assumed diagonal for simplicity, $S \equiv \mathrm{diag}\,[s_{11}, s_{22}]$, $\boldsymbol{x}_F = [x_{1F}, x_{2F}]^{\mathrm{T}}$ is the desired final state at the specified final time T. The Hamiltonian can then be expressed as,

$$H = \frac{1}{2}f^2 + \lambda_1 x_2 + \lambda_2\left(-\omega_n^2 x_1 - \beta x_1^3 + f\right) \tag{5.21}$$

where, λ_1 and λ_2 are the Lagrange multipliers or the system co-states. The necessary conditions that relates the co-states to the controller and minimizes the cost function are derived from Eq. (5.7) as,

$$\begin{aligned}
-\dot{\lambda}_1 &= \frac{\partial H}{\partial x_1} \Rightarrow \dot{\lambda}_1 = \lambda_2 \left(\omega_n^2 + 3\beta x_1^2\right) \\
-\dot{\lambda}_2 &= \frac{\partial H}{\partial x_2} \Rightarrow \dot{\lambda}_2 = -\lambda_1 \\
\frac{\partial H}{\partial f} &= 0 \Rightarrow f = -\lambda_2
\end{aligned} \tag{5.22}$$

where different boundary conditions can also be derived following Table 5.1. In this study, several ill-posed problems of the duffing oscillator are considered: the free final state case, the fixed final state case, and the partially prescribed initial and final states. A simple extension to prescribed periodic solution case is also demonstrated. All these cases are discussed in detail in the following subsections, and are solved by the proposed simple RBF collocation method.

5.3.1 Free Final State Optimal Control Problem

With prescribed initial conditions $\boldsymbol{x}(0) = [x_{10}, x_{20}]^{\mathrm{T}}$, the objective is to find the optimal forcing function that minimizes the performance index in Eq. (5.20). From Eq. (5.6) the boundary conditions imposed on the system of ODEs in Eq. (5.22) are,

$$\begin{aligned}
&x_1(0) = x_{10}, \qquad x_2(0) = x_{20} \\
&\lambda_1(T) = s_{11}(x_1(T) - x_{1F}), \qquad \lambda_2(T) = s_{22}(x_2(T) - x_{2F})
\end{aligned} \tag{5.23}$$

where x_{1F}, x_{2F} are the desired final states. Combining Eq. (5.23) with Eq. (5.22) and Eq. (5.18) yields the system of ordinary differential equations with split boundary conditions for the free final state optimal control problem as,

$$\begin{aligned}
&\left.\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 - \lambda_2\end{aligned}\right\} x_1(0), x_2(0) \quad \text{specified}\\
&\left.\begin{aligned}\dot{\lambda}_1 &= \lambda_2\left(\omega_n^2 + 3\beta x_1^2\right)\\ \dot{\lambda}_2 &= -\lambda_1\end{aligned}\right\}\begin{aligned}\lambda_1(T) &= s_{11}(x_1(T) - x_{1F})\\ \lambda_2(T) &= s_{22}(x_2(T) - x_{2F})\end{aligned}
\end{aligned} \tag{5.24}$$

Applying RBFs collocation to the system of ODEs in Eq. (5.24) the total time is divided into N LGL nodes and the system of $4N$ nonlinear algebraic equations is obtained as,

$$\begin{aligned}
R_1^1 &= x_1^1 - x_{10} = 0\\
R_1^i &= Dx_1^i - x_2^i = 0\\
R_2^1 &= x_2^1 - x_{20} = 0\\
R_2^i &= Dx_2^i + \omega_n^2 x_1^i + \beta(x_1^i)^3 + \lambda_2^i = 0\\
R_3^j &= D\lambda_1^j - \lambda_2^j[\omega_n^2 + 3\beta(x_1^j)^2] = 0\\
R_3^N &= \lambda_1^N - s_{11}(x_1^N - x_{1F}) = 0\\
R_4^j &= D\lambda_2^j + \lambda_1^j = 0\\
R_4^N &= \lambda_2^N - s_{22}(x_2^N - x_{2F}) = 0
\end{aligned} \tag{5.25}$$

where, $i = 2, \ldots, N$, $j = 1, \ldots, N-1$. This formulation accommodates the collocation of the boundary conditions without producing an over-determined system of equations, [140]. The set of $4N$ nonlinear algebraic equations in Eq. (5.25) can then be solved by the classical Newton's method to obtain the values of the states and the co-states at the collocation nodes.

5.3.2 Fixed Final State Optimal Control Problem

For this case both the initial and final conditions are prescribed and the optimal forcing function is to be solved for to minimize the general performance index in Eq. (5.20). The split boundary condition are then given by,

$$\boldsymbol{x}(0) = \boldsymbol{x}_0, \qquad\qquad \boldsymbol{x}(T) = \boldsymbol{x}_F \tag{5.26}$$

Applying the necessary conditions in Eq. (5.7), the optimal control problem can then be formulated as,

$$\begin{aligned} &\left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 - \lambda_2 \end{aligned}\right\} x_1(0), x_2(0), x_1(T), x_2(T) \quad \text{specified} \\ &\left.\begin{aligned} \dot{\lambda}_1 &= \lambda_2\left(\omega_n^2 + 3\beta x_1^2\right) \\ \dot{\lambda}_2 &= -\lambda_1 \end{aligned}\right\} \lambda_1, \lambda_2 \quad \text{free} \end{aligned} \tag{5.27}$$

As in Eq. (5.25), the system of $4N$ nonlinear algebraic equations is constructed as,

$$\begin{aligned} R_1^i &= D x_1^i - x_2^i = 0 \\ R_2^1 &= x_1^1 - x_{10} = 0 \\ R_2^j &= D x_2^j + \omega_n^2 x_1^j + \beta (x_1^3)^j + \lambda_2^j = 0 \\ R_2^N &= x_1^N - x_{1F} = 0 \\ R_3^1 &= x_2^1 - x_{20} = 0 \\ R_3^j &= D\lambda_1^j - \lambda_2^j(\omega_n^2 + 3\beta(x_1^2)^j = 0 \\ R_3^N &= x_2^N - x_{2F} = 0 \\ R_4^i &= D\lambda_2^i + \lambda_1^i = 0 \end{aligned} \tag{5.28}$$

where, $i = 1\ldots, N$, $j = 2, \ldots, N-1$. In this way, there are $4N$ nonlinear algebraic equations for $4N$ unknowns.

5.3.3 Partially Prescribed Initial and Final States

Three additional cases of partially prescribed boundary conditions are formulated in this section. The first case prescribes the initial and final position, $x_1(0), x_1(T)$. The second case prescribes the initial position and final velocity, $x_1(0), x_2(T)$. Finally, the third case prescribes the initial velocity and final position, $x_2(0), x_1(T)$. For each of the three cases the split boundary conditions are derived from Eq. (5.6) to formulate the ill-posed system of first-order ODEs that is parameterized with RBFs collocation and solved with classical Newton's method. For the first case, the ill-posed set of ODEs is given by,

$$\begin{gathered} \left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 - \lambda_2 \end{aligned}\right\} x_1(0), x_1(T) \quad \text{specified} \\ \left.\begin{aligned} \dot{\lambda}_1 &= \lambda_2\left(\omega_n^2 + 3\beta x_1^2\right) \\ \dot{\lambda}_2 &= -\lambda_1 \end{aligned}\right\} \lambda_2(0) = 0, \;\; \lambda_2(T) = s_{22}(x_2(T) - x_{2F}) \end{gathered} \tag{5.29}$$

For the second case where initial position and final velocity are prescribed, the set of first-order ODEs with split boundary conditions is given by,

$$\begin{gathered} \left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 - \lambda_2 \end{aligned}\right\} x_1(0), x_2(T) \quad \text{specified} \\ \left.\begin{aligned} \dot{\lambda}_1 &= \lambda_2\left(\omega_n^2 + 3\beta x_1^2\right) \\ \dot{\lambda}_2 &= -\lambda_1 \end{aligned}\right\} \lambda_2(0) = 0, \;\; \lambda_1(T) = s_{11}(x_1(T) - x_{1F}) \end{gathered} \tag{5.30}$$

Finally, for the case where initial velocity and final position are prescribed, the following equations are obtained,

$$\left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega_n^2 x_1 - \beta x_1^3 - \lambda_2 \end{aligned}\right\} x_2(0), x_1(T) \quad \text{specified}$$
$$\left.\begin{aligned} \dot{\lambda}_1 &= \lambda_2\left(\omega_n^2 + 3\beta x_1^2\right) \\ \dot{\lambda}_2 &= -\lambda_1 \end{aligned}\right\} \lambda_1(0) = 0, \ \ \lambda_2(T) = s_{22}(x_2(T) - x_{2F}) \tag{5.31}$$

For each of these three cases, a collocation scheme which is similar to Eq. (5.25) and Eq. (5.28) is used. The only difference is that the boundary conditions at t_0 and t_F are changed to those in Eq. (5.29), Eq. (5.30), and Eq. (5.31).

5.3.4 Prescribed Harmonic Steady State Achieved by an Optimal Controller

In this case, with a given initial position and velocity, the duffing oscillator is required to achieve a steady harmonic state after a time interval T:

$$\hat{x}(t) = a_1\cos(\omega t) + a_2 cos(3\omega t) + a_3\cos(\frac{1}{3}\omega t) \tag{5.32}$$

where, the frequency, ω, and the amplitudes, a_1, a_2, a_3, are specified, and the same performance index of Eq. (5.19) is considered.

From Eq. (5.17), one can see that the controller is defined after T:

$$f = \ddot{\hat{x}} + \omega_n^2\hat{x} + \beta\hat{x}^3, \quad t \geq T \tag{5.33}$$

And the solution of the controller between $0 \leq t \leq T$ is entirely equivalent to solve the fixed final state optimal control problem, with the following fixed final states:

$$\begin{aligned} x_{1F} &= a_1 \cos(\omega T) + a_2 \cos(3\omega T) + a_3 \cos(\frac{1}{3}\omega T) \\ x_{2F} &= -\left[a_1 \omega \sin(\omega T) + 3a_2 \omega \sin(3\omega T) + \frac{1}{3} a_3 \omega \sin(\frac{1}{3}\omega T) \right] \end{aligned} \tag{5.34}$$

Thus, the same solution procedure given in Eq. (5.28) is used at here, with RBF as trial functions, and LGL nodes as collocation points.

5.3.5 Numerical Results

Numerical experiments are conducted for each case of the Duffing optimal control problem in section 3.1-3.4. Table 5.2 shows the parameters used for the numerical simulations. For each case, 40 LGL nodes within the time interval are used as the RBF source points and collocation points, i.e. $N = 40$. And $c = \frac{N-1}{4T}$ is used for all the examples. the states and the controller resulting from the solution of the NAEs are plotted. And then the obtained initial conditions are fed into a standard numerical integrator, *MATLAB ODE45*. The differences between the solution by collocation and the integrator are plotted at each collocation point, i.e. $\Delta x = x_{RBF} - x_{ODE}$.

Table 5.2: Duffing Optimal Control Problem Parameters, [123]

Parameter	**Value**
Natural frequency, ω_n	1 rad/s
Nonlinearity coefficient, β	≥ 0.9
Initial conditions, $\boldsymbol{x}_0$	$\begin{bmatrix} 0 & 0 \end{bmatrix}^{\mathrm{T}}$
Final conditions, $\boldsymbol{x}_F$	$\begin{bmatrix} 5 & 2 \end{bmatrix}^{\mathrm{T}}$
Harmonic response amplitudes	$a_1 = 1.5$, $a_2 = 2$, $a_3 = 3$
Harmonic response frequency	$\omega = 3$ rad/s

For the first case with the free final conditions and $\beta = 0.9$, Fig. 5.1 shows the states and the controller time history. Figure 5.2 and Fig. 5.3 show the errors between the semi-analytic solution obtained by RBFs collocation and the numerical integrator at each collocation point for the states and the co-states, respectively. As shown from the plots, the errors are in the order of $\approx 10^{-7}$ for both the states and the co-states.

