MODELING, CONTROL DESIGN AND MECHATRONIC IMPLEMENTATION OF CONSTRAINED ROBOTS FOR SURFACE FINISHING APPLICATIONS

By

BIAO YU

BACHELOR OF SCIENCE
Beijing University of Aeronautics and Astronautics
Beijing, P.R. China
1990

MASTER OF SCIENCE
South China University of Technology
Guangzhou, Guangdong, P.R. China
1993

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
December, 2000

MODELING, CONTROL DESIGN AND MECHATRONIC
IMPLEMENTATION OF CONSTRAINED ROBOTS
FOR SURFACE FINISHING APPLICATIONS

Thesis Approved:

prabhakar pagilla

Thesis Adviser

J. T. Wolenark

Gary E. Young

Alfred Larry

Dean of the Graduate College

ACKNOWLEDGMENTS

I wish to express my sincerest appreciation to my major advisor, Dr. Prabhakar R. Pagilla for his intelligent supervision, constructive guidance, inspiration, and friendship.

I would like to extend my warmest thanks to my doctoral committee members: Dr. Larry Hoberock (Chair), Dr. Gary E. Young, and Dr. Gary G. Yen for their support and suggestions in completion of this research. Their guidance and understanding made the development of this thesis a positive learning experience.

I would also like to thank my colleagues at Oklahoma State University Robotics Research Lab: Ramamurthy V. Dwivedula, Fu Yuet, L. Prasad C. Perera, and Gimkhuan Ng. They are among the finest people I know and are a joy to work with. I am grateful for the technical (and moral) support provided by Jerry Dale and James Davis. Without them, this research would not have been possible. I extend my thanks to the people at the School of Mechanical and Aerospace Engineering of Oklahoma State University who, by simply doing their jobs well, contributed to the successful completion of this thesis. They are Daleene Caldwell, Janet Smith, Diane Compton, Kyong Edwards, and Sally Kellenberger.

My most heartfelt thanks go to my parents, Rengdao Yu & Meiye Chang, and newly married wife, Yu Zou. Their unflagging faith in me has been my source of strength throughout my entire life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

Figure **Page**

Figure **Page**

Figure **Page**

Chapter 1

Introduction

Robot manipulators are used extensively in industries to reduce costs, improve productivity, obtain better product quality, and avoid humans working in hazardous environments. The industrial robot applications are usually classified into the following application categories [1]: material handling, machine loading and unloading, spraying, welding, machining, assembly, and other applications such as work part inspection. The robot tasks of welding, machining, and assembly involve robot motion under a geometrical constraint. According to [1], more than 40% of robots used in industry fall into these three application categories.

Applications that involve robotic assembly and machining require interaction between the robot end-effector and its environment. Modeling and control of robot under a constraint are critical to these robot tasks. In a robotic assembly task, robot motion includes the unconstrained motion phase where the robot moves free from work parts and the constrained motion phase where the robot is in contact with the work part. The most basic assembly task, peg-in-a-hole, is to insert one part (the peg) into another part (the hole). Interaction often occurs between the parts being assembled [1, 2]. In some cases, a downward force is required to be exerted in specific direction in order to ensure proper locking of the parts [3]. For these applications, suitable control strategies should be sought which employ not only the end-effector position control but also the contact force control.

In robotic machining applications, surface finishing is one of the important tasks which

involve constrained motion. Conventional machine tools, such as CNC operated machines, are used in general to remove large amount of material to shape a part to its desired geometry. Finishing of the machined part is required to remove material in small amounts to bring the part to the required tolerance. It is known that material finishing operations such as deburring, grinding, chamfering, polishing and other edge finishing operations can be responsible for 10 to 30 percent of all manufacturing costs [4, 5]. Automation of such processes is still in its rudimentary stages. In a recent assessment of critical Pratt & Whitney needs in technology development, it was revealed that the problem of deburring and finishing ranked second in a list of 46 manufacturing problems.

Typically, a complete robot task involves the following sequence of operations: the robot starts from the home position and moves freely for some time, makes contact with the constraint, follows the constraint surface, leaves the surface, finally returns to home position. Such a complete task can be divided into three phases:

- unconstrained motion phase, where the robot is free from the external constraint,
- transition phase, where the end-effector begins to make contact with constraint surface, transition from unconstrained motion to constrained motion takes place in this phase,
- constrained motion phase, where the robot motion is under a geometrical constraint, assembly or surface finishing operation is employed in this phase.

The main focus of this project is to develop and investigate robot controllers for a complete task. Emphasis is laid on simultaneous position/force control, stable transition control, control implementation for complete robot tasks, and mechatronic system integration. In this introductory chapter, literature reviews for robot motion control, position/force control for constrained robot, transition control, control of complete robot tasks, and mechatronic design approaches relating robot system design are given in sections 1.1 - 1.5, sequentially. Section 1.6 contains brief summary of the contributions of this work.

1.1 Robot Motion Control

Robot motion control is essential to industrial robot applications such as material transfer, machine loading and unloading, welding, spray painting, assembly, and surface finishing. Most of the robot motion control systems used in industry are simple independent local PD-type (Proportional and Differential) controllers. Each joint servo has feedback of its own position and velocity. However, the dynamics of a robot manipulator is highly nonlinear and has strong coupling between joints. Trajectory tracking performance of a simple PD controller is not satisfactory in the case of fast robot movement [6]. Advanced robot motion control algorithms have been proposed considering nonlinear and coupling dynamics of robot manipulators. These robot motion controls can be classified into three groups [7]:

- PD feedback with nonlinear gains [8, 9, 10, 11],
- nonlinear control: feedback linearization [12, 13, 14, 15], nonlinear decoupling [16, 17, 18], and robust control [19], and
- inverse dynamics control (computed torque approach) [20, 21].

The performance of these robot motion control schemes were compared experimentally in [22, 23]. The inverse dynamics control is also known as computed torque control in literature because the inverse dynamics of the manipulator is calculated in real-time to obtain the required input torque. The Coriolis and centrifugal terms and gravity terms are also compensated. Since computed torque control requires a large amount of computation in real-time, the computation time limitation is the major obstacle to the implementation of this control method [6]. Benefiting from the developments in electronic and computer engineering, computation power of microprocessors increase significantly in recent times. The robot controllers, which are typically microprocessor systems, are capable of completing sophisticated computation in one sample period. Computational burden in present time will not be the main restrictive factor that prevents extensive adoption of computed torque

control in industrial robots.

Computed torque control relies on the perfect knowledge of system inertial parameters. In practice, parameter uncertainties are unavoidable. In model-based adaptive control, the control parameters in the computed torque controller are replaced by their parameter estimates. Suitable adaptation laws are designed to obtain the parameter estimates that guarantee the states of the closed-loop system to be bounded and the convergence of tracking error to zero [22, 24, 25]. Adaptive control techniques are robust to parameter uncertainty, but not robust to unmodeled dynamics and disturbance [26, 27]. The robustness of model-based adaptive control was considered in [28]. Learning control proposed in [29, 30] is an alternative approach to reduce tracking error caused by disturbance and unmodeled dynamics for repetitive robot tasks. The basic idea is to use the error information in previous cycles due to unmodeled dynamics to generate or adjust input torque for current cycle. In this work, a model-based adaptive motion control available in literature [22] that has been well tested is chosen.

1.2 Robot Force Control

Research on robot force control began in the 1960's [31]. Two types of control approaches, namely logic branching feedback method and continuous feedback method, are available in literature for robots involving constrained motion. Logic branching feedback method consists of strings of statements, which initiate or terminate discrete moves based on discrete events such as contact force exceeding a certain level. A rule-based force control system was developed in [32]. In the 1990's, logic branching weighted algorithm was investigated to train a robot using force sensor information [33].

Continuous feedback force control is based on the position and force/torque information combined with the knowledge of manipulator and environment dynamics. Continuous sequence of force vector, motions, and/or desired relations between force and motion are

controlled. Considerable research in this type of force control schemes has been reported in the last two decades. According to the control goal, fundamental continuous feedback force control algorithms can be categorized into four classes:

- force control schemes which focus on realizing a desired dynamic relationship between the end-effector position and the contact force, including: (1) stiffness control involving the relation between position and applied force, (2) impedance and admittance control involving the relation between velocity and applied force,
- control schemes whose aim is to regulate the applied force, including explicit force control and implicit force control,
- parallel position/force control,
- simultaneously position/force control based on tangential and normal subspaces of the constraint surface. Control schemes in this category are called hybrid position/force control or hybrid impedance control in the robotics literature. In these control schemes, different controls are applied in two complementary orthogonal subspaces, position-controlled subspace and force-controlled subspace, separately.

1.2.1 Stiffness Control

In stiffness control, the joint stiffness matrix is modulated to achieve the desired relation between position and applied force [34]. Stiffness control can be categorized further into passive stiffness control and active stiffness control. Passive stiffness control is achieved by equipping the robot end-effector with a mechanical device composed of passive springs and dampers. In active stiffness control, the robot end-effector stiffness is changed based on the position and/or force feedback signals[35, 36]. A stiffness control scheme is implemented in Cartesian coordinates in [37, 38], where the stiffness was specified in work space. Three translational and three rotational stiffness coefficients were specified for force

control. Based on the difference between desired and actual end effector position, a desired force is obtained. Stiffness may be changed under computer control to match varying task requirements. The implementation of this type of stiffness control was reported for tendon tension control robot in [39, 40]. End-effector stiffness control for over-constrained systems was considered in [41, 42]. An over-constrained system contains more independent inputs than the number of kinematic of degrees of freedom.

Stiffness control is sensitive to the combined stiffness of the environment, the end-effector, and the force sensor. Uncertainty in stiffness value leads to poor force control performance. The effect of robot wrist stiffness on manipulator control was discussed in [35]. Stiffness control combined with adaptive, learning, and robust control techniques were considered in [43, 44, 45, 46, 47] to handle stiffness uncertainty. Adaptive strategies were included in the stiffness controller to maintain consistent performance in the presence of unknown parameters of the robot and the environment in [43, 44]. In [46] and [47] , feedback controller gains were changed based on on-line estimates of stiffness matrix. A reinforcement learning process was proposed in [45] to compute the stiffness values of the end-effector for repetitive tasks.

1.2.2 Impedance and Admittance Control

Impedance control has been investigated as a force control methodology for assembly and machining robot manipulators [37, 48, 49]. The basic idea of impedance control is to specify a desired dynamic behavior, namely generalized dynamic impedance, for the constrained robot [50]. The manipulator controller is designed to track a motion trajectory and realize a desired dynamic relationship between the end-effector position and the contact force [51, 52, 53, 54, 55]. Similar to stiffness control, impedance control is an approach in which "force is controlled by controlling position" [34]. Stiffness control can be regarded as a special case of impedance control with only static model-based compensation [51]. Besides position feedback, impedance control has another feedback loop for velocity and

the effect of the contact force on the velocity [36, 56, 57].

Mechanical impedance reflects the relationship between the velocity ($\dot{x}$) and the applied force f, that is,

$$z_m(t) = \frac{f(t)}{\dot{x}(t)} \text{ or } Z_m(s) = \frac{F(s)}{sX(s)}$$

where z_m denotes mechanical impedance. The goal of impedance control is to maintain the impedance of the controlled system at a desired level. The command-input torque of impedance control consists of two parts. The first part is a position control input using position and velocity signal feedback. The second part includes the effect of the constraint forces on the position and velocity.

Impedance control is suitable for applications where accurate force regulation is not required [50]. These applications include dexterous hands [31, 58], assembly robot [59], etc. The contact force under impedance control is limited during transition phase and reaches a constant value at steady state. However, the path-tracking accuracy is poor during constrained motion phase [51]. The desired impedance needs to be chosen carefully. If the desired impedance is less than twice the combined stiffness of the manipulator and environment, poor position accuracy may cause recurring oscillatory behavior [60]. Many contact tasks require that a specific contact force be applied on the constraint surface. Use of impedance control is limited in such cases.

For impedance control, the dynamic properties of the manipulator and its environment need to be well known in order to control the contact force. Adaptive schemes for controlling the end-effector impedance of robot manipulators were considered in [61, 62, 63]. Adaptive impedance control with inner position loop was investigated to improve position tracking accuracy in [64]. Impedance controller designed for robotic deburring was discussed in [65]. Sliding-mode based approach was adopted to formulate robust impedance controllers in [66, 67, 68, 69]. Iterative learning impedance control problem was formulated and solved for a robot performing repetitive tasks in [70, 71]. Simulation results of an adaptive impedance control were reported in [72] for deburring/milling processes in which

normal contact force is required. An impedance controller with a disturbance observer to estimate contact force was reported in [73]. Neural networks were embedded into adaptive impedance control and/or iterative learning impedance control in [74, 75, 76] considering the dynamic model uncertainties.

Impedance control is also implemented in another form known as admittance control or accommodation control. The mechanical admittance is defined as the inverse of the impedance. Admittance control involves regulating the admittance to achieve desired force response with small error, low overshoot, and rapid rise time [53]. Admittance control is extensively used in robotic assembly where desired force tracking is critical. Application of adaptive, sliding-mode, and learning techniques into admittance control was considered in [64, 77, 78, 79, 80].

Impedance control schemes perform indirect force control by controlling stiffness, impedance or admittance between the end-effector and the environment. Impedance control provides force control solution for those applications where accurate force regulation is not required. It is difficult to specify a desired amount of contact force with an impedance controller, because obtaining an accurate dynamic model between the robot and its environment is difficult. The use of impedance control for the robot tasks that require a specific contact force is limited.

1.2.3 Explicit and Implicit Force Control

Unlike impedance control, explicit force control and implicit force control are force control strategies that focus on force regulation [51]. Explicit force control schemes involve direct force command and measurement of force values. Explicit force control can be categorized into two groups, force based control and inner position loop based control [81]. In explicit force control, reference force input and measured force are compared and processed, then are used for generating control signal directly to the manipulator. Typically, Integral or Proportional-Integral control law is chosen. Comparative experimental results

in [81] show that integral control gives a better performance of the two. Since the measured force signal typically contains high frequency noise, it is filtered by a low pass filter before it is used to generate the control signal. Inner position based explicit force control is same as the admittance control [34]. The reference force is transformed into a reference position through an admittance [82, 83]. Since most commercial manipulators have built-in position controllers, position based explicit force controllers were implemented commonly.

Some conventional techniques such as lead-lag compensation were investigated for explicit force control in [84]. Adaptive approaches were proposed for explicit force regulation on a position-controlled robot manipulator in [85, 77]. Simulation results of an explicit controller for robotic grinding task was reported in [86]. Recent research in [87] shows that proportional explicit force control with force feedforward is essentially equivalent to the second-order impedance control with force feedback.

In implicit force control, a desired force is obtained by controlling the predefined position. The dynamics between the manipulator and the environment is required to obtain this predefined position in order to obtain desired force [31, 88]. An implicit force controller was considered in [88] for an industrial robot with elastic joints. Frequency-shaped techniques were investigated in [89] for implicit force control scheme.

1.2.4 Parallel Force Control

Stiffness, impedance, and explicit/implicit force control schemes perform either position control with indirect force control or force control with indirect position control. Parallel force control has the ability of controlling both position and force [50]. This is achieved by closing an outer force control loop around the inner position control loop [90]. Contact force is regulated to a desired value by proportional-integral or integral force control action. Simultaneously, desired position inner loop provides motion control along the feasible task space (unconstrained subspace). In this force control scheme, force control and position control work in parallel. The integral action on the force error ensures dominance

of the force loop over the position loop during constrained motion. A priority strategy was reported in [91] to manage the conflicting situations between the position and force tasks.

Under the assumption that contact force aligns with normal vector of the constraint surface, parallel position/force controller ensures position tracking and force regulation. Manipulator tracks the desired position in the unconstrained task subspace (tangential subspace to the constraint surface) with regulation of the desired force in the constrained subspace (normal subspace to the constraint surface). However, this control scheme requires that the desired force be correctly planned. When contact is established, desired force needs to be switched from zero value to non-zero value. Implementation of parallel position/force becomes difficult for the cases where constraint is curved, i.e., direction of the desired contact force is not constant. Furthermore, when contact force does not align with the normal vector of the constraint surface, this control scheme cannot ensure tracking error convergence to zero in the motion subspace.

1.2.5 Hybrid Position/Force Control

Hybrid position/force control is best suited for cases where a desired force exerted against the constraint is required together with motion control along the constraint surface. Control schemes in this category assume that the environment is rigid and the manipulator has to continuously keep the end effector in contact with the constraint. Hybrid position/force control scheme was originally introduced in [92]. In the controls literature a hybrid system is a mixture of discrete and continuous systems. In the robotics literature hybrid position/force control is used to mean a hybrid combination of simultaneous force and position control. Since position and force are controlled in two orthogonal subspaces in the hybrid position/force control scheme, perhaps a better name for this scheme may be orthogonal position/force control scheme. However, to be consistent with previous literature in the robotics field, and since the meaning is clear from the context, hybrid position/force control is used in this dissertation.

In hybrid position/force control, the constraint frame is decomposed into two orthogonal subspaces, namely constraint normal direction and tangential subspace, via two complementary projection matrices as shown in Fig. 1.1 [49, 93].

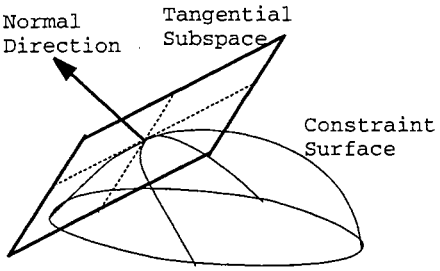

Figure 1.1: Orthogonal subspaces

In this type of force control design, the bodies in contact are assumed to be rigid. Thus force error does not affect position error. Based on this assumption, in the subspace normal to the constraint, only force control is applied because position control is not necessary if contact is maintained. Fig. 1.2 illustrates the basic idea of hybrid position/force control strategy. In Fig. 1.2, $P(x)$ is the orthogonal projection matrix whose image is the constraint surface normal at point x and $Q(x) = I - P(x)$ is the orthogonal projection matrix whose image is the constraint surface tangent at point x.

Appropriate controllers are designed for position tracking and desired force tracking separately [92, 94, 95]. Hybrid position/force control was implemented by closing a force control loop around an internal position loop in a hierarchical way in [96]. Adaptive, robust, and learning controllers have been widely introduced into the hybrid position/force control design. Adaptive model-based schemes were considered to design hybrid position/force control scheme for robot arms under a geometric constraint in [97, 98, 99, 100]. Passivity-based approach was applied to design adaptive control laws in [101, 102]. A model-based adaptive motion and force control for robots performing a complete task was given in [103]. Learning control concept [29, 104] was utilized for hybrid position/force control to improve the tracking error on the basis of previous operation data when the operations are repetitive

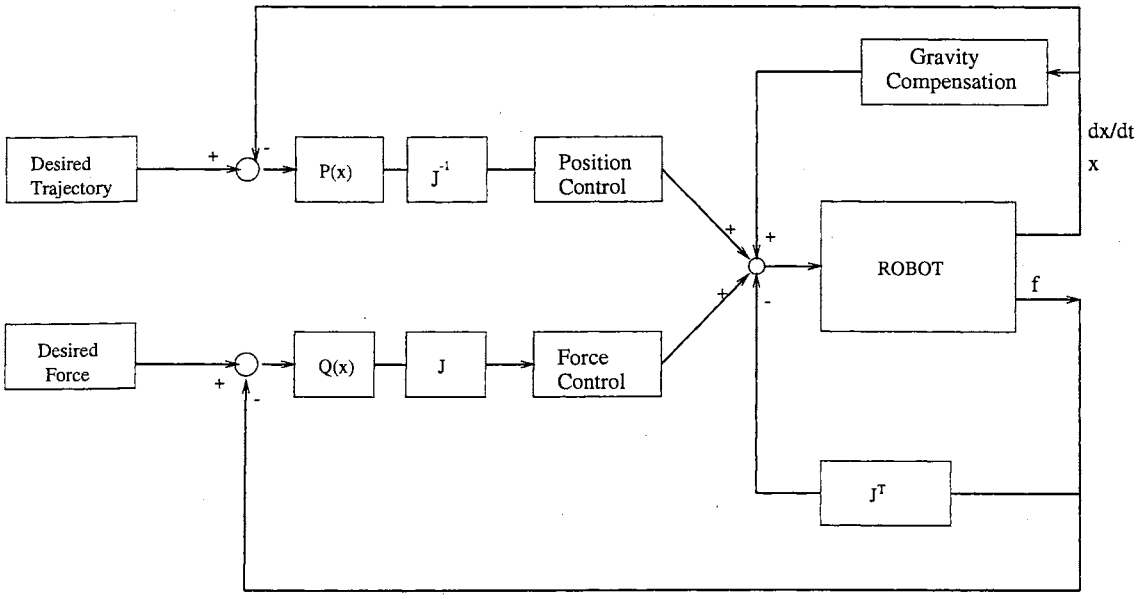

Figure 1.2: Hybrid position/force control

[105, 106, 107, 108, 109]. Deformation of constraint surface was considered to design hybrid position/force controller using learning control [110]. Neural networks were employed for motion and force control of a direct-drive robot in [111]. Fuzzy control was applied in [112] considering model uncertainty. Sliding-mode controller was used to design robust controllers in position control subspace for position/force control in [113, 114]. In the force control subspace, proportional-integral (PI) is still the most commonly used strategy. Recently, a parallel position/force control was suggested in force-controlled subspace to make the system more robust to uncertainties that arise during real-time implementation [115].

Hybrid impedance control was proposed in [116], combining impedance control and hybrid position/force control. In this type of control schemes, a distinction of impedances in force-controlled and position-controlled subspaces is obtained so that the desired velocity (or position) and force trajectory can be followed in these subspaces, respectively [117]. A disturbance observer designed in the constraint frame is incorporated into hybrid impedance control design in [118]. Advanced control techniques such as adaptive and sliding-mode were considered for hybrid impedance control in [119, 120].

1.2.6 Summary of Force Control Schemes

Tables 1.1 and 1.2 give a comparison of various fundamental force control schemes from the point of view of implementation and performance. There is no standard classification for robot force control schemes. It has been shown that some control schemes under different names are essentially equivalent [34, 87]. It is commonly accepted that impedance control and hybrid position/force control are the two major approaches for robot force control [121]. A summary of advanced force control algorithms based on fundamental schemes combined with adaptive control, robust control, learning method, and soft-computing techniques such as neural networks, fuzzy reasoning, and fuzzy-neural networks is shown in Table 1.3.

For the robotic surface finishing tasks, the robot velocity along the constraint surface and the contact force normal to the surface should be controlled separately and simultaneously. In these cases the feed-rate and cutting depth are the critical parameters that affect the finished surface. These parameters are determined by the velocity along the surface and the contact force normal to the surface. Since the material, the shape of the work parts being finished, and the fixture used to hold the work parts are different for different products, obtaining an accurate model of the work part surface is unrealistic. The indirect force control algorithms such as impedance, stiffness, admittance control, and hybrid impedance control that require the model of the environment for force regulation render poor performance if the environment model is not well known. Due to shape complexity of work parts, the desired contact force is a vector with time-varying direction and/or time varying magnitude. Parallel force control scheme that is designed for force regulation problems where magnitude and direction of desired force are all constant is not suitable to surface finishing applications either. In hybrid position/force control scheme, trajectory tracking and normal force control are applied separately. Feed-rate and normal force (cutting depth) can be controlled independently without the knowledge of accurate model of the environment. Hence, hybrid position/force control scheme is suitable for robotic surface finishing applications.

Table 1.1: Fundamental force control algorithms (1)

Algorithm	Measured variables	Modified variables	Modulated objectives
Stiffness control	Position and/or force	Position and contact force	Mechanical Stiffness
Impedance control	Position, velocity, and force	Position, velocity, and contact Force	Mechanical impedance
Admittance control	Contact force	Force error	Mechanical admittance
Explicit force control	Contact force	Force error	Contact force
Implicit force control	Position	Position error	Contact force
Parallel force control	Contact force and position	Force and position error along all directions	Force and position along all directions
Hybrid position/force control	Contact force and position	Force error in normal subspace and position error in tangential subspace	Normal force and position along the constraint surface
Hybrid impedance control	Position, velocity, and force	Position, velocity, and contact Force	Distinct impedances in normal and tangential subspaces

Table 1.2: Fundamental force control algorithms (2)

Algorithm	Dynamic model required	Path tracking	Force control
Stiffness control	Mechanical stiffness	Poor	Indirect force regulation via control of mechanical stiffness
Impedance control	Mechanical impedance	Poor	Indirect force regulation with small overshoot via control of mechanical impedance
Admittance control	Mechanical stiffness	No	Force tracking is achieved indirectly provided system admittance is well known
Explicit force control	No	No	Zero steady-state error for constant reference force; slow force trajectory tracking
Implicit force control	Mechanical stiffness	No	Contact force is controlled based on pre-defined position.
Parallel force control	No, but desired force is planned based on knowledge of system dynamics.	In unconstrained directions	Force regulation in constrained direction(s)
Hybrid position/force control	No	In unconstrained directions	Time-varying desired force tracking in constrained direction(s)
Hybrid impedance control	Impedance	In unconstrained directions	Indirect force control by controlling impedance in constrained direction(s)

Table 1.3: Advanced force control algorithms

	Stiffness, impedance, admittance control	Explicit and implicit force control	Hybrid position/force and impedance control
Adaptive	Adaptive stiffness control [43, 44, 46, 47], Adaptive impedance control [61]-[64], Adaptive admittance control [77, 78]	Adaptive explicit force control [77, 85]	Adaptive hybrid position/force control [97]-[101],[103], Adaptive hybrid impedance control [119]
Robust	Sliding-mode based impedance control [66]-[69], Robust impedance control [79]		Sliding-mode based hybrid position/force control [113, 114], Robust hybrid impedance control [120]
Learning	Learning stiffness control [45], Learning impedance control [70, 71], Learning admittance control[80]		Learning hybrid position/force control [105]-[109]
Fuzzy logic, neural networks	Neural network adaptive impedance control [74]-[76]		Neural network/fuzzy learning hybrid position/force control [110]-[112]

1.3 Transition Control

In the previous section, force control schemes for a robot interacting with an external environment were reviewed. Much of this research has been based on the assumption that the robot is already in contact with the external environment. Typically, in industrial applications such as robotic surface finishing, the robot moves a certain distance freely before making contact with the surface. If the robot impacts a surface with a non-zero velocity, then transition from unconstrained motion to constrained motion involves impulsive forces on the system, and the velocity variable is discontinuous. For the force control schemes discussed in previous section to be practically useful, control of the robot during transition to and from the constraint surface must be addressed. In general, four broad classes of solutions are available for transition control:

- redesign robot end-effector so that its compliance is changed or controlled to avoid high impact force,
- optimize trajectory to reduce contact velocity,
- equip robot with an additional sensor (e.g. proximity sensor) to achieve smooth landing, and
- construct controllers that are stable through the transition.

In the first category listed above, passive mechanical compliance in end-effector is used to reduce impact force during transition [122]. As a consequence, the stiffness required for position control is reduced, which causes poor position accuracy. The contact is accommodated at the cost of position accuracy. Active and semi-active fingertips were considered to control the damping characteristics of a fluid-filled fingertip in [123, 124]. This type of contact transition control method is mostly adopted for robotic assembly [125]. Mechanical

redesign and compliance change are not feasible options for all situations. In some cases, even though very soft fingertips are used, contact transition still tends to incite instability and may result in high impact force [126].

The second category of transition control schemes use kinematic redundancy to reduce the effective inertia in the direction of contact normal, thus avoiding high impact force [127]. Unexpected impact control of a redundant manipulator is investigated using a method known as Full Space Parameterization (FSP) in [128]. The method of generating trajectories with optimal approach velocity was discussed in [129, 130] to control the collision forces between a manipulator and its environment.

In the third category, a proximity sensor is used to detect the target surface before the robot contacts the surface. The desired trajectory is modified such that the velocity profile is smooth during the approach and transition phases [131]. A multi-phase control strategy was employed using external fiber optic proximity sensor combined with a force sensor in [132].

Since mechanical redesign, trajectory modification, kinematic redundancy, or additional sensor is not always an option, the general solution lies in constructing controllers for all phases including the transition phase. In this work, the focus is on the fourth category, i.e., constructing controllers that are stable through the transition, which will be discussed in detail.

1.3.1 Impact Modeling

In the transition phase, the robot may experience severe impact forces due to non-zero impact velocity normal to the surface. Knowledge of the impact phenomena facilitates an efficient design of a stable controller during the transition phase.

Study of impact has been extensive in the mechanics literature [133, 134, 135]. An impact model is chosen to predict the post-impact behavior of the manipulator based on pre-impact conditions. A number of impact models exist in literature [133, 135, 136, 137].

Central to the impact model is the so called coefficient of restitution. There are three distinct definitions of the coefficient of restitution: (1) kinematic coefficient (Newton's coefficient), defined as the ratio of post-impact to pre-impact normal velocity; (2) kinetic coefficient (Poisson's coefficient), defined as the ratio of post-impact to pre-impact impulse; and (3) energetic coefficient, defined as the ratio of post-impact to pre-impact energy. An extensive discussion of different coefficients of restitution can be found in [133, 138]. An experimental study of planar impact of a robot manipulator was reported in [139].

1.3.2 Transition Control Schemes

From the point of view of control design, transition control schemes considering impact phenomenon can be grouped into two categories: (1) uniform control strategies for all phases, (2) discontinuous control schemes including controllers designed for each phase.

Impedance control strategy and impulsive constraint method have been investigated to achieve unified control without switching. There is a rich body of literature on impedance control design to accomplish contact transition tasks [37, 53, 140, 141, 142, 143, 144, 145]. Impedance control for transition phase was developed based on "compliance control" in [53]. Desired force response is achieved by adjusting the impedance. Stiffness control presented in [37] and various damping control algorithms for transition are special cases of impedance control. In the active impact damping control scheme proposed in [140], the velocity gain of the PD force controller was increased for a limited time after impact. The active nonlinear damping controller presented in [143] examines derivative of force signal. A friction term is added to the output force command when this derivative exceeds a threshold. This type of control scheme provides a uniform control approach for both unconstrained and constrained tasks. But after contact is established, the force cannot be regulated unless the environment is perfectly modeled [53, 144].

Parallel to impedance transition control research, unified control designs for multi-phase including transition phase were discussed in literature based on impulsive constraint

analysis [146, 147, 148]. With the introduction of a virtual surface, control discontinuity was avoided by replacing the force discontinuity of the robot end-effector by velocity discontinuity of the constraint surface. The end-effector was assumed to be 'constrained' by a moving virtual constraint surface all the time. The desired contact force was zero before the end-effector reached the real constraint surface. When the virtual constraint surface reached the location of the real constraint surface, the motion of virtual constraint was frozen [147]. A single uniform force control was responsible for unconstrained motion, transition motion, and transition phase.

In contrast to uniformed control strategies, discontinuous control schemes consist of different control algorithms designed for unconstrained motion, constrained motion, and transition phase. Discontinuous control schemes combined with an event-based switching strategy were proposed in [149, 150, 151]. Since this category of controllers is discontinuous in nature, theory of generalized dynamical systems [152] was applied for the stability analysis [151].

There is an increasing interest in discontinuous control design for complete robot tasks. Substantial research has been reported on transition control via discontinuous approach. Experimental results of a discontinuous control scheme were reported in [153]. This scheme consists of three controllers: computed torque position control for unconstrained motion, proportional gain explicit force control for transition phase, and integral gain explicit force control for constrained motion. Model-based adaptive control design for both unconstrained and constrained motions was considered to estimate the dynamic uncertainties in [154, 155]. Positive acceleration feedback was added to the feedback loop to stabilize transition phase and achieve better force regulation performance in [156, 157, 158]. Other methods such as nonlinear proportional and derivative control scheme involving adjusting force control gain during transition phase [140, 159], input command pre-shaping approach that modifies the feed-forward command to suppress post-maneuver vibration [126], and robust control [160] were employed to design a discontinuous controller. Con-

trol of mechanical systems subject to unilateral constraint using the Hamiltonian approach was considered in [161]. Control of contact problem in constrained Euler-Lagrangian systems was investigated in [162, 163]. Nonsmooth Lyapunov theory was used to show stability [164]. Experimental evaluation of a stable transition controller was investigated for geometrically constrained robots in [155, 165]

1.4 Modeling and Control for Surface Finishing Operations

Automation of surface finishing operations is an active area of investigation in the manufacturing industry and also in several national laboratories, including the National Institute of Standards and Technology and Sandia National Laboratories [166, 167, 168]. Research towards automation has focused in many directions from prediction and modeling of burr formation [169] to creating intelligent machines [56, 170] for such operations. The development of a passive one-degree-of-freedom end-effector force control system for efficient deburring was demonstrated in [56]. In [56], experimental results for robotic deburring of two-dimensional parts using an impedance control method have been shown. Deburring using force control and active end-effector system has been presented in [170]. Several important research problem areas have been pointed out in [4, 56, 167]. Accurate position and force control of the robot in the presence of uncertainties and stable transition between unconstrained motion and contact motion were reported to be some of the major factors that need to be addressed to create an advanced deburring system.

Control design for surface finishing processes involving unconstrained motion, transition phase, and constrained motion has been investigated for a number of years. During constrained motion phase, namely surface finishing process, the end effector must exert a certain force on the work-piece. The goal of control design for surface finishing process is to maintain a constant force on the tool in the direction normal to the constraint surface

while following the position trajectory in the direction tangential to the surface.

Models of surface finishing processes are essential to the control design and system performance. Extensive research has been done to obtain theoretical and empirical force models for surface finishing processes such as grinding, deburring, and chamfering [171]. These force models for surface finishing processes formulate the relation between normal force (or tangential force) and the working conditions such as cutting speed, feed-rate, tool dimension, and specific energy. For surface finishing applications, the cutting depth is a variant because it depends on the variation of the burr size, shape, and material hardness. From the point of view of control design, the normal force is preferable to tangential force as it is less sensitive to burr area variation [48]. The deburring process was modeled as a normal force control problem in [172]. It has been shown that both normal and tangential contact forces are approximately proportional to the material removal rate [173].

It is well known that cutting forces play a dominant role in the dynamics of surface finishing processes. Force control is critical for efficient tool utilization and accuracy of the desired profile of finished workpiece. Surface finishing control design implementations via various force control strategies, namely impedance control [48], parallel force control [174], and hybrid position/force control [170, 175, 176, 177, 178, 179], have been reported in literature.

The force control performance of impedance control strategy is strongly dependent on the accurate dynamic model. In many practical applications, obtaining an accurate force model for a surface finishing process is difficult. Nonlinear adaptive observers were considered to estimate the model parameters in [171, 180].

Hybrid position/force control, in which normal force and tangential tracking are controlled simultaneously, is gaining more and more attention recently. A hybrid position/force controller design for multi-axis robotic deburring system was proposed in [176]. Implementation of this control scheme and experimental results were reported in [178]. To accommodate the workpiece location and orientation uncertainties, a vision sensor was used

while implementing hybrid position/force control in [177, 181]. In [182], signals from force and vision sensors were combined during deburring to improve the depth measurement. Other force control strategies such as human skills based adaptive hybrid impedance control [117], chamfering control design using acoustic emission feedback [169], and fuzzy force control for robotic deburring [183] are available in literature.

Overall, hybrid position/force control design is suitable for surface finishing applications where normal force and tangential motion can be controlled simultaneously. Moreover, the hybrid position/force control design is also suitable when the dynamic model of the process is not well known. Hybrid position/force control scheme is generally adopted rather than impedance control because of its robustness with respect to high contact stiffness, which is typical during surface finishing processes such as deburring, chamfering, and polishing. It should be pointed out that most control schemes available in literature ignored the tangential force due to cutting or friction along the motion on the surface. Environmental stick-slip friction during manipulator constrained motion was investigated in [179], where the friction force was compensated as unmodeled uncertainty via a sliding-mode based controller. A model-based adaptive hybrid position/force control design is investigated and implemented in [184] and friction force is modeled and related to normal force by a grinding coefficient in [185].

1.5 Mechatronic Approach for Constrained Robot Control System Design

Besides control algorithm design, architecture design in which control functions are distributed over software and hardware components plays a critical role for the overall system development [186]. A mechatronic approach typically means to conduct system or product design systematically considering the real effects of architecture design choices on the control system performance. This approach is critical to control of fast dynamic systems with

discontinuous dynamics where control switching is required, such as robotic surface finishing systems. It has been shown that architecture design is essential for multi-processor controller for robot system in [187].

Mechatronics is a trademark registered by Yaskawa Electric Company in 1971. Mechatronic systems are becoming common in many engineering applications [188]. According to the definition of IFTMM (International Federation for the Theory of Machines and Mechanism), "Mechatronics is the synergistic combination of precision mechanical engineering, electronic control, and systems thinking in design of products and manufacturing processes" [189]. From an engineering perspective, the most remarkable characteristic of mechatronics is that it crosses the boundaries of mechanical, electronic, and computer science fields. The goal of mechatronic approach is to take into account all the advantages that can result from an integrated design considering the best properties of the components contained within the system.

In the 1980's, significant progress of microcomputer and power electronic technology made more complicated control algorithms realizable in mechatronic systems. In recent years, the emphasis of mechatronic systems is shifting to the software part, which is information processing and decision making. In the 1990's, besides extensive information integration within mechatronic systems, communication technologies played an increasing role [190]. Thus higher level system integration can be achieved using mechatronic design approaches.

The main advantage of a mechatronic approach is its optimal integration of system components using the best properties of mechanical and electronic components in synergistic combination with computer control. A mechatronic system typically involves three aspects: sensing, processing of data, and action to control the process. An overview of a mechatronic system is shown in Fig. 1.3. The areas that were historically considered by mechanical, electronic, computer science, and control engineers separately are viewed in an integral way.

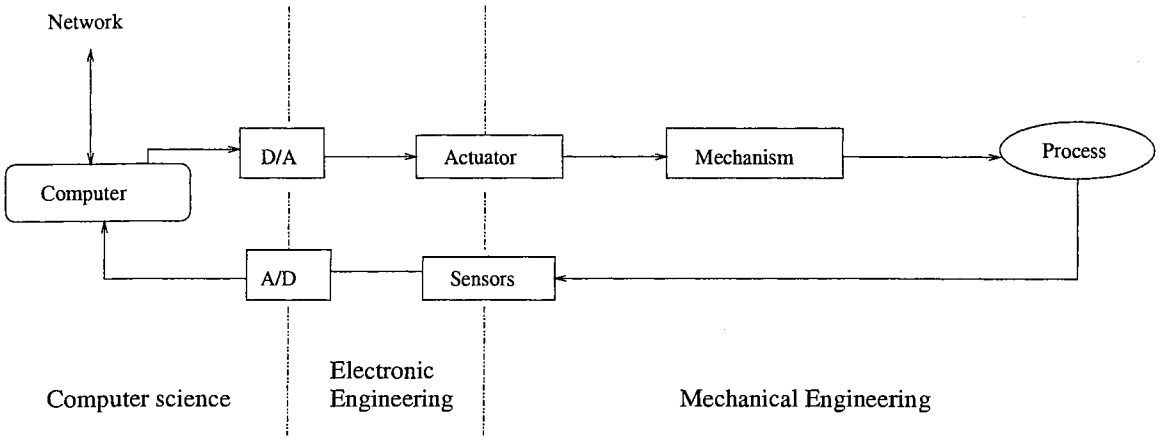

Figure 1.3: Overview of a mechatronic system

Mechatronic approaches are most evident in the robotics field. A typical industrial robot consists of five essential components:

(1) manipulator: a mechanism consisting of several segments or arms,

(2) end-effector: a mechanism holding the workpiece or tools,

(3) actuators: components driving the manipulator arms,

(4) sensor: components collecting information such as position, velocity, and force, and

(5) controller: microcomputer or microprocessor system to generate trajectories and to implement the designed control algorithm.

A robot system is an integration of mechanical, electronic, and controller components. Considering information flow in the system, a robot system sequentially performs the following: collects information from its sensors, processes the information, communicates the information to controller, execute the control algorithm to generate control signal, and amplifies the control signal to drive actuators. This is an information integration procedure that crosses the boundaries of mechanical, electronic, and computer science.

Robot systems performing surface finishing operations require stable and robust control algorithms, fast processors, position/velocity and force sensors, and high performance

surface finishing tools. It also requires a good system design that has optimal usage of its components. For the control design of a system involving impact and contact with the environments, the following issues strongly affect the system behavior:

- timing aspects related to the distribution of control function over its multi-processor structure,
- methods to coordinate different processor operations,
- communications among processors and sensors, and
- data flow management.

For a robot control system, suitable architecture design choices such as sampling frequency, force and position feedback signal quality and rate, communication rate among the system processors, and interrupt-based or polling-based data acquisition are essential to obtain optimal behavior from the overall system. System reconfiguration that lead to optimal usage of the system component for an open architecture experimental robot system was reported in [191]. Systematic function distribution among a multi-processor robot control system was discussed in [187]. Particularly, timing aspects are critical to impact phase, where even a single sampling delay in the reaction of force control and/or control switching may cause significant increase of impact force, as well as serious bouncing of the robot end-effector on the constraint surface. In this work, the architecture design via mechatronic approach was investigated considering the requirements of the discontinuous control scheme for robotic surface finishing system [192], which is a time critical real-time control system.

1.6 Contributions

The contributions of this work are four-fold: (1) a new adaptive position/force control scheme was proposed based on constrained robot dynamics; (2) a unique robust transition

control algorithm was designed for control switching from unconstrained to constrained motion; (3) a discontinuous control strategy was investigated for complete robot tasks including unconstrained, constrained motion, and transition phase; (4) system architecture design accommodating discontinuous control requirements was conducted to optimize overall system performance. An open architecture robot experimental platform was developed. Different system implementation approaches were compared. Extensive experiments have been conducted to test and analyze the proposed control strategy and controller implementation approach.

The dynamics of complete robot tasks involving unconstrained and constrained motions was investigated. A complete dynamic model that describes the dynamic behavior of the robot for surface finishing tasks such as deburring, grinding, chamfering, and polishing was developed. A complete surface finishing task is divided into three phases (unconstrained motion phase, transition phase, and constrained motion phase) depending on the location of the robot end-effector with respect to the constraint surface [103]. A new adaptive position/force control scheme was proposed for the constrained motion phase. Instead of assuming that the tangential force is negligible or regarding it as model uncertainty, it is assumed that normal and tangential forces are related by a grinding coefficient during constrained motion phase. Since the material of the constraint surface, tool rotation speed, and feed rate are unchanged during one process, the grinding coefficient relating the normal and tangential contact force magnitude is assumed to be constant. In practice, the value of the coefficient is not known. An on-line adaptation law for grinding coefficient was proposed [184, 185]. Similar to up and down milling in machining, two types of modes were considered for robotic surface finishing operations. These modes are distinguished from each other based on the relation of the direction of tool rotation and the direction of travel of the tool on the surface. Chamfering experiments were conducted to study the effect of these two types. Experimental results from both surface following and surface finishing validate the proposed control strategy.

A unique transition control algorithm was designed [155]. Most work in literature for constrained motion control assumed that the robot is already on the surface. To make hybrid position/force control algorithms feasible for complete robot tasks involving constrained and unconstrained motions, contact transition problem was considered. Most of the contact transition algorithms in literature assume either the environment is compliant and/or an impact model exists for the surface and can be used in the control algorithm. Furthermore, most of the transition control algorithms that have been proposed have not been experimentally verified for a complete task. In this work, a new stable contact transition controller is designed and is shown to be asymptotically stable. Extensive experiments were conducted for a robot following a surface using the proposed method. Uncertainty in the location of the constraint is considered as the main cause for impact of the robot with the constraint. Experiments were conducted with different levels of constraint uncertainty, trajectory velocity, and constraint rigidity [165]. It is shown that the performance of the proposed control methodology is much improved when compared with direct switching from unconstrained motion control to constrained position and force control.

Controllers designed for unconstrained motion, constrained motion, and transition phase are integrated for a complete robot task. An event driven control switch strategy and an event based trajectory modification approach are developed for this discontinuous multi-phase control scheme [103].

An open architecture experimental platform was developed. The experimental platform comprises of a two-axis direct drive manipulator, a 6-axis force/torque sensor, a surface finishing tool, data acquisition cards, power supply and amplifiers. Real-time control software was developed considering the characteristics of proposed discontinuous control strategy. Real-time software differs from conventional software in that its result must not only be numerically and logically correct, but also must be delivered at the correct time. An efficient mechatronic approach was used to coordinate and integrate hardware and software components of this robot system. To minimize feedback and control algorithm switch-

ing delays, various integration modes that involve efficient data communication and data filtering are considered and compared experimentally. Real-time control software was developed to provide optimal use of the hardware components and available processing capability. Extensive experiments were conducted comparing different integration modes [192]. Emphasis is given to two important aspects, rate of sampling and data filtering.

The remainder of the thesis is organized into the following chapters. Dynamic model for complete surface finishing operation is developed in Chapter 2. Chapter 3 contains controller design for each phase. Closed-loop stability with the proposed controllers is also shown in Chapter 3. In Chapter 4, hardware and software aspects of the open architecture experimental platform for robotic surface finishing operations are discussed in detail. Mechatronic design of the system is also given. Representative experimental results are given and discussed in Chapter 5. Summary and future work are given in Chapter 6. Extensive experiments have been conducted to investigate various aspects of control design for constrained robots. The complete experimental results are given in Appendices. Appendix A contains experimental results of model-based adaptive control for unconstrained motion. Complete experimental results of adaptive hybrid position/force control for constrained motion are included in Appendix B. Appendix C and D give the complete experimental results of free impact and transition control, respectively.

Chapter 2

Dynamic Modeling of Constrained Robot Manipulator

Dynamic modeling plays an important role in control design. In this chapter, the dynamics of a robot performing a complete task is developed. The motion of a complete robot tasks consists of three phases, namely unconstrained motion phase, transition phase, and constrained motion phase. The dynamic models for each phase are derived and analyzed in this chapter. Section 2.1 discusses the robot dynamics based on Euler-Lagrange equations. Geometrical constraint on manipulator is modeled by a unilateral constraint. Based on this constraint model, constrained robot dynamics is derived. In section 2.2, impact modeling issues are considered in order to obtain insight into the robot dynamics during the transition phase. Section 2.3 presents the dynamics of each phase of a complete surface finishing task. Finally in section 2.4, some properties of the robot dynamic model essential for control are given.

2.1 Constrained Robot Dynamics

Let the kinetic and potential energy functions of an n-link robot be given by $\mathcal{K}(q, \dot{q}) = \frac{1}{2}\dot{q}^\top M(q)\dot{q}$ and $\mathcal{P}(q)$, where $q \in \mathbb{R}^n, \dot{q} \in \mathbb{R}^n$ are the generalized position and velocity, re-

spectively, and $M(q) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite mass matrix. The dynamics of the robot is given by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + J^{\top}(q)f \tag{2.1}$$

where $C(q,\dot{q})$ is the matrix composed of Coriolis and centripetal terms, $g(q)$ is the gravity vector, τ is the vector of generalized forces applied by the motors at each joint of the robot, f represents the vector of external forces (contact force), and $J(q)$ is the Jacobian of the manipulator. Let the geometrical constraint on the robot be modeled by the following unilateral constraint,

$$\phi(x(q)) \leq 0, \tag{2.2}$$

where $x(q)$ is the Cartesian position. The presence of the unilateral constraint in the robot workspace divides the state space into the following sets:

$$X_c := \{q, \dot{q} \in \mathbb{R}^n : \phi(q) = 0\} \tag{2.3}$$

$$X_u := \{q, \dot{q} \in \mathbb{R}^n : \phi(q) < 0\} \tag{2.4}$$

$$X_f := \{q, \dot{q} \in \mathbb{R}^n : \phi(q) > 0\} \tag{2.5}$$

where X_c represents the configurations wherein the robot lies on the constraint surface, X_u represents configurations that the robot can freely move, and X_f represents configurations that violate the constraint.

The constraint is assumed to be smooth. Define the following orthogonal projection matrix whose image represents the normal direction of the constraint,

$$P_\phi(q) = \frac{[\nabla\phi(q)]^{\top}[\nabla\phi(q)]}{\|\nabla\phi(q)\|^2} \tag{2.6}$$

where $\nabla\phi(q)$ is the row vector representing the gradient of the function $\phi(q)$ and $\|\cdot\|$ denotes the Euclidean 2-norm of a vector. The kernel of $P_\phi(q)$ gives the tangential direction of the constraint, and is given by

$$Q_\phi(q) = I_n - P_\phi(q)$$

where I_n denotes the $n \times n$ identity matrix. The external force, f, given in (2.1) is the contact force due to the constraint, and can be written as

$$f = n(x)f_n + t(x)f_t \tag{2.7}$$

where f_n and f_t represent the magnitude of normal force and tangential force, respectively, $n(x)$ represents the unit vector normal to the constraint surface and $t(x)$ represents the unit vector in the direction of robot motion tangential to the constraint surface. These two vectors are given by

$$n(x) = \frac{[\nabla\phi(x)]^\top}{\|\nabla\phi(x)\|} \tag{2.8}$$

$$t(x) = \frac{\dot{x}}{\|\dot{x}\|} \tag{2.9}$$

Notice that the projection matrices $P_\phi(q)$ and $Q_\phi(q)$ have rank 1 and rank $(n-1)$, respectively. The unit vector $n(x(q))$ is the vector in the range space of $P_\phi(q)$ and $t(x(q))$ is the vector in the null space of $P_\phi(q)$.

The magnitude of the contact force f depends on the activation/deactivation of the constraint,

$$\phi(q) < 0 \implies \|f\| = 0 \tag{2.10}$$

$$\phi(q) = 0 \implies \|f\| \geq 0 \tag{2.11}$$

A complete task of the robot in the presence of the unilateral constraint can be divided into three phases: (a) when $\phi(q) < 0$, then the robot is said to be in the unconstrained motion phase, (b) when $\phi(q) = 0$ and the velocity normal to the surface is zero, then the robot is said to be in the constrained motion phase, and (c) transition from the unconstrained motion phase to the constrained motion phase is termed as the transition phase. The space X_c defined in (2.3) can be sub-divided into the sets X_{ct} and X_{ca}, i.e., $X_c = X_{ct} \cup X_{ca}$, where

$$X_{ct} := \{q, \dot{q} \in \mathbb{R}^n : \phi(q) = 0, P_\phi(q)\dot{q} \neq 0\} \tag{2.12}$$

$$X_{ca} := \{q, \dot{q} \in \mathbb{R}^n : \phi(q) = 0, P_\phi(q)\dot{q} = 0\} \tag{2.13}$$

where $P_\phi(q)\dot{q}$ indicates the velocity normal to constraint surface. The motivation for this sub-division is that the system can impact the constraint surface with a non-zero normal velocity, and there is a jump condition in velocity. Notice that X_{ct} is the set of all impact points. The transition phase takes place when the robot jumps between X_{ct} and X_u. With the division of the robot state space, the dynamics of the robot in each phase can be written as follows:

If $q \in X_u$, then the dynamic equations are

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau \tag{2.14}$$

If $q \in X_c$, then the dynamic equations are

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + J^\top(q)f \tag{2.15}$$

If $q \in X_{ct}$, then the jump condition for equation (2.15) is given by

$$\dot{q}^+ = \mathcal{D}(q,\dot{q}^-) \tag{2.16}$$

where, $\dot{q}^+$ and $\dot{q}^-$ represent the post-impact velocity and pre-impact velocity, respectively. $\mathcal{D}(\cdot)$ represents an operator which maps the impact velocity to the rebound velocity. This operator can take several forms depending on the choice of the impact model for the constraint surface. The dynamics at impact points is discussed in the following section.

2.2 Impact Modeling

Impacts are generally treated as very large forces acting over a short duration of time. If it is assumed that impact occurs over an infinitesimally small period of time, then (1) all velocities remain finite and (2) there is no change in the position of the system. If Δt, where $\Delta t \to 0$, is the duration of collision then the force impulse f_I due to impact at time t_* is

$$f_I = \int_{t_*}^{t_*+\Delta t} f(\omega)d\omega. \tag{2.17}$$

Integrating (2.1) from t_* to $t_* + \Delta t$, the dynamics during impact becomes

$$M(q)\sigma_{\dot{q}} = J^\top(q) f_I \tag{2.18}$$

where $\sigma_{\dot{q}} := \dot{q}^+ - \dot{q}^-$, $\dot{q}^-$ and $\dot{q}^+$ represent pre-impact and post-impact velocities. In the Cartesian coordinates, the change in velocity is given by

$$\sigma_v := J(q)M^{-1}(q)J^\top(q) f_I \tag{2.19}$$

where $v := \dot{x}$ and $\sigma_v := v^+ - v^-$. One method of obtaining post-impact velocities from pre-impact velocities is to assume Newton's restitution model for normal velocity, i.e.,

$$n^\top(x)v^+ = -e_n n^\top(x)v^- \tag{2.20}$$

where e_n denotes the normal coefficient of restitution and $n(x)$ is the unit normal vector to the constraint surface. Pre-multiplying (2.19) by $n^\top(x)$ and substituting (2.20) results in

$$-(1+e_n)n^\top(x)v^- = n^\top(x)J(q)M^-(q)J^\top(q) f_I \tag{2.21}$$

For an ideal impact there is no change in position of the system, which implies that there is no relative motion between the robot end-effector and the constraint surface during impact. Thus, there is no friction force, which means that there is no tangential impulse. In Cartesian space, the contact force impulse is

$$f_I = f_{nI} n(x) \tag{2.22}$$

where f_{nI} is the magnitude of the normal impulse. In joint space,

$$J^\top(q) f_I = P_\phi(q)J^\top(q) f_I \quad \text{and} \quad Q_\phi(q)J^\top(q) f_I = 0 \tag{2.23}$$

Substituting (2.22) into (2.21) yields

$$f_{nI} = -(1+e_n)\frac{n^\top(x)v^-}{n^\top(x)H(q)n(x)} \tag{2.24}$$

where $H(q) = J(q)M^{-1}(q)J^{\top}(q)$. Pre-multiplying (2.19) with $t(x)$ and using (2.24) results in

$$t^{\top}(x)\sigma_v = t^{\top}(x)H(q)n(x)f_{nI} \tag{2.25}$$

Substituting (2.24) into (2.25) gives

$$t^{\top}v^{+} = t^{\top}(x)v^{-} - \frac{t^{\top}(x)H(q)n(x)}{n^{\top}(x)H(q)n(x)}(1+e_n)n^{\top}(x)v^{-} \tag{2.26}$$

Equation (2.26) together with (2.20) give the post-impact velocity knowing the coefficient of restitution. Notice that even though the tangential impulse is assumed to be zero, there is still a jump in the tangential velocity given by (2.26). This jump is primarily due to the configuration of the robot. From (2.26), notice that the sign of $t^{\top}(x)H(q)n(x)$ determines the sign of the jump in the tangential velocity.

For a non-ideal impact, the contact force impulse, f_I, generally has components both in the normal and tangential directions. The contact force impulse can be expressed as $f_I = f_{nI}n(x) + f_{tI}t(x)$, where f_{nI} and f_{tI} are the magnitude of the normal force impulse and tangential force impulse, respectively. For this non-ideal case of f_I, another equation is required in addition to (2.20) and (2.19) to get post-impact behavior, i.e., to compute f_{nI}, f_{tI} and the post-impact tangential velocity. This additional equation can be obtained by assuming the existence of the kinetic coefficient of restitution [133], that is the ratio of the tangential to normal force impulse. Even though in the non-ideal case there is a component of the contact force impulse in the tangential direction, its effect is small compared to that of the normal component. This is in fact validated by the experimental results, which are shown in sections 5.3.1.

Considering the compliance between the robot and the constraint surface for a non-ideal impact, the magnitude of the pre-impact normal velocity versus the post-impact normal velocity can be assumed to be related as shown in Fig. 2.1. In Fig. 2.1, $\dot{q}_n^{-}(t_*) = P_\phi \dot{q}^{-}(t_*)$, $\dot{q}_n^{+}(t_*) = P_\phi \dot{q}^{+}(t_*)$, and δ_{nv} represents a threshold of normal impact velocity, below which the magnitude of the post-impact normal velocity is zero.

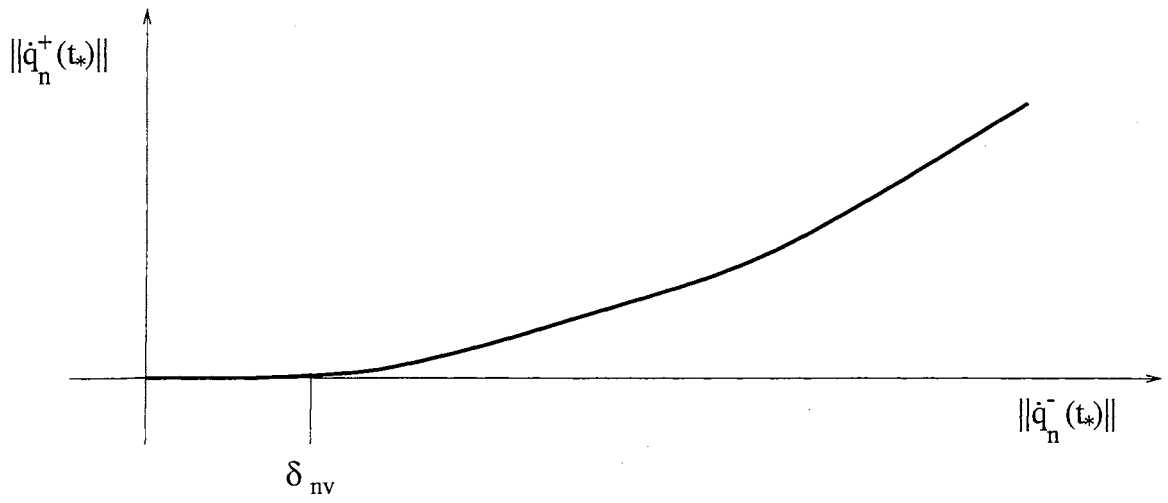

Figure 2.1: Post-impact normal velocity vs. pre-impact normal velocity

Thus far, existence of either kinematic coefficient or kinetic coefficient has been assumed to obtain the post-impact behavior of the system. Another important aspect during impact of a mechanical system is the relation between pre-impact and post-impact kinetic energy. For impacting mechanical systems, it can be assumed that there is a finite amount of kinetic energy reduction of the system due to impact [133]. The kinetic energy reduction during impact is given by

$$\mathcal{K}^+ - \mathcal{K}^- = -\Delta\mathcal{K} < 0 \tag{2.27}$$

where $\mathcal{K}^- = \frac{1}{2}[\dot{q}^-]^T M(q)[\dot{q}^-]$ and $\mathcal{K}^+ = \frac{1}{2}[\dot{q}^+]^T M(q)[\dot{q}^+]$ are pre-impact kinetic energy and post-impact kinetic energy, respectively, and $\Delta\mathcal{K}$ represents the amount of kinetic energy reduction. Due to the quadratic nature of the left-hand-side of (2.27) with embedded mass matrix, it is not possible to explicitly obtain $\dot{q}^+$ from the knowledge of $\dot{q}^-$, $M(q)$, and $\Delta\mathcal{K}$. Nevertheless (2.27) find its use during Lyapunov function based stability analysis of the transition phase.

2.3 Dynamics for a Complete Surface Finishing Task

For dynamics in constrained motion phase where surface finishing operation take place, the contact force due to the constraint has to be considered. In the dynamics of constrained motion phase, the contact force consists of normal and tangential components: f_n and f_t. The contact force, which is grinding force for surface finishing applications, is determined by conditions such as tool rotation speed, feed-rate, material removal rate, etc. The normal force causes the abrasive to wear the surface at a specific depth. When the steady depth is reached, f_n and f_t are constant [193].

In surface finishing applications, grinders are grouped as force-controlled and power-controlled grinders. Power-controlled grinders are used in situation where force sensors are not available. In that case power is the only obtainable measure of force. For force-control grinders, the normal force is the independent input that is the primary cause of cutting depth. The normal force should be greater than a threshold level, f_{th}, to cause material removal. The material volumetric removal rate Q_w is linear with the amount of the normal force above the threshold, f_{th}, that is

$$Q_w = WRP(f_n - f_{th}) \tag{2.28}$$

where WRP is Work Removal Parameter. It has been shown in [194] that when steady contact and feed-rate (velocity along the constraint surface) are reached f_n and f_t are related by a constant coefficient, called the grinding friction coefficient, as

$$f_t = \xi f_n \tag{2.29}$$

where ξ represents the grinding friction coefficient. The robot dynamics during surface finishing process becomes

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + v(q)f_n + v'(q)\xi f_n \tag{2.30}$$

where $v(q)$ and $v'(q)$ are the vectors that map normal and tangential contact force into

corresponding joint forces respectively, and are given by

$$\begin{aligned} v(q) &= J^{\top}(q)n(x) \\ \bar{v}(q) &= v(q)/[v(q)^T v(q)] \\ v'(q) &= J^T(q)t(x) \end{aligned} \tag{2.31}$$

The dynamics for a complete surface finishing task is summarized as follows.

- Unconstrained motion phase:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau \tag{2.32}$$

- Transition phase:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau \quad \text{and} \quad \dot{q}^+ = \mathcal{D}(q,\dot{q}^-) \tag{2.33}$$

- Constrained motion phase:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + v(q)f_n + v'(q)\xi f_n \tag{2.34}$$

2.4 Dynamic Model Properties

The robot dynamic model for a complete task has some useful properties that can be used in control design. The useful dynamic model properties are given below.

1. The inertia matrix, $M(q)$, is a symmetric positive definite matrix. This matrix for all system configurations is bounded from above and below, that is

$$\lambda_m I \leq M(q) \leq \lambda_M I \tag{2.35}$$

where λ_m and λ_M denote the strictly positive finite minimum and maximum eigenvalue of $M(q)$ for all configuration of q, respectively.

2. The matrix $\frac{1}{2}\dot{M}(q) - C(q,\dot{q})$ is skew-symmetric, that is

$$z^\top [\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]z = 0 \tag{2.36}$$

for any vector $z \in \mathbb{R}^n$.

3. The left-hand-side of the robot dynamic model given by (2.32), (2.33), and (2.34) is linear in terms of coupled manipulator inertial parameters, and can be written as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = Y(q,\dot{q},\ddot{q})\beta$$

where $\beta \in \mathbb{R}^p$ is a coupled manipulator parameter vector, and $Y(q,\dot{q},\ddot{q})$ is an $n \times p$ regressor matrix.

4. There is a finite amount of kinetic energy reduction after impact, that is,

$$\frac{1}{2}(\dot{q}^+)^\top M(q)\dot{q}^+ - \frac{1}{2}(\dot{q}^-)^\top M(q)\dot{q}^- = -\Delta\mathcal{K} < 0 \tag{2.37}$$

where $\dot{q}^+ = \mathcal{D}(q,\dot{q}^-)$.

5. The operator $\mathcal{D}(q,\dot{q}^-)$ is such that there exists a threshold value δ_{nv}, when the norm of the pre-impact normal velocity satisfies $\|\dot{q}_n^-(t_*)\| \leq \delta_{nv}$, then the post-impact velocity normal velocity is zero, that is

$$\|\dot{q}_n^+(t_*)\| = 0 \text{ if } \|\dot{q}_n^-(t_*)\| \leq \delta_{nv} \tag{2.38}$$

The first two properties are essential in Lyapunov stability analysis. The adaptive control algorithms are based on the linear parameterization property of the dynamics of robot manipulator. The last two properties regarding kinetic reduction and post-impact velocity due to impact are critical to the stability proof of transition control.

Chapter 3

Control Design for a Complete Task

In the previous chapter, the dynamics of a complete robot task was derived and the properties of the dynamic models were given. In this chapter control algorithms are developed based on the dynamics of each phase, separately. In section 3.1, a model-based adaptive control scheme is described. The control goal in unconstrained motion phase is to track the desired motion trajectory considering manipulator model uncertainties. Since switching directly from unconstrained motion control to constrained motion control may lead to severe repeated impacts of the robot on the surface. A stable transition controller is proposed in section 3.2 to ensure that repeated impacts do not occur. A new adaptive hybrid position/force control scheme is developed for constrained motion phase in section 3.3. During the constrained motion phase, the control goal is to simultaneously track the desired motion in the tangential direction, regulate the desired force normal to the constraint surface, and provide enough cutting force in the tangential direction. A on-line estimate law is designed for grinding friction coefficient that relates the normal and tangential contact forces. In section 3.4, this chapter is concluded with a stable discontinuous control scheme by integrating the controller for each phase via an event based switching strategy.

