# USE OF KINEMATIC REDUNDANCY FOR DESIGN OPTIMIZATION IN ROBOTS 

By<br>Yuyan Wang

B.Eng., Southeast University, Nanjing, China, 1985

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Master of Applied Science

> in

THE FACULTY OF GRADUATE STUDIES DEPARTMENT OF MECHANICAL ENGINEERING

We accept this thesis as conforming to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA
January 1994
(c) Yuyan Wang, 1994

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

## Department of Mechanical Engineering

The University of British Columbia
2075 Wesbrook Place
Vancouver, Canada
V6T 1W5

Date:
$\operatorname{Jan} 121994$


#### Abstract

For a manipulator mounted on a dynamic structure, as in space applications, the interactions with the robot and its supporting structure can play a crucial role. For example, stability and accurate control of space station operations are critical; therefore, methods that minimize the dynamic interactions (force and torque) between the space station and robot manipulators are needed. This thesis evaluates two such approaches, and extends these ideas to the design of space robots. General simulations of trajectory optimization schemes are conducted based on the redundancy model proposed by de Silva [1], [7], [8]. Furthermore, this optimization theory is extended to cover robot parameter design for a redundant manipulator. Two techniques are developed. First method is the "Joint Angle Manifold Approach", which uses a path integral to define the global cost function as the sum of the terms contributed by the reaction forces along the robot trajectory, which are described mathematically as different topological manifolds. The final cost function becomes a polynomial of the design parameters $\boldsymbol{l}_{\boldsymbol{i}}$. After calculating the cost function in each manifold, the one with the minimal value was selected for further optimization to search for the optimal link parameter solution. The second approach uses a global time integral to define the cost function as the sum of two terms; one term describing the time integral of the instantaneous base reaction cost function and the second term describing the peak value of all instantaneous cost function curves. The global cost function is then optimized against the design parameters which are link lengths $l_{i}$, using a numerical optimization method. Both methodologies are simulated for a three degree-of-freedom manipulator.


## Table of Contents

Abstract ..... ii
List of Tables ..... vi
List of Figures ..... vii
Acknowledgment ..... ix
Nomenclature ..... xi
1 Introduction ..... 1
1.1 Preliminary Remarks ..... 1
1.2 Literature Review ..... 3
1.3 Objectives and Contributions of the Research ..... 4
1.4 Scope of the Thesis ..... 5
2 Theory of Base Reaction Minimization ..... 7
2.1 Dynamics of Space Station and Manipulator ..... 7
2.2 Base Reaction Minimization ..... 11
2.3 End Effector Trajectory Generation ..... 14
2.4 Summary ..... 16
3 The Simulation Model of the NASA 7 d.o.f. Manipulator ..... 18
3.1 The Traction Drive Joint ..... 19
3.2 Kinematics ..... 21
3.2.1 Forward Kinematics ..... 22
3.2.2 Differential Kinematics ..... 28
3.3 Dynamics ..... 36
3.4 Summary ..... 44
4 Simulations of Base Reaction Optimization ..... 45
4.1 The Computer Simulation ..... 45
4.2 Simulation Results ..... 52
4.2.1 Simulation Results for the NASA Seven d.o.f. Robot ..... 52
4.2.2 Simulation Results for the Three d.o.f. Planar Robot ..... 54
4.3 Summary of Results ..... 57
5 Parameter Design Optimization ..... 64
5.1 Background ..... 64
5.2 Global Parameter Optimization ..... 66
5.3 Computer Simulation ..... 67
5.4 Results and Discussions ..... 68
5.4.1 Optimization Results ..... 68
5.4.2 Summary of Results ..... 72
6 The Trajectory Manifold Approach ..... 79
6.1 Introduction ..... 79
6.2 Manifold Formulation of Manipulator Kinematics ..... 81
6.2.1 Forword Kinematics ..... 81
6.2.2 Inverse Kinematics ..... 82
6.2.3 Jacobian Relations ..... 83
6.2.4 Manifold Solution for a 3-R Robot ..... 84
6.3 Design Parameter Optimization ..... 85
6.4 Computer Simulation ..... 87
6.5 Summary of Results ..... 90
7 Conclusion and Recommendations for Future Work ..... 94
7.1 Summary of the Thesis ..... 94
7.2 Major Contributions and Shortcomings of the Research ..... 95
7.3 Recommendations for Future Work ..... 97
7.4 Application of the Work ..... 97
Bibliography ..... 99
A Appendix ..... 102
A. 1 Kinematics and Dynamics of a Three d.o.f. Planar Robot ..... 102

## List of Tables

3.1 Denavit-Hartenberg Link Parameters ..... 27
4.2 Link Length Parameters for the Ten Cases in Figure 4.13 ..... 62

## List of Figures

2.1 A Space Station Based Robot ..... 8
2.2 Nomenclature for Dynamics of a Space Station and Robot ..... 10
3.1 The NASA Seven-Degree-of-Freedom Traction-Drive Manipulator ..... 20
3.2 Position Vector Representation in Coordinate Frames ..... 23
3.3 Joint Coordinate Frames for LTM ..... 26
3.4 The Relationship Between the Coordinate Frames of Base, $i-1$ th Joint and $i$ th Joint ..... 38
3.5 Schematic Diagram of Link $i$ ..... 40
3.6 Computational Steps for the Recursive Newton-Euler Equations ..... 42
4.1 The Structure of the Software Modules ..... 48
4.2 Flowchart for Trajectory Optimization Using Kinematic Redundancy ..... 49
4.3 Algorithm for Avoiding Singular Configuration Using Kinematic Redun- dancy ..... 50
4.4 End Effector Trajectory ..... 55
4.5 Base Reaction Cost Function ..... 56
4.6 Joint 1 Trajectory ..... 57
4.7 Joint 2 Trajectory ..... 58
4.8 Joint 3 Trajectory ..... 59
4.9 Joint 4 Trajectory ..... 60
4.10 Joint 5 Trajectory ..... 60
4.11 Joint 6 Trajectory ..... 61
4.12 Joint 7 Trajectory ..... 61
4.13 Optimal Cost Function for Different Link Lengths ..... 62
4.14 Optimal Cost Function with the First Link Length Decreased ..... 63
5.1 Algorithm for Design Optimizatiom of Robot Parameters ..... 69
5.2 A Three-Degree-of-Freedom Planar Robot Mounted on a Space Station ..... 70
5.3 End Effector Trajectory ..... 73
5.4 Optimal Local Cost Function Values for $l=\left[\begin{array}{lll}1 & 7 & 7\end{array}\right]$ (curve 1) and $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$ (curve 2) ..... 74
5.5 Joint 1 Trajectory for $l=\left[\begin{array}{ll}1 & 7 \\ 7 & 7\end{array}\right]$ (curve 1) and $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$ (curve 2) ..... 75
5.6 Joint 2 Trajectory for $l=\left[\begin{array}{ll}1 & 7 \\ 7\end{array}\right]$ (curve 1) and $l=\left[\begin{array}{ll}5 & 5\end{array} 5\right.$ (curve 2) ..... 76
5.7 Joint 3 Trajectory for $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$ ( curve 1 ) and $l=\left[\begin{array}{ll}5 & 5\end{array}\right.$ 5 (curve 2) ..... 77
6.1 A Three-Degree-of-Freedom Planar Robot and a Supporting Space Station ..... 88

## Acknowledgment

I would like to thank my supervisor, Dr. C. W. de Silva, for providing me the opportunity to carry out this research. His continuous help and advice is gratefully acknowledged.

This work has been funded through grants from the Natural Sciences and Engineering Research Council of Canada.

## THIS THESIS IS DEDICATED TO MY PARENTS.

## Nomenclature

| $\boldsymbol{R}_{\text {i }}^{\text {i-1 }}$ | Rotation matrix |
| :---: | :---: |
| $\boldsymbol{A}_{i}^{i-1}$ | D-H transformation matrix |
| $\boldsymbol{f}_{\boldsymbol{i - 1}}$ | Force vector at the i-1 th joint of the $i$ th link |
| $\boldsymbol{N}_{\boldsymbol{i}-1}$ | Moment (torque) vector at the i-1 th joint of the $i$ th link |
| $g_{\text {i }}$ | Vector respresenting the equivalent gravitational force per unit mass at the centroid of the $i$ th link |
| $m_{i}$ | Mass of the $i$ th link |
| $I_{c i}$ | Moment of inertial matrix of the $i$ th link about the centroid, and expressed in the body frame of the link |
| $l_{i}$ | Length of the $i$ th link. |
| $v_{\text {ci }}$ | Velocity of the centroid of the $i$ th link with respect to base frame |
| $\boldsymbol{v}_{\boldsymbol{i}}$ | The linear velocity at the origin of the coordinate frame attached to link $i$. |
| $\omega_{i}$ | Angular velocity of the $i$ th link expressed in the body frame of the link |
| $\boldsymbol{r}_{\boldsymbol{i}, \mathrm{ci}}$ | Position vector of the center of mass of link $i$ from the origin of the coordinate frame of the link. |
| $\boldsymbol{r}_{\text {ci }}$ | Position vector of the center of mass of link $i$ from the origin of the base referece frame. |
| $v_{0}$ | Rectilinear velocity of the space station with respect to the inertial frame |
| $\omega_{s}$ | Angular velocity of the space station with respect to the inertial frame |
| $M_{s}$ | Mass of the space station |
| $I_{s}$ | Inertial matrix of the space station |
| $q_{i}$ | The rotation about $z_{i-1}$-axis with respect to $i$ th joint |

in the coordinate system ( $x_{i-1}, y_{i-1}, z_{i-1}$ )
$\theta_{i}$

## $\boldsymbol{Q}$

$\boldsymbol{X}$

The rotation about $z_{i-1}$-axis with respect to $i$ th joint
in the coordinate system ( $x_{i-1}, y_{i-1}, z_{i-1}$ )
A set of $n$ joint coordinates.
A set of $m$ end-effector coordinates.

## Chapter 1

## Introduction

### 1.1 Preliminary Remarks

As the space technology advances, future space missions will progress to encompass many sophisticated tasks and operations; for example, operating scientific experiments such as constructing and repairing space stations, deploying satellites and delivering multi-phased payloads. Robots need to provide the flexibility, adaptability, programmability, self-reconfigurability and intelligence that would be neccessay for autonomous operation, and also require to meet engineering specifications such as the end effector speed, load levels and trajectory precisions.

Typical tasks of space robots would require extremely precise and accurate motion control. However, manipulator motion control is generally difficult to achieve because, for example, dynamic interaction resulting from the fact that manipulator motions are coupled with the space station produces unwanted disturbances on the system. Consequently, as the arm moves, it will exert dynamic forces and torques on the space station, causing the latter to move in a perturbed manner by changing its orbit and attitude. Since there is dynamic coupling between the robot and the space station, these disturbances also affect the performance of the manipulator. Hence, it is desirable to minimize, if not eliminate, the dynamic base reaction forces and moments through proper design and control of the robot manipulator.

There are many considerations in robot system design. Common criteria such as load capacity, load to mass ratio, low parts count and high modularity are important, but even more critical are considerations like large workspace, high dexterity and end effector dynamics. Preferably it should be a concurrent engineering process; in which the design, planning and developmental tasks are carried out in parallel. However, since many space missions are multi-tasked, it is difficult to compensate for all unwanted space interactions through physical design of the robot. Then, such disturbances have to be minimized by appropriate control of the robot and the space station. Nevertheless, the need for interaction compensation through control has to be reduced to a minimum so as to maintain controller simplicity and fuel efficiency.

Using the minimization of dynamic interactions as the central criterion for the robot system design process, the goal of this research is to develop a unified methodology for optimizing robot parameters for a particular task. Recently, there has been much emphasis in using path planning of redundant manipulators to achieve optimal performance under disturbances. This is however a limited approach; a general theory and methodology would be needed to extend the current solutions to include the geometric and physical parameters of a robot like link length and mass.

In order to implement the theory of design optimization, an accurate model of the dynamic interaction of a space robot and path optimization and planning techniques is required. Kinematic redundancies may be employed in this optimization. This research therefore looks into the aspects of developing a physical model for a robot manipulator in space operations; develops software programs to simulate robot motions and dynamics; and obtains fast and accurate optimization of the trajectory and design parameters for
complex space robot systems such as the NASA seven degree-of-freedom traction drive manipulator.

### 1.2 Literature Review

The earlier work on base reaction minimization was done by Vafa and Dubowsky [3], K.Yamada and K.Tsuchiya [6], and de Silva [1]. Vafa and Dubowsky considered the reduction of altitude variations for satellite bodies by path planning through the model of disturbance map; which shows the direction of the effect of joint velocities on the base station disturbance. It was essentially a mapping of the disturbance vector field for manipulator variables. Vafa and Dubowsky attempted to use a virtual manipulator method for the analysis of base station dynamic disturbance; this method offers the advantage of decoupling manipulator dynamics from the base station because of the selection of an appropriate coordinate frame of reference. Algorithms were also developed to use the disturbance map for trajectory planning in order to achieve minimal base reaction effects. The search for trajectory in disturbance map approach is an extremely time consuming task for high degree-of-freedom manipulators whose kinematics and dynamics are complex. Yamada and Tsuchyi considered the trajectory planning of a robot manipulator through the application of quantified cost function for the reaction forces of the base station. Analytical trajectory is achieved for a simple satellite model. De Silva uses kinematic redundancy to design manipulator trajectories so that base reaction is minimized; In his approach, a local quadratic positive definite cost function is employed to characterize the base disturbance. Unconstrained optimization is used to select the optimal end effector trajectory of the robot according to motion specifications such as acceleration and jerk limits.

### 1.3 Objectives and Contributions of the Research

The objective of this research is to develop a comprehensive methodology for the optimization of design parameters of a robot mounted on a dynamic structure. The research will be based on the foundation of the theory developed by de Silva for base reaction minimization through trajectory planning using kinematic redundancy. Computer simulations are developed using this theory for two systems; a three degree-of-freedom planar robot and a NASA seven-degree-of-freedom traction drive manipulator. The numerical simulations are used to demonstrate the application of the theory, which are expected to show significant reduction in the base reactions for both manipulators, through the use of kinematic redundancy. Since redundancy, in a given application only, allows the possibility for optimization of joint trajectories, the optimization model must be extended to include other geometric and kinematics design variables, such as link parameters, mass, and end effector velocity of the manipulator. Therefore, an important part of this research will be to develop numerical methods for global optimization; including a method of defining a global cost function, as well as methods for optimizing this complex and multi-valued cost function. The concluding task of the research will be to produce numerical simulations to verify the design parameter optimization methodology for a three-degree-of-freedom planar redundant manipulator.

The main contributions of this research are:

1. Development of computer simulations for a three-degree-of-freedom planar manipulator and a NASA seven-degree-of-freedom traction drive manipulator based on the theory developed by de Silva for trajectory planning. Numerical simulations are performed on a SUN SPARC station computer platform using MATLAB ${ }^{〔}$ optimization subroutines. These verify that base reaction minimization is possible for
different types of robots, and further predict that the base reaction forces ( cost function values ) are related to the geometric variables of the robot.
2. Development of optimization methodologies for design parameter optimization. Two approaches are developed. They are:
(a) Use of the topological manifold method to identify different trajectory flows, and to calculate the global cost function for the manipulator as the path integral along each trajectory. The cost function is symbolically estimated using first order terms, and the optimal design variables are calculated by optimization programs in MATLAB ${ }^{\text {© }}$.
(b) A numerically oriented procedure based on the early version of the path planning optimization is developed. This model uses the instantaneous cost function of the original method as the integrand to calculate the new global cost function; however, instead of optimization with respect to path parameters, numerical optimization with respect to the geometric parameters is achieved to obtain somewhat global, optimal design variables.

### 1.4 Scope of the Thesis

This section outlines the body of this thesis. Chapter 1 summarizes the objectives of the work, describes the reason for developing robot design optimization, presents the main results, and lists the contribution of the reseach. Chapter 2 introduces the trajectory optimization using kinematic redundancy and its application to base reaction minimization. The theory of de Silva and the techniques of using redundant degrees of freedom to optimize a cost function of the base reactions are reviewed. Chapter 3 is very important for understanding the remaining chapters. Here, the kinematic and dynamic
model of the NASA seven-degree-of-freedom robot manipulator is developed. In Chapter 4, computer simulations are carried out based on the model derived in Chapter 3. and the computational algorithm and the methodology of optimization are described. Chapter 5 forms the core of the design parameter optimization. In particarlly, the basis that leads to a global cost function definition and the methodology of nonlinear constainted optimization are outlined. A new approach for incorporating a global cost function in the optimization is introduced in Chapter 6. Here, the method of topological manifolds is used to describe the "self motion" of a redundant trajectory, and grouping them into finite number of submanifolds. This approach allows symbolic computation of the global cost function, thereby, enabling design optimization of parameters. Finally, the concluding chapter summarizes the design optimization problem and the theory and computer models developed for resolution of the problem, and provides recommendations for future research.

## Chapter 2

## Theory of Base Reaction Minimization

Before we formulate the theory of design optimization, we shall review the fundamental idea of base reaction minimization through the use of kinematic redundancy for trajectory planning as developed by de Silva [1], [7], [8]. There are four key steps in the approach:

1. Formulation of space-station manipulator dynamics
2. Quantification of the base reaction disturbances in terms of a scalar cost function
3. Resolution of the end effector position and orientation in terms of redundant joint angle variables
4. Optimization of the redundant joint coordinate to minimize the cost function values.

### 2.1 Dynamics of Space Station and Manipulator

Consider a space station that is orbiting around the earth in a circular orbit at a certain attitude. The open chain robot is rigidly mounted on the base station, as illustrated in Figure 2.1. The links are assumed to be rigid, and joint flexibility, friction, and backlash are assumed to be ideal. The inertial frame is located at the center of the earth. The center of mass of space station is located at $\boldsymbol{r}_{b}$ from the base of the robot.

The dynamic equations (both translational and rotational ) of the rigid space station


Figure 2.1: A Space Station Based Robot
are given by

$$
\begin{gather*}
f_{s}+f_{b}=\frac{d}{d t}\left(m_{s} v_{s}\right)  \tag{2.1}\\
N_{s}+N_{b}+r_{b} \otimes f_{b}=\frac{d}{d t}\left(I_{s} \omega_{s}\right)+\omega_{s} \otimes\left(I_{s} \omega_{s}\right) \tag{2.2}
\end{gather*}
$$

in which $f_{s}$ is the resultant external force acting at the centre of mass of the space station and $N_{s}$ is the resultant torque.

Consider the $i$ th link of a $n$ link manipulator as shown in Figure 2.2. The NewtonEuler equations for this link consist of the force-momentum equations:

$$
\begin{equation*}
\boldsymbol{f}_{i-1}-\boldsymbol{f}_{i}+m_{i} \boldsymbol{g}_{i}=\frac{d}{d t}\left(m_{i} \boldsymbol{v}_{c i}\right) \tag{2.3}
\end{equation*}
$$

and the moment-angular momentum equations about the centroid of the link:

$$
\begin{equation*}
\boldsymbol{N}_{i-1}-\boldsymbol{N}_{i}+\boldsymbol{r}_{\mathrm{c} i, i-1} \otimes \boldsymbol{f}_{i-1}-\boldsymbol{r}_{\mathrm{c} i, i} \otimes \boldsymbol{f}_{i}=\frac{d}{d t}\left(I_{c i} \omega_{i}\right)=I_{\mathrm{ci}} \dot{\omega}_{i}+\omega_{i} \otimes\left(I_{\mathrm{ci}} \omega_{i}\right) \tag{2.4}
\end{equation*}
$$

Since $f_{n}=0$ and $N_{n}=0$ for an open chain robot, by summing the $n$ equations given by equation (2.3) and (2.4), the bace-reaction forces and moments are obtained as follows:

$$
\begin{gather*}
f_{b}=\sum_{i=1}^{n} m_{i}\left(\dot{v}_{c i}-\boldsymbol{g}_{i}\right)  \tag{2.5}\\
\boldsymbol{N}_{b}=\sum_{i=1}^{n}\left[I_{c i} \dot{\omega}_{i}+\omega_{i} \otimes\left(I_{c i} \omega_{i}\right)+m_{i} \boldsymbol{r}_{i-1, c i} \otimes\left(\boldsymbol{v}_{c i}-\boldsymbol{g}_{i}\right)\right] \tag{2.6}
\end{gather*}
$$

Note that in these summations, the vectors are assumed to have been expressed in a common reference frame, through suitable coordinate transformations.