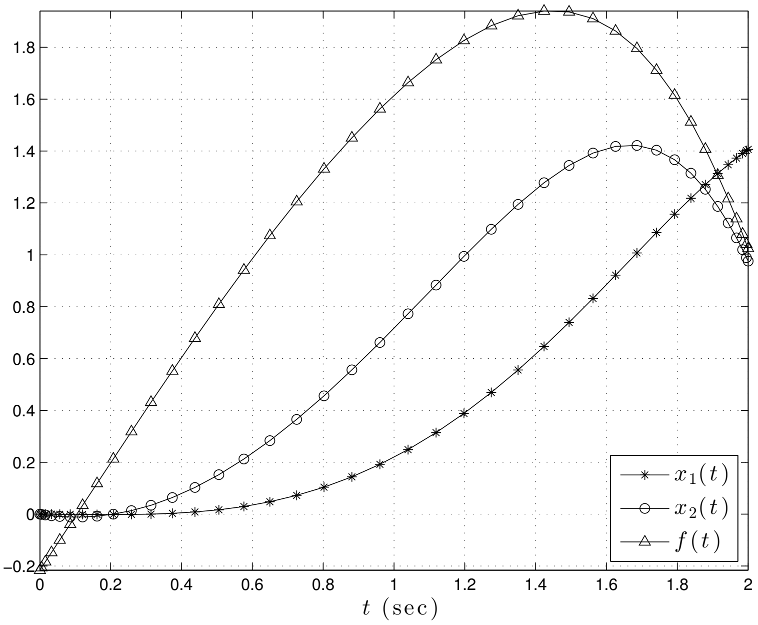

Figure 5.1: States and Controller, Duffing Free Final State, [123]

The results for the fixed final state case with $\beta = 0.9$ are shown in Fig. 5.4 for the states and the controller time history. Figure 5.5 and Fig. 5.5 show the errors in the states and the co-states. Similar to the free final state case the errors in the states and co-states are very small. It is also worth noting that the shooting method implemented at $\beta \geq 0.9$ will diverge with the arbitrary initial guess used for the

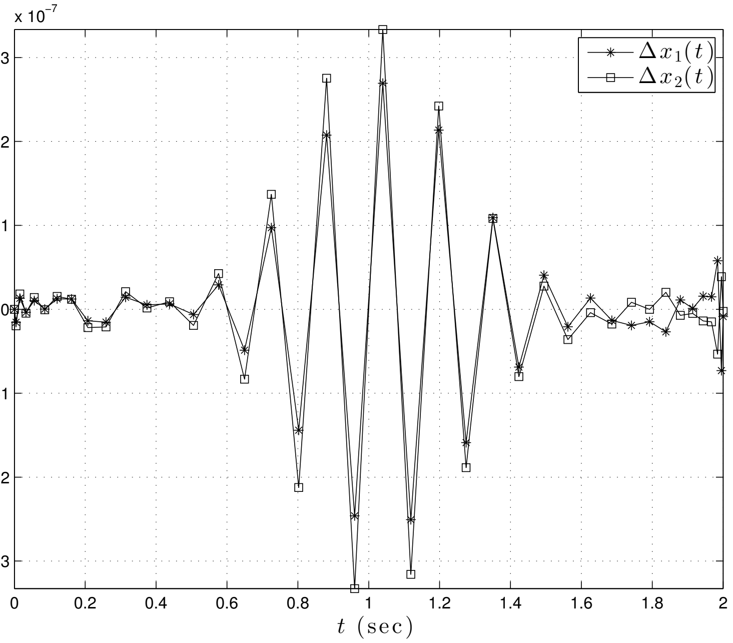

Figure 5.2: States Errors, Duffing Free Final State, [123]

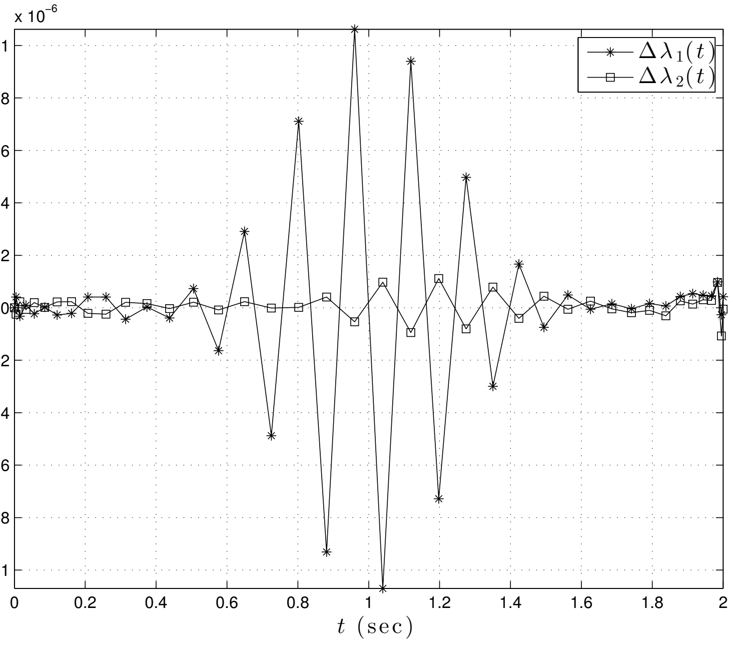

Figure 5.3: Co-states Errors, Duffing Free Final State, [123]

RBFs collocation solution.

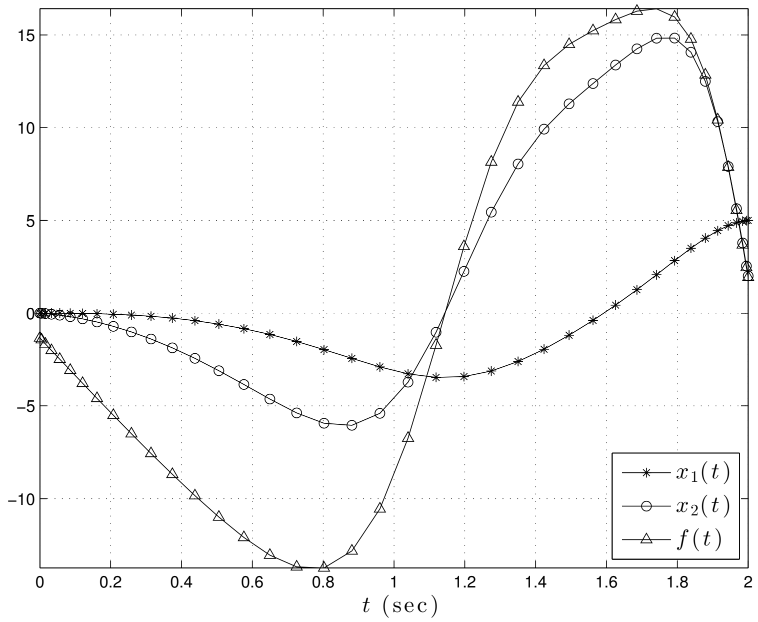

Figure 5.4: States and Controller, Duffing Fixed Final State, [123]

Fig. 5.7, Fig. 5.8 and Fig. 5.8 show the results for the first case of partially prescribed boundary conditions with $x_1(0) = x_{10}$ and $x_1(T) = x_{1F}$. In that case, $\beta = 0.94$. Results for the second case of partial boundary conditions are shown in Fig. 5.10, Fig. 5.11 and Fig. 5.12 with $x_1(0) = x_{10}$ and $x_2(T) = x_{2F}$. Similar to the previous case, the same value for the nonlinearity coefficient is chosen. The third case of partial boundary conditions is given by $x_2(0) = x_{20}$ and $x_1(T) = x_{1F}$ with $\beta = 0.97$. Numerical results for this case are shown in Fig. 5.13, Fig. 5.14 and Fig. 5.15.

Finally, for the case of a prescribed harmonic steady state with $\beta = 0.9$, the results are shown in Fig. 5.16, whereas the errors in the states and the co-states at

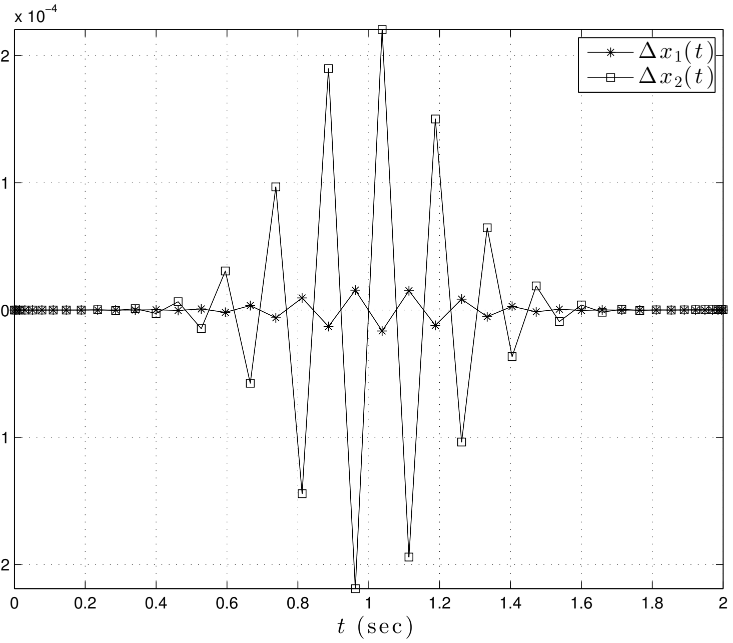

Figure 5.5: States Errors, Duffing Fixed Final State, [123]

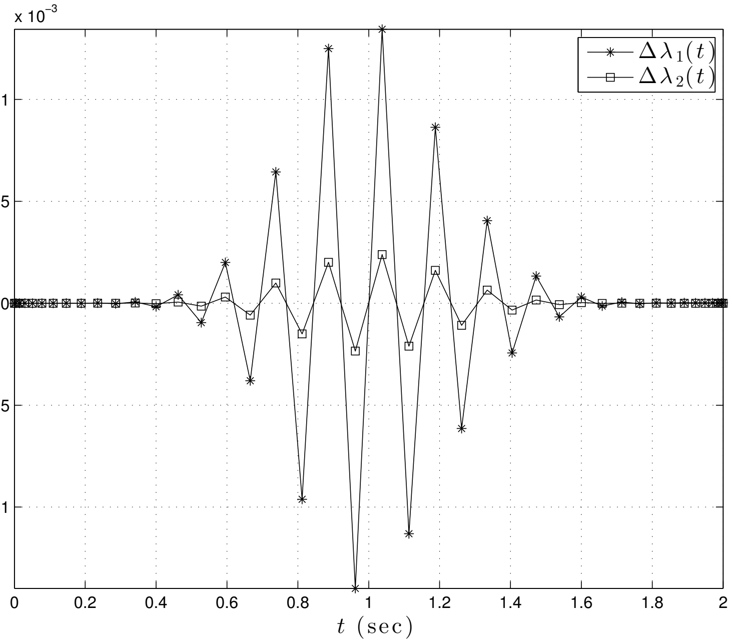

Figure 5.6: Co-states Errors, Duffing Fixed Final State, [123]

