DYNAMIC CONTROL OF KINEMATICALLY REDUNDANT MANIPULATORS

Zhengcheng Lin

A Thesis

in

The Department

of

Electrical & Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montreal, Quebec, Canada

March 1993

ISBN 0-315-84627-5

ABSTRACT

Dynamic Control of Kinematically Redundant Manipulators

Zhengcheng Lin, Ph.D.
Concordia University, 1993

This thesis is concerned with the problem of dynamic control of kinematically redundant manipulators. A robot manipulator is said to be kinematically redundant when it has more degrees of freedom than are necessary to accomplish a particular task. Kinematic redundancy in a robot manipulator is a desirable characteristic since such manipulators have increased dexterity and versatility due to their self-motion. However, kinematic redundancy in a manipulator's structure presents a challenging problem since the richness in the choice of joint motions for the same end-effector trajectory complicates the manipulator kinematics and control problems considerably.

The main objective of the work presented in this thesis is to design useful control strategies for kinematically redundant manipulators in order to enhance their performance to achieve the main task as well as the secondary tasks. In particular, in this thesis the following three problems are considered: First, following the impedance control approach, the problem of minimizing redundant manipulator collision impacts is addressed. The configuration control approach is used to reduce impulsive forces, while a simplified impedance control scheme is formulated to minimize rebound effects. Second, a new Cartesian control strategy for redundant flexible-joint manipulators, namely the *hybrid Cartesian-joint control* scheme, is proposed. The main idea in this scheme is to control not only the manipulator's end-effector, but also its links so as to achieve specified positions and velocities for the end-effector and the links. Finally, a new application of kinematically redundant manipulators is proposed, namely, that of using redundancy resolution to compensate for joint flexibility. This redundancy resolution scheme is incorporated in a control strategy for redundant flexible-joint manipulators. The problem of possible algorithmic singularities is addressed, and a scheme is proposed which makes the controller robust with respect to such singularities.

ACKNOWLEDGEMENTS

I would like to express my thanks to my supervisor, Professor R.V. Patel, for his invaluable guidance, encouragement and support throughout the course of this work. He gave me the opportunity and introduced me to various aspects of robotics and control systems. He contributed a great deal of his time, effort and ideas to the work presented in this dissertation. I consider myself fortunate to have been working in association with him.

I would like to thank Professor K. Khorasani for very helpful discussions and suggestions. I would like to acknowledge the examiners for their valuable comments. Special thanks to Dr. C. Balafoutis for discussions and his comments on the thesis.

I also wish to thank all the members of the Computer support group. Nicky, Gustavo, Dave and Guy have been extremely helpful and have provided excellent support for the computing facilities. Many thanks are extended to all my friends and fellow graduate students for their friendship and good time. Finally, I wish to thank my wife Qin for her encouragement, patience and care over many weekends and week nights I spent working away at school, and my lovely son Bryan for his patience during my preoccupation with this work.

Dedicated to

My wife *Qin*,

and my son *Bryan*

TABLE OF CONTENTS

LIST OF SYMBOLS

Unless mentioned otherwise, the following notations are used in this thesis

n	Number of degrees-of-freedom of a manipulator
m	Number of degrees-of-freedom of the Cartesian (task) space
r	Number of degrees of redundancy ($r = n - m$)
s	Number of flexible joints
$q_l\,(\dot{q}_l, \ddot{q}_l)$	$n \times 1$ link position (velocity, acceleration) vector(s) of a flexible-joint manipulator
$q_m\,(\dot{q}_m, \ddot{q}_m)$	$n \times 1$ motor position (velocity, acceleration) vector(s) of a flexible-joint manipulator
$q\,(\dot{q}, \ddot{q})$	$n \times 1$ joint position (velocity, acceleration) vector(s) of a rigid-joint manipulator, while in the case of flexible-joint manipulator, it denotes $2n \times 1$ extended joint position (velocity, acceleration) vector, where $q = \left[q_l^T\ q_m^T\right]^T$
$x\,(\dot{x}, \ddot{x})$	$m \times 1$ Cartesian (end-effector) position (velocity, acceleration) vector(s)
$\Lambda\,(q)$	$m \times 1$ nonlinear function denoting manipulator forward kinematics
$J_e\,(q)$	$m \times n$ manipulator Jacobian matrix
$J\,(q)$	$n \times n$ augmented Jacobian matrix of the manipulator
$J_e^{\dagger}\,(q)$	$n \times m$ pseudo-inverse matrix of the Jacobian matrix

ξ	$n \times 1$ redundancy resolution vector
ξ_p	$(n-m) \times 1$ transformed redundancy resolution vector
τ	$n \times 1$ joint torque vector
F	$m \times 1$ external force/moment vector
$D_l(q_l)$	$n \times n$ symmetric positive-definite inertia matrix of the manipulator
$C_l(q_l, \dot{q}_l)$	$n \times 1$ Coriolis and centrifugal torque vector
$G_l(q_l)$	$n \times 1$ gravitational torque vector
$F_l(\dot{q}_l)$	$n \times 1$ frictional torque vector
F_c	$m \times 1$ Cartesian force/moment vector
$D_x(q_l)$	$m \times m$ manipulator inertia matrix expressed in Cartesian space
$C_x(q_l, \dot{q}_l)$	$m \times 1$ Cartesian space Coriolis and centrifugal force vector
$G_x(q_l)$	$m \times 1$ Cartesian space gravitational force vector
D_{mr}	$(n-s) \times (n-s)$ motor inertia matrix corresponding to rigid joints
D_{mf}	$s \times s$ motor inertia matrix corresponding to flexible joints
D_m	$n \times n$ motor inertia matrix where $D_m = \begin{bmatrix} D_{mr} & 0 \\ 0 & D_{mf} \end{bmatrix}$
K_s	$n \times n$ constant diagonal stiffness matrix of joint flexibility in the case where all the joints are flexible
K_l	$s \times s$ constant diagonal stiffness matrix of joint flexibility in the case where s joints are flexible
M_1	$m \times m$ positive-definite desired mass matrix in impedance controller
B_1	$m \times m$ positive-definite desired damping matrix in impedance controller
K_1	$m \times m$ positive-definite desired stiffness matrix in impedance controller
W_1	$m \times m$ mobility tensor (or virtual mass matrix) in Cartesian space
ρ	mass ratio coefficient
$L(q, D_l)$	kinematic/dynamic objective function
F_{imp}	$m \times 1$ impulsive force vector
n_c	$m \times 1$ unit vector normal to the plane of collision

μ — scalar constant coefficient of restitution denoting the type of collision

K_{pl}, K_{vl} — $n \times n$ desired stiffness and damping matrices for joint-space control scheme of flexible-joint manipulator

β_{pl}, β_{vl} — $m \times m$ desired stiffness and damping matrices for Cartesian tracking controller of flexible-joint manipulator

K_{pm}, K_{vm} — $n \times n$ desired stiffness and damping matrices for motor tracking controller of flexible-joint manipulator (in the case where s joints are flexible, the dimensionality of these two matrices are $s \times s$)

κ_p, κ_v — $n \times n$ desired stiffness and damping matrices of joint controller in Cartesian/joint hybrid control of flexible-joint manipulator

$e_l, \dot{e}_l$ — $n \times 1$ joint space position and velocity error vectors

$e_x, \dot{e}_x$ — $m \times 1$ Cartesian space position and velocity error vectors

$e_m, \dot{e}_m$ — $n \times 1$ motor position and velocity error vectors

γ — $s \times 1$ link movement vector produced by the manipulator's self-motion

P — $n \times (n-m)$ decomposed matrix from the matrix $(I - J_e^{\dagger} J_e)$

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

With the pressing need for increasing productivity and quality of end-products, the use of computer-controlled robots has increased in manufacturing, and in applications in space, oceans, hazardous environments, etc. Future applications will involve the use of robots in every aspect of life. To support the development of such broad applications, robotics has evolved and must evolve, into a systematic approach that addresses all aspects of the design, manufacture, control and applications of robots.

A robot is a *reprogrammable multi-functional* manipulator designed to move materials, parts, tools, or specialized devices, through variable programmed motions for the performance of a variety of tasks. In short, robot is a programmable general purpose manipulator with sensors that can perform various tasks. More precisely, a robot in the 1990s is better defined as a machine which (1) is programmable; (2) works in contact with its environment, for example, a manipulator arm which performs "pick and place" motions, or "compliant" motion; and (3) behaves in an "intelligent" way, i.e., the robot senses and reacts to changes in its environment. A robot manipulator usually consists of several rigid (or flexible) links interconnected in series (or in parallel) by revolute or prismatic joints. One end of the chain is attached to a supporting base while the other end is free and carries a tool to manipulate objects or perform assembly tasks. The motion of the joints results in relative motion of the links. Thus, mechanically, a robot is composed of an arm with actuators, and an end-effector. The arm is designed to reach any workpiece located within its workspace, The latter is defined as the space of influence of a robot whose arm can deliver the end-effector to any point within this space. Based on the dimensionality of the robot workspace, one can define two-dimensional (planar) robots where the robot movements are restricted to two dimensional space, and three-dimensional robots where the manipulator can move freely in three dimensional space. In the case of three-dimensional manipulators, the arm usually has six degrees-of-freedom in order to achieve a desired position and orientation of the end-effector in three-dimensional space.

Robot manipulators so far have been used primarily for applications where the working environments of the manipulators are well arranged - called "artificial environments". Also, the workpiece that the manipulator is to grasp is placed at an "*easy*" location, and the path and the goal are both in an "*easy*" area within the workspace that is basically free of obstacles, or where the manipulator configuration is not in the neighborhood of singularities or joint limits. However, setting up environments that meet these assumptions often costs more than the manipulatois themselves, and this obviously limits the applications of manip.lators in industrial environments.

To overcome these limitations, it is necessary to develop manipulators that possess functions which allow them to perform more difficuit and sophisticated tasks which cannot usually be done by conventional manipulators. For example, avoiding collisions between a manipulator and objects in its workspace while maintaining proper end-effector trajectory tracking. Such manipulators are said to be dexterous and versatile. The need for dexterous and versatile robotic systems is becoming important for applications in space and undersea missions, and in hazardous environments.

Dexterity and versatility imply the mechanical ability to carry out various kinds of tasks in various situations. If a manipulator is to be dexterous and versatile, it should have more degrees of freedom than a conventional manipulator. For example, a dexterous planar manipulator should possess more than two degrees of freedom while a dexterous three dimensional manipulator should have more than six degrees of freedom. Such manipulators are said to be *redundant.* Furthermore, mechanical redundancy can be divided into *kinematic redundancy* and *actuation redundancy.* In a robotic system with *kinematic redundancy,* we are able to vary the configuration of the manipulator without changing the position and orientation of the end-effector, while *actuation redundancy* is only found in closed-chain mechanisms. In this thesis, we shall only deal with kinematic redundancy or kinematically redundant manipulators. Therefore, the term "redundancy" or "redundant manipulator" will be used to mean kinematic redundancy or kinematically redundant manipulator.

Generally speaking, there are a number of advantages that redundant manipulators have. As in the case of the human arm, they excel in versatility and applicability. More specifically, they have the potential to avoid singularities [11][14], avoid obstacles [2][10], avoid structural limitations (e.g., angle limits of a rotational joint), carry out reasonable actions, e.g., minimum energy motion, optimal velocity motion and minimum torque/force motion [5][12], reach behind an object, crawl into concave space, and so on. Redundancy can also be used to make a manipulator more reliable in the sense that it can perform certain tasks even after a failure of some joints [3], and more accurate [1]. Reliability and accuracy are particularly important in some applications such as space and undersea operations. Therefore, this steadily broadening field of applications determines the growing interest towards the analysis, design, and control of redundant robotic systems.

Of course, redundant manipulators have disadvantages as well. They possess more joints than conventional manipulators. Their structure is more complex, and the richness in choice of joint motions complicates the manipulator control problem considerably. In order to take full advantage of the capabilities of redundant manipulators, appropriate and effective control schemes need to be developed to utilized the redundancy in some useful manner.

Achieving high performance by applying advanced control techniques requires an increased understanding of the dynamics of robot manipulators. A crucial fact in the dynamic modeling of manipulators that has been realized by robotics researchers in recent years is the problem of *joint flexibility.* It has been shown [16][19] that control algorithms which assume a rigid model for the manipulator are limited in their applicability to real robots where the assumption of perfect rigidity is never satisfied exactly. For example, harmonic drives are used as actuators in many robot manipulators. It has been reported experimentally in [19] that the torsional flexibility in the drive system provides lightly damped oscillatory modes in the open-loop response of the manipulators. In addition, torque transducers, drive shaft stiffness or drive belts are considered as sources of joint flexibility for robot manipulators. Therefore, it is important to take joint flexibility into account in the modeling and design of robot controllers if high accurate performance is to be achieved.

1.2 MOTIVATION AND OBJECTIVES OF THE THESIS

Redundant manipulator control and flexible-joint manipulator control are two relatively complex problems in robotics field. Although a great deal of effort has been made and many papers have been published in these two areas, e.g. see [6][13][15][17][7][9][18][20], there still remain several problems which have not been solved adequately. Moreover, the issues of redundancy and joint flexibility can appear together when we deal with the problem of controlling a redundant flexible-joint manipulator. This combination may even create problems that have not been dealt with previously. These are challenging and difficult problems. The need to solve these problems has provided the motivation for the research described in this thesis.

Control of manipulator collision impact is a rather complicated problem which cannot be solved by directly applying a manipulator control scheme developed for free space tracking, or contact motion tracking. Using redundancy for collision impact reduction is a relatively new research topic [4][8][21]. A better understanding of this mechanical phenomenon is needed. and more comprehensive control strategies need to be developed to cope with this complex situation that arises as a result of collisions between a manipulator's end-effector and a workspace object. In this thesis, an impedance control based strategy for controlling collision impacts is presented. The control scheme consists of a *simplified impedance controller* and an *augmented configuration controller*. The purpose of the simplified impedance controller is to reduce both impulsive forces and rebound effects, while the augmented configuration controller is designed for the minimization of impulsive forces in the redundant manipulator.

Existing flexible-joint manipulator control strategies are only based in joint space and suitable for non-redundant manipulators [9][18][20]. However, redundant manipulators may also have joint flexibility, in particular when elastic actuators are used. In this circumstance, existing joint schemes for non-redundant manipulators may no longer be suitable for the redundant case. Therefore, it is necessary to develop Cartesian control schemes for redundant flexible-joint manipulators. In this thesis, a new control strategy called *hybrid Cartesian-joint control* is introduced, and important issues such as disturbances due to joint flexibility, manipulator self-motion control, and the relationship between disturbances arising from joint flexibility and self-motion are addressed. Stability of the proposed scheme is shown using the Lyapunov function approach.

Using kinematically redundant manipulators for obstacle avoidance, singularity avoidance, and kinematic optimization [2][10][11][13][14][17] has become a very active research area. However, almost no attention has been paid to the possibility of using kinematic redundancy in solving the problem of how a redundant manipulator can compensate for joint flexibility by appropriate configurations of the redundant manipulator. To address this problem in this thesis, a Cartesian space based control scheme for redundant flexible-joint manipulators is developed. The pseudo-inverse approach is used for redundancy resolution while the arbitrary vector is determined such that the redundancy is utilized to compensate for joint flexibility. The main idea behind this approach is to use the manipulator's self motion to "shape" the posture of the manipulator in such a way that the internal link motion eliminates the effect of the torsional force due to joint flexibility.

1.3 THESIS OUTLINE

Chapter 2: *Redundant Manipulators: Kinematics and Redundancy Resolution*

This chapter introduces some fundamental concepts which are extensively used throughout the thesis such as the concept of the kinematically redundant manipulators, and various issues in kinematic analysis and redundancy resolution. From a survey of a large number of publications, we also give an up-to-date review of the existing approaches of kinematic redundancy resolution. This forms the basis for the construction of the control strategies proposed in later chapters.

Chapter 3: *Dynamic Modeling of Rigid- and Flexible-Joint Manipulators*

In this chapter, we present the basic dynamic equations for a rigid-link, open-chain robot manipulator. The formulation of a manipulator dynamic model is given both in joint and Cartesian spaces. Also, we specify functional relationships between joint and Cartesian spaces for the nonlinear inertia matrix, Coriolis and centrifugal, gravitational, and frictional terms. Furthermore, the general formulation of the dynamic model for flexible-joint manipulators is introduced using the Lagrangian and the Newton-Euler approach. Finally, a simplified dynamic model for a flexible-joint manipulator, which will be used in this thesis, is discussed.

Chapter 4: *Impedance Control of Redundant Manipulators for Minimization of Collision Impact*

In this chapter, an impedance control based strategy for collision impact control of redundant manipulators is presented. The controller consists of a simplified impedance controller and an augmented configuration controller. The simplified impedance controller is designed to reduce the impulsive forces as well as the rebound effects. The augmented configuration controller is designed to minimize collision impulsive forces. Confirmation of the theoretical analysis is shown by computer simulations.

Chapter 5: *Cartesian Control of Redundant Flexible-Joint Manipulators*

In this chapter, the issue of redundant flexible-joint manipulator control is addressed. A new control strategy called hybrid Cartesian-joint control is introduced which consists of a Cartesian tracking controller, a link tracking controller, and a motor tracking controller. The construction of the proposed control scheme is based on the analysis of a flexible-joint non-redundant joint space scheme, as well as a Cartesian space scheme. Important issues such as the effect of disturbances caused by joint flexibility, control of the manipulator's self-motion, and the relationship between the disturbances and self-motion are also addressed. A stability analysis for the proposed controller is given, and computer simulations for verifying the performance of the proposed control strategy are presented.

Chapter 6: *Dynamic Control of Redundant Manipulators to Compensate for Joint Flexibility*

A general dynamic model of a redundant rigid/flexible-joint manipulator is introduced in this chapter. This allows us consider both rigid as well as flexible joints in a manipulator. Based on this model, a Cartesian space control scheme for rigid/flexible-joint redundant manipulators is developed. In this scheme, following the pseudo-inverse approach, redundancy is resolved to compensate against the effect of joint flexibility. To avoid possible algorithmic singularities, a modified damped least-squares approach is incorporated in the control scheme. Issues concerning the compensation capability and estimation of higher order derivatives are also discussed. Stability of the resulting closed-loop system is examined. Finally, numerical simulations are given to illustrate the application of the proposed control scheme.

Chapter 7: *Conclusions and Future Research*

Based on the proposed control strategies and algorithms for rigid as well as flexible joint redundant manipulators, general conclusions concerning the derivations, analysis, and results of this thesis are given in this chapter. Possible extensions of the results to existing and new problems in redundant manipulator control are also discussed.

1.4 REFERENCES

[1] J. Angeles, F. Ranjbaran, and R.V. Patel, "On the design of the kinematic structure of seven-axes redundant manipulators for maximum conditioning," in *Proc IEEE Int. Conf. Robotics Automat.*, pp. 494-499, 1992.

[2] J. Baillieul, "Avoiding obstacles and resolving kinematic redundancy," in *Proc. IEEE Int. Conf. Robotics and Automat.*, San Francisco, CAL., pp. 1698-1704, 1986.

[3] M.M. Bridges, D.M. Dawson, Z. Qu and S.C. Martindale, "Robust control of rigid-link flexible-joint robots with redundant joint actuators," submitted to *IEEE Trans. Syst., Man, and Cybern.*, 1992.

[4] M.W. Gertz, J.O. Kim and P. Khosla, "Exploiting redundancy to reduce impact force," *IEEE/RSJ Intern. Workshop on Intellig. Robots and Systems IROS'91*, pp. 179-184, 1991.

[5] J.M. Hollerbach and K.C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE J. of Robotics and Automat.*, vol. 3, pp. 308-315, 1987.

[6] P. Hsu, J. Hauser and S. Sastry, "Dynamic control of redundant manipulators," *Proc of IEEE Int. Conf. on Robotics and Automation*, pp. 183-187, 1988.

[7] K.Y. Lian, J.H. Jean and L.C. Fu, "Adaptive force control of single-link mechanism with joint flexibility," *IEEE Trans. Robotics. and Automat.*, vol. 7, no. 4, pp. 540-545, Aug. 1991.

[8] Z. Lin, R.V. Patel and C.A. Balafoutis, "Impedance control of redundant manipulators for minimization of collision impact," *Proc. American Control Conf.* pp. 1237-1238, 1992.

[9] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Trans. Automat. Contr.*, vol. 37, no. 2, pp. 174-181, Feb. 1992.

[10] A.A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robotics Res.*, vol. 4, no. 3, pp. 109-117, 1985.

[11] A.A. Maciejewski and C.A. Klein, "Numerical filtering for the operation of robot manipulators through kinematically singular configurations," *J. Robotic Systems*, vol. 5, no. 6, pp. 527-552, 1988.

[12] D.P. Martin, J. Baillieul and J.M. Hollerbach, "Resolution of kinematic redundancy using optimization techniques," *IEEE Trans. Robotics and Automat.*, vol. 5, no. 4, pp. 529-533, 1989.

[13] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading, MA., 1991.

[14] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *J. Dyn. Syst., Meas. Contr.*, vol. 108, pp. 163-171, 1986.

[15] D.N. Nenchev, "Redundancy resolution through local optimization: A review," *J Robotic Systems*, vol. 6, no. 6, pp. 769-798, 1989.

[16] E.I. Rivin, " Effective rigidity of robot structure: Analysis and enhancement," in *Proc. 1985 ACC*, Boston, MA., June 1985.

[17] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robotics and Automat.*, vol. 5, no. 4, pp. 472-490, 1989.

[18] M.W. Spong, K. Khorasani and P.V. Kokotovic, "An integral manifold approach to the feedback control of flexible joint robots," *IEEE J. Robotics, and Automat.*, vol.

RA-3, no. 4, pp. 291-300, Aug. 1987.

[19] L.M. Sweet and M.C. Good, "Re-definition of the robot motion control problem: Effects of plant dynamics, drive system constraints, and user requirements," in *Proc. 23rd IEEE Conf. Decision and Control*, Las Vegas, NV., pp. 724-731, Dec. 1984.

[20] P. Tomei, "A simple PD controller for robots with elastic joints," *IEEE Trans. Automat. Contr.*, vol. 36, no. 10, pp. 1208-1213, Oct. 1991.

[21] I.D. Walker, "The use of kinematic redundancy in reducing impact and contact effects in manipulation," *Proc. of IEEE Int. Conf. on Robotics and Automation*, Cincinnati, OH, pp. 434-439, 1990.

CHAPTER 2 REDUNDANT MANIPULATORS: KINEMATICS AND REDUNDANCY RESOLUTION

2.1 INTRODUCTION

The desire to obtain robot performance superior to that achievable with conventional manipulators has led researchers to study the capabilities of robot manipulators which possess more degrees-of-freedom than those of the conventional ones. This, therefore, led to the development of *redundant* manipulators that have shown great potentials in various applications. For instance, it is well known that while a non-redundant manipulator is very limited in performing a task for obstacle avoidance, a kinematically redundant manipulator may successfully perform this task. It has also been reported that the singular regions (the neighboring areas around singularities) significantly limit a manipulator's workspace [32]. Redundant manipulators can successfully overcome this problem as well. Furthermore, in addition to the main tracking task performed in Cartesian space, redundant manipulators are capable of optimizing various performance criteria such as joint limit avoidance, minimization of joint velocities, joint accelerations and joint torques, etc. In this chapter, we shall focus on the topic of kinematically redundant manipulators and their kinematic analysis. From a survey of a large number of publications, we shall also give a systematic review of the approaches of kinematic redundancy resolution. Based on this we shall construct the control strategies that are proposed in later chapters in this thesis.

The rest of the sections in this chapter are organized as follows: An introductory analysis for redundant manipulators and their kinematics is given in Section 2.2. In Section 2.3, a systematic review and analysis of redundancy resolution methodologies is presented, and Section 2.4 draws some conclusions.

2.2 REDUNDANT MANIPULATORS AND KINEMATIC ANALYSIS

A robot manipulator is said to be kinematically "redundant" if it possesses more degrees-of-freedom than are necessary for performing a specified task. For example, in two dimensional space, a planar manipulator with three joints is redundant for achieving any end-effector position; whereas the manipulator is non-redundant for tasks that involve positioning as well as orienting of the end-effector. Similarly, in three dimensional space, a manipulator with seven or more joints is redundant since six degrees-of-freedom are sufficient to position and orient the end-effector in any desired configuration. Furthermore, if we consider a broader class of robotic mechanisms (planar robot manipulators, mechanical wrists, fingers or legs, multi-arm cooperating robots, etc.) and a broader class of motion tasks, we say that any robotic mechanism is kinematically redundant whenever $n > m$, where n denotes the number of the degrees of freedom that the robot manipulator possesses, and m represents the number of task variables. The difference $n - m$ denotes the number of *degrees of redundancy*.

Redundancy for a robotic manipulator is a desirable characteristic since such a manipulator has increased dexterity and versatility due to the infinite number of joint motions which result in the same end-effector trajectory. This leads to a variety of applications such as, obstacle avoidance, singularity avoidance, minimum torque motion, etc. However, this richness in choosing the joint motions complicates the kinematic computation as well as the control of redundant manipulators considerably. On the other hand, it is noted that usually for a non-redundant robot manipulators the kinematic relations and the corresponding control strategies are relatively simple. For a prescribed end-effector trajectory and a given pose (such as elbow up or down), the motion of the manipulator is uniquely determined. However, when this motion is undesirable due to collision with obstacles, approaching kinematic singularities, or reaching joint limits, there is no freedom for the non-redundant manipulator to reconfigure itself so as to reach around obstacles, or to avoid singularities and joint limits. In order to take full advantage of the capabilities of redundant manipulators, our goal is to investigate new applications of redundant manipulators, and develop effective control schemes to utilize the redundancy for these applications.

Before going into the details of redundancy resolution, First we shall establish some basic kinematic relations for non-redundant as well as redundant manipulators. *Forward kinematics* are described by a nonlinear differentiable vector function Λ. For each joint space variable q, the function Λ provides the corresponding variable in Cartesian space, and this is expressed by the equation

$$x = \Lambda(q) \tag{2.2.1}$$

Then, the inverse function of Λ, which describes the inverse kinematics, can be defined symbolically by an equation of the form

$$q = \Lambda^{-1}(x) \tag{2.2.2}$$

This equation is known to describe a far more difficult problem than the forward kinematics in (2.2.1). In general, even for non-redundant manipulators, equation (2.2.2) is highly nonlinear. Usually, the inverse kinematics problem cannot be solved analytically (except for those manipulators which are of planar type, or for those manipulators with wrist-partitioned structure), and it is solved using iterative numerical procedures such as the Newton-Raphson method. As an alternative to these "position level" iterative methods for solving the inverse kinematics problem, we first discretise the Cartesian path and assume

q to be some initial joint value, then a small increment of q results in a small increment of x which can be expressed as $x + \Delta x = f(q + \Delta q)$. For a small value of Δx, we can express Δx in terms of Δq as

$$\Delta x = J_e(q)\,\Delta q \tag{2.2.3}$$

where $J_e(q)$ is defined as the manipulator's Jacobian matrix corresponding to the function $\Lambda(q)$, and can be written as

$$J_e(q) = \frac{\partial \Lambda(q)}{\partial q} \tag{2.2.4}$$

For a non-redundant manipulator, the dimensions of the joint variable q and the Cartesian variable x are the same, and thus, $J_e(q)$ is a square matrix. In this approach to inverse kinematics, the displacement Δq in joint space that will guide the end-effector in the direction Δx is determined by solving equation (2.2.3). For accurate path tracking in Cartesian space, the variables Δq and Δx must be infinitesimal quantities. This leads us to the equation

$$\dot{x} = J_e(q)\,\dot{q} \tag{2.2.5}$$

The use of this differential relationship to solve inverse kinematics was first introduced by Whitney [39], and is referred to as the *resolved motion rate control* method. Also, a second order differential relationship between joint and Cartesian space variables can be described by differentiating (2.2.5) with respect to time to get the equation

$$\ddot{x} = \dot{J}_e(q)\,\dot{q} + J_e(q)\,\ddot{q} \tag{2.2.6}$$

Thus, the inverse kinematics problem can also be solved using equation (2.2.6) at the acceleration level, and this method is referred to as *resolved acceleration control* [24].

Equations (2.2.1)-(2.2.6) are valid for redundant manipulators as well. Therefore, the above mentioned techniques can also be used to compute inverse kinematics for redundant manipulators. However, there is an important difference between non-redundant and redundant manipulators as far as the inverse kinematic problem is concerned. For non-redundant manipulators, a finite number of solutions is obtained, while for redundant manipulators, an infinite number of solutions exist. This implies that one must specify a functional form of the redundant degrees of freedom which characterize the particular resolution of redundancy when the inverse kinematic problem is to be solved. This leads us to the problem of redundancy resolution which will be introduced in more details in the next section.

2.3 REDUNDANCY RESOLUTION

Manipulator redundancy resolution is a way to specify a functional form of the redundant degrees of freedom based on some objective function(s) such that, among the infinite number of choices, a specific solution of the inverse kinematics problem is determined. This has been a research topic that has attracted considerable attention in recent years [20][30], and has resulted in a number of different approaches for redundancy resolution.

In order to be able to analyze and evaluate the methods for resolving kinematic redundancy in a systematic way, it is necessary to classify them according to some common criteria, properties or underlying mathematical formulations. There are a number of ways for classifying the methodologies which resolve kinematic redundancy. For example, one can classify the methods based on performance criteria such as kinematic optimality criteria (e.g., manipulability index [40]) or kinetic criteria (e.g., joint torques). Alternatively, one may borrow terminologies from optimization theory and consider a classification into two classes that utilize (1) *local optimization* approaches and (2) *global optimization* approaches. These classifications may have some advantages, but they have no direct connections with the control strategies used for redundant manipulators. Since the aim of this thesis is to develop control strategies for redundant manipulators, we shall consider another classification where the terminology is based on control techniques. In particular, the classification is based on control strategies that incorporate end-effector measurements such as those based on *resolved-position*, *resolved-velocity*, and *res lved-acceleration*. Depending on which equation (i.e., equation (2.2.1), (2.2.5) or (2.2.6)) has been chosen as the basis for solving the inverse kinematics problem, we may say that the redundancy has been resolved at position, velocity or acceleration level. Thus, all of the methodologies which resolve kinematic redundancy will be divided into the following categories: *(I) Redundancy resolution at the position level*, *(II) Redundancy resolution at the velocity level*, and *(III) Redundancy resolution at the acceleration level.*

In the following sections, we provide a brief overview of a number of redundancy resolution methodologies, and highlight the most important approaches which will form the basis for the development of our redundant manipulator control schemes.

2.3.1 Redundancy Resolution at the Position Level

Redundancy resolution at the position level includes methods which "directly" map Cartesian space variables into joint space variables. Symbolically this mapping is defined by equation (2.2.2) where the joint space variables are expressed in terms of the Cartesian space variables, manipulator's kinematic parameters, and performance criteria for utilizing the extra degrees of freedom. There are mainly three approaches that can be categorized in this class, namely, **a**) analytic approaches, **b**) iterative approaches, and **c**) configuration control approaches.

a) *Analytic Approaches*

Analytic approaches can be regarded as a generalization of the analytic solutions of the inverse kinematics of non-redundant manipulators. In this approach, the main objective is to express the joint space variables as explicit functions of Cartesian space variables such that given performance criteria are satisfied. The "inverse function" approach which has been proposed by Baker and Wampler [6] [7] is probably the best known analytic approach for resolving kinematic redundancy. This approach is basically a generalization of the closed-form solution to the inverse kinematics problem for non-redundant manipulators, and consists of two main steps: (1) based on a global view of the Cartesian space, an "invertiable" or "feasible" workspace (i.e., a space where there are no singularities) is selected; (2) a differentiable inverse kinematic function g ($g \equiv \Lambda^{-1}$) is defined such that $\Lambda [g(x)] = x$ for all x in the feasible workspace. This is used to construct a "tracking algorithm" for the function Λ. To accomplish these two steps, Wampler gives some simple examples and procedures to derive a simply connected invertible workspace as well as the inverse function g. Comparing with other possible approaches for resolving kinematic redundancy, there are two main reasons for considering the inverse function approach. First it allows real-time path corrections and second, it is *cyclic* or repeatable in a proper domain. Some of the shortcomings of the approach are: (i) it restricts the workspace of the manipulator; (ii) there is no general procedure for constructing (and optimizing) inverse kinematic functions; and (iii) the use of inverse functions does not allow for any readjustment (self-motion) during the motion. Other examples of analytic approaches can be found in [8][13][22].

b) *Iterative Approaches*

Iterative approaches are based on numerical methods for solving a nonlinear set of algebraic equations. Usually, these equations consist of the equations of forward kinematics and additional equations which optimize some performance criteria. One elegant method for resolving kinematic redundancy is that proposed in [11], and is formulated as a constrained optimization problem. This method can be outlined as follows. First, by using Lagrangian multipliers, the constrained minimization problem is recast as an unconstrained minimization problem. Then, assuming that the manipulator Jacobian matrix is of full rank, a solution is devised based on the nonlinear system of the n algebraic equations consisting of the forward kinematics equation $x = \Lambda(q)$ and a homogeneous equation $Zh = 0$, where h is the gradient vector of the constraint function. The matrix Z is composed of the $n-m$ linearly independent vectors which span the null space of the manipulator's Jacobian matrix. Therefore, the homogeneous equation characterizes the self-motion of the redundant manipulator. The n algebraic equations can be solved iteratively using numerical methods. It should be noted that the formulation of redundancy resolution in this approach is very versatile since the matrix Z depends only on the manipulator Jacobian while the vector h depends only on the desired performance criteria. A potential drawback of the method is the computational complexity since the definition of Z involves selecting and inverting a full rank submatrix based on the manipulator Jacobian J_e. Some other approaches in this category can be found in [1][16].

c) *Configuration Control Approaches*

The main idea behind configuration control approaches (also called augmented kinematics approaches) [33][34] is that first, the forward kinematics function is augmented in such a way that its Jacobian matrix is square, and second, the resultant kinematics realizations are formulated as trajectory tracking problems which can be solved on line using linear or non-linear control theory. The number of kinematic functions in Cartesian or joint space is chosen [33] to reflect the desired additional tasks that will be performed using redundancy. The augmented kinematic function can be viewed as a parametrization of the manipulator "self-motion". By self-motion, we mean the internal movement of the link which does not result in motion of the end-effector. In other words, given an end-effector position/orientation and the augmenting kinematic functions, the additional tasks are used to "shape" the manipulator's configuration. In this approach, the end-effector Cartesian coordinates and the kinematic functions are combined together to form a set of "configuration variables" which describe the physical configuration of the entire manipulator in Cartesian space. This task augmentation at the position level produces a kinematic representation of the manipulator which is no longer redundant, i.e., the dimension of the augmented task space is identical to that of the joint space. Then, conventional techniques for control of non-redundant manipulators are applied directly to achieve task space trajectory tracking.

Since the configuration control approach will form the basis of our control schemes in later chapters, we elaborate more on this approach. In particular, we describe a kinematic optimization scheme which can be formulated as a tracking problem within the framework of configuration control at position level. To see this, let us denote by $L(q)$ the scalar kinematic performance criterion to be optimized by the utilization of redundancy. Then, applying gradient projection optimization theory [19] to $L(q)$, subject to the end-effector constraint $\dot{x} = J_e\dot{q}$, the optimality criterion for the constrained optimization is given [33] by

$$(I - J_e^{\dagger} J_e) \frac{\partial L(q)}{\partial q} = 0 \tag{2.3.2}$$

where $J_e^{\dagger}$ is the pseudo-inverse of the manipulator end-effector Jacobian matrix J_e. The matrix $(I - J_e^{\dagger} J_e) \in \Re^{n \times n}$ is of rank r, where r denotes the degrees of redundancy, and therefore equation (2.3.2) can be reduced to

$$N_e \frac{\partial L(q)}{\partial q} = 0 \tag{2.3.3}$$

where $N_e \in \Re^{r \times n}$ is formed from r linearly independent rows of $(I - J_e^{\dagger} J_e)$. Notice that the rows of N_e span the r-dimensional null-space of J_e. In the configuration control approach, one defines the additional task based on the r kinematic functions of equation (2.3.3) as $z(q) = N_e \frac{\partial L(q)}{\partial q}$ and the desired trajectory as $z_d(t) \equiv 0$. Then, by combining the end-effector trajectory with the additional tasks, the augmented configuration vector y and the augmented Jacobian matrix J are defined as

$$y = \begin{bmatrix} x \\ \cdots\cdots \\ N_e \dfrac{\partial L}{\partial q} \end{bmatrix} \qquad\qquad J = \begin{bmatrix} J_e \\ \cdots\cdots\cdots\cdots \\ \dfrac{\partial}{\partial q}\left(N_e \dfrac{\partial L}{\partial q}\right) \end{bmatrix} \qquad (2.3.4)$$

Finally, a control scheme is applied to ensure that $y(t)$ tracks the desired trajectory $y_d(t) = \begin{pmatrix} x_d(t) \\ z_d(t) \end{pmatrix}$ and at the same time, the kinematic redundancy is resolved such that the optimal kinematic function $L(q)$ is optimized by the manipulator's self-motion.

The most important features of the configuration control approach are the following: (1) By directly controlling the manipulator in Cartesian space, the complicated and time consuming inverse kinematics computations are avoided. (2) Repeatability of the manipulator's motion is maintained. (3) Because it is a Cartesian space scheme, the method can be readily extended to hybrid force and position control.

2.3.2 Redundancy Resolution at the Velocity Level

Redundancy resolution at the velocity level involves finding a set of joint velocities $\dot{q}$ which satisfies equation (2.2.5) for a given end-effector velocity vector $\dot{x}$. This approach is applicable to general manipulators, and is usually well defined in a manipulator's workspace except possibly for some numerical difficulties in the neighborhood of the manipulator's singularities where special care must be taken.

From a linear algebra point of view, depending on the number of joints and the dimension of the task space, Solution of the inverse kinematics problem at the velocity level may have "no solutions" when $m > n$, a unique solution when $m = n$, or an infinite number of solutions when $m < n$. In practice, in the case that there is no solution, we usually wish to find the joint velocity which minimizes the error $\|J_e\dot{q} - \dot{x}\|$, and in the case that there is an infinite number of solutions, we would like to find the minimum norm solution $\|\dot{q}\|$. However, we have $m < n$ in the redundant manipulator case, and thus the problem of redundancy resolution results in the solution of an underdetermined linear system shown in equation (2.2.5). From a mathematical point of view, the solution of this linear system is usually composed of a particular solution, and a general solution or a non-redundant system based solution. In the following, we shall give a brief introduction to some important methodologies in this category.

a) *Pseudo-inverse based particular solution*

A particular solution for the linear system of equation (2.2.5) is the minimum norm solution which is defined by the equation

$$\dot{q} = J_e^{\dagger}(q)\,\dot{x} \qquad (2.3.5)$$

where the pseudo-inverse $J_e^{\dagger}$ [12] [9] of the Jacobian matrix J_e is a generalized inverse which satisfies the equations

$$J_e J_e^{\dagger} J_e = J_e \qquad J_e^{\dagger} J_e J_e^{\dagger} = J_e^{\dagger}$$

$$(J_e^{\dagger} J_e)^T = J_e^{\dagger} J_e \qquad (J_e J_e^{\dagger})^T = J_e J_e^{\dagger} \qquad (2.3.6)$$

With this definition, the pseudo-inverse $J_e^{\dagger}$ for the underdetermined linear system of equation (2.2.5) with J_e of full row rank is explicitly defined as

$$J_e^{\dagger} = J_e^T (J_e J_e^T)^{-1} \qquad (2.3.7)$$

Applications of the pseudo-inverse approach to redundancy resolution has been thoroughly reviewed by Klein and Huang [20]. The most important observation they made in their review is that control of kinematically redundant manipulators based on the pseudo inverse approach produces a drift in joint space even when a cyclic task is performed in

Cartesian space. Another drawback of the pseudo-inverse approach is that it does not avoid singularities. Some other approaches in this category can be found in [21][35][36].

b) *Weighted pseudo-inverse techniques*

The weighted pseudo-inverse approach was first proposed by Whitney [39] who considered the following solution to equation (2.2.5)

$$\dot{q} = (J_e)_w^{+}\dot{x} \tag{2.3.8}$$

where the matrix $(J_e)_w^{+}$ is a weighted pseudo-inverse of the Jacobian matrix J_e, and is defined as

$$(J_e)_w^{+} = \mathcal{W}^{-1} J_e^T (J_e \mathcal{W}^{-1} J_e^T)^{-1} \tag{2.3.9}$$

where $\mathcal{W}$ is a positive definite weighting matrix. The weighted pseudo-inverse approach is essentially similar, although not equivalent [35][36], to the unweighted pseudo-inverse approach. Thus, for example, it does not introduce algorithmic singularities, and its global behavior is the same as that of an unweighted pseudo-inverse approach, i.e., most of the time the resulting motion is not cyclic.

c) *Pseudo-inverse based general solution*

The pseudo-inverse based general solution of a kinematically redundant manipulator was first suggested by Liegeois [23], and has the form

$$\dot{q} = J_e^{+}\dot{x} + (I - J_e^{+}J_e)\,\xi \tag{2.3.10}$$

where ξ is an arbitrary vector which can be used for redundancy resolution, and the matrix $(I - J_e^{+}J_e)$ is a projection matrix which projects the arbitrary vector ξ onto the null space of J_e. Hence, the term $(I - J_e^{+}J_e)\xi$ in equation (2.3.10) stands for the homogeneous solution of the equation $J_e\dot{q} = 0$ while the first term on the right-hand side of equation (2.3.10) corresponds to the particular solution. As can be seen in equation (2.3.10), the pseudo-inverse based general solution is decoupled in the sense that the particular solution is used to realize the desired end-effector velocity $\dot{x}$, while the homogeneous solution contributes tc a motion in joint space only, the so called self-motion of the manipulator.

d) *Damped least-squares techniques*

Application of the damped least-squares techniques to redundant manipulators resulted from application to the singularity avoidance problem. As mentioned above, the pseudo-inverse based solution for redundancy resolution in general does not avoid singularities. To overcome this difficulty, the damped least-squares formulation was independently proposed by Wampler [38], and Nakamura and Hanafusa [28]. In [28] the method is referred to as a *singularity robustness method.* The basic idea in the damped least-squares method is to balance the cost of a large residual error (i.e., the end-effector tracking error) against the cost of a large solution. The singularity avoidance property can be achieved using a well-conditioned formulation based on weighting the accuracy of tracking of the end-effector velocity with the norm of the joint velocity. In other words, in this formulation we minimize the sum

$$\|\dot{x} - J_e\dot{q}\|^2 + \alpha\|\dot{q}\|^2 \qquad (2.3.11)$$

where α is the damping factor. Thus, a damped least-squares pseudo-inverse can be defined [38][28] as

$$(J_e)_d^{\dagger} = J_e^T(\alpha I + J_e J_e^T)^{-1} \qquad (2.3.12)$$

or

$$(J_e)_d^{\dagger} = (\alpha I + J_e^T J_e)^{-1} J_e^T \qquad (2.3.13)$$

As can be seen from equations (2.3.12) and (2.3.13), independent of the rank of J_e, the matrix $(\alpha I + J_e J_e^T)$ (or matrix $(\alpha I + J_e^T J_e)$) is ensured to be of full rank. This implies that even if the n inipulator's configuration is in the neighborhood of a singularity where J_e loses rank, the solution of the pseudo-inverse still exists. Notice that the first equation (2.3.12) is simpler from the computational point of view [28] because in equation (2.3.12) the matrix inversion of a smaller matrix than that of (2.3.13) is required. As can be seen, the damped least-squares pseudo-inverse $(J_e)_d^\dagger$ approaches the pseudo-inverse $J_e^\dagger$ when the damping factor α approaches zero.

In the formulation of the damped least-squares pseudo-inverse, the damping factor α plays an important role. Wampler [38] used a fixed value for α, derived from a reasonable bound on the change in residual error. However, it is desirable to obtain a larger damping factor near singularities, and a smaller or zero damping factor for nonsingular regions. Thus, in [28] the following automatic adjustment technique for α was proposed:

$$\alpha = \begin{cases} \alpha_o \left(1 - \frac{h}{h_o}\right) & h < h_o \\ 0 & h \geq h_o \end{cases} \tag{2.3.14}$$

where

$$h = \sqrt{det(J_e J_e^T)} \tag{2.3.15}$$

is the manipulability measure as defined by Yoshikawa [40], h_o is a threshold that defines the boundary of the neighborhood of singular points, and α_o is the value of the damping factor at singular points. Other versions of automatic damping factor adjustment can be found in [18][10][25].

The damped least-squares technique provides a useful approach for handling critical situations in the neighborhood of singularities. It is also a useful approach for our redundant flexible-joint manipulator scheme. A damped least-squares based technique will be developed to avoid algorithmic singularities in Chapter 6.

e) *Extended Jacobian technique*

This method for obtaining a generalized inverse of the Jacobian matrix has been proposed by Baillieul [3][4][5]. A similar idea has also been suggested by Oh et al. [31]. This approach is based on the introduction of $n-m$ additional holonomic constrained functions which are normally used to specify various performance criteria such as singularity or obstacle avoidance into the underdetermined system of equation (2.2.5), thus extending the dimension of the task space, and making it equal to that of the joint space. As in the other techniques, this approach can also be reformulated in terms of particular and homogeneous parts of the solution. We note that although the final formulation is different, this approach may be considered as a "linearized" version of the configuration control approach.

2.3.3 Redundancy Resolution at the Acceleration Level

Redundancy resolution at the acceleration level involves finding a set of joint accelerations $\ddot{q}$ which satisfies equation (2.2.6) for a given end-effector configuration acceleration $\ddot{x}$. This acceleration level approach is in fact a generalization of the redundancy resolution approach at the velocity level, and is introduced in order to incorporate manipulator dynamics (kinetic energy, joint torques, etc.) for better utilization of redundancy. Most techniques of resolving redundancy at the acceleration level are generalization of velocity level techniques. The most popular techniques in this category are optimization based ones and those that are incorporated in the design of dynamic controllers (dynamic control methods).

a) *Optimization based approach*

Most of the optimization methods discussed so far in the literature are formulated by utilizing Lagrange multipliers, Pontryagin's maximum principle, or the calculus of variations, and other iterative numerical methods [27][26][14][17][37]. Based on the forms and the nature of the criteria functions used in the optimization, we can further classify these optimization techniques into local and global ones. Usually, when a criterion function is defined as the gradient vector of a scalar function that evaluates a global performance index, in a sense the redundancy is utilized instantaneously (i.e., locally), and the solution is not a global optimal solution. For a global optimal solution, the performance criterion needs to be of an integral-type which is evaluated over the entire duration of the motion. The advantages of instantaneous optimization methods over global ones are obviously the simplicity of the formulations and significantly less computational effort. However, local techniques have problems with producing cyclic motion [20], avoiding singularities [2], and reaching global optima [20]. On the other hand, global methods ensure the global optimality but require a large amount of computation.

b) *Dynamic control method*

Because of the structure of the robot manipulator dynamics, redundancy resolution at the acceleration level is suitable for incorporation into the design of manipulator controllers. For example, using an acceleration level redundancy resolution approach an optimum control method is proposed in [29]. An elegant dynamic control scheme, similar to the *resolved acceleration control* [24], was proposed by Hsu et al. [15] which guarantees the tracking of a given end-effector trajectory while provides for the control of the redundant joint velocities. This approach is appealing because some more sophisticated control strategies such as robust or adaptive control schemes can be developed based on this *pseudo-inverse based resolved acceleration control.* Also, we shall see later in this thesis, the pseudo inverse based resolved acceleration control can be extended to the control of flexible-joint manipulators.

The control law proposed in [15] can be written as

$$\tau = D\,[J_e^{\dagger}\,(\ddot{x}_d + \mathcal{K}_v\dot{e} + \mathcal{K}_p e - \dot{J}_e\dot{q}) + \varphi_N] + C + G \qquad (2.3.16)$$

where $e = x_d - x$ is the tracking error, $\mathcal{K}_p$ and $\mathcal{K}_v$ are position and velocity gain matrices, and φ_N is an arbitrary vector lying in the null space of the manipulator Jacobian matrix. The terms D, C, G and τ denote the manipulator's inertia matrix, Coriolis and centrifugal forces, gravitational force, and joint torque vector respectively. In [15], it is shown that the control law of equation (2.3.16) guarantees the convergence of Cartesian tracking error e. However, because the scheme is Cartesian in nature, the joint velocity corresponding to the self-motion (in the null space of Jacobian) becomes unobservable. Thus, the system may result in some undesirable behavior, or even become unstable unless we use the vector φ_N to control the null space joint velocity. As suggested in [15], a way to provide control for the joint space velocities in the null space of J_e for achieving good system performance is to define a vector g, and let the null space joint velocity track the projection of g onto the null space of J_e. As a demonstration in [15], a specific null space vector φ_N was selected in such a way that the manipulator avoids singularity while maintaining Cartesian space end-effector tracking.

2.4 CONCLUSIONS

In this Chapter, we have discussed the issues of kinematic redundancy and its resolution. A comprehensive overview has also been given, which covers position-level, velocity-level and acceleration-level kinematic redundancy resolution methodologies. Furthermore, an in-depth analysis of how redundancy resolution can be best incorporated in the design of manipulator dynamic controllers, or in optimization schemes has paved the framework for developing control strategies in later chapters in this thesis for the control of rigid- and flexible-joint redundant manipulators.

2.5 REFERENCES

[1] J. Angeles, M. Habib and C.S. Lopez-Cajun, "Efficient algorithms for the kinematic inversion of redundant robot manipulators," *Int. J. Robotics Automat.*, vol. 3, pp. 106-116, 1987.

[2] J. Baillieul, J.M. Hollerbach and R. Brockett, "Programming and control of kinematically redundant manipulators," in *Proc. 23rd IEEE Conf. on Decision and Control*, Las Vegas, NV., pp. 768-774, 1984.

[3] J. Baillieul, "Kinematic programming alternatives for redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automat.*, St. Louis,MI., pp. 722-728, 1985.

[4] J. Baillieul, "Avoiding obstacles and resolving kinematic redundancy," in *Proc. IEEE Int. Conf. Robotics and Automat.*, San Francisco, CAL., pp. 1698-1704, 1986.

[5] J. Baillieul, "A constrained oriented approach to inverse problems for kinematically redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Raleigh, NC., pp. 1827-1833, 1987.

[6] D.R. Baker and C.W. Wampler, "Some facts concerning the inverse kinematics of redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Raleigh, NC., pp. 604-609, 1987.

[7] D.R. Baker and C.W. Wampler, "On the inverse kinematics of redundant manipulators," *Int. J. Robotics Res.*, vol. 7, no. 2, pp. 3-21, 1988.

[8] M. Benati, P. Morasso, and V. Tagliasco, "The inverse kinematic problem for Anthropomorphic manipulator Arms," *J. Dyn. Syst. Meas. Contr.*, vol. 104, pp. 110-113, 1982.

[9] T.L. Boullion and P.L. Odell, *Generalized Inverse Matrices*, John Wiley & Sons, New York, 1971.

[10] S.K. Chan and P.D. Lawrence, "General inverse kinematics with the error damped pseudo-inverse," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Philadephia, PA., pp. 834-839, 1988.

[11] P.H. Chang, "A closed-form solution for inverse kinematics of robot manipulators with redundancy," *IEEE Trans. Robotics Automat.*, vol. RA-3, no. 5, pp. 393-403, 1987.

[12] G.H. Golub and Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins Univ.

Press, Baltimore, 1989.

[13] J.M. Hollerbach, "Optimum kinematic design for a seven degrees of freedom manipulator," in *Proc. 2nd Int. Symp. in Robotics Research*, pp. 215-222, H. Hanafusa and H. Inoue, Eds., Cambridge, MA: MIT Press, 1985.

[14] J.M. Hollerbach and K.C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE J. Robotics Automat.*, vol. RA-3, pp. 308-315, 1987.

[15] P. Hsu, J. Hauser and S. Sastry, "Dynamic control of redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Philadephia, PA., pp. 183-187, 1988.

[16] P.J. Jouaneh and S. Rangwala, "Inverse kinematic solutions for redundant manipulators using simulated annealing," in *Proc. IEEE/RSJ Int. Workshop on Intellig Robots and Systems IROS'91*, Osaka, Japan, pp. 1647-1652, 1991.

[17] K. Kazerounian and Z. Wang, "Global versus local optimization in redundancy resolution of robotic manipulators," *Int. J. Robotics Res.*, vol. 7, no. 5, pp. 3-12, 1988.

[18] L.Kelmar and P.K. Khosla, "Automatic generation of kinematics for a reconfigurable modular manipulator system," in *Proc. IEEE Int. Conf. Robotics and Automat.* Philadephia, PA , pp. 663-668, 1988

[19] D.E. Kirk, *Optimal Control Theory: An Introduction.* Englewood Cliffs, NJ., Prentice-Hall, 1970.

[20] C.A. Klein and C.H. Huang, "Review of pseudo-inverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 245-250, 1983.

[21] C.A. Klein and K.B. Kee, "The nature of drift in pseudo-inverse control of kinematically redundant manipulators," *IEEE Trans. Robotics Automat.*, vol. 5, pp. 231-234, 1989.

