1-1-2003

General methodology for multi-objective optimal design of control-structure nonlinear mechanisms with symbolic computing

Jason Carrigan
Iowa State University

General methodology for multi-objective optimal design of control-structure nonlinear mechanisms with symbolic computing

by

Jason Carrigan

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Mechanical Engineering

Program of Study Committee:
Atul G. Kelkar, Major Professor
Prakash Krishnaswami
Judy Vance
Ranga Narayananaswami

Iowa State University

Ames, Iowa

2003

Graduate College
Iowa State University

This is to certify that the master's thesis of

Jason Carrigan

has met the thesis requirements of Iowa State University

Signatures have been redacted for privacy

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES vii

ACKNOWLEDGEMENTS xi

1 INTRODUCTION 1

1.1 Literature Review 3

1.2 Contributions of this Work 6

1.3 Thesis Organization 7

2 MATHEMATICAL MODELLING OF SYSTEM DYNAMICS . . . 8

2.1 Lagrangian Formulation 9

2.2 Constraint Equations and Lagrange Multipliers 12

2.3 Multibody Constrained Formulation 14

2.3.1 Development of the Constraint Equations 16

2.4 Mathematical Modelling of Controller Dynamics 19

2.5 Generic Form of Dynamic Controller 19

2.6 Controller Dynamics in Lagrange Formulation 20

2.7 Controller Dynamics in Multibody Constrained Formulation 20

2.7.1 Important Considerations in Multibody Constrained Formulation 21

3 SENSITIVITY 25

3.1 Preliminaries 25

3.1.1 Sensitivity 25

3.1.2 Formulation of Sensitivity Equations 27

3.1.3 Derivation of Initial Conditions for Sensitivity 29

3.1.4 Error Checking 30

3.2 Sensitivity Equations in Lagrangian Formulation 32

3.3 Sensitivity Equations in the Multibody Constrained Formulation 33

4 INTEGRATED DESIGN 35

4.1 Optimization 35

4.1.1 Optimizing for System Response 38

4.1.2 The Min-Max Problem 38

4.1.3 Optimizing for Minimum Sensitivity 39

4.1.4 Optimizing for Minimum Variability 40

4.1.5 Optimizing for Manufacturability 41

4.2 Symbolic Derivation of Response and Sensitivity Equations 42

5 DOUBLE SLIDER EXAMPLE 43

5.1 Double Slider System Problem Statement 43

5.1.1 Lagrangian Formulation 44

5.1.2 Optimizing for Tracking Performance 48

5.1.3 Optimization for Variability 50

5.1.4 Optimization for Minimum Sensitivity 51

5.1.5 Optimization for Manufacturability 52

5.2 Multibody Constrained Formulation 54

5.2.1 Developing Constraint Matrix Φ and Force Vector Q for Double Slider Mechanism 56

5.2.2 Optimization for Performance with PD controller 60

5.2.3 Performance Optimization with Generic Controller 61

6 CONSTRUCTION LOADER LINKAGE PROBLEM 64

6.0.4 Controller Optimization for Bucket Leveling 65

6.0.5 Control-Structure Integrated Design for Bucket Leveling 68

6.0.6 Sensitivity Optimization Using Integrated Design 68

6.0.7 Manufacturability Optimization Using Integrated Design 69

7 CONCLUSIONS 75

7.1 Suggestions for Future Work 76

BIBLIOGRAPHY 78

LIST OF TABLES

5.1	Optimal design variables for double slider problem	55
5.2	Tolerance values of optimal systems for double slider	56
6.1	Mass, inertia and initial center of mass location for links	67
6.2	Optimal design variables for construction linkage	70
6.3	Tolerance values for construction linkage	71

LIST OF FIGURES

2.1 Coordinate frames for developing constraint equations for a revolute joint between two bodies 17

2.2 Coordinate frame for developing the generalized force vector Q on a body 18

3.1 Function f evaluated at b_0 and $b_0 + \Delta b$ 27

3.2 Error plot 32

4.1 Integrated design scheme 36

4.2 Data flow between modules 37

4.3 Performance objective for the min-max problem 39

5.1 Double slider mechanism 44

5.2 Position of m_2 for optimal tracking 49

5.3 Velocity and desired velocity of m_2 for optimal tracking 49

5.4 θ_L and $\dot{\theta}_L$ for optimal tracking 49

5.5 Variability of the velocity of m_2 for optimal tracking 49

5.6 Control input for optimal tracking 50

5.7 Position of m_2 for optimal variability 51

5.8 Velocity and desired velocity of m_2 for optimal variability 51

5.9 θ_L and $\dot{\theta}_L$ for optimal variability 52

5.10 Variability of the velocity of m_2 for optimal variability 52

5.11 Control input for optimal variability 52
5.12 Position of m_2 for optimal sensitivity 53
5.13 Velocity and desired velocity of m_2 for optimal sensitivity 53
5.14 θ_L and $\dot{\theta}_L$ for optimal sensitivity 54
5.15 Variability of the velocity of m_2 for optimal sensitivity 54
5.16 Control input for optimal sensitivity 54
5.17 Position of m_2 for optimal manufacturability 55
5.18 Velocity and desired velocity of m_2 for optimal manufacturability 55
5.19 θ_L and $\dot{\theta}_L$ for optimal manufacturability 56
5.20 Variability of the velocity of m_2 for optimal manufacturability . 56
5.21 Control input for optimal manufacturability 57
5.22 Position of m_2 for optimal tracking using the multibody constrained formulation 61
5.23 Velocity and desired velocity of m_2 for optimal tracking using the multibody constrained formulation 61
5.24 Control input for optimal tracking using the multibody constrained formulation 62
5.25 Position of m_2 for optimal tracking using the multibody constrained formulation with the generic controller 62
5.26 Velocity and desired velocity of m_2 for optimal tracking using the multibody constrained formulation with the generic controller . . 62
5.27 Control input for optimal tracking using the multibody constrained formulation with the generic controller 63

6.1 Picture of construction linkage on tractor 65
6.2 Picture of construction linkage on tractor 66
6.3 Trajectory of boom for bucket leveling 67

6.4 Trajectory of bucket for bucket leveling 67
6.5 Control input for bucket leveling 68
6.6 Trajectory of boom for bucket leveling using the integrated design approach 69
6.7 Trajectory of bucket for bucket leveling using the integrated design approach 69
6.8 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for bucket leveling using the integrated design approach 70
6.9 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for bucket leveling using the integrated design approach 70
6.10 Control input for bucket leveling using the integrated design approach 71
6.11 Trajectory of boom for sensitivity optimized system using the integrated design approach 71
6.12 Trajectory of bucket for sensitivity optimized system using the integrated design approach 71
6.13 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for sensitivity optimized system using the integrated design approach 72
6.14 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for sensitivity optimized system using the integrated design approach 72
6.15 Control input for sensitivity optimized system using the integrated design approach 72

6.16 Trajectory of boom for optimal manufacturability using the integrated design approach 73

6.17 Trajectory of bucket for optimal manufacturability using the integrated design approach 73

6.18 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for optimal manufacturability using the integrated design approach 73

6.19 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for optimal manufacturability using the integrated design approach 73

6.20 Control input for optimal manufacturability using the integrated design approach 74

1 INTRODUCTION

In spite of the abundance of existing literature, research in mechanism analysis and synthesis continues to be an active research area. With the advent of the computer age, mechanism design methodology has undergone significant changes and improved drastically; however, there is still room for further improvements. One of the primary reasons being, with the availability of superfast computers, the design methods which were perceived to be impractical a few years ago, are now very much in the realm of possibilities. This computing power of present day machines (mainframe as well as desktop) has opened up numerous new research directions.

One of the primary shortcomings of the current design methodologies is the lack of integrated design framework that allows optimization over a combination of several design spaces. Most of the design methodologies use sequential optimization schemes which target one design space at a time. As a result, the design obtained is not *truly* optimal. Moreover, in many design procedures, most design decisions are left to the designers judgement [1]. There exists some iterative procedures where manual intervention is needed to iterate between two optimal designs which are a part of a sequential scheme. Over the last decade, there has been some work done in developing integrated design schemes that optimize over design variables arising from several domains. For example, optimal design of a mechanism may involve design variables related to the geometry or shape of the links, material properties, and possibly control design variables of a control system. This type of integrated approach typically results in a computationally intensive, highly nonlinear, and high order design problem. In addition, the global optimal in such cases is often not possible to find and one has to look for the best possible suboptimal design.

There are several other considerations that enter into the selection of an appropriate optimization method that is suitable for a particular problem at hand. There is no unique methodology that is suited to all problems. Moreover, there are considerations that arise from a designers personal preferences towards possible approaches available to the designer. The designer may not want to use an algorithm like a black box. He or she may want to play an active role in directing the search for an optimal solution. The designer may want to analyze the optimization process at every step to gather more insight into the problem. For example, a designer may want to find out what happens if a particular variable is changed, and how will it affect the overall performance of the system. All such factors make optimization an art and not just engineering. In summary, a good optimization strategy is the one that will allow the designer to explore the *design space* and make better decisions as to what tradeoffs can be made and how those tradeoffs will affect the overall design.

The conventional mechanical system design follows a sequential process. First, the system is designed based on optimal parameters derived by minimizing some performance index which purely depends on the structural strength considerations. Once a structural design is finalized and the system is built, the control designer designs an optimal control system purely from control performance considerations. This approach has been in practice for several years and continues to be the case even today. The design obtained from this sequential approach tends to be not optimal in the true sense since, while optimizing for the structural design, control design parameters and performance were not considered; and similarly, while optimization for control design, structural parameters and performance are not considered. A better optimal design would be the one in which a combined performance function (structural and control) can be minimized with respect to a set of structural and controller parameters. The work of this thesis is focused on this very aspect of optimization. In particular, a more versatile design tool is developed wherein an integrated design of nonlinear multibody controlled mechanical system can be performed with considerations to performance, sensitivity, robustness, variability and manufacturability. The methodology developed also allows numeric as well as symbolic data inputs. The design variables can be derived symbolically which allows analytical computation of sensitivity functions. The feature is extremely valuable in analysis and synthesis as it gives a designer important information about the design space and the effect of variations in design variables on a performance function.

1.1 Literature Review

There has been very little work in the development of general purpose methodology for sensitivity based design and optimization methodology for controlled mechanical systems that combines structural as well as controller parameters in one optimization process. Although there is an abundance of literature on optimization and optimal design in general, there is very limited literature on integrated design of controlled nonlinear mechanical systems and multi-objective design. As a result, the literature review will be fairly brief.

Numerical methods have been used for centuries to solve complex mathematical problems. Computers have been utilized within the last fifty years to perform the tedious calculations that are common in many numerical methods. An iterative numerical method is often used to successively adjust a parameter until a desired result is obtained [2], such a process is usually called optimization. The gradient method was first presented by Cauchy [1847]. Modern optimization methods were pioneered by Courants paper on penalty functions [1943], Dantzig's paper on the simplex method for linear programming [1951], and Karush, Kuhn and Tucker who derived the "KKT" optimality conditions for constrained problems in 1939 and 1951 [3]. Large scale optimization began primarily in 1947 with US Air Force problems in which linear programs were formulated to solve large scale problems with many design variables [4]. Within the last three decades, many numerical optimization methods have been developed and commonly applied to a wide variety of problems [3], [5].

The first step in the optimization problem formulation for dynamic systems is to obtain governing differential equations that describe the motion of the system. Two commonly employed approaches for multibody systems are Newtonian and Lagrangian formulations. The Lagrangian formulation [6], [7], is often preferred over a Newtonian approach due to its simplicity as a result of the energy based approach. Lagrangian formulation also yields nice, closed-form equations which are easy to work with. However, in the case of systems with multiple bodies, the complexity of equations grows exponentially with the number of bodies which leads to the problem of equation swell. An alternative approach which can be used to overcome this shortcoming of Lagrangian approach is the multibody constrained formulation [8]. This approach has gained importance since computational power is not an issue. Another important piece of information that is needed, especially in gradient-based optimization schemes, is sensitivity derivatives of a performance function with respect to design variables. The literature on sensitivity computations include several references, some examples being [1], [9] and [10]. In [9], a direct differentiation-based scheme is given for sensitivity computation whereas in [10], a singular value decomposition method is used.

Optimization for minimum sensitivity using nonlinear programming has been successfully implemented for planar mechanisms [11]. However, there has been very little work done in integrating sensitivity based optimal design in a general optimization framework. One of the few available references on optimal design of structure using sensitivity methods is [12], wherein the optimization approach used did not address the integrated design concept, however, the method could be extended to include control parameters. Moreover, this thesis is specifically focused on the design for vibration suppression. In

[13], a sensitivity based design focused on manufacturability was presented, however a connection between sensitivity and manufacturability was indirect.

Another area that has emerged over the last two decades which had a huge impact on the optimization area is automatic differentiation and symbolic computing. There are several references pertaining to this area some of which include, [14], [15], [16], [17] and [18]. Use of symbolic computing can greatly benefit sensitivity analysis and optimization of mechanical systems. Availability of better, more efficient symbolic software and extended differentiation capabilities of new automatic differentiation algorithms can now enable integration of multi-objective analysis and synthesis tools very effectively. In this research, these capabilities are exploited for developing a general purpose, modular optimization tool that can be used not only for integrated design but also for various other optimization tasks.

Integrated design of the structure and controller has been successfully implemented in the past; however, most of the designs focused on linear systems. A generalized approach to the design of nonlinear controlled mechanisms has not been well developed. Moreover, control-structure integrated design with considerations to performance has been investigated [19] for dynamic controllers such as H_2 and H_∞ controllers; however, a generalized controller formulation has not been explored. A genetic algorithm approach was used in [20] for optimal design of structure and controller. This method is much more computationally intensive and relatively complicated if the user wants to have control over each step of the optimization process. Other related work in the area of control structure integrated design can be found in [21], [22], [23], [24] and [25]. The methods presented in this literature, however, focus on specific objectives; the methodology is not generalized and structural models are linear in most cases.

In [26] and [27], the multi-objective approach is used for integrated control-structure design; however, the design objectives are only related to the quality of the system response, The objectives such as manufacturability or sensitivity were not considered and generalization through a symbolic input capability and a generalized controller framework was not addressed. The only work that has more comprehensive optimal design capabilities can be found in [28]. The methodology presented in this work however, addresses only the robustness aspect in the design.

In summary, based on an extensive literature review, it was concluded that much of the groundwork has been done in the area of dynamic analysis, sensitivity analysis and optimization of linear dynamic systems. However, the work in the area of multi-objective integrated design of controlled nonlinear mechanical systems is very limited. Also, the existing limited work in this area primarily focuses on one particular aspect of performance optimization and often times presents a methodology for a specific problem at hand. A general purpose tool that can be used to solve several types of optimization problems for a generic nonlinear controlled mechanical system is not available. This research focused on developing such a tool using symbolic computing and a generic controller framework. The design tool developed in this research will allow optimal designs for performance, sensitivity, robustness, manufacturability and variability.

1.2 Contributions of this Work

There are four major contributions of this research which can greatly benefit the engineering design community.

1. A general purpose integrated design method for designing nonlinear controlled mechanisms has been developed. This methodology has alleviated the need for linearization of nonlinear systems to obtain an optimal design using integrated design techniques. Moreover, the use of the multibody constrained formulation allows a modular approach to model the dynamics of multibody systems. This approach also allows very efficient reformulation of complex systems in case of a change in the configuration of the system.

2. A symbolic preprocessor capability is developed which enables the designer to develop equations of motion even for large dimensional system and any order controller along with corresponding sensitivity equations very efficiently. With some problem specific information, these equations can be output directly to the optimization program. The modular program structure enables the designer to make changes in the optimization data and monitor each step of the optimization process very efficiently and easily.

3. Multi-objective optimization structure allows the designer to optimize over many different design spaces in order to obtain a better optimal solution.

4. A generalized method for implementing generic dynamic controller structure into closed loop system dynamics is developed which allows input of controller parameters along with structural parameters. Moreover, these inputs can be done in symbolic form. The controller parameters, structure and order can also be changed independently of structure dynamics.

1.3 Thesis Organization

The organization of this thesis is as follows: Chapter 2 presents the mathematical development of a dynamic model for multibody nonlinear system along with integration of equation of a dynamic controller; Chapter 3 focuses on the derivation of sensitivity equations; Chapter 4 presents the formulation of an integrated design framework as the optimization problem. The structure of the symbolic preprocessor and its role in optimization and sensitivity analysis is also discussed. Chapter 5 presents the optimization results for a proof-of-concept double slider example problem; Chapter 6 gives the results for the integrated design of a construction loader problem; and finally Chapter 7 gives conclusions and suggestions for future work.

2 MATHEMATICAL MODELLING OF SYSTEM DYNAMICS

The first step in the optimization of dynamic systems is to obtain a mathematical model representing the dynamics of motion of a system. Two basic approaches to obtain the equations of motion are the Newtonian formulation and the Lagrangian formulation. An alternative approach that has emerged since the advent of computers is the multi-body constrained formulation. The Newtonian formulation is a vector-based iterative method which is cumbersome to use and therefore not widely used in the optimization community. Both, Lagrangian and multibody constrained formulation are widely used, however. The choice of a particular method depends on the problem at hand. In particular, the Lagrangian approach is easy to use when the number of bodies involved in a multibody system is low. With an increase in the number of bodies, the complexity of the Lagrangian formulation grows exponentially. This phenomena is referred to as *equation swell.* the multibody constrained formulation, on the other hand, yields a large number of equations, however each individual equation is relatively simple. Thus, with adequate computational power, the multibody constrained formulation is the method of choice.

In this thesis, both formulations were developed in order to be able to compare and validate correctness of each approach. In particular, for the proof-of-concept system, both approaches were used to formulate the optimization problem and the results were compared to ensure accuracy of the formulations. When symbolic inputs are used, the multibody constrained formulation offers the only viable approach for systems with more that three or four bodies. This chapter will focus on the development of generic equations of motion for multibody nonlinear systems using both approaches.

2.1 Lagrangian Formulation

The Lagrangian formulation is based on an energy-based approach and therefore deals with only scalar quantities. This makes the development of the equations of motion much easier and less prone to errors compared to a Newtonian formulation which uses vector quantities. Given below is the derivation for the Lagrangian formulation taken primarily from [30].

