1990

# Automated linearization of nonlinear coupled differential and algebraic equations

Jeffrey Dean Trom
*Iowa State University*

# INFORMATION TO USERS

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

**The quality of this reproduction is dependent upon the quality of the copy submitted.** Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Order Number 9101382

# Automated linearization of nonlinear coupled differential and algebraic equations

Trom, Jeffrey Dean, Ph.D.

Iowa State University, 1990

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

# Automated linearization of nonlinear coupled differential and algebraic equations

by

Jeffrey Dean Trom

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Major: Mechanical Engineering

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

For the Major Department

Signature was redacted for privacy.

For the Graduate College

Iowa State University
Ames, Iowa
1990

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

## ACKNOWLEDGEMENTS

I would like to thank my friend and advisor, Marty Vanderploeg, for his tireless support and encouragement during my Ph.D. program. I would also like to thank Professors Bernard, Fullmer, Huston and McConnell for their valuable input while serving on my committee.

In addition, I would like to thank my colleagues and friends from 0095E, Alan Lynch, Alan Hufnagel, Terran Boylan, Jim Lynch, Chaeyoun Oh, Craig Weiss and Afshin Mikaili both for the technical help they provided and, more importantly, for the diversions they supplied over the course of this research. Very special thanks to my good friend and squash partner, Jay Shannan.

Finally, I would like to thank my family and Karen, for their love and support.

# DESCRIPTION OF NOMENCLATURE

The mathematical symbols used in this thesis consist of scalars, geometric vectors, arrays and matrices. Scalars are represented by lower-case, lightface characters. Geometric vectors are represented by lower-case, lightface characters with an overscore arrow. Arrays and matrices are represented boldface characters with arrays being lower-case and matrices being upper-case. Differentiation with respect to time is denoted by one or more dots.

| | |
|---|---|
| $a$ | scalar |
| $\vec{a}$ | geometric vector |
| $\mathbf{a}$ | array |
| $\mathbf{A}$ | matrix |
| $\dot{a}$ | first derivative with respect to time |
| $\ddot{a}$ | second derivative with respect to time |
| $\tilde{\mathbf{A}}$ | 3x3 skew-symmetric matrix |

# LIST OF NOMENCLATURE

| | |
|---|---|
| $\mathbf{B}$ | velocity transformation matrix |
| $\mathbf{b}$ | right-hand side of constraint velocity equations |
| $\mathbf{f}$ | force vector in Cartesian coordinates |
| $\mathbf{J}_i'$ | inertia tensor of body $i$ in the body-fixed coordinate system |
| $m_i$ | mass of body $i$ |
| $\mathbf{M}_i$ | mass matrix of body $i$ in Cartesian coordinates |
| $\mathbf{M}$ | system mass matrix in Cartesian coordinates |
| $\bar{\mathbf{M}}$ | system mass matrix in generalized coordinates |
| $n_b$ | number of bodies |
| $n_c$ | number of constraints |
| $n_{gc}$ | number of generalized coordinates |
| $n_r$ | rank of the constraint Jacobian matrix |
| $n_{df}$ | number of degrees-of-freedom for the system |
| $\mathbf{q}$ | generalized coordinate positions |
| $\dot{\mathbf{q}}$ | generalized coordinate velocities |
| $\ddot{\mathbf{q}}$ | generalized coordinate accelerations |
| $T$ | total system kinetic energy |
| $\mathbf{v}_i$ | global translational velocities of body $i$ |

| | |
|---|---|
| $W$ | total work done on the system |
| $\mathbf{y}$ | Cartesian coordinate positions |
| $\dot{\mathbf{y}}$ | Cartesian coordinate velocities |
| $\ddot{\mathbf{y}}$ | Cartesian coordinate accelerations |
| $\dot{\mathbf{z}}$ | independent coordinate velocities |
| $\ddot{\mathbf{z}}$ | independent coordinate accelerations |
| $\boldsymbol{\lambda}$ | vector of Lagrange multipliers |
| $\boldsymbol{\Phi}$ | kinematic constraints |
| $\boldsymbol{\Phi}_{\mathbf{q}}$ | constraint Jacobian matrix |
| $\omega_i$ | angular velocities of body $i$ in global coordinates |
| $\omega_i'$ | angular velocities of body $i$ in the body-fixed coordinate system |

# 1. INTRODUCTION

When designing nonlinear dynamic systems, it is often necessary to obtain local linearizations of the equations of motion about an operating point. These linearizations yield the local eigenvalues which characterize the dynamics of the system near the operating point. In addition, linearizations are essential for design of control methodologies and algorithms. Obtaining linearizations is straightforward for systems in which it is possible to derive closed-form analytical equations of motion. However, linearizing large-order systems is extremely difficult, especially when the system moves in three dimensions.

This thesis deals with methods for obtaining linearizations of large-order, three-dimensional dynamic systems. Methods for obtaining time solutions of these mechanical systems are well developed and make use of numerical equations of motion which are derived at each time step during the integration process. A second, more recent approach for obtaining time solutions makes use of symbolic manipulation packages such as MACSYMA, SMP or REDUCE to derive analytical equations of motion. This symbolic approach has an efficiency advantage over the numeric approach because the equations of motion are assembled only once whereas the numeric methods require an assembly process at each time step. However, as the order of the system increases, the symbolic equations of motion become so large that special re-

cursive techniques for evaluating derivatives must be used for even moderately-sized systems. In this case, the symbolic technique becomes less efficient and results in a numerical form of the equations of motion at each integration time step. Therefore, unless the end user is concerned about integration speed or knows apriori that the systems being studied are small enough to be assembled by symbolic packages, the numeric approach offers a more general means of obtaining a time solution. Because of this generality, the linearization algorithms developed in this thesis are based on a numeric rather than a symbolic formulation.

Using a numeric formulation for the equations of motion, this thesis investigates two approaches for obtaining the local linearizations. The first approach uses finite difference to find the linearized mass, stiffness and damping matrices that characterize motion of a linearized system about an operating point. The advantage to this method is that it is relatively easy to implement. The drawbacks are the inherent difficulties in choosing dithering values for the independent variables and the need to evaluate the linearized terms more than once to check convergence.

A second approach is to analytically differentiate all of the terms evaluated in the numerical equation formulation, evaluate the terms at a given operating point and then assemble the terms to form the linearized matrices. The drawback to this approach is the relative complexity in the initial derivation and implementation of the sensitivity terms. The advantage is that the sensitivities of the individual terms with respect to the independent coordinates are derived analytically. This eliminates the problem of choosing and verifying a dithering value, and, as this work will show, is computationally more efficient than the finite difference method even when compared to only one finite difference evaluation. In this thesis, this technique will be referred

to as an *analytical/numerical* method.

The literature contains only one study that has used this analytical/numerical approach. In 1985, Sohoni and Whitesell [1] developed a linearization algorithm based on a Cartesian coordinate method of formulating the numerical equations of motion. Because the formalism uses a Cartesian coordinate set, transformation to first-order equations resulted in extremely large matrices. These first order system matrices are computed with a finite difference technique and are available through the Gear integration algorithm used in the dynamic analysis package. These matrices are then reduced to a minimal size (i.e., order of two times the number of degrees of freedom) by choosing an independent coordinate set and making use of the constraint Jacobian matrix. Although the method resulted in excellent matches between computed eigenvalues and closed-form solutions for several examples, no method of determining the independent coordinates was presented. In addition, this method requires the system to have steady motion with no Cartesian acceleration terms. This restriction eliminates application of this technique to systems such as a vehicle in a steady turn.

The primary contribution of this research is a new analytical/numerical linearization scheme based on a minimal or so-called relative coordinate set multibody formalism. This method results in smaller matrix operations and eliminates the need to find an independent set of coordinates for open-loop systems. For closed-loop systems, QR decomposition is used to identify the independent coordinates. When the effects of damping must be included, the system is transferred to first-order space after the linearized damping matrix is obtained. Linearization of steady motion systems with Cartesian accelerations is valid using this technique. Several open and

closed-loop examples are presented and compared with finite difference results and Fourier transforms of the nonlinear time responses.

# 2. DYNAMICS OF RIGID BODY SYSTEMS

This chapter presents a brief comparison of the numerous formulations that "automatically" assemble the equations of motion for rigid body systems and presents a more detailed description of the assembly formulation chosen for this research. The chapter begins with an overview of rigid body formulation methods based on an XYZ/Euler parameter Cartesian coordinate system. The kinematic constraint equations for a number of idealized joints are presented as well as the Cartesian Jacobian matrix entries associated with each joint type. The next section deals with the definition of the generalized coordinates related to specific joint types typically referred to as relative coordinates. Also, the position, velocity and acceleration transformations relating the relative coordinates to the Cartesian coordinates are presented. Finally, the equations of motion for general closed-loop systems are obtained using a Lagrangian approach. These equations are formulated in terms of generalized coordinates by starting with Cartesian coordinates and transforming to generalized coordinates using the relationships outlined in this chapter.

## 2.1 Background

Computer simulation of multibody mechanical systems began in the late 1960s and early 1970s on two apparently independent fronts. Multibody formalisms were

developed with the emergence of space technology [2, 3] and parallelled the efforts of researchers working in the machines and mechanisms areas [4, 5, 6].

Since the earliest multibody formalisms, much time and effort has been directed towards the development of accurate, user-friendly multibody system (MBS) programs. To date, the most succesful efforts reside in one of two categories. The first category is composed of symbolic formulations such as NEWEUL [7, 8], MESA VERDE [9] and SD/EXACT [10] while the second category uses numeric formulations such as DADS [11, 12], ADAMS [13] and MEDYNA [14, 15, 16]. For simple systems, symbolic formulations have two advantages over numeric formulations. Symbolic programs are more efficient since they assemble the equations of motion just once rather than at every time step as is required for the numerical programs. Also, the symbolic formulations allow a certain amount of insight since analytical equations are available. Unfortunately, these two advantages are negated when the system is moderately complex. Excessive storage and evaluation requirements for complex systems tend to shift the efficiency scales to the numerical formalisms. Also, the insight provided by the inspection of the analytical equations is lost when the equations become very complex.

## 2.2 Rigid Body Motion in Cartesian Coordinates

The location and orientation of a rigid body with respect to an inertial reference frame can be defined with three translational and three rotational coordinates. The rotational coordinates are commonly defined as three successive rotations about a body-fixed orthonormal axis. Depending on the order of rotations, a total of twelve conventions is possible. Two of the most common conventions are Euler and Bryant

angles. Although these orientation conventions are easily implemented, they result in singularity problems when one or more of the bodies experience large angular rotations [12, 17].

Another set of orientation parameters, called Euler parameters, has no singularity problems and is used extensively in rigid body dynamics [18]. In addition to the singularity advantage, numerical efficiency is increased with Euler parameters due to the fact that direction cosine matrices can be computed without trigonmetric evaluations. Euler parameters relate the orientation of one coordinate system with respect to another through an angle of revolution, $\phi$, about a unit vector, $\mathbf{u}$. The four parameters are defined as

$$\mathbf{p} = \begin{bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\phi}{2}\right) \\ u_1 \sin\left(\frac{\phi}{2}\right) \\ u_2 \sin\left(\frac{\phi}{2}\right) \\ u_3 \sin\left(\frac{\phi}{2}\right) \end{bmatrix} \tag{2.1}$$

Since only three coordinates are needed to define the orientation of a body, the Euler parameters are not independent and are related by the constraint [12]

$$\mathbf{p}^T\mathbf{p} = {e_0}^2 + {e_1}^2 + {e_2}^2 + {e_3}^2 = 1 \tag{2.2}$$

Using this set of Euler parameters, a 3x3 direction cosine matrix relating system $i$ to system $j$ can be computed [12]

$$\mathbf{A}_{ij} = 2\begin{bmatrix} {e_0}^2+{e_1}^2-\frac{1}{2} & e_1e_2-e_0e_3 & e_1e_3+e_0e_2 \\ e_1e_2+e_0e_3 & {e_0}^2+{e_2}^2-\frac{1}{2} & e_2e_3-e_0e_1 \\ e_1e_3-e_0e_2 & e_2e_3+e_0e_1 & {e_0}^2+{e_3}^2-\frac{1}{2} \end{bmatrix} \tag{2.3}$$

Using Euler parameters to define orientations, the Cartesian coordinate set for each body can be written as

$$\mathbf{x}_i = [\mathbf{r}_i{}^T, \mathbf{p}_i{}^T]^T \tag{2.4}$$

where the three position coordinates, $\mathbf{r}_i$, relate the center of gravity of the $i$th body to the inertial reference frame and the vector of Euler parameters, $\mathbf{p}_i$, defines the orientation of the $i$th body with respect to the inertial reference frame. Figure 2.1 depicts the orientation of a body with respect to the inertial reference frame and defines a position vector from the global origin to an arbitrary point on the body. The location of this point, $g$, defined in the inertial reference frame, is given by

$$\begin{aligned} \mathbf{r}_g &= \mathbf{r}_i + \mathbf{s}_{ig} \\ &= \mathbf{r}_i + \mathbf{A}_{i0}\mathbf{s}'_{ig} \end{aligned} \tag{2.5}$$

where $\mathbf{s}'_{ig}$ is the vector from the body-fixed coordinate system to the point $g$ defined in the coordinate system of body $i$ and $\mathbf{A}_{i0}$ is the transformation matrix relating system $i$ to the inertial reference frame.

For a point $g$ fixed in body $i$, the velocity of $g$ is obtained by differentiating Equation 2.5 with respect to time

$$\dot{\mathbf{r}}_g = \dot{\mathbf{r}}_i + \dot{\mathbf{A}}_{i0}\mathbf{s}'_{ig} \tag{2.6}$$

Using the relation [17]

$$\dot{\mathbf{A}}_{i0} = \tilde{\omega}_i \mathbf{A}_{i0} \tag{2.7}$$

Equation 2.6 can be written as

$$\mathbf{v}_g = \mathbf{v}_i + \tilde{\omega}_i \mathbf{A}_{i0}\mathbf{s}'_{ig} \tag{2.8}$$

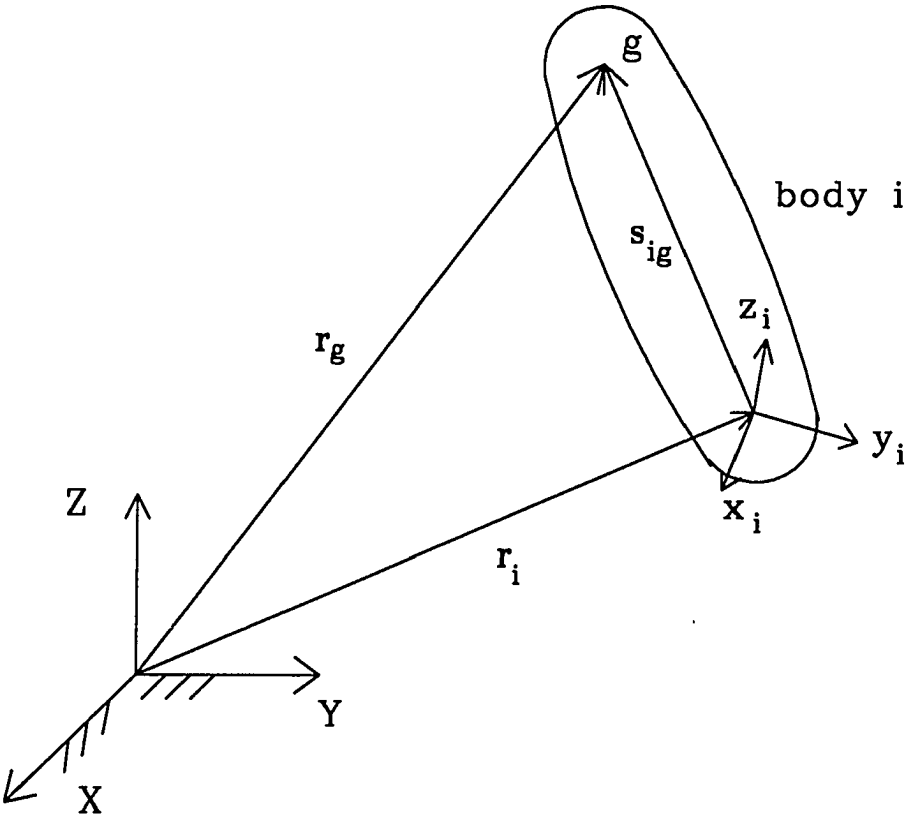



**Figure 2.1: Definition of Position Vectors**

where $\mathbf{v}_g$ is the velocity of point $g$ in the inertial reference frame, $\mathbf{v}_i$ is the translational velocity of the origin of the $i$th body-fixed coordinates and $\tilde{\omega}_i$ is defined as

$$\tilde{\omega}_i = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{2.9}$$

The terms $\omega_x, \omega_y$ and $\omega_z$ represent the angular velocity components of body $i$ with respect to the inertial reference frame.

The velocity of the $i$th body is represented by three translational and three rotational Cartesian velocity components

$$\dot{\mathbf{y}}_i = \left[\mathbf{v}_i{}^T \boldsymbol{\omega}_i{}^T\right]^T{}_{(6\times 1)} \tag{2.10}$$

where the relationships between $\boldsymbol{\omega}_i$ and the Euler parameters, $\mathbf{p}_i$, are given by [17]

$$\begin{aligned} \boldsymbol{\omega}_i &= 2\mathbf{E}_i\dot{\mathbf{p}}_i \\ \dot{\mathbf{p}}_i &= \tfrac{1}{2}\mathbf{E}_i{}^T\boldsymbol{\omega}_i \end{aligned} \tag{2.11}$$

Using the same components to define the accelerations, the Cartesian coordinates positions, velocities and accelerations can be summarized as

$$\begin{aligned} \mathbf{x}_i &= [\mathbf{r}_i{}^T, \mathbf{p}_i{}^T]^T{}_{(7\times 1)} \\ \dot{\mathbf{y}}_i &= \left[\mathbf{v}_i{}^T \boldsymbol{\omega}_i{}^T\right]^T{}_{(6\times 1)} \\ \ddot{\mathbf{y}}_i &= \left[\dot{\mathbf{v}}_i{}^T \dot{\boldsymbol{\omega}}_i{}^T\right]^T{}_{(6\times 1)} \end{aligned} \tag{2.12}$$

## 2.3 Definition of Generalized Coordinates

This thesis defines the generalized coordinate set to be the collection of all relative joint coordinates associated with kinematic joints excluding cut-joint coordinates.

Table 2.1: Definition of Relative Joint Coordinates

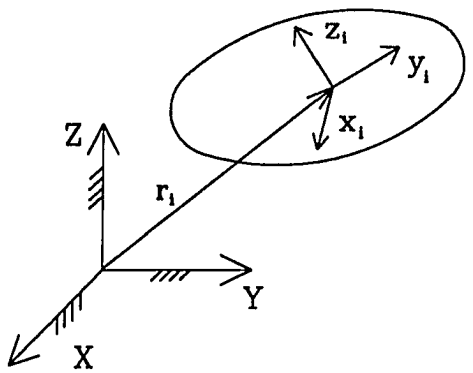
| Joint Type | Number of Generalized Coordinates | Generalized Coordinate Representation | Required Joint Axes |
|---|---|---|---|
| Floating Base Body | 7 | $\mathbf{r}_i, \mathbf{p}_i$ | none |
| Revolute Joint | 1 | $\phi_j$ | ${}^1\mathbf{u}_j$ |
| Translational Joint | 1 | $d_j$ | ${}^1\mathbf{u}_j$ |
| Cylindrical Joint | 2 | $\phi_j, d_j$ | ${}^1\mathbf{u}_j$ |
| Universal Joint | 2 | ${}^1\phi_j, {}^2\phi_j$ | ${}^1\mathbf{u}_i, {}^1\mathbf{u}_j$ |
| Spherical Joint | 3 | ${}^1\phi_j, {}^2\phi_j, {}^3\phi_j$ | ${}^1\mathbf{u}_j, {}^2\mathbf{u}_j, {}^3\mathbf{u}_j$ |

In the case of a floating base body, the seven Cartesian coordinates associated with that body are defined as generalized coordinates. Figure 2.2 with Table 2.1 defines the generalized coordinates associated with a floating base body as well as each of the joint types implemented in this formalism.

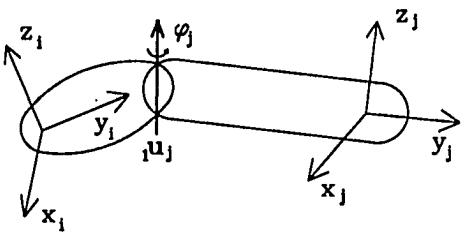
The degrees of freedom for a system can be computed based on the number of generalized coordinates, floating base bodies and constraint equations.

$$n_{df} = n_{gc} - n_{fb} - n_c \tag{2.13}$$

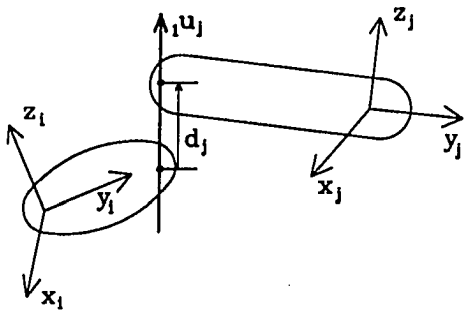
where $n_{gc}$ is the number of generalized coordinates, $n_{fb}$ is the number of floating base bodies and $n_c$ is the number of constraint equations. The constraint equations are composed of driving constraints, cut-joint constraints and Euler parameter constraints associated with floating base bodies. The constraint equations are presented in more detail in Section 2.5.



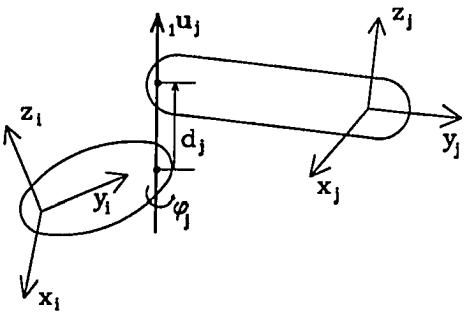

Floating Base Body



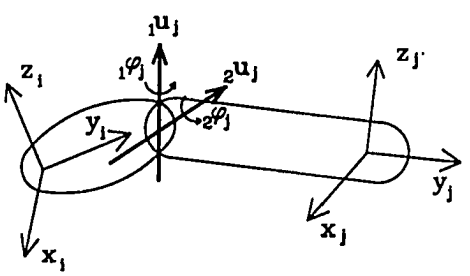

Revolute Joint



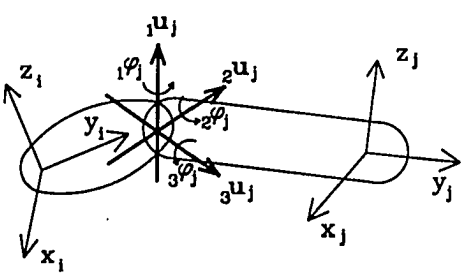

Translational Joint




Cylindrical Joint




Universal Joint




Spherical Joint

Figure 2.2: Definition of Relative Joint Coordinates

## 2.4 Cartesian/Generalized Coordinate Relationships

### 2.4.1 Position Transformation

Figure 2.3 presents three groups of vectors that are required later in this chapter for the formation of the velocity transformation matrix. The joint definition vectors associated with every joint in the system are represented by $\mathbf{u}_{i_x}'$, $\mathbf{u}_{i_y}'$ and $\mathbf{u}_{i_z}'$. These vectors are initially defined in the local coordinate systems of adjacent bodies and are transformed to the global reference frame as the bodies undergo rotational motion

$$\mathbf{u}_i = \mathbf{A}_{i0}\mathbf{u}_i' \tag{2.14}$$

Likewise, the local position vectors, $\mathbf{s}_{ij}'$, defined from the CG of body $i$ to the joint definition point of body $j$ are initially defined in the coordinate system of body $i$ and must also be transformed to the global coordinate system as the system rotates.

$$\mathbf{s}_{ij} = \mathbf{A}_{i0}\mathbf{s}_{ij}' \tag{2.15}$$

The third vector group required for the velocity transformation matrix is termed the distance vector group and denoted by $\mathbf{d}_{ij}$. This vector type is defined from the joint definition point of body $i$ to the CG of body $j$ and can be expressed as a combination of local position vectors and local distance vectors. For the example shown in Figure 2.3, the global distance vector defined from the joint definition point of body $i$ to the CG of body $k$ is given as

$$\mathbf{d}_{ik} = \mathbf{A}_{i0}\left(\mathbf{d}_{ii} + \mathbf{s}_{ij}\right) + \mathbf{A}_{j0}\left(\mathbf{d}_{jj} + \mathbf{s}_{jk}\right) + \mathbf{A}_{k0}\mathbf{d}_{kk} \tag{2.16}$$

and in general can be expressed as

$$\mathbf{d}_{ik} = \sum_{m=i}^{k-1} \mathbf{A}_{m0}\left(\mathbf{d}_{mm} + \mathbf{s}_{m,m+1}\right) + \mathbf{A}_{k0}\mathbf{d}_{kk} \tag{2.17}$$

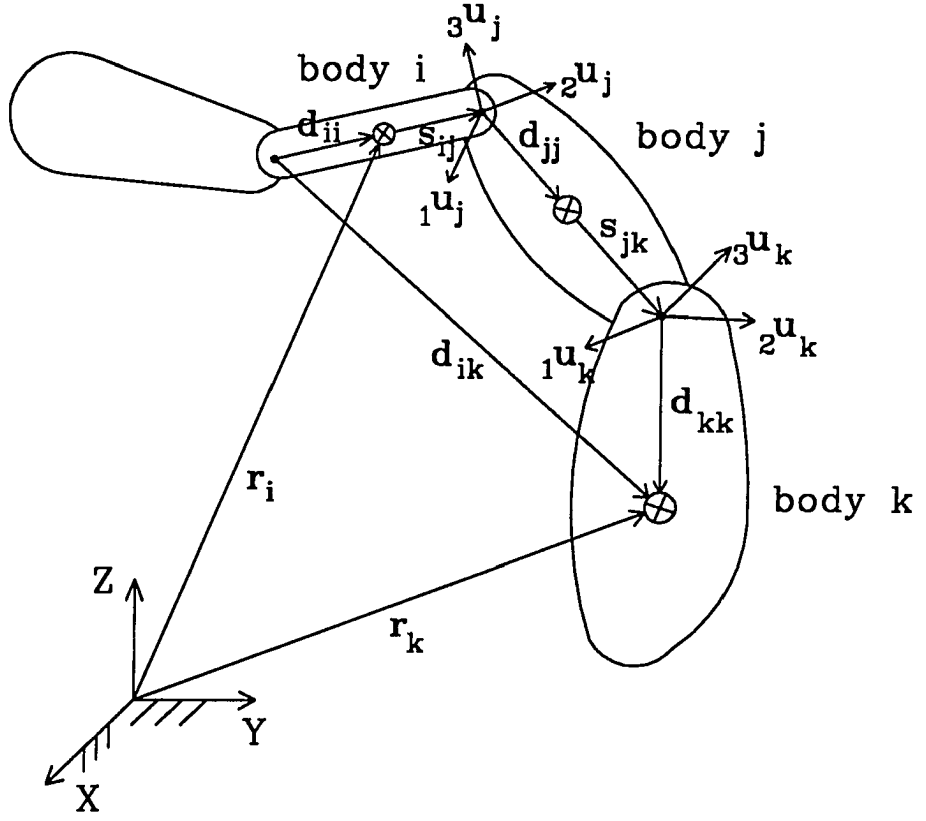



Figure 2.3: Definition of Joint Axes and Distance Vectors

defined from the CG of body $i$ to the joint definition point of body $j$ are initially defined in the coordinate system of body $i$ and must also be transformed to the global coordinate system as the system rotates.

