CONTROL STRATEGIES FOR ROBOTS IN CONTACT

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF AERONAUTICS & ASTRONAUTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Jaeheung Park
March 2006

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Oussama

Oussama Khatib
Department of Computer Science Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Stephen Rock
Department of Aeronautics & Astronautics

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Claire

Claire Tomlin
Department of Aeronautics & Astronautics

Approved for the University Committee on Graduate Studies.

Preface

In the field of robotics, there is a growing need to provide robots with the ability to interact with complex and unstructured environments. Operations in such environments pose significant challenges in terms of sensing, planning, and control. In particular, it is critical to design control algorithms that account for the dynamics of the robot and environment at multiple contacts. The work in this thesis focuses on the development of a control framework that addresses these issues. The approaches are based on the operational space control framework and estimation methods. By accounting for the dynamics of the robot and environment, modular and systematic methods are developed for robots interacting with the environment at multiple locations. The proposed force control approach demonstrates high performance in the presence of uncertainties. Building on this basic capability, new control algorithms have been developed for haptic teleoperation, multi-contact interaction with the environment, and whole body motion of non-fixed based robots. These control strategies have been experimentally validated through simulations and implementations on physical robots. The results demonstrate the effectiveness of the new control structure and its robustness to uncertainties. The contact control strategies presented in this thesis are expected to contribute to the needs in advanced controller design for humanoid and other complex robots interacting with their environments.

Acknowledgements

As I finish the final draft, so many people come to my mind. My adviser, Professor Oussama Khatib, has taught me *robotics* since the beginning of my Ph.D program when I knew nothing about the subject. He has always encouraged my research with endless effort and care. Professor Stephen Rock and Claire Tomlin were the first professors I met at Stanford, and have been my constant guides through my Ph.D.

In fact, I have to admit that I probably owe the most to the people in my research group, the manipulation group in Stanford AI lab. I'd like to thank Vince De Sapio for so many valuable research discussions and help with writing. To Peter Thaulad, with whom I spent the most time, I owe every experiment that I did at Stanford. From Rui Cortesão, I learned so much when we conducted research. Thanks to James Warren, Francois Conti, Luis Sentis, Irena Pashchenko, and Mike Zinn, who bore with me through good times and bad. Thanks also go to Costantinos Stratelos, Mike Stilman, Probal Mitra, and the new lab members: Anya Petrovskaya, Dongjun Shin, Jinsung Kown – good luck to all of them. Many thanks to KAASA members (Korean Aero/Astro Stanford Association) and their family. They have been the people who made me feel at home.

Finally, I cannot thank my family and in-laws enough for their generous support. *Special* thanks to my parents, who have cared for me my whole life. Thanks to Yunjung for help with illustrations. Another gift I've gotten along the way is my own little family - Yunhyung, Kevin and Bryan. My wife, Yunhyung, always has kept me smiling and happy. Words cannot express my gratitude to her, for I could never have reached this point without her. My big boys, Kevin and Bryan, have made my life at Stanford happier and richer.

Contents

List of Figures

Chapter 1

Introduction

In the past three decades conventional robotic systems have been employed in various industries. In most cases these systems have consisted of manipulator arms whose function has been to perform desired tasks using their end-effectors. These tasks have involved programming end-effector motion trajectories and then controlling individual joints to produce the desired motion at the end-effector. This has been accomplished by using kinematic relationships between the end-effector pose and the joint angles and by including the dynamic properties of the system in addition to the kinematics. A characteristic of many of these tasks is that they only involve motion in free space. That is, the end-effector and the manipulator links are not in contact with the environment.

With the increasing complexity of tasks that are required of manipulators contact with the environment has become more common (Figure 1.1). Additionally, applications for robots have expanded beyond traditional industrial settings into human environments. Physical interaction between robots, humans, and the environment at large is no longer a rare occurrence to be avoided but a common operating condition. Consequently, control strategies that deal with these situations are essential to safely achieving desired goals.

The motion of a robot in contact with the environment is often referred to as *constrained motion* in the sense that the motion is not free but rather constrained by the environment [9]. Due to these environmental constraints the control schemes

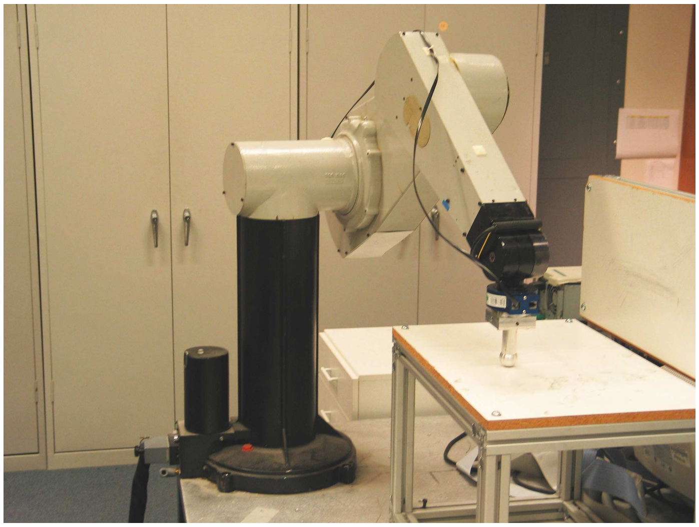


Figure 1.1: **A PUMA560 manipulator in contact.**

that have been adopted are often referred to as *compliant motion* control strategies since the robot must be controlled in a manner that is responsive and compliant to the environment [42]. The goal of these control strategies is to successfully perform the desired tasks without compromising the robot or the environment. While being constrained by the environment the robot must control the motion and contact forces simultaneously and responsively.

One difficulty in controlling robots in contact involves maintaining stability. In particular, instability arises during contact with stiff environments [6]. Since the system response to contact forces must be fast in this case, sampling time is often a limitation when implementing a stable controller. Also, the environment that is in contact with the robot is typically not easily modeled. This results in modeling uncertainties which cause the controller to perform inconsistently in response to small changes in the contact environment or the robot itself. These factors motivate the need for a robust controller.

Given this motivation, a constrained motion controller using a hybrid motion/force control strategy is proposed in this thesis to overcome these difficulties. The hybrid motion/force control structure for the end-effector in [31] provides a control structure that decouples the dynamics of the motion and the contact forces. Using this framework, this thesis presents the design of a controller that is robust to modelling uncertainties including environment stiffness. Active Observers [11] are applied for contact force control in this design. This is a model reference approach using full state feedback with a Kalman estimator. Robust contact force control is achieved using this approach and the implementation has been demonstrated on a PUMA560 robot at the Stanford Artificial Intelligence Laboratory.

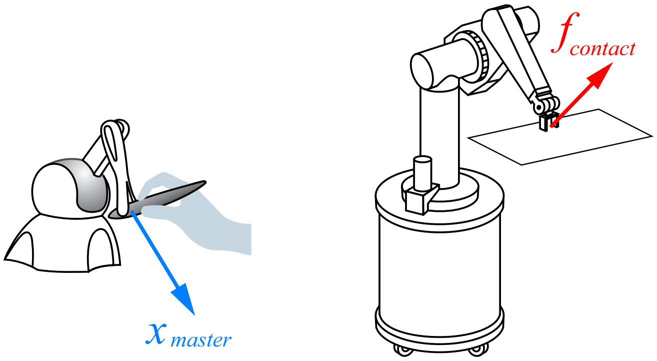


Figure 1.2: **Haptic teleoperation system.** A haptic master device is controlled by a human operator to control a remote slave robot.

One application for this contact force controller is in the area of haptic teleoperation (Figure 1.2). A basic teleoperation system provides control to a remote robotic device through a master controller. The applications of such systems are broad and include space robotics, surgical robotics, and service in nuclear power plants and other hazardous environments. A typical teleoperation system consists of visual feedback to the user from the remote environment and a control interface that allows the user to command the robot through desired position or velocity commands.

When remotely manipulating objects, visual feedback is typically not sufficient for fine and precise manipulation. This fact hinders the speed at which teleoperation tasks can be performed. To mitigate this performance limitations, force reflection is introduced to provide the user with additional feedback. It has been shown that this additional feedback improves the teleoperation task by providing significantly greater precision [21]. *Transparency* refers to the degree to which the force reflection felt by the user at the master device emulates the actual feeling associated with touching the object directly at the remote location. Good transparency implies that the operator attains *tele-presence* with the remote environment; that is, the operator feels as though he or she is actually in the remote environment.

Force reflected teleoperation is fundamentally dependent on an effective compliant motion strategy since the slave robot is expected to make contact with objects in the environment as the user guides it from the master device. A major difficulty with such systems is the trade-off between system performance and stability when force reflection is provided to the user [38]. The criteria for evaluating this performance include how well the slave robot tracks the master robot in free space and how accurately force reflection is provided to the user when the slave robot is in contact with the remote environment. In particular, when the slave robot makes contact the overall system stability, not just the remote system stability, is critical. This is due to the fact that the master and slave systems are connected.

To improve performance while maintaining stability the proposed teleoperation approach uses contact force control on the slave manipulator. This enhances the stability characteristics of the slave robot in contact with unknown objects, as well as the stability of the overall system. A virtual spring is used to connect the master and slave systems such that the slave generates the desired contact force. This contact force is proportional to the relative position difference in the two systems. The user is provided with high fidelity force reflection by having the contact force controller track the desired force on the slave and producing the desired force values on the master device.

The discussion so far has assumed a single contact between the robot and the environment. Another challenge in compliant motion control involves the scenario of multiple contacts. Robotic systems are becoming increasingly complicated with more joints needed to perform complex and subtle motion tasks. Thus, there is a greater likelihood that robots will encounter multiple contacts on a single link or on multiple links. An example of this is illustrated in Figure 1.3, where a humanoid robot is depicted working in a human environment.

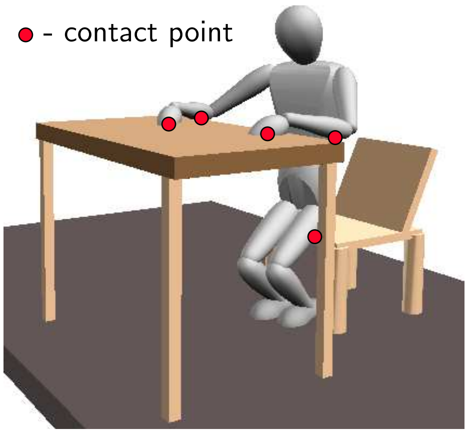


Figure 1.3: **A humanoid robot in multiple contact.**

In such cases, an appropriate compliant motion strategy is necessary, which is able to generate compliant motion over all the contacts. This implies that the controller should be able to control the motion of the manipulator while maintaining all the contacts as desired. Since motion control and force control in the presence of contacts influence each other, a framework that can decouple these controls is imperative to achieving high performance.

The hybrid position/force control framework referred to earlier for a single contact decouples the dynamics and control structures for the end-effector. To deal with multiple contacts over multiple links, however, the framework needs to be generalized. This generalization is initiated by constructing an operational space coordinate associated with each contact normal direction. The contact forces can then be controlled individually in a decoupled manner. The operational space coordinate for motion control can be augmented to the operational space coordinate for force control. Alternatively, the motion of the manipulator can be controlled in the null space of the contact force control space.

As mentioned earlier, humanoid robots often involve multiple contacts. Since Honda released its first humanoid robot [23] two decades ago, a great deal of research effort has been placed on developing humanoid systems; specifically on developing humanoid *walking* controllers. As a result, most current humanoid robots can walk, but not necessarily in a human-like manner. While control strategies have been developed for behaviors other than walking, these have been limited to specialized behaviors. Therefore, a more general strategy is sought for the whole body control of a humanoid robot.

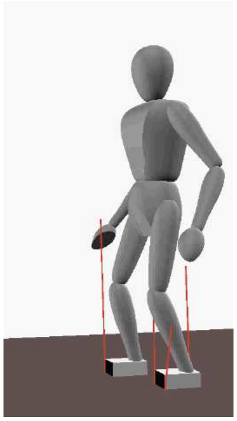


Figure 1.4: **A humanoid robot in contact with the ground.**

Humanoid systems intrinsically require contact with the environment, in particular the ground, in order to achieve stability (Figure 1.4). Therefore, it is critical to correctly deal with contacts when controlling the system. An approach proposed in this thesis is the *contact consistent control framework*. In this framework contact forces are accounted for in the composition of the control torque. That is, the motion/force control of the whole body, including the dynamics of the environment in contact, is considered. In this way the contact forces on the feet are treated as internal forces. The contact forces are, therefore, not explicitly controlled, but generated consistent with the loop closure between the robot and the ground. The contact consistent control framework successfully integrates the generalized hybrid motion/force control structure such that the motion and contact forces are controlled in the same manner as a fixed based manipulator. Figure 1.5 depicts a *climbing* scenario that is presented in this thesis. The robot encounters numerous contacts on both its hands and feet while the motion is controlled to move upward.

The results of implementing this contact consistent control are generated in the

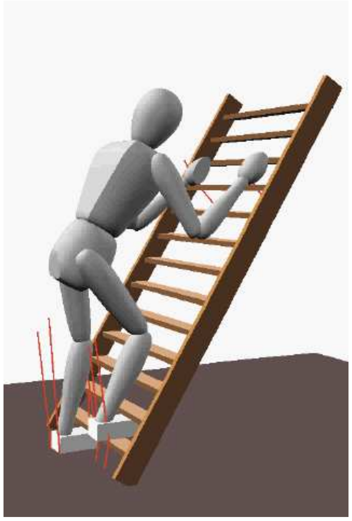


Figure 1.5: **A humanoid robot climbing.**

SAI simulation environment [32]. SAI has been developed in the Stanford AI Laboratory over the past five years. The simulations provide forward dynamics integration, multiple contact resolution, and a graphical user interface/display. The interactive nature of the environment facilitates the development of controllers and the testing of different situations by the user.

To summarize this work, the main contribution has been the development of control frameworks for robots operating at multiple contacts. Specifically, the proposed contact force control approach incorporates a modified Kalman observer (AOB) into the operational space control framework. The experimental results demonstrate its performance and robustness to modelling errors and unknown disturbances. A new haptic teleoperation approach is developed based on this contact force control. The end-effector of the slave robot is rendered transparent by compensating for the highly nonlinear slave dynamics using the force control. The use of a virtual spring and online stiffness estimation provides the teleoperation system with robustness with respect to communication time delay and abrupt changes of the contact environment. Next, the motion/force control framework is further generalized to control contact forces at multiple links and motion while maintaining contact. This generalization is accomplished by constructing an operational space coordinate associated with each contact normal direction. Each contact force can then be controlled independently without affecting each other. Lastly, a contact consistent control framework is proposed for non-fixed base robots such as humanoids. It derives the dynamics of the system by considering a robot in contact with the environment as a constrained system. The constrained dynamics are utilized in the composition of control for the whole body behavior of the robot. This framework enables us to apply the motion/force control framework to the non-fixed base robot. Therefore, the whole body motion can be generated by designing the trajectories not only in joint space but also in task space.

Chapter 2

Contact force control

Robot manipulators must often make contact with the environment when executing tasks. Properly controlling robots in contact is important not only to the successful achievement of the task but also to the mutual safety of the robot, environment, and most importantly any human present in the environment. Since most robots are designed to follow motion trajectories accurately and with high bandwidth, the resulting motion is characteristically *stiff*. This is due to the need for good disturbance rejection characteristics in the presence of unexpected external forces. When a robot is in contact with the environment, however, motion control is not always sufficiently precise due to uncertainties in the models associated with the robot and the environment. Consequently, motion control alone is not sufficient to successfully control a robot in contact, even with detailed environment information. Therefore, compliant motion control strategies are necessary, not only for controlling the contact forces but also to ensure safe interaction with the environment.

Compliant motion control strategies can be categorized into two main areas. These are *indirect* force control and *direct* force control [60, 10, 67]. Indirect force control seeks to create a desirable compliance/impedance state at the robot contact point, most typically the end-effector, in the contact directions. That is, if the robot is in point contact with a plane, the normal direction of the plane would be chosen to be a compliant direction [51] (Figure 2.1). This approach is implemented by choosing the position control gains in the end-effector space differently depending on the directions that one is interested in. For example, the proportional gains for the compliant direction would be chosen to have a desired stiffness and the gains for the other directions would be chosen to achieve the desired motion control bandwidth. This approach is referred to as stiffness control [54] for static behaviors and impedance control [24] for dynamic behaviors. Also, we differentiate between *active* versus *passive* compliant motion control depending on whether an external force sensor is used in generating the desired compliance [57].

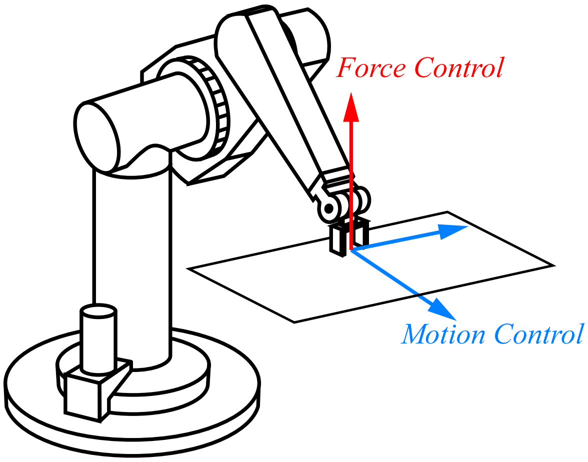


Figure 2.1: **Compliant frame selection.**

Directly controlling the contact forces is different than indirect force control in the sense that the control law is based on the errors between the desired and measured contact forces rather than position errors. Impedance/stiffness control does not explicitly control the contact forces. However, given perfect knowledge of the environment, the desired contact force can be generated by composing the goal position using the environment stiffness. In reality, perfect knowledge is not obtained and it may be necessary to specify contact forces that are to be precisely tracked. In this case different control approaches may be needed, that is, an approach which controls the contact forces directly.

One approach in direct force control involves designing an external feedback loop around the position control loop [57]. This takes advantage of the existing motion control and uses the measurement of the contact forces to correct the input to the motion controller. Design of the external force controller is accomplished by incorporating a model of the environmental stiffness. Using this stiffness model an appropriate position command can be determined to achieve the desired contact forces. Thus, knowledge of the contact environment is critical for proper controller design. In practice, a simple linear spring model is used as a model of the environment since most of the objects in the contact environment are passive static systems [60].

Another direct force control approach is the hybrid position/force control strategy [51, 42]. This approach has two separate control loops providing position control and force control, respectively. The original approach for this control did not fully utilize the dynamic properties at the end-effector; thus, it could not provide a decoupled control structure for each position and force control loop. The operational space approach [31] extended the concept of hybrid position/force control by decoupling the controllers. Since this approach provides a dynamically decoupled control system with feedback linearization, there is more flexibility to choose different control subsystems in the controller design. This approach is used to compose the motion and force controllers in this thesis.

In all of the force control strategies discussed thus far robustness to unknown disturbances and modeling errors is an important and challenging issue. This is also true of stability in contact with stiff environments. Typical direct force controller designs use proportional-integral (PI) control to ensure zero steady state error. A simple interaction model, with an estimated environmental stiffness, is included as a parameter of the controller. Based on this environmental stiffness and the desired bandwidth, the PI gains are adjusted. However, these controllers tend to be very sensitive to disturbances or mismatch in the models, resulting in poor performance or instability in tracking contact forces.

This problem is exacerbated when the environmental stiffness is very high, as it is harder to estimate the real stiffness. Consequently, error in stiffness matching often leads to instability. Also, because of the stiff environment and sensor characteristics, the system response of the contact force is fast compared to the servo rate of the robot controller. This factor further limits the performance of the force controller. Therefore, one of the most important characteristics needed in force control is robustness with respect to modeling errors, especially errors in environment stiffness.

The approach proposed in this thesis is to implement the Active Observer (AOB) design [11] in the operational space control framework. The hybrid motion/force control structure along with the operational space framework realizes decoupled linearized sub-systems through the nonlinear dynamic decoupling [19, 31]. This control architecture allows the specific linear control schemes to be applied individually. The nonlinear dynamic coupling method for robots is effective since inaccuracies of the model used for decoupling have a minor effect compared to unknown disturbances, unmodeled friction, and parameter errors in the environment model. The AOB design is then applied on each linear second order system to deal with these uncertainties.

The AOB design uses a Kalman observer and full state feedback. The model in this Kalman observer includes an additional state, which is an equivalent disturbance at the input command, due to unmodeled dynamics, parameter mismatches, and unknown disturbance. This state is referred to as *active* since the estimated values are directly canceled at the input to the system. This active role forces the system response to closely match the desired closed-loop system response. The active state enables us to increase bandwidth of the system by choosing higher feedback gains, resulting in higher tracking performance. Therefore, the AOB design realizes a model reference control approach [1]. By *model reference control* we mean a control scheme which adaptively follows the desired model of the system response rather than simply tracking a reference trajectory. Another interpretation of the AOB design is to regard it as a disturbance accommodation technique [27]. It allows the existence of input disturbance and compensates for the disturbance by adding its estimate to the input command.

The development of the force controller within the operational space formulation is the main focus of this chapter. The experiments for force control in this chapter are conducted for one point contact at the end-effector. That is, the force control direction is selected to be a normal direction to the contact surface. Other translational directions and orientation are controlled by motion control. Significant improvement on the performance and robustness have been demonstrated over PID controllers through these experiments.

This chapter begins with a brief explanation of the hybrid motion/force control structure. Force control design with the AOB is then discussed to illustrate the design procedure and also to explain how the active state works. Sensitivity to mismatches in stiffness is analyzed and the performance is demonstrated with a PUMA560 robot making contact with different stiffnesses.

2.1 The operational space formulation

2.1.1 Dynamics of the robot in contact

The dynamic equations of a manipulator in free space are described by

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) = \Gamma, \tag{2.1}$$

where q, $A(q)$, $b(q,\dot{q})$, $g(q)$, and Γ are the vector of joint angles, the mass/inertia matrix, the Coriolis/centrifugal torque, the gravity torque in joint space, and the vector of joint torques, respectively. When the end-effector of the robot is in contact, the dynamics of the manipulator include the contact forces at the end-effector. That is,

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) + J^T f_c = \Gamma, \tag{2.2}$$

where J is the Jacobian corresponding to the end-effector and the vector, f_c, is the contact force/moment at the end-effector.