[22] S. Lee and A.K. Bejczy, "Redundant arm kinematic control based on parameterization," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Sacramento, CA., pp. 458-464, 1991.

[23] A. Liegeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, no. 12, pp. 868-871, 1977.

[24] J.Y. Luh, M.W. Walker and R.P. Paul, "Resolved acceleration control of mechanical manipulators," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 468-474, 1980.

[25] A.A. Maciejewski and C.A. Klein, "Numerical filtering for the operation of robot manipulators through kinematically singular configurations," *J. Robotic Systems*, vol. 5, pp. 527-552, 1988.

[26] D.P. Martin, J. Baillieul and J.M. Hollerbach, "Resolution on kinematic redundancy using optimization techniques," *IEEE Trans. Robotics Automat.*, vol. 5, pp. 529-533, 1989.

[27] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading, MA., 1991.

[28] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *J. Dyn. Syst., Meas. Contr.*, vol. 108, pp. 163-171, 1986.

[29] Y. Nakamura and H. Hanafusa, "Optimal redundancy control of robot manipulators," *Int. J. Robotics Res.*, vol. 6, no. 1, pp. 32-42, 1987.

[30] D.N. Nenchev, "Redundancy resolution through local optimization: A review," *J. Robotic Systems*, vol. 6, no. 6, pp. 769-798, 1989.

[31] S.Y. Oh, D. Orin and M. Bach, "An inverse kinematic solution for kinematically redundant robot manipulators," *J. Robotic Systems.*, vol. 1, no. 3, pp. 235-249, 1984.

[32] R.P. Paul and C.N. Stevenson, "Kinematics of robot wrists," *Int. J. Robotics Res.*, vol. 2, pp. 31-38, 1983.

[33] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robotics Automat.*, vol. 5, pp. 472-490, 1989.

[34] H. Seraji, "Task-based configuration control of redundant manipulators," *J. Robotic*

Systems, vol. 9, pp. 411-451, 1992.

[35] T. Shamir, "Remarks on some dynamical problems of controlling redundant manipulators," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 341-344, 1990.

[36] T. Shamir and Y. Yomdin, "Repeatability of redundant manipulators: Mathematical solution of the problem," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 1004-1009, 1988.

[37] K.C. Suh and J.M. Hollerbach, "Local versus global torque optimization of redundant manipulators," in *Proc. IEEE Int. Conf. on Robotics and Automat.*, Raleigh, NC., pp. 619-624, 1987.

[38] C.W. Wampler, "Manipulator inverse kinematics solutions based on vector formulations and damped least-squares methods," *IEEE Trans. Syst., Man, Cybern.*, vol. 16, pp. 93-101, 1986.

[39] D.E. Whitney, "Resolved motion rate control of manipulator and human prostheses," *IEEE Trans. Man-Machine Syst.*, vol. 10, pp. 47-53, 1969.

[40] T. Yoshikawa, "Manipulability and redundancy control of robotic mechanisms," in *Proc. IEEE Int. Conf. on Robotics and Automat.*, St. Louis, Missouri, pp. 1004-1009, 1985.

CHAPTER 3 DYNAMIC MODELING OF RIGID- AND FLEXIBLE-JOINT MANIPULATORS

3.1 INTRODUCTION

Manipulator dynamics deals with the study of the forces and torques required to cause motion of the manipulator links. In order to accelerate or decelerate the manipulator links from some initial velocity to certain final velocity, the joint actuators must provide certain torque (or force) functions. These functions could be complicated functions of the path to be followed by the end-effector as well as the inertial properties of the links and the manipulator's payload, etc. Manipulator dynamic modeling and analysis are fundamental steps towards the development of control schemes.

The main topic of this thesis is the control of kinematically redundant manipulators. As mentioned previously, the kinematic analysis of redundant and non-redundant manipulators is different. However, there is no distinctive difference between non-redundant and redundant manipulators from point of view of dynamic analysis. The only difference that appears is when we convert a joint-space dynamic model into a Cartesian space dynamic model. This is due to the involvement of the Jacobian matrix. Hence, in this chapter we will not specifically distinguish between redundant and non-redundant dynamic models. we shall focus mainly on the problem of dynamic modeling for rigid-joint as well as flexible-joint manipulators. This will be used as the basis for the development of rigid-joint and flexible-joint manipulator controllers. Before going into the details of dynamic modeling, some assumptions that will hold throughout this thesis are stated below.

(a) The manipulators (redundant and non-redundant) under discussion will be assumed to be open kinematic chains with revolute joints. The control schemes to be developed can be extended to the cases where revolute as well as prismatic joints appear in a manipulator system.

(b) The manipulators are modeled as jointed rigid bodies. This implies that link flexibility is not considered in the modeling. Therefore, the word *rigid-joint manipulator* means that the manipulator's joints as well as links are rigid. The word *flexible-joint manipulator* implies that all the joints are flexible but the links are rigid. The term *rigid/flexible-joint manipulator* means that the rigid manipulator links are interconnected by rigid as well as flexible joints.

(c) Although all the computer simulations of the proposed control strategies will be carried out in digital computer systems, the theoretical work is based on a continuous time analysis.

(d) It is assumed that all the dynamic and kinematic parameters of the rigid- and flexible-joint manipulator are known *a priori*.

(e) The desired manipulator Cartesian trajectories are assumed to be known, including first and second derivatives for rigid-joint manipulator control, and first to the fourth derivatives for flexible-joint manipulator control.

The rest of the sections are arranged as follows. In Section 3.2, rigid-joint manipulator dynamic model expressed in joint space is discussed, while the Cartesian version of the model is analyzed in Section 3.3. A general dynamic model for a flexible-joint manipulator is formulated in Section 3.4, and a simplified version corresponding to the general model is discussed in Section 3.5. Finally, Section 3.6 draws some conclusions.

3.2 RIGID-JOINT MANIPULATOR DYNAMICS IN JOINT SPACE

A rigid-joint robot manipulator can be modeled as a set of n moving rigid links connected in a serial chain with one end fixed to the ground and the other end free. The links are joined together with revolute (or prismatic) joints and there is no flexibility in the joints. Several methods are available for formulating the dynamic behavior of a rigid-joint manipulator [6][9][1]. The recursive Newton-Euler formulation [16] leads to a dynamical system of equations which is computationally one of the most efficient dynamic models available. It involves the successive transformation of velocities and accelerations from the base of the manipulator to the end-effector using relationships between the generalized coordinate systems. Forces are then transformed backward from the end-effector to the base to obtain the joint torques. The complete derivation of the Newton-Euler formulation can be found in [6][9][1]. The dynamics of a rigid-joint manipulator can be written in the form of a vector equation [1] as follows:

$$\tau = D_l(q)\ddot{q} + C_l(q, \dot{q}) + G_l(q) + F_f(\dot{q}) + J_e F \qquad (3.2.1)$$

where

$\tau \in \Re^{n \times 1}$: is the vector of joint torques supplied by the actuators,

$q \in \Re^{n \times 1}$: is the vector of joint positions with $q = [q_1, q_2, \ldots, q_n]^T$,

$D_l(q) \in \Re^{n \times n}$: is the manipulator's inertia matrix,

$C_l(q, \dot{q}) \in \Re^{n \times 1}$:represents the torques arising from centrifugal and Coriolis forces,

$G_l(q) \in \Re^{n \times 1}$: represents the torques due to gravity,

$F_f(\dot{q}) \in \Re^{n \times 1}$: represents the torques due to friction acting at the manipulator's joints,

$F \in \Re^{m \times 1}$: denotes external forces and moments acting on the end-effector.

For convenience, sometimes the Coriolis and centrifugal vector $C(q, \dot{q})$ can be rewritten in the form such that the centrifugal terms are separated from the Coriolis terms. Therefore, assuming free space motion, and that there is no friction, the dynamic equation can be expressed as

$$\tau = D_l(q)\ddot{q} + \{C_1(q)\,[\dot{q}\dot{q}] + C_2(q)\,[\dot{q}^2]\} + G_l(q) \qquad (3.2.2)$$

where $C_1(q) \in \Re^{n \times (n(n-1)/2)}$ is a matrix of Coriolis terms, $[\dot{q}\dot{q}] \in \Re^{(n(n-1)/2) \times 1}$ is a vector of joint velocity products expressed by

$$[\dot{q}\dot{q}] = [\dot{q}_1\dot{q}_2, \dot{q}_1\dot{q}_3, \ldots, \dot{q}_{n-1}\dot{q}_n]^T. \qquad (3.2.3)$$

$C_2(q) \in \Re^{n \times n}$ is a matrix of centrifugal terms, and $[\dot{q}^2] \in \Re^{n \times 1}$ is a joint velocity product vector which can be written as

$$[\dot{q}^2] = [\dot{q}_1^2, \dot{q}_2^2, \ldots, \dot{q}_n^2] \qquad (3.2.4)$$

Next, in order to get a better idea of the dynamics of robot manipulators, we discuss the main terms appearing in (3.2.1).

a) *The manipulator inertia matrix $D_l(q)$*

The kinetic energy of a manipulator can be written in a quadratic form [9][2] as

$$KE = \frac{1}{2}\dot{q}^T D_l(q)\dot{q} \qquad (3.2.5)$$

Here, the matrix $D_l(q)$ reflects the mass distribution of a manipulator as a function of the joint vector q. Each element of $D_l(q)$ has units of inertia (kgm^2). To ensure that the quantity in equation (3.2.5) is always positive and represents energy, $D_l(q)$ must be a positive-definite matrix. On the other hand, the potential energy of a manipulator is described by a scalar function of joint positions only, say, a function $\mathcal{P}(q)$. Therefore, the Lagrangian of the manipulator [2] can be written as

$$L = \frac{1}{2}\dot{q}^T D_l(q)\dot{q} - \mathcal{P}(q) \qquad (3.2.6)$$

Now, using the Euler-Lagrange equation,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = \tau \qquad (3.2.7)$$

the dynamic equation of the manipulator can be expressed [2] by the equation

$$D_l(q)\ddot{q} + \{ D_l(q)\dot{q} - \frac{1}{2}\dot{q}^T \left(\frac{\partial D_l(q)}{\partial q}\right)\dot{q}\} + \frac{\partial \mathcal{P}(q)}{\partial q} = \tau \qquad (3.2.8)$$

which verifies that the kinetic energy of a manipulator is given by equation (3.2.5).

As we shall see later in this thesis, the inertia matrix $D_l(q)$ plays an important role in manipulator controller design. Therefore, it is important to see some of its basic properties. As shown in [1], the elements of the inertia matrix are trigonometric functions (sines and cosines) of joint position variables (for revolute joints). Now, since sines and cosines are bounded for any value of their arguments, the inertia matrix $D_l(q)$ is bounded for all q. A consequence of this is that $D_l(q)$ is bounded from above and below. Since $D_l(q)$ is a positive-definite matrix, its inverse exists, and is also positive-definite. Also, it is bounded from above and below.

b) *The Coriolis and centrifugal term* $C_l(q, \dot{q})$

When a multi-body manipulator system moves, its internal nonlinear forces are expressed by a vector function of Coriolis and centrifugal terms. The Coriolis and centrifugal terms are known [2] to be complex nonlinear functions of the variables q and $\dot{q}$, and can be written in several different ways. Besides the form in equation (3.2.2), it is also possible to write $C_l(q, \dot{q})$ in the form

$$C_l(q,\dot{q}) = \begin{bmatrix} \dot{q}^T C_{l1}(q)\,\dot{q} \\ \dot{q}^T C_{l2}(q)\,\dot{q} \\ \vdots \\ \dot{q}^T C_{ln}(q)\,\dot{q} \end{bmatrix} \equiv C_{lm}(q,\dot{q})\,\dot{q} \qquad (3.2.9)$$

where $C_{li}(q) \in \Re^{n \times n}$ are symmetric matrices for $i = 1, 2, \ldots, n$. Furthermore, from the Lagrangian formulation of equation (3.2.8), it can be shown that

$$C_l(q,\dot{q}) = \dot{D}_l(q)\,\dot{q} - \frac{1}{2}\dot{q}^T\left(\frac{\partial D_l(q)}{\partial q}\right)\dot{q} \qquad (3.2.10)$$

from where it follows that [12]

$$C_{lm}(q,\dot{q}) = \frac{1}{2}\{\dot{D}_l(q) + S\} \qquad (3.2.11)$$

where S is some skew symmetric matrix.

c) *The gravity term* $G_l(q)$

The gravity term $G_l(q)$ defines the effect of gravity on the manipulator. Equating equations (3.2.1) and (3.2.8), we find that the gravity term $G_l(q)$ can be written as

$$G_l(q) = \frac{\partial \Psi(q)}{\partial q} \qquad (3.2.12)$$

Therefore, it is obvious that $G_l(q)$ is a function of the joint position q only This function dependence is in terms of *sine* and *cosine* functions only, and therefore term $G_l(q)$ is bounded.

d) *The frictional force term* $F_f(q)$

Frictional forces in the manipulator joints are described by functions which are usually complex and are probably represented only by some appr...mate deterministic models. Basically, there are two types of frictional effects which are usually considered. One is due to viscous friction where the torque required to overcome the friction is proportional to the joint velocity $\dot{q}$. The other is Coulomb friction which is defined by

$$F_{f,Coulomb} = c_o \mathrm{sgn}(\dot{q}) \tag{3.2.13}$$

where c_o is the Coulomb friction constant and its value changes depending on the value of the jc·nt velocity.

Usually, Coulomb friction dominates when the motion is slow, while viscous friction prevails when the motion of the joints is relatively fast. In the dynamic modeling described in Chapter 4, we consider the Coulomb friction force only because the manipulator velocity is usually kept low when contact between the end-effector and a workspace object is imminent.

3.3 RIGID-JOINT MANIPULATOR DYNAMICS IN CARTESIAN SPACE

Usually, for redundant manipulator control schemes, the desired input trajectory and the feedback trajectory information are Cartesian quantities from which the Cartesian tracking errors can be formulated. Thus, it is possible to directly construct a manipulator tracking controller in Cartesian space. Therefore, it is often necessary to develop a Cartesian space formulation of manipulator dynamics [8][10].

The Cartesian-space dynamic equation, which relates the acceleration of the end-effector expressed in Cartesian space to the Cartesian forces and moments acting at the end-effector, can be obtained using the joint-space dynamic model of equation (3.2.1), and the acceleration relation (2.2.6). In the following, the frictional forces are omitted (they depend only on the sign of joint variables), and we also assume free space motion, i.e.,

$F = 0$. Under these conditions, the Cartesian space manipulator dynamic model is defined by the equation

$$F_c = D_x(q)\ddot{x} + C_x(q, \dot{q}) + G_x(q) \tag{3.3.1}$$

where

$F_c \in \Re^{m \times 1}$: is the Cartesian force/moment vector acting on the end-effector,

$\ddot{x} \in \Re^{m \times 1}$: is the Cartesian position/orientation vector of the end-effector,

$D_x(q) \in \Re^{m \times m}$: is the manipulator inertia matrix expressed in Cartesian space,

$C_x(q, \dot{q}) \in \Re^{m \times 1}$: is the nonlinear velocity vector expressed in Cartesian space,

$G_x(q) \in \Re^{m \times 1}$: is the gravitational force vector expressed in Cartesian space

In the case of a non-redundant manipulator, since the Jacobian matrix $J_e(q)$ is square and its inverse exists when the manipulator's configuration is not in the neighborhood of a singularity, the above quantities can be expressed in terms of their joint-space counterparts as follows:

$$D_x(q) = J_e^{-T}(q) D_j(q) J_e^{-1}(q)$$

$$C_x(q, \dot{q}) = J_e^{-T}(q)\,[C_j(q, q) - D_j(q) J_e^{-1}(q) \dot{J}_e(q)]\, J_e^{-1}(q)$$

$$G_x(q) = J_e^{-T}(q) G_j(q) \tag{3.3.2}$$

However, in the case of a redundant manipulator, the expressions in equation (3.3.2) can be modified such that direct inversion of the Jacobian matrix is avoided [10]

$$D_x(q) = [J_e(q) D_j^{-1}(q) J_e^T(q)]^{-1}$$

$$C_x(q, \dot{q}) = D_x(q)\,[J_e(q) D_j^{-1}(q) C_j(q, q) - \dot{J}_e(q) q]$$

$$G_x(q) = D_x(q) J_e(q) D_j^{-1}(q) G_j(q) \tag{3.3.3}$$

The end-effector Cartesian force F_e can be mapped into the equivalent joint torque τ through the Jacobian matrix J_e, i.e., $\tau = J_e^T F_e$. Thus, the dynamic model that relates the required joint torque τ to the end-effector Cartesian coordinates x is given by the equation

$$\tau = J_e^T D_x(q)\ddot{x} + J_e^T C_x(q, \dot{q}) + J_e^T G_x(q) \tag{3.3.4}$$

The manipulator Cartesian-space dynamic model (3.3.1) or (3.3.4) is useful and convenient for the construction of Cartesian-space controllers. However, from the computational point of view, this model is expensive. Moreover, the complete Cartesian-space scheme requires Cartesian-space sensing to obtain precise end-effector position and velocity information. Now, since Cartesian-space sensing is much more difficult to do than the joint-space sensing, for our Cartesian-space controller, we shall assume that the sensing takes place in joint-space and the measured joint outputs are converted to the corresponding Cartesian quantities which are fed back to produce the system error in Cartesian space. More details of the design of these controllers will be presented in later chapters in this thesis.

3.4 GENERAL FLEXIBLE-JOINT MANIPULATOR DYNAMICS

Experimental results have shown that for high precision operations of robot manipulators, the joint flexibility should be taken into account in the dynamic modeling of manipulators [5]. The flexibility in the joints may be caused by the presence of *harmonic drives*. These are gear mechanisms having low weight, small size and high gear ratios to accommodate motors spinning 10 to 100 times faster than the links. Joint flexibility may also be caused by deformation of the gears at a joint due to loading.

In this section, we first introduce the general dynamic model of a flexible-joint manipulator and then, in the next section we make several simplifications to this model.

A general formulation of the dynamic model of flexible-joint manipulators has been proposed in [3][4][7][14] using the Lagrangian or the Newton-Euler approach. In this case, the inertia of the actuators is modeled about three independent axes. Hence, the inertia forces, Coriolis and centrifugal forces, and gravitational forces depend not only on the link variables, but also on the motor velocities and accelerations. In particular, the formulation of the dynamic equations of motion of a general flexible joint manipulator can be written [14] as

$$\begin{bmatrix} D_l(q_l) & D_o(q_l) \\ D_o^T(q_l) & D_m \end{bmatrix} \begin{bmatrix} \ddot{q}_l \\ \ddot{q}_m \end{bmatrix} + \begin{bmatrix} C_{fl}(q_l, \dot{q}_l, \dot{q}_m) \\ C_m(q_l, \dot{q}_l, \dot{q}_m) \end{bmatrix} + \begin{bmatrix} G_{fl}(q_l) \\ G_m(q_l) \end{bmatrix} + \begin{bmatrix} -T_o \\ T_o \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} \qquad (3.4.1)$$

where $q_l \in \Re^{n \times 1}$ denotes the link position vector, and $q_m \in \Re^{n \times 1}$ represents the motor position vector; $D_l(q_l) \in \Re^{n \times n}$ is the link inertia matrix, while $D_m \in \Re^{n \times n}$ denotes the motor inertia matrix; $D_o(q_l) \in \Re^{n \times n}$ represents the coupling, or interaction matrix that gives the dynamic coupling between the motor acceleration and the link acceleration. In a standard model of flexible-joint manipulator, the structure of $D_o(q_l)$ is usually an upper triangular matrix with zero diagonal elements, $C_{fl}(q_l, q_l, q_m) \in \Re^{n \times 1}$ and $G_{fl}(q_l) \in \Re^{n \times 1}$ are the effects on the link shaft due to Coriolis, centrifugal, and gravity forces. Similarly, $C_m(q_l, \dot{q}_l, \dot{q}_m) \in \Re^{n \times 1}$ and $G_m(q_l) \in \Re^{n \times 1}$ represent the effects on the motor due to Coriolis, centrifugal, and gravity forces acting upon the motor It should be noted that in the above model, because the motors are modeled as uniform cylinders, the inertia matrix is a function only of the link variables q_l. This also implies that if we express the Coriolis and centrifugal vectors C_{fl} and C_m as $C_{fl}(q_l, q_l) = C_{fl_v}(q_l, q_l)q$ and $C_m(q_l, \dot{q}_l) = C_{m_v}(q_l, \dot{q}_l)\dot{q}$, the matrices $C_{fl_v}(q_l, q_l)$ and $C_{m_v}(q_l, \dot{q}_l)$ will be independent of the motor variables [15], where $q = \begin{bmatrix} q_l^T & q_m^T \end{bmatrix}^T \in \Re^{2n \times 1}$. The same is also true for the gravity forces acting on the system. However, we note that the motor velocity does enter into the Coriolis and centrifugal terms since $C_{fl_v}(q_l, \dot{q}_l)$ and $C_{m_v}(q_l, \dot{q}_l)$ are multiplied by $\dot{q}$.

The vector $T_o \in \Re^{n \times 1}$ in equation (3.4.1) denotes the coupling function between the motors and the links. In general, this function is a nonlinear function due to backlash and motor saturation [11]. The typical characteristics of the coupling function between a motor and a link is illustrated in Figure 3.1. In the general model equation (3.4.1), the complete coupling is considered which includes, first, the coupling torque between the motors and the links (whose actual characteristics are shown in Figure 3.1), and second, the inertia coupling represented by the off-diagonal terms of the inertia matrix and the associated Coriolis forces.

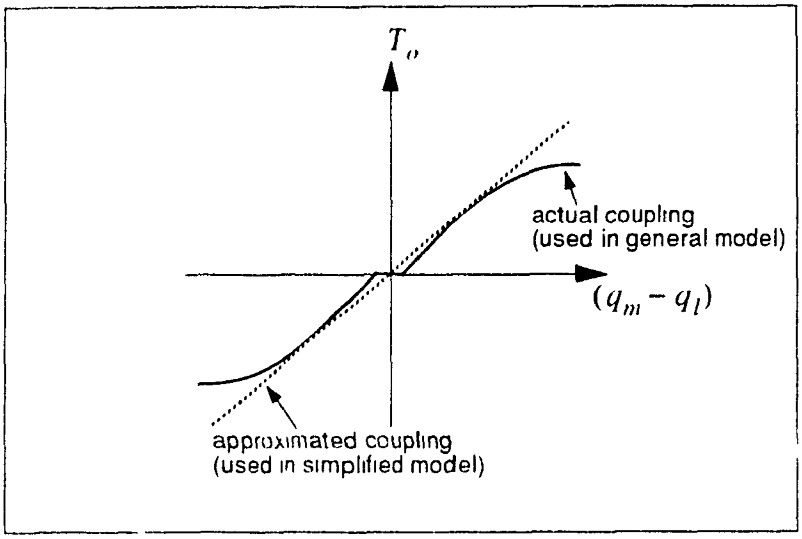

Figure 3.1 **The characteristics of the coupling function in flexible joint**

Obviously, in developing control strategies for flexible-joint manipulators, it is better to use the general model of equation (3.4.1). However, since this model is highly nonlinear and coupled, it is very difficult to design adequate and efficient controllers for this complicated system. To avoid this difficulty, in the next section we introduce an approximate model which leads us to the "simplified flexible-joint manipulator" dynamic model.

3.5 SIMPLIFIED FLEXIBLE-JOINT MANIPULATOR DYNAMICS

A simplified dynamic model for flexible-joint manipulators derived using the

Lagrange-Euler technique has been proposed by Spong [13] and is based on the following assumptions:

(a) The kinetic energy of the motor is due to its own rotation.

(b) The motor is symmetric about the axis of rotation.

Assumption (a) implies that the motors are rotated with respect to the inertial frame, not with respect to the frames attached to moving links. Under this assumption, the off-diagonal submatrix $D_o(q_l)$ in the inertia matrix of the flexible-joint manipulator vanishes. Furthermore, the Coriolis and centrifugal forces which are the gyroscopic forces in this case can be ignored and this leads to $C_m = 0$. Assumption (b) affects the motor dynamic equation directly. With a symmetric motor, it can be shown that $G_m = 0$ [7]. In addition, Spong [13] used a simple torsional spring to model the coupling between the motor and the link, and the model of torsional spring can be expressed as

$$T_o = K_s(q_m - q_l) \tag{3.5.1}$$

where K_s is a diagonal matrix of spring constants. This approximation of coupling is shown graphically in Figure 3.1 by the dotted line, where the slope of the line is determined by K_s.

Under these assumptions, the general dynamic model of equation (3.4.1) is reduced to the following model [13]

$$\begin{bmatrix} D_l(q_l) & 0 \\ 0 & D_m \end{bmatrix} \begin{bmatrix} \ddot{q}_l \\ \ddot{q}_m \end{bmatrix} + \begin{bmatrix} C_l(q_l, \dot{q}_l) \\ 0 \end{bmatrix} + \begin{bmatrix} G_l(q_l) \\ 0 \end{bmatrix} + \begin{bmatrix} -K_s(q_m - q_l) \\ K_s(q_m - q_l) \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} \tag{3.5.2}$$

where C_l is the reduced Coriolis and centrifugal term and G_l denotes the reduced gravity vector. For the simplified model of equation (3.5.2), we can make the following two remarks.

Remark 3.1: It is important to note that the simplified and the general flexible-joint manipulator dynamic models possess different properties with respect to the feedback controller design. In the simplified model of equation (3.5.2) the inertial coupling is ignored, and the torque is only transmitted through the coupling between the motor and the link. On the other hand, in the general model of equation (3.4.1) there are two sources of interaction, namely, the coupling between the motor and the link, and the inertia coupling. The different structure of the off-diagonal matrix $D_o(q_l)$ in the inertia matrix of the general flexible-joint manipulator leads to different considerations for controller design.

Case 1: $D_o(q_l) = 0$. This case corresponds to the simplified model of equation (3.5.2). In this case, a static feedback control law can be derived [13] based only on the knowledge of q_l, q_m, $\dot{q}_l$ and $\dot{q}_m$.

Case 2: $D_o(q_l)$ is a non-singular matrix. This case corresponds to the general model of equation (3.4.1) where $D_o(q_l)$ is a non-singular upper triangular matrix representing the coupling of the acceleration between the motors and the links. As shown in [7], the motor acceleration $\ddot{q}_m$ in the link dynamic equation can be eliminated using the motor dynamic equation. Then the computed torque technique can be applied to formulate the controller.

Case 3: $D_o(q_l) \neq 0$ but singular. As mentioned above, this is the typical case for the general model (3.4.1) where $D_o(q_l)$ is upper triangular with zero diagonal elements. For this case, there is no static feedback linearizing controller suitable for this model. Therefore, other approaches must be considered [3].

Remark 3.2: As we can see in Figure 3.1 when the dead-zone of the motor is relatively small, and the motor operates near the linear portion of the curve, the approximation of the coupling between the motor and the link is fairly good. In practice, this approximation is usually valid.

3.6 CONCLUSIONS

In this chapter, we have described dynamic models for rigid- as well as flexible-joint manipulators. This is an important step in the design of controllers for such manipulators. We first considered rigid-joint manipulators and their dynamic models expressed in joint- as well as in Cartesian-space. The Cartesian space model is useful for designing Cartesian control schemes. This is particularly relevant in the case of designing controllers for redundant manipulators. However, Cartesian-space dynamic models are computationally more expensive than their joint-space counterparts. Dynamic modeling of flexible-joint manipulators was considered and a general model was given. Possible simplifications to this model were described. Finally, issues concerning controller design for these two models were briefly discussed.

3.7 REFERENCES

[1] J.J. Craig, *Introduction to Robotics, Mechanics and Control*, Addison Wesley, 1986.

[2] J.J. Craig, *Adaptive Control of Mechanical Manipulators*, Addison Wesley, 1988.

[3] A. De Luca, "Dynamic control properties of robot arms with joint elasticity," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Philadelphia, PA., pp. 1574-1580, 1988.

[4] A. De Luca, A. Isidori and F. Nicolo, "Control of robot arm with elastic joints via nonlinear dynamic feedback," in *Proc. 24th IEEE Conf. on Decision and Control*, Ft. Lauderdale, FL., pp. 1671-1679, 1985.

[5] M.C. Good, L.M. Sweet and K.L. Strobel, "Dynamic models for control system design of integrated robot and drive systems," *ASME J. Dyn. Syst., Meas. Contr.*, vol. 107, pp. 53-59, 1985.

[6] J.Y. Luh, M.W. Walker and R.P. Paul, "On line computational scheme for mechanical manipulators," *ASME J. Dyn. Syst. Meas. Contr.*, vol. 102, 1980.

[7] S.H. Murphy, J.T. Wen and G.N. Saridis, "Simulation and analysis of flexibly jointed manipulators," in *Proc. 29th IEEE Conf. on Decision and Control*, Honolulu, HA., pp. 545-550, 1990.

[8] C.C. Nguyen, Z. Zhou and G. E. Mosier, "Cartesian-space control of redundant manipulators using a computationally efficient adaptive control scheme," *Robotics & Computer-Integrated Manufacturing*, vol. 9, pp, 159-167, 1992.

[9] R.P. Paul, *Robot Manipulators: Mathematics, Programming and Control*, MIT Press, Cambridge, MA., 1981.

[10] H. Seraji, "Task-based configuration control of redundant manipulators," *J. Robotic Systems*, vol. 9, pp. 411-451, 1992.

[11] P. Sicard and J.T. Wen, "Application of a passivity based control methodology for flexible joint robots to a simplified Space Shuttle RMS," in *Proc. 1992 American Control Conf.*, Chicago, IL., pp. 1690-1694, 1992.

[12] J.E. Slotine and W. Li, "Adaptive manipulator control: A case study," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 995-1003, Nov. 1988.

[13] M.W. Spong, "Modeling and control of elastic joint robots," *J. Dyn. Syst. Meas. Contr.*, vol. 109, pp. 310-319, 1987.

[14] M.W. Spong, "Control of flexible joint robots: A survey," *Coordinated Science Laboratory, Technical Report*, UILU-ENG-90-2203 DC-116, University of Illinois at Urbana-Champaign, Feb. 1990.

[15] M.W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley & Sons, New York, 1989.

[16] M.W. Walker and D.E. Orin, "Efficient dynamic computer simulation of robotic mechanisms," *ASME J. Dyn. Syst. Meas. Contr.*, vol. 104, 1982.

CHAPTER 4 IMPEDANCE CONTROL OF REDUNDANT MANIPULATORS FOR MINIMIZATION OF COLLISION IMPACT

4.1 INTRODUCTION

In the last few years, most of the research on kinematic redundancy resolution has been focused on utilizing redundancy for achieving secondary tasks which do not require interaction with the environment, such as obstacle avoidance, singularity avoidance, optimizing some kinematic objective functions, etc. [1][5][9][11][12]. However, kinematic redundancy resolution can also be used to successfully solve problems where interaction with the environment is necessary. For example, kinematic redundancy resolution can be used in minimizing collision impact in contact tasks by choosing appropriate configurations of the redundant manipulator and appropriate control strategies. In this chapter, the problem of collision impact control for redundant manipulators is addressed and a new control strategy is proposed.

In many practical applications such as moving objects, assembling parts, cleaning surfaces, deburring edges, etc., the manipulator must make certain types of contact with its workspace objects or environment. In applications of this nature the manipulator has to come into physical contact with the object before the desired force and moment can be applied. Thus, in switching from free space motion to constrained force control, one has to analyze the significant problem of impact forces. These impulsive forces can be very large, and in many cases, they can cause the manipulator rebound from the environment to drive an otherwise stable controller into instability. Also, large impulsive forces can cause damages to the objects as well as the manipulator itself. Therefore, for safety reasons and for successful operation, it is desirable to reduce impulsive forces and soften the contact between a manipulator's end-effector and the workspace objects.

In this chapter, an impedance control based strategy for collision impact is proposed [7]. The controller consists of a *simplified impedance controller* and an *augmented configuration controller*. The simplified impedance controller is based on selection of the desired inertia matrix such that its inverse is equal to the mobility tensor of the manipulator defined in Cartesian space. Its purpose is to reduce both impulsive forces and rebound efforts. As we shall see, this judicious choice of the desired inertia matrix reduces the impulsive forces, and also avoids the oscillatory behavior. The augmented configuration controller is designed to choose the proper configuration of the robot arm such that the impulsive forces are minimized. To achieve this goal, the manipulator impulsive contact model is derived in Cartesian space, and the relation of the impulsive forces with respect to the manipulator's configuration is used for the augmentation of the manipulator's Jacobian matrix. Thus, in this scheme, the self-motion of the manipulator is used to configure its posture such that the impulsive forces are minimized at the time of impact. To demonstrate the effectiveness of the proposed controller, numerical simulations have been carried out using a three-link planar redundant manipulator. The results obtained for minimization of impulsive forces, and reduction of rebound efforts confirm the validity of the proposed scheme.

This chapter is organized as follows: Section 4.2 gives a review of the existing literature concerning rigid body impact modeling, non-redundant manipulator impact analysis, and the use of redundancy for impact reduction. An impedance control strategy as well as a simplified impedance control law are introduced in Section 4.3. Section 4.4 presents the redundancy resolution strategy called the augmented kinematics approach, while in Section 4.5 modeling of a manipulator's impulsive contact behavior in Cartesian space is given. Based on concepts introduced in the previous sections, in Section 4.6, the augmented kinematics approach and the simplified impedance control strategy are combined, and it is shown how the manipulator impact effects are altered by the proposed controller. Simulation results are shown in Section 4.7, and finally, Section 4.8 concludes the chapter.

4.2 OVERVIEW OF THE EXISTING METHODOLOGIES

A brief outline of previous work on the analysis of manipulator collision effects and the use of redundancy in contact tasks is given in this section. In [10], stability issue of manipulators during transition to and from compliant motion is addressed. Two stability results for manipulators switching to and from compliant motion are established. One is the global asymptotic connective stability, and other is the joint asymptotic connective stability. In [17], an impact model for a single-axis drive system is investigated by Youcef-Toumi and Gutz based on an energy method. Their investigation shows that integral force compensation with velocity feedback improves force tracking and rejects impact. It is also shown in their study that impact response can be tuned by selecting a favorable dimensionless ratio of force to approaching velocity. In [15] Wang and Mason present a geometric method for modeling impact in the planar case. The basic feature of this method predicts the mode of contact, the total impulsive force and the resultant motions of the objects. Their method also includes friction, inertial effects and elasticity for two objects in collision. Another model for impact which allows the incorporation of complete spatial manipulator dynamics was introduced by Zheng and Hemami in [18]. Their approach reveals two important points. First, the manipulator's joint velocities have abrupt changes at the moment of collision with the environment. These changes are defined with a mathematical model which they derived for establishing a quantitative relationship between the abrupt change and the severity of the collision. Second, internal to the manipulator, large impulsive forces and torques may develop at each joint because of the collision. This relationship is also expressed mathematically to establish a quantitative relation between the impulsive forces/torques and the collision. In [13] Volpe and Khosla looked at the impact problem from an experimental point of view. They applied various force control strategies for impact control and compared the results '' ever, this experimental analysis has been limited to on-redundant manipulators. The idea of using a redundant manipulator for reducing impact and contact effects was first introduced by Walker in [14]. Based on the model proposed by Zheng and Hemami [18], Walker first derived the impact dynamics by expressing the impact force in terms of the mode of collision, the relative velocity between the manipulator end-effector and the environment, the normal direction of the collision plane, and the configuration of the manipulator. This model reveals that the impact force can vary in terms of different configurations of the manipulator while assuming all other parameters fixed. This idea was further analyzed by Gertz, Kim and Khosla in [2] where they modelled the impact events as non-instantaneous instead of the conventional impact modeling of instantaneous effects. Two strategies were proposed in [2]: One involves adding torques to the joints of the redundant manipulator to impede motion into the object with which it collides; the other which is similar in some respects to Walker's approach involves choosing the best configuration for the impact event.

As mentioned above, there are basically two ways to model the collision process. One is to model collision impact as an instantaneous process with infinitesimal time duration [14][18]; the other is to treat the collision impact as a non-instantaneous event with certain finite time duration [2]. Both of these models are valid for analyzing the manipulator collision impact In the collision impact analysis performed in this chapter, we shall mainly concentrate on the former approach, i.e., treating impact as a instantaneous process.

4.3 IMPEDANCE CONTROL STRATEGY

4.3.1 Impedance Control for Compliant Motion

Impedance control is a compliant motion control strategy that involves the regulation of the mechanical impedance of the manipulator's end-effector. The objective of an impedance controller is to maintain a desired dynamic relationship between the end-effector/environment contact force and the manipulator's position. In general, a Cartesian target impedance is usually specified as a second-order linear system [3][4]:

$$M_1(\ddot{x} - \ddot{x}_d) + B_1(\dot{x} - \dot{x}_d) + K_1(x - x_d) = -F \qquad (4.3.1)$$

where M_1, B_1 and $K_1 \in \Re^{m \times m}$ are positive-definite matrices representing the desired mass, damping and stiffness of the closed-loop system respectively; x_d and $x \in \Re^{m \times 1}$ represent the desired and actual end-effector position, and $F \in \Re^{m \times 1}$ denotes the Cartesian force exerted by the end-effector on the environment. The matrices M_1, B_1 and K_1 can be selected by the designer to correspond to various manipulation task objectives. For example, high stiffness is specified along directions where the environment is compliant and positioning accuracy is important. On the other hand, low stiffness is specified in directions where the environment is stiff, or when small interaction forces must be maintained. Similarly, a large value of damping matrix B_1 is specified when energy must be dissipated, while M_1 can be used to provide smoothing in the end-effector response due to external contact. The coefficient matrices, M_1, B_1 and K_1 of equation (4.3.1) need not be diagonal. For some applications, the coupling between the impedance axes due to nondiagonal forms of the matrices M_1, B_1 and K_1, may be useful, while for other tasks M_1, B_1 and K_1 can be considered to be diagonal, i.e., the case of uncoupled impedances.

To design an impedance controller for which the desired closed-loop characteristics (desired impedance) of equation (4.3.1) are achieved, we proceed as follows. The Cartesian space acceleration $\ddot{x}$ is related to the joint space acceleration by the equation

$$\ddot{x} = J_e\ddot{q} + \dot{J}_e\dot{q} \qquad (4.3.2)$$

Now, since the desired mass matrix M_1 has been assumed to be positive definite, we can solve equation (4.3.1) for the Cartesian acceleration $\ddot{x}$ to get

$$\ddot{x} = M_1^{-1} [M_1 \ddot{x}_d - B_1 (\dot{x} - \dot{x}_d) - K_1 (x - x_d) - F] \tag{4.3.3}$$

Also, from the joint-space based manipulator dynamic model

$$D_l(q)\ddot{q} + C_l(q, \dot{q}) + G_l(q) + F_f(\dot{q}) = \tau - J_e^T(q) F \tag{4.3.4}$$

we solve for the joint space acceleration vector $\ddot{q}$. Note that this is always possible since the inertia matrix $D_l(q)$ is positive-definite. Then the manipulator joint acceleration can be expressed as

$$\ddot{q} = D_l^{-1}(q) [\tau - J_e^T(q) F - C_l(q, \dot{q}) - G_l(q) - F_f(\dot{q})] \tag{4.3.5}$$

Now by substituting equations (4.3.3) and (4.3.5) into equation (4.3.2), the impedance controller is determined by the equation

$$\begin{aligned} \tau = J_e^T \{ W_1^{-1} [M_1^{-1} (B_1 (\dot{x}_d - J_e \dot{q}) + K_1 (x_d - \Lambda(q))) + \ddot{x}_d \\ + J_e D_l^{-1} (C_l + G_l + F_f) - \dot{J}_e \dot{q}] + (I - W_1^{-1} M_1^{-1}) F \} \end{aligned} \tag{4.3.6}$$

where $\Lambda(q)$ defines the forward kinematics function which maps joint displacements into Cartesian space, and $W_1 = J_e D_l^{-1} J_e^T$ is usually referred to as the *mobility tensor*, and its inverse (which exists at nonsingular configurations) is called the *virtual mass* [3] of the manipulator in Cartesian space. Note that the controller (4.3.6) is also applicable to redundant manipulators. As we can see in this case, the redundancy resolution problem is solved implicitly by using the *mobility tensor* $W_1 = J_e D_l^{-1} J_e^T$, i.e., the redundancy is resolved following the weighted pseudo-inverse approach with the manipulator inertia matrix being the weighting matrix. Also note that by comparing the impedance controller of equation (4.3.6) with the resolved acceleration control strategy, we can see that impedance control is actually a *dual* of resolved acceleration control [8] in the domain of constrained motion. The impedance controller of equation (4.3.6) forms the basis for our simplified impedance controller which will be presented in the following section.

4.3.2 Simplified Impedance Control

Equation (4.3.6) defines a relatively complicated nonlinear controller which includes a position feedback loop as well as a force feedback loop. Moreover, this impedance controller is designed primarily for manipulator compliant motion and free space trajectory tracking rather than collision impact control [3][4]. Therefore, the impedance controller shown above must be modified, and the gain matrices have to be adjusted such that it is specifically suitable for collision impact control.

In the following, we propose a new controller called the *simplified impedance controller*. The basic idea in deriving this simplified impedance controller is to select the desired inertia (mass) matrix of equation (4.3.6) as

$$M_1^{-1} = W_1 \tag{4.3.7}$$

Equation (4.3.7) states that the inverse of the desired mass matrix is chosen to be identical to the *mobility tensor* (or the natural inertia tensor) as expressed in Cartesian space. With this choice for matrix M_1, the manipulator is commanded to behave exactly as it would actually behave in Cartesian space. In other word, it is the exact physical consequence of accepting the machine's natural inertia. In this case, it can be seen that since $W_1^{-1}M_1^{-1} = I$, the terms corresponding to the force F vanish in equation (4.3.6). This does not imply that the force feedback loop is deleted; rather, because of the specific choice of the desired mass matrix M_1, the force feedback signal going through the feedback loop has been cancelled. With this judicious choice of M_1, the impedance control law of equation (4.3.6) becomes the *simplified impedance controller*

$$\tau = J_e^T \{ B_1(\dot{x}_d - J_e\dot{q}) + K_1(x_d - \Lambda(q)) + W_1^{-1}[J_e D_l^{-1}(C_l + G_l + F_f) + \ddot{x}_d - \dot{J}_e\dot{q}] \} \tag{4.3.8}$$

which obviously results in coupled Cartesian motion due to the nondiagonal structure of M_1. In the following, we shall demonstrate that the simplified impedance controller is indeed suitable for manipulator collision impact control.

Let us choose the following mass matrix

$$M_1 = \rho W_1^{-1} \tag{4.3.9}$$

where the scalar $\rho > 0$. With this choice of M_1, let us see what the best value is for the mass ratio ρ which results in the minimization of the manipulator's collision impact force. Now, substituting equation (4.3.9) into equation (4.3.6), we can express the control force (obtained from the control torque using the relationship $\tau = J_e^T F$) as a function of the mass ratio ρ as follows

$$F = W_1^{-1}(\frac{1}{\rho}W_1(B_1(\dot{x}_d - \dot{x}) + K_1(x_d - x))) + (I - \frac{1}{\rho}W_1^{-1}W_1)F_{imp} + V_n \tag{4.3.10}$$

where $V_n = W_1^{-1}(J_e D_l^{-1}(C_l + G_l + F_f) + \ddot{x}_d - \dot{J}_e\dot{q})$ denotes the nonlinear forces. Equation (4.3.10) can be simplified and the relationship between the mass ratio ρ and the end-effector force F can be expressed as

$$F = \frac{1}{\rho}(\rho - 1)F_{imp} + \frac{1}{\rho}(B_1(\dot{x}_d - \dot{x}) + K_1(x_d - x)) + V_n \qquad (4.3.11)$$

Equation (4.3.11) provides the functional relation of the end-effector force at the time of impact with respect to the coefficient (mass ratio) ρ. By choosing a different desired mass matrix M_1 by changing ρ, we can clearly see that the end-effector force F at the time of impact will be affected. Note that at the time of impact, the impact force F_{imp} is much larger than other forces expressed on the right-hand side of equation (4.3.11). Also, we note that the term $\frac{1}{\rho}(\rho - 1)$ in (4.3.11) plays an important role in determining the magnitude of the end-effector force at the time of impact. In order to clearly see this effect, we graphically show the characteristics of the term $\frac{1}{\rho}(\rho - 1)$ with respect to the mass ratio ρ as follows:

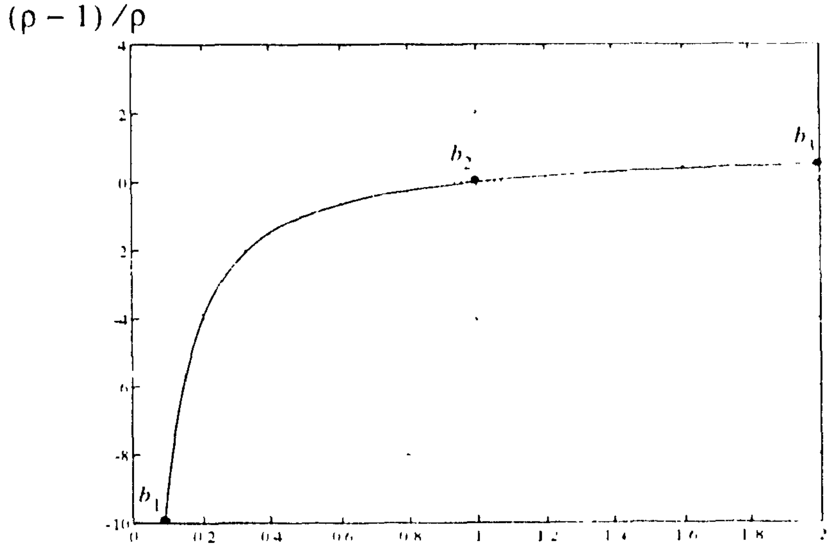

Figure 4.1 **Functional relationship of the term** $(\rho - 1)/\rho$ **with respect to mass ratio** ρ

From Figure 4.1 and equation (4.3.11), we have three possible choices for ρ, namely, (i) $0 < \rho < 1$, (ii) $\rho = 1$, and (iii) $\rho > 1$. For these three choices of ρ, we have the following analysis.

(i) $0 < \rho < 1$: This implies that $(\rho - 1)/\rho < 0$ and the values of the term $\frac{1}{\rho}(\rho - 1)$ correspond to the points on the curve between b_1 and b_2 in Figure 4.1. Based on equation (4.3.9), the implication of this choice of ρ is that we select smaller values for the desired mass matrix. With this choice of ρ, the magnitude of the first term on the right-hand side of equation (4.3.11) becomes more negative when ρ deviates from 1 towards zero. If we take into account the rest of the terms on the right-hand side of equation (4.3.11) and, also consider the closed-loop characteristics of equation (4.3.1), then we see that the deviation of ρ from 1 results in an increase in the imaginary part of the closed-loop system poles, and consequently in the oscillatory behavior of the manipulator, rather than in an increase in the magnitude of the end-effector force at the time of impact. This phenomenon was also observed by Volpe and Khosla [13] through impact experiments. Although in this case the end-effector force at the time of impact is smaller, the resulting oscillatory behavior is not desirable. Moreover, the oscillatory behavior may affect stability of the system, thus, it is not advisable to choose ρ close to zero.

(ii) $\rho > 1$: In this case, the values of the term $\frac{1}{\rho}(\rho - 1)$ correspond to points on the curve between b_2 and b_3 in Figure 4.1. This choice of ρ implies that the desired inertia matrix is larger than the manipulator's true inertia. This selection has two consequences: From the point of view of reducing rebound effects, a larger value of M_1 is desirable. In this case, as pointed out in [13] the manipulator is made to appear so massive that it cannot bounce back. However, from the point of view of impulsive force, a larger value of inertia results in larger impulsive forces. This follows from the first term on the right-hand side of equation (4.3.11). This fact has also been verified experimentally in [13].

(iii) $\rho = 1$: This case corresponds to the point b_2 in Figure 4.1. With this choice of ρ, the desired inertia matrix M_1 is selected to be equal to the manipulator actual inertia in Cartesian space. As can be seen from equation (4.3.11), when $\rho = 1$, the first term vanishes and therefore, the end-effector force at the time of impact is reduced.

From this analysis, it is clear why we select $\rho = 1$ in the reduced-order impedance controller which is intended for use in impact controller. Although in general we cannot say that $\rho = 1$ is the optimal choice, it is obvious that with this choice we are close to the optimum. Also, as we shall see in later sections, this choice of ρ provides a good compromise between reduction of rebound effects and minimization of the magnitude of the impulsive forces.

4.4 AUGMENTED SIMPLIFIED IMPEDANCE CONTROL

In its present formulation, the simplified impedance controller is applicable to non-redundant manipulators. For kinematically redundant manipulators, sol. • modifications must be made in order to resolve the redundancy in a useful manner. In this section, we shall show how we can utilize the redundancy to minimize collision impacts. The redundancy utilization will be based on augmented kinematics and the configuration control approach [11][12] which were discussed briefly in Section 2.3.1.

As mentioned in Section 2.3.1, the optimization of a kinematic or dynamic objective function $L(q, D)$ which will resolve the redundancy is reformulated as a tracking problem. In this problem, the forward kinematic function and its Jacobian matrix have the form $y = \begin{pmatrix} \Lambda(q) \\ z(q, D) \end{pmatrix}$, and $J = \begin{pmatrix} J_e \\ \partial z / \partial q \end{pmatrix}$ respectively, where the augmentation function z is usually defined as follows [11]:

$$z(q, D_l) \equiv N_e \frac{\partial L(q, D_l)}{\partial q} = 0 \qquad (4.4.1)$$

In (4.4.1) the rows of the matrix $N_e \in \Re^{r \times n}$ span the null space of the manipulator's Jacobian matrix J_e. Note that the function z, as defined by equation (4.4.1), optimizes the objective function $L(q, D_l)$ subject to the forward kinematic constraints $x = \Lambda(q)$. Based on this augmentation of the forward kinematics and the augmented Jacobian matrix, the corresponding *augmented simplified impedance control* law is written as

$$\tau = J^T \{ B(\dot{y}_d - J\dot{q}) + K(y_d - y) + W^{-1} [J D_l^{-1} (C_l + G_l + F_f) + \ddot{y}_d - J q] \} \qquad (4.4.2)$$

where $K \in \Re^{n \times n}$ and $B \in \Re^{n \times n}$ represent the augmented stiffness and damping matrices which are chosen to be symmetric and positive-definite, and have the following structure

$$K = \begin{bmatrix} K_1 & 0 \\ 0 & K_z \end{bmatrix} \qquad\qquad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \tag{4.4.1}$$

The augmented stiffness matrix K is composed of the stiffness matrix K_1 and the gain matrix K_z from the augmentation feedback loop, while the augmented damping matrix B is computed based on the augmented stiffness matrix K and an augmented desired inertia matrix M. Details of how the damping matrix B can be computed will be presented later in this chapter. The matrix $W \in \Re^{n \times n}$ in equation (4.4.2), the augmented mobility tensor, is defined as $W = JD_l^{-1}J^T$. Then, the augmented desired inertia matrix M is formed as $M = W^{-1}$.

4.5 MANIPULATOR IMPULSIVE CONTACT MODELING

When a manipulator comes into physical contact with its environment during task execution, the manipulator undergoes an impact with the environment for a very short period of time. This impact creates an impulsive force at the end-effector that is propagated through the manipulator structure. In this section, the dynamical problem of collision impact is addressed, and an impact dynamic model of the manipulator [14] with its environment is analyzed.