The required kinematic relations, specifically expressions for $\boldsymbol{\omega}_{i}, \dot{\omega}_{i}$ and $\boldsymbol{v}_{c i}$ in terms of the joint velocities $\dot{\boldsymbol{q}}$ and joint accelerations $\ddot{\boldsymbol{q}}$ are as follows:


Figure 2.2: Nomenclature for Dynamics of a Space Station and Robot

Angular velocities:

$$
\begin{array}{r}
\omega_{0}=\omega_{s}  \tag{2.7}\\
\omega_{i}=\omega_{i-\mathbf{1}}+\dot{q}_{i} \\
i=2, \ldots, n
\end{array}
$$

Angular Accelerations:

$$
\begin{array}{r}
\dot{\omega}_{o}=\dot{\omega}_{s}  \tag{2.8}\\
\dot{\omega}_{i}=\dot{\omega}_{i-1}+\ddot{q}_{i}+\omega_{i-1} \otimes \dot{q}_{i} \\
i=2, \ldots n
\end{array}
$$

Rectilinear velocities:

$$
\begin{array}{r}
\boldsymbol{v}_{o}=\boldsymbol{v}_{\boldsymbol{s}}+\boldsymbol{\omega}_{\boldsymbol{a}} \otimes \boldsymbol{r}_{\boldsymbol{o}}  \tag{2.9}\\
\boldsymbol{v}_{\boldsymbol{c i}}=\boldsymbol{v}_{\boldsymbol{i}}+\boldsymbol{\omega}_{\boldsymbol{i}} \otimes \boldsymbol{r}_{\boldsymbol{i}, \mathbf{c}} \\
\boldsymbol{v}_{\boldsymbol{i}}=\boldsymbol{v}_{\boldsymbol{i}-\mathbf{1}}+\boldsymbol{\omega}_{\boldsymbol{i}} \otimes \boldsymbol{r}_{\boldsymbol{i}-\mathbf{i}, \boldsymbol{i}} \\
i=1, \ldots . n
\end{array}
$$

Rectilinear accelerations:

$$
\begin{array}{r}
a_{o}=a_{s}+\dot{\omega}_{s} \otimes r_{o}+\omega_{s} \otimes\left(\omega_{s} \otimes r_{o}\right)  \tag{2.10}\\
\dot{v}_{c i}=\dot{v}_{i}+\dot{\omega}_{i} \otimes r_{i, c i}+\omega_{i} \otimes\left(\omega_{i} \otimes r_{i, c i}\right) \\
a_{i}=a_{i-1}+\dot{\omega}_{i} \otimes r_{i-1, i}+\omega_{i} \otimes\left(\omega_{i} \otimes r_{i-1, i}\right) \\
i=1, \ldots, n
\end{array}
$$

### 2.2 Base Reaction Minimization

Using the recursive Newton Euler dynamic formulation, the base reactions obtained above can be used to express the cost function. Specifically, the six dimensional vector of
the base reactions is mapped into a quadratic cost function, using the following measure:

$$
\begin{equation*}
\text { CostFunction }=\boldsymbol{R}^{T} Q \boldsymbol{R} \tag{2.11}
\end{equation*}
$$

in which the reaction vector $\boldsymbol{R}$ is given by :

$$
\begin{equation*}
R=\left[f_{b} N_{b}\right]^{T}-\left[f_{b}^{\circ} N_{b}^{\circ}\right]^{T} \tag{2.12}
\end{equation*}
$$

Here, the vectors $f_{b}^{\circ}$ and $N_{b}^{\circ}$ are the nominal reactions for a given configuration; and $Q$ is a positive definite weighting matrix, which also takes into consideration of the dimensional differeces in the vector $\boldsymbol{R}$.

Let us assume that the end-effector trajectory is expressed in the normal manner as a sixth order vector containing position and orientation of a body frame attached to the end effector; and expressed in the base frame. The incremental joint motion $\boldsymbol{\delta} \boldsymbol{q}$ that would be needed to produce a trajectory increment of $\delta \boldsymbol{y}$ is given as:

$$
\begin{equation*}
\delta y=J \delta q \tag{2.13}
\end{equation*}
$$

where $\boldsymbol{J}$ is the Jacobian matrix of the system.

The velocity of the end effector is given as:

$$
\begin{equation*}
\boldsymbol{v}=\boldsymbol{J} \dot{\boldsymbol{q}} \tag{2.14}
\end{equation*}
$$

and the acceleration relationship is:

$$
\begin{equation*}
a=\dot{\boldsymbol{J}} \dot{\boldsymbol{q}}+\dot{\boldsymbol{J}} \ddot{\boldsymbol{q}} \tag{2.15}
\end{equation*}
$$

The Jacobian matrix can be partitioned into:

$$
\begin{equation*}
J=\left[J_{n}, J_{r}\right] \tag{2.16}
\end{equation*}
$$

where $J_{n}$ is a nonsigular matrix corresponding to a set of independent joint coordinates $\boldsymbol{q}_{\boldsymbol{n}}$.
$\boldsymbol{J}_{\boldsymbol{r}}$ is a submatrix corresponding to a particular choice of redundant joint coordinates $\boldsymbol{q}_{\boldsymbol{r}}$.

Similarly, the vector is partitioned into

$$
\begin{equation*}
\boldsymbol{q}=\left[\boldsymbol{q}_{\boldsymbol{n}}, \boldsymbol{q}_{\boldsymbol{r}}\right] \tag{2.17}
\end{equation*}
$$

Thus, the velocity and acceleration equations are represented as:

$$
\begin{gather*}
\boldsymbol{v}=\boldsymbol{J}_{n} \dot{\boldsymbol{q}}_{n}+\boldsymbol{J}_{\boldsymbol{r}} \dot{\boldsymbol{q}}_{\boldsymbol{r}}  \tag{2.18}\\
\boldsymbol{a}=\dot{\boldsymbol{J}}_{n} \dot{\boldsymbol{q}}_{n}+\dot{\boldsymbol{J}}_{\boldsymbol{r}} \dot{\boldsymbol{q}}_{\boldsymbol{r}}+\boldsymbol{J}_{n} \ddot{\boldsymbol{q}}_{n}+\boldsymbol{J}_{\boldsymbol{r}} \ddot{\boldsymbol{q}}_{r} \tag{2.19}
\end{gather*}
$$

The vectors $\boldsymbol{v}$ and $a$ can be expressed in terms of curtate cycloidal motions, which may be used to limit both acceleration and jerk.

The joint variables and their derivatives are represented as follow:

$$
\begin{gather*}
\dot{\boldsymbol{q}}_{n}=J_{n}^{-1}\left(\boldsymbol{v}-J_{r} \dot{q}_{r}\right)  \tag{2.20}\\
\ddot{\boldsymbol{q}}_{n}=J_{n}^{-1}\left(\boldsymbol{a}-J_{r} \ddot{q}_{r}-\dot{J}_{n} \dot{\boldsymbol{q}}_{n}-\dot{J}_{r} \dot{\boldsymbol{q}}_{r}\right) \tag{2.21}
\end{gather*}
$$

Therefore, the cost function is represented in terms of the redundant variables $\boldsymbol{q}_{\boldsymbol{r}}$, which are expressed as a polynomial of time:

$$
\begin{equation*}
q_{r}=q_{0}+c_{1} t+c_{2} t^{2} \tag{2.22}
\end{equation*}
$$

This in turn means the cost function is depended on the parameters $c_{1}$ and $c_{2}$. The optimization scheme proceeds as follow: divide the trajectory into sufficiently large number of segments, and find the suitable values for the parametes $c_{1}$ and $c_{2}$ to produce a locally optimal solution for all joint coordinates. Such a method is repeated until the end of the task trajectory is reached.

### 2.3 End Effector Trajectory Generation

The motion for the space robot system is primarily dependent on the end effector trajectory. The need for engineering specifications on the acceleration and jerk at the end effector of the robot means trajectory kinematics must be appropriately designed. A curtate cycloidal time trajectory is used to meet such criteria [10]. For simplicity reasons in computer simulations, a linear segment is chosen to depict the motion of the end effector of a manipulator, and also the rotational motion is executed in synchronism for simulation purposes. An Euclidian rectilinear coordinate $y$ and an angular coodinate $\theta$ are chosen to represent the end effector position and orientation respectively.

In the curtate cycloidal motion, $y$ and $\theta$ are synchronized, and the velocity $v=\dot{y}$ and angular velocity $\omega=\dot{\theta}$ are represented parametrically as:

$$
\begin{align*}
& v(p)=b_{1}(1-\cos p)  \tag{2.23}\\
& \omega(p)=b_{2}(1-\cos p)  \tag{2.24}\\
& t(p)=a(p-c \sin p) \tag{2.25}
\end{align*}
$$

$$
a, b_{1}, b_{2}>0,1>c>0, \quad 2 \pi \geq p \geq 0
$$

Here $p$ is the parameter of the motion; corresponding to a scaled time variable. $T$ is the trajectory duration as given by

$$
\begin{equation*}
T=2 \pi a \tag{2.26}
\end{equation*}
$$

The position and orientation vectors of the trajectory are obtained by integrating the equations (2.23) and (2.24); thus

$$
\begin{align*}
& y(p)=a b_{1}\left[1+\left(\frac{c}{2}\right)\right] p-(c+1) \sin p+\left(\frac{c}{4}\right) \sin 2 p  \tag{2.27}\\
& \theta(p)=a b_{2}\left[1+\left(\frac{c}{2}\right)\right] p-(c+1) \sin p+\left(\frac{c}{4}\right) \sin 2 p \tag{2.28}
\end{align*}
$$

The final position and orientation at $t=T$ are

$$
\begin{align*}
& y(T)=a b_{1}\left[1+\frac{c}{2}\right] 2 \pi  \tag{2.29}\\
& \theta(T)=a b_{2}\left[1+\frac{c}{2}\right] 2 \pi \tag{2.30}
\end{align*}
$$

Rectilinear and angular accelerations $a$ and $\alpha$ are obtained by differentiating equations (2.23) and (2.24).

$$
\begin{align*}
& a(p)=\frac{b_{1} \sin p}{a(1-c \cos p)}  \tag{2.31}\\
& \alpha(p)=\frac{b_{2} \sin p}{a(1-c \cos p)} \tag{2.32}
\end{align*}
$$

The maximum acceleration values are

$$
\begin{equation*}
\left|a_{\max }\right|=\frac{b_{1}}{a \sqrt{\left(1-c^{2}\right)}} \tag{2.33}
\end{equation*}
$$

$$
\begin{equation*}
\left|\alpha_{\max }\right|=\frac{b_{2}}{a \sqrt{\left(1-c^{2}\right)}} \tag{2.34}
\end{equation*}
$$

Linear and angular jerks $\dot{\boldsymbol{a}}$ and $\dot{\alpha}$ are:

$$
\begin{align*}
& \dot{a}(p)=\frac{b_{1}}{a^{2}} \frac{(\cos p-c)}{(1-c \cos p)}  \tag{2.35}\\
& \dot{\alpha}(p)=\frac{b_{2}}{a^{2}} \frac{(\cos p-c)}{(1-c \cos p)} \tag{2.36}
\end{align*}
$$

These jerks remain finite throughout the trajectory as desired. There are four stationary values for each expression. However, the largest magnitude is

$$
\begin{align*}
& \left|\dot{a}_{\max }\right|=\frac{b_{1}}{a^{2}(1-c)^{2}}  \tag{2.37}\\
& \left|\dot{\alpha}_{\max }\right|=\frac{b_{2}}{a^{2}(1-c)^{2}} \tag{2.38}
\end{align*}
$$

These occur at the points with $p=0$ and $p=2 \pi$.

### 2.4 Summary

In this chapter, the theoretical framework of the base-reaction optimization of a robot manipulator mounted on a dynamic structure is presented from the point of view of kinematics and Newton-Euler dynamics. Specifically, the physical description of robotic motion and reaction forces is presented using Newton-Euler equations, expressed in a reference frame fixed to the space station. The fundamental variables are joint motions. Furthermore, the base reactions are defined as a positive-definite functional of the force and moment vectors. Using the Jacobian relationship between the joint velocities and the end-effector velocities, the force and moment equations are mapped in terms of the derivatives of the redundant joint varibles $\boldsymbol{q}_{\boldsymbol{r}}$. The optimization of the cost function is
expressed as a stepwise, local, parameter optimization procedure. Finally, a theoretical model for limiting the acceleration and jerk of the end effector is introduced using the curtate cycloidal motion.

## Chapter 3

## The Simulation Model of the NASA 7 d.o.f. Manipulator

As discussed and shown in Chapter 2, the differential kinematics of a robot manipulator can be described using its Jacobian. Furthermore, the dynamics of the robot and a dynamic supporting structure can be formulated using the Newton-Euler equations. In this manner, the complete mathematical relationship between the joint angle velocities, accelerations and the dynamic base reactions of $\boldsymbol{F}$ and $\boldsymbol{N}$ can be linked together. Subsequently, the optimization of base reactions (forces and moments) through trajectory planning may be carried out as formulated in Chapter 2. In order to carry out the optimization, an analytical model has to be developed in this manner, for the specific robot of interest. In this chapter, such a model is presented for a practical robot.

The NASA Langley Research Center sponsored and the Oak Ridge National Laboratory (ORNL) developed a Laboratory Telerobotic Manipulator (LTM) for space applications. This robot will serve as prototype that is used in this chapter for the computer simulation of the optimization scheme. As shown in Figure 3.1, the LTM arm has seven degrees of freedom that provide one degree of joint-space redundancy over that needed for the fundamental task of end-effector placement and orientation. The arm is configured from three common pitch/yaw joints which combine to provide shoulder, elbow, and wrist joints. Each of these joints has two degrees of freedom which are provided by a differential drive mechanism and is termed a traction drive joint, that has two outputs
providing pitch/yaw motions [11], [12]. This mechanism may be represented by two intersecting revolute joints. A wrist roll mechanism, mounted at the wrist joint, provides the seventh degree of freedom. Seven degrees of freedom allow the LTM to reorient itself without changing the end-effector position and orientation.

The theoretical background that is necessary to carry out a computer simulation of the LTM includes kinematics and dynamics. First, a traction drive joint is described. Then, the kinematics of the arm is presented. Finally, the equations of motion are derived.

### 3.1 The Traction Drive Joint

The joint mechanisms for conventional robots are usually gear transmissions, timing belts, sprockets and chain devices. Some others use the technique of harmonic drives. Neither of these mechanisms satisfies the requirement for accurate and repeatable operational specifications for teleoperating robot manipulators in space. In particular, the need for minimal backlash, high flexibility, high stiffness and smooth operations. The NASA Oak Ridge National Laboratory developed the traction drive joint mechanism for a seven d.o.f. manipulator. Traction drive is among the most elementary way of implementing speed changing mechanisms. It simply utilizes the application of the friction wheels of different dimension (diameter) to regulate speed. It can then be construct to give a single, fixed speed ratio, by controlling the rolling surfaces to engage at different radii. Using lubricant films, this joint poses better power transfer performance characteristics than others mechanisms because of minimal backlash.

For the specific case of seven d.o.f. LTM, the traction drives are located at the shoulder, elbow, wrist joints as well as another roll motion at the wrist to provide the extra


Figure 3.1: The NASA Seven-Degree-of-Freedom Traction-Drive Manipulator
degree of redundancy. At each joint, two input rollers are frictionally engaged with two other wheels of larger diameter dimension, forming a differential mechanism. The two differential wheels are also engaged with a output roller. This whole mechnism will produce two different independent vector motions pending on the motion of the two differntial wheels. When they are undergoing opposite motions with same speed, a roll motion is produced. When they are undergoing same direction motion at same speed, a pitch motion is produced about the axis of the output roller. The input rollers are of course controlled by dc servo motors, and any linear combination of pitch and yaw motion can be generated by varying the motion of input rollers.

The dynamic characteristics of the traction drive have been evaluated by de Silva and Hankins using controllability, observability, stability and response analysis [13]. Using state vector and system matrix analysis, the traction drive joint is found to be well behaved with the implementation of a linear quadratic regulator for servo control. In particular, satisfactory performance of the traction drive is possible with joint servo using roll and pitch angle feedback.

### 3.2 Kinematics

The forward-kinematics problem is to find the position and orientation of the endeffector of the maniputor given the joint angles $\theta$. The differential-kinematics problem is to find the corresponding joint volecities that cause the end-effector to move at the desired velocity. The inverse-kinematics problem is to determine the joint angles which will result in a desired end-effector position and orientations, typically given relative to and expressed in a coordinate frame fixed at the base. All of these relations have to be
determined for the LTM arm.

Differential relationships of manipulation are important in several ways. The most obvious is in the case of motions of accommodation, for instance, when a camera observes the manipulator end-effector position and calculates the differential changes in position and orientation in order to reach some goal. In this case, we should be able to convert defferential changes in one coordinate frame (measurement space) into changes in another ( the attraction or control space ). Another use of differential relationships follows directly, which is particularly applicable in this work. Given a differential change in the end-effector position and orientation, we will employ differential relationships to find the corresponding changes in the joint coordinates, since it is the joint velocities and accelerations that appear in the expressions of the base reactions. The Jacobian matrix, which corresponds to a set of linear equations, that has the differential change in position and orientation of the end effector as dependent variables, and the differential change in the joint coordinates as independent variables, as in equation (2.13).

### 3.2.1 Forward Kinematics

First, a means of relating positions on one link with respect to a fixed frame or to another moving link is established. In this manner, the position and orientation of the end-effector can be expressed in terms of the joint coordinates. The principal points are covered here. Details are found in references [15] and [16].

Given a point $P$ and two coordinate frames $F^{i-1}$ and $F^{i}$ as shown in Figure 3.2, the position of $P$ relative to $F^{i-1}$ is expressed as a vector $x^{i-1}$ in $F^{i-1}$, whereas the position of $P$ relative to $F^{i}$ is expressed as $\boldsymbol{x}^{i}$ in $F^{i}$. A superscript denotes which coordinate
system the vector is referred to.


Figure 3.2: Position Vector Representation in Coordinate Frames

The vectors $\boldsymbol{x}^{i-1}$ and $\boldsymbol{x}^{i}$ are related by

$$
\begin{equation*}
x^{i-1}=d^{i-1}+R_{i}^{i-1} x^{i} \tag{3.1}
\end{equation*}
$$

where $\boldsymbol{R}_{\boldsymbol{i}}^{i-1}$ is the matrix of coordinate transformation from Frame $i$ to Frame $i-1$ and $d^{i-1}$ is the distance between the origins of the two frames, expressed in Frame $i-1$. Note that

$$
\boldsymbol{R}_{i}^{i-1}=\left[\begin{array}{ccc}
\cos \theta_{i} & -\cos \alpha_{i} \sin \theta_{i} & \sin \alpha_{i} \sin \theta_{i}  \tag{3.2}\\
\sin \theta_{i} & \cos \alpha_{i} \cos \theta_{i} & -\sin \alpha_{i} \cos \theta_{i} \\
0 & \sin \alpha_{i} & \cos \alpha_{i}
\end{array}\right]
$$

where $\theta_{i}$ is the joint angle from $\boldsymbol{x}_{i-1}$ axis to the $\boldsymbol{x}_{i}$ axis about the $\boldsymbol{x}_{i-1}$ axis. $d_{i}$ is the distance from the orgin of the $i-1$ th coordinate frame to the intersection of the $\boldsymbol{z}_{\boldsymbol{i}-1}$ with the $\boldsymbol{x}_{\boldsymbol{i}}$ axis along the $\boldsymbol{x}_{\boldsymbol{i}-1}$ axis. $a_{i}$ is the offset distance from the intersection of the $\boldsymbol{z}_{i-1}$ axis with the $\boldsymbol{x}_{\boldsymbol{i}}$ axis to the origin of the $i$ th frame along the $\boldsymbol{x}_{\boldsymbol{i}}$ axis. $\alpha_{i}$ is
the offset angle from the $\boldsymbol{z}_{i-1}$ axis to the $\boldsymbol{z}^{i}$ axis about the $\boldsymbol{x}_{i}$ ( using the right-hand rule ).