3.1 Model-based Adaptive Control for Unconstrained Motion Phase

During this phase the robot is away from the constraint surface and is in unconstrained motion. A number of control designs exist in literature that have been shown to work quite well for trajectory tracking. An experimentally well tested passivity type model-based controller is chosen for this phase. If the robot inertial parameters are not known exactly then a model-based adaptive controller can be considered for this phase.

$$\tau = \widehat{M}(q)\ddot{q}_r + \widehat{C}(q,\dot{q})\dot{q}_r + \widehat{g}(q) - F_v e_v \tag{3.1}$$

$$\widehat{\beta}(t) = \beta_0 - \int_0^t \Gamma^{-\top} Y^{\top}(q,\dot{q},\dot{q}_r,\ddot{q}_r) e_v dt \tag{3.2}$$

where F_v, Γ are the positive definite gain matrices, $\widehat{\beta}(t)$ and β_0 are the estimate and initial known value of β, respectively, and

$$\begin{aligned} \dot{q}_r &= \dot{q}_d - \Lambda_p e \\ \ddot{q}_r &= \ddot{q}_d - \Lambda_p \dot{e} \\ e_v &= \dot{q} - \dot{q}_r \end{aligned}$$

where Λ_p is positive definite gain matrix. The reference velocity error e_v is defined as the difference between actual joint velocity $\dot{q}$ and reference velocity q_r. The terms $\widehat{M}(q)$, $\widehat{C}(q,\dot{q})$, and $\widehat{g}(q)$ are estimates of the respective matrices. Using the linear parameterization property (property 3 in section 2.4), the dynamics estimate is parameterized in the terms of estimate of β, i.e., $\widehat{\beta}$.

$$Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widehat{\beta} = \widehat{M}(q)\ddot{q}_r + \widehat{C}(q,\dot{q})\dot{q}_r + \widehat{g}(q)$$

substituting the control law (3.1) into the unconstrained robot dynamics (2.32) yields

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \widehat{M}(q)\ddot{q}_r + \widehat{C}(q,\dot{q})\dot{q}_r + \widehat{g}(q) - F_v e_v \tag{3.3}$$

Subtracting $[M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r]$ from both sides of the above equation and simplifying,

$$\begin{aligned} &M(q)(\ddot{q} - \ddot{q}_r) + C(q,\dot{q})(\dot{q} - \dot{q}_r) + F_v e_v = \\ &(\widehat{M}(q) - M(q))\ddot{q}_r + (\widehat{C}(q,\dot{q}) - C(q,\dot{q}))\dot{q}_r + (\widehat{g}(q) - g(q)) \end{aligned} \tag{3.4}$$

Using the linear parameterization property

$$(\widehat{M}(q) - M(q))\ddot{q}_r + (\widehat{C}(q,\dot{q}) - C(q,\dot{q}))\dot{q}_r + (\widehat{g}(q) - g(q)) = Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widetilde{\beta} \tag{3.5}$$

where $\widetilde{\beta} = \widehat{\beta}(t) - \beta$. Substituting (3.5) into (3.4) yields error dynamics

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widetilde{\beta} \tag{3.6}$$

Stability of the closed-loop system can be shown using Lyapunov's second method.

Stability: Consider the following Lyapunov function candidate for unconstrained motion phase:

$$V_U(e_v,\widetilde{\beta}) = \frac{1}{2}e_v^\top M(q)e_v + \frac{1}{2}\widetilde{\beta}^\top \Gamma \widetilde{\beta} \tag{3.7}$$

The time derivative of the Lyapunov function candidate along the trajectories of (3.6) is

$$\begin{aligned} \dot{V}_U(e_v,\widetilde{\beta}) =& e_v^\top M(q)\dot{e}_v + \frac{1}{2}e_v^\top \dot{M}(q)e_v + \dot{\widetilde{\beta}}^\top \Gamma \widetilde{\beta} \\ =& e_v^\top[-C(q,\dot{q})e_v - F_v e_v + Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widetilde{\beta}] + \frac{1}{2}e_v^\top \dot{M}(q)e_v + \dot{\widetilde{\beta}}^\top \Gamma \widetilde{\beta} \\ =& -e_v^\top F_v e_v + e_v^\top[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]e_v + e_v^\top Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widetilde{\beta} + \dot{\widetilde{\beta}}^\top \Gamma \widetilde{\beta} \end{aligned}$$

Since $[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]$ is a skew-symmetric matrix, $e_v^\top[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]e_v = 0$,

$$\dot{V}_U(e_v,\widetilde{\beta}) = -e_v^\top F_v e_v + e_v^\top Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\widetilde{\beta} + \dot{\widetilde{\beta}}^\top \Gamma \widetilde{\beta} \tag{3.8}$$

The time derivative of the parameter given by adaptation law (3.2) is

$$\dot{\widehat{\beta}} = -\Gamma^{-\top} Y^\top(q,\dot{q},\dot{q}_r,\ddot{q}_r)e_v \tag{3.9}$$

Since vector β is constant, $\dot{\widehat{\beta}} = \dot{\widetilde{\beta}}$ and hence.

$$\dot{\widetilde{\beta}} = -\Gamma^{-\top} Y^\top(q,\dot{q},\dot{q}_r,\ddot{q}_r)e_v \tag{3.10}$$

Substituting (3.10) into (3.8) and simplifying, we obtain

$$\dot{V}_U(e_v, \widetilde{\beta}) = -e_v^\top F_v e_v < 0 \tag{3.11}$$

Therefore, equations (3.7) and (3.11) show that $V_U(e_v, \widetilde{\beta})$ is indeed a Lyapunov function, i.e., V_U is positive definite and its time derivative along the trajectories of (3.6) is negative definite. Because V_U is a positive function that is bounded from below, e_v and $\widetilde{\beta}$ are bounded. Since the function $Y(q, \dot{q}, \dot{q}_r, \ddot{q}_r)$ appearing in (3.6) remains bounded and manipulator inertia matrix $M(q)$ is never singular during the motion, $\dot{e}_v$ is bounded from (3.6). This means that e_v is uniformly continuous. Furthermore, since e_v appears explicitly in $\dot{V}_U$, e_v converges to zero asymptotically.

3.2 Robust Discontinuous Control for Transition Phase

Transition phase starts when the robot makes its first impact with the surface and lasts until the robot makes a stable contact with the surface. The first impact gives the actual location of the constraint surface. The desired motion trajectory of the robot is developed based on a priori knowledge of the location of the constraint. The desired trajectory should be modified based on the actual location of the constraint, which is given by the first impact. After the first impact the desired trajectory is modified using the projection matrices. The modified trajectory represents motion of the end-effector along the tangential direction of the constraint, and position regulation in the normal direction to the actual location of the constraint. Now, the control goal becomes regulation of the end-effector onto the surface and tracking of the modified desired trajectory in the tangential direction. From now on, we assume without loss of generality that the robot inertial parameters are known. In the tangential subspace, the control goal is to track the modified desired trajectory. The following control is chosen:

$$\tau_t = Y(q, \dot{q}, \dot{q}_{rt}, \ddot{q}_{rt})\beta - F_v e_{vt} \tag{3.12}$$

where $\dot{q}_{rt}$ and e_{vt} are reference velocity and reference velocity error projected onto the tangential subspace, i.e.,

$$\dot{q}_{rt} = Q_\phi(\dot{q}_d - \Lambda_p e) \tag{3.13}$$

$$\ddot{q}_{rt} = Q_\phi(\ddot{q}_d - \Lambda_p \dot{e}) + \dot{Q}_\phi(\dot{q}_d - \Lambda_p e) \tag{3.14}$$

$$e_{vt} = Q_\phi[\dot{q} - (\dot{q}_d - \Lambda_p e)] \tag{3.15}$$

In the normal subspace, the control goal is convergence of the normal velocity to zero as quickly as possible. Introduce the normal projection of the reference velocity error as

$$e_{vn} = P_\phi[\dot{q} - (\dot{q}_d - \Lambda_p e)] = P_\phi[\dot{q} + \Lambda_p e] \tag{3.16}$$

The control law for the transition phase in normal direction is chosen as follows:

$$\tau_n = Y(q, \dot{q}, \dot{q}_{rn}, \ddot{q}_{rn})\beta - F_v e_{vn} - \lambda_{tn} P_\phi \ \mathrm{sgn}(e_{vn}) \tag{3.17}$$

where $\mathrm{sgn}(e_{vn})$ is the component-wise sign of vector e_{vn}, and $\dot{q}_{rn}$ is the reference velocity projected onto the normal subspace of the constraint, i.e.,

$$\dot{q}_{rn} = P_\phi(\dot{q}_d - \Lambda_p e) = -P_\phi(\Lambda_p e) \tag{3.18}$$

$$\ddot{q}_{rn} = -P_\phi \Lambda_p \dot{e} - \dot{P}_\phi \Lambda_p e \tag{3.19}$$

The additional term $\lambda_{tn} P_\phi \ \mathrm{sgn}(e_{vn})$ in (3.17) is a discontinuous term based on the reference velocity projected into the normal direction. Only normal position and velocity errors are used to generate this term to prevent robot from leaving the surface or force robot back to the surface in the event of loss of contact. Since no force feedback is involved, this term provides the controller robustness to impact forces.

The transition controller for both normal and tangential directions is obtained by combining (3.12) and (3.17) as

$$\tau = \tau_t + \tau_n \tag{3.20}$$

$$= Y(q, \dot{q}, \dot{q}_{rt}, \ddot{q}_{rt})\beta - F_v e_{vt} \tag{3.21}$$

$$+Y(q, \dot{q}, \dot{q}_{rn}, \ddot{q}_{rn})\beta - F_v e_{vn} - \lambda_{tn} P_\phi \ \mathrm{sgn}(e_{vn})$$

Choosing suitable forms of regressor matrices $Y(q, \dot{q}, \dot{q}_{rt}, \ddot{q}_{rt})$ and $Y(q, \dot{q}, \dot{q}_{rn}, \ddot{q}_{rn})$ yields

$$Y(q, \dot{q}, \dot{q}_{rt}, \ddot{q}_{rt}) + Y(q, \dot{q}, \dot{q}_{rn}, \ddot{q}_{rn}) = Y(q, \dot{q}, (\dot{q}_{rt} + \dot{q}_{rn}), (\ddot{q}_{rt} + \dot{q}_{rn})) \tag{3.22}$$

Then, the control input for normal and tangential subspaces becomes

$$\tau = Y(q, \dot{q}, (\dot{q}_{rt} + \dot{q}_{rn}), (\ddot{q}_{rt} + \dot{q}_{rn})) - F_v(e_{vt} + e_{vn}) - \lambda_{tn} P_\phi \, \mathrm{sgn}(e_{vn}) \tag{3.23}$$

Let $\dot{q}_r = \dot{q}_{rn} + \dot{q}_{rt}$ and $\ddot{q}_r = \ddot{q}_{rn} + \ddot{q}_{rt}$, using (3.13),(3.18) and (3.14), (3.19),

$$\dot{q}_r = Q_\phi \dot{q}_d - (Q_\phi + P_\phi)\Lambda_p e \tag{3.24}$$

$$\ddot{q}_r = Q_\phi \ddot{q}_d - (Q_\phi + P_\phi)\Lambda_p \dot{e} + \dot{Q}_\phi \dot{q}_d - \Lambda_p(\dot{Q}_\phi + \dot{P}_\phi)e \tag{3.25}$$

Since $Q_\phi + P_\phi = I_n$ and $\dot{Q}_\phi + \dot{P}_\phi = 0$, (3.24) and (3.25) can be written as

$$\dot{q}_r = Q_\phi \dot{q}_d - \Lambda_p e \tag{3.26}$$

$$\ddot{q}_r = Q_\phi \ddot{q}_d + \dot{Q}_\phi \dot{q}_d - \Lambda_p \dot{e} \tag{3.27}$$

Also let $e_v = e_{vt} + e_{vn}$, using (3.15) and (3.16), the combined reference velocity becomes

$$\begin{aligned} e_v &= (Q_\phi + P_\phi)\dot{q} - Q_\phi \dot{q}_d + (Q_\phi + P_\phi)\Lambda_p e \\ &= \dot{q} - Q_\phi \dot{q}_d + \Lambda_p e \end{aligned} \tag{3.28}$$

Using the definitions of (3.26), (3.27), and (3.28), the overall control input during transition phase becomes

$$\tau = Y(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\beta - F_v e_v - \lambda_{tn} P_\phi \mathrm{sgn}(e_{vn}) \tag{3.29}$$

Substituting the control law into the dynamic equation (2.33), we obtain

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = Y(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\beta - F_v e_v - \lambda_{tn} P_\phi \mathrm{sgn}(e_{vn}) \tag{3.30}$$

Since

$$Y(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\beta = M(q)\ddot{q}_r + C(q, \dot{q})\dot{q}_r + g(q)$$

(3.30) becomes

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) - F_v e_v - \lambda_{tn} P_\phi \text{sgn}(e_{vn}) \tag{3.31}$$

Simplifying this equation, yields in the closed-loop error dynamics.

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = -\lambda_{tn} P_\phi \text{sgn}(e_{vn}) \tag{3.32}$$

Stability of the closed-loop system with the proposed control law is shown next.

Stability: Consider the following Lyapunov function candidate,

$$V_T(q, e_v) = \frac{1}{2} e_v^\top M(q) e_v \tag{3.33}$$

Taking the time derivative of the Lyapunov function candidate along the trajectories of (3.32) yields

$$\begin{aligned} \dot{V}_T(e_v) &= \tfrac{1}{2} e_v^\top \dot{M}(q) e_v + e_v^\top M(q)\dot{e}_v \\ &= \tfrac{1}{2} e_v^\top \dot{M}(q) e_v - e_v^\top [C(q,\dot{q})e_v + F_v e_v + \lambda_{tn} P_\phi \text{sgn}(e_{vn})] \\ &= e_v^\top [\tfrac{1}{2}\dot{M}(q) - C(q,\dot{q})] e_v - e_v^\top F_v e_v - e_v^\top \lambda_{tn} P_\phi \text{sgn}(e_{vn}) \end{aligned}$$

Using the skew-symmetry of the matrix $[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]$ and simplifying, the above equation becomes

$$\dot{V}_T(e_v) = -e_v^\top F_v e_v - \lambda_{tn} e_v^\top P_\phi \text{ sgn}(e_{vn}) \tag{3.34}$$

Since $e_{vn} = P_\phi e_v$ and $P_\phi^\top = P_\phi$,

$$\dot{V}_T(e_v) = -e_v^\top F_v e_v - \lambda_{tn} e_{vn}^\top \text{sgn}(e_{vn}) \tag{3.35}$$

the derivative of the Lyapunov function candidate can be bounded as follows:

$$\dot{V}_T(e_v) \leq -\sigma_v \|e_v\|^2 - \lambda_{tn} |e_{vn}| \tag{3.36}$$

where σ_v is the minimum eigenvalue of F_v. Therefore, (3.33) and (3.36) mean that $V_T(e_v)$ is indeed a Lyapunov function, i.e., $V_T(e_v)$ is positive definite and its time-derivative along

the trajectories of (3.32) is negative definite. Thus, the system is stable in the sense of Lyapunov. Using Lyapunov theorem, e_v is bounded. Since the orthogonal projection matrix $P_\phi(q)$ is bounded, e_{vn} is bounded. Therefore, the left-hand-side of (3.32) is bounded. Combined with the fact that manipulator matrix $M(q)$ is always non-singular during the motion, from (3.32), it can be shown that $\dot{e}_v$ is bounded. This means that e_v is uniformly continuous. Also, e_v appears explicitly in $\dot{V}(e_v)$, e_v converges to zero asymptotically. Notice that $e_v = P_\phi e_v + Q_\phi e_v$, where the two terms are orthogonal to each other. Therefore, $e_v \to 0$ implies that the projection of e_v in both tangential and normal directions converges to zero, i.e., $e_{vn} \to 0$ and $e_{vt} \to 0$.

In the preceding, it has been shown that the Lyapunov function decreases between impacts. If a series of impacts are involved during the transition phase, decrease of the Lyapunov function after every impact has to be shown. Impact model given by (2.16) describes the behavior of the robot at the moment of impact with the constraint surface. Let A^- and A^+ represent a variable or function before and after an impact. Assume that robot end-effector impacts the constraint surface at time instances t_k, where k represents the number of impact points and is assumed to be finite. Let $e_{vk}^+ = e_v(t_k + \delta)$ and $e_{vk}^- = e_v(t_k - \delta)$, where $\delta \to 0$. The change in Lyapunov function during the k-th impact is

$$\begin{aligned} \Delta V_{T_k} : &= V_{T_k}^+ - V_{T_k}^- \\ &= \frac{1}{2}(e_{vk}^+)^\top M(q)(e_{vk}^+) - \frac{1}{2}(e_{vk}^-)^\top M(q)(e_{vk}^-) \end{aligned} \tag{3.37}$$

From (3.28), the post-impact (e_{vk}^+) and pre-impact reference velocity (e_{vk}^-) are given by

$$e_{vk}^+ = \dot{q}_k^+ - \dot{q}_{kr}^+ \tag{3.38}$$

$$e_{vk}^+ = \dot{q}_k^- - \dot{q}_{kr}^- \tag{3.39}$$

Using (3.26),

$$\dot{q}_{kr}^+ = Q_\phi^+ \dot{q}_d^+ - \Lambda_p e^+ \tag{3.40}$$

$$\dot{q}_{kr}^- = Q_\phi^- \dot{q}_d^- - \Lambda_p e^- \tag{3.41}$$

Since the desired trajectories are smooth and continuous, $\dot{q}_d^+ = \dot{q}_d^-$ and $q_d^+ = q_d^-$ is true at any point. Assuming that impact occurs over an infinitesimally small period of time, there is no change in the position of the system, i.e., $q^+ = q^-$. Hence, $Q_\phi^+ = Q_\phi^-$ and $e^+ = e^-$ because $q^+ = q^-$ and $q_d^+ = q_d^-$. Further

$$\dot{q}_{kr}^+ = \dot{q}_{kr}^- = \dot{q}_{kr} \tag{3.42}$$

Substituting (3.38) and (3.39) into (3.37), we obtain

$$\Delta V_{T_k} = \frac{1}{2}(\dot{q}_k^+ - \dot{q}_{kr})^T M(q)(\dot{q}_k^+ - \dot{q}_{kr}) - \frac{1}{2}(\dot{q}_k^- - \dot{q}_{kr})^T M(q)(\dot{q}_k^- - \dot{q}_{kr}) \tag{3.43}$$

Expanding the above equation and rearranging,

$$\begin{aligned} \Delta V_{T_k} \;=\; & \frac{1}{2}(\dot{q}_k^+)^T M(q)(\dot{q}_k^+) - \frac{1}{2}(\dot{q}_k^-)^T M(q)(\dot{q}_k^-) \\ & -[(\dot{q}_{kr})^T M(q)(\dot{q}_k^+) - (\dot{q}_{kr})^T M(q)(\dot{q}_k^-)] \end{aligned} \tag{3.44}$$

The terms $\frac{1}{2}(\dot{q}_k^+)^T M(q)(\dot{q}_k^+)$ and $\frac{1}{2}(\dot{q}_k^-)^T M(q)(\dot{q}_k^-)$ are manipulator post-impact and pre-impact kinetic energy $\mathcal{K}^+$ and $\mathcal{K}^-$, respectively. Simplifying (3.44) yields

$$\Delta V_{T_k} = \mathcal{K}^+ - \mathcal{K}^- - (\dot{q}_{kr})^\top M(q)(\dot{q}_k^+ - \dot{q}_k^-) \tag{3.45}$$

where $(\dot{q}_k^+ - \dot{q}_k^-) = \sigma_{\dot{q}}$ is the velocity change caused by impact. Hence,

$$\Delta V_{T_k} = \mathcal{K}^+ - \mathcal{K}^- - (\dot{q}_{kr})^\top M(q)\sigma_{\dot{q}} \tag{3.46}$$

At impact points, the robot end-effector is on the surface. This implies that the normal position error is zero, i.e., $P_\phi e_k = 0$. From the definition of reference velocity $\dot{q}_r$ in (3.26),

$$\dot{q}_{kr} = Q_\phi \dot{q}_{kd} - \Lambda_p e_k = Q_\phi(\dot{q}_{kd} - \Lambda_p e_k) \tag{3.47}$$

The Lyapunov function change due to impact becomes:

$$\Delta V_{T_k} = \mathcal{K}^+ - \mathcal{K}^- - [Q_\phi(\dot{q}_{kd} - \Lambda_p e_k)]^\top M(q)\sigma_{\dot{q}} \tag{3.48}$$

Using (2.18) yields

$$\Delta V_{T_k} = \mathcal{K}^+ - \mathcal{K}^- - (\dot{q}_{kd} - \Lambda e_k)^\top Q_\phi^\top J(q)^\top f_I \tag{3.49}$$

Since $Q_\phi^\top = Q_\phi$, using (2.23) results in $Q_\phi^\top J(q)^\top f_I = 0$. Thus,

$$\Delta V_{T_k} = \mathcal{K}^+ - \mathcal{K}^- \tag{3.50}$$

which is the kinetic energy change due to impact. Using the impact model assumption discussed in section 2.2, equation (2.27), yields

$$\Delta V_{T_k} = -\Delta \mathcal{K} < 0 \tag{3.51}$$

Therefore, the Lyapunov function decreases at impact points. Since it has been shown that the Lyapunov function decreases between impact points, the overall system involving a series of impacts is also stable in the sense of Lyapunov. The reference velocity error e_v is bounded during the motion between impact and at the impact points. The right-hand-side of (3.32) is bounded and $M(q)$ is nonsingular during impacts. This implies that $\dot{e}_v$ is bounded during impacts, hence e_v is uniformly continuous during the transition phase. Considering impacts, $e_v \to 0$ as $t \to \infty$ in the transition phase.

3.3 Position/Force Control for Constrained Motion

The constrained motion phase starts when the robot end-effector is in stable contact with the constraint surface. During constrained motion phase, the control goal is to simultaneously track the desired motion in tangential direction, regulate the desired force normal to the constraint surface, and provide enough cutting force in the tangential direction. Motion control including tangential force compensation is applied in the tangential subspace of the constraint. Integral force control is used in the subspace normal to the constraint. An adaptation scheme is designed to estimate the grinding coefficient ξ, which relates the normal and tangential components of the contact force.

As stated in section 2.3, during constrained motion phase, manipulator dynamics is

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + v(q)f_n + v'(q)\xi f_n \tag{3.52}$$

For this phase, the following control law is chosen:

$$\tau = M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) - F_v e_v - v(q)f_{nd} - v'(q)\widehat{\xi} f_n \tag{3.53}$$

$$\widehat{\xi}(t) = \xi_0 - \int_0^t \gamma_f e_v^\top v'(q) f_n dt \tag{3.54}$$

where f_{nd} is the desired normal force, $\widehat{\xi}(t)$ is the estimate of grinding coefficient ξ, and

$$\dot{q}_r = Q_\phi[\dot{q}_d - \Lambda_p e] + \beta_f \bar{v}(q) e_{vfn} \tag{3.55}$$

$$e_v = \dot{q} - \dot{q}_r \tag{3.56}$$

The other variables in (3.53) are

$$v(q) = J^\top n(q)$$

$$v'(q) = J^\top t(q)$$

$$\bar{v}(q) = v(q)/\|v(q)\|^2$$

$$e_{fn} = f_n - f_{nd}$$

$$e_{vfn} = \int_0^t e_{fn}(\omega) d\omega$$

Notice that the reference velocity $\dot{q}_r$ defined in (3.55) is composed of two components that are orthogonal to each other. The first term on the right hand of (3.55), $Q_\phi[\dot{q}_d - \Lambda_p e]$, depends on the desired velocity and the position tracking error in the subspace tangential to the constraint, whereas the second term, $\beta_f \bar{v}(q) e_{vfn}$, is the integral feedback force error in the direction normal to the surface. Since the robot end-effector moves along the constraint during this phase, $P_\phi \dot{q} = 0$ is always true. This implies that $\dot{q} = Q_\phi \dot{q}$, hence (3.56) can be written as

$$e_v = Q_\phi \dot{q} - \dot{q}_r \tag{3.57}$$

Substituting (3.55) into (3.57) yields:

$$e_v = Q_\phi[\dot{e} + \Lambda_p e] - \beta_f \bar{v}(q) e_{vfn} \tag{3.58}$$

Equation (3.58) indicates that the reference velocity error e_v defined by (3.56) can also be partitioned into two parts orthogonal to each other. The first term on right hand side of (3.58) depends on tracking error projected in the tangential subspace. The second term is the integral force error normal to the constraint.

The overall control input in (3.53) applies position control and force control in two subspaces orthogonal to each other. The tangential subspace to the constraint is position control subspace, whereas the normal subspace to the constraint is the force-control subspace. In the position control subspace, a tangential force compensation term is used to provide enough cutting force in motion direction. The tangential force feedforward term $v'(q)\widehat{\xi} f_n$ is related to the magnitude of the normal force f_n via a constant grinding friction coefficient ξ. A gradient type adaptation law (3.54) is designed to estimate the value of ξ.

Substituting the control law (3.53) into the dynamic equation (3.52) yields

$$\begin{aligned} M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) &= M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) - F_v e_v \\ &\quad - v(q) f_{nd} - v'(q)\widehat{\xi} f_n + v(q) f_n + v'(q)\xi f_n \end{aligned} \tag{3.59}$$

Simplifying this equation yields the error dynamics

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = v(q) e_{fn} + v'(q)\widetilde{\xi} f_n \tag{3.60}$$

Stability: Choose the following Lyapunov function candidate:

$$V_C'(e_v, e_{fn}) = \frac{1}{2} e_v^\top M(q) e_v + \frac{1}{2}\beta_f e_{vfn}^2 \tag{3.61}$$

Taking the time derivative of the Lyapunov function candidate along the trajectories of

(3.60) yields

$$\begin{aligned}\dot{V}_C'(e_v, e_{fn}) &= e_v^\top M(q)\dot{e}_v + \frac{1}{2}e_v^\top \dot{M}(q)e_v + \beta_f e_{fn}e_{vfn} \\ &= e_v^\top[-C(q,\dot{q})e_v - F_v e_v + v(q)e_{fn} + v'(q)\tilde{\xi}f_n] + \frac{1}{2}e_v^\top \dot{M}(q)e_v + \beta_f e_{fn}e_{vfn} \\ &= e_v^\top[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]e_v - e_v^\top F_v e_v + e_v^\top v(q)e_{fn} + e_v^\top v'(q)\tilde{\xi}f_n + \beta_f e_{fn}e_{vfn}\end{aligned}$$

since $[\frac{1}{2}\dot{M}(q) - C(q,\dot{q})]$ is skew-symmetric, we have

$$\dot{V}_C'(e_v, e_{fn}) = -e_v^\top F_v e_v + e_v^\top v(q)e_{fn} + e_v^\top v'(q)\tilde{\xi}f_n + \beta_f e_{fn}e_{vfn} \tag{3.62}$$

The term $e_v^\top v(q)e_{fn}$ can be simplified as follows:

$$\begin{aligned}e_v^\top v(q)e_{fn} &= [\dot{q} - \dot{q}_r]v(q)e_{fn} \\ &= [(P_\phi + Q_\phi)\dot{q} - Q_\phi(\dot{q}_d - \Lambda_p e) - \beta_f \bar{v}(q)e_{vfn}]^\top v(q)e_{fn} \\ &= [Q_\phi\dot{q} - Q_\phi(\dot{q}_d - \Lambda_p e) - \beta_f \bar{v}(q)e_{vfn}]^\top v(q)e_{fn} \\ &= [Q_\phi(\dot{e} - \Lambda_p e) - \beta_f \bar{v}(q)e_{vfn}]^\top v(q)e_{fn} \\ &= (\dot{e} - \Lambda_p e)^\top Q_\phi^\top v(q)e_{fn} - \beta_f e_{vfn}\bar{v}^\top(q)v(q)e_{fn} \\ &= -\beta_f e_{vfn}\frac{v^\top(q)}{\|v(q)\|^2}v(q)e_{fn} \\ &= -\beta_f e_{vfn}e_{fn}\end{aligned} \tag{3.63}$$

where the fact that $P_\phi(q)\dot{q} = 0$ is used. Substituting (3.63) into the derivative of the Lyapunov function candidate, (3.62), yields

$$\dot{V}_C'(e_v, e_{fn}) = -e_v^\top F_v e_v + e_v^\top v'(q)\tilde{\xi}f_n \tag{3.64}$$

To show stability in the presence of uncertainty of the grinding friction coefficient, the Lyapunov function candidate V_C' is modified to

$$V_C = V_C' + \frac{1}{2}\gamma_f^{-1}\tilde{\xi}^2 \tag{3.65}$$

The time derivative of the modified Lyapunov function candidate (3.65) is

$$\begin{aligned}\dot{V}_C(e_v, f_n) &= \dot{V}_C' + \gamma_f^{-1}\tilde{\xi}\dot{\tilde{\xi}} \\ &= -e_v^\top F_v e_v + e_v^\top v'(q)\tilde{\xi}f_n + \dot{\tilde{\xi}}\gamma_f^{-1}\tilde{\xi} \\ &= -e_v^\top F_v e_v + [e_v^\top v'(q)f_n + \dot{\tilde{\xi}}\gamma_f^{-1}]\tilde{\xi}\end{aligned} \tag{3.66}$$

Taking the time derivative of adaptation law given by (3.54) yields

$$\dot{\widehat{\xi}} = -\gamma_f e_v^\top v'(q) f_n \tag{3.67}$$

Since that ξ is a constant, i.e., $\dot{\xi} = 0$. We have $\dot{\widetilde{\xi}} = \dot{\widehat{\xi}}$. Hence,

$$\dot{\widetilde{\xi}} = -\gamma_f e_v^\top v'(q) f_n \tag{3.68}$$

Substituting (3.68) into (3.66) yields

$$\dot{V}_C(e_v) = -e_v^\top F_v e_v < 0 \tag{3.69}$$

Since V_C is a positive definite function and $\dot{V}_C$ is negative definite, V_C give by (3.65) is a Lyapunov function. In constrained motion phase, the system is stable in the sense of Lyapunov. Lyapunov function V_C is a positive function that is bounded from below, hence e_v , e_{vfn}, and $\widetilde{\xi}$ are bounded. Notice that e_v consists with two orthogonal terms, $Q_\phi(\dot{e} - \Lambda e)$ and $\beta_f \bar{v}(q) e_{vfn}$. Boundedness of e_v implies that these two terms are bounded too. If assuming impact force is bounded provided the pre-impact velocity is bounded, left-hand-side of (3.60) remains bounded and manipulator inertia matrix $M(q)$ is never singular, $\dot{e}_v$ is bounded from (3.60). This means that e_v is uniformly continuous. Furthermore, since e_v appears explicitly in $\dot{V}_C$, e_v converges to zero asymptotically. Therefore, the two orthogonal terms of e_v, the tangential tracking error $Q_\phi(\dot{e} - \Lambda e)$ and normal force integral e_{vfn}, converge to zero asymptotically.

3.4 Controller for a complete task

The controller for a complete task is obtained by integrating the controller for each phase. An event based control switching strategy is developed for the complete task. In unconstrained motion phase, a model-based adaptive control law is applied for trajectory tracking considering uncertainties in robot model. Upon contact with the surface, control is switched to the one given by (3.29). In the transition phase the primary goal is to stabilize the robot

end-effector onto the surface. When the end-effector is on the surface, robot switches to the constrained motion control law given by (3.53). An event based online trajectory planning is utilized along with the control switch strategy. The desired trajectory is pre-computed based on the initial knowledge of the location of the constraint surface. Immediately after the first impact, pre-computed trajectories are modified such that the desired velocity normal to the constraint surface is zero. After the completion of the transition phase, the trajectory planner prescribes a desired force in the direction normal to the constraint surface and a desired joint space trajectory projected into the tangential direction of the constraint surface. In the following, a summary of the closed-loop error dynamics and Lyapunov functions are given for each phase.

- In the unconstrained motion phase, the error dynamics of the closed loop system is given by

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = Y(q,\dot{q},\dot{q}_r,\ddot{q}_r)\tilde{\beta}$$

where $e_v = \dot{e} + \Lambda_p e$, $\dot{q}_r = \dot{q}_d - \Lambda_p e$, $F_v > 0$, and $\Lambda_p > 0$. Stability can be shown with the Lyapunov function V_U given by

$$V_U(e_v,\tilde{\beta}) = \frac{1}{2}e_v^\top M(q)e_v + \frac{1}{2}\tilde{\beta}^\top \Gamma \tilde{\beta}$$

whose derivative along the trajectories of the error dynamics satisfies

$$\dot{V}_U(e_v,\tilde{\beta}) = -e_v^\top F_v e_v < 0$$

- In the transition phase, the error dynamics of the closed loop system between impact points is given by

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = -\lambda_{tn} P_\phi \mathrm{sgn}(e_{vn})$$

where $e_v = \dot{e} + \Lambda_p e$, $e_{vn} = P_\phi e_v$, $F_v > 0$, $\Lambda_p > 0$, and $\lambda_{tn} > 0$. P_ϕ is the normal projection matrix. Stability can be shown with the Lyapunov function V_T given by

$$V_T(q,e_v) = \frac{1}{2}e_v^\top M(q)e_v$$

whose derivative between impact points can be bounded as follows

$$\dot{V}_T(e_v) \leq -\sigma_v \|e_v\|^2 - \lambda_{tn}|e_{vn}|$$

At impact points, the Lyapunov function change is the kinetic energy change due to impact and is given by

$$\Delta V_{T_k} = -\Delta \mathcal{K} < 0$$

Hence, the Lyapunov function decreases during the transition phase.

- In the constrained motion phase, the error dynamics of the closed-loop system is given by

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + F_v e_v = v(q)e_{fn} + v'(q)\widetilde{\xi} f_n$$

where $e_v = Q_\phi[\dot{e} + \Lambda_p e] - \beta_f \bar{v}(q) e_{vfn}$, Q_ϕ is tangential projection matrix, $F_v > 0$, and $\Lambda_p > 0$. The definition of other variables can be found in section 3.3. The Lyapunov function for this phase is V_C

$$V_C = \frac{1}{2} e_v^\top M(q) e_v + \frac{1}{2}\beta_f e_{vfn}^2 + \frac{1}{2}\gamma_f^{-1}\tilde{\xi}^2$$

and its derivative along the trajectories of the error dynamics satisfies

$$\dot{V}_C(e_v) = -e_v^\top F_v e_v < 0$$

The closed-loop system is shown to be stable in each phase. Notice that the reference velocity error, e_v, in each phase is different even though the same notation is used. So, comparison of Lyapunov function between phases is not practical. It is assumed that time is long enough such that $\|e_v\| < \delta_{ev}$ at the end of each phase. For the stability proof for the overall system, it is essential to verify that the system states and Lyapunov functions are bounded at control switching points.

At the control switching point from the unconstrained motion phase to the transition phase, say at time t_{sw1}, the pre-switch and post-switch reference velocities, i.e., $e_v^-(t_{sw1})$ and $e_v^+(t_{sw1})$, are

$$e_v^-(t_{sw1}) = \dot{e}^-(t_{sw1}) + \Lambda_p e^-(t_{sw1}) \tag{3.70}$$

$$e_v^+(t_{sw1}) = \dot{e}^+(t_{sw1}) + \Lambda_p e^+(t_{sw1}) \tag{3.71}$$

In the case of no constraint uncertainty, robot lands on the constraint smoothly and the desired trajectory is not modified after control switch, i.e., $\dot{e}^-(t_{sw1}) = \dot{e}^+(t_{sw1})$ and $e^-(t_{sw1}) = e^+(t_{sw1})$, hence

$$e_v^-(t_{sw1}) = e_v^+(t_{sw1}) \tag{3.72}$$

In the unconstrained motion phase, the reference velocity error satisfies $\|e_v^-(t_{sw1})\| < \delta_{ev}$. Hence, at the beginning of the transition phase, $\|e_v^+(t_{sw1})\| < \delta_{ev}$ is also true after control switch. Lyapunov function $V_T^+(t_{sw1})$ is bounded and the system states are bounded at the beginning of the transition phase. In the case of constraint uncertainty and/or non-zero normal impact velocity, the desired trajectory is modified after the first impact such that $P_\phi \dot{q}_d^+(t_{sw1}) = 0$ and $P_\phi e^+(t_{sw1}) = 0$, implying $\dot{q}_d^+(t_{sw1}) = Q_\phi \dot{q}_d^-(t_{sw1})$ and $e^+(t_{sw1}) = Q_\phi e^-(t_{sw1})$. The reference velocity after control switch becomes

$$\begin{aligned} e_v^+(t_{sw1}) &= \dot{e}^+(t_{sw1}) + \Lambda_p e^+(t_{sw1}) \\ &= \dot{q}^+(t_{sw1}) - \dot{q}_d^+(t_{sw1}) + \Lambda_p e^+(t_{sw1}) \\ &= \dot{q}^+(t_{sw1}) - Q_\phi \dot{q}_d^-(t_{sw1}) + \Lambda_p Q_\phi e^-(t_{sw1}) \end{aligned}$$

Using the impact model (2.16), $\dot{q}^+(t_{sw1}) = \mathcal{D}(q, \dot{q}^-(t_{sw1}))$, gives

$$e_v^+(t_{sw1}) = \mathcal{D}(q, \dot{q}^-(t_{sw1})) - Q_\phi \dot{q}_d^-(t_{sw1}) + \Lambda_p Q_\phi e^-(t_{sw1}) \tag{3.73}$$

Hence, $e_v^+(t_{sw1})$ can be bounded as follows

$$\|e_v^+(t_{sw1})\| \leq \lambda_1 \|\dot{q}^-(t_{sw1})\| + \lambda_2 \|\dot{q}_d^-(t_{sw1})\| + \lambda_3 \|e^-(t_{sw1})\| \tag{3.74}$$

where λ_1, λ_2, and λ_3 are finite positive values. The three terms on the right-hand-side of above equation are all bounded during the unconstrained motion phase. Then the Lyapunov function $V_T^+(t_{sw1})$ and the system states are bounded after switching from the unconstrained motion phase to the transition phase.

At the control switching point from the transition phase to the constrained motion phase, say at time t_{sw2}, the pre-switch and post-switch reference velocities, i.e., $e_v^-(t_{sw2})$ and $e_v^+(t_{sw2})$, are

$$e_v^-(t_{sw2}) = \dot{e}^-(t_{sw2}) + \Lambda_p e^-(t_{sw2}) \tag{3.75}$$

$$e_v^+(t_{sw2}) = Q_\phi[\dot{e}^+(t_{sw2}) + \Lambda_p e^+(t_{sw2})] - \beta_f \bar{v}(q) e_{vfn} \tag{3.76}$$

where $e_{vfn} = \int_0^{\Delta t_{sw}} (f_n(\omega) - f_{nd}) d\omega$, Δt_{sw} is the switching time. Since both f_n and f_{nd} are finite values, $e_{vfn} \to 0$ as $\Delta t_{sw} \to 0$. At the end of the transition phase, the robot end-effector is on the constraint surface, i.e., $P_\phi \dot{q}^-(t_{sw2}) = 0$ and $P_\phi e^-(t_{sw2}) = 0$. Further, no impact force is involved at this switch point, hence $e^+(t_{sw2}) = e^-(t_{sw2})$ and $\dot{e}^+(t_{sw2}) = \dot{e}^-(t_{sw2})$. Thus,

$$\begin{aligned} e_v^+(t_{sw2}) &= Q_\phi[\dot{e}^-(t_{sw2}) + \Lambda_p e^-(t_{sw2})] \\ &= \dot{e}^-(t_{sw2}) + \Lambda_p e^-(t_{sw2}) \\ &= e_v^-(t_{sw2}) \end{aligned}$$

Therefore, $\|e_v^+(t_{sw2})\| < \delta_{ev}$. The Lyapunov function $V_T^+(t_{sw2})$ is bounded and the system states are bounded after switching from the transition phase to the constrained motion phase. Hence the overall closed loop system is stable in the sense of Lyapunov.

Remarks

- The position/force control proposed considers the contact force in tangential subspace to the constraint surface. Unlike most hybrid position/force control schemes available in the literature, the new adaptive hybrid position/force control algorithm has a tangential force feed-forward term in position control subspace. Instead of assuming that the tangential force is negligible or regarding it as a disturbance, the tangential force is compensated by a term related with normal force by a grinding friction coefficient. However, the control goal in the tangential subspace to the constraint is still to track the desired trajectory, not to regulate tangential force.

- The transition control design strategy developed does not require an accurate impact model. The following are assumed from the impact model: (1) the normal component is the dominant term in contact force impulse at impact points; (2) there exists a threshold of normal impact velocity below which the magnitude of the post-impact velocity is zero; (3) there is a finite amount of kinetic energy reduction due to impact. All the three assumptions are verified by the impact experiments.

Chapter 4

Mechatronic Design and Implementation

The objective of this chapter is to present the open architecture experimental platform developed via a mechatronic approach. The hardware and software components are described in detail and controller implementation issues are addressed in this chapter. Based on the knowledge of system components and system architecture design, different function design modes are discussed in section 4.3 for implementation of the discontinuous control scheme proposed in the previous chapter.

4.1 Hardware

The open architecture robot experimental platform consists of a two-link manipulator, computer for real-time control, force sensor, end-effector containing the surface finishing tool, and the constraint fixture. Fig. 4.1 illustrates the complete hardware platform of the robotic surface finishing system.

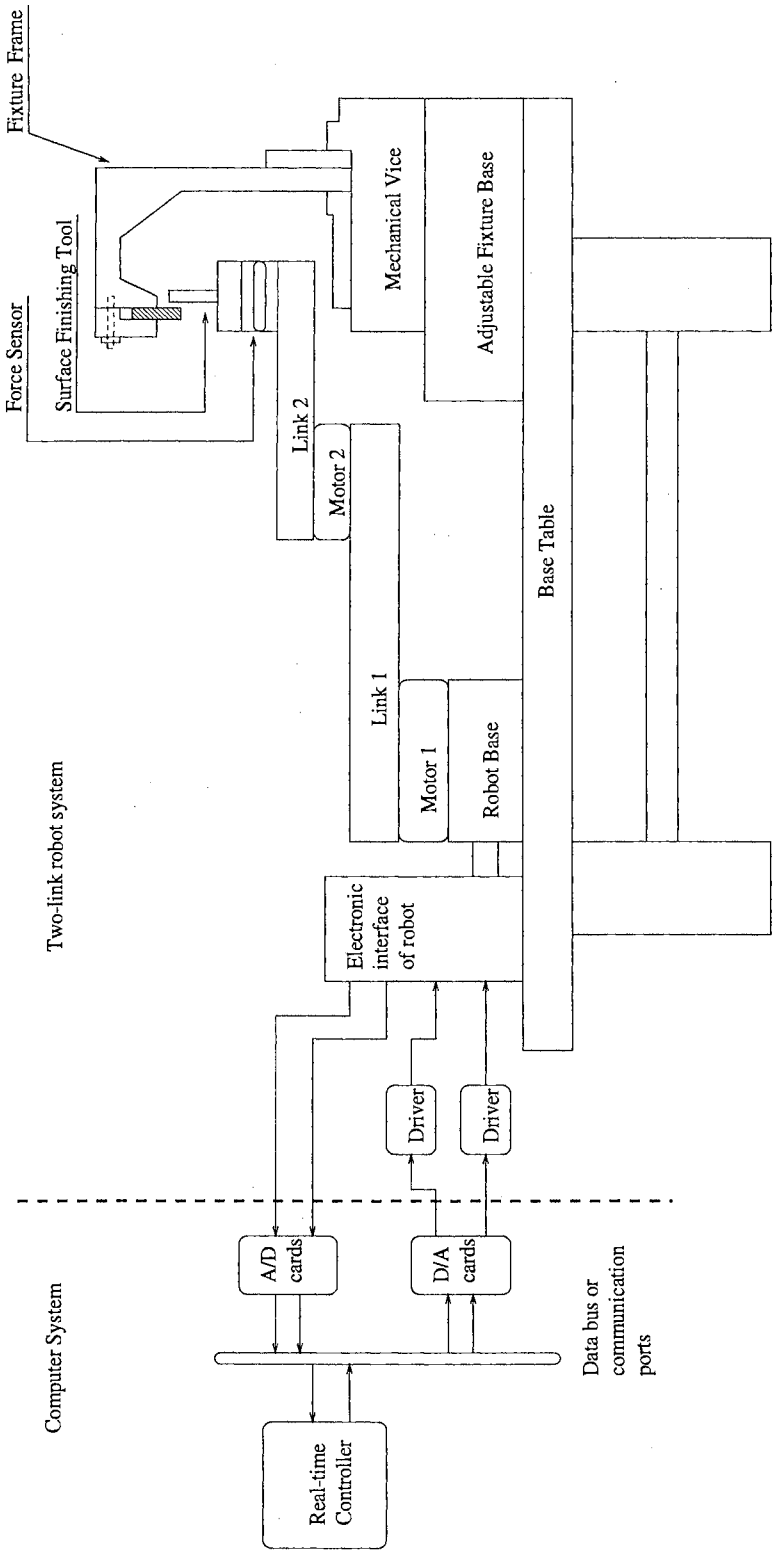

Figure 4.1: Schematic diagram of the open architecture experimental platform

4.1.1 Robot System

The main part of the robot system is a two-axis direct drive manipulator. Direct drive manipulator operates in the absence of the undesirable factors of mechanical backlash and gear train compliance. The direct drive actuator eliminates the need for gear reduction, so repeatability is limited only by the resolution of the position feedback. Also, direct coupling of the motor and load permits tighter and more direct control of the load. Direct drive manipulator systems tend to be difficult to control as payload variations and dynamic uncertainties directly affect the motor performance. Adaptive, learning, and robot nonlinear control schemes for direct drive systems are very active research fields, especially for applications that involve time varying payload, uncertain external forces, and impact forces [195].

Each axis of the robot manipulator is driven by an NSK-Megatorque direct drive servomotor. The NSK-Megatorque motor system consists of motor and its driver unit. This is a stand-alone system that contains all the elements needed for a complete closed-loop servo motor control.

The NSK motor consists of a high torque direct drive brushless actuator, a high-resolution brushless resolver, and a heavy duty precision bearing. The Megatorque motor is capable of producing extremely high torque at low speeds suitable for direct drive applications. The servo-motors are capable of up to 3 revolutions per second maximum velocity and position feedback resolution of up to 156,400 counts per revolution. The base motor delivers up to 245 N-m of torque output, and the elbow motor produces up to 40 N-m torque output. The heavy-duty bearing eliminates the need for separate mechanical support since the motor case can often support the load directly.

The NSK motor driver unit consists of a 16 bit DSP (Digital Signal Processor) system, power amplifier, and resolver interface. A schematic diagram of the driver unit and its relation with real-time robot controller and NSK motor are shown in Fig. 4.2. The DSP, which is a 16 bit microprocessor system, receives position, velocity, and/or torque commands

from host computer via RS232 or from the servo control DSP via a DS2 I/O card. The feedback position and velocity signals are collected by the driver unit DSP from resolvers via a resolver interface circuit. The actual values of the controlled variables such as joint position and NSK motor current are compared with the corresponding command variables from servo controller. Based on the comparison results, the driver unit generates actuation signals continuously.

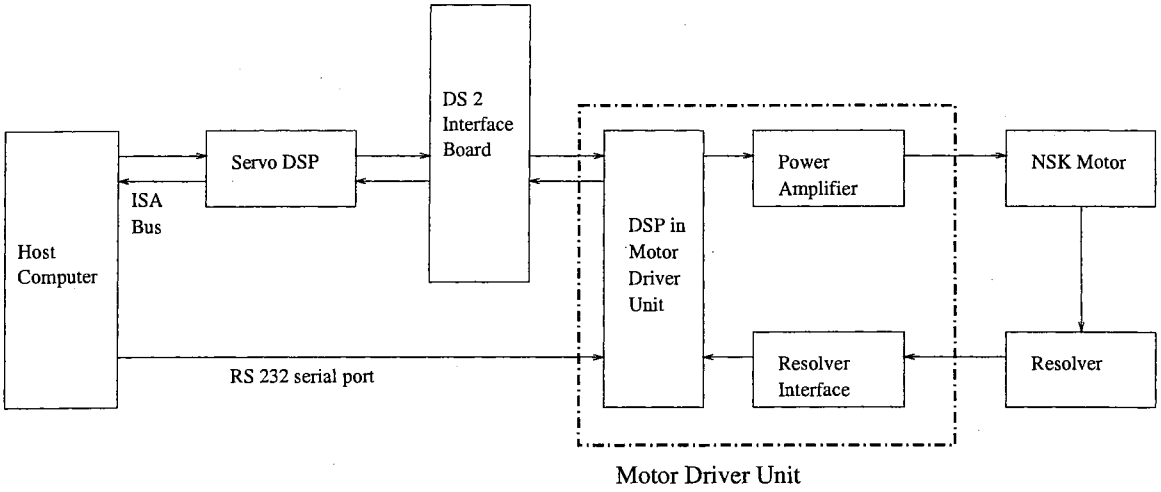

Figure 4.2: Driver unit of NSK motor

Manipulator Dynamics

Fig. 4.3 shows a schematic of the two link robot manipulator of the experimental platform. Table 4.1 and 4.2 show the symbols used to derive the manipulator dynamic equations. The manipulator dynamics are derived using the following Lagrange equations of motion:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \tau_i \tag{4.1}$$

where q_i, $i = 1, 2$, is the joint angle τ_i is the motor torque, and $L = T - P$ is the Lagrangian, where T is the kinetic energy and P is the potential energy of the manipulator. The manipulator is planar and it is installed such that $P = 0$. Let

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \quad \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$

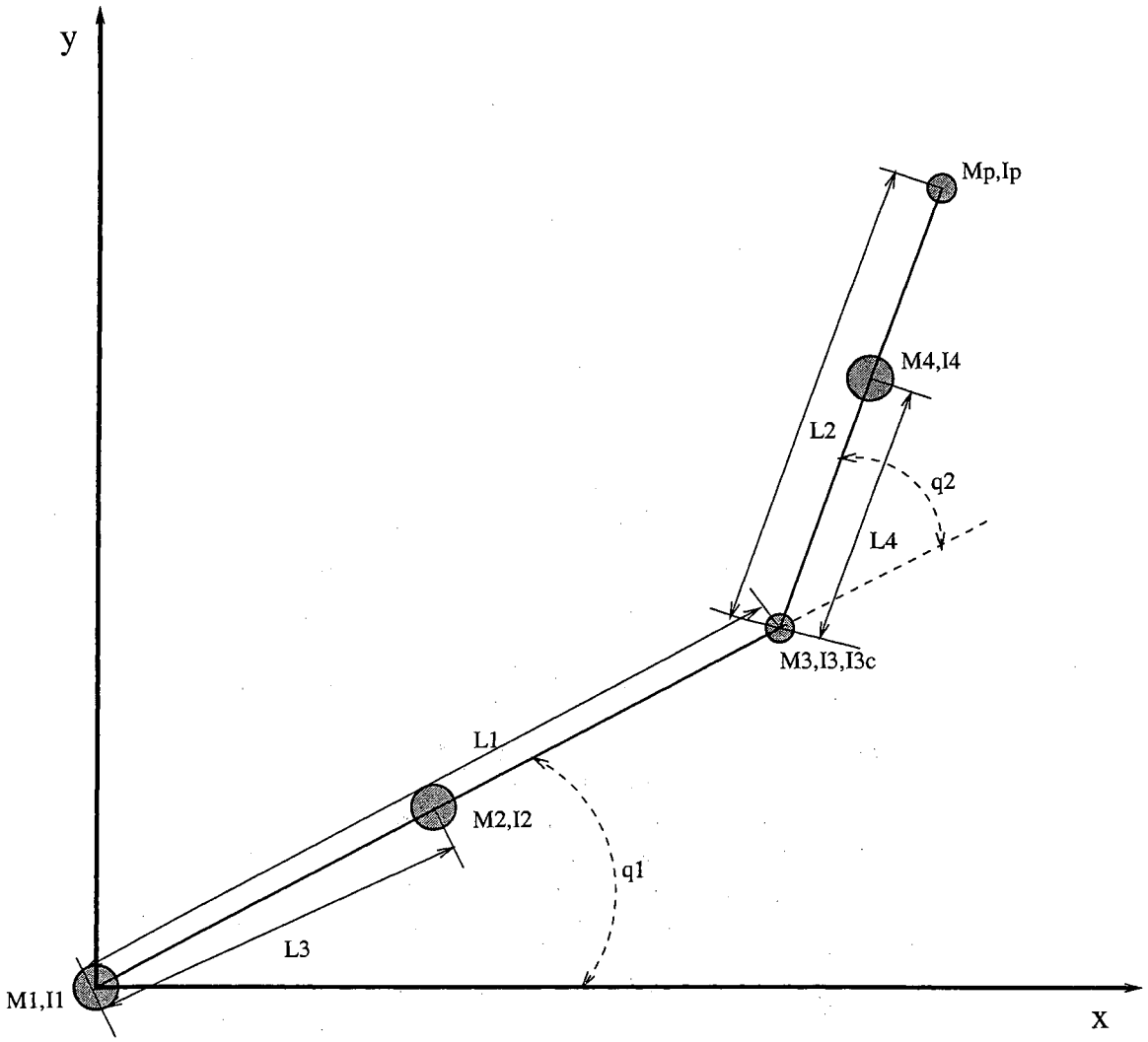

Figure 4.3: Schematic diagram of the two link manipulator

The position and velocity of the center of mass of link 1 are:

$$x_{c2} = \begin{bmatrix} L_3 \cos(q_1) \\ L_3 \sin(q_1) \end{bmatrix} \quad v_{c2} = \frac{dx_{c2}}{dt} = \begin{bmatrix} -L_3 \dot{q}_1 \sin(q_1) \\ L_3 \dot{q}_1 \cos(q_1) \end{bmatrix}$$

The position and velocity of the center of mass of motor 2 are:

$$x_{c3} = \begin{bmatrix} L_1 \cos(q_1) \\ L_1 \sin(q_1) \end{bmatrix} \quad v_{c3} = \frac{dx_{c3}}{dt} = \begin{bmatrix} -L_1 \dot{q}_1 \sin(q_1) \\ L_1 \dot{q}_1 \cos(q_1) \end{bmatrix}$$

I_1	Motor 1 rotor inertia
I_2	Link 1 centroidal moment of inertia
M_1	Motor 1 mass
M_2	Link 1 mass
I_3	Motor 2 rotor inertia
I_4	Link 2 centroidal moment of inertia
M_3	Motor 2 mass
M_4	Link 2 mass
I_{3c}	Motor 2 stater inertia
M_p	Payload mass
I_p	Payload moment of inertia
L_1	Length of link 1
L_2	Length of link 2
L_3	Distance of center of gravity of link 1 from the axis of rotation
L_4	Distance of center of gravity of link 2 from the axis of rotation

Table 4.1: Symbol definitions (1)

The position and velocity of the center of mass of link 2 are:

$$x_{c4} = \begin{bmatrix} L_1\cos(q_1) + L_4\cos(q_1+q_2) \\ L_1\sin(q_1) + L_4\sin(q_1+q_2) \end{bmatrix}$$

$$v_{c4} = \frac{dx_{c4}}{dt} = \begin{bmatrix} -L_1\dot{q}_1\sin(q_1) - L_4(\dot{q}_1+\dot{q}_2)\sin(q_1+q_2) \\ L_1\dot{q}_1\cos(q_1) + L_4(\dot{q}_1+\dot{q}_2)\cos(q_1+q_2) \end{bmatrix}$$

The position and velocity of the center of mass of the payload are:

$$x_{cp} = \begin{bmatrix} L_1\cos(q_1) + L_2\cos(q_1+q_2) \\ L_1\sin(q_1) + L_2\sin(q_1+q_2) \end{bmatrix}$$

q_1	Angle of link 1 with respect to the horizontal
q_2	Angle of link 2 with respect to link 1
τ_1	Torque applied by motor 1
τ_2	Torque applied by motor 2
f_1	Axis 1 friction force
f_2	Axis 2 friction force
x_{c2}	Position vector of center of gravity of link 1
x_{c3}	Position vector of center of gravity of motor 2
x_{c4}	Position vector of center of gravity of link 2
x_{cp}	Position vector of center of gravity of the payload
v_{c2}	Velocity vector of center of gravity of link 1
v_{c3}	Velocity vector of center of gravity of motor 2
v_{c4}	Velocity vector of center of gravity of link 2
v_p	Velocity vector of center of gravity of the payload

Table 4.2: Symbol definitions (2)

$$v_p = \frac{dx_{c4}}{dt} = \begin{bmatrix} -L_1\dot{q}_1\sin(q_1) - L_2(\dot{q}_1+\dot{q}_2)\sin(q_1+q_2) \\ L_1\dot{q}_1\cos(q_1) + L_2(\dot{q}_1+\dot{q}_2)\cos(q_1+q_2) \end{bmatrix}$$

The kinetic energy of the manipulator is given by

$$\begin{aligned} T &= \tfrac{1}{2}[(I_1+I_2+I_{3c})\dot{q}_1^2 + (I_3+I_4+I_p)(\dot{q}_1+\dot{q}_2)^2 \\ &\quad + M_2 v_{c2}^\top v_{c2} + M_3 v_{c3}^\top v_{c3} + M_4 v_{c4}^\top v_{c4} + M_p v_p^\top v_p] \end{aligned} \tag{4.2}$$

Substituting for v_{c2}, v_{c3}, v_{c4} and v_p in the above expression yields:

$$\begin{aligned} T &= \tfrac{1}{2}[(I_1+I_2+I_{3c}+M_3L_1^2+\underline{M_4L_1^2}+M_pL_1^2+\underline{M_2L_3^2})\dot{q}_1^2 \\ &\quad +(I_3+I_4+I_p+M_4L_4^2+M_pL_2^2)(\dot{q}_1+\dot{q}_2)^2 \\ &\quad +2(M_4L_1L_4+M_pL_1L_2)\dot{q}_1(\dot{q}_1+\dot{q}_2)\cos(q_2)] \end{aligned} \tag{4.3}$$

Then,

$$\begin{aligned}\frac{\partial T}{\partial \dot{q}_1} &= (I_1 + I_2 + I_{3c} + M_3 L_1^2 + M_4 L_1^2 + M_p L_1^2 + M_2 L_3^2)\dot{q}_1 \\ &\quad +(I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2)(\dot{q}_1 + \dot{q}_2) \\ &\quad +(M_4 L_1 L_4 + M_p L_1 L_2)(2\dot{q}_1 + \dot{q}_2)\cos(q_2) \end{aligned} \tag{4.4}$$

$$\begin{aligned}\frac{\partial T}{\partial \dot{q}_2} &= (I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2)(\dot{q}_1 + \dot{q}_2) \\ &\quad +(M_4 L_1 L_4 + M_p L_1 L_2)\dot{q}_1 \cos(q_2) \end{aligned} \tag{4.5}$$

$$\frac{\partial T}{\partial q_1} = 0 \tag{4.6}$$

$$\frac{\partial T}{\partial q_2} = -(M_4 L_1 L_4 + M_p L_1 L_2)\dot{q}_1(\dot{q}_1 + \dot{q}_2)\sin(q_2) \tag{4.7}$$

Define the coupled manipulator parameters as

$$\begin{aligned} p_1 =& I_1 + I_2 + I_{3c} + I_3 + I_4 + I_p \\ &+ (M_3 + M_4 + M_p)L_1^2 + M_2 L_3^2 + M_4 L_4^2 + M_p L_2^2 \\ p_2 =& I_3 + I_4 + I_p + M_4 L_4^2 + M_p L_2^2 \\ p_3 =& M_4 L_1 L_4 + M_p L_1 L_2 \end{aligned}$$

Hence

$$\begin{aligned} \frac{d}{dt}\frac{\partial T}{\partial \dot{q}_1} =& p_1\ddot{q}_1 + p_2\ddot{q}_2 + p_3(2\ddot{q}_1 + \ddot{q}_2)\cos(q_2) - p_3(2\dot{q}_1 + \dot{q}_2)\dot{q}_2\sin(q_2) \\ \frac{d}{dt}\frac{\partial T}{\partial \dot{q}_2} =& p_2\ddot{q}_1 + p_2\ddot{q}_2 + p_3\ddot{q}_1\cos(q_2) - p_3\dot{q}_1\dot{q}_2\sin(q_2) \\ \frac{\partial T}{\partial q_1} =& 0 \\ \frac{\partial T}{\partial q_2} =& -p_3\dot{q}_1(\dot{q}_1 + \dot{q}_2)\sin(q_2) \end{aligned}$$

Substituting the above equations into (4.1) yields:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = \tau(t) \tag{4.8}$$

where $M(q)$ is the inertia matrix given by

$$M(q) = \begin{bmatrix} p_1 + 2p_3\cos(q_2) & p_2 + p_3\cos(q_2) \\ p_2 + p_3\cos(q_2) & p_2 \end{bmatrix} \tag{4.9}$$

and $C(q, \dot{q})$ is the Coriolis matrix given by

$$C(q, \dot{q}) = \begin{bmatrix} -\dot{q}_2 p_3 \sin(q_2) & -(\dot{q}_1 + \dot{q}_2) p_3 \sin(q_2) \\ \dot{q}_1 p_3 \sin(q_2) & 0 \end{bmatrix} \tag{4.10}$$

where the mass matrix, $M(q)$, and Coriolis matrix, $C(q, \dot{q})$, are linear in terms of unknown coupled manipulator inertial parameters. Hence, the left-hand-side of (4.8) can be written as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = Y(q, \dot{q}, \ddot{q})\beta$$

where $\beta = [p_1,\ p_2,\ p_3]^\top$ and $Y(q, \dot{q}, \ddot{q})$ is a regressor matrix given by

$$Y = \begin{bmatrix} \ddot{q}_1 & \ddot{q}_2 & \cos(q_2)(2\ddot{q}_1 + \ddot{q}_2) - \dot{q}_2(2\dot{q}_1 + \dot{q}_2)\sin(q_2) \\ 0 & \ddot{q}_1 + \ddot{q}_2 & \ddot{q}_1 \cos(q_2) + \dot{q}_1^2 \sin(q2) \end{bmatrix}$$

4.1.2 Computer System (Controller)

Robot real-time control system consists of the direct drive manipulator controller (host computer), servo DSP, I/O cards, and DSP associated with the sensors. Excluding the DSP in two NSK motor driver units, the overall control system is a three-processor system consisting of a host Pentium processor, a servo DSP (TMS320C30), and a force/torque sensor DSP.

The host processor is used for reference generation, user interface, and coordination of other processors. The servo control algorithm is executed on the servo DSP board, which is a Spectrum TMS320C30 processor. The servo DSP control board is installed on the ISA bus of the host computer and interfaced to the NSK motor driver units through a DS2 motion control I/O card from Integrated Motions Incorporated, Berkeley, CA. DS2 interface card provides 128 Kwords (1 word=32 bits) RAM, 2 A/D converters, 2 shaft encoder interfaces, and 4 bits of parallel I/O ports.

The minimum sampling period that can be implemented is determined by the amount of time required for the host computer to calculate or retrieve references and perform inter-processor communication, approximately 500 μs. The integer and floating-point arithmetic

units equipped on the servo DSP can obtain a peak arithmetic performance of 33.3 million floating-point computations per second. This allows extremely complex algorithms, which are hundreds of lines long to be executed within the minimum 500 μs sample time. The computation power of the servo DSP enables complicated control algorithms implementable with very small sampling period.

The three processor architecture provides the flexibility in terms of collection of force data from the force sensor and position/velocity data from the motor resolvers at different sampling rates. This issue is addressed in more detail in section 4.3. The force sensor DSP is used to filter raw force data, calculate tool offset, and scale/rotate the force data. The next section describes the force sensor and its DSP.