To control the motion and contact force at the end-effector, while compensating for the dynamic effects of the robot, an operational space description of the dynamics is required. We define the Jacobian, J, to correspond to the instantaneous linear/angular velocity, ϑ, of the end-effector. That is,

$$\vartheta = J\dot{q}. \tag{2.3}$$

The dynamics of the end-effector can then be obtained by projecting Equation (2.2) into an operational space specified as the end-effector space [31]. This yields

$$\Lambda(q)\dot{\vartheta} + \mu(q,\dot{q}) + p(q) + f_c = F \tag{2.4}$$

$$\begin{aligned} \Lambda(q) &= (JA^{-1}J^T)^{-1} && (2.5)\\ \mu(q,\dot{q}) &= \Lambda(JA^{-1}b - \dot{J}\dot{q}) && (2.6)\\ p(q) &= \Lambda JA^{-1}g && (2.7) \end{aligned}$$

where $\Lambda(q)$, $\mu(q,\dot{q})$, and $p(q)$ are the inertia matrix, the vector of Coriolis/centrifugal forces, and the vector of gravity forces in operational space, respectively.

2.1.2 Decoupled control structure

The control force, F, in Equation (2.4), can be composed to provide a decoupled control structure by choosing

$$F = \hat{\Lambda}(q)f^* + \hat{\mu}(q,\dot{q}) + \hat{p}(q) + \hat{f}_c \tag{2.8}$$

where the $\hat{.}$ denotes estimates of the quantities. Furthermore, to select the force control and motion control directions, the generalized selection matrices, Ω_f and Ω_m, are used in composing f^*. Raibert and Craig [51] introduced a selection matrix to select force and motion control directions in the Cartesian global frame. A generalized selection matrix was presented in [31]. The generalized selection matrix selects the directions in the contact frame. Chapter 4 describes the selection matrix for the case of multi-contact.

$$f^* = \Omega_m f_m^* + \Omega_f f_f^* \tag{2.9}$$

For the experimental setup of one point contact, the matrix, Ω_f, is chosen to select the normal direction to the contact surface. And the matrix, Ω_m, selects the other translational directions and orientation. In the case that the contact normal direction is the vertical direction in the global frame, the selection matrices are

$$\Omega_f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \Omega_m = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.10}$$

This results in decoupled second order equations in both the force and motion control directions,

$$\ddot{x}_m = f_m^* \tag{2.11}$$

$$\ddot{x}_f = f_f^* \tag{2.12}$$

The composition of the control input, f_m^*, for desired motion can be accomplished by using a linear control method such as PD or PID control. However, the control input for contact force, f_f^*, requires the relationship between motion, x_f, and contact force, f_c. A model of the relation between motion and contact force is described in the following subsection. Subsequently, the generation of the control input, f_f^*, using the model is presented.

After composing proper control inputs, f_m^* and f_f^*, for the decoupled linearized systems, the control force, F, is computed using Equation (2.8). To generate the control force, F, the control torque to the robot is selected as

$$\Gamma = J^T F + N^T \Gamma_0 \tag{2.13}$$

$$N^T = I - J^T \bar{J}^T \tag{2.14}$$
$$\bar{J}^T = \Lambda J A^{-1} \tag{2.15}$$

where N^T is the dynamically consistent null space projection matrix and $\bar{J}$ is the dynamically consistent inverse of J [31]. The dynamically consistent inverse is a generalized inverse that results when the number of rows of the matrix is smaller than the number of columns. Additionally, the dynamically consistent inverse uses the inertia matrix, A, as a weighting term. The first term, $J^T F$, in Equation (2.13) generates the control force, F, on the end-effector and the second term, $N^T \Gamma_0$, is the control in the null space of the end-effector control. The pre-multiplication by the null space projection matrix, N^T, guarantees that this null space control torque, Γ_0, will not generate any force on the end-effector. The block diagram of the operational space control framework is shown in Figure 2.2. In the case of the 6 DOF PUMA560 manipulator, there is no null space if the 6 DOF end-effector is fully controlled by motion and force control.

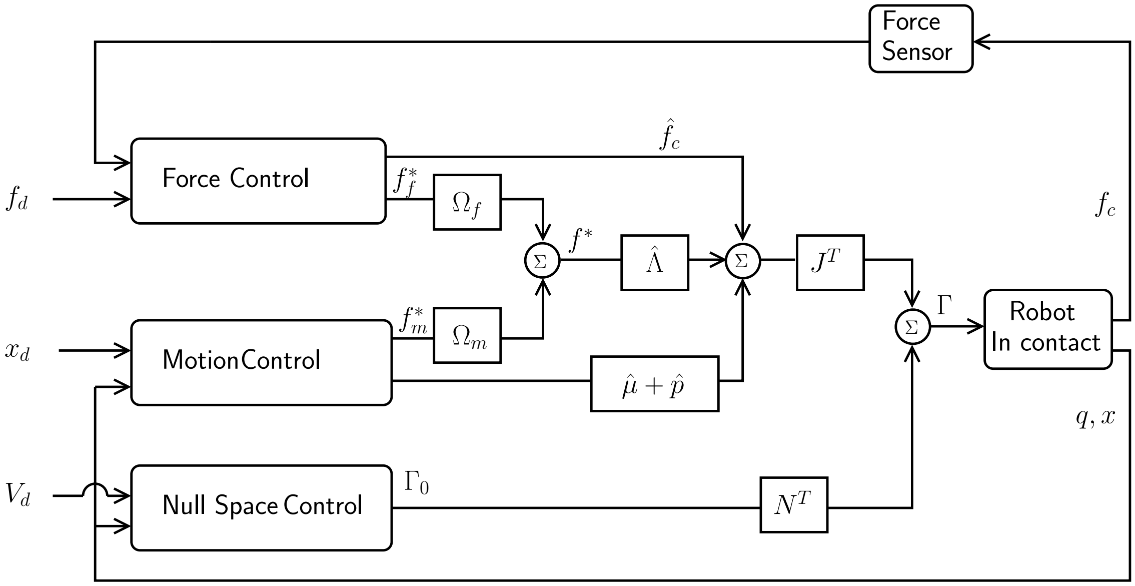


Figure 2.2: **The operational space control framework for a manipulator.**

2.1.3 A linear spring model for the contact environment

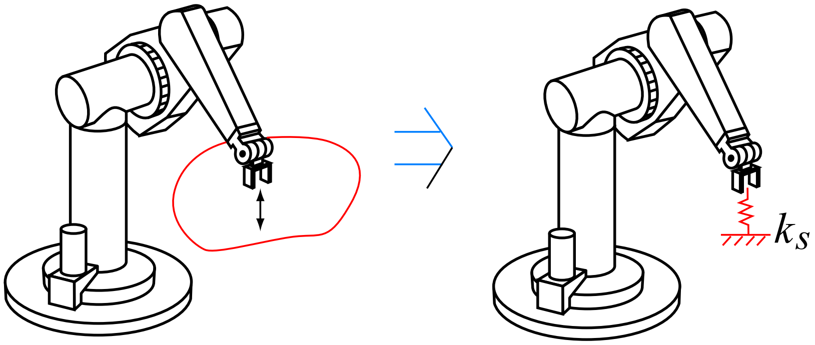


Figure 2.3: **Model of the contact environment.**

The design of force control involves composing the control input, f_f^*, in Equation (2.12). As part of the control design it is necessary to know the relation between the contact force and the motion of the end-effector in the force control direction. In practice it may not be possible to identify a precise mathematical model for the actual contact environment. However, a simple spring model [33] can be used for the controller design. In this case the environment is assumed to have a constant stiffness (Figure 2.3). Although this model seems too simple to represent the environment for control purposes, it captures the important characteristic that contact force on most passive objects increases with deflection.

A higher order model for passive environments is a second order model with mass, damping, and stiffness. The linear spring model is a special case of this model. When the stiffness of the contact object is identified, adding a mass property to the model makes the system slower. Therefore, the simple linear spring model can be considered a conservative model in terms of stability. The use of a linear spring model on the actual second order system may decrease the performance. So, the proposed approach is to utilize the stiffness model and design a controller. Then, the performance issues will be compensated for by an adaptive controller using AOB.

For each contact i, we use the stiffness model

$$\dot{f}_{c,i} = k_{s,i}\vartheta_{c,i}, \tag{2.16}$$

where $f_{c,i}$ is the ith contact force. The term $\vartheta_{c,i}$ is the instantaneous velocity in the contact normal direction and $k_{s,i}$ is the ith contact environment stiffness.

With this model and Equation (2.12), the equations of motion for each contact, i, are

$$\ddot{f}_{c,i} = k_{s,i} f^{*}_{f,i}. \tag{2.17}$$

2.2 Contact force control design

A common approach for contact force control involves the use of a proportional-integral (PI) controller with damping based on the velocity of the end-effector. One of the main difficulties with this approach involves hard contact. In this case, the dynamics of contact with the environment are already very fast, so there is a limitation in the proportional gain that can be employed. Thus, the proportional gain must be kept small, which in practice results in large steady state error. This error can be reduced by adding integral control, however, this is problematic since it may adversely affect the stability of the system.

In addition to this difficulty associated with classical PI controllers, the stiffness of the environment is difficult to identify and may even change during contact when deflection occurs. Classical PI controllers cannot deal with these difficulties since they do not account for uncertainties in the system. These facts motivate a force control strategy which employs an observer that can account for uncertainties in a systematic way.

Active Observers (AOB) [11] use a modified Kalman estimator with an additional state, called an active state. The active state is the estimate of the disturbance to the input of the system. Full state feedback is implemented with estimated states that correspond to the contact force and the derivative of the contact force. In addition, the estimated input disturbance (active state) is directly subtracted from the input to compensate for the error. This AOB method is best applied to systems which can be modeled as linear systems with input disturbance. The linearized contact force control system is one such system. In this case feedback linearization is achieved through the use of the operational space formulation. A simple spring model is used for the environment and as such modeling uncertainties need to be considered. In addition to these modeling uncertainties most robots cannot accurately provide the commanded torque to the system and this mismatch between commanded torque and actual torque can be treated as an input disturbance.

2.2.1 Discretized system plant

We begin our force control design with the second order dynamic equation associated with each contact, Equation (2.17). When there is a dead-time, T_d, (delay of the input command)

$$\ddot{f}_{c,i} = k_{s,i} f^*_{c,i} e^{-sT_d}. \tag{2.18}$$

In practice, it is not easy to stabilize the system with damping based on the derivative of the contact force. Therefore, an additional damping term is composed using the velocity of the end-effector in the contact force direction, i.e.,

$$f^*_{c,i} = -k_d \vartheta_{c,i}. \tag{2.19}$$

where k_d is the damping coefficient. Using the spring model (2.16) this can be expressed as

$$f^*_{c,i} = -\frac{k_d}{k_{s,i}} \dot{f}_{c,i}. \tag{2.20}$$

With this additional damping term the system equation becomes

$$\ddot{f}_{c,i} = -k_d \dot{f}_{c,i} e^{-sT_d} + k_{s,i} f^*_{c,i} e^{-sT_d}. \tag{2.21}$$

The transfer function, $G(s) = F_{c,i}/F^*_{c,i}$, from Equation (2.21) is

$$G(s) = \frac{k_{s,i} e^{-sT_d}}{s(s + k_d e^{-sT_d})}. \tag{2.22}$$

When the time-delay, T_d, is small, it can be approximated by

$$G(s) \approx \frac{k_{s,i}e^{-sT_d}}{s(s+k_d)}. \tag{2.23}$$

for a wide range of frequencies. The equivalent temporal representation is

$$\ddot{y} + k_d\dot{y} = k_{s,i}u(t - T_d) \tag{2.24}$$

where y is the plant output (contact force at the end-effector), f, and u is the input, f^*. Defining the state variables $x_1 = y$ and $x_2 = \dot{y}$, Equation (2.24) can be written as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -k_d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ k_s \end{bmatrix} u(t - T_d). \tag{2.25}$$

and,

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{2.26}$$

In compact form,

$$\begin{cases} \dot{x} = Ax(t) + Bu(t - T_d) \\ y(t) = x_1 \end{cases} \tag{2.27}$$

Discretizing Equation (2.27) with sampling time h, the equivalent discrete time system is

$$\begin{cases} x_{r,k} = \Phi_r x_{r,k-1} + \Gamma_r u_{k-1} \\ y_k = C_r x_{r,k} \end{cases} \tag{2.28}$$

with

$$T_d = (d-1)h + \tau' \tag{2.29}$$

$$0 < \tau' \leq h \tag{2.30}$$

$$x_{r,k} = [x_k^T \quad u_{k-d} \quad \ldots \quad u_{k-2} \quad u_{k-1}]^T \tag{2.31}$$

$$\Phi_r = \begin{bmatrix} \Phi_1 & \Gamma_1 & \Gamma_0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \tag{2.32}$$

$$\Gamma_r = [0 \quad 0 \quad \dots \quad 0 \quad 1]^T \tag{2.33}$$

and

$$C_r = [1 \quad 0 \quad \dots \quad 0 \quad 0]. \tag{2.34}$$

The matrices Φ_1, Γ_0, and Γ_1 are given by

$$\Phi_1 = e^{Ah} = \phi(h) \tag{2.35}$$

$$\Gamma_0 = \int_0^{h-\tau'} \phi(\lambda) d\lambda B \tag{2.36}$$

and

$$\Gamma_1 = \phi(h-\tau') \int_0^{\tau'} \phi(\lambda) d\lambda B \tag{2.37}$$

The term x_k has two states representing the force and force derivative. The other states appear due to dead-time. The continuous state transition and command matrices are

$$\phi(t) = \begin{bmatrix} 1 & \frac{1-e^{-k_d t}}{k_d} \\ 0 & e^{-k_d t} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ k_s \end{bmatrix}. \tag{2.38}$$

2.2.2 AOB design

Recalling the discrete state space representation of Equation (2.28), the theory of AOB [11] can be applied to this system in a straightforward manner to achieve adaptive control in the presence of uncertainties. A special Kalman filter must be designed to achieve a model reference adaptive control architecture. An extra state (active state), p_k, is generated to eliminate an equivalent disturbance referred to the system input. This equivalent disturbance exists whenever the response of the physical system is different from the desired model. This estimated disturbance to the input is directly compensated for at the input by subtracting the value of the active state. Therefore, its role is similar to the integral control in classical PID controllers. However, rather than generating the input by accumulating the error between the desired values and measured values, the active state is generated by the error between the estimated values and measured values. This term, therefore, realizes the model reference adaptive control. In general, the Nth order dynamic model can be applied to the input disturbance [11]. For force control applications a first order AOB will be described and implemented here. The block diagram of the AOB design for force control is shown in Figure 2.4.

Inserting the active state, p_k, in the loop, the overall system can be described by[1]

$$\begin{aligned} x_k &= \Phi x_{k-1} + \Gamma u_{k-1} + \xi_k \\ y_k &= C_a x_k + \eta_k, \end{aligned} \tag{2.39}$$

where

$$x_k = \begin{bmatrix} x_k \\ p_k \end{bmatrix} \quad , \quad \Phi = \begin{bmatrix} \Phi_r & \Gamma_r \\ 0 & 1 \end{bmatrix} \tag{2.40}$$

$$\Gamma = \begin{bmatrix} \Gamma_r \\ 0 \end{bmatrix} \quad , \quad C = [C_r \quad 0] \tag{2.41}$$

and the stochastic inputs ξ_k and η_k are model and measurement uncertainties.

[1] The subscript i is omitted in the state space form since the system equation is the same for all the contacts.

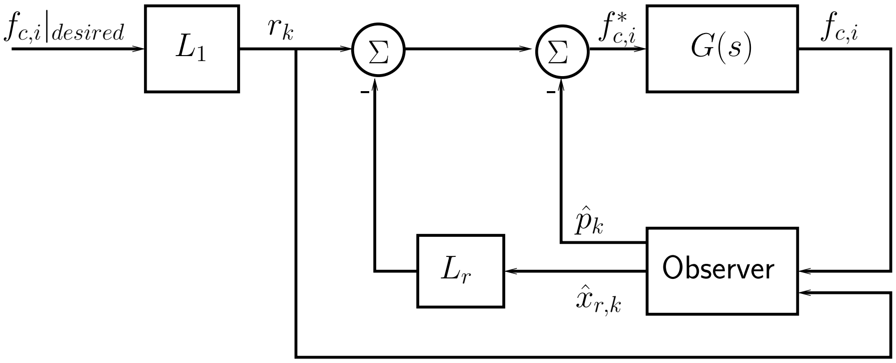


Figure 2.4: **AOB design for force control.** The term $G(s)$ is the system transfer function from the command, $f^*_{c,i}$, to the contact force, $f_{c,i}$. The term $f_{c,i}|_{desired}$ is the desired contact force. The terms r_k, $\hat{x}_k$, and $\hat{p}_k$ are reference input, state estimate, and input error estimate. The terms L_r and L_1 are a full state feedback gain and a scaling factor to compute reference input, r_k, respectively.

A full state feedback gain, L_r, is designed using pole placement method (Ackermann's formula) [18]. Combining the state feedback with the direct compensation of the input error estimate, the input to the system is

$$u_{k-1} = r_{k-1} - L\hat{x}_{k-1} \tag{2.42}$$

$$L = [L_r \quad 1]. \tag{2.43}$$

A Kalman estimator is designed based on Equation (2.39) and (2.42).

$$\hat{x}_k = \hat{x}_{k|(k-1)} + K_k(y_k - \hat{y}_k) \tag{2.44}$$

$$\hat{x}_{k|(k-1)} = \Phi_{closed}\hat{x}_{k-1} + \Gamma r_{k-1} \tag{2.45}$$

$$\Phi_{closed} = \begin{bmatrix} \Phi_r - \Gamma_r L_r & 0 \\ 0 & 1 \end{bmatrix} \tag{2.46}$$

$$\hat{y}_k = C\hat{x}_{k|(k-1)} \tag{2.47}$$

The Kalman gain K_k is

$$K_k = P_{1k}C^T[CP_{1k}C^T + R_k]^{-1} \tag{2.48}$$

with

$$\begin{aligned} P_{1k} &= \Phi P_{k-1}\Phi^T + Q_k && (2.49)\\ P_k &= P_{1k} - K_k C P_{1k}. && (2.50)\end{aligned}$$

The system noise matrix, Q_k, represents model uncertainty. The term R_k is the measurement noise variance matrix. The term P_k is the mean square error matrix of the states.

2.3 Sensitivity analysis

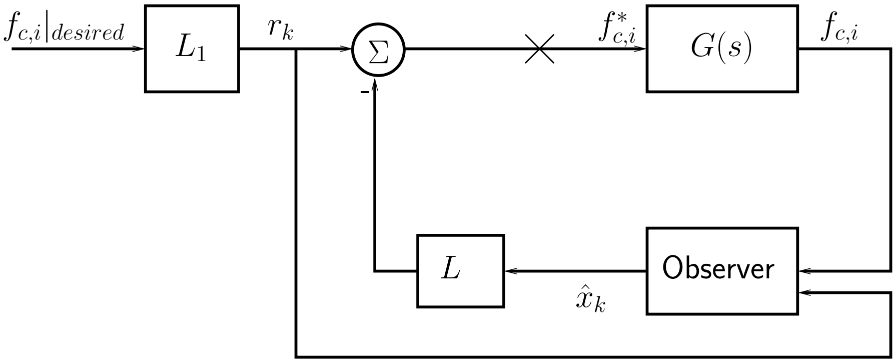


Figure 2.5: **The breakpoint for loop transfer function.**

Sensitivity analysis in the presence of modeling errors is important in designing a controller. The loop transfer function of the system is used to analyze the gain/phase margin of the closed loop system. The break point to derive the loop transfer function is chosen at the input to the real system in Figure 2.5 [15].

The system with an active state is

$$x_k = \Phi x_{k-1} + \Gamma u_{k-1}. \tag{2.51}$$

We define the nominal system matrix, Φ_n, as the system matrix used in the control design, and $\Delta\Phi$ as the error between the real system matrix, Φ, and the nominal system matrix. Thus

$$\Phi = \Phi_n + \Delta\Phi. \tag{2.52}$$

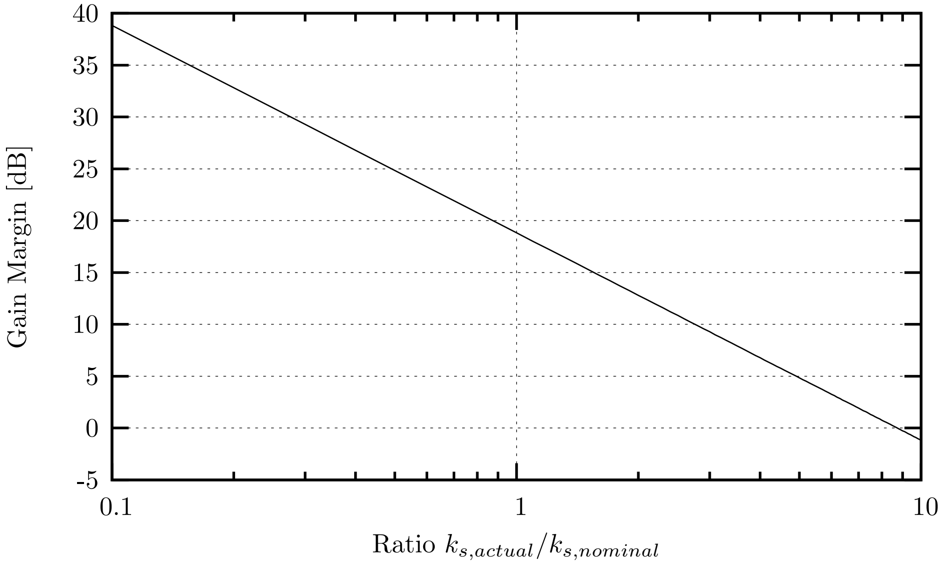


Figure 2.6: **Gain margin when $\mathbf{k_{s,nominal}} = \mathbf{100.0N/m}$.**

The state estimate is based on the nominal system matrix, Φ_n, and is given by

$$\hat{x}_k = \Phi_{n,closed}\hat{x}_{k-1} + K_k[y_k - C(\Phi_{n,closed}\hat{x}_{k-1})] \tag{2.53}$$

where

$$\Phi_{n,closed} = \Phi_n - \Gamma L \tag{2.54}$$

Defining the estimation error of x_k as

$$e_k = x_k - \hat{x}_k, \tag{2.55}$$

we have

$$\begin{bmatrix} \hat{x}_k \\ e_k \end{bmatrix} = \begin{bmatrix} M_{11} & K_k C\Phi \\ M_{21} & (I - K_k C)\Phi \end{bmatrix} \begin{bmatrix} \hat{x}_{k-1} \\ e_{k-1} \end{bmatrix} + \begin{bmatrix} K_k C\Gamma \\ (I - K_k C)\Gamma \end{bmatrix} u_{k-1}, \tag{2.56}$$

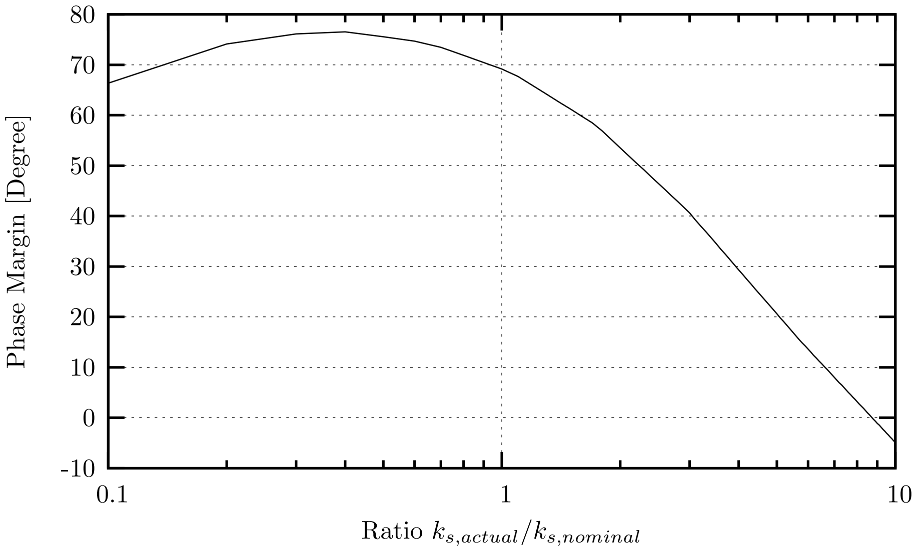


Figure 2.7: **Phase margin when $\mathbf{k_{s,nominal}} = \mathbf{100.0N/m}$.**

where

$$\begin{aligned} M_{11} &= \Phi_n - \Gamma L + K_k C(\Delta\Phi + \Gamma L) \\ M_{21} &= (I - K_k C)(\Delta\Phi + \Gamma L). \end{aligned} \tag{2.57}$$

The output of loop transfer function is $Y_k = L\hat{x}_k$, i.e.

$$Y_k = [L \quad 0] \begin{bmatrix} \hat{x}_k \\ e_k \end{bmatrix} \tag{2.58}$$

From Equation (2.56) and (2.58), the transfer function is given by

$$H_{LTF}(z) = [L \quad 0][I - \Phi_a z^{-1}]^{-1}\Gamma_a z^{-1}, \tag{2.59}$$

where Φ_a and Γ_a are the state transition and command matrices in Equation (2.56), respectively. The Bode plots can be used to analyze the gain and phase margins of the control system with respect to the uncertainty, $\Delta\Phi$.