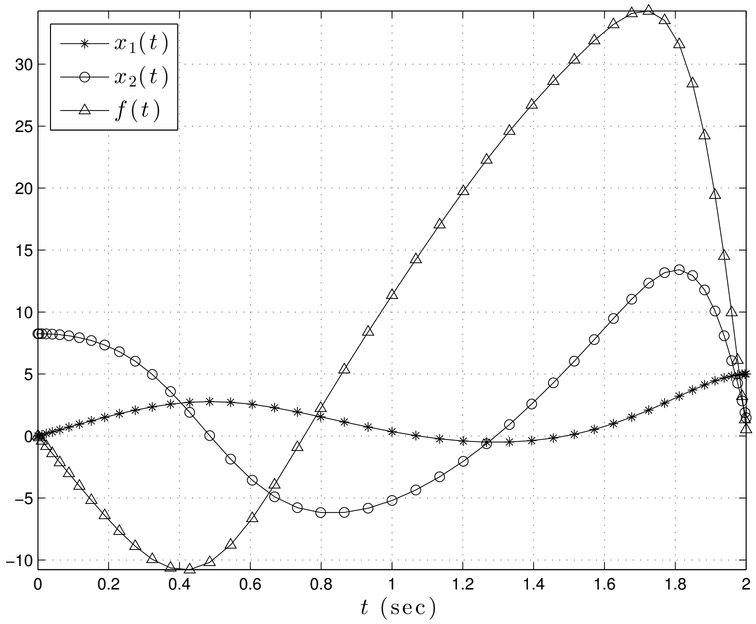

Figure 5.7: States and Controller, Duffing Prescribed Initial and Final Position, [123]

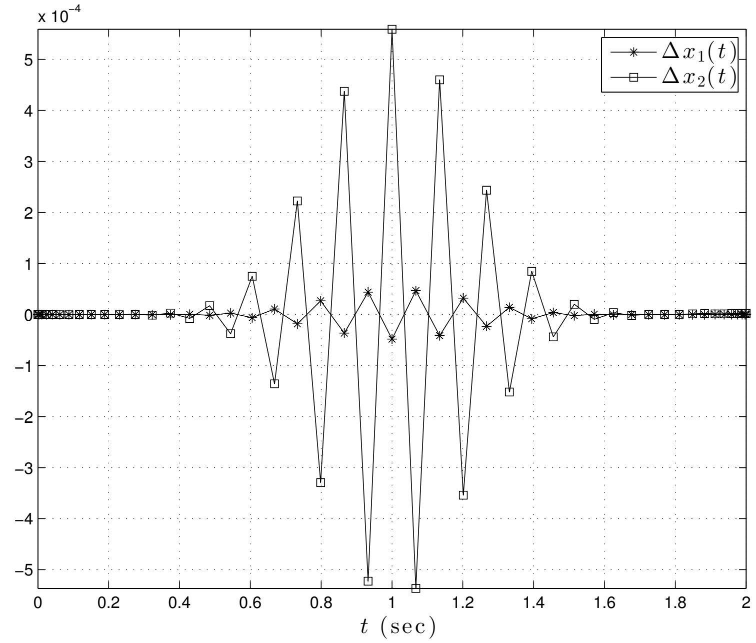

Figure 5.8: States Errors, Duffing Prescribed Initial and Final Position, [123]

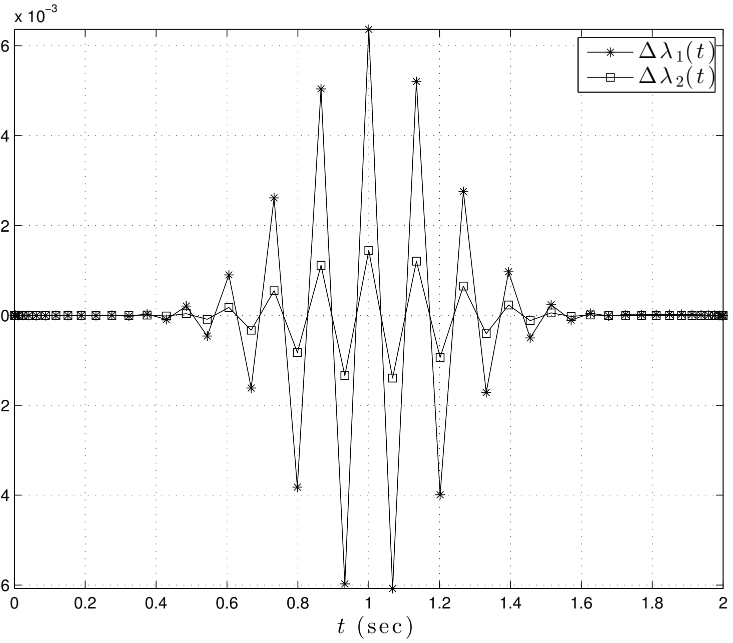

Figure 5.9: Co-states Errors, Duffing Prescribed Initial and Final Position, [123]

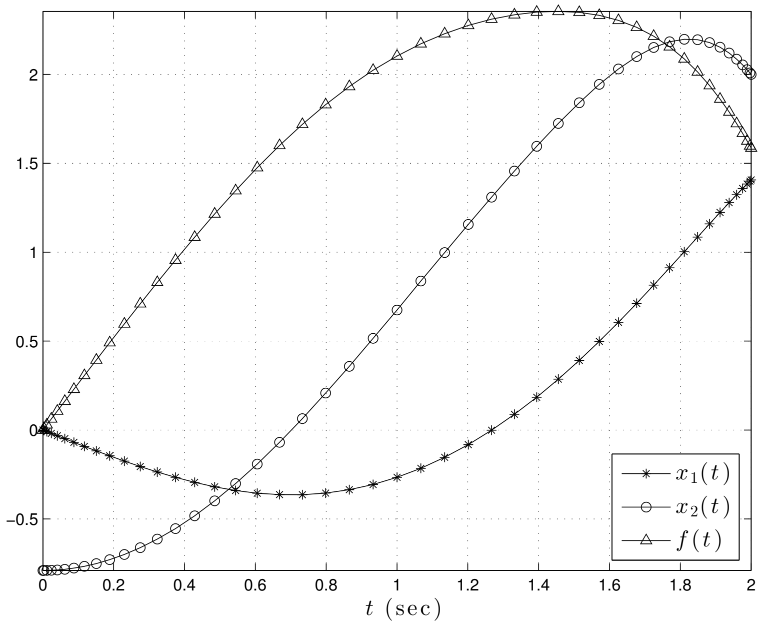

Figure 5.10: States and Controller, Duffing Prescribed Initial Position and Final Velocity, [123]

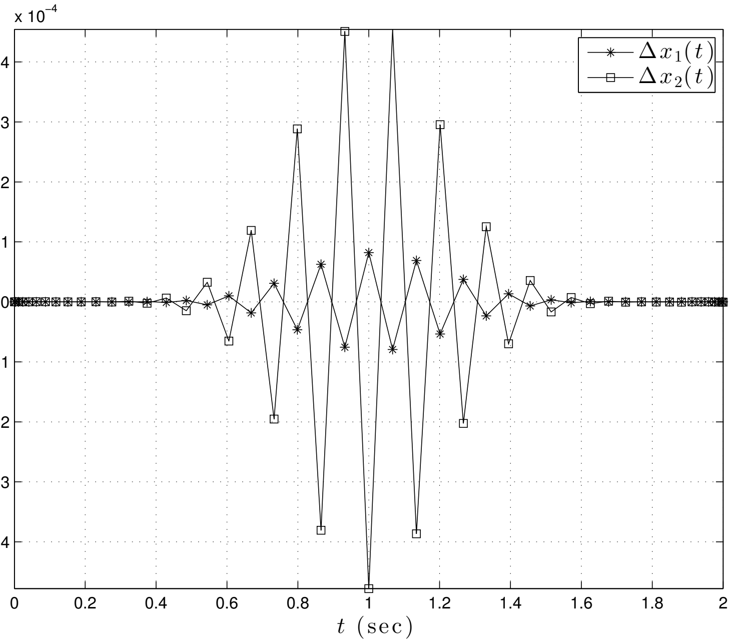

Figure 5.11: States Errors, Prescribed Initial Position and Final Velocity, [123]

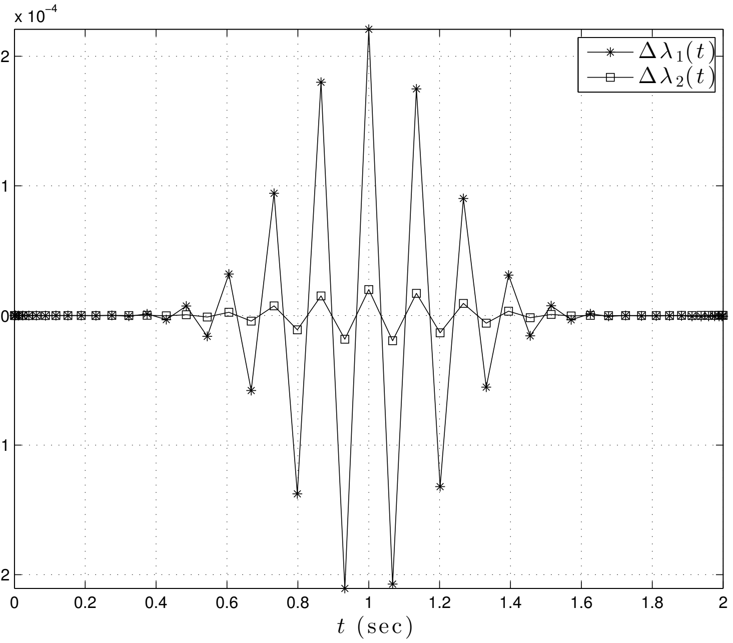

Figure 5.12: Co-states Errors, Prescribed Initial Position and Final Velocity, [123]

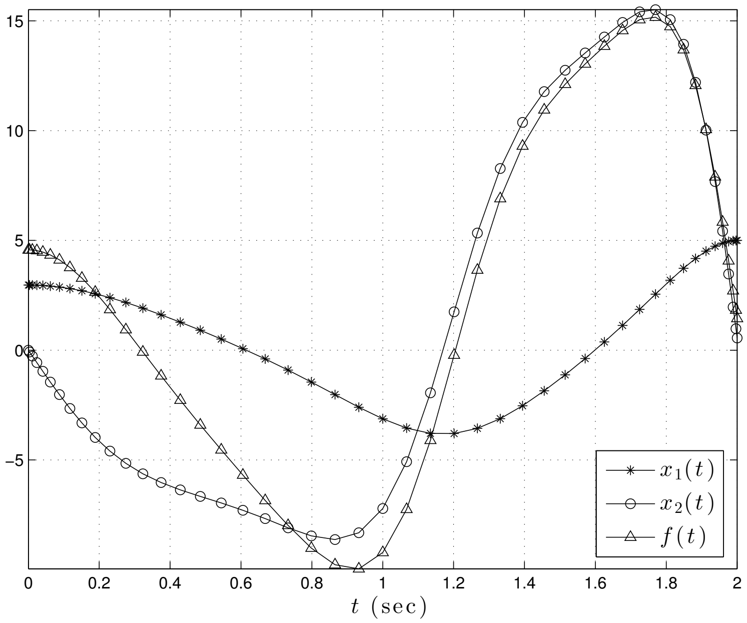

Figure 5.13: States and Controller, Duffing Prescribed Initial Velocity and Final Position, [123]