$$\mathbf{s}_{ij} = \mathbf{A}_{i0}\mathbf{s}_{ij}' \tag{2.18}$$

These vectors require a transformation from local to global coordinates in the form of an A matrix. In addition, the Cartesian coordinates associated with each body are required for external force computations. Since the equations of motion for the system are written in terms of generalized coordinates and Cartesian coordinates are required to compute the external forces and the local-to-global transformation matrices, a position transformation in the form

$$\left[\mathbf{r}_i{}^T, \mathbf{p}_i{}^T\right]^T = \left[\mathbf{r}_i{}^T(\mathbf{q}), \mathbf{p}_i{}^T(\mathbf{q})\right]^T \tag{2.19}$$

is required. Although it is difficult to write the transformation equations explicitly in terms of generalized coordinates, it is straightforward to specify the Cartesian coordinates of one body when the Cartesian coordinates of its reference body and the relative coordinates between the bodies are known. Thus, the Cartesian coordinates for each body in a chain are computed recursively from the base body to the chain-end body along the outward path. Recursive computation of the Euler parameters is accomplished by introducing intermediate-axis Euler parameters computed with generalized coordinates and using quaternion operations for successive rotations [19]. For example, suppose the bodies $i$ and $j$ in Figure 2.3 are connected through a revolute joint. The Euler parameters for body $i$ are known and intermediate-axis

Euler parameters defining the relative motion of body $i$ to body $j$ can be defined as

$$\mathbf{p}_{ij} = \begin{bmatrix} cos\left(\frac{\phi_i}{2}\right) \\ {}_1u_i' sin\left(\frac{\phi_i}{2}\right) \\ {}_2u_i' sin\left(\frac{\phi_i}{2}\right) \\ {}_3u_i' sin\left(\frac{\phi_i}{2}\right) \end{bmatrix} \tag{2.20}$$

where $\mathbf{u}_i'$ is the generalized coordinate axis as defined in the coordinate system of body $i$ and $\phi_i$ is the generalized coordinate associated with the rotation between the two bodies. The Euler parameters for body $j$ can now be computed as

$$\mathbf{p}_j = \widehat{\mathbf{G}}_{ij}\mathbf{p}_i \tag{2.21}$$

where $\widehat{\mathbf{G}}_{ij}$ is a $(3 \times 4)$ matrix of intermediate Euler parameters defined as [18]

$$\widehat{\mathbf{G}} = \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} \tag{2.22}$$

Given the Euler parameters of each body, recursive computation of the body's CG is more straightforward. Using Equation 2.3 to compute the local-to-global transformation matrices, the recursive computation of the body's CG can be computed as

$$\mathbf{r}_j = \mathbf{r}_i + \mathbf{A}_{i0}\mathbf{s}_{ij}' + \mathbf{A}_{j0}\mathbf{d}_{jj}' \tag{2.23}$$

### 2.4.2 Velocity Transformation

The ultimate goal of this chapter is to derive a formalism that will assemble the equations of motion for any general multibody system. To accomplish this,

a velocity transformation which relates the Cartesian and generalized coordinate systems is required. Using a recursive algorithm similar to the one outlined for the position transformation, the Cartesian velocity vector defined in Equation 2.12 can be written as functions of the generalized coordinates and velocities

$$\dot{\mathbf{y}} = \dot{\mathbf{y}}(\mathbf{q}, \dot{\mathbf{q}}) \tag{2.24}$$

As with the position transformation, the angular information is needed first in order to solve for the translational velocities. From Figure 2.3, the angular velocity of body $j$ can be expressed as the sum of the angular velocity of its reference body and the relative angular rotation rate between the two bodies

$$\omega_j = \omega_i + (1 - \epsilon_i)\dot{q}_i \mathbf{u}_i \tag{2.25}$$

where

$$\epsilon_i = \begin{cases} 0 & \text{if joint } i \text{ is a revolute joint} \\ 1 & \text{if joint } i \text{ is a translational joint} \end{cases} \tag{2.26}$$

and $\mathbf{u}_i$ is the unit vector about which the $i$th generalized coordinate $q_i$ rotates. Moving outward through the chain, the angular velocity components of the $k$th body can be written as

$$\omega_k = \omega_i + (1 - \epsilon_i)\dot{q}_i \mathbf{u}_i + (1 - \epsilon_i)\dot{q}_j \mathbf{u}_j \tag{2.27}$$

Thus, a general expression for the angular velocity of a body can be written as

$$\omega_i = \sum_{j=1}^{n_b} \pi_{ij} \Omega_{ij} \tag{2.28}$$

where the path matrix component is defined as

$$\pi_{ij} = \begin{cases} 1 & \text{if body } j \text{ is between the base body and the } i\text{th body} \\ 0 & \text{otherwise} \end{cases} \tag{2.29}$$

The quantity $\boldsymbol{\Omega}_{ij}$ is defined as

$$\boldsymbol{\Omega}_{ij} = \begin{cases} \boldsymbol{\omega}_j & \text{if body } j \text{ is a floating base body} \\ \sum_k \dot{q}_{j_k} \mathbf{u}_{j_k} & \text{if body } j \text{ is attached with a revolute joint } (k=1) \\ \mathbf{0} & \text{if body } j \text{ is attached with a translational joint} \end{cases} \tag{2.30}$$

Equation 2.30 can easily be extended to include universal and spherical by setting $k$ equal to 2 or 3, respectively. The velocities associated with cylindrical joints can also be computed using a combination of revolute and translational joints.

Now that the angular velocity components for the bodies have been defined, the translational velocities can be computed using the same recursive process. Referring to Figure 2.3, the global position of body $j$ can be defined as

$$\mathbf{r}_j = \mathbf{r}_i + \epsilon_j q_j \mathbf{u}_j + \mathbf{s}_{ij} + \mathbf{d}_{jj} \tag{2.31}$$

The translational velocity of body $j$ can now be obtained by differentiating Equation 2.31 with respect to time

$$\mathbf{v}_j = \mathbf{v}_i + \epsilon_j \dot{q}_j \mathbf{u}_j + \tilde{\boldsymbol{\omega}}_j \epsilon_j q_j \mathbf{u}_j + \tilde{\boldsymbol{\omega}}_i \mathbf{s}_{ij} + \tilde{\boldsymbol{\omega}}_j \mathbf{d}_{jj} \tag{2.32}$$

The translational velocity of the gravity center of the $i$th body can then be expressed as

$$\mathbf{v}_i = \sum_{j=1}^{n_b} \pi_{ij} \left( \boldsymbol{\Omega}_{ij} \mathbf{d}_{ij} + \nu_{ij} \right) \tag{2.33}$$

where the term $\nu_{ij}$ accounts for floating base bodies and translational coordinates and is defined as

$$\nu_{ij} = \begin{cases} \mathbf{v}_j & \text{if body } j \text{ is a floating base body} \\ \mathbf{0} & \text{if body } j \text{ is connected with a revolute joint} \\ \dot{q}_j \mathbf{u}_j & \text{if body } j \text{ is connected with a translational joint} \end{cases} \tag{2.34}$$

Equations 2.33 and 2.28 define the Cartesian velocity coordinates in terms of generalized coordinates. From these two equations, it can be seen that each Cartesian velocity is simply a linear combination of the generalized velocities. Therefore, the individual Cartesian velocities expressed in Equations 2.33 and 2.28 can be represented in a more compact form as

$$\dot{\mathbf{y}}_i = \begin{bmatrix} \mathbf{v}_i \\ \omega_i \end{bmatrix} = \sum_{j=1}^{n_b} \mathbf{B}_{ij}\dot{\mathbf{q}}_j \tag{2.35}$$

where $\mathbf{B}_{ij}$ is a function of the generalized coordinates and system topology. These $\mathbf{B}_{ij}$ submatrices have a row length of 6 and and column length of 1 to 6 depending on the joint type associated with the submatrix (see Table 2.1). Table 2.2 presents the submatrices for several of the joint types used in this formalism.

Equation 2.35 can be generalized as

$$\dot{\mathbf{y}} = \begin{bmatrix} \mathbf{v}_1 \\ \omega_1 \\ \mathbf{v}_2 \\ \omega_2 \\ \vdots \\ \mathbf{v}_{n_b} \\ \omega_{n_b} \end{bmatrix} = \mathbf{B}(\mathbf{q})\dot{\mathbf{q}} \tag{2.36}$$

where the assembled $\mathbf{B}$ matrix is termed the velocity transformation matrix [20]. As will be shown in the next chapter, the velocity transformation matrix is used extensively in the computation of the linearized mass, stiffness and damping matrices for the linearization process.

Table 2.2: Block Entries for the Velocity Transformation Matrix

| Joint Type of $j$th Body | $\mathbf{B}_{ij}$ | Matrix Size |
|---|---|---|
| Floating Base Body | $\begin{bmatrix} \mathbf{I} & -\tilde{\mathbf{d}}_{ij} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$ | (6x6) |
| Revolute Joint | $\begin{bmatrix} \tilde{\mathbf{u}}_j \mathbf{d}_{ij} \\ \mathbf{u}_j \end{bmatrix}$ | (6x1) |
| Translational Joint | $\begin{bmatrix} \mathbf{u}_j \\ \mathbf{0} \end{bmatrix}$ | (6x1) |
| Cylindrical Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \mathbf{d}_{ij} & {}_2\mathbf{u}_j \\ {}_1\mathbf{u}_j & \mathbf{0} \end{bmatrix}$ | (6x2) |
| Universal Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \mathbf{d}_{ij} & {}_2\tilde{\mathbf{u}}_j \mathbf{d}_{ij} \\ {}_1\mathbf{u}_j & {}_2\mathbf{u}_j \end{bmatrix}$ | (6x2) |
| Spherical Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \mathbf{d}_{ij} & {}_2\tilde{\mathbf{u}}_j \mathbf{d}_{ij} & {}_3\tilde{\mathbf{u}}_j \mathbf{d}_{ij} \\ {}_1\mathbf{u}_j & {}_2\mathbf{u}_j & {}_3\mathbf{u}_j \end{bmatrix}$ | (6x3) |

### 2.4.3 Acceleration Transformation

A transformation relating the Cartesian coordinate accelerations to the generalized coordinate accelerations is required for the generation of the equations of motion. This relationship is obtained from the time derivative of Equation 2.36

$$\ddot{\mathbf{y}} = \mathbf{B}\ddot{\mathbf{q}} + \dot{\mathbf{B}}\dot{\mathbf{q}} \tag{2.37}$$

where $\dot{\mathbf{B}}$ is the time derivative of the velocity transformation matrix and $\ddot{\mathbf{q}}$ is the generalized acceleration vector. The $\dot{\mathbf{B}}$ matrix is a function of both the generalized position and velocity coordinates and has the same dimension and nonzero entries as the $\mathbf{B}$ matrix. Table 2.3 lists the $\dot{\mathbf{B}}_{ij}$ submatrices associated with the joint types used in this formalism.

## 2.5 Constraints

More often than not, the coordinates chosen to represent a mechanical system are not independent and are related by holonomic constraint equations of the form

$$\Phi_k(\mathbf{q}, t) = 0, \quad k = 1, 2, \ldots, n_C \tag{2.38}$$

where $n_C$ is the number of constraint equations. These constraints can be classified into two categories:

(i) joint constraints
(ii) general constraints

As the term implies, "joint constraints" are constraints imposed on adjacent bodies so that the relative movement of the two bodies is consistent with the type of joint connecting the bodies. The "general constraint" category includes all other constraints.

Table 2.3: Block Entries for the Time Derivative of the Velocity Transformation Matrix

| Joint Type of $j$th Body | $\dot{\mathbf{B}}_{ij}$ |
|---|---|
| Floating Base Body | $\begin{bmatrix} \mathbf{0} & -\widetilde{\dot{\mathbf{d}}_{ij}} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ |
| Revolute Joint | $\begin{bmatrix} \widetilde{\tilde{\omega}_j \mathbf{u}_j} \mathbf{d}_{ij} + \tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} \\ \tilde{\omega}_j \mathbf{u}_j \end{bmatrix}$ |
| Translational Joint | $\begin{bmatrix} \tilde{\omega}_j \mathbf{u}_j \\ \mathbf{0} \end{bmatrix}$ |
| Cylindrical Joint | $\begin{bmatrix} \widetilde{\tilde{\omega}_j {}_1\mathbf{u}_j} \mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} & \tilde{\omega}_j {}_2\mathbf{u}_j \\ \tilde{\omega}_j {}_1\mathbf{u}_j & \mathbf{0} \end{bmatrix}$ |
| Universal Joint | $\begin{bmatrix} \widetilde{\tilde{\omega}_j {}_1\mathbf{u}_j} \mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} & \widetilde{\tilde{\omega}_j {}_2\mathbf{u}_j} \mathbf{d}_{ij} + {}_2\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} \\ \tilde{\omega}_j {}_1\mathbf{u}_j & \tilde{\omega}_j {}_2\mathbf{u}_j \end{bmatrix}$ |
| Spherical Joint | $\begin{bmatrix} \widetilde{\tilde{\omega}_j {}_1\mathbf{u}_j} \mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} & \widetilde{\tilde{\omega}_j {}_2\mathbf{u}_j} \mathbf{d}_{ij} + {}_2\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} & \widetilde{\tilde{\omega}_j {}_3\mathbf{u}_j} \mathbf{d}_{ij} + {}_3\tilde{\mathbf{u}}_j \dot{\mathbf{d}}_{ij} \\ \tilde{\omega}_j {}_1\mathbf{u}_j & \tilde{\omega}_j {}_2\mathbf{u}_j & \tilde{\omega}_j {}_3\mathbf{u}_j \end{bmatrix}$ |

Examples include translational or rotational kinematic drivers and temporary position constraints used during the initial assembly phase.

### 2.5.1 Joint Constraints

The number of joint constraints generated by a multibody simulation package is dependent upon the type of joints modelled and the coordinate representation used by the particular package. For example, formalisms using Cartesian coordinates must generate constraint equations for every joint in the system [11, 13] whereas relative joint formalisms generate constraint equations only for closed-loop systems [7, 9, 14, 20].

The formalism used in this thesis uses a minimal set of generalized coordinates for open-loop tree structure systems [20]. For these open-loop systems, the generalized coordinates are independent and no constraint equations are generated. On the other hand, closed-loop systems are handled by "cutting" a joint in each loop and generating the equations of motion as if the system was an open-loop tree structure. In this case, algebraic constraint equations limiting the relative cut-joint motion must be solved together with the equations of motion of the open-loop system. The number of constraint equations generated is dependent upon the number of degrees of freedom associated with the cut joint. A cut revolute joint generates five constraint equations whereas a spherical joint generates only three.

Figure 2.4 gives the notation used to define constraint equations between two bodies where $\mathbf{d}$ is the vector from the joint definition point on body $i$, $\mathbf{p}_i$, to the joint definition point on body $j$, $\mathbf{p}_j$. This vector is nonexistent for revolute, universal and spherical joints. $\mathbf{u}_i$ and $\mathbf{u}_j$ are unit vector triads that represent the joint axes

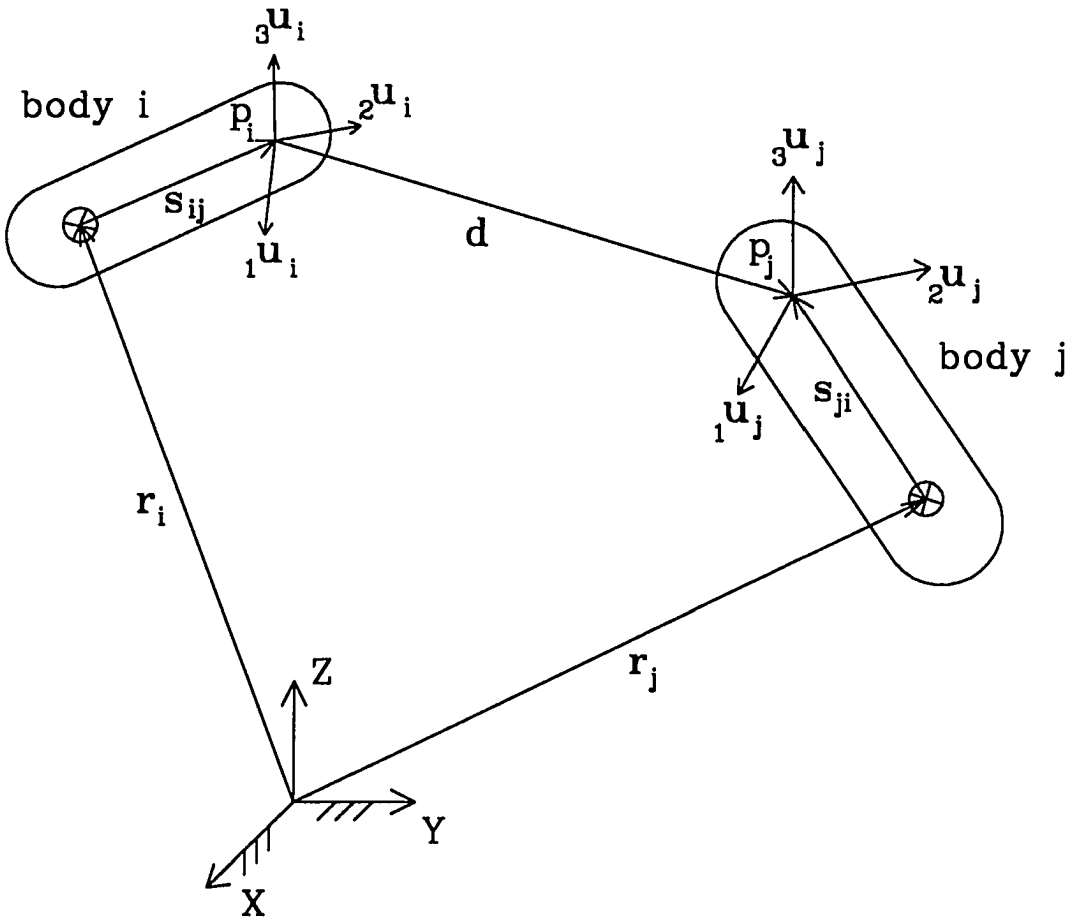



**Figure 2.4: Constraint Equation Notation**

on bodies $i$ and $j$, respectively. All joint constraint equations can be derived from combinations of the following conditions:

(i) $\mathbf{p}_i$ and $\mathbf{p}_j$ coincide
(ii) two vectors imbedded in each body are perpendicular
(iii) two vectors imbedded in each body are parallel

Table 2.4 lists the constraint equations associated with the cut joints used in this thesis.

### 2.5.2 General Constraints

This category encompasses all constraints not included in the joint constraint section. Constraints include drivers on relative coordinates, drivers on Cartesian coordinates, distance constraints and temporary position constraints used for intial assembly. Table 2.5 lists a few of the possibilities.

### 2.5.3 Constraint Jacobian

The constraint Jacobian matrix, $\boldsymbol{\Phi}_{\mathbf{q}}$, is needed to generate the equations of motion for a general multibody system. Because the majority of the constraint equations given in Tables 2.4 and 2.5 are defined using Cartesian coordinates, partial differentiation of the constraint equations with respect to the generalized coordinates is difficult. Kim and Vanderploeg [20] derived a method of computing $\boldsymbol{\Phi}_{\mathbf{q}}$ based on the constraint Jacobian as defined in Cartesian coordinates and the system topology

$$\boldsymbol{\Phi}_{\mathbf{q}} = \boldsymbol{\Phi}_{\mathbf{x}_{ij}} \mathbf{D}_{ij} \mathbf{B}_{ij} \tag{2.39}$$

Table 2.4: Joint Constraints

| Joint Type | Condition | Constraint Equations | Number of Constraint Equations |
|---|---|---|---|
| Spherical | $\mathbf{p}_i = \mathbf{p}_j$ | $\mathbf{r}_i + \mathbf{s}_{ij} - \mathbf{r}_j - \mathbf{s}_{ji} = \mathbf{0}$ | 3 |
| Universal | $\mathbf{p}_i = \mathbf{p}_j$ | $\mathbf{r}_i + \mathbf{s}_{ij} - \mathbf{r}_j - \mathbf{s}_{ji} = \mathbf{0}$ | 3 |
| | $\mathbf{u}_{i_x} \perp \mathbf{u}_{j_x}$ | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_x} = 0$ | 1 |
| Revolute | $\mathbf{p}_i = \mathbf{p}_j$ | $\mathbf{r}_i + \mathbf{s}_{ij} - \mathbf{r}_j - \mathbf{s}_{ji} = \mathbf{0}$ | 3 |
| | $\mathbf{u}_{i_x} \parallel \mathbf{u}_{j_x}$ | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_y} = 0$ | 1 |
| | | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_z} = 0$ | 1 |
| Cylindrical | $\mathbf{u}_{i_x} \parallel \mathbf{u}_{j_x}$ | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_y} = 0$ | 1 |
| | | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_z} = 0$ | 1 |
| | $\mathbf{d} \parallel \mathbf{u}_{j_x}$ | $\mathbf{d}^T \mathbf{u}_{j_y} = 0$ | 1 |
| | | $\mathbf{d}^T \mathbf{u}_{j_z} = 0$ | 1 |
| Translational | $\mathbf{u}_{i_x} \parallel \mathbf{u}_{j_x}$ | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_y} = 0$ | 1 |
| | | $\mathbf{u}_{i_x}^T \mathbf{u}_{j_z} = 0$ | 1 |
| | $\mathbf{d} \parallel \mathbf{u}_{j_x}$ | $\mathbf{d}^T \mathbf{u}_{j_y} = 0$ | 1 |
| | | $\mathbf{d}^T \mathbf{u}_{j_z} = 0$ | 1 |
| | $\mathbf{u}_{i_y} \parallel \mathbf{u}_{j_x}$ | $\mathbf{u}_{i_y}^T \mathbf{u}_{j_x} = 0$ | 1 |

Table 2.5: General Constraints

| Constraint Type | Constraint Equations | Number of Constraint Equations |
|---|---|---|
| Joint Driver | $q_i - q(t) = 0$ | 1 |
| Point Location | $\mathbf{r}_i + \mathbf{s}_i - \mathbf{r}(t) = \mathbf{0}$ | 3 |
| Orientation | $\mathbf{p}_i - \mathbf{p}(t) = \mathbf{0}$ | 4 |
| Relative Orientation | $\mathbf{p}_i - \widehat{\mathbf{E}}_{ij}{}^T(t)\mathbf{p}_j = \mathbf{0}$ | 4 |
| Distance | $\parallel \mathbf{r}_i + \mathbf{s}_i - \mathbf{r}_j - \mathbf{s}_j \parallel - \mathbf{d}(t) = 0$ | 1 |

where $\Phi_{\mathbf{x}_{ij}}$ is the constraint Jacobian submatrix entries defined in Cartesian coordinates, $\mathbf{D}_{ij}$ is defined as

$$\mathbf{D}_{ij} = \left[\begin{array}{cc|cc} \mathbf{I}_3 & \mathbf{0} & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{0}$}} \\ \mathbf{0} & \frac{1}{2}\mathbf{E}_i{}^T & & \\ \hline \multicolumn{2}{c|}{\multirow{2}{*}{$\mathbf{0}$}} & \mathbf{I}_3 & \mathbf{0} \\ & & \mathbf{0} & \frac{1}{2}\mathbf{E}_j{}^T \end{array}\right] \tag{2.40}$$

and $\mathbf{B}_{ij}$ is the block matrix entries for the velocity transformation matrix. A modified Cartesian Jacobian matrix is then defined as the product of the Cartesian Jacobian matrix and $\mathbf{D}_{ij}$.

$$\mathbf{J}_{ij} = \Phi_{x_{ij}}\mathbf{D}_{ij} \tag{2.41}$$

Table 2.6 lists the modified Cartesian Jacobian submatrices for the basic cut-joint constraints listed in Table 2.4.

Table 2.6: Modified Cartesian Jacobian Matrix for Cut-Joint Constraints

| Cut-Joint Type | Modified Cartesian Jacobian Matrix ($\mathbf{J}_{ij}$) |
|---|---|
| Spherical | $\begin{bmatrix} \mathbf{I}_3 & -\tilde{\mathbf{s}}_i & -\mathbf{I}_3 & \tilde{\mathbf{s}}_j \end{bmatrix}_{(3\times12)}$ |
| Universal | $\begin{bmatrix} \mathbf{I}_3 & -\tilde{\mathbf{s}}_i & -\mathbf{I}_3 & \tilde{\mathbf{s}}_j \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_i\mathbf{u}_j)^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_i\mathbf{u}_j)^T \end{bmatrix}_{(4\times12)}$ |
| Revolute | $\begin{bmatrix} \mathbf{I}_3 & -\tilde{\mathbf{s}}_i & -\mathbf{I}_3 & \tilde{\mathbf{s}}_j \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_y})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_y})^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T \end{bmatrix}_{(5\times12)}$ |
| Cylindrical | $\begin{bmatrix} \mathbf{u}_{j_y}^T & (\tilde{\mathbf{s}}_i\mathbf{u}_{j_y})^T & -\mathbf{u}_{j_y}^T & -\left\{(\mathbf{r}_i+\widetilde{\mathbf{s}_i}-\mathbf{r}_j)\mathbf{u}_{j_y}\right\}^T \\ \mathbf{u}_{j_z}^T & (\tilde{\mathbf{s}}_i\mathbf{u}_{j_z})^T & -\mathbf{u}_{j_z}^T & -\left\{(\mathbf{r}_i+\widetilde{\mathbf{s}_i}-\mathbf{r}_j)\mathbf{u}_{j_z}\right\}^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_y})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_y})^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T \end{bmatrix}_{(4\times12)}$ |
| Translational | $\begin{bmatrix} \mathbf{u}_{j_y}^T & (\tilde{\mathbf{s}}_i\mathbf{u}_{j_y})^T & -\mathbf{u}_{j_y}^T & -\left\{(\mathbf{r}_i+\widetilde{\mathbf{s}_i}-\mathbf{r}_j)\mathbf{u}_{j_y}\right\}^T \\ \mathbf{u}_{j_z}^T & (\tilde{\mathbf{s}}_i\mathbf{u}_{j_z})^T & -\mathbf{u}_{j_z}^T & -\left\{(\mathbf{r}_i+\widetilde{\mathbf{s}_i}-\mathbf{r}_j)\mathbf{u}_{j_z}\right\}^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{i_y})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_y})^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_x}\mathbf{u}_{j_z})^T \\ \mathbf{0}^T & (\tilde{\mathbf{u}}_{i_y}\mathbf{u}_{j_z})^T & \mathbf{0}^T & -(\tilde{\mathbf{u}}_{i_y}\mathbf{u}_{j_z})^T \end{bmatrix}_{(5\times12)}$ |

## 2.6 Equations of Motion

Lagrange's equations of motion for an $n$ degree of freedom system are given as [21]

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} - g_j = 0, \quad j = 1, 2, ..., n_{gc} \tag{2.42}$$

where $T$ is the total kinetic energy of the system, $\mathbf{q}$ is the vector of independent generalized coordinates and $g_j$ is the generalized force corresponding to the $j$th generalized coordinate.