For the derivation of the impact force, it is more convenient to express the manipulator dynamics in Cartesian space. Therefore, we rewrite the manipulator Cartesian space dynamic model of equation (3.3.1) with impact force (frictional forces are also included) as follows:

$$F_{imp} + F = D_x(q)\ddot{x} + C_x(q,\dot{q}) + G_x(q) + F_{fx}(q,\dot{q}) \tag{4.5.1}$$

where $F_{imp} \in \Re^{m \times 1}$, $F \in \Re^{m \times 1}$ and $\ddot{x} \in \Re^{m \times 1}$ denote the impulsive force, the Cartesian control force, and the end-effector acceleration; and $D_x \in \Re^{m \times m}$ is the manipulator Cartesian inertia matrix (or Cartesian *virtual mass*) defined as $D_x = [J_e D_l^{-1} J_e^T]^{-1}$. Finally, the quantities $C_x(q,\dot{q})$, $G_x(q)$ and $F_{fx}(q,\dot{q})$ represent the Cartesian space equivalents of the Coriolis and centrifugal forces, the gravitational forces and the frictional forces. Now, assume that the initial manipulator/environment impact occurs at time t and lasts for an infinitesimally short period of time Δt. Then, by integrating both sides of equation (4.5.1) from t to $t + \Delta t$, we get

$$\int_t^{t+\Delta t} F_{imp}dt' + \int_t^{t+\Delta t} F dt' = \int_t^{t+\Delta t} D_x\ddot{x}dt' + \int_t^{t+\Delta t} (C_x + G_x + F_{fx})dt' \tag{4.5.2}$$

Since F, q and $\dot{q}$ are finite quantities at all times, the integrals of the finite functions F and $(C_x + G_x + F_{fx})$ from t to $t + \Delta t$ becomes zero as $\Delta t \to 0$. Thus, equation (4.5.2) reduces to

$$\vec{F}_{imp} = \lim_{\Delta t \to 0} D_x \int_t^{t+\Delta t} \ddot{x}dt = \lim_{\Delta t \to 0} D_x\{\dot{x}(t+\Delta t) - \dot{x}(t)\} \tag{4.5.3}$$

where

$$\vec{F}_{imp} = \lim_{\Delta t \to 0} \int_t^{t+\Delta t} F_{imp}dt \tag{4.5.4}$$

defines the impulsive force at time of impact, and the term $\{\dot{x}(t+\Delta t) - \dot{x}(t)\} \equiv \Delta\dot{x}$ represents the change in the end-effector's velocity before and after the impact. Note that in equation (4.5.4), the impulsive force $\vec{F}_{imp}$ results from the integration of the infinite value F_{imp} during a very short time period Δt. Equation (4.5.3) can be viewed [16] as a generalization of the point mass impact dynamics and therefore, equation (4.5.3) implies that the manipulator impact force is equal to the change in the manipulator's momentum before and after impact. Now, if we use the definition of Cartesian virtual mass D_x, equation (4.5.3) can further be expressed as

$$\Delta \dot{x} = [J_e D_l^{-1} J_e^T] \vec{F}_{imp} \tag{4.5.5}$$

In order to provide a simple analytic definition for the magnitude of the impulsive force $\vec{F}_{imp}$, we assume that the manipulator end-effector collides with a stationary workpiece in its workspace. This implies that the workpiece has zero velocity before and after the collision. Also, from the theory of rigid body collisions [16], we have

$$(\dot{x} + \Delta\dot{x})^T n_o = -\mu \dot{x}^T n_o \tag{4.5.6}$$

where n_o is the unit vector normal to the plane of collision impact between the end-effector and workpiece, and μ is the constant coefficient of restitution denoting the type of collision taking place. We note that μ has a value in the range $0 \le \mu \le 1$; when $\mu = 0$ we have purely plastic collision, i.e., the colliding bodies have zero relative velocity to each other immediately following the collision, and when $\mu = 1$ we have purely elastic collision, i.e., the total dynamic energy right before the collision is equal to the total dynamic energy immediately after the collision. Finally, we note that the impulsive force $\vec{F}_{imp}$ is directed along the normal direction n_o to the contact plane. Therefore, we can write $\vec{F}_{imp} = \hat{F}_{imp} n_o$, where the scalar $\hat{F}_{imp}$ represents the magnitude of $\vec{F}_{imp}$. Now, from equations (4.5.5) and (4.5.6), we can derive the following manipulator collision impact model

$$\hat{F}_{imp} = \frac{-(1+\mu)\dot{x}^T n_o}{n_o^T [J_e(q) D_l^{-1}(q) J_e^T(q)] n_o} \tag{4.5.7}$$

Equation (4.5.7) gives an expression for the magnitude of the impulsive force $\hat{F}_{imp}$ in terms of the manipulator's configuration q at impact, the end-effector velocity $\dot{x}$ right before collision, and the unit vector n_o normal to the plane of collision impact.

4.6 MINIMIZATION OF MANIPULATOR IMPACT EFFECTS

Collision impact occurs when a robot manipulator collides or comes in contact with its environment. At the time of collision, significant impulsive forces could be generated which may jeopardize the stability of the manipulator control system or even damage the manipulator or its environment. In this section, the augmented kinematics approach for redundant manipulators is used to reduce impulsive forces, we shall use the manipulator collision impact model of equation (4.5.7) and the augmented kinematics approach for redundancy resolution in order to reduce the impulsive forces. Also, the simplified impedance control strategy is applied to overcome the after-collision rebound effects as well as the impulsive forces. Thus, as we shall see in this Section, a combination of proper kinematics and control strategies results in the minimization of the total impact effects, (i.e., rebound effects and impulsive forces).

Before we present the details of impact effect reduction, we make the following assumptions: First, we assume that the workpiece with which the manipulator collides behaves as a spring with certain stiffness. This implies that the deformation of the workpiece provides a measure of the impulsive forces. Second, we assume that the links and joints of the manipulator are rigid. Third, we consider collisions or contacts between the manipulator's end-effector and its workpiece only.

4.6.1 Impulsive Force Reduction Using Configuration Control

When a human arm makes contact with an object (for example, a wall) the arm posture is usually adopted such that it minimizes the shock on the hand or the arm. More specifically, if one wants to make a "soft" contact between one's hand and a wall, most likely one will adjust one's forearm to be almost parallel with the surface of the wall rather than perpendicular to it. This is an example of how human beings use the redundancy in their arms to perform soft interaction tasks. Bearing this in mind, in this section we shall design a controller which allows redundant manipulators to perform contact tasks by choosing manipulator configurations that produce minimum impulsive forces.

Equation (4.5.7) indicates that the severity of the impact depends on the manipulator's configuration q. Therefore, as mentioned above, the main idea here is to use a manipulator's self-motion in order to reconfigure its posture in such a way that $\hat{F}_{imp}$ is minimized. To do this, we define the objective function $L(q, D_l)$ in equation (4.4.1) as follows

$$L(q, D_l) = \frac{-(1+\mu)\dot{x}^T n_o}{n_o^T [J_e(q) D_l^{-1}(q) J_e^T(q)] n_o} \tag{4.6.1}$$

Next, we assume that the Cartesian space trajectory of the manipulator has been planned, i.e., the end-effector velocity profile $\dot{x}$ has been determined in advance. Also, since the materials of the end-effector and the workpiece are known, the parameter μ (and thus the type of collision) is assumed to be known. In most cases, we also assume the geometry of the collision, i.e., parameter n_o is known. Under these assumptions, the minimization of $L(q, D_l)$ is equivalent to the minimization of the following simple objective function

$$L(q, D_l) = \frac{1}{n_o^T [J_e(q) D_l^{-1}(q) J_e^T(q)] n_o} \tag{4.6.2}$$

The minimization of this function requires the augmentation function z defined by equation (4.4.1). Note that in implementing this scheme, the z has to be computed at each time instant that the augmented simplified impedance controller is evaluated. In other words, when the manipulator end-effector tracks a given Cartesian trajectory, this additional augmented redundancy resolution loop is executed in parallel with the main feedback loop of the impedance controller, to control the self-motion of the manipulator. This will configure the manipulator links such that $L(q, D_l)$ in equation (4.6.2) is minimized. This implementation is particularly useful when the instant of contact or collision is not known.

It is worth mentioning here that the impact reduction objective function of equation (4.6.1) (or that of equation (4.6.2)) is a highly nonlinear function with respect to q. Thus, at any location along the Cartesian space trajectory there might be more than one local optimum for the objective function $L(q, D_l)$. We shall illustrate this point with a simple example. Suppose the tip of a three-link planar manipulator moves along a Cartesian space trajectory and arrives at the point $X = 2$ $(X = x_1)$, $Y = 0$ $(Y = x_2)$, and makes contacts with the surface of a workpiece as is shown in Figure 4.2. Assuming that we can freeze the time t at that time instant, we change the configuration of the manipulator by varying the angle α_l between the surface of the workpiece and the third link of the manipulator. For all different configurations of the manipulator the normalized impulsive forces F^{*}_{imp} are

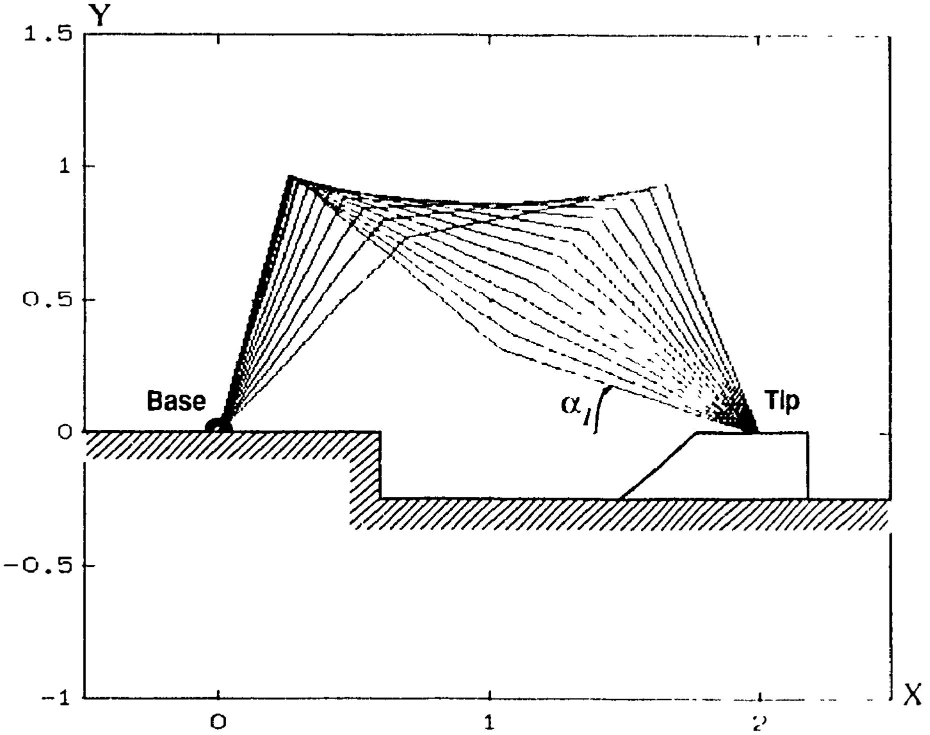

Figure 4.2 **Different configurations for Impact**

calculated using equation (4.6.2) and are plotted against the angle α_l. The plot shown in Figure 4.3 demonstrates that the impulsive force is highly nonlinear and indicates that multiple minimum points may exist.

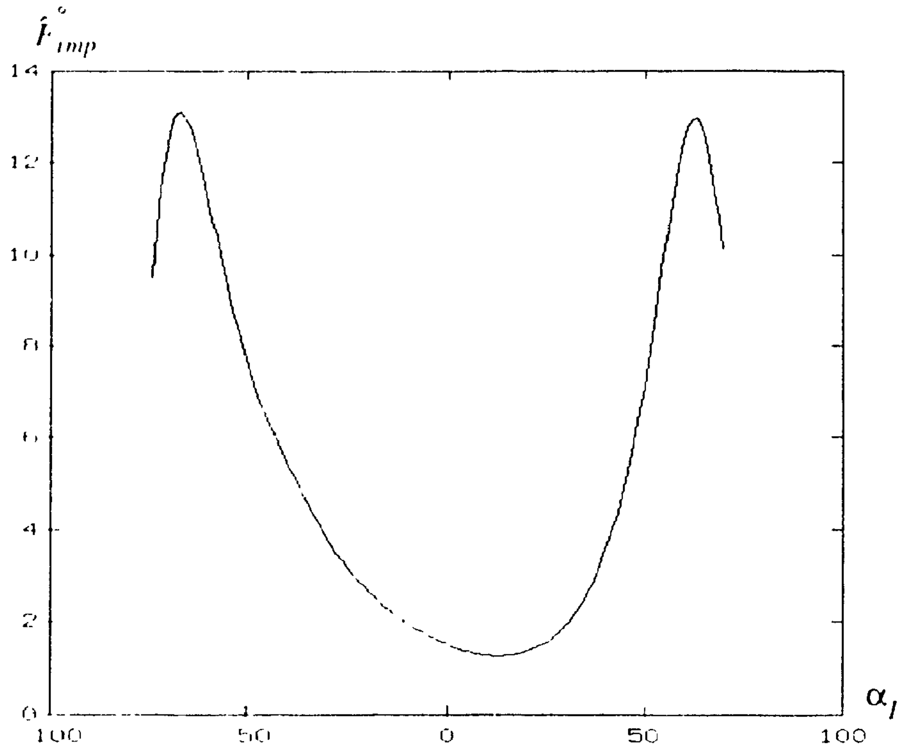

Figure 4 3 Profile of $\hat{F}^c_{imp}$ with respect to α_l

In order to avoid being trapped in an undesirable configuration (due to multiple optima), it is desirable to bring the manipulator to a region close to the point of contact with a desired configuration. Note that it may not always be possible to bring the manipulators to the region of the point of contact with a desirable configuration. This depends on the workspace topology. For example, various objects in the workspace may prevent the manipulator from achieving a desirable configuration. In the following figures, we show various workspace topologies and the corresponding optimal manipulator configurations. To some extent, Figures 4.4 and 4.5 demonstrate that the problem of choosing the optimal configurations may not have a solution due to the presence of obstacles. In this case, we may choose a near-minimum solution which is the solution to the combined problem of manipulator collision impact reduction and obstacle avoidance. Thus, the solution to this "optimal" configuration problem may require consideration of multiple criteria in redundancy resolution.

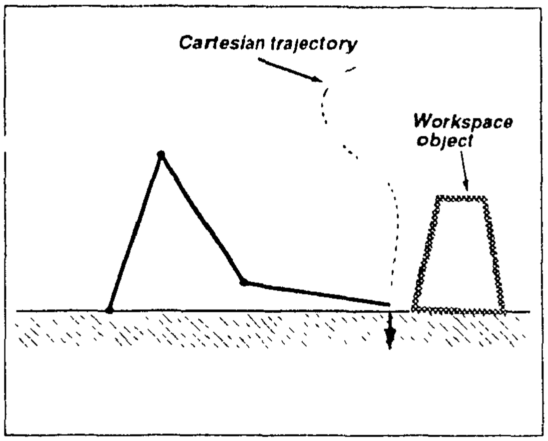

Figure 4 4 **Constraints in workspace: case 1**

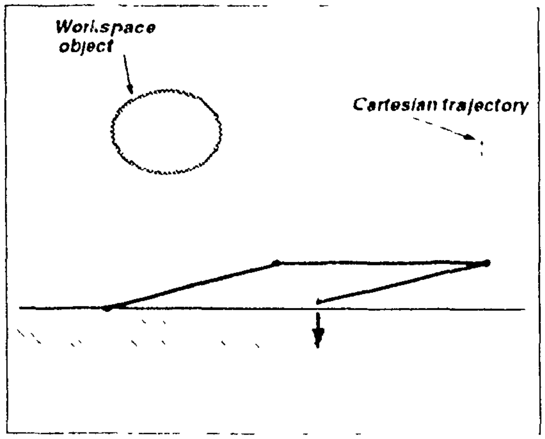

Figure 4 5 **Constraints in workspace: case 2**

Another point to note in our computer simulations concerns the configuration of the last link (to which the end-effector is rigidly attached) with respect to the contact surface of the workspace object. We denote this by the "contact angle", which for the example, in Figure 4.2 is given by α_l. The impulsive force generated at the time of collision depends very much on this angle. It was observed that the point of contact received the smallest amount of impulsive force when the last link is configured to be in parallel with the surface of the workspace object. In our example, all the joints and the links were considered to be rigid and thus, the entire manipulator was treated as a rigid mechanical structure. From the mechanical point of view, the configurations of the remaining links obviously affect the propagation of the impact force from the tip of the last link to the base of the manipulator. Therefore, the variation of the contact angle between the last link and the object surface is important for the magnitudes of both the impulsive forces and the corresponding joint torques. However, this variation is usually kept to a small range. This point is verified in the simulations shown later in this chapter.

4.6.2 Reduction of Rebound Effects Using Augmented Simplified Impedance Control

The reaction force (which depends on the materials used for the object and the end-effector) generated at the time of impact tends to push the manipulator end-effector away from the object surface. Application of an impedance controller will force the end-effector to bounce back toward the object surface when the position, velocity and acceleration errors between the desired and actual Cartesian space trajectories generate large enough impedance forces in the manipulator's controller. Usually, this will cause oscillatory behavior for the manipulator end-effector, and may lead to instability in the closed-loop system. In the worst case scenario, it could even damage the end-effector as well as the object. Ideally, we would like the collision between the manipulator end-effector and object surface to be plastic with small impact forces, i.e., there is no rebound of the end effector after contact. However, in practice the object surface is usually hard, and the end-effector is made of hard metal as well. Therefore, the collision is likely to happen in an elastic way; the degree of elasticity depends on how hard the object surface and the end-effector are. From the foregoing, it is necessary to use a control scheme which will supervise manipulator impact control. Here, we propose a simplified impedance control strategy which reduces the end-effector oscillatory behavior and bouncing effects.

As we saw in Section 4.3.2, the simplified impedance controller has been formulated by choosing the desired inertia matrix M to be identical to the inverse of the manipulator's *mobility tensor* W expressed in Cartesian space. Based on (4.4.2), the complete augmented simplified controller can be rewritten as

$$\tau = J^l \, \{ B\,(\dot{y}_d - J\dot{q}) + K\,(y_d - y) \; \} + (C_l + G_l + F_l) + J^l W^{-1} \, [\ddot{y}_d - \dot{J}\dot{q} \,] \qquad (4.6.3)$$

The terms related to the Coriolis and centrifugal forces, the gravity force and the frictional force are included in the control in order to cancel out the corresponding nonlinear terms in the dynamic model. This can be implemented using an inner feedback loop as shown in Figure 4.6. At the same time, an outer feedback loop carries the Cartesian feedback infor-

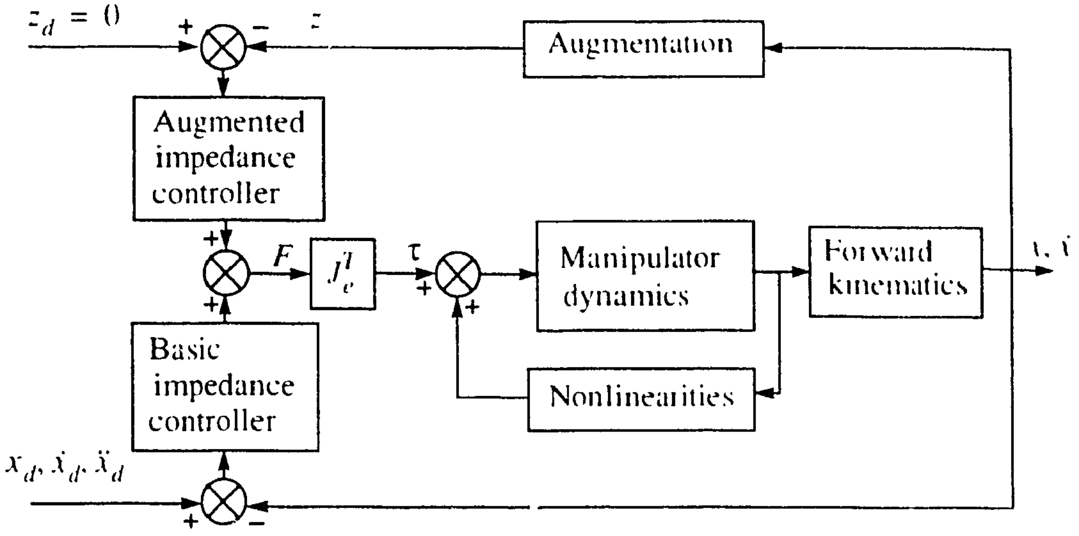

Figure 4 6 **Block diagram of the simplified augmented impedance controller**

mation (calculated through forward kinematics or measured directly) from which the desired system impedance is calculated.

As mentioned previously, the desired inertia matrix M plays an important role in impedance control for impact minimization. Obviously, the larger the inertia that an object has, the more difficult it is to change its state. The object obstructs the motion of the end effector when the manipulator makes contact with it. In order to change the state of the end-effector, a greater effort is needed due to the larger inertia matrix M. Therefore, a larger impulsive force is generated. At the same time, since the impedance controller is usually set to be in critically damped condition, a larger inertia matrix M will also result in larger damping and stiffness matrices. This will further generate larger impedance and, therefore, result in larger impulsive forces. On the other hand, a smaller desired inertia matrix M results in smaller impulsive forces. However, a small inertia matrix M results in oscillatory behavior of the end-effector, and increases rebound effects. Sometimes it may even cause instability in the system. Therefore, considering these two opposite scenarios, the simplified impedance controller provides an "optimal" solution for this problem.

Practically speaking, there are always some mismatches in dynamic models, and the stiffness of an object surface may be variable. This will affect the impact dynamics and consequently may produce oscillations. However, these oscillations can be damped out by setting the impedance controller to provide critical damping.

4.7 COMPUTER SIMULATION RESULTS

In this section, we demonstrate the impact reduction property of the proposed control strategy presenting computer simulations of some examples. The manipulator which we use in these examples is a rigid three-link planar redundant manipulator shown in Figure 4.7. The joint space dynamic model of this simple three-link planar redundant manipulator is described in [6] with the following equations

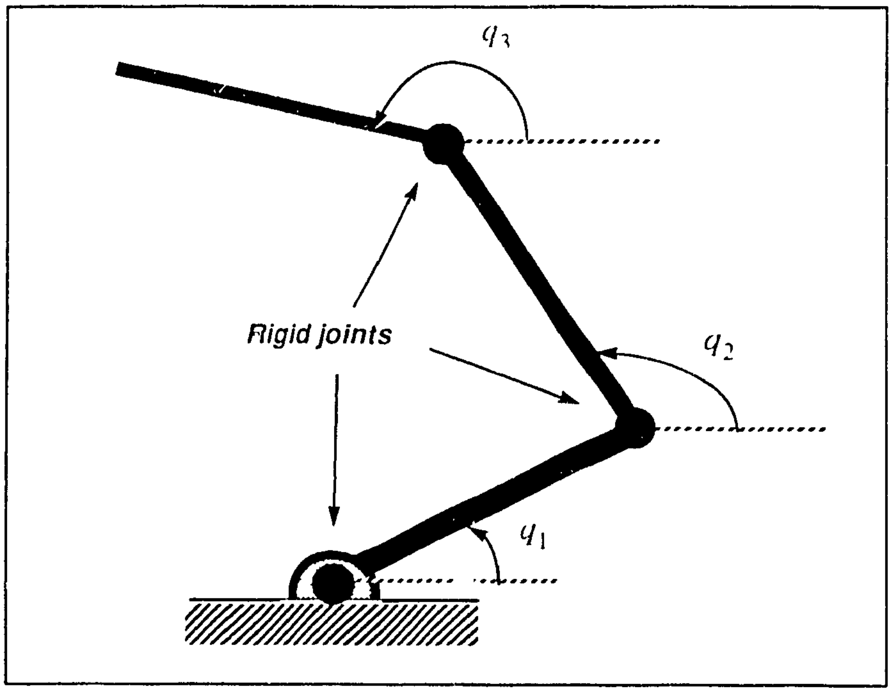

Figure 4 7 **Three-link planar redundant manipulator**

$$\tau = D_l(q)\ddot{q} + C_l(q,\dot{q}) + G_l(q) + F_l(\dot{q}) + J_e F \tag{4.7.1}$$

Each link is assumed to have the same length $l_1 = l_2 = l_3 = 1m$ and the same mass $m_1 = m_2 = m_3 = 10kg$. For simplicity, the links are modeled with point mass at their distal ends. The surface of the workspace object is modeled as a spring with stiffness $K_s = 10^5 \sim 10^6$ to represent a hard surface.

Simulations were performed using MATLAB on a SUN/SPARC-2 workstation. In order to verify that the end-effector receives minimum impulsive force with respect to an appropriate configuration at the time of collision, (see Figure 4.1), we first calculated the impulsive forces using equation (4.6.1) with respect to different configurations at the impact point ($X = 2$ $(X = y_1)$, $Y = 0$ $(Y = y_2)$) with desired Cartesian velocity v. The impulsive force profile for different configurations of the manipulator is shown in Figure 4.2, where the parameter α_l defines the angle between the surface of the object and the third link of the manipulator. Note that there is no end-effector in this simplified example. Therefore, the tip of the third link is to come in contact with the environment. Also, note that each different value of α_l represents a different configuration of the manipulator. As we can see from Figure 4.2, the manipulator receives the minimum impulsive force when $\alpha_l \in [10^\circ, 15^\circ]$. Knowing the minimum impulsive force configuration, we design the following Cartesian trajectory

$$y_{d1} = 1.5 + 0.5\sqrt{2}\cos\left(\omega t - \frac{\pi}{4}\right)$$

$$y_{d2} = 0.5 - 0.5\sqrt{2}\sin\left(\omega t - \frac{\pi}{4}\right) \tag{4.7.2}$$

and select the objective function for redundancy resolution as in equation (4.6.2). The simplified impedance controller of equation (4.6.3) was used as the controller for the redundant manipulator in our examples. Figure 4.8 shows the manipulator initial configuration at $q = [90^\circ, -90^\circ, 0^\circ]^T$, the motion along the desired Cartesian trajectory (defined by (4.7.2)), and the collision between the tip of the third link and the workspace object at the point (2,0). As can be seen in this figure, at the time of collision the angle between the object surface and the third link of the manipulator has a value in the interval [10°,15°] as predicted in Figure 4.2. This implies that the impulsive force at the tip of the third link at impact is minimized. This confirms the theoretical analysis presented in the preceding sections.

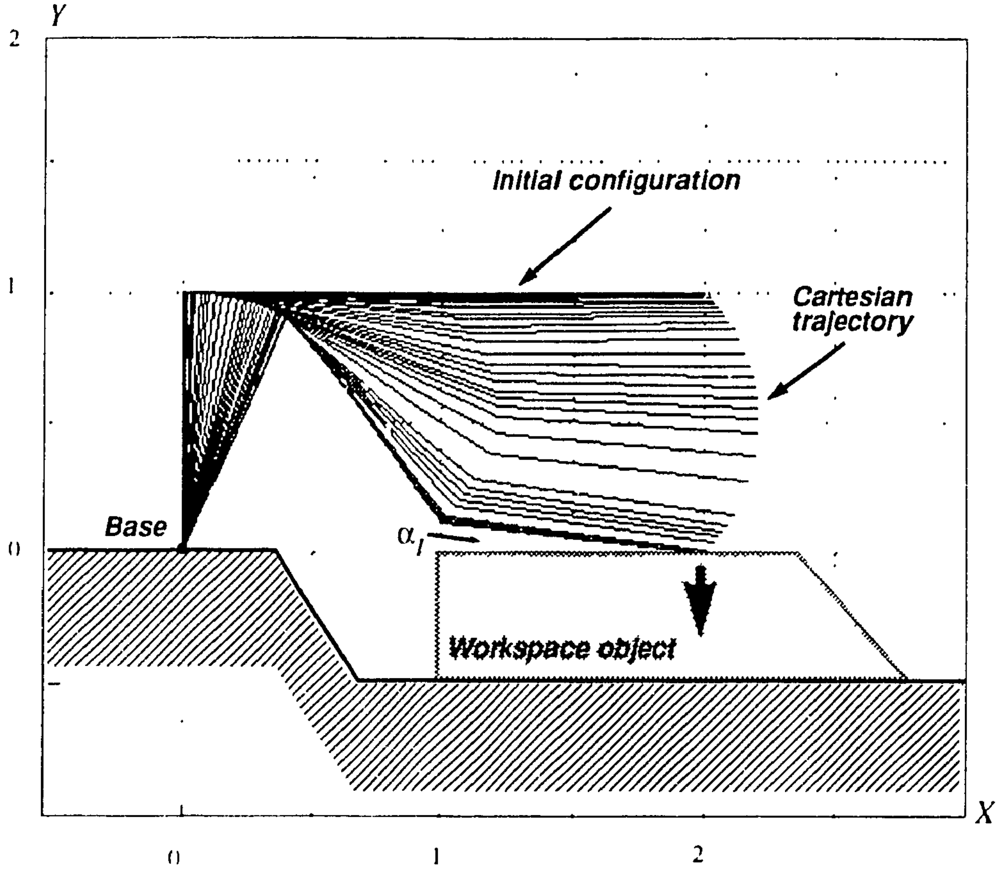

Figure 4 8 **Manipulator configurations before and at the time of collision**

Further details of this computer simulation can be illustrated by analyzing the actual end-effector trajectories in Cartesian space, the actual joint-space trajectories, and the actual torque profiles. Two different simulations were performed. The first simulation, with results shown in Figures 4.9-4.17, uses the proposed simplified impedance controller, i.e., the desired mass matrix is defined as $M = [JD_l^{-1}J^T]^{-1}$. This should be compared with the results of the second simulation shown in Figures 4.18-4.26 where there is significant mismatch between M^{-1} and the mobility tensor $JD_I^{-1}J^T$. As can be seen from Figures 4.9 and 4.10, the end-effector rebound effects are controlled within acceptable levels in comparison with those in Figures 4.18 and 4.19 where the desired inertia matrix M^{-1} is selected to have much higher value than the mobility tensor $JD_I^{-1}J^T$. Also, it can be seen that the resultant impulsive forces/torques in the first simulation (Figures 4.14-4.17) are much lower (at least by a factor of 2) than those in the second simulation (Figures 4.23-4.26). This is to be expected because the desired inertia matrix in the second simulation is large enough to ensure that the end-effector has hardly any rebound, but internally the impulsive forces/torques are very large due to the high value of the desired inertia matrix. As far as the control of collision impact is concerned, this clearly indicates that the simplified impedance control scheme is superior to the conventional impedance controller. It also verifies the assumptions and the analysis which were made earlier in this chapter.

4.8 CONCLUDING REMARKS

In this chapter the problem of controlling redundant manipulators to reduce collision impact effects has been considered, and an augmented kinematics and impedance control scheme has been proposed for its solution. The proposed scheme achieves satisfactory performance by minimizing the magnitudes of impulsive forces as well as reducing rebound effects of the end-effector.

In order to resolve the manipulator's redundancy, the augmented kinematics approach was used so that the manipulator Jacobian matrix is augmented to a square matrix In the proposed control scheme, the augmentation of the Jacobian matrix is based on an impact model derived using the Cartesian space dynamic model of the manipulator. This results in a configuration of the redundant manipulator that gives the smallest amount of impulsive force at the end-effector while the end-effector still follows a prespecified Cartesian trajectory.

The impact controller is also based on a simplified impedance control scheme aimed at reducing the impulsive forces as well as the rebound effects. In the simplified impedance control scheme, the inverse of the desired inertia matrix is chosen to be identical to the mobility tensor of the manipulator in Cartesian space. This ensures that the end-effector at the time of impact generates acceptable rebound effects while keeping the internal impulsive force at an acceptable level. The performance of the proposed controller has been illustrated by computer simulations for several examples.

The topic of impact control and, in particular, impact control using redundancy, is relatively new. The approaches proposed here form only the first step towards the development of more general control schemes which can be applied to a variety of practical manipulator/environment contact situations.

4.9 REFERENCES

[1] Baillieul, "Kinematic programming alternatives for redundant manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 722-728, St. Louis, MO, 1985.

[2] M.W. Gertz, J.O. Kim and P. Khosla, "Exploiting redundancy to reduce impact force," *IEEE/RSJ Intern. Workshop on Intellig. Robots and Systems IROS'91*, pp. 179-184, 1991.

[3] N. Hogan, "Impedance control: An approach to manipulation, parts I-III," *ASME J. Dyn. Syst Meas. Contr.*, vol. 107, pp. 1-24, 1985.

[4] N. Hogan, "Stable execution of contact tasks using impedance control," *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 1047-1054, 1987.

[5] J.M. Hollerbach and K.C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE Int. J. Robotics. Automat.*, vol. RA-3, pp. 308-315, 1987.

[6] A.J. Koivo and S.H. Arnautovic, "Dynamic optimum control of redundant manipulators," in Proc. IEEE Conf. Robotics and Automat., pp. 466-471, 1991.

[7] Z. Lin, R.V. Patel and C.A. Balafoutis, "Impedance control of redundant manipulators for minimization of collision impact," *Proc. American Control Conf.* pp. 1237-1238, 1992.

[8] J.Y.S. Luh, M.W. Walker and R.P. Paul, "Resolved-acceleration control of mechanical manipulators," *IEEE Trans. Auto. Contr.*, vol. AC-25, 1980.

[9] A.A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robotics Res.*, vol. 4, no. 3, pp. 109-117, 1985.

[10] J.K. Mills, "Stability of robotic manipulators during transition to and from compliant motion," *Automatica*, vol. 26, pp. 861-874, 1990.

[11] H. Seraji, "Configuration control of redundant manipulators: theory and implementation," *IEEE Trans. Robotics Automat.*, vol. 5, pp. 472-490, Aug. 1989.

[12] H. Seraji and R. Colbaugh, "Improved configuration control for redundant robots," *J. Robotic. systems.*, vol. 7(6), pp. 897-928, 1990.

[13] R. Volpe and P. Khosla, "Experimental verification of a strategy for impact control," *Proc. of IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, pp. 1854-1860, 1991.

[14] I.D. Walker, "The use of kinematic redundancy in reducing impact and contact effects in manipulation," *Proc. of IEEE Int. Conf. on Robotics and Automation*, Cincinnati, OH, pp. 434-439, 1990.

[15] Y. Wang and T. Mason, "Modeling impact dynamics for robotic operations," *Proc of IEEE Int. Conf. on Robotics and Automation*, Raleigh, NC, pp 678-685, 1987.

[16] J. Wittenberg, *Dynamics of Systems of Rigid Bodies*, Stuttgart: B G. Teubner, 1977.

[17] K. Youcef-Toumi and D. Gutz, "Impact and force control," *Proc of IEEE Conf on Robotics and Automation*, pp. 410-416, 1989.

[18] Y.F. Zheng and H. Hemami, "Mathematical modeling of a robot collision with its environment," *J. Robotic Systems*, vol. 2, pp. 289-307, 1985.

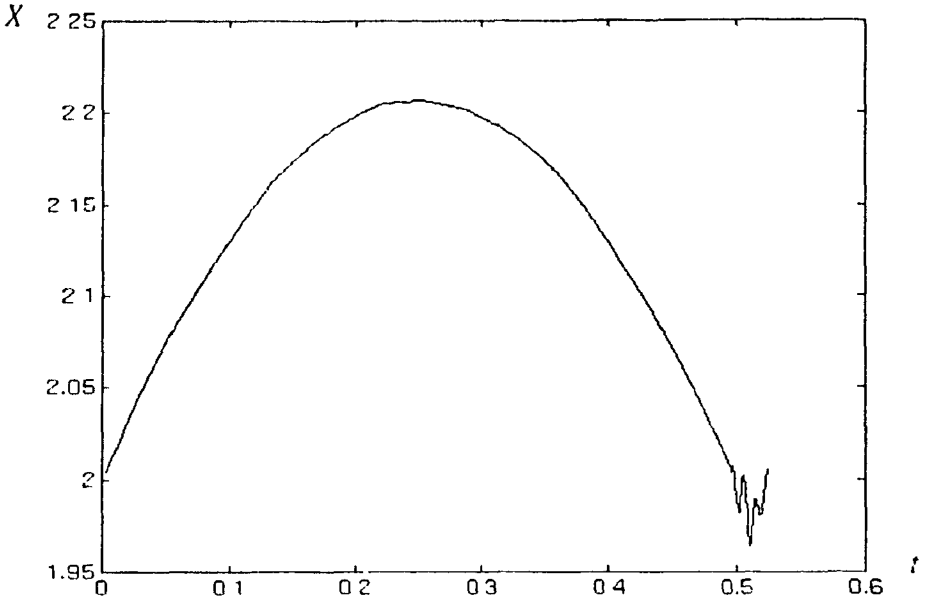

Figure 4.9 **Actual end-effector Cartesian trajectory along *X* axis (proposed approach)**

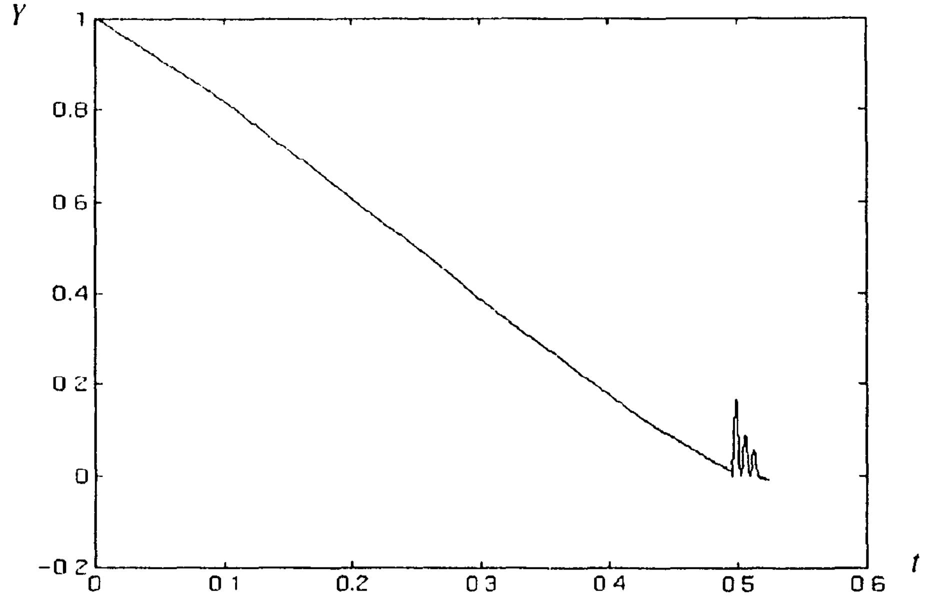

Figure 4.10 **Actual end-effector Cartesian trajectory along Y axis (proposed approach)**

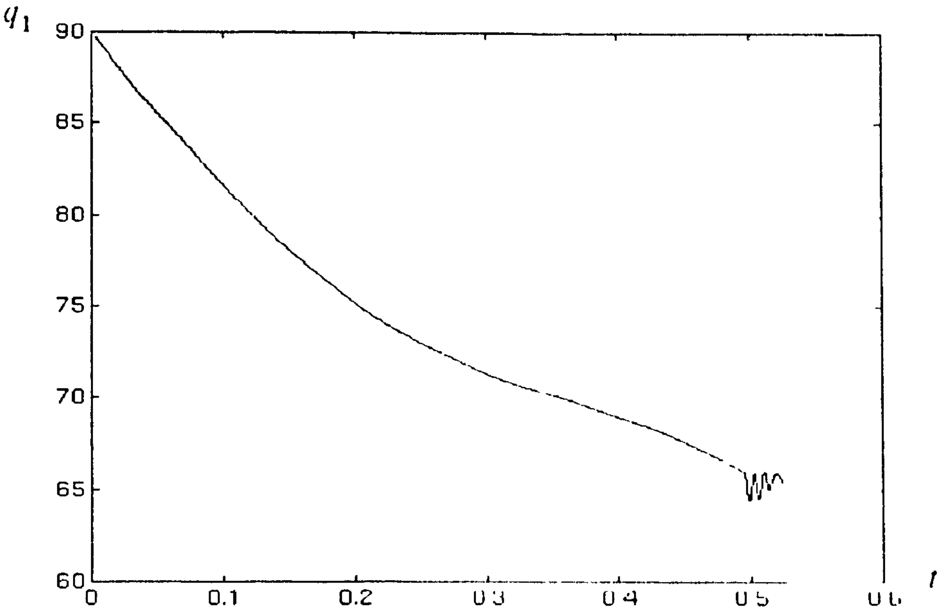

Figure 4.11 **The first joint trajectory (proposed approach)**

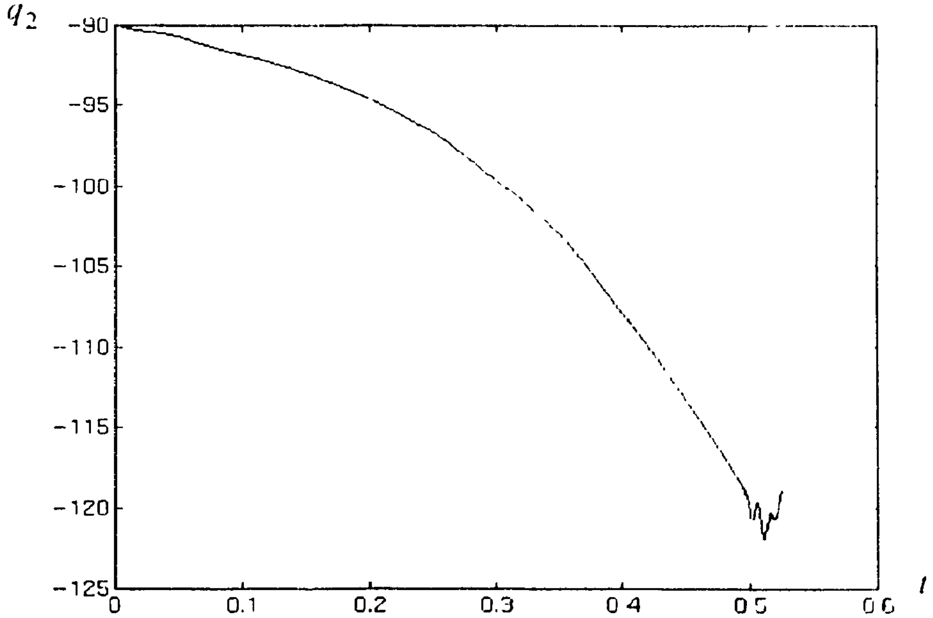

Figure 4 12 **The second joint trajectory (proposed approach)**

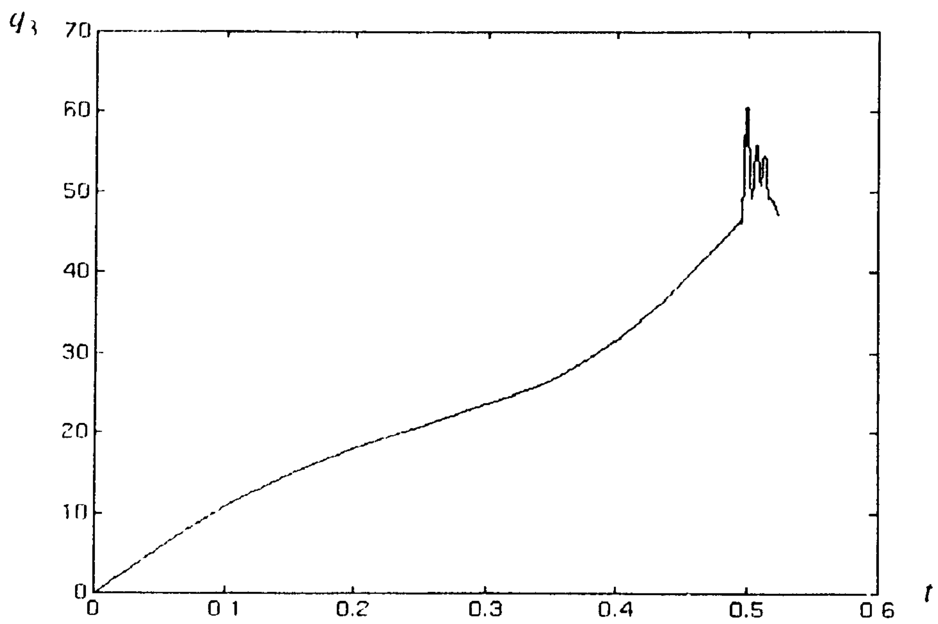

Figure 4.13 **The third joint trajectory (proposed approach)**

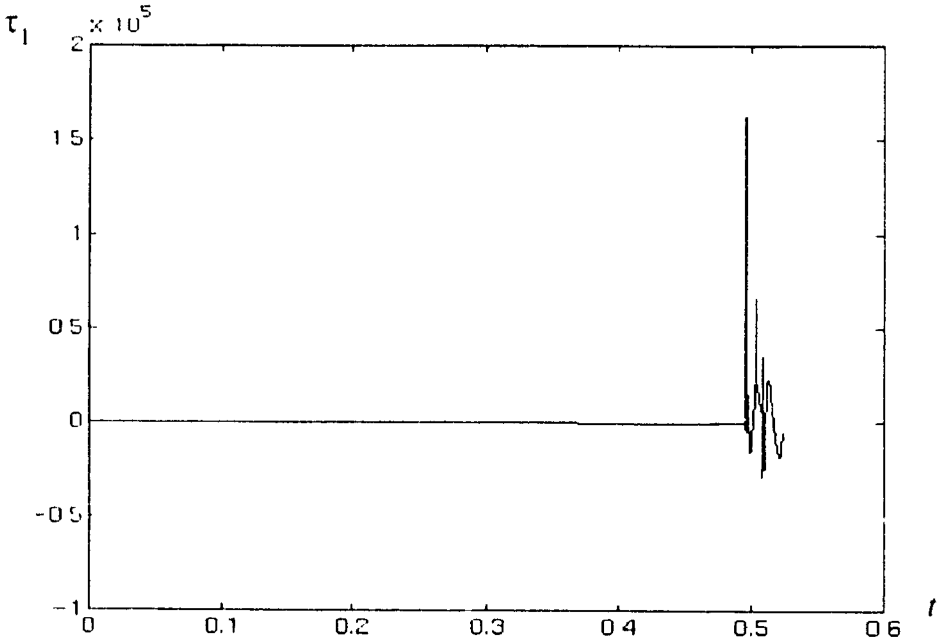

Figure 4 14 **Torque profile for the first joint (proposed approach)**

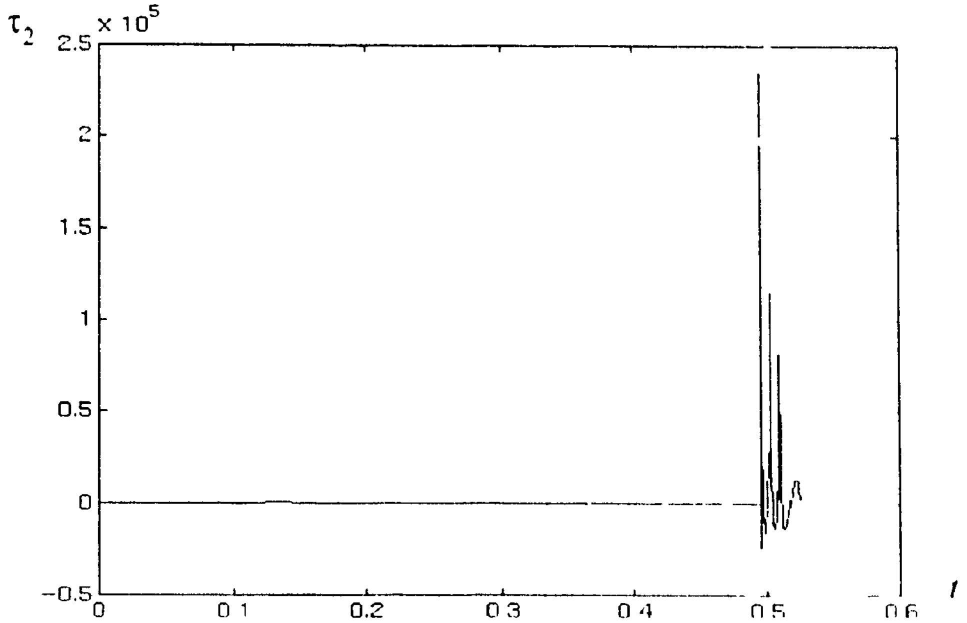

Figure 4.15 **Torque profile for the second joint (proposed approach)**

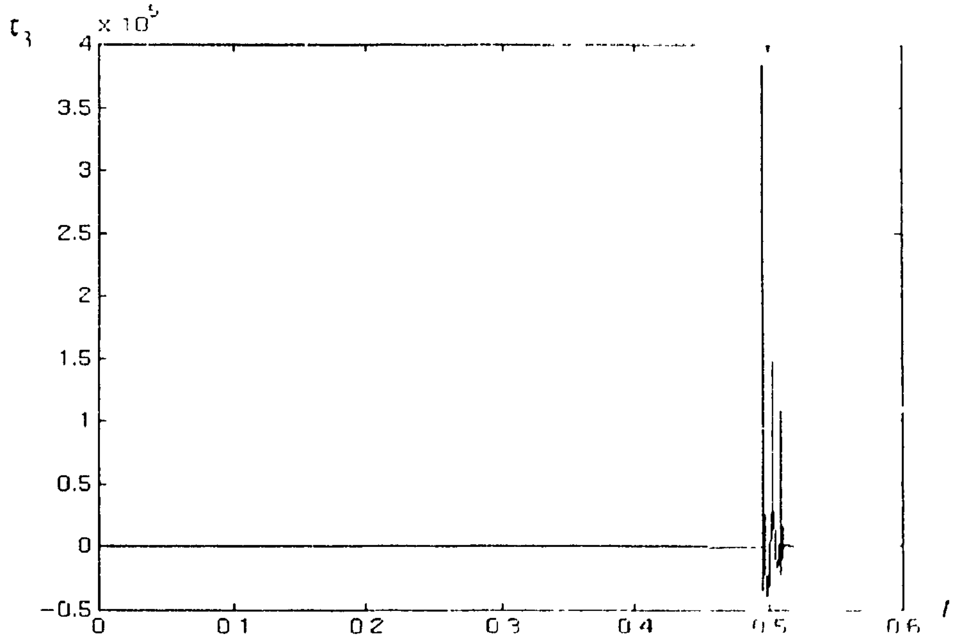

Figure 4.16 **Torque profile for the third joint (proposed approach)**

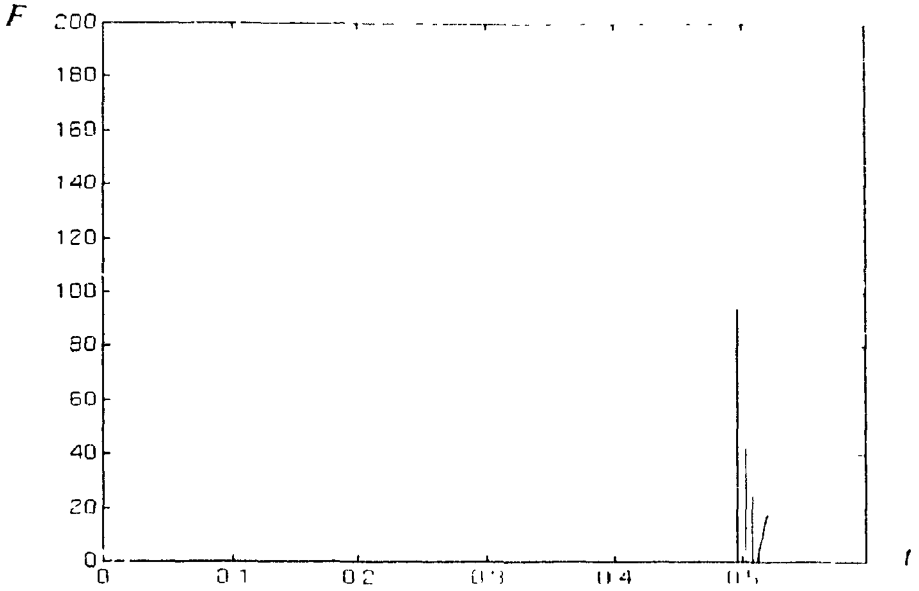

Figure 4 17 **End-effector force profile (proposed approach)**

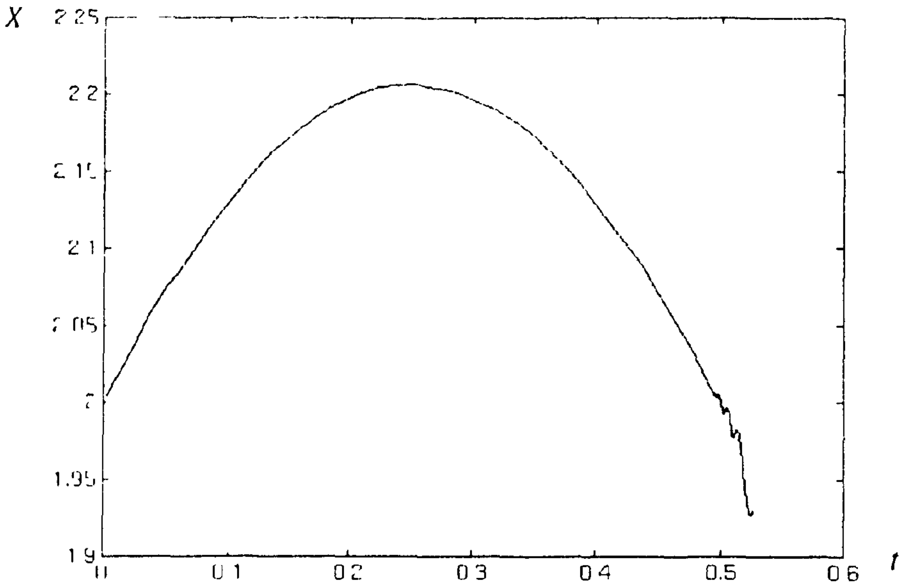

Figure 4.18 **Actual end-effector Cartesian trajectory along *X* axis (conventional approach)**