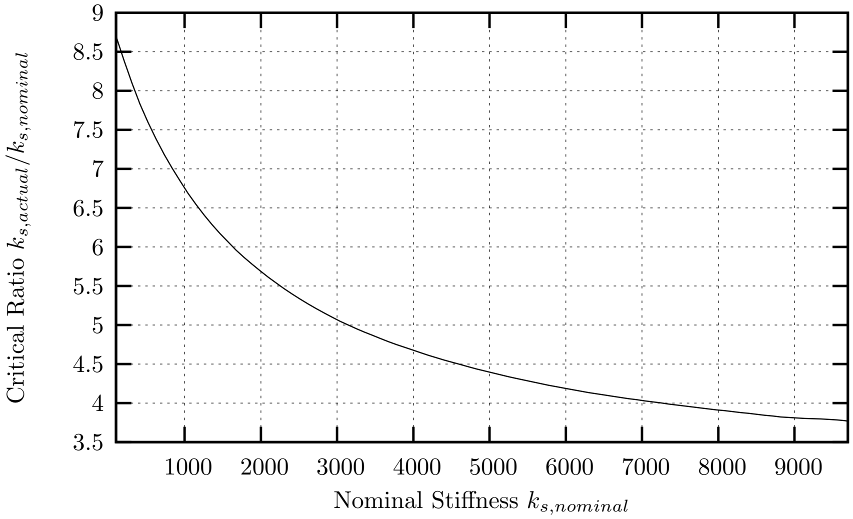


Figure 2.8: **Stability characteristics over nominal stiffnesses.** The critical ratio of $k_{s,actual}$ to $k_{s,nominal}$ indicates when the system becomes unstable. The controller is stable up to the actual stiffness of 8.5 and 3.8 times the nominal stiffness at the nominal stiffness of 100 N/m and 9000 N/m, respectively.

Among the many possible modeling errors in the system, the stiffness of the environment is the most significant uncertainty since the dynamic and kinematic parameters of the robot are relatively well known. In practice, the environment stiffness is not only difficult to measure in advance, but it is also changing over different magnitudes of contact forces applied by the robot. Therefore, it is important to analyze the robustness of the control system with respect to the mismatch of environment stiffness.

By defining $k_{s,n}$ as the nominal stiffness of the environment which will be used for AOB design

$$k_s = k_{s,n} + \Delta k_s. \tag{2.60}$$

The transition matrices Φ, Φ_n, and $\Delta\Phi$ can be computed for the analysis.

Figures 2.6 and 2.7 show the gain and phase margins of the system when the actual stiffness of the system differs from the nominal stiffness. The nominal stiffness is chosen as $100N/m$ for the plot. The general shape of the plots for other nominal stiffnesses remains the same. The gain and phase margins are plotted by changing the ratio of $k_{s,actual}$ to $k_{s,nominal}$. As can be seen in the plot, the system becomes unstable if the ratio exceeds approximately 8.5. The same analysis has been conducted for different nominal stiffnesses and the result is plotted in Figure 2.8. The system becomes unstable if the actual stiffness is beyond this critical ratio of $k_{s,actual}$ to $k_{s,nominal}$. For example, the controller with a nominal stiffness of $100.0N/m$ is not stable if the actual stiffness is bigger than 8.5 times the nominal stiffness of $100.0N/m$.

2.4 Experiments

2.4.1 Experimental setup

A PUMA560 manipulator and a DELTA haptic device (Force Dimension) were used in the development and performance analysis of the force controller. The PUMA560 was connected to a PC running the QNX operating system through a TRC205 amplifier package from Mark V Corporation. This setup allowed a user to program joint torques or motor currents as inputs to the robot. A JR3 force sensor with 6 axis measurements was mounted on the wrist of the manipulator to measure contact forces at the end-effector. To create environments with specified stiffnesses the DELTA haptic device was utilized. The haptic device was programmed to have a specific stiffness and damping. Although this device created the specified stiffnesses in open loop control, the generated stiffness had an error within 20 %. A picture of this setup is shown in Figure 2.9.

Analysis of the performance was conducted in the vertical direction of the robot end-effector. That is, the other translational directions and orientations were controlled by position control of the end-effector. The corresponding selection matrices are in Equation (2.10). No null space control was required since there was no redundancy.

The results of the AOB based force controller are compared by those of a PID controller to clearly demonstrate the improved robustness and performance. The proportional/integral gains and damping coefficient were specified to have the same bandwidth as the AOB based controller. These gains were parameterized with the

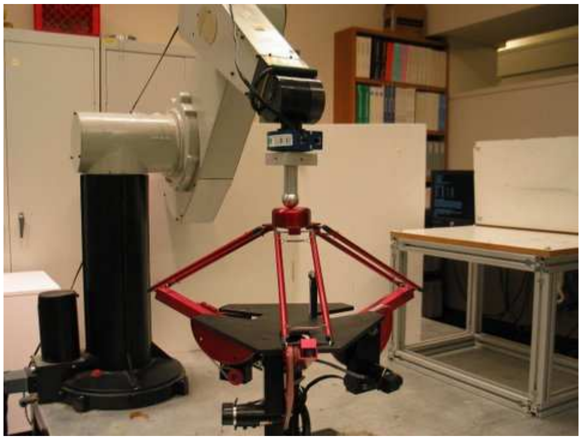


Figure 2.9: **PUMA560 in contact with DELTA.** A DELTA haptic device is programmed to have a specified stiffness.

environment stiffness such that the controller would have the same responses at any known stiffness of the environment. The following sections provide the experimental results using both a PID controller and an AOB based controller.

2.4.2 Environments with known stiffnesses

The experiments were conducted for the case when the environment in contact was known. The purpose is to analyze the performance of the controllers when the environment model is accurately known. However, there are still unmodeled dynamics or disturbances. Therefore, the result of this experiment demonstrates the robustness of the controllers with respect to the unmodeled dynamics.

Both PID and AOB controllers were designed to have the same bandwidth, which was 20 rad/sec in this experiment. In the process of choosing PID gains, proportional and damping terms were first chosen to have the proper bandwidth. The choice of the integral gain was done by trial and error in the experiments. Since the system of Equation (2.17) has a pole at the origin, the integral term is not necessary if the system is ideal. However, the model of a robot in contact possesses many uncertainties which typically affect much of the contact force response. Therefore, the use of integral control is necessary in order to achieve zero steady state error. In the experiment, the tuning of the integral gain was done for an environment with a stiffness of 2000.0 N/m. That is, the DELTA device was programmed to this specified stiffness. Several experimental runs were conducted until the desired response was achieved.

Figure 2.11 (a) shows the results of the PID controller in contact with the DELTA device. The device was programmed to 2000.0 N/m. Several runs are plotted to show the consistency of the controller. The data were gathered by commanding four square inputs from -5 to -15. Thus, it contains four falling steps from -5 to -15 and four rising steps from -15 to -5. The plots of rising steps were converted to the scale of falling steps for easy comparison.

The parameters for the AOB controller were chosen based on the response at the stiffness of 2000.0 N/m. The control gains were chosen to have the same speed of response and the stochastic parameters were chosen to have the desired response of the system. The same experiments as those for the PID controller were conducted and plotted in Figure 2.11 (b). The results demonstrate more consistent performance from the AOB controller than from the PID controller.

This is further demonstrated in additional step responses at various stiffnesses. Figure 2.12 shows the step responses of the PID and AOB force controllers when they are applied to environments with different stiffnesses. The control gains and parameters for both controllers are correspondingly modified to these different stiffnesses. The AOB controller produces consistent results at different stiffnesses. However, the PID controller fails to have consistent results for different stiffnesses. In the responses of the PID controller, the proportional controller always dominates at the beginning of the response but the integral part acts too fast for the softer contact and too slow for the harder contact. This is related to the integral gains which are in fact lower at the harder contact and higher at the softer contact since they are parameterized by the contact stiffness. This observation indicates that the disturbance or unmodeled dynamics are not directly related to the environment stiffness.

2.4.3 Environments with unknown stiffnesses

In this section, the experiments were designed to demonstrate the robustness with respect to the modeling errors in the environment model. Both PID and AOB controllers were tuned for an environment with a stiffness of 5000 N/m. Step responses were gathered while the stiffness of the DELTA device was set to certain stiffnesses, varying from 1000 N/m to 9000 N/m. These stiffnesses of the DELTA were different from those in the controllers. Thus, the step responses demonstrate the performance of the controllers in the presence of the model parameter mismatch. This type of model uncertainty is important to address in addition to the unmodeled dynamics, which was analyzed in the previous section. In most of the applications of contact force control, the model of the environment is not easily obtained. Thus, first order models are typically used. The use of low order models and the uncertainty of the physical parameters introduce unmodeled dynamics. Therefore, the robustness to both facts are important characteristics for given controllers as a part of the performance measurement.

The step responses for this experiment are plotted in Figure 2.13. The results for the PID controller show larger variance and greater inconsistency. The AOB controller succeeds in consistently adapting to different stiffness environments and demonstrates a favorable characteristic of the model reference approach in the AOB controller.

2.4.4 Rigid contact

The last experiment was conducted on a table made of particle board with an aluminum frame (Figure 2.10). The stiffness was at least 50,000 N/m and can be considered as a *rigid* contact. Both PID and AOB controllers were set for the stiffness of 10,000 N/m and tested on the table. The results are plotted in Figure 2.14.

Rigid surfaces are challenging for contact force control since it is difficult to achieve a desired performance without causing instability. Typically, due to the high stiffness of the system, the gains are set low. The integral control gain is limited due to stability characteristics of the system. The resulting responses of PID control exhibit

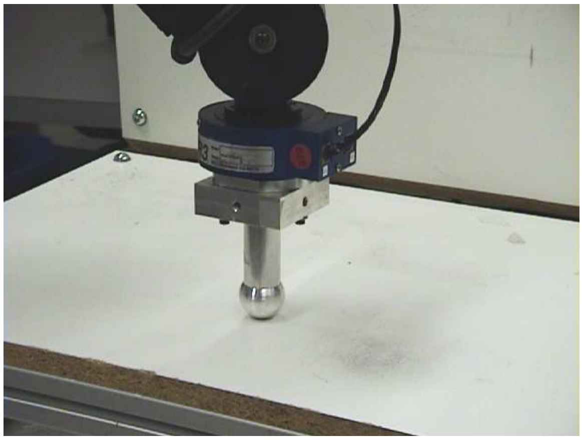


Figure 2.10: **PUMA560 in contact with a table.**

a long settling time and inconsistency.

Figure 2.14 (a) depicts plots for the PID controller, which has a fast system response at the beginning and then slow convergence to the desired value. The initial system responses are fast even with a very low proportional gain because of the fast open loop system characteristics. The integral controller is incorporated to obtain accurate steady state response. This illustrates why it is difficult to design a robust controller using conventional PID control. The theoretically designed controller works very well only in the simulator. Special tuning of the gains is required to achieve the desired response. The tuned values vary a great deal depending on the environmental stiffness and the configuration of the robot. This procedure is even more difficult in dealing with rigid contact. Thus, the performance is very sensitive to changes in the system.

The AOB design alleviates these difficulties in the control design by introducing a model reference adaptive approach. Cancelling out the estimated input disturbance term at the input command forces the system to follow the desired closed loop system model. This differs from the integral part in a PID controller in that the integral controller generates the control input based on the difference between the measurement and reference value.

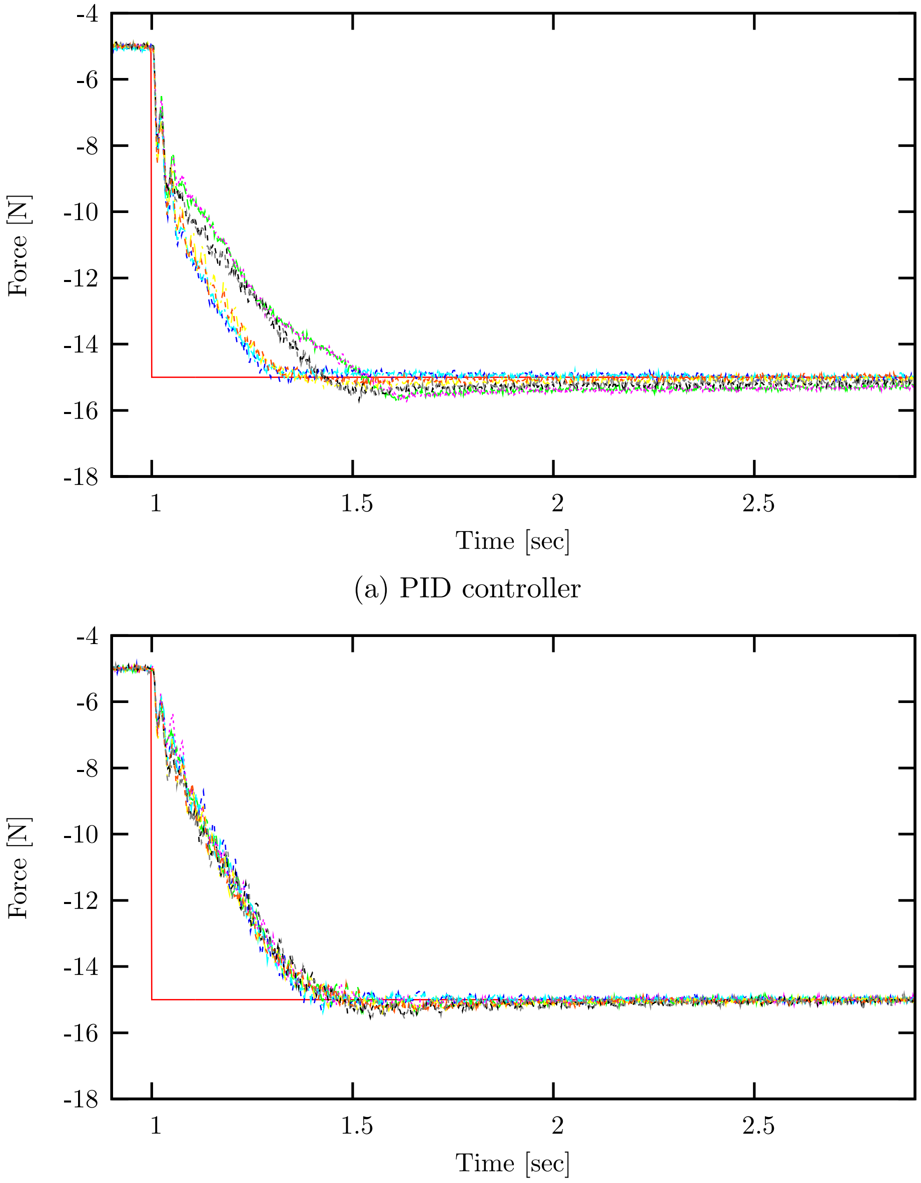
2.5 Conclusion

A force control approach is implemented using Active Observers (AOB) in the operational space framework. The experimental results show the characteristic that the closed loop system is robust to unmodeled dynamics and the mismatch of the parameters in the model. This characteristic critical since these uncertainties are always present whenever we deal with contact. The use of contact force control in teleoperation is an example where model uncertainties are significant. This will be discussed in detail in Chapter 3.

In the composition of the contact force controller, the operational space control framework is applied to dynamically decouple the overall system into linearized subsystems. The AOB approach is then used in the linearized contact force control system. Using the operational space framework we can implement a modular and hierarchical control approach. This approach simplifies each controller and provides various design options. The AOB controllers are implemented on the decoupled linear second order systems for each translational direction. Employing estimators for a highly nonlinear robotic system would yield a very complex and high dimensional control system.

AOB controllers use a Kalman estimator with an additional state associated with the disturbance at the system input. This implementation realizes the model adaptive reference approach. This approach has been demonstrated to be more robust than a conventional PID controller through experiments. The role of the active state is to reduce the differences in the system responses between the closed loop model and the actual system. Therefore, its adaptive response can be more aggressive than the pure integral action. In addition, the design procedure is systematic by allowing the existence of the input disturbance in the model.

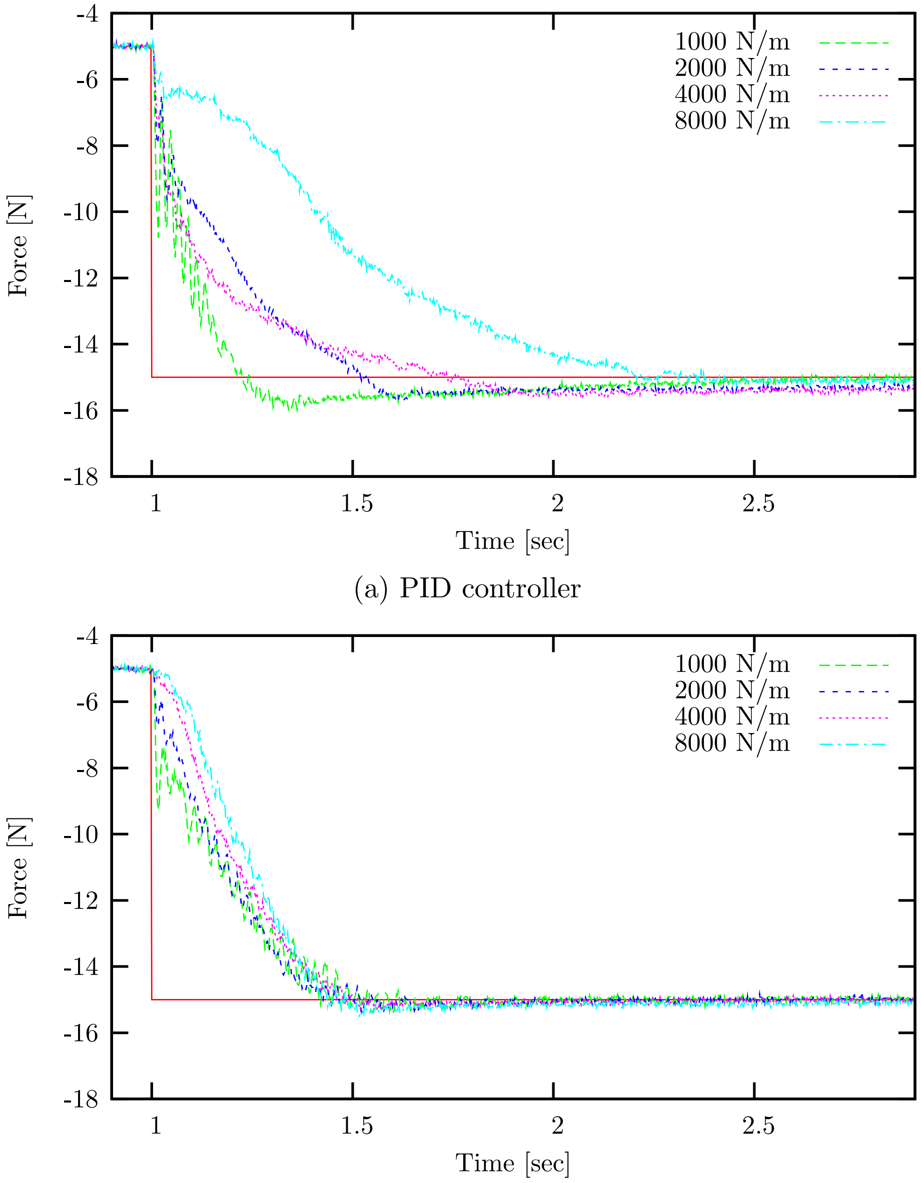
Further demonstration of this force control framework will be presented in the areas of haptic teleoperation and multi-link multi-contact control. The results in these application areas will also demonstrate high performance and robustness.