The total kinetic energy of a system of rigid bodies is the sum of the kinetic energy associated with each body

$$\mathbf{T} = \sum_{i=1}^{n_b} \left[\frac{1}{2}\mathbf{v}_i{}^T \mathbf{M}_i \mathbf{v}_i + \frac{1}{2}\boldsymbol{\omega}'_i{}^T \mathbf{J}_i' \boldsymbol{\omega}'_i\right] \tag{2.43}$$

where $\mathbf{v}_i$ is the global translational velocity vector of the mass center of body $i$, $\boldsymbol{\omega}'_i$ is the local angular velocity vector of body $i$, $\mathbf{M}_i$ is the 3x3 diagonal mass matrix associated with body $i$, $\mathbf{J}_i'$ is the 3x3 central inertia tensor of body $i$ with respect to its body-fixed coordinate system and $N_b$ is the number of rigid bodies in the system. The generalized forces from Equation 2.42 can be expressed as linear combinations of the Cartesian force vector [20]

$$\mathbf{g} = \mathbf{B}^T \mathbf{f} \tag{2.44}$$

Applying Equations 2.44 and 2.43 to Equation 2.42, the final form of the equations of motion in terms of the relative joint coordinates can be computed as [20]

$$\left(\mathbf{B}^T \mathbf{M} \mathbf{B}\right) \ddot{\mathbf{q}} + \mathbf{B}^T \left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}} + \mathbf{h} - \mathbf{f}\right) = \mathbf{0} \tag{2.45}$$

where $\mathbf{B}$ is the velocity transformation matrix defined in Table 2.3 and $\mathbf{M}$ is the

Cartesian mass matrix defined as

$$\mathbf{M} = \left[\begin{array}{c|c|c|c} \begin{matrix}\mathbf{M}_1 & \\ & \mathbf{J}_1\end{matrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \begin{matrix}\mathbf{M}_2 & \\ & \mathbf{J}_2\end{matrix} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \begin{matrix}\mathbf{M}_{n_b} & \\ & \mathbf{J}_{n_b}\end{matrix} \end{array}\right]_{(6n_b \times 6n_b)} \tag{2.46}$$

The term h is the Cartesian Coriolis and centrifugal force vector

$$\mathbf{h} = \left[\begin{array}{cccc} \mathbf{h}_1{}^T & \mathbf{h}_2{}^T & \cdots & \mathbf{h}_{n_b}{}^T \end{array}\right]^T_{(6n_b \times 1)} \tag{2.47}$$

where

$$\mathbf{h}_i = \left[\begin{array}{cc} 0^T & (\tilde{\omega}_i \mathbf{J}_i \omega_i)^T \end{array}\right]^T_{(6\times 1)}$$

and f is defined as the external Cartesian force vector

$$\mathbf{f} = \left[\begin{array}{cccc} \mathbf{f}_1{}^T & \mathbf{f}_2{}^T & \cdots & \mathbf{f}_{n_b}{}^T \end{array}\right]^T_{(6n_b \times 1)} \tag{2.48}$$

where

$$\mathbf{f}_i = \left[\begin{array}{cc} \mathbf{r}_i{}^T & \tau_i{}^T \end{array}\right]^T_{(6\times 1)}$$

$\mathbf{r}_i$ is the external force vector and $\tau_i$ is the external torque vector applied to body $i$ at the CG.

Equation 2.45 can be extended to include closed-loops by introducing constraint equations. Assume there are $n_c$ independent constraint equations of the form

$$\Phi_i(\mathbf{q}, t) = 0 \qquad i = 1, \cdots, n_c \tag{2.49}$$

then equation 2.45 is given as [20]

$$\left(\mathbf{B}^T\mathbf{M}\mathbf{B}\right)\ddot{\mathbf{q}} + \mathbf{B}^T\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}} + \mathbf{h} - \mathbf{f}\right) + \boldsymbol{\Phi}_{\mathbf{q}}{}^T\boldsymbol{\lambda} = \mathbf{0} \tag{2.50}$$

where $\boldsymbol{\Phi}_{\mathbf{q}}{}^T$ is an $n_{df} \times n_c$ constraint Jacobian matrix defined as

$$\boldsymbol{\Phi}_{\mathbf{q}_{ij}} = \frac{\partial \boldsymbol{\Phi}_i}{\partial q_j} \tag{2.51}$$

and $\boldsymbol{\lambda}$ is an $n_c$ vector of Lagrange multipliers [20]. For these types of constrained systems, the coupled differential and algebraic equations must be solved simultaneously. The matrix form of this constrained system is obtained by combining the second time derivatives of the constraint equations with Equation 2.50

$$\begin{bmatrix} \bar{\mathbf{M}} & \boldsymbol{\Phi}_{\mathbf{q}}{}^T \\ \boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{g}} \\ \mathbf{a} \end{bmatrix} \tag{2.52}$$

where $\bar{\mathbf{M}}$ is the generalized mass matrix

$$\bar{\mathbf{M}} = \mathbf{B}^T\mathbf{M}\mathbf{B} \tag{2.53}$$

$\bar{\mathbf{g}}$ is a modified generalized force vector

$$\bar{\mathbf{g}} = \mathbf{B}^T\left(\mathbf{f} - \mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}} - \mathbf{h}\right) \tag{2.54}$$

and $\mathbf{a}$ is the right-hand side of the constraint acceleration equations

$$\mathbf{a} = -\dot{\boldsymbol{\Phi}}_{\mathbf{q}}\dot{\mathbf{q}} - \dot{\boldsymbol{\Phi}}_t \tag{2.55}$$

For independent constraints, the acceleration coefficient matrix is nonsingular and the generalized accelerations, $\ddot{\mathbf{q}}$, and Lagrange multipliers, $\boldsymbol{\lambda}$, can be computed from Equation 2.52.

# 3. LINEARIZATION OF OPEN-CHAIN SYSTEMS

For an $n$ degree-of-freedom, open-chain system, the coupled algebraic constraint equations and differential equations of motion can be reduced to a system of $n$ differential equations using Equation 3.1 and the velocity transformation matrix, $\mathbf{B}$.

$$\mathbf{B}^T\mathbf{M}\mathbf{B}\ddot{\mathbf{q}} + \mathbf{B}^T\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}} + \mathbf{h} - \mathbf{f}\right) = \mathbf{0} \tag{3.1}$$

where

$$\begin{aligned} \mathbf{B} &= \mathbf{B}(\mathbf{q}) \\ \dot{\mathbf{B}} &= \dot{\mathbf{B}}(\mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{h} &= \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{M} &= \mathbf{M}(\mathbf{q}) \end{aligned}$$

This system of equations can be generalized as a function of positions, velocities, accelerations and external forces acting on the system

$$\mathbf{g}\left(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}, \mathbf{f}\right) = \mathbf{0} \tag{3.2}$$

The first variation of this set of functions, $\mathbf{g}$, is

$$\delta\mathbf{g} = \frac{\partial \mathbf{g}}{\partial \ddot{\mathbf{q}}}\delta\ddot{\mathbf{q}} + \frac{\partial \mathbf{g}}{\partial \dot{\mathbf{q}}}\delta\dot{\mathbf{q}} + \frac{\partial \mathbf{g}}{\partial \mathbf{q}}\delta\mathbf{q} + \frac{\partial \mathbf{g}}{\partial \mathbf{f}}\delta\mathbf{f} \tag{3.3}$$

where the variational operator, $\delta$, denotes a small change in the given quantity. For a homogeneous system, $\delta\mathbf{f} = \mathbf{0}$. Furthermore, if the system is evaluated at an operating

point, $\mathbf{q}^* = (\ddot{\mathbf{q}}_0, \dot{\mathbf{q}}_0, \mathbf{q}_0)$, Equation 3.3 can be expressed as

$$\delta \mathbf{g} = \left.\frac{\partial \mathbf{g}}{\partial \ddot{\mathbf{q}}}\right|_{\mathbf{q}^*} \delta\ddot{\mathbf{q}} + \left.\frac{\partial \mathbf{g}}{\partial \dot{\mathbf{q}}}\right|_{\mathbf{q}^*} \delta\dot{\mathbf{q}} + \left.\frac{\partial \mathbf{g}}{\partial \mathbf{q}}\right|_{\mathbf{q}^*} \delta\mathbf{q} = \mathbf{0} \tag{3.4}$$

Define

$$\begin{aligned} \mathbf{M}_l &\equiv \left.\frac{\partial \mathbf{g}}{\partial \ddot{\mathbf{q}}}\right|_{\mathbf{q}^*} & \ddot{\hat{\mathbf{q}}} &\equiv \delta\ddot{\mathbf{q}} \\ \mathbf{C}_l &\equiv \left.\frac{\partial \mathbf{g}}{\partial \dot{\mathbf{q}}}\right|_{\mathbf{q}^*} & \dot{\hat{\mathbf{q}}} &\equiv \delta\dot{\mathbf{q}} \\ \mathbf{K}_l &\equiv \left.\frac{\partial \mathbf{g}}{\partial \mathbf{q}}\right|_{\mathbf{q}^*} & \hat{\mathbf{q}} &\equiv \delta\mathbf{q} \end{aligned} \tag{3.5}$$

Equation 3.4 can then be represented by

$$\mathbf{M}_l \ddot{\hat{\mathbf{q}}} + \mathbf{C}_l \dot{\hat{\mathbf{q}}} + \mathbf{K}_l \hat{\mathbf{q}} = \mathbf{0} \tag{3.6}$$

where $\mathbf{M}_l, \mathbf{C}_l$ and $\mathbf{K}_l$ are defined as the linearized mass, damping and stiffness matrices, and $\ddot{\hat{\mathbf{q}}}, \dot{\hat{\mathbf{q}}}$ and $\hat{\mathbf{q}}$ represent the deviations of the generalized accelerations, velocities and positions, respectively, from the operating point, $\mathbf{q}^*$. The goal of this chapter is to develop a method which can be used to accurately compute the linearized mass, damping and stiffness matrices for any general open-loop, rigid-body system.

## 3.1 Finite Difference Method

One possible solution to this problem is the finite difference method wherein the system states, $\ddot{\mathbf{q}}, \dot{\mathbf{q}}$ and $\mathbf{q}$ are systematically dithered so as to produce the linearized mass, damping and stiffness matrices. For example, the upper-left entry in the stiffness matrix, $\mathbf{K}_l$, can be approximated as

$$k_{1,1} = \frac{g_1(\mathbf{q}^* + \delta q_1) - g_1(\mathbf{q}^*)}{\delta q_1} \tag{3.7}$$

where $g_1$ is the first system equation and $\delta q_1$ represents the dithering value for the first entry in the generalized coordinate position array. At first glance, this approach seems straightforward and easily implemented. Unfortunately, several problems arise under certain circumstances. One difficulty in implementing a finite difference approach arrises when attempting to systematically compute the amount of dithering to introduce to each state variable. For some variables, a dithering value of 2000 (mm) might result in an accurate entry in a linearized matrix whereas another variable might require a value of 0.00001 (radians). A partial solution to this problem is to estimate an initial dithering value based on a small percentage of the variable's absolute value plus an additional small value (this additional small number is needed for those variables that are initially zero or very close to zero). Using this estimate for the dithering value, the entry in the linearized matrix is computed. To determine if this entry is accurate, the dithering value is then reduced by some percentage and the entry recomputed. If the new entry is within a certain percentage of the previous value (e.g., one percent), then the estimated matrix entry is probably "close enough" to the actual value.

Although this method produces accurate system matrices for many types of applications, it is not general enough to work on many complicated systems. The examples discussed in this chapter compare the finite difference results with the method derived in this thesis.

## 3.2 Analytical Derivation

A second method, developed in this thesis, is to analytically derive the terms involved in each of the linearized matrices for each type of joint described in Chap-

ter 3. These terms are then assembled as needed to compute the linearized mass, damping and stiffness matrices.

### 3.2.1 Linearized Mass Matrix

Computation of the linearized mass matrix is straightforward. From Equation 3.5, the definition of the linearized mass matrix is

$$\mathbf{M}_l = \left.\frac{\partial \mathbf{g}}{\partial \ddot{\mathbf{q}}}\right|_{\mathbf{q}^*} = \left.\mathbf{B}^T\mathbf{M}\mathbf{B}\right|_{\mathbf{q}^*} \tag{3.8}$$

Both **B** and **M** are computed each time step in the dynamic analysis portion of the software, and therefore, are readily available to the linearization process developed in this thesis.

### 3.2.2 Linearized Stiffness Matrix

For an equilibrium configuration with zero generalized accelerations, the stiffness matrix can be computed from Equations 3.1 and 3.5

$$\begin{aligned} \mathbf{K}_l &= \frac{\partial\left[\mathbf{B}^T\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{h}-\mathbf{f}\right)\right]}{\partial \mathbf{q}} \\ \mathbf{K}_l &= \frac{\partial \mathbf{B}^T}{\partial \mathbf{q}}\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{h}-\mathbf{f}\right)+\mathbf{B}^T\left(\frac{\partial \mathbf{M}}{\partial \mathbf{q}}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{M}\frac{\partial \dot{\mathbf{B}}}{\partial \mathbf{q}}\dot{\mathbf{q}}+\frac{\partial \mathbf{h}}{\partial \mathbf{q}}-\frac{\partial \mathbf{f}}{\partial \mathbf{q}}\right) \end{aligned} \tag{3.9}$$

Furthermore, if the generalized velocities for a given equilibrium configuration are zero, Equation 3.9 can be simplified by eliminating the terms multiplied by $\dot{\mathbf{q}}$. Also, since the vector of centrifugal forces, **h**, is a function of Cartesian angular velocities and global inertia terms (see Equation 2.47), an equilibrium configuration with zero generalized velocities eliminates the term associated with **h**. Therefore, for a time invariant system at an equilibrium point with zero accelerations and velocities, a

simplified equation for the stiffness matrix becomes

$$\mathbf{K}_l = -\left(\frac{\partial \mathbf{B}^T}{\partial \mathbf{q}}\mathbf{f} + \mathbf{B}^T\frac{\partial \mathbf{f}}{\partial \mathbf{q}}\right) \tag{3.10}$$

To evaluate the linearized stiffness matrix, expressions for $\frac{\partial \mathbf{B}}{\partial \mathbf{q}}$ and $\frac{\partial \mathbf{f}}{\partial \mathbf{q}}$ must be derived. The remainder of the terms in Equation 3.10 is available from the dynamic analysis portion of the code.

Several operations needed to compute the linearized terms are used many times. A few of these operations will now be generalized starting with the operations dealing with the velocity transformation matrix.

**3.2.2.1 Partial Differentiation of the Velocity Transformation Matrix with Respect to the Generalized Coordinates** Table 2.2 contains the block matrix descriptions of the velocity transformation matrix, **B**, for several joint types.

Note that each joint type is composed of some combination of joint axes, $\mathbf{u}_j$, and distance vectors, $\mathbf{d}_{ij}$. Also note that all joint types are combinations of revolute and translational degrees of freedom. For instance, the cylindrical joint is composed of a revolute and a translational joint and the spherical joint is made up of three orthogonal revolute joints. Even the degrees of freedom associated with a floating base body can be thought of as three translational and three rotational degrees of freedom. Thus, a major task needed for deriving the linearized system matrices reduces to computing the partial of the joint axes and distance vectors with respect to a rotation or translation and assembling these gradients in the proper order.

To compute the partials with respect to a rotation angle, the relationship between a $3 \times 3$ rotation matrix and the rotation angle about the axis of rotation must

be obtained. Equation 2.3 presents the 3x3 rotation matrix in terms of Euler parameters. Substituting the definition of the Euler parameters from Equation 2.1 into Equation 2.3 yields

$$\mathbf{A}_{ij} = 2\begin{bmatrix} \cos^2\frac{\phi}{2}+u_1^2\sin^2\frac{\phi}{2}-\frac{1}{2} & u_1u_2\sin^2\frac{\phi}{2}-u_3\sin\frac{\phi}{2}\cos\frac{\phi}{2} & u_1u_3\sin^2\frac{\phi}{2}+u_2\sin\frac{\phi}{2}\cos\frac{\phi}{2} \\ u_1u_2\sin^2\frac{\phi}{2}+u_3\sin\frac{\phi}{2}\cos\frac{\phi}{2} & \cos^2\frac{\phi}{2}+u_2^2\sin^2\frac{\phi}{2}-\frac{1}{2} & u_2u_3\sin^2\frac{\phi}{2}-u_1\sin\frac{\phi}{2}\cos\frac{\phi}{2} \\ u_1u_3\sin^2\frac{\phi}{2}-u_2\sin\frac{\phi}{2}\cos\frac{\phi}{2} & u_2u_3\sin^2\frac{\phi}{2}+u_1\sin\frac{\phi}{2}\cos\frac{\phi}{2} & \cos^2\frac{\phi}{2}+u_3^2\sin^2\frac{\phi}{2}-\frac{1}{2} \end{bmatrix} \tag{3.11}$$

The partial of this transformation matrix with respect to the rotation angle, $\phi$, becomes

$$\begin{aligned}
\frac{\partial a_{1,1}}{\partial\phi} &= 2(u_1^2\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}) \\
\frac{\partial a_{1,2}}{\partial\phi} &= 2u_1u_2\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-u_3\cos^2\tfrac{\phi}{2}+u_3\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{1,3}}{\partial\phi} &= 2u_1u_3\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}+u_2\cos^2\tfrac{\phi}{2}-u_2\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{2,1}}{\partial\phi} &= 2u_1u_2\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}+u_3\cos^2\tfrac{\phi}{2}-u_3\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{2,2}}{\partial\phi} &= 2(u_2^2\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}) \\
\frac{\partial a_{2,3}}{\partial\phi} &= 2u_2u_3\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-u_1\cos^2\tfrac{\phi}{2}+u_1\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{3,1}}{\partial\phi} &= 2u_1u_3\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-u_2\cos^2\tfrac{\phi}{2}+u_2\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{3,2}}{\partial\phi} &= 2u_2u_3\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}+u_1\cos^2\tfrac{\phi}{2}-u_1\sin^2\tfrac{\phi}{2} \\
\frac{\partial a_{3,3}}{\partial\phi} &= 2(u_3^2\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2}-\cos\tfrac{\phi}{2}\sin\tfrac{\phi}{2})
\end{aligned} \tag{3.12}$$

This formula is used when computing the partial of the joint axes and distance vectors with respect to a revolute-type joint. To demonstrate the use of this formula, an example depicting a general rigid body system is presented.

Figure 3.1 is an example of an open-loop chain wherein the first body is grounded in the global coordinate system. Body two is connected to ground through a revolute joint and body three is attached to body two through a universal joint. A translational

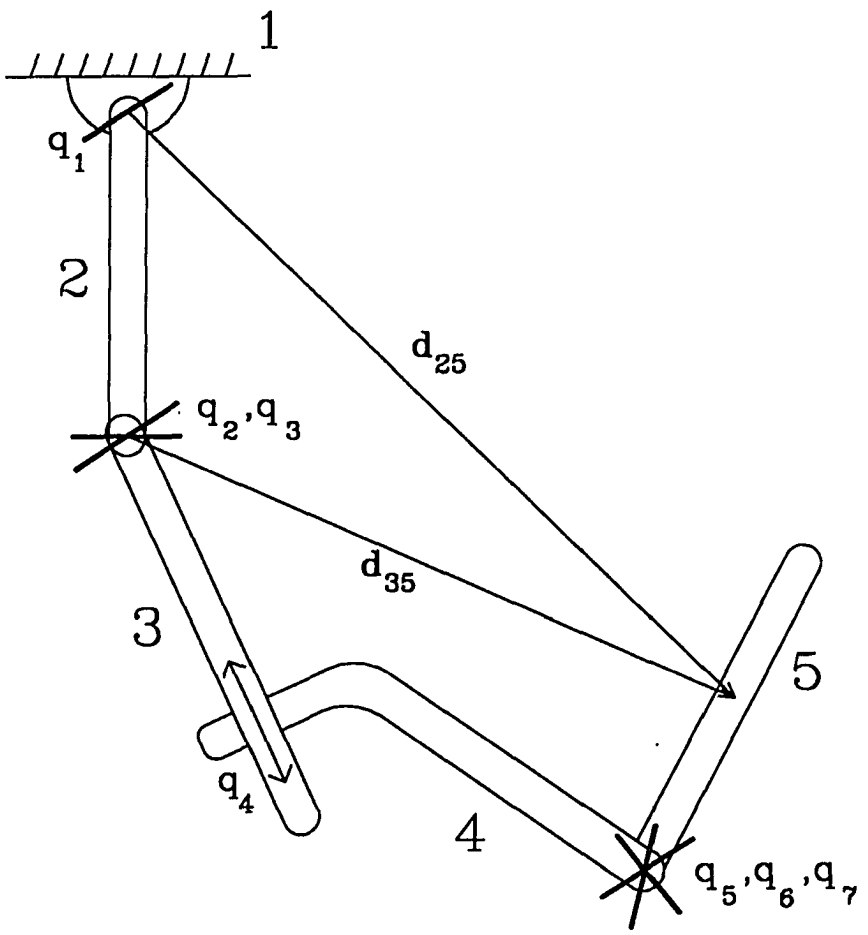



Figure 3.1: Typical Open-Chain System

Table 3.1: Velocity Transformation Matrix for Example 1

| | | | | |
|---|---|---|---|---|
| $\tilde{u}_2 d_{22}$ / $u_2$ | 0 | | 0 | 0 |
| $\tilde{u}_2 d_{23}$ / $u_2$ | ${}_1\tilde{u}_3 d_{33}$ / ${}_1u_3$ | ${}_2\tilde{u}_3 d_{33}$ / ${}_2u_3$ | 0 | 0 |
| $\tilde{u}_2 d_{24}$ / $u_2$ | ${}_1\tilde{u}_3 d_{34}$ / ${}_1u_3$ | ${}_2\tilde{u}_3 d_{34}$ / ${}_2u_3$ | $u_4$ / 0 | 0 |
| $\tilde{u}_2 d_{25}$ / $u_2$ | ${}_1\tilde{u}_3 d_{35}$ / ${}_1u_3$ | ${}_2\tilde{u}_3 d_{35}$ / ${}_2u_3$ | $u_4$ / 0 | ${}_1\tilde{u}_5 d_{55}$ / ${}_1u_5$ ${}_2\tilde{u}_5 d_{55}$ / ${}_2u_5$ ${}_3\tilde{u}_5 d_{55}$ / ${}_3u_5$ |

joint links body four to body three and the final body is connected to four with a spherical joint. Table 3.1 presents the velocity transformation matrix for this system. To examine the steps required to compute the partial differentiation of the velocity transformation matrix with respect to a rotational degree of freedom, one axis of the universal joint located between bodies two and three is used as a specific example. The universal joint has two rotational degrees of freedom associated with it. The first axis of rotation, ${}_1u_3$, is rigidly attached to body number one and the second axis of rotation, ${}_2u_3$, is attached to body two.

To compute the partial of all relevant joint axis vectors with respect to the angle $q_2$ (i.e., the rotation about the first universal axis), the relative position of each joint axis along the chain with respect to $q_2$ must be considered. Since bodies one and two are located further up the chain than the angle $q_2$, the joint axes associated with these bodies are unaffected by $q_2$ and hence, the partials are zero. To compute

the partials of the remaining joint axes, it is important to note that the axes are defined in the global coordinate system. These axes can be defined as a local vector multiplied by a 3x3 transformation matrix. For example, the second axis associated with the spherical joint between bodies four and five can be expressed as

$$ {}^2\mathbf{u}_5 = \left(\mathbf{A}_{5_20}\right) {}^2\mathbf{u}'_5 \tag{3.13} $$

The transformation matrix relating the local and global systems can be defined as a product of two or more matrices so that the above vector can be expressed more precisely as

$$ {}^2\mathbf{u}_5 = \left(\mathbf{A}_{10}\mathbf{A}_{21}\mathbf{A}_{3_12}\mathbf{A}_{3_23_1}\mathbf{A}_{43_2}\mathbf{A}_{5_14}\mathbf{A}_{5_25_1}\right) {}^2\mathbf{u}'_5 \tag{3.14} $$

where $\mathbf{A}_{10}$ is the relative transformation from body one to the global system, $\mathbf{A}_{21}$ is the relative transformation from body two to body one, $\mathbf{A}_{3_12}$ transforms an intermediate system in the universal joint to body two and $\mathbf{A}_{3_23_1}$ transforms the body three system to the intermediate universal system. The matrix $\mathbf{A}_{43_2}$ has been included for formulation purposes only. In general, a translational joint between adjacent coordinate systems always results in an identity matrix. The matrix $\mathbf{A}_{5_14}$ transforms the first intermediate system associated with the spherical joint to the system fixed in body four. Finally, the $\mathbf{A}_{5_25_1}$ matrix transforms the second intermediate system associated with the spherical joint to the first intermediate system. Of the seven relative transformation matrices listed in Equation 3.14, the only matrix that is affected by the $q_2$ coordinate is the transformation relating the intermediate universal system to the body two coordinate system, $\mathbf{A}_{3_12}$. Now, the partial of the middle spherical joint axis with respect to the second generalized coordinate can be expressed as

$$ \frac{\partial_2 \mathbf{u}_5}{\partial q_2} = \mathbf{A}_{pre}\frac{\partial \mathbf{A}_{3_12}}{\partial q_2}\mathbf{A}_{post}{}^2\mathbf{u}'_5 \tag{3.15} $$

where

$$\mathbf{A}_{pre} \equiv \mathbf{A}_{10}\mathbf{A}_{21}$$

$$\mathbf{A}_{post} \equiv \mathbf{A}_{3_2 3_1}\mathbf{A}_{43_2}\mathbf{A}_{5_1 4}\mathbf{A}_{5_2 5_1}$$

In general, the partial of any joint axis with respect to a rotational degree of freedom can be expressed as the partial of a relative rotation matrix pre and post multiplied by an appropriate transformation matrix.

The expression for the partial of any joint axis with respect to a translational degree of freedom is trivial. By definition, joint axis vectors are unit vectors in the direction of the joint axis for a particular joint. A translational degree of freedom has no effect on the direction of a joint axis vector regardless of its location in the chain, therefore, the partial of joint axes vectors with respect to translational degrees of freedom is always zero.