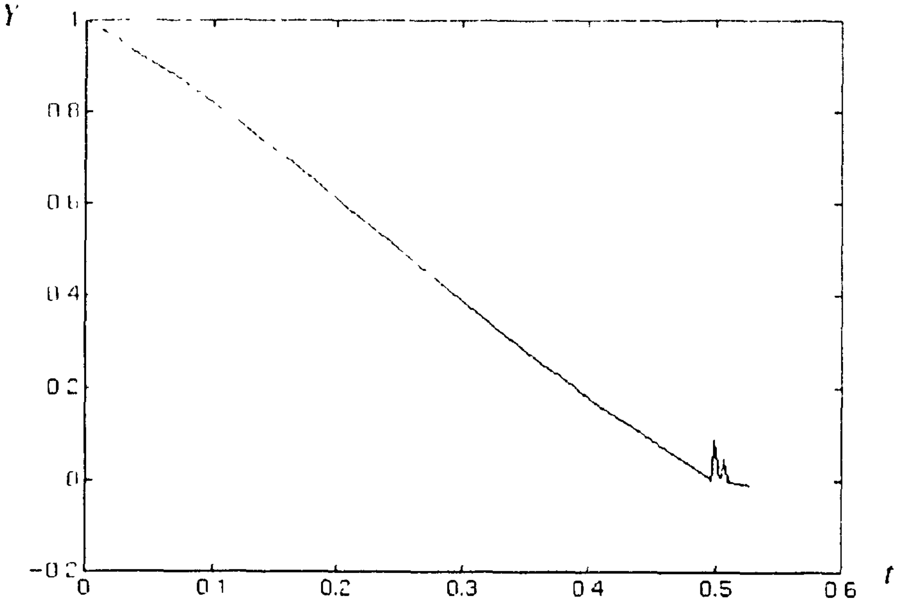

Figure 4.19 **Actual end-effector Cartesian trajectory along Y axis (conventional approach)**

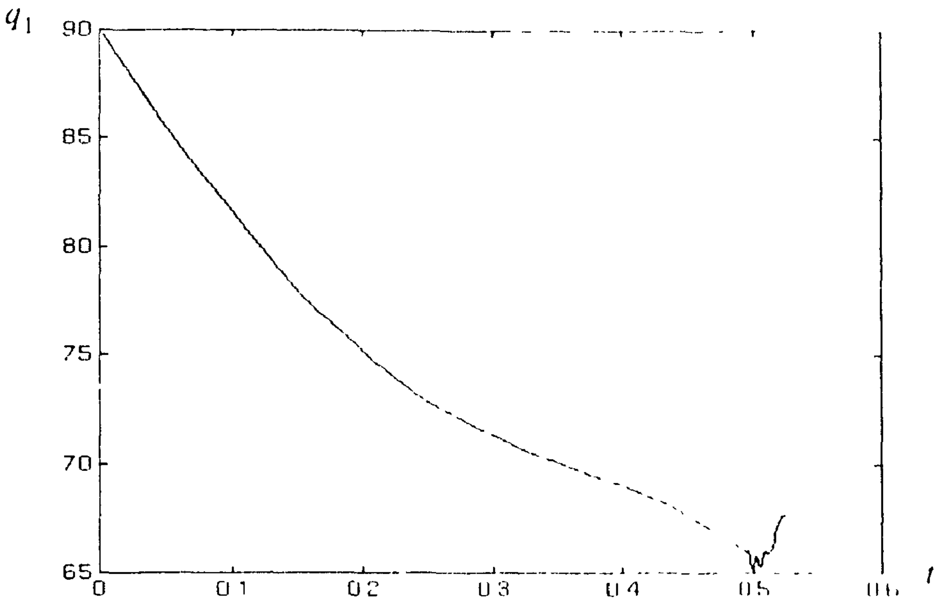

Figure 4.20 **The first joint trajectory (conventional approach)**

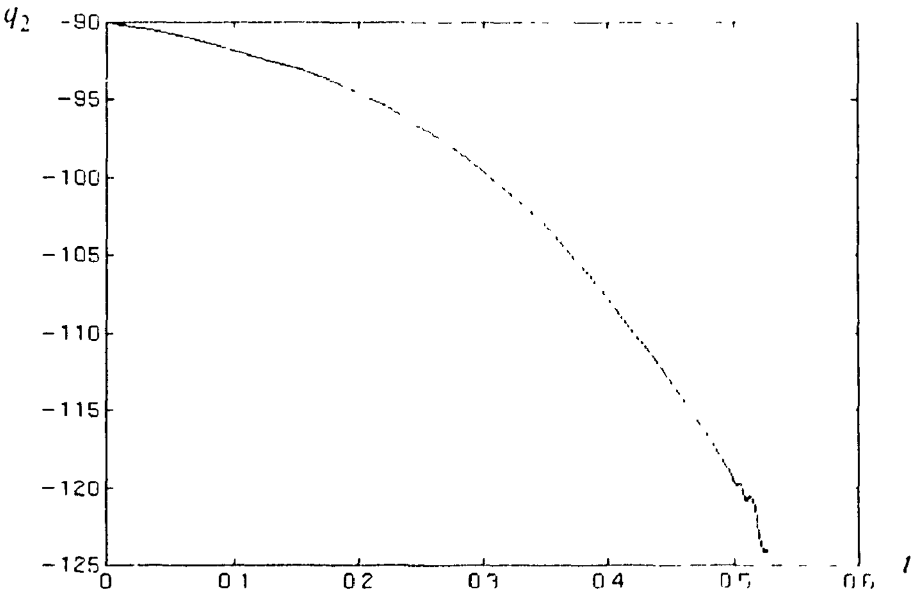

Figure 4 21 **The second joint trajectory (conventional approach)**

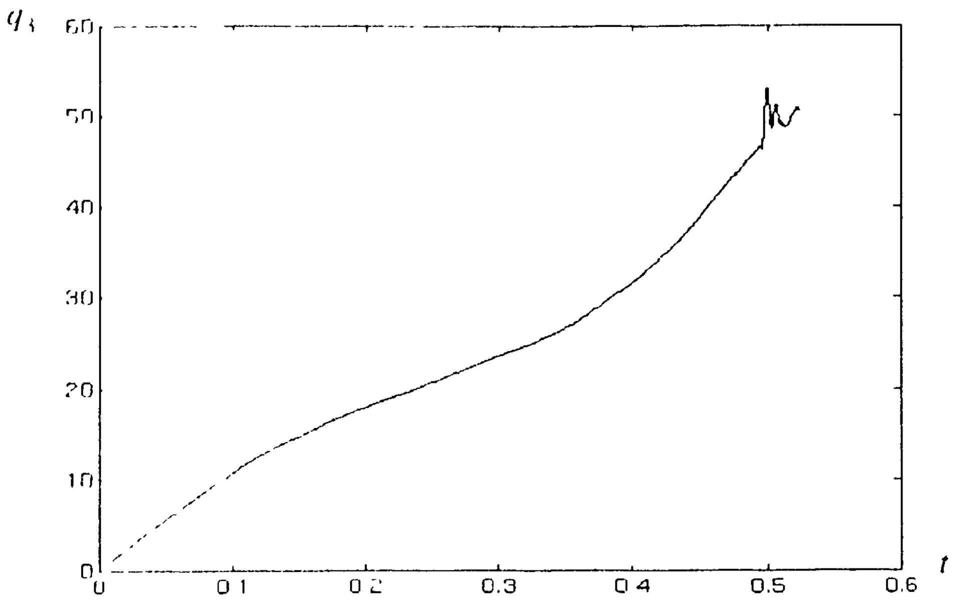

Figure 4.22 **The third joint trajectory (conventional approach)**

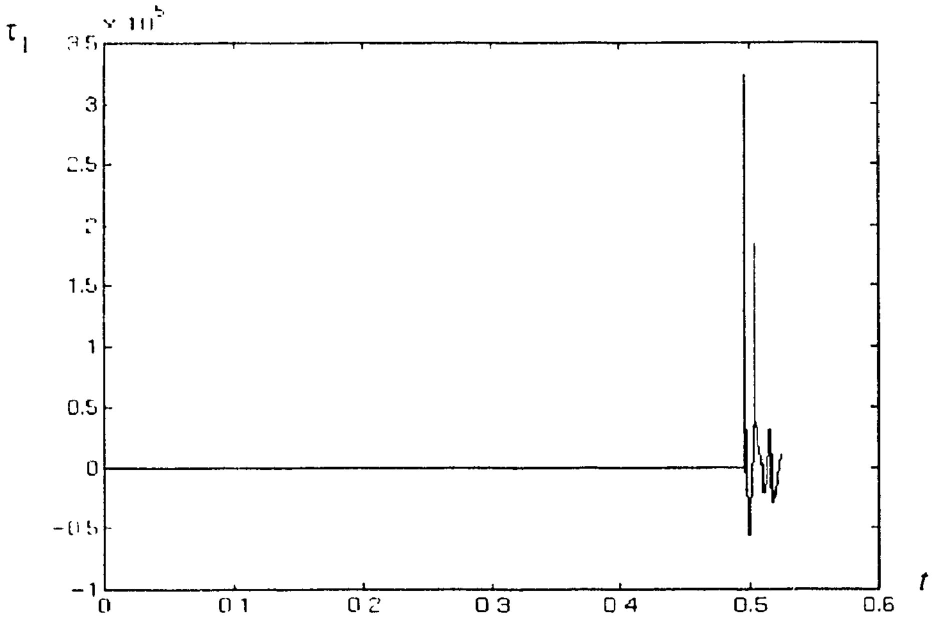

Figure 4 23 **Torque profile for the first joint (conventional approach)**

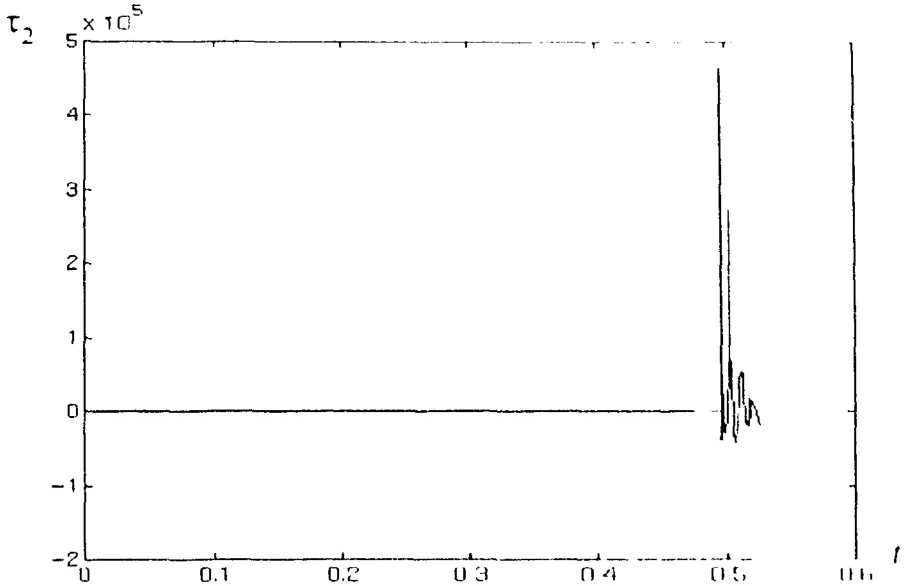

Figure 4.24 **Torque profile for the second joint (conventional approach)**

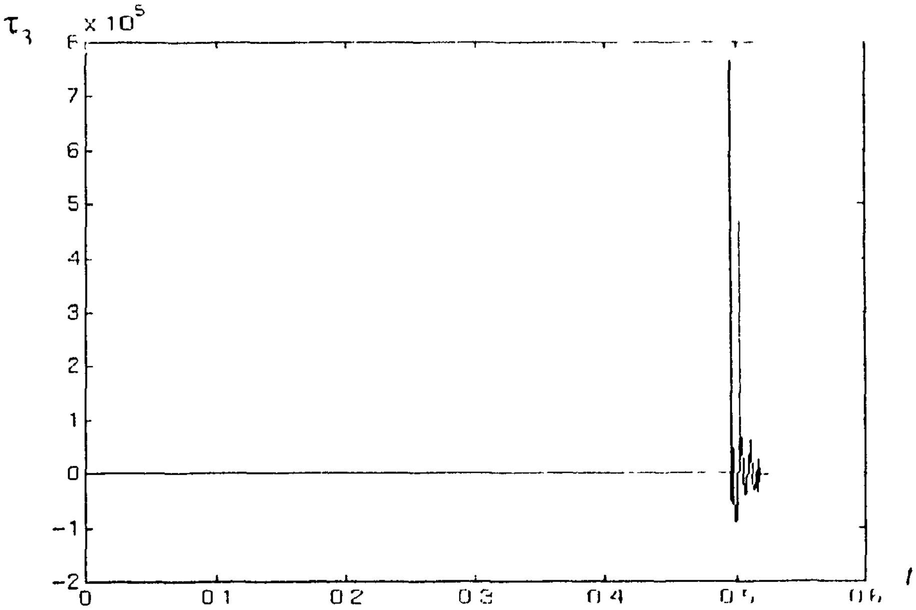

Figure 4.25 **Torque profile for the third joint (conventional approach)**

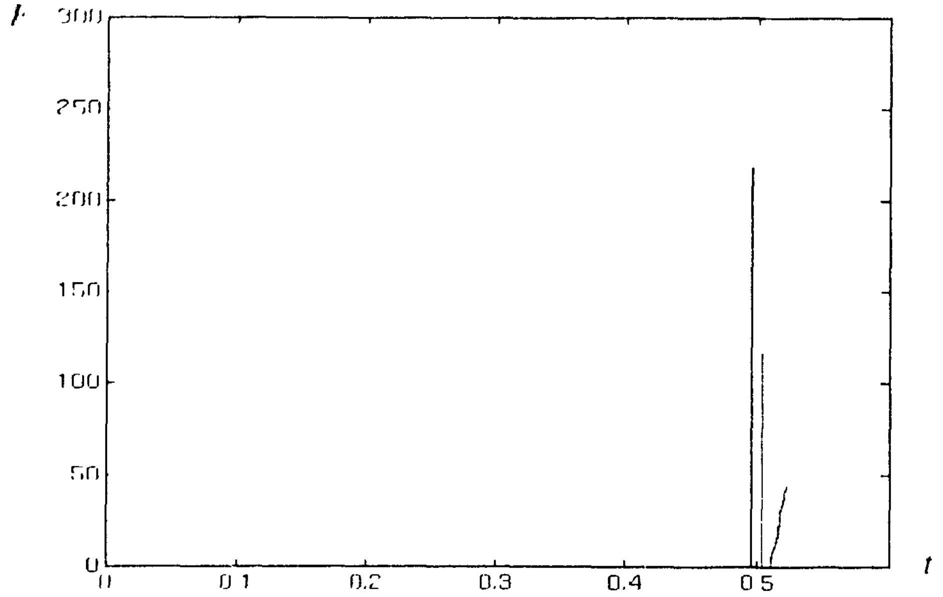

Figure 4.26 **End-effector force profile (conventional approach)**

CHAPTER 5 CARTESIAN CONTROL OF REDUNDANT FLEXIBLE-JOINT MANIPULATORS

5.1 INTRODUCTION

Control of rigid-joint robot manipulators has been thoroughly studied in the last two decades [1][7][10][12][17][21][23]. Many of the approaches have been successfully implemented on industrial robot manipulators. However, with the increasing demands being placed on robot manipulators to perform high speed, high-precision tasks, the problem of dealing with joint flexibility has become important

Joint flexibility usually comes from gear elasticity, shaft wind up, and the use of harmonic drives. Therefore, the dynamics describing rigid-joint manipulators, and the strategies designed to control manipulators with rigid joints may not be applicable to flexible joint manipulators. In order to control manipulators with joint flexibility, first we must consider a more accurate representation of a manipulator's dynamics that involves "flexible" modes, and develop control strategies that can control these modes. As we will see in the literature review in the next section, almost all the existing control schemes for flexible-joint manipulators have been based in joint space. These schemes are suitable for non redundant flexible-joint manipulators. However, as in the non-redundant case, redundant manipulators may also have joint flexibility. Because of the need to resolve redundancy, existing joint space control schemes for non-redundant manipulators may not be suitable for the redundant case. Therefore, it is necessary to develop Cartesian control schemes for redundant flexible-joint manipulators. The dynamic model of a flexible-joint manipulator was deri 'ed in Chapter 3. This chapter will focus on the issue of designing a control strategy for a *redundant* flexible-joint manipulator.

In this chapter, A new control strategy called *hybrid Cartesian-joint control* is introduced. The construction of this control scheme is based on the analysis of flexible-joint non-redundant joint-space schemes and flexible-joint non-redundant Cartesian-space scheme. The important issues of disturbance due to joint flexibility, control of the manipulator's self-motion, and the relationship between joint disturbances and self-motion are also addressed. This leads to the proposal of a *hybrid Cartesian-joint control* scheme, which is compose^d of a Cartesian tracking controller, a link tracking controller, and a motor tracking controller. At the same time, redundancy can still be used for the secondary tasks to be performed by the manipulator while the end-effector tracks a Cartesian trajectory. The rest of the Sections in this chap'er are arranged as follows: Section 5.2 gives an overview of the existing approaches for the control of flexible-joint manipulators. In Section 5.3, a brief review of joint control of flexible-joint non-redundant manipulators based on a nonlinear control strategy is presented, and the extension of this scheme to Cartesian control of flexible-joint non-redundant manipulators is discussed in Section 5.4. Section 5.5 discusses some difficulties in directly extending the approach discussed in Sections 5.3 and 5.4, and presents a new method called *hybrid Cartesian-joint control.* Stability analysis for the proposed controller is given in Section 5.6 using the Lyapunov approach. Section 5.7 discusses some computational issues in the control strategy. Finally, computer simulations to test the proposed control strategy are given in Section 5.8, and Section 5.9 draws some conclusions concerning the topic o" this chapter.

5.2 OVERVIEW OF EXISTING CONTROL STRATEGIES

The techniques currently used for controlling manipulators with joint flexibility can be categorized mainly into: (1) *a singular perturbation formulation* of the dynamic model and the *integral manifold* approach [13][25][26]; (2) *an exact linearization* approach [9][25]; and (3) *a passivity-based design* approach [2][16][28]. All of these approaches are in joint space, and have been developed for non-redundant manipulators.

The singular perturbation formulation for flexible-joint manipulators was used by Spong, Khorasani and Kokotovic [13][25][26]. The model derived using the singular perturbation technique is useful for cases where the elasticity in the joints is of greater significance than gyroscopic interactions between the motors and links. The singular perturbation parameter μ is defined as the inverse of the joint stiffness. In this formulation, the link positions and velocities are the "slow" variables, while the joint torques and their rates are defined as the "fast" variables. In [13][26], the concept of an invariant manifold is utilized. This leads to a reduced-order dynamic model of the same dimension as the rigid model, but incorporating the effects of joint flexibility. Based on this reduced order flexible model, a corrective control strategy is formulated to compensate for flexibility in the joints. The overall control strategy consists of a *rigid controller* designed for the rigid system, and a *corrective controller* to compensate for deviations of the flexible system response.

Feedback linearization and decoupling of flexible-joint manipulator dynamics were discussed in [8][9][24]. In contrast to rigid manipulator dynamics, linearizability of flexible-joint dynamics depends on what type of model is used. It is pointed out in [25] that the manipulator dynamics are feedback linearizable using static state feedback only if a simplified model is used, while the full model is linearizable using dynamic feedback. Control strategies and robustness properties for the linearized system are discussed in [24][25].

The concept of passivity is traditionally defined as an input/output condition describing a class of physical systems that do not generate energy [18]. This property has been used in feedback stabilization for rigid manipulators [27] and flexible-joint manipulators [2][28]. The passivity property for flexible-joint manipulators, i.e., the fact that the motor torque and motor velocity form a passive pair, was recognized in [2], and was used in a proportional-derivative (PD) type controller design. The method requires inherent damping in both links and motors. However, results that do not require inherent damping have recently appeared in [28], where system stability and robustness analysis using a simple PD controller are given with respect to uncertainties on the manipulator parameters. But in [28], only the regulation problem has been considered. The tracking problem was solved in [20] by adding a feedforward controller to the original feedback system. This feedforward controller basically generates the nominal operating point for the feedback controller. Based on the feedforward and feedback structure, an adaptive version of passivity design has been proposed in [16] where it has been shown theoretically that joint position and velocity tracking errors converge to zero, and all the signals are bounded. The advantage of the approach in [16] is that the joint flexibility value is not assumed to be known *a priori*.

In another recent approach proposed in [3][15], a two-stage controller was designed consisting of a link controller and a motor controller similar to the feedforward and feedback control structure in the passivity design approach.

The above mentioned control schemes for flexible-joint manipulators are all based in joint space. In order to solve the redundancy resolution problem effectively for redundant flexible-joint manipulators, we need to address the problem of Cartesian control of flexible-joint manipulators. We start with a joint-space control scheme for non-redundant rigid-joint manipulator. Based on this scheme, we construct an equivalent Cartesian-space scheme. We then propose a hybrid Cartesian-joint controller for redundant flexible-joint manipulators.

5.3 JOINT CONTROL OF NON-REDUNDANT FLEXIBLE-JOINT MANIPULATORS

Consider the simplified flexible-joint manipulator dynamic model (3.5.2) proposed in Chapter 3

$$\begin{bmatrix} D_l & 0 \\ 0 & D_m \end{bmatrix}\begin{bmatrix} \ddot{q}_l \\ \ddot{q}_m \end{bmatrix} + \begin{bmatrix} C_l \\ 0 \end{bmatrix} + \begin{bmatrix} G_l \\ 0 \end{bmatrix} + \begin{bmatrix} K_s(q_l - q_m) \\ -K_s(q_l - q_m) \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} \qquad (5.3.1)$$

The link dynamic equation can be rewritten as

$$D_l\ddot{q}_l + C_l + G_l + Kq_l = K_s(q_{md} - e_m) \qquad (5.3.2)$$

where $e_m \in \Re^{n\times 1}$ is the motor error vector defined by $e_m = q_{md} - q_m$. Extending the resolved acceleration control strategy for a rigid-joint manipulator [17] to the flexible joint case, the desired motor position, q_{md}, is given by

$$q_{md} = K_s^{-1}\{D_l[\ddot{q}_{ld} + K_{vl}\dot{e}_l + K_{pl}e_l] + C_l + G_l + K_s q_l\} \qquad (5.3.3)$$

where $e_l = (q_{ld} - q_l) \in \Re^{n\times 1}$ and $\dot{e}_l \in \Re^{n\times 1}$ represent the link position and velocity error vectors; and $K_{pl} \in \Re^{n\times n}$ and $K_{vl} \in \Re^{n\times n}$ are the constant link position and velocity feedback gain matrices respectively. Actually, we can treat equation (5.3.3) as a *feed forward signal* which provides the nominal motor position trajectory in terms of the desired link trajectory as well as link dynamics. This feedforward signal is somewhat different from those in [3][15], and results in different closed-loop stability properties.

Once the feedforward nominal motor position is obtained, the motor controller can be constructed based on the motor dynamic equation in (5.3.1):

$$\tau_c = D_m\{(\dot{q}_{md} + K_{vm}\dot{e}_m + K_{pm}e_m) - K_s(q_l - q_m) \qquad (5.3.4)$$

where $\tau_c \in \Re^{n\times 1}$ is the motor control torque, and $K_{pm} \in \Re^{n\times n}$ and $K_{vm} \in \Re^{n\times n}$ are constant feedback motor position and velocity gain matrices respectively. Equation (5.3.4) can be regarded as the representation of a *feedback controller*. Therefore, the overall joint level controller is composed of feedforward and feedback controllers.

For the feedforward controller, we substitute (5.3.3) into (5.3.2). The closed-loop equation at the link level is given by

$$\ddot{e}_l + K_{vl}\dot{e}_l + K_{pl}e_l = D_l^{-1}K_s e_m \tag{5.3.5}$$

The matrices K_{pl} and K_{vl} can be specified so as to ensure that the left-hand-side of equation (5.3.5) has characteristic roots in the left-half of the complex plane. But the link tracking error e_l approaching zero asymptotically depends not only on the motor tracking error e_m approaching zero asymptotically on the right-hand-side of (5.3.5), but also on the magnitude of the product $D_l^{-1}K_s$. However, we note that the link inertia matrix $D_l(q_l)$ is positive-definite, and can be bounded as $\underline{D}_l \le D_l(q_l) \le \overline{D}_l$ corresponding to any given q_l within the manipulator's workspace [6]. Therefore, its inverse D_l^{-1} is also bounded as $D_l^{-1} \le D_l^{-1}(q_l) \le \underline{D}_l^{-1}$. The matrix K_s, which is a diagonal matrix of the spring constants, is clearly bounded. This implies that the product of D_l^{-1} and K_s is bounded.

Substituting (5.3.4) into the motor dynamic equation in (5.3.1), we obtain the motor tracking closed-loop equation

$$\ddot{e}_m + K_{vm}\dot{e}_m + K_{pm}e_m = 0 \tag{5.3.6}$$

By properly selecting the motor gain matrices K_{pm} and K_{vm}, we can ensure that e_m as well as $\dot{e}_m$ approach zero asymptotically. To prove stability of the system, a state-space equation is formed based on the combination of the closed-loop equations (5.3.5) and (5.3.6). Knowing that $D_l^{-1}K_s$ is bounded, stability of the system can be proven using the

Lyapunov approach. Details of the proof can be found in [3][15]

The joint-based controller shown above is similar to the schemes in [3][15] where the intermediate variable $z = q_m - q_l$ is defined as the system state instead of q_m. However, it appears that using q_m rather than z requires less computation.

5.4 CARTESIAN CONTROL OF NON-REDUNDANT FLEXIBLE-JOINT MANIPULATORS

For rigid-joint non-redundant manipulators, a Cartesian controller can be easily derived from an equivalent joint-space one. The connection between the Cartesian control force F and the joint control torque τ_c is usually obtained using J_e, e.g. $\tau_c = J_e^T F$ [5] However, little attention has been paid to Cartesian controllers for flexible-joint manipulators. The reason for this is that the control of joint elasticity can be directly done in joint space. Since flexibilty exists between a manipulator's motors and links, it is much more difficult for the controller at the motor side to indirectly control the link position A Cartesian control scheme goes one step further in that the controllers at the motor side indirectly control the end-effector of the manipulator through appropriate control of the link position. A joint-based control scheme, however, prevents us from applying more sophisticated control scheme to a flexible-joint manipulator, such as hybrid control, impedance control, and redundant manipulator control which are achieved using Cartesian space control.

We first develop a Cartesian control scheme for flexible-joint non-redundant manipulators based on the joint-space scheme described in the previous section The controller can be written as

$$q_{md} = K_s^{-1}\{D_l J_e^{-1}[\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e\dot{q}_l] + C_l + G_l + K_s q_l\} \qquad (5.4.1)$$

where $e_x = (x_d - x) \in \Re^{m \times 1}$ and $\dot{e}_x \in \Re^{m \times 1}$ are the Cartesian position and velocity tracking error vectors, x_d and $\dot{x}_d$ are the desired Cartesian position and velocity vectors, while x and $\dot{x}$ are the actual Cartesian position and velocity vectors respectively; $\beta_{pl} \in \Re^{m \times m}$ and $\beta_{vl} \in \Re^{m \times m}$ denote the Cartesian position and velocity gain matrices. Assuming that the manipulator configuration is not in the neighborhood of a singularity, the closed-loop system can be formed by substituting equation (5.4.1) into the link dynamic equation in (5.3.1)

$$\ddot{e}_x + \beta_{vl}\dot{e}_x + \beta_{pl}e_x = J_e D_l^{-1} K_s e_m \tag{5.4.2}$$

Again, by adjusting the gain matrices β_{pl} and β_{vl}, we can ensure that the left-hand-side of equation (5.4.2) has a characteristic polynomial that is Hurwitz. It is also noted that the $J_e D_l^{-1} K_s$ is bounded. A proof of stability for this control scheme will be given later on in this chapter

Remark: Since this is the non-redundant case, the *differential* mapping between joint space and Cartesian space is one to one (after an appropriate kinematic branch has been selected). At the motor level, there is a motor tracking controller which already takes into account the link dynamics. This controller ensures that the motor follows the computed feedforward signal q_{md}, while at the link level the unique differential inverse kinematic solution $\ddot{q}_{ld} = J_e^{-1}(\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e\dot{q}_l)$ transforms Cartesian motion into corresponding joint motions. Therefore, asymptotic Cartesian tracking guarantees asymptotic joint tracking due to the strict kinematic relationship. However, we will see in the next section that this is not always true for redundant flexible-joint manipulators.

5.5 CARTESIAN CONTROL OF REDUNDANT FLEXIBLE-JOINT MANIPULATORS

The control of redundant manipulators falls into two categories. The first category features separate kinematic and dynamic processes. The desired Cartesian trajectory is converted into joint trajectories using an appropriately designed kinematic control scheme, and at the same time redundancy is resolved for the desired joint self-motion [4][29][30]. The generated joint trajectories are then fed into a joint space control scheme On the other hand, the second category combines the kinematics as well as the dynamics of the manipulator. The controller is established in Cartesian space, and the inverse kinematics and redundancy resolution problems are solved implicitly in the process of dynamic control [11][19]. In this Chapter, we will use the latter approach to develop a Cartesian space based redundant manipulator controller.

We first recall the resolved-acceleration pseudo-inverse control for a rigid-joint redundant manipulator:

$$\tau_c = D_l J_e^{\dagger} (\ddot{x}_d + \beta_{vl} \dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l) + D_l (I - J_e^{\dagger} J_e) \xi + C_l + G_l \tag{5.5.1}$$

Notice that the first term on the right-hand side of (5.5.1) is the Cartesian trajectory tracking torque which lies in the range of J_e, and the second term represents the joint self motion torque which lies in the null space of J_e. If we apply this controller to a flexible-joint redundant manipulator, equation (5.4.1) becomes

$$\begin{aligned} q_{md} = K_s^{-1} \{ & D_l J_e^{\dagger} (\ddot{x}_d + \beta_{vl} \dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l) \\ & + D_l (I - J_e^{\dagger} J_e) \xi + C_l + G_l + K_s q_l \} \end{aligned} \tag{5.5.2}$$

Together with the motor controller (5.3.4), this feedforward controller can be applied to the flexible joint manipulator. However, in some cases this controller causes the manipulator's joint configuration to deviate from its designated path, which may also result in deviation of the manipulator's end-effector from its Cartesian trajectory. This could eventually lead to instability. This situation is best illustrated by an example.

The three-link planar flexible joint manipulator is required to track a Cartesian trajectory, and the redundancy resolution is set to minimize the joint acceleration $\|\ddot{q}_l\|$. Figure

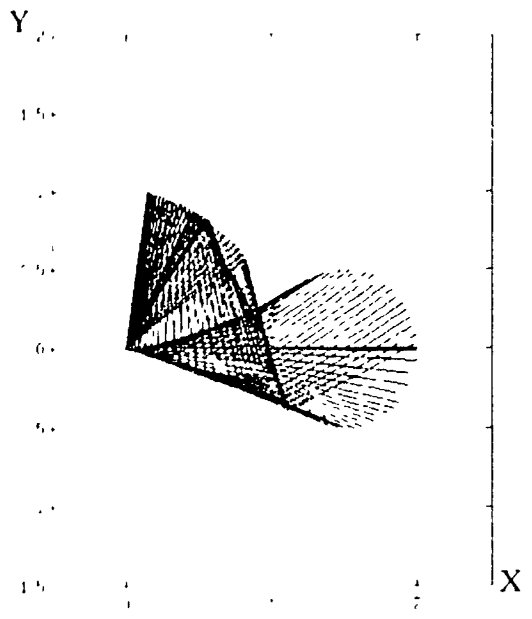

Figure 5 1 **Desired configuration history**

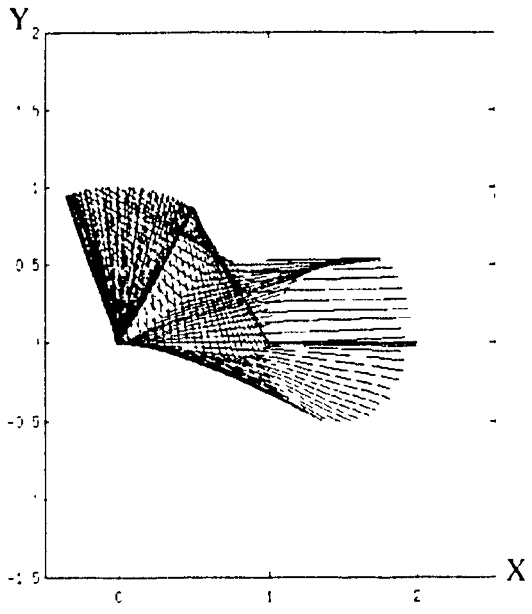

Figure 5.2 **Actual configuration history**

5.1 shows the desired joint configuration history while the tip of the manipulator follows a circular trajectory. In Figure 5.2, equations (5.3.4) and (5.5.2) are used as the motor and link controllers respectively. By contrast with Figure 5.1, it is obvious that in Figure 5.2

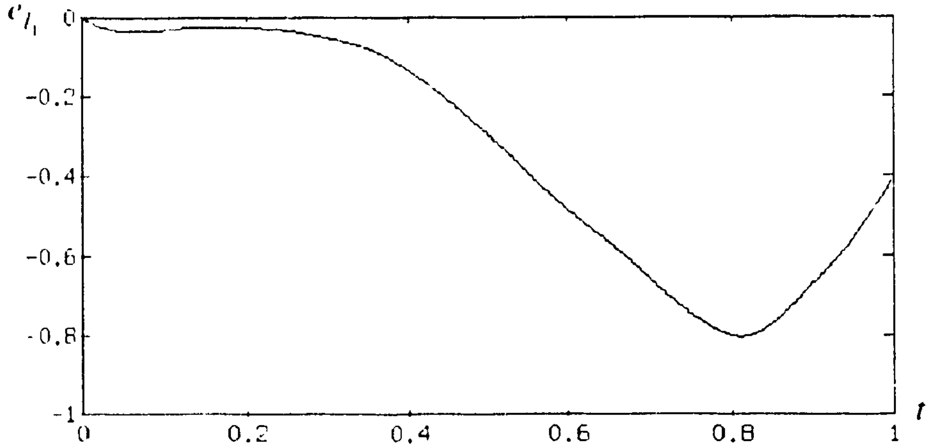

Figure 5 3 **The first link tracking error** the manipulator's joint configuration deviates from its desired path when it is in the process of completing the upper half circle. This deviation causes the tip of the manipulator to move away from its nominal Cartesian trajectory. Figures 5.3, 5.4, and 5.5 show the link tracking errors. It can be seen that the links are actually not tracking the desired trajectories. This is a fairly typical example. In the next section, we will analyze this phenomenon

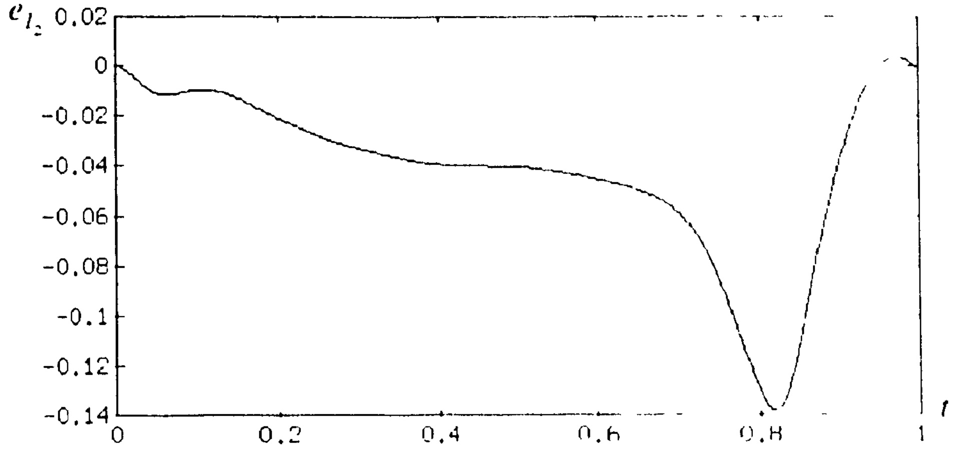

Figure 5.4 **The second link tracking error**

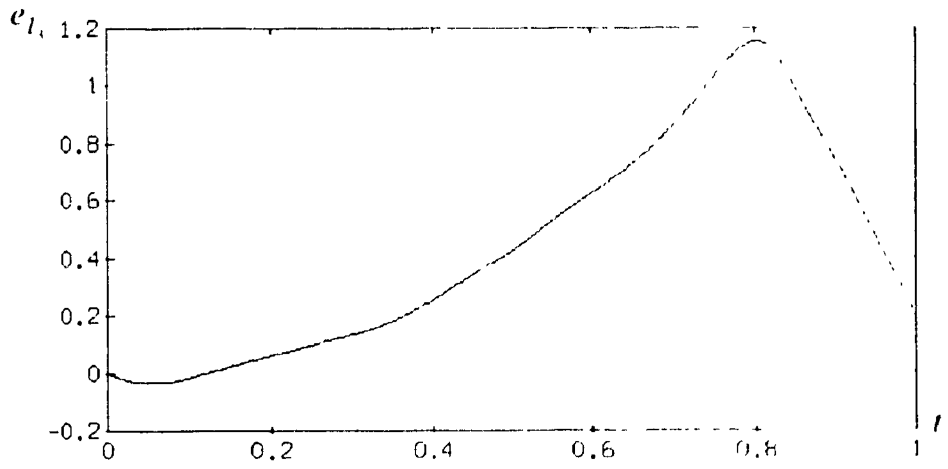

Figure 5 5 **The third link tracking error**

5.5.1 Disturbances Due to Joint Flexibility

For a redundant flexible-joint manipulator, there are two points worth noting. One concerns the torsional forces existing in the flexible joints, and the other is the redundant manipulator's self-motion which is not controlled by the feedback loop.

Torsional forces play an important role in flexible-joint manipulators. On the one hand, torsional forces are affected by link movements; on the other hand, control torques are transmitted from motors and generate torsional forces in order to control the motion of the links. These forces are highly nonlinear and vary in a wide range. It can be assumed that joint flexibility behaves as a spring which stores and releases energy, providing significant disturbances to the system. Furthermore, joint flexibility in one joint has an effect on the other joints, which makes it even more difficult to control the links. Another issue is that because of the differential kinematic one-to-one mapping for non-redundant manipulators, closed-loop control of the end-effector can be translated into closed-loop link/joint control (see Figure 5.6). However, because of self-motion in redundant manipulators, end-effector Cartesian tracking control and motor control do not necessarily translate into closed-loop link control. Usually, we introduce a vector, for example ξ, for the purpose of elbow control, obstacle avoidance, etc., or we optimize some kinematic and dynamic criteria by defining a function $f(q_l) \in \Re^{n \times 1}$, and then choosing ξ in the negative gradient direction $\xi = -\nabla f(q_l)$. The manipulator then adjusts its link configuration and gradually approaches the optimal point while the end-effector tracks a desired Cartesian trajectory. But we note that the null space or self-motion vector ξ is not inside the feedback loop (see

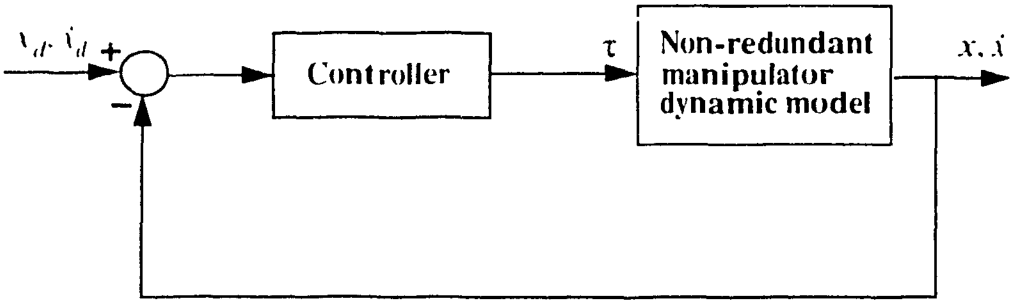

Figure 5.6 **Non-redundant manipulator control block diagram**

Figure 5.7), and its convergence depends only on the negative gradient.

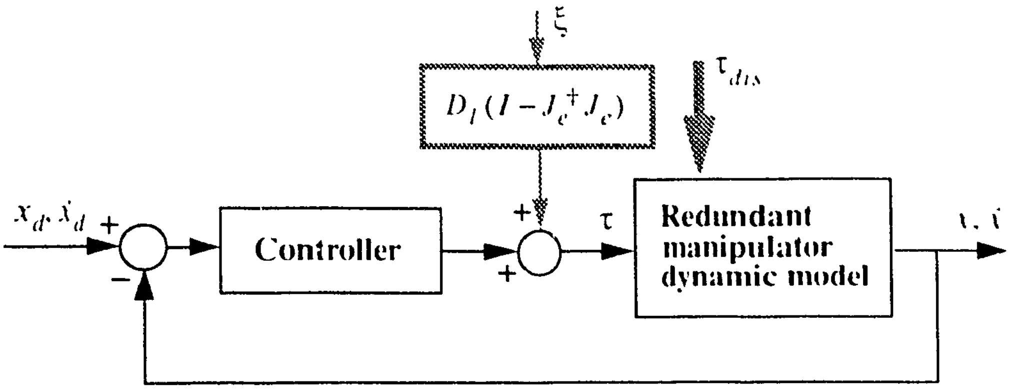

Figure 5.7 **Redundant manipulator control block diagram**

Based on the analysis above, it is clear that a disturbance $\tau_{dis} \in \Re^{n \times 1}$ is introduced in the system by joint flexibility

$$q_{md} = K_s^{-1} \{ D_l J_e^+ (\ddot{x}_d + \beta_{vl} \dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l) + D_l (I - J_e^+ J_e) \xi$$

$$+ C_l + G_l + K_s q_l + \tau_{dis} \} \qquad (5.5.3)$$

Theoretically, τ_{dis} could affect the Cartesian tracking variables x and $\dot{x}$, or through q_{md}, influence the motor tracking variables $q_m$and $\dot{q}_m$. However, these variables are inside the feedback loops of the Cartesian tracking and the motor tracking controllers respectively. The sensitivity function of Cartesian tracking with respect to τ_{dis} is usually proportional to the inverse of the gain matrices β_{pl} and β_{vl}. Hence, the tracking errors due to τ_{dis} can be made arbitrarily small by making β_{pl} and β_{vl} sufficiently large. A similar procedure can be used for the motor controller to make motor tracking errors as small as possible. However, the sensitivity function of the self-motion ξ with respect to τ_{dis} can be large because ξ is outside the feedback loop. Therefore, the influence of τ_{dis} on the self-motion torque

$D_l(I - J_e^{\dagger}J_e)\xi$ can become significant. Two possibilities can be observed:

(i) If $\|\tau_{dis}\| \ll \|D_l(I - J_e^{\dagger}J_e)[-\nabla f(q_l)]\|$, the disturbance τ_{dis} is relatively small and it affects the self-motion torque, but it does not prevent ξ from approaching the optimal point. This implies that using joint self-motion to reach the optimal point is still feasible but it may take a longer time.

(ii) If $\|\tau_{dis}\| \approx \|D_l(I - J_e^{\dagger}J_e)[-\nabla f(q_l)]\|$, the joint torsional force disturbance τ_{dis} is large enough to affect the magnitude as well as the direction of the self-motion torque. In other words, τ_{dis} can prevent ξ from approaching its optimal point. The manipulator in this case may lose the desired redundancy resolution. In the worst case scenario, this loss of redundancy resolution means loss of link control and may even result in instability.

5.5.2 Hybrid Cartesian-Joint Control

The above analysis indicates that we must either add another controller to the self-motion of the manipulator, or somehow bring the self-motion vector ξ into the feedback loop such that all the link motions can be controlled. Unfortunately, the desired link trajectories q_{ld}, $\dot{q}_{ld}$ and $\ddot{q}_{ld}$ are not directly available for constructing a tracking system for the links. However, it is possible to compute the desired link trajectories indirectly. Therefore, we propose a *hybrid Cartesian-joint controller* for flexible-joint redundant manipulators.

Using differential inverse kinematics for a redundant manipulator, the acceleration $\ddot{q}_{ld}$ can be expressed as

$$\ddot{q}_{ld} = J_e^{\dagger}(\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e\dot{q}_{ld}) + (I - J_e^{\dagger}J_e)\xi \tag{5.5.4}$$

Since the initial desired link position $q_{ld}(0)$ and velocity $\dot{q}_{ld}(0)$ are known, we can compute the desired future link acceleration using equation (5.5.4), and from where the desired future velocity $\dot{q}_{ld}$ and the position q_{ld} can be obtained by integration.

If all the link motions, resulting from Cartesian tracking and link self-motion, are controlled by feedback, this could prevent τ_{dls} from affecting link self-motion. In addition to the Cartesian tracking control, this controller reinforces link control. The structure of the controller is given by

$$q_{md} = K_s^{-1} \{ D_l J_e^{\dagger} (\ddot{x}_d + \beta_{vl} \dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l) + D_l (I - J_e^{\dagger} J_e) \xi$$

$$+ D_l [\kappa_v (\dot{q}_{ld} - \dot{q}_l) + \kappa_p (q_{ld} - q_l)] + C_l + G_l + K_s q_l \} \qquad (5.5.5)$$

where $\kappa_p \in \Re^{n \times n}$ and $\kappa_v \in \Re^{n \times n}$ are link position and velocity gain matrices, and q_{ld} and $\dot{q}_{ld}$ are the desired joint position and velocity variables. To control a flexible-joint redundant manipulator, we need Cartesian tracking control as well as joint control. This can be achieved using (5.5.5). In addition to this, the motor controller that takes the same structure as the one for non-redundant manipulator joint control, i.e., equation (5.3.4), is also used to control the motor dynamics. The properties of the closed-loop system resulting from applying the controllers (5.5.5) and (5.3.4) to the system (5.3.1) can be analyzed as follows.

It should be noted that the overall system has three controllers. From the outer-loop to the inner-loop, it has a Cartesian tracking controller, a link-level controller and a motor level controller. Let us start with the Cartesian tracking controller. Based on the differential inverse kinematics and equation (5.5.5), the joint acceleration can be expressed as

$$\ddot{q}^{\circ}_{ld} = J_e^{\dagger} (\ddot{x}_d + \beta_{vl} \dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l) + (I - J_e^{\dagger} J_e) \xi \qquad (5.5.6)$$

Premultiplying by J_e on both sides of (5.5.6), and using the property $J_e(I - J_e^{\dagger}J_e) = 0$, we can express the Cartesian tracking closed-loop equation as

$$\ddot{e}_x + \beta_{vl}\dot{e}_x + \beta_{pl}e_x = 0 \tag{5.5.7}$$

where asymptotic stability can be ensured by proper choice of β_{pl} and β_{vl}. For the link-level controller, express (5.5.5) based on (5.5.6), we get

$$q_{md} = K_s^{-1}\{D_l[\ddot{q}^{o}_{ld} + \kappa_v(\dot{q}_{ld} - \dot{q}_l) + \kappa_p(q_{ld} - q_l)\ ] + C_l + G_l + Kq_l\ \} \tag{5.5.8}$$

Note that equation (5.5.8) denotes a joint space controller. The closed-loop characteristic equation can be formed by combining the controller (5.5.8) and the link dynamic equation in (5.3.1)

$$\ddot{e}_l + \kappa_v\dot{e}_l + \kappa_p e_l = D_l^{-1}K_s e_m \tag{5.5.9}$$

5.6 STABILITY ANALYSIS

System stability can be shown using the Lyapunov function approach. First, note that for equation (5.5.7) the Cartesian tracking errors e_x and $\dot{e}_x$ are not coupled with other variables, its stability can be ensured by proper choices of the gain matrices β_{pl} and β_{vl} independently. Second, for the two sets of closed-loop characteristic equations shown in (5.5.9) and (5.3.6) for link tracking and motor tracking, the equivalent state-space representation can be written as

$$\dot{Y} = AY \tag{5.6.1}$$

where $Y \in \Re^{4n \times 1}$ is defined as $Y = \begin{bmatrix} e_l^T & \dot{e}_l^T & e_m^T & \dot{e}_m^T \end{bmatrix}^T$, and $A \in \Re^{4n \times 4n}$ can be written as

$$A(q_l) = \begin{bmatrix} 0 & I_n & 0 & 0 \\ -K_p & -K_v & -D_l^{-1}(q_l) K_s & 0 \\ 0 & 0 & 0 & I_n \\ 0 & 0 & -K_{pm} & -K_{vm} \end{bmatrix} \tag{5.6.2}$$

Moreover, for the purpose of analysis, we partition Y and $A(q_l)$ as

$$A(q_l) = \begin{bmatrix} A_{11} & A_{12}(q_l) \\ 0 & A_{22} \end{bmatrix} \quad \text{and} \quad Y = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \tag{5.6.3}$$

We note that the system matrix $A(q_l)$ is time-varying because the submatrix $A_{12}(q_l)$ in $A(q_l)$ is a function of the manipulator inertia matrix. Of course, this time-varying property of the matrix $A(q_l)$ makes stability analysis more difficult since the standard approaches for analyzing linear time-invariant systems cannot be used [22]. Thus, it is necessary to consider the application of Lyapunov's direct method for studying the stability of this nonlinear time-varying system.

The main idea in proving stability of (5.6.1) is to derive a simple scheme which is similar to applying Lyapunov's direct method to linear time-invariant systems. In the case of linear time-invariant systems, a necessary and sufficient condition for $\dot{x}_o = A_o x_o$ to be strictly stable is that, for any symmetric positive-definite matrix Q_o, the unique solution P_o of the Lyapunov equation $A_o^T P_o + P_o A_o = -Q_o$ must be positive-definite. In the case of a time-varying system such as the system in (5.6.1), some modifications must be made to accommodate the time-varying property when we apply Lyapunov's direct method

The following Lemma is required in the proof of stability of the system in (5.6.1). Its proof is given in [14].

Lemma 5.1: Let matrix $Z \in \Re^{(n_1+m_1) \times (n_1+m_1)}$ have the structure

$$Z = \begin{bmatrix} H & R \\ R^T & S \end{bmatrix} \tag{5.6.4}$$

where $H \in \Re^{n_1 \times n_1}$ and $S \in \Re^{m_1 \times m_1}$ are symmetric positive-definite matrices. Then the matrix Z is positive-definite if

$$\lambda_{min}(H)\,\lambda_{min}(S) > \|R\|^2 \tag{5.6.5}$$

and

$$\lambda_{min}(Z_{min})\,I_{n_1+m_1} \le Z \le \lambda_{max}(Z_{max})\,I_{n_1+m_1} \tag{5.6.6}$$

where matrices $Z_{min} \in \Re^{2\times 2}$ and $Z_{max} \in \Re^{2\times 2}$ are

$$Z_{min} = \begin{bmatrix} \lambda_{min}(H) & \|R\| \\ \|R\| & \lambda_{min}(S) \end{bmatrix} \tag{5.6.7}$$

$$Z_{max} = \begin{bmatrix} \lambda_{max}(H) & \|R\| \\ \|R\| & \lambda_{max}(S) \end{bmatrix} \tag{5.6.8}$$

and $\|R\|$ denotes the 2-norm of R.

Now consider the system (5.6.1) with matrix A defined in (5.6.2), we shall show the stability of the system using an approach similar to that in [3][15].

Consider the equation

$$A^T P + PA = -Q \tag{5.6.9}$$

where the matrices P and Q are partitioned into four $2n \times 2n$ submatrices as

$$P = \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \tag{5.6.10}$$

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^{T} & Q_{22} \end{bmatrix} \tag{5.6.11}$$

To show that there exist P and Q which are uniformly positive-definite, we substitute A in (5.6.3), (5.6.10) and (5.6.11) into (5.6.9), and express the Lyapunov equation as

$$A_{11}^{T}P_{11} + P_{11}A_{11} = -Q_{11} \tag{5.6.12}$$

$$P_{11}A_{12} = -Q_{12} \tag{5.6.13}$$

$$A_{22}^{T}P_{22} + P_{22}A_{22} = -Q_{22} \tag{5.6.14}$$

In order to ensure the stability of the systems (5.6.1), a proper procedure to find the matrices P and Q must be followed. First, given any positive definite matrix Q_{11}, the matrix P_{11} can be obtained using (5.6.12). The positive definiteness of P_{11} is guaranteed since A_{11} is a time-invariant asymptotically stable matrix. We can then calculate Q_{12} using (5.6.13). It should be noted that Q_{12} is time-varying because of the time-varying nature of A_{12}. Next, we note that since A_{22} is a time-invariant asymptotically stable matrix, a constant matrix P_{22} can be obtained by choosing a constant positive-definite matrix Q_{22} and solving equation (5.6.14). Since the matrices Q_{11} and Q_{12} have been formed already, the matrix Q_{22} should be selected so as to guarantee uniform positive definiteness of Q. This can be ensured as follows: First from (5.6.13) we get

$$
\begin{aligned}
\|Q_{12}\| &= \|P_{11}A_{12}(q_l)\| \\
&\le \lambda_{max}(P_{11})\,\|A_{12}(q_l)\| \\
&= \lambda_{max}(P_{11})\,\|D_l^{-1}(q_l)K_s\| \\
&\le \lambda_{max}(P_{11})\,\lambda_{max}(K_s)\,\underline{D}_l^{-1} \\
&= \sigma_m
\end{aligned}
\tag{5.6.15}
$$

Then by Lemma 5.1, Q is positive-definite if

$$\lambda_{min}(Q_{22}) > \frac{\|Q_{12}\|^2}{\lambda_{min}(Q_{11})} \tag{5.6.16}$$

Therefore, using the result in (5.6.15), if Q_{22} is chosen such that

$$\lambda_{min}(Q_{22}) > \frac{\sigma_m^2}{\lambda_{min}(Q_{11})} \tag{5.6.17}$$

then, Q will be uniformly positive-definite. Thus, we can always find uniformly positive-definite matrices P and Q such that equation (5.6.9) is satisfied. This implies that the system (5.6.1) is asymptotically stable.