For a system of N particles, the total kinetic energy of the system is given by

$$T = \frac{1}{2}\sum_{j=1}^{3N} m_j \dot{x}_j^2 \tag{2.1}$$

where m_j is the mass of particle j and x_j is the generalized coordinate of particle j. Now each cartesian coordinate x_j can be described as a function of generalized coordinates $q_i \ \ (i = 1, 2, ..., N)$ and time t as follows:

$$\begin{aligned} x_1 &= f_1(q_1, q_2, ..., q_n, t) \\ x_2 &= f_2(q_1, q_2, ..., q_n, t) \\ \vdots \quad & \quad \vdots \qquad\qquad \vdots \\ x_{3N} &= f_{3N}(q_1, q_2, ..., q_n, t). \end{aligned} \tag{2.2}$$

The kinetic energy of the system can be written in terms of these generalized coordinates as

$$T = \frac{1}{2}\sum_{j=1}^{3N} m_j \left(\sum_{i=1}^{n} \frac{\partial x_j}{\partial q_i}\dot{q}_i + \frac{\partial x_j}{\partial t} \right)^2 \tag{2.3}$$

where,

$$\dot{x}_j = \sum_{i=1}^{n} \frac{\partial x_j}{\partial q_i}\dot{q}_i + \frac{\partial x_j}{\partial t}. \tag{2.4}$$

The general momentum can be expressed as

$$p_i = \frac{\partial T}{\partial \dot{q}_i} \tag{2.5}$$

substituting for T from Eqn. (2.3) into Eqn. (2.5) yields

$$p_i = \sum_{j=1}^{3N} m_j \dot{x}_j \frac{\partial \dot{x}_j}{\partial \dot{q}_i} \tag{2.6}$$

which, after substituting for $\dot{x}_j$ from Eqn. (2.3) can be expressed as

$$p_i = \sum_{j=1}^{3N} m_j \dot{x}_j \frac{\partial x_j}{\partial q_i}. \tag{2.7}$$

The time rate of change of the generalized momentum is given by

$$\dot{p}_i = \frac{dp_i}{dt} = \sum_{j=1}^{3N} m_j \ddot{x}_j \frac{\partial x_j}{\partial q_i} + \sum_{j=1}^{3N} m_j \dot{x}_j \frac{d}{dt}\left(\frac{\partial x_j}{\partial q_i}\right). \tag{2.8}$$

The last term in Eqn. (2.8) can be evaluated as

$$\frac{d}{dt}\left(\frac{\partial x_j}{\partial q_i}\right) = \sum_{k=1}^{n} \frac{\partial^2 x_j}{\partial q_i \partial q_k} \dot{q}_k + \frac{\partial^2 x_j}{\partial q_i \partial t} = \frac{\dot{x}_j}{q_i} \tag{2.9}$$

substituting Eqn. (2.8) in Eqn. (2.1),

$$\frac{\partial T}{\partial q_i} = \frac{1}{2} \sum_{j=1}^{3N} m_j \dot{x}_j \frac{\partial \dot{x}_j}{\partial q_i}. \tag{2.10}$$

Now, combining Eqns. (2.8), (2.9) and (2.10) the time rate of change of momentum can be expressed as

$$\frac{dp_i}{dt} = \sum_{j=1}^{3N} m_j \ddot{x}_j \frac{\partial x_j}{\partial q_i} + \frac{\partial T}{\partial q_i} \tag{2.11}$$

According to Newton's law of motion,

$$m_j \ddot{x}_j = F_j + R_j \tag{2.12}$$

where F_j is a set of all the externally applied forces and R_j is a set of the forces due to workless constraints on the system. Combining Eqns. (2.12) and (2.11),

$$\dot{p}_i = \frac{dp_i}{dt} = \sum_{j=1}^{3N} F_j \frac{\partial x_j}{\partial q_i} + \sum_{j=1}^{3N} R_j \frac{\partial x_j}{\partial q_i} + \frac{\partial T}{\partial q_i}. \tag{2.13}$$

Noting that $\sum_{j=1}^{3N} F_j \frac{\partial x_j}{\partial q_i} = Q_i$ and R_j's do not do any work, $\sum_{j=1}^{3N} R_j \frac{\partial x_j}{\partial q_i} = 0$. Therefore, Eqn. (2.13) can be simplified to

$$\frac{dp_i}{dt} = Q_i + \frac{\partial T}{\partial q_i}. \tag{2.14}$$

Substituting Eqn. (2.5) into Eqn. (2.13) yields

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} = Q_i, \quad (i = 1, 2, ..., n) \tag{2.15}$$

which is known as the fundamental form of Lagrange's equations. It is to be noted that only the kinetic energy of the system is accounted for on the left hand side and the potential energy inputs are taken care of in the generalized force vector. This is not the most desirable form of Lagrange's equation. A more desirable form of Lagrange's equation uses the concept of the "Lagrangian" defined as

$$L = T - V \tag{2.16}$$

where T is the kinetic energy and V is the potential energy of the system. The conservative forces can be derived from the potential energy function and hence separated from Q_i in Eqn. (2.15) as follows,

$$Q_i = Q_i^c + Q_i^{nc} \tag{2.17}$$

where Q_i^c represents the conservative forces and Q_i^{nc} represents the non-conservative forces. Q_i^c's are derived from V using,

$$Q_i^c = -\frac{\partial V}{\partial q_i} \tag{2.18}$$

It should be noted that V is *not* a function of generalized velocities $\dot{q}$. Therefore, equations (2.15), (2.16) and (2.18) can be combined to give the standard form of Lagrange's equation as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i^{nc}, \quad (i = 1, 2, ..., n) \tag{2.19}$$

where Q_i^{nc} are the generalized, non-conservative forces not derivable from the potential energy function.

2.2 Constraint Equations and Lagrange Multipliers

If the system represented by Eqn. (2.19) is constrained, then the degrees of freedom for the system are reduced by the number of constraints acting on the system. For example, if there are m active constraints on the system described by Eqn. (2.19), the degrees of freedom for the system reduces to $n-m$ where n is the total number of generalized coordinates used to describe the system. In the case of constraints, the equations of motion for the system (2.19) have to be augmented to include the constraint relations. Let the constraint equations be given by

$$\Phi_j(q,t)=0, \quad (j=1,2,...,m) \tag{2.20}$$

where m is the number of constraints. If the constraints in Eqn. (2.20) are holonomic in nature, in theory, one can eliminate constraint equation by using these equations to describe m of the n generalized coordinates in terms of remaining $n-m$ generalized coordinates, and thereby reducing the number of equations in Eqn. (2.19) from n to $n-m$. However, in the case of non-holonomic constraints, the constraint equations cannot be used to eliminate the "extra" (dependent) generalized coordinates. In such a case, constraint equations have to be used together with Eqn. (2.19) to completely describe the system dynamics. In the case of non-holonomic constraints, the constraint equations cannot be given by Eqn. (2.20) as such constraints are not integrable. A more generic form of constraints which can be used to represent both types of constraints is given by

$$\sum_{i=1}^{n}\alpha_{ij}dq_j+\alpha_{it}dt=0 \qquad (i=1,2,...,m). \tag{2.21}$$

In the case of holonomic constraints $\alpha_{ij}=\frac{\partial\Phi_i}{\partial q_j}$ and $\alpha_{it}=\frac{\partial\Phi_i}{\partial t}$, where $\Phi_i=\Phi_i(q,t)$ are holonomic constraint equations.

In the presence of non-holonomic constraints, and sometimes in the case of holonomic constraints (for simplicity) one has to augment the system equations with the constraint equations using Lagrange multipliers. This results in the introduction of new m independent variables to be determined during the solution process. Given below is the procedure to augment the system equations using Lagrange multipliers. From the principle of virtual work [9], the virtual displacements of generalized coordinates satisfy the following condition in the case of frictionless constraints.

$$\sum_{j=1}^{n} \alpha_{ij}\delta q_j = 0 \qquad (i = 1, 2, ..., m) \tag{2.22}$$

Also, since the constraints are frictionless, the work done by the generalized constraint forces (C_i) in virtual displacements is zero, for example,

$$\sum_{j=1}^{n} C_j \delta q_j = 0 \quad \Rightarrow \quad C^T \delta q = 0 \tag{2.23}$$

Now, multiplying each of Eqn. (2.21) by the Lagrange multiplier λ_i $(i = 1, 2, ..., m)$ and subtracting from Eqn. (2.23),

$$\left[\sum_{j=1}^{n} C_j - \sum_{i=1}^{m} \alpha_{ji}\lambda_i\right] \delta q_j = 0 \tag{2.24}$$

In vector matrix form, Eqn. (2.24) can be written as,

$$\left[C - \alpha^T \lambda\right] [\delta q] \tag{2.25}$$

For Eqn. (2.25) to hold for any arbitrary $[\delta q]$, λ needs to be chosen such that

$$C - \alpha^T \lambda = 0 \quad \Rightarrow \quad C = \alpha^T \lambda \tag{2.26}$$

For a holonomic system, Eqn. (2.26) can be written as

$$C = \Phi_q^T \lambda \tag{2.27}$$

where $\Phi_q^T = \left[\frac{\partial \Phi_i}{\partial q_j}\right]^T$.

Now augmenting Eqn. (2.19) to include the constraint forces from Eqn. (2.27) and governing constraint equations from Eqn. (2.21),

$$\begin{array}{c} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i^{nc} + \sum_{j=1}^{m} \alpha_{ij}\lambda_j \qquad (i = 1, 2, ..., n) \\ \sum_{i=1}^{n} \alpha_{ij} dq_j + \alpha_{it} dt = 0 \qquad (i = 1, 2, ..., m) \end{array} \tag{2.28}$$

Note that the system of equations (2.28) has $(n+m)$ equations and $(n+m)$ unknowns and can be solved simultaneously. One advantage of using Lagrange multipliers is that constraint forces can be easily determined once λ's are known. For holonomic systems, Eqn. (2.28) takes the form

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i^{nc} + \Phi_q^T \lambda \qquad (i = 1, 2, ..., n) \\ \sum_{i=1}^{n} \frac{\partial \Phi_i}{\partial q_j} dq_j + \frac{\partial \Phi_i}{\partial t} dt = 0 \qquad (i = 1, 2, ..., m) \end{aligned} \tag{2.29}$$

For multibody systems with a small number of bodies, the formulation is simple and straight forward. However, when the number of bodies increases even beyond three or four, the differential equation for each coordinate is usually very complex and fairly messy. Also, if the system configuration is changed (for example, the addition another link, a change of joint location, a change of the controller type, etc.) the entire derivation has to be repeated again as contributions from individual bodies are not identifiable in system equations. This is one major drawback of the Lagrangian formulation. The complexity of the equations also proves to be prohibitive for use of symbolic computation. These shortcomings of the Lagrangian formulation can be avoided if one uses the multibody constrained formulation, especially if the computational power is not an issue.

2.3 Multibody Constrained Formulation

The multibody constrained formulation essentially uses the Lagrange approach at its core. The main idea is to use the maximum set of generalized coordinates for each body in conjunction with associated constraint equations for each body. For example, for a system of N bodies, there will ba $6N$ generalized coordinates leading to $6N$ differential equations. If each body is constrained by m independent constraints, then there will be $N \times m$ constraint equations. Therefore, this formulation leads to a large number of differential and constraint equations. However, each individual equation is very simple.

The simplicity of equations allows for symbolic computation and the modular structure allows for significant flexibility if the configuration of the system changes as it does not require reformulation of the complete set of equations. Thus, in the multibody constrained formulation, there is very little equations swell when a symbolic package, such as Maple©, is used to symbolically derive the equations. This is a very significant advantage over the Lagrange formulation. A more detailed discussion on this issue is given at the end of this section.

Given below is the formulation of equations of motion using the multibody constrained formulation. As it will be seen, it is an extension of the Lagrange formulation.

Recall the system of equations derived in the previous section given by Eqn. (2.28) for nonholonomic constraints and Eqn. (2.29) for holonomic constraints. Assume the case of holonomic constraints for simplicity. Then, in the multibody constrained formulation, Lagrange's equation is used to describe the dynamics of each body independently leading to six differential equations for each body. That is,

$$\frac{d}{dt}\left(\frac{\partial L^i}{\partial \dot{q}_{ij}}\right) - \frac{\partial L^i}{\partial q_{ij}} = Q_{ij}^{nc} + \lambda \Phi_q^{i^T} \qquad (i = 1, 2, ..., N, j = 1, 2, ..., 6) \tag{2.30}$$

where N is the number of bodies. The constraint equations are given by,

$$\Phi_j^i(q, t) = 0 \qquad (i = 1, 2, ..., N, j = 1, 2, ..., 6). \tag{2.31}$$

Equation (2.30) can be written in compact form as

$$M\ddot{q} - \Phi_q^T \lambda = Q^{nc} \tag{2.32}$$

where, M is a mass matrix, Φ is a matrix of constraint equations, λ is the vector of Lagrange multipliers and Q is a vector of generalized forces. Note that Q incorporates all generalized forces including conservative, nonconservative and externally applied. Combining (2.32) with the constraint condition given in Eqn. (2.31) yields a system of differential algebraic equations. The algebraic constraint equations are converted into differential equations by differentiating Eqn. (2.31) twice to yield a Gaussian form as follows

$$\Phi_q \ddot{q} = -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{tq}\dot{q} - \Phi_{tt} \tag{2.33}$$

Now combining (2.32) and (2.33) yields the following set of differential algebraic equations which can be solved using standard numerical methods.

$$\begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ -\lambda \end{bmatrix} = \begin{bmatrix} Q^{nc} \\ -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{tq}\dot{q} - \Phi_{tt} \end{bmatrix} \tag{2.34}$$

Although a matrix inversion is needed in the process to solve the above set of equations, the matrix that needs to be inverted is sparsely populated and the expressions are fairly small. So, there is no danger of numerical problems related to matrix inversion.

2.3.1 Development of the Constraint Equations

This section will briefly discuss how to develop the constraint equations and derive the forces needed for the multibody constrained formulation. The software developed during the course of this research requires that only the constraint matrix Φ be developed manually. The additional information that a user must supply to the program includes problem dimension, controller dimension, number of design variables, and some fixed problem specific data.

2.3.1.1 Example of Developing the Constraint Matrix Φ

In order to assemble a mechanism and give it the appropriate number of degrees of freedom, a set of constraint equations are necessary. This section will illustrate with a simple example how to develop constraint equations for two planar bodies connected by a revolute joint, and how to specify forces on a body. Joints are expressed in terms of algebraic equations of the form

$$\Phi(q, t) = 0$$

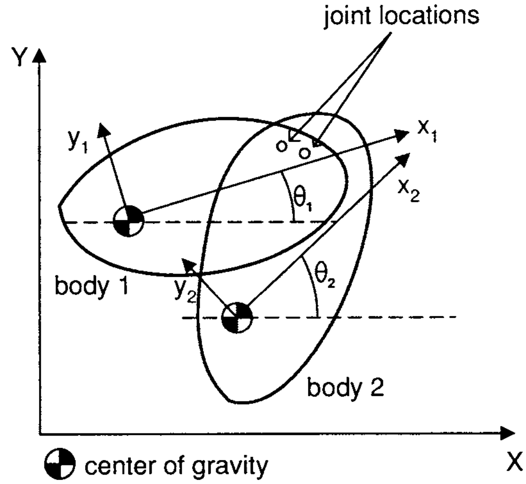

Figure 2.1 Coordinate frames for developing constraint equations for a revolute joint between two bodies

For example, the revolute joint that connects bodies 1 and 2 in Fig. 2.1 can be expressed by the following equations,

$$\begin{aligned} &\text{X:} \quad (X_i + x_i cos(\theta_i) - y_i sin(\theta_i)) - (X_j + x_j cos(\theta_i) - y_j sin(\theta_i)) = 0 \\ &\text{Y:} \quad (Y_i + x_i sin(\theta_i) + y_i cos(\theta_i)) - (Y_j + x_j sin(\theta_i) + y_j cos(\theta_i)) = 0 \end{aligned} \tag{2.35}$$

These constraints ensure that the distance between the joint locations on two bodies is exactly zero. This is basically a loop closure equation. Similar constraints can be developed for other types of joints.

2.3.1.2 The Generalized Force Q

One tricky aspect of the constrained formulation is assigning forces and torques. Figure 2.2 illustrates a force and a torque acting on a body. Forces and torques are also expressed in terms of algebraic equations of the form

$$Q(x_i, y_i, \theta_i) = f$$

So for a torque,

$$Q(\theta) = T \tag{2.36}$$

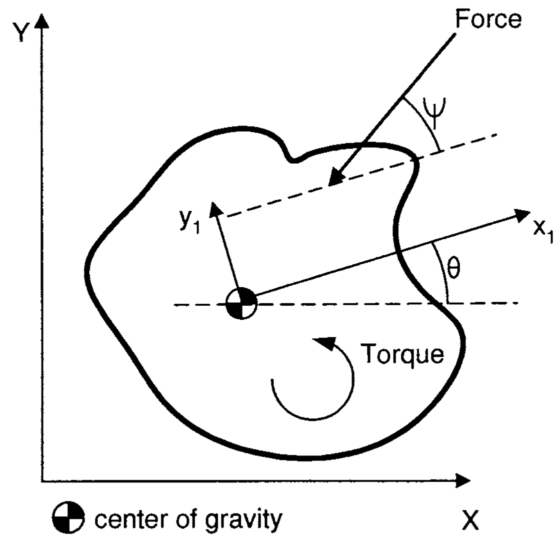

Figure 2.2 Coordinate frame for developing the generalized force vector Q on a body

Since a torque is a free vector, it can be moved anywhere on the body. That is to say, it can be considered to be acting at the centroid of the body. The force, however, requires careful consideration. As an example, the force in Fig. (2.2) can be expressed by the following set of equations,

$$\begin{aligned} Q_F(x) &= Fcos(\Psi + \theta) \\ Q_F(y) &= Fsin(\Psi + \theta) \\ Q_F(\theta) &= -(Fcos(\Psi))Y + (Fsin(\Psi))X \end{aligned} \tag{2.37}$$

where, $Q_F(x)$, $Q_F(y)$ and $Q_F(\theta)$ represent the X and Y components of F and torque developed by the force F, respectively. Note that the joint forces cannot be in the force vector Q because they do not do any work. However, it is possible to compute the joint forces using the Lagrange multipliers. A more detailed reference for developing constraint functions can be found in [36].

2.4 Mathematical Modelling of Controller Dynamics

For the integrated design problem to be discussed in Chapter 4, we will be considering a multibody system in closed-loop configuration. That is, the dynamic model of the overall system will involve multibody system dynamics along with associated automatic controller dynamics. The controller dynamics can be easily incorporated in the system equations (Eqn. (2.29) or (2.34)) through generalized force vector Q. The controller input can be considered as an externally applied force. Depending on whether the controller is dynamic or static, additional controller state equations will be needed to augment the system of equations to obtain closed-loop system dynamics.

Given below is the procedure to include controller dynamics into the system equations.

Incorporation of controller dynamics into the system equations allows us to treat controller parameters as a part of the design variables along with the structural design variables. This unified framework is extremely powerful as it truly integrates the design of the entire closed-loop system and not just the controller or the mechanism.

2.5 Generic Form of Dynamic Controller

Assume a generic form of the dynamic controller expressed in state-space form as

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c y_c \\ u &= C_c x_c + D_c y_c \end{aligned} \tag{2.38}$$

where, x_c is the $n_c \times 1$ controller state vector, y_c is the controller input vector, u is the controller output vector, and A_c, B_c, C_c and D_c are the controller matrices consisting of controller parameters. All of these controller matrices could be treated as the controller design variables. The order of the controller (n_c) can be decided by the designer apriori and changed iteratively during optimization process if desired.

2.6 Controller Dynamics in Lagrange Formulation

Recall Eqn. (2.29) from Chapter 2 which represents the equations of motion of the multibody system in the Lagrange formulation.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i^{nc} + \Phi_q^T \lambda, \quad (i = 1, 2, ..., n) \tag{2.39}$$

When there are externally applied forces, like control input, the above equation can be modified to include these forces as the additional forcing terms on the right hand side and the system equations can be augmented to include the controller dynamics. That is,

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} &= Q_i^{nc} + \Phi_q^T \lambda + u, \quad (i = 1, 2, ..., n) \\ \dot{x}_c &= A_c x_c + B_c y_c \end{aligned} \tag{2.40}$$

where, $Q_i = Q_i^{nc} + u$ and $u = C_c x_c + D_c y_c$. y_c is typically the sensed output of the plant and can be expressed as a function of q, $\dot{q}$, and $\ddot{q}$.

The set of equations given in Eqn. (2.40) can now be numerically integrated using methods like fourth order Runga-Kutta method to obtain a closed-loop response.

2.7 Controller Dynamics in Multibody Constrained Formulation

Controller dynamics can be incorporated in multibody constraint formulation in a similar way as in the case of Lagrangian formulation. The closed-loop system dynamics in this case can be given by

$$\begin{aligned} \begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ -\lambda \end{bmatrix} &= \begin{bmatrix} Q^{nc} + u \\ -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{tq}\dot{q} - \Phi_{tt} \end{bmatrix} \\ \dot{x}_c &= A_c x_c + B_c y_c \end{aligned} \tag{2.41}$$

where, $u = C_c x_c + D_c y_c$. The state vector of the closed-loop system is given by

$$[q \;\; \dot{q} \;\; x_c]^T$$

If the controller is static (constant-gain) the only controller design variable is the D_c matrix, and all other controller matrices are zero. When the controller is dynamic all controller matrices A_c, B_c, C_c, and D_c are design variables. One can reduce the number of design variables by providing a pre-defined structure to the controller matrices. For example, the A_c matrix can be assumed to be in one of the canonical forms where only the elements in the bottom row of the matrix are the design variables. That is, the number of design variables in A_c reduces from n_c^2 to n_c. Similarly, B_c and C_c can be assumed to be in the canonical form as well. Another simplification can be made is to assume that the controller is strictly proper which eliminates the D_c matrix. The controller can be constrained to be stable by constraining A_c to have only negative eigenvalues.

The multibody constraint formulation also allows the change of the controller structure rather easily during the design process. Due to the modular structure of this formulation, any change in the system configuration does not affect the system equation significantly to warrant reformulation as in the case of Lagrangian approach. The changes have to be made only in a handful of elements in system matrices. This is very beneficial while using the symbolic inputs.

2.7.1 Important Considerations in Multibody Constrained Formulation

There are a few important issues that need to be mentioned about the multibody constrained formulation and they are addressed here.

2.7.1.1 Equation Swell

The equation swell phenomena was addressed briefly at the beginning of this section. Equation swell is the rapid growth and increased complexity of a symbolically generated equation. As mentioned previously, the Lagrange formulation produces relatively complex and messy equations as the number of bodies involved increases beyond three or four. On the contrary, in the multibody constrained formulation the individual expressions (expressions for individual bodies) are relatively simple and easily manageable. Matrix dimension can grow quickly, however, the matrices used in the multibody constrained formulation are very sparsely populated and can be easily manipulated. So equation swell is not an issue with the multibody constrained formulation. In fact, this is probably the only known method that can be used with symbolic computations in multibody analysis.

2.7.1.2 Joint Drift

One area of concern with the multibody constrained formulation is the problem of possible joint drift. The joint drift phenomena occurs when the constraint equation $\Phi(q,t) = 0$ is not exactly satisfied due to computational errors. That is to say, when such errors occur, the constraint equation becomes,

$$\Phi(q,t) = 0 + \delta, \quad \text{where } \delta \text{ is a small deviation}$$

When joint drift occurs, links begin to separate as the simulation progresses and the system response is inaccurate. Correcting this problem is not straight forward because it is not clear as to where exactly the joints need to be re-placed.