Thus far, the partial of a joint axis vector with respect to rotational and translational degrees of freedom has been computed. Now the distance vectors must be considered. To compute the partial of a distance vector with respect to a rotational degree of freedom, again, the location of the joint associated with the rotation must be considered. In Figure 3.1, the distance vector $\mathbf{d}_{25}$ can be expressed as

$$\mathbf{d}_{25} = -\mathbf{s}_{21} + \mathbf{s}_{23} + \mathbf{d}_{35} \tag{3.16}$$

or, if $\mathbf{d}_{35}$ is expressed in the local coordinate system of the intermediate universal axis

$$\mathbf{d}_{25} = -\mathbf{s}_{21} + \mathbf{s}_{23} + \mathbf{A}_{20}\mathbf{A}_{3_1 2}\mathbf{d}'_{(35)_1} \tag{3.17}$$

The partial of the distance vector with respect to the rotational degree of freedom, $q_2$, becomes

$$\frac{\partial \mathbf{d}_{25}}{\partial q_2} = \mathbf{A}_{20}\frac{\partial \mathbf{A}_{3_1 2}}{\partial q_2}\mathbf{d}'_{(35)_1} \tag{3.18}$$

To compute the partial of $\mathbf{d}_{25}$ with respect to the translational degree of freedom, $q_4$, the distance vector is defined as

$$\mathbf{d}_{25} = \mathbf{d}_{13} + \mathbf{s}_{34} + q_4\mathbf{u}_4 + \mathbf{d}_{45} \tag{3.19}$$

Due to the fact that none of the terms in Equation 3.19 is a function of $q_4$ (except $q_4$ itself), the partial of $\mathbf{d}_{25}$ with respect to the translational degree of freedom is simply

$$\frac{\partial \mathbf{d}_{25}}{\partial q_4} = \mathbf{u}_4 \tag{3.20}$$

As previously mentioned, the degrees of freedom associated with a floating base body can be thought of as three translational and three rotational degrees of freedom. Due to the fact that a floating base body is always the first body in a chain, the partial differentiation of the joint axes and distance vectors with respect to the degrees of freedom associated with the floating base body can be simplified.

Table 3.2 defines the partials of arbitrary joint axes and distance vectors with respect to rotational and translational degrees of freedom. Note that the distance vectors are defined in the coordinate system of body $k$ (i.e., $\left(\mathbf{d}_{kj}\right)'_k = \mathbf{A}^{-1}_{k0}\mathbf{d}_{kj}$). The terms defined in Table 3.2 are used throughout the computation of the linearized damping and stiffness matrices.

Tables 3.3 and 3.4 show the form of the block matrix entries for $\frac{\partial \mathbf{B}}{\partial \mathbf{q}}$. These terms are obtained by applying the results of Table 3.2 to the block matrix terms of the velocity transformation matrix shown in Table 2.2. Note that the terms in Tables 3.3 and 3.4 are three-dimensional. When the assembled three-dimensional matrix is multiplied by the (6xNGC) array of Equation 3.10, an (NGCxNGC) matrix is formed.

Table 3.2: Partial of Joint Axes and Distance Vectors with Respect to Rotational and Translational Degrees of Freedom

| | |
|---|---|
| $\frac{\partial \mathbf{u}_j}{\partial \phi_k} = \begin{cases} \mathbf{A}_{k0} \frac{\partial \mathbf{A}_{k+1,k}}{\partial \phi_k} \mathbf{A}_{j,k+1} \mathbf{u}'_j & \\ \mathbf{0} & \end{cases}$ | if $j > k$<br>otherwise |
| $\frac{\partial \mathbf{u}_j}{\partial b_k} = \mathbf{0}$ | always |
| $\frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} = \begin{cases} \mathbf{A}_{k-1,0} \frac{\partial \mathbf{A}_{k,k-1}}{\partial \phi_k} \left(\mathbf{d}_{kj}\right)'_k & \\ \mathbf{A}_{k-1,0} \frac{\partial \mathbf{A}_{k,k-1}}{\partial \phi_k} \left(\mathbf{d}_{ij}\right)'_k & \\ \mathbf{0} & \end{cases}$ | if $i \leq k$ and $j \geq k$<br>if $i, j > k$<br>otherwise |
| $\frac{\partial \mathbf{d}_{ij}}{\partial b_k} = \begin{cases} \mathbf{u}_k & \\ \mathbf{0} & \end{cases}$ | if $i < k$ and $j \geq k$<br>otherwise |

Table 3.3: Block Entries for the Partial of the Velocity Transformation Matrix with Respect to a Generalized Rotation

| Joint Type of $j$th Body | $\frac{\partial \mathbf{B}_{ij}}{\partial \phi_k}$ |
|---|---|
| Floating Base Body | $\begin{bmatrix} \mathbf{0} & -\frac{\partial \widetilde{\mathbf{d}_{ij}}}{\partial \phi_k} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ |
| Revolute Joint | $\begin{bmatrix} \frac{\partial \widetilde{\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + \tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} \\ \frac{\mathbf{u}_j}{\partial \phi_k} \end{bmatrix}$ |
| Translational Joint | $\begin{bmatrix} \frac{\partial \mathbf{u}_j}{\partial \phi_k} \\ \mathbf{0} \end{bmatrix}$ |
| Cylindrical Joint | $\begin{bmatrix} \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} & \frac{\partial \widetilde{{}_2\mathbf{u}_j}}{\partial \phi_k} \\ \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k} & \mathbf{0} \end{bmatrix}$ |
| Universal Joint | $\begin{bmatrix} \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} & \frac{\partial \widetilde{{}_2\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_2\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} \\ \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k} & \frac{\partial \widetilde{{}_2\mathbf{u}_j}}{\partial \phi_k} \end{bmatrix}$ |
| Spherical Joint | $\begin{bmatrix} \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} & \frac{\partial \widetilde{{}_2\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_2\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} & \frac{\partial \widetilde{{}_3\mathbf{u}_j}}{\partial \phi_k}\mathbf{d}_{ij} + {}_3\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial \phi_k} \\ \frac{\partial \widetilde{{}_1\mathbf{u}_j}}{\partial \phi_k} & \frac{\partial \widetilde{{}_2\mathbf{u}_j}}{\partial \phi_k} & \frac{\partial \widetilde{{}_3\mathbf{u}_j}}{\partial \phi_k} \end{bmatrix}$ |

Table 3.4: Block Entries for the Partial of the Velocity Transformation Matrix with Respect to a Generalized Translation

| Joint Type of $j$th Body | $\frac{\partial \mathbf{B}_{ij}}{\partial b_k}$ |
|---|---|
| Floating Base Body | $\begin{bmatrix} \mathbf{0} & -\widetilde{\frac{\partial \mathbf{d}_{ij}}{\partial b_k}} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ |
| Revolute Joint | $\begin{bmatrix} \tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} \\ \mathbf{0} \end{bmatrix}$ |
| Translational Joint | $\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$ |
| Cylindrical Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ |
| Universal Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} & {}_2\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ |
| Spherical Joint | $\begin{bmatrix} {}_1\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} & {}_2\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} & {}_3\tilde{\mathbf{u}}_j \frac{\partial \mathbf{d}_{ij}}{\partial b_k} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$ |

**3.2.2.2 Partial Differentiation of the Cartesian Force Vector with Respect to the Generalized Coordinates** The second term on the right hand side of Equation 3.10 includes the partial differentiation of the Cartesian force vector with respect to the generalized coordinates. Recall from the previous chapter that the Cartesian force vector is composed of all external forces that act on a body (excluding joint forces which are internal workless forces). Typically, this force vector contains gravitational, translational springs (TSDA), rotational springs (RSDA) and extraneous forces defined through a user-supplied subroutine.

The partial differentiation of the force vector with respect to the generalized coordinates can be computed analytically for the gravitational, TSDA and RSDA type forces. The analytical computation of these terms are discussed below.

For a time-invariant gravitational force field, the partial differentiation of the Cartesian force vector with respect to the generalized coordinates is trivial. Both the external force vector and the gravitational force field are defined in the inertial reference frame. Therefore, the contribution of the gravitational forces to the external force vector remains constant for any system configuration and hence, the partial of the gravitational forces with respect to the generalized coordinates is always zero.

To determine the partial differentiation of a translational spring force with respect to a generalized coordinate, the effects of both rotational and translational degrees of freedom at different locations in the chains are included. These examples can then be generalized to include all combinations of revolute/translational joint types as well as floating base bodies. Figure 3.2 presents a schematic of a TSDA force between two bodies. The force and moment exerted by the translational

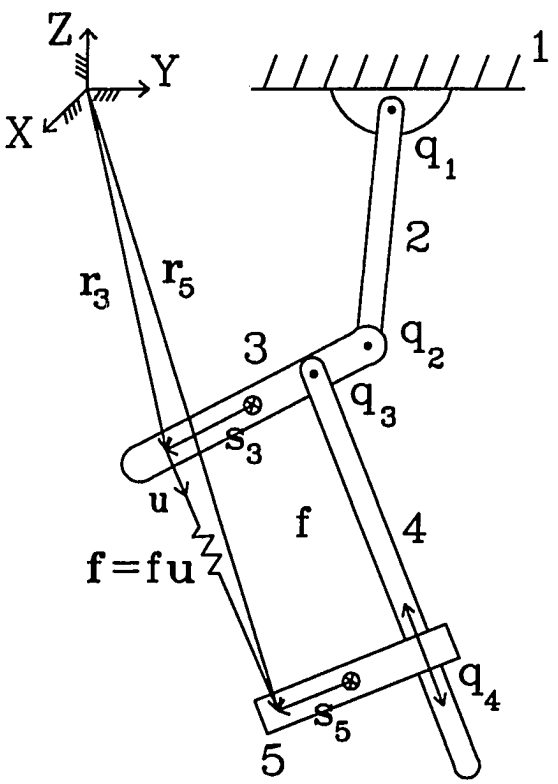



**Figure 3.2: Translational Spring-Damper-Actuator Forces**

spring on bodies $i$ and $j$ is given by

$$\begin{aligned} \mathbf{f}_i &= f\mathbf{u} & \mathbf{f}_j &= -f\mathbf{u} \\ \mathbf{M}_i &= f\tilde{\mathbf{s}}_i\mathbf{u} & \mathbf{M}_j &= -f\tilde{\mathbf{s}}_j\mathbf{u} \end{aligned} \tag{3.21}$$

where $\mathbf{u}$ is a unit vector in the direction defined from the spring attachment point on body $i$ to the spring attachment point on body $j$, $f$ is the magnitude of the force (where tension is positive) and $\mathbf{s}_i$ is the vector from the mass center of body $i$ to the spring attachment point on body $i$. The partial differentiation of these forces and moments with respect to a generalized coordinate, $q$, is given by

$$\begin{aligned} \frac{\partial \mathbf{f}_i}{\partial q} &= \left(\frac{\partial f}{\partial \Delta_l}\right)\left(\frac{\partial \Delta_l}{\partial q}\right)\mathbf{u} + f\frac{\partial \mathbf{u}}{\partial q} \\ \frac{\partial \mathbf{f}_j}{\partial q} &= -\left(\frac{\partial f}{\partial \Delta_l}\right)\left(\frac{\partial \Delta_l}{\partial q}\right)\mathbf{u} - f\frac{\partial \mathbf{u}}{\partial q} \\ \frac{\partial \mathbf{M}_i}{\partial q} &= \left(\frac{\partial f}{\partial \Delta_l}\right)\left(\frac{\partial \Delta_l}{\partial q}\right)\tilde{\mathbf{s}}_i\mathbf{u} + f\left(\frac{\partial \tilde{\mathbf{s}}_i}{\partial q}\mathbf{u} + \tilde{\mathbf{s}}_i\frac{\partial \mathbf{u}}{\partial q}\right) \\ \frac{\partial \mathbf{M}_j}{\partial q} &= -\left(\frac{\partial f}{\partial \Delta_l}\right)\left(\frac{\partial \Delta_l}{\partial q}\right)\tilde{\mathbf{s}}_j\mathbf{u} - f\left(\frac{\partial \tilde{\mathbf{s}}_j}{\partial q}\mathbf{u} + \tilde{\mathbf{s}}_j\frac{\partial \mathbf{u}}{\partial q}\right) \end{aligned} \tag{3.22}$$

where $\Delta_l$ is the deflection of the spring evaluated at the equilibrium state. The variables $f, \mathbf{u}, \mathbf{s}_i$ and $\mathbf{s}_j$ can be computed as outlined in Chapter 3. Also, the partial differentiation of the force magnitude with respect to the spring deflection, $\frac{\partial f}{\partial \Delta_l}$, is a constant for all linear springs and can be evaluated analytically for any nonlinear force/deflection relationship defined using a curve fit technique.

The three remaining terms in Equation 3.22, $\frac{\partial \Delta_l}{\partial q}$, $\frac{\partial \mathbf{s}}{\partial q}$ and $\frac{\partial \mathbf{u}}{\partial q}$ must be evaluated in general with respect to both a translational and a rotational degree of freedom. The first of these three variables is evaluated using the the definition of the spring deflection

$$\Delta_l = \sqrt{(\mathbf{r}_j - \mathbf{r}_i)^T(\mathbf{r}_j - \mathbf{r}_i)} - l_0 \tag{3.23}$$

The partial differentiation of this deflection with respect to a generalized coordinate becomes

$$\frac{\partial \Delta_l}{\partial q} = \frac{(\mathbf{r}_j - \mathbf{r}_i)^T}{\sqrt{(\mathbf{r}_j - \mathbf{r}_i)^T(\mathbf{r}_j - \mathbf{r}_i)}} \left( \frac{\partial \mathbf{r}_j}{\partial q} - \frac{\partial \mathbf{r}_i}{\partial q} \right) \tag{3.24}$$

To compute the partial of the spring attachment points with respect to a generalized coordinate, $\frac{\partial \mathbf{r}_i}{\partial q}$ and $\frac{\partial \mathbf{r}_j}{\partial q}$, both the type of generalized coordinate (i.e., translational or rotational) and the location of the coordinate in the chain with respect to the spring attachment point must be considered. If the body associated with the generalized coordinate is not located in the chain leading up to the spring attachment point in question, the partial of the spring attachment vector with respect to the coordinate is zero. For example, in Figure 3.2, the coordinate $q_3$ is associated with body 4. Since body 4 is not in the chain leading to the spring attachment point in body 3, $\frac{\partial \mathbf{r}_3}{\partial q_4}$ is zero. Body 4, however, is located in the chain leading to the spring attachment point on body 5, therefore, $\frac{\partial \mathbf{r}_5}{\partial q_4}$ is not zero. In this case, the partial of the spring attachment vector with respect to a rotational coordinate associated with a body $k$ is given by

$$\frac{\partial \mathbf{r}_j}{\partial \phi} = \mathbf{A}_{pre} \frac{\partial \mathbf{A}_\phi}{\partial \phi} \mathbf{A}^T{}_{\phi 0} \left( \mathbf{d}_{kj} + \mathbf{s}_j \right) \tag{3.25}$$

where $\mathbf{A}_{pre}$ is the matrix which transforms coordinates from the coordinate system adjacent to the joint axis to the global coordinate system, $\frac{\partial \mathbf{A}_\phi}{\partial \phi}$ is the partial of the transformation matrix relating the joint axis coordinate system to the adjacent system and $\mathbf{A}^T{}_{\phi 0} \left( \mathbf{d}_{kj} + \mathbf{s}_j \right)$ represents the vector from the joint axis to the spring attachment point defined in the joint axis coordinate system. The partial of the spring attachment vector with respect to a translational coordinate associated with

a body $k$ becomes

$$\frac{\partial \mathbf{r}_j}{\partial b} = \mathbf{u}_k \tag{3.26}$$

Next, the partial of the local spring attachment vectors with respect to the generalized coordinates is computed. As with $\frac{\partial \Delta_l}{\partial q}$, the $\frac{\partial \mathbf{s}_j}{\partial q}$ term is nonzero only when the body associated with the generalized coordinate (i.e., body $k$) is located in the path leading to body $j$. In this case, the partial differentiation of the local spring attachment vector with respect to a rotational coordinate is given by

$$\frac{\partial \mathbf{s}_j}{\partial \phi} = \frac{\partial \mathbf{A}_{k0}}{\partial \phi} \left(\mathbf{s}_j\right)'_k \tag{3.27}$$

where $\left(\mathbf{s}_j\right)'_k$ is the local spring attachment vector defined in the coordinate system of body $k$. The partial of the local spring attachment vector with respect to a translational coordinate is always zero.

The last term from Equation 3.22 that needs to be computed is the partial differentiation of the spring direction vector with respect to the generalized coordinates. This unit vector is defined as

$$\mathbf{u} = \frac{(\mathbf{r}_j - \mathbf{r}_i)}{\sqrt{(\mathbf{r}_j - \mathbf{r}_i)^T (\mathbf{r}_j - \mathbf{r}_i)}} \tag{3.28}$$

The partial of this unit vector with respect to a generalized coordinate is given by

$$\frac{\partial \mathbf{u}}{\partial q} = \frac{\left(\frac{\partial \mathbf{r}_j}{\partial q} - \frac{\partial \mathbf{r}_i}{\partial q}\right)}{\sqrt{(\mathbf{r}_j - \mathbf{r}_i)^T (\mathbf{r}_j - \mathbf{r}_i)}} - \frac{(\mathbf{r}_j - \mathbf{r}_i)(\mathbf{r}_j - \mathbf{r}_i)^T}{\left[(\mathbf{r}_j - \mathbf{r}_i)^T (\mathbf{r}_j - \mathbf{r}_i)\right]^{\frac{3}{2}}} \left(\frac{\partial \mathbf{r}_j}{\partial q} - \frac{\partial \mathbf{r}_i}{\partial q}\right) \tag{3.29}$$

where $\frac{\partial \mathbf{r}_j}{\partial q}$ is defined for a translational and a rotational coordinate in Equations 3.26 and 3.25, respectively. Using Equations 3.29, 3.27 and 3.24, the partial of differentiation of a translation spring with respect to a generalized coordinate can be computed from Equation 3.22.

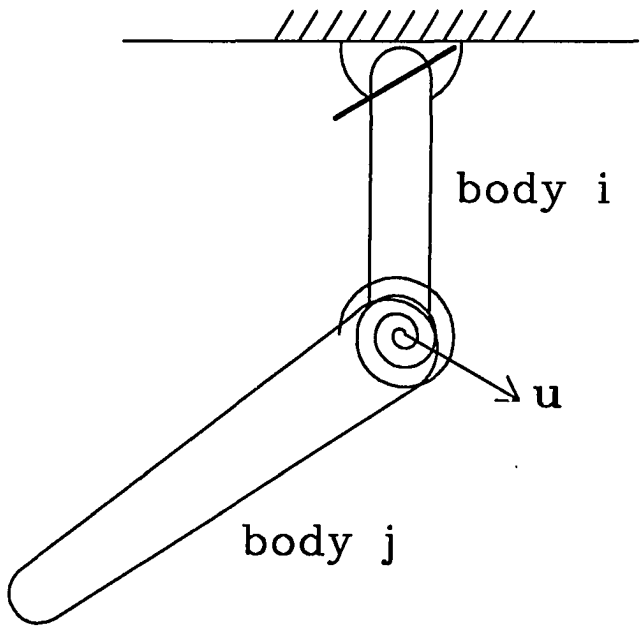



Figure 3.3: Rotational Spring-Damper Forces

Finally, Figure 3.3 presents a schematic of a system containing a rotational spring. The moments exerted by the spring on bodies $i$ and $j$ are defined by

$$\mathbf{M}_i = t\mathbf{u} \quad \mathbf{M}_j = -t\mathbf{u} \tag{3.30}$$

where $t$ is the magnitude of the moment and $\mathbf{u}$ is the unit vector in the direction of the applied torque. The partial differentiation of these moments can be generalized as

$$\frac{\partial \mathbf{M}_i}{\partial \theta} = \frac{\partial t}{\partial \theta}\mathbf{u} + t\mathbf{A}_{pre}\frac{\partial \mathbf{A}_\theta}{\partial \theta}\mathbf{A}_{post}\mathbf{u} \tag{3.31}$$

Thus far, the partial differentiation of the gravitational, TSDA and RSDA forces

with respect to the generalized coordinates have been computed. The only remaining external forces to be addressed are special user-defined forces such as tires. Unfortunately, it is impossible to predict the relationship between the user-supplied external forces and the generalized coordinates. In many cases, these forces are based on highly nonlinear empirical data where a closed-form relationship between the force and the generalized coordinates is impossible to identify. Therefore, the partial differentiation of the Cartesian forces associated with user-supplied subroutines are obtained either by "hard-coding" the force sensitivities (if they are known), or by using a finite difference approach.

$$\left(\frac{\partial \mathbf{f}}{\partial \mathbf{q}}\right) i = \frac{\Delta \mathbf{f}}{\Delta q_i} \tag{3.32}$$

### 3.2.3 Linearized Damping Matrix

An expression for the linearized damping matrix is obtained by applying Equation 3.5 to Equations 3.1 and 3.2

$$\mathbf{C}_l = \mathbf{B}^T \frac{\partial \left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}} + \mathbf{M}\dot{\mathbf{B}} + \mathbf{h} - \mathbf{f}\right)}{\partial \dot{\mathbf{q}}} \tag{3.33}$$

As with the stiffness matrix, equilibrium configurations in which the generalized velocities and accelerations are zero, will be examined first. In this case, the terms associated with $\dot{\mathbf{q}}$ and $\mathbf{h}$ are eliminated and Equation 3.33 can be reduced to to

$$\mathbf{C}_l = \mathbf{B}^T \left(\mathbf{M}\dot{\mathbf{B}} - \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{q}}}\right) \tag{3.34}$$

To evaluate this expression analytically, an expression for $\frac{\partial \mathbf{f}}{\partial \dot{\mathbf{q}}}$ must be derived.

#### 3.2.3.1 Partial Differentiation of the External Cartesian Force Vector with Respect to the Generalized Coordinate Velocities

As mentioned

earlier, the external force vector is composed of gravitational forces, translational spring forces and torques, rotational spring torques and other extraneous forces and torques that act on the system. To determine the partial differentiation of the external force vector with respect to the generalized coordinate velocities, expressions representing the contribution of each force type are derived independently. The summation of these expressions provide the total external force vector sensitivities.

As discussed in Section 3.2.2.2, the gravitational forces and the external force vector are both defined in the inertial reference frame, therefore, the contribution of the gravitational forces to the external force vector remains constant for any system configuration and the partial differentiation of the gravitational forces with respect to the generalized coordinate velocities is always zero.

The forces and moments exerted by a translational spring on two bodies $i$ and $j$ are given by Equation 3.21. The partial differentiation of these forces with respect to a generalized coordinate velocity, $\dot{q}$, is given by

$$\begin{aligned}
\frac{\partial \mathbf{f}_i}{\partial \dot{q}} &= \left(\frac{\partial f}{\partial \dot{\Delta}_l}\right)\left(\frac{\partial \dot{\Delta}_l}{\partial \dot{q}}\right)\mathbf{u} \\
\frac{\partial \mathbf{f}_j}{\partial \dot{q}} &= -\left(\frac{\partial f}{\partial \dot{\Delta}_l}\right)\left(\frac{\partial \dot{\Delta}_l}{\partial \dot{q}}\right)\mathbf{u} \\
\frac{\partial \mathbf{M}_i}{\partial \dot{q}} &= \left(\frac{\partial f}{\partial \dot{\Delta}_l}\right)\left(\frac{\partial \dot{\Delta}_l}{\partial \dot{q}}\right)\tilde{\mathbf{s}}_i\mathbf{u} \\
\frac{\partial \mathbf{M}_j}{\partial \dot{q}} &= -\left(\frac{\partial f}{\partial \dot{\Delta}_l}\right)\left(\frac{\partial \dot{\Delta}_l}{\partial \dot{q}}\right)\tilde{\mathbf{s}}_j\mathbf{u}
\end{aligned} \tag{3.35}$$

where $\dot{\Delta}_l$ is the time change in deflection length of the spring, $\mathbf{u}$ is a unit vector with direction defined from the spring attachment point on body $i$ to the spring attachment point on body $j$, $f$ is the force magnitude and $\tilde{\mathbf{s}}$ is the vector from the body CG to the spring attachment point. With the exception of $\frac{\partial \dot{\Delta}_l}{\partial \dot{q}}$, expressions

for all the terms in Equation 3.35 have been derived in previous sections.

### 3.2.4 Two Degree of Freedom Slider-Pendulum

In order to examine the linearization method in detail, this first example computes the linearized equations associated with a two degree of freedom slider-pendulum. This simple system can easily be linearized analytically and therefore, is an excellent problem for comparison purposes. Figure 3.4 is a schematic of the two degree of freedom slider-pendulum. Ground is defined as body number one, the slider is body number two and the pendulum is body number number three. The nonlinear equations of motion for this system are

$$\begin{aligned}(m_2+m_3)\ddot{y} + \frac{m_3 l}{2}cos\theta\ddot{\theta} - \frac{m_3 l}{2}sin\theta\dot{\theta}^2 + c\dot{y} + ky = 0 \\ \frac{m_3 l}{2}cos\theta\ddot{y} + \left(\frac{m_3 l^2}{4}+j_{3_x}\right)\ddot{\theta} + \frac{m_3 gl}{2}sin\theta = 0\end{aligned} \tag{3.36}$$

Choosing the equilibrium configuration $\theta = 0, y = 0$, and linearizing about this point yields the mass, damping and stiffness matrices.