Note that equation (3.1) represents a rotation and a translation. This transformation between $x^{i-1}$ and $x^{i}$ may be represented by a single matrix through the use of homogeneous vectors. An extra unity element is added to the vectors, so that given

$$
x=\left[\begin{array}{l}
x  \tag{3.3}\\
y \\
z
\end{array}\right]
$$

the vector $\boldsymbol{X}$ is defined

$$
\boldsymbol{X}=\left[\begin{array}{c}
x  \tag{3.4}\\
y \\
z \\
1
\end{array}\right]
$$

Note that, the original vectors $\boldsymbol{x}^{i-1}$ and $\boldsymbol{x}^{i}$ are augmented by adding a " 1 " as the fouth element so that the result is a $4 \times 1$ vector.

Now, the transformations are expressed as

$$
\begin{equation*}
X^{i-1}=A_{i}^{i-1} X^{i} \tag{3.5}
\end{equation*}
$$

where, the homogeneous transformation matrix is given by

$$
\boldsymbol{A}_{i}^{i-1}=\left[\begin{array}{cccc} 
& & & d_{x}  \tag{3.6}\\
& \boldsymbol{R}_{i}^{i-1} & & d_{y} \\
& & & d_{z} \\
0 & 0 & 0 & 1
\end{array}\right]
$$

By applying equation(3.5) successively, a vector $\boldsymbol{x}^{i}$ expressed in Frame $F^{i-1}$ may be given by a vector $\boldsymbol{X}$ in Frame $F^{0}$. Specifically, we have

$$
\begin{equation*}
\boldsymbol{X}=\boldsymbol{A}_{1}^{0} \boldsymbol{A}_{2}^{1} \ldots . \boldsymbol{A}_{i}^{i-1} \boldsymbol{X}^{i} \tag{3.7}
\end{equation*}
$$

The combined transformation which is a chain of $\boldsymbol{A}$ matrices is denoted by the $\boldsymbol{T}$ transformation matrix, as

$$
\begin{equation*}
\boldsymbol{X}=\boldsymbol{T}_{\boldsymbol{i}} \boldsymbol{X}^{\boldsymbol{i}} \tag{3.8}
\end{equation*}
$$

where

$$
\boldsymbol{T}_{i}=\boldsymbol{A}_{1}^{0} \boldsymbol{A}_{2}^{1} \ldots \boldsymbol{A}_{i}^{i-1}=\left[\begin{array}{cccc}
n_{x} & s_{x} & a_{x} & d_{x} \\
n_{y} & s_{y} & a_{y} & d_{y} \\
n_{z} & s_{z} & a_{z} & d_{z} \\
0 & 0 & 0 & 1
\end{array}\right]
$$

In order to relate positions and velocities at any point on each link of the robot to a fixed base frame, a coordinate frame is attached to each link. Denavit-Hartenberg convention [16] is used in this work for establishing coordinate frames. For a manipulator that has a revolute joint $i$, the joint variable $\theta_{i}$ is taken to describe the rotation of joint $\boldsymbol{i}$ about the $\boldsymbol{z}_{\boldsymbol{i - 1}}$ axis. The $\boldsymbol{x}_{\boldsymbol{i}}$ axis is chosen normal to both $\boldsymbol{z}_{\boldsymbol{i - 1}}$ and $\boldsymbol{z}_{\boldsymbol{i}}$ from the cross product $\boldsymbol{x}_{\boldsymbol{i}-1} \otimes \boldsymbol{z}_{\boldsymbol{i}}$. The intersection is the origin. The $\boldsymbol{y}_{\boldsymbol{i}}$ axis completes the coordinate axis triplet according to the right-hand rule, $\boldsymbol{y}_{\boldsymbol{i}}=\boldsymbol{x}_{\boldsymbol{i}} \otimes \boldsymbol{x}_{\boldsymbol{i}}$. The following parameters describe the relationship between neighbouring links:

- $a_{i}$ is the distance between $\boldsymbol{z}_{\boldsymbol{i}}$ and $\boldsymbol{z}_{\boldsymbol{i - 1}}$ measured along $\boldsymbol{x}_{\boldsymbol{i}}$
- $d_{i}$ is the distance between $\boldsymbol{x}_{\boldsymbol{i}}$ and $\boldsymbol{x}_{\boldsymbol{i}-1}$ measured along $\boldsymbol{x}_{\boldsymbol{i}-1}$
- $\alpha_{i}$ is the angle between $\boldsymbol{x}_{i-1}$ and $\boldsymbol{x}_{i}$ measured about $\boldsymbol{x}_{i}$, using the right hand rule.


Figure 3.3: Joint Coordinate Frames for LTM
The local coordinate frames of the LTM, established in this manner are shown in Figure 3.3. The Choice of coordinate frames is important, since the whole description of the behaviour of the robot, from trajectories to dynamics equations, rests on it. Table 3.1 gives the kinematic parameters for the LTM.

The next step is to obtain the $\boldsymbol{A}_{i}$ matrix from the choice of coordinate frames. Using the above definitions, a vector $\boldsymbol{X}^{i}$ expressed in link $i$ coordinate frame may be reexpressed in terms of link $i-1$ coordinate frame as $X^{i-1}$ by performing the following sequence of sub-transformations:

- first, rotate by an angle $\theta_{i}$ about $\boldsymbol{z}_{\boldsymbol{i - 1}}$ to bring the $\boldsymbol{x}_{\boldsymbol{i}}$ axis parallel to the $\boldsymbol{x}_{\boldsymbol{i - 1}}$ axis
- second, translate along $\boldsymbol{x}_{\boldsymbol{i - 1}}$ a distance so that $\boldsymbol{x}_{\boldsymbol{i}}$ and $\boldsymbol{z}_{\boldsymbol{i - 1}}$ coincide
- third, translate along $\boldsymbol{x}_{i}=\boldsymbol{x}_{i-1}$ a distance $a_{i}$ to bring the two origins together
- finally, rotate by an angle $\alpha_{i}$ about $x_{i}$ to bring the coordinate frames into coincidence.

Table 3.1: Denavit-Hartenberg Link Parameters

| $\theta_{i}$ | $d_{i}(\mathrm{~m})$ | $\alpha_{i}(\operatorname{deg})$ | $a_{i}(\mathrm{~m})$ | $\theta_{i}(\operatorname{deg})$ |
| :---: | :---: | :---: | :---: | :---: |
| $\theta_{1}$ | 0 | -90 | 0 | -90 |
| $\theta_{2}$ | 0 | 90 | $l_{2}$ | 0 |
| $\theta_{3}$ | 0 | 90 | 0 | 90 |
| $\theta_{4}$ | 0 | -90 | $l_{3}$ | 0 |
| $\theta_{5}$ | 0 | 90 | 0 | 0 |
| $\theta_{6}$ | 0 | 90 | 0 | 90 |
| $\theta_{7}$ | $d_{7}$ | 0 | 0 | 0 |

The result is the general matrix $\boldsymbol{A}_{\mathbf{i}}$ given by

$$
A_{i}^{i-1}=\left[\begin{array}{cccc}
\cos \theta_{i} & -\cos \alpha_{i} \sin \theta_{i} & \sin \alpha_{i} \sin \theta_{i} & a_{i} \cos \theta_{i}  \tag{3.9}\\
\sin \theta_{i} & \cos \alpha_{i} \cos \theta_{i} & -\sin \alpha_{i} \cos \theta_{i} & a_{i} \sin \theta_{i} \\
0 & \sin \alpha_{i} & \cos \alpha_{i} & d_{i} \\
0 & 0 & 0 & 1
\end{array}\right]
$$

For the LTM, the resulting $\boldsymbol{A}_{\boldsymbol{i}}$ matrices, $i=0,1, . .7$ are given below:

$$
\boldsymbol{A}_{0}^{b}=\left[\begin{array}{cccc}
1 & 0 & 0 & l_{1} \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{array}\right] \quad \boldsymbol{A}_{1}^{0}=\left[\begin{array}{cccc}
C_{1} & 0 & -S_{1} & 0 \\
S_{1} & 0 & C_{1} & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{array}\right]
$$

$$
\boldsymbol{A}_{3}^{2}=\left[\begin{array}{cccc}
C_{2} & 0 & S_{2} & l_{2} C_{2} \\
S_{2} & 0 & -C_{2} & l_{2} S_{2} \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{array}\right] \quad \boldsymbol{A}_{3}^{2}=\left[\begin{array}{cccc}
C_{3} & 0 & S_{3} & 0 \\
S_{3} & 0 & -C_{3} & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{array}\right]
$$

$$
\left.\begin{array}{ll}
\boldsymbol{A}_{4}^{3}=\left[\begin{array}{cccc}
C_{4} & 0 & -S_{4} & l_{4} C_{4} \\
S_{4} & 0 & C_{4} & l_{4} S_{4} \\
0 & -1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{array}\right] \\
\boldsymbol{A}_{6}^{5}=\left[\begin{array}{ccc}
C_{5} & 0 & S_{5} \\
0 \\
S_{5} & 0 & -C_{5} \\
0 \\
0 & 1 & 0 \\
0 \\
0 & 0 & 0
\end{array}\right]
\end{array}\right]
$$

where

$$
C_{i}=\cos \theta_{i}, S_{i}=\sin \theta_{i}, C_{i j}=\cos \left(\theta_{i}+\theta_{j}\right), S_{i j}=\sin \left(\theta_{i}+\theta_{j}\right)
$$

and, $l_{1}$ is the distance between $\boldsymbol{z}_{0}$ and $\boldsymbol{z}_{6}$ measured along $\boldsymbol{x}_{0}, \boldsymbol{l}_{2}$ is the distance between $\boldsymbol{z}_{2}$ and $\boldsymbol{z}_{1}$ measured along $\boldsymbol{x}_{2}, \boldsymbol{l}_{3}$ is the distance between $\boldsymbol{z}_{4}$ and $\boldsymbol{z}_{3}$ measured along $\boldsymbol{x}_{4}$, $d_{7}$ is the distance between $\boldsymbol{x}_{7}$ and $\boldsymbol{x}_{6}$ measured along $\boldsymbol{z}_{7}$.

### 3.2.2 Differential Kinematics

For differential kinematics, the main problem is to obtain the Jacobian matrix. Here we just briefly discuss some main principles. More details are found in reference [15] and [16].

In Figure 3.2, assume that point $P$ is fixed to Frame $F^{i}$. We have:

$$
\begin{equation*}
x_{i}^{i-1}=d_{i}^{i-1}+R_{i}^{i-1} x^{i} \tag{3.10}
\end{equation*}
$$

Note that the superscript indicates which coordinate system the vector is referred to.

The expression for the velocity is obtained by differentiating this expression with respect to time:

$$
\begin{equation*}
v_{i}^{i-1}=v_{d_{i}}^{i-1}+\dot{R}_{i}^{i-1} x^{i}+R_{i}^{i-1} v^{i} \tag{3.11}
\end{equation*}
$$

Note that the point $P$ is fixed with respect to Frame $F^{i}$. It can be shown that, for the present case,

$$
\begin{equation*}
v_{i}^{i-1}=v_{d_{i}}^{i-1}+\dot{R}_{i}^{i-1} R_{i}^{i-1 T} R_{i}^{i-1} x^{i} \tag{3.12}
\end{equation*}
$$

Note that the $\boldsymbol{R}_{\boldsymbol{i}}^{i-1}$ matrix is given by

$$
\boldsymbol{R}_{i}^{i-1}=\left[\begin{array}{ccc}
\cos \theta_{i} & -\cos \alpha_{i} \sin \theta_{i} & \sin \alpha_{i} \sin \theta_{i}  \tag{3.13}\\
\sin \theta_{i} & \cos \alpha_{i} \cos \theta_{i} & -\sin \alpha_{i} \cos \theta_{i} \\
0 & \sin \alpha_{i} & \cos \alpha_{i}
\end{array}\right]
$$

Differentiating $\boldsymbol{R}_{\boldsymbol{i}}^{i-1}$ with respect to time yields:

$$
\dot{\boldsymbol{R}}_{i}^{i-1}=\left[\begin{array}{ccc}
-\sin \theta_{i} & -\cos \alpha_{i} \cos \theta_{i} & \sin \alpha_{i} \cos \theta_{i}  \tag{3.14}\\
\cos \theta_{i} & -\cos \alpha_{i} \sin \theta_{i} & \sin \alpha_{i} \sin \theta_{i} \\
0 & 0 & 0
\end{array}\right] \dot{\theta}_{i}
$$

Mutiplying $\boldsymbol{R}_{i}^{i-1}$ and $\dot{\boldsymbol{R}}_{\boldsymbol{i}}^{i-1}$ yields:

$$
\dot{\boldsymbol{R}}_{i}^{i-1} \boldsymbol{R}_{i}^{i-1} T=\left[\begin{array}{ccc}
0 & -1 & 0  \tag{3.15}\\
1 & 0 & 0 \\
0 & 0 & 0
\end{array}\right]{\dot{\theta_{i}}}
$$

Hence, we note that $\dot{\boldsymbol{R}}_{\boldsymbol{i}}^{i-1} \boldsymbol{R}_{i}^{i-1}{ }^{\boldsymbol{T}} \boldsymbol{R}_{i}^{i-1}=\dot{\boldsymbol{R}}_{\boldsymbol{i}}^{i-1}$
For a revolute joint that rotates about $z$ axis without translation,

$$
\begin{equation*}
v_{d}^{i-1}=0 \tag{3.16}
\end{equation*}
$$

So, from equation (3.12), we have

$$
\begin{equation*}
\boldsymbol{v}_{i}^{i-1}=\left(\dot{\boldsymbol{R}}_{i}^{i-1} \boldsymbol{R}_{i}^{i-1 T}\right) \boldsymbol{R}_{i}^{i-1} \boldsymbol{x}^{i} \tag{3.17}
\end{equation*}
$$

$$
\boldsymbol{v}_{i}^{i-1}=\left[\begin{array}{ccc}
0 & -1 & 0  \tag{3.18}\\
1 & 0 & 0 \\
0 & 0 & 0
\end{array}\right] \dot{\boldsymbol{\theta}}_{i} \boldsymbol{R}_{i}^{i-1}[\boldsymbol{X}]_{i}
$$

If we consider $\boldsymbol{X}_{\boldsymbol{i}}$ as the origin point of the end effector frame $n$, measured with respect to frame $i$, equation (3.18) can be rewritten as follows:

$$
\begin{align*}
\boldsymbol{v}_{i}^{i-1} & =\left[\begin{array}{lll}
0 & -1 & 0 \\
1 & 0 & 0 \\
0 & 0 & 0
\end{array}\right] \dot{\theta}_{i} R_{i}^{i-1}\left[\begin{array}{c}
P_{x} \\
P_{y} \\
P_{z}
\end{array}\right]_{n}^{i}  \tag{3.19}\\
& =\left[\begin{array}{lll}
0 & -1 & 0 \\
1 & 0 & 0 \\
0 & 0 & 0
\end{array}\right] \dot{\theta}_{i}\left[\begin{array}{c}
P_{x} \\
P_{y} \\
P_{z}
\end{array}\right]_{n}^{i-1} \\
& =\left[\begin{array}{c}
-P_{y} \\
P_{x} \\
0
\end{array}\right]_{n}^{i-1} \dot{\theta}_{i}
\end{align*}
$$

We multiply the equation (3.19) by the appropriate $\boldsymbol{R}$ transformation matrices to express the end effector velocity in Frame $F^{i}$ with respect to the base coordinate frame. Specifically,

$$
\boldsymbol{v}_{i}^{0}=\boldsymbol{R}_{1}^{0} \boldsymbol{R}_{2}^{1} \ldots \boldsymbol{R}_{i-1}^{i-2}\left[\begin{array}{c}
-P_{y}  \tag{3.20}\\
P_{x} \\
0
\end{array}\right]_{n}^{i-1} \dot{\boldsymbol{\theta}}_{i}
$$

where,

$$
\boldsymbol{R}_{1}^{0} \boldsymbol{R}_{2}^{1} \ldots . \boldsymbol{R}_{i-1}^{i-2}=\left[\begin{array}{ccc}
n_{x} & s_{x} & a_{x}  \tag{3.21}\\
n_{y} & s_{y} & a_{y} \\
n_{z} & s_{z} & a_{z}
\end{array}\right]_{i-1}^{0}
$$

Hence,

$$
v_{i}^{0}=\left[\begin{array}{lll}
n_{x} & s_{x} & a_{x}  \tag{3.22}\\
n_{y} & s_{y} & a_{y} \\
n_{z} & s_{z} & a_{z}
\end{array}\right]_{i-1}^{0}\left[\begin{array}{c}
-P_{y} \\
P_{x} \\
0
\end{array}\right]_{n}^{i-1} \dot{\theta}_{i}
$$

From this equation, the translational velocity of the end-effector with respect to the base frame is found:

$$
\left[\begin{array}{c}
\dot{x}  \tag{3.23}\\
\dot{y} \\
\dot{z}
\end{array}\right]_{i}^{0}=\left[-\left[\begin{array}{l}
n_{x} \\
n_{y} \\
n_{z}
\end{array}\right]_{i-1}^{0}\left[P_{y}\right]_{n}^{i-1}+\left[\begin{array}{c}
s_{x} \\
s_{y} \\
s_{z}
\end{array}\right]_{i-1}^{0}\left[P_{x}\right]_{n}^{i-1}\right] \dot{\theta}_{i}
$$

The joint variable $\boldsymbol{\theta}_{\boldsymbol{i}}$ is defined to be a rotation about the $\boldsymbol{z}_{\boldsymbol{i}-1}$.

$$
\begin{align*}
& \boldsymbol{\theta}_{i}=\left[\begin{array}{l}
0 \\
0 \\
1
\end{array}\right] \boldsymbol{\theta}_{i}  \tag{3.24}\\
& \dot{\theta}_{i}=\left[\begin{array}{l}
0 \\
0 \\
1
\end{array}\right] \dot{\theta}_{i} \tag{3.25}
\end{align*}
$$

Then the angular velocity of the end-effector with respect to the base frame is obtained by multipling the corresponding $R$ trasformation matrices.

$$
\begin{align*}
{\left[\begin{array}{c}
\dot{\theta}_{x} \\
\dot{\theta}_{y} \\
\dot{\theta}_{z}
\end{array}\right]_{i}^{0} } & =\left[\begin{array}{lll}
n_{x} & s_{x} & a_{x} \\
n_{y} & s_{y} & a_{y} \\
n_{z} & s_{z} & a_{z}
\end{array}\right]_{i-1}^{0}\left[\begin{array}{l}
0 \\
0 \\
1
\end{array}\right] \dot{\theta}_{i}  \tag{3.26}\\
& =\left[\begin{array}{l}
a_{x} \\
a_{y} \\
a_{z}
\end{array}\right]_{i-1}^{0} \dot{\theta}_{i}
\end{align*}
$$