There are a few ways to address this issue of joint drift:

1. Neglect the drift, but verify it is there. Use integration error control to limit Φ

2. Use a single step integrator (fixed time step integrator), and after each step reassemble the mechanism just like computing initial conditions. Note that when initial conditions are calculated, n independent coordinates are chosen, where n is equal to the number of degrees of freedom of the system

3. Constraint stabilization [29]: The basic idea of this method is to treat each joint as a controller. In this case, the constraint dynamics become,

$$\Phi_q \ddot{q} = -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{tq}\dot{q} - \Phi_{tt} - \alpha\Phi + 2\beta\dot{\Phi}$$

   where the term $-\alpha\Phi+2\beta\dot{\Phi}$ is a correction factor which behaves like a PD-controller when the joints drift. Both α and β must be tuned to minimize the drift properly.

4. Use a variable step integrator, but check the independent coordinates at each time step. Restart the simulation if the set of constraints changes. For this method n independent coordinates are also needed.

5. Hand code a function to handle the joint drift. This is a problem specific fix, the problem of joint drift must be studied for the problem at hand and addressed for the specific problem.

Most of the time, joint drift is minimal, however, it is not to be overlooked. Two of the methods given above require that independent coordinates be selected. There are a few ways that independent coordinates can be selected:

1. Select independent coordinates based on gaussian elimination of Φ with full pivoting [7].

2. Select independent coordinates based on

$$\Delta q \approx \dot{q}\Delta t + \frac{1}{2}\ddot{q}\Delta t^2$$

the independent coordinate is the largest Δq. This method is most commonly used for one degree of freedom systems. It can be used for multi degree of freedom systems, however, after choosing the first independent coordinate, all further dependencies must be eliminated.

3. The final method is just a hand coded selection of the independent coordinate.

To check if joint drift is occurring, simply evaluate $\Phi(q,t)$ at each time step. $\Phi(q,t)$ should be zero at each time step, but it is not for the reasons described above. Usually joint drift is not a problem in simple systems.

3 SENSITIVITY

This chapter is focused on the development of sensitivity equations which is the most important step in the integrated design framework presented in this thesis. Sensitivity equations are used in multiple problem formulations. For example, performance optimization, sensitivity optimization, robustness analysis and sensitivity analysis. This chapter will present a basic background of sensitivity followed by a generic paradigm to develop sensitivity equations for the optimization problem formulation.

3.1 Preliminaries

This section will present a preliminary background of sensitivity and an example will be given at the end to illustrate the basic concepts.

3.1.1 Sensitivity

The definition of sensitivity will be given in the context of a function and a variable. The sensitivity of a function with respect to a particular variable can be defined as the percentage change in the function value due to a change in that variable. In this context, the function that is of interest is the performance function to be optimized and the variable is the design variable of interest. Consider any performance function f as follows,

$$f = f(\vec{q}, \vec{\dot{q}}, \vec{\ddot{q}}, \vec{\lambda}, \vec{b}, \vec{t}) \tag{3.1}$$

where, $\vec{b}$ is a vector of design variables, $\vec{q}$ is the vector of generalized coordinates, and $\vec{\lambda}$ is the vector of Lagrange multipliers. The sensitivity, or change of f with respect to a design variable, b, can be written as

$$S_{\vec{b}}^{f} \triangleq \left[\frac{\partial f}{\partial \vec{b}}\right]_{1\times n_b} \tag{3.2}$$

where it is assumed that $f(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}, \vec{\lambda}, \vec{b}, \vec{t})$ is a scalar and $\vec{b}$ has dimension $n_b \times 1$. If f is a vector of performance functions (now $\vec{f}$), then $S_{\vec{b}}^{\vec{f}}$ becomes a matrix,

$$S_{\vec{b}}^{\vec{f}} = \left[\frac{\partial \vec{f}}{\partial \vec{b}}\right]_{n_f\times n_b} \tag{3.3}$$

where, n_f is the length of the vector $\vec{f}$. To illustrate these functions, consider the following example.

Let the performance function for some system be given by,

$$f = b\cos(\theta)\dot{\theta}$$

Then, the change in f due to a change in design variable, b, or in other words, the sensitivity of f with respect to b, is given by,

$$S_{\vec{b}}^{\vec{f}} = \frac{\partial f}{\partial b} = \cos(\theta)\dot{\theta} + -b\sin(\theta)\dot{\theta}\frac{\partial \theta}{\partial b} + b\cos(\theta)\frac{\partial \dot{\theta}}{\partial b}$$

If the state information is known, the only unknowns are the $\frac{\partial(\cdot)}{\partial b}$ terms. More insight can be gained into the meaning of sensitivity function by considering Fig. 3.1. The figure shows magnified plots of the function f for a small perturbation of the design variable b. The change in the value of f due to this perturbation in b at any time $t = t^*$ should equal the sensitivity of f with respect to b multiplied by the perturbation in b. That is,

$$f(b_0 + \Delta b)|_{t=t^*} - f(b_0)|_{t=t^*} \approx \frac{\partial f}{\partial b}\Delta b \tag{3.4}$$

Equation (3.4) is very useful particularly for error checking in sensitivity formulations which will be addressed later in this chapter.

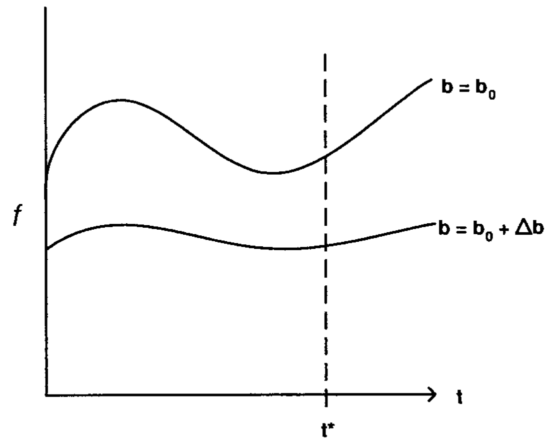

Figure 3.1 Function f evaluated at b_0 and $b_0 + \Delta b$

3.1.2 Formulation of Sensitivity Equations

For the system given in Eqn. (3.1), the general form of the sensitivity function is given by:

$$\frac{\partial f}{\partial \vec{b}} = \left.\frac{\partial f}{\partial \vec{b}}\right|_{exp} + \left.\frac{\partial f}{\partial \vec{q}}\right|_{exp}\frac{\partial \vec{q}}{\partial \vec{b}} + \left.\frac{\partial f}{\partial \vec{\dot{q}}}\right|_{exp}\frac{\partial \vec{\dot{q}}}{\partial \vec{b}} + \left.\frac{\partial f}{\partial \vec{\ddot{q}}}\right|_{exp}\frac{\partial \vec{\ddot{q}}}{\partial \vec{b}} + \left.\frac{\partial f}{\partial \vec{\lambda}}\right|_{exp}\frac{\partial \vec{\lambda}}{\partial \vec{b}} + \left.\frac{\partial f}{\partial \vec{t}}\right|_{exp}\frac{\partial \vec{t}}{\partial \vec{b}} \tag{3.5}$$

where, exp denotes that the derivative is taken explicitly with respect to the independent variable involved. Also, $\frac{\partial t}{\partial \vec{b}} = 0$ as time does not get affected by b.

The formulation of differential equations for sensitivity and corresponding initial conditions will be discussed next.

The idea is to obtain a governing differential equation for the sensitivity function. To this effect, an assumption is made that the derivative terms can be written with respect to b as follows,

$$\begin{aligned} \frac{d\dot{q}}{db} &= \frac{d}{db}\frac{dq}{dt} = \frac{d}{dt}\frac{dq}{db} = \frac{d}{dt}q_b = \dot{q}_b \\ \frac{d\ddot{q}}{db} &= \frac{d}{db}\frac{d^2q}{dt^2} = \frac{d^2}{dt^2}\frac{dq}{db} = \frac{d^2}{dt^2}q_b = \ddot{q}_b \end{aligned} \tag{3.6}$$

Now, using Eqn. (3.6), Eqn. (3.5) can be rewritten as

$$f_b = f_b^{exp} + f_q^{exp}q_b + f_{\dot{q}}^{exp}\dot{q}_b + f_{\ddot{q}}^{exp}\ddot{q}_b + f_\lambda^{exp}\lambda_b \tag{3.7}$$

where, $f^{exp}_{(\cdot)} = \frac{\partial f}{\partial(\cdot)}\Big|_{exp}$. Equation (3.7) is essentially a differential equation for the new variable q_b, which will be called the state sensitivity. Now Eqn. (3.7) can be treated as just another state equation and can be solved together with a set of system equations (Eqn. (2.29)). Thus the closed-loop system of Eqn. (2.29) and the whole system can be solved together. The complete state vector for the system then consists of plant states, controller states and sensitivity states. Next, a simple example is given to illustrate the formulation of sensitivity differential equations.

Consider the following second order nonlinear system:

$$\ddot{\theta} = f(\theta, \dot{\theta}, b) = b\sin(\theta) + b^2\dot{\theta}$$

where b is a design variable. Define state variables as,

$$[x_1 \quad x_2]^T = [\theta \quad \dot{\theta}]^T$$

then, the system equations can be rewritten as a set of two first order equations,

$$[\dot{x_1} \quad \dot{x_2}]^T = [x_2 \quad b\sin(x_1) + b^2 x_2]^T \tag{3.8}$$

These equations can be solved with a one-step integration routine. Once the state solution is known, the state sensitivities can be obtained as follows. Note that

$$\frac{\partial\ddot{\theta}}{\partial b} = \frac{\partial f}{\partial b} + \frac{\partial f}{\partial \theta}\frac{\partial \theta}{\partial b} + \frac{\partial f}{\partial \dot{\theta}}\frac{\partial \dot{\theta}}{\partial b}$$

This can be rewritten in terms of state sensitivities as,

$$\ddot{\theta}_b = f_b + f_\theta \theta_b + f_{\dot{\theta}}\dot{\theta}_b$$

Once the state solutions θ and $\dot{\theta}$ are available, the only unknowns in the above sensitivity differential equation are θ_b and $\dot{\theta}_b$, which can be solved for in the same manner as the state differential equation . For the example under consideration, the sensitivity state can be defined as,

$$[x_3 \quad x_4]^T = [\theta_b \quad \dot{\theta}_b]^T$$

and rewrite the sensitivity differential equations in the following first order form,

$$[\dot{x_3} \quad \dot{x_4}]^T = \left[x_4 \quad [[\sin(x_1) + 2bx_2] + [b\cos(x_1)x_3] + [b^2 x_4]]\right]^T \tag{3.9}$$

Now, the system described by Eqn. (3.8) can be augmented by Eqn. (3.9) to obtain a combined system described by a set of four first order differential equations.

$$[\dot{x_1} \quad \dot{x_2} \quad \dot{x_3} \quad \dot{x_4}]^T = \left[x_2 \quad [b\sin(x_1) + b^2 x_2] \quad x_4 \quad [[\sin(x_1) + 2bx_2] + [b\cos(x_1)x_3] + [b^2 x_4]]\right]^T \tag{3.10}$$

where,

$$[x_1 \quad x_2 \quad x_3 \quad x_4 \quad]^T = [\theta \quad \dot{\theta} \quad \theta_b \quad \dot{\theta}_b]^T$$

It is to be noted that while solving the combined system of state and sensitivity equations the sensitivity equations must be located at the bottom of the stack of equations because the state solution is needed to solve for sensitivities.

In general, for a system with n states and m design variables, solving for the combined system will yield $n + n \times m$ total states. This can be seen from the previous example.

The fact that both sensitivities and system response can be obtained simultaneously is very significant because it allows for simultaneous error control on both state and sensitivity. The sensitivity terms can be used directly in the design for robustness, design for variability and design for manufacturability, which will be discussed in Chapter 4.

Another significant benefit from this formulation is that the designer obtains valuable physical insight to the problem using the sensitivity data and can make good judgements in the design process.

3.1.3 Derivation of Initial Conditions for Sensitivity

When solving sensitivity differential equations, initial conditions are needed on the state sensitivities. The method for finding the initial conditions for the state sensitivities can be best described by an example. Consider a system described by two states, position and velocity, the initial conditions are,

$$\begin{aligned} x(0): &\quad b\sin(\theta_0) = constant \\ v(0): &\quad \dot{\theta}_0 = 0 \end{aligned}$$

the initial conditions for the associated sensitivity terms are found by taking the partial derivative of $x(0)$ and $v(0)$ with respect to the design variable. The initial condition of the sensitivity is nothing but the change in the state initial condition with respect to the change in design variable b.

$$\frac{\partial x(0)}{\partial b} = x_b(0) = \sin(\theta_0) + \cos(\theta_0)\frac{\partial \theta_0}{\partial b} = 0$$

which can be rearranged as

$$\frac{\partial \theta_0}{\partial b} = \frac{-\sin(\theta_0)}{\cos(\theta_0)} = -\tan(\theta_0).$$

The equation above gives the initial condition for the sensitivity term x_b. The initial condition for the sensitivity $v_b$is obtained similarly. Taking the partial derivative of $v(0)$ initial condition with respect to the design variable b, yields,

$$\frac{\partial v(0)}{\partial b} = v_b(0) = \frac{\partial \dot{\theta}_0}{\partial b} = 0$$

Which gives four initial conditions for four states. As previously stated the total number of states is $n+n\times m = 2+2\times 1 = 4$ which includes two system states and two sensitivity states.

3.1.4 Error Checking

The equations in the sensitivity formulation can get fairly complicated. As a result, a method to do error checking for the equations is necessary. Recall from Section 3.1.1, Fig. 3.1 and Eqn. (3.4) that, for small changes in the design variable, the change in the function of the design variable should be approximately equal to the change in sensitivity of that function with respect to the design variable multiplied by the change in the design variable. Equation (3.4) can be rewritten as,

$$(\theta - \tilde{\theta}) = \text{avg}(\theta_b, \tilde{\theta}_b)\Delta b \tag{3.11}$$

where θ and $\tilde{\theta}$ denote the system state variables corresponding to the nominal and perturbed values of the design variable, respectively. Similarly, θ_b and $\tilde{\theta}_b$ are the state sensitivities with a nominal and perturbed design variable, respectively. Since the sensitivities also change with a perturbation in b, an average value is taken in Eqn. (3.11) and Δb is the amount of perturbation in the design variable. This method of error checking is known as the finite difference check. For finite differences in design variable (the amount of perturbation), the sensitivity state should change by a known amount. For typical systems, if the design variable is perturbed by 0.1%, the left and right side of Eqn. (3.11) should be within 1.0% of each other. Alternately, the difference equation can be written as

$$\%\text{error} = \frac{(\theta - \tilde{\theta}) - \text{avg}(S_b^{\theta}, S_{\tilde{b}}^{\theta})\Delta b}{(\theta - \tilde{\theta})} \tag{3.12}$$

Equation (3.12) can be verified numerically.

In some cases, an error plot will have spikes which can be easily mistaken for discontinuity in the sensitivity. However, this phenomenon typically occurs where there is a phase change between the two functions and most of the times it can be ignored. This phenomenon is illustrated in Fig. 3.2. Most of the time, the spikes can be ignored. Since the error is determined at each time instance, there are instances where the two curves change phase and the difference in response is very large in comparison to the sensitivity.

Another observation from Fig. 3.2 is that the error between the actual change in response and predicted change in response is very close to zero. This method of error checking is used in this work for all sensitivities.

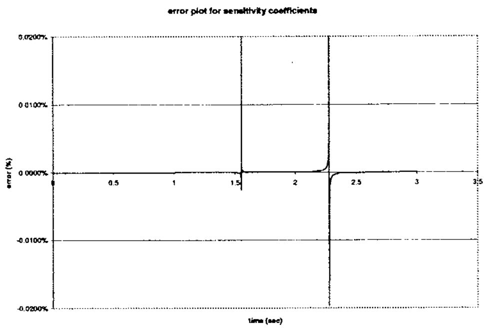

Figure 3.2 Error plot

3.2 Sensitivity Equations in Lagrangian Formulation

This section will discuss the inclusion of sensitivity equations in the Lagrangian formulation. Sensitivity derivations in Lagrange's formulation is fairly straightforward.

Assume that Lagrange's equations are given in a set of second order differential equations. The process of deriving the sensitivity equations, then, is simply a matter of taking the derivative of the differential equations with respect to the design variable. If $g = f(\theta, \dot{\theta}, b)$, and it represents a differential equation, then the sensitivity can be derived as,

$$\frac{\partial g}{\partial b} = \left.\frac{\partial g}{\partial b}\right|_{exp} + \left.\frac{\partial g}{\partial \theta}\right|_{exp} \frac{\partial \theta}{\partial b} + \left.\frac{\partial g}{\partial \dot{\theta}}\right|_{exp} \frac{\partial \dot{\theta}}{\partial b} \tag{3.13}$$

Equation (3.13) will yield necessary sensitivity differential equations.

3.3 Sensitivity Equations in the Multibody Constrained Formulation

Recall from Chapter 2 the closed-loop equations of motion obtained using the multibody constrained formulation in Eqn. (2.41) repeated here,

$$\begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ -\lambda \end{bmatrix} = \begin{bmatrix} Q^{nc} + u \\ -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{tq}\dot{q} - \Phi_{tt} \end{bmatrix}$$

$$\dot{x}_c = A_c x_c + B_c y_c \tag{3.14}$$

where, $u = C_c x_c + D_c y_c$. The sensitivity of the constraint equation can be determined as,

$$\frac{d\Phi}{db} = \Phi_q q_b + \Phi_b = 0 \quad \Rightarrow \quad \Phi_q q_b = -\Phi_b \tag{3.15}$$

Now, differentiating Eqn. (3.15) twice with respect to time yields the Gaussian form as follows,

$$\Phi_q \ddot{q}_b = -\ddot{\Phi}_q q_b - 2\dot{\Phi}_q q_b - \ddot{\Phi}_b \tag{3.16}$$

Similarly, the system equations and the controller equations can be differentiated with respect to the design variable b to yield,

$$M\ddot{q}_b - \Phi_q^T \lambda_b = Q_q q_b + Q_{\dot{q}} \dot{q}_b + Q_b + u_b (\Phi_q^T \lambda)_q q_b + (\Phi_q^T \lambda)_b - (M\ddot{q})_b \tag{3.17}$$

$$(\dot{x}_c)_b = A_c (x_c)_b|_{exp} + (A_c \tilde{x}_c)_b + B_c (y_c)_b|_{exp} + (B_c \tilde{y}_c)_b + B_c (y_c)_q q_b + B_c (y_c)_{\dot{q}} \dot{q}_b \tag{3.18}$$

where,

$$u_b = C_c (x_c)_b|_{exp} + (C_c \tilde{x}_c)_b + D_c (y_c)_b|_{exp} + (D_c \tilde{y}_c)_b + D_c (y_c)_q q_b + D_c (y_c)_{\dot{q}} \dot{q}_b \tag{3.19}$$

Combining Eqns. (3.15), (3.17), (3.19) and (3.18) yields the following system of differential algebraic equations

$$u_b = C_c(x_c)_b|_{exp} + (C_c\tilde{x}_c)_b + D_c(y_c)_b|_{exp} + (D_c\tilde{y}_c)_b + D_c(y_c)_q q_b + D_c(y_c)_{\dot{q}}\dot{q}_b$$

$$\begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q}_b \\ -\lambda_b \end{bmatrix} = \begin{bmatrix} Q_q q_b + Q_{\dot{q}}\dot{q}_b + Q_b + (\Phi_q^T\lambda)_q q_b + (\Phi_q^T\lambda)_b - (M\ddot{q})_b + u_b \\ -\ddot{\Phi}_q q_b - 2\dot{\Phi}_q q_b - \ddot{\Phi}_b \end{bmatrix} \tag{3.20}$$

Equations (2.34), (3.20) and (3.18) can be solved simultaneously as discussed previously.

4 INTEGRATED DESIGN

This chapter will present the integrated design approach for controlled multibody dynamic systems with a particular focus on multi-link mechanical systems. The methodology developed has emerged as a valuable design and analysis tool for various optimization problems. The symbolic input capability and ability to handle constrained nonlinear systems is of significant benefit to the design field. The methodology is based on a modular approach which lends itself for potential parallelization using multiple processing units. Automated symbolic code generation is also an attractive feature in sensitivity and robustness analysis. This chapter is organized as follows: Section 4.1 will present a generic formulation to the optimization problem for integrated design. Subsequent sections will focus on different types of design objectives and present the formulation of the optimization problem for the respective objectives.

4.1 Optimization

The integrated design problem is formulated as an optimization problem where a performance function of interest is minimized with respect to a set of design variables chosen from more than one design domain. For the purpose of this thesis, the two domains of interest are structure and control design domains. That is, the set of design variables consists of structural design variables and control design variables.