5.7 COMPUTATIONAL CONSIDERATIONS

It is obvious that control of flexible-joint manipulators is much more difficult than control of rigid-joint ones. This difficulty can be illustrated by the amount of computational effort required to form the flexible-joint manipulator controller.

The link-level controller (5.5.5), which provides the desired motor position, is a function of both Cartesian and joint variables

$$q_{md} = q_{md}(x_d, \dot{x}_d, \ddot{x}_d, x, \dot{x}, q_{ld}, \dot{q}_{ld}, q_l, \dot{q}_l) \qquad (5.7.1)$$

It requires only position and velocity information of both Cartesian and joint variables which are both measurable. However, careful examination of the motor-level controller (5.3.4) tells us that not only the desired motor position q_{md} is required, but the desired motor velocity and acceleration are also needed:

$$\dot{q}_{md} = \dot{q}_{md}(x_d, \dot{x}_d, \ddot{x}_d, \dddot{x}_d, x, \dot{x}, \ddot{x}, q_{ld}, \dot{q}_{ld}, \ddot{q}_{ld}, q_l, \dot{q}_l, \ddot{q}_l) \qquad (5.7.2)$$

$$\ddot{q}_{md} = \ddot{q}_{md}(x_d, \dot{x}_d, \ddot{x}_d, \dddot{x}_d, x_d^{(4)}, x, \dot{x}, \ddot{x}, x^{(3)}, q_{ld}, \dot{q}_{ld}, \ddot{q}_{ld}, q_{ld}^{(3)}, q_l, \dot{q}_l, \ddot{q}_l, q_l^{(3)}) \qquad (5.7.3)$$

This means that higher derivatives are required in the feedback loop. However, from a practical point of view, the measurements for higher derivatives are not feasible yet. Even though it is possible somehow to measure (or estimate) the acceleration, the measurement is usually corrupted with noise. Therefore, we must rely on some algorithm(s) to compute these higher derivatives.

In (5.7.2) and (5.7.3), the key point is to evaluate $\ddot{q}_l$ and $q_l^{(3)}$. The other higher derivatives can be computed from $\ddot{q}_l$ and $q_l^{(3)}$. Fortunately, because of the special structure of the flexible joint dynamics, the higher-order derivatives of q_l can be calculated recursively in terms of its lower-order ones. Then, $\ddot{q}_l$ and $q_l^{(3)}$ can be evaluated using

$$\ddot{q}_l = -D_l^{-1}(q_l)\,[C_l(q_l, \dot{q}_l) + G_l(q_l) + K_s(q_l - q_m)] \qquad (5.7.4)$$

$$q_l^{(3)} = -D_l^{-1}(q_l)\,[\dot{D}_l(q_l, \dot{q}_l)\ddot{q}_l + \dot{C}_l(q_l, \dot{q}_l, \ddot{q}_l) + \dot{G}_l(q_l, \dot{q}_l) + K_s(\dot{q}_l - \dot{q}_m)] \qquad (5.7.5)$$

Having obtained $\ddot{q}_l$ and $q_l^{(3)}$, higher order derivatives of x can be calculated using forward kinematics, while higher order derivatives of q_{ld} can be computed using equation (5.5.4) and its derivatives. The higher-order derivatives of the desired Cartesian trajectory x_d will be obtained by directly differentiating the given trajectory up to the 4th derivative to get $\dot{x}_d, \ddot{x}_d, x_d^{(3)}$ and $x_d^{(4)}$.

Another important point from the computational point of view is the integration used to calculate the desired joint position and velocity from (5.5.4). This integration is very difficult to perform analytically because of the complicated nonlinear nature of (5.5.4). Therefore, in general numerical integration is required.

5.8 COMPUTER SIMULATIONS

To demonstrate the applicability of the proposed hybrid Cartesian-joint control strategy for redundant flexible-joint manipulators, we apply the scheme to the three DOF planar redundant flexible-joint manipulator shown in Figure 5.8. Each link is assumed to

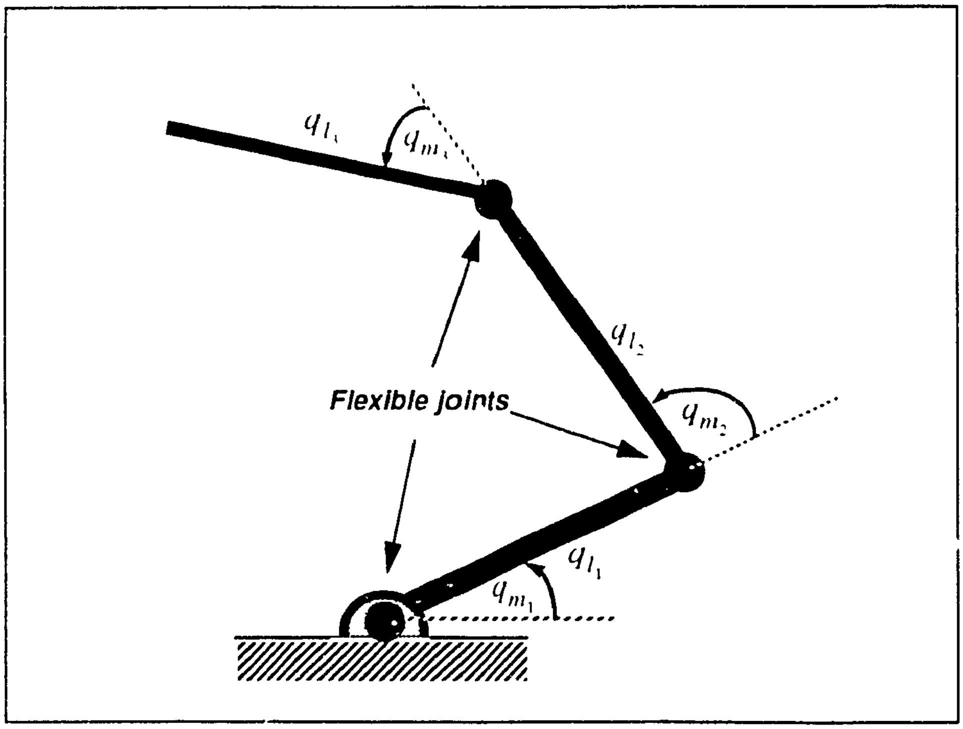

Figure 5 8 **Three-link planar redundant flexible-joint manipulator** have the same length $l_1 = l_2 = l_3 = 1m$, and the same mass $m_1 = m_2 = m_3 = 10kg$. The links are modelled under the assumption of point masses at their distal ends. The motor inertia matrix is assumed to be $D_m = diag\{D_{m_1}, D_{m_2}, D_{m_3}\}$ with $D_{m_1} = D_{m_2} = D_{m_3} = 1kg$. It is also assumed that the joint stiffness constant matrix $K_s = diag\{K_{s1}, K_{s2}, K_{s3}\}$ with $K_{s1} = K_{s2} = K_{s3} = 100Nt$. The dynamic model of the entire manipulator is formed by combining the dynamics of the standard three DOF planar robot manipulator and the joint flexibility. In the simulations, we must solve forward dynamics of the manipulator in order to mimic the dynamic behavior of the real manipulator. Because of the joint flexibility, the differential equations describing the manipulator dynamics become *stiff*. Using a higher-order Runge-Kutta method (7/8th order) in the simulations, the results are satisfactory with respect to the relatively flexible stiffness matrix K_s.

Simulations were performed using MATLAB on a SUN/SPARC-2 workstation. The three controllers, namely the Cartesian tracking controller, the joint tracking controller, and the motor tracking controller, were designed to stabilize the end-effector Cartesian motion, the link motion and the motor motion. For the Cartesian controller, the gain matrices β_{pl} and β_{vl} were selected as $\beta_{pl} = diag\{200, 200\}$ and $\beta_{vl} = diag\{28.5, 28.5\}$ to ensure proper Cartesian tracking. At the same time, the link tracking controller was also designed with its gain matrices $\kappa_p = diag\{150, 150, 150\}$ and $\kappa_v = diag\{24.5, 24.5, 24.5\}$ to effectively control link positions as well as velocities. The motor controller was also set up in a similar way to guarantee that the motors track their "desired" trajectories. Typical values of the gain matrices for . motor controller are $K_{pm} = diag\{100, 100, 100\}$ and $K_{vm} = diag\{10, 10, 10\}$.

In the simulations, various Cartesian trajectories were tested. To show the applicability of the proposed control strategy, two sets of the typical examples from a number of simulations are given in this section. The first set of simulations involves straight line trajectory tracking in Cartesian space, while the second set illustrates the redundant flexible joint manipulator following a circular Cartesian trajectory. Each set of simulations has different redundancy resolution criteria.

For the first set of simulations, minimum acceleration $\|\ddot{q}_l\|$ is selected as the redundancy resolution criterion. Figure 5.9 illustrates the evolution of the redundant manipulator configurations. The manipulator was initially at rest with $q(0) = \begin{bmatrix} 60^o & -120^o & 60^o \end{bmatrix}^T$ which corresponds to $x(0) = \begin{bmatrix} 2 & 0 \end{bmatrix}^T$ in Cartesian space. As show in Figure 5.9, the manipulator end-effector tracks a straight line trajectory and finally reaches the goal point $x(t_f) = \begin{bmatrix} 1.0 & 0.5 \end{bmatrix}^T$. Figures 5.10 and 5.11 show the Cartesian position tracking errors along X axis $(X = x_1)$ and Y axis $(Y = x_2)$ respectively, while Figures 5.12 and 5.13 illustrate the Cartesian velocity tracking errors along X and Y axes respectively. Corresponding to the straight-line motion of the end-effector in Cartesian space, the actual link position and velocity profiles are shown in Figures 5.14-5.19. With respect to the "desired" link positions q_{ld} computed using equation (5.5.4), the link tracking errors are shown in Figures 5.20-5.22, while the motor tracking errors between the desired motor position q_{md} and actual motor position q_m are shown in Figures 5.23-5.25. Finally, the control torque profiles resulting in the straight line motion of the manipulator are plotted in Figures 5.26-5.28.

The second example involves the redundant flexible-joint manipulator tracking a curved Cartesian trajectory. For redundancy resolution, a quadratic function of joint positions was chosen

$$f(q_l) = \sum_i \frac{1}{2} q_i^2 \qquad i = 1, 2, 3 \tag{5.8.1}$$

The redundancy resolution vector ξ can be expressed as $\xi = -\lambda \nabla f(q_l)$, where λ is a constant gain. The evolution of the configurations of the redundant flexible-joint manipulator is shown in Figures 5.29 and 5.30, where the manipulator starts from the configuration $q_l(0) = \left[60^{\circ}\ -120^{\circ}\ 60^{\circ}\right]^T$ at $x(0) = \left[2\ 0\right]^T$, and the end-effector follows the elliptical path clockwise for two complete circles, and finally comes back to the initial point. In order not to overlap the configurations, the evolutions of the configurations for the first and the second circles are illustrated in Figures 5.29 and 5.30 respectively. It can be seen in Figure 5.29 that the actual end-effector motion for the first half of the circle deviates from the desired trajectory because of joint flexibility and initial over-shoot. But after the initial period the tracking errors are reduced to an acceptable level for the rest of the motion. The details of the Cartesian position and velocity tracking errors can be seen in Figures 5.31-5.34, while in Figures 5.35-5.40 the corresponding link position and velocity profiles are shown. To see how good the link controller and the motor controller are, the link tracking errors and the motor tracking errors are illustrated in Figures 5.41 5.43, and in Figures 5.44-5.46 respectively. The control torques transmitted to the actuator in the joints during the tracking of the elliptical trajectory are shown in Figures 5.47-5.49.

5.9 CONCLUDING REMARKS

A novel scheme called *hybrid Cartesian-joint control* for controling a redundant flexible-joint manipulator has been developed in this chapter, In order to develop this scheme, we have studied joint-space control of flexible-joint non-redundant manipulators, Cartesian-space control of flexible-joint non-redundant manipulators, and Cartesian-space control of flexible-joint redundant manipulators. The effect of disturbances due to joint flexibility, and manipulator link self-motion were analyzed. The hybrid Cartesian-joint nonlinear control scheme proposed in this chapter consists of a Cartesian tracking controller, a link tracking controller, and a motor tracking controller. A stability analysis for the proposed control scheme has also been given. The applicability of the proposed hybrid Cartesian-joint control scheme has been illustrated by computer simulations for a three-link planar redundant flexible-joint manipulator. In formulating the controller, we have assumed that the flexible joint stiffness matrix K_s is a constant diagonal matrix. However, the proposed scheme can also be extended to the case where K_s is nonlinear and time-varying.

5.10 REFERENCES

[1] S. Arimoto and F. Miyazaki, "Stability and robustness of PID feedback control for robot manipulators of sensory capability," in *Robotics Research: The First Int. Symp.*, ed. M. Brady and R. Paul (MIT Press), pp. 783-799, 1984.

[2] S. Arimoto and F. Miyazaki, "Stability and robustness of PD feedback control with gravity compensation for robot manipulators," *ASME Winter Meeting*, Anaheim, CA, pp. 67-72, Dec. 1986.

[3] K.P. Chen and L.C. Fu, "Nonlinear adaptive motion control for a manipulator with flexible joints," *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 1201-1206, 1989.

[4] R. Colbaugh, K. Glass and H. Seraji, "An adaptive inverse kinematics algorithm for robot manipulators," in *Proc. of American Control Conf.*, San Diego, CA., pp. 2281-2286, 1990.

[5] J.J. Craig, *Introduction to Robotics: Mechanics and Control*, Addison Wesley, 1986.

[6] J.J. Craig, *Adaptive Control of Mechanical Manipulators*, Addison Wesley, 1988.

[7] J.J. Craig, P. Hsu and S. Sastry, "Adaptive control of mechanical manipulators," *Int. J. Robotics. Res.*, vol. 6, no. 2, pp. 16-28, 1987.

[8] A. De Luca, A. Isidori and F. Nicolo, "Control of a robot arm with elastic joints via nonlinear dynamic feedback," *Proc. 24th IEEE Control and Decision Conf.*, pp.

1671-1679, 1985.

[9] A. De Luca, "Dynamic control properties of robot arms with joint elasticity," *Proc IEEE Int. Conf. on Robotics and Automation*, pp. 1574-1580, Philadelphia, PA., 1988.

[10] E. Freund, "Fast nonlinear control with arbitrary pole-placement for industrial obots and manipulators," *Int. J. Robotics Res.*, vol. 1, no. 1, pp. 65-78, 1982.

[11] P. Hsu, J. Hauser and S. Sastry, "Dynamic control of redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automat.*, Philadephia, PA., pp. 183-187, 1988.

[12] O. Khatib, "A unified approach for motion and force control of robot manipulators," *IEEE J. Robotics Automat.*, vol. RA-3, pp. 43-53, 1987.

[13] K. Khorasani, "Adaptive control of flexible-joint robots," *IEEE Trans. Robotics Automat.*, vol. 8, pp. 250-267, April 1992.

[14] D.E. Koditschek, "Quadratic Lyapunov functions for mechanical systems," *Electrical Engineering Dept., Technical Report*, no. 7803, Yale University, March 1987

[15] K.Y. Lian, J.H. Jean and L.C. Fu, "Adaptive force control of single-link mechanism with joint flexibility," *IEEE Trans Robotics. Automat.*, vol 7, pp 540 545, Aug. 1991.

[16] R Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Trans. Automat. Contr*, vol. 37, pp. 174-181, Feb. 1992.

[17] J.Y.S. Luh, M.W. Walker and R.P. Paul, "Resolved-acceleration control of mechanical manipulators," *IEEE Trans. Automat. Contr.*, vol. AC-25, no. 3, 1980.

[18] P.J. Moylan, "Implications of passivity in a class of nonlinear systems," *IEEE Trans. on Automat. Control*, vol. 19, pp. 373-381, Aug. 1974.

[19] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robotics Automat.*, vol. 5, pp. 472-490, 1989.

[20] P. Sicard and J. T. Wen, "Application of a passivity based control methodology for flexible joint robots to a simplified Space Shuttle RMS," *1992 Proc American*

Contr. Conf., pp. 1690-1694, 1992.

[21] J.J. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. Robotics. Res.*, vol. 6, no. 3, pp. 49-59, 1987.

[22] J.J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, NJ., 1991.

[23] J.J. Slotine and S.S. Sastry, "Tracking control of nonlinear systems using sliding surfaces with application to robot manipulators," *Int. J. Contr.*, vol. 38, pp. 465-492, 1983.

[24] M.W. Spong, "Modeling and control of elastic joint robots," *ASME J. Dyn. Syst. Meas. Contr.*, vol. 109, pp. 310-319, 1987.

[25] M.W. Spong, "Control of flexible joint robots: A survey," *Coordinated Science Laboratory, Technical Report*, UILU-ENG-90-2203 DC-116, University of Illinois at Urbana-Champaign, Feb. 1990.

[26] M.W. Spong, K. Khorasani and P.V. Kokotovic, "An integral manifold approach to the feedback control of flexible joint robots," *IEEE J. Robotics. Automat.*, vol. RA-3, pp. 291-300, Aug. 1987.

[27] M. Takegaki and S. Arimoto, "A new feedback method for dynamic control of manipulators," *ASME J. Dyn. Syst. Meas. and Cont.*, vol. 102, June 1981.

[28] P. Tomei, "A simple PD contioller for robots with elastic joints," *IEEE Trans. Automat. Contr.*, vol. 36, pp. 1208-1213, Oct. 1991.

[29] R.J. Vaccaro and S.D. Hill, "A joint-space command generator for Cartesian control of robotic .nanipulators," *IEEE J. Robotics Automat.*, vol. 4, pp. 70-76, 1988.

[30] W.A. Wolovich and H. Elliot, "A computational technique for inverse kinematics," in *Proc. 23rd IEEE Conf. on Decision and Control*, Las Vegas, NV., pp. 1359-1363, 1984.