Then the total velocity relation is formed by combining equations (3.23) and (3.27). Specifically,

$$
\left[\begin{array}{c}
\dot{x}  \tag{3.27}\\
\dot{y} \\
\dot{z} \\
\dot{\theta}_{x} \\
\dot{\theta}_{y} \\
\dot{\theta}_{z}
\end{array}\right]_{i}^{0}\left[\begin{array}{c}
-\left[\begin{array}{c}
n_{x} \\
n_{y} \\
n_{z}
\end{array}\right]_{i-1}^{0}\left[P_{y}\right]_{n}^{i-1}+\left[\begin{array}{l}
s_{x} \\
s_{y} \\
s_{z}
\end{array}\right]_{i-1}^{0}\left[P_{x}\right]_{n}^{i-1} \\
{\left[\begin{array}{l}
a_{x} \\
a_{y} \\
a_{z}
\end{array}\right]_{i-1}^{0}}
\end{array}\right] \dot{\theta}_{i}
$$

The velocity relationship between frames can be converted into the differential form, which results in the well-known Jacobian matrix. The Jacobian matrix is a $6 \times n$ matrix consisting of differential translation and rotation vector elements.

$$
\left[\begin{array}{c}
d x  \tag{3.28}\\
d y \\
d z \\
d \theta_{x} \\
d \theta_{y} \\
d \theta_{z}
\end{array}\right]=J(\theta)\left[\begin{array}{c}
d \theta_{1} \\
d \theta_{2} \\
d \theta_{3} \\
\cdot \\
\cdot \\
\cdot \\
d \theta_{n}
\end{array}\right]
$$

where $J$ is given by the right-hand side matrix of equation(3.27)
Each column of the Jacobian matrix consists of the differential translation and rotation vectors corresponding to the differential changes of each joint coordinates. For the LTM, if $q_{1}$ is picked as the redundant degree of freedom, the Jacobian is found to be ( see equation (2.16) ):

$$
\begin{equation*}
\boldsymbol{J}=\left[\boldsymbol{J}_{n}, \boldsymbol{J}_{\boldsymbol{r}}\right] \tag{3.29}
\end{equation*}
$$

where the vector

$$
J_{r}=\left[\begin{array}{c}
C_{1} C_{2} C_{3} C_{4} C_{5} C_{6} d_{7}-S_{1} S_{3} C_{4} C_{5} C_{6} d_{7}+C_{1} S_{2} S_{4} C_{5} S_{6} d_{7}-C_{1} C_{2} S_{3} S_{5} S_{6} d_{7}- \\
S_{1} C_{3} C_{5} S_{6} d_{7}-C_{1} S_{2} C_{4} C_{6} d_{7}+C_{1} C_{2} C_{3} S_{4} C_{6} d_{7}-S_{1} S_{3} S_{4} C_{6} d_{7}- \\
l_{3} C_{1} C_{2} C_{3} C_{4}+l_{3} S_{1} S_{3} C_{4}-l_{3} C_{1} S_{2} S_{4}-l_{2} C_{1} C_{2}-l_{1} \\
-S_{1} C_{2} S_{3} S_{4} C_{5} S_{6} d_{7}-C_{1} S_{3} C_{4} C_{5} S_{6} d_{7}-S_{1} S_{2} S_{4} C_{5} S_{6} d_{7}+S_{1} C_{2} S_{3} S_{5} S_{6} d_{7}+ \\
C_{1} C_{3} S_{5} S_{6} d_{7}+S_{1} S_{2} C_{4} C_{6} d_{7}-S_{1} C_{2} C_{3} S_{4} C_{6} d_{7}-C_{1} S_{3} S_{4} C_{6} d_{7}+ \\
l_{3} S_{1} C_{2} C_{3} S_{4}+l_{3} C_{1} C_{3} C_{4}+l_{3} S_{1} S_{2} S_{4}+l_{2} S_{1} S_{2} \\
0 \\
0 \\
0
\end{array}\right]
$$

and the $6 \times 6$ square matrix

$$
\boldsymbol{J}_{n}=\left[\begin{array}{lllll}
\boldsymbol{J}_{n 1} & J_{n 2} & J_{n 3} & J_{n 4} & J_{n 5}  \tag{3.30}\\
J_{n 6}
\end{array}\right]
$$

where

$$
J_{n 1}=\left[\begin{array}{c}
C_{1} S_{2} C_{3} C_{4} C_{5} S_{6} d_{7}-C_{1} C_{2} S_{4} C_{5} S_{6} d_{7}-C_{1} S_{2} S_{3} S_{5} S_{6} d_{7}+C_{1} S_{2} C_{3} S_{4} C_{6} d_{7}- \\
C_{1} C_{2} C_{4} C_{6} d_{7}-l_{3} C_{1} S_{2} C_{3} C_{4}+l_{3} C_{1} C_{2} S_{4}-l_{2} C_{1} S_{2} \\
S_{1} S_{2} C_{3} C_{4} C_{5} S_{6} d_{7}-S_{1} C_{2} S_{4} C_{5} S_{6} d_{7}-S_{1} S_{2} S_{3} S_{4} C_{6} d_{7}+S_{1} S_{2} C_{3} S_{4} C_{6} d_{7}+ \\
S_{1} C_{2} C_{4} C_{6} d_{7}-l_{3} S_{1} S_{2} C_{3} C_{4}+l_{3} S_{1} C_{2} S_{4}-l_{2} S_{1} S_{2} \\
C_{2} C_{3} C_{4} C_{5} S_{6} d_{7}+S_{2} S_{4} C_{5} S_{6} d_{7}-C_{2} S_{3} S_{5} S_{6} d_{7}+C_{2} C_{3} S_{4} C_{6} d_{7}- \\
S_{2} C_{4} C_{6} d_{7}-l_{2} C_{2} C_{3} C_{4}-l_{3} S_{2} S_{4}-l_{2} S_{2} \\
-S_{1} \\
C_{1} \\
0
\end{array}\right]
$$

$$
\begin{gathered}
J_{n 2}=\left[\begin{array}{c}
C_{1} C_{2} S_{3} C_{4} C_{5} S_{6} d_{7}+S_{1} C_{3} C_{4} C_{5} S_{6} d_{7}-S_{1} S_{3} S_{5} S_{6} d_{7}+C_{1} C_{2} C_{3} S_{5} C_{6} d_{7}+ \\
S_{1} C_{3} S_{4} C_{6} d_{7}+C_{1} C_{2} S_{3} S_{4} C_{6} d_{7}-l_{3} C_{1} C_{2} S_{3} C_{4}-l_{3} S_{1} C_{3} S_{4} \\
S_{1} C_{2} S_{3} C_{4} C_{5} S_{6} d_{7}-C_{1} C_{3} C_{4} C_{5} S_{6} d_{7}-C_{1} C_{3} S_{4} C_{6} d_{7}+S_{1} C_{2} C_{3} S_{5} S_{6} d_{7}+ \\
S_{1} C_{2} S_{3} S_{4} C_{6} d_{7}+l_{3} C_{1} C_{3} C_{4}-l_{3} S_{1} C_{2} S_{3} C_{4} \\
-S_{2} C_{3} C_{4} C_{5} S_{6} d_{7}+S_{2} C_{3} C_{5} S_{6} d_{7}-S_{2} S_{3} S_{4} C_{6} d_{7}+l_{3} S_{2} S_{3} C_{4} \\
C_{1} S_{2} \\
S_{1} S_{2} \\
C_{2}
\end{array}\right] \\
J_{n 3}=\left[\begin{array}{c}
C_{1} C_{2} C_{3} S_{4} C_{5} S_{6} d_{7}-C_{1} S_{2} C_{4} C_{5} S_{6} d_{7}-S_{1} S_{3} S_{4} C_{5} S_{6} d_{7}-C_{1} C_{2} C_{3} C_{4} C_{6} d_{7}+ \\
S_{1} S_{3} C_{4} C_{6} d_{7}-C_{1} S_{2} S_{4} C_{6} d_{7}+l_{3} C_{1} S_{2} C_{4}+l_{3} S_{1} S_{3} S_{4} \\
S_{1} C_{2} C_{3} S_{4} C_{5} S_{6} d_{7}-S_{1} S_{2} C_{4} C_{5} S_{6} d_{7}+C_{1} S_{3} S_{4} C_{5} S_{6} d_{7}-S_{1} C_{2} C_{3} C_{4} C_{6} d_{7}- \\
C_{1} S_{3} C_{4} C_{6} d_{7}-S_{1} S_{2} S_{4} S_{6} d_{7}+l_{3} S_{1} S_{2} C_{4}-l_{3} C_{1} S_{3} S_{4}+l_{3} S_{1} C_{2} C_{3} S_{4} \\
-S_{2} C_{3} S_{4} C_{5} S_{6} d_{7}-C_{2} C_{4} C_{5} S_{6} d_{7}+S_{2} C_{3} C_{4} C_{6} d_{7}- \\
C_{2} S_{4} C_{6} d_{7}+L_{3} S_{2} C_{3} S_{4}+L_{3} C_{2} C_{4} \\
C_{1} C_{2} S_{3}+S_{1} C_{3} \\
S_{1} C_{2} S_{3}-C_{1} C_{3} \\
-S_{2} S_{3}
\end{array}\right]
\end{gathered}
$$

$$
\begin{aligned}
& J_{n 4}=\left[\begin{array}{c}
C_{1} C_{2} C_{3} C_{4} S_{5} S_{6} d_{7}+C_{1} C_{2} C_{3} C_{5} S_{6} d_{7}-S_{1} S_{3} C_{4} S_{5} S_{6} d_{7}+ \\
C_{1} S_{2} S_{4} S_{5} S_{6} d_{7}+S_{1} C_{3} C_{5} S_{6} d_{7} \\
S_{1} C_{2} C_{3} C_{4} S_{5} S_{6} d_{7}-C_{1} C_{3} C_{5} S_{6} d_{7}+S_{1} C_{2} C_{3} C_{5} S_{6} d_{7}+ \\
C_{1} S_{3} C_{4} S_{5} S_{6} d_{7}+S_{1} S_{2} S_{4} S_{5} S_{6} d_{7} \\
-S_{2} S_{3} C_{5} S_{6} d_{7}+C_{2} S_{4} S_{5} S_{6} d_{7}-S_{2} C_{3} C_{4} S_{5} S_{6} d_{7} \\
-C_{1} C_{2} C_{3} S_{4}+S_{1} S_{3} S_{4}+C_{1} S_{2} C_{4} \\
-S_{1} C_{2} C_{3} S_{4}-C_{1} S_{3} S_{4}+S_{1} S_{2} C_{4} \\
S_{2} C_{3} S_{4}+C_{2} C_{4}
\end{array}\right] \\
& J_{n 5}=\left[\begin{array}{c}
-C_{1} C_{2} C_{3} C_{4} C_{5} C_{6} d_{7}+C_{1} C_{2} C_{3} S_{4} S_{6} d_{7}+S_{1} S_{3} C_{4} C_{5} C_{6} d_{7}+C_{1} C_{2} S_{3} S_{5} C_{6} d_{7}- \\
C_{1} S_{2} S_{4} C_{5} C_{6} d_{7}-S_{1} S_{3} S_{4} S_{6} d_{7}-C_{1} S_{2} C_{4} S_{6} d_{7}+S_{1} C_{3} S_{5} C_{6} d_{7} \\
-S_{1} C_{2} C_{3} C_{4} C_{5} C_{6} d_{7}+S_{1} C_{2} C_{3} S_{4} S_{6} d_{7}-S_{1} S_{2} S_{4} C_{5} C_{6} d_{7}+S_{1} C_{2} S_{3} S_{5} C_{6} d_{7}- \\
C_{1} S_{3} C_{4} C_{5} C_{6} d_{7}-S_{1} S_{2} C_{4} S_{6} d_{7}+C_{1} S_{3} S_{4} S_{6} d_{7}-C_{1} C_{3} S_{5} C_{6} d_{7} \\
S_{2} C_{3} C_{4} C_{5} S_{6} d_{7}-C_{2} S_{4} C_{5} C_{6} d_{7}+S_{2} C_{3} S_{4} S_{6} d_{7}-C_{2} C_{4} S_{6} d_{7}+S_{2} S_{3} S_{5} C_{6} d_{7} \\
C_{1} C_{2} C_{3} C_{4} S_{5}-S_{1} S_{3} C_{4} S_{5}+C_{1} C_{2} S_{3} C_{5}+S_{1} C_{3} C_{5} \\
S_{1} C_{2} C_{3} C_{4} S_{5}+C_{1} S_{3} C_{4} S_{5}+S_{1} S_{2} S_{4} S_{5}+S_{1} C_{2} S_{3} C_{5}+C_{1} C_{3} C_{5} \\
-S_{2} C_{3} C_{4} S_{5}+C_{2} S_{4} S_{5}-S_{2} S_{3} C_{5}
\end{array}\right]
\end{aligned}
$$

$$
J_{n 6}=\left[\begin{array}{c}
0 \\
0 \\
0 \\
-C_{1} C_{2} C_{3} C_{4} C_{5} S_{6}+S_{1} S_{3} C_{4} C_{5} S_{6}-C_{1} S_{2} S_{4} C_{5} S_{6}+C_{1} C_{2} S_{3} S_{5} S_{6}+ \\
S_{1} C_{3} S_{5} S_{6}-C_{1} C_{2} C_{3} S_{4} C_{6}+S_{1} S_{3} C_{4} C_{6}+C_{1} S_{2} C_{4} C_{6} \\
-S_{1} C_{2} C_{3} C_{4} C_{5} S_{6}-C_{1} S_{3} C_{4} C_{5} S_{6}-S_{1} S_{2} S_{4} C_{5} S_{6}+S_{1} C_{2} S_{3} S_{5} S_{6}- \\
C_{1} C_{3} S_{5} S_{6}-S_{1} C_{2} C_{3} S_{4} C_{6}-C_{1} S_{3} S_{4} C_{6}+S_{1} S_{2} C_{4} C_{6} \\
S_{2} C_{3} C_{4} C_{5} S_{6}+C_{2} S_{4} C_{5} S_{6}-S_{2} S_{3} S_{5} S_{6}+S_{2} C_{3} S_{4} C_{6}+C_{2} C_{4} C_{6}
\end{array}\right]
$$

### 3.3 Dynamics

The Newton-Euler equations of motion are used for deriving the dynamic equations in this work mainly because of the computational efficiency that results from the associated recursive formulation.

The necessary mathematical relations are briefly described here. More details are found in reference [16] and [15].

In Figure 3.2, suppose that Frame $F^{i}$ is rotating and translating with respect to Frame $F^{i-1}$. Also suppose that a particle $p$ with mass $m$ is located by vectors $\boldsymbol{x}^{i}$ and $\boldsymbol{x}^{i-1}$ with respect to the origins of the coordinate frames $F^{i}$ and $F^{i-1}$. The origin of $F^{i}$ is given by a vector $\boldsymbol{d}^{i-1}$ with respect to the origin of Frame $F^{i-1}$.

Then, we have the relation between the position vectors $\boldsymbol{x}^{i}$ and $\boldsymbol{x}^{i-1}$ :

$$
\begin{equation*}
x^{i-1}=x^{i}+d^{i-1} \tag{3.31}
\end{equation*}
$$

where all vectors are assumed to have been expressed with same coordinate frame.
Furthermore, if we define $v^{i-1}$ and $v^{i}$ as the velocities of the particle $p$ reletive to the frames $F^{i-1}$ and $F^{i}$, and if $\boldsymbol{v}_{d}$ is the velocity of the frame $F^{i}$ relative to the frame $F^{i-1}$, the velocity of the particle $p$ with respect to the coordinate system $F^{i-1}$ is given by:

$$
\begin{gather*}
v^{i-1}=v^{i}+v_{d}  \tag{3.32}\\
\boldsymbol{v}^{i-1}=\frac{d x^{i}}{d t}+\frac{d d^{i-1}}{d t}=\frac{d^{*} x^{i}}{d t}+\omega \otimes x^{i}+\frac{d d^{i-1}}{d t} \tag{3.33}
\end{gather*}
$$

where $d^{*}() / d t$ is defined as time derivative with respect to the coordinate system $F^{i}$, and $\boldsymbol{\omega}$ is the angular velocity of this frame.

Similarly, the acceleration of the particle $p$ with respect to the coordinate frame $F^{i-1}$ is found to be:

$$
\begin{equation*}
\boldsymbol{a}^{i-1}=\frac{d^{2} x^{i}}{d t^{2}}+\frac{d^{2} d^{i-1}}{d t^{2}}=a^{i}+a_{d} \tag{3.34}
\end{equation*}
$$

where $a^{i-1}$ and $\boldsymbol{a}^{i}$ are the accelerations of the moving particle $p$ relative to the coordinate frames $F^{i-1}$ and $F^{i}$, and $a_{d}$ is the acceleration of Frame $F^{i}$ relative to Frame $F^{i-1}$. Accordingly,

$$
\begin{equation*}
a^{i-1}=\frac{d^{* 2} x^{i}}{d t^{2}}+2 \omega \frac{d^{*} x^{i-1}}{d t}+\omega \otimes\left(\omega \otimes x^{i}\right)+\frac{d \omega}{d t} \otimes x^{i}+\frac{d^{2} d^{i-1}}{d t^{2}} \tag{3.35}
\end{equation*}
$$

We wish to apply these results to link coordinate systems of the robot, to obtain the kinematic relationship of the moving links of a robot arm with respect to the base coordinate system.

Assume that the base coordinate system is $\left(x_{0}, y_{0}, z_{0}\right)$. The coordinate system ( $x_{i-1}$, $y_{i-1}, z_{i-1}$ ) with origin $O_{i-1}$ is attached to link $i-1$ and the coordinate system ( $x_{i}, y_{i}$, $z_{i}$ ) with origin $O_{i}$ is attached to link $i$, as shown in Figure 3.4. Origin $O_{i}$ is located by a position vector $\boldsymbol{p}_{\boldsymbol{i}}$ with respect to the origin $O$ and by a position vector $\boldsymbol{p}$ from the origin $O_{i-1}$ with respect to the base coordinate frame. Origin $O_{i-1}$ is located by a position vector $\boldsymbol{p}_{\boldsymbol{i}-1}$ from origin $O$ with respect to base coordinate frame.


Figure 3.4: The Relationship Between the Coordinate Frames of Base, i-1 th Joint and $i$ th Joint

Let $\boldsymbol{v}_{\boldsymbol{i - 1}}$ and $\boldsymbol{\omega}_{\boldsymbol{i - 1}}$ be the linear and angular velocities of the coordinate frame $F^{\boldsymbol{i - 1}}$ ( $x_{i-1}, y_{i-1}, z_{i-1}$ ) with respect to the base coordinate system. The angular velocity of the frame $F^{i}$ with respect to the frame $F^{i-1}$ is $\omega_{i}^{i-1}$.

Then, the linear velocity $\boldsymbol{v}_{i}$ and the angular velocity $\boldsymbol{\omega}_{i}$ of the coordinate frame $F^{i}$
$\left(x_{i}, y_{i}, z_{i}\right)$ with respect to the base coordinate system are given by:

$$
\begin{gather*}
\boldsymbol{v}_{i}=\frac{d^{*} \boldsymbol{p}}{d t}+\omega_{i-1} \otimes \boldsymbol{p}+\boldsymbol{v}_{i-1}=\omega_{i} \otimes \boldsymbol{p}+\boldsymbol{v}_{i-1}  \tag{3.36}\\
\omega_{i}=\omega_{i-1}+\omega_{i}^{i-1} \tag{3.37}
\end{gather*}
$$

Here $d^{*}() / d t$ denotes time derivative with respect to the frame $F^{i-1}$, and $\frac{d^{*} \boldsymbol{p}}{d t}=\omega_{i}^{i-1} \otimes \boldsymbol{p}$ since the joint is revolute and not prismatic. Also, $\boldsymbol{\omega}_{i}^{i-1}=\boldsymbol{x}_{i-1} \dot{q}_{i}$

The linear and the angular accelerations $\dot{v}_{i}$ and $\dot{\omega}_{i}$ of the coordinate system ( $x_{i}, y_{i}$, $z_{i}$ ) with respect to the base frame are:

$$
\begin{align*}
\dot{v}_{i} & =\frac{d^{* 2} p}{d t^{2}}+2 \omega_{i-1} \otimes \frac{d^{*} p}{d t}+\omega_{i-1} \otimes p+\omega_{i-1} \otimes\left(\omega_{i-1} \otimes p\right)+\dot{v}_{i-1}  \tag{3.38}\\
& =\dot{\omega}_{i} \otimes p+\omega_{i} \otimes\left(\omega_{i} \otimes p\right)+\dot{v}_{i-1}
\end{align*}
$$

$$
\begin{equation*}
\dot{\omega}_{i}=\dot{\omega}_{i-1}+\dot{\omega}_{i}^{i-1}+\omega_{i-1} \otimes \omega_{i}^{i-1} \tag{3.39}
\end{equation*}
$$

Here, $\frac{d^{\mu 2} p}{d t^{2}}=\frac{d \omega_{i}^{i-1}}{d t} \otimes p+\omega_{i}^{i-1} \otimes\left(\omega_{i-1}^{i} \otimes p\right)$ since the joint is revolute.