In general, optimization can be thought of as a journey and not a destination. The journey is that of discovering the design space the designer has to work with. This philosophy becomes increasingly applicable when the optimization problem is non-convex and/or non-smooth. When the problem is non-convex, one can potentially get stuck with local minima and a suboptimal solution is the result. When the problem is non-convex, solving the optimization problem tends to be an art as a one-shot solution is typically not possible. In such cases, several intelligent solution strategies have to be employed to obtain a desirable optimal solution. Some of these techniques include changing initial conditions and constraint functions, fixing selected design variables, and even possibly changing the performance function or design variables.

Figure 4.1 shows the basic program structure used for the integrated design framework. The specific optimization problems may slightly deviate, but the core program remains the same.

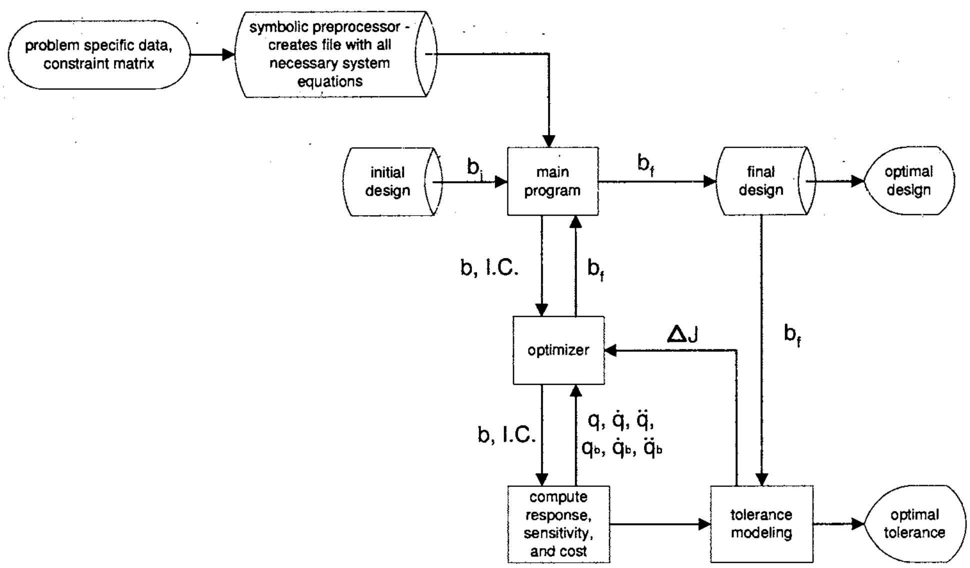

Figure 4.1 Integrated design scheme

A data flow diagram which shows how information flows between different computational modules/programs is shown in Fig. 4.2.

The optimizer used for this research is the *fmincon*() function from the optimization toolbox within Matlab©. The optimization tool box of Matlab© was found to be adequate for the problem addressed in this research. Also, in this work, the actual

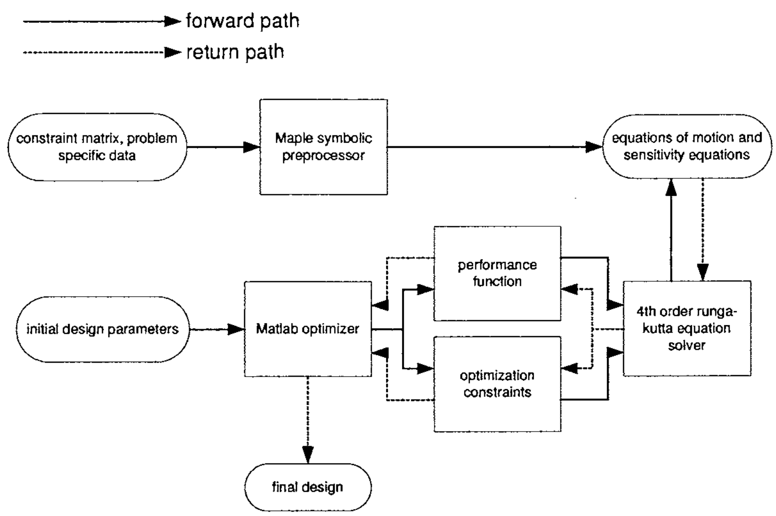

Figure 4.2 Data flow between modules

process of exploring the design space through optimization is the primary focus of the optimization in this research. The framework represented in Figs. 4.1 and 4.2 can be used to solve a variety of integrated design problems depending on the performance objective of interest. All of the optimization problems will be handled within a unified non-linear programming framework. This framework is presented below in the form of a generic optimization problem.

$$
\begin{aligned}
&\text{Determine the optimal value of the vector of design variables } \vec{b} \\
&\text{which solves the following problem:} \\
&\qquad \text{Minimize } f(\vec{q}, \vec{\dot{q}}, \vec{\ddot{q}}, \vec{\lambda}, \vec{q_b}, \vec{\dot{q}_b}, \vec{\ddot{q}_b}, \vec{\lambda_b}, \vec{b}, \vec{t}) \\
&\text{subject to} \\
&\qquad g_j(\vec{q}, \vec{\dot{q}}, \vec{\ddot{q}}, \vec{\lambda}, \vec{q_b}, \vec{\dot{q}_b}, \vec{\ddot{q}_b}, \vec{\lambda_b}, \vec{b}, \vec{t}) \leq 0 \\
&\qquad\qquad (j = 1, ..., N_g) \\
&\qquad h_k(\vec{q}, \vec{\dot{q}}, \vec{\ddot{q}}, \vec{\lambda}, \vec{q_b}, \vec{\dot{q}_b}, \vec{\ddot{q}_b}, \vec{\lambda_b}, \vec{b}, \vec{t}) = 0 \\
&\qquad\qquad (k = 1, ..., N_h)
\end{aligned}
\tag{4.1}
$$

where, N_g and N_h are the number of inequality and equality constraints, respectively. The functions f, g and h are selected depending on the design objective of interest. Given below are various categories of optimization objectives that are considered in this research for performing integrated design.

4.1.1 Optimizing for System Response

In the case of the controlled multibody systems, one of the most basic and very common optimization objectives is to optimize for the system performance which is typically expressed in terms of regulating or tracking performance. This problem usually involves minimization of integral error between a desired response and the actual response of the system. The difference between the actual and the desired response is penalized in the performance function. The optimization algorithm then searches over the domain of design variables to determine the optimal set such that the performance function is minimized, for example, the tracking error is minimized.

The performance function for this case is given by $f := J(q, \dot{q}, \ddot{q}, \lambda, b, t)$, where, J is a suitable measure of performance such as,

$$J = \int_{t_0}^{t_f} (E(t))^2 dt$$

where, $E(t)$ is the error between the desired and the actual system trajectories. The constraints ($\vec{g}$ and $\vec{h}$) on the design variables and system trajectories are determined from problem specific data.

4.1.2 The Min-Max Problem

In optimization problems where the focus is on the minimization of sensitivity or variability the problems can be formulated as the min-max problem. As the name suggests, every min-max problem is aimed at minimizing the maximum value of some desired function. While optimizing for performance, typically all of the performance criteria are placed in the performance function and the constraints are typically placed in terms of some function and/or bound on design variables. In the case of the min-max problem, however, the optimal performance is first determined by solving for performance and then this optimal performance is used as an inequality constraint; for example, the constraint could be that the performance must be within ±5.0% of the optimal performance. The performance function for the min-max problem is some desirable objective function or variable. The optimization minimizes this upper bound, while conforming to the constraints of the system. Mathematically, this can be expressed as,

$$\text{Minimize } \overline{f}$$
$$\text{subject to } \|J - J_{optimal}\| - .05 J_{optimal} \leq 0$$

where, $\overline{f}$ is some upper bound on a function or variable. Figure 4.3 gives a graphical representation of the min-max problem.

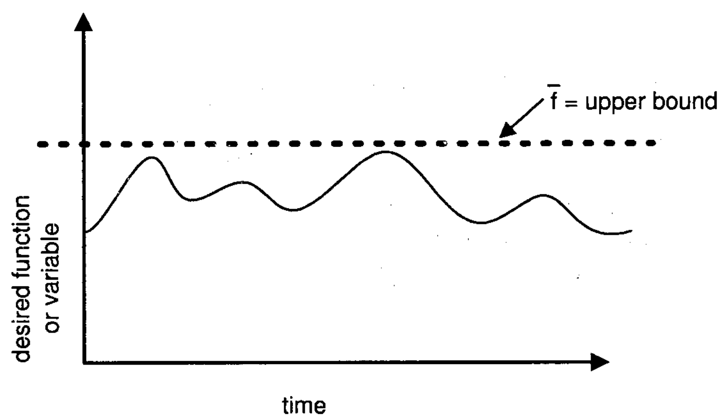

Figure 4.3 Performance objective for the min-max problem

4.1.3 Optimizing for Minimum Sensitivity

The problem of optimizing for robustness can be expressed in terms of minimizing the upper bound on a sensitivity function. The performance function, in such a case, is given by $f = \max\left(S(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}, \vec{\lambda}, \vec{q_b}, \dot{\vec{q_b}}, \ddot{\vec{q_b}}, \vec{\lambda_b}, \vec{b}, \vec{t})\right)$ where S is a suitable measure of sensitivity. When the sensitivity function is minimized, which represents the sensitivity of the system response to the design variable, the system response becomes robust to the change in design variables. Choosing which sensitivity function to be minimized is problem specific and is to be determined from the application of interest. The constraints on the design variables and system trajectories are determined from problem specific data.

4.1.4 Optimizing for Minimum Variability

Before introducing the minimum variability problem, it is important to understand the notion of variability.

4.1.4.1 Variability

Variability is the measure of variation in the performance function due to the variation in the design variables which is the result of manufacturing inaccuracies. Variability is closely related to the sensitivity, in fact, variability is computed from the sensitivity as follows.

$$V = \Delta f = f_b \Delta b$$

where, Δf is the variation in performance function f_b is the sensitivity of the performance function, and Δb is the variation in design variable due to manufacturing inaccuracies. Variability can be defined in two ways, the upper variability and the lower variability,

$$V^u \quad \Delta f(t)^u = \textstyle\sum_{i=1}^{n} \Delta f_i^u \tag{4.2}$$

$$V^l \quad \Delta f(t)^l = \textstyle\sum_{i=1}^{n} \Delta f_i^l \tag{4.3}$$

where,

$$\Delta f_i^u = \begin{cases} \frac{\partial f}{\partial b_i} \Delta b_i^u & \text{if } \frac{\partial f}{\partial b_i} \geq 0 \\ \frac{\partial f}{\partial b_i} \Delta b_i^l & \text{if } \frac{\partial f}{\partial b_i} < 0 \end{cases} \tag{4.4}$$

and

$$\Delta f_i^l = \begin{cases} \frac{\partial f}{\partial b_i} \Delta b_i^u & \text{if } \frac{\partial f}{\partial b_i} < 0 \\ \frac{\partial f}{\partial b_i} \Delta b_i^l & \text{if } \frac{\partial f}{\partial b_i} \geq 0 \end{cases} \tag{4.5}$$

The difference between sensitivity and variability is that variability is weighted sensitivity where an appropriate weighting is given to sensitivities of components that vary the most/least. Another way of looking at variability is,

$$b_i^{nom} + \Delta b_i^l \leq b_i^{nom} \leq b_i^{nom} + \Delta b_i^u \tag{4.6}$$

Variability and tolerance is very commonly confused. There is a subtle difference between the two, tolerance is chosen by the designer as an acceptable range of error for a given dimension, variation is the actual deviation from the nominal value for a given dimension that is a result of manufacturing inaccuracies . Yet another way to put it is, variation is what you get and tolerance is what you want.

The problem of optimizing for robustness can also expressed using the min-max problem approach where the performance function is the upper bound on variability. That is, $f = \max\left(V(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}, \vec{\lambda}, \vec{q_b}, \dot{\vec{q_b}}, \ddot{\vec{q_b}}, \vec{\lambda_b}, \vec{b}, \vec{\Delta b}, \vec{t})\right)$ where $\vec{\Delta b}$ is the amount of variation in the design variables which is known from the manufacturer. The constraints ($\vec{g}$ and $\vec{h}$) on the design variables and system trajectories are determined from problem specific data.

4.1.5 Optimizing for Manufacturability

The problem of optimizing for manufacturability can be expressed in terms of optimizing the tolerance band while retaining the system performance as specified. The design variables in this case are the upper and the lower limits of the design variables. The design variable values obtained from the sensitivity optimization are used as nominal values, and allowed to vary as long as the system performance is within ±0.2% of the sensitivity optimized performance. The performance function can be given by,

$$f = T(b^l, b^u) = -\left[\sum_{i=1}^{N}(\|b_i^u - b_i^l\|)\right]$$

where b_i^u and b_I^l denote the upper and lower tolerance values for design variable b_i, respectively, and N is the number of design variables. Once again, the constraints on the design variables and system trajectories are determined from problem specific data.

Optimizing for maximum tolerance is a more computationally intensive problem than the previously presented problems. Optimizing for maximum tolerance is directly related to a savings in product cost, and therefore very appealing to mass produced systems.

4.2 Symbolic Derivation of Response and Sensitivity Equations

One strength of the proposed design methodology is the ability to process symbolic data to compute sensitivity and response functions. Given the constraint matrix Φ, the external force vector Q, and some other problem specific data for any given system, the software developed in this research can symbolically generate expressions for closed-loop response and sensitivity. Using the symbolic computation software Maple©, symbolic expressions are generated using a direct differentiation method and then exported to Matlab© which is used for numerical data processing. Previous research has utilized symbolic methods for generating dynamic equations and sensitivity but are limited to simple systems and current, state-of-the-art software enables better software integration and automation.

The user must hand derive and input the following data: constraint matrix (Φ), generalized force vector (Q), number of structural and controller design variables, number of joints, number of bodies, body masses and inertias, controller order, number of controller inputs and outputs and the location of the controller inputs and outputs. With all of this problem specific information, the equations for response and sensitivity are generated, then solved numerically and used in the integrated design package. The multibody constrained formulation lends itself nicely to symbolic generation of equations. A balance of symbolic computing with numeric computing gives the designer a good handle on the problem. Specifically, the designer can determine exactly which equations relate to each body. Since it is very computationally expensive to solve the system in symbolic form, the system is solved numerically.

5 DOUBLE SLIDER EXAMPLE

This chapter will demonstrate the integrated design methodology presented in previous chapters on a proof-of-concept example system. The proof-of-concept example system used is a double slider system shown in Fig. 5.1. The design steps are illustrated in detail using hand derivations to help understand the methodology. In Chapter 6 the methodology will be extended to a linkage mechanism of a real-life generic construction loader.

5.1 Double Slider System Problem Statement

In order to demonstrate the proposed integrated design methodology, a simple double slider system of Fig. 5.1 was used. The system consists of a rod connected to two masses which are constrained to slide on perpendicular surfaces. One of the masses sliding on the horizontal surface is connected by a spring and a damper to the vertical surface. The controller is assumed to apply an external torque at the center of mass of the system. A PD-type controller is chosen to control the system and to track a desired position and velocity trajectory. Both, the Lagrangian as well as the multibody constrained formulation are used to model the system dynamics. The equations for the multibody constrained formulation were derived symbolically as well as manually for verification purposes.

The following sections will present a detailed description of all of the steps involved in the design of the double slider problem, using both the Lagrange and multibody

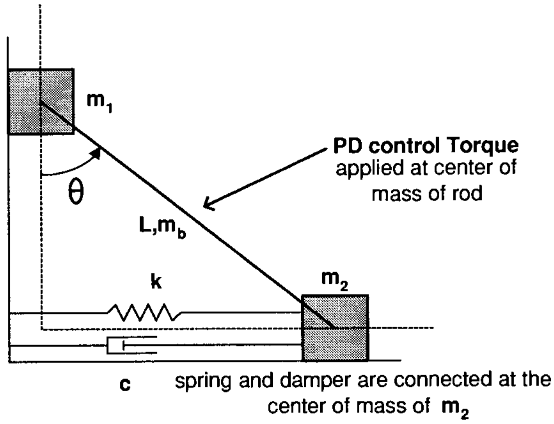

Figure 5.1 Double slider mechanism

constrained formulation. The integrated design is performed for various performance objectives. The results from each case are presented and discussed. The four optimization categories used include: tracking a desired trajectory, robustness in terms of sensitivity and variability, and manufacturability.

5.1.1 Lagrangian Formulation

For double slider mechanism the following generalized coordinates are used in the Lagrangian formulation.

$$q = [\theta \ \ \dot{\theta}]^T$$

The five design variables are the length of rod (L), spring constant(k), damping constant(c), and PD-controller gains (K_p and K_d). That is,

$$b = [L \ \ k \ \ c \ \ K_p \ \ K_d]^T$$

The system has a generalized input of,

$$Q = [-(K_p(\theta - \theta_{des}) + K_d(\dot{\theta} - \dot{\theta}_{des})) - c\dot{\theta}]$$

which includes the input from the controller and the damper. The damper is in the generalized force vector because it is a nonconservative force, unlike the spring, which is a conservative force.

5.1.1.1 Equations of Motion

The system dynamics and sensitivities were calculated using the Lagrangian formulation. Section 2.1 gave the generic derivation for the Lagrange equations. This section will go through the development of the equations specific to the double slider system based on the formulation in Section 2.1.

The kinetic energy of the system is

$$T = \frac{1}{2}m_1 v_1^2 + \frac{1}{2}m_2 v_2^2 + \frac{1}{2}m_b v_b^2 + \frac{1}{2}I_b\dot{\theta}^2$$

and the potential energy is

$$V = m_2 g L \cos\theta + m_b g \frac{L}{2}\cos\theta + \frac{1}{2}kL^2(\sin\theta - \sin\theta_0)^2$$

The Lagrangian of the system is then given by,

$$L = T - V \tag{5.1}$$

Differentiating Eqn. (5.1) to obtain the standard form (refer to Eqn. 2.19) as,

$$\begin{gathered} L^2 \sin\theta cos\theta\dot{\theta}^2(m_2 - m_1) - gL\sin\theta(m_2 + \tfrac{1}{2}m_b) - kL^2(\sin\theta - \sin\theta_0)\cos\theta \\ +L^2\ddot{\theta}(m_1\cos^2\theta + m_2 sin^2\theta + \tfrac{1}{3}m_b) = -(K_p(\theta - \theta_{des}) + K_d(\dot{\theta} - \dot{\theta_{des}}) + cL\cos\theta\dot{\theta}) \end{gathered}$$

The terms on the right hand side of the equation are the nonconservative forces. The above equation can be rewritten as,

$$\begin{gathered} \ddot{\theta} = \Big[-L^2\sin\theta cos\theta\dot{\theta}^2(m_2 - m_1) + gL\sin\theta(m_2 + \tfrac{1}{2}m_b) + \\ kL^2(\sin\theta - \sin\theta_0)\cos\theta - (K_p(\theta - \theta_{des}) + K_d(\dot{\theta} - \dot{\theta_{des}}) + cL\cos\theta\dot{\theta})\Big] \\ \Big[L^2\ddot{\theta}(m_1\cos^2\theta + m_2 sin^2\theta + \tfrac{1}{3}m_b)\Big]^{-1} \end{gathered} \tag{5.2}$$

Defining $[x_1 \quad x_2]^T = [\theta \quad \dot{\theta}]^T$, Eqn. (5.2) can be broken into two first-order differential equations as follows,

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \ddot{x}_2 &= \Big[-L^2\sin x_1 cos x_1 x_2^2(m_2 - m_1) + gL\sin x_1(m_2 + \tfrac{1}{2}m_b) + \\ &\quad kL^2(\sin x_1 - \sin x_{10})\cos x_1 - (K_p(x_1 - \theta_{des}) + K_d(x_2 - \dot{\theta}_{des}) + cx_2)\Big] \\ &\quad \Big[L^2\theta(m_1\cos^2 x_1 + m_2 sin^2 x_1 + \tfrac{1}{3}m_b)\Big]^{-1} \end{aligned} \tag{5.3}$$

This set of equations is solved numerically, using a fourth order Runga-Kutta method to yield the system states.

Next, the sensitivity equations are formulated. Since, there are five design variables, there are five different sensitivity equations all having the form

$$\frac{\partial g}{\partial b} = \left.\frac{\partial g}{\partial b}\right|_{exp} + \left.\frac{\partial g}{\partial \theta}\right|_{exp}\frac{\partial \theta}{\partial b} + \left.\frac{\partial g}{\partial \dot{\theta}}\right|_{exp}\frac{\partial \dot{\theta}}{\partial b}$$

In the Lagrange formulation, these equations tend to get very messy, which is why the multibody constrained formulation is advantageous.

The first term in each sensitivity equation, $\left.\frac{\partial g}{\partial b}\right|_{exp}$ will be different for each design variable. There are two terms that will be the same for every sensitivity equation, $\frac{\partial g}{\partial \theta}$ and $\frac{\partial g}{\partial \dot{\theta}}$. Finally, the remaining two terms, $\frac{\partial \theta}{\partial b}$ and $\frac{\partial \dot{\theta}}{\partial b}$ will be solved for numerically.