4.1.3 Force Sensor

A six-axis force/torque sensor is mounted on the end of the second-link of the robot manipulator. The force transducer of this sensor has three pairs of low-noise silicon gauge bridges. The signals from these bridges are fed into a low-pass filter and multiplexer, amplified, and connected to a high speed 12-bit A/D converter for use in the control algorithm in the host computer. Fig. 4.4 shows the transducer diagram.

The force sensor has an on-board controller, which can provide force sensor data up to 4 kHz. The primary function of the sensor DSP is to convert strain gauge data to Cartesian force/torque components. Provision is also available for tool weight offset, data filtering, temperature compensation, and frame transfer. Fig. 4.5 is the architecture of the controller card working with the force transducer. The calibration matrix is kept in EEPROM and the tool transformation matrix, transducer conditions and PC commands are stored in dual port RAM on the sensor DSP board.

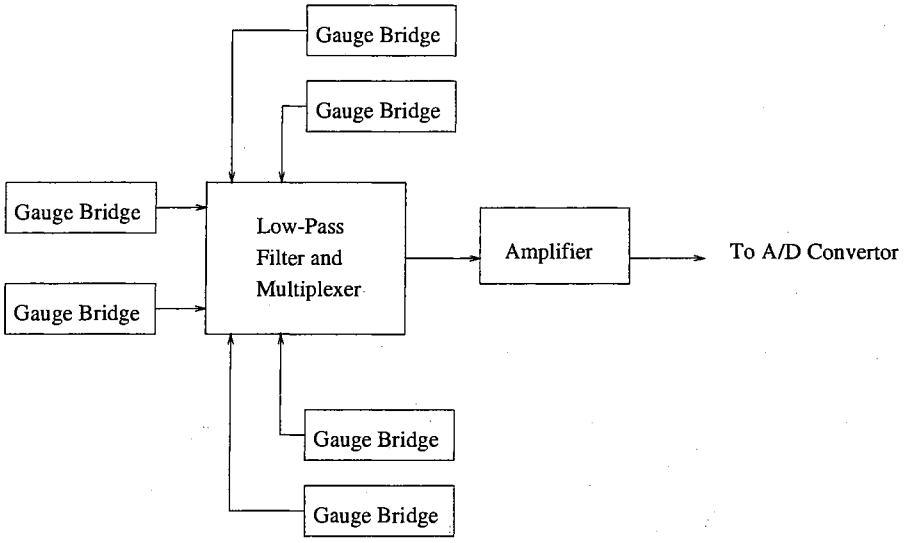

Figure 4.4: Force/torque transducer

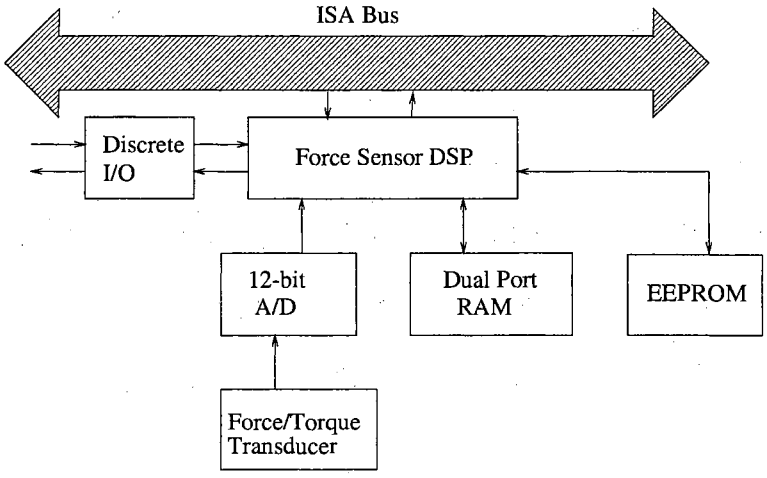

Figure 4.5: ISA force controller card

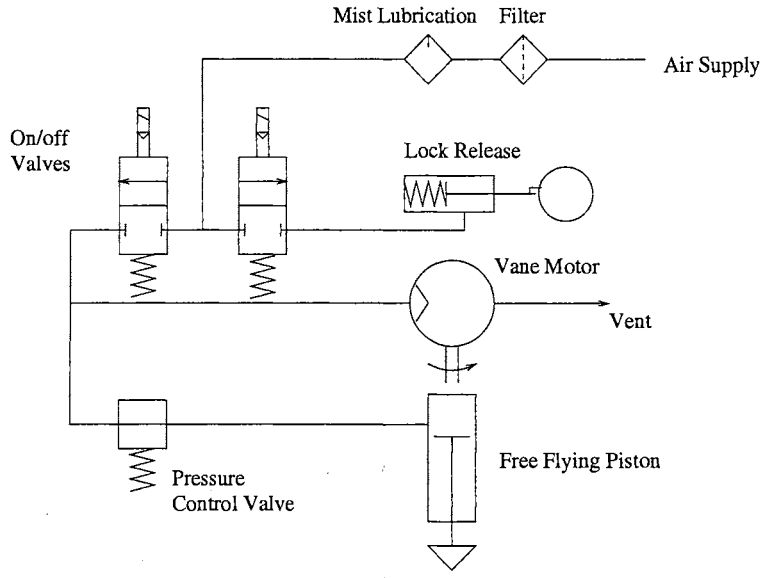

Figure 4.6: Robotic surface finishing tool

4.1.4 Surface Finishing Tool

The robotic surface finishing tool used in the system is a high-speed multi-position robotic deburring tool with integrated axial compliance. The deburring tool consists of an air driven vane motor and a rotary file. Fig. 4.6 gives the schematic of the multi-position deburring tool and its pneumatic system. The idle speed of the air driven vane motor can reach up to 30,000 RPM. Depending on the load, the operating speed ranges from 18,000 RPM to 25,000 RPM. When the motor works at 25,000 RPM, its power is 250 W. The axial force due to the axial pressure can be regulated from 1 N to 25 N.

One of the important features of this tool is its ability to provide axial motion. In the robotic applications, a deviation between the burr edge and the desirable path exists. This tool can compensate for this deviation because the rotary file is free to move 8 mm in the axial direction, which is pneumatically controlled. By means of a special lock-mechanism, this multi-position deburring tool can be fixed in 24 different position related to the bracket. This feature equips the tool with an ability to reach burr edges with difficult access.

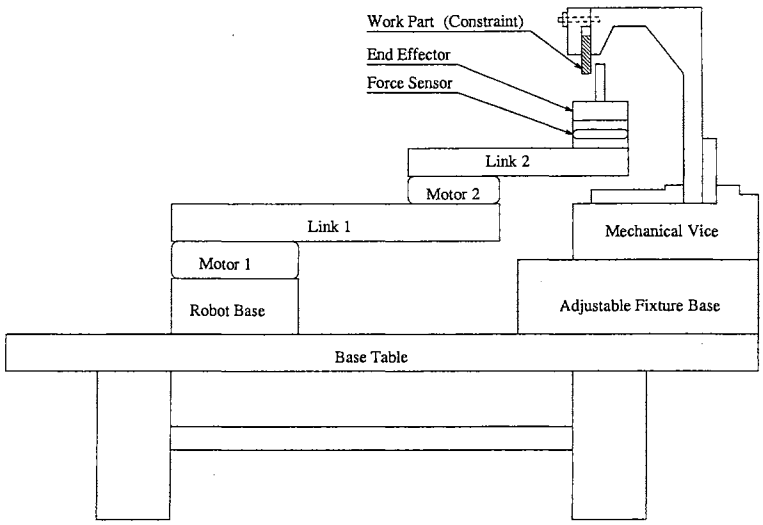

Figure 4.7: Upward fixture

4.1.5 Fixture

Two types of fixture designs were chosen for the work part (or constraint surface). The two-link robot manipulator and fixture are fixed on the base table. The upward fixture consists of a mechanical vice and upward fixture frame as shown in Fig. 4.7. In the downward fixture, the work part is fixed directly to a mechanical vice as shown in Fig. 4.8. The rigidity of the constraint surface in the downward fixture only depends on the rigidity of the surface material, whereas the upward fixture frame adds compliance to the constraint surface. Experiments were conducted on both types of fixtures to compare the affects of compliance on the performance of the proposed control algorithm.

4.2 Software

The software of the robotic surface finishing system can be divided into two parts: off-line and real-time control parts as shown in Fig. 4.9. Desired trajectories are generated by an off-line C module using location and dimension specifications of the trajectories. For desired trajectories involving constrained motion, there is a constraint locating and calculating module which is used to obtain constraint location and dimension information from

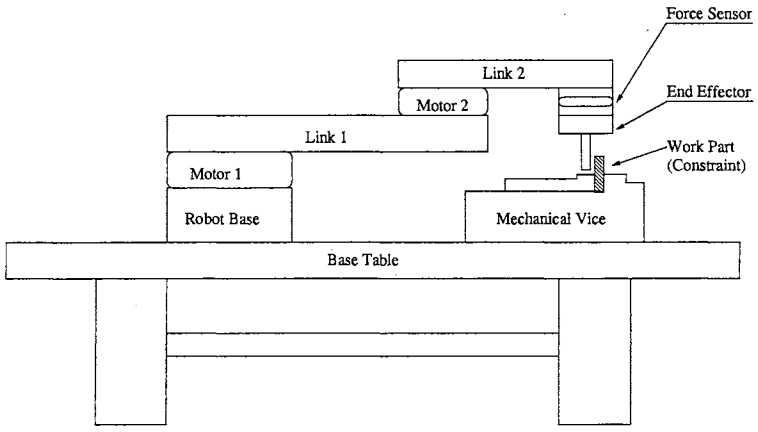

Figure 4.8: Downward fixture

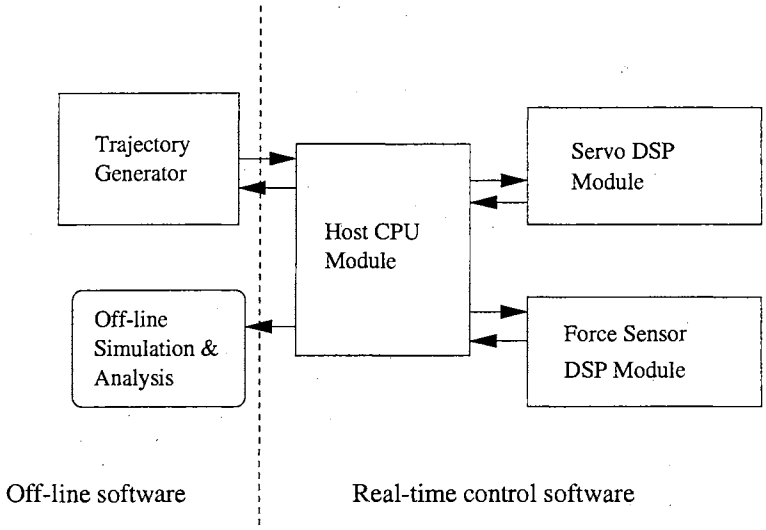

Figure 4.9: Software of the open architecture experimental platform

the data collected from test trials. Besides generating the desired trajectories with smooth position, velocity, and acceleration, the trajectory generator conducts robot reachable space check, singular point check, and output torque limit check. MATLAB is used as off-line simulator to test the proposed control schemes before implementation.

The real-time control part consists of three modules shown in Fig. 4.9, namely host CPU module, servo DSP module, and force sensor DSP module. Corresponding to the multi-processor structure, three software modules run on the three processors. Real-time data exchange among these three modules takes place through dual port RAMs on the DSP

boards.

4.2.1 Host CPU Module

As shown in Fig. 4.10, the software module running on host computer CPU can be broken into several functional modules, each of which performs a specific task such as interpolation, inverse kinematics or dual port RAM operations.

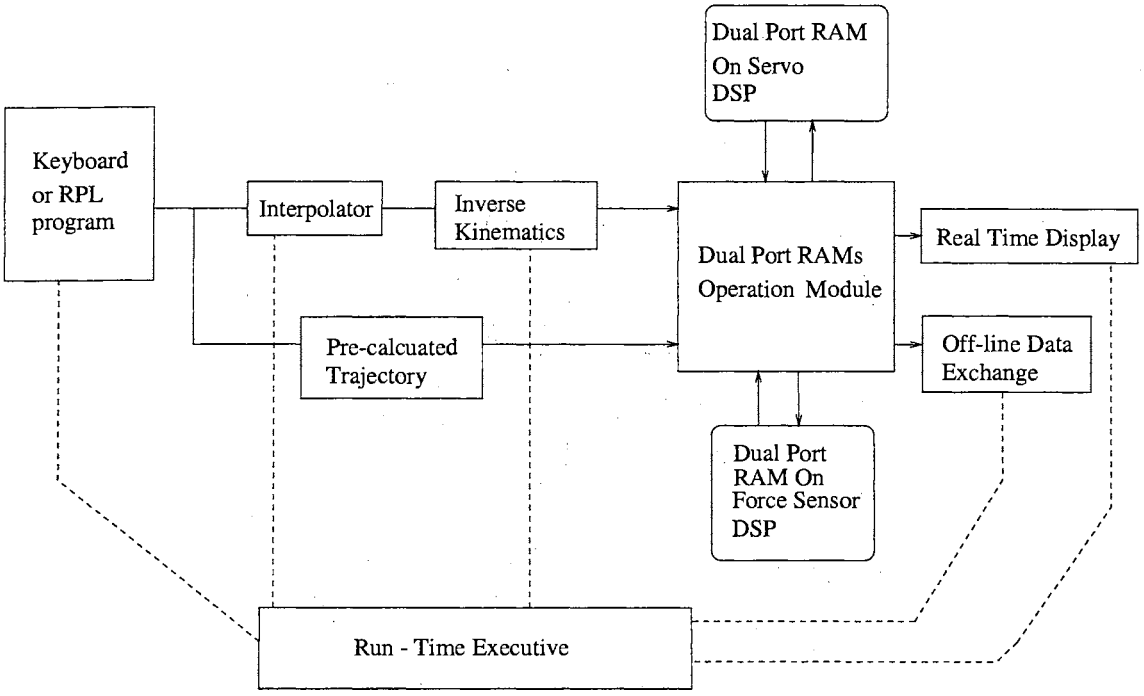

Figure 4.10: Real-time modules on host CPU

As shown Fig. 4.10, the desired trajectories can be obtained by defining path end points and/or downloading trajectory generated off-line. In the first method, either a RPL (Robot Program Language) program or the keyboard is used to specify the path end points. Then the interpolator and inverse kinematics modules generate the desired trajectory at every sampling period. Using this approach, it is difficult to specify complicated trajectories. Furthermore, the choice of sampling period is limited due to inverse kinematic computations in real-time. In the second method, the desired trajectory data is downloaded from pre-calculated trajectory files. Complicated trajectory with special characteristics such as

smooth velocity, acceleration, even jerk may be obtained via this approach. In the experiments, the control algorithms being tested require smooth velocity and acceleration. RPL programs or keyboard inputs are used to move the robot from its home position to the first point of the desired trajectory on which proposed control algorithms are applied. The desired trajectory is calculated off-line and is downloaded directly to the dual port RAM on the servo DSP board.

The data saved in the dual port RAM on the servo DSP can be read by a display module and uploaded to the host computer for further analysis. The dual port RAM operation module in Fig. 4.10 manages the data flow among the host CPU, force sensor DSP, and servo DSP, initiation and data collection commands are sent to the force sensor DSP board; real-time force data is collected from the sensor DSP dual port RAM; force data and desired trajectory data are sent to the servo DSP. The run-time executive block acts as a user interface and does the following:

- coordinates data flow among software modules,
- performs scheduling and synchronizes the servo DSP, sensor DSP, and the host CPU,
- handles real-time communication of data among the three processors,
- performs safety checks and "housekeeping functions", and
- performs the homing sequence.

4.2.2 Servo DSP Module

The servo control module running on the servo DSP generates the appropriate voltage output, between 10 V and -10 V, to the driver units each sample period. The control algorithm being tested is sent to the control module as shown in Fig. 4.11. The control algorithms are written in the C language and are complied by TMS320C30C compiler. The main function

of the control module is to implement the control algorithm using position and velocity feedback signals.

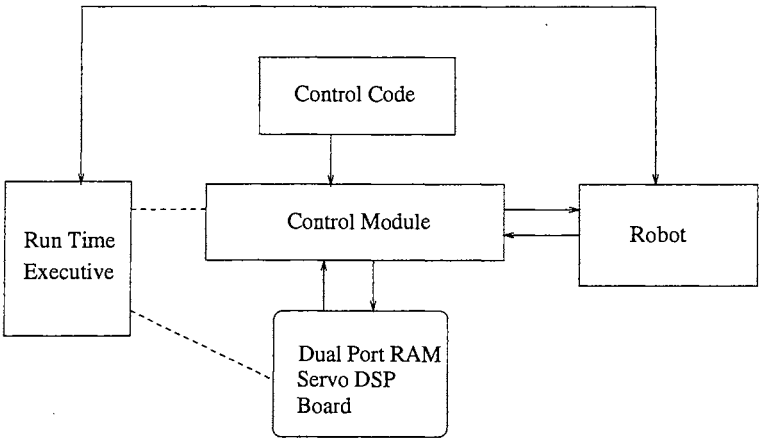

Figure 4.11: Real-time modules on servo DSP board

The run time executive module handles the communication between the host processor and servo DSP and delivers the output signals to the motor driver units via two DS2 I/O interface cards. This module also monitors the status of external switches such as the position limit switches and the emergency stop switch and runs the control module once every sampling period. The control module can read data or write data to the dual port RAM on the servo DSP.

4.2.3 Force Sensor DSP Module

The force sensor DSP module collects raw data from the force transducer, processes the raw data, saves the processed data in the dual port RAM, and sends the force data to host processor via the host computer ISA bus. In Fig. 4.12, the run time executive module controls the ISA bus interface and the dual port RAM on the board. The signal processing block shown in Fig. 4.12 conducts saturation check, bias, temperature compensation, transducer condition check and other calculations on strain gauge data for every sampling period. The flow chart of the signal processing block is shown in Fig. 4.13.

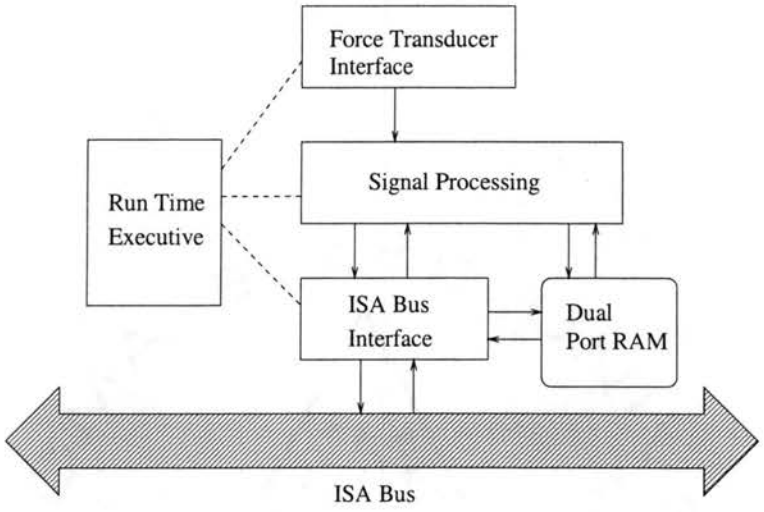

Figure 4.12: Real-time modules on force sensor DSP

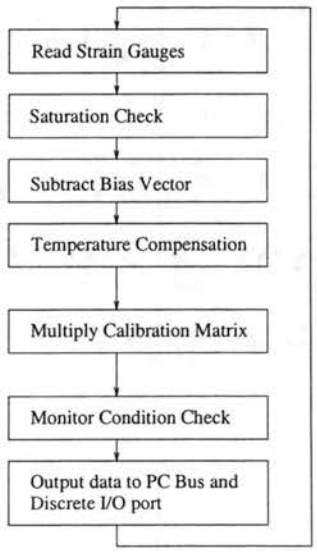

Figure 4.13: Flow chart of signal processing module

4.2.4 Data Exchange

The software modules running on the host computer, servo DSP, and force sensor DSP exchange data via dual port RAM's as shown in Fig. 4.14.

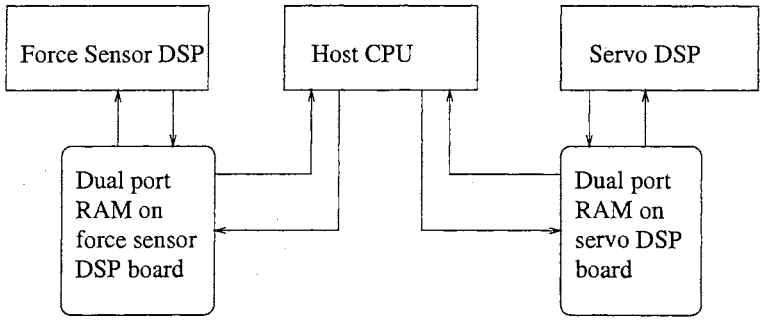

Figure 4.14: Data exchange among modules

Dual port RAM on the force sensor DSP board connects force sensor module and the module on host computer. The RPL program running on the Host CPU sends commands to the dual port RAM on the force sensor DSP board to initialize the transducer and its controller. The commands can set sampling frequency and low pass filter parameters for force sensor, initialize the temperature compensation matrix and frame transfer matrices. After the transducer has been initialized, the software module on the force sensor DSP board puts the force/torque data on to the dual port RAM at every sampling period. Whenever the force/torque data is refreshed, the dual port RAM operation module on the host computer reads the data. After data processing such as filtering, the host CPU delivers the force data to servo DSP.

The host CPU exchanges data with the servo DSP via the dual port RAM on the servo DSP board. Whenever the host CPU reads new force data from the sensor, the RPL program modifies the data field in the servo DSP dual port RAM corresponding to force data. The control program running on the servo DSP reads this data once per control period. At the same time, the run time executive module in Fig. 4.11 modifies the units corresponding to current positions, velocities, acceleration, and some user defined variables. These variables

are read by the host computer for real-time display. The host computer and the servo DSP exchange data via dual port RAM after and before servo control, also. The pre-calculated trajectory data is downloaded on to the DSP dual port RAM before real-time control. The experiment data is uploaded from the servo DSP to the host computer after real-time control and saved in data files for further analysis.

4.3 Implementation of Control Scheme

In this section implementation of the switching control scheme proposed in the Chapter 3 will be discussed. The noisy force signals from the force sensor cannot be directly used in the control algorithm. Control switching and force signal filtering are essential to stability and performance of the closed-loop control system. The method in which data is acquired and processed is critical to the overall real-time system performance. For the multi-processor robot system, the following five integration modes are considered and their performance in terms of sampling delay and control algorithm switching delay is discussed.

(1) **Mode 1:** Synchronous operation mode (Fig. 4.15).

(2) **Mode 2:** Asynchronous operation mode with multi-frequency sampling (Fig. 4.16).

(3) **Mode 3:** Asynchronous operation mode with multi-frequency sampling using fixed length average filter on the sensor DSP board (Fig. 4.17).

(4) **Mode 4:** Asynchronous multi-sampling frequency mode using sliding fixed length average filter on the host processor (Fig. 4.18).

(5) **Mode 5:** Asynchronous multi-sampling frequency mode using variable length sliding average filter on the host processor (Fig. 4.19).

In Fig. 4.15, the force sensor DSP and the servo DSP use the same sampling period ($t_s = T_s$), where t_s is the force sensor sampling period and T_s is the control sampling period of the servo DSP, and the operations are synchronized. The signal processing module

runs on an interrupt mode and the interrupt is triggered once per sample period. The data exchange module on the host computer works in polling mode. Whenever the force data is ready, the host computer fetches the data and delivers it to the servo DSP. Let Δt be the time required by the servo DSP to run the control algorithm once and to read the force signal at the begin of its program code. From Fig. 4.15, the force sampling delay, which is defined as the time delay between sampling of the force signal and the control output being sent to the motor drivers, is $T_s + \Delta t$.

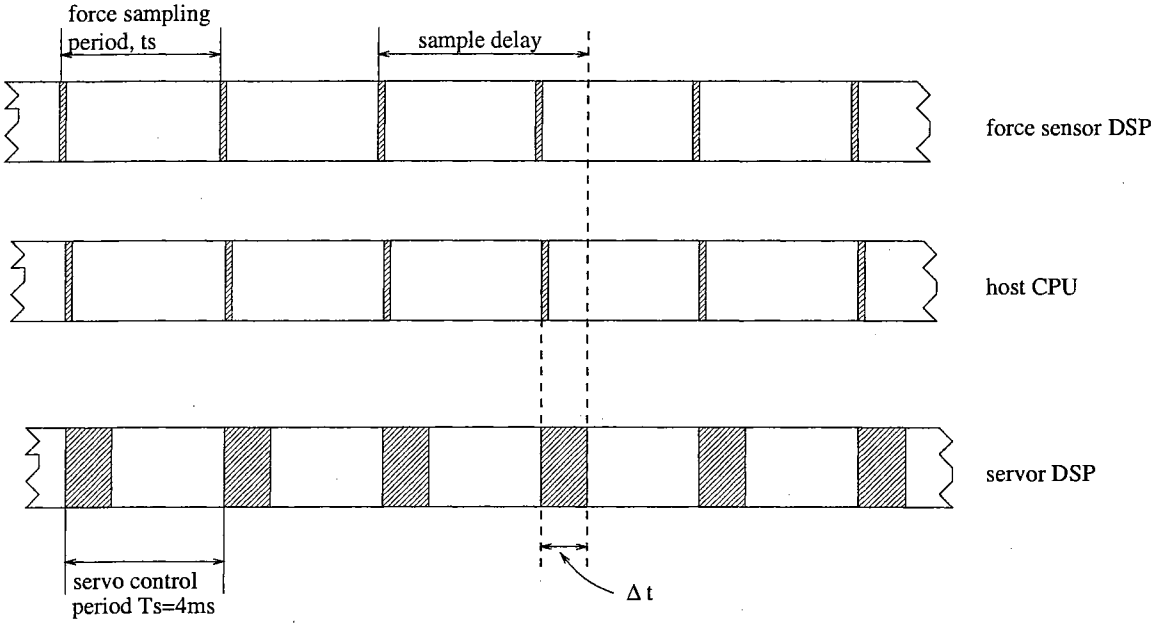

Figure 4.15: Mode 1: Synchronous operation

It is assumed that the control sample period cannot be changed due to servo processor computation capability and complexity of the control algorithm. In mode 2, asynchronous operation mode is implemented, where the fastest possible sampling rate available for the force sensor is chosen to minimize the force sampling delay. Because most of the computation load for computing the control input is on servo DSP, increasing the force sensor sampling period does not affect the implementation of the control algorithm. The force sensor that is used in this experimental platform can provide force data at a rate up to 4 kHz. A control sampling frequency of 250 Hz is used. Fig. 4.16 shows asynchronous

operation, where the force sensor sampling frequency is considerably higher than the servo sampling frequency. In this case, the maximum sampling delay is $t_s + \Delta t$, where $t_s \ll T_s$. Hence, the sampling delay is minimized without changing the servo sampling frequency.

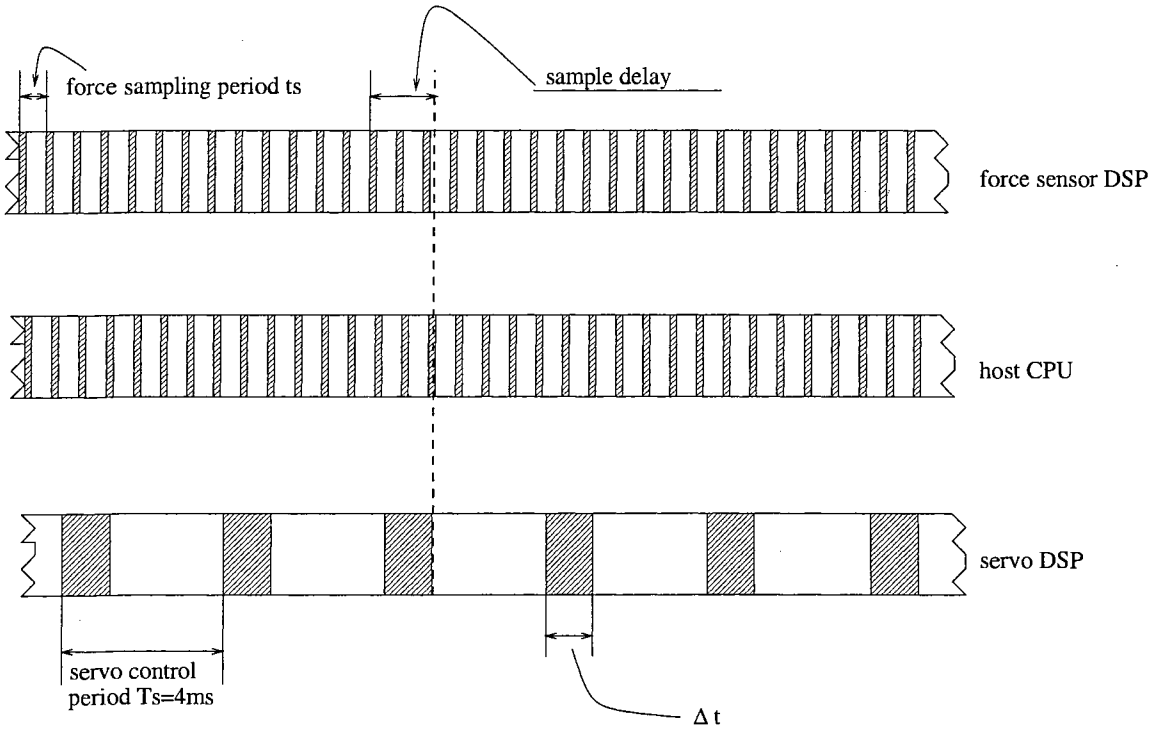

Figure 4.16: Mode 2: Asynchronous operation

To alleviate the problem caused by force signal noise, low pass filtering of the force signal is considered in mode 3. An eight sample running average filter of the force signal is implemented on the force sensor DSP. Instead of sending each force data signal to the host processor, the force sensor DSP collects eight samples, calculates the average, then sends it to the host processor, as shown in Fig. 4.17. In mode 3, a control sampling frequency of 250 Hz and a force sampling frequency of 2 kHz are used. The maximum time and minimum time delay due to force sampling are $T_s + \Delta t + \Delta T_f$ and $\Delta t + \Delta T_f$, respectively. ΔT_f represents the time delay due to filtering of the force data.

In Mode 4, a fixed length sliding average filter is implemented on the host processor as shown in Fig. 4.18. The host processor working in the polling mode fetches the force signal at a rate of force sampling frequency, $1/t_s$. This frequency can be higher than the

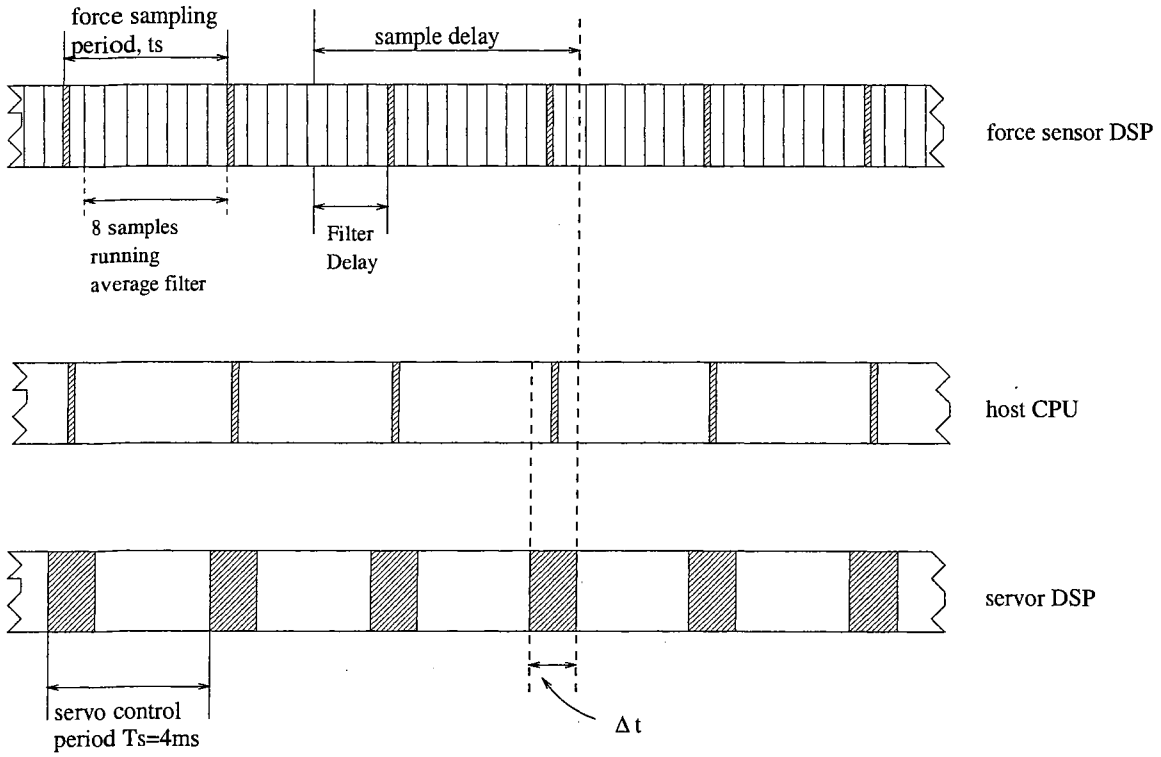

Figure 4.17: Mode 3: Asynchronous operation, fixed length average filter on force sensor DSP

system control frequency $1/T_s$. Using the previous $n-1$ points of the force signal, the host processor implements a n sample length sliding filter at every data sampling period. Fig. 4.18 indicates that the maximum force sample delay is $t_s + \Delta t + \Delta T_f$, and the minimum sample delay is $\Delta t + \Delta T_f$.

Recall that the controller for a complete task is a switching type of control algorithm. Switching of the control algorithm from unconstrained motion to transition motion is triggered based on a threshold force signal level. If the end-effector makes contact with the surface just before the servo DSP reads the force data for the next control period and the force signal after filtering is still less than the threshold value, switching to the transition control scheme is delayed by one control period, T_s. This time delay can cause a large impact force during transition phase.

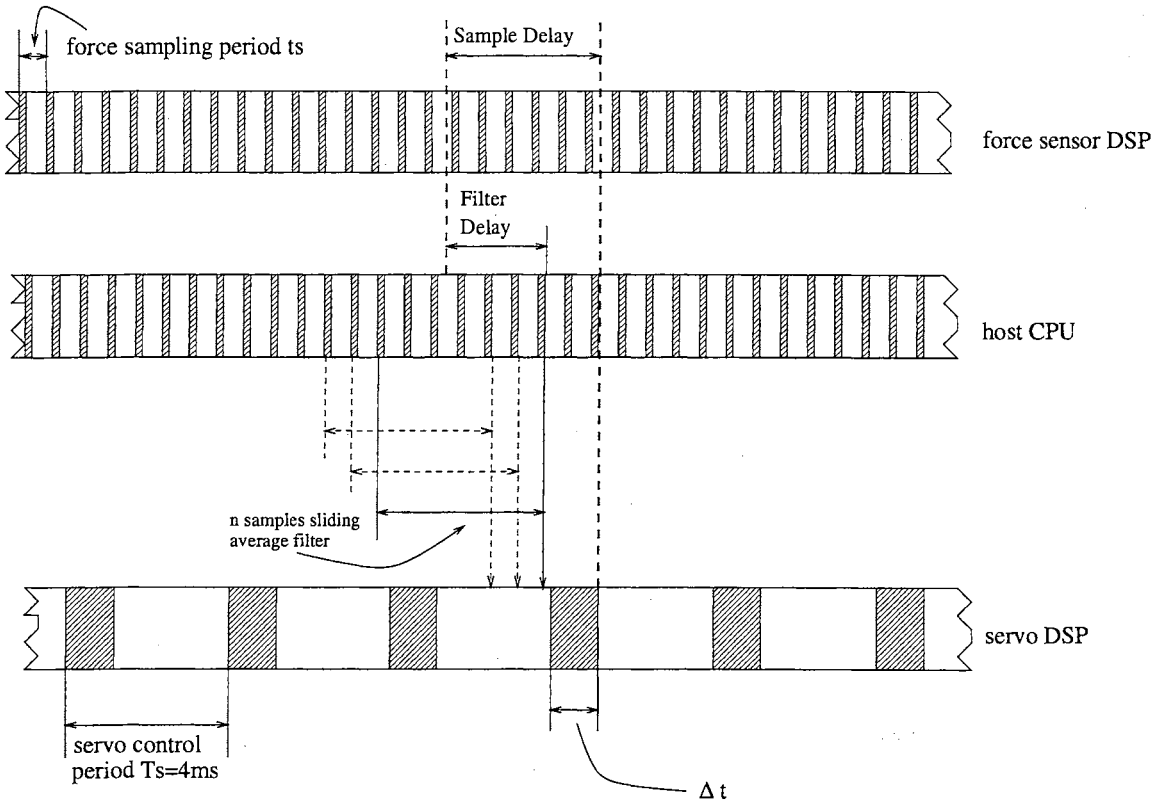

Figure 4.18: Mode 4: Asynchronous operation, fixed length sliding average filter on host CPU

To circumvent this problem, flexible length sliding filter is proposed in Fig. 4.19. The solid black blocks indicate the force signal greater than the threshold. Integration mode 5 is similar to mode 4 except in the case when the host processor reads a force signal greater than the threshold value, the filter length is modified to one. Further, the length of the filter is increased as the new force data is available until it equals the fixed filter length. Using mode 5, the controller can switch to transition control algorithm at the earliest control period after the end-effector makes contact with the surface.

For mode 5, the maximum and minimum force sampling delay is $t_s + \Delta t + \Delta T_f$ and $\Delta t + \Delta T_f$, respectively. Notice that the range of the force sampling delay is same as in mode 4. However, the control algorithm switching delay lies in the range $(\Delta T_f + \Delta t, \Delta T_f + T_s + \Delta t)$ for mode 4, and $(\Delta t, T_s + \Delta t)$ for mode 5. The control switching delay is critical to system performance in the presence of constraint uncertainty. Table 4.3 gives a summary

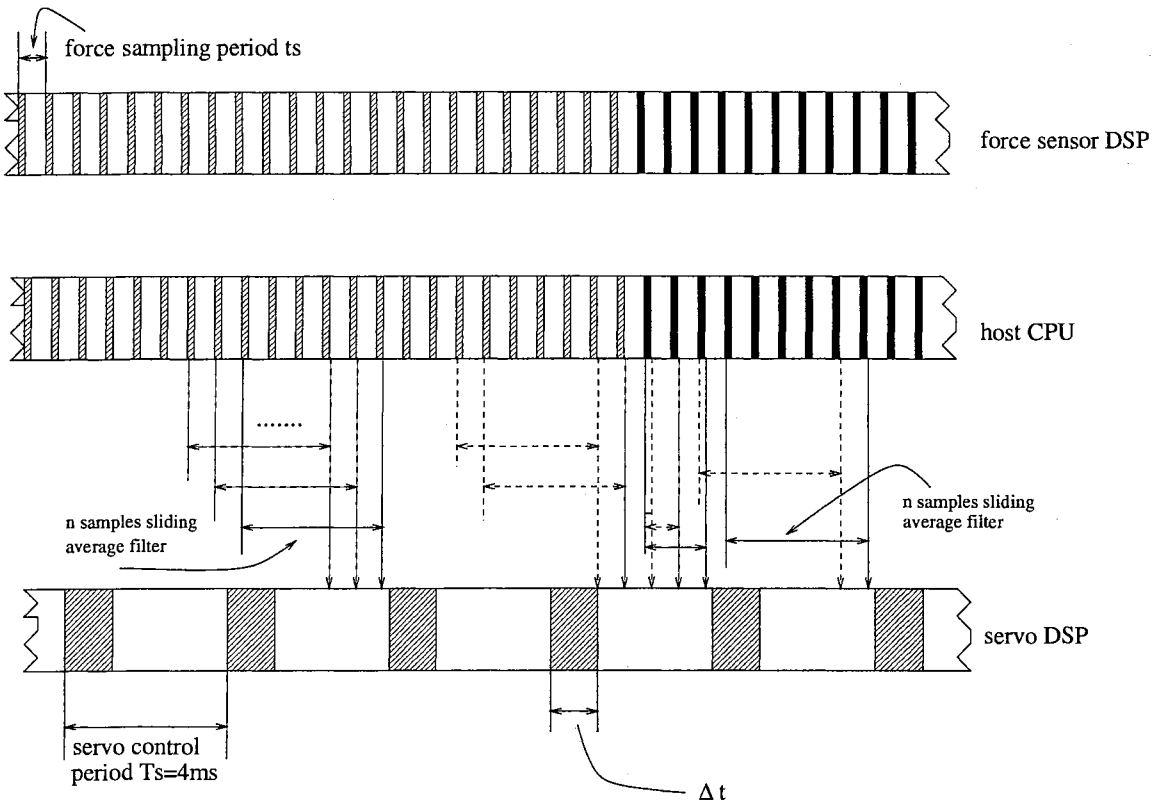

Figure 4.19: Mode 5: Asynchronous operation, variable length sliding average filter on host CPU

of force sampling delay and switching delay range of the five controller implementation modes. The following observations can be made:

- compared to mode 2, the force sample delay and algorithm switch delay can be reduced by using high force sampling frequency;
- the filter delay introduced in mode 3 increases sampling delay and switching delay;
- in mode 4, the filter delay is reduced via multi-sampling rate;
- mode 5 is able to reduce the switching delay further.

In this chapter, the hardware and software components of the open architecture robotic experimental platform are described. An efficient mechatronic design has been considered in developing a robotic surface finishing system. Different controller implementation

Mode	Sample Delay	Switching Delay
1	$T_s + \Delta t$	$T_s + \Delta t$
2	$(\Delta t, t_s + \Delta t)$	$t_s + \Delta t$
3	$(\Delta t + \Delta T_f, T_s + \Delta t + \Delta T_f)$	$(\Delta T_f + \Delta t, 2T_s + \Delta T_f + \Delta t)$
4	$(\Delta t + \Delta T_f, t_s + \Delta t + \Delta T_f)$	$(\Delta T_f + \Delta t, \Delta T_f + T_s + \Delta t)$
5	$(\Delta t + \Delta T_f, t_s + \Delta t + \Delta T_f)$	$(\Delta t, T_s + \Delta t)$

Table 4.3: Sample and switching delay

modes are compared emphasizing data communication between processors and data filtering.

Chapter 5

Experiments

In the previous chapter, mechatronic design of the open architecture experimental platform is described. In this chapter, the experiments conducted on the experimental platform for complete robot tasks are discussed. The effectiveness of the proposed control design and the efficiency of system architecture design are analyzed and compared based on experimental results. Manipulator dynamic model uncertainties, force control during constrained motion, impact force and transition control, and effects of different controller implementation modes are investigated experimentally.

First, a model-based adaptive motion control algorithm for robot motion control is investigated. In order to show its performance in presence of manipulator uncertainties, an experimental case study has been developed. Tracking performance of adaptive control is compared to that of computed torque control with and without dynamic model uncertainties. Second, experiments are conducted to validate the effectiveness of the adaptive position/force control scheme proposed in section 3.3. Besides tangential tracking and normal force regulation performance, adaptive estimation of grinding coefficient under different contact conditions is investigated. Then, impact phenomena which is common during transition from unconstrained motion to constrained motion is investigated experimentally. An experimental understanding of impact phenomena can facilitate better transition control design. The performance of the transition control design proposed in section 3.2 is com-

pared with ordinary discontinuous control strategies without transition control. In order to show the overall performance of the control scheme proposed for the robot performing a complete task, surface finishing application is considered. Two different surface finishing modes were compared.

In this chapter, control implementation issues are also investigated with experimental comparison. In order to achieve optimal system performance, the requirements of proposed discontinuous control scheme for robot complete tasks was considered. Different controller implementation modes were implemented and compared in terms of signal filtering, filtering delay, and control switching delay.

The rest of this chapter is organized as follows: Section 5.1 gives experimental results of unconstrained motion control. Two desired trajectories are used in unconstrained motion control experiments. Emphasis is given to the system performance with manipulator model uncertainties. In section 5.2, experiment results of complete robot tasks involving constrained motion are shown. First, experimental evaluation of the proposed adaptive position/force control algorithm is given for the case of known grinding coefficient. Then, experimental results are presented for the case where the grinding coefficient is not well known. An adaptation law is used to estimate the grinding coefficient on-line. Section 5.3 starts with an experimental study of free impact of robot manipulator. An experimental comparison between the proposed transition control and ordinary direct switching strategy is given in this section considering constraint uncertainty. The results of surface finishing experiments are discussed in section 5.4. In section 5.5, the control implementation modes discussed in previous chapter are compared and analyzed experimentally.

5.1 Adaptive Control for Unconstrained Motion

A model based adaptive control algorithm discussed in section 3.1 was implemented on the two link experimental platform. The main emphasis of these experiments is to investigate

the adaptation law for manipulator inertial parameters.

$$\begin{aligned} \tau &= \widehat{M}(q)\ddot{q}_r + \widehat{C}(q,\dot{q})\dot{q}_r + \widehat{g}(q) - F_v e_v \\ \widehat{\beta}(t) &= \beta_0 - \int_0^t \Gamma^{-T} Y^{\top}(q,\dot{q},\dot{q}_r,\ddot{q}_r) e_v dt \end{aligned}$$

To show the control performance under different conditions, experiments have been conducted on two different desired trajectories.

5.1.1 Desired Trajectories

The desired trajectory 1 shown in Fig. 5.1 is such that link 1 and link 2 start at home position, move to $\pi/3$ in 1 second, stop for 1 second, move back to the home position in 1 second again. The robot manipulator workspace (dotted line) is also illustrated in Fig. 5.1. As shown in Fig. 5.2, this trajectory is designed such that the desired joint position, velocity, and acceleration are all smooth and the desired velocity and acceleration are zero at the first and last points of the path. Notice that the desired trajectories of link 1 and link 2 are exactly the same in this case.

The desired trajectory 2 shown in Fig. 5.3 is to finish a circle trajectory in the Cartesian space in 4 seconds. The desired end-effector velocity increases and decreases smoothly along the circle circumference. Fig. 5.4 shows the desired trajectory in joint space.

5.1.2 Experiment Results

Experimental results of unconstrained motion control are shown in Fig. 5.5 to Fig. 5.10. Figs. 5.5, 5.6 and 5.7, 5.8 show the tracking errors for trajectory 1 and 2, respectively. The trajectories were repeated 8 times for each experiment. For each trajectory, controller performance for the following three cases were compared:

- Case 1: computed torque with manipulator model uncertainties. The controller assumes inertial parameters as 50% of the true values. The followings are the controller

gains used in the experiments:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 1.5 \\ 0.2 \\ 0.2 \end{bmatrix}$$

- Case 2: computed torque without manipulator model uncertainties. The controller gains are:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 3.162 \\ 0.12 \\ 0.18 \end{bmatrix}$$

- Case 3: model-based adaptive control with initial model uncertainties using the following control and estimation gains:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 1.5 \\ 0.2 \\ 0.2 \end{bmatrix} \quad \Gamma^\top = \begin{bmatrix} 2.0 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$$

Experimental results of case 1 and case 2 are presented in the upper and middle plots in the figures. The first two plots in Figs. 5.5- 5.8 indicate that the system performance is sensitive to the uncertainties of the manipulator model. For both trajectories, the tracking errors of link 1 and link 2 can be significantly reduced provided the manipulator dynamics are well known. From the third plot in Figs. 5.5 - 5.8, it can be observed that the adaptive control law can reduce tracking errors while applying the adaptation law (3.2). In the presence of model uncertainties, model-based adaptive control out-performs computed torque control using the same gains and dynamic model. After several cycles, the tracking errors of model-based adaptive control is as good as that of computed torque control using accurate manipulator model.

Experimental results shown in Fig. 5.9 and 5.10 indicate that the estimates of manipulator inertia parameters appear to converge to constant values. However, the converged

values and the convergence speeds are different for two different trajectories. Since the trajectory 2 which is a circle in workspace satisfies persistency of excitation condition [196], the regressor matrix is persistently exciting for this trajectory. Hence, better parameter convergence is observed for trajectory 2.

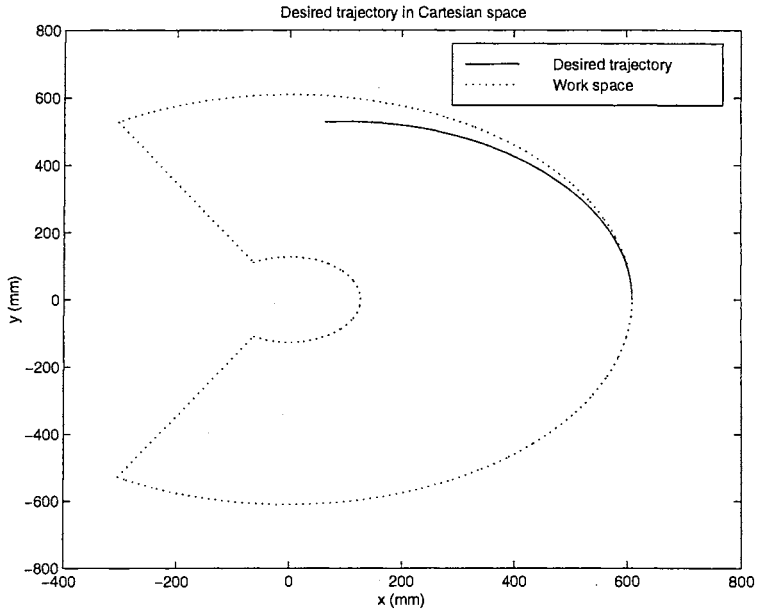

Figure 5.1: Desired trajectory 1 in Cartesian space

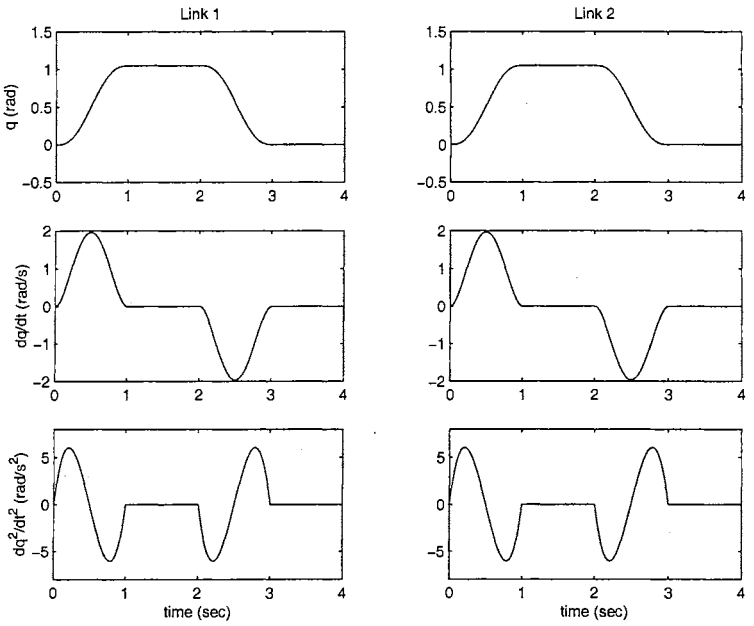

Figure 5.2: Desired trajectory 1 in joint space

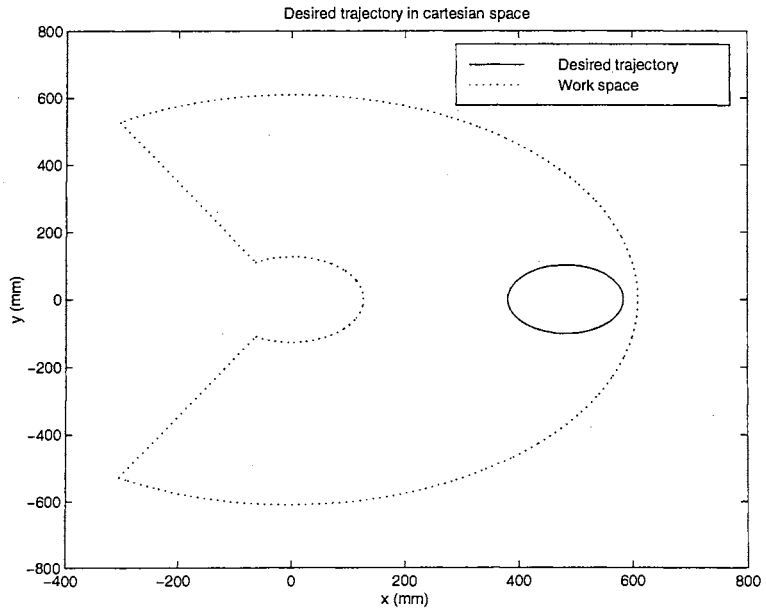

Figure 5.3: Desired trajectory 2 in Cartesian space

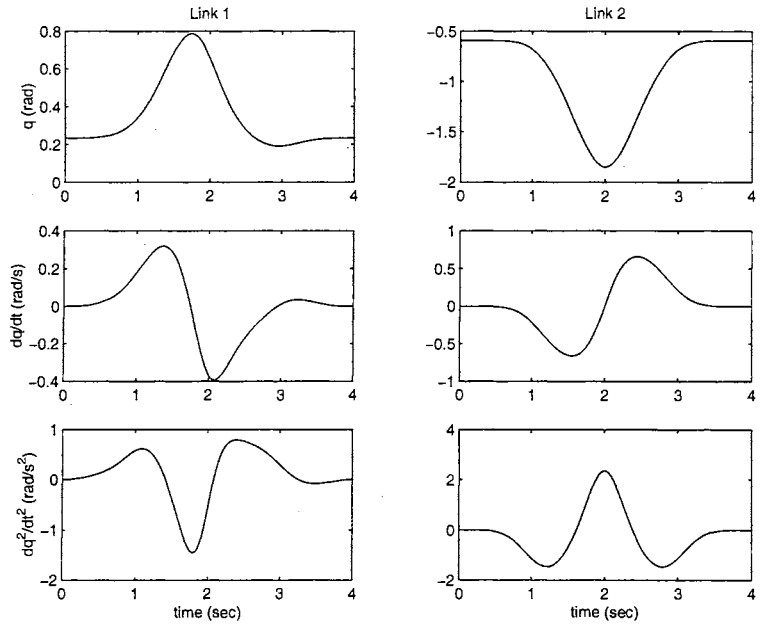

Figure 5.4: Desired trajectory 2 in joint space

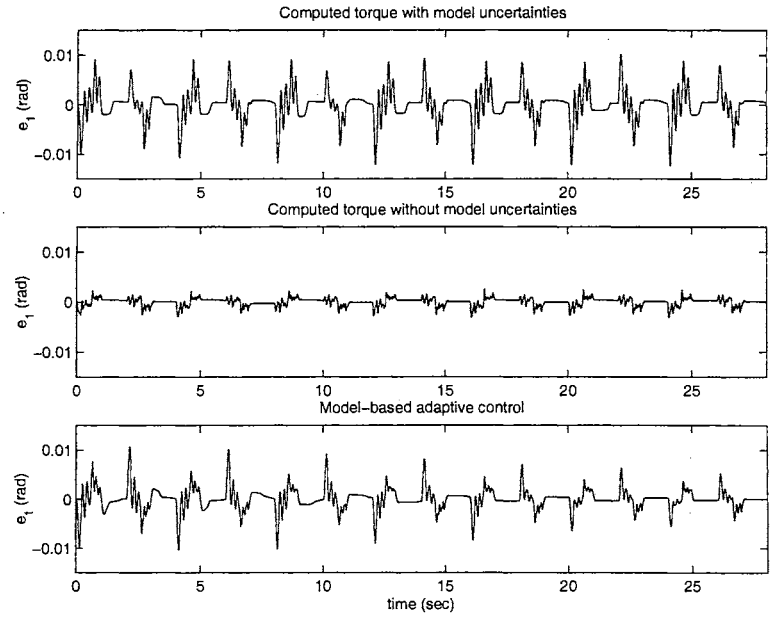

Figure 5.5: Link 1 tracking error of trajectory 1

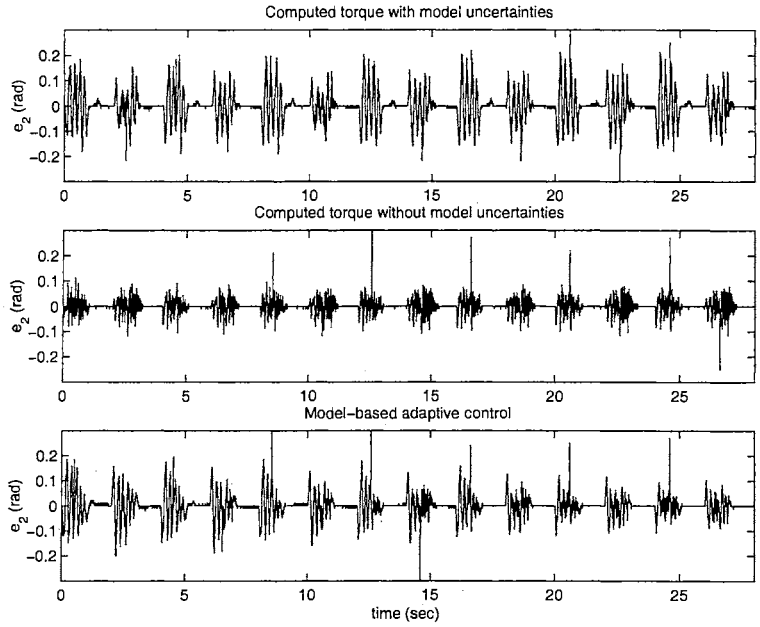

Figure 5.6: Link 2 tracking error of trajectory 1

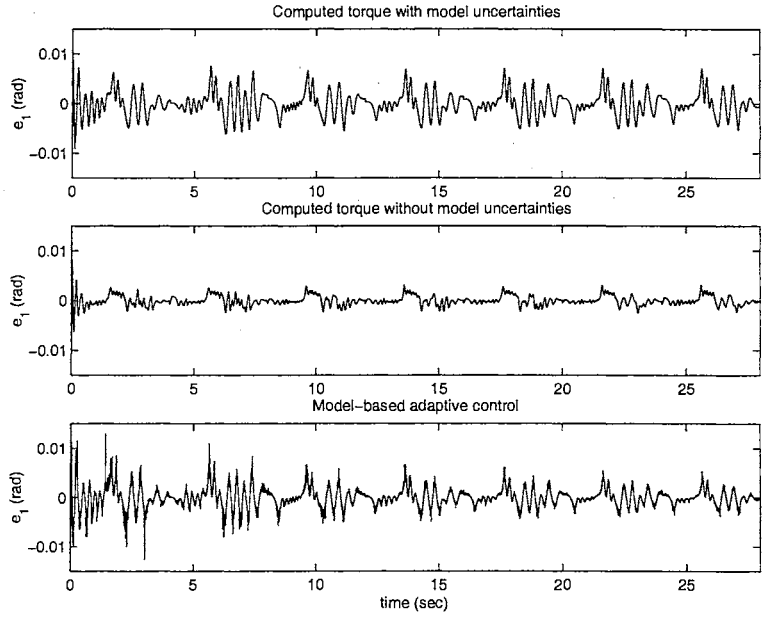

Figure 5.7: Link 1 tracking error of trajectory 2

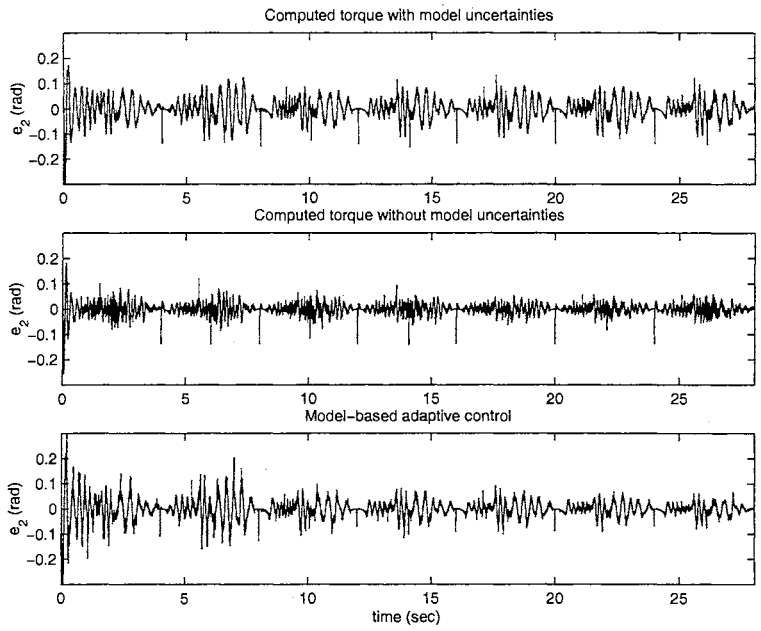

Figure 5.8: Link 2 tracking error of trajectory 2

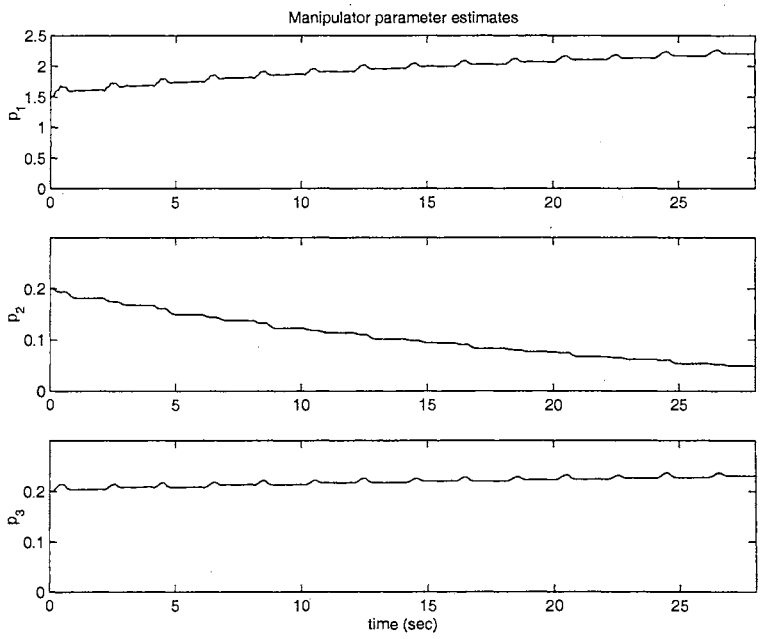

Figure 5.9: Manipulator parameter estimates (trajectory 1)

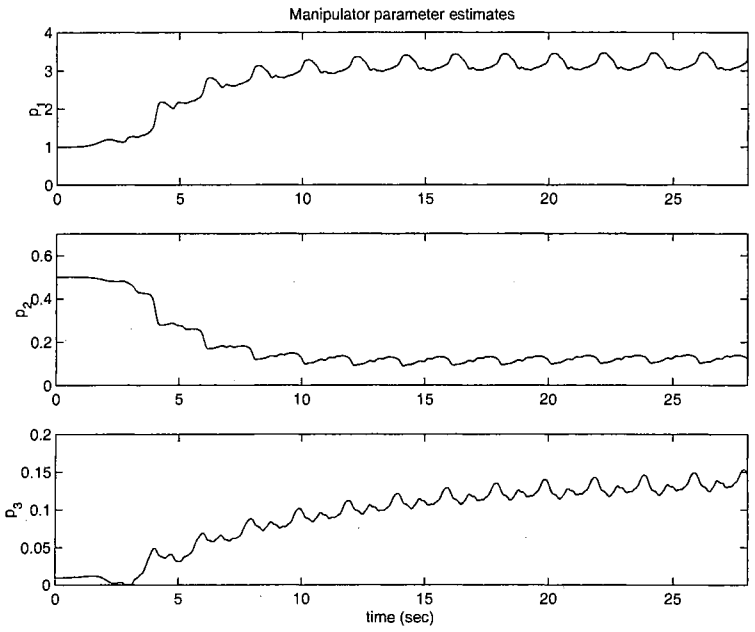

Figure 5.10: Manipulator parameter estimates (trajectory 2)

5.2 Position/Force Control for Constrained Motion

In the previous section, the effectiveness of the control algorithm for motion phase was confirmed by experimental results. Position/force control design is investigated experimentally in this section. Two robot desired trajectories that interact with straight and curve constraint surfaces are considered. In the first group of experiments, it is assumed the dynamics during constrained motion is well known, position/force control algorithm is tested. In the second group, grinding coefficient uncertainty is considered. A gradient adaptation law is used for on-line estimation of the grinding coefficient, ξ. Experiment results of both groups are analyzed.