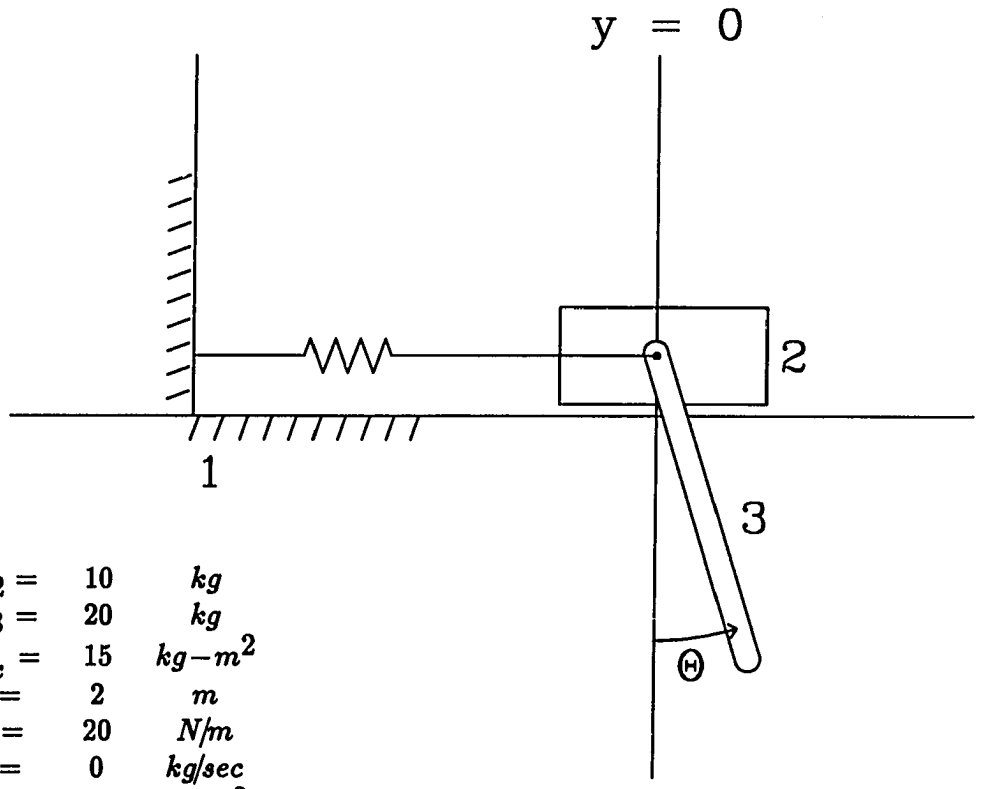
$$\begin{bmatrix} m_2+m_2 & \frac{m_2 l}{2} \\ \frac{m_2 l}{2} & \frac{m_2 l^2}{4}+j_{2_x} \end{bmatrix}\begin{bmatrix}\ddot{y}\\ \ddot{\theta}\end{bmatrix} + \begin{bmatrix} c & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix}\dot{y}\\ \dot{\theta}\end{bmatrix} + \begin{bmatrix} k & 0 \\ 0 & \frac{m_2 gl}{2} \end{bmatrix}\begin{bmatrix} y \\ \theta \end{bmatrix} = \mathbf{0}$$

$$\mathbf{M}_l\begin{bmatrix}\ddot{y}\\ \ddot{\theta}\end{bmatrix} + \mathbf{C}_l\begin{bmatrix}\dot{y}\\ \dot{\theta}\end{bmatrix} + \mathbf{K}_l\begin{bmatrix} y \\ \theta \end{bmatrix} = \mathbf{0} \tag{3.37}$$

where $\mathbf{M}_l, \mathbf{C}_l$ and $\mathbf{K}_l$ are evaluated as

$$\mathbf{M}_l = \begin{bmatrix} 30 & 20 \\ 20 & 35 \end{bmatrix} \qquad \mathbf{C}_l = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad \mathbf{K}_l = \begin{bmatrix} 20 & 0 \\ 0 & 196.2 \end{bmatrix}$$

Now we wish to compare these these results with the linearized matrices computed using the method outlined in this chapter. For the same equilibrium configuration




| | | |
|---|---|---|
| $m_2 =$ | 10 | $kg$ |
| $m_3 =$ | 20 | $kg$ |
| $j_{3_x} =$ | 15 | $kg-m^2$ |
| $l =$ | 2 | $m$ |
| $k =$ | 20 | $N/m$ |
| $c =$ | 0 | $kg/sec$ |
| $g =$ | 9.81 | $m/sec^2$ |

**Figure 3.4: Two Degree of Freedom Slider-Pendulum System**

as above, Figure 3.4 shows that the joint definition axis for the translational joint associated with the slider is defined along the $y$ axis. The joint definition axis for the revolute joint associated with the pendulum is defined along the $x$ axis. Also, the distance vector, $\mathbf{d}_{33}$, is defined along the local $-z$ axis. Using these vectors and Table 2.2, the velocity transformation matrix and the Cartesian mass matrix are computed as

$$\mathbf{B}^T = \left[\begin{array}{cc|cc} \mathbf{u}_2 & \mathbf{0} & \mathbf{u}_2 & \mathbf{0} \\ \hline \mathbf{0} & & \tilde{\mathbf{u}}_3\mathbf{d}_{33} & \mathbf{u}_3 \end{array}\right] = \left[\begin{array}{cccccc|cccccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{array}\right] \tag{3.38}$$

where

$$\mathbf{u}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \mathbf{u}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{d}_{33} = \begin{bmatrix} 0 \\ 0 \\ -\frac{l}{2} \end{bmatrix}$$

$$
\mathrm{M} = \begin{bmatrix}
m_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & m_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & m_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & j_{2x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & j_{2y} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & j_{2z} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & m_3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & m_3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & j_{3x} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & j_{3y} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & j_{3z}
\end{bmatrix} \tag{3.39}
$$

The external force vector contains the spring, damping and gravitational forces applied to each body at the given configuration.

$$
\begin{aligned}
\mathbf{f}^T &= \left[\begin{array}{cccccc|cccccc} 0 & (-ky_2 - c\dot{y}_2) & -m_2 g & 0 & 0 & 0 & 0 & 0 & -m_3 g & 0 & 0 & 0 \end{array}\right] \\
\mathbf{f}^T &= \left[\begin{array}{cccccc|cccccc} 0 & 0 & -98.1 & 0 & 0 & 0 & 0 & 0 & -196.2 & 0 & 0 & 0 \end{array}\right]
\end{aligned} \tag{3.40}
$$

The last two terms required to compute the linearized mass and stiffness matrices are $\frac{\partial \mathbf{B}}{\partial \mathbf{q}}$ and $\frac{\partial \mathbf{f}}{\partial \mathbf{q}}$. From Equation 3.38, the form of the partial of the velocity transformation matrix with respect to the translational and rotational generalized coordinates

is Tables 3.3 and 3.4.

$$\frac{\partial \mathbf{B}}{\partial q_1} = \left[\begin{array}{c|c} \begin{matrix}0\\0\end{matrix} & 0 \\ \hline 0 & \tilde{u}_3 \frac{\partial d_{33}}{\partial q_1} \\ 0 & 0 \end{array}\right]$$

$$\frac{\partial \mathbf{B}}{\partial q_2} = \left[\begin{array}{c|c} \begin{matrix}\frac{\partial u_2}{\partial q_2}\\0\end{matrix} & 0 \\ \hline \begin{matrix}\frac{\partial u_2}{\partial q_2}\\0\end{matrix} & \begin{matrix}\widetilde{\frac{\partial u_3}{\partial q_2}} d_{33} + \tilde{u}_3 \frac{\partial d_{33}}{\partial q_2}\\ \frac{u_3}{\partial q_2}\end{matrix} \end{array}\right] \tag{3.41}$$

Following the rules stated in Table 3.2, the terms involved in computing the partial of the velocity transformation matrix are evaluated as

$$\frac{\partial u_2}{\partial q_1} = \frac{\partial u_2}{\partial q_2} = \frac{\partial u_3}{\partial q_1} = \frac{\partial u_3}{\partial q_2} = \frac{\partial d_{33}}{\partial q_1} = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$\frac{\partial d_{33}}{\partial q_2} = A_{20} \frac{\partial A_{32}}{\partial q_2} (d_{33})'_3 = \begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix} \begin{bmatrix}0&0&0\\0&0&-1\\0&1&0\end{bmatrix} \begin{bmatrix}0\\0\\-1\end{bmatrix} = \begin{bmatrix}0\\1\\0\end{bmatrix} \tag{3.42}$$

inserting these terms into Equation 3.41 yield the two matrix planes of $\frac{\partial \mathbf{B}}{\partial \mathbf{q}}$.

$$\frac{\partial \mathbf{B}^T}{\partial q_1} = \left[\begin{array}{cccccc|cccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$
$$\frac{\partial \mathbf{B}^T}{\partial q_2} = \left[\begin{array}{cccccc|cccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}\right] \tag{3.43}$$

The partial of the force vector with respect to the generalized coordinate positions is given by

$$\frac{\partial \mathbf{f}^T}{\partial q_1} = \left[\begin{array}{cccccc|cccccc} 0 & -k & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$
$$\frac{\partial \mathbf{f}^T}{\partial q_2} = \left[\begin{array}{cccccc|cccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{3.44}$$

Using these terms and Equations 3.8 and 3.10, the linearized mass and stiffness matrices are computed and compared to the closed-form results. Table 3.5 compares the mass and stiffness matrices as well as the eigenvalues for this system. In addition to the closed-form and analytical/numerical results, a finite difference solution is also compared. The finite difference solution uses the same linearized mass matrix (i.e., Equation 3.8) as the analytical solution but computes the stiffness matrix using Equation 3.10 with finite difference solutions for the $\frac{\partial \mathbf{B}^T}{\partial \mathbf{q}}$ and $\frac{\partial \mathbf{f}}{\partial \mathbf{q}}$ terms. A dithering value of 0.0001 for the two generalized coordinates was used to compute the finite difference terms. As can be seen from the table, the computer generated analytical/numerical method produced exactly the same results as the closed-form analysis and the finite difference stiffness method produced nearly the same results.

A CPU time comparison between the computer generated analytical method and the finite difference stiffness method is also given in the table. The finite difference

Table 3.5: Comparison of the Stiffness Matrix, Eigenvalues and CPU Time for a Two Degree of Freedom System

| | $K_l$ |
|---|---|
| Closed-form | $\begin{bmatrix} 20.000000000000 & 0.000000000000 \\ 0.000000000000 & 196.200000000000 \end{bmatrix}$ |
| Analytical Method | $\begin{bmatrix} 20.000000000000 & 0.000000000000 \\ 0.000000000000 & 196.200000000000 \end{bmatrix}$ |
| Finite Difference | $\begin{bmatrix} 19.999999999978 & 0.000000000000 \\ 0.000000000000 & 196.199999673000 \end{bmatrix}$ |

| | $\lambda_1$ | $\lambda_2$ | CPU Seconds[a] |
|---|---|---|---|
| Closed-form | 0.63569205 | 9.49661564 | |
| Analytical Method | 0.63569205 | 9.49661564 | 0.18 |
| Finite Difference | 0.63569205 | 9.49661563 | 0.25 |

[a] CPU comparisons exclude the initial assembly time common to both methods. A VAX 11/785 computer was used for all examples discussed in this thesis.

method required 39 percent more CPU time than the analytical method. Although the overall CPU times are relatively small, this is a substantial difference between the two linearization methods. This difference becomes increasingly important as the system size increases and the process is used for analyses that require repeated linearizations such as design sensitivity and optimization.

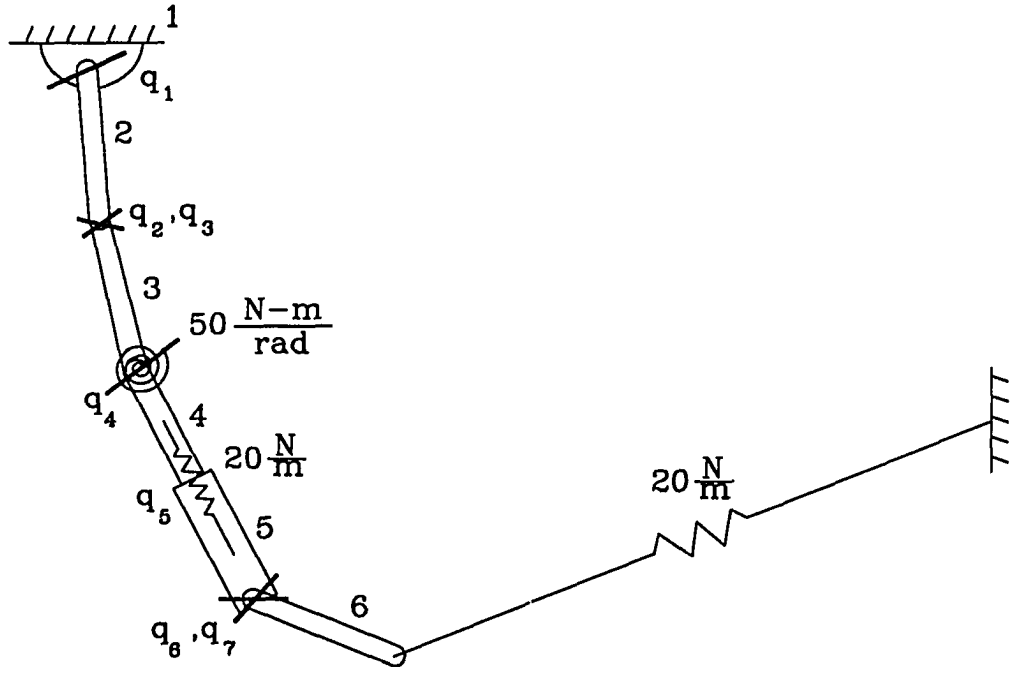
### 3.2.5 Seven Degree of Freedom Example

Figure 3.5 shows a seven degree of freedom open-loop system containing six rigid bodies, three types of joints, one rotational spring and two translational springs. Each uniform rod has a length of 2 meters where the mass and inertia properties vary as shown in the figure. A rotational spring is attached to the revolute joint between bodies three and four and a translational spring is attached to the translational joint defined between bodies four and five. Rod number six has a translational spring attached from its bottom tip to the side wall.

An equilibrium position for this system was obtained by applying damping to all generalized coordinates and performing a dynamic simulation until the system came to rest. Figure 3.5 depicts the approximate equilibrium configuration for the spring constants listed in the figure.

Equation 3.45 presents the velocity transformation matrix for this system.

$$
\mathbf{B} = \left[\begin{array}{c|cc|c|c|cc}
\tilde{u}_2 d_{22} & \multicolumn{2}{c|}{0} & 0 & 0 & \multicolumn{2}{c}{0} \\
u_2 & & & & & & \\
\hline
\tilde{u}_2 d_{32} & {}_1\tilde{u}_3 d_{33} & {}_2\tilde{u}_3 d_{33} & 0 & 0 & \multicolumn{2}{c}{0} \\
u_2 & {}_1u_3 & {}_2u_3 & & & & \\
\hline
\tilde{u}_2 d_{42} & {}_1\tilde{u}_3 d_{43} & {}_2\tilde{u}_3 d_{43} & \tilde{u}_4 d_{44} & 0 & \multicolumn{2}{c}{0} \\
u_2 & {}_1u_3 & {}_2u_3 & u_4 & & & \\
\hline
\tilde{u}_2 d_{52} & {}_1\tilde{u}_3 d_{53} & {}_2\tilde{u}_3 d_{53} & \tilde{u}_4 d_{54} & u_5 & \multicolumn{2}{c}{0} \\
u_2 & {}_1u_3 & {}_2u_3 & u_4 & 0 & & \\
\hline
\tilde{u}_2 d_{62} & {}_1\tilde{u}_3 d_{63} & {}_2\tilde{u}_3 d_{63} & \tilde{u}_4 d_{64} & u_5 & {}_1\tilde{u}_6 d_{66} & {}_2\tilde{u}_6 d_{66} \\
u_2 & {}_1u_3 & {}_2u_3 & u_4 & 0 & {}_1u_6 & {}_2u_6
\end{array}\right] \tag{3.45}
$$




$g = 9.81\ m/sec^2$

| | | | |
|---|---|---|---|
| $m_2 = 5kg$ | $j_{2_x} = 2.0kg-m^2$ | $j_{2_y} = 2.0kg-m^2$ | $j_{2_z} = 0.5kg-m^2$ |
| $m_3 = 3kg$ | $j_{3_x} = 1.2kg-m^2$ | $j_{3_y} = 1.2kg-m^2$ | $j_{3_z} = 0.3kg-m^2$ |
| $m_4 = 2kg$ | $j_{4_x} = 0.8kg-m^2$ | $j_{4_y} = 0.8kg-m^2$ | $j_{4_z} = 0.2kg-m^2$ |
| $m_5 = 1kg$ | $j_{5_x} = 0.4kg-m^2$ | $j_{5_y} = 0.4kg-m^2$ | $j_{5_z} = 0.1kg-m^2$ |
| $m_6 = 1kg$ | $j_{6_x} = 0.4kg-m^2$ | $j_{6_y} = 0.4kg-m^2$ | $j_{6_z} = 0.1kg-m^2$ |
| $m_7 = 1kg$ | $j_{7_x} = 0.4kg-m^2$ | $j_{7_y} = 0.4kg-m^2$ | $j_{7_z} = 0.1kg-m^2$ |

Figure 3.5: Seven Degree of Freedom Open-Loop Example

To verify the eigenvalue results obtained from the linearization process, an FFT analysis was obtained from a dynamic simulation of the system. The simulation was started at the system equilibrium position except for the small generalized coordinate velocities listed below.

$$\dot{\mathbf{q}} = \begin{bmatrix} 0.02 & -0.03 & -0.03 & -0.03 & 0.05 & 0.03 & -0.03 \end{bmatrix}^T$$

The magnitudes and signs of the initial velocities were chosen to give the system a "random" motion. The dynamic simulation was run for 200 seconds and all seven coordinates were sampled at a rate of 5 samples per second. The results of this simulation were analyzed using an FFT routine [22] and Figure 3.6 shows the results this analysis. Table 3.6 compares the natural frequencies obtained from the FFT routine, the analytical linearization process and the finite difference stiffness matrix method. The FFT frequencies are accurate to within plus or minus 0.0025 Hertz. All frequencies computed from the analytical linearization process and the finite difference stiffness matrix method fell within the margin of error surrounding the FFT frequencies. Also, the analytical method and the finite difference method resulted in very similar frequencies, differing by at most 0.002 percent. The finite difference stiffness method required 38 percent more CPU time than the analytical method although the overall times were not excessive for either method. The CPU times included both the computation of the linearized matrices and the eigenvalue solution.

### 3.2.6 Eigenanalysis with Viscous Damping

For the undamped systems presented in the previous section, the natural frequencies were obtained from a straightforward eigenanalysis of the system matrix,

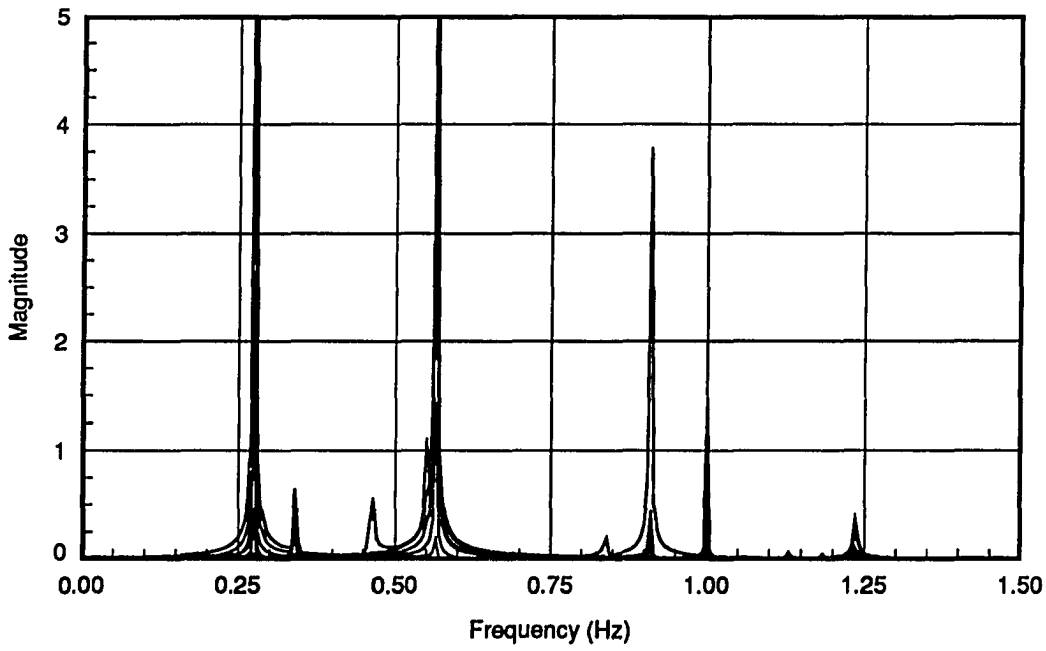



Figure 3.6: FFT Analysis of the Seven Degree of Freedom System

Table 3.6: Comparison of the Natural Frequencies and CPU Times for a Seven Degree of Freedom System

| | FFT | Analytical Method | Finite Difference |
|---|---|---|---|
| $\omega_1$ | 0.275 $\pm$.0025 | 0.27414837 | 0.27415091 |
| $\omega_2$ | 0.340 | 0.34053509 | 0.34054240 |
| $\omega_3$ | 0.460 | 0.46232840 | 0.46232681 |
| $\omega_4$ | 0.565 | 0.56345305 | 0.56345308 |
| $\omega_5$ | 0.910 | 0.90809103 | 0.90809238 |
| $\omega_6$ | 1.000 | 0.99868744 | 0.99867463 |
| $\omega_7$ | 1.235 | 1.23476960 | 1.23477312 |
| CPU Seconds | | 1.93 | 2.66 |

$-\left[\mathrm{M}^{-1}\mathrm{K}\right]$. If viscous damping is present, an alternative technique must be used to obtain the eigenvalues. For the case of proportional damping, where the damping matrix is a linear combination of the mass and stiffness matrices, the mass, stiffness and damping matrices can be transformed to diagonal matrices using the eigenvectors obtained from the undamped system [23]. A second technique involves specifying the damping ratios of the diagonalized system based on experimental results or an estimate of the damping ratio [23].. This technique is referred to as *modal damping* and is used only in the case of light damping.

A third, and more general technique for obtaining the eigenvalues of a damped system involves converting the set of $n$ second order differential equations into an equivalent set of $2n$ first order equations. In 1938, Frazer presented two methods which are still widely used to reduce the second order equations to first order [23,

24, 25]. The first method uses the $n$ generalized momenta, $\mathbf{z} = \mathbf{M}\dot{\mathbf{q}}$, as auxiliary variables to convert to first order. The second method, implemented in this thesis, uses the generalized velocities, $\dot{\mathbf{q}}$, as the auxiliary variables. For this transformation, the set of second order equations of motion with order $n$

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{0} \tag{3.46}$$

can be reduced to a set of first order equations of order $2n$

$$\mathbf{M}_{aux}\dot{\mathbf{y}} + \mathbf{K}_{aux}\mathbf{y} = \mathbf{0} \tag{3.47}$$

where

$$\mathbf{y} \equiv \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}, \qquad \mathbf{M}_{aux} = \mathbf{I}, \qquad \mathbf{K}_{aux} = \left[\begin{array}{c|c} \mathbf{0} & -\mathbf{I} \\ \hline \mathbf{M}^{-1}\mathbf{K} & \mathbf{M}^{-1}\mathbf{C} \end{array}\right]$$

Now the eigenvalues and eigenvectors for the damped system can be obtained from the system matrix

$$\mathbf{A} = -\mathbf{M}^{-1}_{aux}\mathbf{K}_{aux} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{I} \\ \hline -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C} \end{array}\right] \tag{3.48}$$

In general, the eigenvalues of this system can be real, pure imaginary or complex. Pure imaginary and complex eigenvectors appear in pairs of complex conjugates where the pure imaginary eigenvalues represent modes with no damping. Real eigenvalues represent overdamped modes of vibration and complex eigenvalues represent underdamped modes. As an example, Table 3.7 presents the eigenvalues for the system shown in Figure 3.4 both with and without a damping term added to the slider coordinate.

Table 3.7: Eigenvalues for a Damped System

| damping | real part | imaginary part |
|---|---|---|
| c = 0 | 0.000000 | ±0.797303 |
| | 0.000000 | ±3.081658 |
| c = 20 | 0.226284 | ±3.023550 |
| | 0.312177 | ±0.747815 |

# 4. EXTENSIONS TO CLOSED-LOOP SYSTEMS

## 4.1 Decomposition Techniques

Several methods exist which decompose a constrained system to a system with an independent coordinate set. This process involves reducing the set of dependent coordinates into a subset of independent coordinates. The selection of independent coordinates cannot be made arbitrarily because systems that undergo large displacements can present singularity problems in the constraints under certain circumstances. One method of defining a set of independent coordinates involves choosing a subset of $n_{df}$ independent coordinates from the existing dependent coordinate set. Wehage [19] used LU partitioning of the constraint Jacobian matrix to identify the independent generalized coordinates.

A second decomposition technique involves reformulating the system using a different coordinate set and in the process reducing the number of coordinates from $n_{gc}$ to $n_{df}$. Singular value decomposition is one technique that has also been used to determine the independent coordinate set [26]. Kim used a QR decomposition method developed by Golub [27] to identify the rank and orthogonal subspaces of the constraint Jacobian matrix [20]. He then reformulated the system of constrained equations to an independent system of equations. Lynch [28] applied Kim's method to a symbolic formulation based on Kane's equations. This thesis also uses the QR

decomposition technique to reduce a set of constrained linearized equations to an independent system. The independent system allows for a straightforward eigenanalysis to be performed on constrained systems.

It should be noted that the eigenvalues of a constrained system can be computed without reducing to a set of independent coordinates. The algorithm of Moler and Stewart [29] can compute the eigenvalues of a constrained system but is much less efficient than the QR/eigenanalysis method. Moler's eigenvalue algorithm is of order ${n_{gc}}^3$ calculations. The QR decomposition and the matrix multiplication required by the transformation is of order $2\left({n_{gc}}^2\right)$ [30]. The eigenanalysis required by the condensed system is of order ${n_{dof}}^3$ calculations so that the total number of calculations required by the QR/eigenanalysis becomes $2\left({n_{gc}}^2\right) + {n_{dof}}^3$. As an example, the closed-loop passenger car system presented in this chapter has 29 generalized coordinates and ten degrees of freedom. The QR decomposition method results in an efficiency savings of 900 percent when compared to Moler's method. The eigenvalues are the same for both methods.

## 4.2 QR Decomposition of the Constraint Jacobian Matrix

If the constraint Jacobian matrix $\mathbf{\Phi_q}$ has full row rank, then there exists an $(n_{gc} \times n_{gc})$ orthogonal matrix $\mathbf{Q}$ and an $(n_{gc} \times n_c)$ matrix $\mathbf{R}$ [20] such that

$$\mathbf{\Phi_q}^T = \mathbf{QR} \tag{4.1}$$

The matrix $\mathbf{R}$ is defined as

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} \tag{4.2}$$

where $\mathbf{R}_1$ is an $(n_c \times n_c)$ upper triangular matrix. The matrix $\mathbf{Q}$ can be partitioned as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \tag{4.3}$$

where the columns of the $(n_{gc} \times n_c)$ matrix $\mathbf{Q}_1$ form an orthonormal basis for the constraint surface (i.e., the row space of ${\Phi_\mathbf{q}}^T$) and the columns of the $(n_{gc} \times n_{df})$ matrix $\mathbf{Q}_2$ form an orthonormal basis for the constraint tangent surface (i.e., the null space of ${\Phi_\mathbf{q}}^T$). Using these definitions of $\mathbf{Q}$ and $\mathbf{R}$, Equation 4.1 can be written as

$${\Phi_\mathbf{q}}^T = \mathbf{Q}_1\mathbf{R}_1 \tag{4.4}$$

Since $\mathbf{Q}$ is an orthogonal matrix, the inner product of any two columns in $\mathbf{Q}$ must be zero. Therefore, premultiplying Equation 4.4 by ${\mathbf{Q}_2}^T$ yields

$${\mathbf{Q}_2}^T{\Phi_\mathbf{q}}^T = {\mathbf{Q}_2}^T\mathbf{Q}_1\mathbf{R}_1 = \mathbf{0} \tag{4.5}$$

or

$$\Phi_\mathbf{q}\mathbf{Q}_2 = \mathbf{0} \tag{4.6}$$

This relation will be used to reduce the linearized set of constrained equations to an unconstrained system of equations.