The dynamic equations of a rigid body can be represented by two equations; one describing the translational motion of the center of mass, called Newton's equation of motion for a mass particle, while the other describing the rotational motion about the center of mass, called Euler's equation of motion.

Let us consider an individual link $i$ as shown in Figure 3.5. All the forces and moments acting on the link are shown in the figure. Let $\boldsymbol{v}_{\mathrm{ci}}$ be the linear velocity of the center of mass of the link with respect to the base frame ( $x_{0}, y_{0}, z_{0}$ ), $\omega_{i}$ be the angular


Figure 3.5: Schematic Diagram of Link $i$
velocity of the link and $I_{c i}$ be the moment of inertia of the link with respect to center of mass.

The equation of translational motion is obtained by using Newton's second law,

$$
\begin{equation*}
f_{i-1}-f_{i}+m_{i} \boldsymbol{g}_{i}=\frac{d}{d t}\left(m_{i} v_{c i}\right) \tag{3.40}
\end{equation*}
$$

and the rotational motion is described by Euler's equation.

$$
\begin{equation*}
\boldsymbol{N}_{i-1}-\boldsymbol{N}_{i}+\boldsymbol{r}_{c i, i-1} \otimes \boldsymbol{f}_{i-1}-\boldsymbol{r}_{c i, i} \otimes \boldsymbol{f}_{i}=I_{c i} \dot{\omega}_{i}+\boldsymbol{\omega}_{i} \otimes\left(I_{c i} \boldsymbol{\omega}_{i}\right) \tag{3.41}
\end{equation*}
$$

where $\omega_{i}$ is the angular velocity of the body frame of the link.

The N-E equations of motion consist of a set of forward and backward recursive equations. Equations (3.36), (3.37), (3.38), and (3.39) form the forward equations, while equations (3.40) and (3.41) form the backward equations. For the forward equations,
linear velocity and acceleration, angular velocity and acceleration of each link, are computed from the base reference frame to the end effector. For the backward recursive equation, the moments and forces acting on each link are computed recursively from the end effector to base reference system. Hence, the forward equations compute the kinematics of each link from the base frame to the end effector, while the backward equations compute the necessary moments/forces for each joint from the end effector to the base reference frame. The computational algorithm is shown in Figure 3.6.

For the LTM, the necessary equations are summarized below:
Angular velocities:

$$
\begin{align*}
& \quad \omega_{0}=\omega_{\mathbf{s}}  \tag{3.42}\\
& \omega_{1}=\omega_{s}+z_{0} \dot{q}_{1} \\
& \omega_{2}=\omega_{1}+z_{1} \dot{q}_{2} \\
& \omega_{3}=\omega_{2}+z_{2} \dot{q}_{3} \\
& \omega_{4}=\omega_{3}+z_{3} \dot{q}_{4} \\
& \omega_{5}=\omega_{4}+z_{4} \dot{q}_{5} \\
& \omega_{6}=\omega_{5}+z_{5} \dot{q}_{6} \\
& \omega_{7}=\omega_{6}+z_{6} \dot{q_{7}}
\end{align*}
$$

Angular Accelerations:

$$
\begin{array}{r}
\dot{\omega}_{o}=\dot{\omega}_{s}  \tag{3.43}\\
\dot{\omega}_{1}=\dot{\omega}_{s}+\ddot{q}_{1}+\omega_{\mathrm{a}} \otimes z_{o} \dot{q}_{1} \\
\dot{\omega}_{2}=\dot{\omega}_{1}+\ddot{q}_{2}+\omega_{1} \otimes z_{1} \dot{q}_{2}
\end{array}
$$



Figure 3.6: Computational Steps for the Recursive Newton-Euler Equations

$$
\begin{aligned}
& \dot{\omega}_{3}=\dot{\omega}_{2}+\ddot{q}_{3}+\omega_{2} \otimes z_{2} \dot{q}_{3} \\
& \dot{\omega}_{4}=\dot{\omega}_{3}+\ddot{q}_{4}+\omega_{3} \otimes z_{3} \dot{q}_{4} \\
& \dot{\omega}_{5}=\dot{\omega}_{4}+\ddot{q}_{5}+\omega_{4} \otimes z_{4} \dot{q}_{5} \\
& \dot{\omega}_{6}=\dot{\omega}_{5}+\ddot{q}_{6}+\omega_{5} \otimes z_{5} \dot{q}_{6} \\
& \dot{\omega}_{7}=\dot{\omega}_{6}+\ddot{q}_{7}+\omega_{6} \otimes z_{6} \dot{q}_{7}
\end{aligned}
$$

## Linear accelerations:

$$
\begin{array}{r}
\dot{v}_{o}=\dot{v}_{a}+\dot{\omega}_{s} \otimes r_{o}+\omega_{s} \otimes\left(\omega_{s} \otimes r_{0}\right)  \tag{3.44}\\
\dot{v}_{1}=\dot{v}_{o}+\dot{\omega}_{1} \otimes r_{o, 1}+\omega_{1} \otimes\left(\omega_{1} \otimes r_{0,1}\right) \\
\dot{v}_{2}=\dot{v}_{1} \\
\dot{v}_{3}=\dot{v}_{2}+\dot{\omega}_{3} \otimes r_{2,3}+\omega_{3} \otimes\left(\omega_{3} \otimes r_{2,3}\right) \\
\dot{v}_{4}=\dot{v}_{3} \\
\dot{v}_{5}=\dot{v}_{4}+\dot{\omega}_{5} \otimes r_{4,5}+\omega_{5} \otimes\left(\omega_{5} \otimes r_{4,5}\right) \\
\dot{v}_{6}=\dot{v}_{5} \\
\dot{v}_{7}=\dot{v}_{6}+\dot{\omega}_{7} \otimes r_{6,7}+\omega_{7} \otimes\left(\omega_{7} \otimes\right) r_{6,7}
\end{array}
$$

## Linear accelerations of the centers of mass:

$$
\begin{array}{r}
\dot{v}_{c 1}=\dot{v}_{1}+\dot{\omega}_{1} \otimes r_{1, c 1}+\omega_{1} \otimes\left(\omega_{1} \otimes r_{1, c 1}\right) \\
\dot{v}_{c 2}=\dot{v}_{c 1} \\
\dot{v}_{c 3}=\dot{v}_{3}+\dot{\omega}_{3} \otimes r_{3, c 3}+\omega_{3} \otimes\left(\omega_{3} \otimes r_{3, c 3}\right) \\
\dot{v}_{c 4}=\dot{v}_{c 3} \\
\dot{v}_{c 5}=\dot{v}_{5}+\dot{\omega}_{5} \otimes r_{5, c 5}+\omega_{5} \otimes\left(\omega_{5} \otimes r_{5, c 5}\right)
\end{array}
$$

$$
\begin{array}{r}
\dot{\boldsymbol{v}}_{c 6}=\dot{\boldsymbol{v}}_{c 5} \\
\dot{\boldsymbol{v}}_{c 7}=\dot{\boldsymbol{v}}_{7}+\dot{\omega}_{7} \otimes \boldsymbol{r}_{7, c 7}+\boldsymbol{\omega}_{7} \otimes\left(\boldsymbol{\omega}_{7} \otimes \boldsymbol{r}_{7, c 7}\right)
\end{array}
$$

Note that all vectors here are expressed in the base coordinate frame.

The efficient, recursive, Newton-Euler equations of motion, which Luh et al. [14] improved are used. In their approach, all velocities, accelerations, inertial matrices, locations of the centers of mass of the links, and the forces and moments are referenced to the local coordinate systems, thereby improving the computational effeciency.

### 3.4 Summary

In this chapter, the theoretical framework of kinematics and dynamics of the NASA Seven-Degree-of-Freedom traction drive manipulator is presented. The kinematic equations of the robot are built based on the Denevit-Hartenberg convention. The differential kinematics of the LTM is characterized by Jacobian matrix, which was derived from the first principle. The dynamics of the LTM is described using the recursive Newton-Euler equations. All the dynamic variables derived are expressed in terms of vectors in the base coordinate frame. This chapter provides the fundamental mechanics for the modelling of dynamic interaction between a robot and its support structure.

## Chapter 4

## Simulations of Base Reaction Optimization

This chapter will describe the computer simulation process in applying the base reaction optimization technique that has been formulated in the previous chapters. The definition of an end effector path trajectory from the start point to the end point, the division of the trajectory into time interval steps, the computation of the joint kinematics at each interval using the inverse Jacobian and computing the instantaneous cost function at each interval, and the local optimization of the cost function at each interval using kinematic redundancy, constitute the main steps of the simulation process. The associated software modules and their structure are described. They are utilized to obtain joint trajectories for the seven d.o.f. NASA robot as well as a three d.o.f. planar revolute robot, that minimize a quadratic function in base reaction. The optimal results are compared with non-optimal baseline results. A preliminary design optimization with respect to robot parameters is also given.

### 4.1 The Computer Simulation

Based on the kinematic and dynamic model derived in the previous chapter, numerical software tools have been developed to show the feasibility of using kinematic redundancy for trajectory planning, with the objective of minimizing dynamic interaction of a robot and its support structure. Initially, a simple three degrees of freedom robot for planar motions is used. Subsequently, the current version of a NASA traction-drive
robot with seven degrees of freedom that is capable of complex operations in the three dimensional space is used. Finally, the three d.o.f. planar robot is used for parametric studies, aimed at design optimization.

The numerical simulation is developed using the MATLAB ${ }^{(\subset)}$ software [18] on a SUN SPARC station platform. The structure of the software modules is shown in Figure 4.1. The flowcharts for the simulation procedure and the algorithm for sigularity avoidance are given in Figure 4.2 and Figure 4.3 respectively. The MATLAB ${ }^{〔}$ program consists of nine main subroutines:

- initcost.m This routine determines the best initial joint configuration that has the minimal cost function value.
- optim.m This routine initialize the physical parameters of the program. Constants such as mass, vectors like initial position and orientation of the end effector, and arrays like the cost function are defined in this program. Also, This is the main program that contains the procedures for the optimization algorithm, including generating the end effector trajectory, and optimizing joint angles by calling other subroutines like Kinematic.m, costf.m and other.
- costf.m This is the key program that calculates the instantaneous cost function at any time along the trajectory using the Newton-Euler equations, the Jacobian matrix, and the local cost function.
- Kinematic. $m$ This routine contains the inverse kinematic equations, and it calculates the joint angles using the end effector position and orientation vector as the input.
- grad.m This routine contains the gradient function of the inverse kinematic relationship for the robot manipulator, and it is used for faster and more efficient computation in the it Kinematic.m module.
- Jnfunc.m This routine contains the functional equations for the normal joint Jacobian matrix.
- Jrfunc.m This routine contains the functional equations for the redundant joint Jacobian matrix.
- Jrdotfunc.m This routine contains the partial derivative of the jacobian matrix for the redundant joint variable.
- Jndotfunc.m This routine contains the partial derivative of the jacobian matrix for the normal joint variable.

The initcost.m carries out the initial optimization using an exhaustive search method to obtain initial configuration for the robot, that minimizes the cost function.

The main routine is optim.m and it performs optimization of the trajectory. It divides the end effector trajectory into many time intervals. For each interval, it calls the MATLAB ${ }^{(C)}$ OPTIMIZATION TOOL " fminu" to perform an optimization of the trajectory by selecting a set of parameters $c_{2}, c_{3}$, which corresponds to selecting the trajectory segment for the redundant joint, that would give a minimal value for the cost function.

The routine costf.m calculates the value of the cost function for a particular value of the redundant coordinate. The six main steps of the computational scheme are:


Figure 4.1: The Structure of the Software Modules


Figure 4.2: Flowchart for Trajectory Optimization Using Kinematic Redundancy


Figure 4.3: Algorithm for Avoiding Singular Configuration Using Kinematic Redundancy

1. The first joint coordinate $q_{1}$ is chosen as the redundant variable $q_{r}$. It is represented as a polynomial function of in $t$.

$$
q_{1}=c_{1}+c_{2} t+c_{3} t^{2}
$$

The parameters $c_{1}, c_{2}$ and $c_{3}$, which are the coefficients of the polynomial, are to be determined in the optimization process by calling function $c=$ fminu (' costf ', $\left[\begin{array}{ll}1 & 1\end{array}{ }^{\prime}\right.$ ); where [11] is the initial guess of the solution for $\left[c_{2}, c_{3}\right]$.
2. Given the values for coordinate $q_{r}$, the remaining kinematic variables are constrained by the kinematic equations of the manipulator. Thus, by solving the inverse kinematic problem numerially, we obtain the values for the remaining joint angles. MATLAB ${ }^{(\subset)}$ subroutines Kinematic.m and grad.m are used for such computations; Kinematic.m contains the inverse-kinematic equation. Also grad.m is the gradient matrix of the kinematic equation. MATLAB ${ }^{(\mathcal{C}}$ solves the equation using the OPTIMIZATION TOOL" fsolve2".
3. The Jacobian matrix for a seven-degree-of-freedom manipulator is very complex. The functional form of each of the matrix elements has been derived from first principles. They are listed in the programs Jnfunc.m and Jrfunc.m. Their corresponding derivatives are derived and listed in Jndotfunc.m and Jrdotfunc.m respectively.
4. The dynamic variables of $\dot{q}_{1}, \dot{q}_{n}, \ddot{q}_{1}$, and $\ddot{q}_{n}$ are then calculated using the Jacobian equations along with the values of the end effector velocity and acceleration, and Jocobian matrix $J_{n}$ and $J_{r}$. Next the determinent of $J_{n}$ is checked. If it is zero, $\dot{q}_{1}$ is replaced with another coordinate in $\dot{q}_{n}$, and return to step 1 . Else the steps are continued.
5. The joint variables $v_{1}, v_{n}, \omega_{1}$ and $\omega_{n}$ and their time derivatives are calculated using the dynamic equations.
6. The force and torque at the base point are calculated using the Newton-Euler equations.
7. Finally for the given weighting matrix $Q$ the Cost Function is calculated .

$$
\operatorname{cost}=\boldsymbol{R}^{\boldsymbol{T}}(Q \boldsymbol{R})
$$

### 4.2 Simulation Results

### 4.2.1 Simulation Results for the NASA Seven d.o.f. Robot

First consider the seven-degree-of-freedom NASA-ORNL traction-drive robot. In this simulation, the space station is assumed to be orbiting the earth at geosynchronous period of 24 hours; this means that the circular orbital radius is approximately $R_{0}=$ 42330 km . The tangential velocity of the space station is calculated using the conservation of kinetic and potential energy.

$$
\begin{gather*}
\frac{1}{2} m v_{s}^{2}=\frac{G m M_{e}}{R_{0}}  \tag{4.1}\\
v_{s}=\sqrt{\frac{2 G M_{e}}{R_{0}}} \tag{4.2}
\end{gather*}
$$

where $M_{e}$ is the mass of the earth, and $G$ is the universal gravitational constant.
Also, from simple kinematics,

$$
\begin{equation*}
\boldsymbol{\omega}_{s}=\frac{\boldsymbol{v}_{\mathbf{s}}}{\boldsymbol{R}_{\mathbf{0}}} \tag{4.3}
\end{equation*}
$$

The space station parameters values used are:
Moment of inertia about the axis of rotation $I_{a}=19.7 \times 10^{6} \mathrm{~kg} \cdot \mathrm{~m}^{2}$
Mass $M_{s}=18.0 \times 10^{3} \mathrm{~kg}$

The link parameters values used are assumed identical links:
Link length $l_{i}=5.0 \mathrm{~m}$
Link mass $m=20.0 \mathrm{~kg}$
Link moment of inertia about the joint axis $I_{i}=200.0 \mathrm{~kg} \cdot \mathrm{~m}^{2}$

The reference frame is located at the centre of mass of the space station. The end effector is assumed to move along a straight line from the point (7.5, $6.5,6.5$ ) to (7.2, 5.5, 6.2), and it is assumed to be rotated to the appropriate orientation at the end point of the trajectory. The distance travelled is $y=\sqrt{(7.5-7.2)^{2}+(6.5-5.5)^{2}+(6.5-6.2)^{2}}=1.09$ m. Assume that the time duration of motion $T$ (equation (2.26)) is 2.0 seconds, the acceleration limit $a$ (equation (2.33) ) is $10 \mathrm{~m} / \mathrm{s}^{2}$ and the angular acceleration limit $\alpha$ ( equation (2.34)) is $10 \mathrm{rad} / \mathrm{s}^{2}$. Then the curtate cycloidal parameters $a, b_{1}, b_{2}$ and $c$ (see equations (2.23) through (2.25) ) are obtained for this specific end effector trajectory, as $a=0.31831, b_{1}=0.3597, b_{2}=0.0297$ and $c=0.9934$. The linear jerk limit ( equation (2.37) ) and the angular jerk limit (equation (2.38)) are calculated as $\dot{a}=814.98 \mathrm{~m} / \mathrm{s}^{3}$ and $\dot{\alpha}=672.93 \mathrm{rad} / \mathrm{s}^{3}$, using equations (2.37) and (2.38) respectively.

The end effector trajectory is divided into 200 intervals. During the initial step, the optimal joint configuration of the robot is determined by carryiny out an optimization ( an exhaustive search method) as, $q_{1}=-2.2605 \mathrm{rad}, q_{2}=1.1250 \mathrm{rad}, q_{3}=1.5507 \mathrm{rad}$, $q_{4}=2.0675 \mathrm{rad}, q_{5}=-0.6290 \mathrm{rad}, q_{6}=1.2293 \mathrm{rad}$, and $q_{7}=-0.1662 \mathrm{rad}$. Next, the unconstrained optimization is carried out for each successive interval of the trajectory to obtain the optimal joint trajectory.

A nonoptimal case is also simulated for the same end effector trajectory, by selecting an arbitrary trajectory for $q_{r}$ by assigning the values for $c_{1}=1$ and $c_{2}=1$ for the
ploynomial coefficients; and then obtaining $\boldsymbol{q}_{\boldsymbol{n}}$ and other variables to satisfy the specified end effector trajectory.

The results are shown in Figures 4.4 through 4.12. These figures compare the optimal case with the nonoptimal case.

Figure 4.4 shows the end effector trajectory for both the optimal and the nonoptimal cases, which are constrained to be identical. The time interval is divided into small enough steps so that there would be no deviation present due to numerical errors.

Figure 4.5 shows the optimal and the nonoptimal cost functions as functions with respect to trajectory time. The relative magnitude of the optimal cost function is much lower in comparision to that of the nonoptimal one, thereby indicating the potential reduction of the dynamic interaction through trajectory optimization using the presented approach. The corresponding joint trajectories ( $q_{1}$ to $q_{7}$ ) are shown in Figures 5.5 through 4.12.

### 4.2.2 Simulation Results for the Three d.o.f. Planar Robot

The parametric studies are carried out using a three-degree-of-freedom planar robot with revolute joints. Its kinematic and dynamic models are given in Appendix A. The parameters of the space station remained the same as in last simulation while the parameters of the robot are listing in Table 4.2.2. Noted that the mass of each link remained constant, while the link lengths and the moment of inertia for each link have been changed accordingly. Specifically, the moment of inertia changes proportionaly to the square of the link length.