The derivative of $\ddot{\theta}$ with respect to design variable (length of rod (L), spring constant (k), damping constant (c), P-control coefficient (K_p), and D-control coefficient (K_d)) will be presented.

The explicit derivative of $\ddot{\theta}$ with respect to design variable L is,

$$\begin{aligned}
\tfrac{\partial \ddot{\theta}}{\partial L} = & \Big[((-2Lsin(\theta)cos(\theta)\dot{(\theta)}^2(m_2 - m_1) + gsin(\theta)(m_2 + \tfrac{1}{2}m_b) \\
& -2kLsin(\theta - \theta_0)cos(\theta) - (ccos(\theta)\dot{(\theta)}))(L^2(m_1cos^2(\theta) + m_2sin^2(\theta) + \tfrac{1}{3}m_b))) \\
& -(2L(m_1cos^2(\theta) + m_2sin^2(\theta) + \tfrac{1}{3}m_b)(-L^2sin(\theta)cos(\theta)\dot{\theta}^2(m_2 - m_1) \\
& +gLsin(\theta)(m_2 + \tfrac{1}{2}m_b) - kL^2sin(\theta - \theta_0)cos(\theta) \\
& -(K_p(\theta - \theta_{des}) + K_d(\dot{\theta} - \dot{\theta_{des}}) + cL\cos(\theta)\dot{\theta})))\Big] \\
& \Big[(L^2(m_1cos^2(\theta) + m_2sin^2(\theta) + \tfrac{1}{3}m_b)\Big]^{-2}
\end{aligned} \tag{5.4}$$

The explicit derivative of $\ddot{\theta}$ with respect to design variable k is,

$$\frac{\partial \ddot{\theta}}{\partial k} = \Big[L^2sin(\theta - \theta_0)cos(\theta)\Big]\Big[(L^2(m_1cos^2(\theta) + m_2sin^2(\theta) + \frac{1}{3}m_b)\Big]^{-1} \tag{5.5}$$

The explicit derivative of $\ddot{\theta}$ with respect to design variable c is,

$$\frac{\partial \ddot{\theta}}{\partial c} = \Big[-Lcos(\theta)\dot{\theta}\Big]\Big[(L^2(m_1cos^2(\theta) + m_2sin^2(\theta) + \frac{1}{3}m_b)\Big]^{-1} \tag{5.6}$$

The explicit derivative of $\ddot{\theta}$ with respect to design variable K_p is,

$$\frac{\partial \ddot{\theta}}{\partial K_p} = [-(\theta - \theta_{des})]\left[(L^2(m_1 cos^2(\theta) + m_2 sin^2(\theta) + \frac{1}{3}m_b)\right]^{-1} \tag{5.7}$$

Finally, the explicit derivative of $\ddot{\theta}$ with respect to design variable K_d is,

$$\frac{\partial \ddot{\theta}}{\partial K_d} = \left[-(\dot{\theta} - \dot{\theta_{des}})\right]\left[(L^2(m_1 cos^2(\theta) + m_2 sin^2(\theta) + \frac{1}{3}m_b)\right]^{-1} \tag{5.8}$$

Next, the derivative of $\ddot{\theta}$ with respect to the state variables θ and $\dot{\theta}$ will are given.

The derivative of $\ddot{\theta}$ with respect to θ is given as,

$$\begin{aligned}
\tfrac{\partial \ddot{\theta}}{\partial \theta} = \Big[(((-L^2\dot{\theta}^2(cos^2(\theta) - sin^2(\theta)))(m_2 - m_1)\\
+gLcos(\theta)(m2 + .5mb)\\
-kL^2((sin(\theta) - sin(\theta_0))cos^2(\theta) - (sin(\theta) - sin(\theta_0))sin(\theta))) - (K_p))\\
(L^2(m_1 cos^2(\theta) + m_2 sin^2(\theta) + \tfrac{1}{3}m_b)))\\
-((2L^2 sin(\theta)cos(\theta)(m_2 - m_1))\\
(-L^2 sin(\theta)cos(\theta)\dot{\theta}^2(m_2 - m_1)\\
+gLsin(\theta)(m_2 + \tfrac{1}{2}m_b)\\
-kL^2(sin(\theta) - sin(\theta_0))cos(\theta)\\
- -(K_p(\theta - \theta_{des}) + K_d(\dot{\theta} - \dot{\theta}_{des}) + \dot{c(\theta)})))\Big]\\
[(L^2(m_1 cos^2(\theta) + m_2 sin^2(\theta) + (1/3)m_b))]^{-2};
\end{aligned} \tag{5.9}$$

The derivative of $\ddot{\theta}$ with respect to $\dot{\theta}$ is,

$$\begin{aligned}
\tfrac{\partial \ddot{\theta}}{\partial \dot{\theta}} = \Big[(-2L^2 sin(\theta)cos(\theta)\dot{\theta}(m2 - m1) - ((cLcos(\theta)) + (K_d)))\Big]\\
\Big[L^2(m_1 cos^2(\theta) + m_2 sin^2(\theta) + \tfrac{1}{3}m_b)\Big]^{-1}
\end{aligned} \tag{5.10}$$

Each of Eqns. (5.4), (5.5),(5.6),(5.7) and (5.8) can be split into two first order differential equations by defining additional state variables as in Eqn. (5.3). The state vector for the double slider problem is,

$$\begin{aligned}
[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_10 \quad x_11 \quad x_12] =\\
\Big[\theta \quad \dot{\theta} \quad \tfrac{\partial \theta}{\partial L} \quad \tfrac{\partial \dot{\theta}}{\partial L} \quad \tfrac{\partial \theta}{\partial k} \quad \tfrac{\partial \dot{\theta}}{\partial k} \quad \tfrac{\partial \theta}{\partial c} \quad \tfrac{\partial \dot{\theta}}{\partial c} \quad \tfrac{\partial \theta}{\partial K_p} \quad \tfrac{\partial \dot{\theta}}{\partial K_p} \quad \tfrac{\partial \theta}{\partial K_d} \quad \tfrac{\partial \dot{\theta}}{\partial K_d}\Big]
\end{aligned} \tag{5.11}$$

and the corresponding initial conditions chosen are,

$$[30^o \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0] \tag{5.12}$$

Now the combined system of equations given by Eqns. (5.2), (5.4), (5.5),(5.6),(5.7), (5.8), (5.9) and (5.10) with initial conditions given by Eqn. (5.12), each second order differential equation can be split into two first order differential equations similar to Eqn. (5.3) and solved for numerically to obtain the state and sensitivity information

Once the sensitivities are obtained, variability can be calculated. The variability of a function f is of the form,

$$\frac{\partial f}{\partial b} = \left[\left.\frac{\partial f}{\partial b}\right|_{exp} + \left.\frac{\partial f}{\partial \theta}\right|_{exp} \frac{\partial \theta}{\partial b} + \left.\frac{\partial f}{\partial \dot{\theta}}\right|_{exp} \frac{\partial \dot{\theta}}{\partial b} \right] \Delta b$$

where Δb is the variation in design variable. For a more strict definition of variability, refer to 4.1.4.1.

For this example, the system will first be optimized for response only, next, minimum sensitivity only, then minimum variability, and finally optimized for manufacturability. Each of these optimization problems are discussed in detail in Chapter 4

5.1.2 Optimizing for Tracking Performance

The system was first optimized for performance. The performance criteria is for the mass on the horizontal surface, m_2 from Fig. 5.1, to follow a desired velocity profile. Therefore, the performance function to be minimized is,

$$f = \int_{t_0}^{t_f} (Lcos(\theta)\dot{\theta} - Lcos(\theta_{desired})\dot{\theta}_{desired})^2$$

where, $\theta_{desired}$ and $\dot{\theta}_{desired}$ are given by a fifth order polynomial function. The boundary conditions for the polynomial are at initial time and final time, velocity and acceleration are zero, the initial angle is 30^o and the final angle is 90^o. The design variables are subject to the constraints,

$$g = [.25 - L \quad L - 2.5 \quad -k \quad k - 500 \quad -c \quad c - 500$$
$$-K_p \quad K_p - 500 \quad -K_d \quad K_d - 500]^T$$
$$h = [\ ]$$

where g is the inequality constraints and h is the equality constraints.

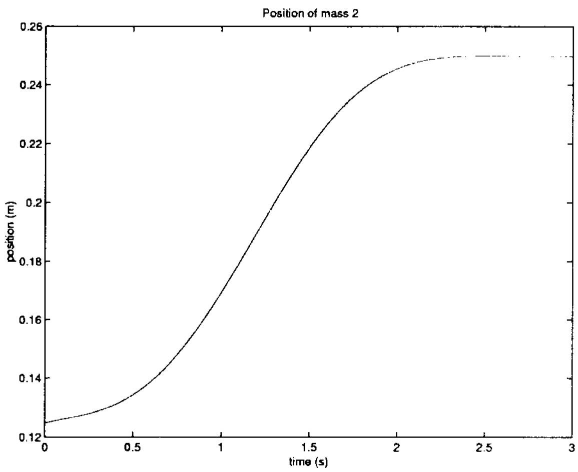

Figure 5.2 Position of m_2 for optimal tracking

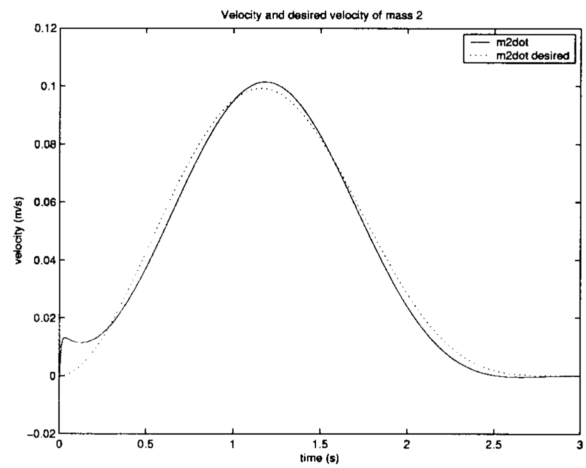

Figure 5.3 Velocity and desired velocity of m_2 for optimal tracking

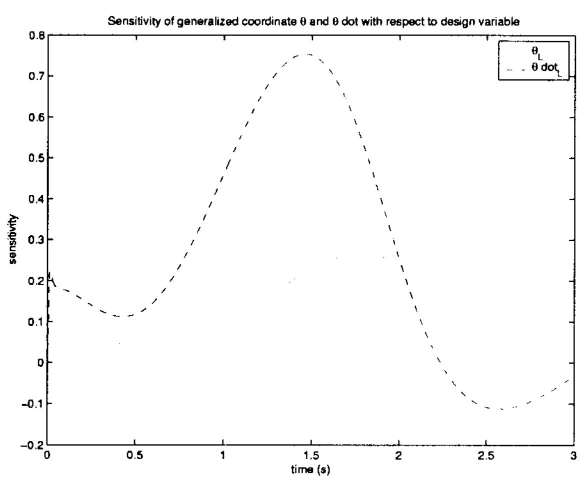

Figure 5.4 θ_L and $\dot{\theta}_L$ for optimal tracking

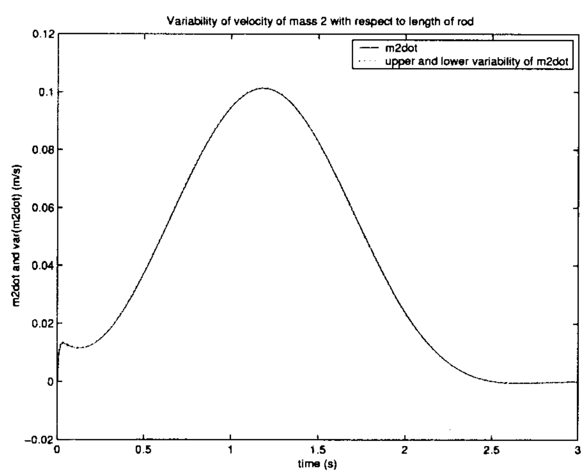

Figure 5.5 Variability of the velocity of m_2 for optimal tracking

The results are shown in Figs. 5.2, 5.3, 5.4, 5.5 and 5.6. Values for all design variables, cost, and tolerances will be presented in table 5.1.

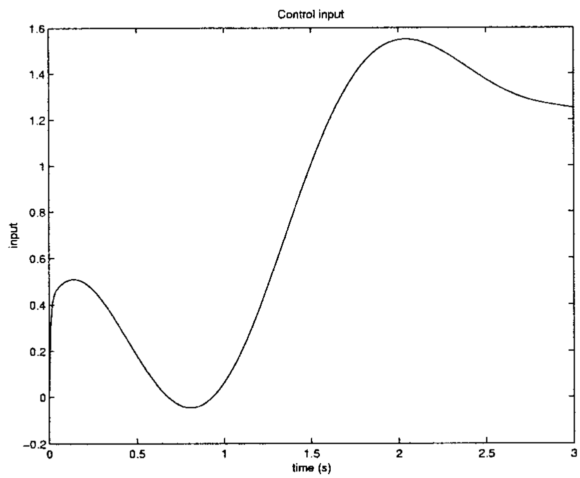

Figure 5.6 Control input for optimal tracking

5.1.3 Optimization for Variability

Next, the system was optimized for variability. The function to be bound is,

$$\begin{aligned} V = \Big[& 2\left[Lcos(\theta)\dot{\theta} - Lcos(\theta_d)\dot{\theta}_d\right]\left[cos(\theta)\dot{(\theta)} - cos(\theta_d)\dot{\theta}_d\right] \\ & -2\left[Lcos(\theta)\dot{\theta} - Lcos(\theta_d)\dot{\theta}_d\right]\left[Lsin(\theta)\dot{(\theta)}\theta_b\right] \\ & +2\left[Lcos(\theta)\dot{\theta} - Lcos(\theta_d)\dot{\theta}_d\right]\left[Lcos(\theta)\dot{(\theta)}_b\right]\Big]\Delta L \end{aligned}$$

and the performance function is to minimize the upper bound of V. The constraints of the system are given by,

$$\begin{aligned} g = [& .25 - L \quad L - 2.5 \quad -k \quad k - 500 \quad -c \quad c - 500 \quad -K_p \quad K_p - 500 \\ & -K_d \quad K_d - 500 \quad (J_{opt} - J) - J_{opt} * .05 \quad V_t - \bar{b}]^T \end{aligned}$$

$$h = [\ ]$$

where g is the inequality constraint, h is the equality constraint and V_t is the variability at each time step, and $\bar{b}$ is the upper bound on the maximum variability function value. J_{opt} is the optimal cost obtained from the optimization for system response problem.

Results from the variability optimization are presented in Figs. 5.7, 5.8, 5.9, 5.10 and 5.11.

It is not so obvious that from the variability optimized system, that the variability band decreased significantly. The maximum variability actually did decrease by 5%.

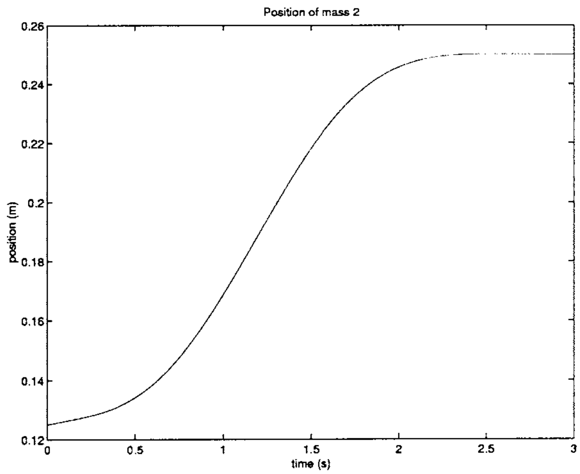

Figure 5.7 Position of m_2 for optimal variability

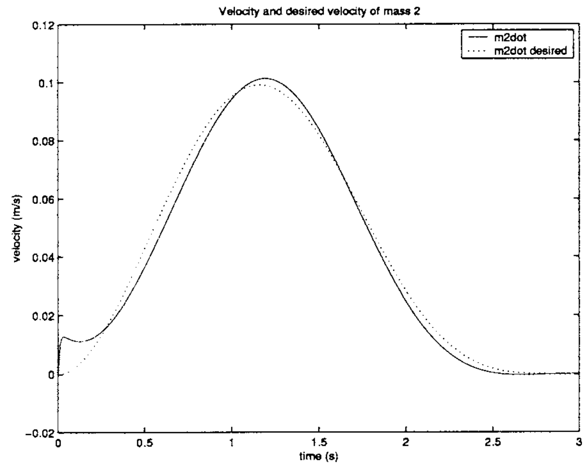

Figure 5.8 Velocity and desired velocity of m_2 for optimal variability

5.1.4 Optimization for Minimum Sensitivity

The only sensitivity function that is optimized in this case is the sensitivity of response due to a change in the length of the rod. The length of the rod is the most sensitive variable in the double slider mechanism. Therefore, the function to be minimized is,

$$S = max\left(\frac{\partial\theta}{\partial L}, \frac{\partial\dot{\theta}}{\partial L}\right)$$

and the constraints of the system are given by,

$$g = [.25 - L \quad L - 2.5 \quad -k \quad k - 500 \quad -c \quad c - 500 \quad -K_p \quad K_p - 500$$
$$-K_d \quad K_d - 500 \quad (J_{opt} - J) - J_{opt} * .05 \quad S_t - \bar{b}]^T$$
$$h = [\ ]$$

where g is the inequality constraint, h is the equality constraint, S_t is the sensitivity at each time step, and $\bar{b}$ is the upper bound on the maximum sensitivity function value. J_{opt} is the optimal cost obtained from the optimization for system response problem.

Results from minimum sensitivity optimization are presented in Figs. 5.12, 5.13, 5.14, 5.15 and 5.16. Notice that although the sensitivity drastically decreased, the controller

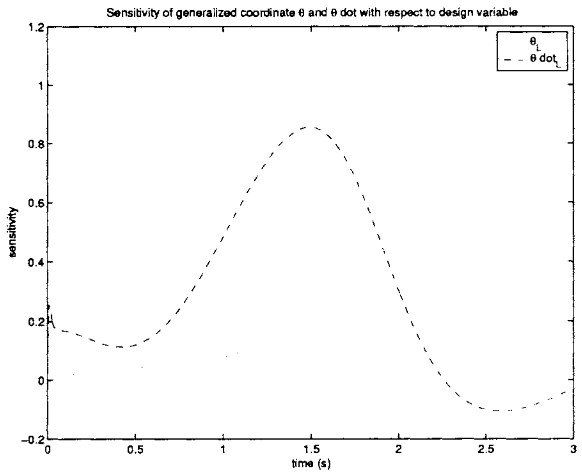

Figure 5.9 θ_L and $\dot{\theta}_L$ for optimal variability

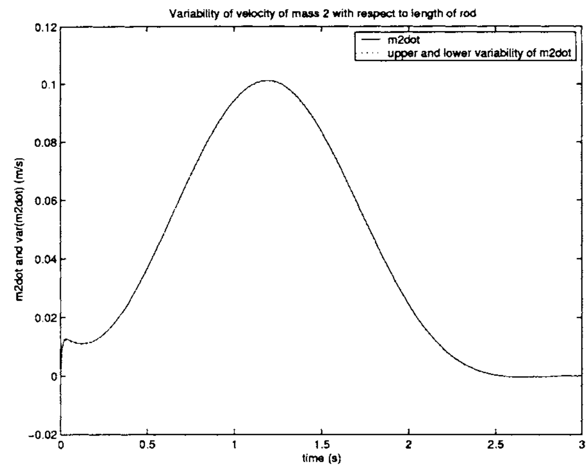

Figure 5.10 Variability of the velocity of m_2 for optimal variability

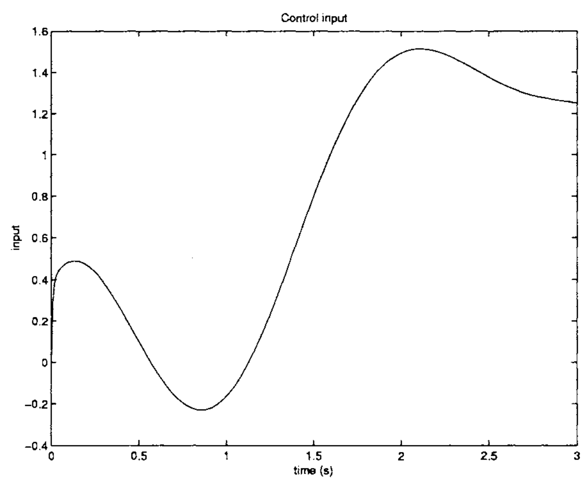

Figure 5.11 Control input for optimal variability

input increased quite a bit. This gives insight to the design space that is being explored. In order to decrease sensitivity, more control input is required.