## 4.3 QR Application to the Linearized Equations of Motion for Constrained Systems

The unconstrained linearized equations of motion can be extended to constrained systems by including the Lagrange multiplier term ${\Phi_\mathbf{q}}^T\boldsymbol{\lambda}$ from Equation 2.50. The Lagrange multipliers are not independent, and in fact, are chosen such that the virtual displacements of the generalized coordinates, $\delta\mathbf{q}$, can be chosen independently

[21]. Because of this dependence, condition 3.2 holds for both constrained and unconstrained systems and Equation 3.6 can be modified to represent constrained systems by adding the terms associated with the partial differentiation of $\Phi_{\mathbf{q}}{}^{T}\boldsymbol{\lambda}$ with respect to $\mathbf{q}, \dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. However, $\Phi_{\mathbf{q}}{}^{T}\boldsymbol{\lambda}$ is not a function of the generalized velocities or accelerations and therefore, only the linearized stiffness matrix is affected by the Lagrange multiplier term. Adding this term to Equation 3.6, the set of constrained linearized equations of motion can be written as

$$\mathbf{M}_l\ddot{\hat{\mathbf{q}}} + \mathbf{C}_l\dot{\hat{\mathbf{q}}} + \left[\frac{\partial\left(\Phi_{\mathbf{q}}{}^{T}\boldsymbol{\lambda}\right)}{\partial \mathbf{q}} + \mathbf{K}_l\right]\hat{\mathbf{q}} = \mathbf{0} \tag{4.7}$$

There are $n_{gc}$ unknown generalized coordinates plus $n_c$ unknown Lagrange multipliers associated with the system represented by Equation 4.7 but only $n_{gc}$ equations. The QR decomposition method outlined in the previous section can be used to convert the system to $n_{df}$ independent equations.

First, the linearized coordinate basis vector, $\mathbf{q}$, is partitioned as

$$\mathbf{q} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{d} \\ \mathbf{z} \end{bmatrix} \tag{4.8}$$

where $\mathbf{d}$ is an $n_c$ vector of dependent coordinates and $\mathbf{z}$ is an $n_{df}$ vector of independent coordinates. If the new generalized coordinate vector is chosen to lie on the tangent to the constraint surface, then $\mathbf{Q}_1\mathbf{d} = \mathbf{0}$ and

$$\mathbf{q} = \mathbf{Q}_2\mathbf{z} \tag{4.9}$$

For constraints which are not explicitly time dependent, the generalized velocity and acceleration vectors can be expressed as

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{Q}_2\dot{\mathbf{z}} \\ \ddot{\mathbf{q}} &= \mathbf{Q}_2\ddot{\mathbf{z}} \end{aligned} \tag{4.10}$$

Using Equations 4.9 and 4.10 and premultiplying by $\mathbf{Q}_2$, Equation 4.7 becomes

$$\mathbf{Q}_2{}^T\mathbf{M}_l\mathbf{Q}_2\ddot{\hat{\mathbf{z}}} + \mathbf{Q}_2{}^T\mathbf{C}_l\mathbf{Q}_2\dot{\hat{\mathbf{z}}} + \mathbf{Q}_2{}^T\left[\frac{\partial\left(\boldsymbol{\Phi}_\mathbf{q}{}^T\boldsymbol{\lambda}\right)}{\partial\mathbf{q}} + \mathbf{K}_l\right]\mathbf{Q}_2\hat{\mathbf{z}} = \mathbf{0} \tag{4.11}$$

The decomposed linearized stiffness matrix can be written as the sum of three terms

$$\mathbf{K}_{dl} = \mathbf{Q}_2{}^T\frac{\partial\boldsymbol{\Phi}_\mathbf{q}{}^T}{\partial\mathbf{q}}\boldsymbol{\lambda}\mathbf{Q}_2 + \mathbf{Q}_2{}^T\boldsymbol{\Phi}_\mathbf{q}{}^T\frac{\partial\boldsymbol{\lambda}}{\partial\mathbf{q}}\mathbf{Q}_2 + \mathbf{Q}_2{}^T\mathbf{K}_l\mathbf{Q}_2 \tag{4.12}$$

Since the partial differentiation of the constraint Jacobian with respect to $\mathbf{q}$ is orthogonal to the vector of Lagrange multipliers, the first term in the above expression is zero. Also, Equation 4.5 can be applied to eliminate the second term in the expression and the resulting linearized unconstrained equations of motion are expressed as

$$\mathbf{Q}_2{}^T\mathbf{M}_l\mathbf{Q}_2\ddot{\hat{\mathbf{z}}} + \mathbf{Q}_2{}^T\mathbf{C}_l\mathbf{Q}_2\dot{\hat{\mathbf{z}}} + \mathbf{Q}_2{}^T\mathbf{K}_l\mathbf{Q}_2\hat{\mathbf{z}} = \mathbf{0} \tag{4.13}$$

## 4.4 4-Bar Mechanism

To examine the terms involved in a closed-loop system, a classic one degree of freedom problem will be analyzed. Figure 4.1a is a schematic of a closed-loop, spatial four-bar mechanism. The system is modelled with two revolute, one universal and one spherical joint. By cutting the spherical joint and adding the three associated constraint equations to the system as shown in Figure 4.1b, the closed-loop mechanism can be modelled as an open-loop system with constraints. This modelling technique results in four generalized coordinates coupled with three constraint equations and one degree of freedom.

A closed-form solution for the eigenvalue can be obtained by treating the system as a compound pendulum.

$$I_O\ddot{\theta} + M_O = 0 \tag{4.14}$$

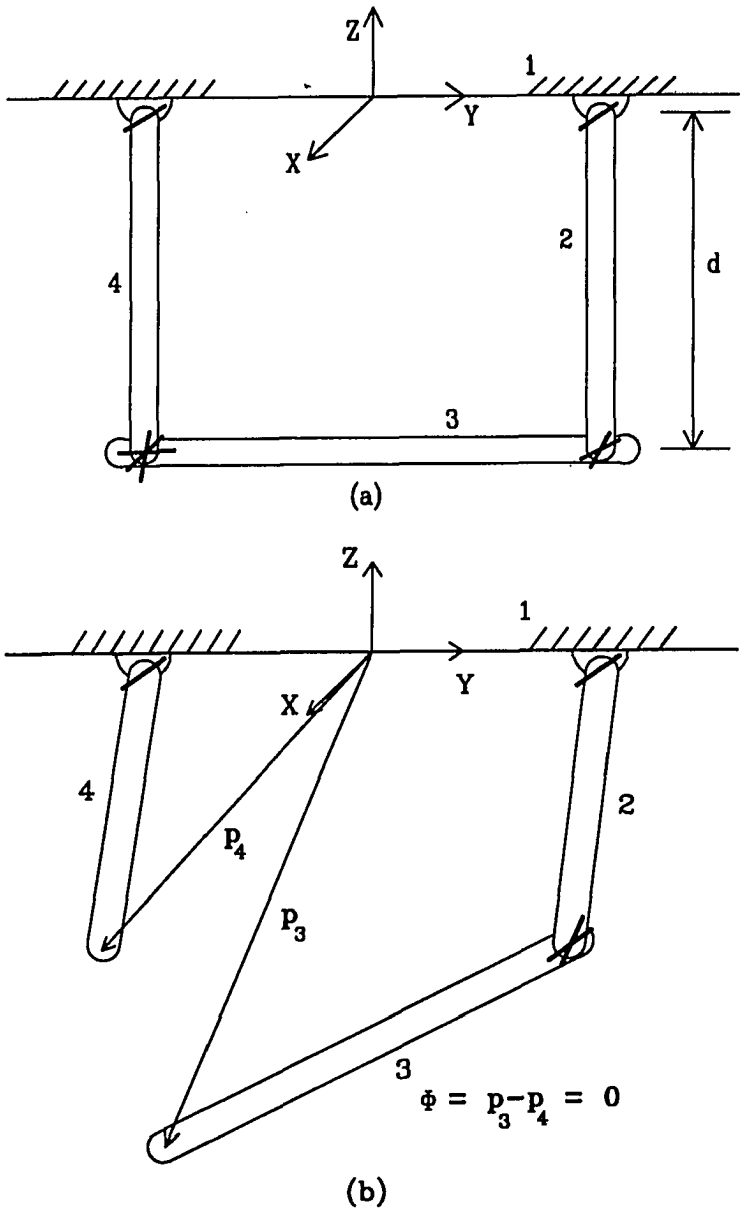



Figure 4.1: Closed-Loop, Four-Bar Mechanism

$I_o$ is the moment of inertia about the point $o$ and is computed as

$$\begin{aligned} I_o &= I_2 + I_4 + m_2\left(\frac{d}{2}\right)^2 + m_3(d)^2 + m_4\left(\frac{d}{2}\right)^2 \\ &= 5 + 5 + 10\left(\frac{1}{2}\right)^2 + 10(1)^2 + 10\left(\frac{1}{2}\right)^2 = 25\ kg{-}m^2 \end{aligned} \tag{4.15}$$

$M_o$ is the external moment about the point $o$ and for small motions is computed as

$$\begin{aligned} M_o &= \tfrac{d}{2}m_2 g\theta + dm_3 g\theta + \tfrac{d}{2}m_4 g\theta \\ &= \tfrac{1}{2}10(9.81)\theta + 10(9.81)\theta + \tfrac{1}{2}10(9.81)\theta = 196.2\theta\ N{-}m \end{aligned} \tag{4.16}$$

Equation 4.14 can now be written as

$$25\ddot{\theta} + 196.2\theta = 0 \tag{4.17}$$

and the eigenvalue is computed to be 7.848.

Using the automated linearization method, the open-loop system linearized mass and stiffness matrices were computed to be

$$\mathbf{M}_l = \begin{bmatrix} 25 & 7.5 & 0 & 0 \\ 7.5 & 7.5 & 0 & 0 \\ 0 & 0 & 7.5 & 0 \\ 0 & 0 & 0 & 7.5 \end{bmatrix}, \quad \mathbf{K}_l = \begin{bmatrix} 147.15 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 49.05 \end{bmatrix} \tag{4.18}$$

and a QR decomposition of the constraint Jacobian matrix yielded a $\mathbf{Q}_2$ matrix of

$$\mathbf{Q}_2 = \begin{bmatrix} 0.57735027 \\ -0.57735027 \\ 0.00000000 \\ 0.57735027 \end{bmatrix} \tag{4.19}$$

Note that the third entry in the $\mathbf{Q}_2$ vector is zero. This is due to the fact that the $\mathbf{Q}_2$ matrix is orthogonal to the constraint tangent surface. In this example, the third

generalized coordinate represents the Z-axis rotation of the universal joint shown in Figure 4.1. The direction of motion of body number three associated with this coordinate is normal to the plane of motion associated with the four-bar system and therefore, a QR decomposition eliminates any contribution of this coordinate to the decomposed system.

Using Equation 4.13, the linearized equation of motion for the four-bar mechanism becomes

$$\left[\mathbf{Q}_2{}^T \mathbf{M}_l \mathbf{Q}_2\right]_{(1\times 1)} \ddot{\hat{\mathbf{z}}} + \left[\mathbf{Q}_2{}^T \mathbf{K}_l \mathbf{Q}_2\right]_{(1\times 1)} \hat{\mathbf{z}} = 0 \tag{4.20}$$

The stiffness value was computed using both the analytical/numerical process and the finite difference method. Table 4.1 compares the constrained system mass, stiffness and eigenvalue results as well as the CPU time for this example. The eigenvalues obtained from both automated linearization methods matched the closed-form eigenvalue to within nine significant digits. Note that the mass and stiffness values obtained from the closed-form analysis differ from the other methods. This is due to the change in coordinate basis from the QR decomposition technique. The system matrices from the QR methods differ from the closed-form matrices by a constant factor of three.

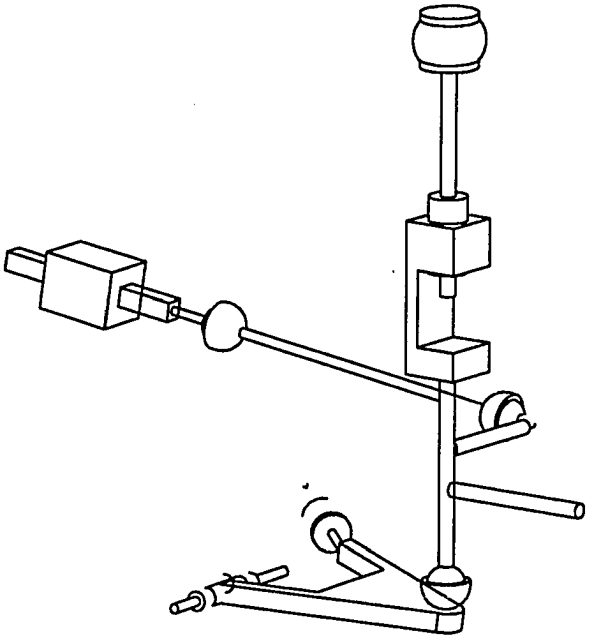
## 4.5 McPherson Strut and Twist Axle Suspensions

To further demonstrate the capabilities of the linearization technique on closed-loop systems, the suspension models depicted in Figure 4.2 were analyzed. The damping was removed from the time response models to facilitate comparison of the undamped natural frequencies computed by the analytical/numerical technique,

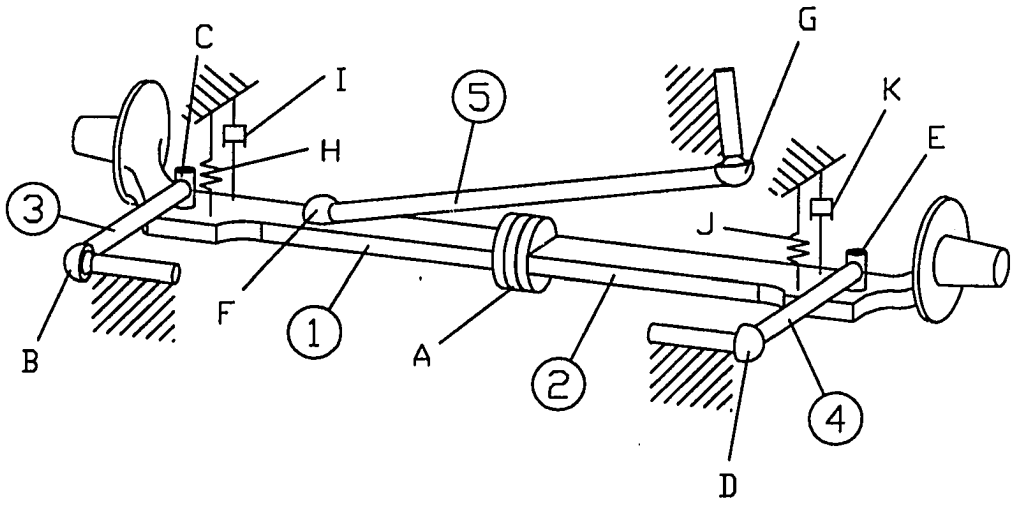
Table 4.1: Comparison of the Mass Matrix, Stiffness Matrix, Eigenvalue and CPU Times for a Closed-Loop, Four-bar System

| | Closed-Form | Analytical/Numerical | Finite Difference |
|---|---|---|---|
| mass | 25.0 | 8.3333333 | 8.3333333 |
| stiffness | 196.2 | 65.4000000 | 65.3999989 |
| $\lambda$ | 7.84800000 | 7.84800000 | 7.84799999 |
| CPU Seconds | | 0.93 | 1.07 |

finite difference technique and the FFT of a time response. Table 4.2 presents a comparison of the undamped frequencies obtained for both the McPherson strut model and the twist-axle model. Note that all frequencies agree to within Fourier transform tolerances. Also note that the finite difference method requires about 75 percent more CPU time than the analytical/numerical method.



McPherson Strut Front Suspension




Twist Axle Rear Suspension

Figure 4.2: Multibody Model of Two Suspension Systems

Table 4.2: Comparison of the Undamped Natural Frequencies and CPU Times for Two Suspension Models

| | FFT | Analytical/Numerical | Finite Difference |
|---|---|---|---|
| strut $\omega$ | $11.600 \pm .025$ | 11.5974 | 11.5973 |
| CPU Seconds[a] | | 0.77 | 1.30 |
| axle $\omega_1$ | $10.333 \pm .033$ | 10.35017 | 10.35741 |
| axle $\omega_2$ | $10.800 \pm .033$ | 10.78386 | 10.78481 |
| CPU Seconds | | 1.17 | 2.07 |

[a] The CPU seconds associated with the analytical method reflect a finite difference computation of the partial differentiation of the tire forces with respect to the generalized coordinates as well as the analytical computation of the remaining linearization terms.

# 5. EQUILIBRIUM STATES WITH NON-ZERO VELOCITIES AND ACCELERATIONS

The systems analyzed thus far have equilibrium states with zero generalized velocities and accelerations. Another group of systems can be categorized as having equilibrium states with non-zero generalized velocities. This type of system typically involves rotating objects about a fixed point in the inertial reference frame. A rotating pendulum and a spinning top are two examples. In some systems, an equilibrium state exists with non-zero generalized accelerations in addition to non-zero velocities. Typically, these systems contain a floating base body with some type of external forces which tend to propel the body in an arc trajectory. A vehicle in a steady turn and an aircraft during a constant $g$ maneuver are two examples of systems that fall into this category. Although these types of systems are in a state of steady motion, the centripetal accelerations associated with the curved trajectories must be accounted for if the eigenvalues of the linearized system are to have the traditional meaning.

To account for the non-zero generalized velocities and accelerations, the linearized coefficient matrices developed in Chapter 3 are expanded to include the terms associated with $\dot{\mathbf{q}}$ and $\tilde{\mathbf{q}}$. Since M and B are not functions of the generalized velocities or accelerations, the linearized mass matrix defined in Equation 3.8 does not

change. The stiffness matrix contains an additional term for steady motion systems with non-zero generalized accelerations and both the stiffness and damping matrices contain additional terms for non-zero generalized velocities. The expressions for the general stiffness and damping matrices are derived below.

## 5.1 Additional Stiffness Matrix Terms

Equation 3.10 represents the stiffness matrix for systems with zero generalized velocities and accelerations. Extending this equation to include all systems with steady motion, the definition for the stiffness matrix in Equation 3.5 is applied to Equation 3.1

$$\mathbf{K}_l = \frac{\partial\left(\mathbf{B}^T\mathbf{M}\mathbf{B}\right)}{\partial \mathbf{q}}\ddot{q} + \mathbf{B}^T\frac{\partial(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{h}-\mathbf{f})}{\partial \mathbf{q}} + \frac{\partial \mathbf{B}^T}{\partial \mathbf{q}}\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{h}-\mathbf{f}\right)$$

$$\mathbf{K}_l = 2\left(\frac{\partial \mathbf{B}}{\partial \mathbf{q}}\right)^T\mathbf{M}\mathbf{B}\ddot{\mathbf{q}} + \mathbf{B}^T\frac{\partial \mathbf{M}}{\partial \mathbf{q}}\mathbf{B}\ddot{q} + \frac{\partial \mathbf{B}^T}{\partial \mathbf{q}}\left(\mathbf{M}\dot{\mathbf{B}}\dot{\mathbf{q}}+\mathbf{h}-\mathbf{f}\right) + \mathbf{B}^T\left(\frac{\partial \mathbf{M}}{\partial \mathbf{q}}\dot{\mathbf{B}}\dot{\mathbf{q}} + \mathbf{M}\frac{\partial \dot{\mathbf{B}}}{\partial \mathbf{q}}\dot{\mathbf{q}} + \frac{\partial \mathbf{h}}{\partial \mathbf{q}} - \frac{\partial \mathbf{f}}{\partial \mathbf{q}}\right) \tag{5.1}$$

Although this stiffness matrix equation is considerably more involved than the expression developed for zero velocity and acceleration steady motion systems, many of the terms in the above equation have already been derived either in Chapter 3 of this thesis or through the dynamic analysis portion of the simulation package. Before Equation 5.1 can be assembled, the following three new terms must derived

$$\frac{\partial \mathbf{M}}{\partial \mathbf{q}},\quad \frac{\partial \dot{\mathbf{B}}}{\partial \mathbf{q}},\quad \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \tag{5.2}$$

### 5.1.1 Partial Differentiation of the Cartesian Mass Matrix with Respect to the Generalized Coordinates

Recall from Equation 2.46 that the Cartesian mass matrix is defined as a diagonal matrix with $6 \times 6$ submatrices on the diagonals. Each submatrix is composed of 3 mass terms and 9 global inertial terms.

$$\mathbf{M}_{ii} = \left[\begin{array}{ccc|c} m_i & & & \\ & m_i & & \mathbf{0} \\ & & m_i & \\ \hline & \mathbf{0} & & \mathbf{J}_i \end{array}\right] \tag{5.3}$$

where the $3 \times 3$ global inertia matrix is a function of the constant local inertia matrix and the transformation matrix relating body system $i$ to the inertial reference frame

$$\mathbf{J}_i = \mathbf{A}_{i0}\mathbf{J}'_i\mathbf{A}^T{}_{i0} \tag{5.4}$$

Since the $m_i$'s and $\mathbf{J}'_i$'s are constant, the partial differentiation of the Cartesian mass submatrix reduces to

$$\frac{\partial \mathbf{M}_{ii}}{\partial \mathbf{q}} = \left[\begin{array}{ccc|c} 0 & & & \\ & 0 & & \mathbf{0} \\ & & 0 & \\ \hline & \mathbf{0} & & 2\left(\frac{\partial \mathbf{A}_{i0}}{\partial \mathbf{q}}\right)\mathbf{J}'_i\mathbf{A}^T{}_{i0} \end{array}\right] \tag{5.5}$$

where $\frac{\partial \mathbf{A}_{i0}}{\partial \mathbf{q}}$ is defined in Equation 3.12.

### 5.1.2 Partial Differentiation of the Time Derivative of the Velocity Transformation Matrix with Respect to the Generalized Coordinates

Table 2.3 contains the block matrix entries for the time derivative of the velocity transformation matrix. All the terms in the block matrices are composed of some combination of the following $3 \times 1$ vectors:

| | |
|---|---|
| $\mathbf{d}_{ij}$ | distance vectors |
| $\dot{\mathbf{d}}_{ij}$ | time derivatives of distance vectors |
| $\mathbf{u}_j$ | joint axis vectors |
| $\omega_j$ | global angular velocity vectors |

The partial differentiation of the distance vectors, $\mathbf{d}_{ij}$, and joint axis vectors, $\mathbf{u}_j$, have been derived in Chapter 3 and are presented in Table 3.2.

Although the magnitudes of the global angular velocity vectors are not functions of the generalized coordinates, the direction of the $\omega_j$'s are affected by a change in a generalized rotational coordinate. Clearly, the partial differentiation of $\omega_j$ with respect to a translational coordinate or with respect to a rotational coordinate that is outward on the chain with respect to body $j$ is zero. To compute the partial differentiation of $\omega_j$ with respect to an inward rotational coordinate, the global angular velocity vector is first written as combination of two vectors

$$\omega_j = \omega_i + \omega_k \tag{5.6}$$

where $\omega_i$ is the component of global angular velocity up to and including the joint axis in question. $\omega_k$ is the component of angular velocity from the joint axis in question outward on the chain to body $j$. The inward component of angular velocity is not affected by a change in the rotational coordinate, $q_n$, and therefore, the partial

of the global angular velocity vector can be written as

$$\frac{\partial \omega_j}{\partial q_n} = \frac{\partial(\omega_i + \omega_k)}{\partial q_n} = \omega \frac{\partial \mathbf{u}}{\partial q_n} \tag{5.7}$$

where $\omega$ is the magnitude and u is the direction of the angular velocity vector $\omega_k$. u can be expressed in the local coordinate system of the joint, $q_n$, as

$$\mathbf{u} = \mathbf{A}_{j0}\mathbf{A}_{j+1,j}\mathbf{u}' \tag{5.8}$$

where $\mathbf{A}_{j+1,j}$ is the transformation matrix associated with the generalized coordinate $q_n$ and $\mathbf{A}_{j0}$ is the transformation matrix relating the adjacent system to the inertial system. Combining Equations 5.7 and 5.8, the partial differentiation of the global angular velocity vector with respect to a generalized coordinate can be expressed as

$$\frac{\partial \omega_j}{\partial q_n} = \begin{cases} \mathbf{0} & \text{if } q_n \text{ is a translational coordinate or if } q_n \text{ is located outward on the chain with respect to body } j. \\ \omega \mathbf{A}_{j0} \dfrac{\partial \mathbf{A}_{j+1,j}}{\partial q_n} \mathbf{u}' & \text{otherwise} \end{cases} \tag{5.9}$$

where $\frac{\partial \mathbf{A}_{j+1,j}}{\partial q_n}$ is given by Equation 3.12.

The remaining term required to compute the partial differentiation of $\dot{\mathbf{B}}$ with respect to q is the partial differentiation of the time derivative of the distance vector with respect to the generalized coordinate vector, q. Figure 5.1 snows a distance vector $\mathbf{d}_{25}$ which can be expressed as the sum of the vectors $\mathbf{s}_{21}$, $\mathbf{s}_{23}$ and $\mathbf{d}_{35}$. The time derivative of $\mathbf{d}_{25}$ can then be expressed as

$$\begin{aligned} \dot{\mathbf{d}}_{25} = \; & -\dot{\mathbf{s}}_{21} + \dot{\mathbf{s}}_{23} + \dot{\mathbf{A}}_{20}\mathbf{A}_{32}\mathbf{d}'_{(35)_1} + \\ & \mathbf{A}_{20}\dot{\mathbf{A}}_{32}\mathbf{d}'_{(35)_1} + \mathbf{A}_{20}\mathbf{A}_{32}\dot{\mathbf{d}}'_{(35)_1} \end{aligned} \tag{5.10}$$

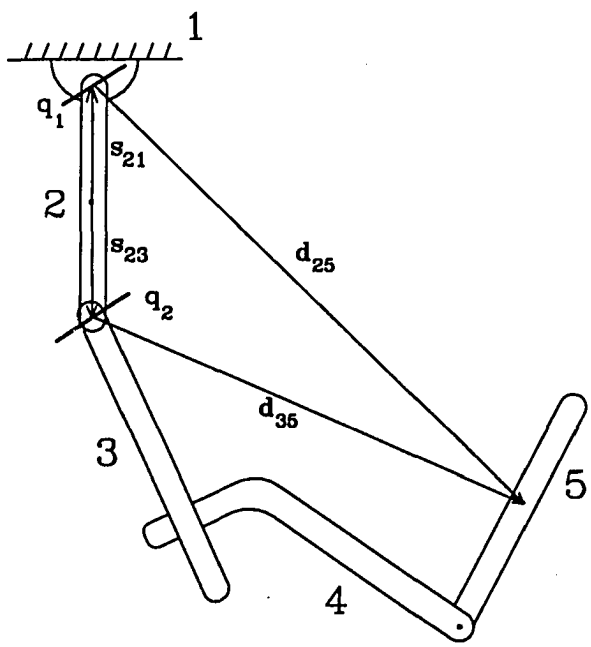



Figure 5.1: Distance Vector

where $\mathbf{A}_{32}$ represents the transformation matrix associated with joint coordinate $q_2$. The partial differentiation of $\dot{\mathbf{d}}_{25}$ with respect to the coordinate, $q_2$, is given by

$$\begin{aligned}\frac{\partial \dot{\mathbf{d}}_{25}}{\partial q_2} = \; & \dot{\mathbf{A}}_{20}\frac{\partial \mathbf{A}_{32}}{\partial q_2}\mathbf{d}'_{(35)_1} + \\ & \mathbf{A}_{20}\frac{\partial \dot{\mathbf{A}}_{32}}{\partial q_2}\mathbf{d}'_{(35)_1} + \\ & \mathbf{A}_{20}\frac{\partial \mathbf{A}_{32}}{\partial q_2}\dot{\mathbf{d}}'_{(35)_1}\end{aligned} \tag{5.11}$$

where $\frac{\partial \mathbf{A}_{32}}{\partial q_2}$ is defined by Equation 3.12 and $\frac{\partial \dot{\mathbf{A}}_{32}}{\partial q_2}$ can be computed as

$$\frac{\partial \dot{\mathbf{A}}_{32}}{\partial q_2} = \tilde{\omega}\frac{\partial \mathbf{A}_{32}}{\partial q_2} \tag{5.12}$$

In this context, $\omega$ represents the global rotational velocity component associated with the joint $q_2$. The partial differentiation of the time derivative of a distance vector with respect to a translational coordinate is zero.