(a) PID controller

(b) AOB controller

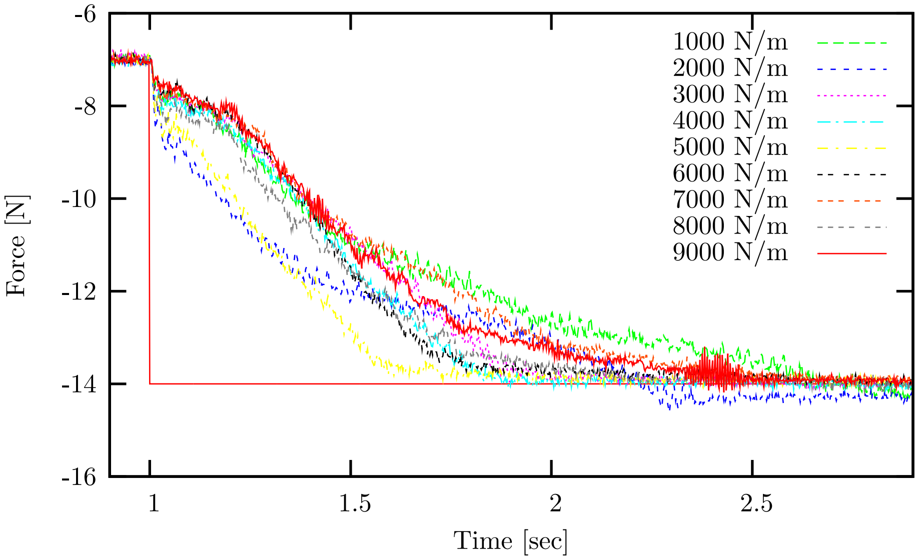
Figure 2.11: **Step responses of force controllers for a known stiffness of 2000 N/m.** The controllers are designed for a known stiffness of 2000 N/m. Results from eight runs are plotted. (a) results from PID controller (b) results from AOB controller



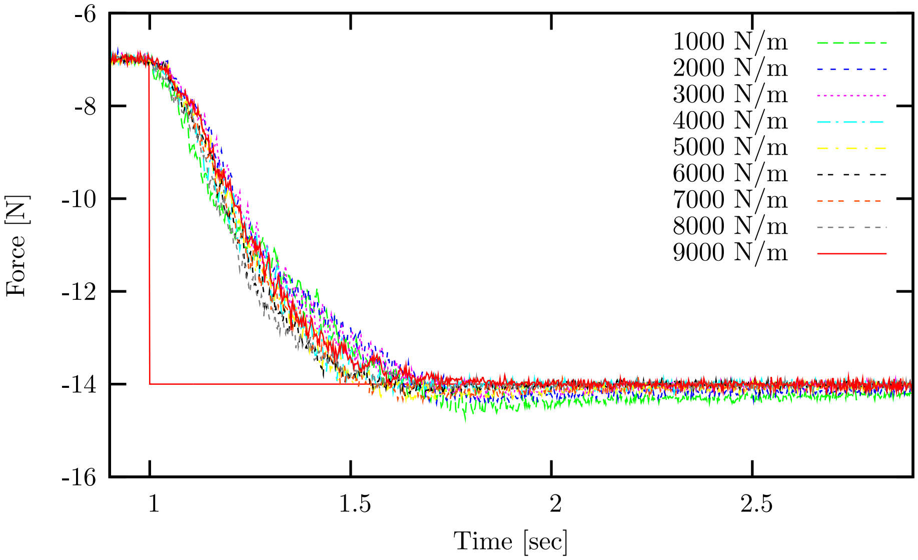
(a) PID controller

(b) AOB controller

Figure 2.12: **Step responses of force controllers for various known stiffnesses.** The controllers are designed for each stiffness, ranging from 1000 N/m to 8000 N/m. (a) results from PID controllers (b) results from AOB controllers

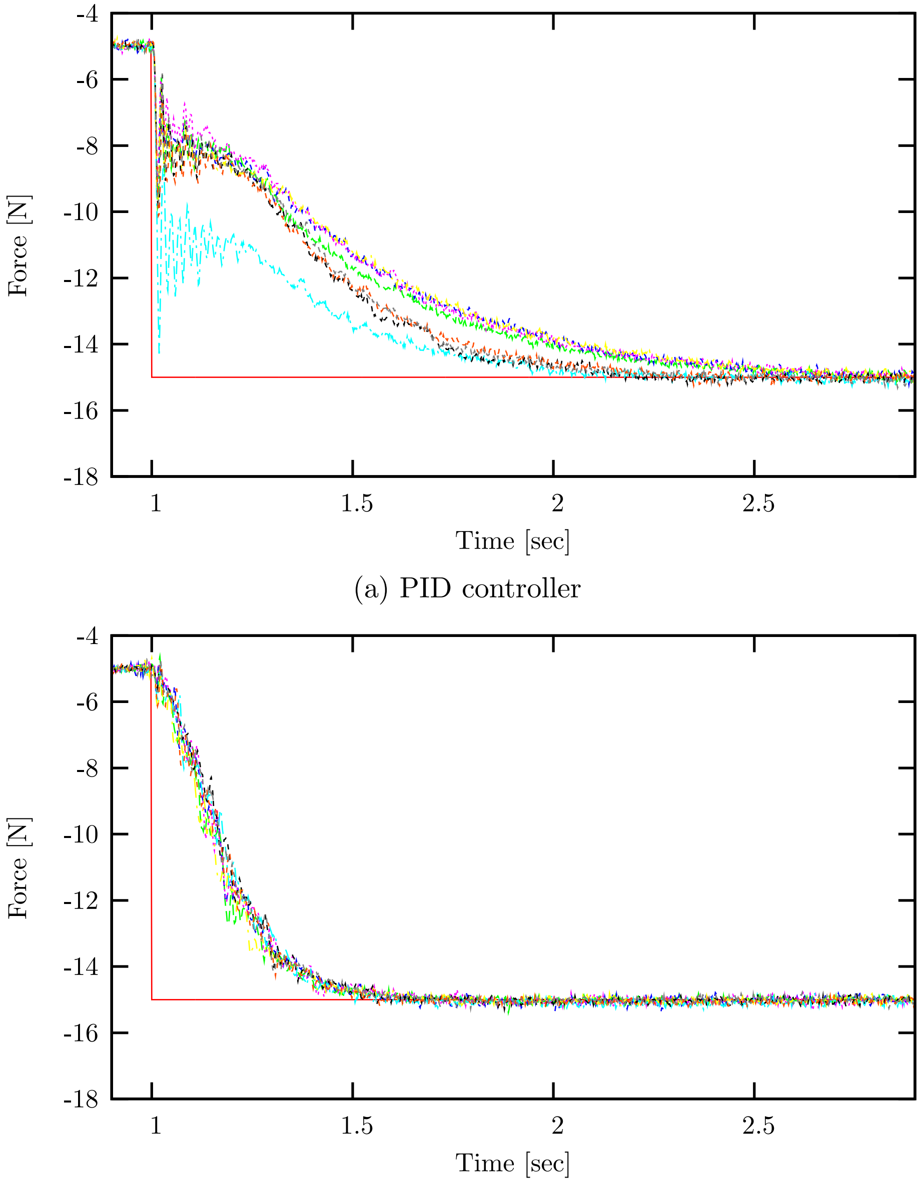


(a) PID controller



(b) AOB controller

Figure 2.13: **Step responses of force controllers for a stiffness of 5000 N/m with various unknown stiffnesses.** The controllers designed for a stiffness of 5000 N/m is tested for different unknown stiffnesses. (a) results from PID controller (b) results from AOB controller



(a) PID controller

(b) AOB controller

Figure 2.14: **Step responses of force controllers in contact with a table.** The controllers are designed for a stiffness of 10000 N/m. Results from eight runs are plotted. (a) PID controller (b) AOB controller

Chapter 3

Haptic teleoperation

The goal of haptic teleoperation is to allow a user to remotely control a slave robot through a master device while feeling forces from the remote environment. Such systems offer great potential, but connecting master/slave stations in a coherent way is a challenging task. While the master station is controlled by a human operator, the slave station often interacts with an unknown and dynamic environment. The nature of this interaction greatly influences overall system performance.

Many teleoperation schemes have been developed to improve telepresence and stability when position and force measurements are available on both the master and slave [38, 68, 36]. Telepresence is achieved when transparency of the teleoperation system is realized, i.e., accurate position tracking in free space operation, and force or impedance matching during contact [22, 38, 66]. A common control architecture is to use PD type position feedback control with direct feedforward force control to track the position and contact force of the counterpart system. This approach would provide perfect telepresence and stability in an ideal situation where measurements of acceleration are available and the feedforward contact force is perfectly applied [38, 66]. In practice, however, these conditions are not easily met. Specifically, the feedforward contact force command may not be realized due to uncertainties such as friction and modelling errors.

To address this difficulty, local force control is proposed in [20, 36, 2, 66, 22]. One of the main challenges in this approach is to design a local force controller that works for an environment that is not known a priori [20]. Also, the overall stability is degraded when the measured contact force of one system is used as the desired contact force of its counterpart. This problem is exacerbated if the mass properties of the master and slave differ significantly [14].

Another inherent characteristic of teleoperation systems is time delay in the communication link. Enhanced robustness to time delays using local force control is presented in [22]. To guarantee the stability of the overall system, passivity-based approaches have been extensively studied [2, 46, 47, 53]; however, loss of performance is inevitable in the approaches.

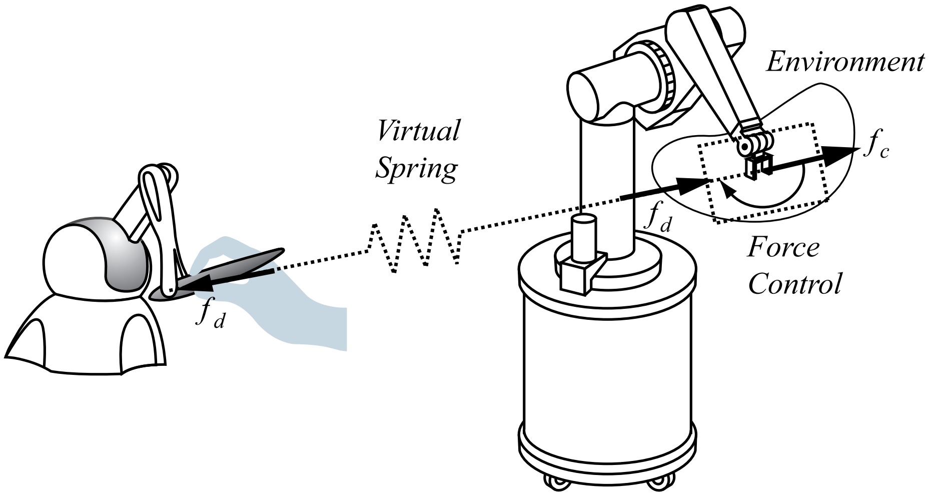


Figure 3.1: **Illustration of teleoperation approach with a virtual spring and force control.** The desired force, f_d, is produced by the virtual spring based on the position difference between the master and slave robot end-effectors. The force controller on the slave robot enforces the contact force, f_c, to track the desired force while the desired force is fed-back to the user at the master device.

This chapter introduces a new teleoperation approach, which is based on three components: a virtual spring to connect the master and slave systems, the operational space framework to provide a decoupled dynamic controller, and a local contact force controller to realize tracking of the contact force. This approach is illustrated in Figure 3.1 and the block diagram is shown in Figure 3.2. In this approach, a *virtual spring* connects the master and slave systems. When the positions of the master and

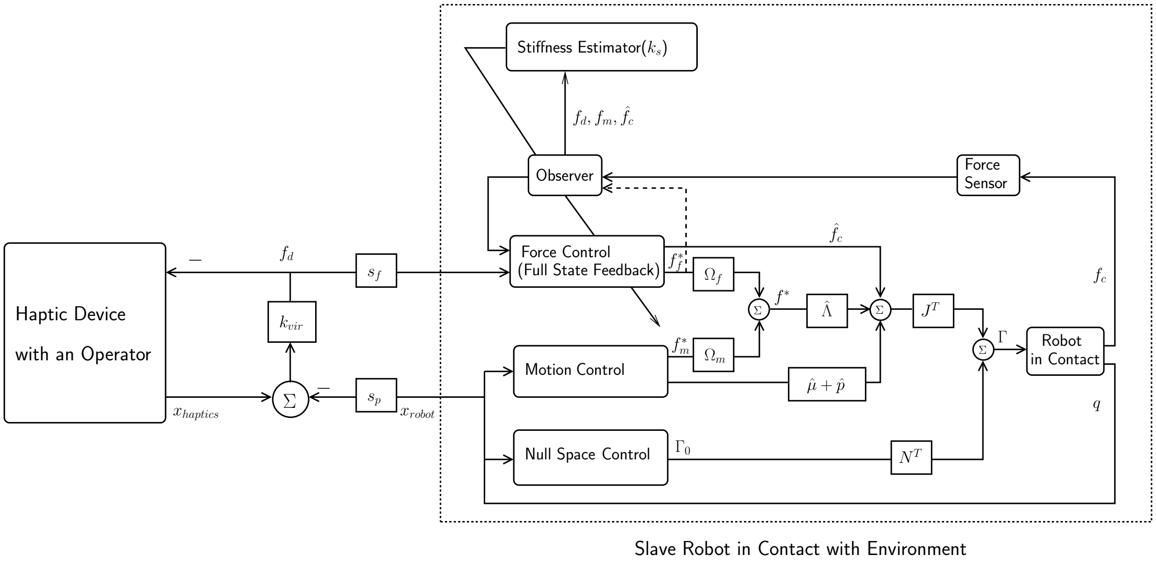


Figure 3.2: **A block diagram for the proposed teleoperation approach.** The master and slave system are connected by a virtual spring with a spring constant, k_{vir}. The terms, s_p and s_f, are the scale factors for position and force, which are used to adjust different workspaces and force magnitudes for the two systems. The block diagram in the dotted block on the right side shows the motion/force control structure for redundant robots.

slave system do not match, the virtual spring produces a force proportional to the difference in positions. This force acts as a desired contact force which is tracked by local force control on each side. This scheme thus provides the human operator with all contact forces within the bandwidth of the force controller. The robot control for each system is simply contact force control. Even in free space operation of the slave system the controller assumes that the robot is in contact with a very compliant environment.

Position tracking in free space is implicitly accomplished by the force control and the virtual spring. When the slave robot is in free space, the force control at the slave commands the robot to move toward the master's position until the difference in positions is zero, since the virtual spring produces the desired contact force in that direction. This approach greatly simplifies the overall teleoperation architecture. Furthermore, the stability characteristics with respect to time delays and the difference between the inertial properties of the master and slave system are improved since the measured contact force is not used as the desired contact force at the counterpart system. No switching is required in the robot control structure since the robot is considered to be always in contact with the environment, even in free space.

Local force control is the most important part of the proposed approach since telepresence depends on how much bandwidth the force controller has. Also, implementation on a complex mobile manipulation system is non-trivial. The operational space formulation [31] decouples the dynamics of the mobile manipulator into end-effector task dynamics and posture dynamics. Moreover, each end-effector DOF can be independently controlled. The control of the base can be separately synthesized since its dynamics are decoupled from that of the end-effector. Based on this formulation a local force control and teleoperation scheme is applied for each end-effector DOF.

To deal with uncertainties and time-varying parameters (e.g. dynamic environments), the force control on the slave robot uses Active Observers (AOB) [11] that modify the Kalman estimation structure to achieve model-reference adaptive control. The AOB is designed to cover a medium range of stiffness values. For large variations, on-line stiffness estimation is necessary [12] to improve robustness and telepresence. This on-line stiffness estimation is important in order to produce a significantly large bandwidth for the force controller over different environments, so that the teleoperation system can provide the user with accurate contact forces. In addition, the virtual spring stiffness is modified with the change in the estimated environmental stiffness for better telepresence.

This architecture is especially suited for systems where force sensing is limited to the slave robot and when the master device is relatively light and frictionless. Specifically, our setup which uses a PHANTOM device and a PUMA robot mounted on a mobile base, meets this criteria. While it is imperative to use local force control on the slave mobile manipulation system, the light-weight frictionless haptic device generates relatively accurate commanded forces. Time delay associated with the wireless LAN network is also analyzed for our system.

3.1 Control for a manipulator

3.1.1 Task control

The dynamic equation of a mobile manipulator is described by

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) + J_c(q)^T f_c = \Gamma, \tag{3.1}$$

where q, $A(q)$, $b(q, \dot{q})$, and $g(q)$ are the vector of joint angles, the mass/inertia matrix, the Coriolis/centrifugal torque, and the gravity torque in joint space, respectively. The term J_c denotes the Jacobian for the contact point. The equations of motion for the end-effector of a robotic manipulator can be described using the operational space formulation [31]. This yields

$$\Lambda(q)\dot{\vartheta} + \mu(q, \dot{q}) + p(q) + f_c = F \tag{3.2}$$

where $\Lambda(q)$, $\mu(q, \dot{q})$, and $p(q)$ are the inertia matrix, the vector of Coriolis/centrifugal forces, and the vector of gravity forces in operational space, respectively. The term ϑ denotes the instantaneous velocity in operational space coordinates and f_c is the contact force at the end-effector. The control torque is selected as,

$$\begin{aligned} \Gamma &= J^T F + N^T \Gamma_0 && (3.3) \\ F &= \hat{\Lambda}(q) f^* + \hat{\mu}(q, \dot{q}) + \hat{p}(q) + \hat{f}_c && (3.4) \end{aligned}$$

where N^T is the dynamically consistent null space projection matrix and f^* is the command to the unit mass system. The $\hat{}$ indicates an estimate of a particular quantity. The following decoupled equations of motion for the end-effector are obtained when the estimates are perfect.

$$\dot{\vartheta} = f^* \tag{3.5}$$

The command f^* is composed of force and motion control components that are projected by the selection matrices, Ω_f and Ω_m, respectively.

$$f^* = \Omega_f f_f^* + \Omega_m f_m^* \tag{3.6}$$

In the experimental setup, force control is used to control only the Cartesian position of the end-effector since the master device does not provide force feedback on the orientation. The selection matrices are

$$\Omega_f = \begin{bmatrix} I_3 & 0_3 \\ 0_3 & 0_3 \end{bmatrix}, \qquad \Omega_m = \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & I_3 \end{bmatrix} \tag{3.7}$$

where I_3 is the 3×3 identity matrix and 0_3 is the 3×3 zero matrix.

The control of the mobile base is applied to Γ_0 in Equation (3.3). The dynamically consistent null space projection matrix N^T prevents control of the mobile base from affecting the end-effector control. The overall control framework is illustrated in Figure 2.2.

3.1.2 Posture control

In the case of a redundant robot posture control is composed in the null space of the task control. This control can be designed by taking account of the task dynamics in addition to the whole system dynamics [34]. This approach ensures the performance of the posture control. Without accounting for task and robot dynamics in the composition of the posture control the posture behavior may not be consistent in different configurations of the robot although it is still guaranteed that the posture control will not interfere with the task control.

One of the experimental setups uses a mobile manipulator, which is considered as a redundant robot. In this case, the task is the motion and force control of the end-effector. That is the task coordinate is x_t. The position and orientation of the base are chosen as the posture coordinates, x_p, and the corresponding Jacobian is defined as J_p. The null space torque, Γ_0, is then selected as $J_p^T F_{p|t}$. With the control torque in Equation (3.3), the posture dynamics are obtained by projecting Equation (3.1) into the posture coordinates.

$$\Lambda_{p|t}\dot{\vartheta}_p + \mu_{p|t} + p_{p|t} + \bar{J}_{p|t}^T J^T f_c = F_{p|t} + \bar{J}_{p|t}^T J^T F, \tag{3.8}$$

where

$$\begin{aligned} \Lambda_{p|t}^{-1} &= J_p A^{-1} N^T J_p^T && (3.9) \\ \bar{J}_{p|t}^T &= \Lambda_{p|t} J_p A^{-1} && (3.10) \\ \mu_{p|t} &= \bar{J}_{p|t}^T b(q,\dot{q}) - \Lambda_{p|t}\dot{J}_p\dot{q} && (3.11) \\ p_{p|t} &= \bar{J}_{p|t}^T g(q), && (3.12) \end{aligned}$$

and F is the control force for the task.

The control force, $F_{p|t}$, is composed to compensate the dynamics and the control input for the task.

$$F_{p|t} = \hat{\Lambda}_{p|t} f_p^* + \hat{\mu}_{p|t} + \hat{p}_{p|t} + \bar{J}_{p|t}^T J_c^T \hat{f}_c - \bar{J}_{p|t}^T J^T F. \tag{3.13}$$

The total torque to be applied to the robot is

$$\Gamma = J^T F + N^T J_p^T F_{p|t}. \tag{3.14}$$

This results in

$$\dot{\vartheta}_p = f_p^*. \tag{3.15}$$

The operational space control structure provides nonlinear dynamic decoupling and dynamic consistency for the task and posture. The task and posture behaviors of the decoupled systems are described in Equations (3.5) and (3.15). The control inputs f^* and f_p^* can be designed using a simple PD controller or any other controller. The stability and performance designed for a given controller at the decoupled system, Equation (3.5), are achieved at the nonlinear highly coupled system of Equation (3.2) through the nonlinear dynamic decoupling provided by the control structure of Equation (3.3) and (3.4) [19, 31]. Because of the redundancy, the asymptotic stability of the redundant robot (3.1) requires the posture controller of Equation (3.15) to be asymptotically stable [31]. In the next section we present a specific control strategy for the force control portion f_f^* of f^* in Equation (3.6). The motion control inputs for the task, f_m^*, and posture, f_p^*, will be designed using simple PD controllers.

3.2 Contact force control with stiffness estimation

The decoupled unit mass system for each translational direction, Equation (3.5), is used for force controller design. With the contact model, Equation (2.16), the equations of motion of contact force for each direction in operational space are,

$$\ddot{f}_c = k_s f^* \tag{3.16}$$

The system transfer function for contact force control is derived from a decoupled sub-system (3.16). With an additional damping, $k_v \vartheta$, to f^* for better stability and a system input delay, $T_{input,d}$, the overall system can be approximated by

$$G(s) = \frac{k_s e^{-sT_{input,d}}}{s(s+k_v)}, \tag{3.17}$$

where k_v is a positive scalar. The discretized state space form of Equation (3.17) is used for discrete Kalman estimation and control.