5.2.1 Desired Trajectories

Two constraint surfaces, a thick straight aluminum sheet and a steel cylinder which are firmly held by a vice, are considered. Fig. 5.11 illustrates the two-link robot (top view) and the straight constraint surface in Cartesian space, where d is the location of the surface on the x axis and α is the angle between the constraint and the y axis. The constraint surface equation in Cartesian space is

$$x + y\tan(\alpha) = d \tag{5.1}$$

The desired motion trajectory is generated using the straight line equation given in (5.1). Assuming that two points on the constraint surface are known, the trajectory generator creates the desired trajectory shown in Fig. 5.13. Robot end-effector follows the points C, D, A and B sequentially. The desired velocity on the segment of $A \rightarrow B$ is set to be a constant v_c. On segment $C \rightarrow D$, robot moves in free space. The motion on this segment consists of an acceleration period followed by a deceleration period. On segment $D \rightarrow A$, appropriate acceleration is chosen to increase the velocity magnitude until it is v_c. After the velocity magnitude reaches the constant level v_c, it is maintained at that level. Hence, at point A where robot switches from unconstrained motion to constrained motion, there is no jump in the desired velocity and the desired velocity normal to the surface is zero. Segment $B \rightarrow C$ is symmetric to segment $A \rightarrow D$. Fig. 5.14 shows the desired trajectory position, velocity, and acceleration in joint space.

For the second case, a steel cylinder firmly held by a vice, shown in Fig. 5.12, is the constraint surface. The constraint surface equation is

$$(x - x_o)^2 + (y - y_o)^2 = R^2 \tag{5.2}$$

where (x_o, y_o) and R are the coordinates of cylinder center and radius, respectively.

The desired trajectory was generated such that the robot moves from a point in free space approaching to the constraint surface, makes contact with the surface with zero normal velocity, moves along the surface with constant line speed, then leaves the surface and

returns to the original point. Fig. 5.15 shows the desired trajectory position, velocity, and acceleration in joint space. Notice that this trajectory is in C^2, i.e., two times continuously differentiable.

Fig. 5.16 shows the two desired trajectories where robot interacts with straight and cylindrical constraint surfaces. In Fig. 5.16, the dotted lines represent the robot desired path of unconstrained motion and the solid lines represent constrained motion. The $\star$ in Fig. 5.16 represent the transition phases where robot switches from unconstrained motion to constrained motion or vice versa. In both cases, the desired complete task of the robot is to move towards the constraint surface, make contact with the surface, follow the surface while maintaining a desired normal contact force, and leave the surface to return to the starting point.

Both the desired trajectories shown in Fig. 5.16 are designed in such a way that without any uncertainty in the location of the constraint, the robot lands on the surfaces smoothly, i.e., there is no normal velocity at contact. The desired velocity of the robot end-effector in Cartesian coordinates during constrained motion, i.e., along solid line, is chosen to be a constant. Also, the desired trajectories are such that the entire task is completed in 12 seconds for straight constraint and 10 seconds for cylindrical constraint. Control sampling period of 4 milli-seconds and force sampling period of 0.5 milli-seconds are used in all experiments.

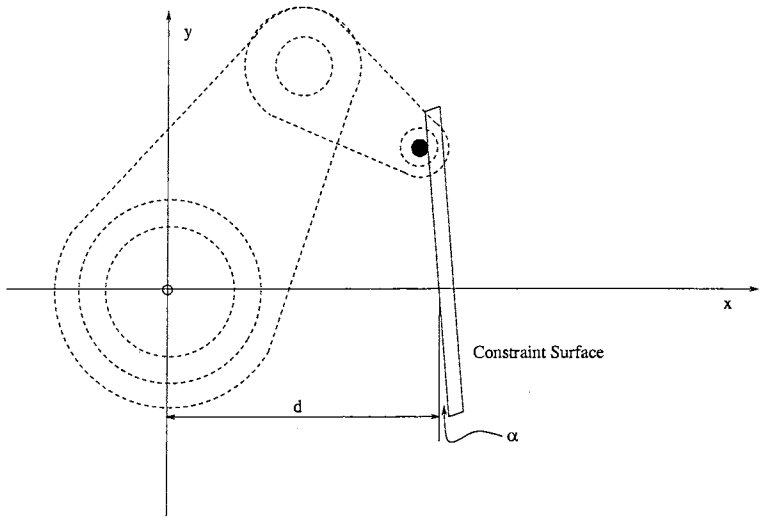

Figure 5.11: Robot and straight constraint (top view)

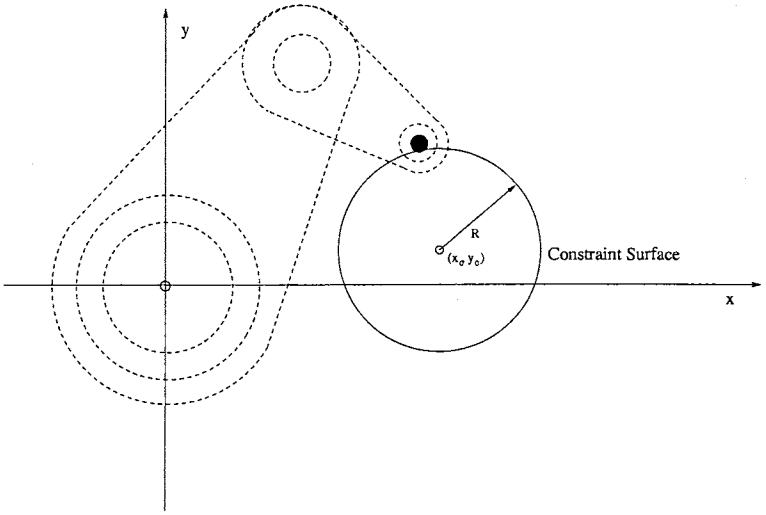

Figure 5.12: Robot and cylindrical constraint (top view)

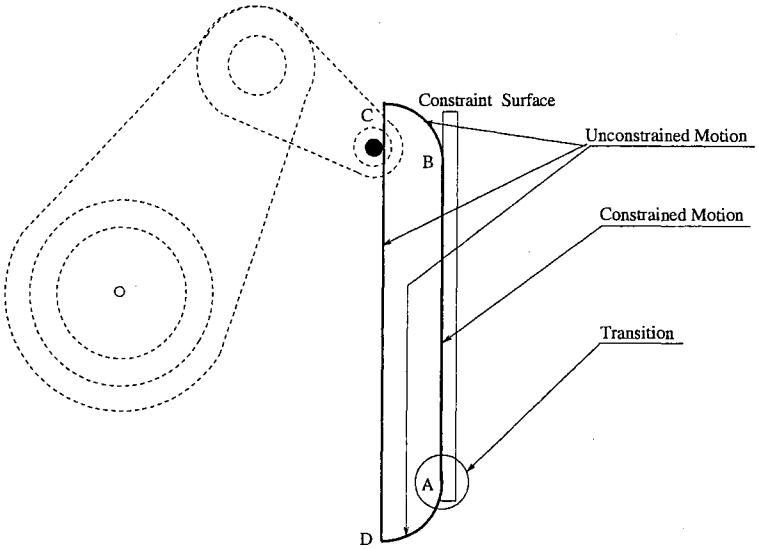

Figure 5.13: Desired trajectory of straight constraint

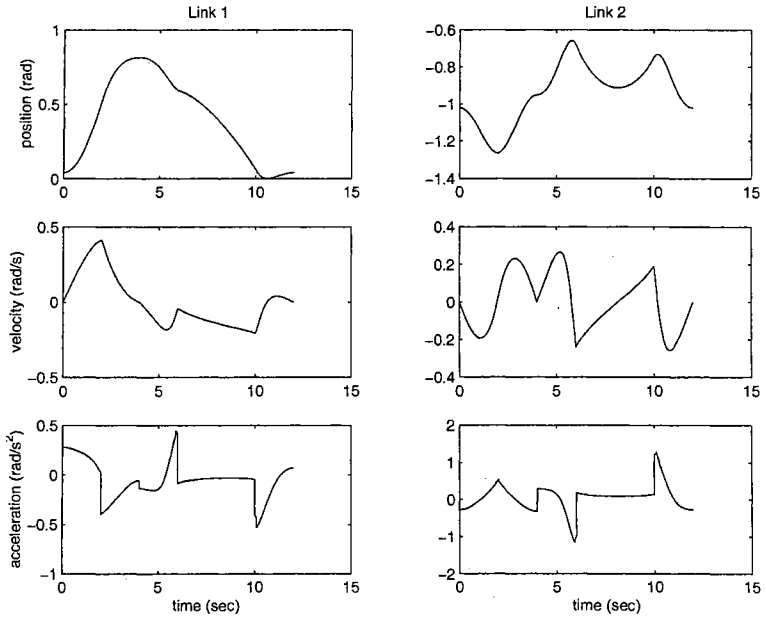

Figure 5.14: Desired trajectory in joint space (straight constraint)

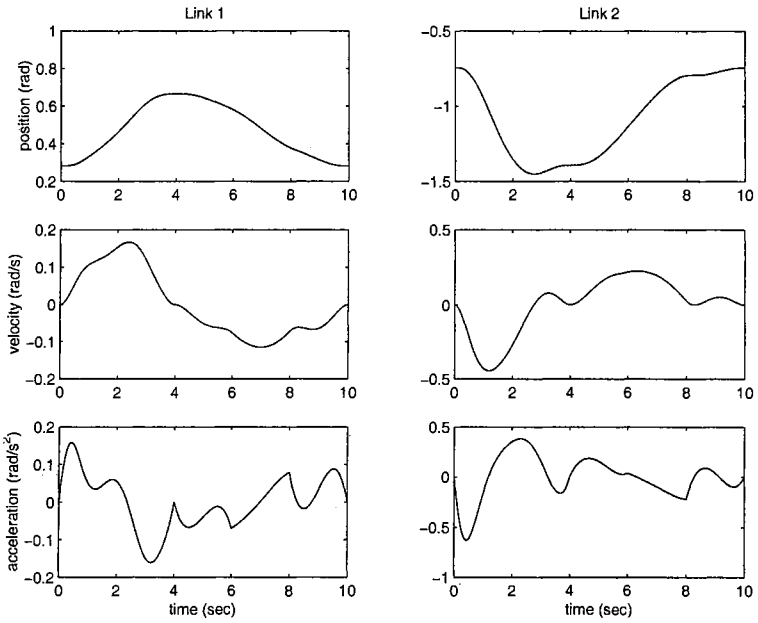

Figure 5.15: Desired trajectory in joint space (cylindrical constraint)

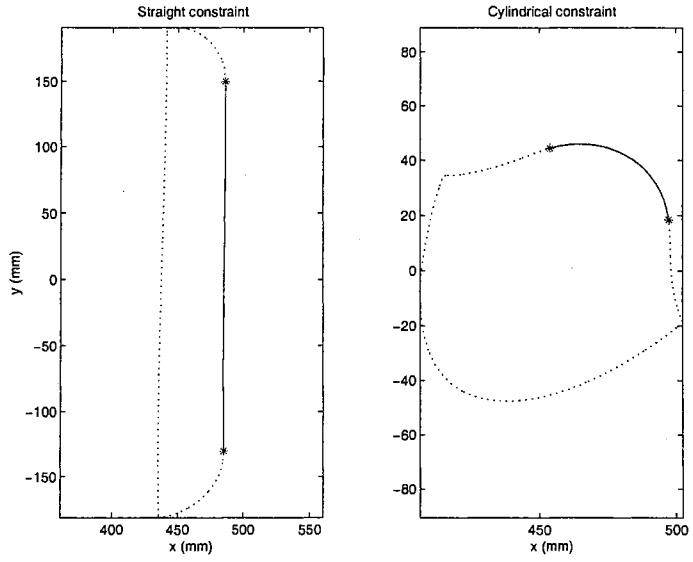

Figure 5.16: Desired trajectories in Cartesian space

5.2.2 Experimental Results

The emphasis of these experiments is on validating the effectiveness of the adaptive position/force control laws (3.53) and (3.54) which were proposed in section 3.3 for constrained motion control, i.e., position tracking tangential to the surface and force regulation normal to the surface. The unconstrained motion control laws (3.1) and (3.2) that have been validated in section 5.1 were employed during unconstrained motion phase.

During unconstrained motion phase, it is expected that the robot not only tracks the desired trajectory but also picks up optimal values of manipulator parameters as it did in the unconstrained motion control experiments shown in section 5.1. Initially, the parameters are chosen to be $p_1 = 1.0$, $p_2 = 0.1$ and $p_3 = 0.1$, i.e., approximately 50% of their true values. The adaptation gains for these parameters are 2.0, 0.2 and 0.2 for p_1, p_2, and p_3, respectively.

Fig. 5.17 shows that, after the initial adaptation, link 1 tracking error is reduced significantly. After first 4 seconds, the tracking errors of link 1 and link 2 due to parameter uncertainties became very small. Fig. 5.18 indicates that the estimates of parameter p_1, p_2 and p_3 approach some constant values. Notice that these constant values may not be the true values of the parameter because the desired trajectory is not persistently exciting. Moreover other factors such as friction in motors and unmodeled dynamics can affect parameter adaptation. Various adaptation gains were used and similar experimental results were observed. More experimental results with different gains can be found in Appendix A. From the experimental results shown in Fig. 5.17, it can be seen that by using model-based adaptive control during unconstrained motion, the effect of the uncertainties of manipulator dynamic model uncertainties is reduced and can be ignored after the first cycle.

In this section, the emphasis is given to the performance of position/force control scheme proposed for the constrained motion phase. Recall that the following control and

adaptation laws were used during constrained motion phase.

$$\tau = M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) - F_v e_v - v(q)f_{nd} - v'(q)\widehat{\xi} f_n$$

$$\widehat{\xi} = \xi_0 - \int_0^t \gamma_f e_v^\top v'(q) f_n dt$$

The definition of the variables used in the above equations can be found in section 3.3.

Experimental Results without Grinding Coefficient Estimation

To simplify the problem, in this group of experiments, it is assumed that the parameters of the manipulator and grinding coefficient ξ are known and the constraint surface is exactly known. Surface following is conducted during constrained motion phase to test the position/force control strategy. A plain cylindrical metal tip is used for surface following experiments. To compare the surface following performance with various desired contact force levels the following gains were used for all the experiments in this section.

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 3.162 \\ 0.12 \\ 0.18 \end{bmatrix} \quad \Gamma^\top = 0 \quad \gamma_f = 0$$

The grinding friction coefficient is chosen to be zero for the straight constraint surface, and 0.2 for the circular constraint surface.

In the first experiment, desired normal force value was taken as zero, i.e., $f_{nd} = 0$. Fig. 5.19 shows the normal and tangential forces measured with a force sensor. Fig. 5.19 shows that no normal force was applied to the constraint surface while robot moves along the constraint. Fig. 5.20 indicates that there is a large burr on the surface. Due to the presence of burrs along the surface, contact force values may be observed.

In the second experiment, desired normal force value was taken to be 20 N. Fig. 5.21 shows that the normal force is regulated around the desired force level. Since the constraint surface is assumed to be known, there was no impact when switching controllers from

unconstrained motion to constrained motion. Smooth landing of the robot on to the surface is evident from the force signal in the normal and tangential direction. In addition, from Fig. 5.21, it can be observed that there is a disturbance component with constant frequency in normal and tangential force signals. This is mainly due to the friction in the bearing used to hold the steel tip that is connected to the force sensor. Fig. 5.22 shows that the presence of burrs on the surface did not affect the tracking error as much as it did in the previous experiments because a desired normal force of 20 N was applied on the surface.

In the third experiment, desired normal force level was raised to 40 N. Fig. 5.23 shows the normal and tangential force signals from the force sensor. Regulation of the desired contact force is achieved. Fig. 5.24 shows the normal and tangential position tracking error. Notice that in Fig. 5.24 there are positive jumps in the normal position error immediately after the robot switches to constrained motion control. This is primarily due to the combined compliance of the constraint surface and steel tip on the force sensor. It is also observed that the friction force in the tangential direction results in large a tracking error in the tangential direction.

Fig. 5.25 and Fig. 5.26 show the contact force and the tracking errors of experiments with the steel cylindrical constraint. Since the constraint surface is not straight the normal direction is time varying in this case, the desired normal force is a vector with constant magnitude and time-varying direction. Fig. 5.25 shows that the normal force magnitude during constrained motion was regulated at a desired force level 30 N. Notice that the tangential force value was also stable at a value related to normal force with a constant coefficient. The value of the coefficient is different from the value relating f_n and f_t in Fig. 5.21 and Fig. 5.23 because the material of the constraint surface is different. The profile of normal tracking error during the constrained motion phase indicates the existence of the uncertainties of circle center and radius or imperfect shape of the cylinder. Due to friction force, tangential tracking error with non-zero mean value was observed during the constrained motion phase. Experimental results of different desired force with cylindrical

steel constraint surface can be found in Appendix B.1.

Experimental Results with Grinding Coefficient Estimation

In practical situations, it is difficult to obtain the value of grinding friction coefficient because this value depends on the material of the constraint surface and the robot end-effector, feed-rate, normal force level, grain size of abrasive tools, etc. On the other hand, the above conditions are generally fixed during the constrained motion phase. Without loss of generality, it can be assumed that the normal and tangential forces are related by an unknown constant friction coefficient. An adaptation law is utilized to estimate the value of this friction coefficient. Experiments have been conducted to test this adaptation law. The following control gains are used in these experiments:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 3.162 \\ 0.12 \\ 0.18 \end{bmatrix}$$

Different initial value and adaptation gain are chosen for grinding friction coefficient in each experiment. Experiments have been conducted using a combination of the following gains:

$$\xi_0 = 1.2, 0.3 \quad \gamma_f = 0.1, 0.3$$

for the straight constraint surface and

$$\xi_0 = 1.2, 0.3 \quad \gamma_f = 0.1, 0.3$$

for the circular constraint surface.

Fig. 5.27 and Fig. 5.28 show that the estimate of the grinding friction coefficient with the straight aluminum surface and steel surface, respectively. The results show that the estimates of this friction coefficient converge to constant values in experiments with various initial values and adaptation gains. The friction coefficient values are same for all

experiments with same grinding conditions regardless of their initial values and adaptation gains.

Fig. 5.29 and Fig. 5.30 show the L_2 norm of the normal and tangential tracking error in the constrained motion phase for each cycle. Since the normal tracking errors are determined by constraint surface, normal tracking error during constrained motion does not change for every cycle. However, the experimental results show that tangential tracking is improved with the adaptation of grinding coefficient. The L_2 norm of the normal force error for each surface is shown in Fig. 5.31 and Fig. 5.32. Notice that the on-line adaptation of the friction coefficient can also help to improve the normal force tracking. This is due to the fact that an accurate estimate of grinding coefficient leads to better tangential force compensation.

In summary, the experimental results in this section verify the performance of the position/force controller and the gradient-type adaptation algorithm for friction coefficient proposed in section 3.3. The adaptation law was designed to estimate the friction coefficient for different contact conditions. On-line estimation of the friction coefficient was implemented to improve tangential force compensation. Experimental results show much improved performance of the proposed control strategy with tangential force compensation and friction coefficient adaptation when compared with the results of the hybrid position/force control available in literature, which did not consider force compensation.

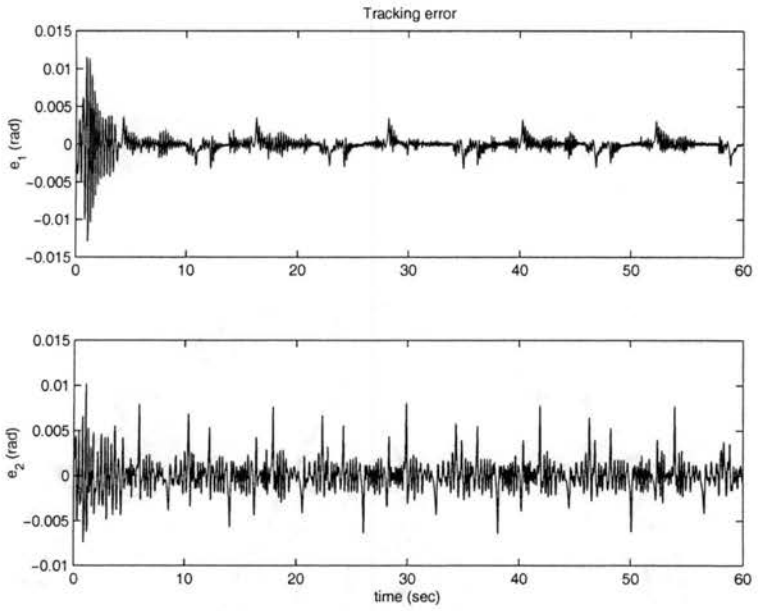

Figure 5.17: Tracking error of model-based adaptive control

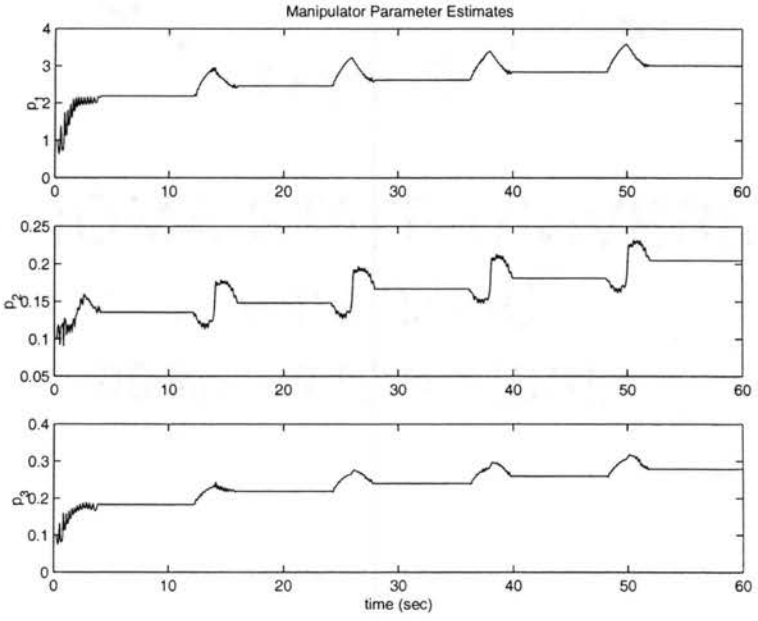

Figure 5.18: Manipulator parameter estimates for model-based adaptive control

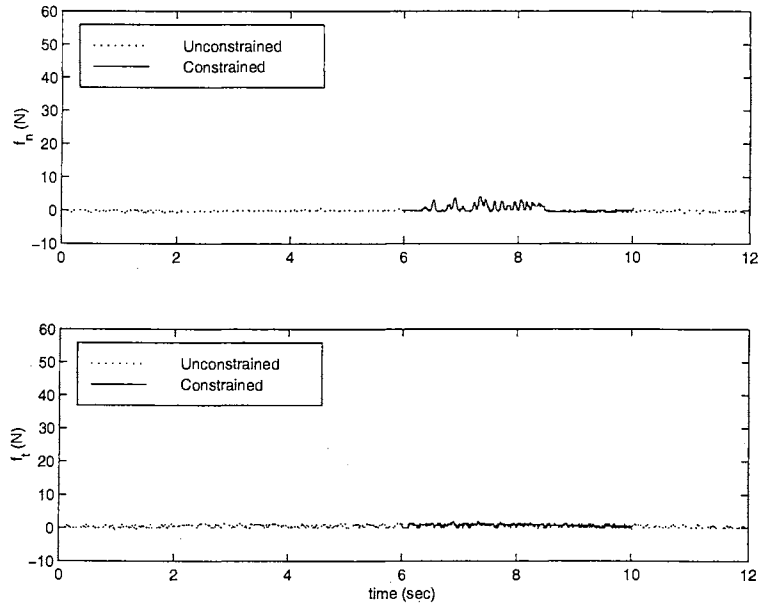

Figure 5.19: Contact force under straight constraint: $f_{nd} = 0$

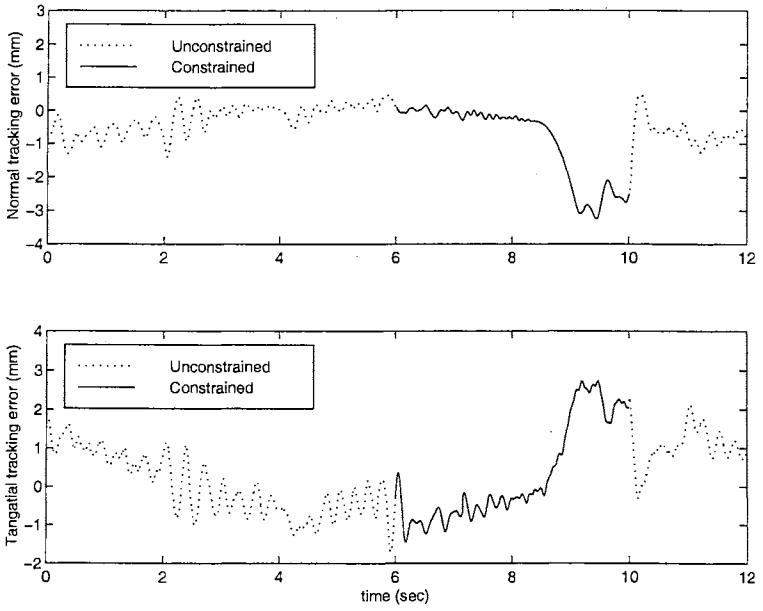

Figure 5.20: Tracking error under straight constraint: $f_{nd} = 0$

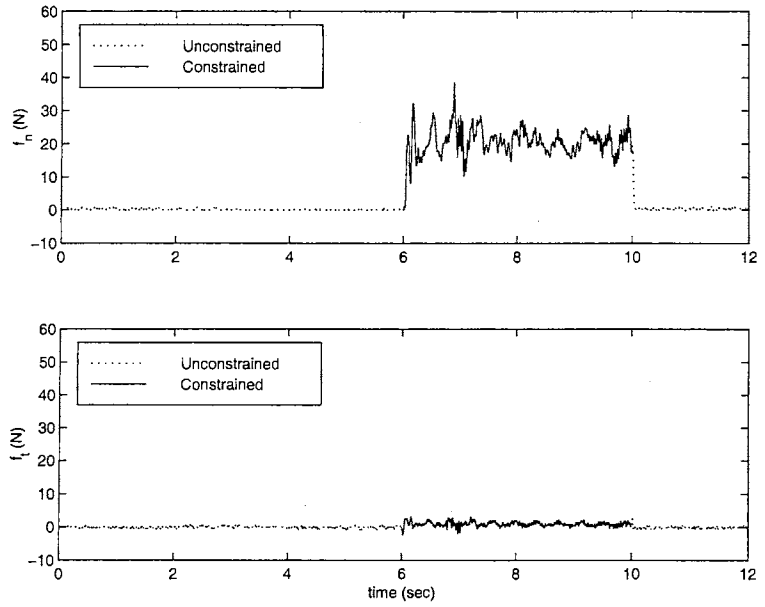

Figure 5.21: Contact force under straight constraint: $f_{nd} = 20$ N

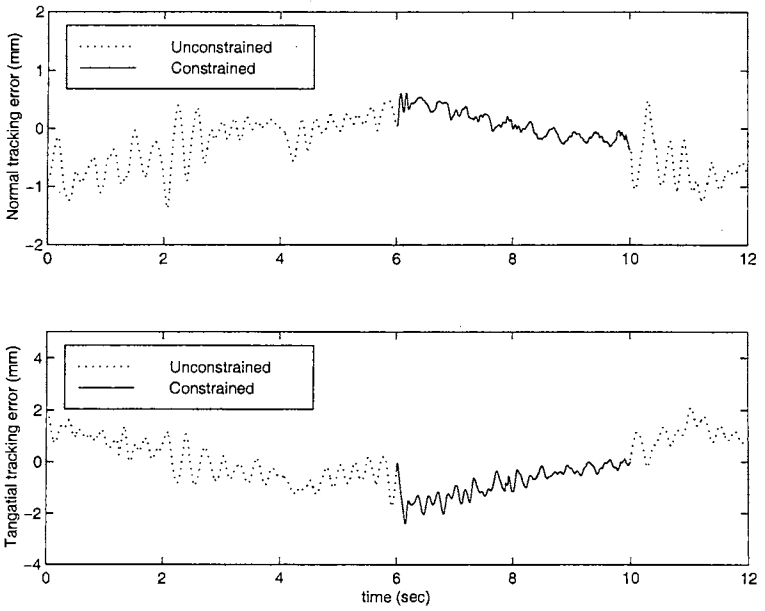

Figure 5.22: Tracking error under straight constraint: $f_{nd} = 20$ N

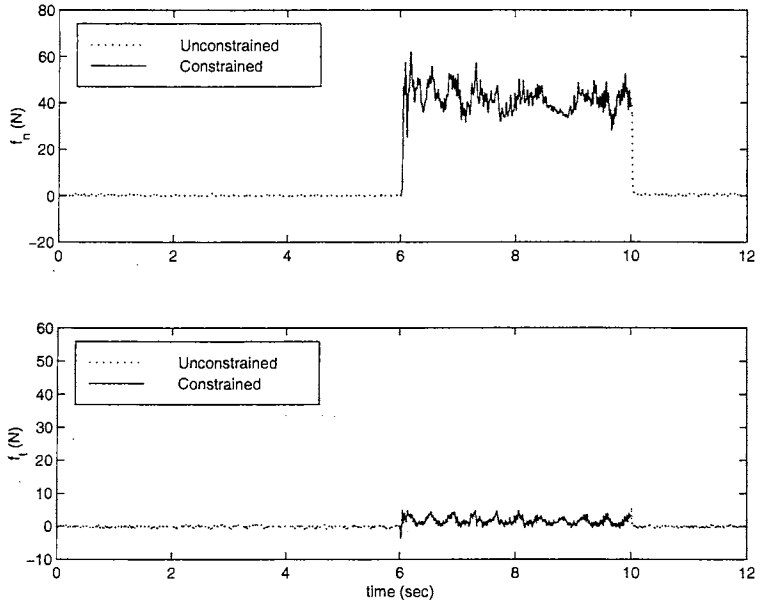

Figure 5.23: Contact force under straight constraint: $f_{nd} = 40$ N

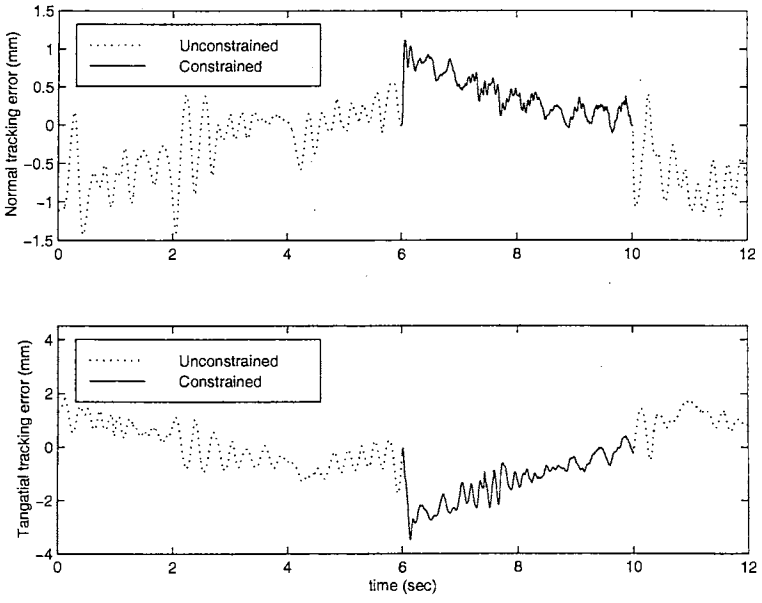

Figure 5.24: Tracking error under straight constraint: $f_{nd} = 40$ N

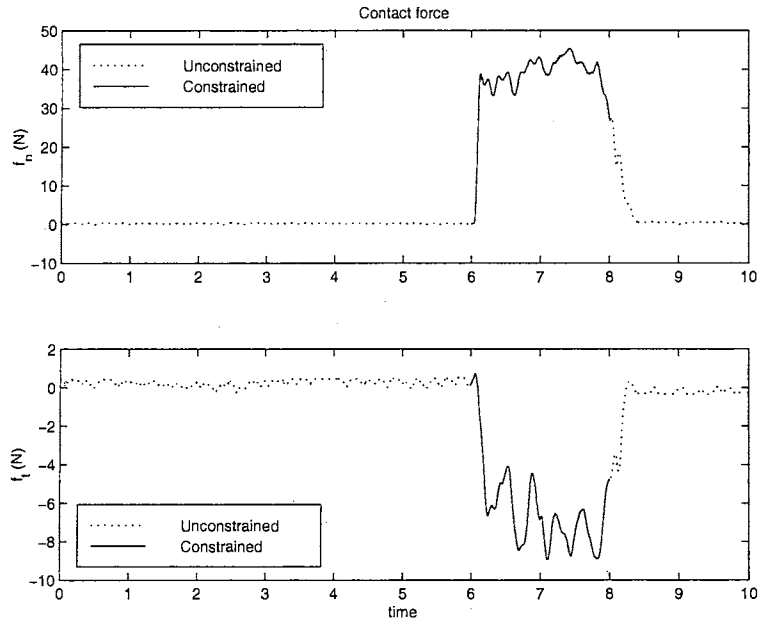

Figure 5.25: Contact force under cylindrical constraint: $f_{nd} = 40$ N

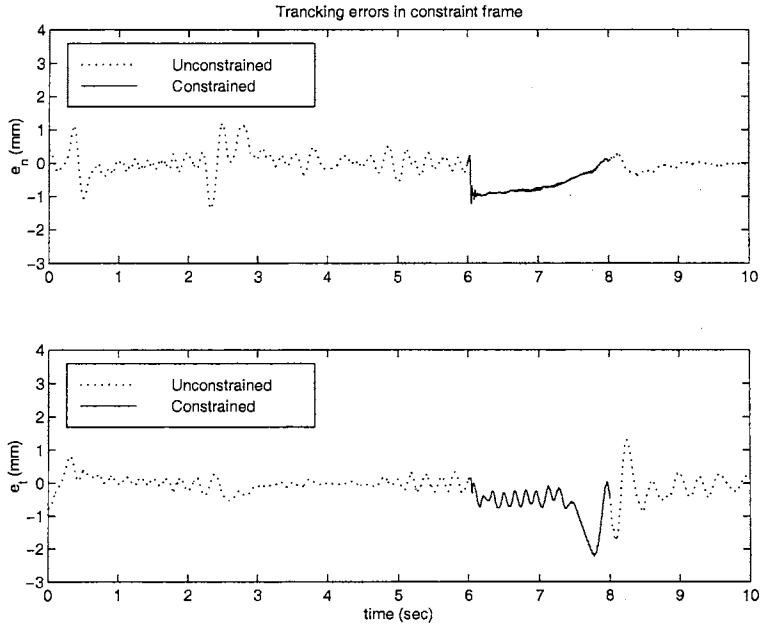

Figure 5.26: Tracking error under cylindrical constraint: $f_{nd} = 40$ N

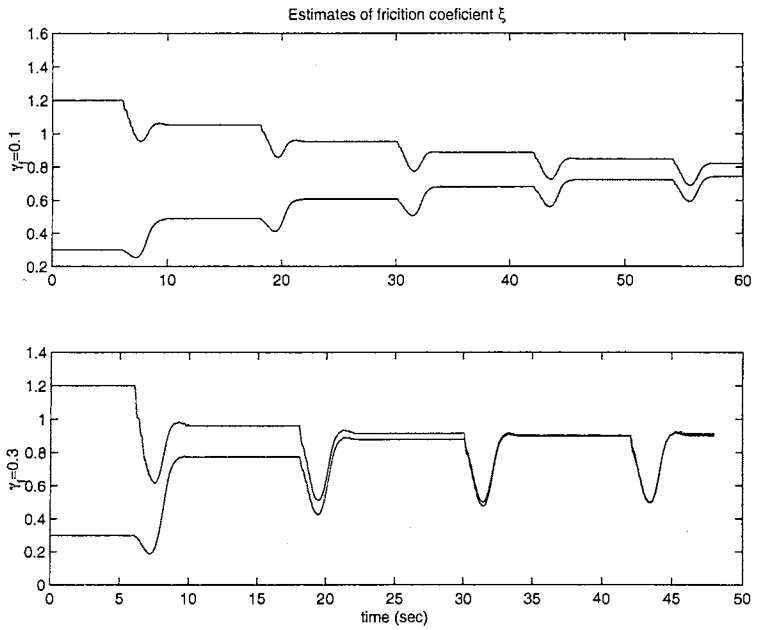

Figure 5.27: Estimates of friction coefficient under straight constraint

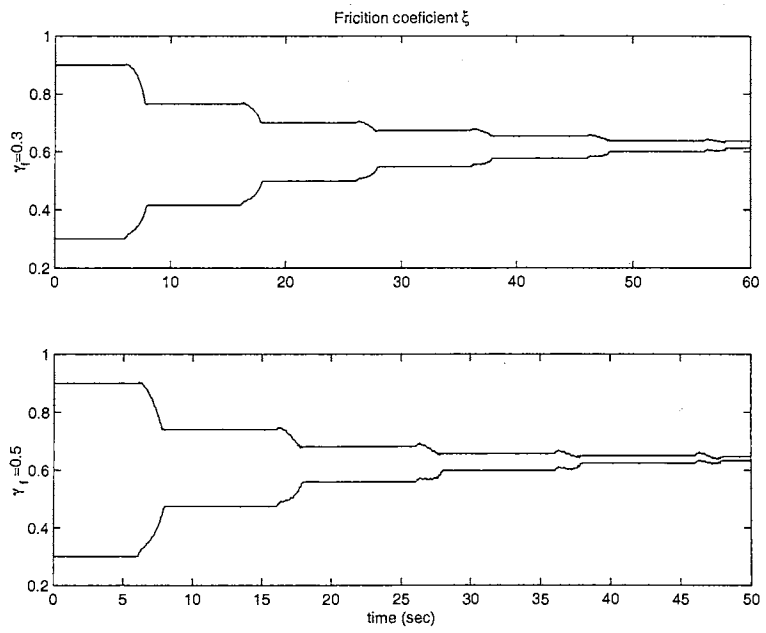

Figure 5.28: Estimates of friction coefficient under cylinder constraint

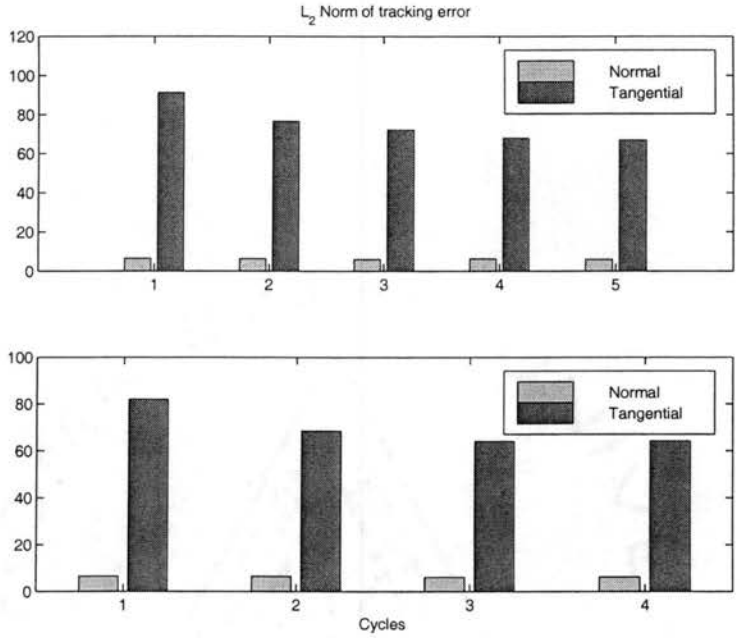

Figure 5.29: L_2 norm of tracking error under straight constraint

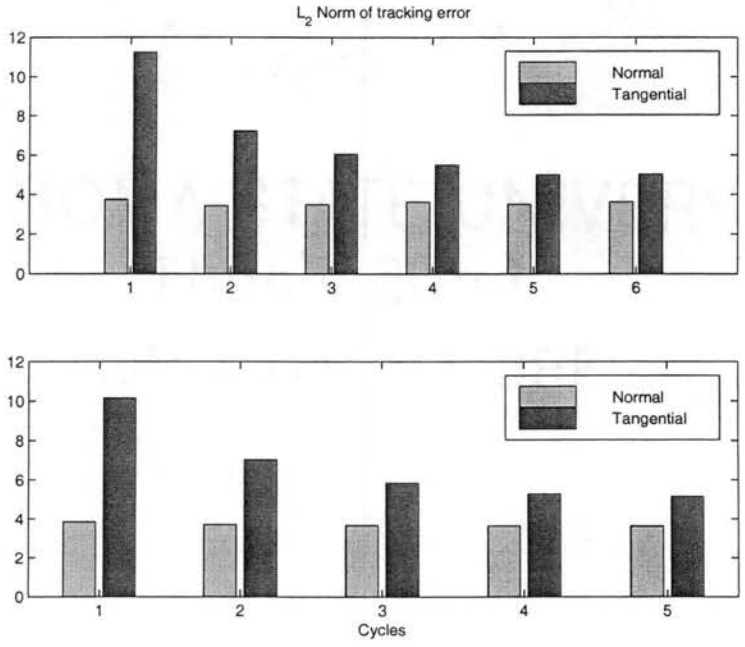

Figure 5.30: L_2 norm of tracking error under cylindrical constraint

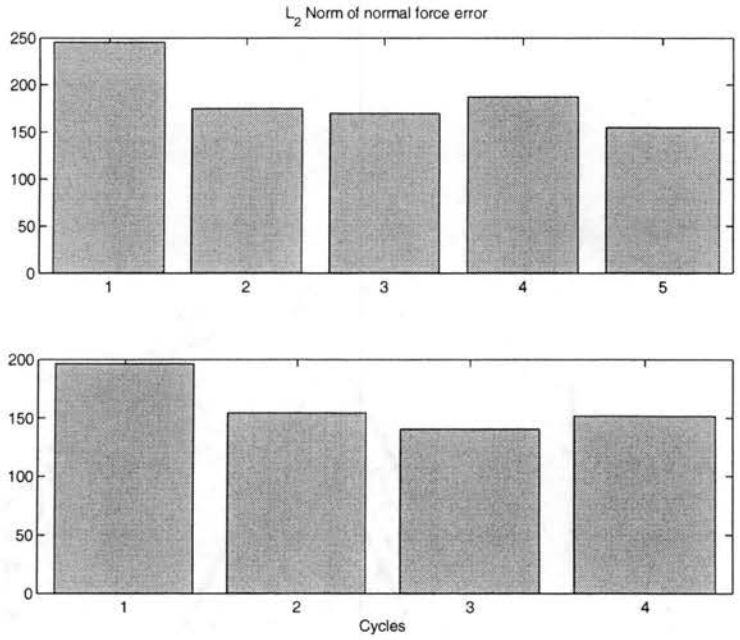

Figure 5.31: L_2 norm of normal force error of straight constraint

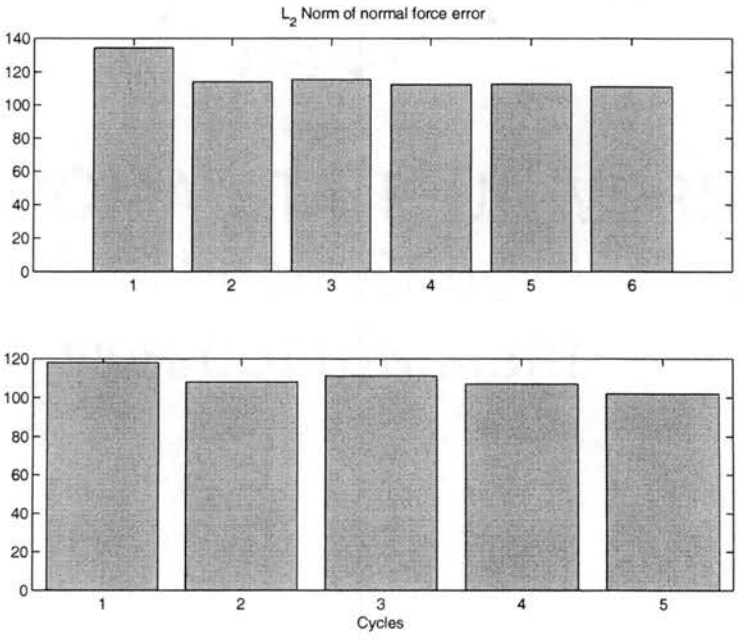

Figure 5.32: L_2 norm of normal force error of cylindrical constraint

5.3 Transition Control

For a complete robot task involving constrained motion, robot moves in free space before the end-effector makes contact with the constraint. If the robot end-effector impacts the surface with a non-zero normal velocity then the end-effector has to be stabilized on to the surface before any operation, such as polishing, deburring, or chamfering, can be performed. From a practical point of view, impact is difficult to completely avoid during transition phase because of constraint uncertainties, tracking error, and other disturbances. If the constraint surface is rigid, impact forces on the robot during the transition phase may become very large even when the velocity normal to the surface is very small at contact. In this section, an experimental study of free impact of the robot manipulator is presented in section 5.3.1 to gain insight into the impact phenomena that takes place during the transition phase. Desired trajectory used for transition control experiments is described in section 5.3.2. Experimental results for a complete task with transition control are given in section 3.2. The goal of these experiments is to test the effectiveness of the transition control algorithm proposed in section 3.2 for rigid surface.

5.3.1 Experimental Study of Free Impact of the Manipulator

An experimental understanding of the impact phenomena will facilitate design of a stable controller during the transition phase. It has been observed that the post-impact robot behavior varies considerably for different pre-impact conditions. Some observed aspects include a jump in the tangential velocity, and in some cases reversal of tangential velocity component at the point of impact. Further, for some pre-impact conditions, the post-impact tangential velocity may increase. Different pre-impact conditions and manipulator configurations are considered in this section. Insight from these experiments can facilitate better

understanding of the impact behavior and can lead to more efficient controller design for the transition phase in robotics.

The experimental procedure consists of the manipulator moving towards the surface at a certain impact angle and making contact with the surface. The procedure involves maintaining a prescribed velocity of the end-effector and the angle of impact using the motor torques. Just prior to the end-effector impacting the surface the motor torques are shut-off to mimic free impact with the prescribed velocity and angle. Joint angles, joint velocities, and force on the end-effector are collected every four milli-seconds. The two manipulator impact configurations, up-elbow and down-elbow, are shown in Figs. 5.33 and 5.34, respectively, where α represents the angle of impact. For each configuration three velocities of impact (0.1 m/s, 0.2 m/s, and 0.3 m/s) and four different angles of impact (30 deg., 45 deg., 60 deg., and 90 deg.) were considered. Impact velocities larger than 0.3 m/s were not tried as the impact force exceeds the force sensor safety limit.

The normal and tangential velocity profile of the end-effector for an impact velocity of 0.2 m/s and different angles of impact are shown in Figs. 5.35 and 5.36 for up and down-elbow configurations, respectively. Notice that there is a jump in the normal and tangential velocity components at impact. The experimental results indicate that the normal velocity component always decreases in magnitude after impact. However, for different manipulator configurations, the tangential velocity component may increase in magnitude and/or reverse its direction after impact for the same pre-impact velocity. It is interesting to observe that the tangential velocity jump increases with increase in the angle of impact.

Figs. 5.37 and 5.38 show the tangential and normal impact force magnitude near the impact for up-elbow and down-elbow configurations. Notice that the time axis interval is only of 0.2 seconds duration. The normal and tangential impact forces shown in the figure are forces on the robot as recorded by the force sensor. For both up-elbow and down-elbow configurations the normal and tangential impact forces increase with increase in the impact angle for both configurations. Observe that the tangential impact force for

up-elbow configuration is in different direction to that of the down-elbow configuration, which is consistent with the tangential velocity jump. Furthermore, the tangential force changed sign during impact for down-elbow configuration in Fig. 5.38. This is consistent with velocity reversal observed in Fig. 5.36. Also, notice that the tangential impact force is around 10 percent or less of the normal impact force, which validates the assumption made in section 2.2 that the normal force impulse dominates the tangential force impulse.

The robot path in the Cartesian space near impact is shown in Figs. 5.39 and 5.40 for up-elbow and down-elbow configurations, respectively. The dashed lines in the figures represents the constraint surface and the solid line represents the robot path. Figs. 5.39 and 5.40 show that the angle of departure of the end-effector is smaller than the angle of impact. This would mean that the ratio of normal and tangential velocity before impact is smaller than that of the velocity after impact. Notice that there is a reversal in direction of the robot end-effector for the down-elbow configuration for angles of impact greater than 45 degrees. For both configurations, decrease in the magnitude of normal velocity was observed for all cases. However, the change in tangential velocity can only be predicted provided that the pre-impact velocity, the manipulator mass matrix, and the Jacobian matrix are all known. From our experimental results, decrease and increase may happen in the magnitude of tangential velocity. Thus, the impact dynamics given by (2.19) is validated. Further, observe that the robot end-effector tip appears to go into the surface, which is due to the overall compliance of the end-effector and the surface.

Figs. (5.41) and (5.42) show the kinetic energy profile of the manipulator during impact for up-elbow and down-elbow configuration with impact velocity 0.2 m/s and various impact angles. The kinetic energy drops after impact. Increase in impact angle results in an increase in the amount of kinetic energy drop. The post-impact behavior for impact velocity of $v = 0.1$ m/s and $v = 0.3$ m/s is similar to the case of 0.2 m/s shown in this section. Complete experimental results for three impact velocities can be found in Appendix C. Figs. 5.43 and 5.44 give a summary of ratio of drop in manipulator kinetic energy

during impact to the pre-impact kinetic energy for various conditions. The ratio of drop in manipulator kinetic energy is computed as follows:

$$\text{Kinetic energy drop ratio} = \frac{(\dot{q}^+)^\top M(q)\dot{q}^+ - (\dot{q}^-)^\top M(q)\dot{q}^-}{(\dot{q}^-)^\top M(q)\dot{q}^-} \tag{5.3}$$

Experimental results indicate that the kinetic energy drop increases with increase of impact velocity and increase of impact angle.

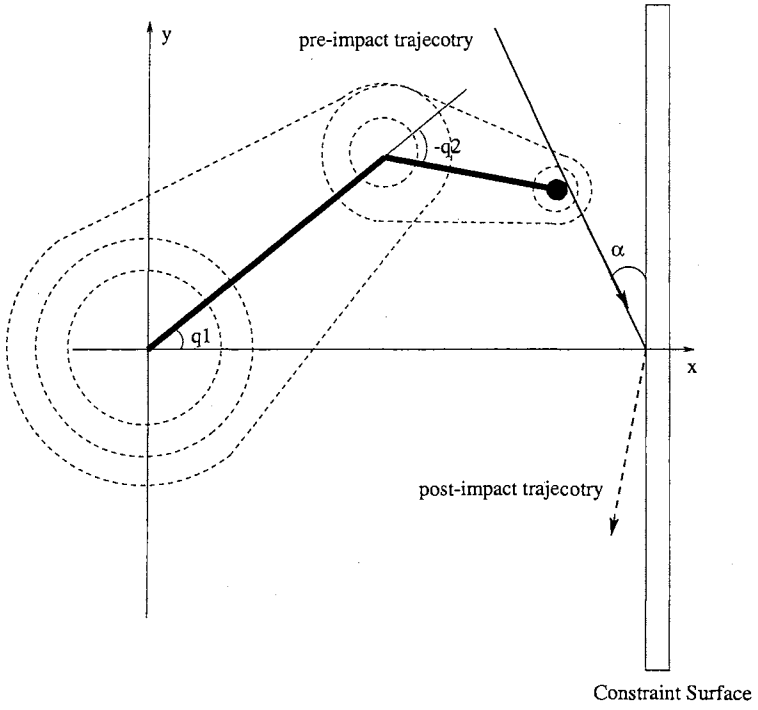

Figure 5.33: Up-elbow configuration

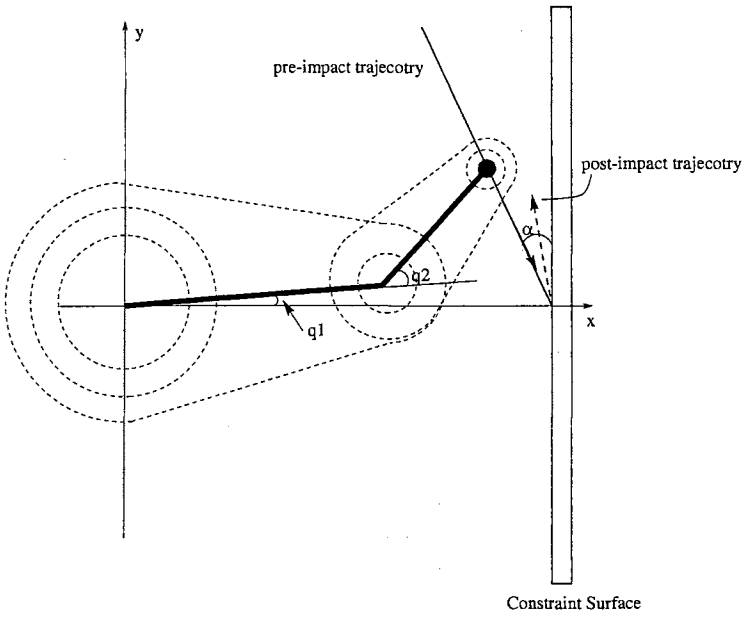

Figure 5.34: Down-elbow configuration

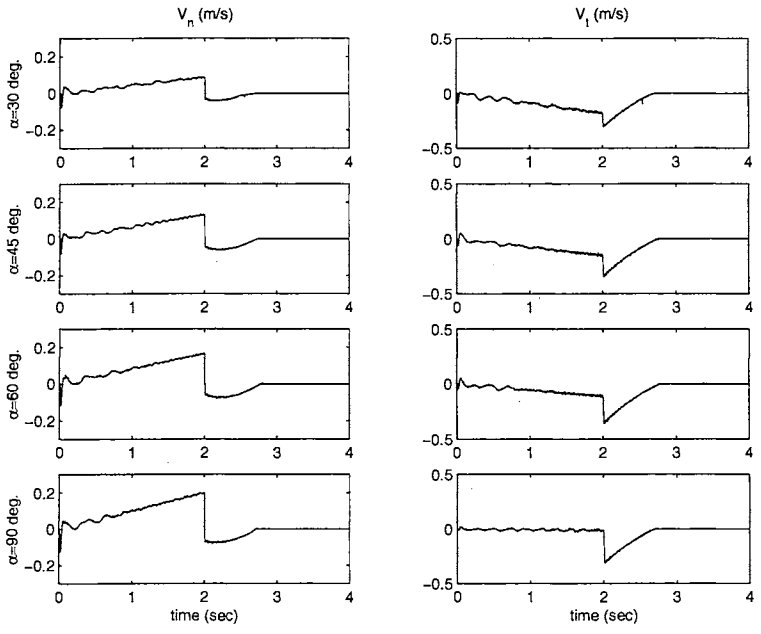

Figure 5.35: Velocity, pre-impact velocity $v = 0.2$ m/s, up-elbow

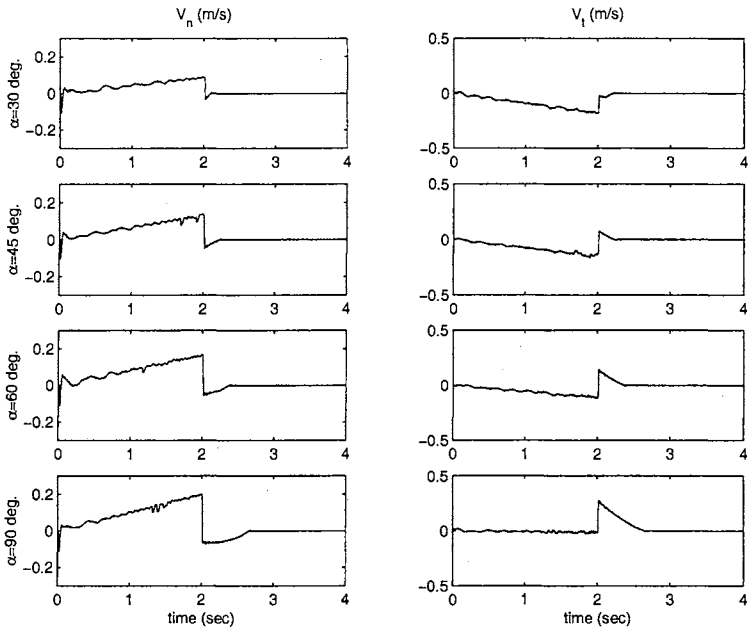

Figure 5.36: Velocity, pre-impact velocity $v = 0.2$ m/s, down-elbow

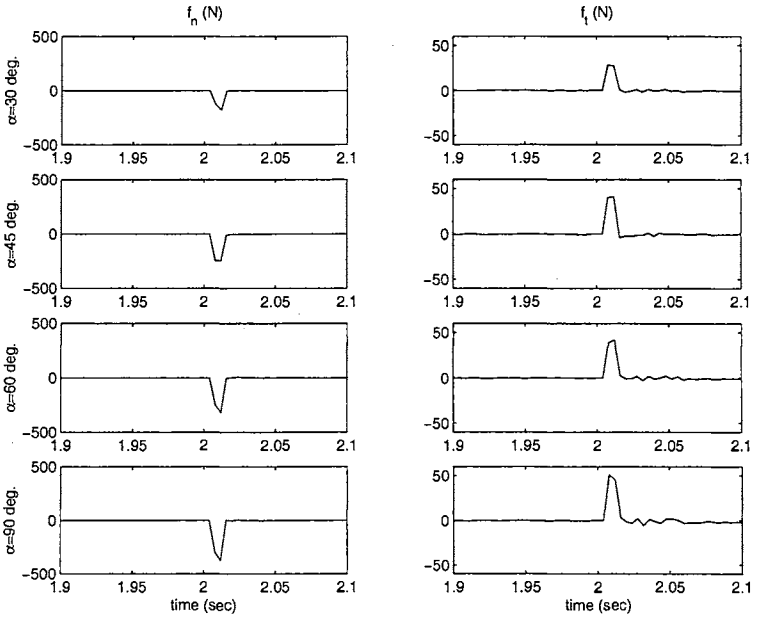

Figure 5.37: Impact force, pre-impact velocity $v = 0.2$ m/s, up-elbow

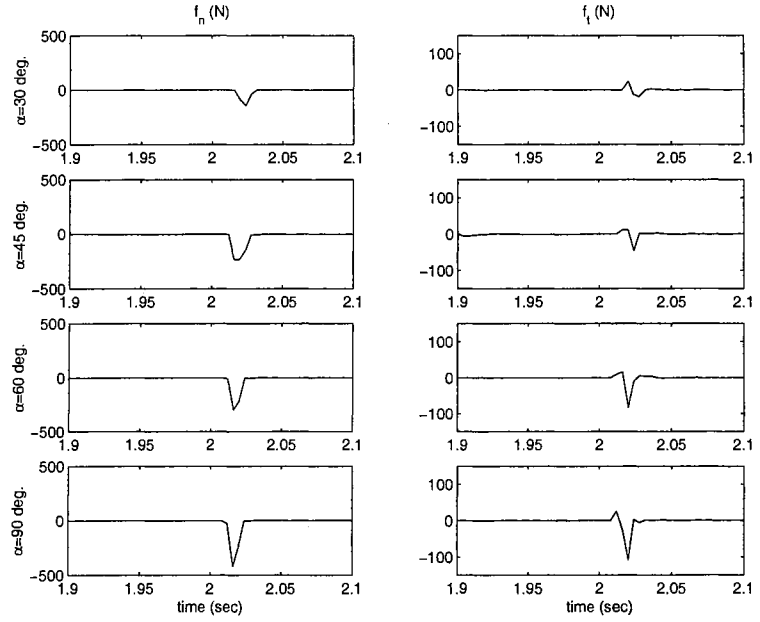

Figure 5.38: Impact force, pre-impact velocity $v = 0.2$ m/s, down-elbow

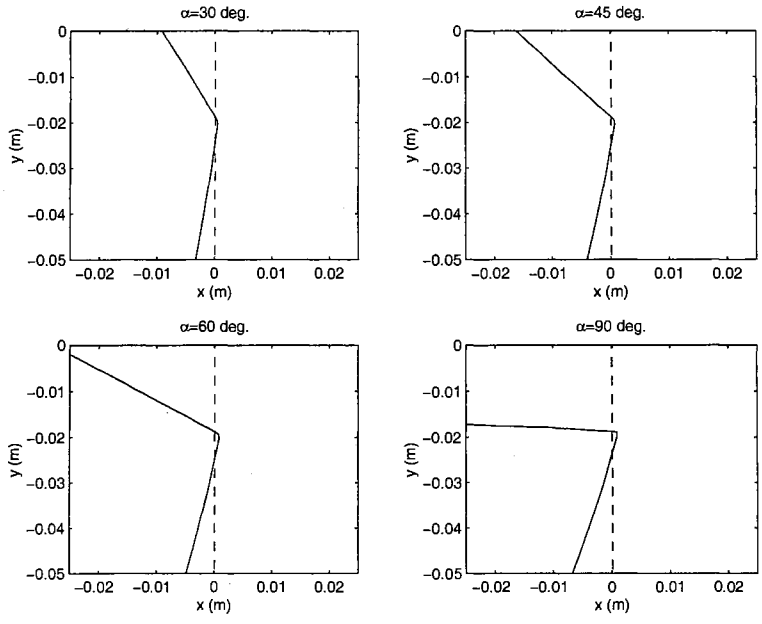

Figure 5.39: Robot path near impact, pre-impact velocity $v = 0.2$ m/s, up-elbow

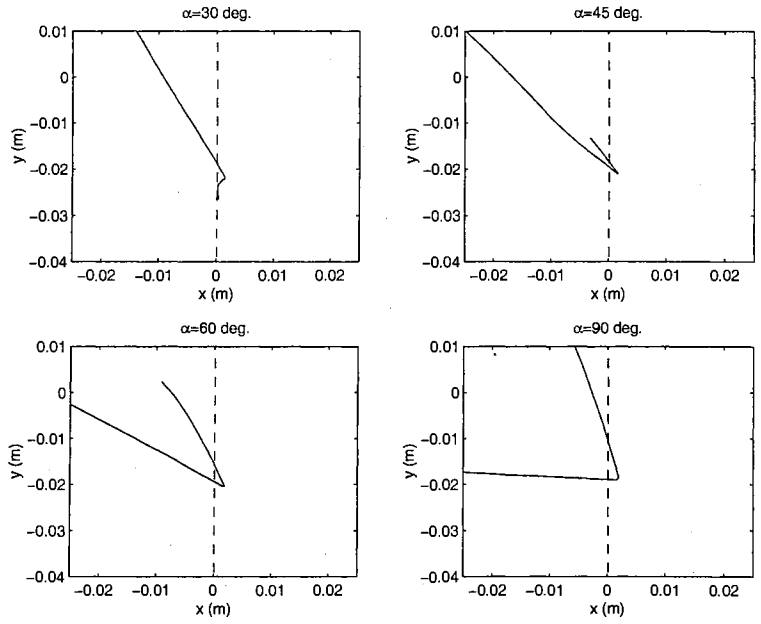

Figure 5.40: Robot path near impact, pre-impact velocity $v = 0.2$ m/s, down-elbow

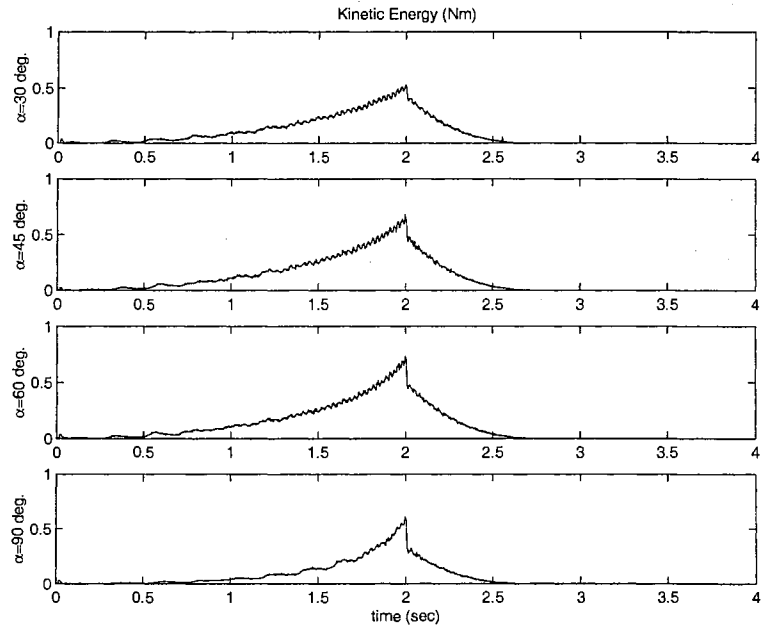

Figure 5.41: Robot kinetic energy, pre-impact velocity $v = 0.2$ m/s, up-elbow

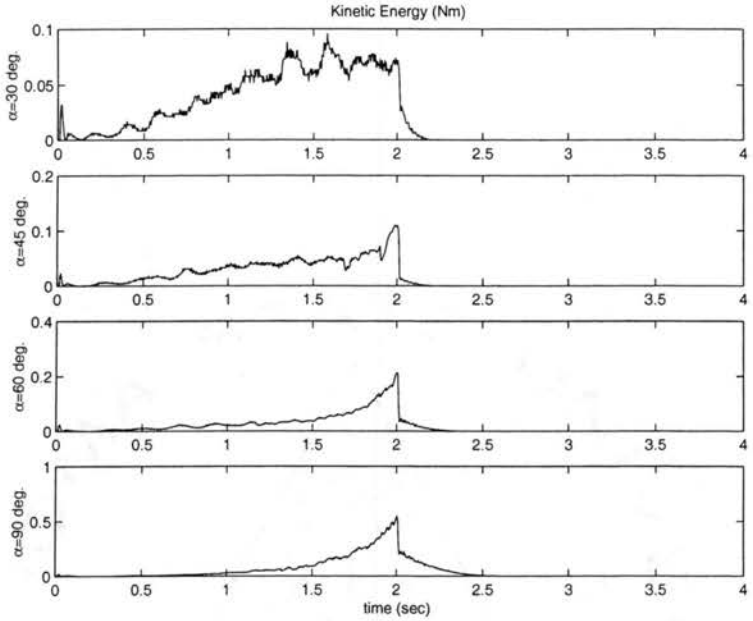

Figure 5.42: Robot kinetic energy, pre-impact velocity $v = 0.2$ m/s, down-elbow

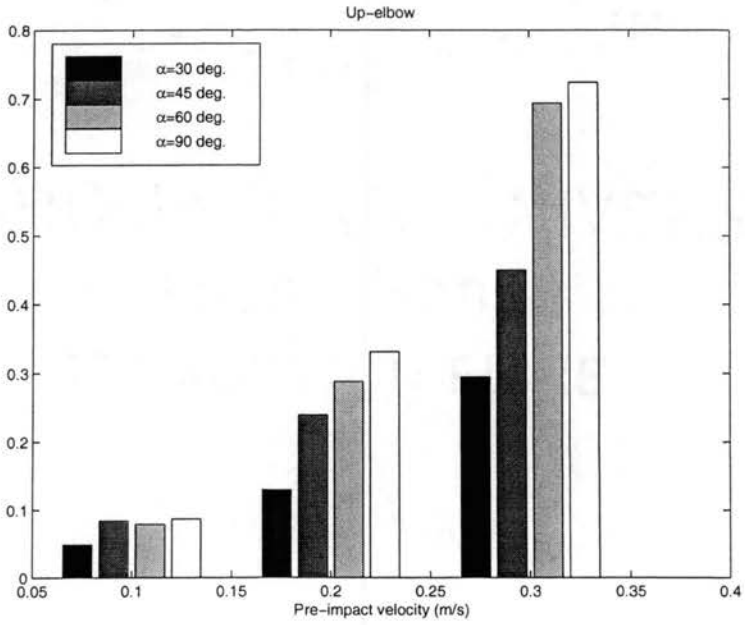

Figure 5.43: Kinetic energy drop ratio during impact, up-elbow

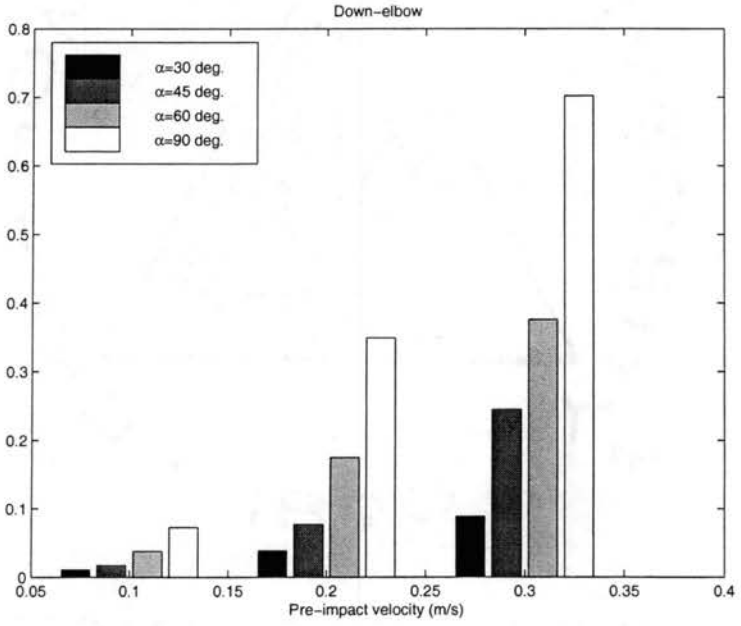

Figure 5.44: Kinetic energy drop ratio during impact, down-elbow

5.3.2 Desired Trajectory

In transition control experiments, the constraint surface is chosen as a rigid straight wall, which is a thick aluminum sheet firmly held by a vice as shown in Fig. 4.1. The constraint surface is given in (5.1). Fig. 5.45 illustrates the two-link robot (top view) and the constraint surface in the Cartesian space. In Fig. 5.45, the desired trajectory of the robot is **CDABC** (bold line). Uncertainty in the constraint location is represented by ζ. Therefore, depending on the uncertainty, there are three possibilities: (1) $\zeta = 0$, there is no normal velocity at contact, i.e., the robot lands on the surface smoothly; (2) $\zeta > 0$, the robot end-effector will impact the surface with non-zero normal velocity anywhere along the path **DA**; and (3) $\zeta < 0$, the robot will not make contact with the surface. When case (3) is encountered, the proposed control algorithm simply performs motion control along the desired trajectory. Experimental results of case (1) and case (2) with the proposed control strategy are shown in this section. The angle of the constraint surface with the y-axis, α in Fig. 5.45, is 0.0460 radians and is assumed to be known. The known location of the constraint used to construct the desired trajectory is $d = 0.513$ m.