Figure 4.4: End Effector Trajectory

The end effector is assumed to move along a straight line, from point $(7.5,4.0)$ to $(8.0,4.5)$, through a distance $y=\sqrt{(7.5-8.0)^{2}+(4.0-4.5)^{2}}=0.71 \mathrm{~m}$. Assume the trajectory duration $T$ (equation (2.26) ) as 0.7 s , the acceleration limit $a$ (equation (2.33) ) as $10 \mathrm{~m} / \mathrm{s}^{2}$. Then the curtate cycloidal parameters $a, b_{1}$, and $c$ of the end-effector ( equations (2.23), (2.24), and (2.25) ) are obtained for this trajectory, as $a=0.11141$, $b_{1}=0.73406$ and $c=0.75224$. The linear jerk limit is computed using equation (2.37) as $\dot{a}=963.43 \mathrm{~m} / \mathrm{s}^{3}$.

The end effector trajectory is divided into 200 intervals. The initial optimized joint configuration is found to be $q_{1}=1.48 \mathrm{rad}, q_{2}=-0.42 \mathrm{rad}, q_{3}=-1.44 \mathrm{rad}$. Simulations are carried out with different link lengths for the parametric studies. The results are shown in Figures 4.13 and 4.14.


Figure 4.5: Base Reaction Cost Function

Figure 4.13 shows the optimal cost functions for different link lengths. The results of curve 7 and curve 8 show clearly to be better than that of other curves, indicating an enhanced optimal case. Curve 7 indicates the situation of having the length of the first link decreased, and curve 8 indicates the situation of the length of the first link decreased while the length of the third link is increased.

Figure 4.14 shows the curves 7 and 8 in Figure 4.13. Note that the link lengths for the case of smaller cost function (curve 1) are $l_{1}=2.5 \mathrm{~m}, l_{2}=5 \mathrm{~m}$ and $l_{3}=7.5 \mathrm{~m}$, and for the larger one (curve 2) they are $l_{1}=2.5 \mathrm{~m}, l_{2}=5 \mathrm{~m}$ and $l_{3}=5 \mathrm{~m}$.

It shows that the cost function is highly depended on the parameter values of link lengths. The functional dependance is complex, but can be numerically simulated and optimized. In particular, reducing the length of the first link for the manipulator seems


Figure 4.6: Joint 1 Trajectory
to have the biggest impact on minimizing the cost function values. This phenamenon is in agreement with physical interpretation because the first link produces the biggest direct torque contribution to the base station, which is directly proportional to the length of the first link. Therefore, It implies there are opportunities for design optimization.

### 4.3 Summary of Results

Using the framework of trajectory optimization, that was presented in the previous chapter, several simulation studies were carried out. The method utilizes the kinematically redundant degrees of freedom of a robot as means for instantaneous local optimization of a quadratic cost function of the disturbance forces and moments at the support structure of the robot. In this manner the best joint angle trajectories may be


Figure 4.7: Joint 2 Trajectory
planned. The computer simulation has been developed using MATLAB ${ }^{(\subset)}$ software on a SUN SPARC station platform. The software is structured into a modular high level form. The processing time of simulation is excessive because of the complexity of the optimization. The simulation algorithm is however robust, with a built in capability to avoid singular configuration of the robot and options to select a level of accuracy for the optimization. Simulations were carried out based on the model of optimizing the joint angle trajectory to satisfy a specific straight-line trajectory for the end effector. The difference between optimal and non-optimal cost function values ranges from 2 dB to 3 dB . The individual joint angle motions of the optimal trajectory are also found to be smoother than those of the non-optimal trajectory. Figure 4.13 illustrates how the cost function changes with the link length of the manipulator. This plot shows that a locally optimized cost function is still functionally dependent on the physical parameters such as link lengths of the robot. This strongly suggest that the optimization using kinematic


Figure 4.8: Joint 3 Trajectory
redundancy may be further utilized to do design optimization. However, a method of defining a global cost function is required, and a technique to optimize such a cost function must be developed and implemented. This aspect is addressed in the next chapter.


Figure 4.9: Joint 4 Trajectory


Figure 4.10: Joint 5 Trajectory


Figure 4.11: Joint 6 Trajectory


Figure 4.12: Joint 7 Trajectory

Table 4.2: Link Length Parameters for the Ten Cases in Figure 4.13

| Case No. | $l_{1}(\mathrm{~m})$ | $l_{2}(\mathrm{~m})$ | $l_{3}(\mathrm{~m})$ | $I_{\mathrm{c} 1}\left(\mathrm{~kg} \cdot \mathrm{~m}^{2}\right)$ | $I_{\mathrm{c} 2}\left(\mathrm{~kg} \cdot \mathrm{~m}^{2}\right)$ | $I_{\mathrm{c} 3}\left(\mathrm{~kg} \cdot \mathrm{~m}^{2}\right)$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1 | 5 | 5 | 5 | 200 | 200 | 200 |
| 2 | 2.5 | 2.5 | 2.5 | 50 | 50 | 50 |
| 3 | 4 | 4 | 4 | 128 | 128 | 128 |
| 4 | 6 | 6 | 6 | 288 | 288 | 288 |
| 5 | 7.5 | 5 | 5 | 450 | 200 | 200 |
| 6 | 7.5 | 5 | 2.5 | 450 | 200 | 50 |
| 7 | 2.5 | 5 | 5 | 50 | 200 | 200 |
| 8 | 2.5 | 5 | 7.5 | 50 | 200 | 450 |
| 9 | 5 | 5 | 7.5 | 200 | 200 | 450 |
| 10 | 5 | 5 | 2.5 | 200 | 200 | 50 |



Figure 4.13: Optimal Cost Function for Different Link Lengths


Figure 4.14: Optimal Cost Function with the First Link Length Decreased

## Chapter 5

## Parameter Design Optimization

### 5.1 Background

Design engineering for space robots covers many aspect of the space system; desirable parameters include high capacity/mass ratio, low parts count, modularity in configuration, large work envelope for the activity environment, dexerterity in the end effector and also low dynamic interaction with the supporting structure which is typically the space station.

Reduction of dynamic interactions in a space robotic system by path planning forms the basis of present thesis. This will minimize the required control effort, and thus reduce control fuel in the space operations. This idea may be extended to the design of a space robot system. Specifically, minimization of dynamic interactions between a robot and the space station is used as the criterion for parameter design optimization that is described in this chapter. The dynamic interaction is quantified by a mapping of reaction forces and torques into a scalar functional. Such a mapping is expressed as a cost function, which has to be minimized in the design procedure. Both local and global disturbance effects have to be studied using an appropriate cost function, in the design optimization.

As discussed in chapters 2 and 3 , in addition to its use in dexterity improvement and sigularity avoidance, kinematic redundancy can be successfully applied to minimize the
base reactions through appropriate planning of the joint trajectory, for a specified end effector trajectory. This is possible because the extra and redundant degrees of joint angle freedom can be utilized to locally minimize a cost function in terms of reactional forces and torques at the support structure, throughout the trajectory. In particular, it is attempted in this manner, to maintain the dynamic forces and moments at the location of interaction with the support structure ( space station) to be the same as reference values set in the cost functional formulation.

A unified approach for robot trajectory optimization, design and control has been proposed by de Silva [24]. Here the design of the robot system is linked to the problem of trajectory planning. This approach integrates the latest technology in each field in the spirit of concurrent engineering. The main steps of the procedure of the unified design optimization are:

1. Define the robotic tasks.
2. Group the tasks into appropriate, general subtasks, using a group technology appropriate.
3. Design the end effector trajectory for each subtask.
4. Select robotic parameters for individual groups.
5. Select a common set of parameters for all groups.
6. Optimize trajectory of each subtask using kinematic redundancy built into the robot.
7. Shape the optimal cost function by parameter redesign if needed.
8. Design the controller to achieve the optimized joint trajectories.

An effort to apply this general approach for a simple 3-R robot is given in the following section.

### 5.2 Global Parameter Optimization

Optimization of the joint angle trajectories of a redundant manipulator to minimize the reaction forces and torques generated at the base of the manipulator, according to a cost function defined locally along the trajectory, is the fundamental problem in the present work. This locally optimized cost function may be reshaped globally through parameter changes, and could constitute a design approach for the robot system.

In a unified parameter design optimization approach, it follows that, a global optimization can be quite useful. Unlike joint variables, geometric parameters such as link lengths and mass are considered global parameters. These parameters affect the dynamic forces and interactions because link lengths and inertia are directly present in the dynamic equations. For example, the link length is a global parameter in the sense that variations in it affects the workspace of the robot manipulator irrespective of joint angle configurations. Therefore, the end effector path imposes a constraint on the link lengths and vice versa. Note that link parameters do not have the same level of freedom as the local variables such as joint angles.

To optimize the choice of the design parameters $\boldsymbol{l}_{\boldsymbol{i}}$ for a particular task, a global cost function $F_{g}$ is defined as:

$$
\begin{equation*}
F_{g}=\alpha \int_{0}^{t_{f}} F_{c} d t+\beta F_{m a x} \tag{5.1}
\end{equation*}
$$

$$
\begin{equation*}
F_{\max }=M a x\left(F_{\mathrm{c}}\right) \tag{5.2}
\end{equation*}
$$

where $\alpha, \beta$ are positive constants, $t_{f}$ is the final time of the trajectory, $F_{c}$ is the instantaneous cost function value at the $i$ th interval, as defined in the previous chapter, and $F_{\max }$ is the peak value of the entire cost function along the end effector trajectory. In other words, the global cost function is the sum of the time integral of the local cost function and a weighted measure of the peak value of that cost function. The purpose of this cost function is to limit the peak values of the previous cost function, and also to restrain the global time average of dynamic reaction of the manipulator. In other words, in the global optimization process the local cost function would be reshaped through the proper selection of the design parameters.

The parameters of optimization here are $l_{i}$, which may represent, for example, link lengths and link inertia values. In what follows, we assume that the design parameters $l_{i}$ are link length alone. Then, the link inertia values could be adjusted depending on the link lengths according to same design strategy.

### 5.3 Computer Simulation

Based on the global cost function presented in the previous section, numerical simulations are carried out in the MATLAB ${ }^{\text {© }}$ environment. The flowchart of the simulations is shown in Figure 5.1. The method of optimization used is the nonuniform convergence of a feasible set using a gradient line search algorithm.

The procedure of global optimization of the base reactions using link lengths $l_{i}$ as the
optimization parameters, for a given end effector trajectory, is as follows:

1. Assume an initial best guess for $\boldsymbol{l}_{\boldsymbol{i}}$.
2. Use joint angle redundancy to minimize the instantaneous forces and moments at the supporting structure using a quadratic cost function, with the design parameters $l_{i}$ being kept unchanged.
3. Integrate the instantaneous cost function, and identify the overall peak value of the cost function. Calculate the global cost function for the end effector trajectory using this information.
4. The global cost function $F_{g}$ is used to globally optimize the system using link parameters $l_{i}$ subject to physical constraints. During each search step, goto Step 2 to obtain a best fit joint trajectory for a specific end effector trajectory.

The flowchart of computer simulation of this design procedure is shown in Figure 5.1

### 5.4 Results and Discussions

### 5.4.1 Optimization Results

A 3-R planar robot manipulator in $\Re^{2}$, as shown in Figure 5.2, serves as the model for the computer simulation, for parameter optimization. Assume that the reference frame is located at the centre of mass of the space station, with the $x-y$ plane coinciding with the motion plane. The link length is denoted by $l_{i}$. The joint angle $q_{1}$ is the angle between link 1 and $x$-axis. The joint angle $q_{2}$ is the angle between link 2 and link 1 , and $q_{3}$ is the angle between the links 3 and 2.


Figure 5.1: Algorithm for Design Optimizatiom of Robot Parameters


Figure 5.2: A Three-Degree-of-Freedom Planar Robot Mounted on a Space Station

The space station is assumed to be orbiting the earth at a geosynchronous period of 24 hours. The following parameters values are used for the space station:

Moment of inertia about the axis of rotation $I_{s}=19.7 \times 10^{6} \mathrm{~kg} \cdot \mathrm{~m}^{2}$.
Mass $M_{s}=18.0 \times 10^{3} \mathrm{~kg}$.
The link mass values used are :
Link mass $m_{i}=20.0 \mathrm{~kg}$.
Therefore, the moment of inertia will also be changing as the square of the link length, which is to be optimized.

An initial guess of link length is set at $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$. In each optimization search step, for every change in link length, program calls the local cost function optimization routines to obtain the best set of joint trajectories.

The optimization program is set to search for the best numerical solution that satisfy a given physical constraint [20].

$$
\begin{align*}
l_{1}+l_{2}+l_{3} & =15  \tag{5.3}\\
l_{3} & \leq l_{2} \\
0.5 l_{2} & \leq l_{3}
\end{align*}
$$

The constraint equations set the limits of the workspace of the manipulator as well as the ranges of the values for link 2 and link 3 based on physical considerations.

During local optimization period, the end effector is assumed to move along the same straight line, from point (7.5,4.0) to point (8.0, 4.5), traveling the distance
$y=\sqrt{(7.5-8.0)^{2}+(4.0-4.5)^{2}}=0.71 \mathrm{~m}$. Assume that the time of the trajectory motion ( $T$ in equation (2.26) ) to be 0.7 second, and the acceleration limit ( $a$ in equation (2.33) ) to be $10 \mathrm{~m} / \mathrm{s}^{2}$. Then the curtate cycloidal parameters $a, b_{1}$, and $c$ are obtained for this specific end effector trajectory as, $a=0.11141, b_{1}=0.73406$ and $c=0.75224$. The end effector trajectory is divided into 200 time intervals. The starting joint configurations are obtained through optimization within the first step, and was found to be $q_{1}=1.48 \mathrm{rad}, q_{2}=-0.42 \mathrm{rad}, q_{3}=-1.44 \mathrm{rad}$, This remained unchanged in each global optimization.

The globally optimal set of link variables was found to be $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$ with the global cost function value of $f_{g}=1.19 \times 10^{7}$. Figures 5.3 and 5.7 present the results. Figure 5.3 gives the end-effector trajectory used in the simulation. In Figure 5.4 the locally optimized cost function for link lengths $l=\left[\begin{array}{ll}5 & 5\end{array} 5\right]$ is shown as the reference (curve 2 ) with which the global optimization began. In the same figure, the instantaneous cost function for the final step of global optimization, which corresponds to $l=\left[\begin{array}{ll}1 & 7\end{array} 7\right]$, is also shown. As expected, the instantaneous cost function for $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$ is found to be substantially smaller than that of $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$ throughout the end effector trajectory. This shows the potential of using the dynamic reactions at the support structure as the performance index.

### 5.4.2 Summary of Results

Based on the simulation model for trajectory planning using kinematic redundancy, a global cost function is defined as the time integral of the local cost function with an additional term given by a linear function of the overall maximum value of that local cost function. The optimization algorithm uses a recursive technique to optimize the


Figure 5.3: End Effector Trajectory


Figure 5.4: Optimal Local Cost Function Values for $l=\left[\begin{array}{ll}177\end{array}\right]$ (curve 1 ) and $l=\left[\begin{array}{ll}5 & 5\end{array}\right]$ (curve 2 )


Figure 5.5: Joint 1 Trajectory for $l=\left[\begin{array}{lll}1 & 7 & 7\end{array}\right]$ (curve 1) and $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$ (curve 2)


Figure 5.6: Joint 2 Trajectory for $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$ ( curve 1 ) and $l=\left[\begin{array}{ll}5 & 5\end{array}\right]$ (curve 2)


Figure 5.7: Joint 3 Trajectory for $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$ (curve 1) and $l=\left[\begin{array}{lll}5 & 5 & 5\end{array}\right]$ (curve 2 )
link length parameters globally and the joint trajectory locally, in succession. In this manner, the link lengths of the robot are optimized as well as the joint trajectories. The link lengths are optimized with respect to a set of physical constraints while the joint trajectories are considered unconstrained. The computer simulation were carried out for a three-degree-of-freedom, revolute, planar manipulator. This method of optimization takes many hours of CPU time because of the high level of computational complexity of the global optimization. However, the results indicate that the algorithm and the numerical model are robust, with the cost function of the final optimal design being 1dB to 2 dB below that of the initially, locally optimized case. As shown in the computer simulation results, a set of optimized design parameter values is obtained as $l=\left[\begin{array}{ll}1 & 7\end{array}\right]$. It is expected that the optimal link length values will always be depend ent on the definition of the global cost function and the physical link constraints. This is a brute force method of doing design optimization. The methodology is essentially a local analysis approach because the global cost function is still directly related to the locally optimized joint trajectory. This functional relationship is embeded in the cost function definition, and the looped optimization procedure. A more global approach to exploit the fundamental relationship between link parameters and the cost function would be desirable. In the next chapter, a relatively global technique for analysing of inverse kinematics of a redundant manipulator from a topological point of view will be used for the parameter design optimization of the manipulator.

## Chapter 6

## The Trajectory Manifold Approach

### 6.1 Introduction

In the investigation described in the previous chapter, the trajectory optimization of robots has been based on the instantaneous kinematics approach. This approach has been applied to both local trajectory optimization and global design optimization with the cost function that expressed in terms of the dynamic forcs and torques at the support structure. Burdick [27] has, however, look at the global application of redundant kinematics of manipulators from a topological point of view. Here, the infinite number of trajectory solutions, which is possible in view of the kinematic redundacy, is grouped into a finite and bounded set of disjoint and continuous manifolds, known as self motion manifold. This concept of self motion manifold for a redundant robot manipulator is extended in the present chapter into a systematic and convienient method for redundancy resolution and parameter design optimization.

Smooth surfaces may be generalized as manifolds in an abstract, multidimensional hyperspace. In manifold theory, a smooth $n$ dimensional surface may be characterize by a set of flat Euclidian surfaces in the $n$ dimensional Euclidian space. In such a framework, useful mathematical concepts like differentiation become meaningful in the neighborhood of any locally defined coordinates. The concept of manifold has useful applications in robotics because the trajectories of joint angle coordinates for a general $n$
degree-of-freedom manipulator can be treated as an $n$ dimensional manifold flow, and the mathematical tools used in manifold theory can be applied to solve robotics problems. This can provide new insights into understanding robotic dynamics through a new geometric perspective. In particular, for kinematically redundant manipulators, manifold theory allows for the resolution of infinitely many possible trajectories into a finite group of submanifolds with similar characteristics. Furthermore, this approach allows for the implementation of a global cost function through a natural definition using the concept of a path integral along a family of submanifold.

The forward kinematic function of a manipulator is a nonlinear vector function that maps joint angle vectors into the end effector position and orientation vectors. For a redundant manipulator, there exist infinitely many joint angle configurations that can be mapped into a single end effector position and orientation. However, this set of infinite points can be grouped into a finite set of manifolds, each with a set of distinct global geometric characteristics. These submanifolds are known as "self-motion" manifolds. In other words, the motion in joint angle configuration in a self-motion manifold leaves the end effector motionless.

From the foregoing discussion it should be clear that a specific manipulator task maps out a specific end effector trajectory, which in turn maps out a family of flow submanifold, or "self-motion" submanifold. By defining the global cost function, which represents dynamic interactions between the robot and its support structure, along each submanifold, the design parameters of the robot system can be optimized by computing the cost function. Here the concept of Path Integrals is introduced. Since the full description of reaction forces and torques is complex for symbolic computation, only the first order approximations of $\boldsymbol{F}$ and $\boldsymbol{N}$ are considered. In other words, only higher order terms in
$\boldsymbol{F}$ and $\boldsymbol{N}$ containing link variables $l_{i}$ are considered in evaluating the path integral in the global cost function calculation.

### 6.2 Manifold Formulation of Manipulator Kinematics

In order to analyse the global characteristics of redundant manipulators, one needs to understand the kinematics of the robot in terms of manifold mapping instead of local kinematic analysis [26], [27].