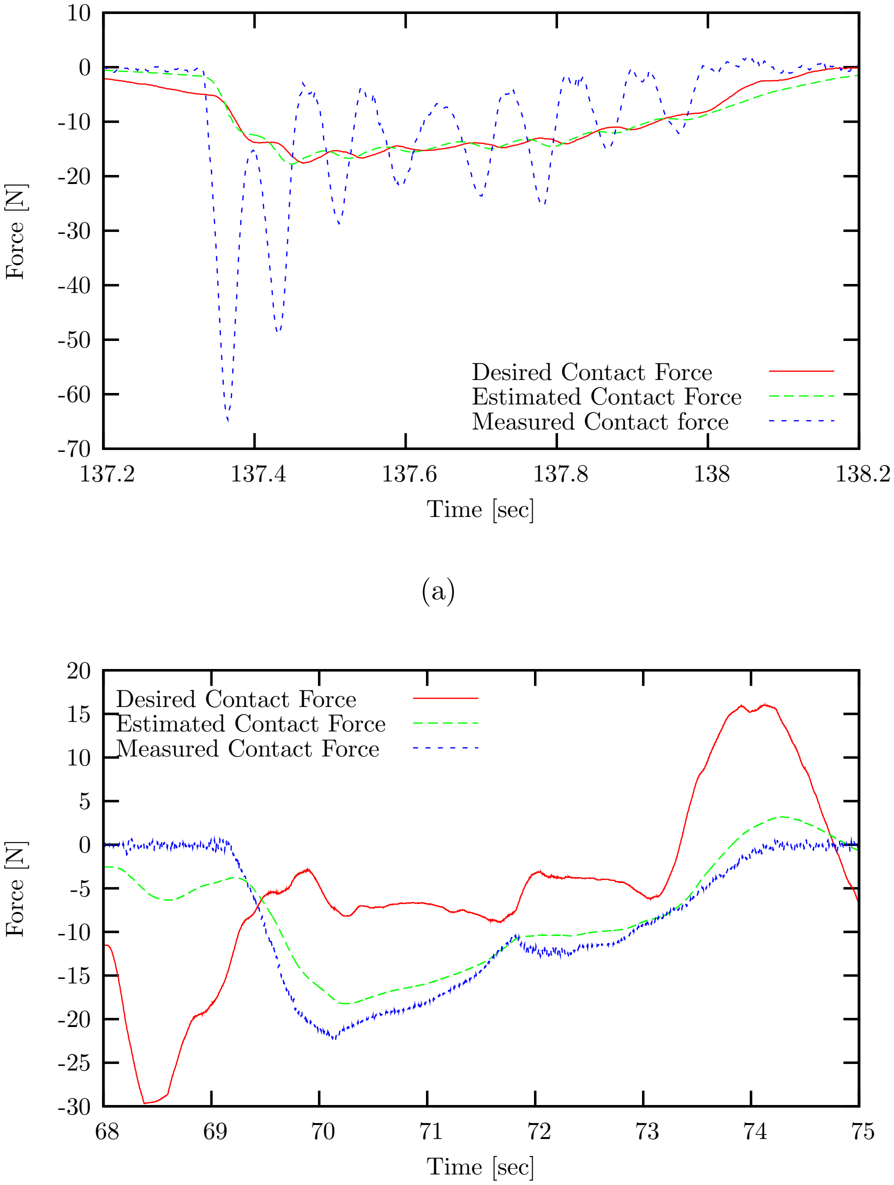
The detailed formulation and analysis is explained in Chapter 2. The overall force control scheme is illustrated in Figure 2.4. A Kalman estimator is designed to estimate the states of the system and the additional state, input error. The estimate of the input error is then directly canceled at the input command (Figure 2.4). Full state feedback is applied with the estimated states of the system. The use of input error as a state provides an alternative way of implementing integral control. Also, since most manipulators do not have joint torque sensors, it is not certain how accurately the input torque command is applied to the corresponding joint. This fact contributes to one of the major model uncertainties in the system. Thus, the estimate of input error plays an important role in actual manipulators.

3.2.1 Stiffness adaptation

The slave manipulator in teleoperation experiences contact with different environments. The knowledge of the stiffness, k_s, is important not only for force control but also for modifying the virtual spring, k_{vir}, in order to provide better telepresence to the operator. The changes in the environment's stiffness can be abrupt and large in magnitude. Although the contact force controller designed with AOB is robust to the change of environmental stiffness, its performance will degrade when there is a mismatch between estimated and actual stiffness and the system may be unstable if the mismatch is beyond the stability margin. These facts are demonstrated in the experiments in the presence of a large mismatch of the environment stiffness, as shown in Figure 3.3. Thus, a fast on-line stiffness estimation strategy is required to cope with these changes.

A review of estimation methods for contact stiffness and damping is presented in [16]. A signal processing method, an indirect adaptive controller [58], a model reference adaptive controller [61], and a recursive least-squares estimation technique [41] are reviewed. The signal processing method requires off-line implementation while the other three methods are implemented on-line. However, these on-line methods still require exciting signals over time to compute the model parameters. These algorithms are based on quantities such as the measured contact force and deflected position. Although these approaches have demonstrated convergence between estimated and actual stiffness, the time required to achieve this convergence increases as the robot's motion slows or when the motion is less than the resolution of the encoders. This situation often occurs in haptic teleoperation when the robot touches a rigid surface or is stationary in free space. Therefore, although stiffness identification is accomplished, the performance of the control is severely degraded prior to convergence of the estimated stiffness.

While the stiffness estimation methods reviewed in [16] seek to accurately estimate stiffness, in haptic teleoperation obtaining a highly accurate estimate of the stiffness is not as critical as achieving consistent performance at all times. Therefore, a different estimation approach is sought which satisfies the specific requirements for haptic teleoperation.



(a)

(b)

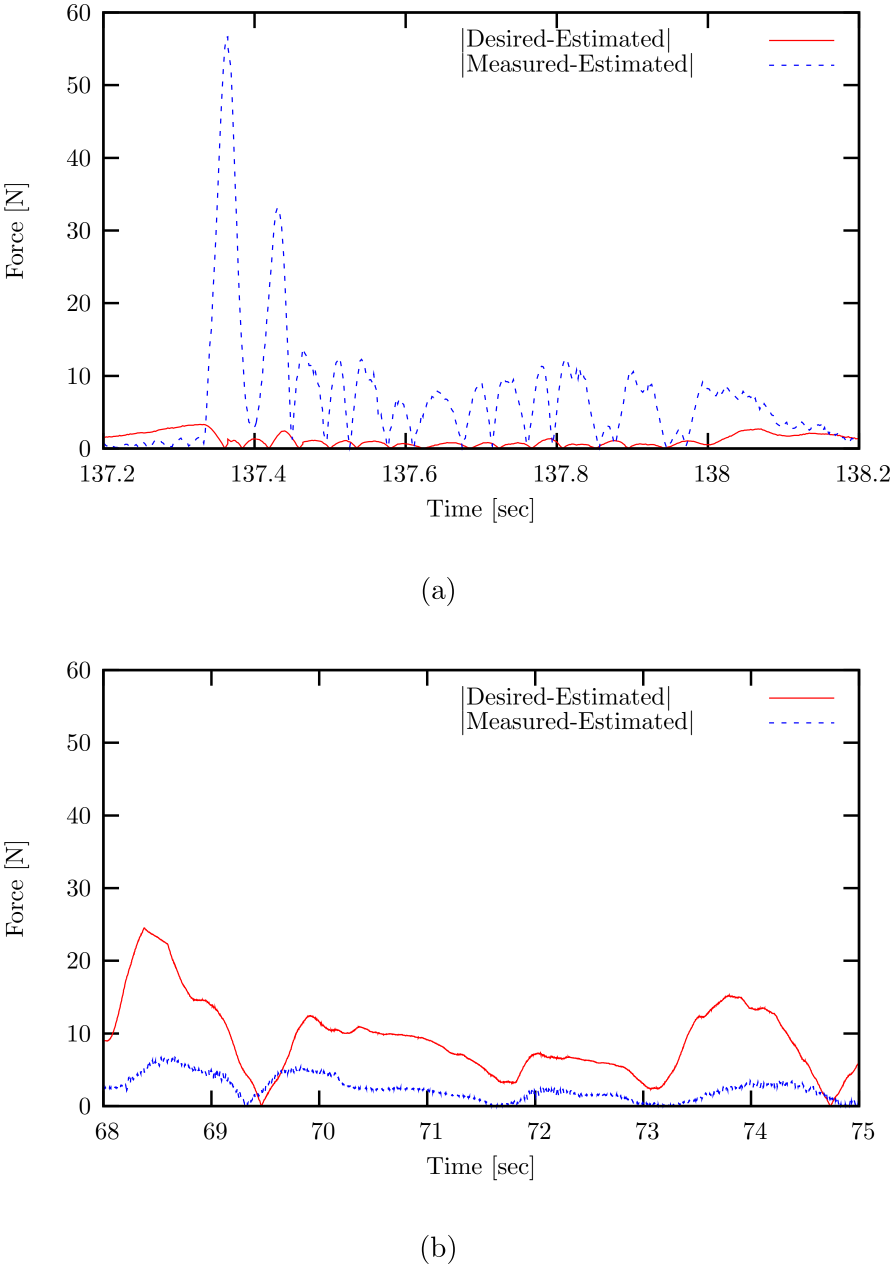
Figure 3.3: **The results of force control without adaptation.** (a) Nominal stiffness, $\hat{k}_s$, is 100 N/m and k_s changes from free space to 3000 N/m. (b) Nominal stiffness, $\hat{k}_s$, is 3000 N/m and k_s changes from free space to 300 N/m.

An approach for stiffness estimation in a haptically teleoperated system will be presented. The basis for the stiffness adaptation law used in this approach is derived from the fact that the responses of the measured and estimated contact forces are correlated to the stiffness modelling errors. Large deviations between the measured and estimated force responses indicate that the stiffness modelling error is larger than expected.

The different relationships between the desired, measured, and estimated contact force (f_d, f_m, and f_e), shown in Figure 3.3(a) and (b), are noticeable. The estimated or nominal stiffness $\hat{k}_s$ is used in the design of full state feedback and a Kalman estimator. Figure 3.3(a) shows the results when $k_s \gg \hat{k}_s$. The high deviation of contact force in a short time period occurs due to the under-estimation of stiffness. Because the controller perceives the environment to be more compliant than it actually is high feedback gains are chosen to control the contact force. In this case the controller over-compensates for the errors, resulting in under-damped responses. Eventually instability could arise if the stiffness mismatch becomes too large.

In contrast, Figure 3.3(b) demonstrates the typical results when $k_s \ll \hat{k}_s$. In this case the full state feedback gains are too small due to the over-estimation of stiffness. This results in under-compensation for errors and a sluggish response.

We notice that the measured and estimated contact forces in the first case ($k_s \gg \hat{k}_s$) are oscillatory in a very short time period. The estimated contact forces from the Kalman estimator are computed based on the weighting between two stochastic parameter matrices (measurement and processing noise uncertainties). The oscillatory response of the measurement in the first case causes the estimated values to be closer to the desired values than to the measured values. This is because the measurement updates in the Kalman estimator are unable to track the rapid changes in the measured value. Thus the estimate tends to average out over the measurement oscillations. In the second case ($k_s \ll \hat{k}_s$) we notice that the estimated contact force tracks the measured contact force reasonably well but that the desired contact force is not tracked well because of the low gain response of the controller. Figure 3.4 illustrates the differences between these two cases. It is observed that the difference between f_m and f_e is larger than the difference between f_d and f_e when $k_s \gg \hat{k}_s$.



(a)

(b)

Figure 3.4: **Comparison among the desired, estimated, and measured contact forces in teleoperation without adaptation.** (a) Nominal stiffness, $\hat{k}_s$, is 100 N/m and k_s changes from free space to 3000 N/m. (b) Nominal stiffness, $\hat{k}_s$, is 3000 N/m and k_s changes from free space to 300 N/m.

Conversely, the difference between f_d and f_e is larger than the difference between f_m and f_e when $k_s \ll \hat{k}_s$. The first part of the stiffness adaptation law is motivated by these very different response characteristics. The second part of the adaptation law is motivated by the fact that the system stiffness increases when larger contact force is applied.

The stiffness adaptation law combines these two aforementioned ideas.

$$\hat{k}_s^i = \hat{k}_{s,1}^{f,i} + \hat{k}_{s,2}^{f,i}. \tag{3.18}$$

where the superscript i indicates the discrete time step and the superscript f indicates the filtered value. The first part of the estimation is based on the relation between f_d, f_m, and f_e.

$$\hat{k}_{s,1}^i = \hat{k}_{s,1}^{i-1} + \Delta\hat{k}_{s,1}^i, \tag{3.19}$$

where

$$\begin{aligned}\Delta\hat{k}_{s,1}^i =& k_1|f_m - f_e|\,\sigma_d\left(c, \frac{|f_m - f_e|}{|f_e| + a_1} - b_1\right) \\ & - k_2|f_d - f_e|\,\sigma_d\left(c, \frac{|f_d - f_e|}{|f_e| + a_2} - b_2\right),\end{aligned} \tag{3.20}$$

and

$$\sigma_d(c, x) = \frac{1}{1 + e^{-cx}}. \tag{3.21}$$

The terms $k_1, k_2, a_1, a_2, b_1, b_2$ and c are positive parameters. The minimum of $\hat{k}_{s,1}$ is set to 0 N/m. The second part is

$$\hat{k}_{s,2} = k_{min} + k_3\,\sigma_d\left(c_0, |f_m| - f_0\right), \tag{3.22}$$

where f_0, c_0, and k_3 are positive parameters. The term k_{min} is set to 100 N/m in the experiments. Finally, low-pass filters are used to prevent jerking motions due to quick changes in the stiffness estimation. All the parameters are obtained experimentally: a_1, a_2, b_1, b_2, c, and c_0 are 1.0, 0.1, 1.5, 1.0, 5.0, and 0.2 respectively. The parameters f_0, k_1, k_2, and k_3 are 20 N, 10 m^{-1}, 10 m^{-1}, and 3000 N/m respectively.

The adaptation heuristic has been designed from the experimental data by trial and error. The robustness and effectiveness have been verified through numerous experiments. Stability of the system with this adaptation is guaranteed since the proposed adaptation increases the stiffness estimation whenever it shows the underdamped response. This lowers the control feedback gains such that the system is stabilized. Since this adaptation law is designed and adjusted for the specific experimental setup, a more general automatic procedure, such as neural networks or learning techniques, would be useful in applications to other systems. The successful implementation of this adaptation law shows great potential for refinement using advanced learning/adaptive techniques.

3.3 Teleoperation

The teleoperation approach is developed for each direction in operational space coordinates (i.e. one DOF system) since the control structure in Section 3.1 enables each Cartesian direction of a manipulator end-effector to be controlled independently. Figure 3.5 illustrates the proposed teleoperation approach where the master and slave systems are connected by a virtual spring. Force control is then used on the slave manipulator to eliminate the dynamics of the slave robot.

The desired force, f_d, for both master and slave systems is generated by the virtual spring constant, k_{vir}, due to the position error. The contact force on the slave end-effector is controlled to track the desired force, f_d. The force controller on the slave system is implemented using a modified Kalman Estimator with full state feedback (AOB). However, only feedforward control is used to generate the desired force on the master side since the device is light weight and has low friction.

Stability characteristics of the system are improved by providing the desired contact force to the operator rather than the measured contact force. The direct use of the measured contact force causes a delay in the loop and the stability of the system is greatly dependent upon the mass ratio of the master and slave systems [14].

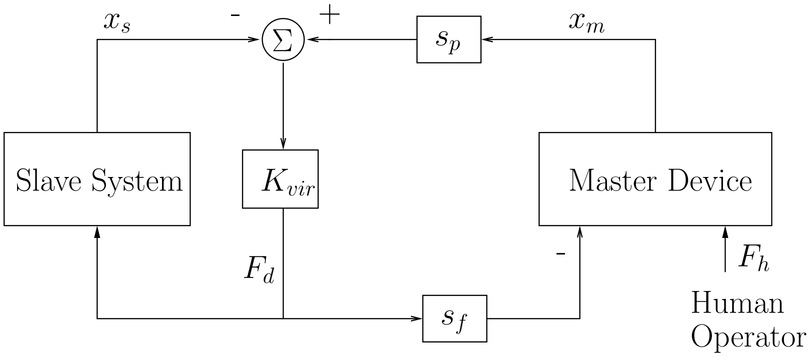


Figure 3.5: **Teleoperation approach.** The terms x_m, x_s, s_p and s_f are the master position, slave position, position scaling and force scaling, respectively. The terms s_p and s_f are 2.0 and 0.1 in the experimental setup and k_{vir} is the virtual spring constant that generates the desired force, F_d.

3.3.1 Telepresence

The user is always provided with the contact force, scaled by s_f, through the haptic device if the force control in the slave robot tracks the desired contact force well. Moreover, the transfer function, $\frac{X_m(s)}{F_h(s)}$, from the force of a human operator to the master position represents the compliance that the human operator feels at the master device [38]. Telepresence would be realized if the transfer function closely matches the slave system compliance.

In Figure 3.5, the master device is modeled as a mass damper system, having a transfer function of $1/(m_m s^2 + c_m s)$. The slave system represents the force controlled robot in contact with environment. Thus, the transfer function from the desired force to the slave position, $\frac{X_s(s)}{F_d(s)}$, is represented by $G_{se}(s)$. The equations of motion for the master and slave are

$$\begin{aligned}(m_m s^2 + c_m s)X_m(s) = \\ F_h(s) - s_f k_{vir}\{s_p X_m(s) - X_s(s)\}\end{aligned} \tag{3.23}$$

$$G_{se}(s)k_{vir}\{s_p X_m(s) - X_s(s)\} = X_s(s), \tag{3.24}$$

where $X_m(s)$ and $X_s(s)$ are the Laplace Transform of x_m and x_s. Moreover, the environment in contact is modeled to have a certain stiffness, k_s,

$$f_c = k_s x_s, \tag{3.25}$$

where f_c is the contact force with the environment. Therefore, $G_{se}(s)$ can be represented by

$$G_{se}(s) = \frac{X_s(s)}{F_d(s)} = \frac{1}{k_s}\frac{F_c(s)}{F_d(s)}. \tag{3.26}$$

$G_s(s) = \frac{F_c(s)}{F_d(s)}$ is the closed loop transfer function of the force controller in the slave system; thus, $G_{se}(s) \approx \frac{1}{k_s}$ within the bandwidth of the force controller. From (3.23) and (3.24), the transfer function $\frac{X_m(s)}{F_h(s)}$ can be derived as

$$\frac{X_m(s)}{F_h(s)} = \frac{k_{vir} + G_{se}^{-1}(s)}{\Delta}, \tag{3.27}$$

where

$$\Delta = (m_m s^2 + c_m s + k_{vir} s_p s_f)(k_{vir} + G_{se}^{-1}(s)) - k_{vir}^2 s_p s_f. \tag{3.28}$$

Equation (3.27) shows the characteristics of the proposed teleoperation approach. If $k_{vir} \gg |G_{se}^{-1}|$ at a low frequency range, $|m_m s^2 + c_m s| \ll k_{vir} s_p s_f$, the compliance that a human operator feels will be close to the environment compliance, $\frac{X_m(s)}{F_h(s)} \approx \frac{1}{s_p s_f k_s}$. At a high frequency range, $|m_m s^2 + c_m s| \gg k_{vir} s_p s_f$, it will be $\frac{X_m(s)}{F_h(s)} \approx \frac{1}{m_m s^2 + c_m s}$. Therefore, the key aspect for telepresence is to maintain $k_{vir} \gg |G_{se}^{-1}|$, i.e. $k_{vir} \gg k_s$, in addition to having a large force control bandwidth. The value of k_{vir} is limited by the stability. To maintain the ratio k_{vir}/k_s as large as possible within this limit, k_{vir} is updated on-line based on the estimated environment stiffness. That is, k_{vir} is increased with the estimate of k_s. The following equation is used in the experiments (Figure 3.6).

$$k_{vir} = 2000.0\sigma_d(0.007(\hat{k}_s - 1000.0)) + 1000.0 \tag{3.29}$$

where $\sigma_d(x) = \frac{1}{1+e^{-x}}$.

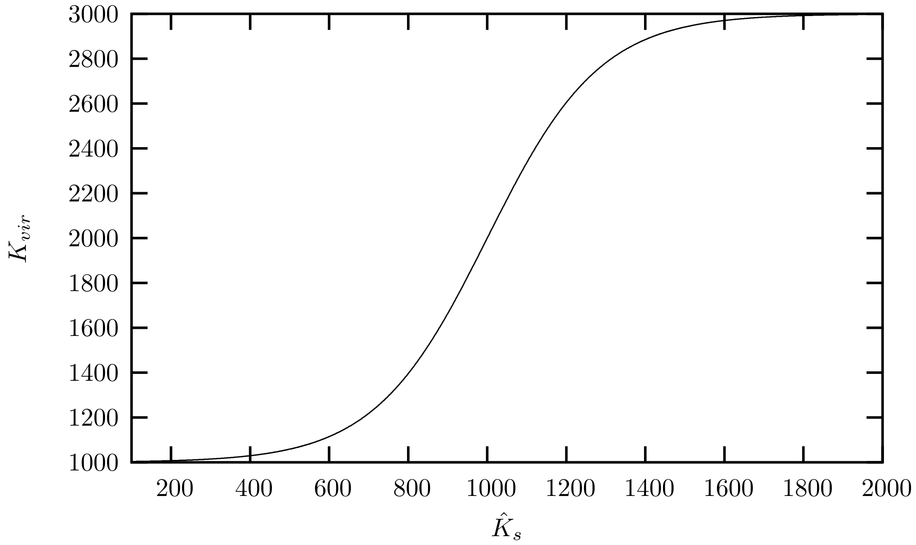


Figure 3.6: **Updates of k_{vir} with the estimate of k_s.** The term $\hat{k}_s$ is the estimate of the environment stiffness k_s.

3.3.2 Stability

The characteristic equation Δ of the loop is

$$\begin{aligned}\Delta =&(m_m s^2 + c_m s)G_{se}(s)^{-1}\\ &+ k_{vir}(m_m s^2 + c_m s + s_p s_f G_{se}(s)^{-1}).\end{aligned} \tag{3.30}$$

The system is stable for any k_{vir} if the model is perfect because $G_{se}(s)$ is a stable minimum system with a constant DC value. However, the feedback gains k_{vir} and k_{vir}/k_s are bounded by the physical limitations of the master device and the slave robot. Figure 3.7 shows the local feedback systems at the master and slave. Specifically, k_{vir} cannot exceed the maximum stiffness that the master device can produce. Also, k_{vir}/k_s is limited by the motion bandwidth of the slave manipulator. In free space motion, where k_s is small, this results in a greater limitation on the magnitude of k_{vir}.

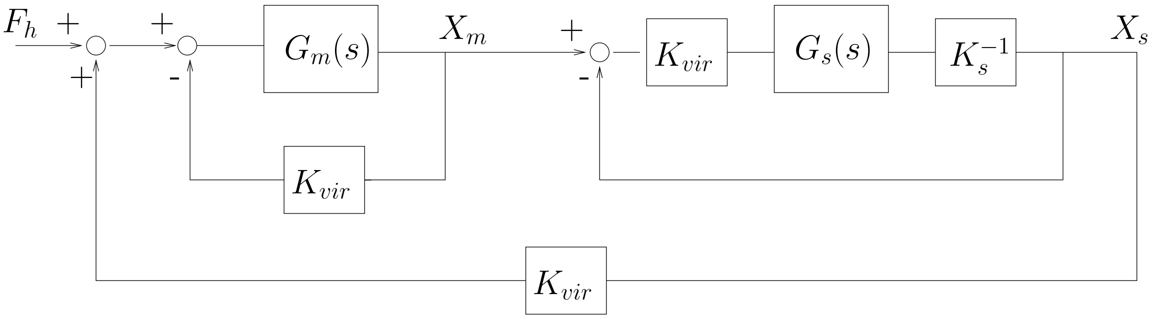


Figure 3.7: **Rearranged block diagram.** This highlights local feedback for each system in the proposed teleoperation approach. The terms $G_m(s)$ and $G_s(s)$ represent dynamics of the master device and the closed loop force control system of the slave manipulator. i.e. $G_m(s) = \frac{X_m(s)}{F(s)} = (m_m s^2 + c_m s)^{-1}$ and $G_s(s) = \frac{F_c(s)}{F_d(s)}$. The terms X_m and X_s are the position of the master and slave system. The term F_h is the force from a human operator and F_c and F_d are the contact force and the desired contact force in the slave manipulator. The term k_s is the environment stiffness.