5.1.5 Optimization for Manufacturability

The design variables from the minimum sensitivity optimization are used and the new design variables are the upper and lower bound on design variables. The function

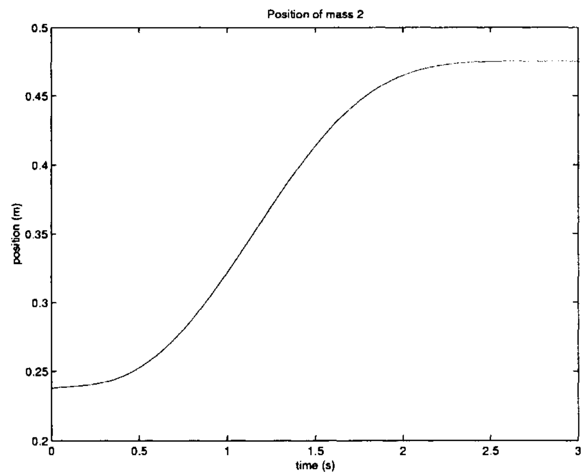

Figure 5.12 Position of m_2 for optimal sensitivity

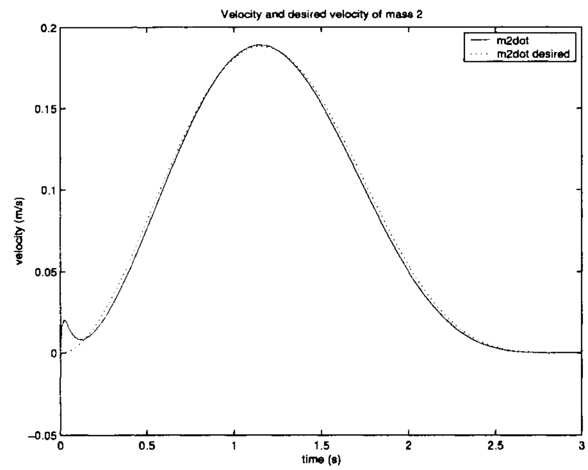

Figure 5.13 Velocity and desired velocity of m_2 for optimal sensitivity

to be minimized in this optimization problem is,

$$T = -(b_u + b_l)$$

where b_u and b_l are the upper and lower tolerances of the nominal value for L, length of the rod. Only the tolerance on the length of the rod is considered for this problem. The system is subject to the constraints,

$$\begin{aligned} g &= [(\|J_{opt} - J_u\| - J_{opt} * .002) \quad (\|J_{opt} - J_l\| - J_{opt} * .002)]^T \\ h &= [\ \ ] \end{aligned}$$

where g is the inequality constraints and h is the equality constraints. J_u and J_l is the cost evaluated at the upper and lower tolerance value of L, respectively. J_{opt} is the optimal cost obtained from the optimization for system response problem.

The results from this optimization are presented in table 5.2, and in Figs. 5.17, 5.18, 5.19, 5.20 and 5.21. The tolerance band increased approximately 3883% from the tolerance for the system response optimized system.

Table 5.1 is a summary of the design variables for the optimization problems above, performance optimization (P-opt), variability optimization (V-opt), and sensitivity opti-

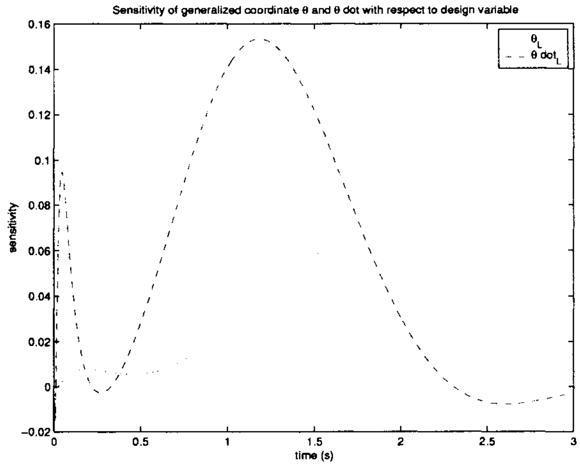

Figure 5.14 θ_L and $\dot{\theta}_L$ for optimal sensitivity

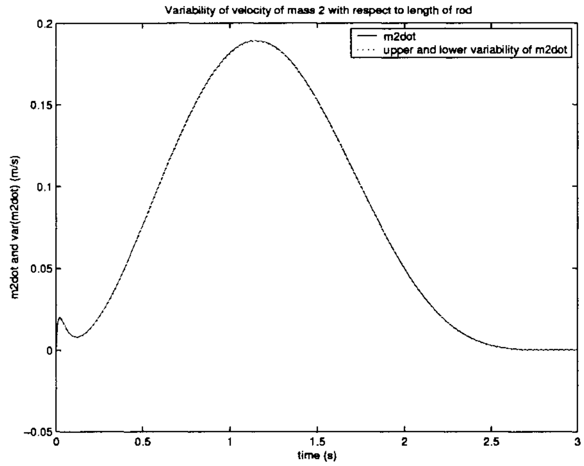

Figure 5.15 Variability of the velocity of m_2 for optimal sensitivity

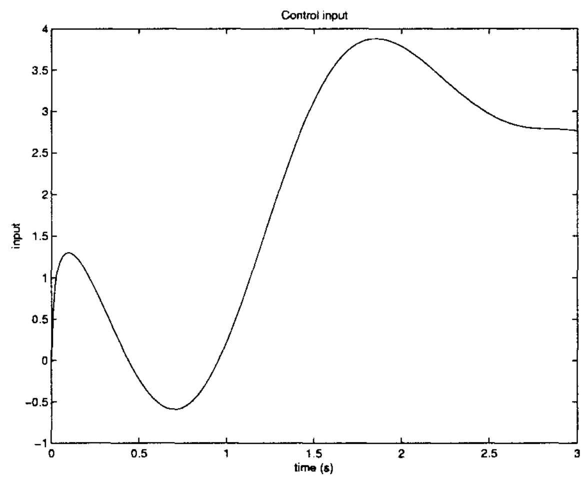

Figure 5.16 Control input for optimal sensitivity

mization (S-opt). Table 5.2 is a summary of the manufacturability optimization (T-opt).

5.2 Multibody Constrained Formulation

The same double slider problem was solved using the multibody constrained formulation. Only the optimization for tracking will be presented for this formulation to verify that both formulations will yield similar results. Two methods of controller im-

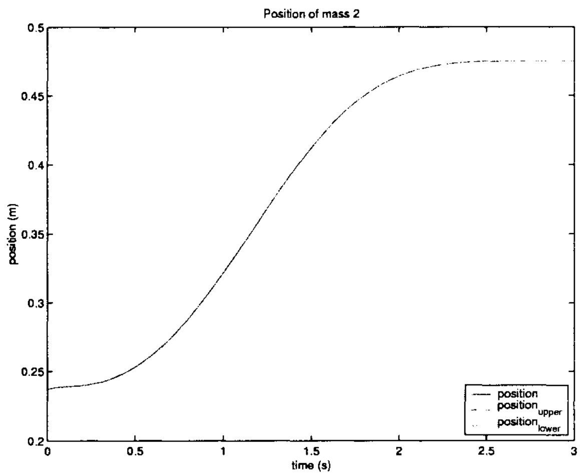

Figure 5.17 Position of m_2 for optimal manufacturability

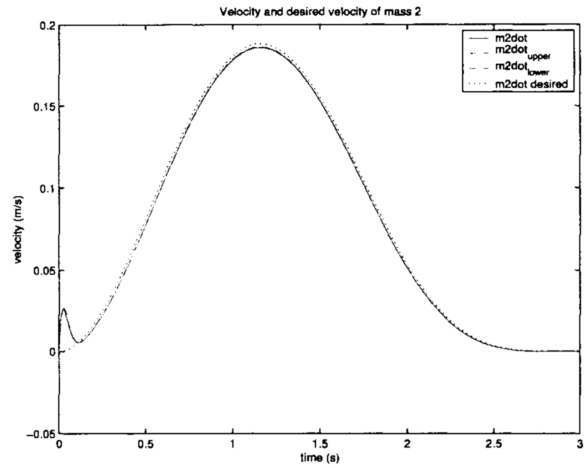

Figure 5.18 Velocity and desired velocity of m_2 for optimal manufacturability

Table 5.1 Optimal design variables for double slider problem

	L (m)	**k (N/m)**	**c (N·s/m)**	K_p	K_d	**cost**
P-opt	0.2500	146.7215	19.0766	44.8211	6.147	0.002122068
V-opt	0.2500	149.1385	21.4106	46.2066	6.0764	0.002228171
S-opt	0.4748	26.5086	8.0820	390.9941	12.4895	0.002228199

plementation will be presented for the performance optimization. The first method is a PD-type controller implemented exactly the same way as the PD-type controller in the Lagrange formulation. The second controller is a generic controller implemented as derived in Section 2.7.

The formulation of this new problem will be briefly presented as well. A complete formulation will not be presented because the method has worked for the Lagrangian formulation, and the multibody constrained formulation gave similar results.

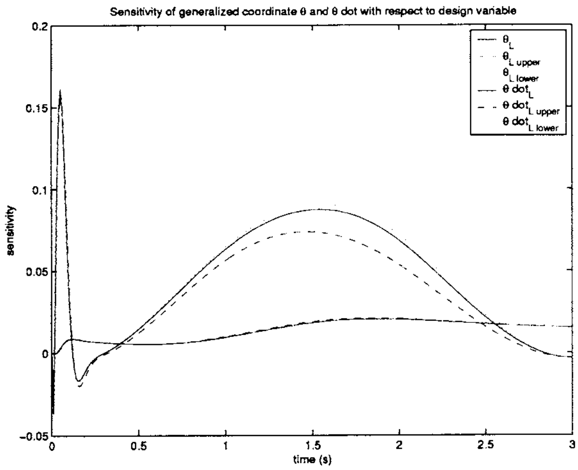

Figure 5.19 θ_L and $\dot{\theta}_L$ for optimal manufacturability

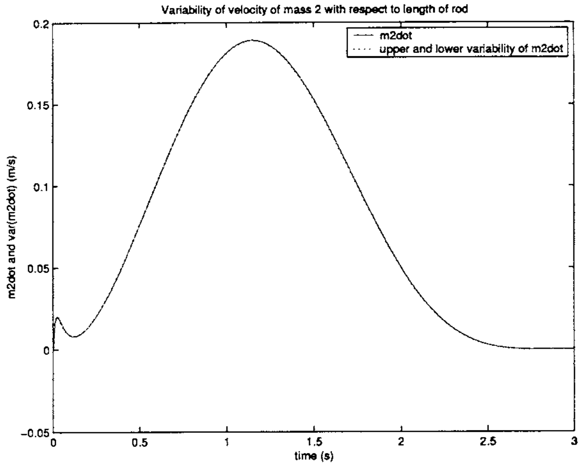

Figure 5.20 Variability of the velocity of m_2 for optimal manufacturability

Table 5.2 Tolerance values of optimal systems for double slider

	upper tol. (m)	**lower tol. (m)**	**tol. band (m)**
P,V,S-opt	+0.005124357	+0.005823467	0.0006991
T-opt	+0.0227490862	-0.004093213	0.0268423

5.2.1 Developing Constraint Matrix Φ and Force Vector Q for Double Slider Mechanism

The constraint matrix Φ and force matrix Q will be derived in this section. For an explanation of the method used to develop Φ and Q, please refer to section 2.3.1. All equations, for both controller implementations, were first derived by hand, and then by the symbolic preprocessor.

The double slider has four bodies, the ground (body 1), the upper mass (body2), the rod (body 3), and the lower mass (body 4), refer to Fig. 5.1. The maximal set of general coordinates is used, therefore the generalized coordinate vector is,

$$q = [x_1 \;\; x_2 \;\; x_3 \;\; x_4 \;\; y_1 \;\; y_2 \;\; y_3 \;\; y_4 \;\; \theta_1 \;\; \theta_2 \;\; \theta_3 \;\; \theta_4]^T$$

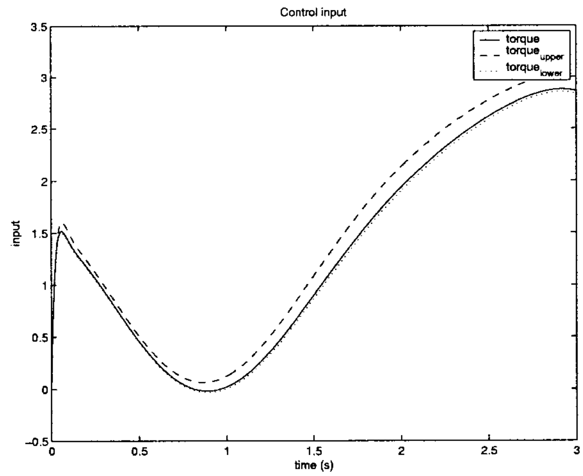

Figure 5.21 Control input for optimal manufacturability

and the velocity terms are,

$$\dot{q} = [\dot{x}_1 \;\; \dot{x}_2 \;\; \dot{x}_3 \;\; \dot{x}_4 \;\; \dot{y}_1 \;\; \dot{y}_2 \;\; \dot{y}_3 \;\; \dot{y}_4 \;\; \dot{\theta}_1 \;\; \dot{\theta}_2 \;\; \dot{\theta}_3 \;\; \dot{\theta}_4]^T$$

The ground constraint is completely constrained. The ground coordinates are x_1, y_1 and θ_1 (q_1, q_5 and q_9), all three coordinates are set to zero. The next constraint is the translational joint between the masses and the ground. The translational joint constrains both masses so they cannot rotate, body 2 cannot move in the x-direction and body 4 cannot move in the y-direction, therefore coordinates x_2, θ_2, y_4 and θ_4 (q_2, q_{10}, q_8 and q_{12}), are also set to zero. The final set of constraints is the revolute joint between the rod (body 3) and the two masses (body 2 and body 4). Using Eqn. (2.35) for the revolute joints, the constraint matrix is,

$$\Phi = \begin{bmatrix} q_1 \\ q_5 \\ q_9 \\ q_2 \\ q_{10} \\ q_8 \\ q_{12} \\ (q2 + 0 - 0) - (q_3 + -\frac{L}{2}cos(q_{11}) - 0) \\ (q6 + 0 - 0) - (q_7 + -\frac{L}{2}cos(q_{11}) - 0) \\ (q4 + 0 - 0) - (q_3 + \frac{L}{2}cos(q_{11}) - 0) \\ (q8 + 0 - 0) - (q_7 + \frac{L}{2}cos(q_{11}) - 0) \end{bmatrix}$$

The forces in this system are: the spring force, the damper force, the controller force and the force of the weight of each body. The forces are applied at the corresponding generalized coordinate location. The Q vector is,

$$Q = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{2}k(q_4 - q_{40})^2 \\ -m_1 g \\ -m_2 g \\ -m_3 g \\ -m_4 g \\ 0 \\ 0 \\ -(K_p(q_{11} - q_{11desired}) + K_d(\dot{q}_{11} - \dot{q}_{11desired}) + c\dot{q}_{11}) \\ 0 \end{bmatrix}$$

The Q vector for the general controller case is slightly different,

$$Q = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{2}k(q_4 - q_{40})^2 \\ -m_1 g \\ -m_2 g \\ -m_3 g \\ -m_4 g \\ 0 \\ 0 \\ -(u + c\dot{q}_1 1) \\ 0 \end{bmatrix}$$

where

$$u = Cx + Dy$$

$$C = [c_1 \quad c_2]$$

$$D = [d]$$

$$y = [\theta - \theta_{desired}]$$

The general controller is only dependent on $\theta - \theta_{desired}$ whereas the PD controller is dependent on $\theta - \theta_{desired}$ and $\dot{\theta} - \dot{\theta}_{desired}$, so there will be some difference in the results, however, this controller should be adequate.

With the Φ and Q matrices developed, the rest of the terms can be obtained by taking appropriate derivatives.

5.2.2 Optimization for Performance with PD controller

Similar to the tracking optimization for the Lagrangian formulation, tracking optimization for the multibody constrained formulation of the mechanism was conducted. The performance function to be minimized, once again, is,

$$f = \int_{t_0}^{t_f} (Lcos(\theta)\dot{\theta} - Lcos(\theta_{desired})\dot{\theta}_{desired})^2$$

and the constraints are,

$$\begin{aligned} g = [&.25 - L \quad L - 2.5 \quad -k \quad k - 500 \quad -c \quad c - 500 \\ &- K_p \quad K_p - 500 \quad -K_d \quad K_d - 500]^T \\ h = [&\ ] \end{aligned}$$

Results are shown in Figs. 5.22, 5.23 and 5.24. Sensitivity and variability plots were similar to the Lagrange case, but not shown here.

The results obtained by this optimization are similar to those obtained in the Lagrange formulation.

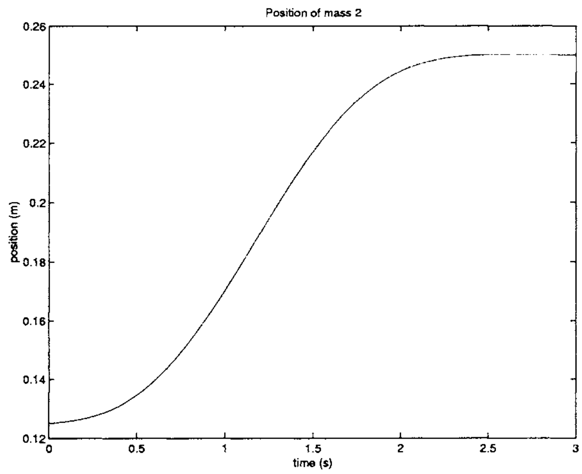

Figure 5.22 Position of m_2 for optimal tracking using the multibody constrained formulation

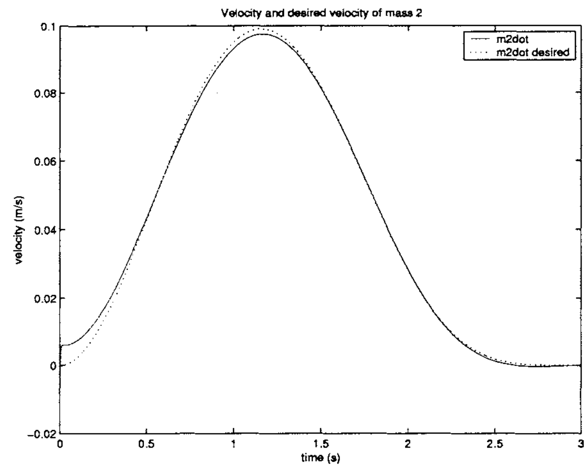

Figure 5.23 Velocity and desired velocity of m_2 for optimal tracking using the multibody constrained formulation

5.2.3 Performance Optimization with Generic Controller

For this case, the general controller was implemented as discussed in Chapter 2. In order to verify this method, the same performance optimization was conducted and results compared. The results can be seen in Figs. 5.25, 5.26 and 5.27.

The results are very similar to the Lagrange formulation, therefore, the generic controller method will work for designing controllers with the multibody constrained formulation.