The experiments is distinguished with two different desired trajectories in terms of the desired travel speed tangential to the surface during constrained motion phase, i.e., on segment $A \rightarrow B$. The desired travel speed during the constrained motion phase is denoted by v and is equal to 0.068 m/s, Fig. 5.46, and 0.14 m/s, Fig. 5.47, for low and high speed trajectories, respectively. The duration of each cycle of the desired trajectory **CDABC** (see Fig. 5.45) is equal to 12 and 6 seconds for the low speed and high speed, respectively.

5.3.3 Experimental Results of Transition Control

Extensive experiments were conducted with the proposed control methodology with different levels of constraint uncertainty and different desired travel speeds. Recall that the

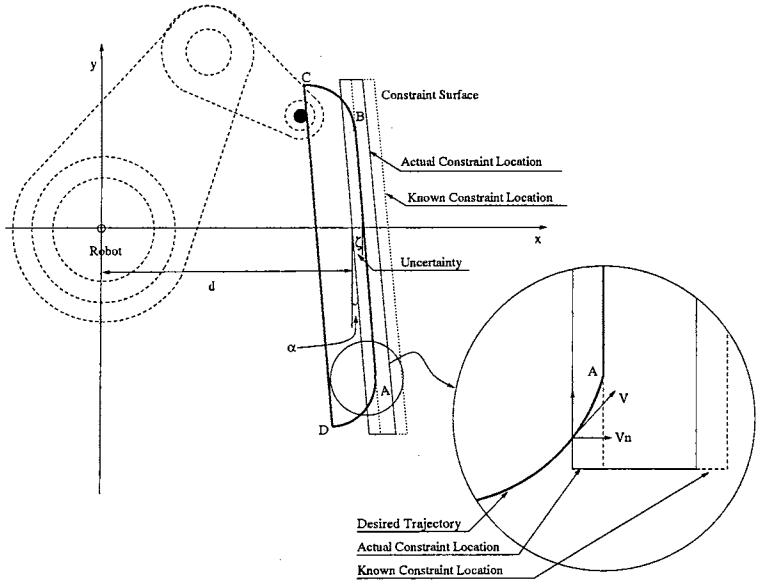

Figure 5.45: Desired trajectory and constraint uncertainty

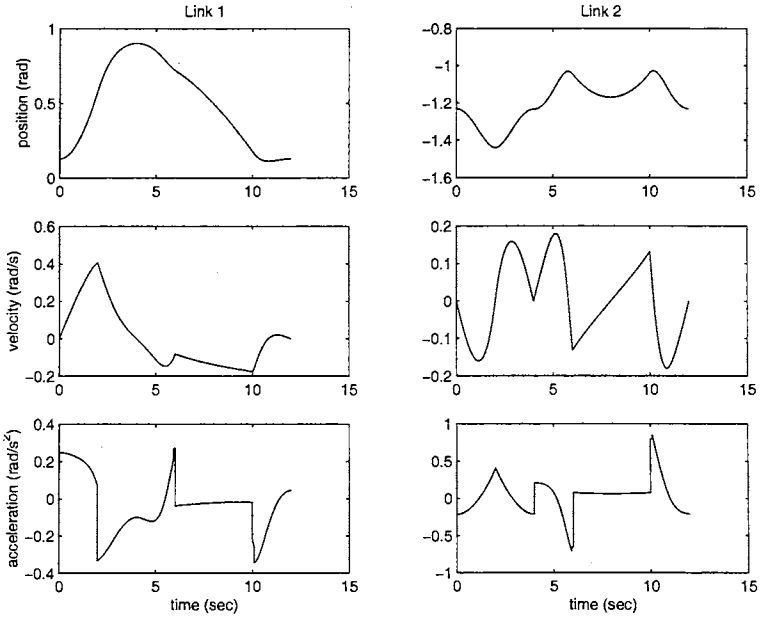

Figure 5.46: Desired trajectory, low speed

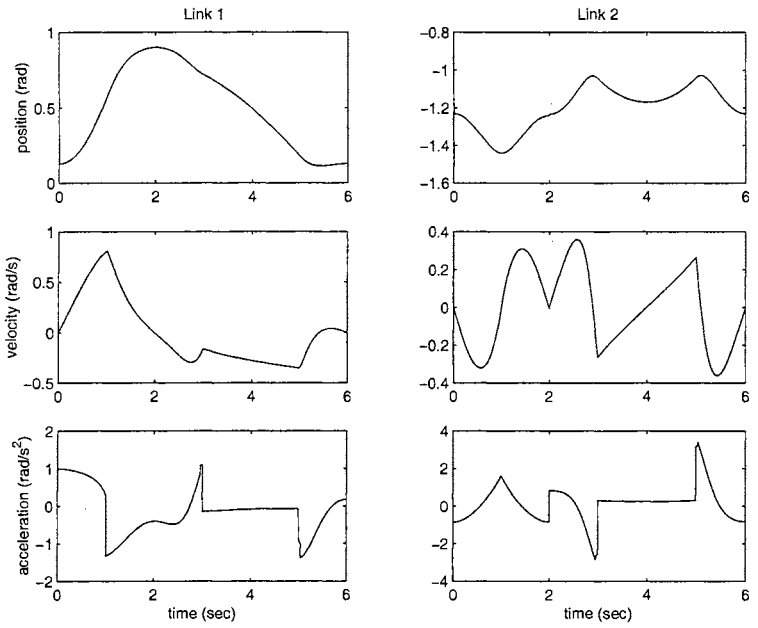

Figure 5.47: Desired trajectory, high speed

following control algorithm (3.29) is used during the transition phase.

$$\tau = Y(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\beta - F_v e_v - \lambda_{tn} P_\phi \text{sgn}(e_{vn})$$

This section presents experimental results for three different levels of uncertainty ($\zeta = 2.5$ mm, $\zeta = 5$ mm and $\zeta = 7.5$ mm) and two different velocities of travel ($v = 0.068$ m/s and $v = 0.14$ m/s) on the surface during the constrained motion phase. Also, for all the experiments, the proposed control scheme is compared with an algorithm that directly switches from unconstrained motion control to constrained motion control without considering transition.

The experimental procedure is to follow the desired trajectory **CDABC** as shown in Fig. 5.45. Unconstrained motion control is applied in the trajectory sections **CD**, **DA**, and **BC**. Constrained motion and force control is applied along the surface **AB**. Transition control is activated at first impact. In the direct switch algorithm, constrained motion and force control is activated according to the design of the pre-computed desired trajectory, i.e., at point **A**, ignoring transition. In all the experiments, a servo sampling rate of 250 Hz is employed. Since force measurements are very noisy, a fixed length moving average filter is

used to filter noisy force measurements. Also, since the force data can be collected at higher sampling rates, a force sensor sampling rate of 2 kHz is used. The higher force sampling rate reduces delay due to filtering. The issues related to control algorithm implementation will be addressed experimentally in section 5.5. The following are the control gains used in these experiments:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 3.162 \\ 0.12 \\ 0.18 \end{bmatrix} \quad \lambda_{tn} = \begin{bmatrix} 1.2 \\ 0.6 \end{bmatrix}$$

The results for each experiment, shown in Fig. 5.48 through Fig. 5.59, correspond to consecutive implementation of three cycles of the desired trajectory. Fig. 5.48 shows that the normal position error for low speed ($v = 0.068$ m/s) and for three cases: (1) smooth landing (no uncertainty, $\zeta = 0$ mm), (2) direct switch ($\zeta = 5$ mm), and (3) transition control ($\zeta = 5$ mm). Fig. 5.49 gives experimental results for high speed ($v = 0.14$ m/s). The desired normal force is 45 N. Notice that without transition control at impact the robot bounces severely on the constraint surface. When the transition controller is used during transition phase, the robot settles on the surface after the first bounce. For the direct switch case, it should be observed that the end-effector tip seems to go into the surface, this is due to the compliance of the end-effector assembly and/or surface. Figs. 5.50 and 5.51 give the normal force for low and high speeds, respectively, for the three cases. The results show that directly switching from unconstrained motion control to constrained motion and force control results in severe bounces of the robot on the surface. This effect is magnified for the high speed case. Figs. 5.52 and 5.53 give the tangential force for low and high speeds, respectively. The tangential force data shows that the surface has some friction. To better illustrate the bounces on the surface, Cartesian trajectories are plotted in Figs. 5.54 and 5.55. In these figures, the dashed line represents the actual location of the constraint surface. The dotted line represents the pre-computed desired trajectory of the robot tip, which is based on a priori knowledge of the location of the constraint. The solid line is the

actual path followed by the robot tip.

Figs. 5.56 and 5.57 give the L_2 norm of the normal force error for direct switch and transition control. With no constraint uncertainty, $\zeta = 0.0$ mm, both direct switch and transition control give similar performance as expected. With constraint uncertainties $\zeta = 2.5$ mm and $\zeta = 5.0$ mm, the performance is much better with the transition controller.

Further, to illustrate the performance of the proposed control algorithm, experimental results with an uncertainty of 7.5 mm is shown in Figs. 5.58 through 5.60. To focus on the behavior of the robot tip in the transition phase, Fig. 5.60 shows the Cartesian trajectory and the normal force for a short duration of time around the transition phase. The first impact with the constraint surface is denoted by '$\star$' in Fig. 5.60. Notice that the normal force becomes zero after a short duration of time after the first impact, which means that at point 1 the robot leaves the constraint surface. Also, from the Cartesian trajectory it appears that at points 2 and 3 the robot tip leaves the surface. But this is not the case, since the normal force magnitude at points 2 and 3 is non-zero. Although the surface is rigidly fixed to the vice, there is some compliance in the surface and the end-effector assembly that causes small vibrations after the impact. Therefore, with the proposed transition controller even for a constraint uncertainty of $\zeta = 7.5$ mm, it was observed the stable landing of the robot tip on the surface with just one bounce. For this level of uncertainty, the direct switch algorithm renders the system unstable.

In addition to the given figures, the results of numerous experiments was summarized in Table 5.1 and Table 5.2. Table 5.1 gives a summary of the peak normal force and Table 5.2 gives the number of bounces of the robot tip on the constraint surface. Experimental data summarized in Tables 5.1 and 5.2 show that the impact force peaks and the number of bounces are reduced when transition control is applied. The complete experimental results for transition control can be found in Appendix D. The experimental data shows that uncertainty in the location of the constraint surface will cause impact of the robot with the surface. The impact force and bounces during the transition phase are affected not only by

	$v = 0.14$ m/s		$v = 0.068$ m/s	
Constraint certainty	Direct switch	Transition control	Direct switch	Transition control
0.0 mm	62.3	56.1	73.7	56.4
2.5 mm	101.5	63.5	136.9	52.4
5.0 mm	293.8	101.3	282.8	67.9
7.5 mm	No data	104.0	No data	65.5

Table 5.1: Summary: peak normal force (N)

	$v = 0.14$ m/s		$v = 0.068$ m/s	
Constraint certainty	Direct switch	Transition control	Direct switch	Transition control
0.0 mm	0	0	0	0
2.5 mm	2	0	2-3	0
5.0 mm	4-15	1	4	0
7.5 mm	No data	1	No data	0

Table 5.2: Summary: number of rebounds

constraint and robot model uncertainties but are also affected by the approaching velocity of the robot and the rigidity of the constraint. Higher approach speeds will result in longer transition phase. Increase in rigidity of the constraint surface results in larger impact forces. Furthermore, with direct switch, the peak impact force may not happen at the first impact. Control of transition from free motion to constrained motion is essential to maintain stability of the system when the constraint location is uncertain. The impact force and the number of bounces due to impact are significantly reduced by using the robust discontinuous controller proposed in section 5.3.3. The effectiveness of the transition controller and the entire control strategy is validated by these experimental results.

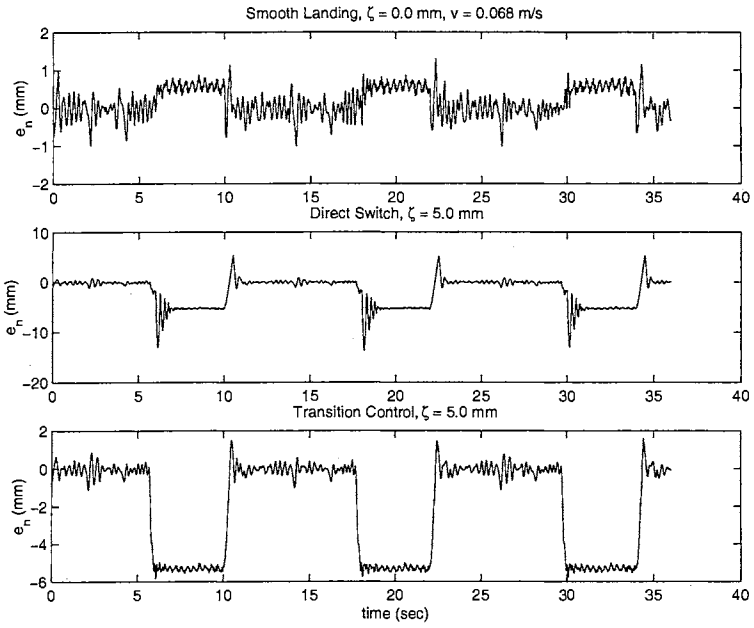

Figure 5.48: Normal tracking error, low speed

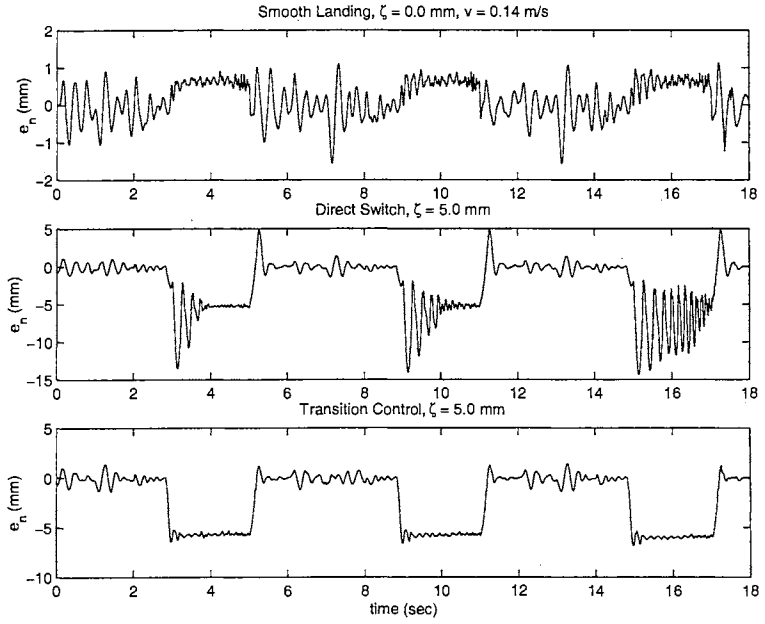

Figure 5.49: Normal tracking error, high speed

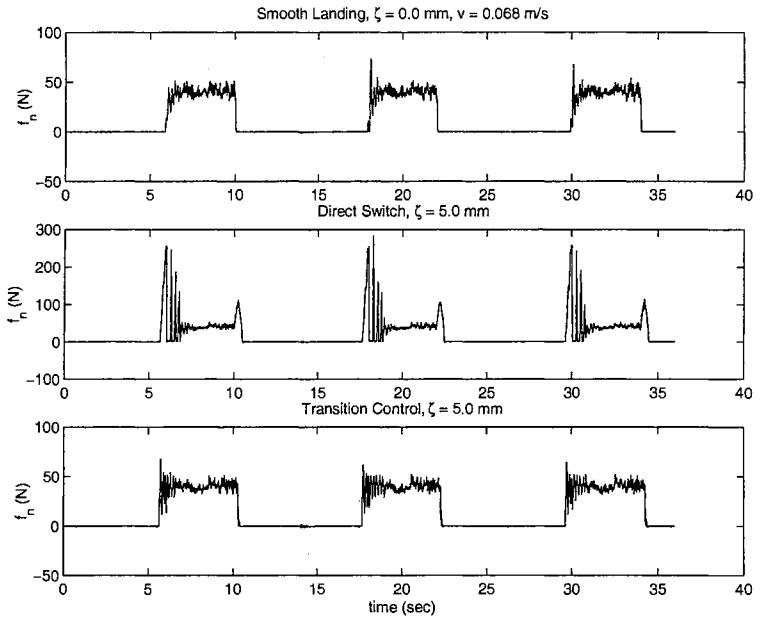

Figure 5.50: Normal contact force, low speed

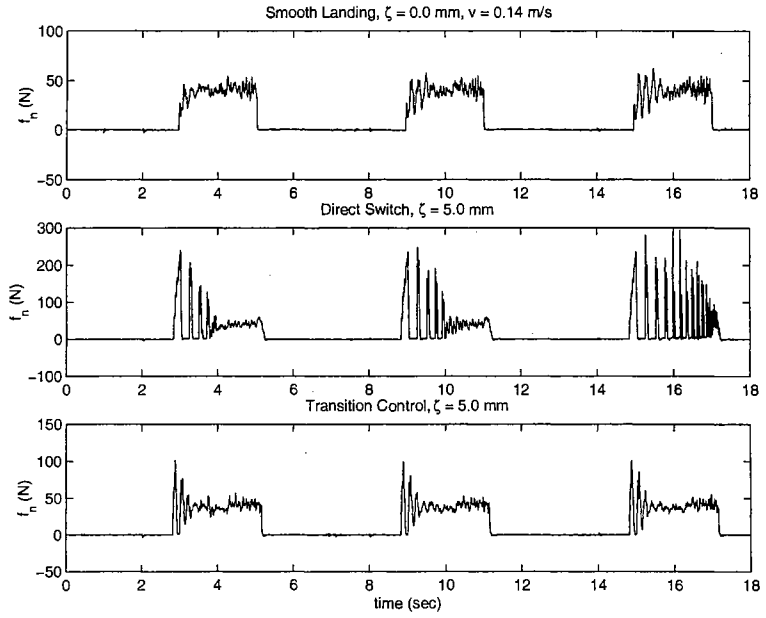

Figure 5.51: Normal contact force, high speed

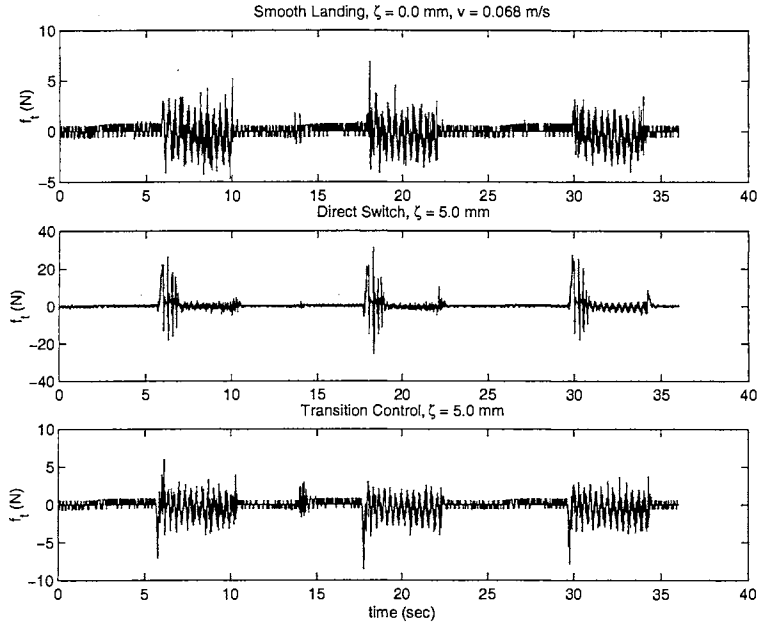

Figure 5.52: Tangential contact force, low speed

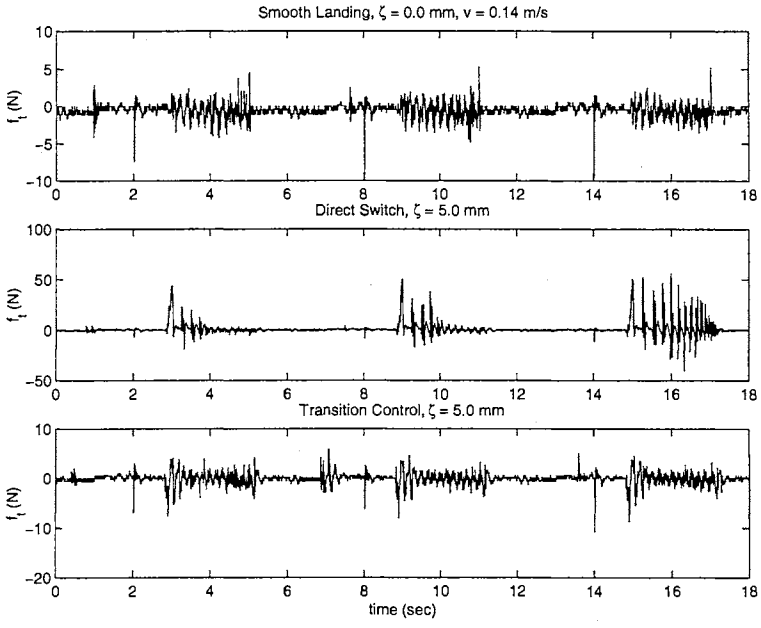

Figure 5.53: Tangential contact force, high speed

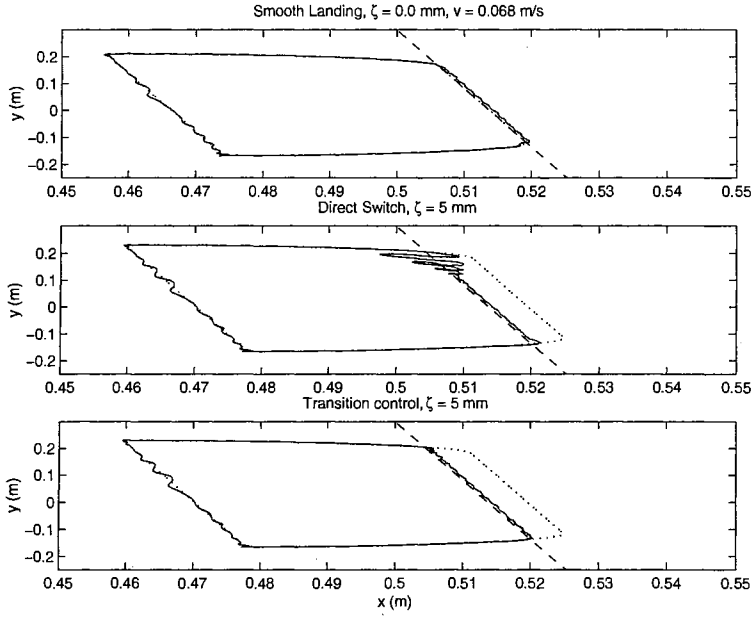

Figure 5.54: Cartesian trajectory, low speed

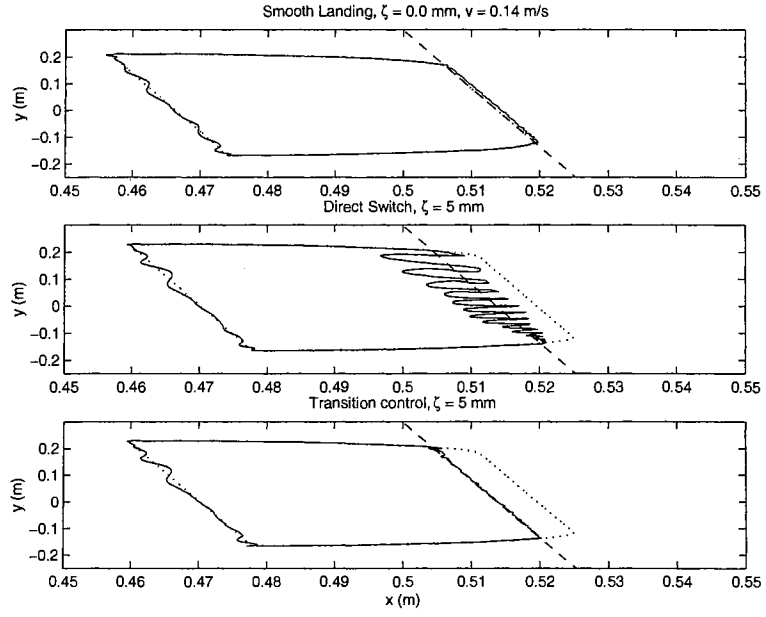

Figure 5.55: Cartesian trajectory, high Speed

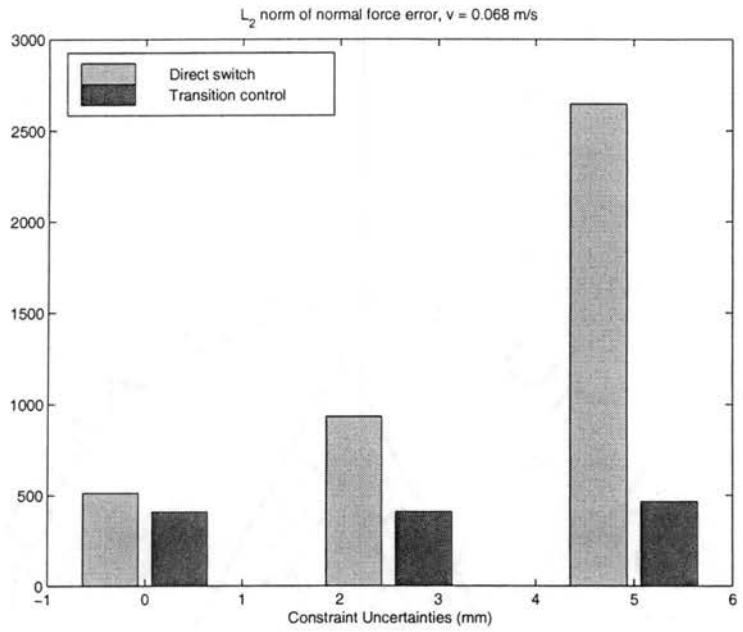

Figure 5.56: Normal contact force error L_2 norm, low speed

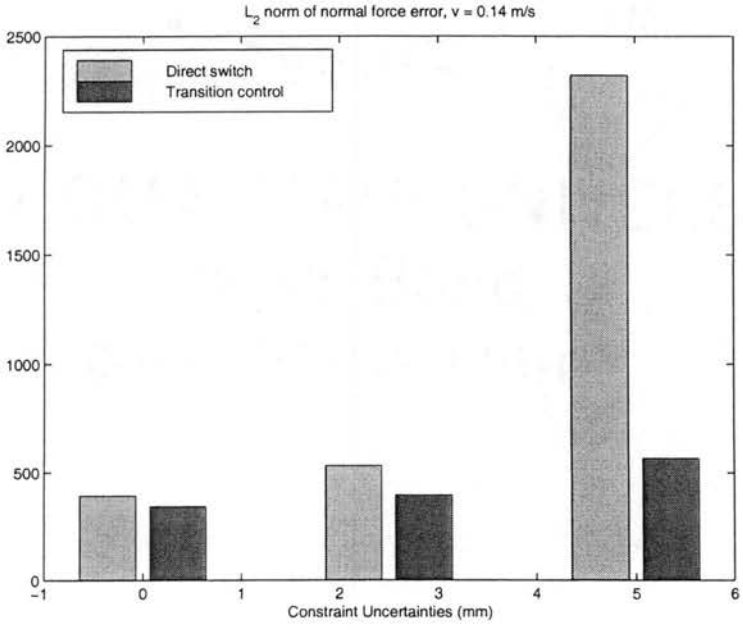

Figure 5.57: Normal contact force error L_2 norm, high speed

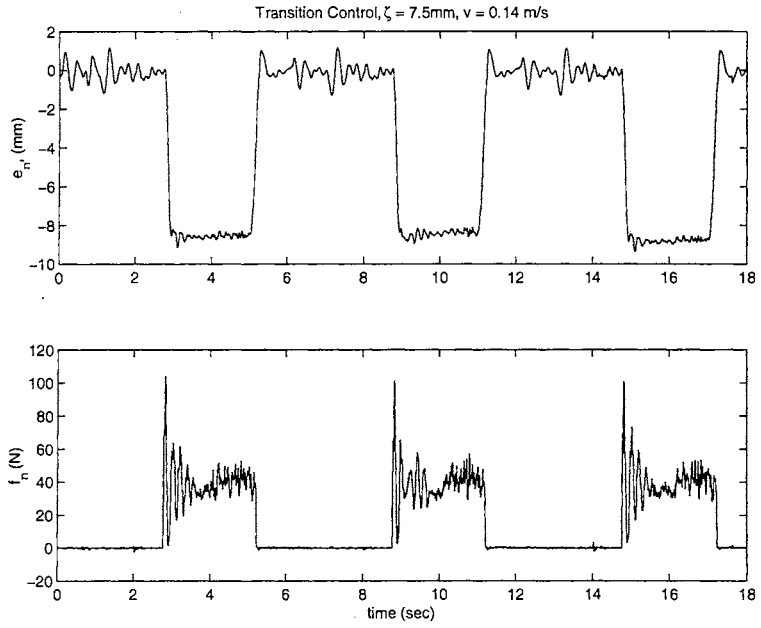

Figure 5.58: Normal tracking error and normal contact force using transition control, $\zeta = 7.5$ mm

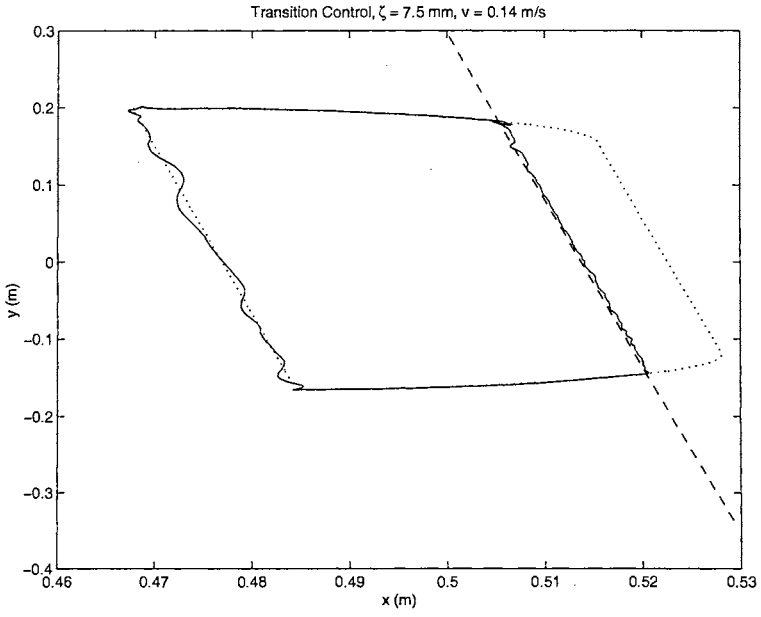

Figure 5.59: Cartesian trajectory using transition control, $\zeta = 7.5$ mm

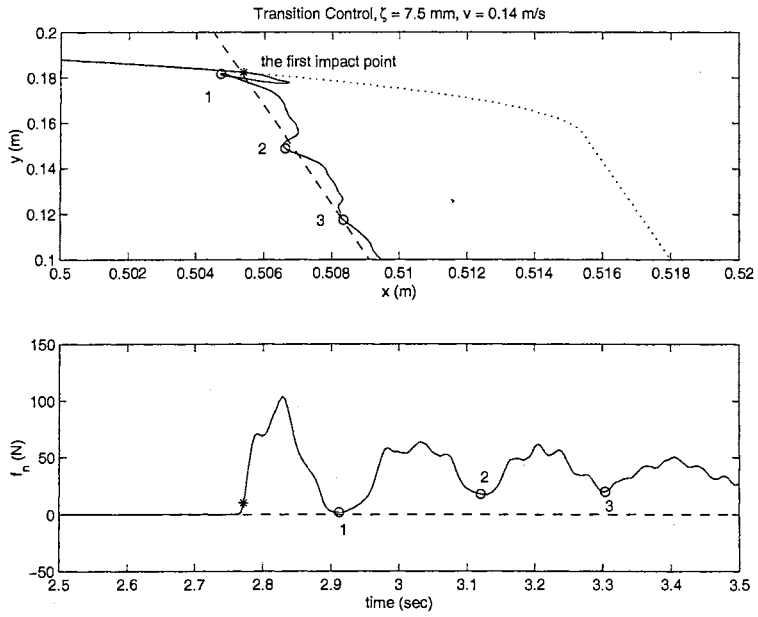

Figure 5.60: Motion and contact force near impact, transition Control, $\zeta = 7.5$ mm

5.4 Surface Finishing

In this section, experimental results of surface finishing are presented. The constraint surface $\phi(q)$ of these experiments is defined in equation (5.1) in section 5.2.1. Desired trajectory is the same as the trajectory 1 shown in Fig. 5.13 and Fig. 5.14 in section 5.2.1. For all the experiments in this section the desired normal force is 15 N. The following control gains are used:

$$F_v = \begin{bmatrix} 20 & 0 \\ 0 & 4 \end{bmatrix} \quad \Lambda_p = \begin{bmatrix} 100 & 0 \\ 0 & 40 \end{bmatrix} \quad \beta_0 = \begin{bmatrix} 3.162 \\ 0.12 \\ 0.18 \end{bmatrix} \quad \lambda_{tn} = \begin{bmatrix} 1.2 \\ 0.6 \end{bmatrix}$$

Similar to up and down milling in machining, a robotic surface finishing operation can be categorized into two modes depending on the tool rotation and the direction of travel of the tool mounted on the robot end-effector. The two modes are illustrated in Fig. 5.61 and Fig. 5.62. For mode 1, the normal force f_n caused by surface finishing process pushes the tool away from the surface. In contrast, the normal force holds the tool on the surface in mode 2. Hence, more stable contact can be achieved in mode 2. In mode 1, the tangential force f_t is in the same direction as the tangential motion along the surface. In mode 2, f_t is in the opposite direction of motion. Hence, the magnitude and sign of the coefficient of grinding friction ξ in cutting force model are different for mode 1 and 2. For tangential force compensation, grinding friction coefficient of 0.8 and 1.2 are used for mode 1 and mode 2, respectively.

Surface finishing experiments were conducted in both mode 1 and mode 2 with different work piece location uncertainty. Experimental results of mode 1 are shown in Fig. 5.63 and Fig. 5.64. Fig. 5.63 contains normal and tangential position tracking errors with constraint uncertainty $\zeta = 0$ and $\zeta = 1$ mm. During the surface finishing process, given by time interval between 6 and 10 seconds on the plots, the compliance of the tool contributes to

the normal tracking error. Stable contact is achieved. Fig. 5.64 shows that normal force is regulated around the desired level of 15 N during surface finishing. Constraint uncertainty causes a larger peak force and oscillations when the tool makes contact with the surface.

Experimental results of mode 2 are shown in Fig. 5.65 and Fig. 5.66. Fig. 5.65 indicates that the normal position tracking error of mode 2 is similar to that of mode 1. However, the tangential tracking error is different for mode 1 and mode 2. This is due to different tangential force direction of mode 1 and mode 2. Fig. 5.66 shows that normal forces are regulated around the desired level of 15 N during surface finishing with and without constraint uncertainty. It should be observed that peak normal force and force oscillation when contact is made are lower for mode 2 when compared with mode 1.

Stable contact and normal force regulation is achieved in both surface finishing modes using the proposed control strategy. Comparing experimental results of mode 1 and mode 2 indicates that better surface finish can be obtained in mode 2. Fig. 5.67 and Fig. 5.68 give the discrete FFT of normal and tangential force signals. Normal forces of mode 1 and mode 2 all contain DC component whose magnitude is 15 N. This implies that force normal to the surface is regulated at desired level for all cases. Also, the tangential forces are kept at constant level determined by the value of ξ. Since the magnitude of f_t may be larger than that of f_n, tangential force compensation becomes critical in obtaining the necessary material removal and surface finish. For mode 1, noise components around 1.5 Hz are observed. In contrast, no significant noise components in f_n and f_t signals are observed in mode 2.

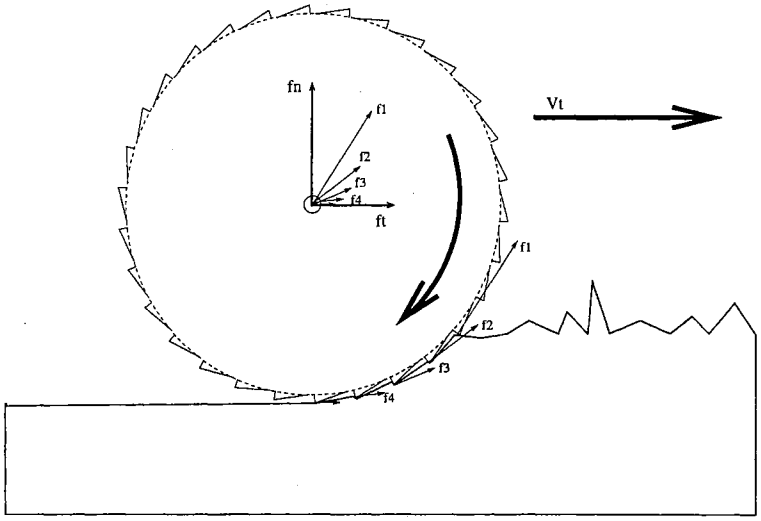

Figure 5.61: Surface finishing mode 1

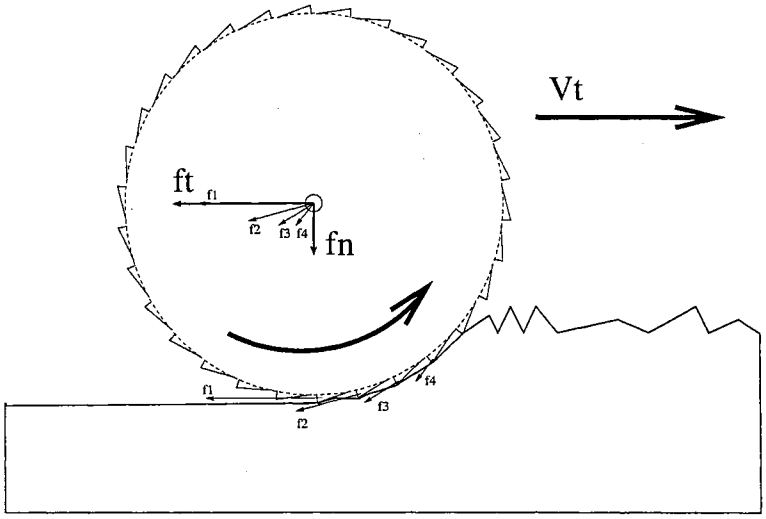

Figure 5.62: Surface finishing mode 2

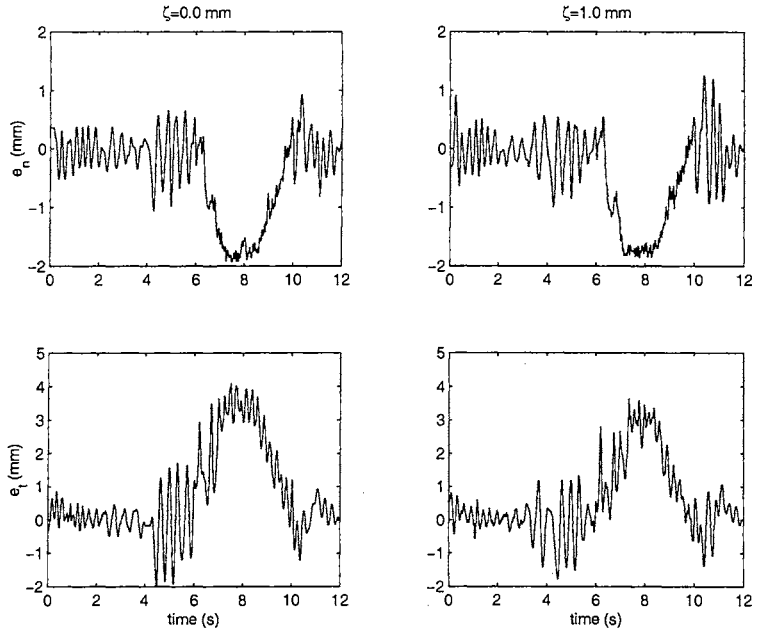

Figure 5.63: Tracking errors of surface finishing, mode 1

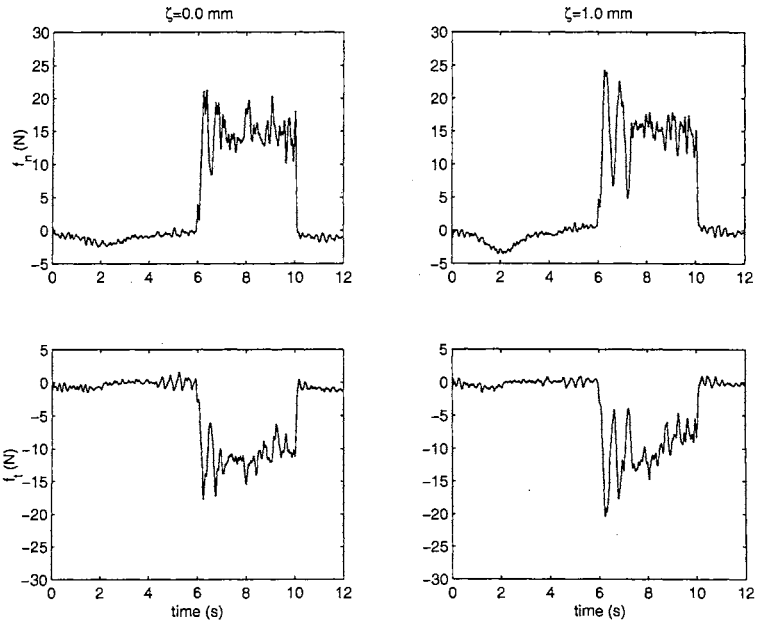

Figure 5.64: Forces of surface finishing mode 1

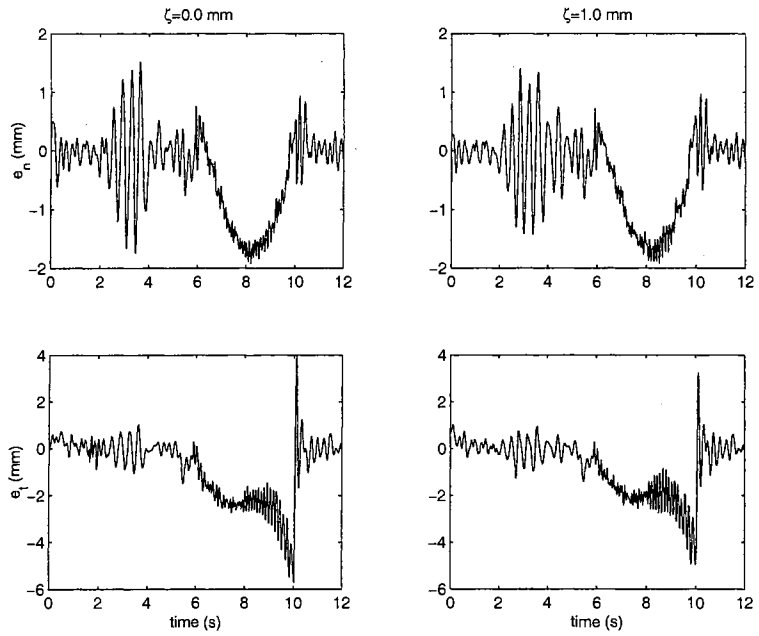

Figure 5.65: Tracking errors of surface finishing, mode 2

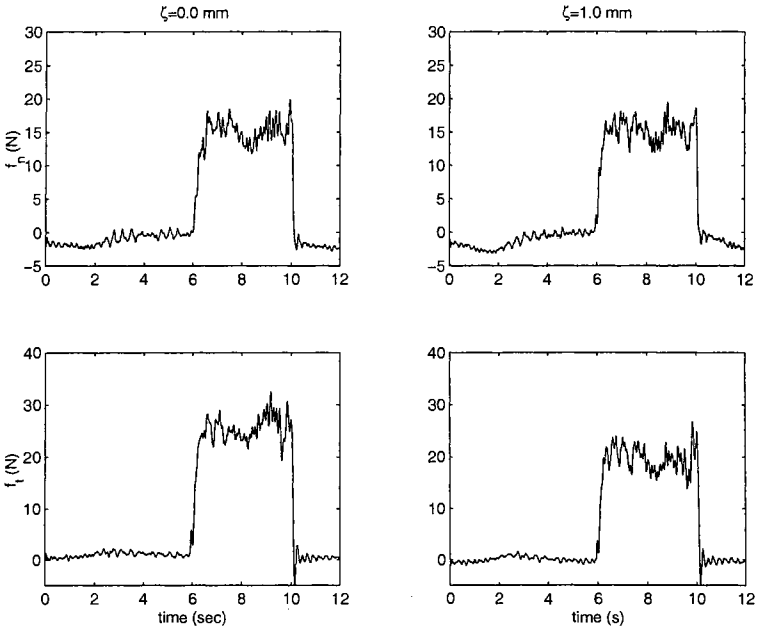

Figure 5.66: Forces of surface finishing mode 2

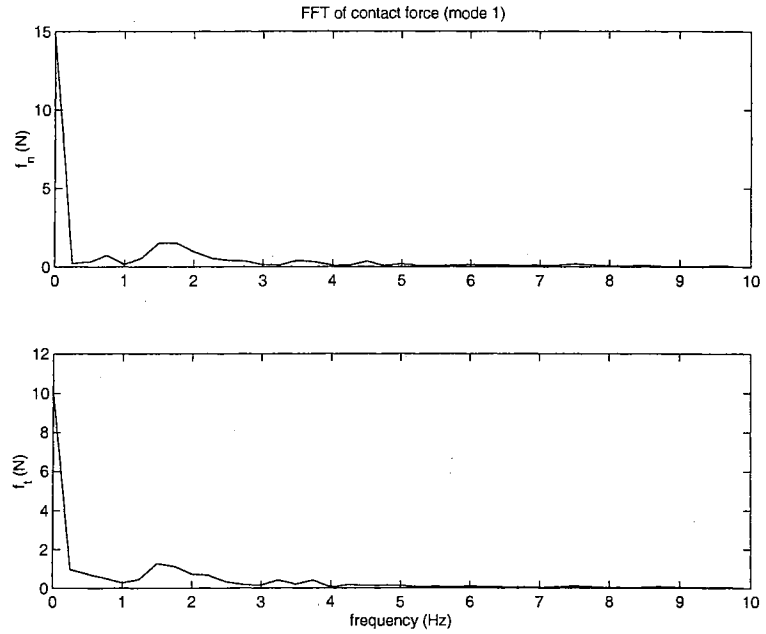

Figure 5.67: FFT of force signal, mode 1

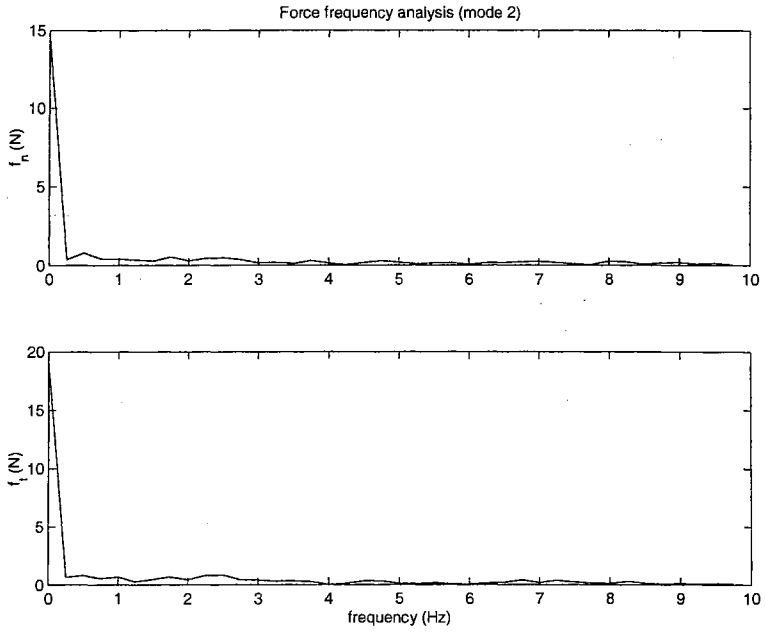

Figure 5.68: FFT of force signal, mode 2

5.5 Investigation of Control Implementation Modes

In this section, mechatronic control implementation modes discussed in section 4.3 are investigated experimentally. Emphasis is given to force signal filtering, filtering delay, and control switching delay. The constraint surface $\phi(x(q))$ is chosen as a rigid straight wall, which is the same as the one described in section 5.2.1. Fig. 5.13 illustrates the two-link robot (top view), constraint surface, and the desired trajectory of the robot (DABC). The desired trajectory is constructed based on the known constraint location. Fig. 5.14 shows the desired position, velocity, and acceleration of the trajectory in joint frame. As shown in Fig. 5.45, uncertainty in the location of the constraint surface leads to the robot impacting the surface with a non-zero normal velocity. Different levels of constraint uncertainty were considered.

Extensive experiments were conducted for the different modes described in the section 4.3. A representative sample of the experimental results is presented. All the experiments use a control sampling frequency of 250 Hz, i.e., $T_s = 4$ milli-seconds. The cycle time to traverse the desired trajectory DABC is 12 seconds. The desired normal force during the constrained motion phase is 45 N. Force signal, its transient at contact, and its FFT are plotted and analyzed.

Experimental results for real-time system operation in mode 1 and mode 2 are shown in Figs. 5.69, 5.70 and 5.71. The top plots in all the three figures correspond to synchronous operation, i.e., mode 1. Two other force sampling frequencies, 1 kHz and 3 kHz, are considered to compare the performance of asynchronous operation to synchronous operation. Choosing higher force sampling rate reduces force signal delay. Fig. 5.70 is the transient force signal of three experiments after the robot end-effector makes contact with the constraint surface. The experimental data shows that choosing higher force sensor sampling frequency results in a better transient force response. Fig. 5.71 shows existence of high fre-

quency components (noise) in the force signal. The amplitude and frequency of the noise is influenced by the force sensor sampling frequency. Higher force sampling frequency introduce more high frequency components in the force signal.

Notice the appearance of frequency components in the range of 30 to 40 Hz in the force signal, as shown in Fig. 5.71. These components are undesirable as the position/force control algorithm in the constrained motion phase uses force feedback signal. Noisy force signals lead to noisy control torque for each motor. To alleviate this noise problem, low pass filtering of the force signal is considered in mode 3. In mode 3, an eight sample running average filter is implemented on the force sensor DSP. The servo DSP control sampling frequency is 250 Hz, the force sampling frequency is 2 kHz. Fig. 5.72 shows the normal force signal, its transient and FFT. Comparison of the force signal in mode 1 and mode 2 with mode 3 shows that the high frequency components no longer appear in the force signal in mode 3. Due to the time delay introduced by the low pass filter, the rise time of the normal force transient response becomes larger for mode 3 when compared with mode 1 and mode 2.

To improve the performance of the discontinuous controller proposed, in mode 4, experiments were conducted with a fixed length sliding average filter on the host processor. Three force sensor sampling frequencies, 250 Hz, 1 kHz, and 3 kHz, were considered in the surface following experiments using mode 4. Fig. 5.73, Fig. 5.74, and Fig. 5.75 show the normal force, its transient, and its FFT, respectively. Comparing with experimental results of mode 2 given by Fig. 5.70, it can be observed that the rise time increases because of the time delay, ΔT_f, introduced by filter. The experimental data also shows that choosing higher force sensor sampling frequency results in better transient response.

When there is uncertainty in the constraint location as shown in Fig. 5.45, there is a tendency of the robot end-effector to bounce on the surface. There is even a possibility of the end-effector leaving the surface. In this case, it is critical that the controller switches to the transition phase control algorithm as quickly as possible after the first impact. The

force signal delay due to filtering may delay the switching of the controller. To solve this problem, mode 5 is implemented. The uncertainty in the constraint surface location is 5 mm, the control sampling frequency is unchanged at 250 Hz, three force sensor sampling frequencies 250 Hz, 1 kHz, and 3 kHz, were considered in the experiments. Fig. 5.76 shows the transient response of the force signal in mode 4 and mode 5. Notice that mode 5 has faster rise time and a smaller impact force peak when compared with mode 4. This is mainly due to the fact that switching delay is reduced in the case of mode 5 using variable length sliding average filter as explained in section 4.3. The experimental data also shows that choosing higher force sensor sampling frequency results in smaller force signal overshoot over the desired force level.

The following observations can be made based on the experimental results. In mode 2, the force sampling delay and algorithm switching delay are reduced by using higher force sampling frequency. Force signal directly from force sensor is noisy and thus cannot be used directly in the control algorithm. In mode 3, a running average filter is implemented on the force sensor DSP. Filtering introduces time delay. In mode 4, force sampling due to filtering can be reduced by increasing the force sensor sampling frequency and running the filter on the host processor. Mode 5 is able to reduce the algorithm switching delay.

This section analyzed the performance of the closed-loop system with different control implementation modes, which mainly differ with respect to the communication between different processors and force signal filtering strategies. An efficient mechatronic design has been considered in developing a robotic experimental platform interacting with an external environment.