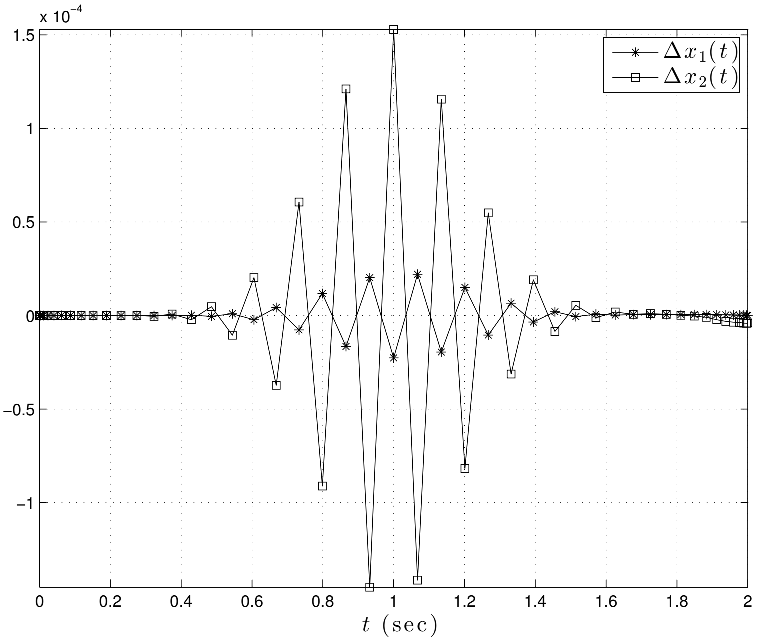

Figure 5.14: States Errors, Prescribed Initial Velocity and Final Position, [123]

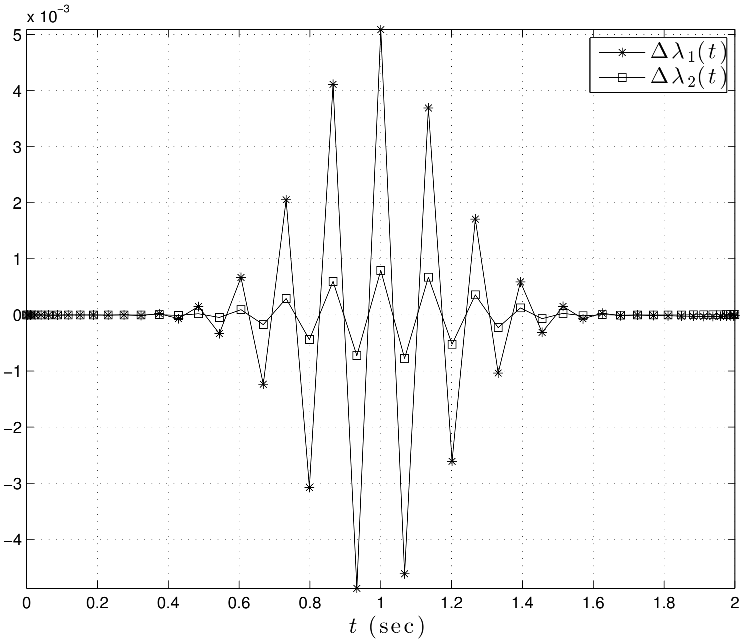

Figure 5.15: Co-states Errors, Prescribed Initial Velocity and Final Position, [123]

each collocation point are shown in Fig. 5.17 and Fig. 5.18, respectively, which is in the order of 10^{-7}.

As a summary for all the presented results, Table 5.3 shows the errors between the prescribed boundary conditioned and the ones obtained from solving the set of nonlinear algebraic equations. It is shown that the discretization of the fixed time-interval optimal control problem using RBFs as trial functions, and simple collocation at the LGL nodes, has demonstrated its high accuracy for all cases of boundary conditions explored in this study.

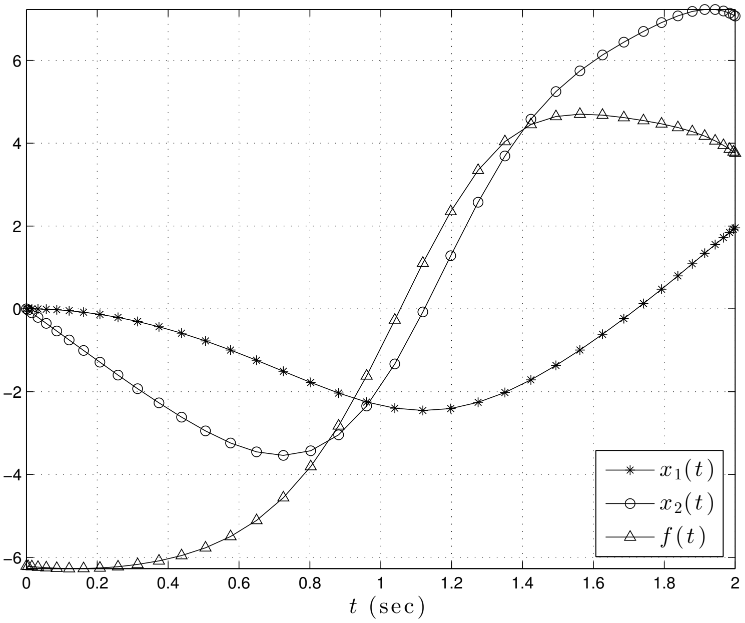

Figure 5.16: States and Controller, Duffing Prescribed Harmonic State, [123]

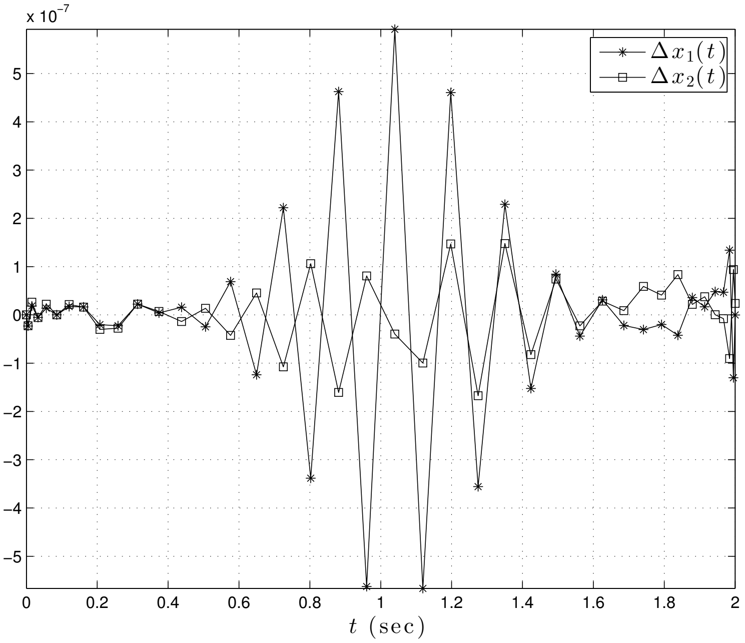

Figure 5.17: States and Controller, Duffing Prescribed Harmonic State, [123]

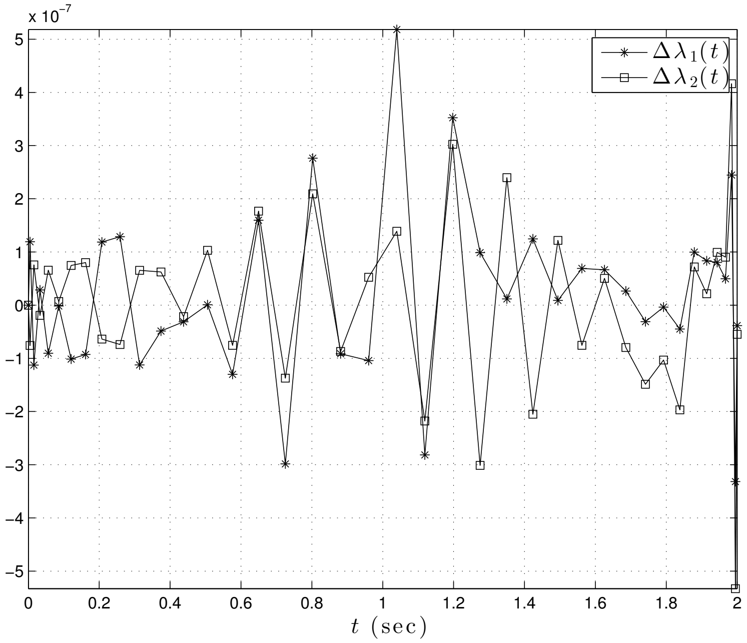

Figure 5.18: States Errors, Duffing Prescribed Harmonic State, [123]

Table 5.3: Errors in Boundary Conditions, [123]

Cases of different BCs	**Error at t_0**	**Error at t_F**
Free final state	$\begin{Bmatrix} 1 \times 10^{-22} \\ 4.2 \times 10^{-22} \end{Bmatrix}$	$\begin{Bmatrix} 0 \\ 2.2 \times 10^{-16} \end{Bmatrix}$
Fixed final state	$\begin{Bmatrix} 9.9 \times 10^{-18} \\ 0 \end{Bmatrix}$	$\begin{Bmatrix} 1.9 \times 10^{-17} \\ 0 \end{Bmatrix}$
Prescribed Initial and final position	$\begin{Bmatrix} 3.8 \times 10^{-17} \\ 6.8 \times 10^{-19} \end{Bmatrix}$	$\begin{Bmatrix} 0 \\ 1.1 \times 10^{-16} \end{Bmatrix}$
Prescribed initial position and final Velocity	$\begin{Bmatrix} 6 \times 10^{-19} \\ 1.4 \times 10^{-19} \end{Bmatrix}$	$\begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$
Prescribed Initial Velocity and final Position	$\begin{Bmatrix} 3.2 \times 10^{-17} \\ 3 \times 10^{-17} \end{Bmatrix}$	$\begin{Bmatrix} 0 \\ 2.2 \times 10^{-16} \end{Bmatrix}$
Harmonic steady state	$\begin{Bmatrix} 4.3 \times 10^{-19} \\ 1.4 \times 10^{-18} \end{Bmatrix}$	$\begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$

5.4 Orbital Transfer Two-Point Boundary Value Problem

Another classical problem in celestial mechanics is the two-point boundary value problem of orbital transfer. This problem is known as Lambert's problem after Johann Heinrich Lambert (1728–1779) who was the first to state and to solve the problem. The objective is to find the transfer orbit that connects two points in space given a flight time. Figure 5.19 illustrates the geometry of Lambert's problem with $t_0, \boldsymbol{r}_0$ the initial time and position, $t_F, \boldsymbol{r}_F$ the desired final time and position, $\boldsymbol{v}_0$ the initial velocity to be solved for that would generate the transfer orbit and $\boldsymbol{v}_F$ the terminal velocity at the desired position.