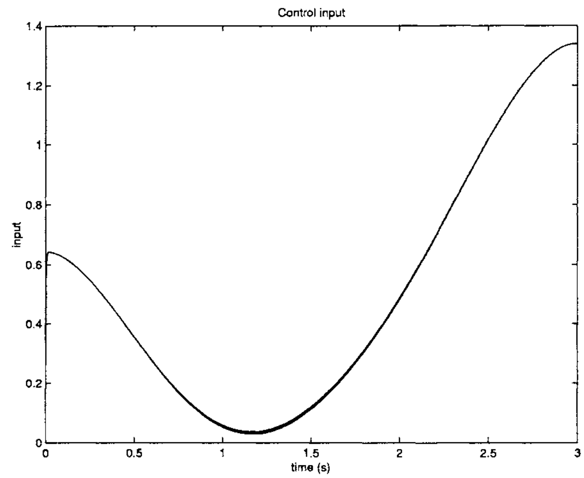

Figure 5.24 Control input for optimal tracking using the multibody constrained formulation

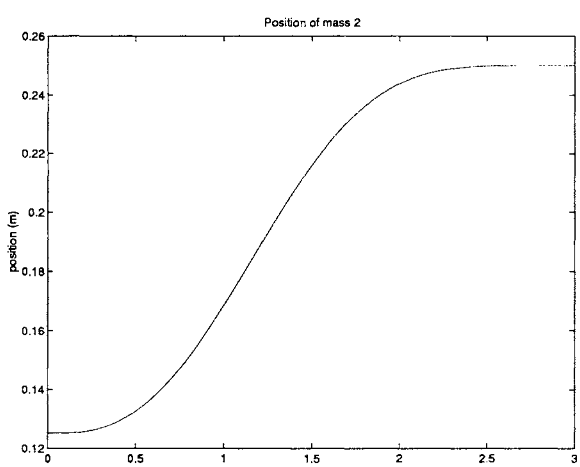

Figure 5.25 Position of m_2 for optimal tracking using the multibody constrained formulation with the generic controller

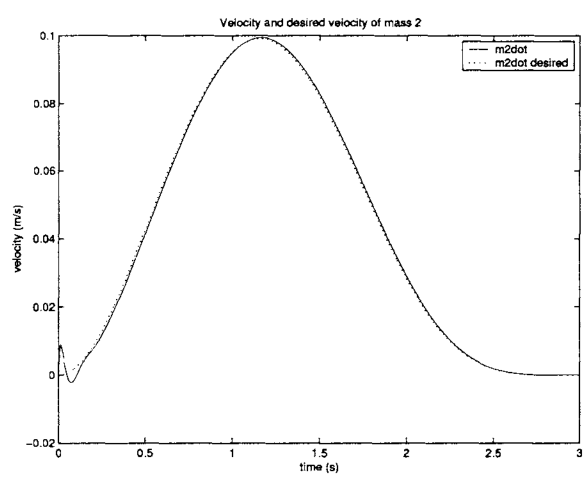

Figure 5.26 Velocity and desired velocity of m_2 for optimal tracking using the multibody constrained formulation with the generic controller

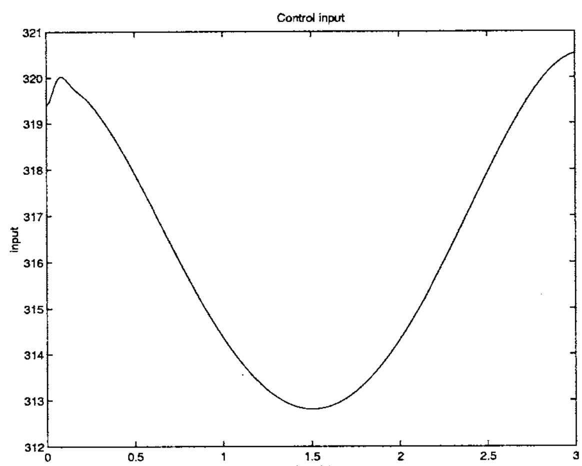

Figure 5.27 Control input for optimal tracking using the multibody constrained formulation with the generic controller

6 CONSTRUCTION LOADER LINKAGE PROBLEM

Once validated on a proof-of-concept system, the integrated design methodology was demonstrated on a real-life system, namely, the design of a construction loader linkage (see an example in Fig. 6.1). This type of loader is commonly used in heavy construction applications. The boom which holds the bucket extends approximately 3 meters and weighs approximately 1500 kilograms. The bucket weighs approximately 1700 kilograms. The system has two degrees of freedom which are controlled by hydraulics. Two hydraulic cylinders control the lift of the boom and one hydraulic cylinder controls the tilt of the bucket. These will be referred to as lift and tilt hydraulics, respectively. The global reference frame is located at the pin joint that connects the boom to the tractor, see Fig. 6.2. The local reference frame for each body is located at the center of mass of the respective body. The linkage has nine bodies, therefore, the dynamic formulation has 27 generalized coordinates. Table 6.1 gives the mass and inertia properties, as well as the start position of the center of mass of each body. Note that since there are two hydraulic cylinders and barrels to control the lift, the mass and inertias are doubled. The system has two degrees of freedom yielding 25 constraint equations. Primarily, the constraint equations are in the form of the revolute joint equations given by Eqn. 2.35. The constraint equations for the sliding joints (the hydraulic cylinders), are slightly different, however the same principle can be applied to develop those equations

The parameters of the controller that regulates the tilt angle of the bucket and the location of the pin joint to ground connection for the tilt hydraulic barrel are the design variables for this problem. For the problem formulation under consideration, only the

Figure 6.1 Picture of construction linkage on tractor

linkage dynamics are considered. Dynamics of the hydraulic actuators is omitted. In the first case, only the controller optimization is considered for keeping the bucket leveled while the boom is raised. In the next case, an integrated design approach is used to show how the structure and the controller can be optimized simultaneously in an integrated framework for bucket leveling. Finally, the same integrated design approach is used to design for minimum sensitivity and manufacturability. The boom motion considered is its travel from the start position (see Fig. 6.2) to an angle of approximately 45^o in 3 seconds. The start position corresponds to the bucket resting on the ground with no bucket tilt, as shown in Fig. 6.2. Although several other performance measures can be used, the problem of bucket leveling is used as an example to demonstrate the capabilities of the approach developed in this research on a realistic problem of large scale.

6.0.4 Controller Optimization for Bucket Leveling

This section presents the results from the optimization of the controller parameters only for bucket leveling. The performance function used for this optimization and all

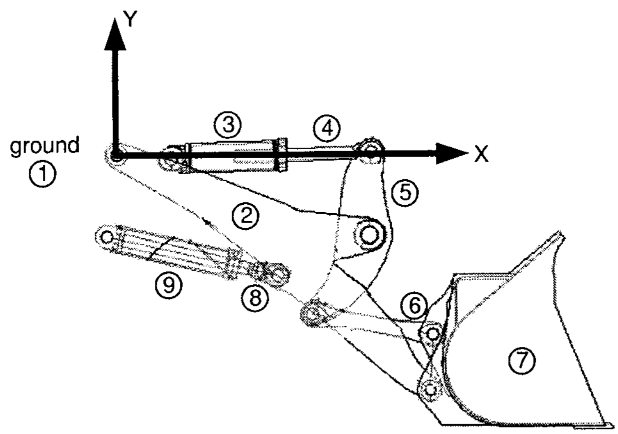

Figure 6.2 Picture of construction linkage on tractor

other optimizations (excluding optimization for sensitivity and manufacturability) is,

$$J = \int_{t_0}^{t_f} \theta_{bucket}^2 + \dot{\theta}_{bucket}^2 + u^T R u$$

where θ is the bucket angle relative to the global reference frame, $\dot{\theta}$ is the angular velocity of the bucket, u is the control input and R is the control penalty used as a design parameter. A second order dynamic controller was used in generic form for optimization. The controller state-space model used is given below where a_i and c_i are the controller design parameters.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_1 & -a_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} y$$

$$u = [c_1 \quad c_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The input to the controller y is the bucket angle θ, and the output u is the force applied to the hydraulic piston. The controller was optimized for the bucket leveling problem. The boom and bucket responses and control input time history are given in Figs. 6.3,

Table 6.1 Mass, inertia and initial center of mass location for links

Body	Name	Mass (kg)	Inertia ($kg \cdot m^2$)	X (m)	Y (m)	θ (rad)
1	ground	0	1	0	0	0
2	boom	1437.68	731.00	1.34	-.764	0
3	barrel	148.96	19.51	.931	-.015	.026
4	cylinder	99.46	14.67	1.52	-.001	.026
5	bell	322.434	48.20	1.85	-.649	0
6	link	88.61	10.85	1.98	-1.33	-.173
7	bucket	1910.11	764.39	3.00	-1.56	0
8	cylinder	2*101.96	2*17.13	.789	-.829	.228
9	barrel	2*102.146	2*13.91	.413	-.750	.228

6.4 and 6.5. The optimal controller design gave the control parameters given in Table 6.2.

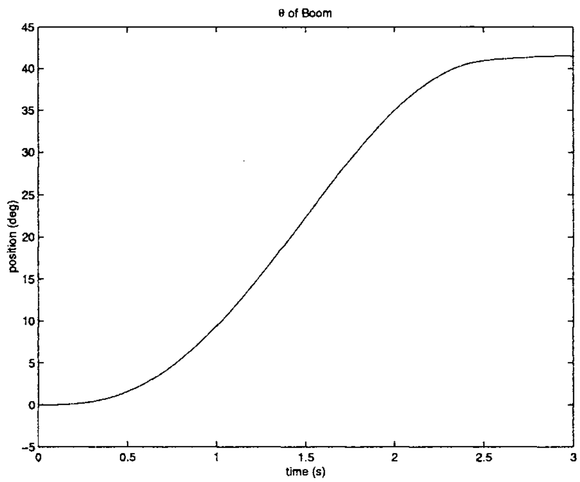

Figure 6.3 Trajectory of boom for bucket leveling

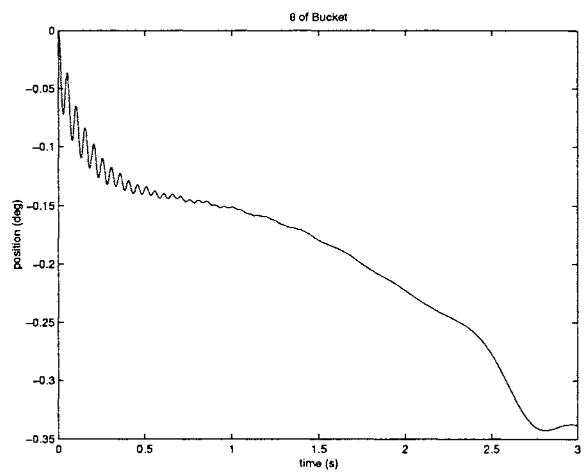

Figure 6.4 Trajectory of bucket for bucket leveling

The results show that a controller can effectively keep the bucket leveled while the boom travels from the start position to an angle of approximately 45^o in 3 seconds.

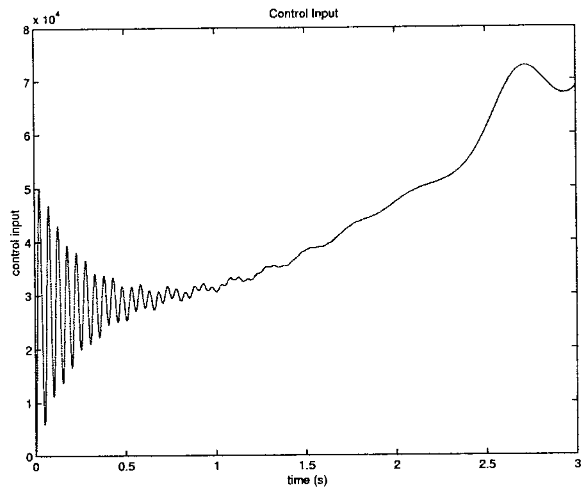

Figure 6.5 Control input for bucket leveling

6.0.5 Control-Structure Integrated Design for Bucket Leveling

In the second case, instead of just optimizing the controller, both the controller and the structure is optimized. The design variables considered are the controller design variable, same as in the previous section, and the x- and y-locations of the pin joint for the tilt hydraulics. The optimization results are given in Table 6.2, and the time responses corresponding to these design variables are given in Figs. 6.6, 6.7, 6.8, 6.9 and 6.10.

These results show that much less control force is needed in order to keep the bucket leveled compared to the previous case. Additionally, the integrated design approach resulted in better leveling performance, numerically represented by a 57% decrease in cost.

6.0.6 Sensitivity Optimization Using Integrated Design

In this case the design variables are once again, the controller design variables and the x- and y-location of the pin joint for the tilt hydraulics. The results for the sensitivity optimization are given in Table 6.2 and the time response in Figs. 6.11, 6.12, 6.13, 6.14 and 6.15.

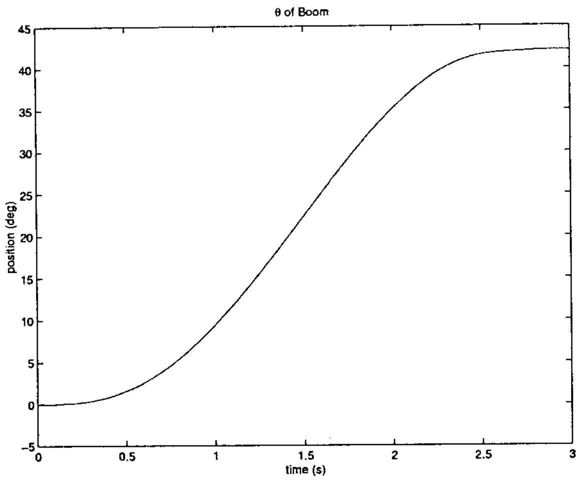

Figure 6.6 Trajectory of boom for bucket leveling using the integrated design approach

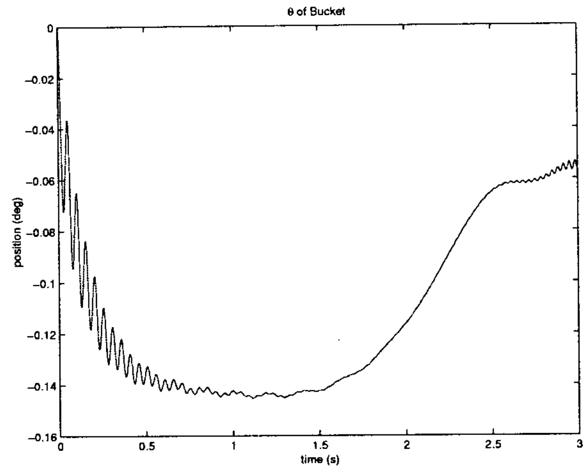

Figure 6.7 Trajectory of bucket for bucket leveling using the integrated design approach

It is clear from the plots that the sensitivity with respect to the design variables is greatly reduced. This means, for example, that the location of the tilt hydraulic barrel pin joint can change (vary) without greatly affecting the system performance. This fact can be used to relax the tolerances used for manufacturing. This case will be presented in the next section.

In the double slider problem, optimization for variability was also presented, but since the performance function has no dependence on design variables, it is not relevant in this problem.

Table 6.2 is a summary of the design variables for the three optimization problems above, controller optimization for performance (P-opt), integrated control-structure optimization for performance (P-opt int) and integrated control-structure optimization for sensitivity (S-opt int).

6.0.7 Manufacturability Optimization Using Integrated Design

The design variables in this case are the tolerances associated with the pin joint for the tilt hydraulics to ground pin joint connection. The results can be seen in Figs. 6.16,

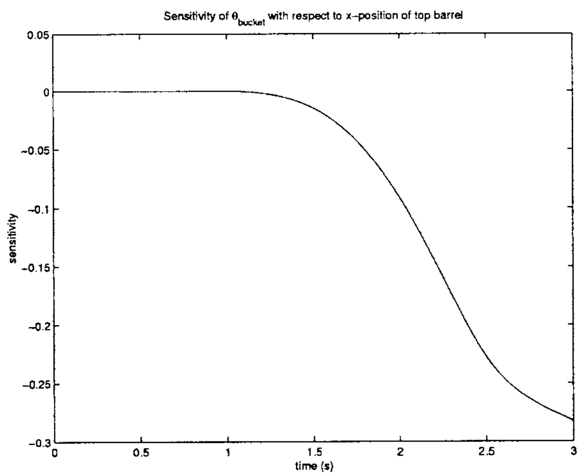

Figure 6.8 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for bucket leveling using the integrated design approach

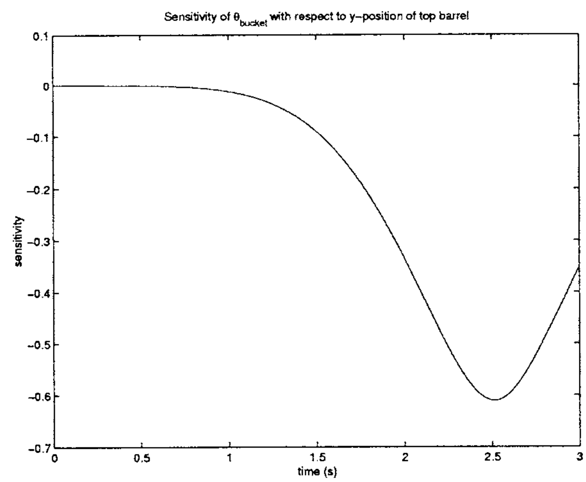

Figure 6.9 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for bucket leveling using the integrated design approach

Table 6.2 Optimal design variables for construction linkage

	x (m)	**y** (m)	a_1	a_2	c_1	c_2	**cost**
P-opt	0.4211	-0.0288	31011.33	1008.88	3.59432e11	5.19268e10	3.967
P-opt int	-0.0221	0.6457	31011.30	1008.63	3.59432e11	5.19268e10	2.278
S-opt int	-0.3279	0.5797	31011.30	1008.64	3.59432e11	5.19268e10	2.290

6.17, 6.18, 6.19, 6.20, and table 6.3.

The results show that by increasing the tolerances, the same performance can be achieved. The current and manufacturability optimized (T-opt) results are shown.

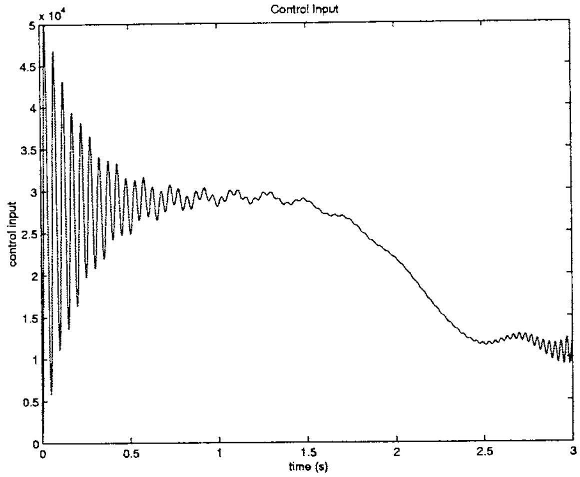

Figure 6.10 Control input for bucket leveling using the integrated design approach

Table 6.3 Tolerance values for construction linkage

		upper tol. (m)	lower tol. (m)	tol. band (m)
current	x-tol	-0.001000	0.001000	0.002000
current	y-tol	-0.001000	0.001000	0.002000
T-opt	x-tol	-0.081970	0.099321	0.181291
T-opt	y-tol	-0.016188	0.022528	0.038716

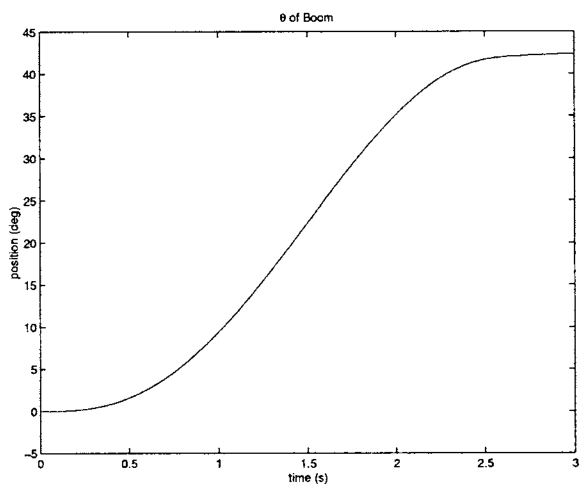

Figure 6.11 Trajectory of boom for sensitivity optimized system using the integrated design approach

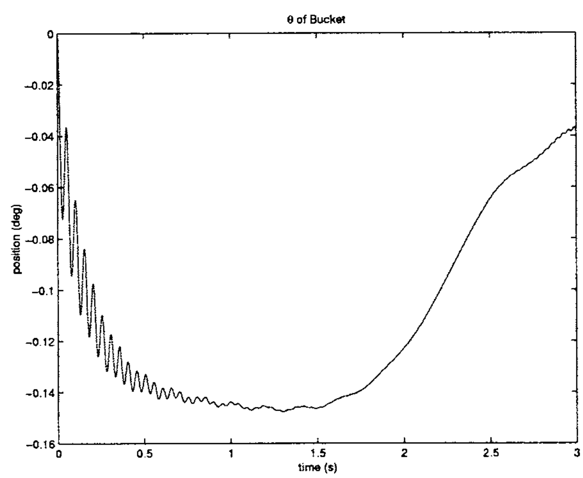

Figure 6.12 Trajectory of bucket for sensitivity optimized system using the integrated design approach

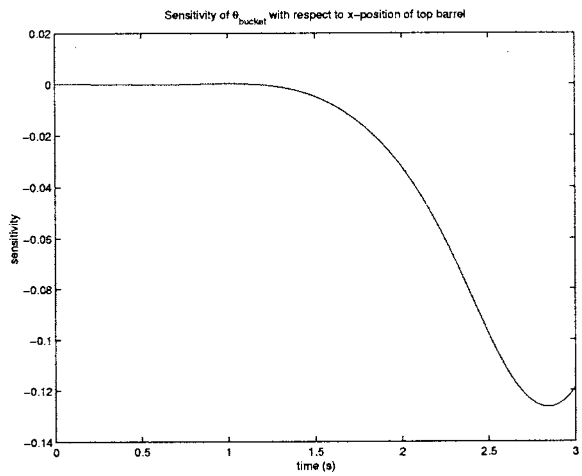

Figure 6.13 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for sensitivity optimized system using the integrated design approach

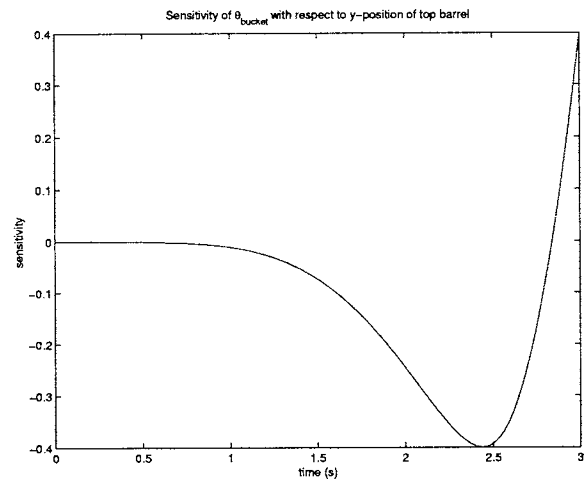

Figure 6.14 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for sensitivity optimized system using the integrated design approach