3.3.3 Time delay

The block diagram in the presence of time delay is shown in Figure 3.8. Accounting for the time delay, Equations (3.23) and (3.24) become

$$\begin{aligned}(m_m s^2 + c_m s)X_m(s) =& F_h(s) - s_f k_{vir}\{s_p X_m(s)\\ &- X_s(s)e^{-T_d s}\}\end{aligned} \tag{3.31}$$

$$G_{se}(s)k_{vir}\{s_p X_m(s)e^{-T_d s} - X_s(s)\} = X_s(s) \tag{3.32}$$

Now, the transfer function from the force of the human operator to the position of the master device is

$$\frac{X_m(s)}{F_h(s)} = \frac{k_{vir} + G_{se}^{-1}}{\Delta'}, \tag{3.33}$$

where

$$\begin{aligned}\Delta' =& (m_m s^2 + c_m s + k_{vir} s_p s_f)(k_{vir} + G_{se}^{-1})\\ &- k_{vir}^2 s_p s_f e^{-2T_d s}.\end{aligned} \tag{3.34}$$

The effect of time delay on the performance is investigated using Padé approximation for small time delay, $e^{-2T_d s} = \frac{1-T_d s}{1+T_d s}$.

$$\left[\frac{F_h(s)}{X_m(s)}\right]_{\text{w delay}} = \left[\frac{F_h(s)}{X_m(s)}\right]_{\text{w/o delay}} + \frac{k_{vir}^2 s_p s_f \frac{2T_d s}{1+T_d s}}{k_{vir} + G_{se}(s)^{-1}} \tag{3.35}$$

The additional term, $\frac{k_{vir}^2 s_p s_f \frac{2T_d s}{1+T_d s}}{k_{vir}+G_{se}(s)^{-1}}$, can be further approximated as $2sT_d k_{vir} s_p s_f$ at a low frequency range when $k_{vir} \gg |G_{se}^{-1}|$. This shows the damping effect of the time delay. Stability is no longer guaranteed for large T_d.

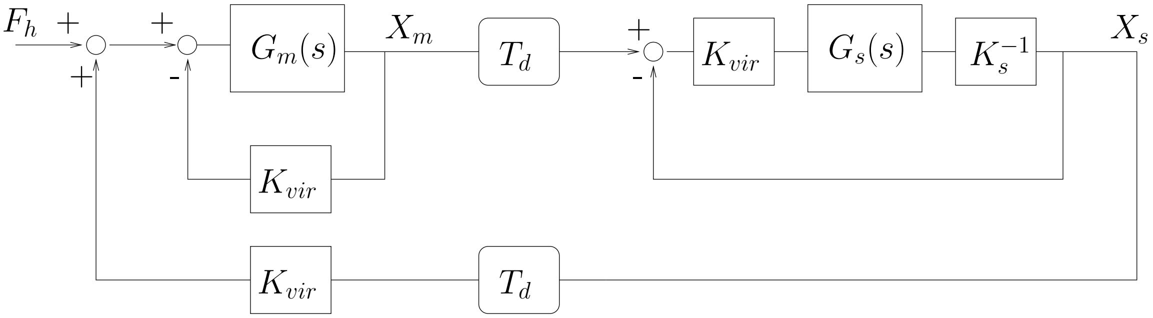


Figure 3.8: **Re-arranged block diagram with time delay.** Time delay is added to the block diagram in Figure 3.7.

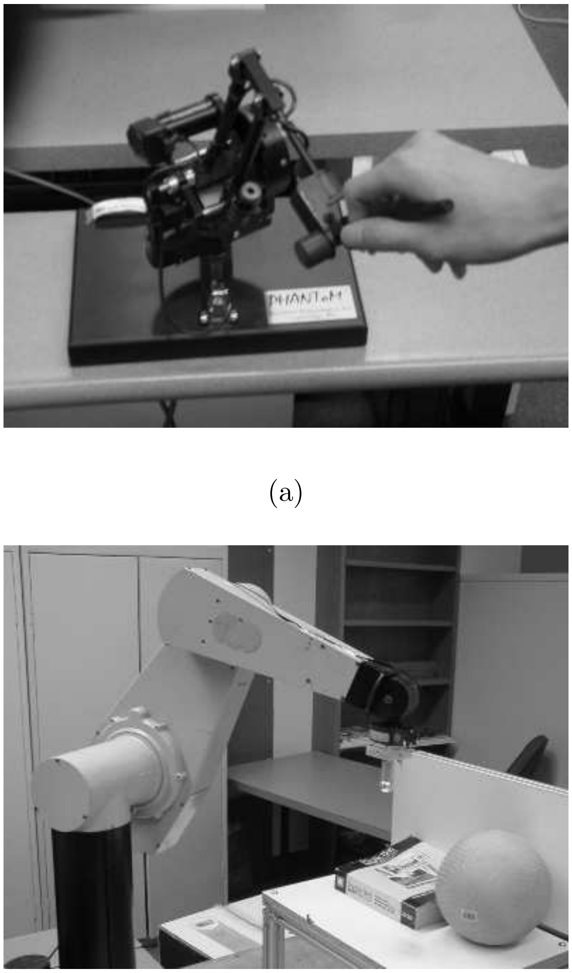
3.4 Experiments

3.4.1 Teleoperation without time delay

Experimental setup

The teleoperation approach with contact force control was implemented on an experimental setup. The experiments were conducted using a PUMA560 manipulator as a slave robot and PHANTOM 1.0A (SensAble technologies) as a master device. Figure 3.9 displays the photographs of the master and slave devices.

The PUMA560 manipulator was controlled through a TRC205 amplifier package from Mark V Automation Corporation. This amplifier package was connected to a



(a)

(b)

Figure 3.9: **Teleoperation system setup.** (a) PHANTOM device controlled by a human operator. (b) PUMA robot with soft and stiff objects in the workspace.

PC, running a QNX real-time operating system. The servo program in QNX commanded torques to the robot through this setup. The servo rate was set to 500 Hz. This PUMA560 robot has 6 DOF, including 3 revolute joints at the wrist. A JR3 force sensor was mounted at the wrist to measure the contact forces at the end-effector.

The PHANTOM 1.0A has 6 DOF for the position and orientation of the end-effector and it has 3 motors for the first three joints. Therefore, only force feedback on the translational motion was available from the device. The device ran on a PC with a LINUX operating system. Since it was not a real-time operating system, the servo rate varied slightly. An average servo rate was about 10,000Hz.

The two stations were connected by a local area network. Therefore, the time delay between the two systems was negligible. The experiment was designed to demonstrate the performance of the system when there was no time delay. Throughout the experiments, the translational directions of the PUMA end-effector were controlled by teleoperation through force control. The end-effector rotation was controlled to maintain a fixed orientation. Direct vision was provided to the operator at the master station. No training was required to operate the system since the teleoperation system was intuitive to operators.

Experimental results

Force control stability depends on the accuracy of the model [11]. In our setup, it can be shown that stability is assured if the nominal stiffness does not deviate too much from the actual stiffness. However, stiffness adaptation is mandatory to guarantee robustness in teleoperation tasks, where the robot interacts with various objects.

Figure 3.11 shows a teleoperation experiment using a PID force controller. The slave robot was in free space at the beginning of the experiment. It was then teleoperated to contact three different contact surfaces: a sponge, a book, and a table. The robot was in free space[1] between contacts. The gains for the PID force controller were chosen experimentally such that the bandwidth and the performance of the force controller were maximized ensuring stability at hard contact. However, the bandwidth of the force controller varied for different contacts because the fixed gains were used

[1] The measured force data in free space is not zero due to the gripper inertia.

when the slave system itself was changed. The system responded well in the soft contact (region 1 in Figure 3.11); however, the contact forces on the hard surfaces were marginally stable (region 2 and 3 in Figure 3.11). Also, the bandwidth during free space motion was too small; thus, the human operator could feel the significant position errors and this made it difficult to manipulate the slave robot. The results again emphasize the need for stiffness adaptation and a robust control scheme for varying stiffness.

Figure 3.12 shows the same experiment as in Figure 3.11 using the proposed force control method. Measured and estimated forces closely matched the desired force independent of the contact surface. In free space motion the slave robot tracked the master position with a designed high bandwidth; thus, the user only felt very small drag. Much better telepresence was achieved due to significantly improved performance of force control in contact and motion tracking in free space.

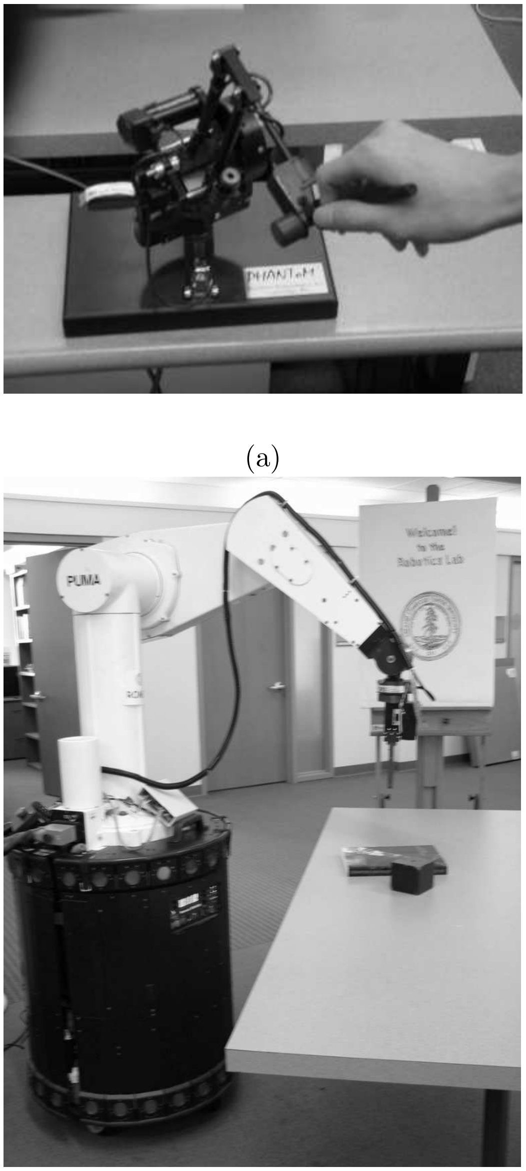
In Figure 3.12(b), the stiffness estimation was proportional to the real stiffness, and in free space it reached the minimum value of 100 N/m. The stiffnesses for the sponge, book and table were identified off-line as 300, 6000, and 50000 N/m respectively at low contact forces.

Comparison between Figure 3.13 and Figure 3.4 show that the stiffness adaptation significantly reduced the differences among the desired, measured, and estimated fores at the contact with different environments.

3.4.2 Teleoperation with a mobile manipulator

Experimental setup

The Stanford mobile platform was used in this experiments (Figure 3.10). The mobile platform consists of a PUMA560 manipulator mounted on a XR4000 mobile base (Nomadic Technologies). The Nomadic XR4000 mobile base is a holonomic robotic vehicle which has 4 powered casters. The driver for the XR4000, implemented in QNX, took as input two translational forces and one torque about the vertical axis. Thus, the robot was treated as having two prismatic joints and one revolute joint [25]. Combined with the PUMA560 manipulator, the mobile manipulator had 9



(a)

(b)

Figure 3.10: **Teleoperation system using a mobile manipulator.** (a) PHANTOM device controlled by a human. (b) PUMA robot mounted on XR4000 (ROMEO)

degrees of freedom. The PHANTOM 1.0A was again used as a master device. The teleoperation system used a wireless local network system for communication. The system experienced a 26ms delay in one direction.

The experiment demonstrated not only the time delayed teleoperation system but also the application of the teleoperation approach to a mobile manipulation platform with task redundancy. The translational directions of the end-effector were controlled by teleoperation through force control while the end-effector rotation was controlled to maintain a fixed orientation. The XR4000 mobile base was controlled to track a trajectory autonomously in the null space.

Experimental results

Two sets of experiments were conducted to show the decoupling of the end-effector control from the base control. Only the PUMA robot was controlled through teleoperation in the first set of experiments while the base was turned off. The results are shown in Figures 3.14 through 3.17. The second set of experiments was conducted while both the PUMA and the base were operational. The base was controlled to move in the lateral direction (i.e. along the table in Figure 3.10) using null space control. The desired trajectory was a sine function with an amplitude of 20 cm and a period of 12 seconds as shown in Figure 3.18. The results are shown in Figures 3.19 through 3.22. The fact that there was little difference in the performance of the two sets of experiments demonstrated the effectiveness of the decoupled control structure.

In both experiments, the operator began moving the slave manipulator in free space (i.e. no contact) through teleoperation. Different objects were then contacted sequentially by the end-effector of the slave manipulator. These included a sponge, a book, and a table. The robot was in free space between contacts (i.e. where $F_m \approx 0$). The operator was asked to contact different objects with the same amount of force. The stiffness of the sponge and the book were computed off-line using the measured contact force and contact position. They were approximately 300 and 6000 N/m, respectively, at the low values of contact force (up to about 20 N). Their stiffnesses increased as much larger forces are applied. The table in the experiments was composed of aluminum plates and frames. The off-line computation of its stiffness was limited due to the resolution of the end-effector position measurement. Based on the material properties and the geometry of the table, the stiffness was computed to be approximately 50,000 N/m.

Figures 3.14 and 3.19 show the contact force in the vertical direction at the end-effector of the slave manipulator. The manipulator was in free space in the region where the measured contact forces were near zero. These results demonstrate that the force controller with AOB and stiffness adaptation performs well even in the presence of vastly different environmental changes. The estimated environment stiffness $\hat{k}_s$ was updated quickly and accurately enough to achieve the designed performance and stability criteria as shown in Figures 3.16(a) and 3.21(a). The effectiveness of the stiffness adaptation can be observed by comparing Figures 3.15 and 3.20 with Figure 3.4. The differences among the desired, measured, and estimated fores have been significantly reduced by effective adaptation to the different environments.

It is noted that stable transition from free space to the three surfaces was achieved. The velocities of the end-effector were approximately 5 cm/sec at the impacts to all three environments as shown in Figures 3.16(b) and 3.21(b). The stability was maintained, particularly during the transition to hard surfaces (the book and the table). The updates of the stiffness in these cases were within 0.1 *sec*. The stability of the contact force control in transition was realized by effective stiffness adaptation and robust contact force control.

The desired contact force for the slave end-effector and the applied haptic force are compared in Figures 3.17(a) and 3.22(a). Without time-delay, they would be exactly the same. However, the effect of time-delay results in large differences especially in free space motion. This contributes to the damping effect that the operator feels in free space operation. Figures 3.17(b) and 3.22(b) compare the measured contact force with the haptic force applied to the operator. These plots show how close the contact force measurement and the haptic feedback are. The force feedback to the user closely matches the measured force during contact. However, the user feels a damping effect in free space motion.

Position tracking performance is shown in Figures 3.16(b) and 3.21(b). It is noted that the positions of the haptic device and the robot end-effector successfully follow each other in free space motion. In contrast, the virtual spring generates the desired contact force for both master and slave robots based on the difference between the master and slave positions.

3.5 Conclusion

The integration of contact force control with stiffness adaptation and a virtual spring provides a modular and systematic control structure for teleoperation. The translational motion of the robot is always controlled by the contact force control and on-line stiffness estimation without any discrete switching. In the case of no contact it is assumed that the robot is in a very compliant contact.

Active force control enables the overall teleoperation approach to have the robustness and performance needed for providing force feedback to a human operator. The bandwidth of the active force control determines how realistic the force feedback is. Transitions between environments with various stiffnesses do not involve any switching in the control structure, as the on-line stiffness adaptation performs effectively to match large changes in the environment. The stability margin due to the transmission time delay was analyzed and the implementation on the experimental setup demonstrated that the system is stable in the presence of 52 ms round-trip communication delay. The experimental results on the mobile platform demonstrated that the proposed teleoperation approach fits well within the operational space formulation. The user only controls the translational motion of the end-effector through teleoperation, while the other DOF are autonomously controlled without disturbing the task.

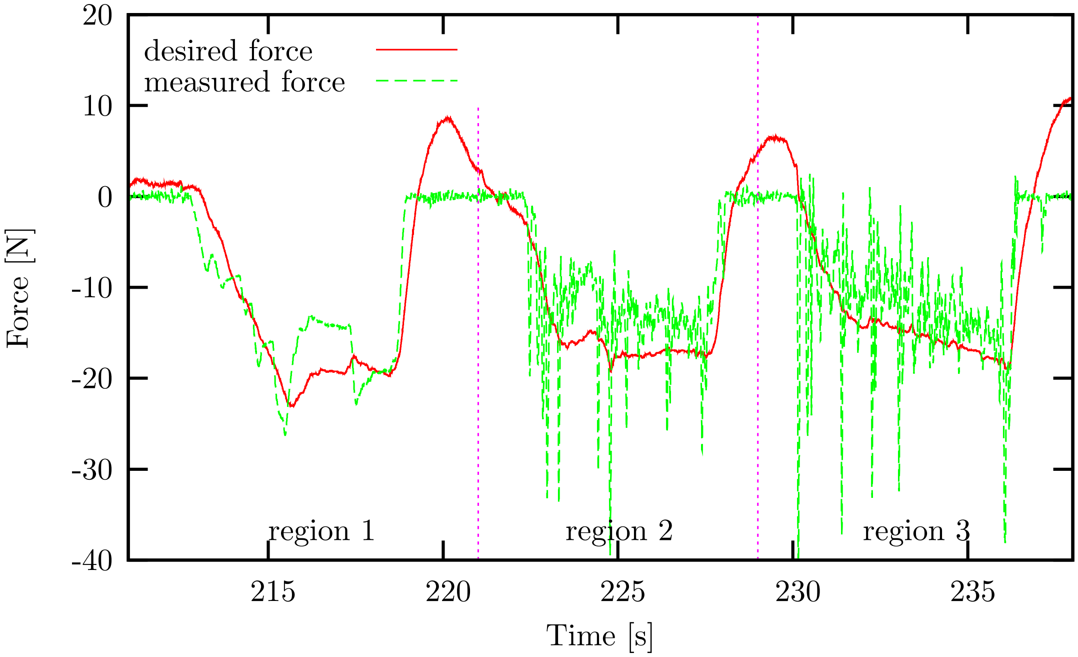
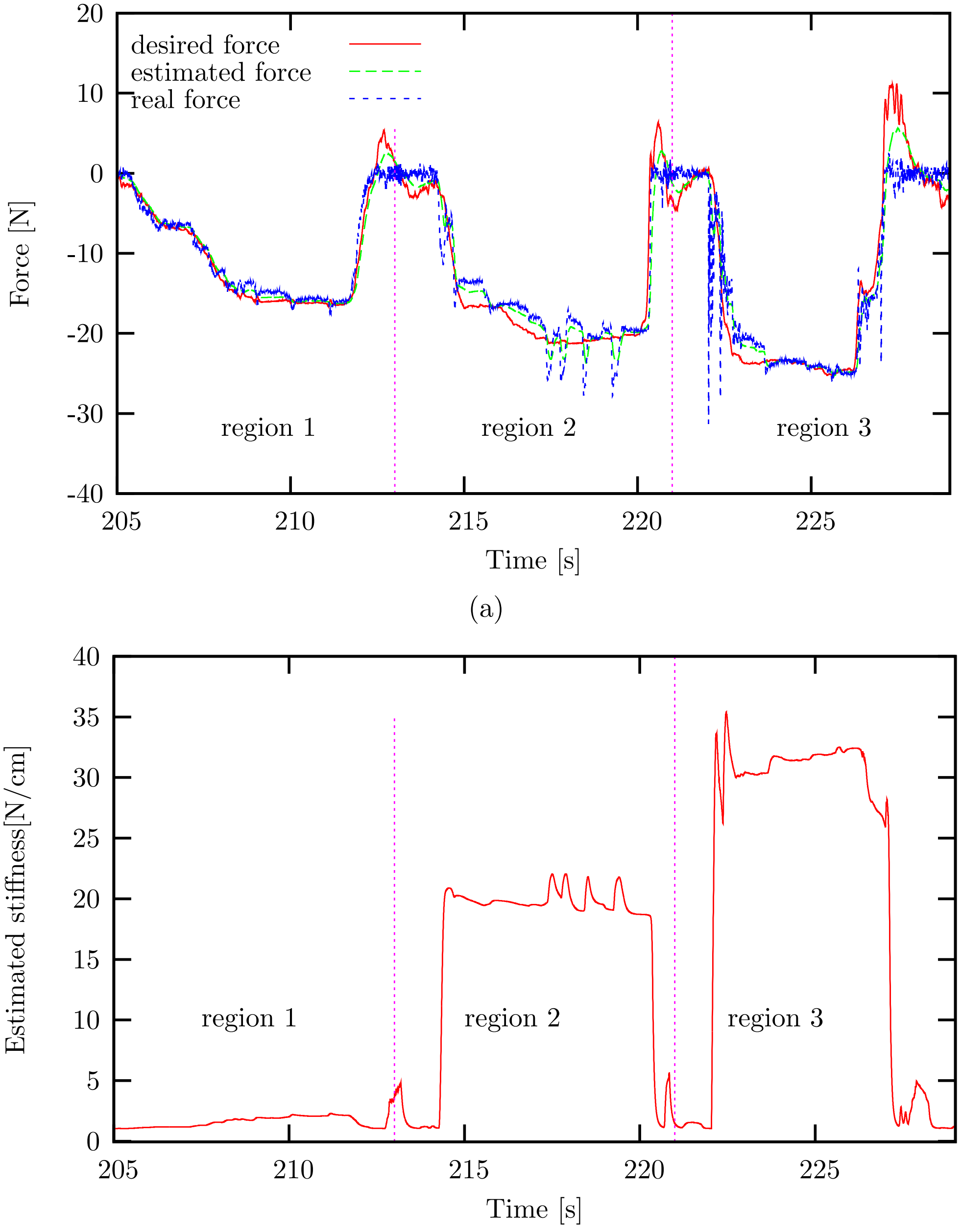


Figure 3.11: **Teleoperation using a PID force controller - force response in the vertical direction.** Sponge contact (region 1). Book contact (region 2). Desk contact (region 3).