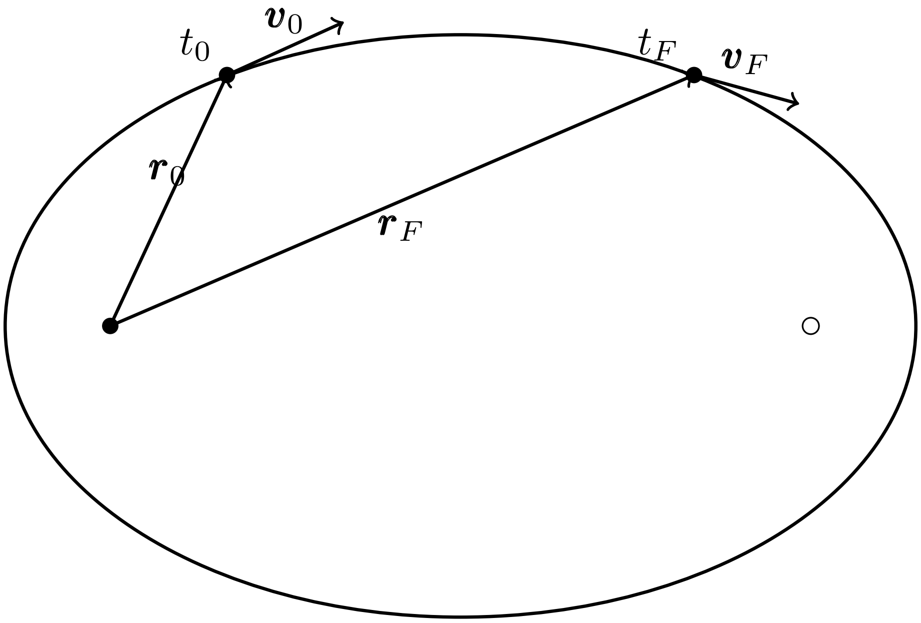

Figure 5.19: Illustration of the Orbital Transfer Problem, [123]

The dynamics of the unperturbed relative two-body problem is obtained from

Newton's famous universal gravitational law as,

$$\ddot{\boldsymbol{r}} = \frac{-\mu}{r^3}\boldsymbol{r} \tag{5.35}$$

where, $\boldsymbol{r} = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$ is the position vector in the inertial frame, μ is the Earth gravitational parameter $\mu \approx 3.986 \times 10^{14}\ \mathrm{m^3 s^{-2}}$ and $r = \sqrt{x^2 + y^2 + z^2}$.

Solving the unperturbed Lambert's problem analytically was discussed in detail in [83]. The solution developed still has a singularity for transfer angles of ±180°. In [84], a numerical iterative method is introduced to handle both singularities and gravitational perturbations in Lambert's problem. The method is essentially a shooting algorithm where a sufficiently good initial guess for the initial velocity is needed to improve convergence. Generally, the initial guess for the velocity vector is obtained such that the target position is reached but not necessarily in the required transfer time. The present solution based on RBFs collocation starts with an arbitrary initial guess, and can readily handle any perturbations to provide a semi-analytic solution for the transfer orbit problem.

As a first order system of equations the unperturbed two-body problem is written as,

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{-\mu}{r^3} x_1 \\
\dot{y}_1 &= y_2 \\
\dot{y}_2 &= \frac{-\mu}{r^3} y_1 \\
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= \frac{-\mu}{r^3} z_1
\end{aligned} \tag{5.36}$$

The RBFs collocation can then be applied to Eq. (5.36) to produce a set of $6N$

nonlinear algebraic equations where N is the number of LGL collocation nodes:

$$\begin{aligned}
R_1^i &= Dx_1^i - x_2^i = 0 \\
R_2^i &= Dy_1^i - y_2^i = 0 \\
R_3^i &= Dz_1^i - z_2^i = 0 \\
R_4^1 &= x_1^1 - x_{10} = 0 \\
R_4^j &= Dx_2^j + \frac{\mu}{r^j} x_1^j = 0 \\
R_4^N &= x_1^N - x_{1F} = 0 \\
R_5^1 &= y_1^1 - y_{10} = 0 \\
R_5^j &= Dy_2^j + \frac{\mu}{r^j} y_1^j = 0 \\
R_5^N &= y_1^N - y_{1F} = 0 \\
R_6^1 &= z_1^1 - z_{10} = 0 \\
R_6^j &= Dz_2^j + \frac{\mu}{r^j} z_1^j = 0 \\
R_6^N &= z_1^N - z_{1F} = 0
\end{aligned} \tag{5.37}$$

where, $i = 1, \ldots, N$ and $j = 2, \ldots, N-1$ which produces a system of $6N$ equations in $6N$ unknowns. As a numerical example an orbit is examined with initial and final position given by,

$$\begin{aligned}
\boldsymbol{r}_0 &= \begin{bmatrix} 2.87 & 5.19 & 2.85 \end{bmatrix}^{\mathrm{T}} \times 10^6 \,\mathrm{m} \\
\boldsymbol{r}_F &= \begin{bmatrix} 2.09 & 7.82 & 0 \end{bmatrix}^{\mathrm{T}} \times 10^6 \,\mathrm{m}
\end{aligned} \tag{5.38}$$

The transfer time is chosen to be $t_F = 0.05$ days or $t_F = 4.32 \times 10^3$ seconds. The number of LGL nodes is set as, $N = 47$, with the shaping parameter, $c = \frac{N+3}{4T}$. The set of nonlinear algebraic equations in Eq. (5.37) is solved with an arbitrary initial guess. And the resulting orbit is compared against the closed form Lagrange/Gibbs

(F& G) solution, [83,84], considering the initial position and velocity vector obtained from the RBFs collocation method. The resulting position and velocity are compared in Fig. 5.20 and Fig. 5.21, respectively.

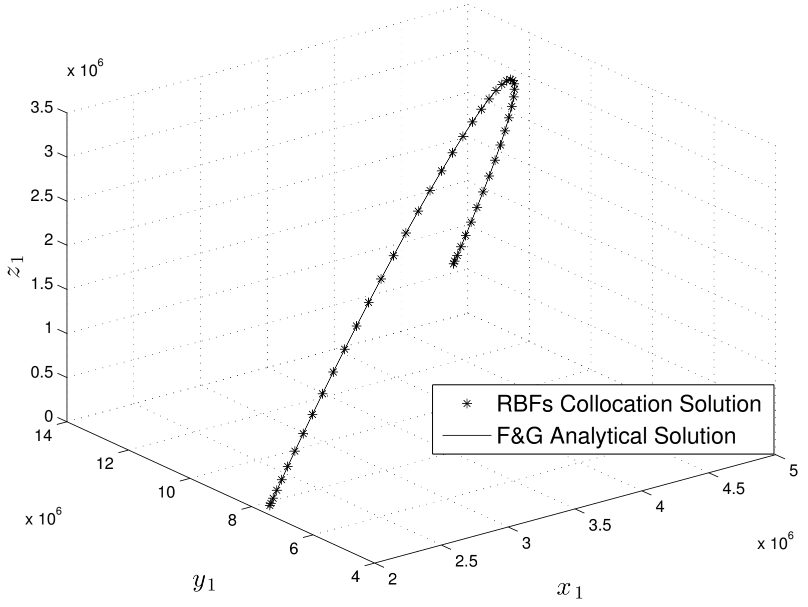

Figure 5.20: Transfer Orbit Position Propagation, [123]

The errors of the initial and the terminal boundary conditions are compared as in Table 5.3 in Table 5.4. The initial conditions obtained by the RBF collocation method drives the object to the desired final position with millimeter accuracy. This approach using RBFs collocation thus is quite advantageous compared to previous analytical and numerical methods of solving the Lambert's problem because of the ease of extension to accommodate perturbations. It can be extended in future to address perturbations and obtain what is known as pork-chop plots for the selection of launch and arrival times while minimizing fuel or some other specified parameters.

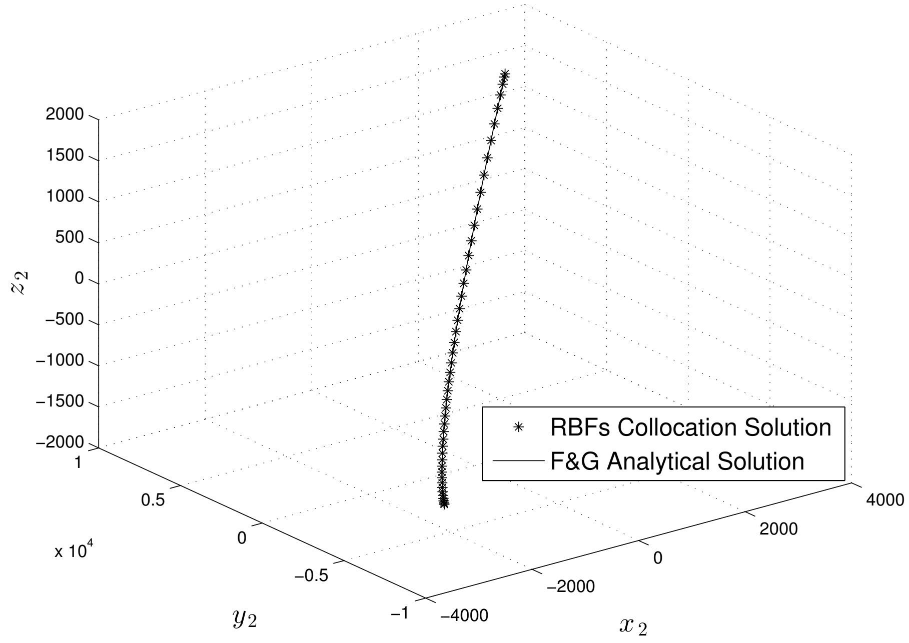

Figure 5.21: Transfer Orbit Velocity Propagation, [123]

Table 5.4: Errors in Boundary Conditions, [123]

Initial Boundary Error	**Terminal Boundary Error**
$\begin{Bmatrix}0\\0\\0\end{Bmatrix}$ m	$\begin{Bmatrix}9.610^{-3}\\16.3\times10^{-3}\\10.2\times10^{-3}\end{Bmatrix}$ m

5.5 Summary

The present simple collocation scheme based on radial basis functions and LGL collocation points is proven to be very accurate and efficient in solving time domain inverse problems. Starting with the Duffing OCP, all the cases considered achieved very high accuracy in the initial and final conditions. The solution is insensitive to the initial guess and does not require any insight into the physics of the problem. Several extensions are possible for the OCP to include intermediate boundary conditions or inequality constraints along the trajectory. The orbital transfer problem based on the same formulation achieved millimeter accuracy when compared to the analytical Lagrange/Gibbs F&G solution. Unlike the shooting method which requires the problem to be solved first for an arbitrary time and the solution to fed in as the initial guess, the RBFs collocation approach started at an arbitrary set of initial conditions and achieved very high accuracy in determining the transfer orbit. It has to be noted that the case presented in this work is for a fraction of an orbit, however, the *RBF-Coll* algorithm can be developed to address the multiple orbit problems and consequently provide pork-chop plots for launch and arrival times for mission design which will be explored in future studies. For all the cases considered in this work, the generated set of NAEs are solved with the classical Newton's method whereas several Jacobian inverse free methods exist and can be explored in future studies, [29, 112, 137].

6. DISCUSSION AND CONCLUSION

In summary, several research thrusts have been presented in this work. The diversity of problems presented and their applications can be correlated under solution methodologies for nonlinear systems in mechanics. In structural mechanics, the Scalar Homotopy Methods are applied to the solution of post-buckling and limit load problems of plane frames considering geometrical nonlinearities. Nodal displacements of the equilibrium state are iteratively solved for, without inverting the Jacobian (tangent stiffness) matrix and without using complex arc-Length methods. This simple method thus saves computational time and avoids the problematic behavior of the Newton's method when the Jacobian matrix is singular. The simple Williams' toggle is presented however, extension to general finite element analyses of space frames, plates, shells and elastic-plastic solids can be considered.