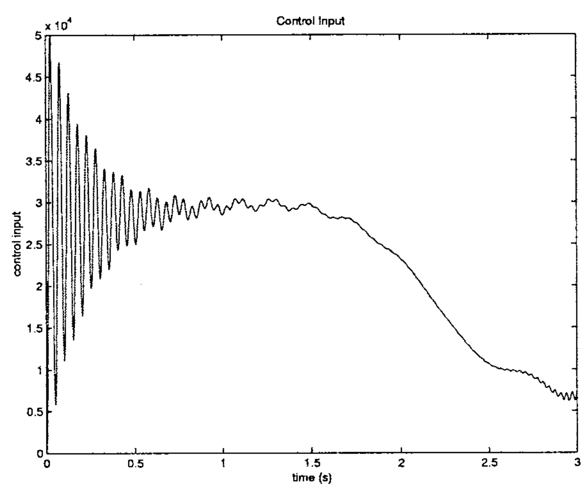

Figure 6.15 Control input for sensitivity optimized system using the integrated design approach

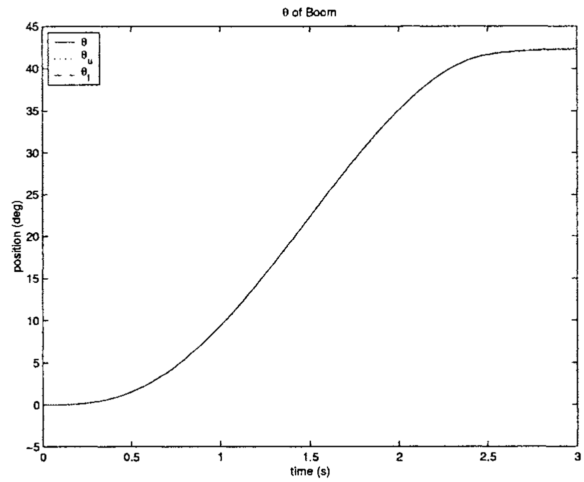

Figure 6.16 Trajectory of boom for optimal manufacturability using the integrated design approach

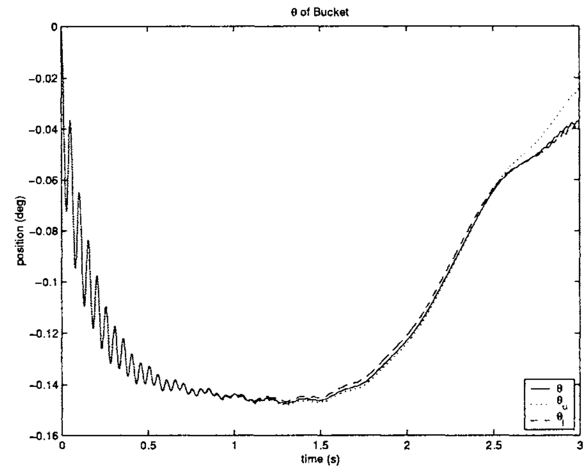

Figure 6.17 Trajectory of bucket for optimal manufacturability using the integrated design approach

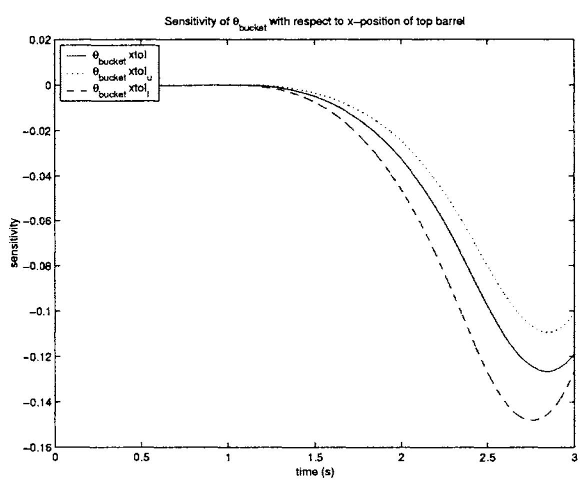

Figure 6.18 Sensitivity of bucket angle with respect to x-position of tilt hydraulic pin to ground joint for optimal manufacturability using the integrated design approach

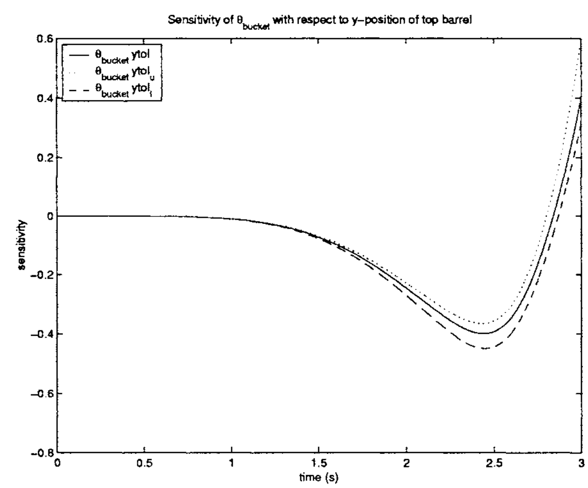

Figure 6.19 Sensitivity of bucket angle with respect to y-position of tilt hydraulic pin to ground joint for optimal manufacturability using the integrated design approach

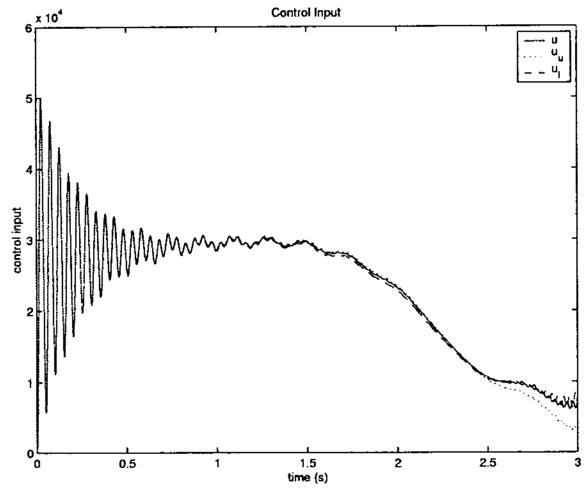

Figure 6.20 Control input for optimal manufacturability using the integrated design approach

7 CONCLUSIONS

A general methodology for designing optimal, nonlinear, controlled mechanisms in an integrated framework was presented. An integrated approach to the design using multidisciplinary optimization proved to be extremely beneficial as it yields a *truly* optimal design. A high degree of automation can be achieved enabling a designer to change designs and reformulate problems quickly. The methodology developed enables the designer to optimize for many different performance objectives, including tracking, sensitivity, variability and manufacturability. To the best of our knowledge, this capability is unavailable in the current design methodologies.

The methodology developed in this research is validated using a proof-of-concept system as well as a real-life mechanical system. Each step of the design process has been validated to ensure the dependability of the method. The sensitivity equations were validated with the finite difference check method. Two independent dynamic formulations were used to cross check results. The controller formulation used two different types of controllers. The equations for the double slider mechanism were first hand derived and then symbolically derived to show that the symbolic preprocessor is a viable tool for generating necessary closed-form equations.

Two different dynamic formulations were used to validate the methodology and to illustrate the advantages of using the multibody constrained formulation. The multibody constrained formulation lends itself to a high level of automation via symbolic generation of equations without equation swell. Although matrices can get relatively large, they are very sparsely populated, and often times narrow banded. Additionally, a system configuration can be changed easily without complete reformulation. Finally, a generalized approach to controller implementation is straight forward and the controller dynamics can be solved simultaneously with the system. The generalized controller enables a designer to change the controller parameters and type of controller rather easily independent of system dynamics. The program could be extended to enable the controller order to be a design variable as well. Both SISO and MIMO controller implementations are simple procedures.

The sensitivity based approach is the heart of the proposed methodology. A sensitivity based method enables the designer to get valuable insight into the problem at hand. Since the order of integration for the state sensitivity derivatives can be interchanged, it is possible to solve for state sensitivities and states simultaneously.

Symbolic derivation of equations is very convenient for large scale systems. Although it adds some computational cost, it allows the designer to quickly reformulate a new system or modify the existing system. Symbolic generation of equations increases the level of automation in the formulation and allows for more flexibility in the methodology.

The integrated framework enables the designer to explore multiple design spaces while making design decisions. The integrated design method entails a multi-objective optimization problem. In this thesis, four different performance/design objectives are considered for the optimization problem. Each optimization problem gives valuable insight into the different design considerations. A *truly* optimal design can be accomplished as both structure and control parameters can be considered as design variables and adjusted simultaneously to achieve the desired performance.

7.1 Suggestions for Future Work

The methodology developed in the course of this research was illustrated on controlled planar nonlinear mechanical systems. The software developed can be readily applied to many other applications including vehicular analysis, deployable structures, robotic systems, precision machine tools, etc. Furthermore, this approach can be extended to numerous classes of integrated systems, not just the systems presented. The software developed can be improved further to include spatial mechanical systems, or flexible structures. The approach can be extended to include a variety of performance functions and a variety of design variables.

The computational efficiency is one of the areas of improvement. Currently, a large scale optimization problem (approximately 500 states) takes approximately eight to twelve hours to run on a desktop computer with a single 1.6MHz processor and 512MB of RAM. A parallel programming technique can be implemented to speed up the solution process. Currently the optimization problem runs in a Matlab© environment. More efficient code, such as C or C++ can be utilized to improve the run-time. An optimizer with a more sophisticated search method can also be employed to better explore the design space through the optimization problem and speed up the search.

Robustness analysis can be further explored. A better understanding between the system robustness and sensitivity can be developed and used to design more robust systems. Uncertainties can be directly modelled in the system in a traditional robust design framework which will allow for a less conservative design.

The controller formulation can also be improved further. The standard templates for LQR, LQG, H_∞, and other optimal controller formulations can be employed. Controller type also can be considered as a design variable.

A virtual reality post processing interface can be developed so that as the design is being developed and modified, it can also be tested through a haptic interface in a virtual environment.

BIBLIOGRAPHY

[1] Krishnaswami, P., and Kelkar, A. G., *Optimal Design of Controlled Multibody Dynamic Systems for Performance, Robustness and Tolerancing*, 2002.

[2] Daniels, R., *An Introduction to Numerical Methods and Optimization Techniques*, North-Holland, New York, 1978.

[3] Belegandu, A. D., and Chandraputla, T. R., *Optimization Concepts and Applications in Engineering*, Prentice Hall, 1999.

[4] Cottle, R. W., Krarup, J., *Optimisation Methods*, The English Universities Press Ltd., London, 1974.

[5] Arora, J., *Introduction to Optimum Design*, McGraw-Hill, Inc., 1989.

[6] Orlandea, N., Chace, M.A., and Calahan, D.A., *A Sparsity Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems, Parts I and II*, ASME Journal of Engineering for Industry, Ser. B, v99, 1977.

[7] Wehage, R.A., and Haug, E.J., *Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic Systems*, ASME Journal of Mechanical Design, v104, 1982.

[8] Ashrafiuon, H., and Mani, N. K., *Analysis and Optimal design of Spatial Mechanical Systems*, Journal of Mechanical Design, v112, 1990, pp. 200-207.

[9] Krishnaswami, P., and Bhatti, M. A., *A General Method for Design Sensitivity Analysis of Constrained Dynamic Systems*, ASME Paper 84-DET-132, 1984.

[10] Mani, N. K., and Haug, E. J., *Singular Value Decomposition for Dunamic System Design Sensitivity Analysis*, Engineering with Computers, v112, 1990, pp. 200-207.

[11] Patil, U., and Krishnaswami, P., *Minimum Sensitivity Design of Planar Kinematic Systems*, ASME Design Automation Conference, 1990.

[12] Kajiwara, I., and Nagamatsu, A., *Optimum Design of Structure by Sensitivity Analysis*, Proceedings of the 1991 Asia-Pacific Vibration Conference, 1991, pp. 2.61-2.66.

[13] Gutkowski, Witold, Bauer, Jacek, *Sensitivity and optimum structural design*, Proceedings of the IASS-ASCE International Symposium 1994 on Spatial, Lattice and Tension Structures, 1994, p 330-339.

[14] Krishnaswami, P., and Bhatti, M.A., *Symbolic Computing in Optimal Design of Dynamic Systems*, ASME Paper 85-BET-76, 1985.

[15] Krishnaswami, P., *Symbolic Computing and Automatic Differentiation for Sensitivity Analysis*, Proceedings of the EF/ASME Optimization in Industry Conference, 1997.

[16] Bischof, C.H., *On the Automatic Differentiation of Computer Programs and an Application to Multibody Systems*, Tech. Rep. MCS-P507-0495, Argonne National Lab., 1995.

[17] Eberhard, P., *Analysis and Optimization of Complex Multibody Systems using Advanced Sensitivity Methods*, Ph.D. Dissertation, Dept. of Mathematics, University of Stuttgart, 1995.

[18] Ashrafiuon, H. *Applications of symbolic computing for numerical analysis of mechanical systems* American Society of Mechanical Engineers, Design Engineering Division (Publication) DE, v 19-3, n pt 3, 1989.

[19] Kajiwara, I. and Nagamatsu, A., *Integrated Design of Structure and Control System considering Performance and Stability*, Proceedings of IEEE 8th Conference on Control Applications, 1999, pp. 86-91.

[20] Kajiwara, I. and Nagamatsu, A., *Concurrent Optimum Design of Structure and Control System by Genetic Algorithm*, Proceedings of the 3rd International Conference on Motion and Vibration Control, Vol. 3, 1996-9, pp.327-332.

[21] Kajiwara, I., Tsujioka, K., and Nagamatsu, A., *Approach for Simultaneous Optimization of a Structure and Control System*, AIAA Journal, Vol. 32, No. 4, 1994, pp. 866-873.

[22] Tsujioka, K., Kajiwara, I. and Nagamatsu, A., *Integrated Optimum Design of Structure and Control System*, AIAA Journal, Vol. 34, No. 1, 1996, pp. 159-165.

[23] Tsujioka, K., Kajiwara, I. and Nagamatsu, A., *Integrated Optimum Design of Structure and H^∞ Control System*, AIAA Journal, Vol. 34, No. 1, January 1996.

[24] Mexxac, A., and Malek, K., *Control-Structure Integrated Design*, AIAA Journal, Vol. 30, No. 8, 1992, pp. 2124-2131.

[25] Cheng, F. Y., and Li, D., *Multiobjective Optimization of Structures with and Without Control*, Journal of Guidance, Control and Dynamics, Vol. 19, No. 2, 1996, pp. 392-397.

[26] Milman, M.; Salama, M.; Wette, M.; Bruno, R., *Multiobjective approach to integrated control, structure, and optical design*, Proceedings of the 32nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 1991, p. 846.

[27] Santoro, Esamuele, *Global methods in multi-objective optimization and their application to a mechanical design problem* Computers in Industry, v 18, n 2, Feb, 1992, p 169-175.

[28] Kalsi, M., Hacker, K., and Lewis, K., 2001, *A Comprehensive Robust Design Approach for Decision Trade-Offs in Complex Systems Design*, Journal of Mechanical Design, Vol. 123, No. 1, pp.1-10.

[29] Baumgarte, J. *Stabilization of Constraints and Integrals of Motion in Dynamical Systems*, Computational Methods in Applied Mechanics and Engineering, v1, pp. 1-16, 1972.

[30] Greenwood, D. T., *Principles of Dynamics, Second Edition*, Prentice Hall, 1988.

[31] Dias, J. P., and Pereira, M. S., *Design for vehicle crashworthiness using multibody dynamics*, International Journal of Vehicle Design, v15, n6, 1994, pp. 563 - 577.

[32] Chang, C. O., and Nikravesh, P. E., *Optimal Design of Mechanical Systems with Constraint Violation Stabilization Method*, Journal of Mechanisms, transmissions, and Automatic Design, v107, December 1985.

[33] Lee, M. Y., Erdman, A. G.,Faik, S., *Generalized Performance Sensitivity Synthesis Methodology for Four Bar Mechanisms*, Mechanism and machine Theory, v34, n7, 1999, pp. 1127-1139.

[34] Dhingra, A. K., and Rao, S. S., *Integrated Optimal Design of Planar Mechanisms Using Fuzzy Theories*, American Society of Mechanical Engineers, Design Engineering Division (Publication) DE, v19-2, n pt2, 1989, Advances in Design Automation September 17-21 1989.

[35] Sanchez, N.E. *Nonlinear dynamics and control of a four-wheel steering vehicle using symbolic-numerical approach* International Journal of Vehicle Design, v 15, n 1-2, 1994, p 81-98.

[36] Haug, E.J. *Computer-Aided Kinematics and Dynamics of Mechanical Systems*, Allyn and Bacon Publishers, 1989.

[37] Ohashi, F., Kajiwara, I., Omori, T. and Arisaka, T., Simultaneous *Design of Mechanism and Control System for Smart Strucutres (Optimization of Piezoelectric Placement and Application to Magnetic Disk Drive)*, Proceedings on SICE Annual Conference 2002, 2002, CD-ROM(No.MP06-1).

[38] Kajiwara, I., Yambe, K. and Nishidome, C., *A Control Method for Non-Linear Time-Varying System Using Mixed Control: Position and Force Control of 2-Link Manipulator*, 2001 ASME Design Engineering Technical Conference, 2001, CD-ROM(No. VIB-21326).

[39] Kajiwara, I., Guo, Z. and Nagamatsu, A., *An Integration of Experimental State-Space Modeling Based on ERA and Control Design*, JSME International Journal (C), Vol. 40, No. 2, 1997, pp.197-202.

[40] Guo, Z., Kajiwara, I. and Nagamatsu, A., *Robust Identificatin and Control Design for Truck Cab Model Device*, JSME International Journal, Vol. 41, No. 2, 1998, pp. 171-177.

[41] Kajiwara, I., and Nagamatsu, A., *Optimum Design of Structure by Sensitivity Analysis, Proceedings of the 11th International Modal Analysis Conference*, 1993, pp. 883-889.

[42] Kajiwara, I., and Nagamatsu, A., *Optimum Design of Structure by Sensitivity Analysis*, DE-Vol.38, Modal Analysis, Modelling, Diagnostics, and Control - Analytical and Experimental, ASME, 1991, pp. 189-194.

[43] Kajiwara, I., and Nagamatsu, A., *An Approach to Simultaneous Optimum Design of Structure and Control Systems by Sensitivity Analysis*, DE-Vol.38, Modal Analysis, Modelling, Diagnostics, and Control - Analytical and Experimental, ASME, 1991, pp. 179-184.

[44] Tsujioka, K., Kajiwara, I., and Nagamatsu, A., *Simultaneous Optimum Design of Structure and Control System using Complex Method*, ASME PVP Conference, PVP-Vol. 305, 1995, pp.297-303.

[45] Tsujioka, K., Kajiwara, I., and Nagamatsu, A., *Simultaneous Optimum Design of Structure and Control System using Complex Method*, JSME, Proceedings of the 2nd International Conference on Motion and Vibration Control, 1994, pp.456-461.

[46] Kajiwara, I., Tsujioka, K., and Nagamatsu, A., *Integrated Optimum Design of Structure and Control System by Modal Analysis*, 1995 ASME Design Engineering Technical Conference, Vol. 3, Part C, (DE-Vol. 84-3), 1995, pp. 487-494.

[47] Kajiwara, I., and Nagamatsu, A., *Simultaneous Optimum Design of Structure and Control Systems by Sensitivity Analysis*, Finite Elements in Analysis and Design, Elsevier, Vol. 14, Nos. 2,3, 1993, pp. 187-195.

[48] Park, J.-H, and Asada, H., *Integrated Structure/Control Design of a Two-Link Nonrigid Robot Arm for High Speed Positioning*, Proceedings of the 1992 IEEE International Conference on Robotics and Automation, Nice, France, May 1992.

[49] Onoda, J. and Haftka, R. T., *An Approach to Structure/Control Simultaneous Optimization for Large Flexible Spacecraft*, AIAA Journal, Vol.25, No. 8, August 1987, pp. 1133-1138.

[50] Hsieh, Jer-Guang, Hsiao, Feng-Hsiag, *Multi-objective controller design for discrete stochastic optimal systems with non-linear time-varying unmodelled dynamics and noise spectral uncertainties*, International Journal of Systems Science, v 21, n 10, Oct, 1990, p 1977-1996.

[51] Liu, G.P., Dixon, R., Daley, S., *Multi-objective optimal-tuning proportional-integral controller design for the ALSTOM gasifier problem*, Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering, v 214, n 6, 2000, p 395-404.

[52] Dell'Amico, M., Maffioli, F., and Martello, S. *Annotated Bibliographies in Combinatorial Optimization*, Wiley Interscience Publication, 1997.