### 6.2.1 Forword Kinematics

Let $\boldsymbol{Q}=\left\{\boldsymbol{q}_{\boldsymbol{i}}\right\}$ be the set of joint angle coordinates for the manipulator. Here, $\left\{q_{i}\right\}$ is of dimension $n$. Let $X=\left\{x_{i}\right\}$ be the set of end effector coordinates in vector space $\Re^{m}$, of dimension $m$. Note that, for kinematic redundancy of a manipulator we must have

$$
n \geq m+1
$$

The forward kinematics of a manipulator may be expressed as a nonlinear vector function which relates a set of $n$ joint coordinates, $\boldsymbol{Q}$, to a set of $m$ end effector coordinates:

$$
\begin{equation*}
\boldsymbol{X}=f(\boldsymbol{Q}) \tag{6.1}
\end{equation*}
$$

The forward kinematic function in equation (6.1) maps a given joint configuration, $\boldsymbol{Q}$, to a unique end effector location, $\boldsymbol{X}$. The set of joint configurations forms the "configuration space", $C$, and the end effector locations form the "workspace", $W$. Therefore, the forward kinematic function can be understood as a mapping of points from the configuration space to the workspace.

$$
\begin{equation*}
f(\boldsymbol{Q}): C \Longrightarrow W \tag{6.2}
\end{equation*}
$$

where, $C=$ configuration space; $W=$ workspace. The configuration space has an $n$ dimesional torus manifold. The workspace has a sheet liked structure.

### 6.2.2 Inverse Kinematics

The inverse-kinematics relation is given by:

$$
\begin{equation*}
Q=f^{-1}(X) \tag{6.3}
\end{equation*}
$$

For a redundant manipulator, and a given end-effector position and orientation, there exists an infinite number of solutions in $\left\{q_{i}\right\}$ to satisfy the $f^{-1}$ operation. In other words, $f^{-1}$ is a one-to-many mapping, that is, for a fixed $\boldsymbol{X}=\left\{\boldsymbol{x}_{i}\right\}$ then $\boldsymbol{Q}=\left\{\boldsymbol{q}_{i}\right\}$ is not a compact set. However, a non-redundant manipulator would have a bounded set of discrete and finite points in the configuration space for a given end effector position and orientation in the workspace.

For a redundant manipulator, let $r=n-m$ be the degree of redundancy. The inverse-kinematic solution is represented as a group of one or more disjoint r-dimensional manifolds $\mathcal{M}_{i}$, such that,

$$
\begin{equation*}
f^{-1}(\boldsymbol{X})=\bigcup_{i}^{4} \mathcal{M}_{i}(\psi) \tag{6.4}
\end{equation*}
$$

and

$$
\begin{equation*}
\mathcal{M}_{i}(\boldsymbol{\psi}) \bigcap \mathcal{M}_{j}(\boldsymbol{\psi})=\emptyset \quad \forall i \neq j \tag{6.5}
\end{equation*}
$$

Note that $\mathcal{M}_{i}$ is the $i$ th $r$-dimensional submanifold, with $i=1,2, \ldots s$, where, $s$ is the number of self motion in the inverse kinematics solution. Furthermore, $\boldsymbol{\psi}$ is a vector of $r$ independent parameters, $\psi=\left\{\psi_{1}, \ldots, \psi_{r}\right\}$, which are the generalized coordinates for the self motions.

Each of these submanifolds is thought of as a "self-motion" manifold, which is a set of continuous joint motions that has no effect on the end effector motion.

### 6.2.3 Jacobian Relations

Differential relationship between the joint displacement and the end effector position is given by:

$$
\begin{equation*}
\delta \boldsymbol{X}=\boldsymbol{J} \delta \boldsymbol{Q} \tag{6.6}
\end{equation*}
$$

where $J$ is the $m \times n$ Jacobian matrix of the redundant manipulator. The incremental self motion $\delta \boldsymbol{\psi}$, which corresponds to the incremental joint motion $\delta \boldsymbol{Q}$, is given by the differential relationship.

$$
\begin{equation*}
\delta \psi=J_{\psi} \delta Q \tag{6.7}
\end{equation*}
$$

where $J_{\psi}$ is the $r \times n$ Jocabian matrix, and $r=n-m$.

Let us consider the end effector coordinates and the self motion variables as an augmented task vector $\{\boldsymbol{X}, \boldsymbol{\psi}\}$. The differential relationship corresponding to this new set of generalized coordinates is:

$$
\left[\begin{array}{c}
\delta \boldsymbol{X}  \tag{6.8}\\
\delta \psi
\end{array}\right]=\boldsymbol{J}_{A} \delta \boldsymbol{Q}
$$

where $J_{A}$ is an $n \times n$ augmented Jacobian matrix, which is given by

$$
\boldsymbol{J}_{A}=\left[\begin{array}{c}
\boldsymbol{J}  \tag{6.9}\\
\boldsymbol{J}_{\psi}
\end{array}\right]=\left[\begin{array}{c}
\partial \boldsymbol{X} / \partial \boldsymbol{Q} \\
\partial \psi / \partial \boldsymbol{Q}
\end{array}\right]
$$

Note that the augmented Jacobian $\boldsymbol{J}_{A}$ will be singular whenever $\boldsymbol{J}$ loses rank and/or when $\boldsymbol{J}_{\psi}$ is not full rank. Assuming that this is not the case, the joint velocities are
obtained from equation (6.8), as

$$
\dot{\boldsymbol{Q}}=J_{A^{-1}}\left[\begin{array}{c}
\dot{X}  \tag{6.10}\\
\dot{\psi}
\end{array}\right]
$$

and the joint accelerations are obtained by differentiating equation (6.8), as

$$
\ddot{Q}=J_{A}{ }^{-1}\left[\begin{array}{c}
\dot{X}  \tag{6.11}\\
\ddot{\psi}
\end{array}\right]-J_{A}{ }^{-1} \dot{J}_{A} \dot{Q}
$$

### 6.2.4 Manifold Solution for a 3-R Robot

Consider the three-degree-of-freedom planar robot that has been discussed in the previous chapters. There are two possible sets of joint configurations for this 3 - R manipulator.

$$
\begin{equation*}
\text { 1. }\left\{\theta_{1 a}, \theta_{2 a}, \theta_{3 a}\right\}=\{(\alpha+\eta),(\gamma-\pi),(\psi-\alpha-\eta-\gamma+\pi)\} \tag{6.12}
\end{equation*}
$$

2. $\left\{\theta_{1 b}, \theta_{2 b}, \theta_{3 b}\right\}=\{(\alpha-\eta),(\pi-\gamma),(\psi-\alpha+\eta+\gamma-\pi)\}$
where,

$$
\begin{gather*}
\alpha=\tan ^{-1}\left(\frac{y_{e}-l_{3} \sin \psi}{x_{e}-l_{3} \cos \psi}\right)  \tag{6.14}\\
\gamma=\cos ^{-1}\left(\frac{l_{1}^{2}+l_{2}^{2}-R_{2}^{2}}{2 l_{1} l_{2}}\right)  \tag{6.15}\\
\eta=\cos ^{-1}\left(\frac{l_{1}^{2}-l_{2}^{2}+R_{2}^{2}}{2 l_{1} l_{2}}\right)  \tag{6.16}\\
R=\sqrt{x_{e}^{2}+y_{e}^{2}+l_{3}^{2}-2 l_{3}\left(x_{e} \cos \psi+y_{e} \sin \psi\right)} \tag{6.17}
\end{gather*}
$$

Here, $\left\{\theta_{1 a}, \theta_{2 a}, \theta_{3 a}\right\}$ is a set of joint motions in manifold $a$. $\left\{\theta_{1 b}, \theta_{2 b}, \theta_{3 b}\right\}$ is a set of joint motions in manifold $b$. $\left\{x_{e}, y_{e}\right\}$ is the position vector of end-effector. Also, $\psi$ is a redundant independent variable which is defined as the orientation of the third link relative to the base coordinate frame, and $l_{1}, l_{2}, l_{3}$ are lengths of links 1,2 , and 3 respectively. As $\boldsymbol{\psi}$ changes in its feasible range $[\pi,-\pi]$, equations ( (6.12) and (6.13)) will define two 1-dimensional manifolds in the configuration space. These manifolds are disjoint for all values of $\boldsymbol{\psi}$ unless the Jacobian submatrix formed by links 1 and 2 is not full rank.

### 6.3 Design Parameter Optimization

As in the previous chapter, the objective of the parameter design problem for a space robot is considered as the optimal selection of the most suitable set of link length $\left\{l_{i}\right\}$ for the manipulator such that the dynamic interactions with the support structure are minimized in a global manner. In particular, we wish to select a set of suitable design parameters such as link lengths to find an appropriate trajectory $\mathcal{L}_{c}$, such that,

$$
\begin{equation*}
\int_{\mathcal{L}_{c}} F_{c} d t=\min \left[\int_{\mathcal{L}} F_{c} d t\right] \tag{6.18}
\end{equation*}
$$

subject to the following constraints:

$$
\begin{align*}
l_{1}+l_{2}+l_{3} & =15  \tag{6.19}\\
l_{3} & \leq l_{2} \\
0.5 l_{2} & \leq l_{3}
\end{align*}
$$

where $F_{c}$ is the global cost function. $F_{c}$ is path dependent, and hence depends on the fundamental joint motion variable such as $q, \dot{q}$, and $\ddot{q} . F_{c}$ is also dependent on the design
parameters $\left\{l_{i}\right\}$.

The cost function used for parameter optimization must be global and representative description of the dynamic objective (e.g. minimization of the dynamic interaction ). Also, it must explicitly depend on the design parameters. However, for the simplification of numerical simulations, the partial functional dependance is considered to include the most significant contributing terms. Suppose that the functional form of the global cost function is

$$
\begin{equation*}
F c=\int_{t_{o}}^{t_{f}}\left(\ddot{q}_{i} l_{i}^{2}+\dot{q}_{i} \dot{q}_{k} l_{i} l_{k}\right) d t \tag{6.20}
\end{equation*}
$$

The first term of the integrand is representive of the direct inertia forces while the second term is representative of the cross coupling effects between joints.

The procedures used in the design parameter optimization using the monifold method are as following:

1. Initialize the self motion parameter $\boldsymbol{\psi}$ and the design parameters (link length $\boldsymbol{l}_{\boldsymbol{i}}$ ).
2. Generate the end effector trajectory. Divide the trajectory into a suitable number of time intevals.
3. Compute the multiple set of self-motion manifolds.
4. Compute the augmented Jacobian matrix $J_{A}$, and also $J_{A}{ }^{-1}, \boldsymbol{J}_{A}$, along with the end-effector velocity $\boldsymbol{v}$ and acceleration $\boldsymbol{a}$. Compute the joint velocities $\dot{\boldsymbol{Q}}$ and accelerations $\ddot{\boldsymbol{Q}}$.
5. Repeat Steps 2, 3 and 4 for the complete trajectory.
6. Compute the global cost function for each self motion monifold. Select the manifold with the smallest cost function value, for use in the optimization.
7. Increment the link lengths so as to minimize the cost function, subject to the physical constraints of these parameters.
8. If the minimum cost is not reached, go to Step 2. Else stop.

### 6.4 Computer Simulation

As before, let us consider a $3-\mathrm{R}$ planar robot manipulator in $\Re^{2}$ as the model for the computer simulation. This robot and supporting space station are schematically shown in Figure 6.1. Assume that the reference frame is located at the centre of mass of the base, with the $x-y$ plane coinciding with the motion plane. The link lengths are denoted by $l_{i}$. The joint angle $q_{1}$ is the angle between link 1 and the $x$-axis. The joint angle $q_{2}$ is the angle between links 2 and link 1 , and $q_{3}$ is the angle between the links 3 and 2 .

Let the self motion parameter $\boldsymbol{\psi}$ be defined as the orientation of the third link relative to the $x-y-z$ inertial reference frame. For a given end effector position ( $x_{e}, y_{e}$ ), there are two possible sets of joint configurations.

Suppose that the global cost function is defined by:

$$
\begin{equation*}
F_{c}=\int_{t_{0}}^{t_{f}}\left(\ddot{q}_{1} l_{1}^{2}+\ddot{q}_{2} l_{2}^{2}+\ddot{q}_{3} l_{3}^{2}+\dot{q}_{1} \dot{q}_{2} l_{1} l_{2}+\dot{q}_{1} \dot{q}_{3} l_{1} l_{3}+\dot{q}_{3} \dot{q}_{2} l_{3} l_{2}\right) d t \tag{6.21}
\end{equation*}
$$

In order to compute the global cost function, let us assume that the initial position of the end effector is set at $\left\{x_{o}, y_{0}\right\}$. Also the end effector velocity $v$ and acceleration $a$ are obtained from cycloid motion assumptions as described in Chapter 2.


Figure 6.1: A Three-Degree-of-Freedom Planar Robot and a Supporting Space Station

The self motion parameter $\psi$, which is defined as the angle between the third link of the robot and the inertial reference frame, is given by:

$$
\begin{equation*}
\psi=q_{1}+q_{2}+q_{3} \tag{6.22}
\end{equation*}
$$

The Jacobian matrix $\boldsymbol{J}$ of the $3-\mathrm{R}$ planar manipulator is given by:

$$
\boldsymbol{J}=\left[\begin{array}{ccc}
l_{1} S_{1}-l_{2} S_{12}-l_{3} S_{123} & -l_{2} S_{12}-l_{3} S_{123} & -l_{3} S_{123}  \tag{6.23}\\
l_{1} C_{1}-l_{2} C_{12}-l_{3} C_{123} & -l_{2} C_{12}-l_{3} C_{123} & -l_{3} C_{123}
\end{array}\right]
$$

and the Jacobian matrix $J_{\psi}$ corresponding to incremental self motion $\delta \psi$ and the incremental joint angle $\delta \boldsymbol{Q}$ is given by:

$$
\boldsymbol{J}_{\psi}=\left[\frac{\partial \psi}{\partial \boldsymbol{Q}}\right]=\left[\begin{array}{lll}
1 & 1 & 1 \tag{6.24}
\end{array}\right]
$$

the augmented Jacobian matrix $J_{A}$, is given by:

$$
J_{A}=\left[\begin{array}{c}
J  \tag{6.25}\\
J_{\psi}
\end{array}\right]=\left[\begin{array}{ccc}
l_{1} S_{1}-l_{2} S_{12}-l_{3} S_{123} & -l_{2} S_{12}-l_{3} S_{123} & -l_{3} S_{123} \\
l_{1} C_{1}-l_{2} C_{12}-l_{3} C_{123} & -l_{2} C_{12}-l_{3} C_{123} & -l_{3} C_{123} \\
1 & 1 & 1
\end{array}\right]
$$

Now, the joint velocities can be obtained from equation (6.10), and the joint accelerations from equation (6.11).

The space station model is assumed to be the same as in the previous simulations. The robot parameters are as follows:

Link length $l_{i}=5.0 \mathrm{~m}$.
Link mass $m_{i}=20.0 \mathrm{~kg}$.
Link moment of inertia about the joint axis $I_{i}=200.0 \mathrm{~kg} . \mathrm{m}^{2}$.
It is assumed that the end effector moves along the straight line, from point $(7.5,4.0)$ to $(8.0,4.5)$, as before through a distance $y=\sqrt{(7.5-8.0)^{2}+(4.0-4.5)^{2}}=0.71 \mathrm{~m}$.

Assume that the time duration of the trajectory ( $T$ in equation (2.26) ) is 0.7 s , the trajectory acceleration limit ( $a$ in equation (2.33) ) is $10 \mathrm{~m} / \mathrm{s}^{2}$. Then the curtate cycloidal parameters $a, b_{1}$, and $c$ are obtained for this specific end effector trajectory, as $a=$ $0.11141, b_{1}=0.73406$ and $c=0.75224$. The linear jerk limit (as given by equation (2.37) ) is obtained as $\dot{a}=963.43 \mathrm{~m} / \mathrm{s}^{3}$. The end effector trajectory is divided into 200 time intervals. The global cost function is calculated for the two manifolds, and the smaller is chosen for the cost function optimizatiom. The optimal set of link lengths is found to be $l=\left[\begin{array}{ll}5.4 & 6.4 \\ 3.2\end{array}\right]$ with the minimum value of the global cost function as $f_{g}=4.78 \times 10^{3}$.

Table 6.1 presents a set of global cost function values and the corresponding link lengths. The optimal result is given by the shaded row.

### 6.5 Summary of Results

In this chapter, the idea of flow manifolds, the separation of inverse-kinematic solutions of a robot into self motion manifolds, and the application of the concept of self motion for design optimization have been developed. Base on the definition of the global cost function as a path integral of design parameters on individual self motion manifolds, the cost function was computed along all self motion manifolds using the augmented Jacobian matrix, and the one with minimum value was set to optimize for the best design parameters. The global cost function is defined through a heuristic interpretations of the dependance of the base dynamic interaction on the link parameters, such that only a few quadratic terms are used to approximate the functional relationships. This method was used to simulate the parameter design optimization for a 3-R planar robot, and simulation results shown on Table 6.1 indicated that the method is capable for design optimization. This is another alternative to the brute force design optimization process

Table 6.1 Cost Function Values versus Link Lengths

|  | Link Lengths |  |  | Cost Function |
| :---: | :---: | :---: | :---: | :---: |
|  | 11 | 12 | 13 |  |
| Optimal | 2.0 | 7.0 | 6.0 | $7.16 \times 10$ e3 |
|  | 3.6 | 4.6 | 6.8 | 7.80×10e3 |
|  | 3.8 | 6.4 | 4.8 | $5.62 \times 10$ e3 |
|  | 4.0 | 7.6 | 3.4 | $4.90 \times 1003$ |
|  | 4.4 | 4.0 | 5.6 | $5.94 \times 10 ө 3$ |
|  | 4.8 | 5.2 | 5.0 | $5.77 \times 10$ e3 |
|  | 5.4 | 6.4 | 32 | 4.78*1083 |
|  | 5.0 | 6.6 | 3.4 | $4.82 \times 1003$ |
|  | 5.2 | 6.0 | 3.8 | $4.99 \times 10$ e3 |
|  | 6.0 | 5.4 | 3.6 | $5.04 \times 1003$ |
|  | 7.8 | 3.2 | 4.0 | $5.41 \times 10$ e3 |
|  | 7.6 | 4.0 | 3.4 | $5.61 \times 10 \times 3$ |
|  | 8.0 | 4.0 | 3.0 | $5.66 \times 1003$ |

as described in the previous chapter. This method provides a more transparent design optimization because of the simplified cost function. Also, as a result of the computations being able to be done symbolically rather than numerically, the computed global cost function becomes a simple polynomial (quadratic) function of the design parameters. This method has its limitations as well. One is the difficulty to resolve the redundant, inverse kinematics into self motion manifolds for a manipulator with large number of degrees of freedom. Another is the error introduced as a result of using a simplified cost function.

In comparision to the results obtained by the method of the previous simulations, the optimal set of link variables obtained by the manifold method, in this Chapter, is different for the following reasons:

1. The global cost functions used are different. In the manifold method, the symbolic cost function consists only of a few terms expressed as a quadratic form of $l_{i}$, whereas the cost function in the original simulation in Chapter 5 uses the complete expression for the base reactions.
2. The approach in Chapter 5 utilizes the joint angle redundancy to locally minimize the instantaneous forces and moments along each interval of the end effector trajectory, whereas in the manifold method, the joint angles are computed using the manifold equations.

Topology is the study of global geometric characteristics of mathematical objects, such as the curvature of surfaces, which is invariant under continuous transformations. For deisgn optimization, this approach is suitable and convienient because of many simiplications that can be used. These advantages include the ability to an group the infinite number of inverse-kinematic solutions into a finite number of classes which depend on
the link parameters. This provides a more heuristic and direct definition of a global cost function with respect to design parameters, even though not exact. Also, a simplified understanding of the functional relationship between the design parameters and the cost function is possible through such a cost function. Thereby, the optimization complexity is reducded which is an advantege in numerical simulations.