On the other hand, an analytic approach is adopted to address the problem of a flexible rotating spacecraft. Analytic transfer functions for the system frequency response are developed. The fact that nontrivial problems can be solved by these methods, using distributed parameter models, does not appear to be widely appreciated. These special case models and control design methods serve important roles in evaluating the applicability and validity of the approximations implicit in more generally applicable spatial discretization methods. Several other boundary conditions and constitutive assumptions can be applied and the analogous steps to those presented can be followed in order to obtain the analytical distributed parameter solution. By utilizing the full transfer function solution provided by the GSS approach any control problem design in the frequency domain can be addressed. A case study is constructed for the gain selection of a Lyapunov stable control law. By looking at the frequency response and changing the gains an acceptable performance was achieved driving the structure from a stationary initial state to a target state while suppressing the beam vibrations. The GSS approach can be considered a platform through which distributed parameters models can be addressed. Moreover, the presented control problem has potential for several extensions. Optimization was not considered in this work whereas several techniques exists for optimization in the frequency domain based on Parseval's theorem. In general the GSS solution provides a general framework for any control scheme in the frequency domain. This is shown to be a very powerful tool. When it is possible to use discretization and truncation-free distributed parameter model transfer function solutions provided by the GSS approach, any control problem design in the frequency domain can be addressed rigorously.

In the fields of geodesy and celestial mechanics, two analytic methods are introduced. The Earth-Centered Earth-Fixed (ECEF) to geodetic coordinate transformation is examined with a non-iterative expansion based approach inspired by the Earth's perturbed geometry, where the expansion parameter is the ratio of the Earth semi-major axis and semi-minor axis subtracted from 1. The expansion is carried out to second, third, and fourth orders with numerical examples to compare the accuracy at each order of expansion. Significant improvement in accuracy is demonstrated as the order of expansion is increased, and at fourth order, millimeter accuracy is achieved in height and 10^{11} degree error in latitude. Those errors at such low orders of the expansion are proof of the effectiveness of the method and its potential in solving such a highly nonlinear transformation non-iteratively. The method can be further streamlined for timing studies, but in general it is a clean straightforward approach to the coordinate transformation problem that utilizes a physical perturbation parameter and that proved to be very accurate and efficient. Orbit propagation of the two-body problem is studied via the *Analytic-Cont* algorithm. The algorithm is shown to be highly accurate, fast and very simple to implement for short as well as long-term orbit propagation problems. The comparison against *Matlab ODE45* and *RKN12(10)* shows the advantages of the *Analytic-Cont* algorithm that enables larger step-size with relatively high solution accuracy while maintaining low computational cost. The case of 20 full HEO orbits shows the potential of the algorithm for adaptive step size schemes. Gravitational perturbations can also be easily handled with the *Analytic-Cont* algorithm and will be explored in future works.

The *RBF-Coll* algorithm is applied to Lambert's problem and successfully achieved millimeter accuracy when compared to the analytical Lagrange/Gibbs F&G solution. Unlike the shooting method which requires the problem to be solved first for an arbitrary time and the solution to fed in as the initial guess, the *RBF-Coll* approach started at an arbitrary set of initial conditions and achieved very high accuracy in determining the transfer orbit. The numerical example presented here addresses the orbital transfer problem for a fraction of an orbit whereas, the multiple orbit problem and the consequent development of pork-chop plots for launch and arrival times for mission design will be explored in future works. The algorithm also has potential applicability in reachability problems in rendezvous and relative motion problems.

For general highly nonlinear dynamical systems, the *RBF-Coll* algorithm is shown to be highly accurate, fast and very simple to implement for various types of IVPs. Comparing the algorithm versus several implicit and explicit numerical integration methods clearly shows the advantages of such an algorithm that enables larger time step, high solution accuracy while maintaining a relatively low computational cost. The algorithm is shown to accurately and simply handle short and long periods of time integration, single and multi degrees of freedom systems and periodic and finally transient and periodic solutions. *RBF-Coll* algorithm can have a significant potential in handling various types of dynamical systems governed by second or higher order differential equations. Areas of study that will be explored in future works.

The *RBF-Coll* algorithm is also applied to general nonlinear time domain inverse problems. Starting with the Duffing OCP, all the cases considered achieved very high accuracy in the initial and final conditions. The solution is insensitive to the initial guess and does not require any insight into the physics of the problem. Several extensions are possible for the utilization of the *RBF-Coll* method in orbit propagation in celestial mechanics, dynamic buckling problems and optimal control and two-point boundary value problems with intermediate boundary conditions or inequality constraints along the trajectory.

Finally, while the *RBF-Coll* method has been implemented with the LGL node, this is but one good choice of nodes which typically reduces the Runge effects. Also, the sharpness of the exponential basis functions were held constant over each approximation interval. However, particular dynamical systems may have very non-uniform nonlinearity, so it seems one area of future investigation should be the investigation of methods to adaptively locate nodes and select the sharpness of the basis functions. It is possible that these ideas should be pursued using redundant nodes and least squares approximation (in lieu of strict collocation) so that the residuals at all nodes can be used to guide the location and sharpness of the basis functions added to achieve recursive refinement (area recommended for further study).

REFERENCES

[1] Hirsch, M. W. and Smale, S., "On algorithms for solving f (x)= 0," *Communications on Pure and Applied Mathematics*, Vol. 32, No. 3, 1979, pp. 281–312.

[2] Bryson, A. E. and Ho, Y.-C., *Applied optimal control: optimization, estimation and control*, CRC Press, 1975.

[3] Rao, S. S. and Rao, S., *Engineering optimization: theory and practice*, John Wiley & Sons, 2009.

[4] Crassidis, J. L. and Junkins, J. L., *Optimal estimation of dynamic systems*, CRC press, 2011.

[5] Davidenko, D. F., "On a new method of numerical solution of systems of nonliear equations," *Dokl. Akad. Nauk SSSR*, Vol. 88, 1953, pp. 601–602.

[6] Davidenko, D., "On the approximate solution of sets of nonlinear equations," *Ukr. Mat. Zh*, Vol. 5, No. 2, 1953, pp. 196–206.

[7] Wempner, G. A., "Discrete approximations related to nonlinear theories of solids," *International Journal of Solids and Structures*, Vol. 7, No. 11, 1971, pp. 1581–1599.

[8] Riks, E., "The application of Newton's method to the problem of elastic stability," *Journal of Applied Mechanics*, Vol. 39, No. 4, 1972, pp. 1060–1065.

[9] Crisfield, M. A., "An arc-length method including line searches and accelerations." *International journal for numerical methods in engineering*, Vol. 19, No. 9, 1983, pp. 1269–1289.

[10] Lam, W. and Morley, C., "Arc-length method for passing limit points in structural calculation." *Journal of structural engineering*, Vol. 118, No. 1, 1992, pp. 169–185.

[11] Fletcher, R., *Practical methods of optimization*, John Wiley & Sons, 2013.

[12] Broyden, C. G., "A class of methods for solving nonlinear simultaneous equations," *Mathematics of computation*, pp. 577–593.

[13] Broyden, C. G., "Quasi-Newton methods and their application to function minimisation," *Mathematics of Computation*, pp. 368–381.

[14] Broyden, C. G., "The convergence of a class of double-rank minimization algorithms 1. general considerations," *IMA Journal of Applied Mathematics*, Vol. 6, No. 1, 1970, pp. 76–90.

[15] Broyden, C. G., "The convergence of a class of double-rank minimization algorithms 2. The new algorithm," *IMA Journal of Applied Mathematics*, Vol. 6, No. 3, 1970, pp. 222–231.

[16] Dennis, J. E., Jr and Moré, J. J., "Quasi-Newton methods, motivation and theory," *SIAM review*, Vol. 19, No. 1, 1977, pp. 46–89.

[17] Liu, C.-S. and Atluri, S. N., "A novel time integration method for solving a large system of non-linear algebraic equations," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 31, No. 2, 2008a, pp. 71–83.

[18] Liu, C.-S. and Atluri, S. N., "A novel fictitious time integration method for solving the discretized inverse Sturm-Liouville problems, for specified eigenvalues," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 36, No. 3, 2008b, pp. 261–286.

[19] Liu, C.-S. and Atluri, S. N., “A fictitious time integration method (FTIM) for solving mixed complementarity problems with applications to non-linear optimization,” *CMES: Computer Modeling in Engineering & Sciences*, Vol. 34, No. 2, 2008c, pp. 155–178.

[20] Kirszenblat, A. and Chetrit, M., “Least squares nonlinear parameter estimation by the iterative continuation method,” *AIAA Journal*, Vol. 12, No. 12, 1974, pp. 1751–1752.

[21] Dunyak, J., Junkins, J., and Watson, L. T., “Robust nonlinear least squares estimation using the Chow-Yorke homotopy method,” *Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 6, 1984, pp. 752–755.

[22] Liu, C. S., Yeih, W., Kuo, C. L., and Atluri, S. N., “A scalar homotopy method for solving an over/under determined system of non-linear algebraic equations,” *Computer Modeling in Engineering and Sciences*, Vol. 53, No. 1, 2009, pp. 47–71.

[23] Ku, C.-Y., Yieh, W., and Liu, C.-S., “Solving non-linear algebraic equations by a scalar Newton-homotopy continuation method,” *International Journal of Nonlinear Sciences and Numerical Simulation*, Vol. 11, No. 6, 2010, pp. 435–450.

[24] Liu, C.-S. and Atluri, S. N., “Simple” residual-norm” based algorithms, for the solution of a large system of non-linear algebraic equations, which converge faster than the Newton’s method,” *Computer Modeling in Engineering & Sciences(CMES)*, Vol. 71, No. 3, 2011, pp. 279–304.

[25] Liu, C.-S. and Atluri, S. N., “An Iterative Algorithm for Solving a System of Nonlinear Algebraic Equations, F (x)= 0, Using the System of ODEs with an

Optimum α in $\dot{x} = \lambda[\alpha F + (1-\alpha)B^T F]$; $B_{ij} = \partial F_i / \partial x_j$," *Computer Modeling in Engineering & Sciences(CMES)*, Vol. 73, No. 4, 2011, pp. 395–431.

[26] Liu, C.-S., Dai, H.-H., and Atluri, S. N., "A Further Study on Using $x = \lambda[\alpha R + \beta P](P = F - R(F.R)/\left\|R\right\|^2)$ and $\dot{x} = \lambda[\alpha F + \beta P*](P* = R - F(F.R)/\left\|F\right\|^2)$ in Iteratively Solving the Nonlinear System of Algebraic Equations F (x)= 0," *Computer Modeling in Engineering and Sciences*, Vol. 81, No. 2, 2011, p. 195.

[27] Liu, C.-S., Dai, H.-H., and Atluri, S. N., "Iterative Solution of a System of Nonlinear Algebraic Equations F (x)= 0, Using $x = \lambda[\alpha R + \beta P]$ or $x = \lambda[\alpha F + \beta P*]$, R is a Normal to a Hyper-Surface Function of F, P Normal to R, and $P*$ Normal to F," *Computer Modeling in Engineering & Sciences(CMES)*, Vol. 81, No. 3-4, 2011, pp. 335–362.

[28] Dai, H., Yue, X., Yuan, J., and Atluri, S. N., "A time domain collocation method for studying the aeroelasticity of a two dimensional airfoil with a structural nonlinearity," *Journal of Computational Physics*, Vol. 270, 2014, pp. 214–237.

[29] Elgohary, T. A., Dong, L., Junkins, J. L., and Atluri, S. N., "Solution of Post-Buckling & Limit Load Problems, Without Inverting the Tangent Stiffness Matrix & Without Using Arc-Length Methods," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 98, No. 6, 2014, pp. 543–563.