(a)

(b)

Figure 3.12: **Teleoperation using the AOB and adaptation - force response and estimated stiffness in the vertical direction.** Sponge contact (region 1). Book contact (region 2). Desk contact (region 3). (a) force data. (b) estimated stiffness.

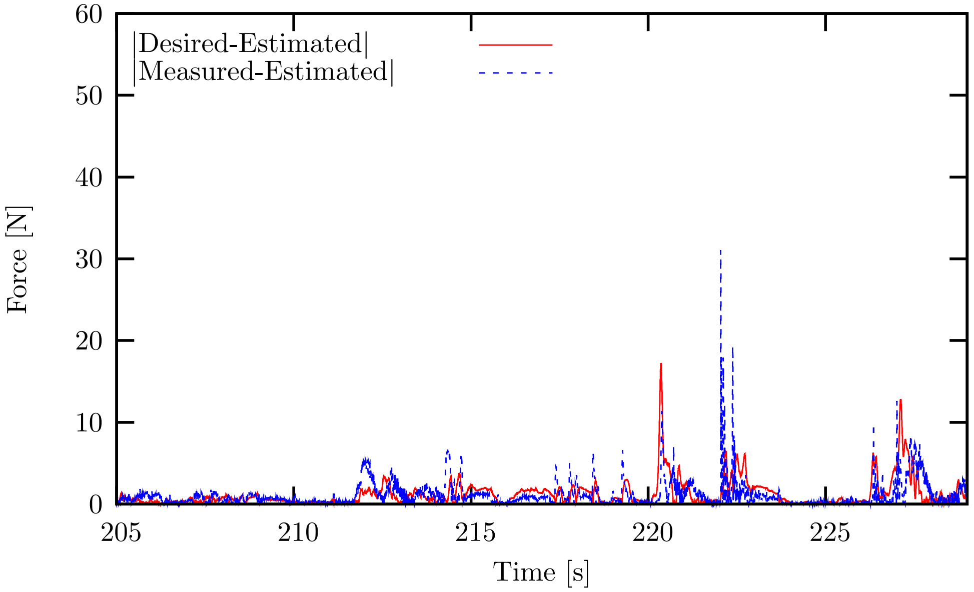
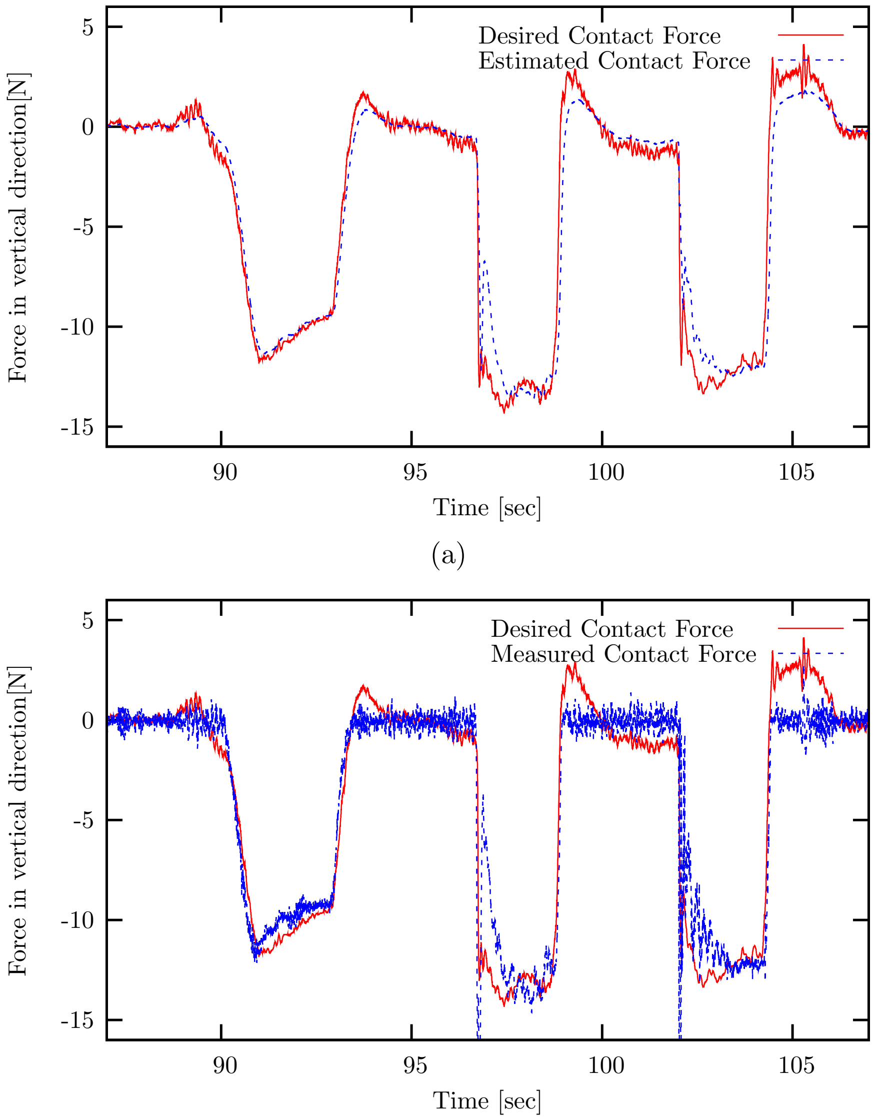


Figure 3.13: **Teleoperation using the AOB and adaptation - comparison among the desired, estimated, and measured contact forces.**



(a)

(b)

Figure 3.14: **Fixed base teleoperation with time delay - force response in the vertical direction.** The desired, estimated, and measured force at the end-effector of the slave manipulator are compared in (a) and (b). The desired force is generated by a virtual spring, $k_{vir}(x_m - x_s)$. The estimated force is from the AOB (a modified Kalman Estimator). The force is measured by the JR3 wrist force sensor.

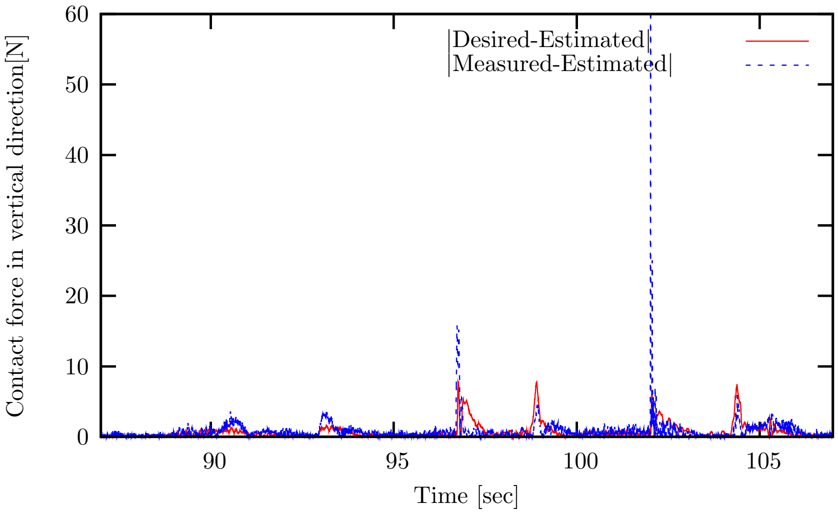
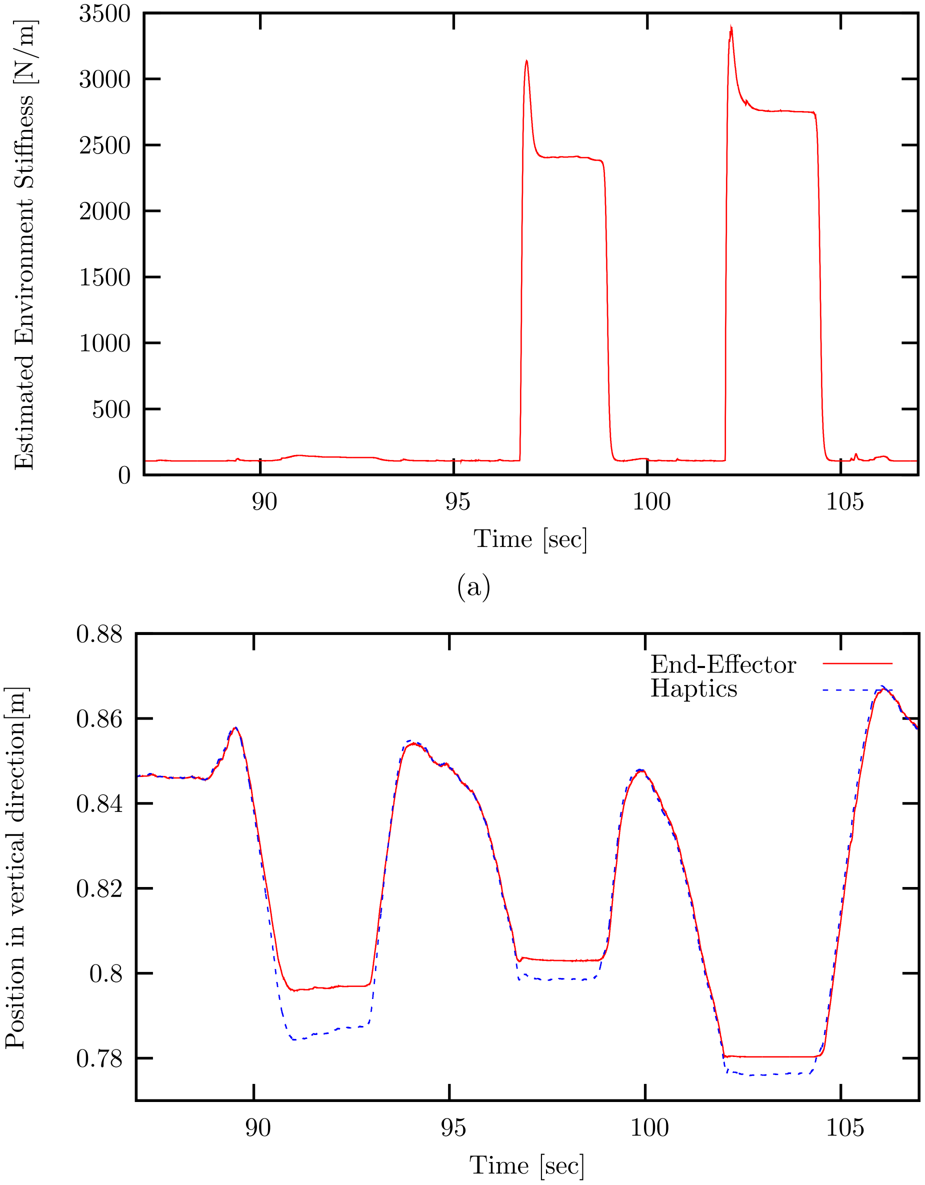


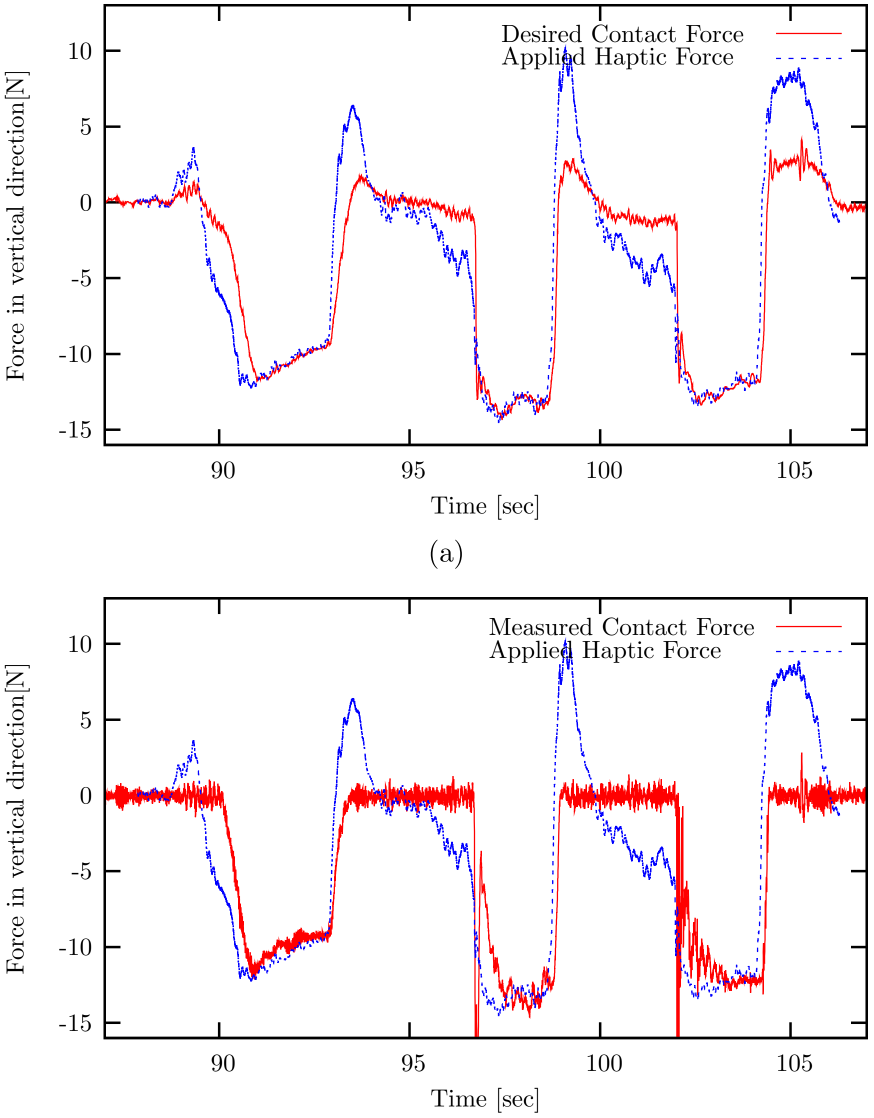
Figure 3.15: **Fixed base teleoperation with time delay - comparison among the desired, estimated, and measured contact forces at the end-effector of the slave manipulator.**



(a)

(b)

Figure 3.16: **Fixed base teleoperation with time delay - $\hat{k}_s$ and position tracking in the vertical direction.** (a) The estimated stiffness $\hat{k}_s$ for the environment that the end-effector of the slave manipulator is in contact with. (b) The end-effector of the slave manipulator tracking haptic position in the vertical direction.



(a)

(b)

Figure 3.17: **Fixed base teleoperation with time delay - force comparison between the slave and master.**(a) The haptic force and the desired contact force. The differences are associated with the effect of time-delay. The haptic force is multiplied by a scaling factor of 10 for comparison. (b) The haptic force and the measured contact force.

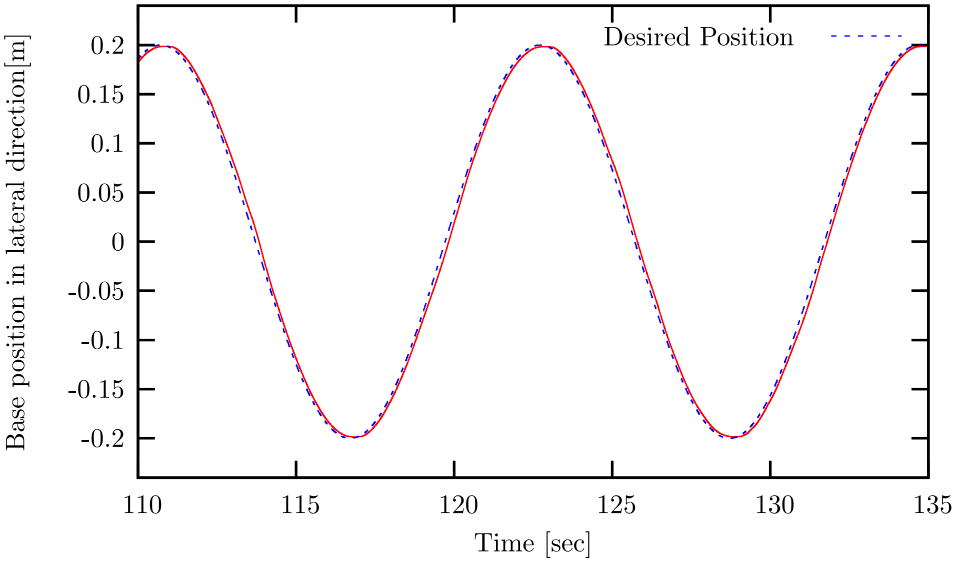
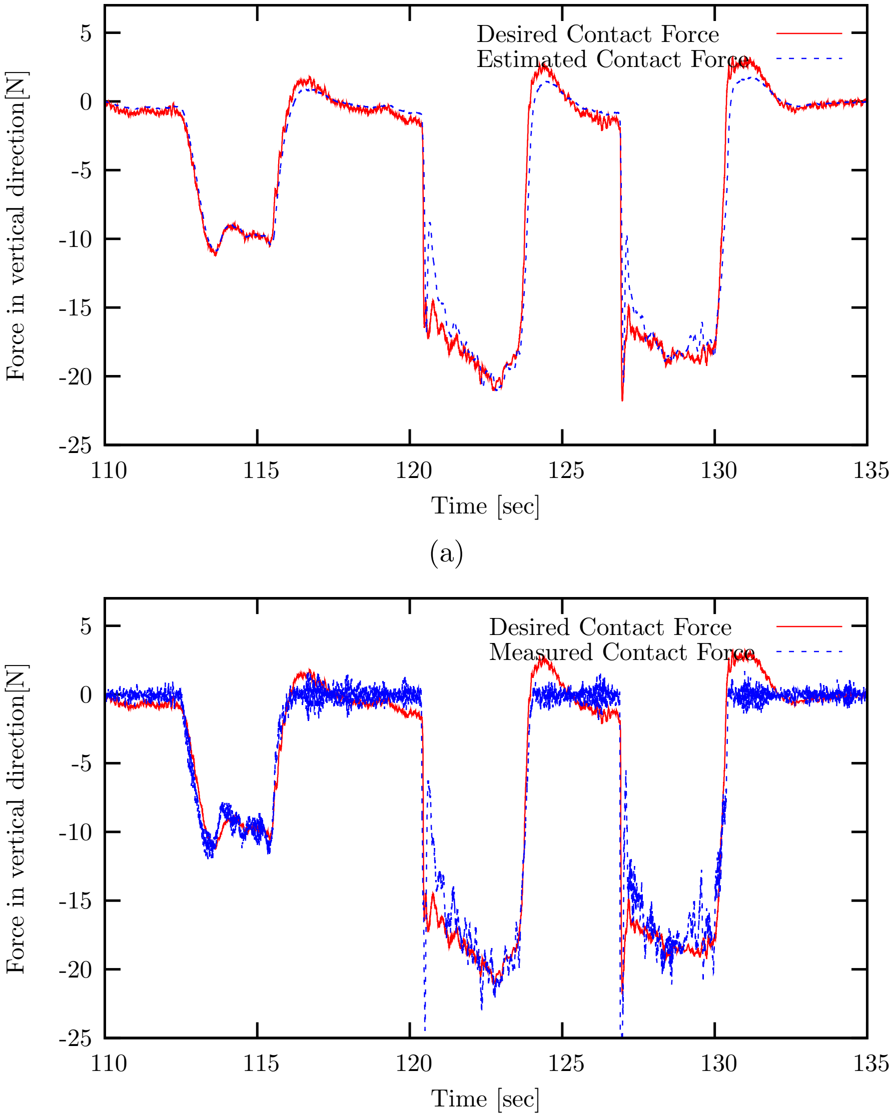


Figure 3.18: **Moving base teleoperation with time delay - base motion.** The base moves in the lateral direction, (i.e. along the table in Figure 3.10) while the end-effector of the slave manipulator is controlled by teleoperation. The amplitude is 20 cm and the period is 12 seconds.



(a)

(b)

Figure 3.19: **Moving base teleoperation with time delay - force response in the vertical direction.** The desired, estimated, and measured force at the end-effector of the slave manipulator are compared in (a) and (b). The desired force is generated by a virtual spring, $k_{vir}(x_m - x_s)$. The estimated force is from the AOB (a modified Kalman Estimator). The force is measured by the JR3 wrist force sensor.