### 5.1.3 Partial Differentiation of h with Respect to the Generalized Coordinates

As presented in Chapter 2, the vector $\mathbf{h}$ is composed of a group of $6\times 1$ subvectors associated with each body in the system. These subvectors are the cross product of the angular velocity of each body with it's angular momentum and can be expressed using the local $3 \times 3$ inertia matrix as

$$\mathbf{h}_i = \left[\; 0^T, \;\; \left(\tilde{\omega}_i \mathbf{A}^T{}_{i0}\mathbf{J}'{}_i\mathbf{A}_{i0}\omega_i\right)T \;\right]^T \tag{5.13}$$

The partial of $\mathbf{h}_i$ with respect to a generalized coordinate can be expressed as

$$\frac{\partial \mathbf{h}_i}{\partial q} = \begin{bmatrix} \mathbf{0} \\ 2\tilde{\omega}_i\left(\frac{\partial \mathbf{A}_{i0}}{\partial q}\right)T\mathbf{J}'{}_i\mathbf{A}_{i0}\omega_i + \frac{\partial \tilde{\omega}_i}{\partial q}\mathbf{J}_i\omega_i + \tilde{\omega}_i\mathbf{J}_i\frac{\partial \omega_i}{\partial q} \end{bmatrix} \tag{5.14}$$

## 5.2 Additional Damping Matrix Terms

Equation 3.33 represents the damping matrix for a general system under steady motion including non-zero generalized velocities and accelerations. This equation can be expanded to examine the individual sensitivity terms in the damping matrix

$$\mathbf{C}_l = \mathbf{B}^T \left( \mathbf{M} \frac{\partial \dot{\mathbf{B}}}{\partial \dot{\mathbf{q}}} \dot{\mathbf{q}} + \mathbf{M}\dot{\mathbf{B}} + \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{q}}} \right) \tag{5.15}$$

The partial derivative of the external Cartesian force vector with respect to the velocities was derived in Section 3.2.3.1. To evaluate Equation 5.15, expressions for the following two terms must be derived

$$\frac{\partial \dot{\mathbf{B}}}{\partial \dot{\mathbf{q}}}, \quad \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}} \tag{5.16}$$

### 5.2.1 Partial Differentiation of the Time Derivative of the Velocity transformation Matrix with Respect to the Generalized Coordinate Velocities

Table 2.3 defines the block matrix entries of the time derivative of the velocity transformation matrix. From this table, it is evident that $\omega_j$ and $\dot{\mathbf{d}}_{ij}$ are the only terms in the time derivative of the velocity transformation matrix that are functions of the generalized velocities. To compute the partial differentiation of a rotational velocity vector with respect to a generalized velocity, the rotational velocity vector is first expressed as

$$\omega_i = \omega_l + \omega_m + \omega_n \tag{5.17}$$

where $\omega_l$ is the angular velocity vector of the reference body associated with $\dot{q}_j$, $\omega_m$ is the relative angular velocity vector of the body associated with $\dot{q}_j$ with respect

to its reference body and $\omega_n$ is the relative angular velocity vector of body $i$ with respect to the body associated with $\dot{q}_j$. Note that only $\omega_m$ is affected by a change in $\dot{q}_j$ and therefore, the partial differentiation of $\omega_i$ with respect to $\dot{q}_j$ becomes

$$\frac{\partial \omega_i}{\partial \dot{q}_j} = \frac{\partial \omega_m}{\partial \dot{q}_j} = \begin{cases} \mathbf{u}_j & \text{if } q_j \text{ is a rotational coordinate} \\ \mathbf{0} & \text{if } q_j \text{ is a translational coordinate} \end{cases} \tag{5.18}$$

where $\mathbf{u}_j$ is the joint axis associated with the generalized coordinate, $q_j$.

The partial derivative of the time derivative of a distance vector, $\dot{\mathbf{d}}_{ij}$, with respect to a generalized velocity, $\dot{q}_i$, can be generalized by examining rotational and translational coordinates separately. Returning to Figure 3.1, the distance vector $\mathbf{d}_{25}$ has both rotational and translational coordinates in the chain. The time derivative of Equation 3.17 can be expressed as

$$\begin{aligned} \dot{\mathbf{d}}_{25} = \; & -\dot{\mathbf{s}}_{21} + \dot{\mathbf{s}}_{23} + \dot{\mathbf{A}}_{20}\mathbf{A}_{3_1 2}\mathbf{d}'_{(35)_1} + \\ & \mathbf{A}_{20}\dot{\mathbf{A}}_{3_1 2}\mathbf{d}'_{(35)_1} + \mathbf{A}_{20}\mathbf{A}_{3_1 2}\dot{\mathbf{d}}'_{(35)_1} \end{aligned} \tag{5.19}$$

The partial differentiation of this vector with respect to the time derivative of the rotational coordinate associated with the intermediate universal axis becomes

$$\begin{aligned} \frac{\partial \dot{\mathbf{d}}_{25}}{\partial \dot{q}_2} &= \mathbf{A}_{20}\frac{\partial \dot{\mathbf{A}}_{3_1 2}}{\partial \dot{q}_2}\mathbf{d}'_{(35)_1} \\ &= \mathbf{A}_{20}\frac{\partial\left(\tilde{\mathbf{u}}_2' \dot{q}_2\right)}{\partial \dot{q}_2}\mathbf{A}_{3_1 2}\mathbf{d}'_{(35)_1} \\ &= \mathbf{A}_{20}\tilde{\mathbf{u}}_2'\mathbf{A}_{3_1 2}\mathbf{d}'_{(35)_1} \end{aligned} \tag{5.20}$$

where $\mathbf{A}_{20}$ is the transformation matrix which transforms the coordinate system of body 2 to ground, $\mathbf{u}_2'$ is the local coordinate vector associated with the first axis in

the universal joint, $A_{3_1 2}$ is the coordinate system which transforms the intermediate universal axis system to body system 2 and $\mathbf{d}'_{(35)_1}$ is the 35 distance vector defined in the intermediate coordinate system of the universal joint.

To compute the partial differentiation of the time derivative of the distance vector with respect to a time derivative of a translational coordinate, the time derivative of Equation 3.19 is expressed as

$$\dot{\mathbf{d}}_{25} = \dot{\mathbf{d}}_{13} + \dot{\mathbf{s}}_{34} + \dot{q}_4 \mathbf{u}_4 + q_4 \dot{\mathbf{u}}_4 + \dot{\mathbf{d}}_{45} \tag{5.21}$$

The partial differentiation of this vector with respect to a time derivative of the translational coordinate becomes

$$\frac{\partial \dot{\mathbf{d}}_{25}}{\partial \dot{q}_4} = \mathbf{u}_4 \tag{5.22}$$

Using Equations 5.20 and 5.22, the partial differentiation of the time derivative of a distance vector can be generalized as

$$\frac{\partial \dot{\mathbf{d}}_{ij}}{\partial \dot{q}_k} = \begin{cases} 0 & \text{if the body associated with } q_k \text{ is greater than } j \\ \mathbf{A}_{m0} \tilde{\mathbf{u}}_k{}' \mathbf{A}_{jk} \mathbf{d}_{kj}{}' & q_k \text{is a rotational coordinate} \\ \mathbf{u}_k & q_k \text{is a translational coordinate} \end{cases} \tag{5.23}$$

where $m$ is the reference coordinate system of the joint axis, $\mathbf{u}_k{}'$ is the joint axis associated with $q_k$ defined in the $m$ local coordinate system and $\mathbf{d}_{kj}{}'$ is the local distance vector defined from the joint definition point of coordinate $q_k$ to the CG of body $j$ and defined in the coordinate system associated with $q_k$.

Applying Equations 5.18 and 5.23 to Table 2.3, all the block entries for the partial differentiation of the time derivative of the velocity transformation matrix with respect to the generalized velocities can be computed and added to the linearized damping matrix defined in Equation 5.15.

### 5.2.2 Partial Differentiation of h with Respect to the Generalized Coordinate Velocities

Using Equation 2.47, the partial differentiation of h with respect to a generalized coordinate velocity can be written as

$$\frac{\partial \mathbf{h}}{\partial \dot{q}_j} = \begin{bmatrix} \mathbf{0} \\ \left(\widetilde{\frac{\partial \omega_1}{\partial \dot{q}_j}}\right)\mathbf{J}_1\omega_1 + \tilde{\omega}_1\mathbf{J}_1\frac{\partial \omega_1}{\partial \dot{q}_j} \\ \hline \mathbf{0} \\ \left(\widetilde{\frac{\partial \omega_2}{\partial \dot{q}_j}}\right)\mathbf{J}_2\omega_2 + \tilde{\omega}_2\mathbf{J}_2\frac{\partial \omega_2}{\partial \dot{q}_j} \\ \hline \vdots \\ \hline \mathbf{0} \\ \left(\widetilde{\frac{\partial \omega_{n_b}}{\partial \dot{q}_j}}\right)\mathbf{J}_{n_b}\omega_{n_b} + \tilde{\omega}_{n_b}\mathbf{J}_{n_b}\frac{\partial \omega_{n_b}}{\partial \dot{q}_j} \end{bmatrix} \tag{5.24}$$

The partial derivative of $\omega_j$ with respect to a generalized velocity is defined in Equation 5.18. Using this equation, a general expression for the partial differentiation of $\mathbf{h}_i$ with respect to a generalized velocity, $\dot{q}_j$, can be expressed as

$$\frac{\partial \mathbf{h}_i}{\partial \dot{q}_j} = \begin{cases} \mathbf{0} & \text{if } q_j \text{ is a translational coordinate or if } i \text{ is less than the body associated with } q_j \\ \tilde{\mathbf{u}}_j\mathbf{J}_i\omega_i + \tilde{\omega}_i\mathbf{J}_i\mathbf{u}_j & \text{otherwise} \end{cases} \tag{5.25}$$

Equation 5.25 holds for any velocity configurations at an equilibrium point. Note that if the coordinate velocities are zero at equilibrium, then Equation 5.25 is zero always.

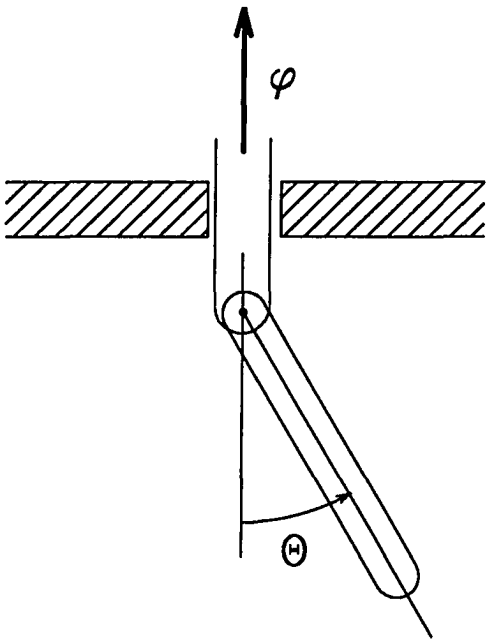



Figure 5.2: Rotating Pendulum

## 5.3 Rotating Pendulum

As an example of a system linearization with non-zero generalized velocities, the pendulum show in Figure 5.2 is free to rotate about the vertical axis. The Lagrangian for this system is expressed as

$$L = \frac{1}{2}ml^2\dot{\theta}^2 + \frac{1}{2}ml^2\sin^2\theta\dot{\phi}^2 + mgl\cos\theta \tag{5.26}$$

and the closed-form solution can be obtained by applying Lagrange's equations

$$\begin{aligned} ml^2\ddot{\theta} - ml^2\sin\theta\cos\theta\dot{\phi}^2 + mgl\sin\theta = 0 \\ 2ml^2\sin\theta\cos\theta\dot{\theta}\dot{\phi} + ml^2\sin^2\theta\ddot{\phi} = 0 \end{aligned} \tag{5.27}$$

Table 5.1: Parameters for the Rotating Pendulum

| | |
|---|---|
| $m$ | $= 10\ kg$ |
| $l$ | $= 0.5\ m$ |
| $\theta_o$ | $= 30\ degrees$ |
| $\dot{\phi}_o$ | $= 4.75975\ \frac{rad}{sec}$ |
| $g$ | $= 9.81\ \frac{m}{sec^2}$ |
| Resulting Eigenvalues: | $0.00000 \pm 0.00000i$ |
| | $0.00000 \pm 8.58076i$ |

When these equations are expanded in a Taylor series, linearized and and assembled in matrix form they become

$$\left[\begin{array}{c|c} ml^2 & 0 \\ \hline 0 & ml^2 \sin^2\theta_o \end{array}\right]\left[\begin{array}{c} \ddot{\theta} \\ \ddot{\phi} \end{array}\right] + \left[\begin{array}{c|c} 0 & -2ml^2 \sin\theta_o \cos\theta_o \dot{\phi}_o \\ \hline 2ml^2 \sin\theta_o \cos\theta_o \dot{\phi}_o & 0 \end{array}\right]\left[\begin{array}{c} \dot{\theta} \\ \dot{\phi} \end{array}\right]$$

$$+ \left[\begin{array}{c|c} ml^2\dot{\phi}^2{}_o(\sin^2\theta_o - \cos^2\theta_o) + mgl\cos\theta_o & 0 \\ \hline 0 & 0 \end{array}\right]\left[\begin{array}{c} \theta \\ \phi \end{array}\right] = \mathbf{0} \tag{5.28}$$

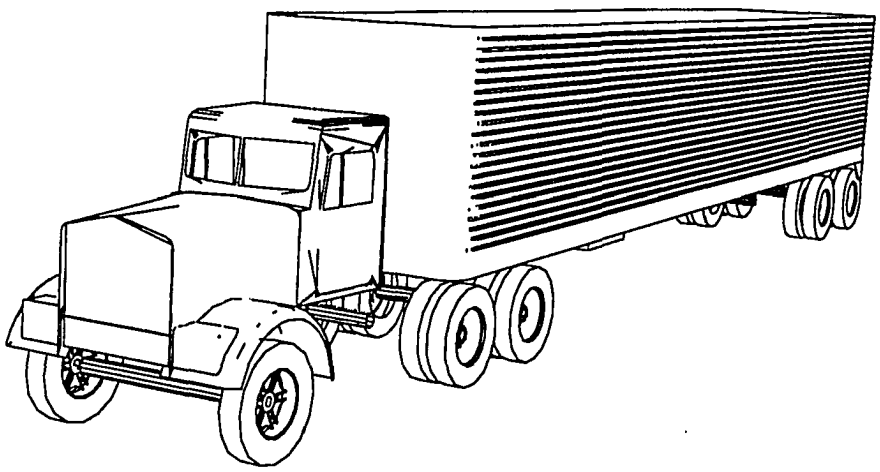
Note that the "damping" matrix is skew symmetric with zero diagonal terms. The off diagonal terms show the velocity coupling between the two coordinates and the zero diagonal terms indicate that the motion of the linearized system is undamped. For the parameters listed in Table 5.1, both the closed-form solution and the analytical/numerical solution resulted in the same eigenvalues to within eight significant digits.

## 5.4 5-Axle Tractor Semi-Trailer

As an example of a system with non-zero generalized accelerations, a 19 degree of freedom tractor semi-trailer is linearized in a steady turn. Figure 5.3 contains a schematic of the truck as well as a description of the 19 generalized coordinates associated with the system. In this example, the tractor has a constant centripetal acceleration of 0.25 $g$'s with a forward speed of about 50 $mph$.

Table 5.2 presents the complex eigenvalues, damped frequencies, undamped frequencies and the damping ratios from the linearized system. Several of the eigenvectors are plotted in Figures 5.4 through 5.12. The eigenvector plots contain both magnitude and phase for the 19 generalized coordinates associated with the truck. The magnitudes are plotted against the left-hand axis and are represented by bars. The phase angles are plotted against the right-hand axis and are represented by solid squares.

The eigenanalysis examples presented in this chapter demonstrate the generality of linearization method by applying this technique to systems with non-zero velocities and accelerations.



1 global X-position of the tractor
2 global Y-position of the tractor
3 global Z-position of the tractor
4 tractor rotation about the global X axis
5 tractor rotation about the global Y axis
6 tractor rotation about the global Z axis
7 first relative rotation between the tractor and trailer
8 second relative rotation between the tractor and trailer
9 third relative rotation between the tractor and trailer
10 relative translation between the front axle and the tractor
11 relative rotation between the front axle and the tractor
12 relative translation between the front drive axle and the tractor
13 relative rotation between the front drive axle and the tractor
14 relative translation between the rear drive axle and the tractor
15 relative rotation between the rear drive axle and the tractor
16 relative translation between the first traile axle and the trailer
17 relative rotation between the first trailer axle and the trailer
18 relative translation between the second trailer axle and the trailer
19 relative rotation between the second trailer axle and the trailer

Figure 5.3: Degrees of Freedom for the 5-Axle Tractor Semi-Trailer

Table 5.2: Eigenanalysis of a 5-Axle Truck

| Eigenvalue Number | Real Part | Imaginary Part | Damped Fequency | Undamped Frequency | Damping Ratio |
|---|---|---|---|---|---|
| 1 | 0.000 | 0.000 | -- | -- | -- |
| 2 | 0.000 | 0.000 | -- | -- | -- |
| 3 | -0.478 | 0.000 | -- | -- | -- |
| 4 | -2.382 | 0.000 | -- | -- | -- |
| 5 | -10.951 | 0.000 | -- | -- | -- |
| 6 | -22.673 | 0.000 | -- | -- | -- |
| 7 | -0.020 | -0.121 | 0.019 | 0.019 | 0.161 |
| 8 | -0.020 | 0.121 | 0.019 | 0.019 | 0.161 |
| 9 | -5.838 | -3.949 | 0.629 | 1.122 | 0.828 |
| 10 | -5.838 | 3.949 | 0.629 | 1.122 | 0.828 |
| 11 | -2.161 | -4.689 | 0.746 | 0.822 | 0.419 |
| 12 | -2.161 | 4.689 | 0.746 | 0.822 | 0.419 |
| 13 | -0.567 | -15.881 | 2.528 | 2.529 | 0.036 |
| 14 | -0.567 | 15.881 | 2.528 | 2.529 | 0.036 |
| 15 | -1.395 | -20.294 | 3.230 | 3.237 | 0.069 |
| 16 | -1.395 | 20.294 | 3.230 | 3.237 | 0.069 |
| 17 | -4.585 | -29.126 | 4.636 | 4.693 | 0.156 |
| 18 | -4.585 | 29.126 | 4.636 | 4.693 | 0.156 |
| 19 | -13.067 | -57.888 | 9.213 | 9.445 | 0.220 |
| 20 | -13.067 | 57.888 | 9.213 | 9.445 | 0.220 |
| 21 | -13.126 | -58.143 | 9.254 | 9.487 | 0.220 |
| 22 | -13.126 | 58.143 | 9.254 | 9.487 | 0.220 |
| 23 | -18.779 | -65.115 | 10.363 | 10.786 | 0.277 |
| 24 | -18.779 | 65.115 | 10.363 | 10.786 | 0.277 |
| 25 | -18.928 | -71.090 | 11.314 | 11.708 | 0.257 |
| 26 | -18.928 | 71.090 | 11.314 | 11.708 | 0.257 |
| 27 | -18.860 | -71.501 | 11.380 | 11.769 | 0.255 |
| 28 | -18.860 | 71.501 | 11.380 | 11.769 | 0.255 |
| 29 | -14.331 | -92.384 | 14.703 | 14.879 | 0.153 |
| 30 | -14.331 | 92.384 | 14.703 | 14.879 | 0.153 |
| 31 | -15.418 | -97.420 | 15.505 | 15.698 | 0.156 |
| 32 | -15.418 | 97.420 | 15.505 | 15.698 | 0.156 |
| 33 | -18.318 | -116.175 | 18.490 | 18.718 | 0.156 |
| 34 | -18.318 | 116.175 | 18.490 | 18.718 | 0.156 |
| 35 | -20.186 | -119.821 | 19.070 | 19.339 | 0.166 |
| 36 | -20.186 | 119.821 | 19.070 | 19.339 | 0.166 |
| 37 | -20.497 | -122.560 | 19.506 | 19.777 | 0.165 |
| 38 | -20.497 | 122.560 | 19.506 | 19.777 | 0.165 |