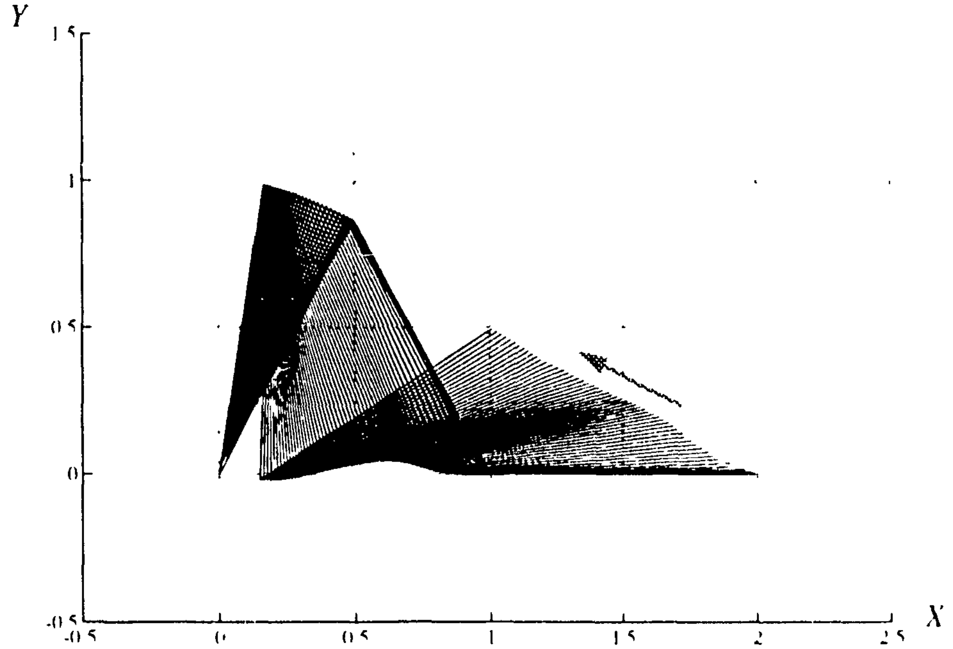

Figure 5.9 **Straight line tracking**

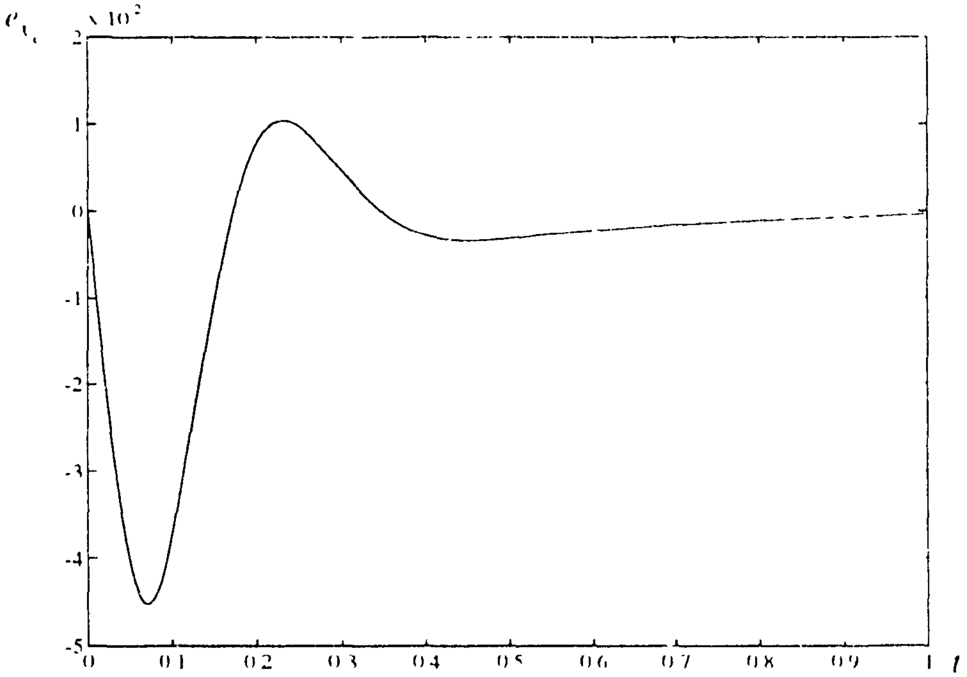

Figure 5.10. **Position tracking error in X direction**

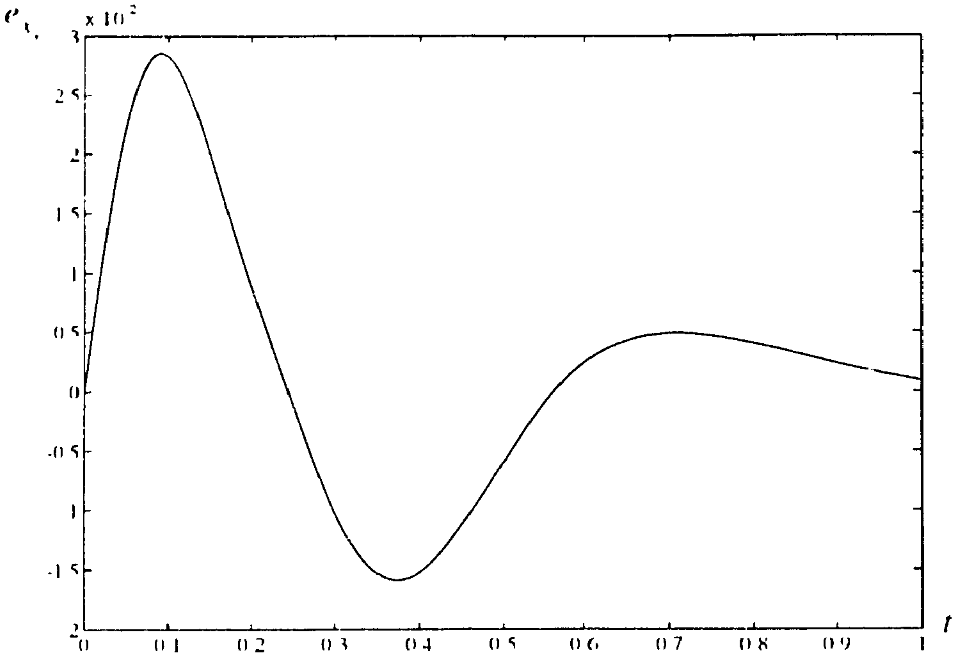

Figure 5 11 **Position tracking error in Y direction**

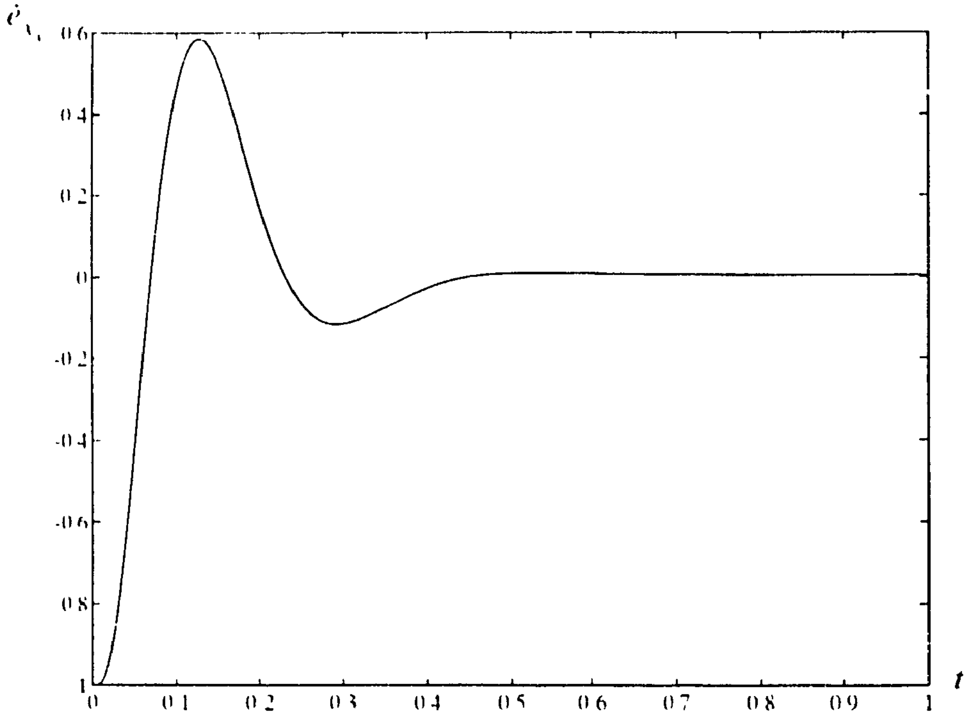

Figure 5.12 **Velocity tracking error in X direction**

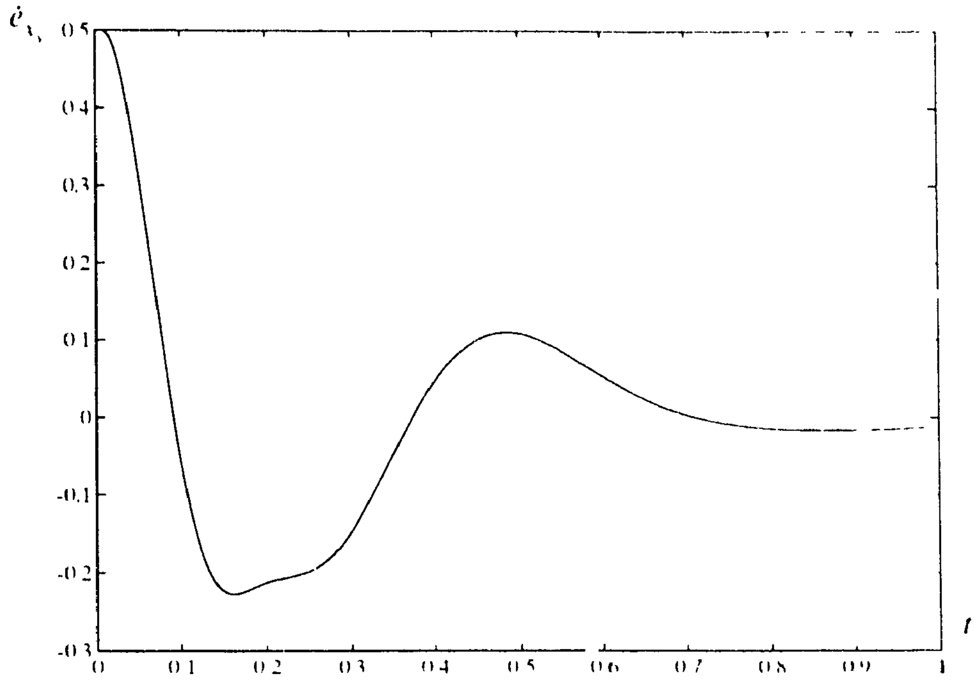

Figure 5.13 Velocity tracking error in Y direction

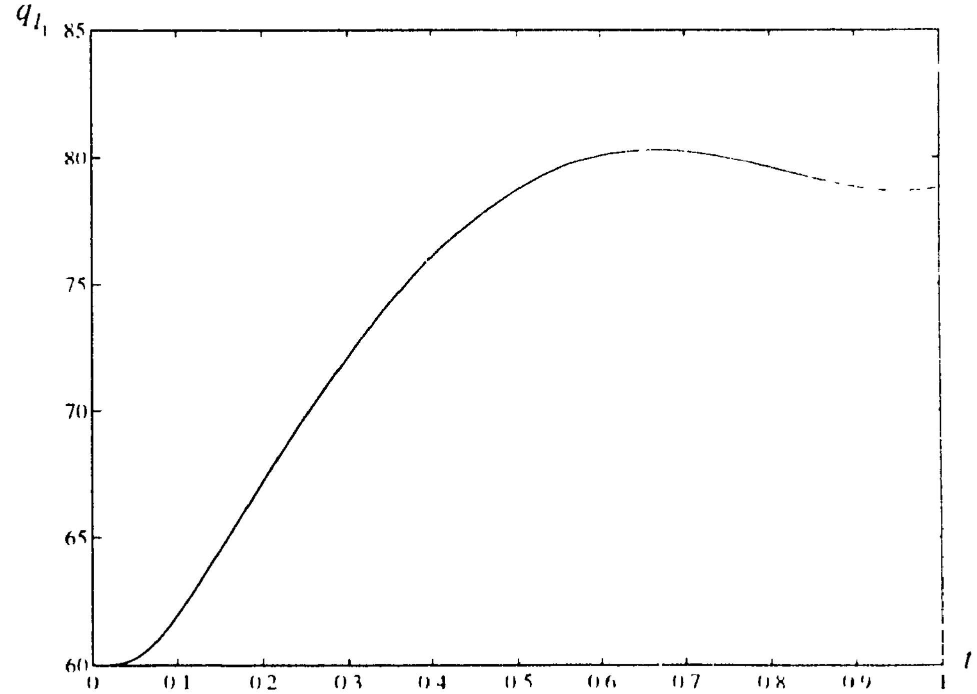

Figure 5 14 The actual trajectory for the first link

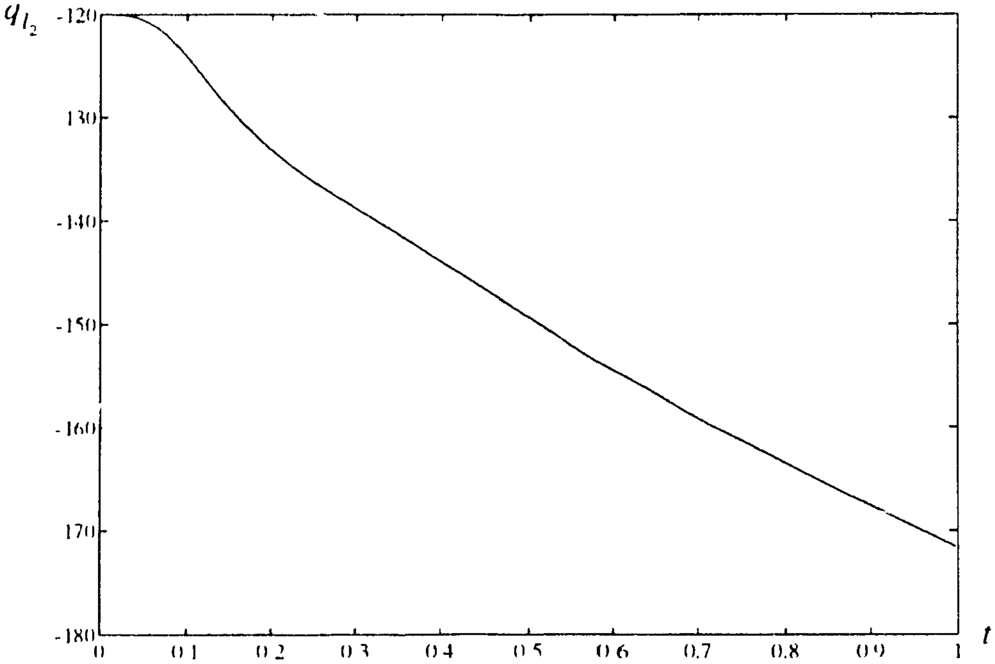

Figure 5.15 **The actual trajectory for the second link**

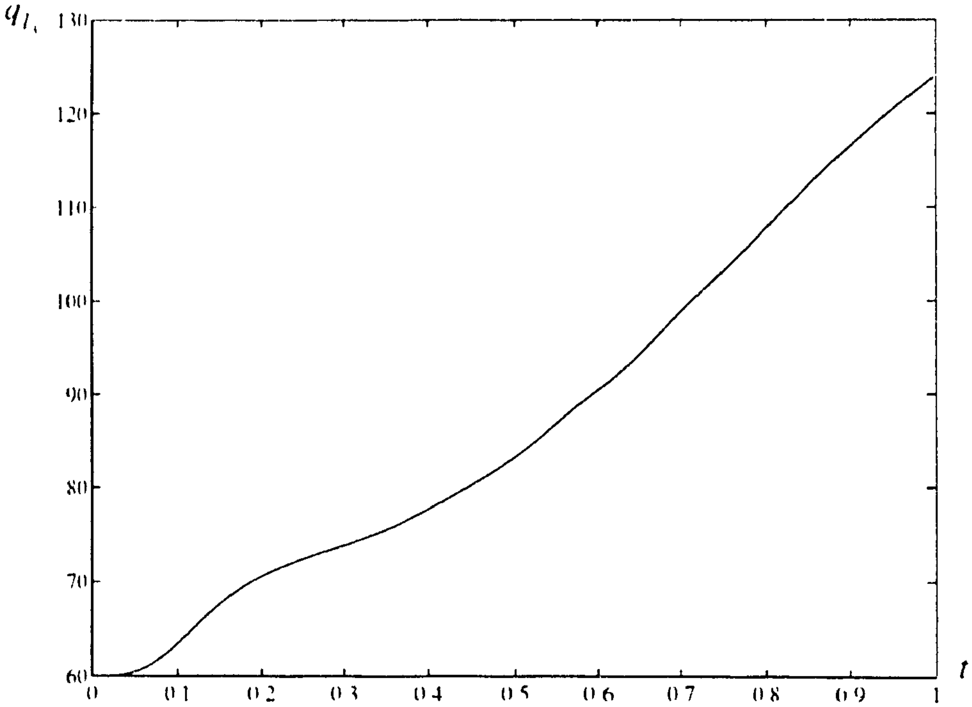

Figure 5 16 **The actual trajectory for the third link**

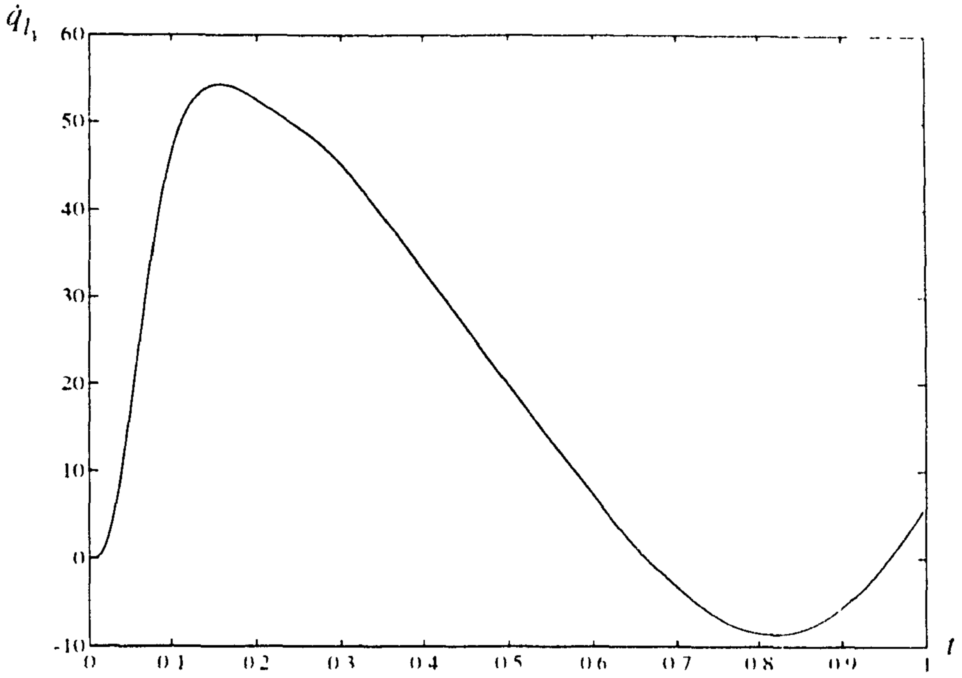

Figure 5.17 **The actual velocity trajectory for the first link**

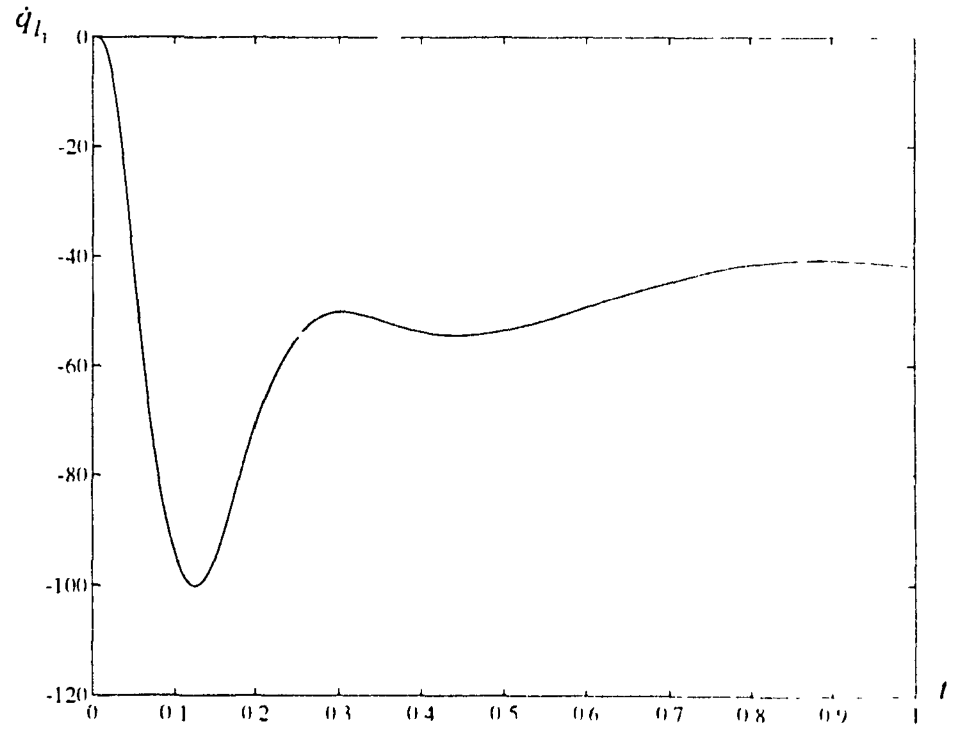

Figure 5.18 **The actual velocity trajectory for the second link**

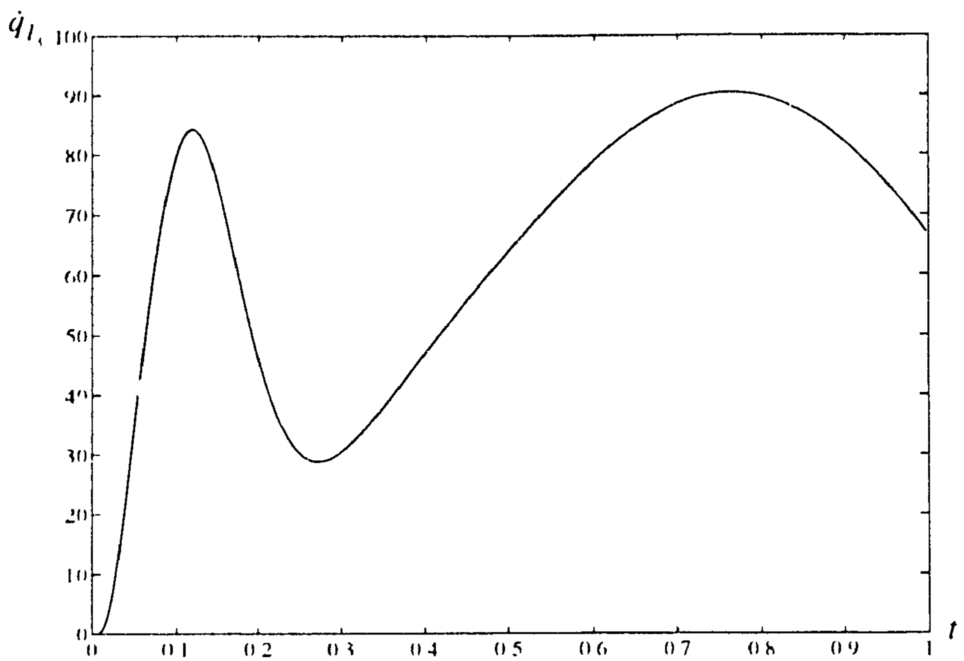

Figure 5.19 **The actual velocity trajectory for the third link**

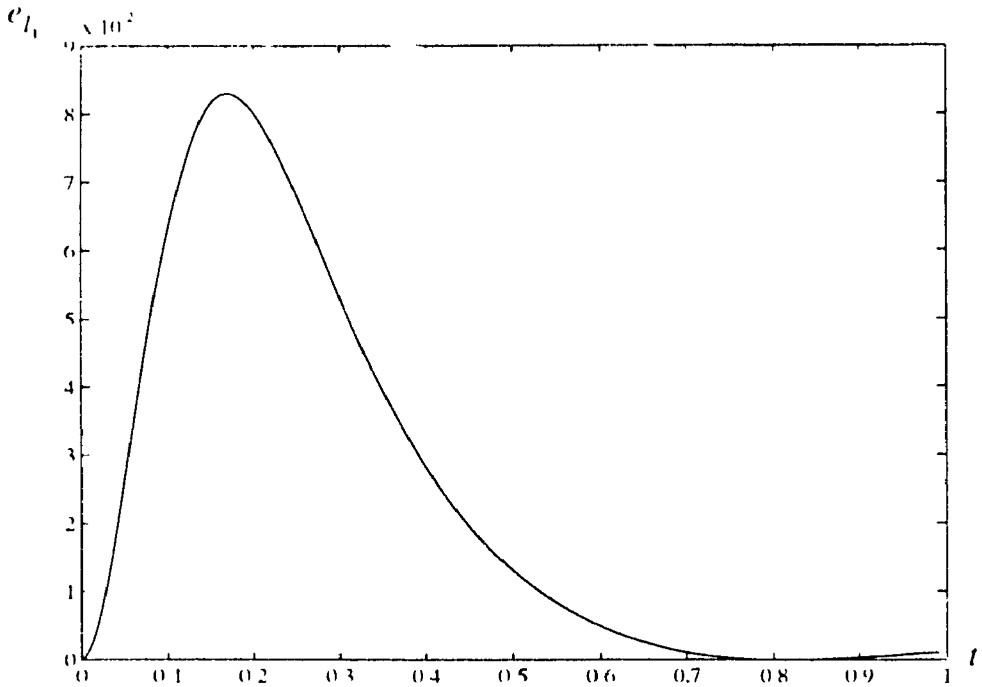

Figure 5 20 **The first link position tracking error**

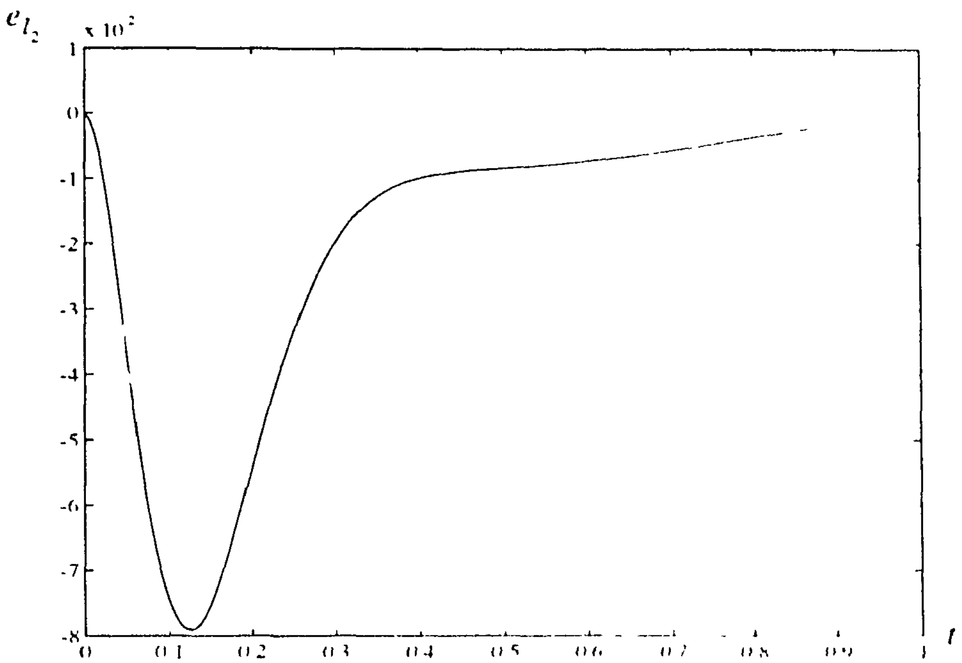

Figure 5 21 **The second link position tracking error**

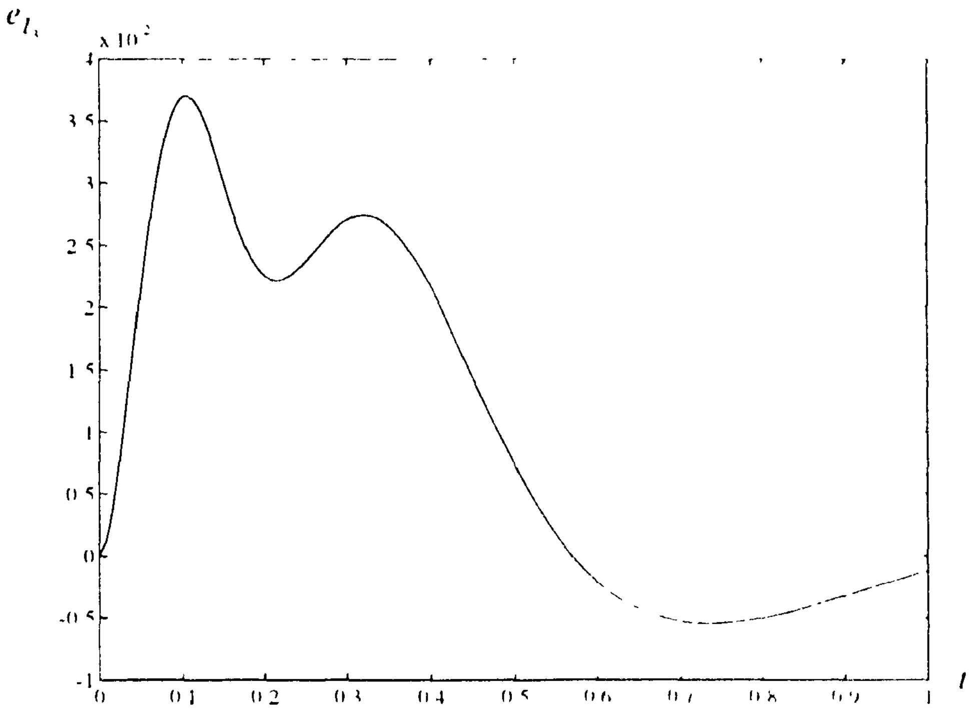

Figure 5 22 **The third link position tracking error**

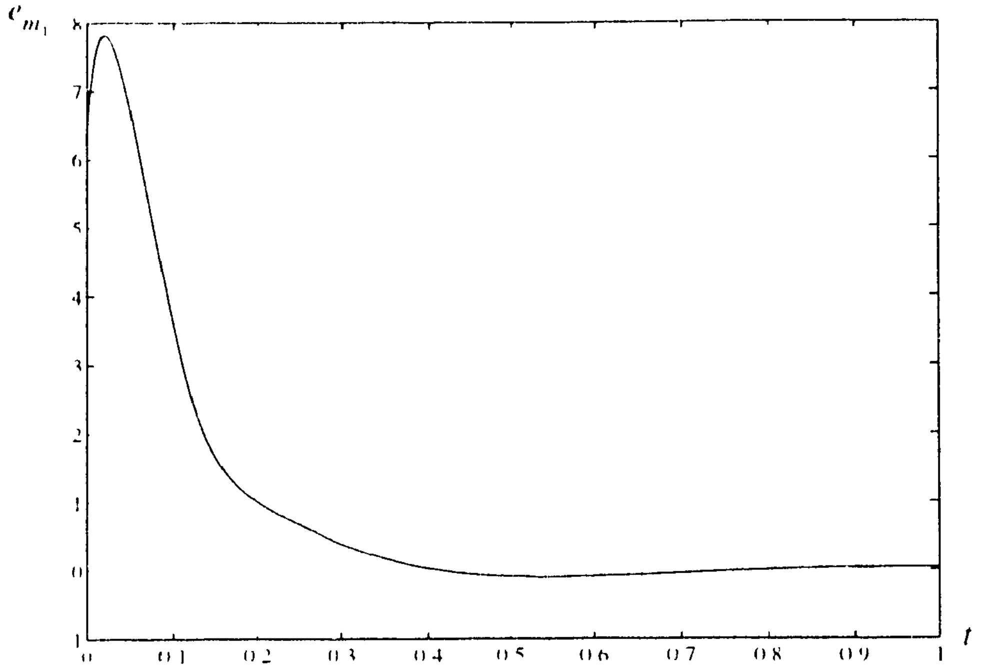

Figure 5 23 **The first motor tracking error**

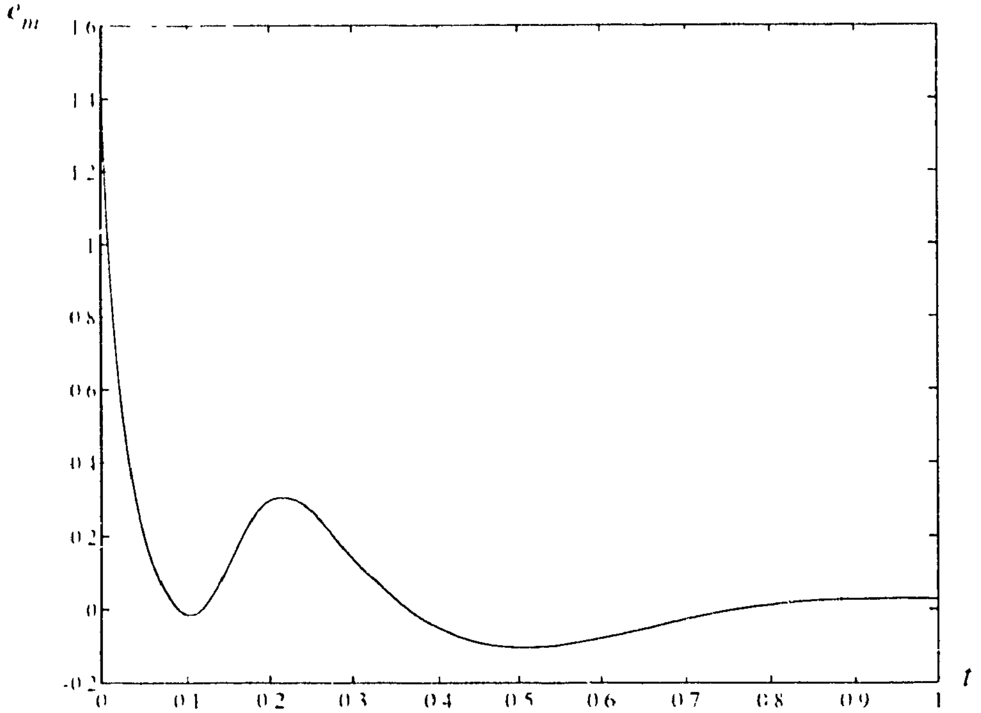

Figure 5.24 **The second motor tracking error**

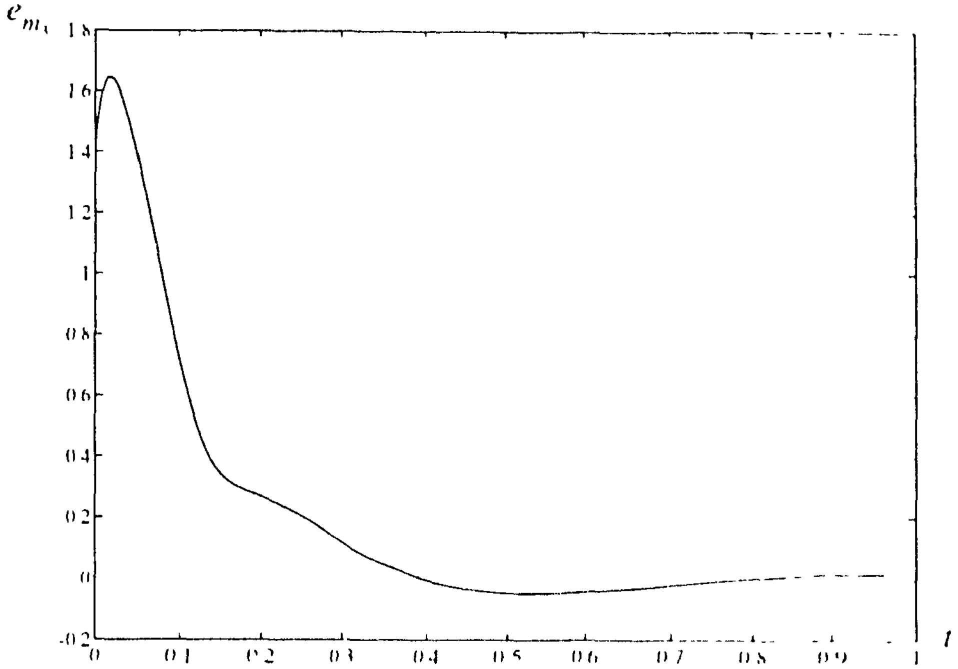

Figure 5.25 **The third motor tracking error**

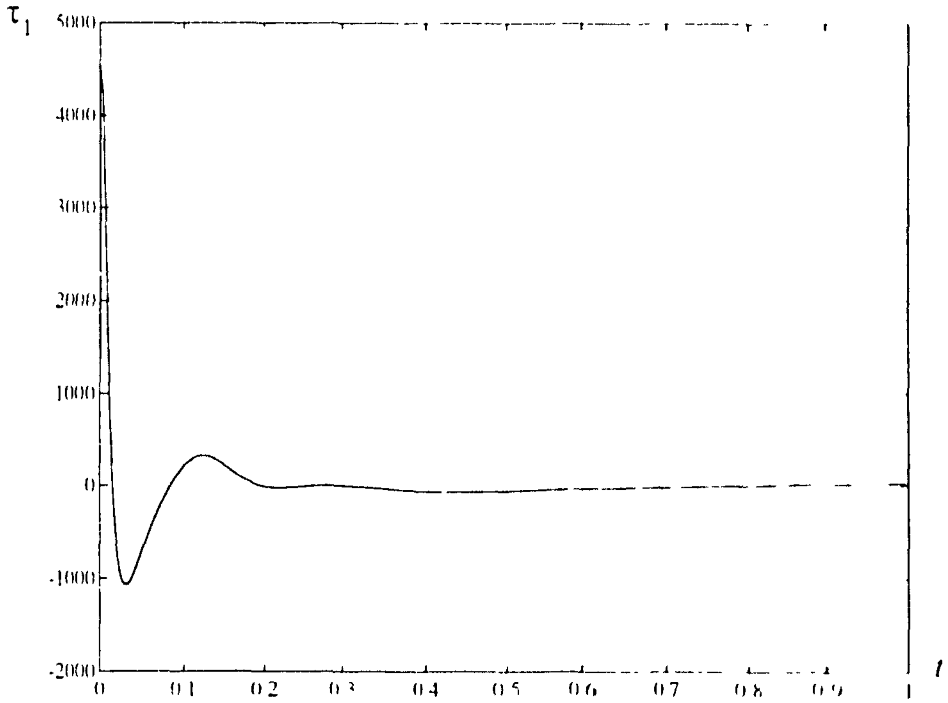

Figure 5.26 **The control torque for the first joint**

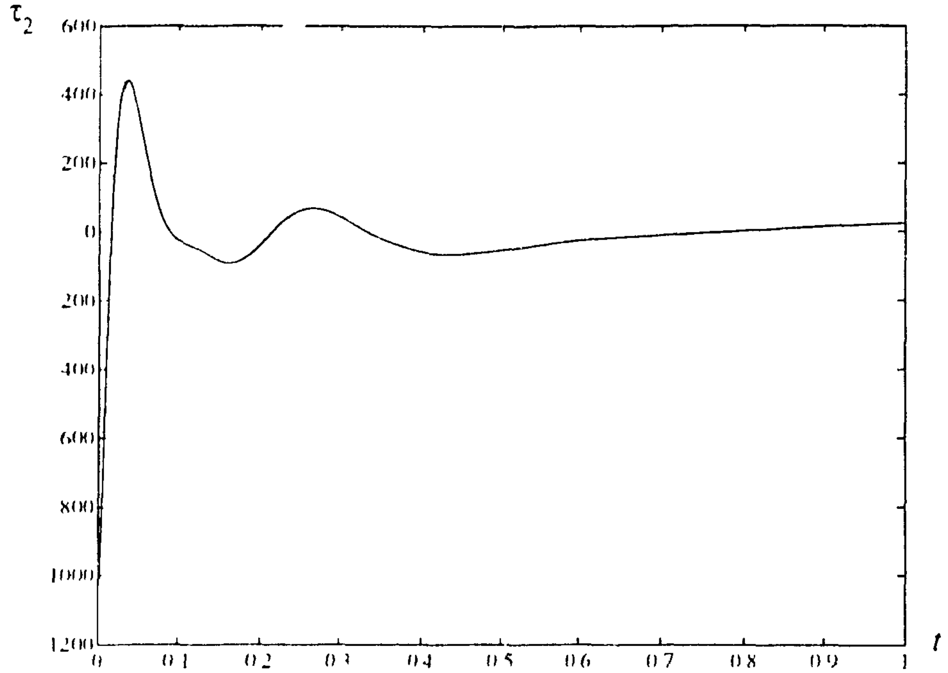

Figure 5 27 **The control torque for the second joint**

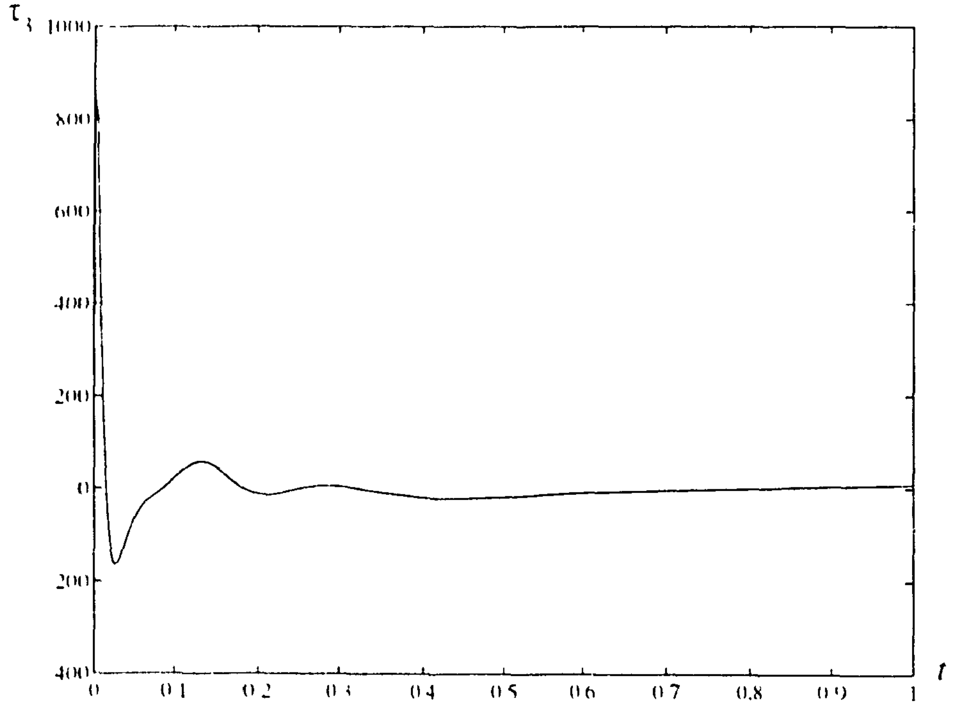

Figure 5 28 **The control torque for the third joint**

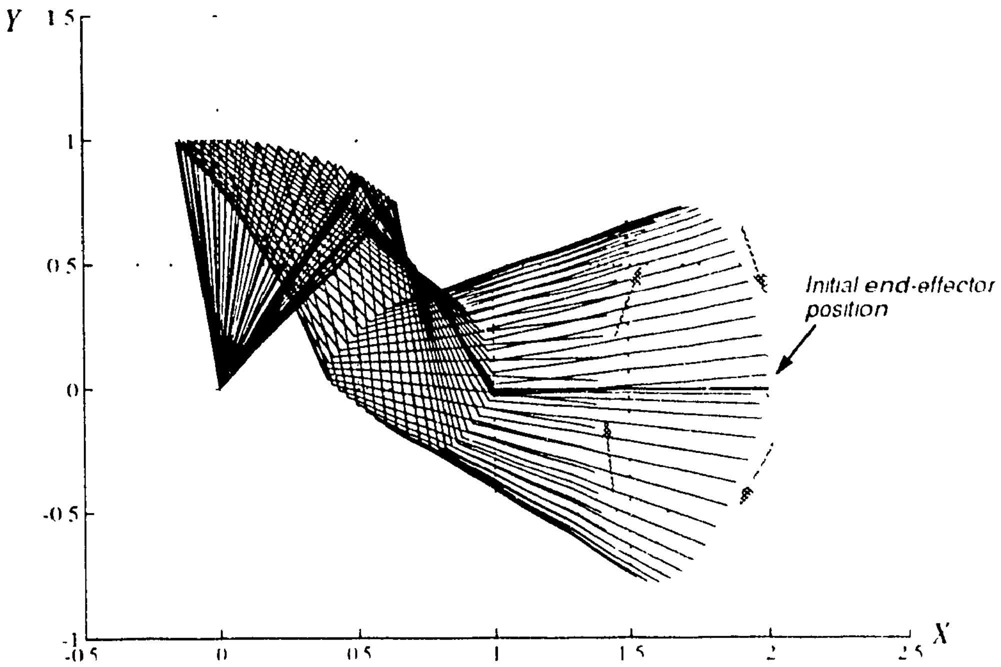

Figure 5.29 **Tracking a Cartesian ellipse (the first revolution)**

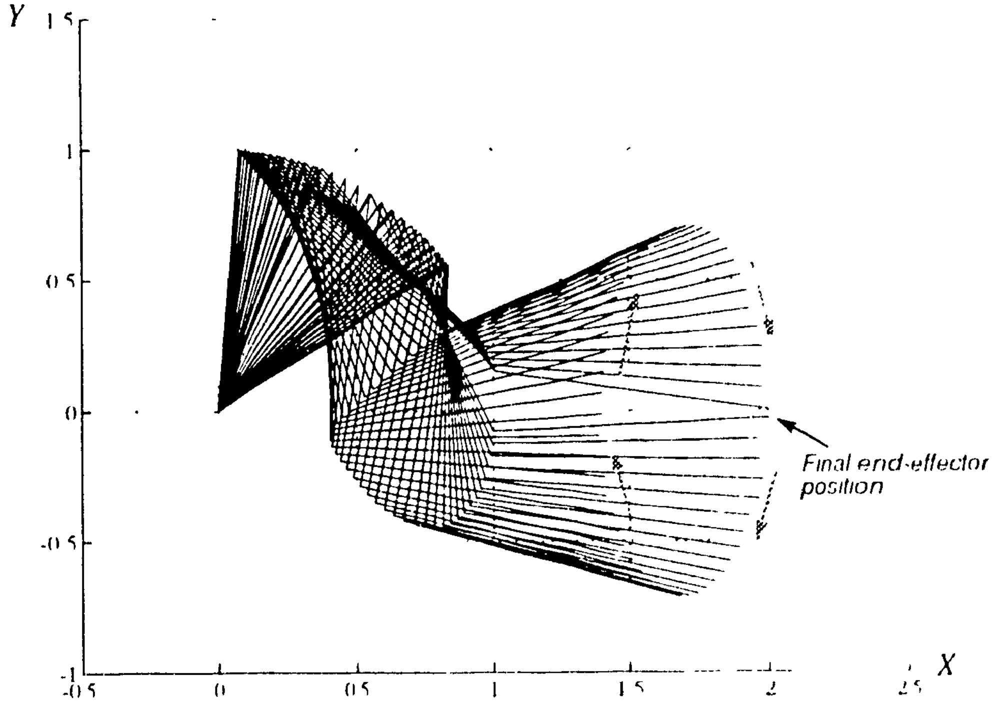

Figure 5.30 **Tracking a Cartesian ellipse (the second revolution)**

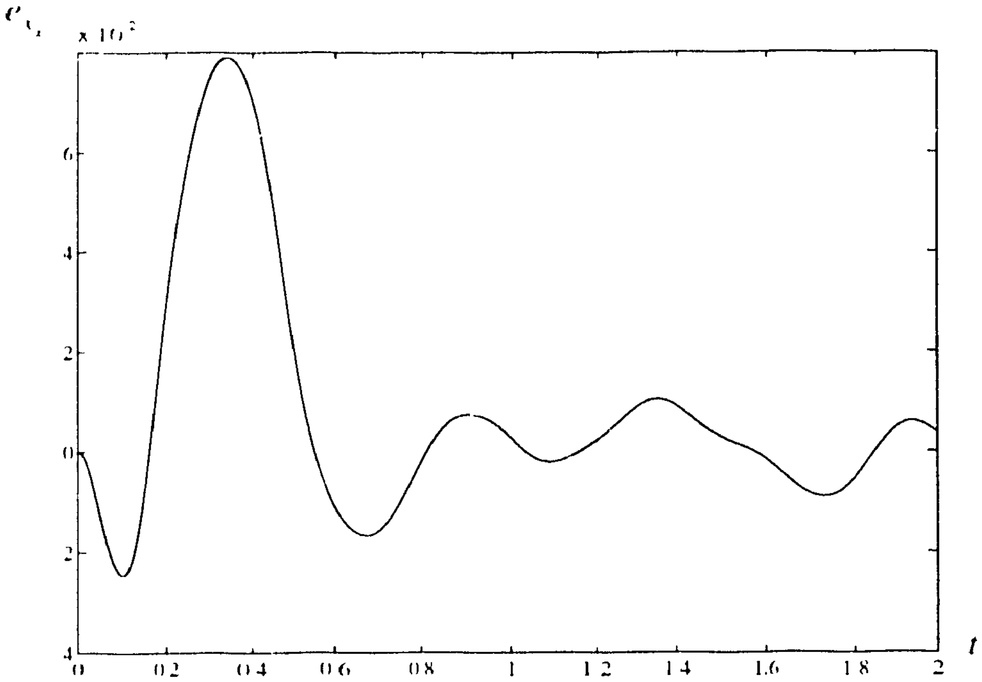

Figure 5.31 **Position error in X direction**

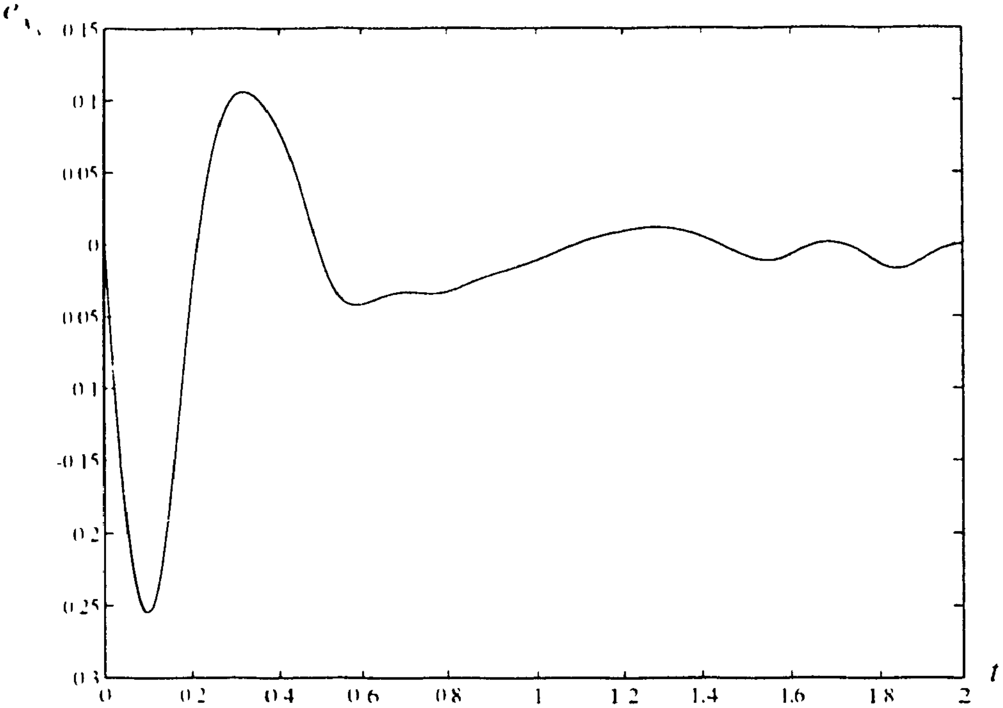

Figure 5 32 **Position error in Y direction**

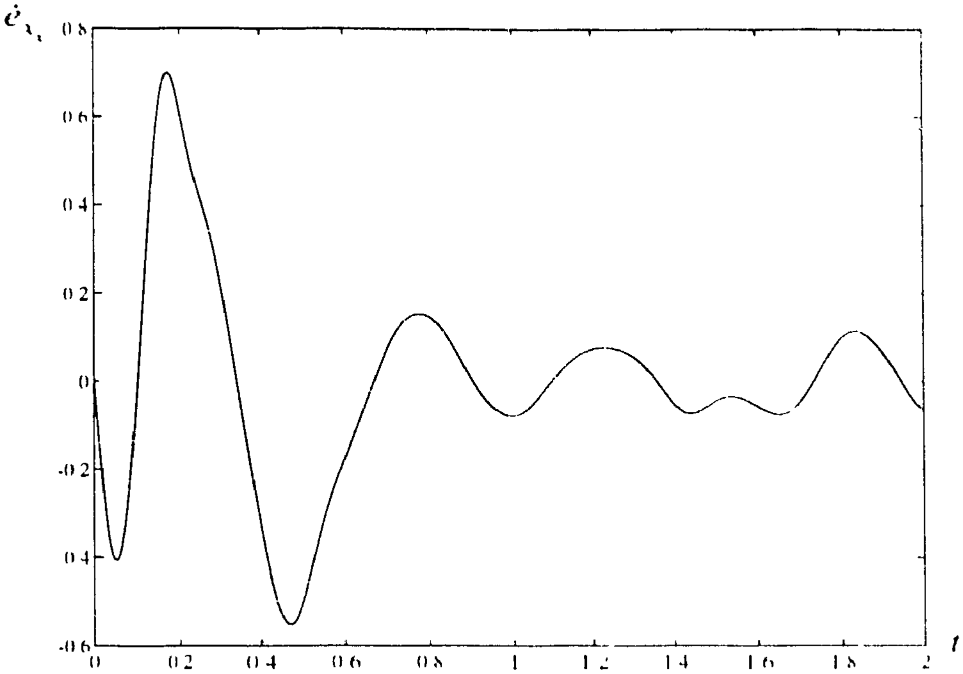

Figure 5.33 **Velocity error in X direction**

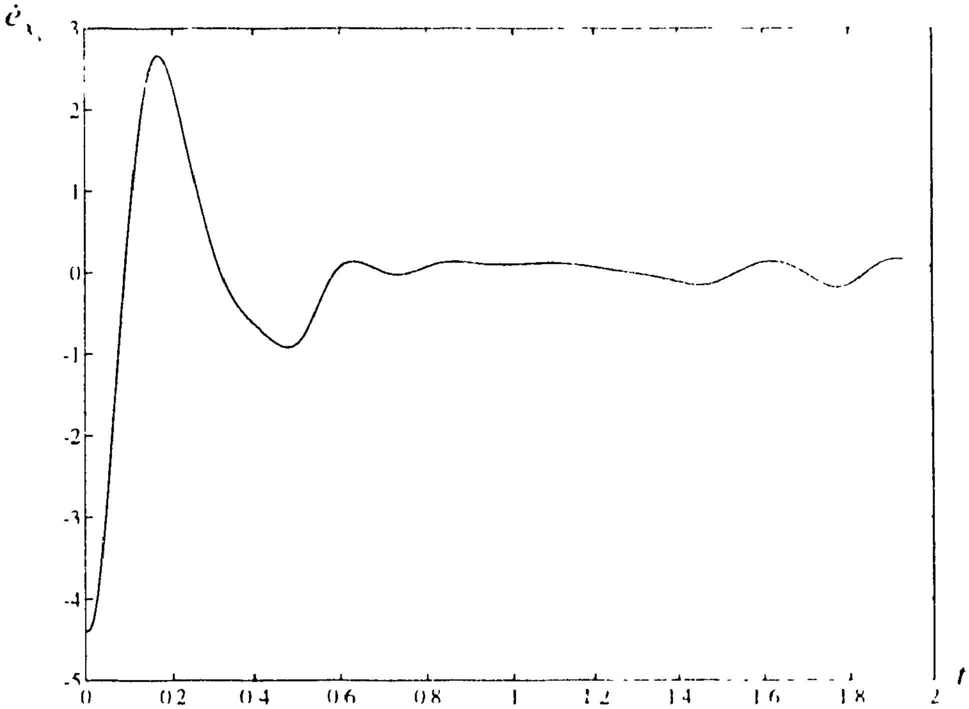

Figure 5.34 **Velocity error in Y direction**

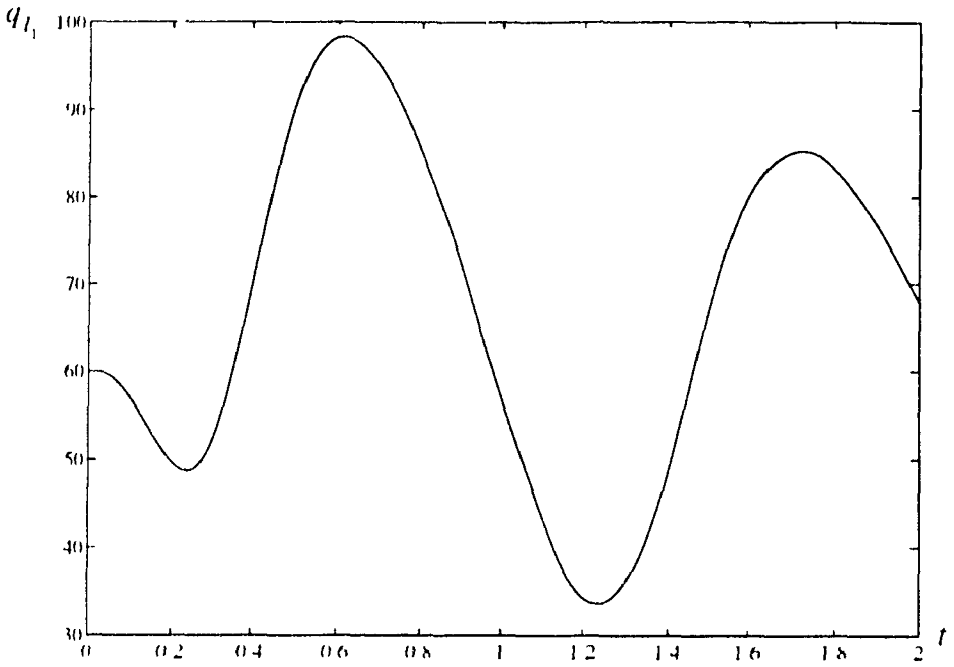

Figure 5.35 **The actual trajectory for the first link**

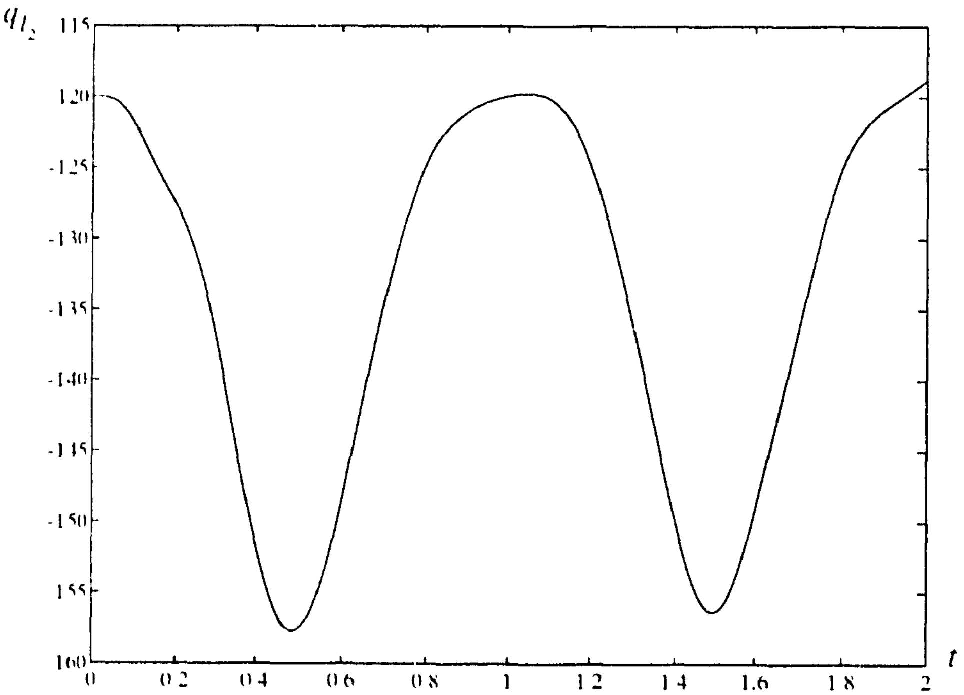

Figure 5 36 **The actual trajectory for the second link**

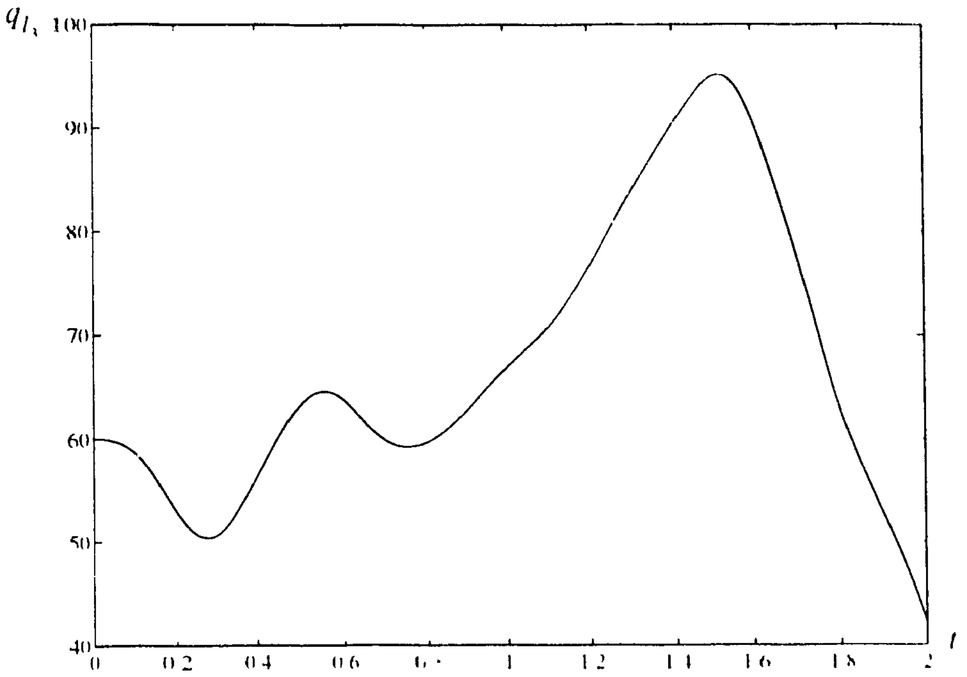

Figure 5.37 **The actual trajectory for the third link**

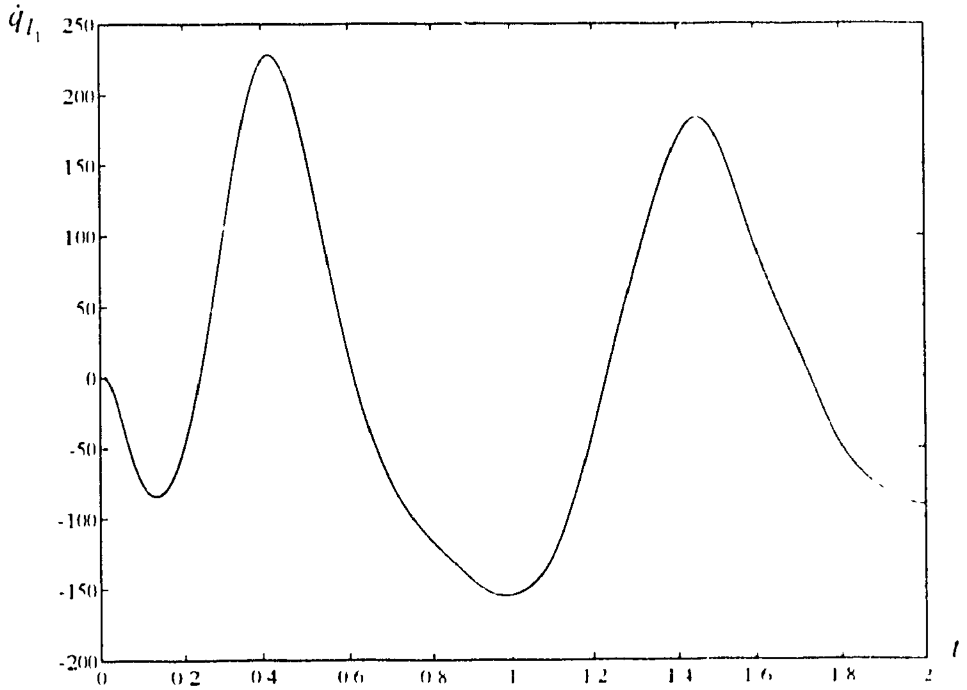

Figure 5.38 **The actual velocity trajectory for the first link**

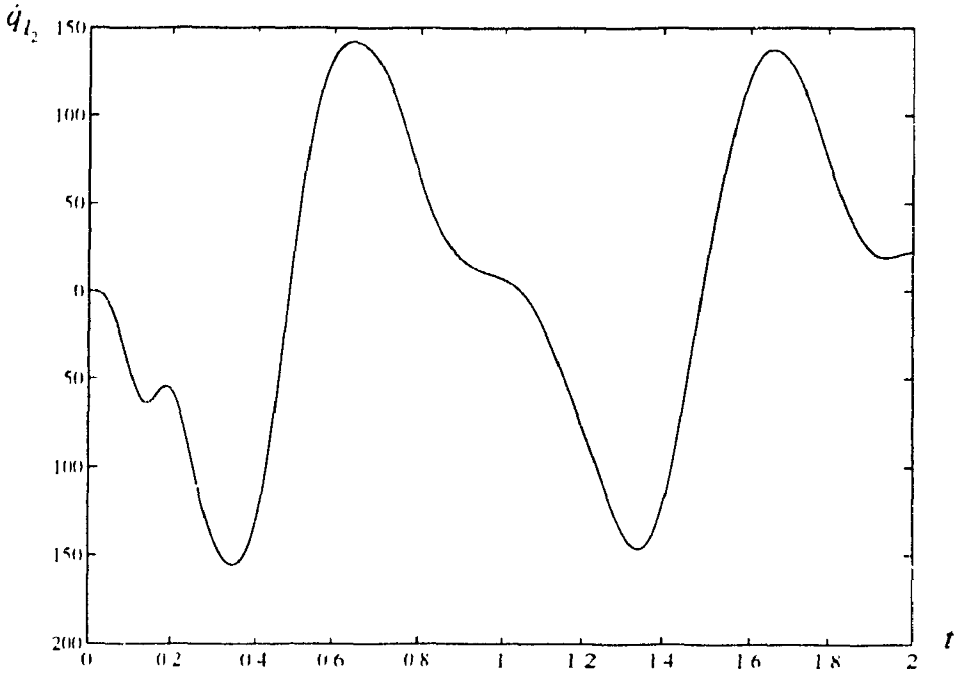

Figure 5 39 The actual velocity trajectory for the second link

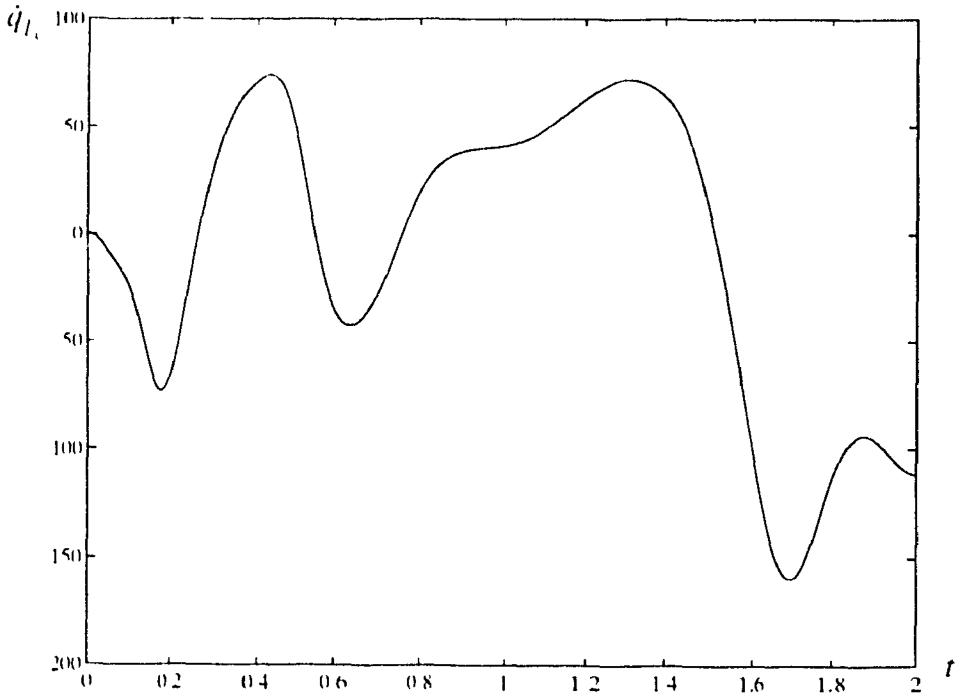

Figure 5 40 The actual velocity trajectory for the third link

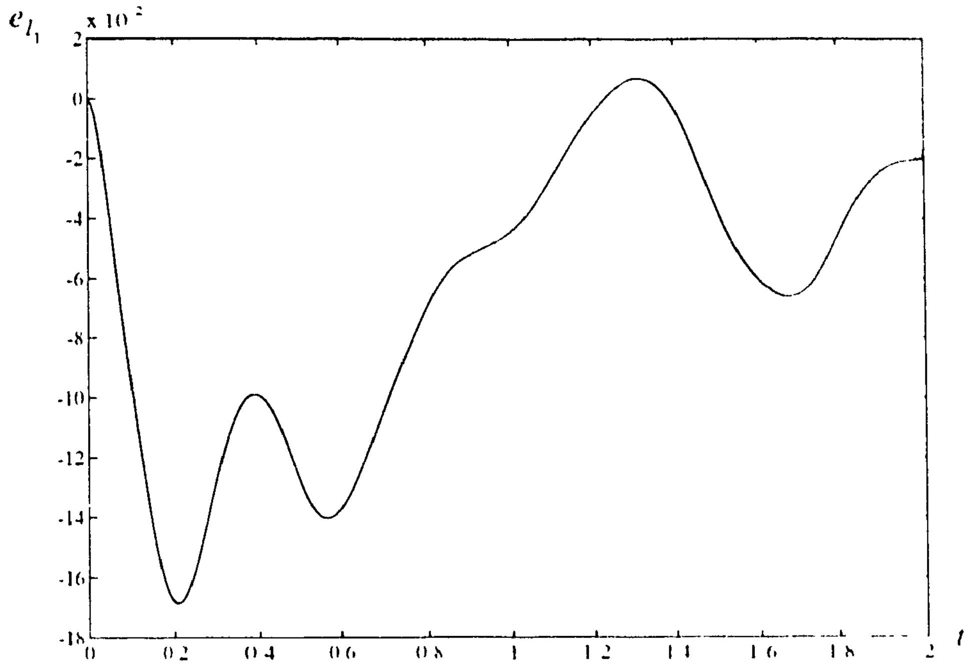

Figure 5.41 **The first link position tracking error**

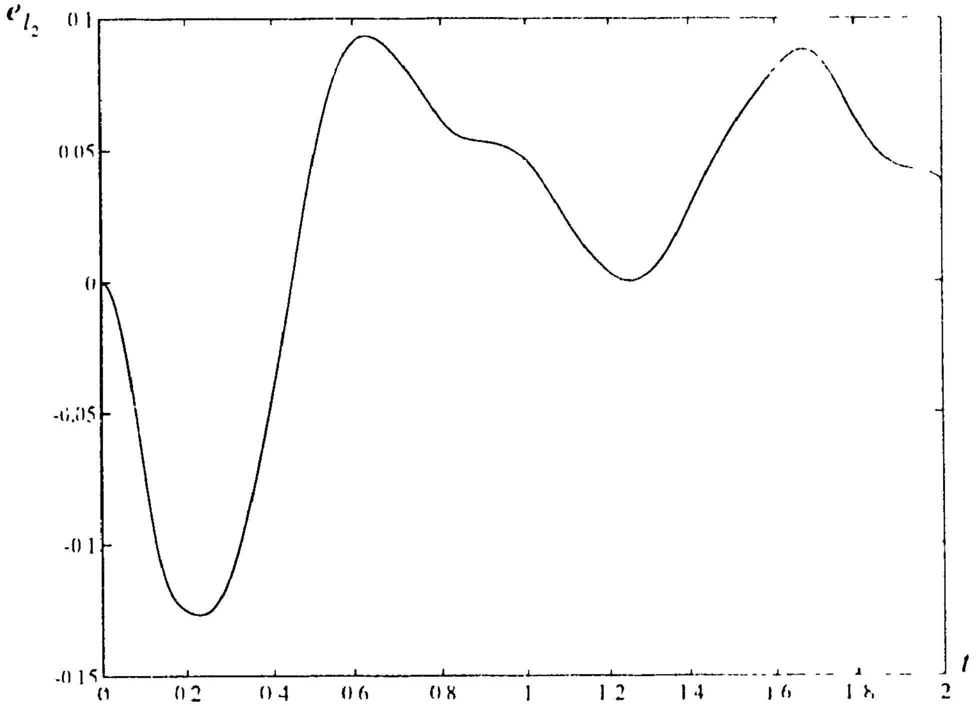

Figure 5 42 **The second link position tracking error**

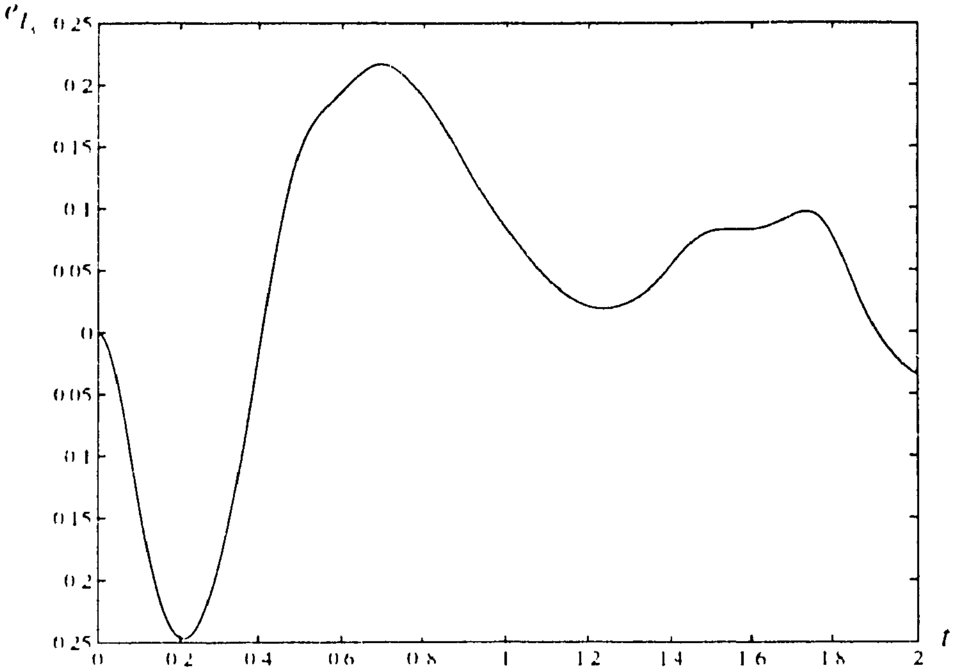

Figure 5.43 **The third link position tracking error**

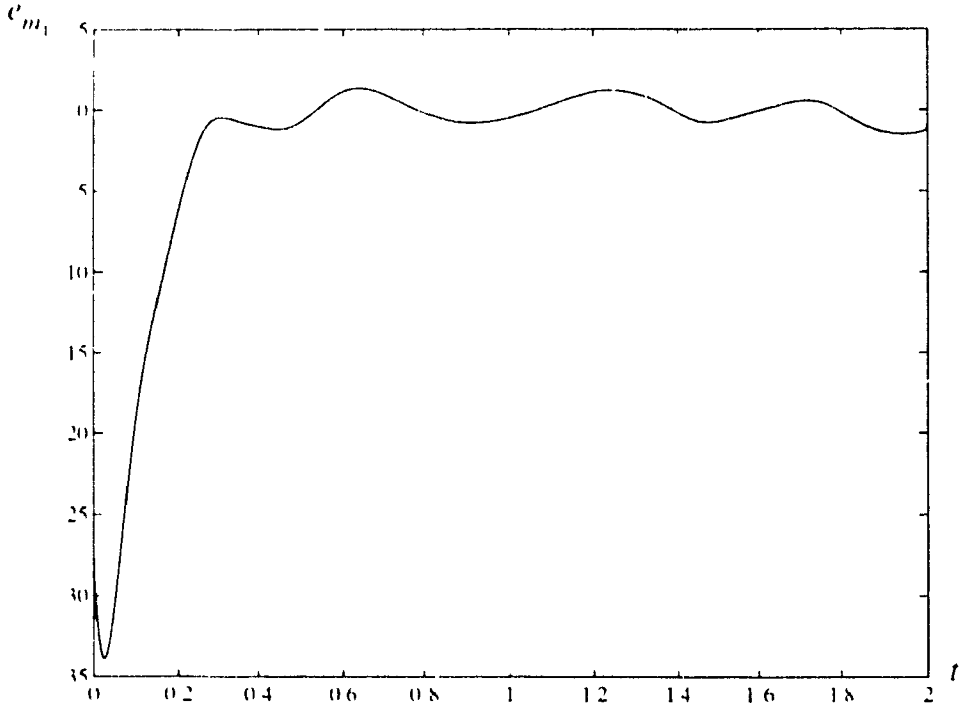

Figure 5 44 **The first motor tracking error**

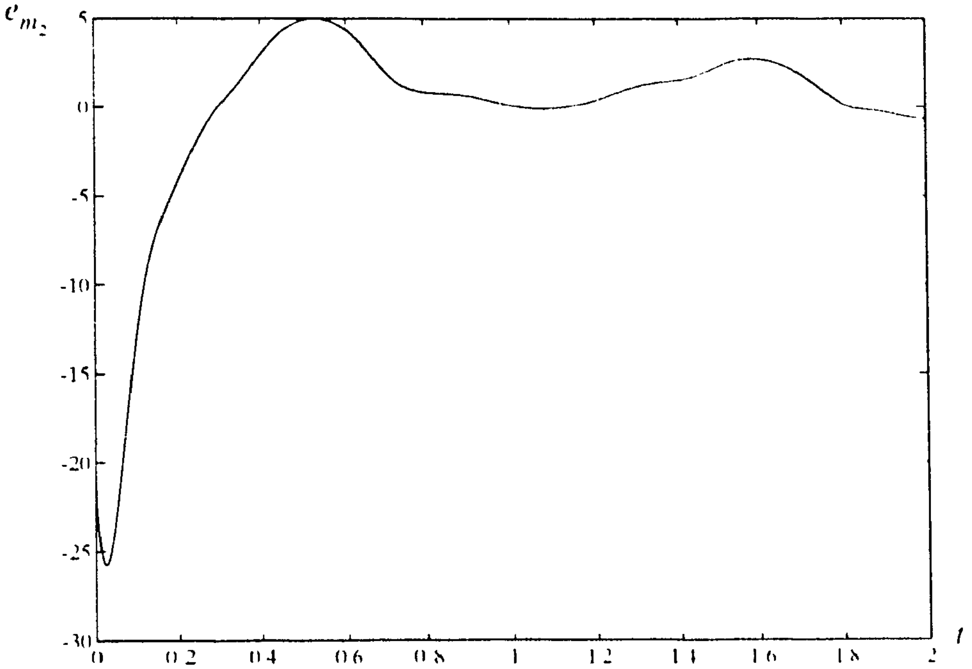

Figure 5.45 **The second motor tracking error**

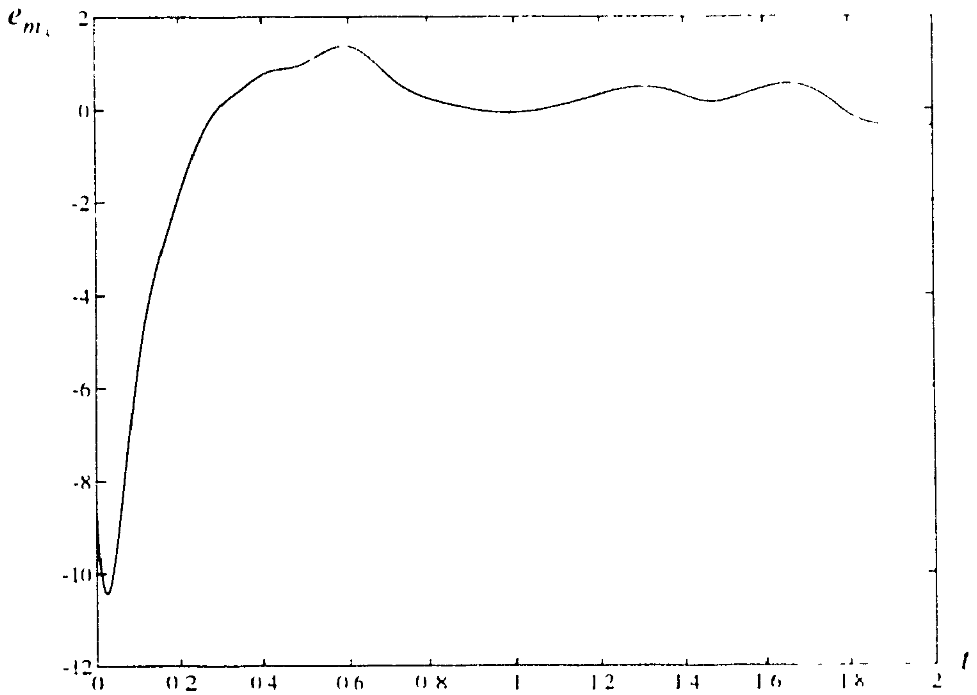

Figure 5.46 **The third motor tracking error**

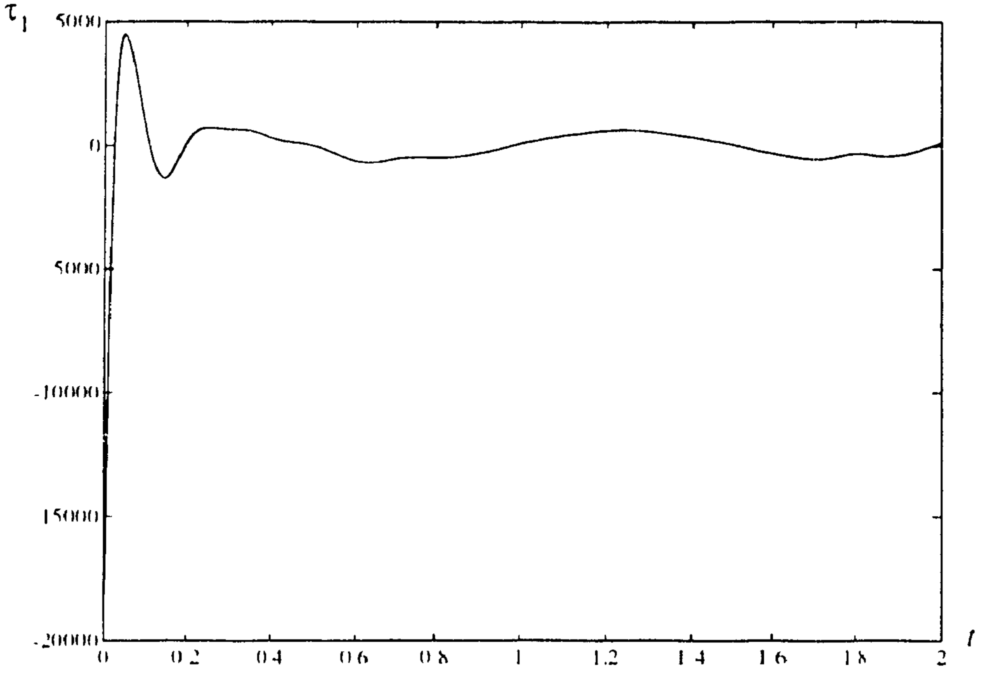

Figure 5.47 **Control torque for the first joint**

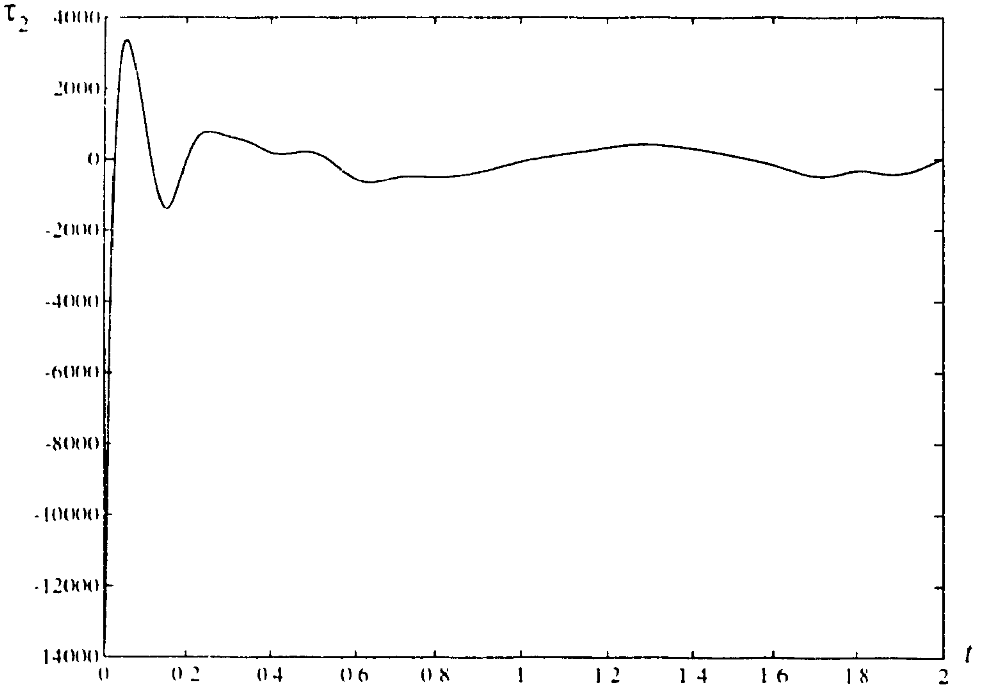

Figure 5 48 **Control torque for the second joint**

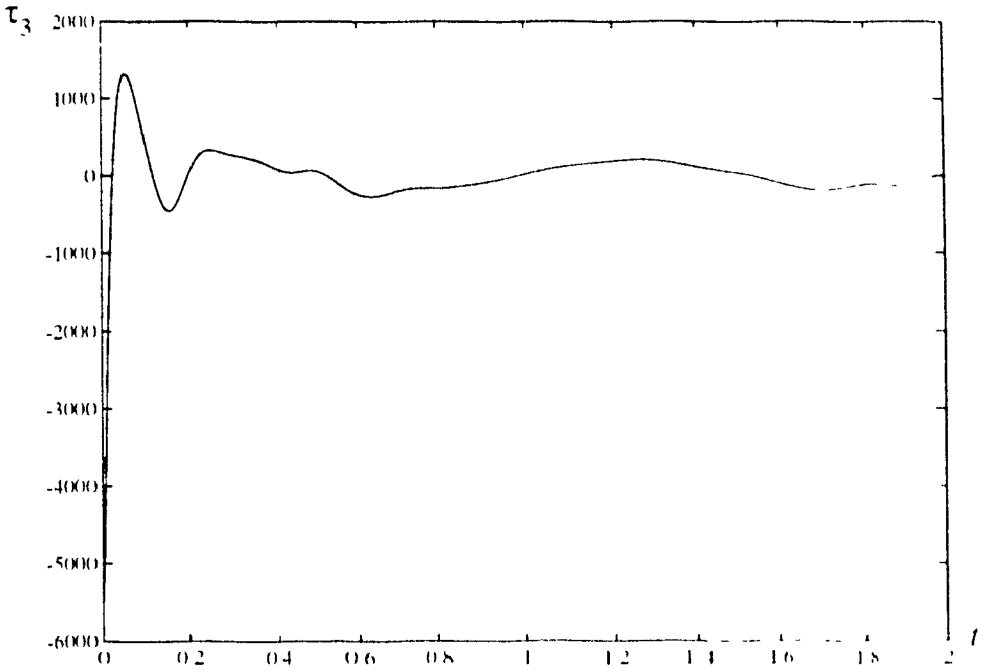

Figure 5 49 **Control to!que for the third joint**

CHAPTER 6 DYNAMIC CONTROL OF REDUNDANT MANIPULATORS TO COMPENSATE FOR JOINT FLEXIBILITY

6.1 INTRODUCTION

Using kinematically redundant manipulators for obstacle avoidance, singularity avoidance and kinematic optimization [3][8][10][15][16][22] has become a very active area of research. However, almost no attention has been paid to the possibility of using kinematic redundancy in solving the problem of how a redundant manipulator can compensate for joint flexibility by appropriate configurations of the redundant manipulator. To our knowledge, there is no published literature on the topic of control of rigid/flexible-joint coupled redundant manipulators whose redundancy is resolved to compensate for joint flexibility. This problem is addressed in this chapter.