[30] Turner, J. D. and Elgohary, T. A., "A Simple Perturbation Algorithm for Inverting the Cartesian to Geodetic Transformation," *Mathematical Problems in Engineering*, Vol. 2013.

[31] Kondoh, K. and Atluri, S. N., "A simplified finite element method for large deformation, post-buckling analyses of large frame structures, using explicitly

derived tangent stiffness matrices," *International journal for numerical methods in engineering*, Vol. 23, No. 1, 1986, pp. 69–90.

[32] Williams, F. W., "An approach to the non-linear behaviour of the members of a rigid jointed plane framework with finite deflections," *The Quarterly Journal of Mechanics and Applied Mathematics*, Vol. 17, No. 4, 1964, pp. 451–469.

[33] Heiskanen, W. A. and Moritz, H., "Physical geodesy," *Bulletin Géodésique (1946-1975)*, Vol. 86, No. 1, 1967, pp. 491–492.

[34] Borkowski, K. M., "Accurate algorithms to transform geocentric to geodetic coordinates," *Bulletin géodésique*, Vol. 63, No. 1, 1989, pp. 50–56.

[35] Lin, K.-C. and Wang, J., "Transformation from geocentric to geodetic coordinates using Newton's iteration," *Bulletin Geodesique*, Vol. 69, No. 4, 1995, pp. 300–303.

[36] Fukushima, T., "Fast transform from geocentric to geodetic coordinates," *Journal of Geodesy*, Vol. 73, No. 11, 1999, pp. 603–610.

[37] Pollard, J., "Iterative vector methods for computing geodetic latitude and height from rectangular coordinates," *Journal of Geodesy*, Vol. 76, No. 1, 2002, pp. 36–40.

[38] Bowring, B. R., "Transformation from spatial to geographical coordinates," *Survey review*, Vol. 23, No. 181, 1976, pp. 323–327.

[39] Vanicek, P. and Krakiwsky, E. J., *Geodesy: the concepts*, Elsevier, 2013.

[40] Pick, M. and Šimon, Z., "Closed formulae for transformation of the cartesian coordinate system into a system of geodetic coordinates," *Studia geophysica et geodaetica*, Vol. 29, No. 2, 1985, pp. 112–119.

[41] Fotiou, A., “A pair of closed expressions to transform geocentric to geodetic coordinates,” *Zeitschrift für Vermessungswesen*, Vol. 123, No. 1998, 1998, pp. 133–135.

[42] Vermeille, H., “Computing geodetic coordinates from geocentric coordinates,” *Journal of Geodesy*, Vol. 78, No. 1-2, 2004, pp. 94–95.

[43] Strang, G. and Borre, K., *Linear algebra, geodesy, and GPS*, Siam, 1997.

[44] Teunissen, P. J., Kleusberg, A., and Teunissen, P., *GPS for Geodesy*, Vol. 2, Springer Berlin, 1998.

[45] Hofmann-Wellenhof, B., Lichtenegger, H., and Collins, J., *Global positioning system: theory and practice*, Springer Science & Business Media, 2012.

[46] Torge, W., “Regional gravimetric geoid calculations in the north sea test area,” *Marine Geodesy*, Vol. 3, No. 1-4, 1980, pp. 257–271.

[47] Lupash, L. O., “A new algorithm for the computation of the geodetic coordinates as a function of earth-centered earth-fixed coordinates,” *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 6, 1985, pp. 787–789.

[48] Feltens, J., “Vector method to compute the Cartesian (X, Y, Z) to geodetic ({\ phi}, λ, h) transformation on a triaxial ellipsoid,” *Journal of Geodesy*, Vol. 83, No. 2, 2009, pp. 129–137.

[49] Zhang, C., Hsu, H., Wu, X., Li, S., Wang, Q., Chai, H., and Du, L., “An alternative algebraic algorithm to transform Cartesian to geodetic coordinates,” *Journal of Geodesy*, Vol. 79, No. 8, 2005, pp. 413–420.

[50] Fukushima, T., “Transformation from Cartesian to geodetic coordinates accelerated by Halleys method,” *Journal of Geodesy*, Vol. 79, No. 12, 2006, pp. 689–693.

[51] Turner, J. D., "A non-iterative and non-singular perturbation solution for transforming Cartesian to geodetic coordinates," *Journal of Geodesy*, Vol. 83, No. 2, 2009, pp. 139–145.

[52] Ligas, M. and Banasik, P., "Conversion between Cartesian and geodetic coordinates on a rotational ellipsoid by solving a system of nonlinear equations," *Geodesy and Cartography*, Vol. 60, No. 2, 2011, pp. 145–159.

[53] Lupi, V., Turner, J., and Chun, H., "Transform Methods for Precision Continuum and Control Models of Flexible Space Structures," in "Proceedings of the AIAA/AAS ASTRODYNAMICS Conference," Portland, Oregon, 1990, pp. 680 – 689.

[54] Lupi, V., Chun, H., and Turner, J., "Distributed Modeling and Control of Flexible Structures," in "Presented at the IFAC Workshop in Dynamics and control of Flexible Aerospace Structures: Modeling and Experimental Verification," Huntsville, Alabama, 1991.

[55] Lupi, V., Chun, H., and Turner, J., "Distributed Control and Simulation of a Bernoulli-Euler Beam," *Journal of Guidance, Control and Dynamics*, Vol. 15, No. 3, 1992, pp. 729 – 734.

[56] Meirovitch, L. and Quinn, R., "Equations of motion for maneuvering flexible spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 5.

[57] Meirovitch, L. and Stemple, T., "Hybrid equations of motion for flexible multibody systems using quasicoordinates," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 4.

[58] Junkins, J. L. and Kim, Y., *Introduction to dynamics and control of flexible structures*, AIAA, 1993.

[59] Lee, S. and Junkins, J., "Explicit Generalizations of Lagranges' Equations for Hybrid Coordinate dynamical Systems," *Journal of Guidance, Control and Dynamics*, Vol. 15, No. 6, 1992, pp. 1443 – 1452.

[60] Junkins, J. and Turner, J., *Optimal Spacecraft Rotational Maneuvers, studies in Astronautics*, Elsevier Scientific Publishing company, New York, NY, 1985.

[61] Turner, J. and Junkins, J., "Optimal large-angle single-axis rotational maneuvers of flexible spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 3, No. 6.

[62] Turner, J. and Chun, H., "Optimal distributed control of a flexible spacecraft during a large-angle maneuver," *Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 3, 1984, pp. 257 – 264.

[63] Wie, B., Sinha, R., and Liu, Q., "Robust time-optimal control of uncertain structural dynamic systems," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 5.

[64] Boškovic, J., Li, S., and Mehra, R., "Robust adaptive variable structure control of spacecraft under control input saturation," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1.

[65] Turner, J. D. and Elgohary, T. A., "Generalized Frequency Domain State-Space Models for Analyzing Flexible Rotating Spacecraft," in "Advances in Astronautical Science: The Kyle T. Alfriend Astrodynamics Symposium," Vol. 139, 2011, pp. 483 – 500.

[66] Elgohary, T. A. and Turner, J. D., "Generalized Frequency Domain Modeling and Analysis For A Flexible Rotating Spacecraft," in "AIAA Modeling and Simulation Technologies (MST) Conference," AIAA, Boston, MA, 2013.

[67] Turner, J. D. and Elgohary, T. A., "Generalized Frequency Domain State-Space Models for Analyzing Flexible Rotating Spacecraft," *The Journal of the Astronautical Sciences*, Vol. 59, No. 1-2, 2012, pp. 459–476.

[68] HALE, A. L., Lisowski, R. J., and DAHL, W. E., "Optimal simultaneous structural and control design of maneuvering flexible spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 1.

[69] Maganti, G. B. and Singh, S. N., "Simplified adaptive control of an orbiting flexible spacecraft," *Acta Astronautica*, Vol. 61, No. 7, 2007, pp. 575 – 589.

[70] Junkins, J., Rahman, Z., and Bang, H., "Near-minimum-time control of distributed parameter systems-Analytical and experimental results," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 2.

[71] Breakwell, J., "Optimal feedback slewing of flexible spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 4, No. 5.

[72] Ben-Asher, J., Burns, J., and Cliff, E., "Time-optimal slewing of flexible spacecraft," *Journal of guidance, control, and dynamics*, Vol. 15, No. 2.

[73] Singh, G., Kabamba, P., and McClamroch, N., "Planar, time-optimal, rest-to-rest slewing maneuvers of flexible spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 12, No. 1.

[74] Dwivedy, S. and Eberhard, P., "Dynamic analysis of flexible manipulators, a literature review," *Mechanism and Machine Theory*, Vol. 41, No. 7, 2006, pp. 749–777.

[75] Elgohary, T. A., Turner, J. D., and Junkins, J. L., "Dynamics and Controls of a Generalized Frequency Domain Model Flexible Rotating Spacecraft," in "AIAA SpaceOps Conference," AIAA, Pasadena, CA, 2014.

[76] Elgohary, T. A., Turner, J. D., and Junkins, J. L., “Analytic Transfer Functions for the Dynamics & Control of Flexible Rotating Spacecraft Performing Large Angle Maneuvers,” Submitted to the Journal of Astronautical Sciences.

[77] Hu, Q. and Ma, G., “Variable structure control and active vibration suppression of flexible spacecraft during attitude maneuver,” *Aerospace Science and Technology*, Vol. 9, No. 4, 2005, pp. 307–317.

[78] Hu, Q. and Ma, G., “Spacecraft vibration suppression using variable structure output feedback control and smart materials,” *Journal of vibration and acoustics*, Vol. 128, No. 2, 2006, pp. 221–230.

[79] Elgohary, T. A. and Turner, J. D., “Generalized Frequency Domain Solution for a Hybrid Rigid Hub Timoshenko Beam Rotating Aerospace Structure,” in “AIAA/AAS Astrodynamics Specialist Conference,” AIAA, San Diego, CA, 2014.

[80] Turner, J. D., Elgohary, T. A., Majji, M., and Junkins, J. L., “High Accuracy Trajectory and Uncertainty Propagation Algorithm for Long-Term Asteroid Motion Prediction,” *Adventures on the Interface of Mechanics and Control: K. T. Alfriend, M. Akella, J. E. Hurtado, J. Juang, and J. D. Turner*, pp. 15–34.

[81] Elgohary, T., Turner, J. D., and Junkins, J. L., “High-Order Analytic Continuation and Numerical Stability Analysis for the Classical Two-Body Problem,” *Advances in Astronautical Sciences: The Jer-Nan Juan Astrodynamics Symposium*, Vol. 147, 2012, pp. 627–646.

[82] Elgohary, T. A. and Turner, J. D., “State Transition Tensor Models for the Uncertainty Propagation of the Two-Body Problem,” *Advances in Astronautical*

Sciences: AAS/AIAA Astrodynamics Conference, Vol. 150, 2014, pp. 1171–1194.

[83] Battin, R. H., *An introduction to the mathematics and methods of astrodynamics*, AIAA, 1999.

[84] Schaub, H. and Junkins, J. L., *Analytical mechanics of space systems*, AIAA, 2003.

[85] Fehlberg, E., "Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems," Tech. rep., NASA, 1969.

[86] Filippi, S. and Gräf, J., "New Runge–Kutta–Nyström formula-pairs of order 8 (7), 9 (8), 10 (9) and 11 (10) for differential equations of the form $y" = f(x, y)$," *Journal of computational and applied mathematics*, Vol. 14, No. 3, 1986, pp. 361–370.