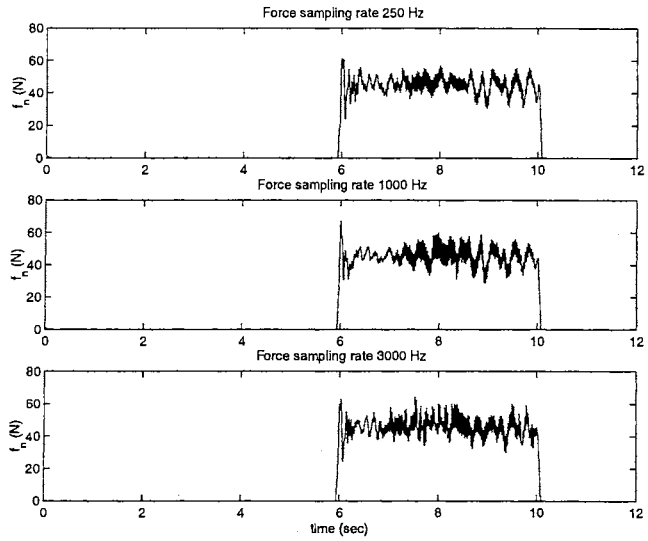

Figure 5.69: Mode 1 and mode 2: normal force

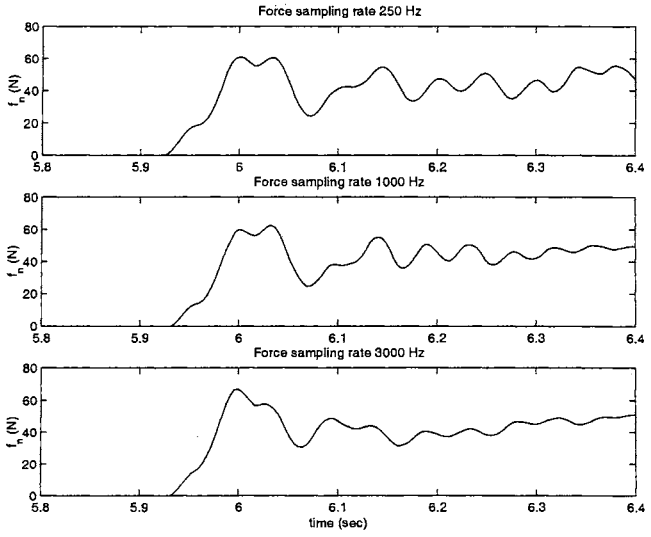

Figure 5.70: Mode 1 and mode 2: transient response of normal force

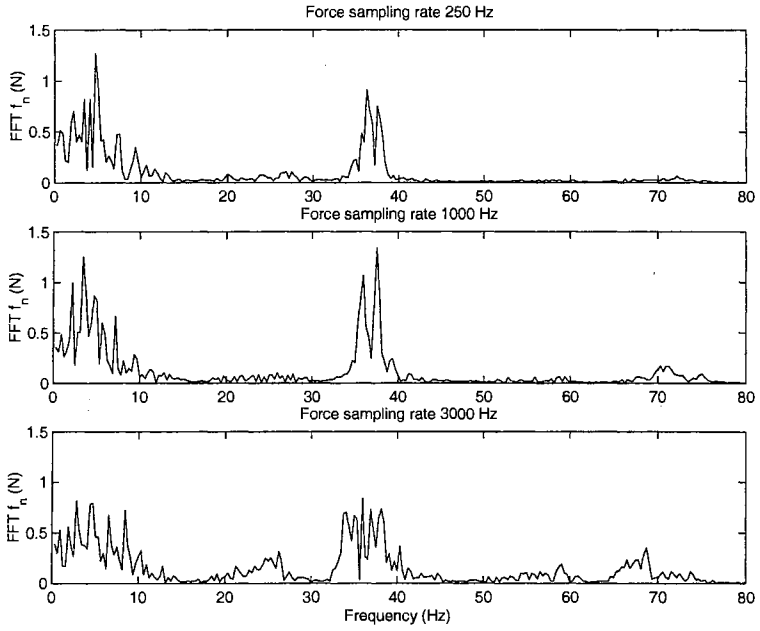

Figure 5.71: Mode 1 and mode 2: FFT of normal force

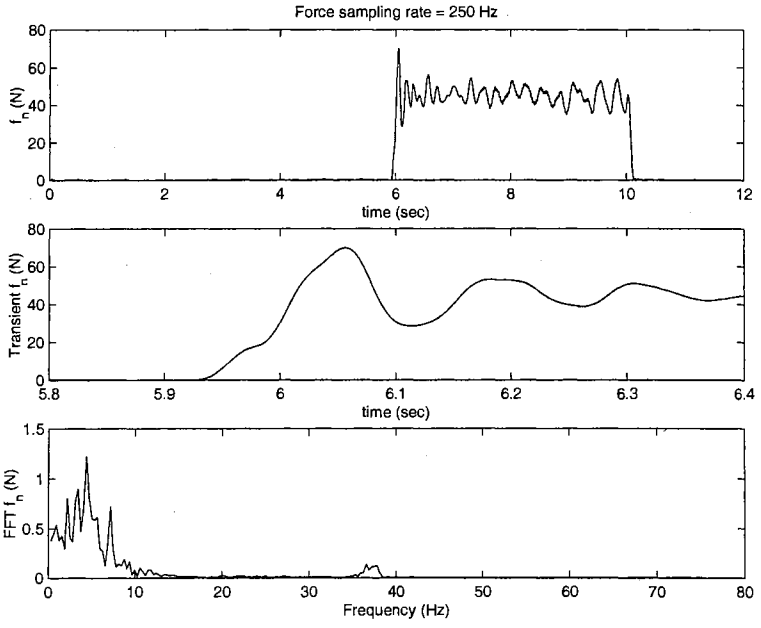

Figure 5.72: Mode 3: transient response and FFT of normal force

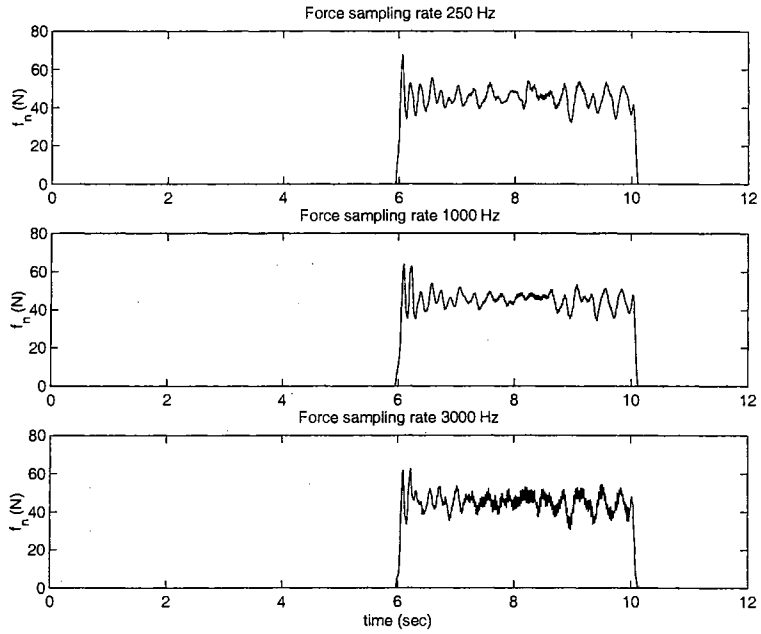

Figure 5.73: Mode 4: normal force

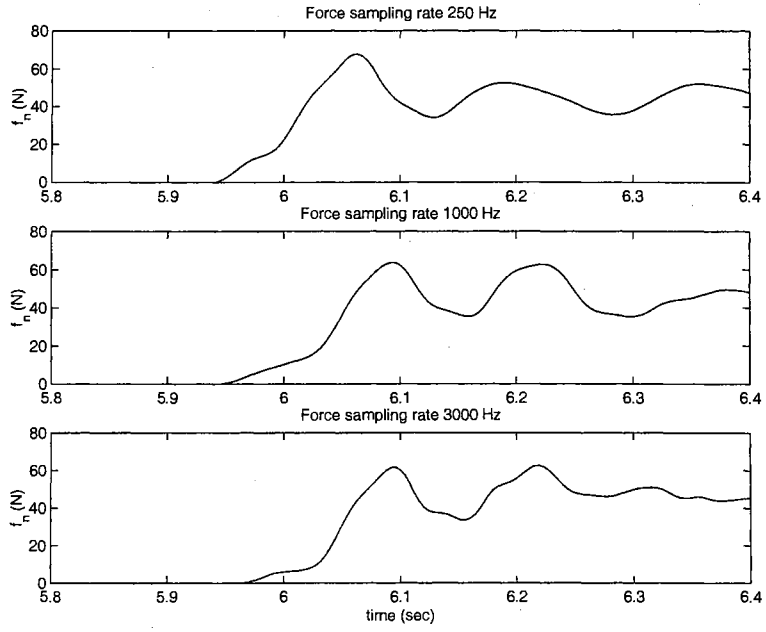

Figure 5.74: Mode 4: transient response of normal force

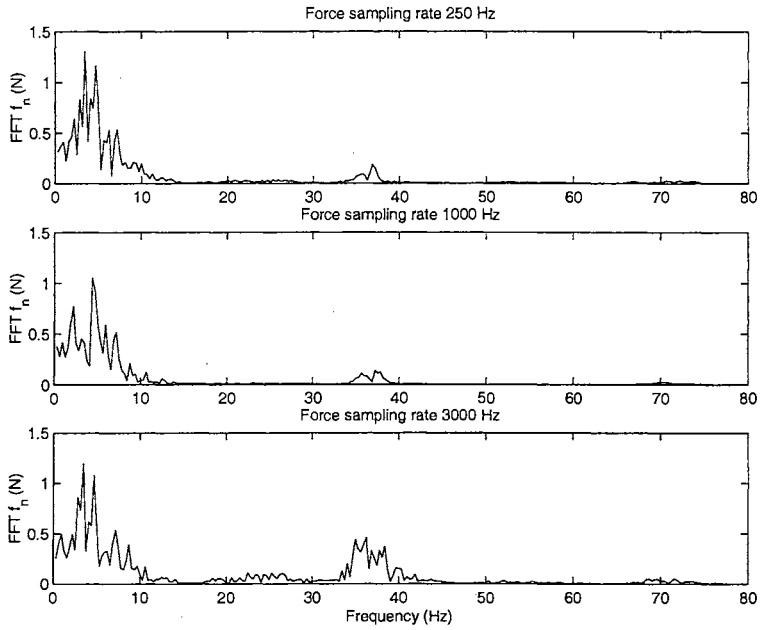

Figure 5.75: Mode 4: FFT of normal force

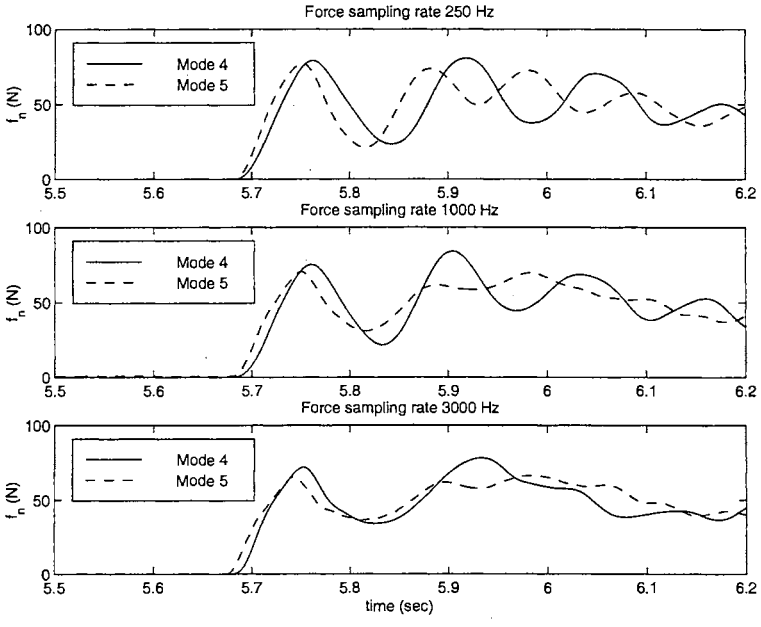

Figure 5.76: Mode 4 and mode 5: transient response of normal force

Chapter 6

Summary and Future Work

6.1 Summary

Modeling, control design, and mechatronic implementation of a class of constrained robots for robotic surface finishing applications is considered in this work. Dynamic model for complete robot tasks involving unconstrained motion, transition motion, and constrained motion has been developed. Control algorithm for each phase is proposed based on the dynamic model developed. A robust discontinuous control scheme is proposed for position and force control for a complete robot task. An event based switching strategy is developed. An open architecture experimental platform has been developed via a mechatronic approach. Extensive experiments have been conducted to investigate the control design and mechatronic implementation.

First, dynamics of a complete robot task involving constrained motion is investigated. The robot motion of a complete task can be divided into three phases, namely, unconstrained motion phase, transition phase, and constrained motion phase. Dynamic model for each phase is given. Robot dynamics of unconstrained motion phase is obtained based on Euler-Lagrange equations. Impact model of a planar manipulator is investigated to facilitate transition control design. The following properties are obtained from the impact model: (1) the contact force impulse lies in the direction normal to the surface at impact

points, or, the normal component is the dominant term in contact impulse at impact points; (2) there exists a threshold of normal impact velocity below which the magnitude of the post-impact velocity normal to the constraint is zero; (3) there is a finite amount of kinetic energy reduction due to impact. In constrained motion phase, geometrical constraint on manipulator is modeled by a unilateral constraint. Based on the assumption that conditions such as feed-rate, material of the environment and robot end-effector, and cutting depth are relatively stable during surface finishing operation (constrained motion phase), a simplified constraint force model is obtained. Constrained robot dynamics is derived based on unilateral constraint model and the simplified constraint force model.

Second, based on the dynamics developed, control algorithm is designed for each phase. A model-based adaptive controller is chosen for unconstrained motion control. Since directly switching from the unconstrained motion phase to the constrained motion phase can lead to non-smooth landing of the robot end-effector onto the surface, a unique robust discontinuous transition control scheme that guarantees stable convergence of the robot end-effector onto the surface is proposed. This transition control algorithm is robust to impact force and does not require the knowledge of an impact model. Constrained motion (surface finishing operation) starts at the end of the transition phase. A new model-based adaptive position/force control algorithm is proposed for the constrained motion phase. Most algorithms in literature assume that the tangential force due to contact is negligible. Since material removal is involved during surface finishing operations, tangential force cannot be neglected. A feed-forward term is introduced into the control algorithm to compensate for tangential forces due to friction and/or material removal. This feed-forward term is based on a widely used empirical model in grinding literature. Stability of the closed-loop system with the proposed controllers is shown for each phase. The controller for a complete task is obtained by integrating the controller for each phase. An event based control switching strategy is developed for complete tasks.

Third, an open architecture experimental platform has been developed for surface fin-

ishing applications with optimal integration of system components via a mechatronic approach. The system consists of a two-link direct drive manipulator, a three-processor real-time controller, a 6-axis force/torque sensor, robotic deburring tool, and work part fixture. Different controller implementation modes are compared by emphasizing data communication between processors and data filtering. An efficient mechatronic design has been considered in developing this experimental robotic surface finishing system.

Finally, extensive experiments have been conducted on the developed experimental platform to test the proposed discontinuous control scheme and compare different controller implementation modes. Experimental results are summarized below:

- The adaptive model-based motion control algorithm used in unconstrained motion phase results in tracking of the desired trajectory in the presence of uncertainties in manipulator model parameters.

- Experimental results show that the position/force controller proposed for constrained motion phase gives position trajectory tracking in the tangential direction as well as desired force regulation in the normal direction. Compared to the hybrid position/force robot control algorithms available in literature, the proposed algorithm with tangential force feed-forward term results in improved trajectory tracking in the tangential direction and better force regulation normal to the constraint. Using the adaptation law designed for grinding friction coefficient ξ, convergence of estimated grinding coefficient to constant values was observed regardless of the initial values of ξ and the value of adaptation gain γ_f.

- The impact model and relevant assumptions were confirmed by the experimental results. The experimental results of free impact experiments show that (1) the normal component in impact force impulse is the dominant term at impact points, (2) when the normal impact velocity is below a threshold the post-impact velocity normal to the surface is zero; (3) there is a finite amount of kinetic energy reduction due to

impact.

- Directly switching from the unconstrained motion phase to constrained motion phase can cause the robot end-effector to bounce on the surface resulting in large impact forces on the robot. The impact force and bounces during the transition phase are affected not only by the constraint uncertainties and tracking errors but also by the approach velocity of the robot end-effector and the rigidity of the constraint. Higher approach velocities will result in longer transition phase. Increasing the rigidity of the constraint surface results in larger impact forces. Furthermore, with direct switch, the peak impact force may not occur at the first impact. Use of the proposed transition control scheme results in significant reduction of peak impact force for various levels of constraint uncertainties, approach velocity, and rigidity of the constraint surface. Extensive comparative experiments show the effectiveness of the proposed transition control scheme.

- Surface finishing experiments show that the relation of tool rotation with respect to the direction of tool travel on the surface affects the tangential force, normal force, and the contact condition. Similar to up and down milling in machining, two working modes are defined (see Figs. 5.61 and 5.62). In mode 2, which involves same direction of tool rotation and tool travel on the surface, a more stable contact between tool and the surface has been observed.

- To obtain optimal overall performance, different control implementation modes were investigated experimentally emphasizing data communication between processors and data filtering. Experimental results show that using higher force sampling frequency via asynchronous integration mode reduces the force sampling delay and algorithm switching delay. The force sampling delay due to filtering can be reduced by increasing the force sensor sampling frequency and running the filter on the host processor. Further, using a sliding average filter with variable length can reduce con-

trol algorithm switching delay. The system using an asynchronous multi-sampling frequency mode is an efficient system design approach for constrained robot control.

In summary, a dynamic model for a complete robot task has been developed. A discontinuous control scheme is proposed for position and force control for a class of constrained robots. Control design and mechatronic system design have been validated by the results of extensive experiments conducted on an open architecture experimental platform developed.

6.2 Future work

In this work, control design and mechatronic implementation were investigated for complete robot tasks. Future work should focus on some potential improvement to the proposed algorithm which are given below.

Design of the transition control algorithm should consider not only normal velocity discontinuity but also tangential velocity discontinuity due to impact. The transition control algorithm proposed results in better transition performance in the normal direction to the constraint surface. Experimental results of robot free impact shows that the velocity of the robot end-effector may be discontinuous in all directions. Even in frictionless case, impact causes discontinuous velocity in both normal and tangential directions to the constraint. The sign of the velocity jump in the tangential direction is predictable based on the mass matrix and the Jacobian of the robot manipulator at impact point. To achieve smooth transition in normal and tangential directions, new transition control algorithm should consider the discontinuity of tangential velocity due to impact.

More investigation is needed on contact force model for different machining conditions. In this work, tangential force feedforward term is obtained from a simplified contact force model. There are many sophisticated grinding force models available in literature. Proper simplification is required for embedding these contact force models in an adaptive position/force control designed for constrained motion phase. Further, a better contact

force model will help to choose suitable desired normal force and initial value of grinding coefficient for a specific machining condition.

The proposed control strategy needs to be implemented on a six DOF (degree of freedom) robot. In this work, dynamic modeling, control design, and stability proof are based on a general case without limitations on manipulators degrees of freedom. However, computation burden, communication rate, data filter, and other control implementation issues are affected by the number of degrees of freedom of the manipulator. The proposed control strategy was implemented successfully on a two-link robot in this work. Implementation needs to be done on a six DOF robot to confirm the effectiveness of the control scheme for industrial applications.

In this work, focus was given to dynamic modeling, control design, and control implementation. Software design and system integration were considered via a mechatronic approach. More work needs to be done on hardware design aspects. Based on the open architecture experimental platform, controller hardware designed for surface finishing system should consider the needs of surface finishing operations in industrial environment. The surface-finishing tool used in this work cannot provide stable power for aggressive cutting files for chamfering and deburring applications. Further, the vane motor of the surface-finishing tool introduces low frequency noise into force signal. A better surface-finishing tool should be designed.

Bibliography

[1] J. D. Meyer, "Applications of robots," in *International encyclopedia of robotics*, pp. 50–56, Wiley Interscience, 1988.

[2] K. Du, X. Huang, M. Wang, and J. Hu, "Assembly robotics research: a survey," *International Journal of Robotics and Automation*, vol. 14, no. 4, pp. 171–183, 1999.

[3] S. Chan and H. Liaw, "Experimental implementation of impedance based control schemes for assembly task," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 29, no. 1, pp. 93–110, 2000.

[4] N. Ramachandran, N. Pande, and N. Ramakrishnan, "Role of deburring in manufacturing: A state-of-the-art survey," *Journal of Materials Processing Technology*, vol. 44, no. 1-2, pp. 1–13, 1994.

[5] R. Komanduri, M. Merchant, and M. Shaw, "Mechining and grinding: A historical review of the classical papers," *Applied Mechanics Reviews*, vol. 46, no. 3, 1993.

[6] S. Arimoto and F. Miyazaki, "Stability and robustness of pd feedback control with gravity compensation for robot manipulator," *ASME, Dynamic Systems and Control Division (Publication)*, vol. 3, pp. 67–72, 1986.

[7] M. Brady, *Robotics Science*. London, England: The MIT Press, 1989.

[8] R. Shoureshi, M. J. Corless, and M. D. Roesler, "Control of industrial manipulators with bounded uncertainties," *Journal of Dynamic Systems, Measurement and Control*, vol. 109, no. 1, pp. 53–59, 1987.

[9] R. Kelly and R. Carelli, "A class of nonlinear pd-type controllers for robot manipulators," *Journal of Robotic Systems*, vol. 13, no. 12, pp. 793–802, 1996.

[10] R. Kelly, "Global positioning of robot manipulators via pd control plus a class on nonlinear integral actions," *IEEE Transactions on Automatic Control*, vol. 43, no. 7, pp. 934–938, 1998.

[11] T. P. Bucklaew and C. S. Liu, "New nonlinear gain structure for pd-type controllers in robotic applications," *Journal of Robotic Systems*, vol. 16, no. 11, pp. 627–649, 1999.

[12] T. J. Tarn, "Nonlinear feedback in robot arm control," in *Proceedings of the IEEE Conference on Decision and Control Including The Symposium on Adaptive Processes*, vol. 2, pp. 736–751, 1984.

[13] M. W. Spong, "Modeling and control of elastic joint robots," *Journal of Dynamic Systems, Measurement and Control*, vol. 109, no. 4, pp. 310–319, 1987.

[14] R. G. Menon and D. P. Garg, "Exact linearization of nonlinear manipulator models using adaptive control," in *Proceedings of the American Control Conference*, vol. 2, pp. 924–929, June 1988.

[15] T. Tarn, A. Bejczy, and X. Yun, "New nonlinear control algorithms for multiple robot arms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 5, pp. 571–583, 1988.

[16] E. Freund, "Fast nonlinear control with arbitrary pole-placement for industrial robots and manipulators," *International Journal of Robotics Reserach*, vol. 1, no. 1, pp. 65–78, 1982.

[17] S. J. Fournier and R. J. Schilling, "Decoupling of a two-axis robotic manipulator using nonlinear state feedback: A case study," *International Journal of Robotics Research*, vol. 3, no. 3, pp. 76–86, 1984.

[18] J. Jeon and C. Lee, "Simplification of manipulator dynamic model for nonlinear decoupled control," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 1, pp. 592–597, December 1988.

[19] J. K. Mills and A. A. Goldenberg, "Robust control of a robotic manipulator with joint variable feedback," *ASME Dynamic Systems and Control Division (Publication)*, vol. 3, pp. 43–49, 1986.

[20] J. M. Hollerbach, "Recursive lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 10, no. 11, pp. 730–736, 1980.

[21] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line computational scheme for mechanical manipulators," *Journal of Dynamic Systems, Measurement and Control*, vol. 102, no. 2, pp. 69–76, 1980.

[22] J. J. E. Slotine and W. Li, "Adaptive manipulator control: A case study," *IEEE Transactions on Automatic Control*, vol. 33, no. 11, pp. 995–1003, 1988.

[23] G. Niemeyer and J. J. E. Slotine, "Performance in adaptive manipulator control," *International Journal of Robotics Research*, vol. 10, no. 2, pp. 149–161, 1991.

[24] J. J. Craig, *Adaptive Control of Mechanical Manipulator*. Reading, Mass. : Addison-Wesley, 1988.

[25] R. Ortega and M. W. Spong, "Adaptive motion control of rigid robots: A tutorial," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 1575–1584, December 1989.

[26] J. S. Reed and P. A. Ioannou, "Instability analysis and robust adaptive control of robotic manipulators," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 381–386, 1989.

[27] G. Tao, "On robust adaptive control of robot manipulators," *Automatica*, vol. 28, no. 4, pp. 803–807, 1992.

[28] H. Berghuis, R. Ortega, and H. Nijmeijer, "A robust adaptive controller for robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1876–1881, May 1992.

[29] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems," in *Proceedings of the IEEE Conference on Decision and Control Including The Symposium on Adaptive Processes*, pp. 1064–1069, 1984.

[30] S. Arimoto, "Robustness of learning control for robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1528–1533, May 1990.

[31] D. Whitney, "Historical perspective and state of art in robot force control," *International Journal of Robotics Research*, vol. 6, no. 1, pp. 3–14, 1987.

[32] R. Paul and B. Shimano, "Compliance and control," in *Proceedings of Japan Automatic Control Conference*, pp. 694–699, 1976.

[33] A. K. Jaura, N. Krouglicof, and M. Osman, "Using a logic branching weighted algorithm to train robots for splined shaft-hole assembly," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 29, no. 3, pp. 277–283, 1999.

[34] G. Zeng and A. Hemami, "An overview of robot force control," *Robotica*, vol. 15, no. 5, pp. 473–482, 1997.

[35] R. Roberts, R. Paul, and S. Kim, "The effect of wrist force sensor stiffness on the control of robot manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 417–426, 1985.

[36] M. Spong and J. Viyasagar, *Robot Dynamics and Control.* John & Sons, Inc., 1989.

[37] J. K. Salisbury, "Active stiffness control of a manipulator in cartesian coordinates," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 1, pp. 95–100, December 1980.

[38] J. Salisbury and J. Craig, "Articulated hands: force control and kinematic issues," *International Journal of Robotics Research*, vol. 1, no. 1, pp. 4–17, 1982.

[39] G. P. Starr, "Cartesian stiffness control of the jpl/stanford/salisbury hand," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 636–647, 1988.

[40] Y. T. Lee, J. H. Kim, W. K. Chung, and Y. Youm, "Stiffness control of the coupled tendon-drive robot hand," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 710–715, October 1993.

[41] R. A. Freeman, "Feedforward stiffness control of overconstrained mechanisms/robotic linkage systems," *ASME Dynamic Systems and Control Division (Publication)*, vol. 11, pp. 251–269, November 1988.

[42] B. J. Yi, R. A. Freeman, and D. Tesar, "Open-loop stiffness control of overconstrained mechanisms/robotic linkage systems," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 1340–1345, 1989.

[43] S. Chiaverini, B. Siciliano, and L. Villani, "Parallel force/position control with stiffness adaptation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1136–1141, April 1997.

[44] M. Pelletier and L. Daneshmend, "Dynamic performance of robot contact tasks using adaptive damping control," in *ASME Design Engineering Division (Publication)*, vol. 18, pp. 237–243, September 1989.

[45] P. K. Bose, "Learning stiffness control for compliant motion," in *Proceedings of the Sixth European Conference on Artificial Intelligence*, pp. 115–126, 1984.

[46] D. Jeon and M. Tomizuka, "Stiffness identification method for self-tuning adaptive force control," in *Proceedings of the Japan - USA Symposium on Flexible Automation*, pp. 1245–1250, July 1992.

[47] B. H. Kim, N. Y. Chong, S. R. Oh, I. H. Suh, and Y. J. Cho, "Intelligent compliance control for robot manipulators using adaptive stiffness characteristics," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2134–2139, May 1999.

[48] H. Kazerooni, T. Sheridan, and P. Houpt, "Robust compliant motion for manipulators," *IEEE Journal of Robotics and Automation*, vol. 2, no. 2, pp. 83–92, 1986.

[49] M. Mason, "Compliance and force control for computer controller manipulator," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 11, no. 6, pp. 418–432, 1981.

[50] C. C. de Wit, B. Siciliano, and G. Bastin, *Theory of robot control*. Great Britain: Springer-Verlag London, 1996.

[51] S. Chiaverini, B. Siciliano, and L. Villani, "A survey of robot interaction control schemes with experimental comparison," *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 3, pp. 273–285, 1999.

[52] N. Hogan, "Impedance control of industrial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 1, no. 1, pp. 97–113, 1984.

[53] N. Hogan, "Impedance control: An approach to manipulation: Part 1, 2, 3," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 107, no. 1, pp. 1–24, 1985.

[54] A. A. Goldenberg, "Implementation of force and impedance control in robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1626–1632, 1988.

[55] J. K. Mills and G. J. Liu, "Robotic manipulator impedance control of generalized contact force and position," in *International Workshop on Intelligent Robot and Systems*, vol. 2, pp. 1103–1108, November 1991.

[56] H. Kazerooni and M. G. Her, "Robotic deburring of two-dimensional parts with unknown geometry," in *Proceedings of IEEE International Symposium on Intelligent Control*, vol. 1, pp. 459–464, 1989.

[57] W. McCormick and H. Schwartz, "Investigation of impedance control for robot manipulators," *International Journal of Robotics Research*, vol. 12, no. 5, pp. 473–489, 1993.

[58] R. Colbaugh and M. Jamshidi, "Robot manipulator control for hazardous waste-handling applications," *Journal of Robotic Systems*, vol. 9, no. 2, pp. 215–250, 1992.

[59] J. M. Schimmels, "Linear space of admittance control laws that guarantees force-assembly with friction," *IEEE Transaction on Robotics and Automation*, vol. 13, no. 5, pp. 656–667, 1997.

[60] B. Heinrichs and N. Sepehri, "A limitation of position based impedance control in static force regulation: theory and experiments," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2165–2170, May 1999.

[61] D. Matko, R. Kamnik, and T. Bajd, "Adaptive impedance control of an industrial robot," in *Proceedings of the IEEE International Symposium on Industrial Electronics*, vol. 1, pp. 129–133, 1999.

[62] Z. M. and R. G. Botev, "A model reference approach to adaptive impedance control of robot manipulators," in *Proceedings of the International Conference on Intelligent Robots and Systems*, vol. 2, pp. 727–733, July 1993.

[63] R. Colbaugh, H. Seraji, and K. Glass, "Direct adaptive impedance control of manipulators," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 3, pp. 2410–2415, 1991.

[64] W. S. Lu and Q. H. Meng, "Impedance control with adaptation for robotic manipulations," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 408–415, 1991.

[65] N. Hogan, "Impedance control applied to automated deburring," *Adaptive and Learning Systems: Theory and Applications*, pp. 359–366, 1986.

[66] A. Hace, S. Uran, K. Jezernik, and B. Curk, "Robust sliding mode based impedance control," in *Proceedings of the IEEE International Conference on Intelligent Engineering Systems*, pp. 77–82, September 1997.

[67] B. Fernandez, G. W. Bae, and L. J. Everett, "Control of robot manipulator through robust sliding linearization," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 124–129, 1990.

[68] Z. Lu and A. Goldenberg, "Robust impedance control and force regulation. theory and experiments," *International Journal of Robotics Research*, vol. 14, no. 3, pp. 225–254, 1995.

[69] A. Hace, K. Jezernik, and S. Uran, "Robust impedance control," in *Proceedings of the IEEE Conference on Control Applications*, vol. 1, pp. 583–587, September 1998.

[70] C. Cheah and D. Wang, "Learning impedance control for robotic manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 2150–2155, May 1995.

[71] D. Wang and C. Cheah, "Iterative learning control scheme for impedance control of robotic manipulators," *International Journal of Robotics Research*, vol. 17, no. 10, pp. 1091–1104, 1998.

[72] S. Jung and T. Hsia, "Adaptive force tracking impedance control of robot for cutting non-homogeneous workpiece," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1800–1805, May 1999.

[73] T. Murakami, R. Nakamura, F. Yu, and K. Ohnishi, "Force sensorless impedance control by disturbance observer," in *Power Conversion Conference*, pp. 352–357, July 1993.

[74] S. Jung and T. Hsia, "Neural network impedance force control of robot manipulator," *IEEE Transaction on Industrial Electronics*, vol. 45, no. 3, pp. 451–461, 1998.

[75] M. Cohen and T. Flash, "Learning impedance parameters for robot control using associative search network," *IEEE Transaction on Robotics and Automation*, vol. 7, no. 3, pp. 382–390, 1991.

[76] T. Tsuji, K. Ito, and P. Morasso, "Learning of robot arm impedance in operational space using neural networks," in *Proceedings of the International Joint Conference on Neural Networks*, vol. 1, pp. 635–638, 1993.

[77] H. Seraji, "Adaptive admittance control: an approach to explicit force control in compliant motion," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, pp. 2705–2710, May 1994.

[78] R. Colbaugh, K. Glass, and S. H., "Decentralized adaptive control of manipulators: theory and experiments," in *Proceedings of the Conference on Decision and Control*, vol. 1, pp. 153–158, December 1993.

[79] S. Jung, T. Hsia, and R. Bonitz, "On robust impedance force control of robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2057–2062, April 1997.

[80] V. Gullapalli and A. G. Barto, "Learning admittance mappings for force-guided assembly," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2633–2638, May 1994.

[81] R. Volpe and P. Khosla, "A theoretical and experimental investigation of explicit force control strategies for manipulators," *IEEE Transactions on Automatic Control*, vol. 38, no. 11, pp. 1634–1650, 1993.

[82] J. Maples and J. Becker, "Experiments in force control of robotic manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 695–702, 1986.

[83] H. Ishikawa, C. Sawada, K. Kawase, and M. Takata, "Stable compliance control and its implementation for a 6 dof manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 98–103, 1989.

[84] H. Faessler, "Concurrent motion and force control for manipulators constrained by stiff contact," in *Proceedings of the IEEE International Workshop on Intelligent Motion Control*, vol. 1, pp. 321–329, 1990.

[85] K. Wedeward, R. Colbaugh, and A. Engelmann, "Adaptive explicit force control of position-controlled manipulators," *Journal of Robotic Systems*, vol. 13, no. 9, pp. 603–618, 1996.

[86] M. Minami, T. Asakura, L. Dong, and Y. Huang, "Position control and explicit force control of constrained motions of a manipulator for accurate grinding tasks," *Advanced Robotics*, vol. 11, no. 3, pp. 285–300, 1997.

[87] R. Volpe and P. Khosla, "Equivalence of second-order impedance control and proportional gain explicit force control," *International Journal of Robotics Research*, vol. 14, no. 6, pp. 574–589, 1995.

[88] P. Rocco, G. Ferretti, and G. Magnani, "Implicit force control for industrial robots in contact with stiff surfaces," *Automatica*, vol. 33, no. 11, pp. 2041–2047, 1997.

[89] J. Borowiec and A. Tzes, "Frequency-shaped implicit force control of flexible link manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 913–918, May 1995.

[90] S. Chiaverini, B. Siciliano, and L. Villani, "Force and position tracking: parallel control with stiffness adaptation," *IEEE Control Systems Magazine*, vol. 18, no. 1, pp. 27–33, 1998.

[91] S. Chiaverini and L. Sciavicco, "Parallel approach to force/position control of robotic manipulators," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 4, pp. 361–373, 1993.

[92] J. Craig and M. Raibert, "A systematic method of hybrid position/force control of a manipulator," in *Proceedings of the IEEE Computer Software and Applications Conference*, vol. 1, pp. 446–451, 1979.

[93] H. Lipkin and J. Duffy, "Hybrid twist and wrench control for a robotic manipulator," *Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 110, no. 2, pp. 138–144, 1988.

[94] T. Yoshikawa, "Dynamic hybrid position/force control of robot manipulators - description of hand constraint and calculation of joint driving force," *IEEE Journal of Robotics and Automation*, vol. 3, no. 5, pp. 386–392, 1987.

[95] T. Yoshikawa, T. Sugie, and M. Tanaka, "Dynamic hybrid position/force control of robot manipulators: Controller design and experiment," *IEEE Journal of Robotics and Automation*, vol. 4, no. 6, pp. 699–705, 1988.

[96] V. Perdereau and M. Drouin, "New scheme for hybrid force-position control," *Robotica*, vol. 11, no. 5, pp. 453–464, 1993.

[97] J. E. Slotine and W. Li, "Adaptive strategies in constrained manipulation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 595–601, 1987.

[98] S. Arimoto, Y. H. Liu, and T. Naniwa, "Model-based adaptive hybrid control for geometrically constrained robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 618–623, May 1993.

[99] S. Arimoto and T. Naniwa, "Learning control for robot tasks under geometric endpoint constraint," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1914–1919, May 1992.

[100] L. L. Whitcomb, S. Arimoto, and T. Naniwa, "Adaptive model-based hybrid control of geometrically constrained robot arms," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 1, pp. 105–116, 1997.

[101] N. M'Sirdi and A. Benali, "Application of passive system approach for adaptive hybrid force-position control," in *International Federation of Automatic Control Symposia Series, Proceedings of a Triennial World Congress*, vol. 8, pp. 125–130, July 1992.

[102] S. Arimoto, *Control Theory of non-linear mechanical systems – a passivity-based and circuit – theoretic approach*. New York: Oxford University Press Inc., 1996.

[103] P. R. Pagilla and B. Yu, "Adaptive motion and force control of robots performing a complete task," in *Proceedings of the ASME Design Engineering Technical Conference*, September 1999.

[104] L. M. Sweet and M. C. Good, "Redefinition of the robot motion-control problem," *IEEE Control Systems Magazine*, vol. 5, no. 3, pp. 18–25, 1985.

[105] K. Guglielmo and N. Sadegh, "Theory and implementation of a hybrid learning force control scheme," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 659–664, May 1993.

[106] H. Yu, L. Seneviratne, and S. Earales, "Adaptive force/motion control of constrained robot manipulators," in *Proceedings of the IEEE Conference on Control Applications*, vol. 2, pp. 505–510, September 1993.

[107] D. Jeon and M. Tomizuka, "Learning hybrid force and position control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 4, pp. 423–431, 1993.

[108] P. Lucibello, "Learning algorithm for hybrid force control of robot arms," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 654–658, May 1993.

[109] P. Lucibello, "Learning algorithm for improved hybrid force control of robot arms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 28, no. 2, pp. 241–244, 1998.

[110] S. Kawamura, W. Choi, and T. Morizono, "Study of learning hybrid force and position control for robot manipulators," in *Proceedings of the Japan - USA Symposium on Flexible Automation*, vol. 2, pp. 1295–1300, July 1992.

[111] R. Maass, V. Zahn, M. Dapper, and Eckmiller, "Hard contact surface tracking for industrial manipulators with (sr) position based force control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1481–1486, May 1999.

[112] F. Y. Hsu and L. C. Fu, "Adaptive fuzzy hybrid force/position control for robot manipulators following contours of an uncertain object," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2232–2237, April 1996.

[113] J. S. Liu and S. L. Chen, "Robust hybrid control of constrained robot manipulators via decomposed equations," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 23, no. 1, pp. 45–70, 1998.

[114] K. P. Jankowski and H. A. ElMaraghy, "Robust hybrid position/force control of redundant robots," *Robotics and Autonomous Systems*, vol. 27, no. 3, pp. 111–127, 1999.

[115] C. Natale, B. Siciliano, and L. Villani, "Robust hybrid force/position control with experiments on an industrial robot," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 956–960, September 1999.

[116] R. J. Anderson and M. W. Spong, "Hybrid impedance control of robotic manipulators," *IEEE Journal of Robotics and Automation*, vol. 4, no. 5, pp. 549–556, 1988.

[117] G. J. Liu and A. A. Goldenberg, "Robust hybrid impedance control of robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 287–292, April 1991.

[118] R. Bickel, M. Tomizuka, and W. Chung, "Hybrid impedance control in constraint coordinates using a disturbance observer," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1974–1979, December 1996.

[119] K. Guglielmo and N. Sadegh, "Implementing a hybrid learning force control scheme," *IEEE Control Systems Magazine*, vol. 14, no. 1, pp. 72–79, 1994.

[120] R. Carelli and R. Kelly, "Adaptive hybrid impedance/force controller for robot manipulators," in *International Federation of Automatic Control Symposia Series, Proceedings of a Triennial World Congress*, vol. 5, pp. 255–261, August 1990.

[121] T. Yoshikawa, "Force control of robot manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 220–226, 2000.

[122] S. H. Drake, "Using compliance instead of sensory feedback for high speed robot assembly," in *SME Technical Paper (series)*, p. 14p, October-November 1977.

[123] P. N. Akella, R. Siegwart, and M. R. Cutkosky, "Manipulation with soft fingers: Contact force control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 652–657, April 1991.

[124] P. N. Akella and M. R. Cutkosky, "Contact transition control with semiactive soft fingertips," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 6, pp. 859–867, 1995.

[125] D. E. Whitney, "A survey of manipulation and assembly: Development of the field and open research issues," in *Robotics Science*, pp. 291–348, MIT Press, 1989.

[126] J. M. Hyde and M. R. Cutkosky, "Controlling contact transition," *IEEE Control Systems Magazine*, vol. 14, no. 1, pp. 25–30, 1994.

[127] I. D. Walker, "The use of kinematic redundancy in reducing impact and contact effects in manipulation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 434–439, 1990.

[128] K. A. Morgansen and F. G. Pin, "Impact mitigation using kinematic constrains and the full space parameterization method," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1897–1902, April 1996.

[129] K. Kitagaki and M. Uchiyama, "Optimal approach velocity of end-effector to the environment," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1928–1934, May 1992.

[130] K. Kitagaki and M. Uchiyama, "Optimal approach velocity of an end-effector to the environment," *Advanced Robotics*, vol. 8, no. 2, pp. 123–137, 1994.

[131] B. Allotta and G. Buttazzo, "Impact handling by proximity and force sensing," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2032–2037, May 1992.

[132] Y. Li, "Sensor-based robot transition control strategy," *International Journal of Robotics Research*, vol. 15, no. 2, pp. 128–136, 1996.

[133] R. Brach, *Mechanical Impact Dynamics: Rigid Body Collisions*. NY: John Wiley & Sons, 1991.

[134] W. Goldsmith, *Impact: The Theory and Physical Behaviour of Colliding Solids*. Edward Arnold Publishers, 1960.

[135] V. Kozlov and D. Treschev, "Billiards: A genetric introduction to the dynamics of systems with impacts," *AMS Translations of Mathematical Monographs*, 1991.

[136] C. E. Smith, "Predicting rebounds using rigid body dynamics," *ASME Journal of Applied Mechanics*, vol. 58, no. 3, pp. 754–758, 1991.

[137] W. J. Stronge, "Two-dimensional rigid-body collisions with friction," *Journal of Applied Mechanics*, vol. 60, no. 2, pp. 564–566, 1993.

[138] C. E. Smith and P. P. Liu, "Coefficients of restitution," *ASME Journal of Applied Mechanics*, vol. 59, no. 4, pp. 963–969, 1992.

[139] P. R. Pagilla and B. Yu, "An experimental study of planar impact of a robot manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2001. submitted.

[140] O. Khatib and J. Burdick, "Motion and force control of robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1381–1386, 1986.

[141] N. Hogan, "Stable execution of contact tasks using impedance control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1047–1054, 1987.

[142] H. Kazerooni, B. Waibel, and S. Kim, "On the stability of robot compliant motion control. theory and experiments," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 112, no. 3, pp. 417–426, 1990.

[143] H. P. Qian and J. De Schutter, "Introducing active linear and nonlinear damping to enable stable high gain force control in case of stiff contact," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1374–1379, May 1992.

[144] S. A. Schneider and R. H. J. Cannon, "Object impedance control for cooperative manipulation: Theory and experimental results," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 383–395, 1992.

[145] R. Vossoughi and M. Donath, "Robot hand impedance control in the presence of mechanical nonlinearities," *ASME, Production Engineering Division (Publication)*, vol. 15, pp. 67–77, 1985.

[146] N. Sarkar, X. Yun, and R. Ellis, "Live-constraint-based control for contact transitions," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 5, pp. 743–754, 1998.

[147] N. Sarkar and X. Yun, "Design of a continuous controller for contact transition task based on impulsive constraint analysis," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 2000–2005, April 1996.

[148] S. Djerassi and S. P. Faibish, "Three phase motion control of robotic arms," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3695–370, April 1996.

[149] J. K. Mills, "Manipulator transition to and from contact tasks: A discontinuous control approach," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 440–446, May 1990.

[150] J. K. Mills and D. M. Lokhorst, "Experimental results in manipulator contact task control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1645–1651, April 1991.

[151] J. K. Mills and D. M. Lokhorst, "Stability and control of robotic manipulators during contact/ noncontact task transition," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 3, pp. 335–345, 1993.

[152] S. Gutman, "Uncertain and dynamical systems - a lyapunov minmax approach," *IEEE Transactions on Automatic Control*, vol. 24, no. 3, pp. 437–443, 1979.

[153] R. Volpe and P. Khosla, "Experimental verification of a strategy for impact control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1854–1860, 1991.

[154] P. Akella, V. Parra-Vega, S. Arimoto, and K. Tanie, "Discontinuous model-based adaptive control for robots executing free and constrained tasks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3000–3007, May 1994.

[155] P. R. Pagilla and B. Yu, "A stable transition controller for constrained robots," *IEEE/ASME Transactions on Mechatronics*, 2000. accepted for publication.

[156] G. Marth, T. Tarn, and A. Beljczy, "Stable phase transition control for robot arm motion," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 355–362, 1993.

[157] T. J. Tarn, Y. Wu, N. Xi, and A. Isidori, "Force regulation and contact transition control," *IEEE Control Systems Magazine*, vol. 16, no. 1, pp. 32–40, 1996.

[158] Y. Wu, T. J. Tarn, N. Xi, and A. Isidori, "On robust impact control via positive acceleration feedback for robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1891–1896, April 1996.

[159] Y. Xu, J. M. Hollerbach, and D. Ma, "Force and contact transient control using nonlinear pd control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 924–930, May 1994.

[160] L. Cai and A. A. Goldenberg, "Robust control of position and force for robot manipulators involving both free and constrained motion," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 3, pp. 1943–1948, December 1990.

[161] P. R. Pagilla and M. Tomizuka, "Control of mechanical systems subject to unilateral constraint," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, pp. 4311–4316, December 1995.

[162] P. R. Pagilla and M. Tomizuka, "Contact transition control of nonlinear mechanical systems subject to a unilateral constrain," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 119, pp. 749–759, December 1997.

[163] P. R. Pagilla, "Control of contact problem in constrained euler-lagrangian systems," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1249–1250, 1997.

[164] P. R. Pagilla, *Control of Constrained Nonlinear Mechanical Systems: Applications to Robot Manipulators*. Ph.d. thesis, University of Califernia, Berkeley CA, May 1996.

[165] P. R. Pagilla and B. Yu, "Design and experimental evaluation of a stable transition controller for geometrically constrained robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1308–1313, April 2000.

[166] J. Koelsch, "Banish manual deburring," *Manufacturing Engineering*, vol. 105, no. 1, pp. 71–75, 1990.

[167] E. Erickson, "Automated robotic deburring produces quality components," *Automation*, pp. 50–51, 1991.

[168] K. Stouffer, R. Russel, R. Archacki, T. Engel, R. Dansereau, and A.Grot, "Advanced deburring and chamfering system (adacs): Final report," Technical Report NISTIR 5915, NIST, 1997.

[169] D. A. Dornfeld, "Intelligent deburring of precision components," in *Proceedings of the International Conference on Industrial Electronics, Control, Instrumentation, and Automation*, vol. 2, pp. 953–960, 1992.

[170] G. Bone, M. Elbestawi, R. Lingarkar, and L. Liu, "Force control of robotic deburring," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 113, no. 3, pp. 395–400, 1991.

[171] H. Tonshoff, J. Peters, I. Inasaki, and T. Paul, "Modelling and simulation of grinding processes," *CIRP Annals*, vol. 41, no. 2, pp. 677–688, 1992.

[172] T. M. Stepien, L. M. Sweet, M. C. Good, and M. Tomizuka, "Control of tool/workpiece contact force with application to robotic deburring," *IEEE Journal of Robotics and Automation*, vol. 3, no. 1, pp. 7–18, 1987.

[173] J. Radford, *Production Engineering Technology*. MacMillan Press Ltd., 3rd ed., 1980.

[174] L. Cai and A. Goldenberg, "Position and force control approach to automatic deburring by a robot manipulator," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 784 –789, 1989.

[175] R. Hollowell and R. Guile, "Analysis of robotic chamfering and deburring," *ASME Dynamic Systems and Control Division (Publication)*, vol. 6, pp. 73–79, 1987.

[176] G. Duelen, H. Munch, D. Surdilovic, and J. Timm, "Automated force control schemes for robotic deburring: Development and experimental evaluation," in *Proceedings of the International Conference on Decision and Control*, vol. 2, pp. 912 –917, 1991.

[177] C. B. Selleck and C. S. Loucks, "A system for automated edge finishing," in *Proceedings of the IEEE International Conference on Systems Engineering*, pp. 423–429, August 1990.

[178] G. Duelen, H. Munch, and Y. Zhang, "A comparison of control strategies for force constrained cooperating robots," in *Proceedings of the International Conference on Industrial Electronics, Control, Instrumentation, and Automation*, vol. 1, pp. 708–709, 1992.

[179] G. Song and L. Cai, "Robust position/force control of robot manipulators during constrained tasks," *Control Theory and Applications*, vol. 145, no. 4, pp. 427–433, 1998.

[180] L. Guo, A. Schone, and X. Ding, "A comprehensive approach to nonlinear adaptive control and its application to form grinding processes," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1267 –1272, 1992.

[181] G. Bone, M. Elbestawi, R. Lingarkar, and L. Liu, "Force control for robotic deburring," *Journal of Dynamic Systems, Measurement and Control*, vol. 113, no. 3, pp. 395–400, 1991.

[182] G. M. Bone and M. A. Elbestawi, "Sensing and control for automated robotic edge deburring," *IEEE Transactions on Industrial Electronics*, vol. 41, no. 2, pp. 137–146, 1994.

[183] M. H. Liu, "Robotic deburring based on fuzzy force control," in *Proceedings of the lEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 782 –789, 1992.

[184] P. R. Pagilla and B. Yu, "Robotic surface finishing processes: modeling, control, and experiments," *ASME Journal of Dynamic Systems, Measurement and Control*, 1999. in review for publication.

[185] P. R. Pagilla and B. Yu, "Adaptive control of robotic surface finishing processes," in *American Control Conference*, 2001. submitted.

[186] G. Ferretti, G. Magnani, and P. Rocco, "Toward the implementation of hybrid position/force control in industrial robots," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 6, pp. 838–845, 1997.

[187] R. Andersson, "Systems design for robot control with a scalar supercomputer," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1210–1215, May 1990.

[188] D. M. Auslander, "What is mechatronics?," *IEEE/ASME Transactions on Mechatronics*, vol. 1, no. 1, pp. 5–9, 1996.

[189] R. Comerford, "Mecha... what?," *IEEE Spectrum*, vol. 31, no. 8, pp. 46–49, 1994.

[190] H. Hashimoto, "Intelligent mechatronics," in *Proceedings of Industrial Electronics Conference*, vol. 1, pp. 445–450, November 1993.

[191] K. Nilsson, R. Johansson, G. Bolmsjo, and G. Olsson, "Experimental industrial robotics," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, pp. 3904–3909, December 1996.

[192] P. R. Pagilla and B. Yu, "Mechatronic desing and control of a robot system interacting with an external environment," in *Proceedings of the 7th Mechatronics Forum*

International Conference, September 2000. also in review for publication in the *International Journal of Mechatronics*.

[193] T. Tawakoli, *High Efficiency Deep Grinding*. Mechanical Engineering Publications Ltd., 1993.

[194] R. King and R. Hahn, *Handbook of Modern Grinding Technology*. Chapman and Hall, 1986.

[195] Y. Ono, "Direct-drive motor control system and its features," *Advanced Robotics*, vol. 6, no. 2, pp. 243–253, 1992.

[196] S. Sastry and M. Bodson, *Adaptive control: stability, convergence, and robustness*. Prentice-Hall, Inc., 1989.

Appendix A

Experiments of Motion Control for Unconstrained Motion Phase

Desired trajectory: straight constraint	In Cartesian space: Fig. A.1 In joint space: Fig. A.2		
Adaptation gain matrix $\Gamma^{-\top}$	Initial parameter vector $\beta_0^\top$	Estimates of parameters	Tracking error
$\mathrm{diag}(1.0, 0.1, 0.1)$	$[1.0, 0.1, 0.1]$	Fig. A.3	Fig. A.4
$\mathrm{diag}(1.5, 0.1, 0.1)$	$[1.0, 0.1, 0.1]$	Fig. A.5	Fig. A.6
$\mathrm{diag}(2.0, 0.2, 0.2)$	$[1.0, 0.1, 0.1]$	Fig. A.7	Fig. A.8

Table A.1: Experimental result for unconstrained motion phase

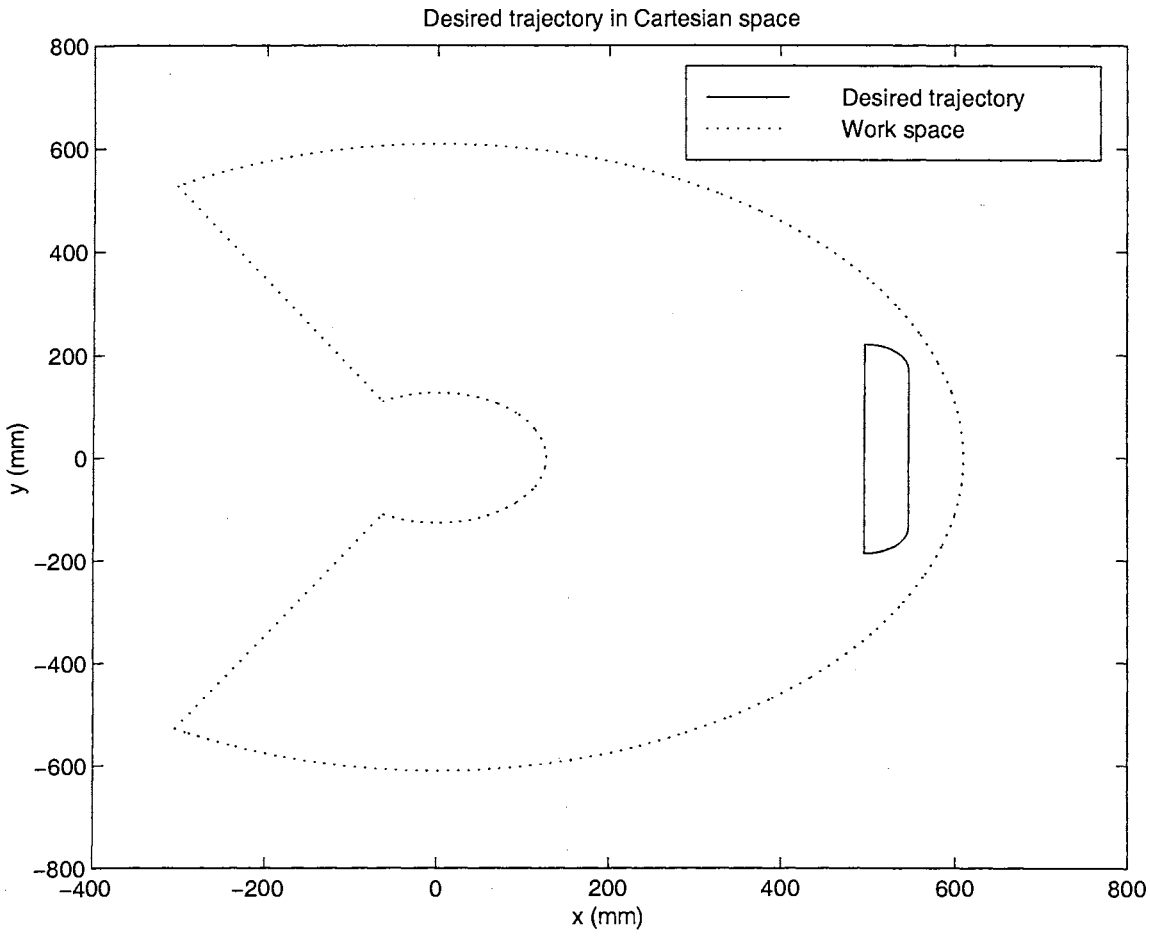

Figure A.1: Desired trajectory in Cartesian space

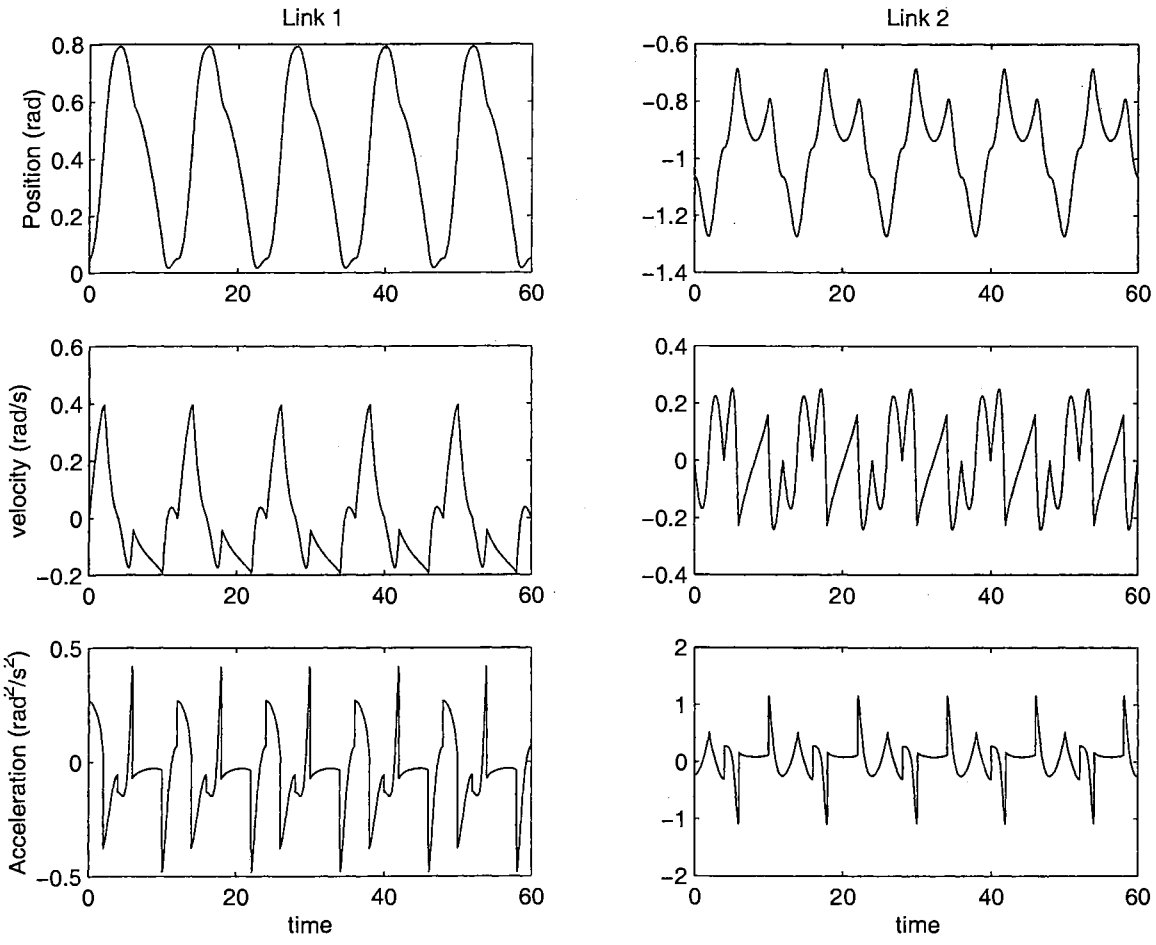

Figure A.2: Desired trajectory in join space

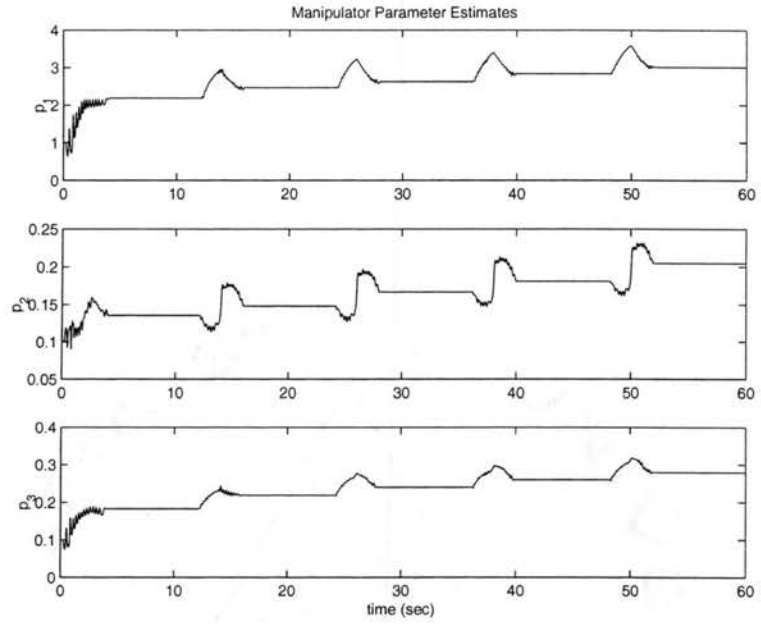

Figure A.3: Manipulator parameter estimates during free motion, adaptation gain $\Gamma^{-\top} = diag([1.0, 0.1, 0.1)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

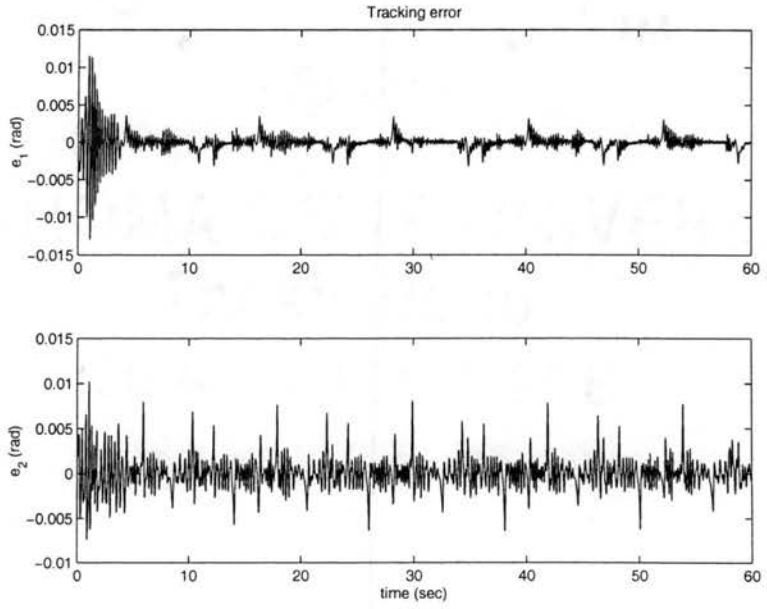

Figure A.4: Tracking error of free motion, adaptation gain $\Gamma^{-\top} = diag(1.0, 0.1, 0.1)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

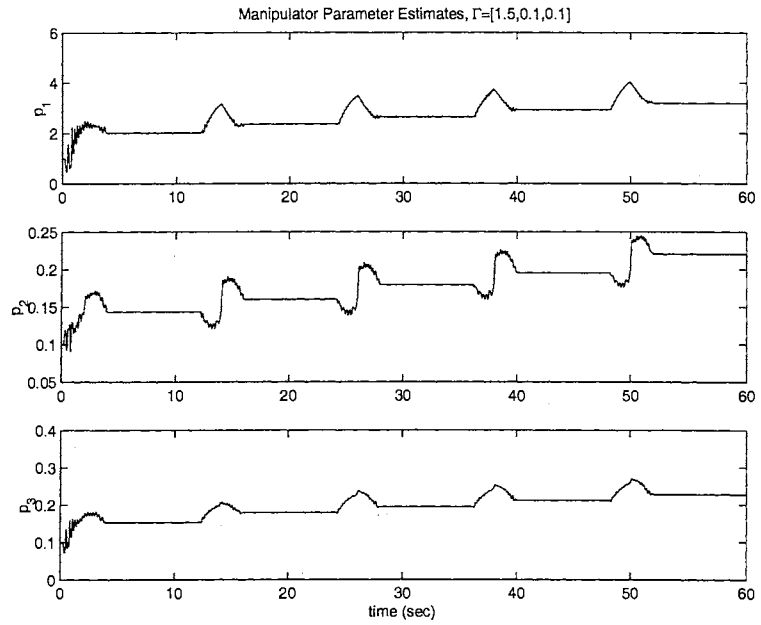

Figure A.5: Manipulator parameter estimates during free motion, adaptation gain $\Gamma^{-\top} = diag(1.5, 0.1, 0.1)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

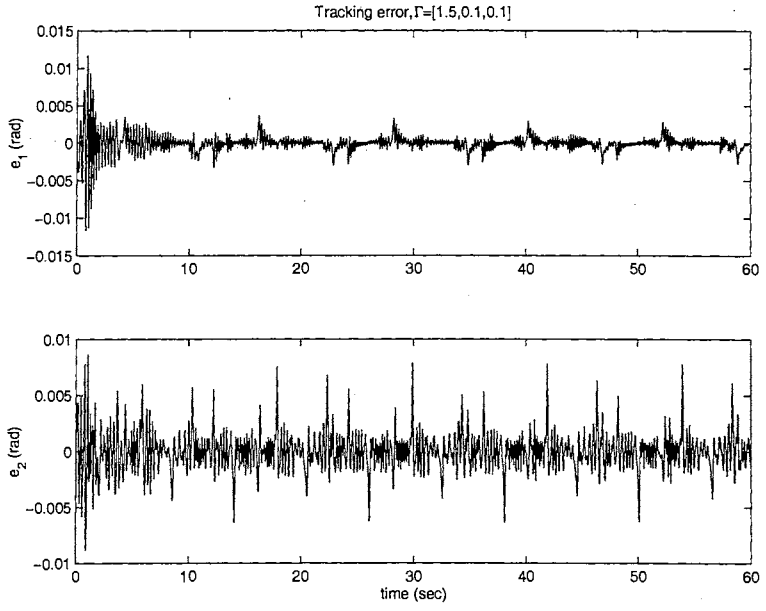

Figure A.6: Tracking error of free motion, adaptation gain $\Gamma^{-\top} = diag(1.5, 0.1, 0.1)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