Many control strategies have been proposed in the last two decades for the dynamic control of robot manipulators whose dynamics are modeled by the rigid body equations of motion of open kinematic chains [1][6][9][11][14][24][25]. However, most of these approaches are limited in their applicability to real robot manipulators where the assumption of perfect rigidity is never satisfied. For example, manipulators with harmonic drives [29], torque transducers [20] or drive belts [21] exhibit joint flexibility. Therefore specific control schemes must be designed to deal with these non-rigid joint manipulators. As mentioned in Chapter 5, there are a few approaches that have been proposed to solve the problem of controlling robot manipulators with joint elasticity [2][7][12][13][27][28][30].

All of these approaches are designed in joint space and they have been developed for non-redundant manipulators only. Besides, there is another fact worth noting: all these approaches including the Cartesian space approach proposed in Chapter 5 are only suitable for the situation where all the joints of a manipulator are flexible. Hence, the existing methods for the control of flexible-joint manipulators may not be applicable for the cases where manipulators possess rigid as well as flexible joints. Due to the increasing applications of robot manipulators and the rapid development of robotics technology, new types of robot manipulators, and different kinds of actuators are being developed. We may have a robot manipulator equipped with several different types of actuators. We may also have a manipulator with one type of actuators, but different sizes may be used to drive different joints. These actuators may exhibit rather different dynamical characteristics under different payloads. Some of them may exhibit flexibility, while others may remain relatively rigid. For these reasons we shall focus on developing a control strategy in this chapter which is suitable for a control of the rigid/flexible-joint coupled redundant manipulator.

In this chapter, we first develop a Cartesian space based control scheme for flexible joint redundant manipulators. A pseudo-inverse based approach is used for redundancy resolution where the arbitrary vector ξ is defined such that the redundancy is utilized to compensate for joint flexibility in Cartesian space. The main idea behind this is to use the manipulator's self-motion to "shape" the posture of the manipulator such that the internal link motion eliminates the effect of the torsional force due to joint flexibility. After redundancy resolution, the robot arm can be viewed as a "rigid-joint" manipulator.

The rest of the sections in this chapter are organized as follows. In Section 6.2, a brief review of the relevant literature is given. A general dynamic model of a rigid/flexible-joint redundant manipulator is formulated in Section 6.3. Section 6.4 describes the framework of redundancy resolution for compensating for joint flexibility. An approach to avoiding algorithmic singularities is proposed in Section 6.5, while in Section 6.6 the ability of the co ol scheme to compensate for joint flexibility under different degrees of redundancy is discussed. In Section 6.7, the issue of calculating higher order derivatives of tl [illegible] nt variables is discussed and an estimation algorithm is proposed. Stability of the closed-loop system is shown in Section 6.8. To show the applicability of the proposed approach, results of numerical simulations are given and analyzed in Section 6.9. Finally, Section 6.10 draws some conclusions concerning the approach proposed in this chapter.

6.2 LITERATURE REVIEW

As shown in Chapter 5, there are several approaches to the control of flexible-joint manipulators. As pointed out earlier, these approaches are only suitable for the joint control of non-redundant manipulators. To our knowledge, there are no publications yet concerning modeling and control of rigid/flexible-joint coupled redundant manipulators whose redundancy is used to compensate for joint flexibility. However, in this Section, we give a brief review of two issues which are closely related to this topic in order to show the status of redundancy resolution in the context of compensating against joint flexibility.

Nguyen et. al. [18][19] have proposed a scheme using redundancy to compensate for link flexibility. A computed torque based nonlinear controller is used. However, their scheme only deals with the regulation problem for flexible-link manipulators. It is not clear that their approach is able to solve the tracking problem in the flexible-link case. This implies that the problem of using redundancy to compensate for link flexibility is still far from completely solved. Another point is that their scheme creates algorithmic singularities in resolving redundancy, and there is no indication that they can avoid singularities. Our experience tells us that this kind of singularities is encountered very frequently in a manipulator's workspace, and the redundancy resolution algorithm fails if the manipulator's configuration is in the neighborhood of an algorithmic singularity. A similar problem has also been mentioned in [10], and this is still a open research problem.

In a recent paper [4][5], Baillieul discusses the relationship between manipulator redundancy and flexible components (flexible joints or flexible links). Both kinematic and dynamic aspects are addressed for flexible robots. To deal with redundancies, Baillieul uses the extended Jacobian method by placing the off diagonal submatrix D_{12} of the manipulator's inertia matrix into the Jacobian matrix. It is known that in the general model of the flexible-joint manipulator, the magnitudes of the elements in submatrix D_{12} are usually very small. Therefore, the extension of the Jacobian matrix may not be well-conditioned. In the rest of the paper, he focuses on the problem of generating a trajectory for the manipulator such that the end-effector moves between prescribed endpoints without any net storage of elastic energy in the flexible joints. He uses an interpolation technique such as fifth-order splines to construct Cartesian trajectories for a flexible-joint manipulator.

In this chapter, we first give a general dynamic model of the rigid/flexible-joint coupled redundant manipulator, and then propose a control strategy for this type of manipulators.

6.3 GENERAL DYNAMIC MODEL OF REDUNDANT RIGID/FLEXIBLE JOINT MANIPULATORS

A dynamic model of a robot manipulator with flexibility in all its joints has been described in [26]. However, a more general model, can be developed to show that a manipulator may have flexibility in some (or all) of its joints. The main reason for considering flexibility only exi^sting in some joints is that there may be different types of actuators used in different joints, or there may be different transmission distances for joints closer to the base of the manipulator from those farther away. Another reason is that even if the same type of actuator is used for the joints of a manipulator, the dynamic performance of each actuator will be different due to different loads imposed on each joint. Consequently, some of them may exhibit more flexibility than others. Therefore, a model of a manipulator which allows for rigid as well as flexible joints (and different joint flexibilities) is much more general than that proposed in [26].

It is assumed that the manipulator has $n+1$ rigid links interconnected by n joints, and it is further assumed that there are s flexible joints $(s \le n)$, while the remaining joints $(n-s)$ are rigid. The general dynamic model can be expressed as

$$\begin{bmatrix} D_{11}(q_{l1}, q_{l2}) + D_{mr} & D_{12}(q_{l1}, q_{l2}) & 0 \\ D_{21}(q_{l1}, q_{l2}) & D_{22}(q_{l1}, q_{l2}) & 0 \\ 0 & 0 & D_{mf} \end{bmatrix} \begin{bmatrix} \ddot{q}_{l1} \\ \ddot{q}_{l2} \\ \ddot{q}_{m} \end{bmatrix} + \begin{bmatrix} C_{l1}(q_{l1}, q_{l2}, \dot{q}_{l1}, \dot{q}_{l2}) \\ C_{l2}(q_{l1}, q_{l2}, \dot{q}_{l1}, \dot{q}_{l2}) \\ 0 \end{bmatrix}$$

$$+ \begin{bmatrix} G_{l1}(q_{l1}, q_{l2}) \\ G_{l2}(q_{l1}, q_{l2}) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ K_t(q_{l2} - q_m) \\ -K_t(q_{l2} - q_m) \end{bmatrix} = \begin{bmatrix} \tau_{l1} \\ 0 \\ \tau_m \end{bmatrix} \qquad (6.3.1)$$

where $q_{l1} \in \Re^{(n-s)\times 1}$ represents the vector of link angles (motor angles) corresponding to the rigid joints, $q_{l2} \in \Re^{s \times 1}$ and $q_m \in \Re^{s \times 1}$ represent the vectors of link angles and motor angles respectively, corresponding to the flexible joints; the submatrix $D_l \equiv \begin{bmatrix} D_{11} + D_{mr} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \in \Re^{n \times n}$ represents the symmetric and positive-definite inertia matrix of the manipulator $(D_{12} = D_{21}^T)$, while $D_{mr} \in \Re^{(n-s)\times(n-s)}$ and $D_{mf} \in \Re^{s \times s}$ denote the inertia matrices of the actuators corresponding to rigid and flexible joints respectively. The vectors $C_{l1} \in \Re^{(n-s)\times 1}$, $C_{l2} \in \Re^{s \times 1}$ represent the Coriolis and centrifugal forces, $G_{l1} \in \Re^{(n-s)\times 1}$, $G_{l2} \in \Re^{s \times 1}$ represent the gravitational forces, and $\tau_{l1} \in \Re^{(n-s)\times 1}$, $\tau_m \in \Re^{s \times 1}$ denote the input torques to the actuators at the rigid and flexible joints respectively. The term $K_t \in \Re^{s \times s}$ denotes the diagonal stiffness matrix for the flexible joints.

Due to the special dynamic structure of the manipulator, the existing rigid-joint as well as flexible-joint manipulator controllers are no longer applicable. Therefore, it is necessary to construct a new controller for this type of manipulators.

6.4 REDUNDANCY RESOLUTION FOR COMPENSATING AGAINST JOINT FLEXIBILITY

Let $q_l = \begin{bmatrix} q_{l1} \\ q_{l2} \end{bmatrix}$, and $q = \begin{bmatrix} q_l \\ q_m \end{bmatrix}$; for the rigid joints, the controller can be easily constructed as

$$\tau_{l1} = \begin{bmatrix} D_{11} \; D_{12} \end{bmatrix} \{ J_e^{+} \, (\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - J_e\dot{q}_l)$$

$$+ \, (I - J_e^{+} J_e) \, \xi \; \} + C_{l1} + G_{l1} \qquad (6.4.1)$$

where $\xi \in \Re^{n \times n}$ is the redundancy resolution vector which will be specified later in this Chapter. Similar to the controller specified in Chapter 5, the link-level controller for flexible-joint manipulator can be designed as

$$q_{md} = K_t^{-1} \{ \begin{bmatrix} D_{21} \; D_{22} \end{bmatrix} \{ J_e^{+} \, (\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e q_l \,)$$

$$+ \, (I - J_e^{+} J_e) \, \xi \; \} \; + C_{l2} + G_{l2} + K_t q_l \; \} \qquad (6.4.2)$$

Here, the problem becomes one of finding the vector ξ as an implicit controller to compensate for the joint flexibility.

We rewrite the motor dynamic equation in (5.1) as

$$D_{mf}\ddot{q}_m - K_t\,(q_{l2} - q_m) \; = \; \tau_m . \qquad (6.4.3)$$

The motor-level controller can then be defined as

$$\tau_m = D_{mf}[\ddot{q}_{md} + K_{vm}(\dot{q}_{md} - \dot{q}_m) + K_{pm}(q_{md} - q_m)] - K_t(q_{l2} - q_m) \qquad (6.4.4)$$

where $K_{pm} \in \Re^{s \times s}$ and $K_{vm} \in \Re^{s \times s}$ are motor control gain matrices. Notice that the term corresponding to the torsional force $K_t(q_{l2} - q_m)$ in motor controller (6.4.4) attempts to cancel the torsional force in motor dynamic equation (6.4.3) so that the rest of the terms in (6.4.3) and (6.4.4) form a standard second-order system. However, in order to use redundancy to compensate for joint flexibility, instead of directly calculating the term $K_t(q_{l2} - q_m)$ in (6.4.4), the crucial point here is to utilize self-motion to cancel out the torsional force in (6.4.3). In other words, at any time instant the configuration of the manipulator evolves in a way that enables the link angles to equal the corresponding their motor angles for the flexible joints. This implies that the torsional forces existing in the flexible joints vanish as a result of the manipulator's configuration change due to its self-motion. This can be achieved using the redundancy in q_{md} in (6.4.2). Therefore, the motor-level controller can be written without the term $K_t(q_{l2} - q_m)$ as

$$\tau_m = D_{mf}[\ddot{q}_{md} + K_{vm}(\dot{q}_{md} - \dot{q}_m) + K_{pm}(q_{md} - q_m)] \qquad (6.4.5)$$

The details of the controller design are as follows:

We propose two different approaches to resolve manipulator redundancy to compensate for joint flexibility. The first approach directly obtains the redundancy resolution vector ξ_p (a transformed version of ξ, which will be specified later in this section), while the second approach finds ξ_p in an indirect way.

Direct approach:

We first define q^o_{md} as

$$q^o_{md} = K_t^{-1}\{[D_{21}\ D_{22}]J_e^{\dagger}(\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e\dot{q}_l) + C_{l2} + G_{l2} + K_t q_l\} \qquad (6.4.6)$$

Hence, the link-level controller q_{md} in (6.4.2) can be rewritten as

$$q_{md} = q^{\circ}_{md} + K_t^{-1}\left[D_{21}\ D_{22}\right](I - J_e^{\dagger} J_e)\,\xi. \tag{6.4.7}$$

Taking the first and the second derivatives of q_{md} in (6.4.7), and substituting q_{md}, $\dot{q}_{md}$ and $\ddot{q}_{md}$ into (6.4.5), we obtain from equation (6.4.5)

$$\tau_m = D_{mf}[\,\ddot{q}^{\circ}_{md} + K_{vm}(\dot{q}^{\circ}_{md} - q_m) + K_{pm}(q^{\circ}_{md} - q_m)\,] + T_N \tag{6.4.8}$$

The torque T_N can be expressed as

$$T_N = D_{mf}\{\Omega_1 (I - J_e^{\dagger} J_e)\ddot{\xi} + \Omega_2 (I - J_e^{\dagger} J_e)\dot{\xi} + \Omega_3 (I - J_e^{\dagger} J_e)\xi\} \tag{6.4.9}$$

where matrices Ω_1, Ω_2 and $\Omega_3 \in \Re^{s \times n}$ are given by

$$\Omega_1 = K_t^{-1}\left[D_{21}\ D_{22}\right] \tag{6.4.10}$$

$$\Omega_2 = 2K_t^{-1}(\left[\dot{D}_{21}\ \dot{D}_{22}\right] - \left[D_{21}\ D_{22}\right]J_e^{\dagger}\dot{J}_e) + K_{vm}K_t^{-1}\left[D_{21}\ D_{22}\right] \tag{6.4.11}$$

$$\begin{aligned}\Omega_3 = {} & K_t^{-1}\{\left[\ddot{D}_{21}\ \ddot{D}_{22}\right] + \left[D_{21}\ D_{22}\right]J_e^{\dagger}(2\dot{J}_e J_e^{\dagger}\dot{J}_e - \ddot{J}_e) - 2\left[\dot{D}_{21}\ \dot{D}_{22}\right]J_e^{\dagger}\dot{J}_e\} \\ & + K_{vm}K_t^{-1}(\left[\dot{D}_{21}\ \dot{D}_{22}\right] - \left[D_{21}\ D_{22}\right]J_e^{\dagger}\dot{J}_e) + K_{pm}K_t^{-1}\left[D_{21}\ D_{22}\right]\end{aligned} \tag{6.4.12}$$

They are functions of known variables and constants so that Ω_1, Ω_2 and Ω_3 can be computed at each step. Therefore, the torque T_N could be used to produce the same amount of torque as the torsional force in the flexible joints so as to eliminate the effect of joint flexi bility. This can be achieved by letting T_N equal to the torsional force $K_t(q_{l2}-q_m)$,

$$D_{mf}\{\Omega_1(I-J_e^{\dagger}J_e)\ddot{\xi}+\Omega_2(I-J_e^{\dagger}J_e)\dot{\xi}+\Omega_3(I-J_e^{\dagger}J_e)\xi\} = K_t(q_{l2}-q_m) \quad (6.4.13)$$

The matrix $(I-J_e^{\dagger}J_e)$ is symmetric with rank $n-m$, and can be written as PP^T where $P\in\Re^{n\times(n-m)}$ has rank $n-m$. Furthermore, writing $\xi_p = P^T\xi\in\Re^{(n-m)\times 1}$, $\dot{\xi}_p = P^T\dot{\xi}$, $\ddot{\xi}_p = P^T\ddot{\xi}$, we can express equation (6.4.13) in the form

$$\Psi_1\ddot{\xi}_p+\Psi_2\dot{\xi}_p+\Psi_3\xi_p = \Gamma \quad (6.4.14)$$

where $\Gamma = K_t(q_{l2}-q_m)$; $\Psi_1\in\Re^{s\times(n-m)}$, $\Psi_2\in\Re^{s\times(n-m)}$ and $\Psi_3\in\Re^{s\times(n-m)}$ are formed as $\Psi_1 = D_{mf}\Omega_1P$, $\Psi_2 = D_{mf}\Omega_2P$ and $\Psi_3 = D_{mf}\Omega_3P$. Usually, the redundancy resolution vector ξ_p can be obtained by solving the second-order differential equation (6.4.14) directly, and ξ_p can then be substituted into the controllers (6.4.1) and (6.4.2) to compensate the joint flexibility by using the manipulator's self-motion. However, care has to be taken in solving for ξ_p because *algorithmic singularities* may arise in (6.4.14). This problem will be addressed in the next section.

Indirect approach:

We let the second term in equation (6.4.7) be denoted by $\gamma\in\Re^{s\times 1}$, i.e.,

$$\gamma = K_t^{-1}\begin{bmatrix}D_{21} & D_{22}\end{bmatrix}(I-J_e^{\dagger}J_e)\xi \qquad \text{or} \qquad \gamma = K_t^{-1}\begin{bmatrix}D_{21} & D_{22}\end{bmatrix}P\xi_p \quad (6.4.15)$$

and

$$q_{md} = q^{\circ}{}_{md}+\gamma. \quad (6.4.16)$$

The vector γ represents the amount of link movement produced by the manipulator's self-motion. Differentiating (6.4.16) once and twice, and substituting q_{md} $\dot{q}_{md}$ and $\ddot{q}_{md}$ into the motor controller (6.4.5), we get the following expression for the motor controller:

$$\tau_m = D_{mf}[\ddot{q}^o{}_{md} + K_{vm}(\dot{q}^o{}_{md} - \dot{q}_m) + K_{pm}(q^o{}_{md} - q_m)]$$

$$+ D_{mf}[\ddot{\gamma} + K_{vm}\dot{\gamma} + K_{pm}\gamma] \qquad (6.4.17)$$

The first term in square brackets on the right-hand side of (6.4.17) represents the torque required for tracking the motor variables, while the second term corresponds to the null-space torque which is used to eliminate the effect of the torsional forces. Then, the following equation holds

$$\ddot{\gamma} + K_{vm}\dot{\gamma} + K_{pm}\gamma = D_{mf}^{-1}K_t(q_m - q_{l2}) \qquad (6.4.18)$$

The link position γ due to self-motion can be obtained by solving (6.4.18). This value of γ can be used in equation (6.4.15) to solve for the redundancy resolution vector ξ_p. Finally, ξ_p is substituted in the expressions for the rigid as well as flexible link controllers (6.4.1) and (6.4.2) to ensure that the manipulator self-motion will compensate for the joint flexibility. However, the product of the inertia matrix $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix}$ and the projection matrix $(I - J_e^{\dagger}J_e)$ in equation (6.4.15) may . always be full rank along some prespecified trajectory and its inverse may not exist. Hence, it is necessary to develop an algorithm which avoids this algorithmic singularity. This problem will be addressed in the next Section.

Comparing the direct and the indirect approaches, we find that the indirect approach is superior over the direct approach from accuracy as well as stability point of view. we will discuss these issues later in this Chapter. As can be seen in equations (6.4.10), (6.4.11) and (6.4.12), the coefficient matrices of the second order differential equation (6.4.14) for *direct approach* are highly nonlinear and time-varying. It is thus difficult in general to prove the stability of this differential equation. Therefore, in the following discussion we will mainly focus on the *indirect approach*.

6.5 AVOIDING ALGORITHMIC SINGULARITIES: A DAMPED LEAST SQUARES APPROACH

Since the matrix K_t is always positive-definite, equation (6.4.15) can be written in the following form

$$([D_{21}\ D_{22}]P)\xi_p = K_t\gamma \qquad (6.5.1)$$

In order to solve for ξ_p in equation (6.5.1) after obtaining the vector γ, the inversion of the matrix $[D_{21}\ D_{22}]P \in \Re^{s \times (n-m)}$ must be performed explicitly, i.e.,

$$\xi_p = ([D_{21}\ D_{22}]P)^{-1}K_t\gamma \qquad \text{or} \qquad \xi_p = ([D_{21}\ D_{22}]P)^{\dagger}K_t\gamma \qquad (6.5.2)$$

However, the matrix $([D_{21}\ D_{22}]P)$ might be rank-deficient at certain points along a trajectory. Therefore, it is necessary to analyze this case, and find a way to overcome the problem caused by rank deficiency of $([D_{21}\ D_{22}]P)$.

Notice that the matrix $[D_{21}(q_l)\ D_{22}(q_l)]$ is a part of the manipulator's inertia matrix, and the matrix $P(q_l)$ results from the independent columns of the projection matrix $[I - J_e^{\dagger}(q_l)J_e(q_l)]$. Both of them are time-varying. When the manipulator moves along a prespecified trajectory, and its configuration evolves, these two matrices are updated continuously. At some point in the trajectory, the matrix $[D_{21}\ D_{22}]P$ becomes rank deficient. We call these singular points *algorithmic singularities* because they result from the specific algorithm that is used. As mentioned earlier, a similar problem has also been encountered when redundancy was utilized to minimize joint torques for rigid manipulators [10][17].

As shown in Chapter 2, the application of the *damped least-squares* technique to redundant manipulators is related mainly to the problem of singularity avoidance [16][31].

In the case of avoiding singularities due to rank deficiency of the Jacobian matrix, the basic idea in the damped least-squares method is to balance the cost of a large residual error (i.e., the end-effector tracking error) against the cost of a large solution by minimizing the objective function in (2.3.11). This idea can be modified and applied here in our flexible-joint redundant manipulator control scheme for avoiding algorithmic singularities. Details of the modifications and the derivations of the damped least-squares technique for our scheme are as follows.

In our case, corresponding to equation (2.3.11), the following objective function must be minimized for the system of equation (6.5.1):

$$\left\| \begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P \xi_p - K_t \gamma \right\|^2 + \alpha \| \xi_p \|^2 \qquad (6.5.3)$$

where $\alpha > 0$ is a *damping factor*. The sum in (6.5.3) can be written as

$$\left\| \begin{pmatrix} \begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P \\ \sqrt{\alpha} I \end{pmatrix} \xi_p - \begin{pmatrix} K_t \gamma \\ 0 \end{pmatrix} \right\|^2 \qquad (6.5.4)$$

Applying standard least-squares theory, the unique minimizer ξ_p is given by

$$\begin{aligned} \xi_p &= \begin{pmatrix} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P) \\ \sqrt{\alpha} I \end{pmatrix}^{\dagger} \begin{pmatrix} K_t \gamma \\ 0 \end{pmatrix} \\ &= \{ \alpha I + (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P) \}^{-1} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T K_t \gamma \end{aligned} \qquad (6.5.5)$$

Note that the matrix $(\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)$ is symmetric positive semi-definite while $\alpha > 0$ can be found such that $\{ \alpha I + (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P) \}$ is always positive definite. Thus, using equation (6.5.5), instead of using equation (6.5.2), we are able to avoid algorithmic singularities. Then, substituting ξ_p into the original controllers (6.4.1) and (6.4.2), we can control the manipulator so that the manipulator self-motion compensates for joint flexibility while avoiding algorithmic singularities.

Note that the damping factor α plays an important role in the damped least-squares formulation of equation (6.5.5). Large values of α result in good performance of singularity avoidance, but produce relatively large errors between the damped and undamped solutions for ξ_p in nonsingular regions. On the other hand, very small values of α will not give satisfactory performance for singularity avoidance, but result in accurate solutions for ξ_p in nonsingular regions. However, Nakamura et. al. [16] have proposed an *automatic adjustment* technique for α where the damping factor α is adjusted based on the *manipulability measure* [32], and the adjustment procedure is based on equations (2.3.14) and (2.3.15). In our algorithmic singularity avoidance scheme, proper values of α_o and h_o in equation (2.3.14) can be selected based on the value of $\left\| \begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P \right\|$ and manipulator kinematic parameters. Moreover, It can be seen in equation (6.5.5) that if α approaches zero, the damped least-squares pseudo-inverse formulation reduces to the original pseudo-inverse formulation of equation (6.5.2).

Although we have only applied the damped least-squares technique to the *indirect approach*, the same technique can also be applied to *direct approach* that we discussed in the last section. Notice that in equation (6.4.14), we need to invert the matrix $\Psi'_1 \in \Re^{s \times (n-m)}$. The matrix Ψ'_1 can be expressed as $\Psi'_1 = D_{mf} K_t^{-1} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)$. It is therefore possible for Ψ'_1 to be rank deficient because of the time-varying property of the matrices $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix}$ and P. Thus, technique similar to that discussed above can be applied in order to avoid algorithmic singularities in this *direct approach*.

6.6 REDUNDANCIES AND COMPENSATING CAPABILITY

The capability of using redundancy to compensate for joint flexibility largely depends on the degree of redundancy, and the number of flexible joints to be compensated for.

Hence, it is necessary to analyze the compensating capability of a redundant manipulator. Depending on the degree of redundancy and number of flexible joints, there are three different scenarios that can be identified:

(1) $n - m = s$, the number of degrees of redundancy is equal to the number of flexible joints. In this case, joint flexibility can in general be fully compensated for by all degrees of redundancy.

(2) $n - m > s$, the number of degrees of redundancy is greater than the number of flexible joints. In this case, joint flexibility can in general be fully compensated for using only some of the degrees of redundancy.

(3) $n - m < s$, the number of degrees of redundancy is smaller than the number of flexible joints. In this case, full joint flexibility compensation cannot be achieved even though all the degrees of redundancy are utilized.

In the following, we will give specific analysis of how the compensating capability for flexible joints affects the solution of equation (6.5.4) as well as the performance of the manipulator.

Scenario (1): *$n - m = s$*

In this scenario, $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P$ is a square matrix. The damped least-squares solution of ξ_p is

$$\xi_p = \{ \alpha I + (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P) \}^{-1} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T K_f \gamma \qquad (6.6.1)$$

Since the matrix $\{ \alpha I + (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P) \}$ is square and nonsingular, the solution for ξ_p given by (6.6.1) is unique. This uniqueness implies that all the degrees of redundancy must be utilized in order to compensate for all the flexible joints. In this scenario, the redundant manipulator is able to track the prespecified Cartesian trajectory with complete compensation of joint flexibility by self-motion.

The trade-off between the manipulator tracking performance and the algorithmic sin gularity avoidance capability depends on the priorities assigned by the designer. This assignment can be done by changing the value of the damping factor α (or α_o and h_o in equation (2.3.14) if the automatic adjustment technique is used). Note that this assignment of α for different priorities of tasks is also applicable to the following scenarios.

Scenario (2): $n - m > s$

In this case, the manipulator has more degrees of redundancy than the number of flexible joints. The matrix $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P \in \Re^{s \times (n-m)}$ is rectangular with more columns than rows. From (6.5.1) with rectangular $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P$, the general solution of ξ_p based on the *pseudo-inverse* approach is

$$\xi_p = (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} K_t \gamma + (I - (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d) \sigma \qquad (6.6.2)$$

where σ is an arbitrary vector that can be used to resolve the remaining redundancies after those for compensating for joint flexibility have been resolved. The matrix $(\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} \in \Re^{(n-m) \times s}$ is a *damped least-squares pseudo-inverse* of matrix $\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P$, and is given by

$$(\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} = (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^I \{\alpha I + (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)(\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)^T\}^{-1}, \qquad (6.6.3)$$

Note that in equation (6.6.2) the damped least-squares pseudo-inverse based general solution of ξ_p consists of the first term $(\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} K_t \gamma$ which corresponds to the minimum norm solution, and the second term $(I - (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d^{\dagger} (\begin{bmatrix} D_{21} & D_{22} \end{bmatrix} P)_d) \sigma$ which corresponds to the homogeneous solution. In this scenario, joint flexibility can be fully compensated for using only some of the degrees of redundancy, while ensuring end-effector Cartesian tracking. The remaining redundancies can be reso' ed by defining additional objective functions to determine the arbitrary vector σ.

Scenario (3): $n - m < s$

This scenario refers to the case where the number of flexible joints is greater than the number of degrees of redundancy. In this scenario, $[D_{21}\ D_{22}]P \in \Re^{s \times (n-m)}$ is a rectangular matrix with more rows than columns. The corresponding damped least-squares pseudo-inverse solution of ξ_p is written as

$$\xi_p = ([D_{21}\ D_{22}]P)_d^{\dagger} K_t \gamma. \tag{6.6.4}$$

where the damped least-squares pseudo-inverse of the matrix $[D_{21}\ D_{22}]P$ is defined as

$$([D_{21}\ D_{22}]P)_d^{\dagger} = \{\alpha I + ([D_{21}\ D_{22}]P)^T ([D_{21}\ D_{22}]P)\}^{-1} ([D_{21}\ D_{22}]P)^T \tag{6.6.5}$$

As can seen, of the s flexible joints, only $n-m$ can be compensated for by the $n-m$ degrees of redundancy. There are no extra degrees of redundancy available for compensating for the remaining flexible joints ($s-n+m$). This implies that the uncompensated flexibility at the joints will affect the tracking performance of the manipulator. This may even cause instability for the system eventually. In this case, it is advisable that appropriate considerations be given to flexible-joint effects in designing controllers for the flexible joints that are not compensated for by redundancies.

6.7 COMPUTATIONAL CONSIDERATIONS FOR HIGHER ORDER DERIVATIVES

Control of manipulators with flexible joints is always more difficult than that of manipulators with rigid joints. One of the reasons for this difficulty is that the controller requires higher order derivatives of the states. However, in the case when all the joints are flexible, as discussed in Chapter 5, we can compute the higher order derivatives recursively from lower order ones. This is a result of the special structure of the flexible-joint manipulator dynamics which can be seen in equations (5.7.4) and (5.7.5). In this chapter, we deal with the flexible-joint manipulator dynamic model that contains flexible as well as rigid joints. Unfortunately, this rigid/flexible coupled structure creates difficulty when it comes to calculating joint accelerations and jerks. To solve this problem, one possibility is that we could construct an observer to estimate the joint accelerations and jerks [27]. But then the observer needs to be reformulated in a way that it is suitable for the control strategy used in this chapter. However, this problem has motivated the investigation of other possible approaches for estimating the higher order derivatives for the rigid/flexible coupled system We propose an approach in this section that is essentially based on the estimation of the torque corresponding to the rigid joints. Details of this method are as follows.

In equation (6 ‾ 1), if only the link dynamics are considered, the link acceleration can be expressed as

$$\ddot{q}_l = \begin{bmatrix} D_{11} & D_{12} \\ D_{12}^T & D_{22} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} \tau_{l1} \\ K_l\,(q_m - q_{l2}) \end{bmatrix} - \begin{bmatrix} C_{l1} \\ C_{l2} \end{bmatrix} - \begin{bmatrix} G_{l1} \\ G_{l2} \end{bmatrix} \right\}. \qquad (6.7.1)$$

Obviously, we cannot directly feedback torque τ_{l1} in order to compute the link acceleration But we can estimate the rigid joint torque τ_{l1} using the computed torque control. Therefore, the estimated torque can be written as

$$\hat{\tau}_{l1} = \begin{bmatrix} D_{11} & D_{12} \end{bmatrix} \{ J_e^{\dagger} \, (\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl} e_x - \dot{J}_e \dot{q}_l \,) + (I - J_e^{\dagger} J_e)\, \xi \,\} + C_{l1} + G_{l1} \quad (6.7.2)$$

Furthermore, the estimated jerk can be calculated recursively based on differentiating (6.7.1) as

$$\hat{q}_l^{(3)} = \begin{bmatrix} D_{11} & D_{12} \\ D_{12}^T & D_{22} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} \dot{\hat{\tau}}_{l1} \\ K_l\,(\dot{q}_m - \dot{q}_{l2}) \end{bmatrix} - \begin{bmatrix} C_{l1} \\ C_{l2} \end{bmatrix} - \begin{bmatrix} G_{l1} \\ G_{l2} \end{bmatrix} - \begin{bmatrix} D_{11} & D_{12} \\ D_{12}^T & D_{22} \end{bmatrix} \hat{\ddot{q}}_l \right\} \qquad (6.7.3)$$

where the derivative of $\hat{\tau}_{l1}$ will be evaluated by differentiating the estimated control torque (6.7.2)

$$\begin{aligned}\dot{\hat{\tau}}_{l1} = {} & \left(\left[\dot{D}_{11}\ \dot{D}_{12}\right] - \left[D_{11}\ D_{12}\right] J_e^{\dagger}\dot{J}_e\right) \{ J_e^{\dagger}\,(\ddot{x}_d + \beta_{vl}\dot{e}_x + \beta_{pl}e_x - \dot{J}_e\dot{q}_l) \\ & + (I - J_e^{\dagger}J_e)\,\xi\ \} + \left[D_{11}\ D_{12}\right]\{ J_e^{\dagger}\,(x_d^{(3)} + \beta_{vl}\ddot{e}_x + \beta_{pl}\dot{e}_x - \ddot{J}_e\dot{q}_l - \dot{J}_e\hat{\ddot{q}}_l) \\ & + (I - J_e^{\dagger}J_e)\,\dot{\xi}\ \} + \dot{C}_{l1} + \dot{G}_{l1} \qquad (6.7.4)\end{aligned}$$

This procedure to estimate joint acceleration and jerk can be summarized in the following flow chart:

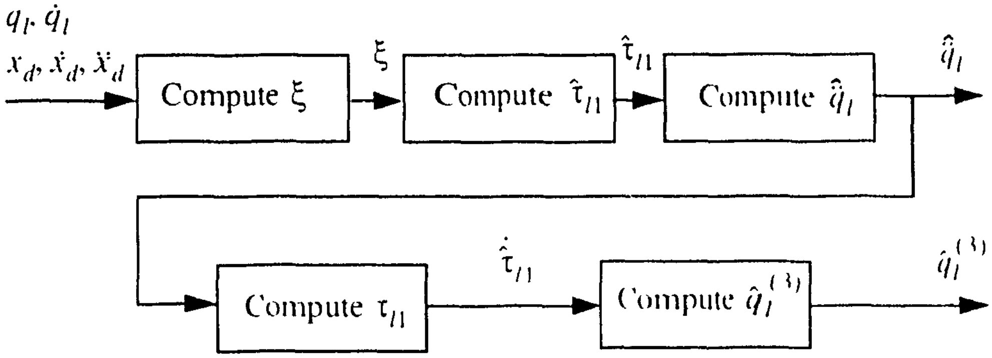

Figure 6.1 Estimation of joint acceleration and jerk

Theoretically speaking, real values of $\ddot{q}_l$ and $q_l^{(3)}$ could be obtained if torque τ_{l1} and its derivative $\dot{\tau}_{l1}$ were measurable. Now, instead of measuring τ_{l1} and $\dot{\tau}_{l1}$ we use the estimated values $\hat{\tau}_{l1}$ and $\dot{\hat{\tau}}_{l1}$. Assuming that the system dynamics do not change fast enough (or the sampling period Δt is chosen small enough), the errors between the real and estimated values for torque and its derivative will be kept within acceptable tolerances.

As far as the computational effort for calculating the estimated joint accelerations and jerks are concerned, obviously a large amount of computing effort is required. But what we gain is the saving in the expense of instrumentation to directly measure higher order derivatives, which is difficult or even impossible to accomplish with present technology.

Although the algorithm for estimating $\ddot{q}_l$ and $q_l^{(3)}$ is not realistic at present for real-time applications, it might be possible someday with much more powerful computing and real-time systems.

From the implementation point of view, it may be possible to avoid the heavy computational burden of estimating higher order derivatives (as discussed above) by using difference formulas to obtain these derivatives (joint accelerations and jerks) from position and velocity measurements. Moreover, the present limitations on sensing technology as well as the presence of noise in the measurements may make the estimation of the higher-order derivatives inaccurate. Also, the problem is made more difficult by the fact that the higher-order derivatives are more sensitive to variations of the parameters in the dynamic model. Therefore, further research is needed to find feasible and efficient means of dealing with higher-order derivatives before the control strategy proposed in this chapter can be implemented effectively.

6.8 STABILITY ANALYSIS OF THE CLOSED-LOOP SYSTEM

System stability for this coupled rigid/flexible-joint redundant manipulator can be proven using the Lyapunov function approach. Before showing system stability, let us obtain the corresponding closed-loop system equations.

Firstly, based on the dynamic model in (6.3.1), the link dynamics can be expressed as

$$\begin{bmatrix} D_{11}+D_{mm} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_{l1} \\ \ddot{q}_{l2} \end{bmatrix} + \begin{bmatrix} C_{l1}+G_{l1} \\ C_{l2}+G_{l2} \end{bmatrix} + \begin{bmatrix} 0 \\ K_t q_{l2} \end{bmatrix} = \begin{bmatrix} \tau_{l1} \\ K_t\,(q_{md}-e_m) \end{bmatrix} \qquad (6.8.1)$$

where q_{md} is the desired motor position, and e_m is the motor tracking error defined by $e_m = q_{md} - q_m$. The closed-loop equation corresponding to Cartesian tracking is formed by substituting the rigid link controller τ_{l1} in (6.4.1) and the link controller corresponding to the flexible joint q_{md} in (6.4.2) into (6.8.1). After some algebraic manipulations, the Cartesian tracking closed-loop equation can be written as

$$\ddot{e}_s + \beta_{vl}\dot{e}_s + \beta_{pl}e_s = J_e D_l^{-1}\begin{bmatrix} 0 \\ K_t e_m \end{bmatrix}. \qquad (6.8.2)$$

Secondly, the motor-level closed-loop equation can also be formulated in terms of the motor dynamic equation in (6.3.1), and the motor tracking controller (6.4.17). Notice that in the motor tracking controller (6.4.17), not only the motor tracking torque but also the dynamical relationship between the redundancy resolution vector ξ_p and the torsional force $K_t(q_{l2} - q_m)$ is included. It should be noted further that in (6.4.18) the term $(\ddot{\gamma} + K_{vm}\dot{\gamma} + K_{pm}\gamma)$ attempts to cancel out the torsional force $K_t(q_{l2} - q_m)$. Therefore, when we include the motor tracking controller, it also takes into account the dynamical relationship of the second-order differential equation in (6.4.18). Furthermore, stability of the second-order differential equation (6.4.18) depends very much on the motor tracking gain matrices K_{pm} and K_{vm}. Fortunately, these two matrices can be selected such that equation (6.4.18) as well as the motor tracking controller is stable. The motor tracking closed-loop equation can then be written as

$$\ddot{e}_m + K_{vm}\dot{e}_m + K_{pm}e_m = 0 \qquad (6.8.3)$$

where $e_m = q_{md} - q_m$ and $q_{md} = q^{\circ}{}_{md} + \gamma$. Now, we consider the two closed-loop equations (6.8.2) and (6.8.3) together. The equivalent state-space representation can be written as

$$\dot{\hat{Y}} = \Pi\hat{Y} \qquad (6.8.4)$$

where $\hat{Y} \in \Re^{2(m+s)}$ is defined as $\hat{Y} = \begin{bmatrix} e_s^T & \dot{e}_s^T & e_m^T & \dot{e}_m^T \end{bmatrix}^T$, and $\Pi \in \Re^{2(m+s) \times 2(m+s)}$ can be expressed as

$$\Pi = \begin{bmatrix} 0 & I_m & 0 & 0 \\ -\beta_{pl} & -\beta_{vl} & -J_e D_l^{-1}\begin{bmatrix} 0 \\ K_t e_m \end{bmatrix} & 0 \\ 0 & 0 & 0 & I_s \\ 0 & 0 & -K_{pm} & -K_{vm} \end{bmatrix} \tag{6.8.5}$$

where I_m and I_s denote identity matrices with dimensions $m \times m$ and $s \times s$ respectively. The system matrix Π possesses the same structure as the matrix A in (5.6.2) in Chapter 5. Hence, closed-loop stability of the coupled rigid/flexible-joint redundant manipulator with the control scheme of equation (6.8.4) can be shown using the Lyapunov approach as was done in Chapter 5.

6.9 NUMERICAL SIMULATIONS

In this Section, we apply the preceding control strategy to a model of the three DOF planar redundant manipulator shown in Figure 6.2 whose third joint is taken to be flexible, while the first and the second joints are assumed to be rigid. Each link has the same length $l_1 = l_2 = l_3 = 1m$, and the same mass $m_1 = m_2 = m_3 = 10kg$. The links are modelled with point mass at their distal ends. The motor inertia matrix is assumed to be $D_m = diag\{D_{mr}, D_{mf}\}$ with $D_{mr} = \begin{bmatrix} D_{mr_1} & 0 \\ 0 & D_{mr_2} \end{bmatrix}$ $(D_{mr_1} = 1kg,\ D_{mr_2} = 1kg)$ and $D_{mf} = 1kg$. It is also assumed that the joint stiffness constant $K_t = 100Nt$. The dynamic model of the entire manipulator is formed by combining the dynamics of the standard three DOF planar robot manipulator [23], and the joint flexibility for its third joint. The dynamic model possesses the same structure as in (6.3.1) with $n = 3$, $m = 2$ and $s = 1$. Having three DOFs, this planar manipulator is redundant when only the positioning of the end-effector is considered. This implies that there is an extra degree of freedom that the manipulator possesses when it performs a positioning task in two dimensional space. Therefore, an infinite number of joint trajectories will in general yield the same end-effector trajectory, and we can exploit this redundancy by imposing that the condition of compensating for the joint flexibility be performed at the same time as tracking. As in Chapter

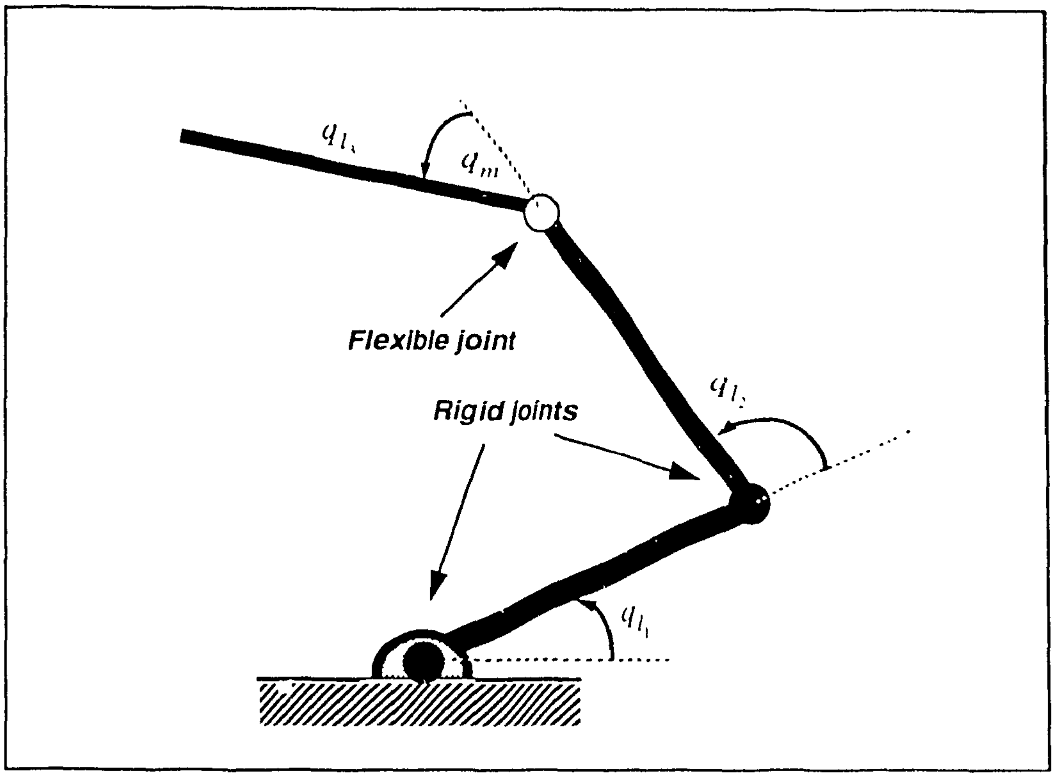

Figure 6.2 **Three DOF planar redundant rigid/flexible-joint manipulator**

5, due to the presence of joint flexibility, the dynamic model of the redundant manipulator results in stiff differential equations. In order to simulate the dynamic behavior of the manipulator, the stiff differential equations are solved numerically using a seventh/eighth order Runge-Kutta method. The results obtained are satisfactory with respect to the relatively flexible stiffness $K_l = 100Nt$. In addition, we must also solve the second-order differential equation (6.4.18) to obtain the variable γ from which we calculate the redundancy resolution vector ξ_p. Some simple but fast methods can be used here. For example, *trapezoidal integration* or *low-order predictor-corrector* integration method are appropriate for this differential equation. In the simulations, a *low-order predictor-corrector* integration method was used for the purpose of solving variable γ.

The proposed algorithm together with the dynamic model were coded in MATLAB and inplemented on a SUN/SPARC-2 workstation. To ensure stability of the rigid/flexible manipulator system, the gain matrices corresponding to the Cartesian tracking controller as well as the motor tracking controller were adjusted such that the Cartesian space and the motor tracking controllers were stable. Notice that both the rigid-joint controller and flexible-joint controller contain the Cartesian tracking controller. For this Cartesian controller, the gain matrices β_{pl} and β_{vl} were selected as $\beta_{pl} = diag\{700, 700\}$ and $\beta_{vl} = diag\{135, 135\}$ to ensure proper Cartesian tracking. At the same time, the motor tracking controller was also designed with typical gain values $K_{pm} = 730$ and $K_{vm} = 90$ to guarantee that the motor in the third joint tracks its "desired" trajectory.

To show the applicability of the proposed control scheme, Simulations were carried out using the three DOF planar redundant manipulator with two rigid joints and one flexible joint. Numerous Cartesian trajectories were tested by simulation. Some of the typical simulation results are shown here to demonstrate the use of the manipulator's self-motion to compensate for joint flexibility while ensuring that the end-effector tracks a Cartesian trajectory. The simulation results show the three DOF manipulator tracking a Cartesian space straight line trajectory while the one degree of redundancy compensates for the flexibility in one joint. The manipulator was initially at rest with $q(0) = \begin{bmatrix} 60^\circ & -120^\circ & 60^\circ \end{bmatrix}^T$ which corresponds to $x(0) = \begin{bmatrix} 2 & 0 \end{bmatrix}^T$ in Cartesian space. As shown in Figure 6.3, the manipulator end-effector tracks a straight line trajectory and finally reaches the goal point $x(t_f) = \begin{bmatrix} 0.0 & -1.0 \end{bmatrix}^T$. The manipulator's Cartesian tracking performance as well as the evolution of the configurations are illustrated in Figure 6.3. Besides, the manipulator's end-effector Cartesian position and velocity tracking errors are shown in Figures 6.4 to 6.7 respectively. Figures 6.8 to 6.13 illustrate manipulator's joint position and velocity profiles, while Figures 6.15 to 6.17 show the joint-space position tracking errors. For the flexible third joint, its motor tracking error is shown in Figure 14. To demonstrate the compensation for joint flexibility by the self-motion of the manipulator, we plot the curve of the quantity $q_m - q_{l2}$ in Figure 6.18. It can seen that the quantity $q_m - q_{l2}$ is controlled close to zero by the self-motion of the manipulator. This directly implies that the torsional force $K_t (q_{l2} - q_m)$ is also controlled to zero due to self-motion. This, therefore, verifies the theoretical analysis in this Chapter. Finally, in Figures 6.19-6.21, the control torque profiles are shown.

Remark 1: In Figure 6.3, we find that the manipulator's posture varies in a way that involves both rotation and translation when it is in the initial oscillation period. This should be compared with the results shown in Chapter 5 (Figure 5.9) where the manipulator's posture changes almost "in parallel" one configuration after another. This difference is obvious because in the case of compensation for joint flexibility, when the end-effector oscillates about the nominal trajectory or deviates from it, the error driven controller generates certain amount of control torque to reduce the error. This torque affects the torsional force in the flexible joint. Therefore, due to the special structure of the redundancy resolution vector ξ_p, the manipulator uses its self-motion to reconfigure itself such that the torsional force $K_t (q_{l2} - q_m)$ approaches zero. This is the reason why we see a fair amount of link movement in the initial stages of tracking in Figure 6.3. On the other hand, in the case of other redundancy resolution objective functions, for example, minimum joint acceleration as shown in Chapter 5, the deviation of the end-effector does not directly relate to the self-motion of the manipulator. That is why we do not see much self-motion when manipulator's end-effector has small oscillations in the early stage of tracking the straight-line in Figure 5.9.

Remark 2: In the Cartesian and motor tracking controllers, some damping has been added via the gain matrices. However, some oscillatory behavior can still be observed in the joint position and velocity profiles. This oscillation results first from the joint flexibility of the manipulator, and second from the way the redundancy is resolved. As mentioned in Remark 1, the manipulator must be reconfigured such that the torsional force vanishes. This reconfiguration creates another torque which affects the torsional force again

Although this process converges, it produces more oscillations than in those cases where the redundancy resolution objective functions are selected for functions other than compensation of joint flexibility. Therefore, more damping is needed to damp out these oscillations in this case.

6.10 CONCLUDING REMARKS

In this chapter, an unified theoretical framework has been presented for control of redundant flexible-joint manipulators whose redundancy is resolved to compensate for joint flexibility. A novel nonlinear control strategy for the rigid/flexible-joint coupled redundant manipulator system has been proposed. A Cartesian-space controller has been constructed in order to control the redundant manipulator in Cartesian space. The redundancy resolution vector ξ_p has been derived so as to enable us to resolve redundancy for the purpose of compensating for the joint flexibility. Moreover, the problem of possible algorithmic singularities has also been discussed, and the modified damped least squares approach has been incorporated to avoid numerical difficulties due to these singularities. Because of the special structure of the rigid/flexible-joint coupled system, not only the position and velocity information but also the higher-order derivatives are required in order to construct the controller. Therefore, in Section 6.7 a scheme to estimate the higher-order derivatives was developed. It was shown that stability of the proposed control strategy can be established using a Lyapunov function approach. Finally, Numerical simulations were given to demonstrate the main results presented in this chapter.

Finally, it should be noted that the nonlinear control strategy for rigid/flexible-joint coupled redundant manipulators presented in this chapter can be used as a basis for developing other advanced control strategies: robust control, adaptive control, etc..

6.11 REFERENCES

[1] S. Arimoto and F. Miyazaki, "Stability and robustness of PID feedback control for

robot manipulators of sensory capability," in *Robotics Research. The First Int. Symp.*, ed. M. Brady and R. Paul (MIT Press), pp. 783-799, 1984.

[2] S. Arimoto and F. Miyazaki, "Stability and robustness of PD feedback control with gravity compensation for robot manipulator," *ASME Winter Meeting*, Anaheim, CA, pp 67-72, Dec. 1986.

[3] Baillieul, "Kinematic programming alternatives for redundant manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 722-728, St. Louis, MO, 1985.

[4] J. Baillieul, "Kinematic redundancy and the control of robots with flexible components," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 715-721, Nice, France, May 1992.

[5] J. Baillieul, "Kinematic redundancy and the control of robots with flexible components," *IEEE Control Systems Magazine*, vol. 13, no. 1, pp. 15-21, Feb. 1993.