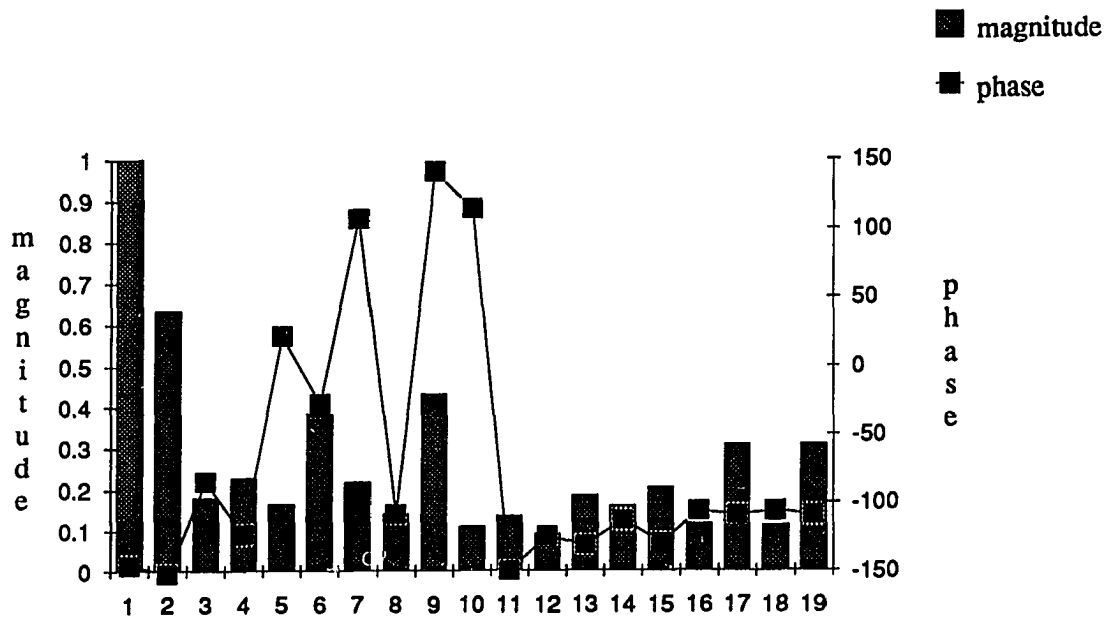



Figure 5.4: 5-Axle Eigenvector Associated with Eigenvalue 10

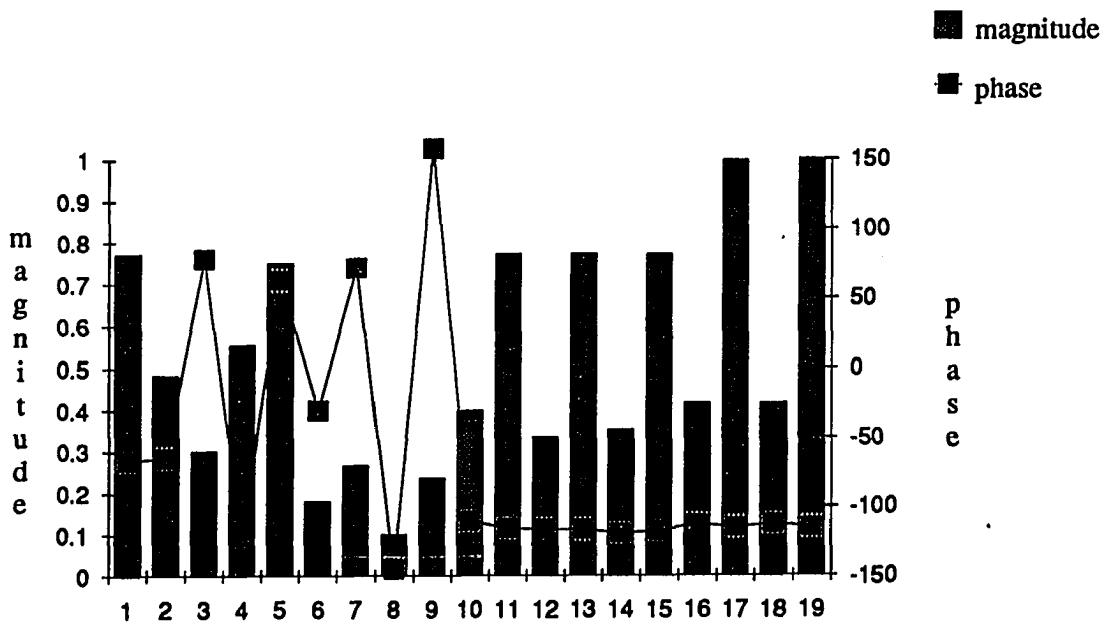



Figure 5.5: 5-Axle Eigenvector Associated with Eigenvalue 12

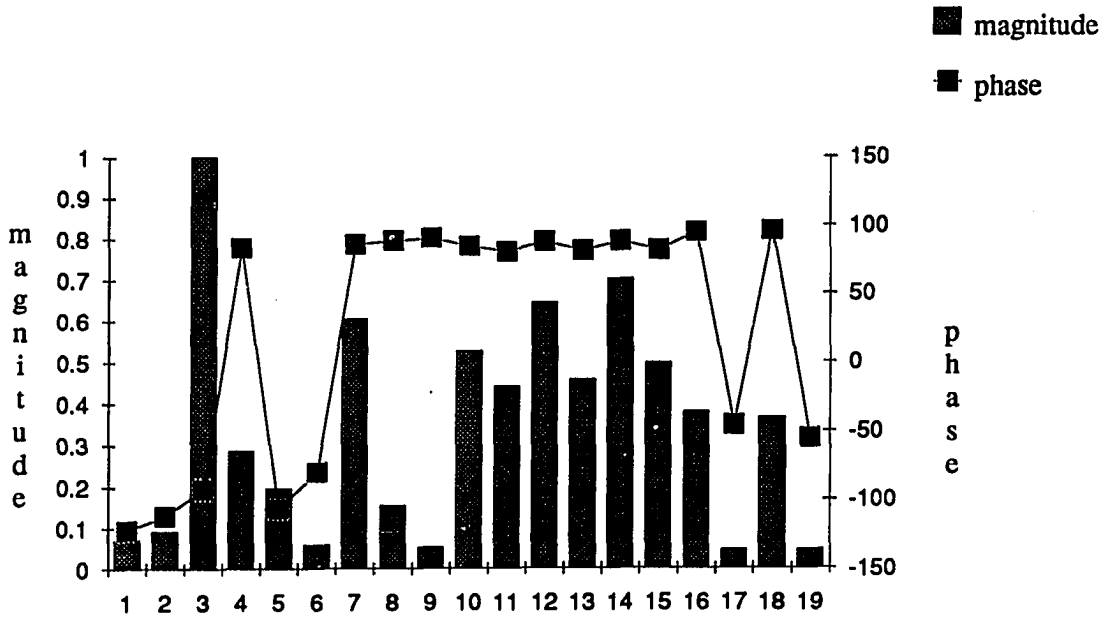



Figure 5.6: 5-Axle Eigenvector Associated with Eigenvalue 14

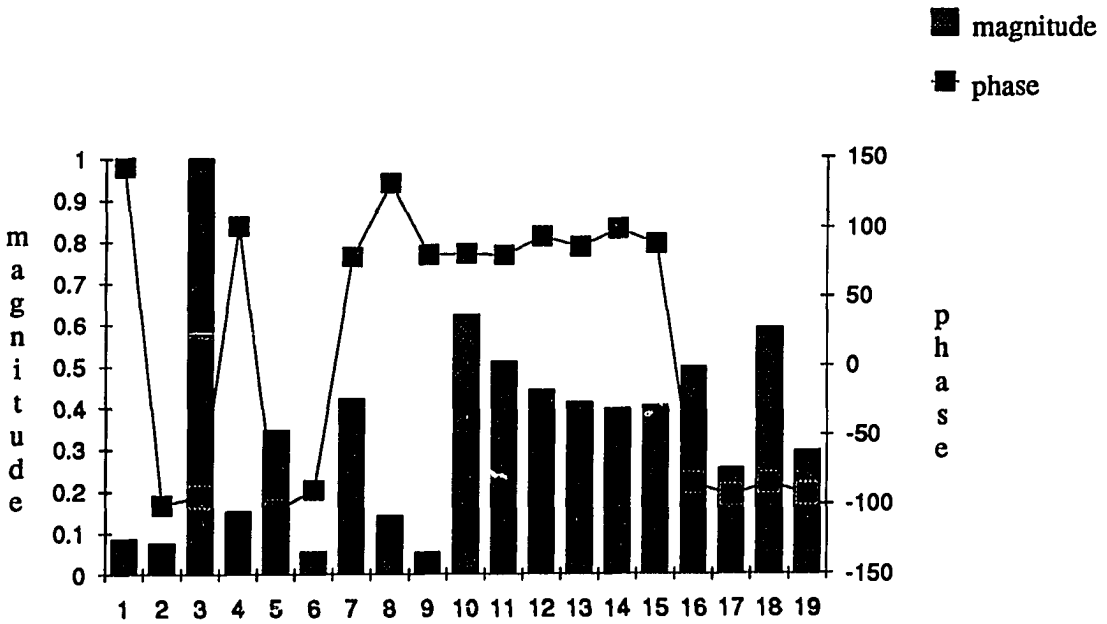



Figure 5.7: 5-Axle Eigenvector Associated with Eigenvalue 16

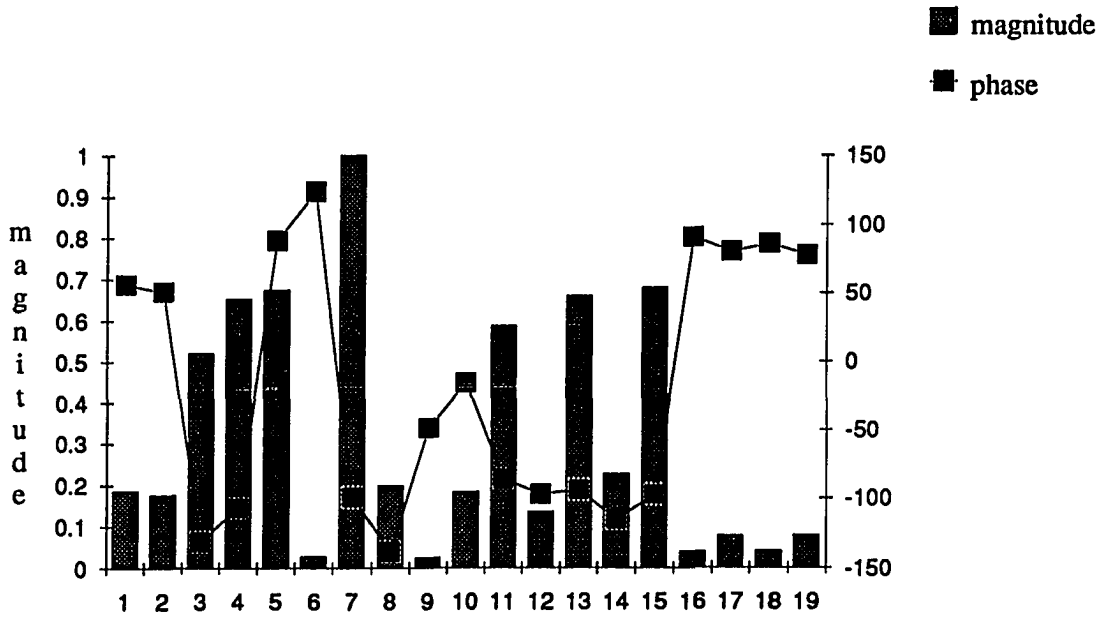



Figure 5.8: 5-Axle Eigenvector Associated with Eigenvalue 18

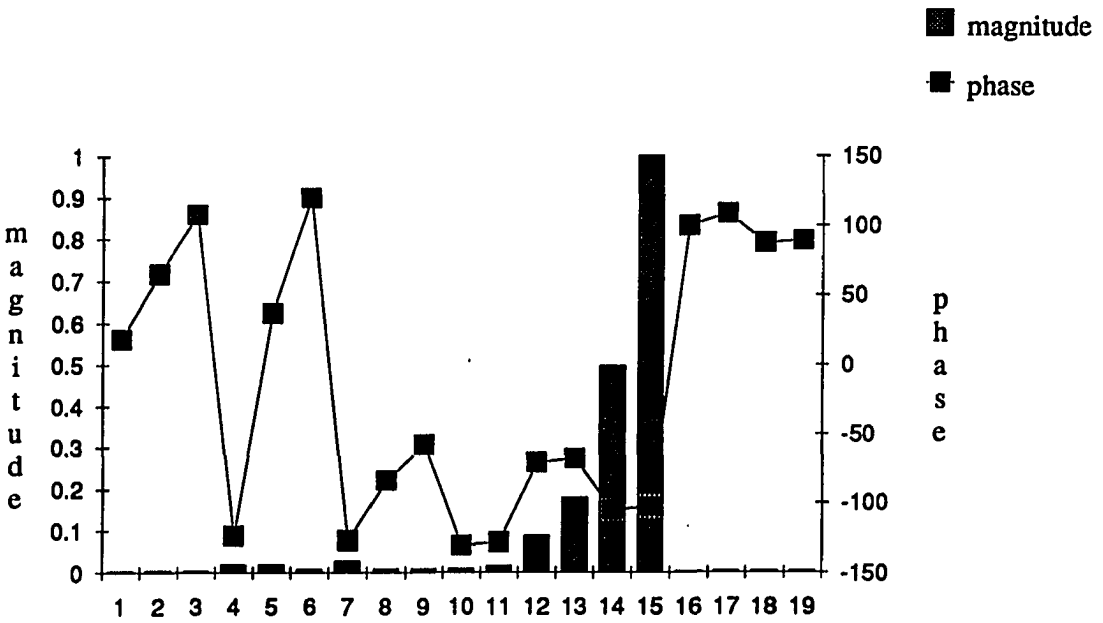



Figure 5.9: 5-Axle Eigenvector Associated with Eigenvalue 22

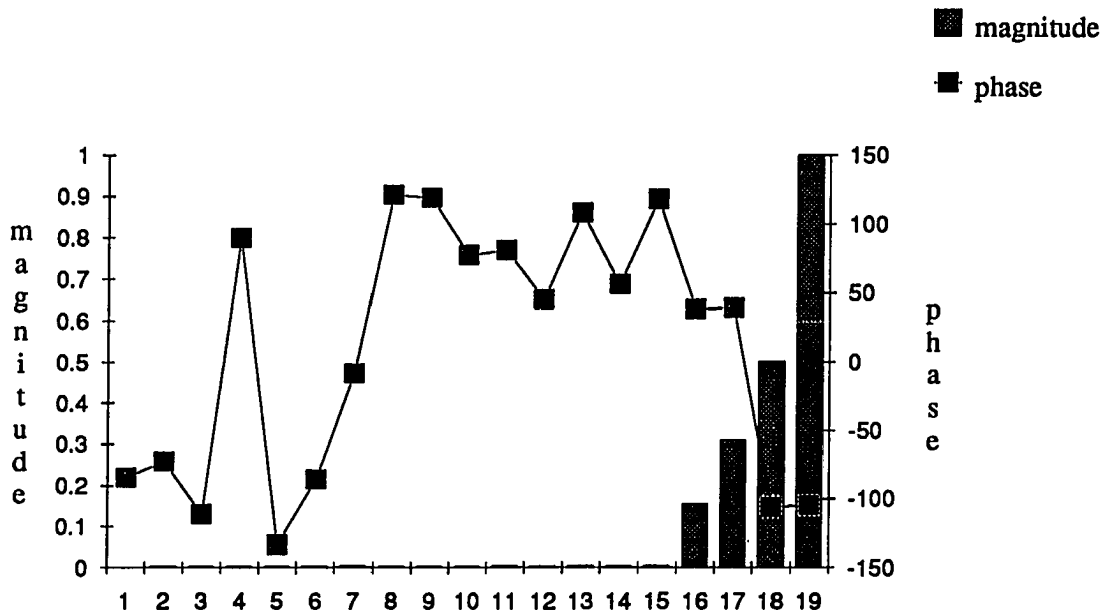



Figure 5.10: 5-Axle Eigenvector Associated with Eigenvalue 28

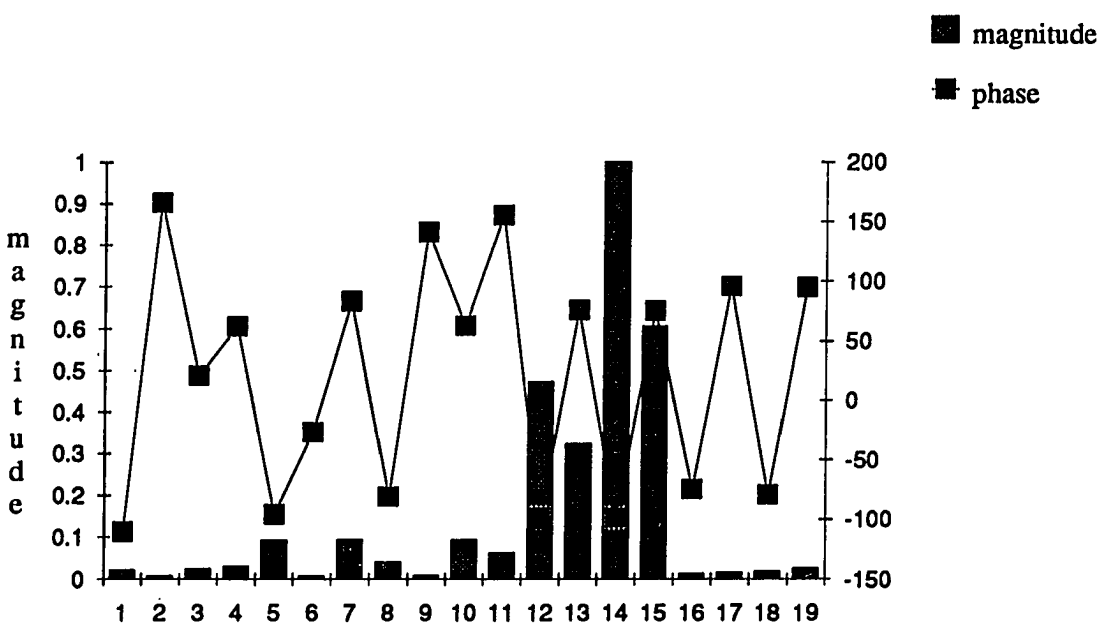



Figure 5.11: 5-Axle Eigenvector Associated with Eigenvalue 32

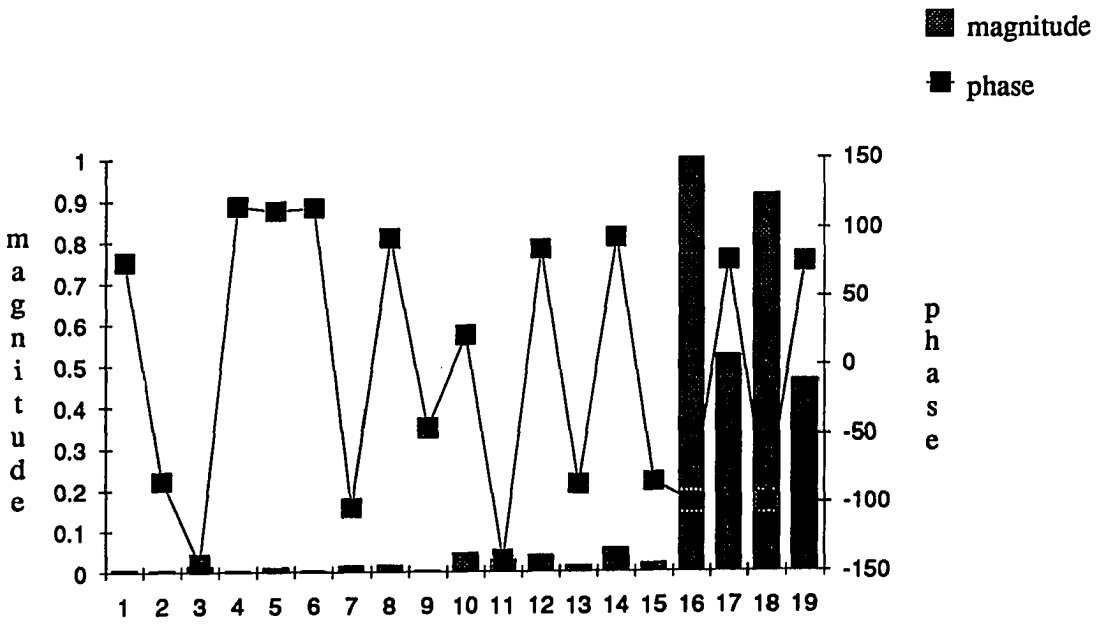



Figure 5.12: 5-Axle Eigenvector Associated with Eigenvalue 38

# 6. CONCLUSIONS

This thesis presented a new approach for linearization of large multibody dynamic systems. The approach uses an analytical differentiation of terms evaluated in a numerical equation formulation. Because the method is based on a relative coordinate formalism, it is more efficient than any finite difference method without the concern of determining the proper dithering values. This new linearization approach was generalized to include systems with closed-loops, damping, and steady motion with non-zero accelerations. A number of examples were presented to illustrate the accuracy and efficiency of the algorithm.

# 7. BIBLIOGRAPHY

[1] Sohoni, V. N. and Whitesell, J. "Automatic Linearization of Constrained Dynamic Models." *Journal of Mechanisms, Transmissions and Automation in Design*, 108, 300-304, 1986.

[2] Roberson, R. E., and Wittenburg, J. "A Dynamic Formalism for an Arbitrary Number of Interconnected Rigid Bodies, with Reference to the Problem of Satellite Attitude Control." Proc. 3rd IFAC Congress, London, 1966, paper 46D.

[3] Roberson, R. E. "Computer-Oriented Dynamical Modeling of Spacecraft: Historical Evaluation of Eulerian Mutibody Formalisms Since 1750." 28th Int. Astronaut. Congr., Prag, 1977, IAF paper 77-A11.

[4] Paul, B., and Krajcinovic, D. "Computer Analysis of Machines with Planar Motion - Part I: Kinematic; Part II: Dynamics." *Journal of Applied Mechanics*, Transactions of ASME, Ser. E., 37, 697-712, 1970.

[5] Chase, M.A., and Smith, D.A. "DAMN - A Digital Computer Program for the Dynamic Analysis of Generalized Mechanical Systems." SAE paper 710244, January, 1971.

[6] Sheth, P.N., and Uicker, J.J., Jr. "IMP (Integrated Mechanics Program), A Computer Aided Design Analysis System for Mechanisms and Linkages." *Journal of Engineering for Industry*, Transactions of ASME, Ser. B, 94, 454-464, 1972.

[7] Schiehlen, W.O. "Computer Generation of Equations of Motion." *Computer Aided Analysis and Optimization of Mechanical System Dynamics.* NATO ASI Series, Vol. F9. Berlin: Springer-Verlag, 1984.

[8] Kreuzer, E.J. and Schiehlen, W.O. "Generation of Symbolic Equations of Motion for Complex Spacecraft." AAS/AIAA Astrodynamics Specialist Conference, Lake Placid, NY, 1983, Paper 83-302.

[9] Wittenburg, J. and Wolz, U. "MESA VERDE: A Symbolic Program for Nonlinear Articulated Rigid Body Dynamics." ASME paper 85-DET-151, 1985.

[10] Rosenthal, D. and Sherman, M. "Symbolic Multibody Equations via Kane's Method." AAS/AIAA Astrodynamics Specialist Conference, Lake Placid, NY, 1983, Paper 83-303.

[11] Wehage, R.A. and Haug, E.J. "Generalized Coordinate Partitioning of Dimension Reduction Analysis of Constrained Dynamic Systems." *Journal of Mechanisms and Design*, Transactions of ASME, 104, 247-255, 1982.

[12] Nikravesh, P.E. and Chung, I.S. "Application of Euler Parameters to the Dynamic Reduction in Analysis of Constrained Dynamic Systems." *Journal of Mechanisms and Design*, Transactions of ASME, 104, 785-791, 1982.

[13] Orlandea, M.A., Chase, M.A., and Calahan, D.A. "A Sparsity-Oriented Approach to Dynamic Analysis and Design of Mechanical Systems, Parts I and II." *Journal of Engineering for Industry*, Transactions of ASME, 99, 773-784, 1977.

[14] Duffek, W., and Jaschinski, A. "Efficient Implemetation of Wheel-Rail Contact Mechnics in Dynamic Curving." *Vehicle Systems Dynaimcs*, 10, 184-192, 1981.

[15] Duffek, W., Fuhrer, C., Schwartz, W., and Wazllrapp, O. "Analysis and Simulation of Rail and Road Vehicles with the Program MEDYNA." 9th IAVSD Symposium on the Dynaimcs of Vehicles on Road and on Tracks, Linkoping, Sweden, 1985.

[16] Kortum, W. Schiehlen, W. "General Purpose Vehicle System Dynamics Software Based on Multibody Formalisms." *Vehicle Systems Dymanics*, 14, 229-263, 1985.

[17] Wittenburg, J. *Dynamics of Systems of Rigid Bodies.* B.G. Teubner, Stuttgart, 1977.

[18] Nikravesh, P.E. *Computer-Aided Analysis of Mechanical Systems.* Prentice-Hall, Englewood Cliffs, New Jersey, 1988.

[19] Wehage, R.A. "Quaternions and Euler Parameters - A Brief Exposition." The Advanced Institute of NATO Conference on Dynaimc Analysis and Dynamic Optimization of Mechanical Systems, Iowa City, Iowa, August, 1983.

[20] Kim, S.S. and Vanderploeg, M.V. "A State Space Formulation for Multibody Dynamic Systems Subject to Control." Technical Report No. 84-20, Center for Computer Aided Design, University of Iowa, Iowa City, Iowa, 1984.

[21] Greenwood, D.T. *Priciples of Dynamics.* Prentice-Hall, Englewood Cliffs, New Jersey, 1965.

[22] IMSL User's Manual. Version 1.0. IMSL Inc., Houston, TX, 1987.

[23] Timoshenko, Stephen. *Vibration Problems in Engineering.* John Wiley & Sons, New York, 1974, p. 342.

[24] Frazer, R.A., Duncan, W.J. & Collar, A.R. *Elementary Matrices.* The Cambridge University Press, London, 1965, p. 289.

[25] Meirovitch, Leonard. *Analytical Methods in Vibrations.* The Macmillan Company, New York, 1967, p. 410.

[26] Mani, N.K. "Use of Singular Value Decomposition for Analysis and Optimization of Mechanical System Dynamics." Ph.D. dissertation, The University of Iowa, 1984.

[27] Golub, G. H. "Numerical Methods for Solving Linear Least Square Problems." *Numerical Mathematics*, 7, 206-216, 1965.

[28] Lynch, A.G. "A Symbolic Fomulation for Equations of Motion of Multibody Systems." Ph.D. dissertation, Iowa State University, 1988.

[29] Moler, C. B. and Stewart, G. W. "An Algorithm for Generalized Matrix Eigenvalue Problems." *SIAM Journal of Numerical Analysis*, 10, 241-256, 1973.

[30] Atkinson, K. E. *An Introduction to Numerical Analysis.* John Wiley & Sons, New York, 1978.

[31] Fox, R.L. and Kapoor, M.P. "Rates of Change of Eigenvalues and Eigenvectors." *AIAA Journal*, 6, No. 12, 2426-2429, 1968.

[32] Thompson, R.B. "Derivatives of Frequency Response Peaks." M.S. Thesis, Michigan State University, 1983.

# 8. APPENDIX: OPTIMIZATION OF FREQUENCY RESPONSE CHARACTERISTICS

During a design procedure, it is often desirable to vary system parameters so as to change some frequency response characteristic of the system. This may include lowering the magnitude of the peak frequency responses, changing the natural frequencies themselves, or altering the relative motion of the system parts. To accomplish these changes, one or more design parameters must be dithered in a systematic way so as to implement the desired change without substantially altering the current system. Typically, an optimization algorithm of some type is use to accomplish this task. These algorithms require continuous knowledge of the frequency characteristic sensitivities with respect to design parameters. This involves repeated eigenanalysis of the linearized system as the optimization algorithm molds the system characteristics to the desired shape.

Because an accurate and efficient linearization technique is essential to an optimization problem of this type, the technique developed in this thesis is ideal for such optimizations. This appendix presents the fundamental sensitivity tools required for optimization of frequency response characteristics.

## 8.1 Eigenvalue Sensitivities

To determine the sensitivity of a complex eigenvalue with respect to a design parameter, the first order, homogeneous linearized system from Equation 3.47 is used.

$$\mathbf{M}_{aux}\dot{\mathbf{y}} + \mathbf{K}_{aux}\mathbf{y} = \mathbf{0} \tag{8.1}$$

From this point on, the auxiliary $\mathbf{M}_{aux}$ and $\mathbf{K}_{aux}$ matrices will be referred to simply as $\mathbf{M}$ and $\mathbf{K}$. Assume a solution to Equation 8.1 in the form

$$\mathbf{y} = \mathbf{u}_i e^{j\lambda_i t} \tag{8.2}$$

where $\lambda_i$ and $\mathbf{u}_i$ represent the $i$th eigenvalue and eigenvector of the system, respectively. Substituting Equation 8.2 into Equation 8.1 yields

$$\lambda_i \mathbf{M}\mathbf{u}_i + \mathbf{K}\mathbf{u}_i = \mathbf{0} \tag{8.3}$$

The rate of change of the $i$th eigenvalue with respect to a design variable, $\delta$, can be obtained by computing the partial derivative of this equation with respect to $\delta$ [31]

$$\frac{\partial \lambda_i}{\partial \delta} = \frac{\mathbf{u}_i^T \left( \frac{\partial \mathbf{K}}{\partial \delta} + \frac{\partial \mathbf{M}}{\partial \delta}\lambda_i \right) \mathbf{u}_i}{\mathbf{u}_i^T \mathbf{M} \mathbf{u}_i} \tag{8.4}$$

## 8.2 Eigenvector Sensitivities

Since the eigenvectors form a basis for the system motion, the partial derivative of an eigenvector with respect to a design parameter can be expressed as a linear combination of the eigenvectors

$$\mathbf{u}_i' = \sum_{j=1}^{2n} a_j \mathbf{u}_j \tag{8.5}$$

where $n$ is the number of second order differential equations in the system. Using Equation 8.5, the eigenvector sensitivities can be obtained by computing the partial derivative of Equation 8.3 with respect to $\delta$. This quantity is given by the expression [31]

$$a_g = \frac{\mathbf{u}_g{}^T \left( \mathbf{K}' - \lambda_i{}' - \lambda_i \mathbf{M}' \right) \mathbf{u}_i}{\lambda_i - \lambda_g} \tag{8.6}$$

## 8.3 Damping Ratio and Undamped Natural Frequency Sensitivities

In 1983, Thompson [32] extended eigenvalue sensitivity to include derivatives of damping ratios and undamped natural frequencies. From the definition of a damped system eigenvalue

$$\lambda_j = -\zeta_j \omega_j + i\omega_j \sqrt{1 - \zeta_j{}^2} \tag{8.7}$$

the damping ratio and undamped natural frequency can be obtained from the real and imaginary components of the partial differentiation of Equation 8.7 with respect to a design parameter

$$\frac{\partial \zeta}{\partial \delta} = \frac{-\sqrt{1 - \zeta_j{}^2} \left( \sqrt{1 - \zeta_j{}^2} Re\left( \frac{\partial \lambda_j}{\partial \delta} \right) - \zeta_j Im\left( \frac{\partial \lambda_j}{\partial \delta} \right) \right)}{\omega_j} \tag{8.8}$$

$$\frac{\partial \omega}{\partial \delta} = \frac{-\zeta_j \omega_j Re\left( \frac{\partial \lambda_j}{\partial \delta} \right) + \omega_j \sqrt{1 - \zeta_j{}^2} Im\left( \frac{\partial \lambda_j}{\partial \delta} \right)}{\omega_j} \tag{8.9}$$

With the sensitivity Equations 8.4, 8.6, 8.8 and 8.9, the frequency response characteristics of any system can be optimized.