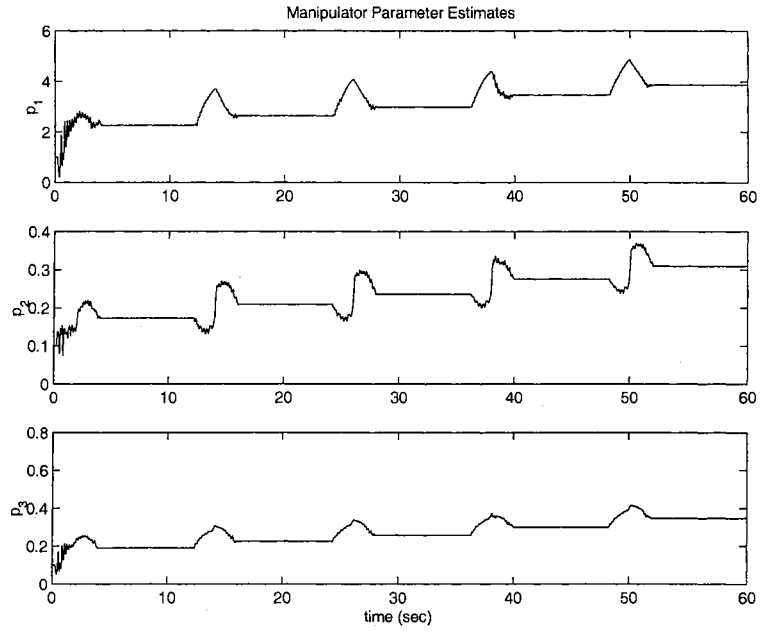

Figure A.7: Manipulator parameter estimates during free motion, adaptation gain $\Gamma^{-\top} = diag(2.0, 0.2, 0.2)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

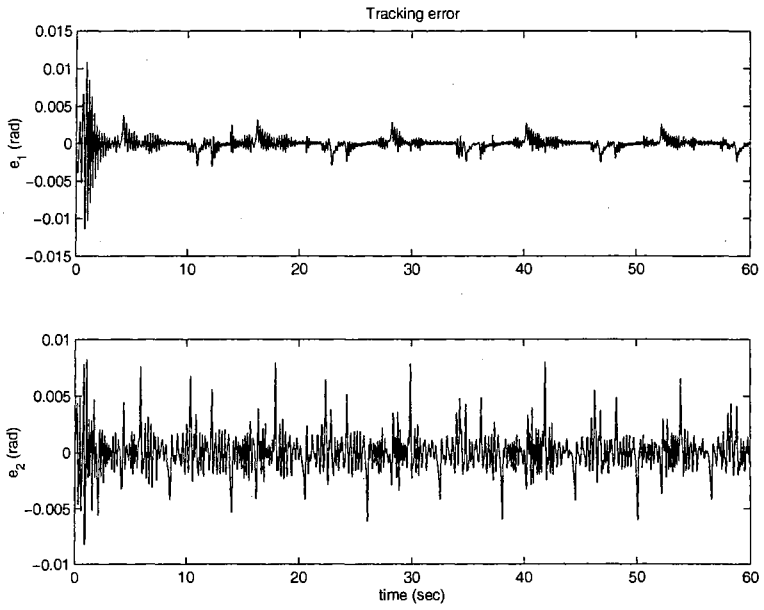

Figure A.8: Tracking error of free motion, adaptation gain $\Gamma^{-\top} = diag(2.0, 0.2, 0.2)$, initial value $\beta_0 = [1.0, 0.1, 0.1]^\top$

Appendix B

Experiments of Force Control for Constrained Motion

B.1 Force/position control with different desired force

Desired trajectory: straight constraint	In Cartesian: Fig. B.1 In joint space: Fig. B.3		
Desired normal force	$f_{nd} = 0$ N	$f_{nd} = 20$ N	$f_{nd} = 40$ N
Tracking error	Fig. B.5	Fig. B.7	Fig. B.9
Contact force	Fig. B.6	Fig. B.8	Fig. B.10
Desired trajectory: cylindrical constraint	In work space: Fig. B.2 In joint space: Fig. B.4		
Desired normal force	$f_{nd} = 20$ N	$f_{nd} = 30$ N	$f_{nd} = 40$ N
Tracking error	Fig. B.11	Fig. B.13	Fig. B.15
Contact force	Fig. B.12	Fig. B.14	Fig. B.16

Table B.1: Force/position control with different desired force

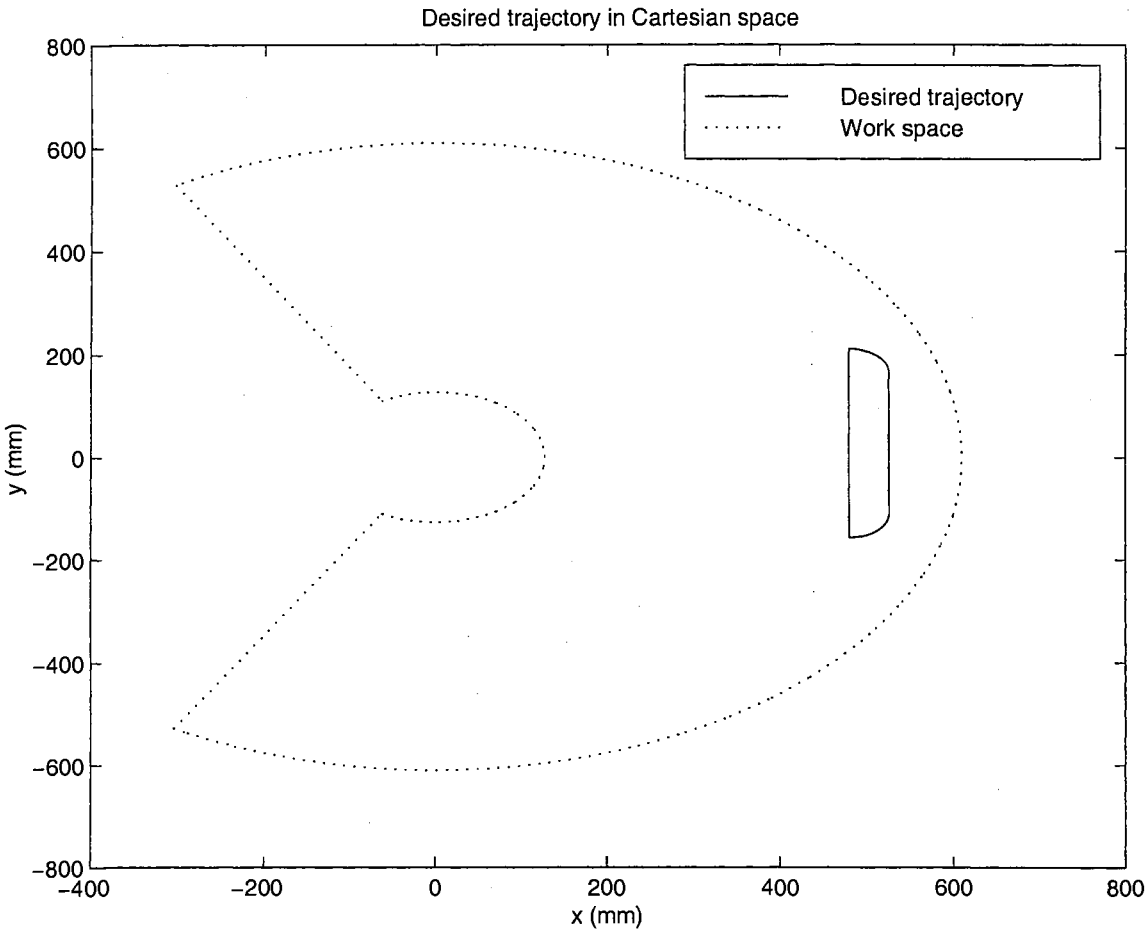

Figure B.1: Desired trajectory in Cartesian space: straight constraint

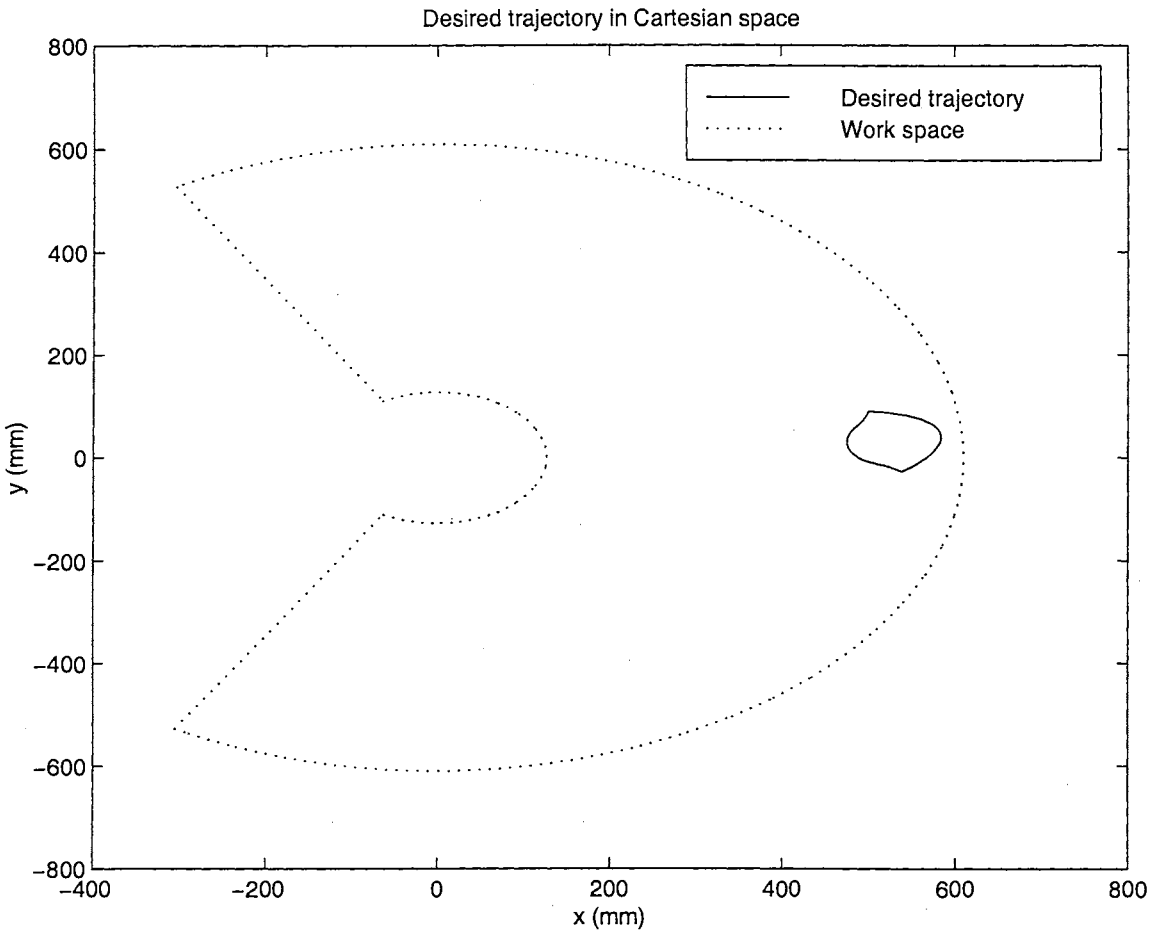

Figure B.2: Desired trajectory in Cartesian space: cylindrical constraint

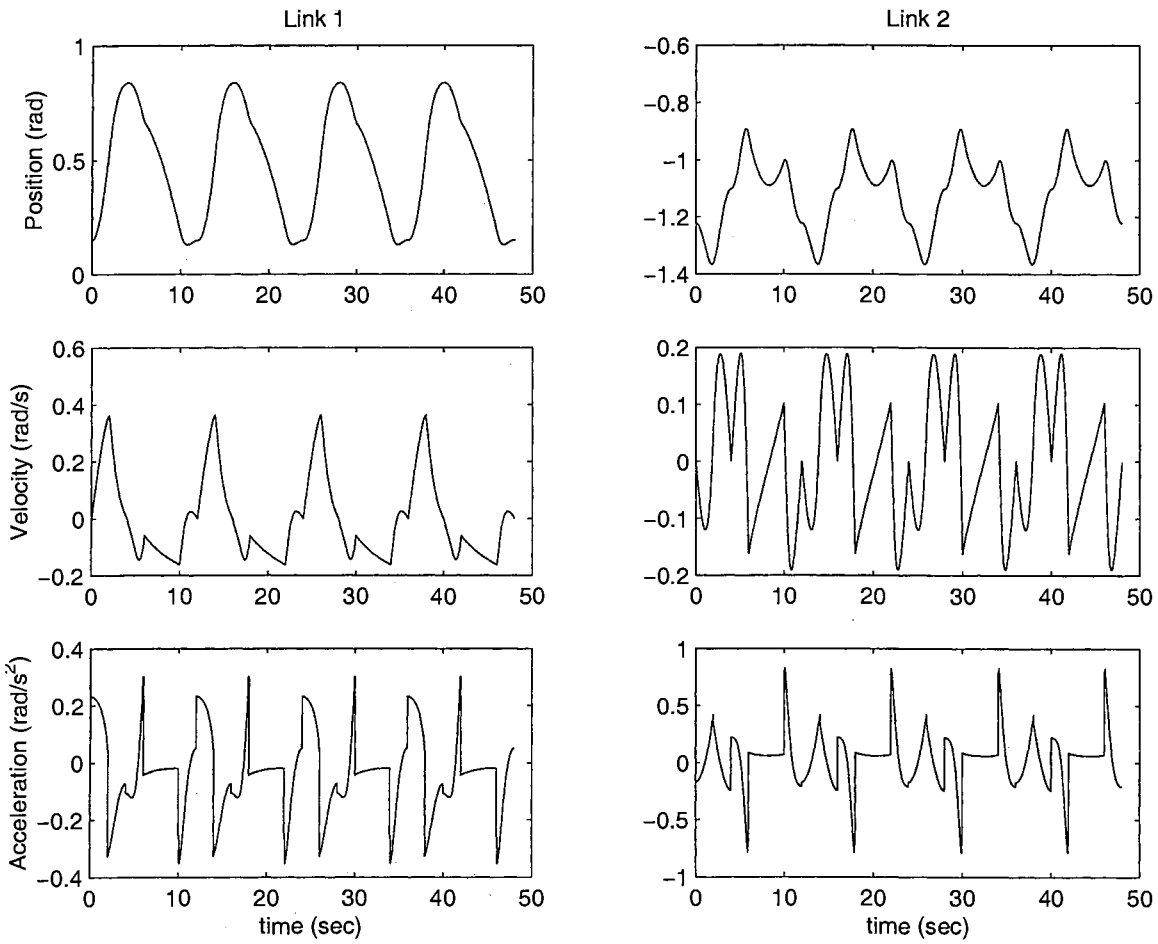

Figure B.3: Desired trajectory in join space: straight constraint

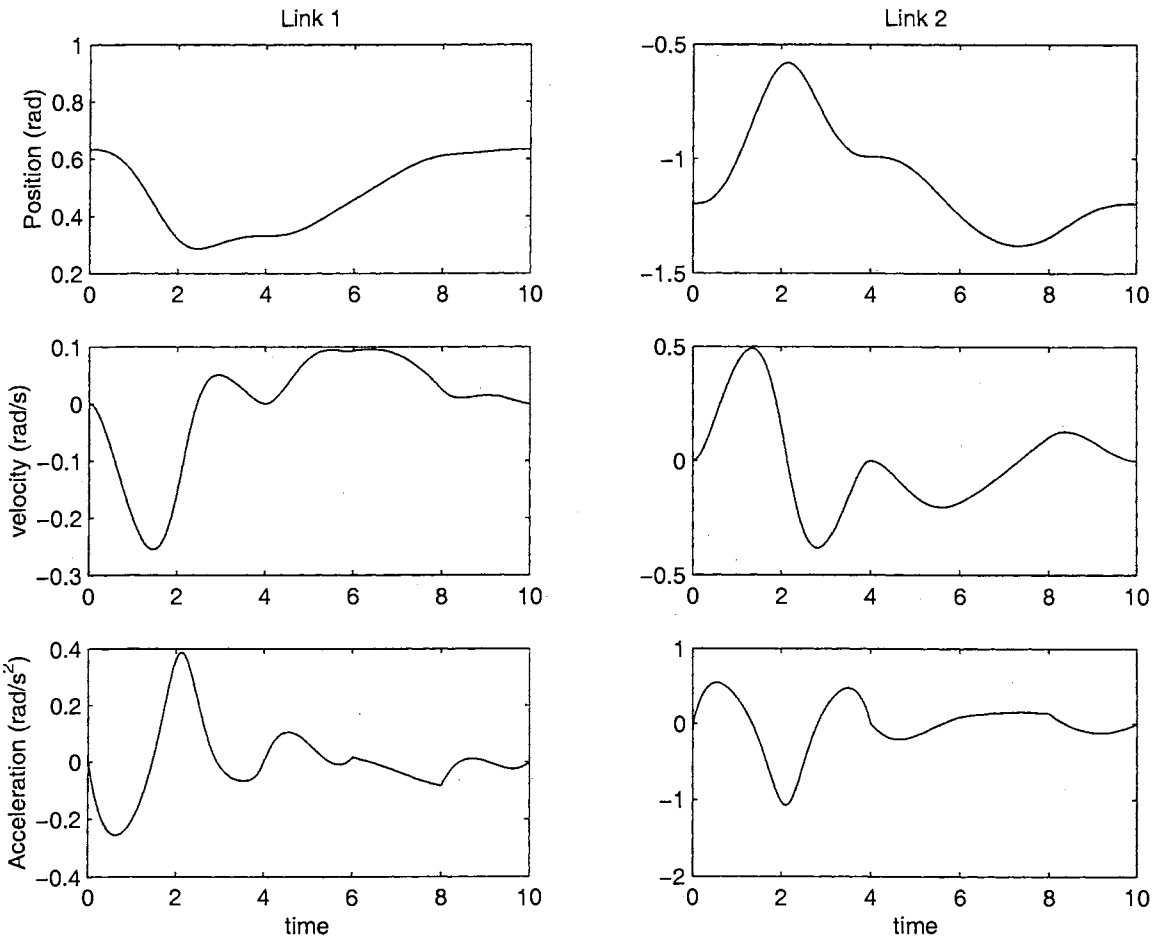

Figure B.4: Desired trajectory in join space: cylindrical constraint

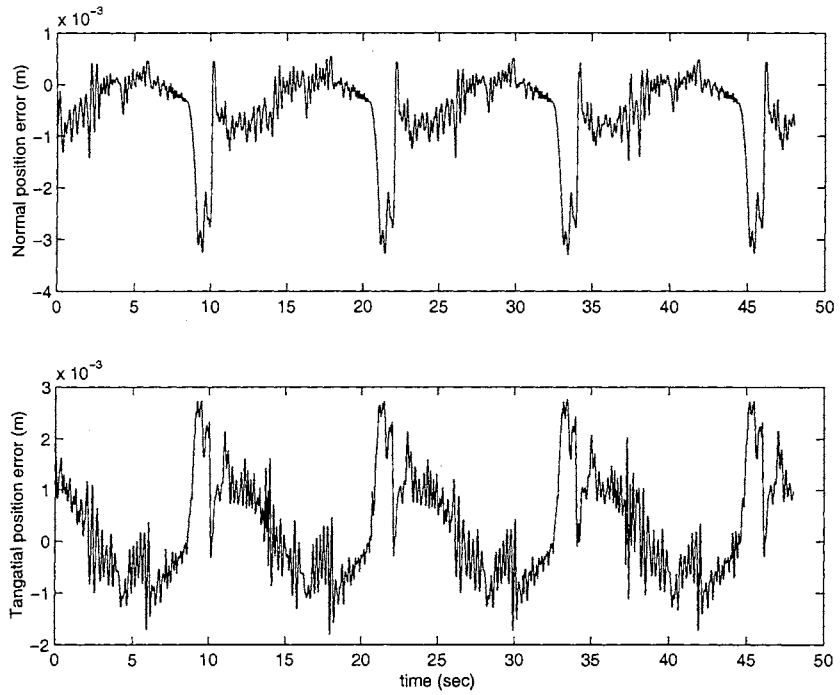

Figure B.5: Tracking error, $f_{nd} = 0$, straight constraint

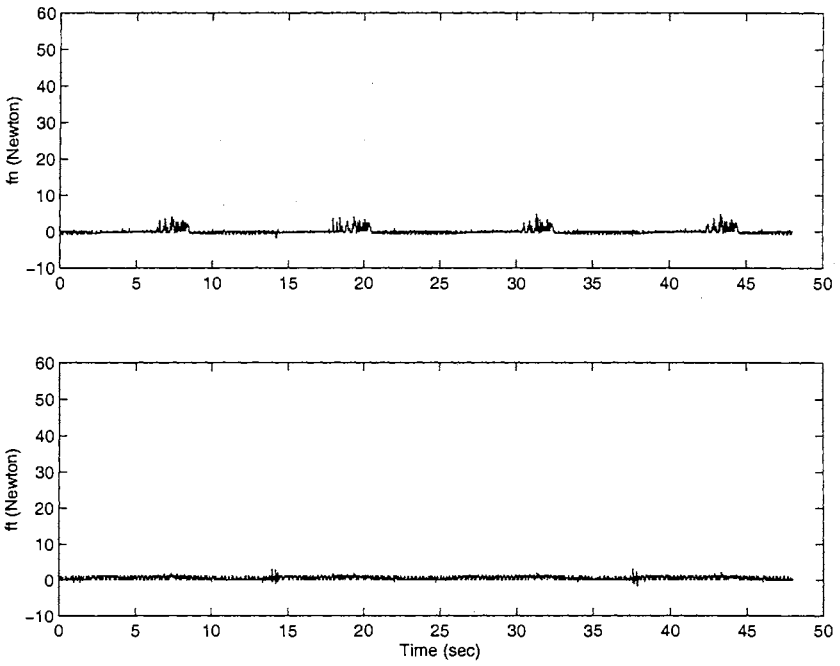

Figure B.6: Contact force, $f_{nd} = 0$, straight constraint

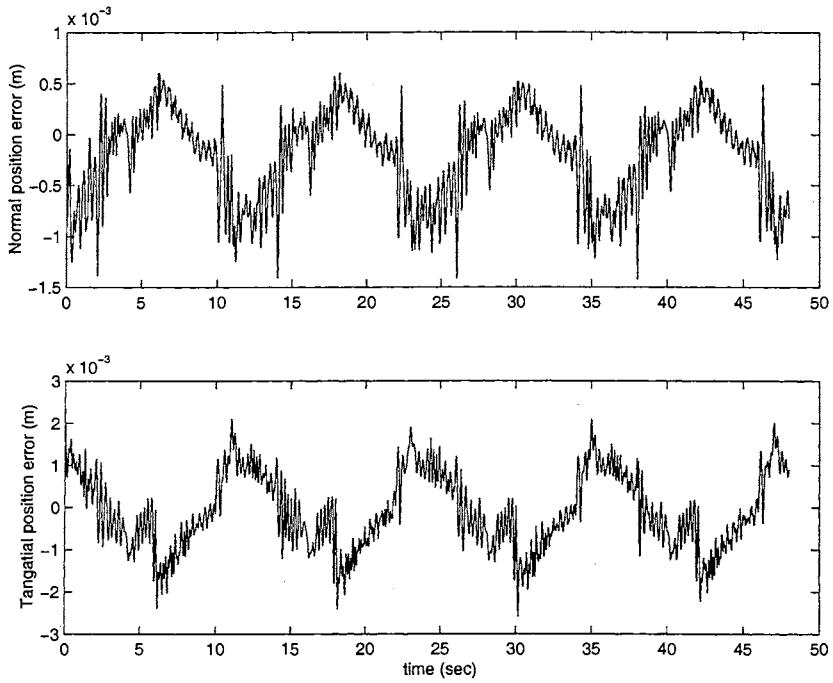

Figure B.7: Tracking error, $f_{nd} = 20$ N, straight constraint

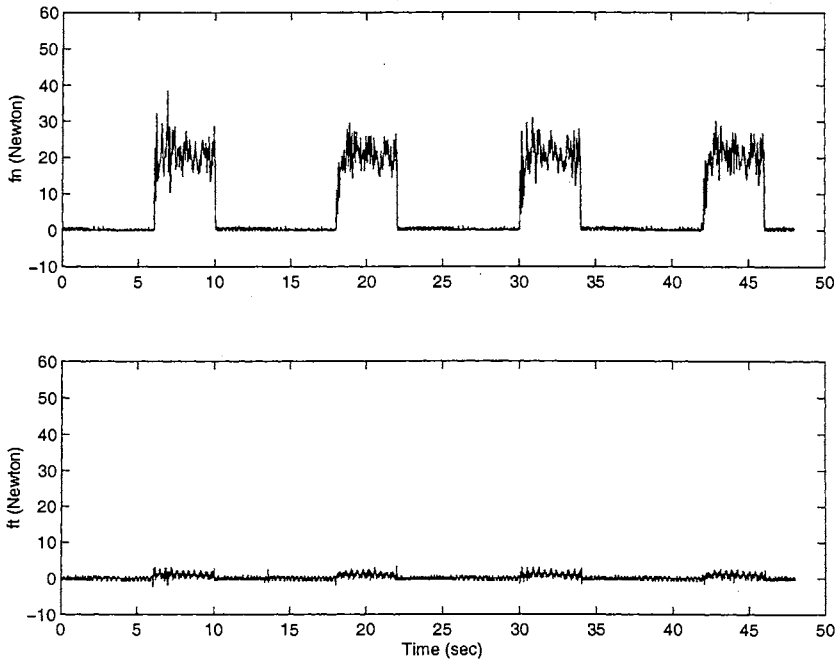

Figure B.8: Contact force, $f_{nd} = 20$ N, straight constraint

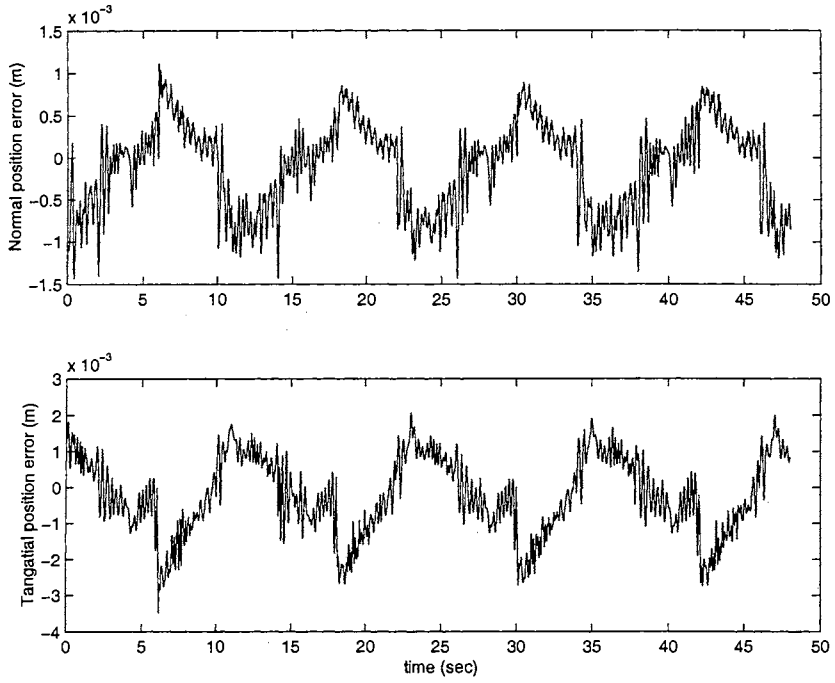

Figure B.9: Tracking error, $f_{nd} = 40$ N, straight constraint

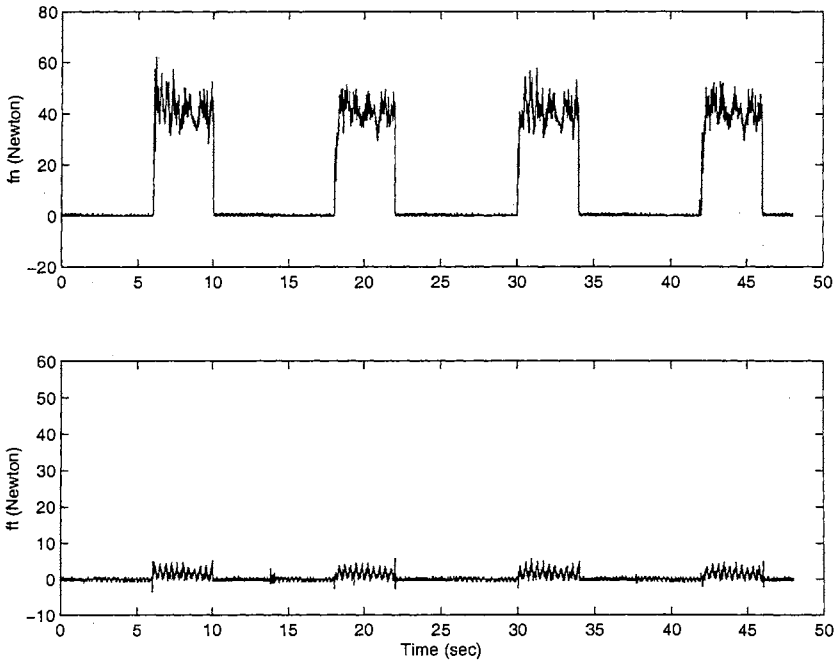

Figure B.10: Contact force, $f_{nd} = 40$ N, straight constraint

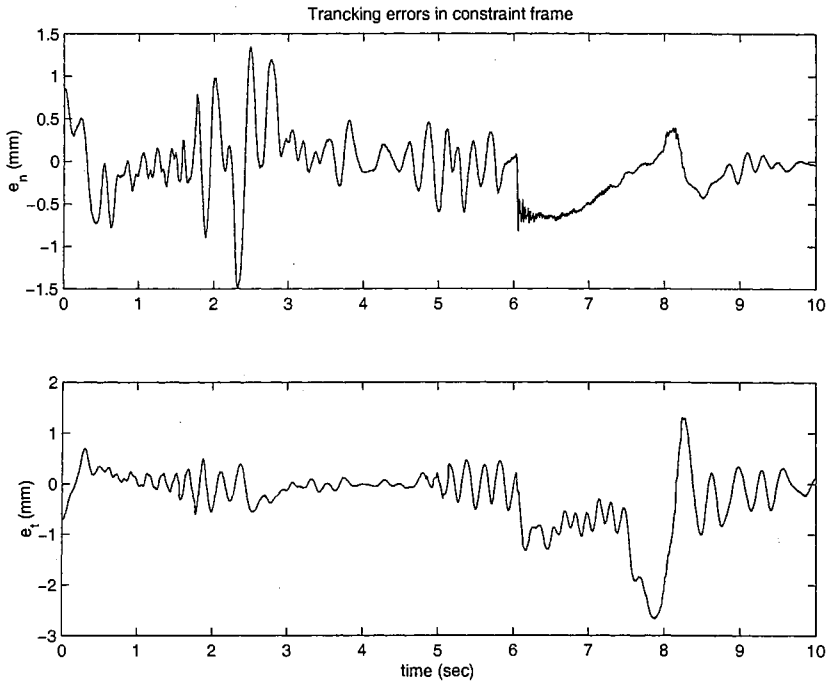

Figure B.11: Tracking error, $f_{nd} = 20$ N, cylindrical constraint

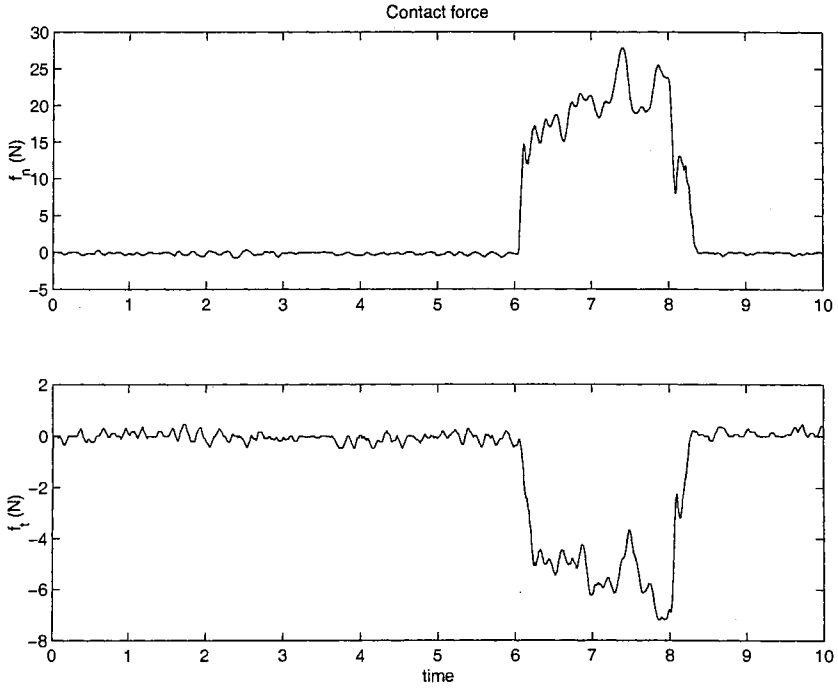

Figure B.12: Contact force, $f_{nd} = 20$ N, cylindrical constraint

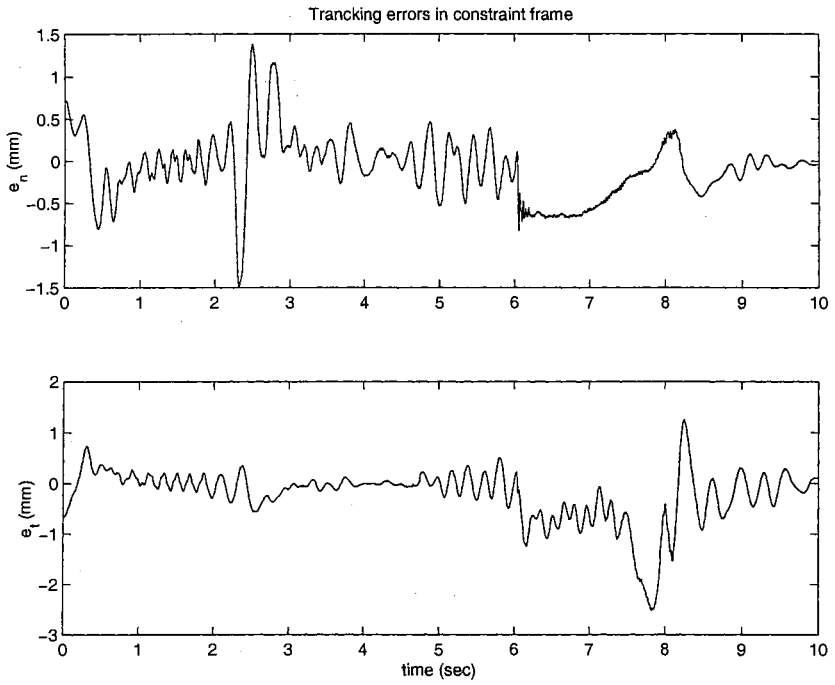

Figure B.13: Tracking error, $f_{nd} = 30$ N, cylindrical constraint

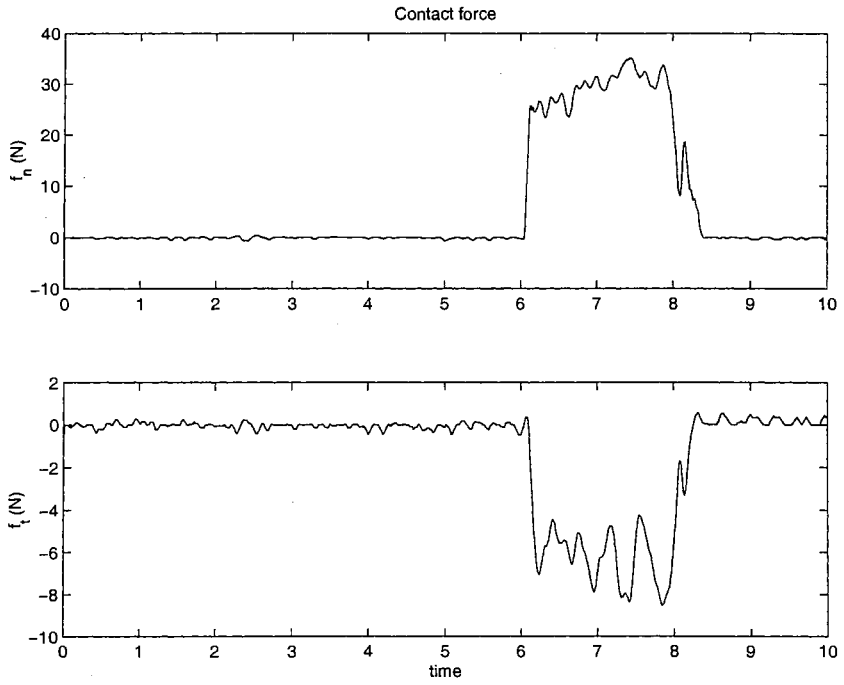

Figure B.14: Contact force, $f_{nd} = 30$ N, cylindrical constraint

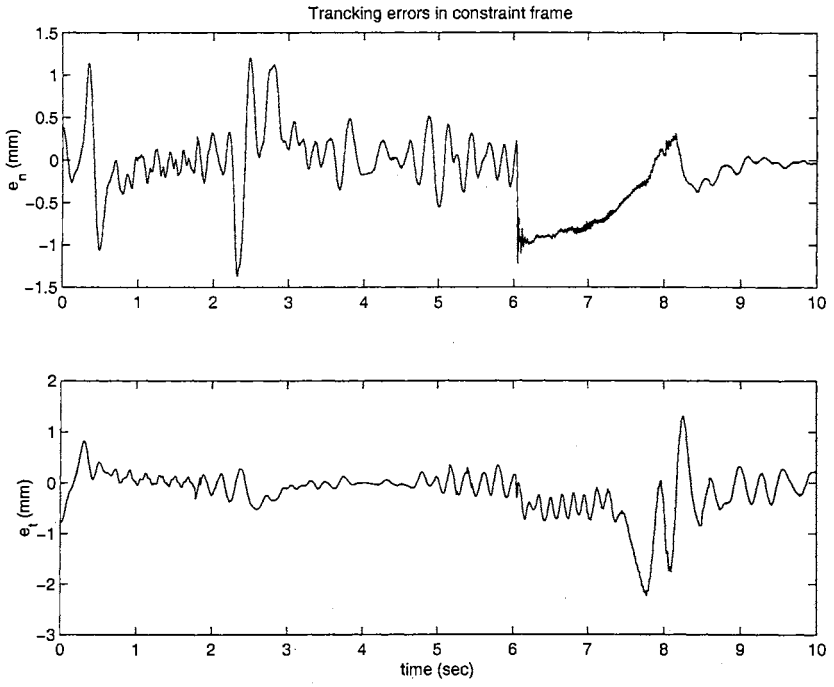

Figure B.15: Tracking error, $f_{nd} = 40$ N, cylindrical constraint

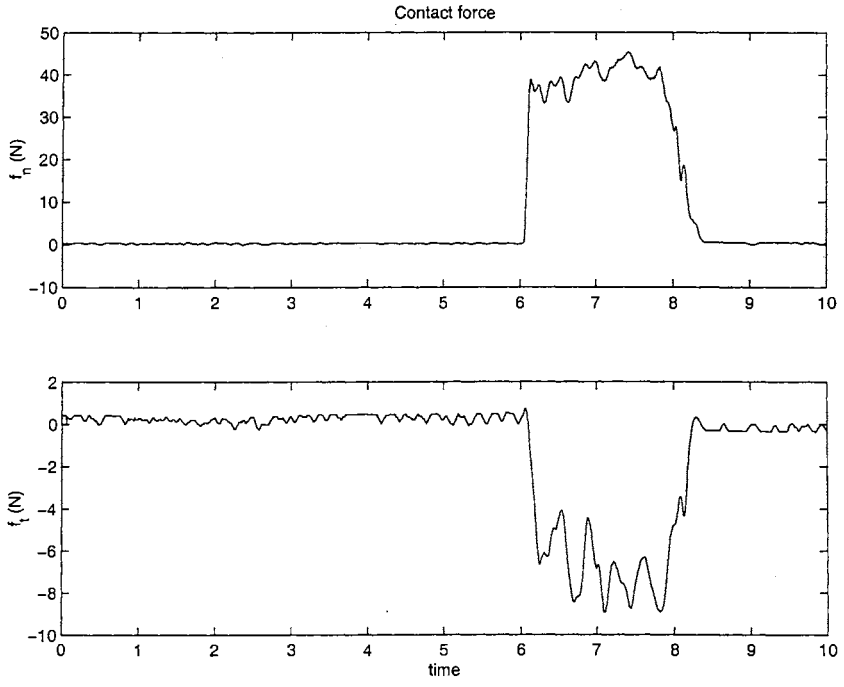

Figure B.16: Contact force, $f_{nd} = 40$ N, cylindrical constraint

B.2 Adaptive force/position control with grinding coefficient adaptation

Desired trajectory: straight constraint	In Cartesian space: Fig. B.1 In joint space: Fig. B.3	
Adaptation gain of ξ	γ_f=0.1	γ_f=0.3
Estimate of ξ	Fig. B.17	
Tracking errors in joint space	Fig. B.18	Fig. B.21
Tracking errors in constraint frame	Fig. B.19	Fig. B.22
Contact force	Fig. B.20	Fig. B.23
Desired trajectory: cylindrical constraint	In Cartesian space: Fig. B.2 In joint space: Fig. B.4	
Adaptation gain of ξ	γ_f=0.1	γ_f=0.3
Estimate of ξ	Fig. B.24	
Tracking errors in joint space	Fig. B.25	Fig. B.28
Tracking errors in constraint frame	Fig. B.26	Fig. B.29
Contact force	Fig. B.27	Fig. B.30

Table B.2: Force/position control with friction coefficient adaptation

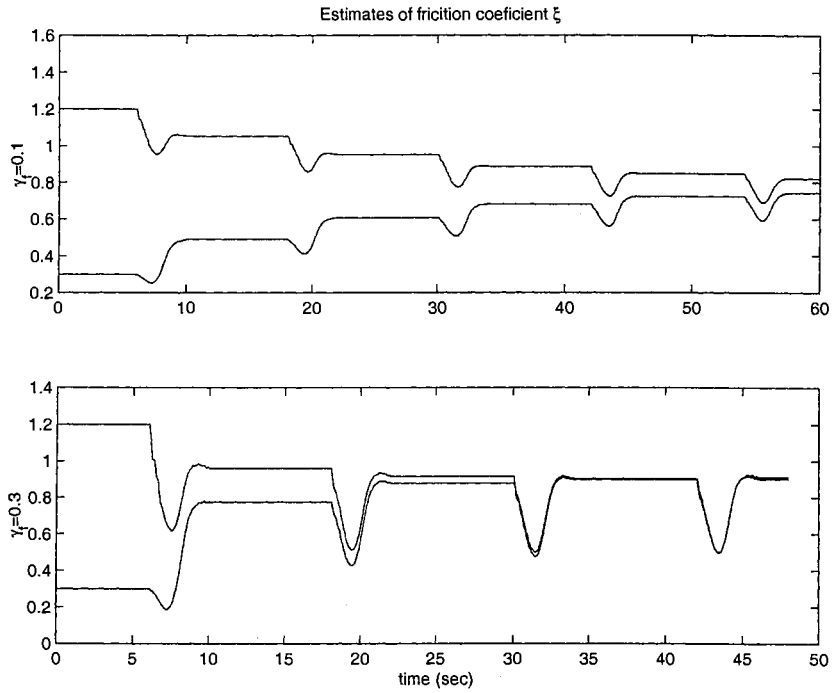

Figure B.17: Estimate of friction coefficient $\widehat{\xi}$ with straight constraint

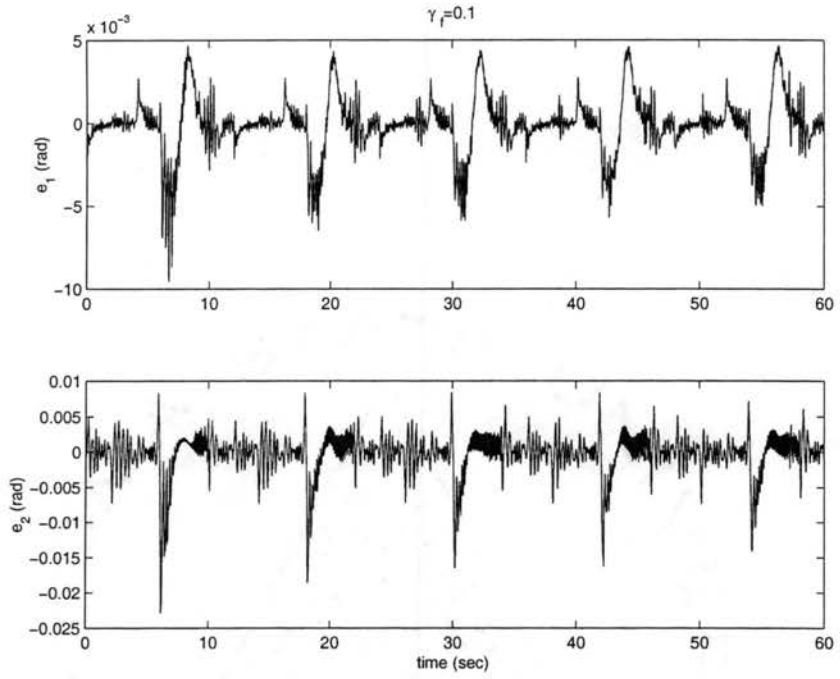

Figure B.18: Tracking errors in joint space, $\gamma_f = 1.0$, straight constraint

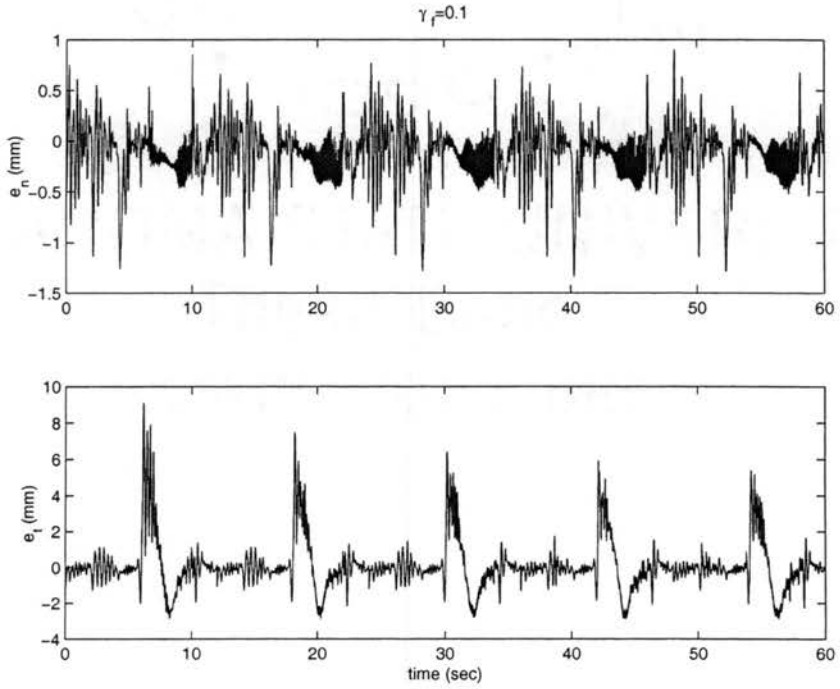

Figure B.19: Tracking errors in constraint frame, $\gamma_f = 1.0$, straight constraint

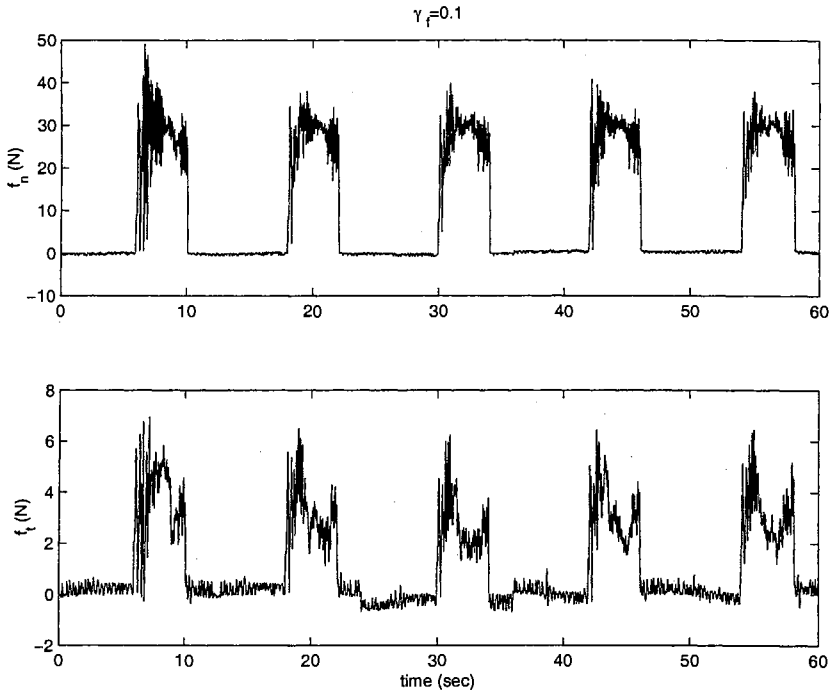

Figure B.20: Contact force, $\gamma_f = 1.0$, straight constraint

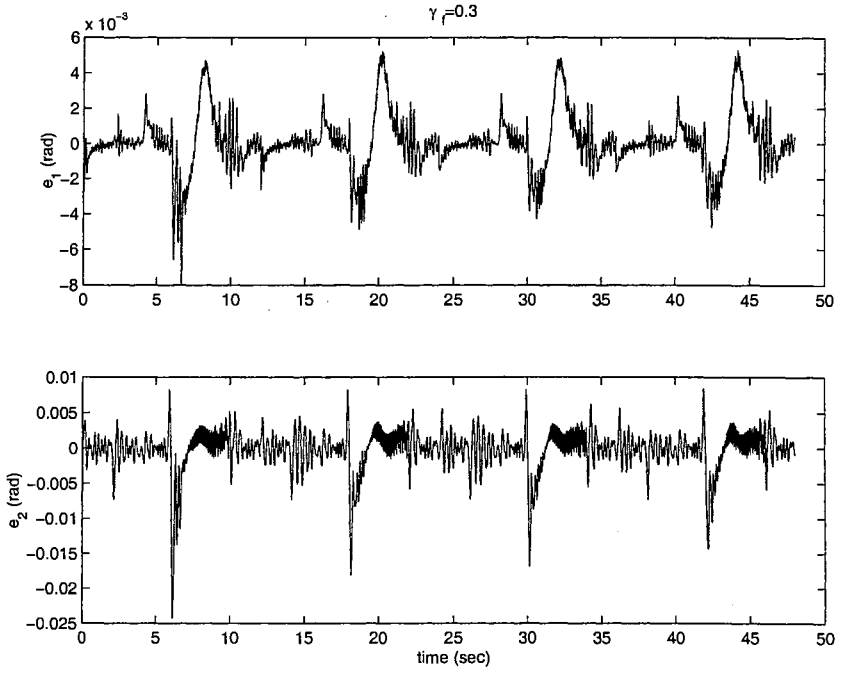

Figure B.21: Tracking errors in joint space, $\gamma_f = 3.0$, straight constraint

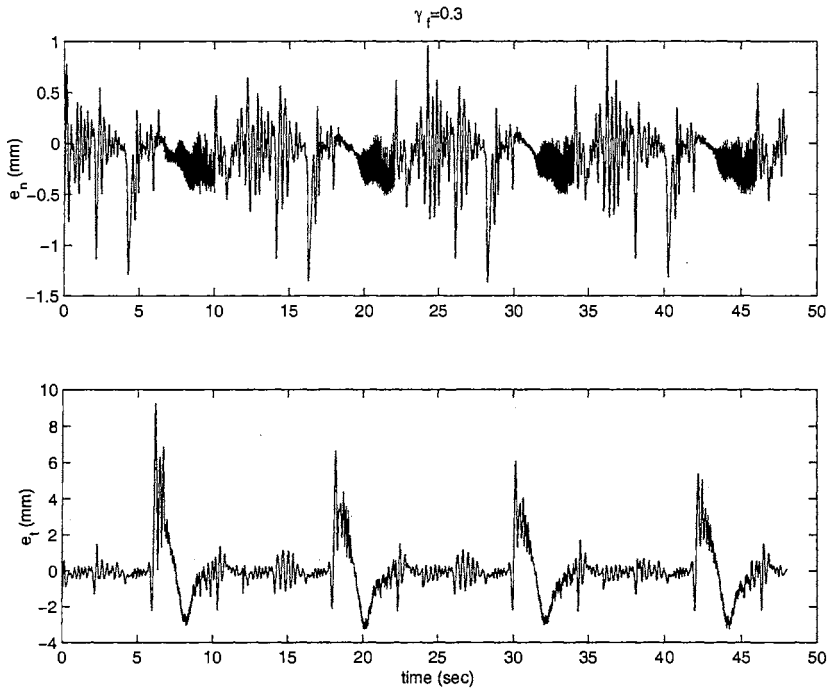

Figure B.22: Tracking errors in constraint frame, $\gamma_f = 3.0$, straight constraint

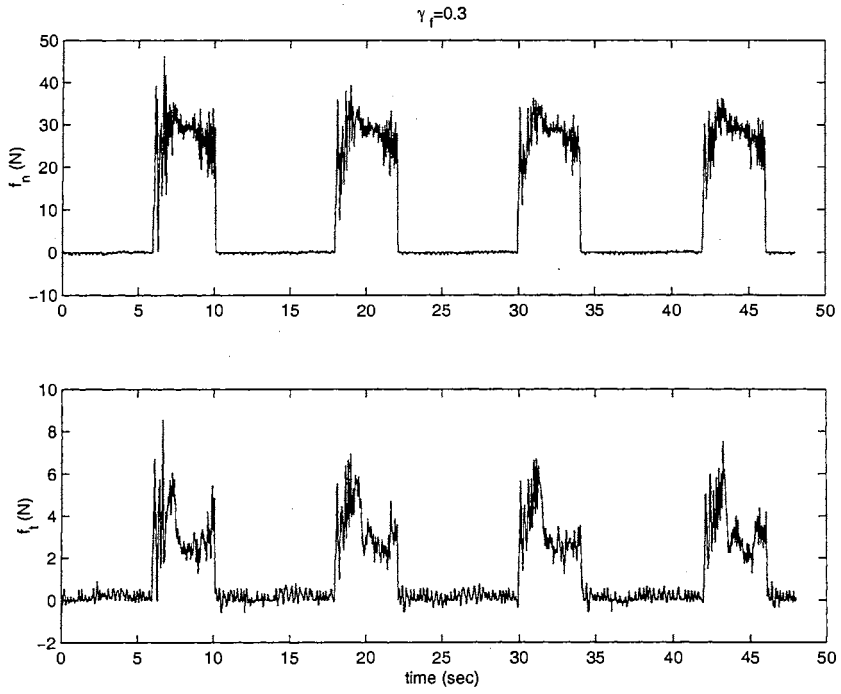

Figure B.23: Contact force, $\gamma_f = 3.0$, straight constraint

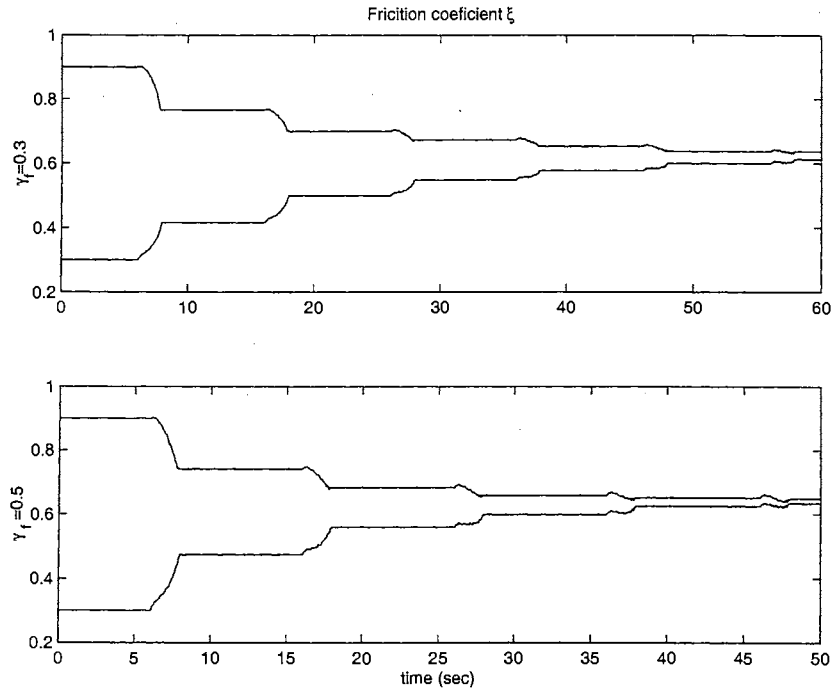

Figure B.24: Estimate of friction coefficient $\widehat{\xi}$ with cylindrical constraint

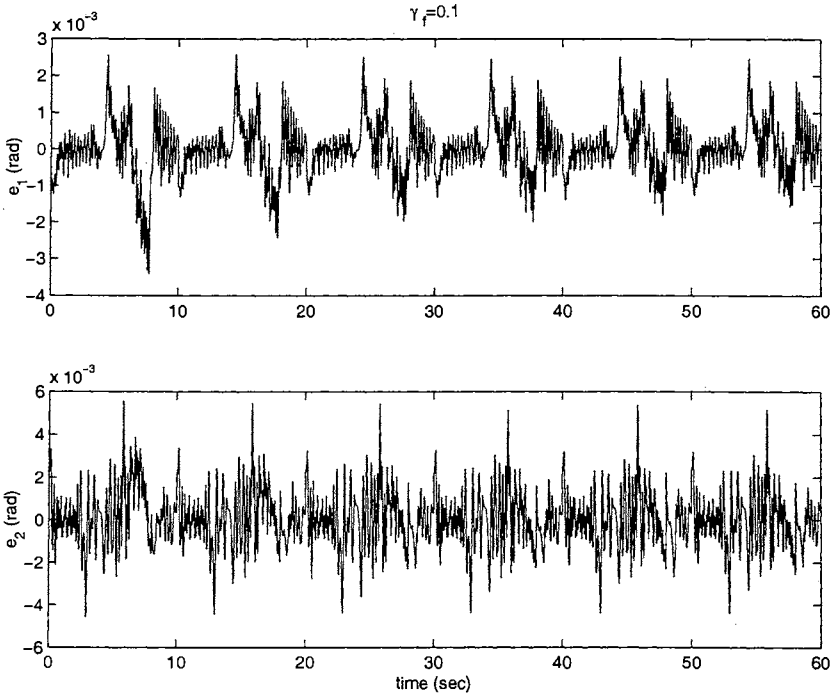

Figure B.25: Tracking errors in joint space, $\gamma_f = 1.0$, cylindrical constraint

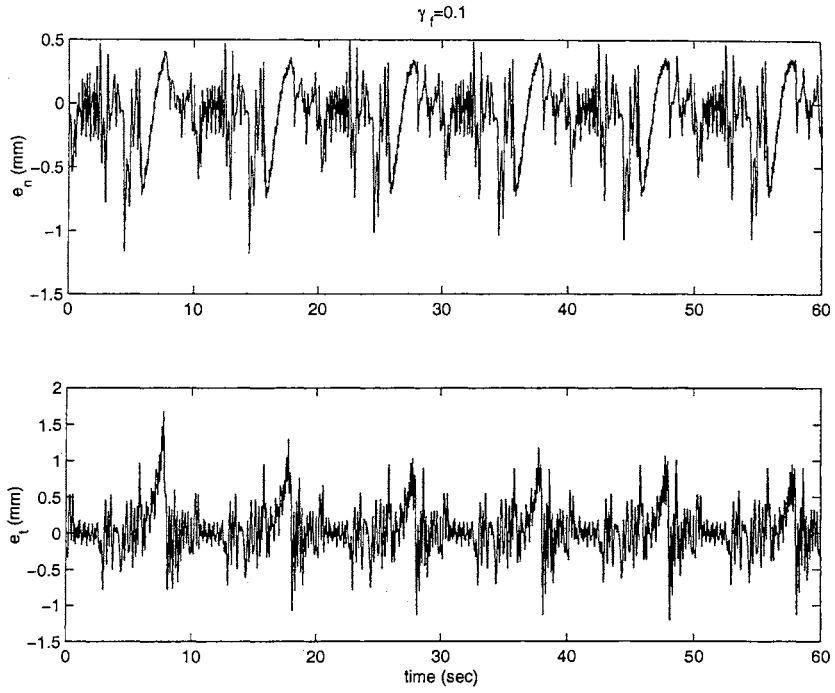

Figure B.26: Tracking errors in constraint frame, $\gamma_f = 1.0$, cylindrical constraint

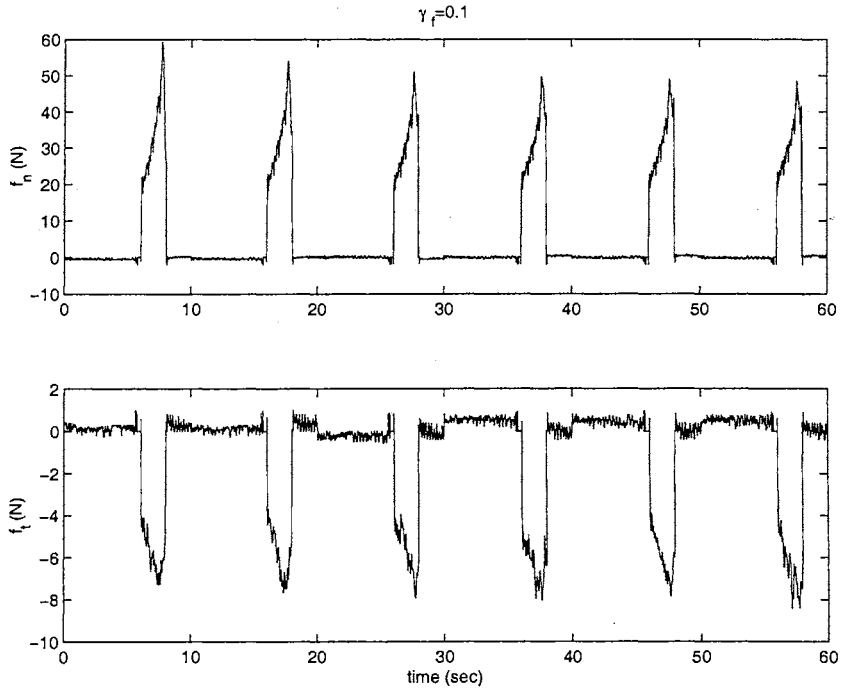

Figure B.27: Contact force, $\gamma_f = 1.0$, cylindrical constraint

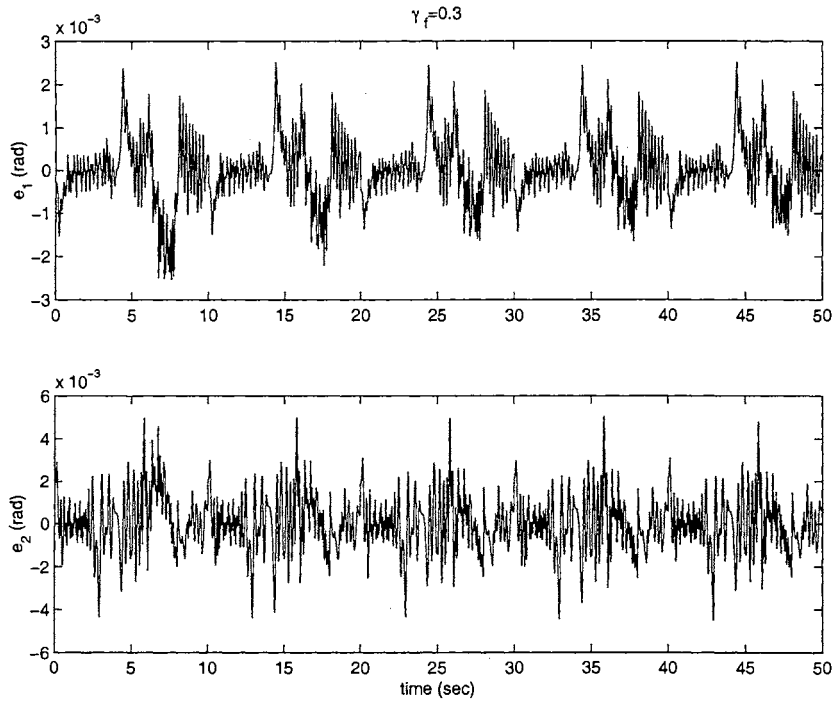

Figure B.28: Tracking errors in joint space, $\gamma_f = 3.0$, cylindrical constraint