[6] J.J. Craig, P. Hsu and S. Sastry, "Adaptive control of mechanical manipulators," *Int J. of Robot. Res.*, vol. 6, no. 2, pp. 16-28, 1987.

[7] A. De Luca, "Dynamic control properties of robot arms with joint elasticity," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1574-1580, Philadelphia, PA., 1988.

[8] O. Egeland, "Cartesian control of a hydraulic redundant manipulator," *Proc IEEE Int. Conf. on Robotics and Automat.*, pp. 2081-2086, Raleigh, NC, April 1987.

[9] E. Freund, "Fast nonlinear control with arbitrary pole-placement for industrial robots and manipulators," *Int. J. Robotics Res.*, vol. 1, no. 1, pp. 65-78, 1982.

[10] J.M. Hollerbach and K.C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE Int. J. of Robot. and Automat* , vol. 3, pp. 308-315, 1987.

[11] O. Khatib, "A unified approach for motion and force control of robot manipulators," *IEEE J. of Robotics and Automat.*, vol. 3, pp. 43-53, 1987.

[12] K. Khorasani, "Adaptive control of flexible-joint robots," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 2, pp. 250-267, April 1992.

[13] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Trans. Automat. Contr.*, vol. 37, no. 2, pp. 174-181, Feb. 1992.

[14] J.Y.S. Luh, M.W. Walker and R.P. Paul, "Resolved-acceleration control of mechanical manipulators," *IEEE Trans. Auto. Contr.*, vol. AC-25, no. 3, 1980.

[15] A.A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robotics Res.*, vol. 4, no. 3, pp. 109-117, 1985.

[16] Y. Nakanura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *J. Dyn. Syst., Meas., Contr.*, vol. 108, no. 3, pp. 163-171, 1986.

[17] D.N. Nenchev, "Redundancy resolution through local optimization: A review," *J. of Robotic Systems*, vol. 6, no. 6, pp. 769-798, 1989.

[18] L.A. Nguyen, I.D. Walker and R. DeFigueiredo, "Control of flexible, kinematically redundant robot manipulators," *SPIE Cooperative Intelli. Robot. in Space*, vol. 1387, pp. 296-312, 1990.

[19] L.A. Nguyen, I.D. Walker and R. DeFigueiredo, "Dynamic control of flexible, kine matically redundant robot manipulators," *IEEE Trans. on Robotics and Automat.*, vol. 8, no. 6, Dec. 1992.

[20] L.E. Pfeffer, O. Khatib and J. Hake, "Joint torque sensory feedback in the control of a PUMA manipulator," *IEEE J. Robotics and Automat.*, vol. RA-5, pp. 418-425, Aug. 1989.

[21] E.I. Riven, *Mechanical Design of Robots*, McGraw Hill, 1988.

[22] H. Seraji, "Configuration control of redundant manipulators: theory and implementation," *IEEE Trans. Robotics Automat.*, vol. 5, no. 4, pp. 472-490, Aug. 1989.

[23] H. Seraji and R. Colbaugh, "Improved configuration control for redundant robots," *J. Robot. syst.*, vol. 7(6), pp. 897-928, 1990.

[24] J.J. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. of*

Robot. Res., vol. 6, no. 3, pp. 49-59, 1987.

[25] J.J. Slotine and S.S. Sastry, "Tracking control of nonlinear systems using sliding surfaces with application to robot manipulators," *Int. J. Contr.*, vol. 38, pp. 465-492, 1983.

[26] M.W. Spong, "Modeling and control of elastic joint robots," *ASME J. Dyn. Syst. Meas. and Cont.*, vol. 109, pp. 310-319, 1987.

[27] M.W. Spong, "Control of flexible joint robots: A survey," *Coordinated Science Laboratory, Technical Report*, UILU-ENG-90-2203 DC-116, University of Illinois at Urbana-Champaign, Feb. 1990.

[28] M.W. Spong, K. Khorasani and P.V. Kokotovic, "An integral manifold approach to the feedback control of flexible joint robots," *IEEE J. of Robot. and Automat.*, vol. RA-3, no. 4, pp. 291-300, Aug. 1987.

[29] L.M. Sweet and M.C. Good, "Re-definition of the robot control problem," *IEEE Control System Mag.*, vol. 5, no. 3, pp.18-25, 1985.

[30] P. Tomei, "A simple PD controller for robots with elastic joints," *IEEE Trans. Automat. Contr.* vol. 36, no. 10, pp. 1208-1213, Oct. 1991

[31] C.W. Wampler, "Manipulator inverse kinematics solutions based on vector formulations and damped least-squares methods," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 93-101, 1986.

[32] T. Yoshikawa, "Manipulability and redundancy control of robotic mechanisms," in *Proc. IEEE Int. Conf. on Robotics and Automat.*, St. Louis, Missouri, pp. 1004-1009, 1985.

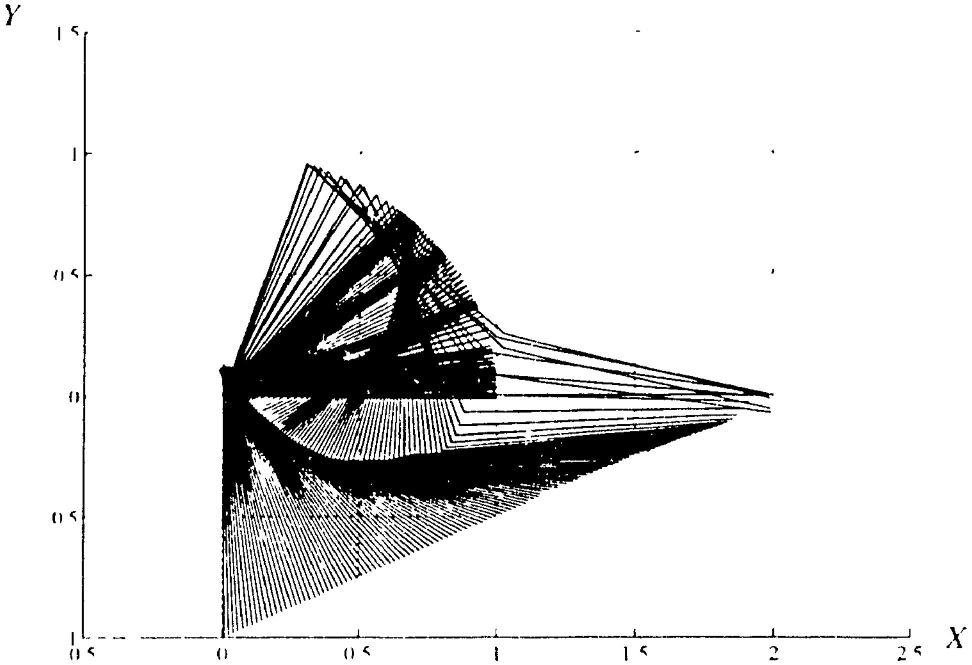

Figure 6 3 **Cartesian straight line tracking**

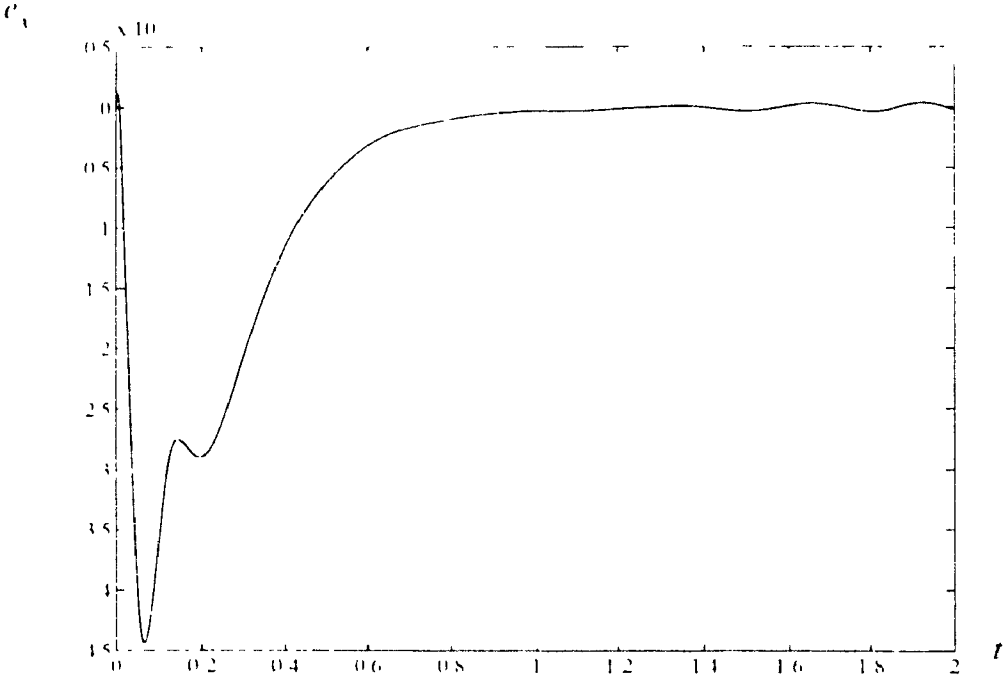

Figure 6 4 **position tracking error in X direction**

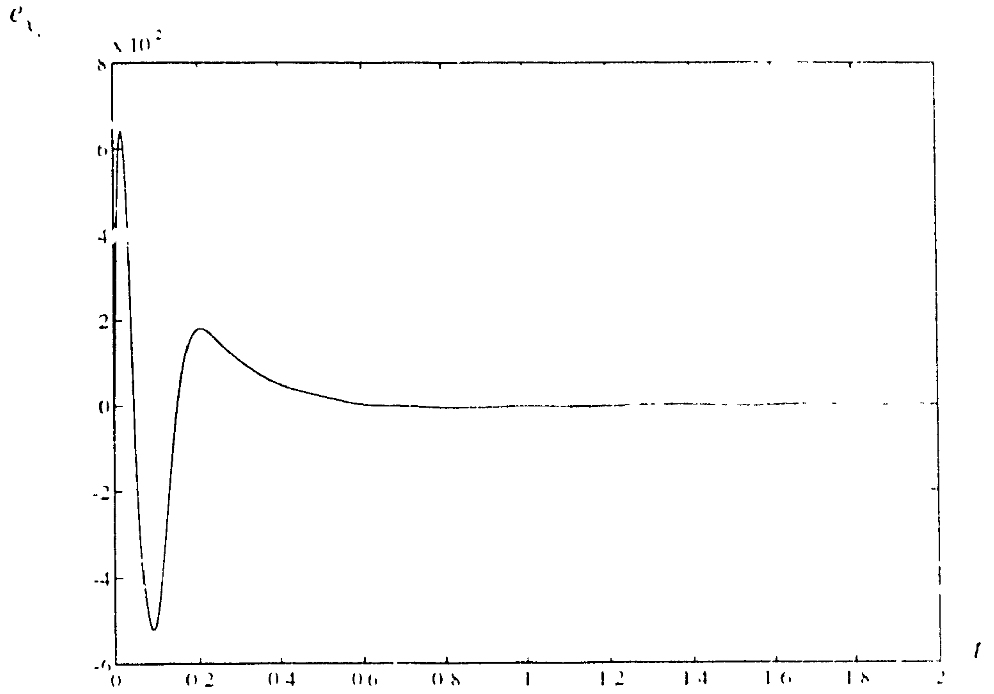

Figure 6.5 **Position tracking error in Y direction**

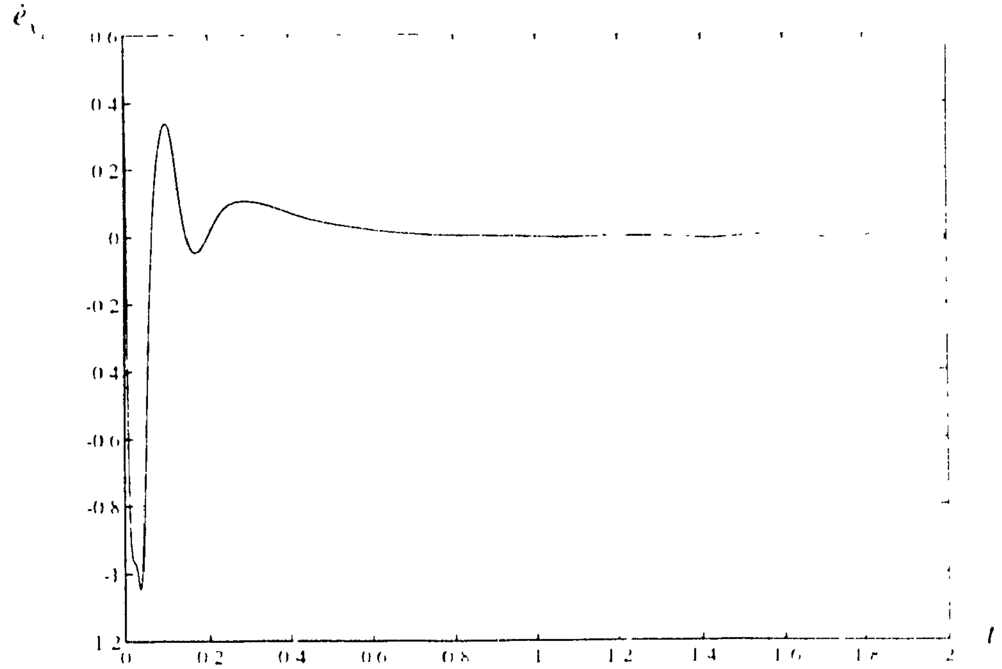

Figure 6 6 **Velocity tracking error in X direction**

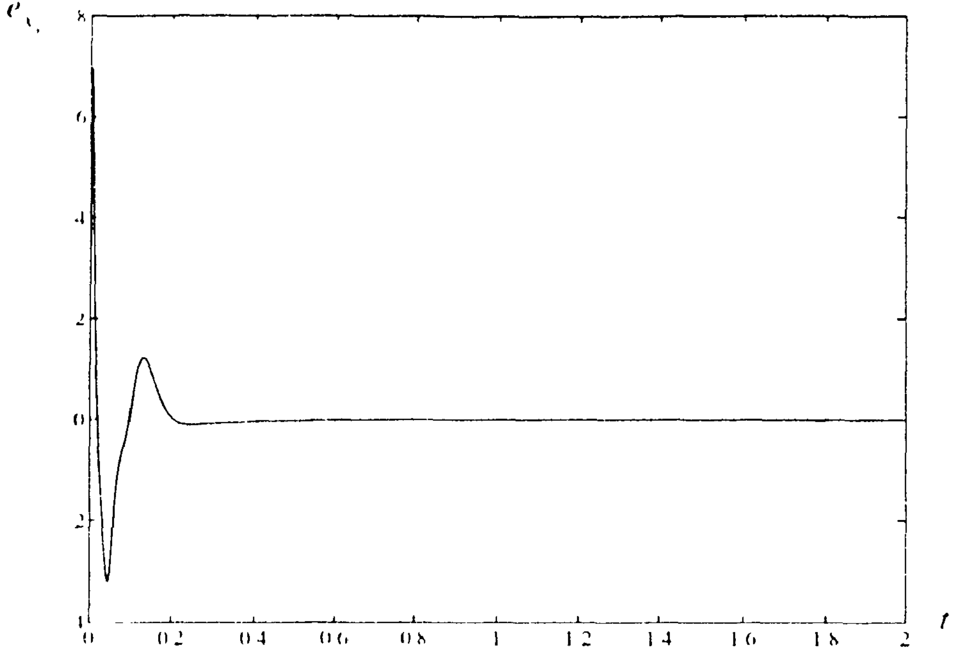

Figure 6 7 **Velocity tracking error in Y direction**

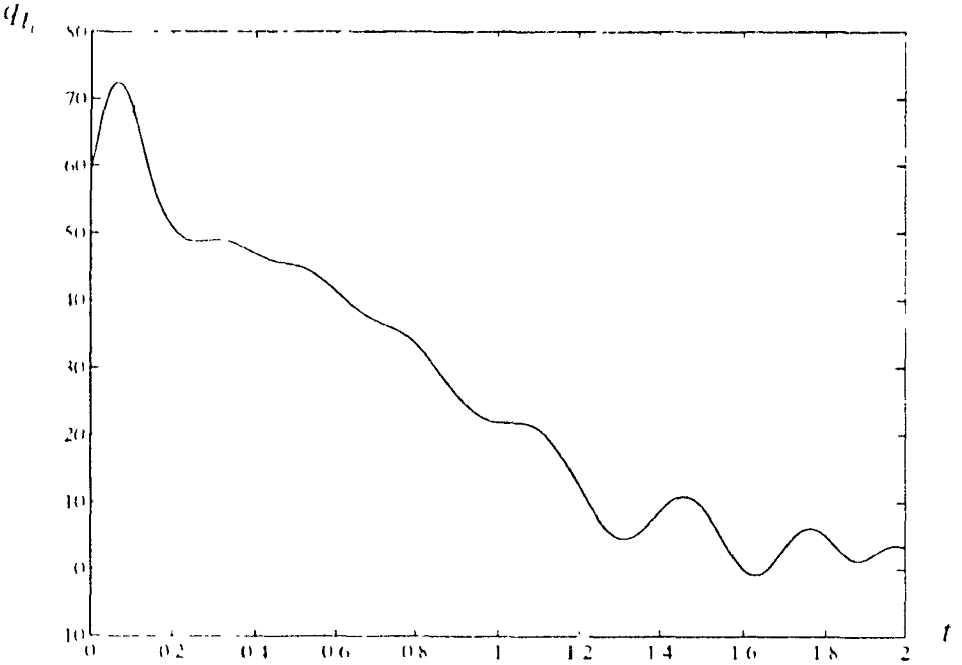

Figure 6 8 **The actual trajectory for the first link**

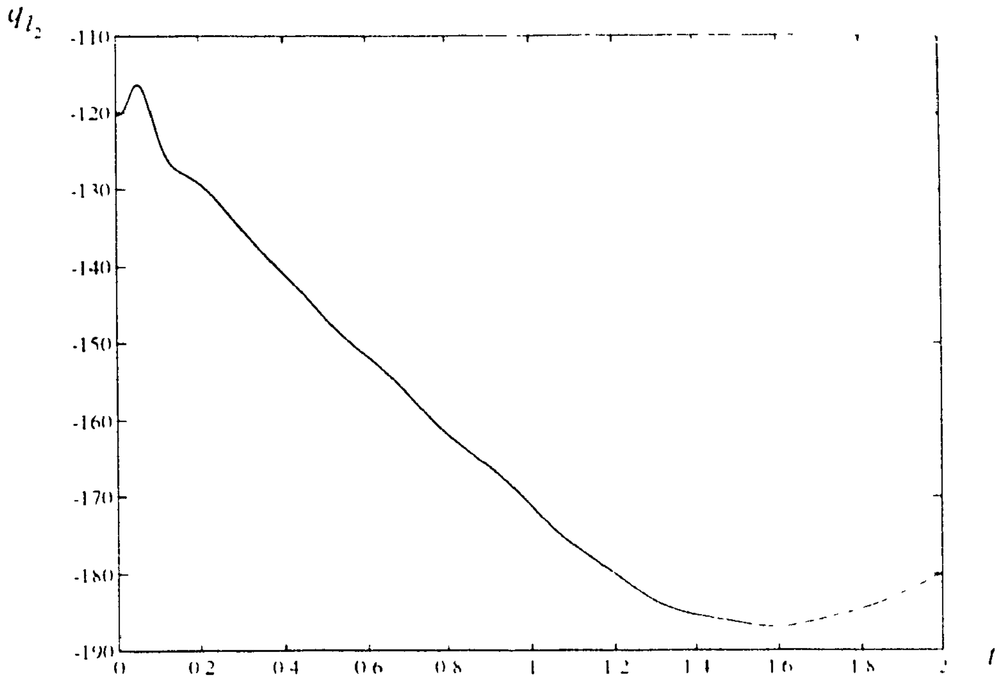

Figure 6 9 The actual trajectory for the second link

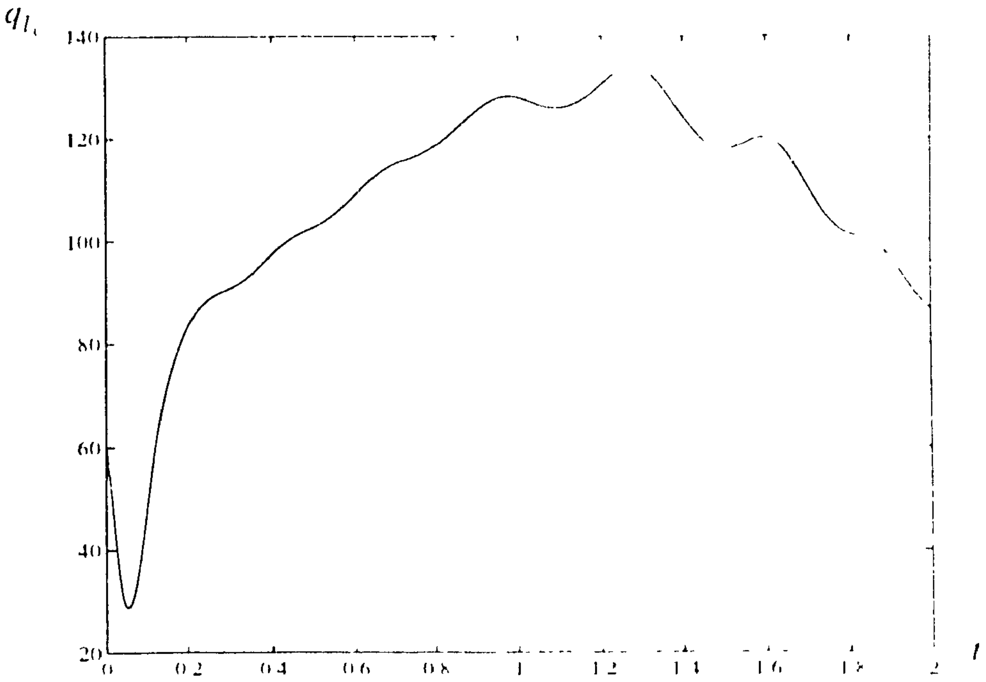

Figure 6 10 The actual trajectory for the third link

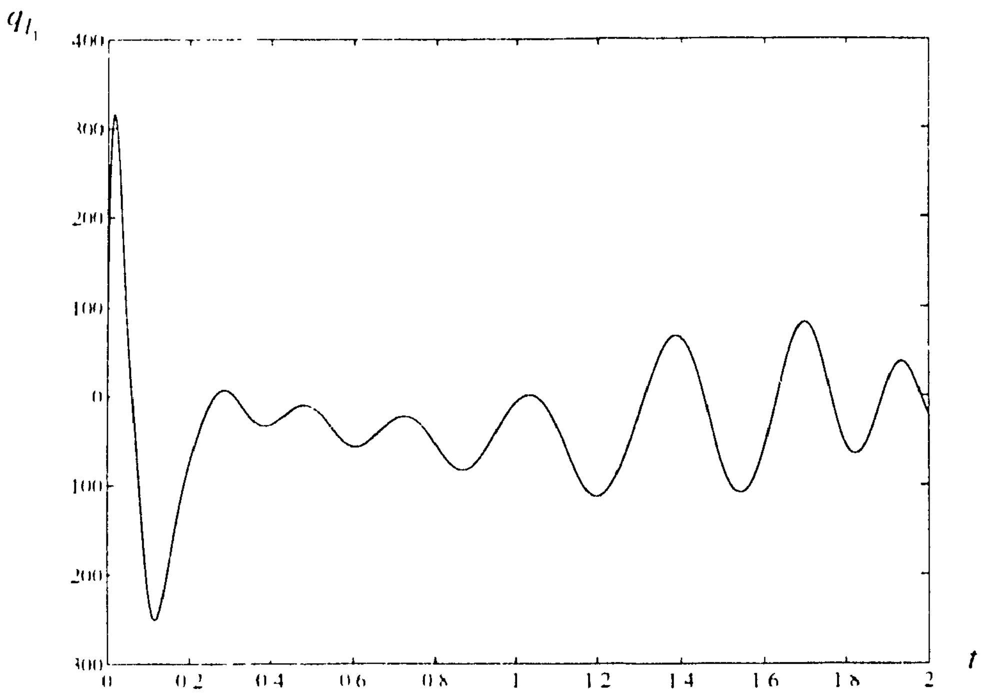

Figure 6 11 **The actual velocity trajectory for the first link**

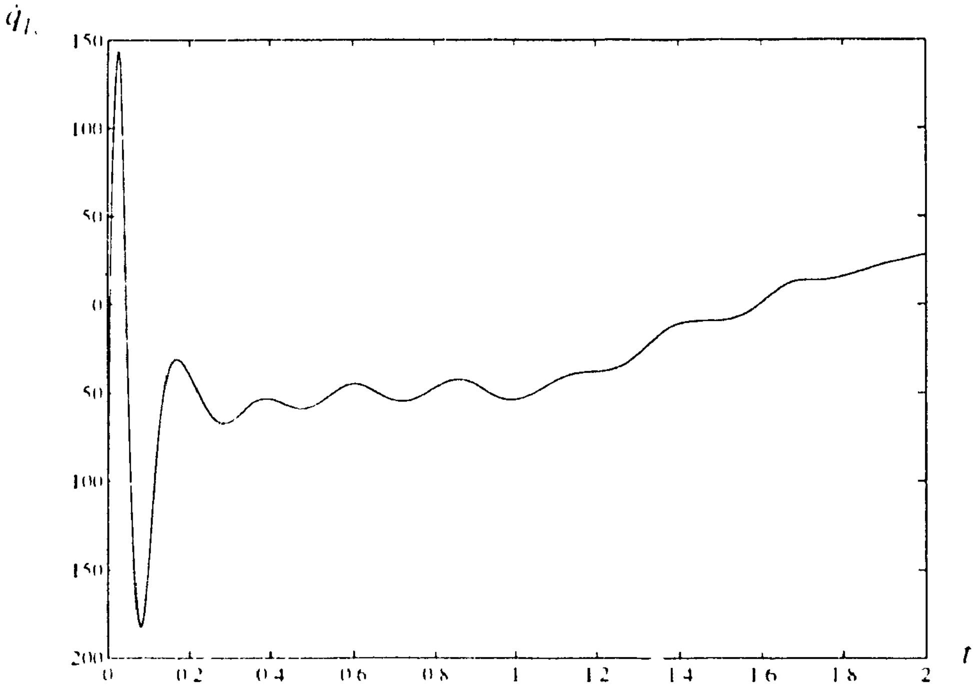

Figure 6 12 **The actual velocity trajectory for the second link**

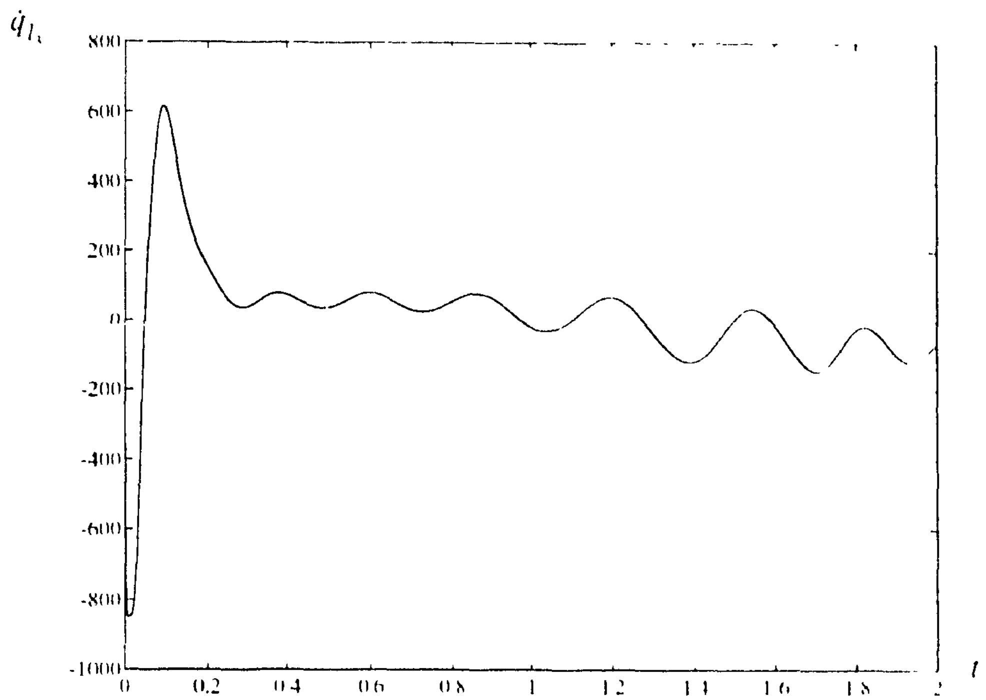

Figure 6.13 **The actual velocity trajectory for the third link**

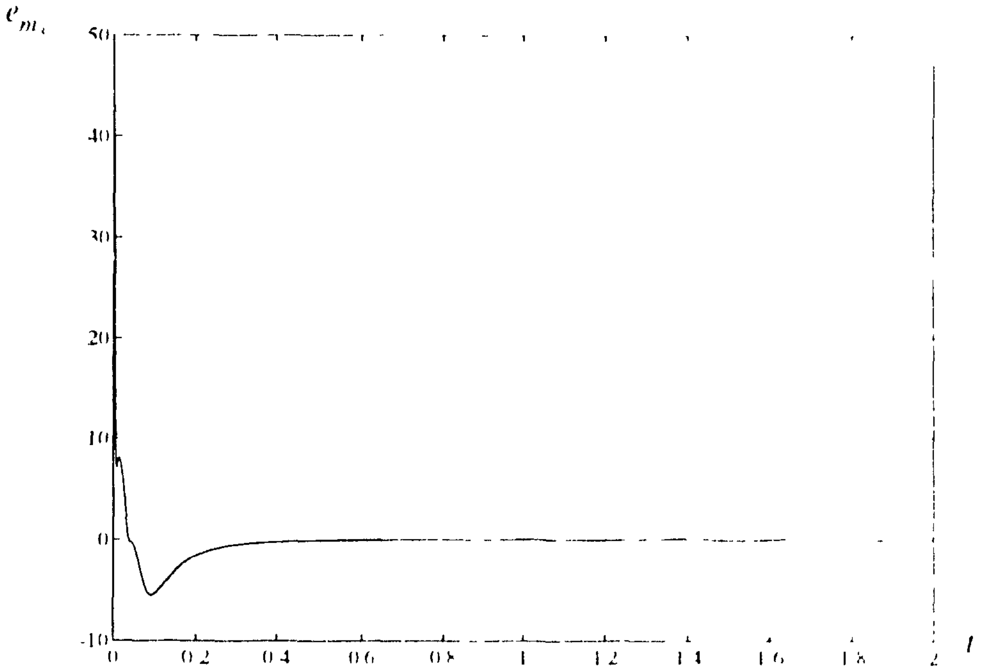

Figure 6 14 **The third motor tracking error**

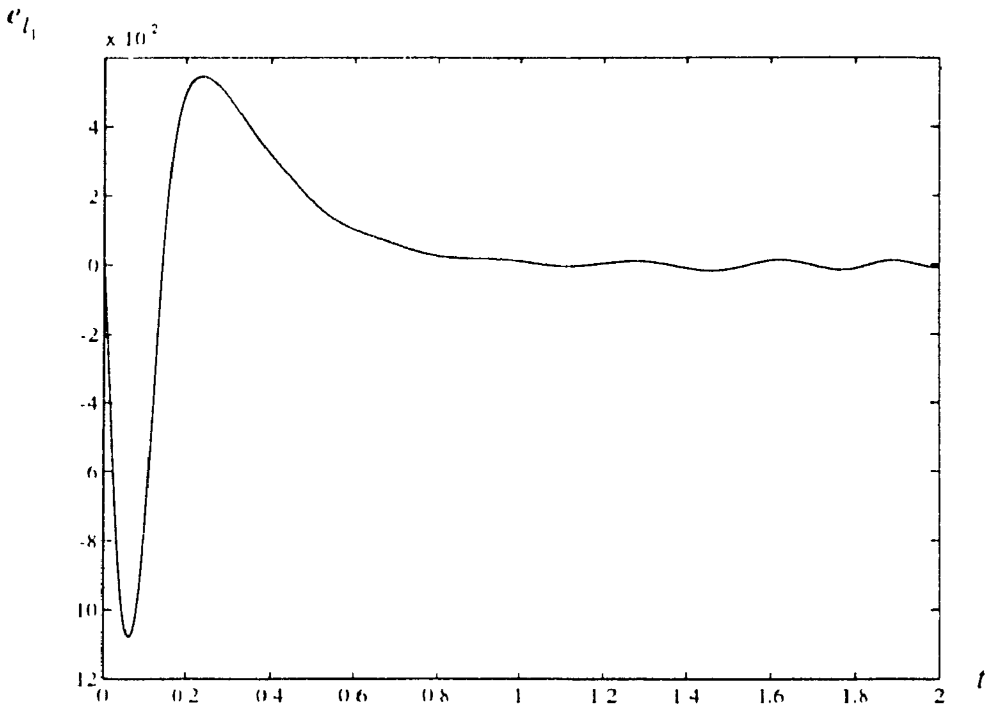

Figure 6 15 **The first link tracking error**

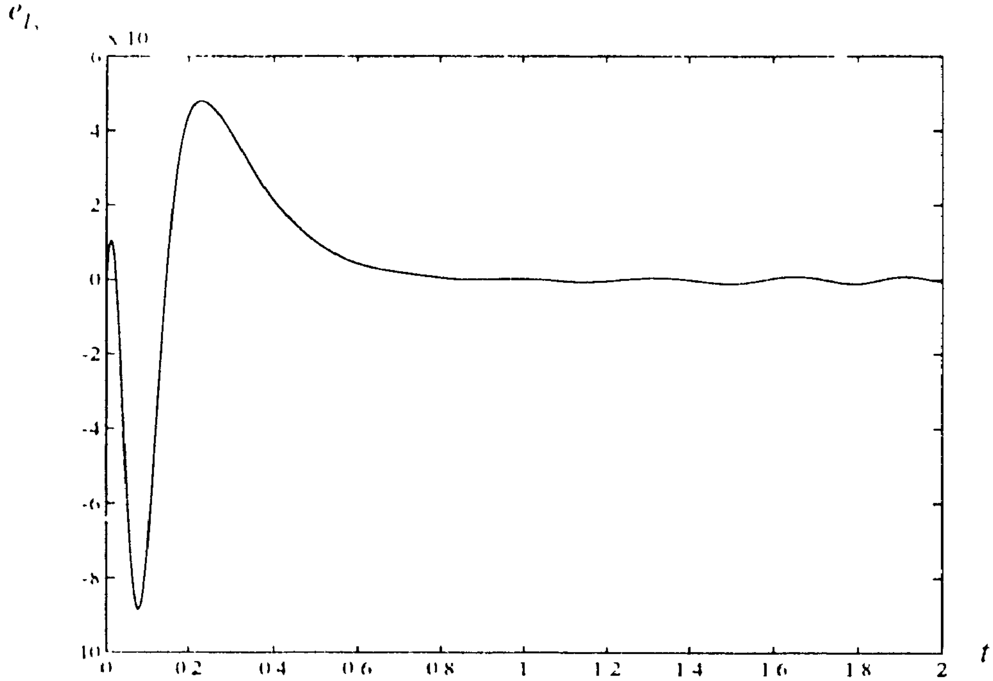

Figure 6 16 **The second link tracking error**

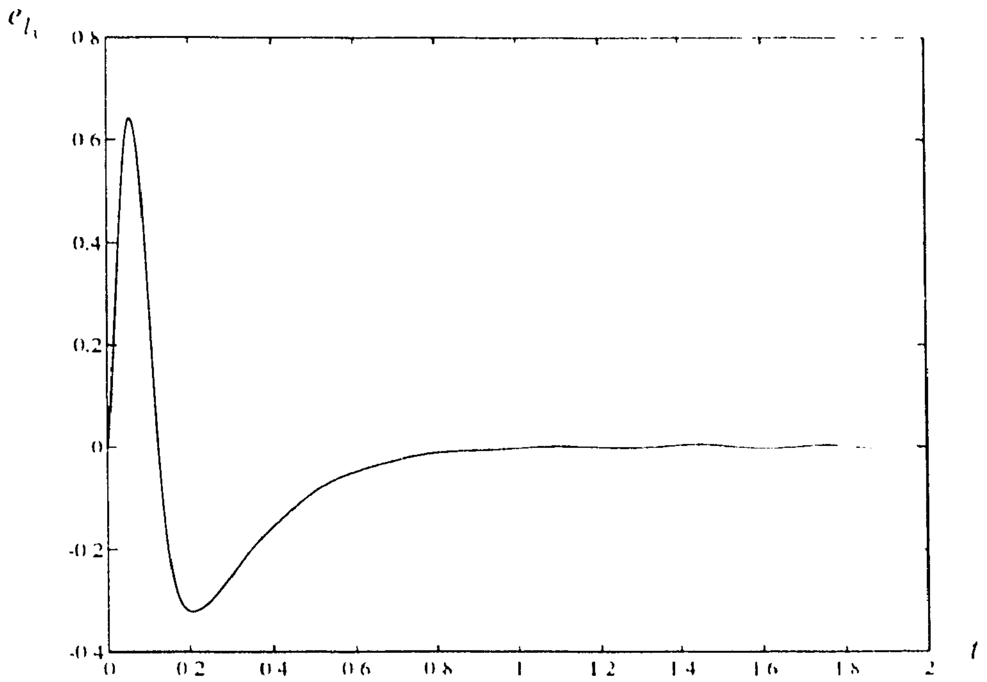

Figure 6.17 **The third link tracking error**

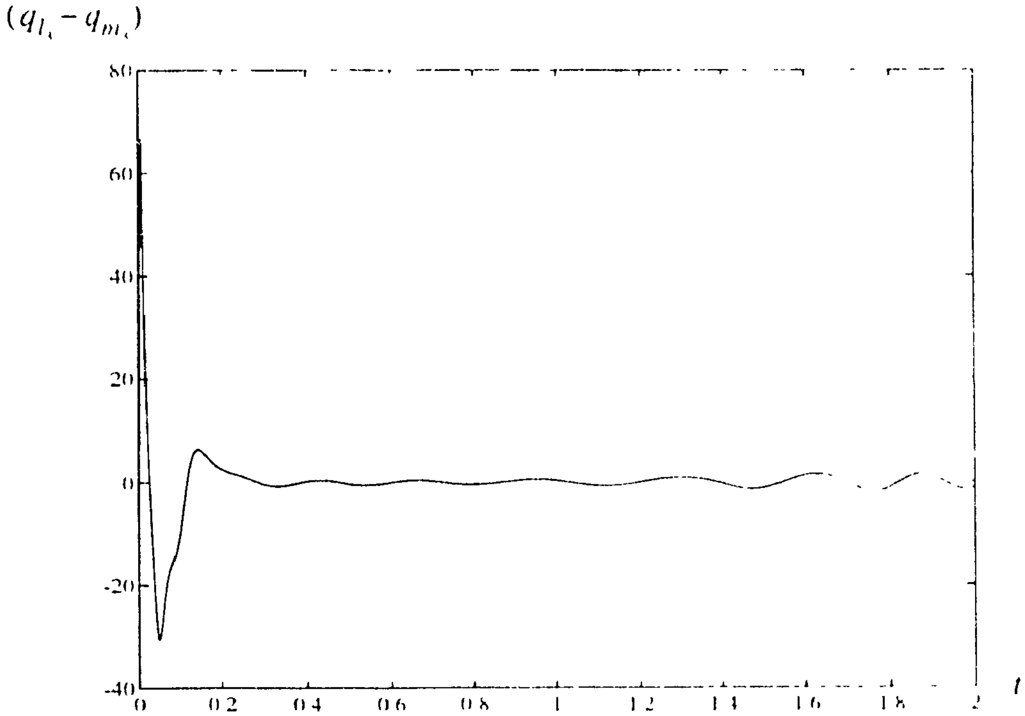

Figure 6.18 **The error between the third motor and the third link**

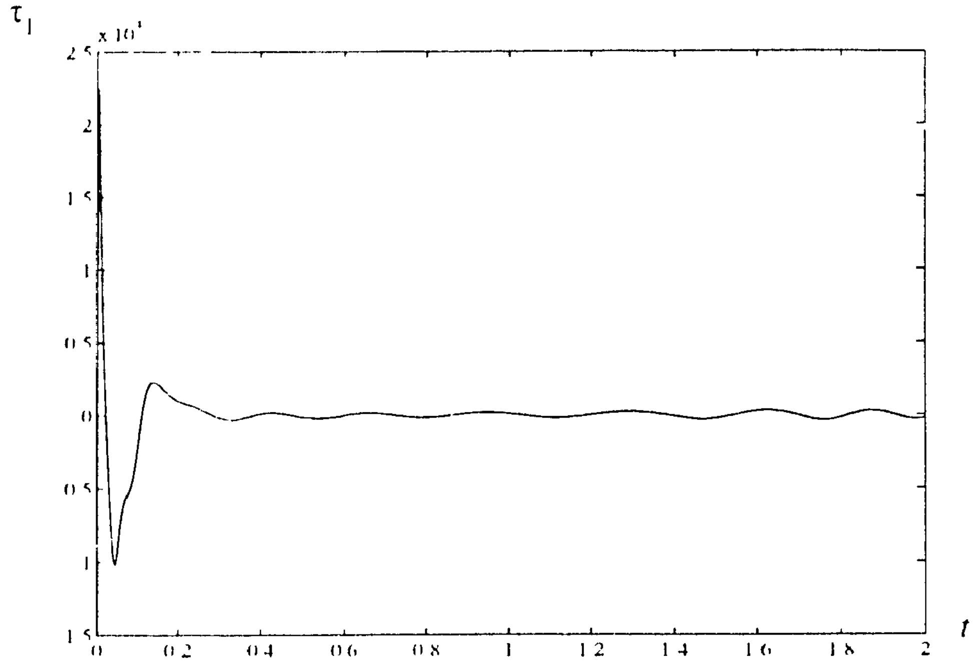

Figure 6 19 **The control torque for the first joint**

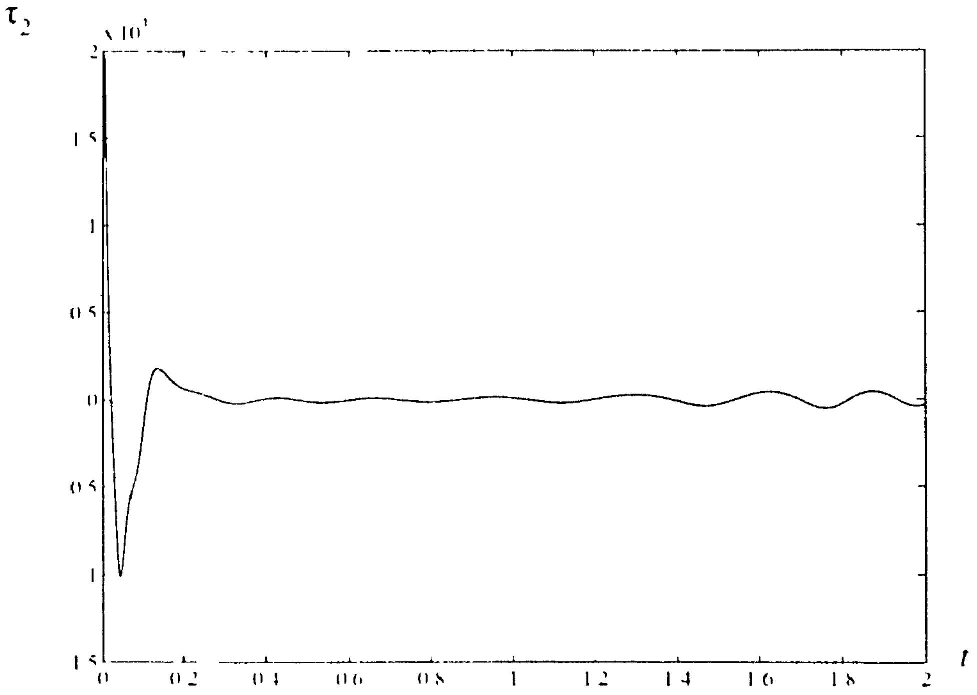

Figure 6 20 **The control torque for the second joint**

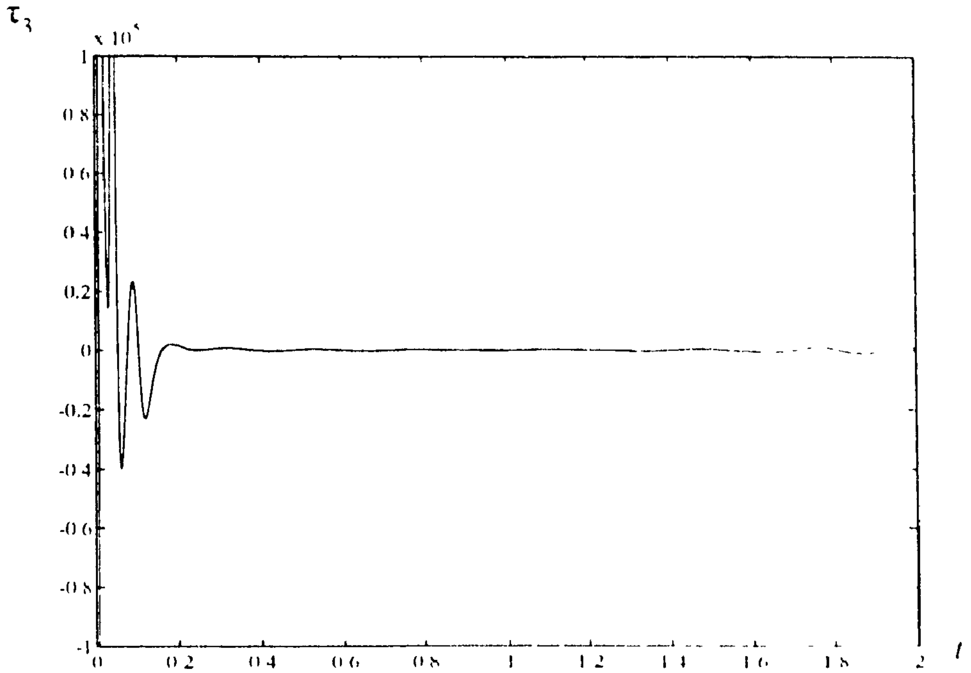

Figure 6.21 **The control torque for the third joint**

CHAPTER 7 CONCLUSIONS AND FUTURE RESEARCH

7.1 CONCLUSIONS

In this thesis, various issues of dynamic control of kinematically redundant manipulators have been addressed. In particular, within the framework of kinematically redundant manipulator control, three main contributions have been made:

(1) Development of an impedance control based redundant manipulator control scheme for collision impact minimization.

(2) Design of a hybrid Cartesian-joint controller for redundant flexible-joint manipulators.

(3) Development of a Cartesian space based control strategy for rigid/flexible-joint redundant manipulators where the redundancy is utilized to compensate for joint flexibility.

Briefly, these contributions are summarized below.

7.1.1 Impact Control for Redundant Manipulators

The problem of controlling redundant manipulators in order to reduce the effect of collision impacts has been solved using an augmented kinematics and an impedance control approach. The solution to this problem is achieved by minimizing the magnitudes of impulsive forces, and reducing rebound effects of the end-effector. Following the augmented kinematics approach, the manipulator Jacobian matrix was augmented in terms of an impact dynamic model, so that the resulting configuration of the redundant manipulator produced the smallest amount of impulsive force in the end-effector at the time of impact. Furthermore, in order to reduce the impulsive forces and the rebound effects, a simplified impedance control scheme was designed by setting the inverse of the desired inertia matrix to be identical to the mobility tensor of the manipulator in Cartesian space.

The proposed controller has a modular structure and, therefore, other redundancy resolution methods can also be incorporated into the formulation of this control strategy. For example, the pseudo-inverse approach [4] can be used such that the arbitrary vector is utilized for impact minimization.

The problem of manipulator impact control is a complex one whose solution depends not only on theoretical analysis but also on experimental experience. This problem is far from completely solved, and further work is needed in order to develop a scheme suitable for real-time practical applications.

7.1.2 Cartesian Control of Redundant Flexible-joint Manipulators

The Cartesian control scheme for flexible-joint redundant manipulators that has been proposed in this thesis is a new contribution A novel controller called the hybrid Cartesian-joint controller has been formulated. This consists of a Cartesian tracking controller, a link tracking controller, and a motor tracking controller. This hybrid controller ensures not only Cartesian trajectory tracking but also link and motor motions, which in turn ensure proper self-motion of the manipulator and reject disturbance due to joint flexibility.

The dynamic model of the flexible-joint redundant manipulator that we have used for analysis as well as for computer simulations is a simplified model where the nonlinear coupling between the motors and the links is represented by simple torsional springs with constant stiffnesses. However, as mentioned in Chapter 3 this motor-link coupling is a nonlinear, time-varying function. With a small change in formulation, the Cartesian control scheme proposed in Chapter 5 can be extended to the case where the coupling between motors and links is nonlinear and time-varying

7.1.3 Control of Redundant Manipulators to Compensate for Joint Flexibility

A new application of kinematically redundant manipulators has been proposed in this thesis, namely, that of using redundancy resolution to compensate for joint flexibility. This redundancy resolution scheme has been incorporated in a control strategy for redundant flexible-joint manipulators. The extension of this strategy to rigid/flexible-joint coupled redundant manipulators to our knowledge, appears here for the first time in the published literature. The basic idea in this approach is the use of redundancy to compensate for joint flexibility. Moreover, the problem of possible algorithmic singularities was analyzed, and a scheme was proposed which makes the controller robust with respect to such singularities.

7.2 SUGGESTIONS AND FUTURE RESEARCH

Kinematically redundant manipulators and corresponding control problems are very active research areas in robotics. The issues tackled in this thesis are relatively new topics in robotics. There are many interesting issues arising from the research work described in Chapters 4-6 of this thesis for which further systematic research is needed. It is also worthwhile extending the ideas and algorithms developed in this thesis to other problems of similar type in robotics. Some suggestions and ideas for future work are as follows:

(a) In Chapter 4, we have considered only the minimum impact problem when we applied the augmented kinematics approach for optimization of the kinematic/dynamic objective function $L(q, D)$. In fact, this problem can be extended to the case where the issues of minimum impact and manipulator link collision avoidance are considered at the same time. This can be achieved by formulating the optimization problem as a constrained optimization scheme where the constraints are derived based on geometric conditions between manipulator links and workspace objects.

(b) The impact problem for rigid-link and rigid-joint manipulators was discussed in Chapter 4. This can be extended to the case where a manipulator has rigid links but flexible joints, or rigid joints but flexible links, or even both flexible joints and flexible links. We may expect a manipulator to receive smaller impulsive forces when the flexible components are present in its structure. However, it is possible that impacts can excite higher frequency modes, and thus result in oscillatory or unstable behavior.

(c) An adaptive scheme for Cartesian control of flexible-joint redundant manipulators can be developed based on the control strategy proposed in this thesis, and the joint space adaptive schemes in [3][5][6]. This adaptive scheme will be especially useful when the parameters of the flexible-joint redundant manipulator dynamic model are not known or are partially known.

(d) The Cartesian control scheme developed in Chapter 5 can be extended to a Cartesian space based impedance control scheme for flexible-joint redundant manipulators to perform compliant motion. The advantage of using a flexible-joint manipulator in performing compliant motion is that a flexible-joint manipulator has higher frequency range. However, in this case care has to be taken to ensure system stability due to the higher frequency modes.

(e) The redundancy resolution and control scheme proposed in Chapter 6 were designed for flexible-joint redundant manipulators. The same idea can also be applied with some changes in formulation to the case where a manipulator is redundant with flexible links. It may be noted that although this topic has been discussed in [1][2][7][8], the problem is still far from being completely resolved

7.3 REFERENCES

[1] J. Baillieul, "Kinematic redundancy and the control of robots with flexible components," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp 715-721, Nice, France, May 1992.

[2] J. Baillieul, "Kinematic redundancy and the control of robots with flexible components," *IEEE Control Systems Magazine*, vol. 13, no. 1, pp. 15-21, Feb. 1993.

[3] K.P. Chen and L.C. Fu, "Nonlinear adaptive motion control for a manipulator with flexible joints," *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 1201-1206, 1989.

[4] C.A. Klein and C.H. Huang, "Review of pseudo-inverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 245-250, 1983.

[5] K.Y. Lian, J.H. Jean and L.C. Fu, "Adaptive force control of single-link mechanism with joint flexibility," *IEEE Trans. Robotics, Automat.*, vol. 7, pp. 540-545, Aug. 1991.

[6] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Trans. Automat. Contr.*, vol. 37, pp. 174-181, Feb. 1992.

[7] L.A. Nguyen, I.D. Walker and R. DeFigueiredo, "Control of flexible, kinematically redundant robot manipulators," *SPIE Cooperative Intelli. Robot. in Space*, vol. 1387, pp. 296-312, 1990.

[8] L.A. Nguyen, I.D. Walker and R. DeFigueiredo, "Dynamic control of flexible, kinematically redundant robot manipulators," *IEEE Trans. Robotics Automat.*, vol. 8, Dec. 1992.