## Chapter 7

## Conclusion and Recommendations for Future Work

### 7.1 Summary of the Thesis

Advances in the space technology and the development of robotics will lead to more and more sophisticated robotic designs and advanced control techniques. Unwanted dynamic disturbances can adversely affect the performance of a space robot system. The objective of this research has been to explore and develop a methodolgy for the optimization of robot design parameters, by exploiting kinematic redundancy in the robot. The scope of this research includes the analytical formulation and computer simulation of the optimization of trajectory planning using kinematic redundancy, and the development of approaches and software tools for optimal parameter design of a robot. Initially, a robot manipulator with redundant degree of freedom is considered for minimization of dynamic interaction between the robot and its support structure. Using this as the preformance measure as expressed by a quadratic cost function, local optimization along the end effector trajectory, for a specific task, is formulated using the redundant joint motion as the means of optimization. On this basis, numerial models have been built using the MATLAB © software language. Computer simulations for both 3-R planar robot and the NASA seven d.o.f. traction drive manipulator have shown that the base reactions can be effectively minimized through the optimization process. These software tools are subsequently extended for the analysis of the parameter design problem for a robot. Two approaches have been developed for this purpose. The first is a
brute force method that numerically describes the global cost function for the parameter design problem as the sum of a time integral of the previously explored instantaneous cost function and the scaled maximum of the instantaneous cost function. The second method uses the mathematical concept of topological manifolds to separate the trajectory flows of redundant manipulators into different but finite number of "self motion" submanifolds that have similar geometric characteristics. In each submanifold, the end effector is motionless, and hence an optimization may be carried out with a proper definition of global cost function for base reaction minimization. This function is typically a simple quadratic of the design parameters and can be easily computed and eventually optimized. Even though there are computational advantages, because the cost function is simplified, the design results become less accurate. Both methods have been verified through the computer simulation of a 3-R planar robot manipulator, and have shown to perform effectively.

### 7.2 Major Contributions and Shortcomings of the Research

The key accomplishments of this research are:

1. Formulation of the optimization problem for minimizing dynamic interactions between a robot and its support structure, in terms of kinematic redundacy.
2. The MATLAB ${ }^{\mathbb{C}}$ modelling of the associated trajectory optimization problem.
3. The development of global techniques for design parameter optimization. This includes:
(a) Definition of a global cost function.
(b) Development of computer simulation algorithms for optimizing the global cost function.
(c) The application of the manifold flow concept to redundant robots for computing the multivalued global cost function as a path integral along the redundant d.o.f. trajectory.
(d) Development of computer simulation for carrying out the design optimization process.
(e) Carrying out several computer simulations to illustrate the techniques.

Even though an approach for design optimization has been developed, which was found to be effective, there are still some issues that need to be addressed, and techniques that need to be further developed. For example:

1. The overall optimization process is computationally inefficient.
2. The results of the design optimization depond on the specific definition of the global cost function. The numerical stability of the optimization methods, results to various parameters of the deisgn problem have not been investigated.
3. The manifold flow method, in this stage, serves only as illustrative purposes. In particular, only a quadratic function in the design parameters is used to compute the path integral which forms the global cost function.
4. In the design simulations, only the link length have been used as the optimization parameters. The explicit use of other parameters such as mass, moment of inertia, and strength, in the design has not been addressed.

### 7.3 Recommendations for Future Work

This thesis has made contributions towards the use of kinematic redundancy for minimizing dynamic interactions between a robot and its support structure, when carrying out specific classes of trajectory following tasks by an end effector. This research has also made progress towards the development of a robot parameter design methodology, on that basis. In order to enhance the theory and to realistically apply this work, there are several reconmmendations that can be made for future research:

1. Development of efficient numerical algorithms for constrained optimization of a nonlinear global cost function.
2. Extension of the concepts developed here to include other robot parameters and structures.
3. Development of control techniques that will guarantee the execution of the designed, optimal trajectories.
4. Enhancement of the design optimization to cover several robotic tasks, through , for example, a group technology approach.
5. Application of the techniques to a realistic design, and evaluation of the design.

### 7.4 Application of the Work

Since the software modules developed here are general, they may be applied to simulate any robot. Specific application considered here is in space robotics. For example, the methodology developed can be used to design a robot manipulator for space station operation. Consider a robotic task such as deployment of a scientific instrument like the Hubble Telescope using a redundant Canadian Arm on a NASA space shuttle mission.

For the purpose of saving fuel and minimizing the dynamic disturbance on the space station, the design parameters of the Canada Arm can be optimized according to the task specifications of the mission. First, kinematic and dynamic requirement of the end effector of the arm are used to calculate the kinematic parameters in the curtate cycloidal motion planning for the manipulator. Secondly, physical considerations on dynamic disturbance can be translate into an appropriate weigthing matrix $Q$ for the description of the cost function at the base station. The robot arm trajectory, either a straight line or more complex motion, is used to help generate the appropriate joint trajectory for the robot arm using inverse kinematics in the kinematic.m software module. The link length optimization for the Canada arm can be done using the brute force method described in Chapter 5. In this case, the Jacobian matrix for the Canada arm need to be derived, the physical parameters of mass, moment of inertia, and the parameters in the description of the global cost function all need to be realistically estimated according to the engineering specification of the space mission, but the methodology of design optimization is identical to the procedures outlined in the thesis. After the optimal selection of link parameters, the redundant trajectory planning software module can also be applied to generate the best joint angle path for the Canada Arm to deploy the instrument from the start point to the end point according to the specified end effector trajectory, either a straight line or a curve. However, there is of course the need to design controllers to regulate the joints of the robot accordingly. But nevertheless, the Canada Arm can now deploy a payload swiftly and smoothly.

## Bibliography

[1] De Silva, C.W., "1987 NASA-ASEE Summer Fellowship at the Lewis Research Center", Final report of the NASA-ASEE Case-lewis summer faculty fellowship program, pp. D-47-54, September 1987.
[2] De Silva, C.W., Ching, C.L., and Lawrence, C., "Base Reaction Optimization of Robotic Manipulators for Space Applications", proc. of the 19th International Symposium on Industrial Robots, Springer-Verlag, New York, Nov. 1988, pp. 829-852.
[3] Vafa, Z. and Dubowski, S., "Minimization of Spacecraft Disturbances in Space Robotic Systems", Proc. 11th AAS Guidance and Control Conf. Keystone, CO, 1988.
[4] Papadoponlos, E. and Dubowsky, S., " On the Nature of Control Algorithms for Space Manipulators" proc. IEEE International Conf. on Robotics and Automation, pp. 1102-1108, 1989.
[5] Smith,J.H., Estus, J., Heneghan, C., and Nainan, C., "The Space Station Sreedom Evolution-Phase: Crew-EVA Demand for Robotic substitution by Task Primitive", proc. IEEE International Conf. on Robotics and Automation, Scottsdale, AZ, pp. 1472-1477, 1989.
[6] Yamada, K. and Tsuchiya, K., "Trajectory Planning for a Space Manipulator", Proc. AAS/AIAA Astrodynamics Specialist Conference, Stowe, VM, Paper No. AAS89440, August 1989.
[7] De Silva, C.W., "Trajectory Design for Robotic Manipulator in Space Applications". J. Guidance, Control and Dynamics, Vol.14, No.3, pp. 670-674, 1991.
[8] De Silva, C.W., "Use of Kinematic Redundancy for Minimizing Dynamic Interaction in Space-Station Robots", Modeling and simulation, Instrument Society of America Publication, Vol. 24,Part 4,pp. 2231-2238, April 1992.
[9] Wee, L.B. and Walker, M.W., "Space Based Robot Manipulators: Dynamics of Contact and Trajectory Planning for Impact Minimization", Proc. American Control Conference, Chicago,IL,pp. 771-775, June 1992.
[10] Shigley, J.E., Theory of Machines and Mechanisms, McGraw Hill Book Co., New York, 1980.
[11] Herndon, J.N., Hamel, W.R., and Kuban, D.P., "Traction-Drive Telerobot for Space Manipulation", proc. IEEE International Conf. on Robotics and Automation, Raleigh, NC, 1987.
[12] Kuban, D. P., and Hamel, W.R., "Application of A Traction - Drive Seven - Degrees of Freedom Telerobot to Space Manipulation", proc. American Astronautical Society 10th Annual Guidance and Control Conference, Keystone, Colorado, 1987.
[13] De Silva, C.W., and Hankins, W.W., " Dynamic Evaluation of the NASA-ORNL traction-Drive Joint", Journal of Guidance, Control, and Dynamics, vol. 14, no. 3, pp. 688-692, May-June 1991.
[14] Luh, J.Y.S., Walker, M.W., and Paul, R.P., "On-Line Computational Scheme for Mechanical Manipulators", J. Dynamic systems, Measurement, and Control, Trans. ASME, vol. 120, pp. 69-76, 1980.
[15] Asada, H. and Slotine,J.J.E., Robot Analysis and Control, Wiley, New York, 1986.
[16] Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., Robotics: Control, Sensing, Vision, and Intelligence, McGraw-Hill, New York, 1987.
[17] Ranky, P.G., and Ho, C.Y., Robot Modelling Control and Applications with Software, Springer-Verlag, New York, 1985.
[18] MATLAB User's Guide, Published by The MathWorks, Inc. 1990.
[19] Rao, S.S., Optimization Theory and Applizations, John Wiley and Sons, New York, 1985.
[20] Mayorga, R.V., Ressa, B., and Wong, A.K.C., "A Kinematic Criterion for the Design Optimization of Robot Manipulators", proc. IEEE International Conf. on Robotics and Automation, Sacramento, CA, pp. 578-583, 1991.
[21] Wolfe, M.A., Numerical Methods for Uncontrained Optimization, Van Nostrand Reinhold Company, New York, 1978.
[22] Gill, P.E., and Murray, W., Numerical Methods for Constrained Optimization, Academic Press, London,1974.
[23] Stadler, W., Multicriteria Optimization in Engineering and in the Sciences, Plenum Press, New York, 1988.
[24] De Silva, C.W., and Wang,Y., "Use of Kinematic Redundancy for Design Optimization in Space-Robot Systems", proc. 1993 American Control Conference, San Francisco, CA, vol. 2, pp. 1806-1810, June 1993.
[25] Ighal, H. Dubey, R.V., and Euler, J.A., " Efficient Gradient Projection Optimization for Manipulator with Multiple Degree of Redundancy", , proc. Internationa Conf. on Robotics and Automation, pp1006-1011, 1990.
[26] Burdick,J.W. and Sereji, H. "Charecterization and Control of Self Motions in Redundant Manipulators", NASA Conference on Space Telerobotics, Pasadena, Calif. Vol. 2, pp.3-14, 1989.
[27] Burdick,J.W., " On the Inverse Kinematics of Redundant Manipulator: Characterization of the Self-Motion Manifolds", proc. International Conf. on Robotics and Automation, Scottsdale, AZ, Vol.1, pp264-270, 1989.
[28] Gottlieb, D.H., "Robots and Topology", proc. International Conf. on Robotics and Automation, San Francisco, CA, pp. 1689-1691, 1987.
[29] Guillemin, V., and Pollack, A., Differential Topology, Prentice-Hall, Inc., Englewood Cliff, New Jersey, 1974.
[30] Massey, W.S., Algebraic Topology: An Introduction, Springer-verlag, New York, 1967.

## Appendix A

## Appendix

## A. 1 Kinematics and Dynamics of a Three d.o.f. Planar Robot

For a 3 d.o.f. planar robot with rotary joints, the Jacobian matrix is given by:

$$
J=\left[\begin{array}{ccc}
-l_{1} S_{1}-l_{2} S_{12}-l_{3} S_{123} & -l_{2} S_{12} & -l_{3} S_{123}  \tag{A.1}\\
-l_{1} C_{1}-l_{2} C_{12}-l_{3} C_{123} & -l_{2} C_{12} & -l_{3} C_{123}
\end{array}\right]
$$

The positions of the centers of mass of the links with respect to the base reference frame are:

$$
\begin{gather*}
\boldsymbol{r}_{c 1}=\left[\begin{array}{l}
\left(\boldsymbol{r}_{c 1}\right)_{x} \\
\left(\boldsymbol{r}_{c 1}\right)_{x}
\end{array}\right]=\left[\begin{array}{l}
-l_{c 1} C_{1} \\
-l_{c 1} S_{1}
\end{array}\right]  \tag{A.2}\\
\boldsymbol{r}_{c 2}=\left[\begin{array}{l}
\left(\boldsymbol{r}_{c 2}\right)_{x} \\
\left(\boldsymbol{r}_{c 2}\right)_{x}
\end{array}\right]=\left[\begin{array}{l}
l_{c 2} C_{1}+l_{c 2} C_{12} \\
l_{c 2} S_{1}+l_{c 2} S_{12}
\end{array}\right]  \tag{A.3}\\
\boldsymbol{r}_{c 3}=\left[\begin{array}{c}
\left(\boldsymbol{r}_{c 3}\right)_{x} \\
\left(\boldsymbol{r}_{c 3}\right)_{x}
\end{array}\right]=\left[\begin{array}{l}
l_{c 2} C_{1}+l_{c 2} C_{12}+l_{c 3} C_{123} \\
l_{c 2} S_{1}+l_{c 2} S_{12}+l_{c 3} S_{123}
\end{array}\right] \tag{A.4}
\end{gather*}
$$

where

$$
\begin{aligned}
& C_{i}=\cos \theta_{i}, S_{i}=\sin \theta_{i}, C_{i j}=\cos \left(\theta_{i}+\theta_{j}\right), S_{i j}=\sin \left(\theta_{i}+\theta_{j}\right) \\
& C_{i j}=\cos \left(\theta_{i}+\theta_{j}+\theta_{k}\right), S_{i j}=\sin \left(\theta_{i}+\theta_{j}+\theta_{k}\right)
\end{aligned}
$$

The velocities of the centers of mass of the links are obtained by differentiating equations (A.2) through (A.4):

$$
\begin{gather*}
v_{c 1}=\left[\begin{array}{c}
-l_{c 1} \dot{q}_{1} S_{1} \\
l_{c 1} \dot{q}_{1} C_{1}
\end{array}\right]  \tag{A.5}\\
v_{c a}=\left[\begin{array}{c}
-l_{c 1} \dot{q}_{1} S_{1}-l_{c 2}\left(\dot{q}_{1}+\dot{q}_{2}\right) S_{12} \\
l_{c 1} \dot{q}_{1} C_{1}+l_{c 2}\left(\dot{q}_{1}+\dot{q}_{2}\right) C_{12}
\end{array}\right]  \tag{A.6}\\
v_{c 3}=\left[\begin{array}{c}
-l_{c 1} \dot{q}_{1} S_{1}-l_{c 2}\left(\dot{q}_{1}+\dot{q}_{2}\right) S_{12}-l_{c 3}\left(\dot{q}_{1}+\dot{q}_{2}+\dot{q}_{3}\right) S_{123} \\
l_{c 1} \dot{q}_{1} C_{1}+l_{c 2}\left(\dot{q}_{1}+\dot{q}_{2}\right) C_{12}+l_{c 3}\left(\dot{q}_{1}+\dot{q}_{2}+\dot{q}_{3}\right) C_{123}
\end{array}\right] \tag{A.7}
\end{gather*}
$$

The accelerations of the centers of mass of the links are obtained by differentiating equations (A.5) through (A.7):

$$
\begin{gather*}
\dot{v}_{c 1}=\left[\begin{array}{c}
-l_{c 1} \dot{q}_{1}^{2} C_{1}-l_{c 1} \ddot{q}_{1} S_{1} \\
-l_{c 1} \dot{q}_{1}^{2} S_{1}+l_{c 1} \ddot{q}_{1} C_{1}
\end{array}\right]  \tag{A.8}\\
\dot{v}_{c 2}=\left[\begin{array}{c}
-l_{1} \dot{q}_{1}^{2} C_{1}-l_{1} \ddot{q}_{1} S_{1}-l_{c 2}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}\right)^{2} C_{12}-L_{c 2}\left(\ddot{q}_{1}+\ddot{q}_{2}\right) S_{12} \\
-l_{1} \dot{q}_{1}^{2} S_{1}+l_{1} \ddot{q}_{1} C_{1}-l_{c 2}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}\right)^{2} S_{12}+l_{c 2}\left(\ddot{q}_{1}+\ddot{q}_{2}\right) C_{12}
\end{array}\right]  \tag{A.9}\\
\dot{v}_{c 3}=\left[\begin{array}{r}
-l_{1} \dot{q}_{1}^{2} C_{1}-L_{1} \ddot{q}_{1} S_{1}-l_{2}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}\right)^{2} C_{12}-l_{2}\left(\ddot{q}_{1}+\dot{q}_{2}\right) S_{12} \\
-l_{c 3}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}+\dot{q}_{3}^{2}\right)^{2} C_{123}-l_{c 3}\left(\ddot{q}_{1}+\ddot{q}_{2}+\ddot{q}_{3}^{2}\right) S_{123} \\
-l_{1} \dot{q}_{1}^{2} S_{1}-l_{1} \ddot{q}_{1} C_{1}-l_{2}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}\right)^{2} S_{12}-l_{2}\left(\ddot{q}_{1}+\ddot{q}_{2}\right) C_{12} \\
-l_{c 3}\left(\dot{q}_{1}^{2}+\dot{q}_{2}^{2}+\dot{q}_{3}^{2}\right)^{2} C_{123}-l_{c 3}\left(\ddot{q}_{1}+\ddot{q}_{2}+\ddot{q}_{3}^{2}\right) S_{123}
\end{array}\right]
\end{gather*}
$$

The angular accelerations of the three links are:

$$
\begin{equation*}
\dot{\omega}_{1}=\ddot{q}_{1} \tag{A.10}
\end{equation*}
$$

$$
\begin{equation*}
\dot{\omega}_{2}=\ddot{q}_{1}+\ddot{q}_{2} \tag{A.11}
\end{equation*}
$$

$$
\begin{equation*}
\dot{\omega}_{3}=\ddot{q}_{1}+\ddot{q}_{2}+\ddot{q}_{3} \tag{A.12}
\end{equation*}
$$

The base reaction forces and moment are given by:

$$
\begin{equation*}
\boldsymbol{f}_{x}=m_{1}\left(\dot{\boldsymbol{v}}_{c 1}\right)_{x}+m_{2}\left(\dot{v}_{c 2}\right)_{x}+m_{3}\left(\dot{v}_{c 3}\right)_{x} \tag{A.13}
\end{equation*}
$$

$$
\begin{equation*}
f_{y}=m_{1}\left(\dot{v}_{c 1}\right)_{y}+m_{2}\left(\dot{v}_{c 2}\right)_{y}+m_{3}\left(\dot{v}_{c 3}\right)_{y} \tag{A.14}
\end{equation*}
$$

$$
\begin{align*}
\boldsymbol{N}_{z}= & m_{1}\left(\boldsymbol{r}_{c 1}\right)_{x}\left(\dot{\boldsymbol{v}}_{c 1}\right)_{y}-m_{1}\left(\boldsymbol{r}_{c 1}\right)_{y}\left(\dot{\boldsymbol{v}}_{\mathrm{c} 1}\right)_{x}  \tag{A.15}\\
& +m_{2}\left(\boldsymbol{r}_{c 2}\right)_{x}\left(\dot{v}_{c 2}\right)_{y}-m_{2}\left(\boldsymbol{r}_{c 2}\right)_{y}\left(\dot{v}_{c 2}\right)_{x} \\
& +m_{3}\left(\boldsymbol{r}_{c 3}\right)_{x}\left(\dot{v}_{c 3}\right)_{y}-m_{3}\left(\boldsymbol{r}_{c 3}\right)_{y}\left(\dot{\boldsymbol{v}}_{c 3}\right)_{x} \\
& +I_{c 1} \dot{\boldsymbol{\omega}}_{1}+I_{c 2} \dot{\omega}_{2}+I_{c 3} \dot{\boldsymbol{\omega}}_{3}
\end{align*}
$$