[87] Fox, K., "Numerical integration of the equations of motion of celestial mechanics," *Celestial mechanics*, Vol. 33, No. 2, 1984, pp. 127–142.

[88] Berry, M. M. and Healy, L. M., "Implementation of Gauss-Jackson integration for orbit propagation," *The Journal of the Astronautical Sciences*, Vol. 52, No. 3, 2004, pp. 331–357.

[89] Dormand, J., El-Mikkawy, M., and Prince, P., "High-order embedded Runge-Kutta-Nystrom formulae," *IMA Journal of Numerical Analysis*, Vol. 7, No. 4, 1987, pp. 423–430.

[90] Montenbruck, O., "Numerical integration methods for orbital motion," *Celestial Mechanics and Dynamical Astronomy*, Vol. 53, No. 1, 1992, pp. 59–69.

[91] Hadjifotinou, K. and Gousidou-Koutita, M., "Comparison of numerical methods for the integration of natural satellite systems," *Celestial Mechanics and Dynamical Astronomy*, Vol. 70, No. 2, 1998, pp. 99–113.

[92] Sharp, P. W., "N-body simulations: The performance of some integrators," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 32, No. 3, 2006, pp. 375–395.

[93] Bai, X. and Junkins, J. L., "Modified Chebyshev-Picard Iteration Methods for Orbit Propagation," *The Journal of the Astronautical Sciences*, Vol. 58, No. 4, 2011, pp. 583–613.

[94] Bai, X. and Junkins, J. L., "Modified Chebyshev-Picard iteration methods for solution of initial value problems," *The Journal of the Astronautical Sciences*, Vol. 59, No. 1-2, 2012, pp. 327–351.

[95] Elgohary, T., Dong, L., Junkins, J., and Atluri, S., "A Simple, Fast, and Accurate Time-Integrator for Strongly Nonlinear Dynamical Systems," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 100, No. 3, 2014, pp. 249–275.

[96] Belytschko, T., "A survey of numerical methods and computer programs for dynamic structural analysis," *Nuclear Engineering and Design*, Vol. 37, No. 1, 1976, pp. 23–34.

[97] Noor, A. K. and Lambiotte Jr, J. J., "Finite element dynamic analysis on CDC STAR-100 computer," *Computers & Structures*, Vol. 10, No. 1, 1979, pp. 7–19.

[98] Belytschko, T., Lin, J. I., and Chen-Shyh, T., "Explicit algorithms for the nonlinear dynamics of shells," *Computer methods in applied mechanics and engineering*, Vol. 42, No. 2, 1984, pp. 225–251.

[99] Newmark, N. M., "A Method of Computation for Structural Dynamics," *Journal of the Engineering Mechanics Division*, Vol. 85, No. 3, 1959, pp. 67–94.

[100] Hilber, H. M., Hughes, T. J., and Taylor, R. L., "Improved numerical dissipation for time integration algorithms in structural dynamics," *Earthquake Engineering & Structural Dynamics*, Vol. 5, No. 3, 1977, pp. 283–292.

[101] Subbaraj, K. and Dokainish, M., "A survey of direct time-integration methods in computational structural dynamics-II. Implicit methods," *Computers & Structures*, Vol. 32, No. 6, 1989, pp. 1387–1401.

[102] Dokainish, M. and Subbaraj, K., "A survey of direct time-integration methods in computational structural dynamics-I. Explicit methods," *Computers & Structures*, Vol. 32, No. 6, 1989, pp. 1371–1386.

[103] Atluri, S. N., *Methods of computer modeling in engineering & the sciences*, Vol. 1, Tech Science Press Palmdale, 2005.

[104] Dong, L., Alotaibi, A., Mohiuddine, S., and Atluri, S., "Computational methods in engineering: a variety of primal & mixed methods, with global & local interpolations, for well-posed or ill-Posed BCs," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 99, No. 1, 2014, pp. 1–85.

[105] Dai, H.-H., Schnoor, M., and Atluri, S. N., "A simple collocation scheme for obtaining the periodic solutions of the Duffing equation, and its equivalence to the high dimensional harmonic balance method: subharmonic oscillations," *Computer Modeling in Engineering and Sciences*, Vol. 84, No. 5, 2012, pp. 459–497.

[106] Bai, X., *Modified Chebyshev-Picard iteration methods for solution of initial value and boundary value problems*, Ph.D. thesis, Texas A&M University, 2010.

[107] Probe, A., Macomber, B., Kim, D., Woollands, R., and Junkins, J., "Terminal Convergence Approximation Modified Chebyshev Picard Iteration for Efficient Numerical Integration of Orbital Trajectories," Advanced Maui Optical and Space Surveillance Technologies Conference, Maui, Hawaii, 2014.

[108] Kim, D., Junkins, J. L., and Turner, J. D., "Multisegment Scheme Applications to Modified Chebyshev Picard Iteration Method for Highly Elliptical Orbits," *Mathematical Problems in Engineering*, Vol. 2015.

[109] Younes, A. B., *ORTHOGONAL POLYNOMIAL APPROXIMATION IN HIGHER DIMENSIONS: APPLICATIONS IN ASTRODYNAMICS*, Ph.D. thesis, Texas A&M University, 2013.

[110] Buhmann, M. D., *Radial basis functions: theory and implementations*, Vol. 5, Cambridge university press Cambridge, 2003.

[111] Elnagar, G. N. and Kazemi, M. A., "Pseudospectral Legendre-based optimal computation of nonlinear constrained variational problems," *Journal of computational and applied mathematics*, Vol. 88, No. 2, 1998, pp. 363–375.

[112] Liu, C.-S. and Atluri, S. N., "A Globally Optimal Iterative Algorithm Using the Best Descent Vector $x = \lambda[\alpha_c F + B^T F]$, with the Critical Value α_c, for Solving a System of Nonlinear Algebraic Equations F (x)= 0," *Computer Modeling in Engineering and Sciences*, Vol. 84, No. 6, 2012, p. 575.

[113] Fornberg, B. and Wright, G., "Stable computation of multiquadric interpolants for all values of the shape parameter," *Computers & Mathematics with Applications*, Vol. 48, No. 5, 2004, pp. 853–867.

[114] Fornberg, B. and Piret, C., "A stable algorithm for flat radial basis functions on a sphere," *SIAM Journal on Scientific Computing*, Vol. 30, No. 1, 2007, pp.

60–80.

[115] Fornberg, B., Larsson, E., and Flyer, N., “Stable computations with Gaussian radial basis functions,” *SIAM Journal on Scientific Computing*, Vol. 33, No. 2, 2011, pp. 869–892.

[116] Rippa, S., “An algorithm for selecting a good value for the parameter c in radial basis function interpolation,” *Advances in Computational Mathematics*, Vol. 11, No. 2-3, 1999, pp. 193–210.

[117] Fasshauer, G. E. and Zhang, J. G., “On choosing optimal shape parameters for RBF approximation,” *Numerical Algorithms*, Vol. 45, No. 1-4, 2007, pp. 345–368.

[118] Fasshauer, G. F., *Meshfree approximation methods with MATLAB*, World Scientific Publishing Co., Inc., 2007.

[119] Cvetićanin, L., “Ninety years of Duffing’s equation,” *Theoretical and Applied Mechanics*, Vol. 40, No. 1, 2013, pp. 49–63.

[120] Jones, B. A., “Orbit propagation using Gauss-Legendre collocation,” in “Proceedings of the 2012 AIAA/AAS Astrodynamics Specialist Conference, AIAA,” Vol. 4967, 2012, pp. 1–16.

[121] Jones, B. A. and Anderson, R. L., “A survey of symplectic and collocation integration methods for orbit propagation,” in “AAS/AIAA Spaceflight Mechanics Conference,” , 2012, pp. 1–20.

[122] Bradley, B. K., Jones, B. A., Beylkin, G., Sandberg, K., and Axelrad, P., “Bandlimited implicit Runge–Kutta integration for astrodynamics,” *Celestial Mechanics and Dynamical Astronomy*, Vol. 119, No. 2, 2014, pp. 143–168.

[123] Elgohary, T. A., Dong, L., Junkins, J. L., and Atluri, S. N., "Time Domain Inverse Problems in Nonlinear Systems Using Collocation & Radial Basis Functions," *CMES: Computer Modeling in Engineering & Sciences*, Vol. 100, No. 1, 2014, pp. 59–84.

[124] Lewis, F. L. and Syrmos, V. L., *Optimal Control*, Wiley, New York, 1995.

[125] Lewis, F. L., Vrabie, D., and Syrmos, V. L., *Optimal control*, John Wiley & Sons, 2012.

[126] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical recipes 3rd edition: The art of scientific computing*, Cambridge university press, 2007.

[127] von Stryk, O. and Bulirsch, R., "Direct and indirect methods for trajectory optimization," *Annals of operations research*, Vol. 37, No. 1, 1992, pp. 357–373.

[128] Bai, X. and Junkins, J. L., "Modified Chebyshev-Picard iteration methods for solution of boundary value problems," *The Journal of the Astronautical Sciences*, Vol. 58, No. 4, 2011, pp. 615–642.

[129] Kang, W. and Bedrossian, N., "Pseudospectral Optimal Control Theory Makes Debut Flight, Saves NASA $ 1 M in," *SIAM news*, Vol. 40, No. 7.

[130] Ross, I. M. and Karpenko, M., "A review of pseudospectral optimal control: From theory to flight," *Annual Reviews in Control*, Vol. 36, No. 2, 2012, pp. 182–197.

[131] Elnagar, G., Kazemi, M. A., and Razzaghi, M., "The pseudospectral Legendre method for discretizing optimal control problems," *Automatic Control, IEEE Transactions on*, Vol. 40, No. 10, 1995, pp. 1793–1796.

[132] Benson, D. A., Huntington, G. T., Thorvaldsen, T. P., and Rao, A. V., "Direct trajectory optimization and costate estimation via an orthogonal collocation method," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1435–1440.

[133] Fahroo, F. and Ross, I. M., "Costate estimation by a Legendre pseudospectral method," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 2, 2001, pp. 270–277.

[134] Elnagar, G. N. and Kazemi, M. A., "Pseudospectral Chebyshev optimal control of constrained nonlinear dynamical systems," *Computational Optimization and Applications*, Vol. 11, No. 2, 1998, pp. 195–217.

[135] Garg, D., Patterson, M., Hager, W. W., Rao, A. V., Benson, D. A., and Huntington, G. T., "A unified framework for the numerical solution of optimal control problems using pseudospectral methods," *Automatica*, Vol. 46, No. 11, 2010, pp. 1843–1851.

[136] Rad, J., Kazem, S., and Parand, K., "A numerical solution of the nonlinear controlled Duffing oscillator by radial basis functions," *Computers & Mathematics with Applications*, Vol. 64, No. 6, 2012, pp. 2049–2065.

[137] Dai, H., Yue, X., and Atluri, S., "Solutions of the von Kármán plate equations by a Galerkin method, without inverting the tangent stiffness matrix," *Journal of Mechanics of Materials and Structures*, Vol. 9, No. 2, 2014, pp. 195–226.

[138] Hu, N. and Wen, X., "The application of Duffing oscillator in characteristic signal detection of early fault," *Journal of Sound and Vibration*, Vol. 268, No. 5, 2003, pp. 917–931.

[139] Suhardjo, J., Spencer Jr, B., and Sain, M., "Non-linear optimal control of a Duffing system," *International journal of non-linear mechanics*, Vol. 27, No. 2, 1992, pp. 157–172.

[140] Trefethen, L. N., *Spectral methods in MATLAB*, Vol. 10, SIAM, 2000.