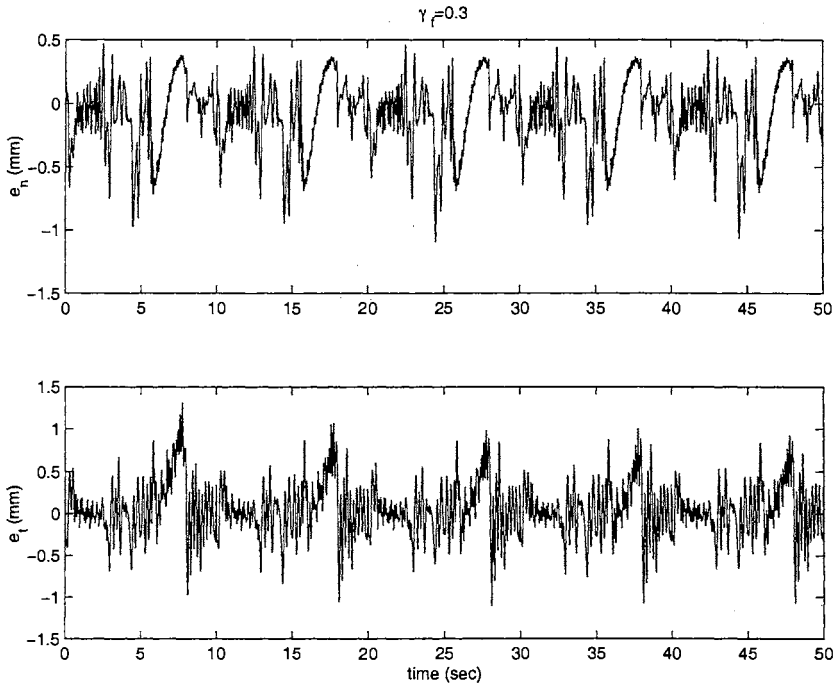

Figure B.29: Tracking errors in constraint frame, $\gamma_f = 3.0$, cylindrical constraint

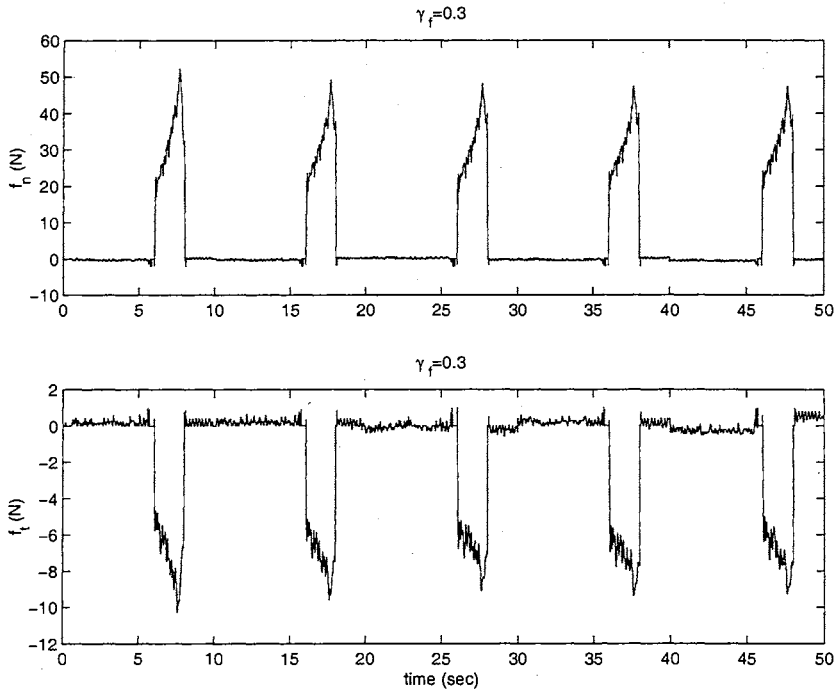

Figure B.30: Contact force, $\gamma_f = 3.0$, cylindrical constraint

Appendix C

Experiments of Free Impact

Up-elbow configuration			
Impact velocity	$v = 0.1$ m/s	$v = 0.2$ m/s	$v = 0.3$ m/s
Velocity	Fig. C.1	Fig. 5.35	Fig. C.4
Impact force	Fig. C.2	Fig. 5.37	Fig. C.5
Manipulator path near impact	Fig. C.3	Fig. 5.39	Fig. C.6
Down-elbow configuration			
Impact velocity	$v = 0.1$ m/s	$v = 0.2$ m/s	$v = 0.3$ m/s
Velocity	Fig. C.7	Fig. 5.36	Fig. C.10
Impact force	Fig. C.8	Fig. 5.38	Fig. C.11
Manipulator path near impact	Fig. C.9	Fig. 5.40	Fig. C.12

Table C.1: Experimental results of free impact

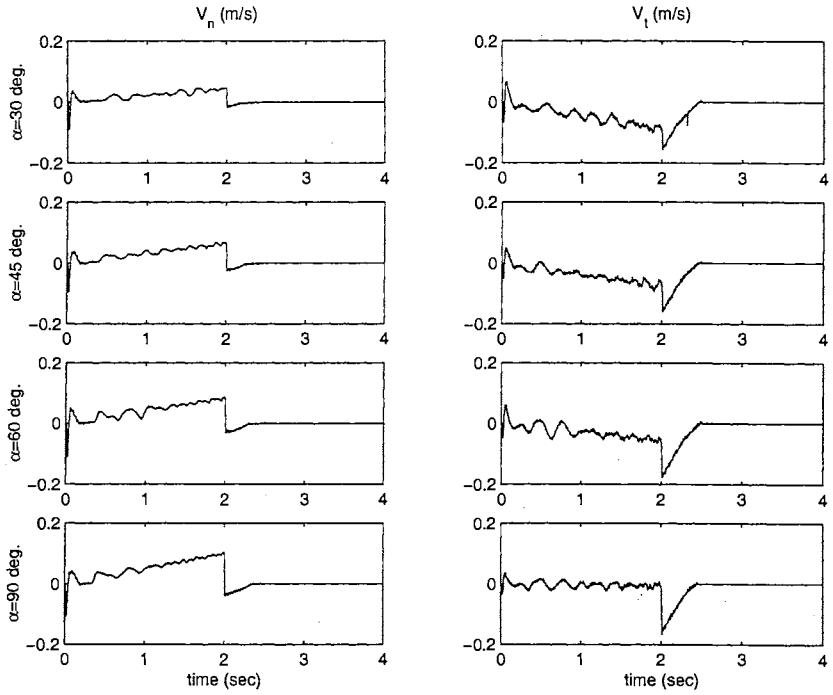

Figure C.1: Velocity, pre-impact velocity $v = 0.1$ m/s, up-elbow

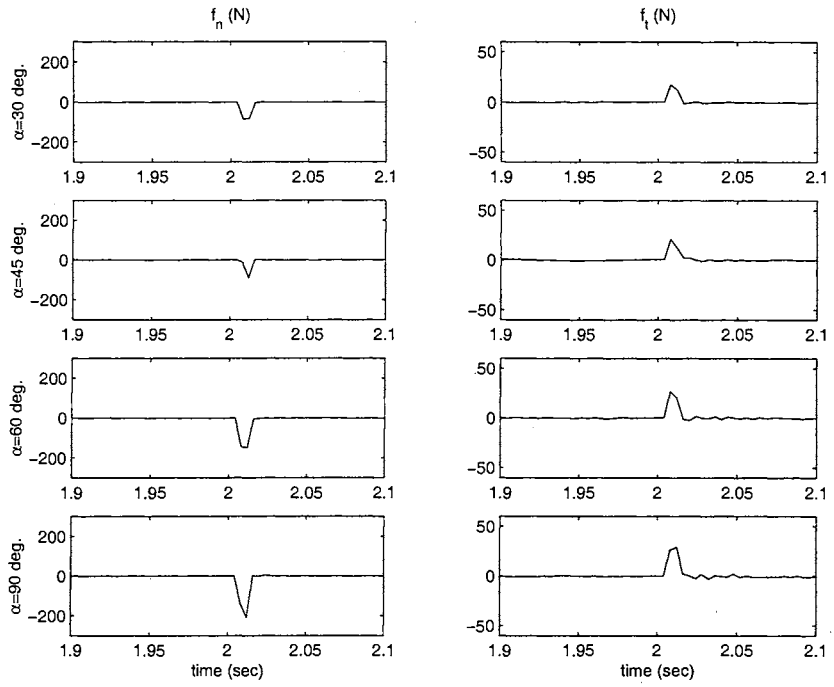

Figure C.2: Impact force, pre-impact velocity $v = 0.1$ m/s, up-elbow

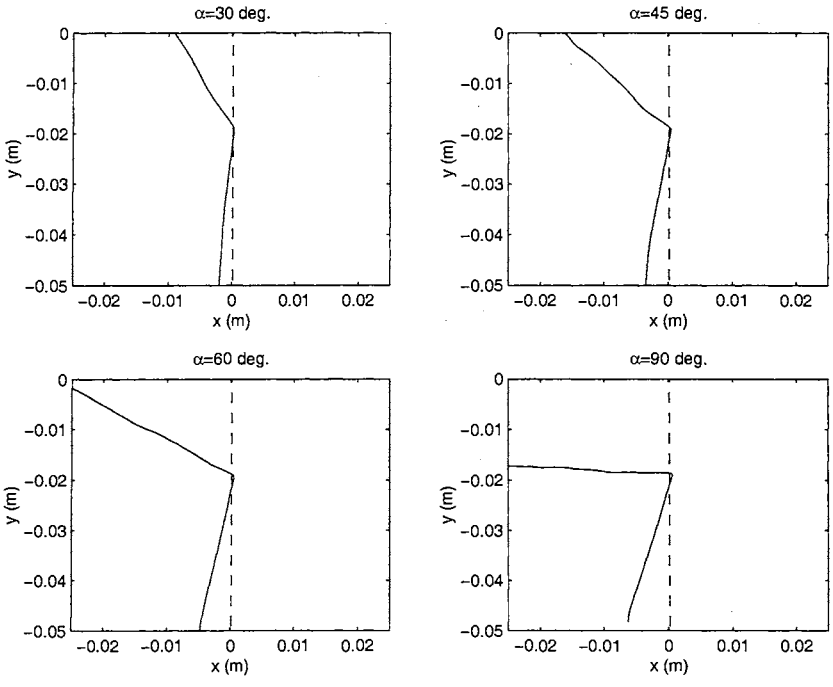

Figure C.3: Robot path, pre-impact velocity $v = 0.1$ m/s, up-elbow

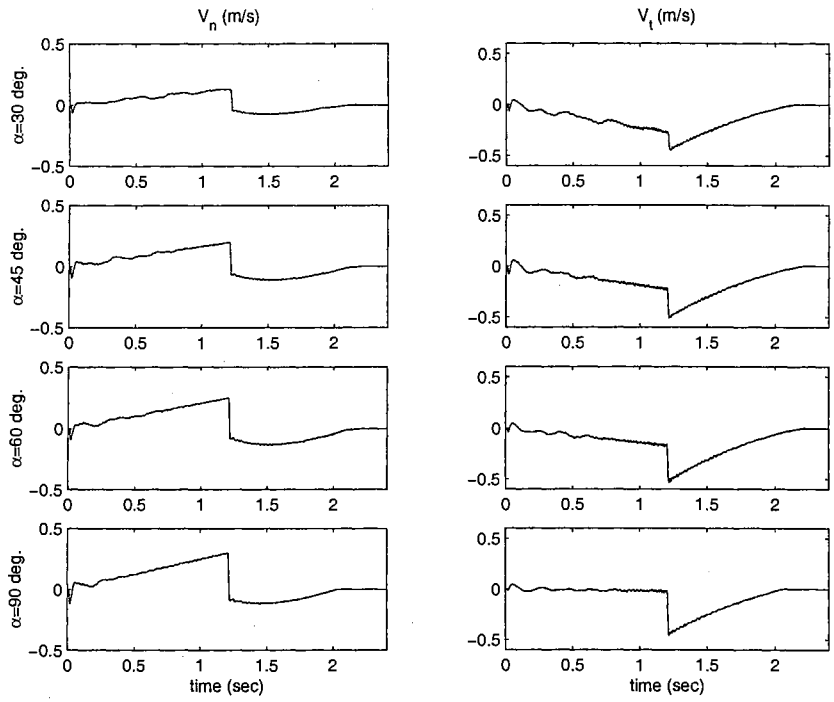

Figure C.4: Velocity, pre-impact velocity $v = 0.3$ m/s, up-elbow

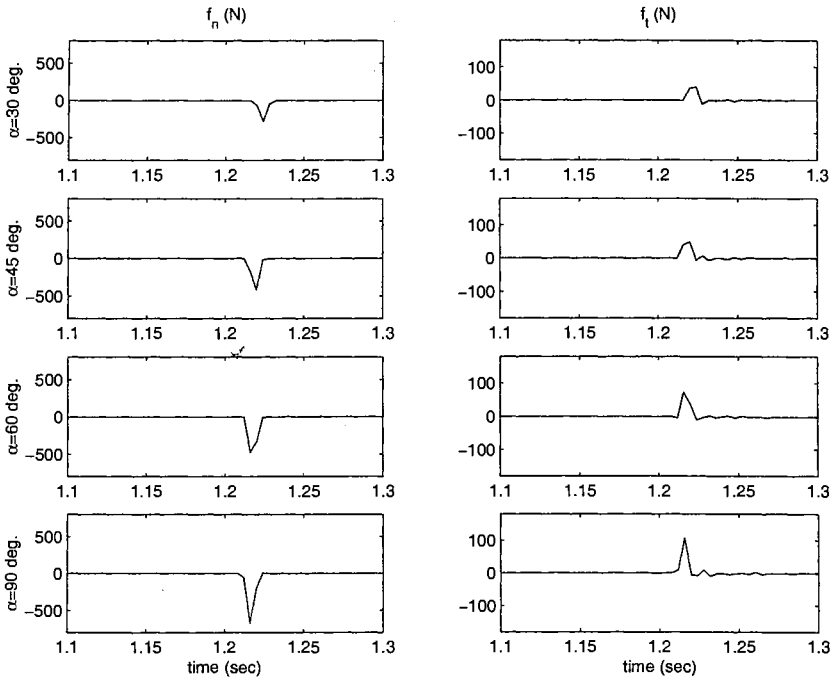

Figure C.5: Impact force, pre-impact velocity $v = 0.3$ m/s, up-elbow

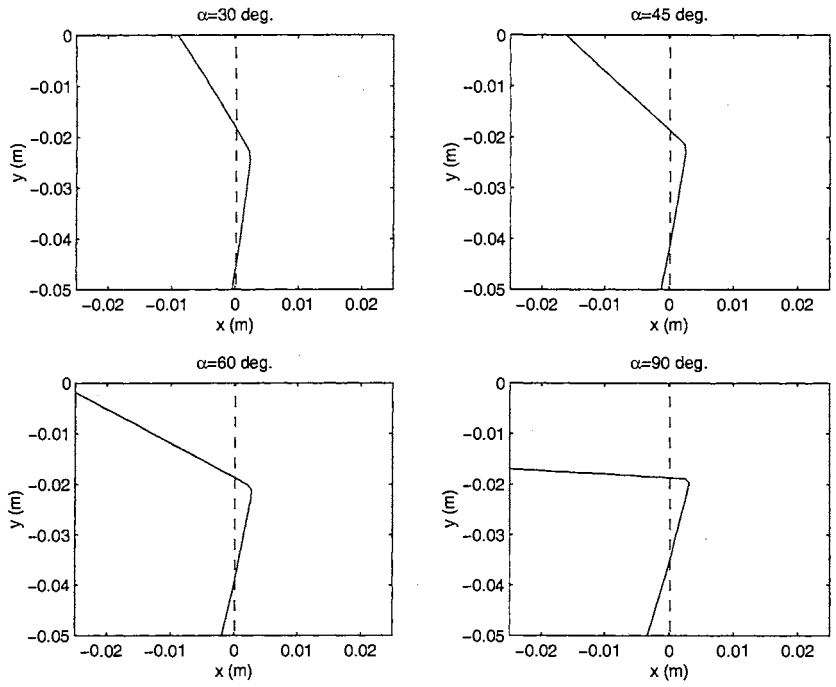

Figure C.6: Robot path, pre-impact velocity $v = 0.3$ m/s, up-elbow

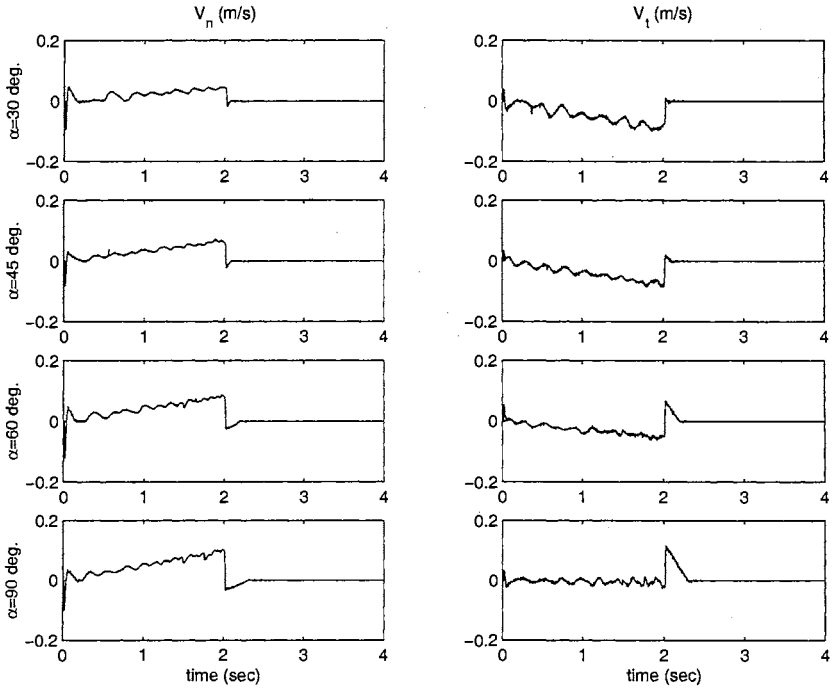

Figure C.7: Velocity, pre-impact velocity $v = 0.1$ m/s, down-elbow

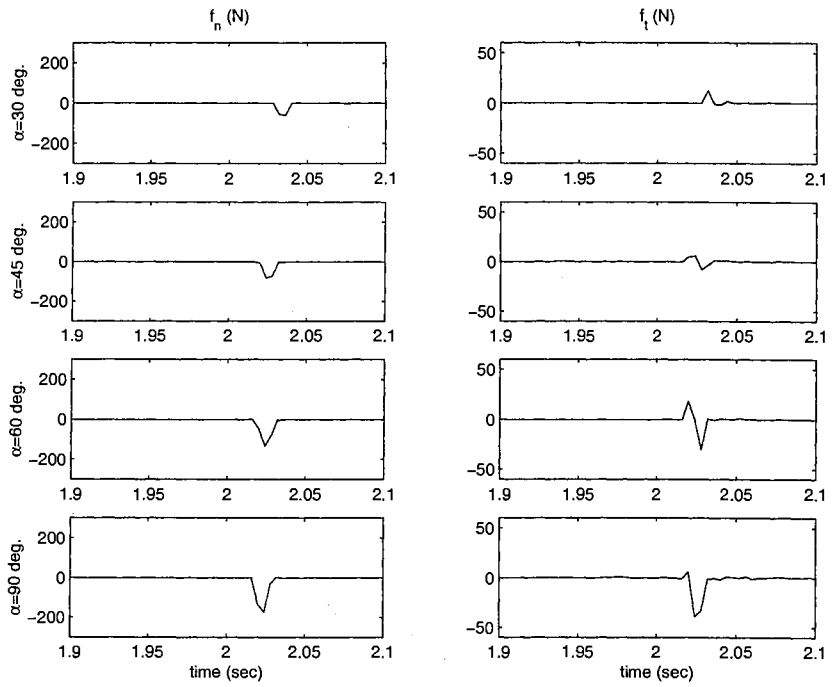

Figure C.8: Impact force, pre-impact velocity $v = 0.1$ m/s, down-elbow

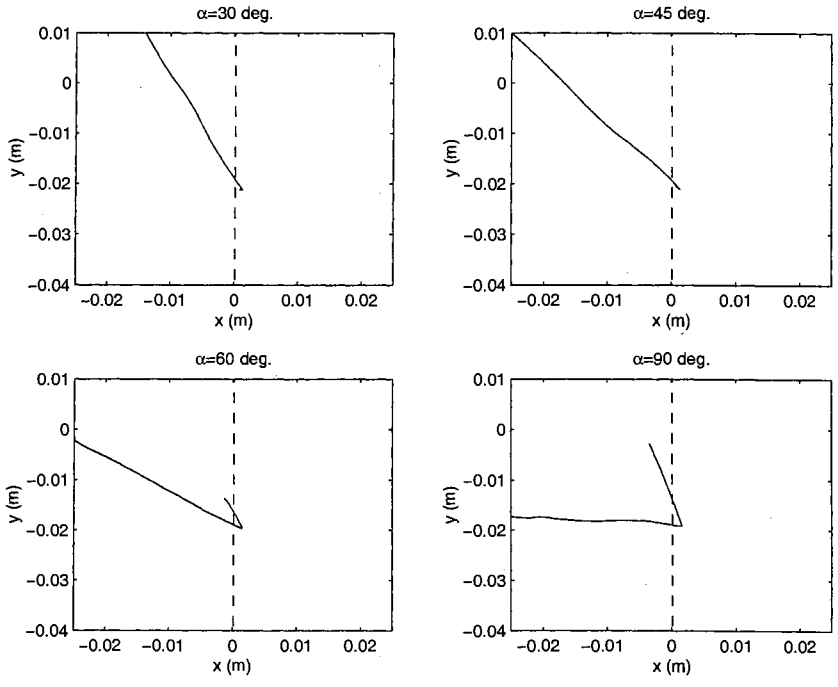

Figure C.9: Robot path, pre-impact velocity $v = 0.1$ m/s, down-elbow

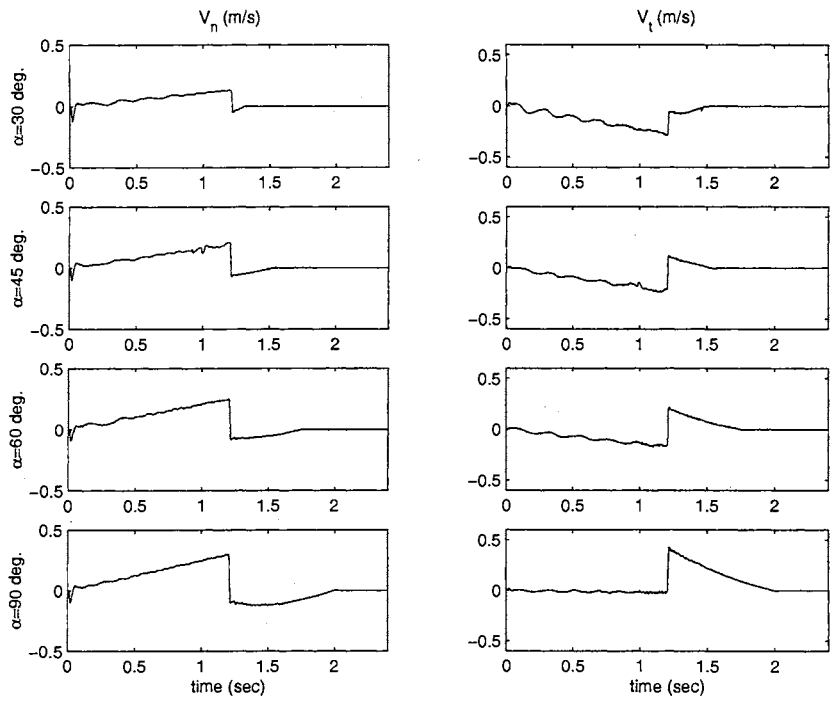

Figure C.10: Velocity, pre-impact velocity $v = 0.3$ m/s, down-elbow

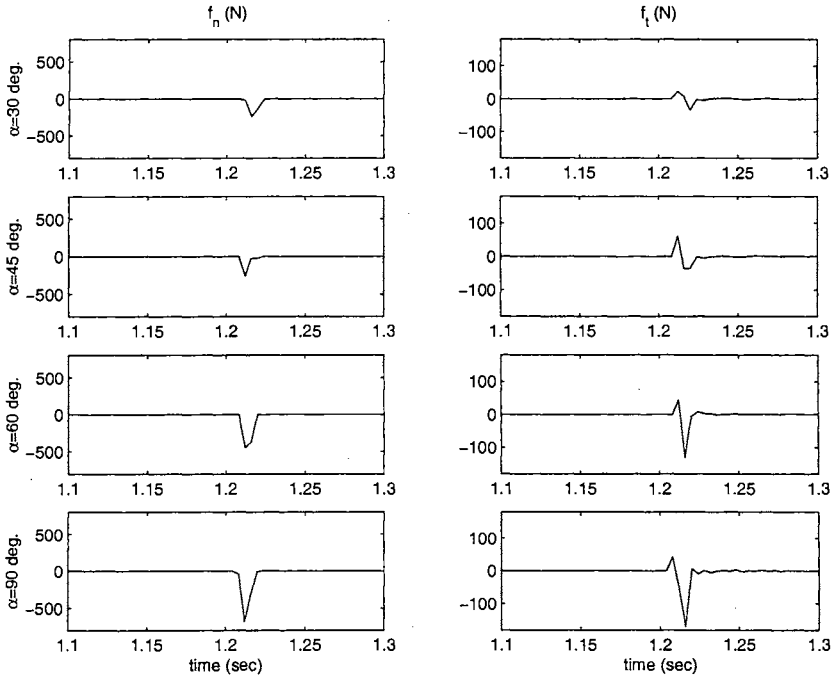

Figure C.11: Impact force, pre-impact velocity $v = 0.3$ m/s, down-elbow

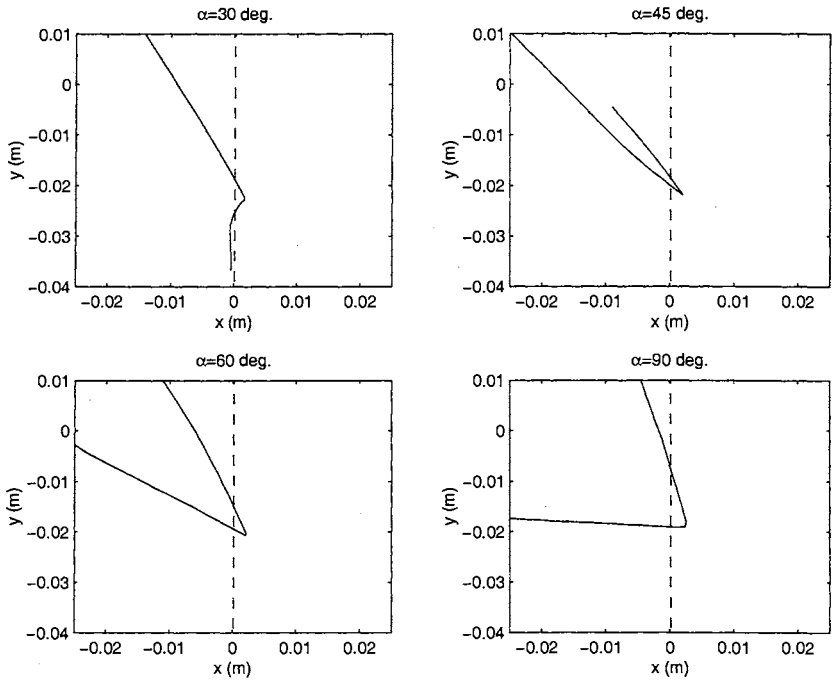

Figure C.12: Robot path, pre-impact velocity $v = 0.3$ m/s, down-elbow

Appendix D

Experiments of Transition Control

	Upward fixture Low speed	Upward fixture High Speed	Downward fixture Low speed
Desired trajectory in Cartesian space	Fig. D.1		
Desired trajectory in joint space	Fig. D.2	Fig. D.3	Fig. D.4
Normal tracking error	Fig. D.5	Fig. D.11	Fig. D.17
Tangential tracking error	Fig. D.6	Fig. D.12	Fig. D.18
Path tracking	Fig. D.7	Fig. D.13	Fig. D.19
Normal force f_n	Fig. D.8	Fig. D.14	Fig. D.20
Tangential force f_t	Fig. D.9	Fig. D.15	Fig. D.21
L_2 norm of f_n, f_t error	Fig. D.10	Fig. D.16	Fig. D.22

Table D.1: Experimental results of transition control

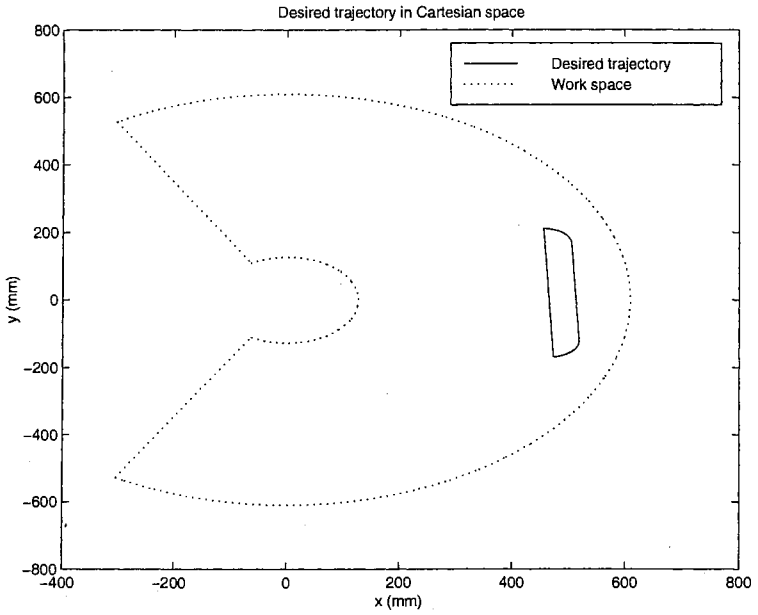

Figure D.1: Desired trajectory in Cartesian space for transition control experiments

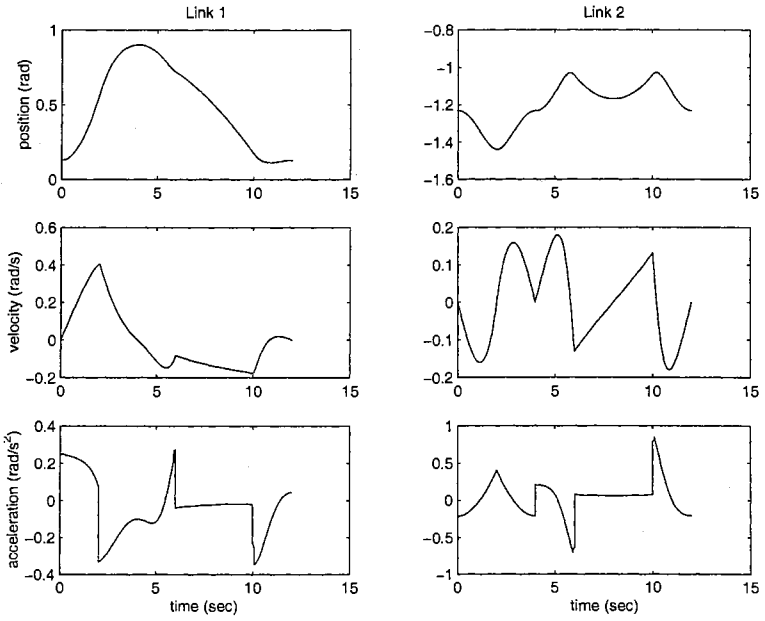

Figure D.2: Desired trajectory in joint space, low speed, upward fixture

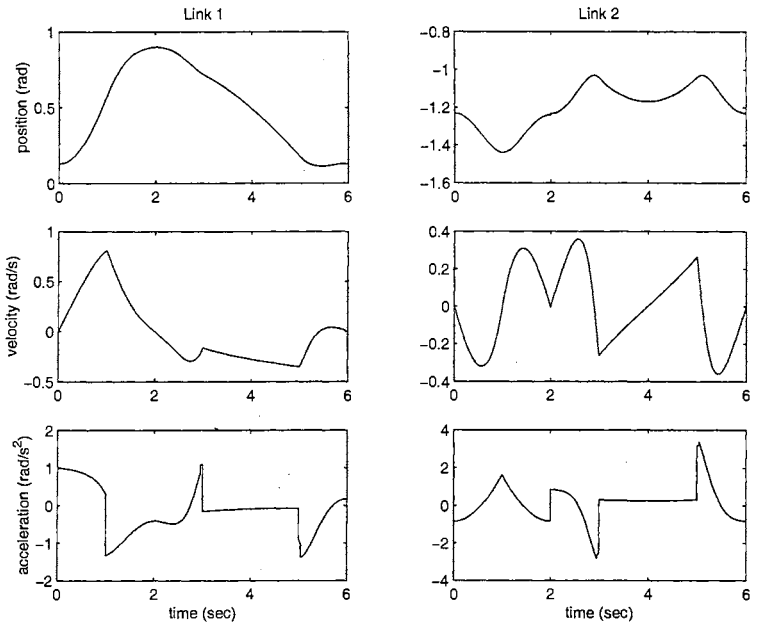

Figure D.3: Desired trajectory in joint space, high speed, upward fixture

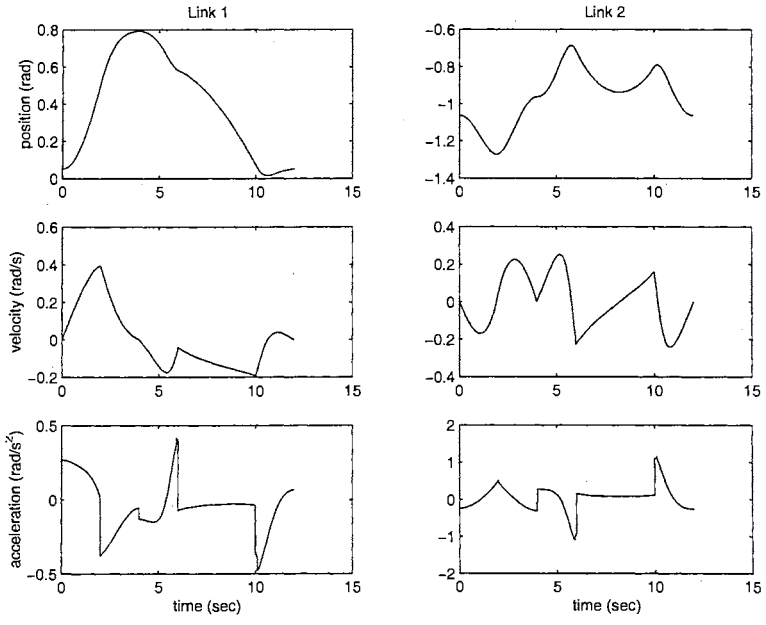

Figure D.4: Desired trajectory in joint space, low speed, downward fixture

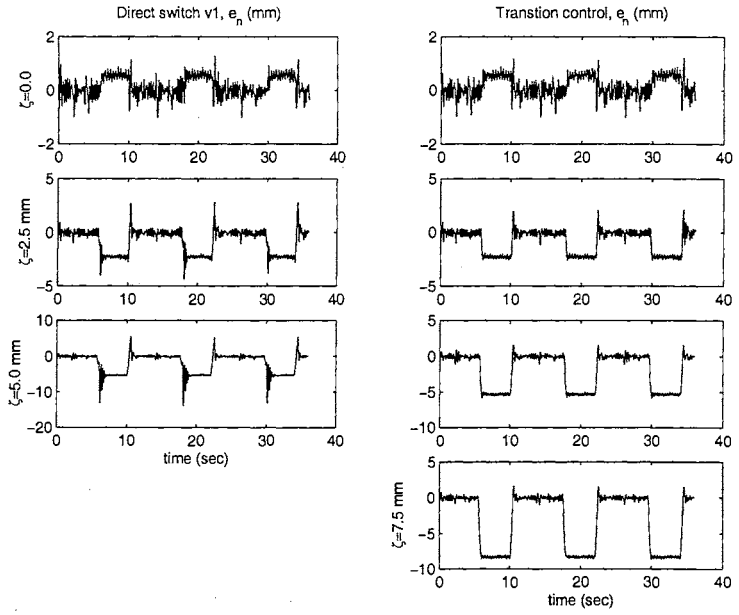

Figure D.5: Normal tracking error, low speed, upward fixture

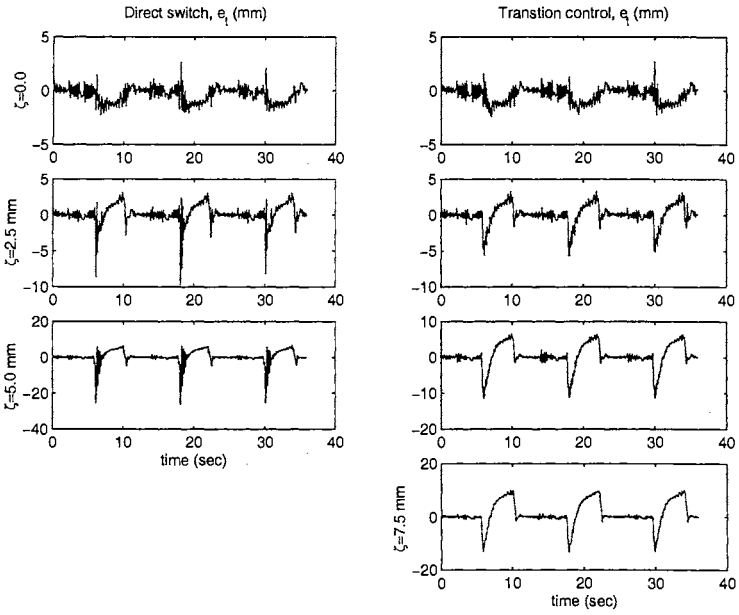

Figure D.6: Tangential tracking error, low speed, upward fixture

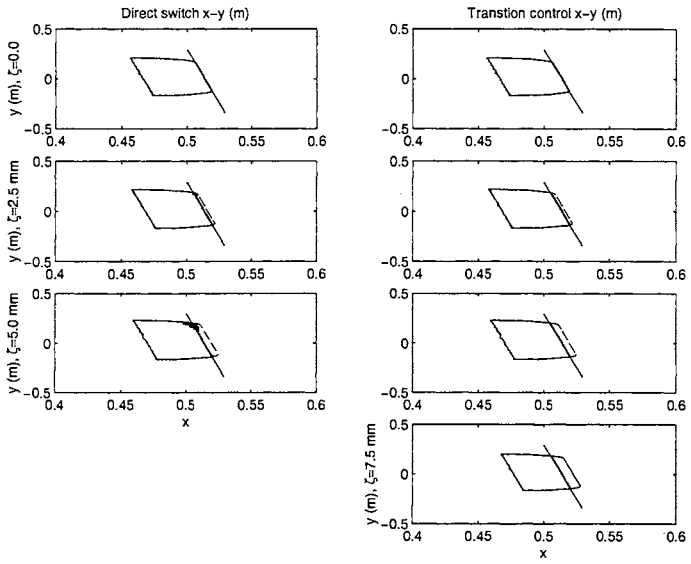

Figure D.7: Trajectory tracking performance, low speed, upward fixture

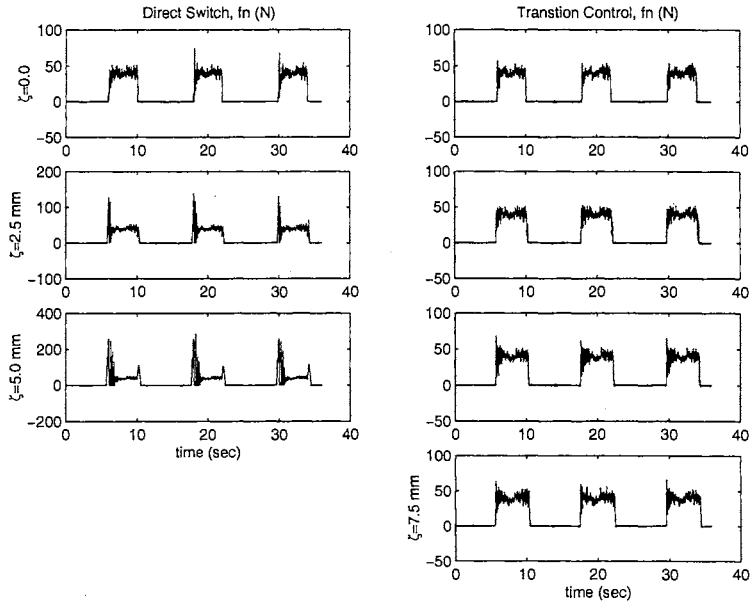

Figure D.8: Normal force, low speed, upward fixture

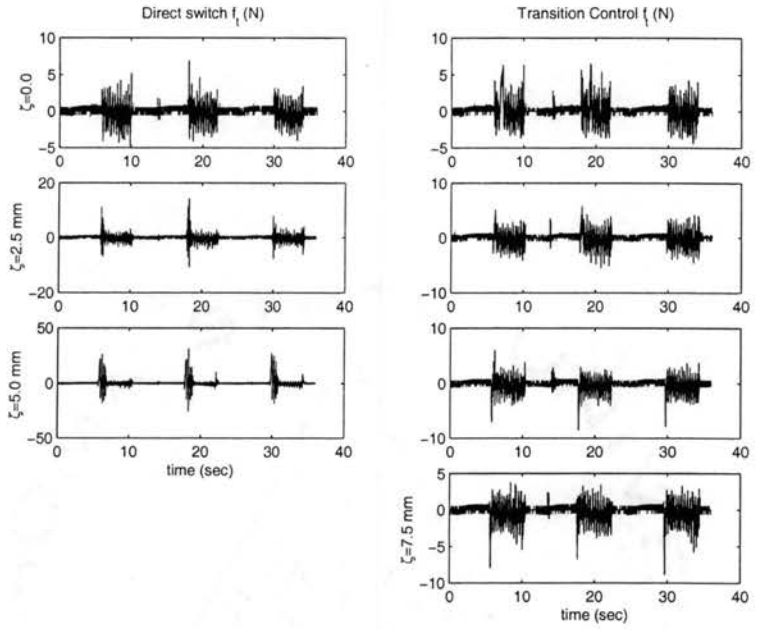

Figure D.9: Tangential force, low speed, upward fixture

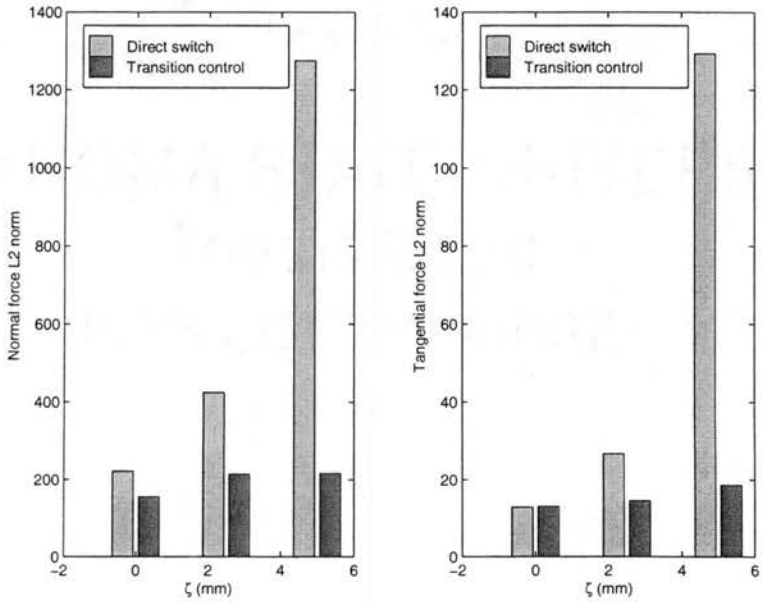

Figure D.10: L_2 norm of contact force errors, low speed, upward fixture

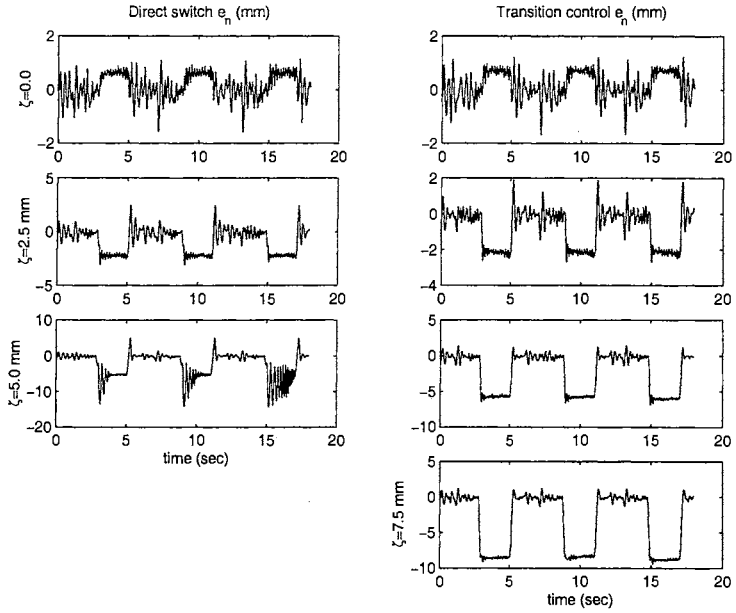

Figure D.11: Normal tracking error, high speed, upward fixture

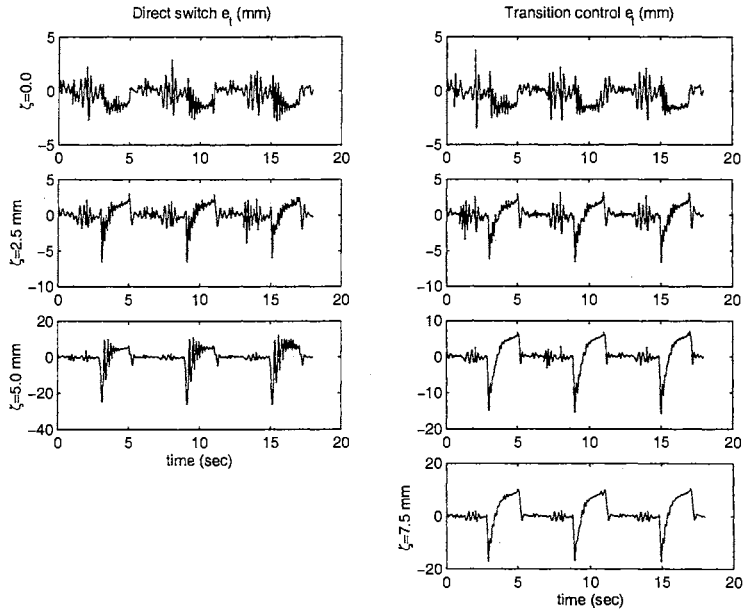

Figure D.12: Tangential tracking error, high speed, upward fixture

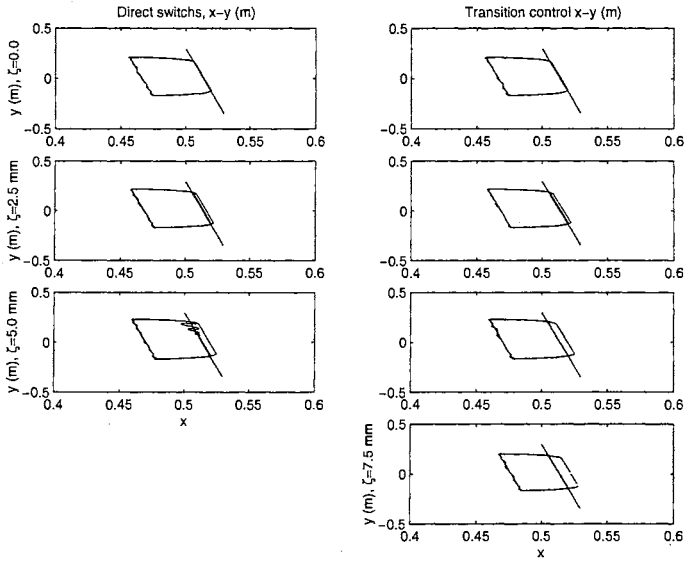

Figure D.13: Trajectory tracking performance, high speed, upward fixture

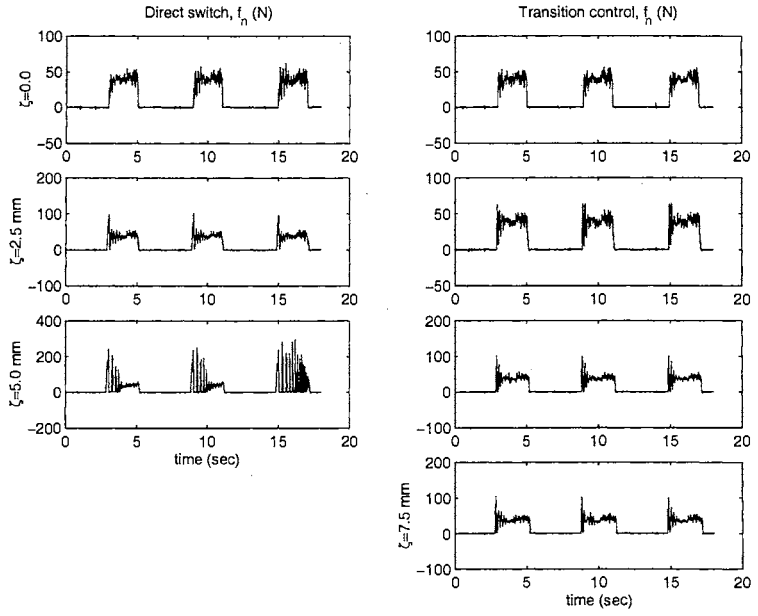

Figure D.14: Normal force, high speed, upward fixture

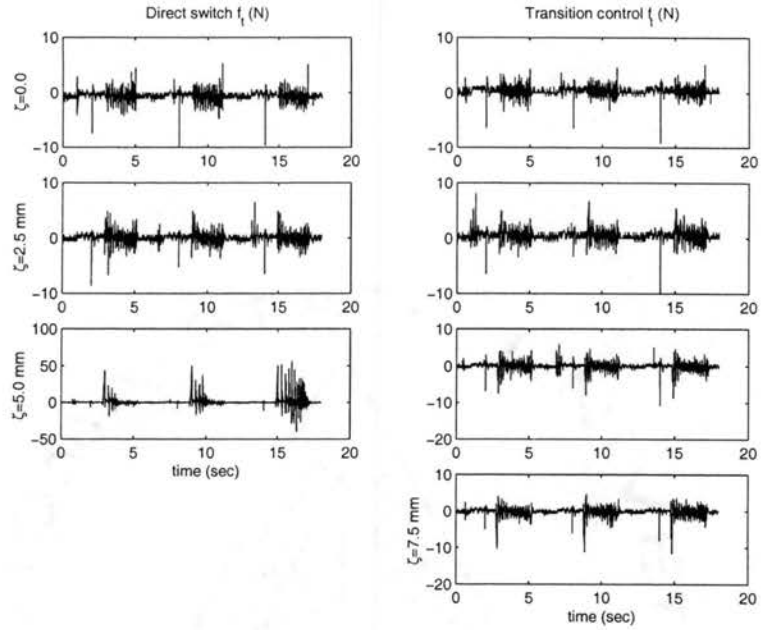

Figure D.15: Tangential force, high speed, upward fixture

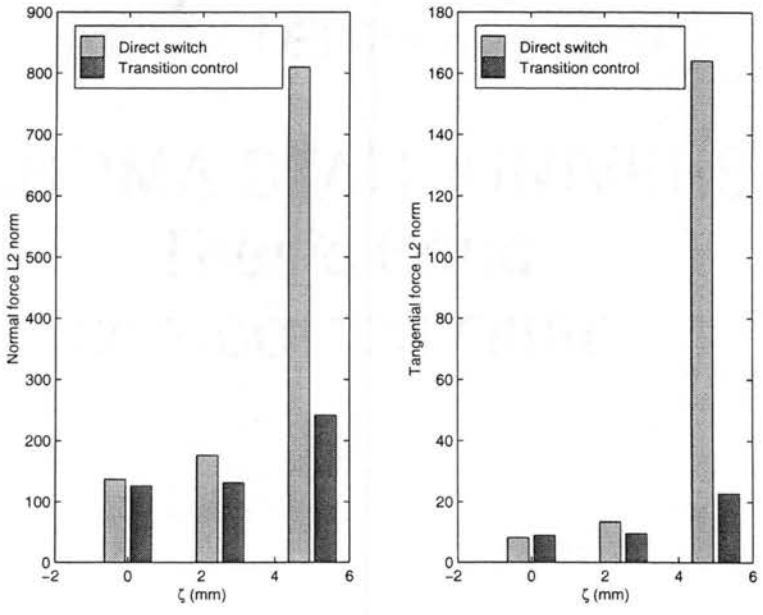

Figure D.16: L_2 norm of contact force errors, high speed, upward fixture

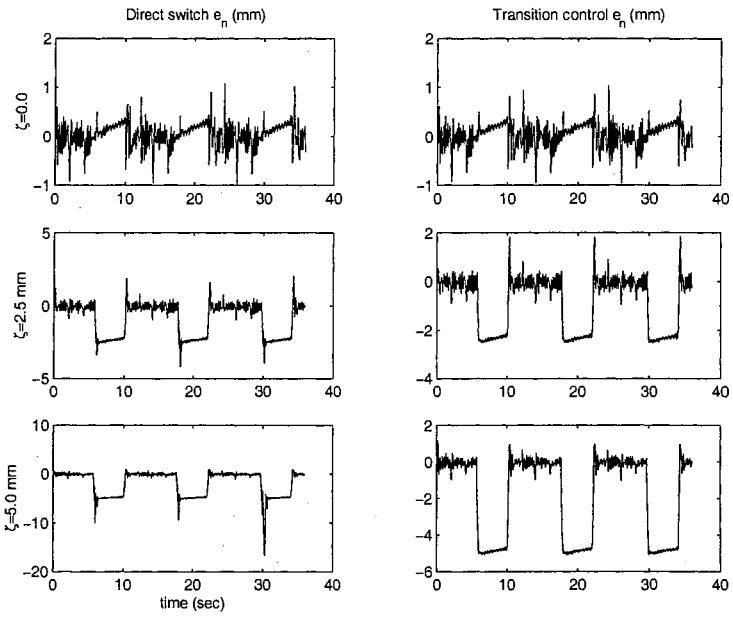

Figure D.17: Normal tracking error, low speed, downward fixture

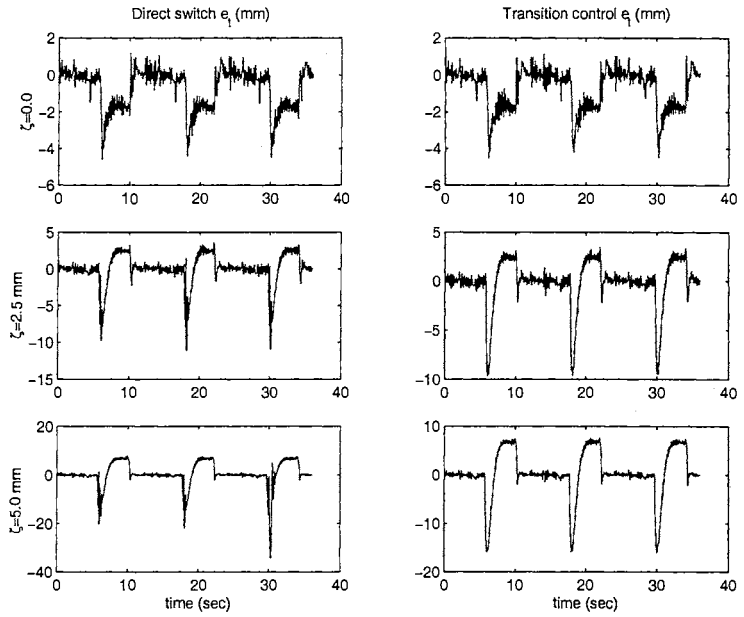

Figure D.18: Tangential tracking error, low speed, downward fixture

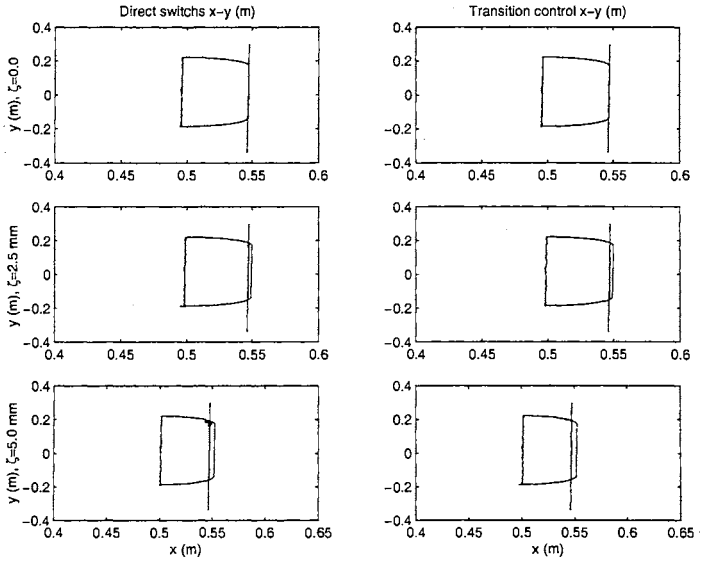

Figure D.19: Trajectory tracking performance, low speed, downward fixture

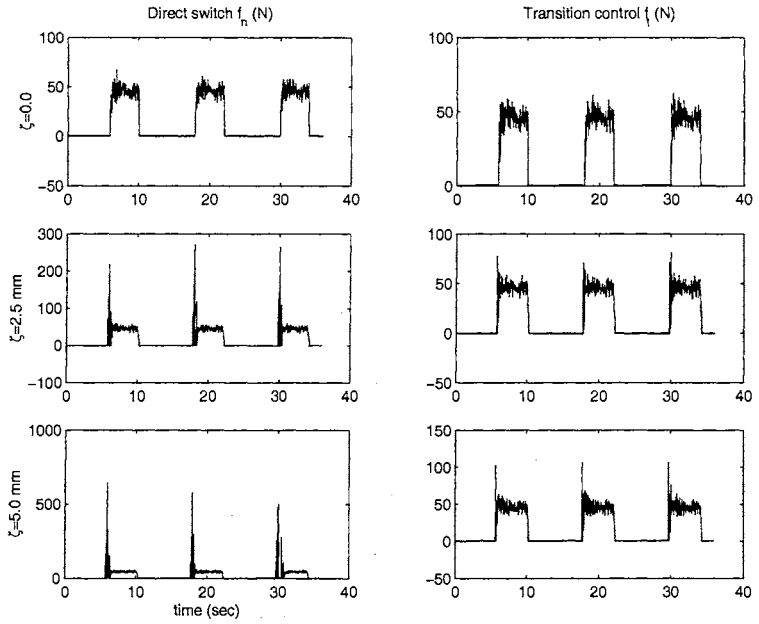

Figure D.20: Normal force, low speed, downward fixture

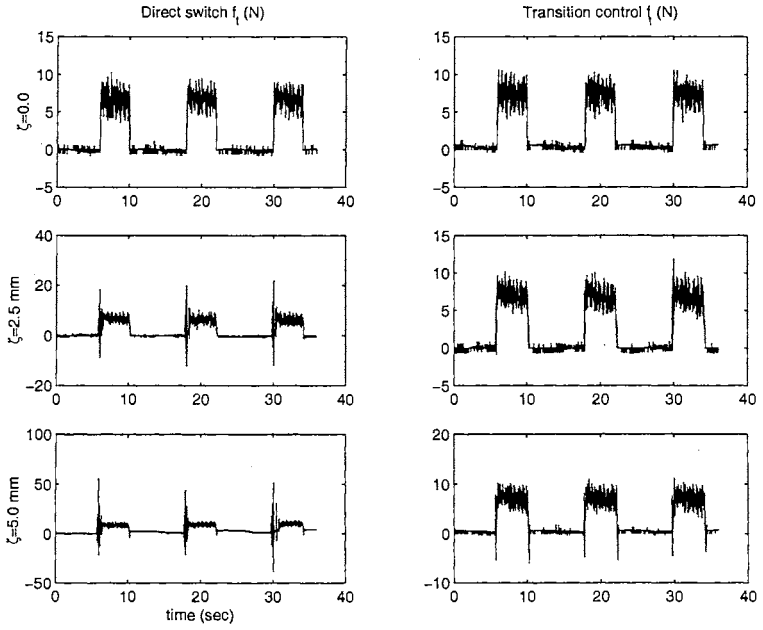

Figure D.21: Tangential force, low speed, downward fixture

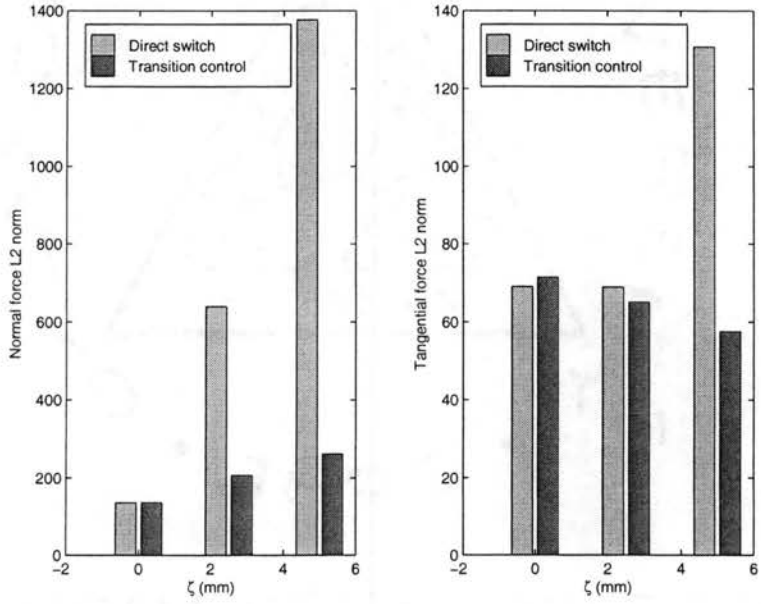

Figure D.22: L_2 norm of contact force errors, low speed, downward fixture

VITA

Biao Yu

Candidate for the Degree of

Doctor of Philosophy

Thesis: MODELING, CONTROL DESIGN AND MECHATRONIC IMPLEMENTATION OF CONSTRAINED ROBOTS FOR SURFACE FINISHING APPLICATIONS

Major Field: Mechancial Engineering

Biographical:

Personal Data: Born in Wuhan, P.R. China, on December 03, 1968, the son of Rendong Yu and Meiye Change.

Education: Graduated from Tongshan High School, Tongshan, Hubei, China; received the B.S. and M.S. degrees from Beijing University of Aeronautics and Astronautics, Beijing, China, South China University of Technology, Guangzhou, China, in 1990 and 1993, in manufacture engineering and mechanical engineering respectively. Completed the requirements for the Doctor of Philosophy degree with a major in Mechanical Engineering and a minor in Electrical and Computer Engineering at Oklahoma State University in December, 2000.

Experience: Research and teach assistant at Oklahoma State University from 1997 to present; Co-founder and manager of R & D of Guangzhou Auto Control Engineering Ltd. from 1994-1996; Product engineer for Yangcheng Science and Technology Inc. from 1992-1994.

Professional Memberships: American Society of Mechanical Engineers, Institute of Electrical an d Electronics Engineers.