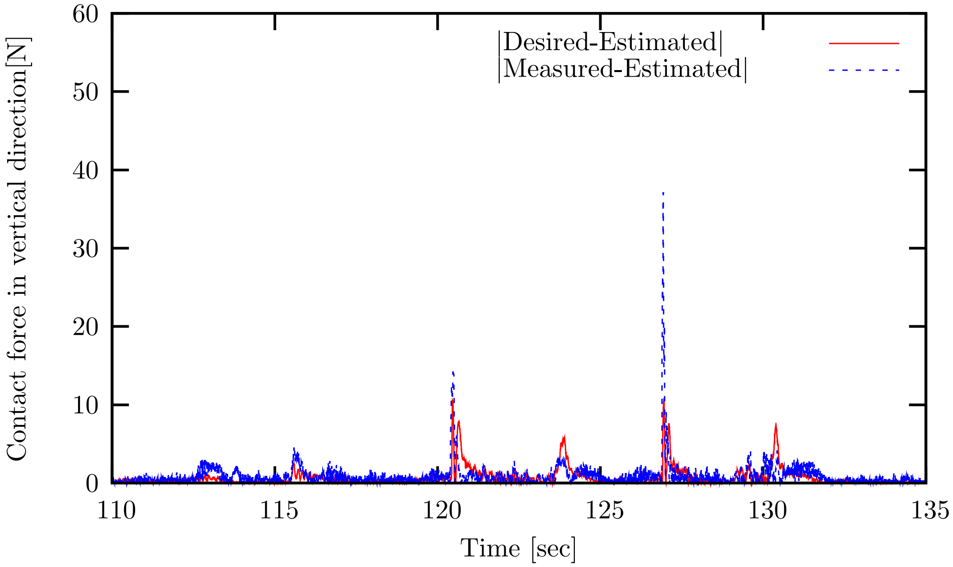
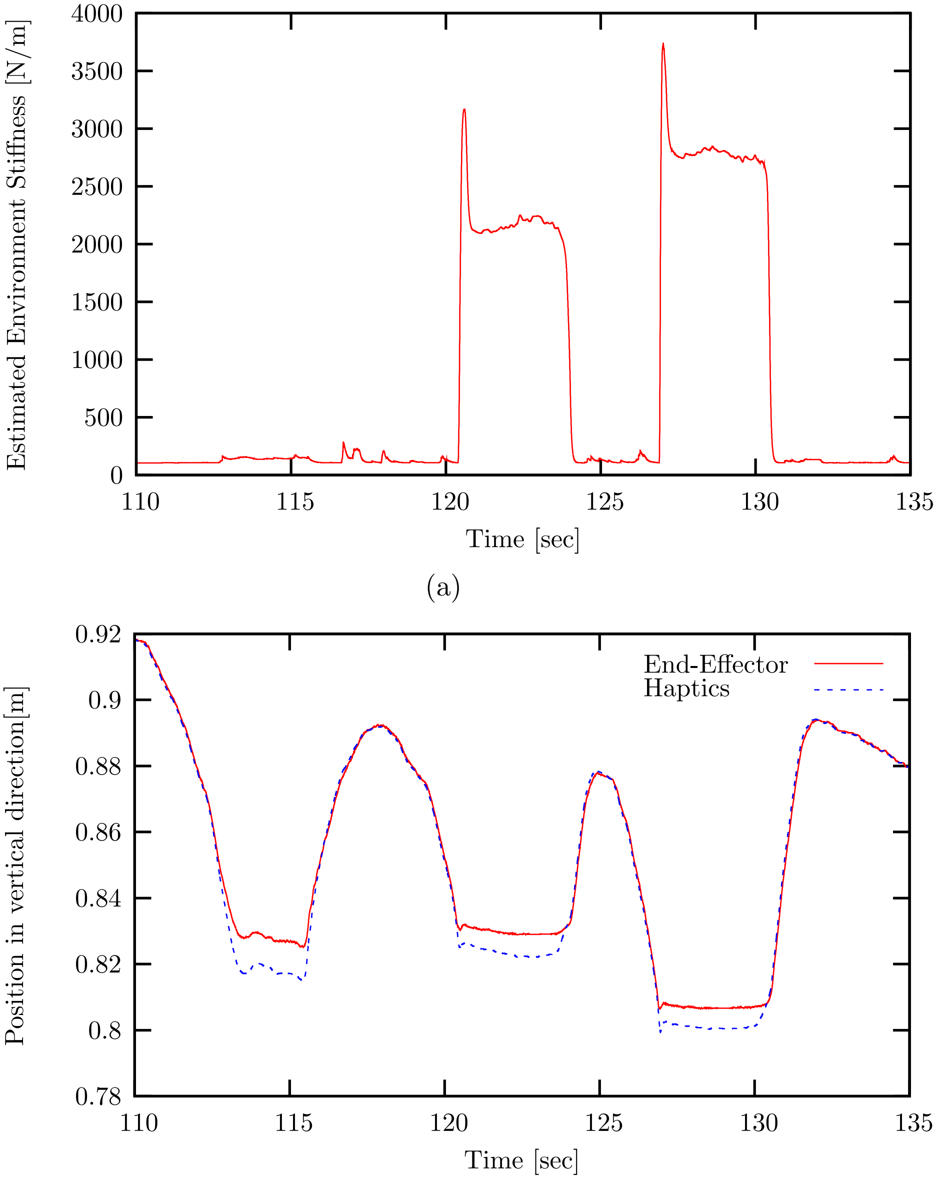


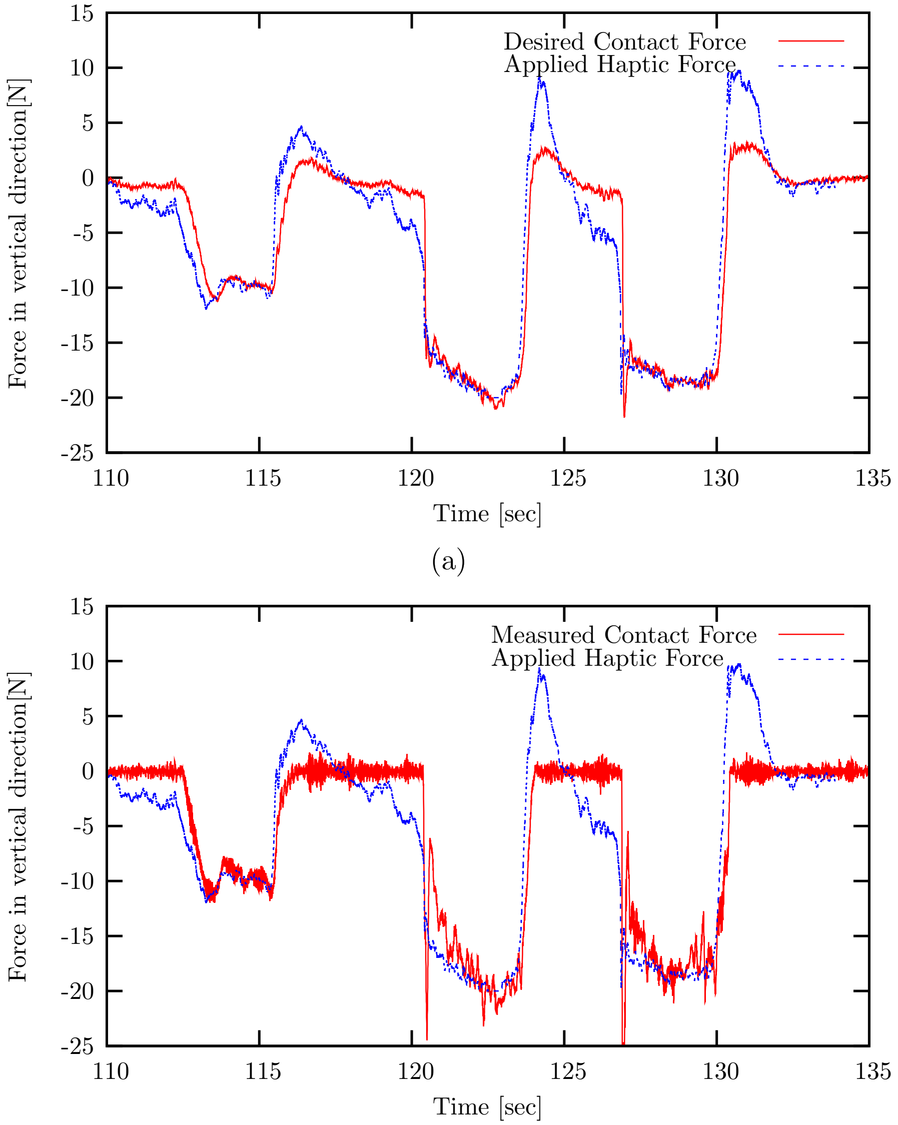
Figure 3.20: **Moving base teleoperation with time delay - comparison among the desired, estimated, and measured contact forces at the end-effector of the slave manipulator.**



(a)

(b)

Figure 3.21: **Moving base teleoperation with time delay - $\hat{k}_s$ and position tracking in the vertical direction.** (a) The estimated stiffness $\hat{k}_s$ for the environment that the end-effector of the slave manipulator is in contact with. (b) The end-effector of the slave manipulator tracking haptic position in the vertical direction.



(a)

(b)

Figure 3.22: **Moving base teleoperation with time delay - force comparison between the slave and master.** (a) The haptic force and the desired contact force. The differences are associated with the effect of time-delay. The haptic force is multiplied by a scaling factor of 10 for comparison. (b) The haptic force and the measured contact force.

Chapter 4

Multi-link multi-contact force control

In recent years the robotics community has been witnessing the emergence of humanoid robots that are designed to operate in complex human environments. Operating under these conditions inevitably involves complex interactions with the physical environment. For this reason it is critical to incorporate a robust framework for multi-contact resolution into future controllers.

Most research in motion and force control strategies have dealt with the contact at the end-effector of manipulators. Compliant frame selection matrices were introduced by [51] to select *compliant* directions to interact with the environment. Generalized selection matrices were later presented [31] to describe the decomposition of the end-effector space in the contact frame. However, these selection matrices are still limited in the sense that orthogonal decomposition is used at the control point of the end-effector. In the general contact configuration of the end-effector the contact force space and motion space may not be orthogonal to each other. This problem has been discussed by [17, 4, 39, 64] and they presented more general kinematic contact models. These references represent the contact kinematic model in the case of rigid contacts between the robot and environment.

A control strategy should account for the compliance or dynamics of the environment since the robot may have other contacts than rigid body contact. In practice, there is always compliance between the object and the robot. Therefore, introducing a model of stiffness at the contact produces a more general control framework. My contribution is developing a control framework combining a multi-contact kinematic model with an environment model such that the controller can deal with a general contact state.

More complex tasks with higher DOF manipulators may involve more contacts on multiple links, rather than contact exclusively at the end-effector. The multi-contact models [17, 4, 39, 64] are specifically focused on contacts at the end-effector. Therefore, the extension to the case of multi-link is necessary. Liu et al. [40] present an adaptive control approach for multiple geometric constraints using joint-space orthogonalization. The joint-space orthogonalization scheme projects the desired joint space velocity, and errors, onto the tangential plane of the geometric surface for joint space motion control. In this manner the contact force control is decoupled from the motion control. Since the projection is on joint space, the geometric constraints can be any function of the joint angles. However, the adaptive hybrid control approach used in this scheme does not provide a decoupled control structure for each contact, although it does guarantee convergence to the desired contact forces. Also, a model of the contact environment is not included in the control design.

In this chapter, a multi-contact multi-link control framework is proposed to incorporate the multi-link kinematic model and the environment model. The approach composes the operational space coordinates with contact force space including all the contacts over the links. The operational space formulation is then applied to obtain the dynamic equation for the contact force and motion control in the null space. Control points (operational points) are chosen at the links that are in contact with the environment. The multi-contact model [17] divides the space of each control point into a contact normal space and a free motion space. The composition of the contact normal spaces on the multiple links defines the operational space coordinates. The contact environment model is then incorporated into the operational space framework, resulting in a decoupled linear second-order system for each contact force.

In each contact force direction the contact force controller again utilizes the AOB [11], which was introduced and fully explained in Chapter 2. Experiments were conducted on a PUMA560 manipulator to demonstrate the multi-contact and multi-link approach. The experiments on multiple contacts over the multiple links are, to my knowledge, the first demonstration of multi-contact motion and force control on multiple links.

4.1 Multi-contact kinematic model

For the lth link having r contacts with environment, the free motion space and contact normal space with respect to the control point of the link can be described by[1] [17]

$$\begin{aligned} \vartheta^l_{c.p.} &= T^l_t \beta^l_t & (4.1) \\ f^l_{c.p.} &= N^l_c f^l_c, & (4.2) \end{aligned}$$

where $\vartheta^l_{c.p.}$ is the free motion space velocity of the control point, which does not disturb the contact forces. The term $f^l_{c.p.}$ is the force/moment at the control point due to the contact forces and f^l_c is the vector of r contact forces/moments. The dimensions of $f^l_{c.p.}$ and f^l_c are 6 and r, respectively. The matrix T^l_t spans the free motion space and N^l_c spans the contact normal space. We note that N^l_c and T^l_t are orthogonal to each other, i.e. ${N^l_c}^T T^l_t = 0$. These spaces and matrices are described at the control point of the link that has contact with environment; thus, the dimensions of N^l_c and T^l_t are $6 \times r$ and $6 \times (6 - r)$.

A link in two point contact is illustrated in Figure 4.1. In this example, The matrix, N^l_c, is composed of n_1, n_2, l_1, and l_2. Therefore, each component of f^l_c represents the contact forces on each contact.

$$N^l_c = \left[\begin{pmatrix} n_1 \\ n_1 \times l_1 \end{pmatrix} \quad \begin{pmatrix} n_2 \\ n_1 \times l_2 \end{pmatrix} \right] \tag{4.3}$$

[1] To clearly distinguish contact forces, f^l_c, from the measured contact force/moment at the control point, $f^l_{c.p.}$, some of the notation is changed from that in [17]. Also, the subscript c is introduced to the contact normal space matrix, N^l_c, to differentiate it from the null space projection matrix, N^T_0, which will be introduced in the following subsection.

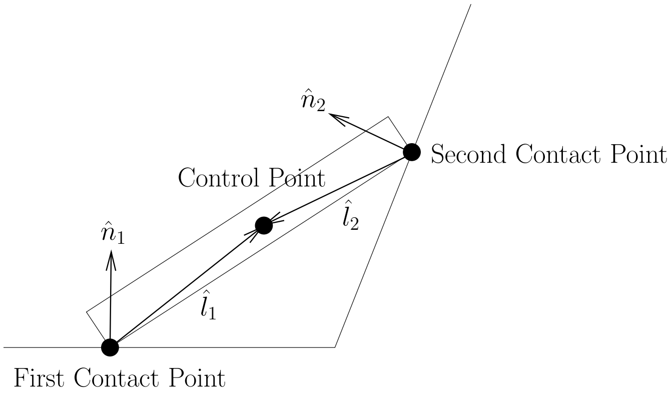


Figure 4.1: **Multi-contact on a link.** The terms n_1 and n_2 are unit vectors normal to the contact surfaces, respectively. The terms l_1 and l_2 are the vectors from the corresponding contact point to the control point of the link.

To achieve contact force control, the control force has to be applied in the space of N_c^l and the desired motion has to be designed in T_t^l for the corresponding control point. The implementation of this multi-contact force control can be accomplished by using projection matrices when only one link has contact with environment. The projection matrices for motion space and contact force space are obtained from the kinematic model, i.e. N_c^l and T_t^l [17]. Having the general projection matrices, Ω_f and Ω_m, the hybrid motion/force control structure presented in Chapter 2 is applied to obtain the decoupled dynamic equations for each space [48].

However, when multiple links have contact with the environment, multiple control points on the individual links have to be chosen to describe the contacts using contact normal spaces. Therefore, the use of projection matrices for control points is no longer as straightforward as in the case of a single control point. A better approach is to define the operational space coordinates to be the displacements in the contact normal spaces.

4.1.1 Operational space coordinates for contact space

Equation (4.2) provides the contact normal space matrix, N_c^l, for the contacts on the lth link. The operational space coordinates are defined to be the displacements in the contact normal spaces, N^l_c. The instantaneous velocity of the coordinates will be denoted as ϑ^l_c. The Jacobian for these coordinates is

$$J^l_c = N^{l^T}_c J^l, \tag{4.4}$$

where J^l is the Jacobian for the control point of the lth link.

For n link contacts, the Jacobian is obtained by constructing the composition of these Jacobians for each link.

$$J_c = \begin{pmatrix} J^l_c \\ J^{l+1}_c \\ \vdots \\ J^{l+n-1}_c \end{pmatrix}. \tag{4.5}$$

Similarly, a concatenation of ϑ^l_c vectors forms the instantaneous velocity of the operational space coordinate, ϑ_c, and a concatenation of f^l_c forms the contact force vector, f_c.

In fact, this operational space coordinate can be obtained in an alternative way. That is, we define the extended operational space as the position and orientation of all the contact points. Then, the contact force/moment directions can be selected to be the operational space coordinates. The resulting operational space and contact Jacobian, J_c, will be exactly the same as derived in this section.

4.2 Control structure

4.2.1 Contact force control

The equations of motion for manipulators are

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) + J_c(q)^T f_c = \Gamma, \tag{4.6}$$

where q is the vector of joint space coordinates. Γ is the vector of joint torques. $A(q)$ is the joint space inertia matrix. $b(q, \dot{q})$ is the vector of Coriolis and centrifugal terms. $g(q)$ is the vector of gravity terms.

The joint torque, Γ, is chosen to be composed of the torque for contact force control and null space torque [31].

$$\Gamma = J_c^T F_c + N_c^T \Gamma_0, \tag{4.7}$$

where the first term, $J_c^T F_c$, is the control torque for the contact force control and the second term, $N_c^T \Gamma_0$, is the torque in the null space of the contact force control. The equation of motion for ϑ_c is then

$$\Lambda_c(q)\dot{\vartheta}_c + \mu_c(q, \dot{q}) + p_c(q) + f_c = F_c, \tag{4.8}$$

where

$$\begin{aligned}
\Lambda_c^{-1}(q) &= J_c(q)A(q)^{-1}J_c(q)^T && (4.9)\\
\bar{J}_c^T(q) &= \Lambda_c(q)J_c(q)A(q)^{-1} && (4.10)\\
N_c^T &= I - J_c^T \bar{J}_c^T && (4.11)\\
\mu_c(q, \dot{q}) &= \bar{J}_c(q)^T b(q, \dot{q}) - \Lambda_c(q)\dot{J}(q)\dot{q} && (4.12)\\
p_c(q) &= \bar{J}_c(q)^T g(q). && (4.13)
\end{aligned}$$

The control force, F_c, in Equation (4.8) can be designed by compensating the dynamic effects with the estimates of the matrices, $\hat{\Lambda}_c(q)$, $\hat{\mu}_c(q, \dot{q})$, $\hat{p}_c(q)$, and $\hat{f}_c$,

$$F_c = \hat{\Lambda}_c(q)f_c^* + \hat{\mu}_c(q, \dot{q}) + \hat{p}_c(q) + \hat{f}_c. \tag{4.14}$$

The resulting equations of motion form the decoupled unit mass system for each contact,

$$\begin{aligned}
\dot{\vartheta}_c &= f_c^*. && (4.15)\\
\text{i.e.} \quad \dot{\vartheta}_{c,i} &= f_{c,i}^*, && (4.16)
\end{aligned}$$

where i denotes each contact.

Controller design

Designing the contact force controller requires knowledge of the contact environment. A precise model of the environment is difficult to construct; thus, a robust controller is designed to deal with modeling errors while a simple environment model is used as in Chapter 2. The contact environment model used in the formulation is a spring model [33], in which the environment is assumed to have a constant stiffness. For each contact i,

$$\dot{f}_{c,i} = k_{s,i}\vartheta_{c,i}, \tag{4.17}$$

where $f_{c,i}$ is the ith contact force. The term, $\vartheta_{c,i}$, is the instantaneous velocity in the contact normal direction. The term, $k_{s,i}$, is the ith contact environment stiffness.

With this model and Equation (4.16) the equations of motion for each contact i are

$$\ddot{f}_{c,i} = k_{s,i}f^*_{c,i}.$$

Based on this second order equation of motion for each contact, a controller is implemented with AOB design. The details are provided in Chapter 2.

4.2.2 Motion control in the null space

The null space control torque, Γ_0, is used for motion control. The dynamically consistent null space projection matrix, N_c^T, projects the torque, Γ_0, into the null space of the contact forces; thus, the contact forces are not affected by Γ_0.

Having the task-posture decomposition control structure for the contact force and motion control, the task consistent dynamic equation for motion control can be obtained [34]. The dynamic equation is

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) + J_c(q)^T f_c = J_c^T F_c + N_c^T \Gamma_0. \tag{4.18}$$

We define the operational space coordinate for motion as x_m and the corresponding Jacobian as J_m. The control torque for motion, Γ_0, that applies control force, $F_{m|c}$, into the null space, is chosen as $J_m^T F_{m|c}$. The dynamic equation in the motion space is obtained by pre-multiplying by $\Lambda_{m|c} J_m A^{-1}$.

$$\Lambda_{m|c}\ddot{x}_m + \mu_{m|c} + p_{m|c} + \bar{J}_{m|c}^T J_c^T f_c = \bar{J}_{m|c}^T J_c^T F_c + F_{m|c}, \tag{4.19}$$

where

$$\begin{aligned}
\Lambda_{m|c}(q)^{-1} &= J_m(q)A(q)^{-1}N_c^T J_m(q)^T &(4.20)\\
\bar{J}_{m|c}^T(q) &= \Lambda_{m|c}(q)J_m(q)A(q)^{-1} &(4.21)\\
\mu_{m|c}(q,\dot{q}) &= \bar{J}_{m|c}(q)^T b(q,\dot{q}) - \Lambda_{m|c}(q)\dot{J}_m(q)\dot{q} &(4.22)\\
p_{m|c}(q) &= \bar{J}_{m|c}(q)^T g(q). &(4.23)
\end{aligned}$$

Note that this dynamic equation of motion is consistent or constrained to the task dynamics. That is, the control force, $F_{m|c}$, is applied to the null-space of the contact force control. Therefore, the torque component that will affect the contact force control will be eliminated by the null-space projection matrix, N_c^T.

Based on Equation (4.19) and the composed control force, F_c, for the contact force control, the control force in motion control can be computed as

$$F_{m|c} = \hat{\Lambda}_{m|c} f_m^* + \hat{\mu}_{m|c} + \hat{p}_{m|c} + \bar{J}_{m|c}^T J_c^T \hat{f}_c - \bar{J}_{m|c}^T J_c^T F_c, \tag{4.24}$$

resulting a unit mass system for motion outside of singular configurations.

$$\ddot{x}_m = f_m^*. \tag{4.25}$$

The total torque to be applied to the robot is

$$\Gamma = J_c^T F_c + N_c^T J_m^T F_{m|c}. \tag{4.26}$$

4.3 Noise characteristics in motion

The noise variance of the force measurement influences the estimation strategy. If the noise variance increases, the force estimation should rely more on the force model rather than measured data. Experimental tests have shown that the measurement noise varies while the robot is moving in contact mainly due to friction. The amount of variation is related to the magnitude of contact force since the normal contact force increases the tangential friction force. Therefore, on-line noise identification has been performed to adapt the state estimation.

4.3.1 Noise variance computation

Maybeck [43] uses the maximum likelihood equations to estimate the measurement covariance matrix. The method optimizes the estimation of the measurement variance using the model and measurements. However, this approach is too sensitive to disturbances, such as the case when the robot starts or stops a motion or forces are applied to other contact points. Practically, the disturbance or the compensator response due to the disturbance create low-frequency output of contact force because of the nature of the controller. Particularly, when the noise variance must be computed on-line this component of the controlled signal should be filtered out before computation. Therefore, a high-pass filter is used for the measured force data before noise analysis.

$$\alpha_f(z) = G_f(z)\alpha(z), \tag{4.27}$$

where $G_f(z)$ is the discrete time first order high-pass filter which has a zero and a pole with a DC gain of 1.0. The term $\alpha(z)$ is the measured contact force for each contact force space. The zero and the pole have been chosen as 3 Hz and 60 Hz in our experiments, respectively, according to our closed loop design specifications. The estimation of the measurement noise, $\hat{R}(t_i)$, is given by

$$\hat{R}(t_i) = \frac{1}{N} \sum_{j=i-N+1}^{i} \{[\alpha_f(t_j) - \bar{\alpha}_f][\alpha_f(t_j) - \bar{\alpha}_f]^T\}, \tag{4.28}$$

where $\bar{\alpha}_f$ is the mean of the filtered force over a time window, i.e.

$$\bar{\alpha}_f = \frac{1}{N} \sum_{j=i-N+1}^{i} \alpha_f(t_j). \tag{4.29}$$

Fifty samples have been used in the experiments.

This practical approach may not compute the exact noise variance. However, this adaptation captures the changes of the variance of the noise quickly and is not sensitive to the large disturbances to the measurement.

4.4 Experiments

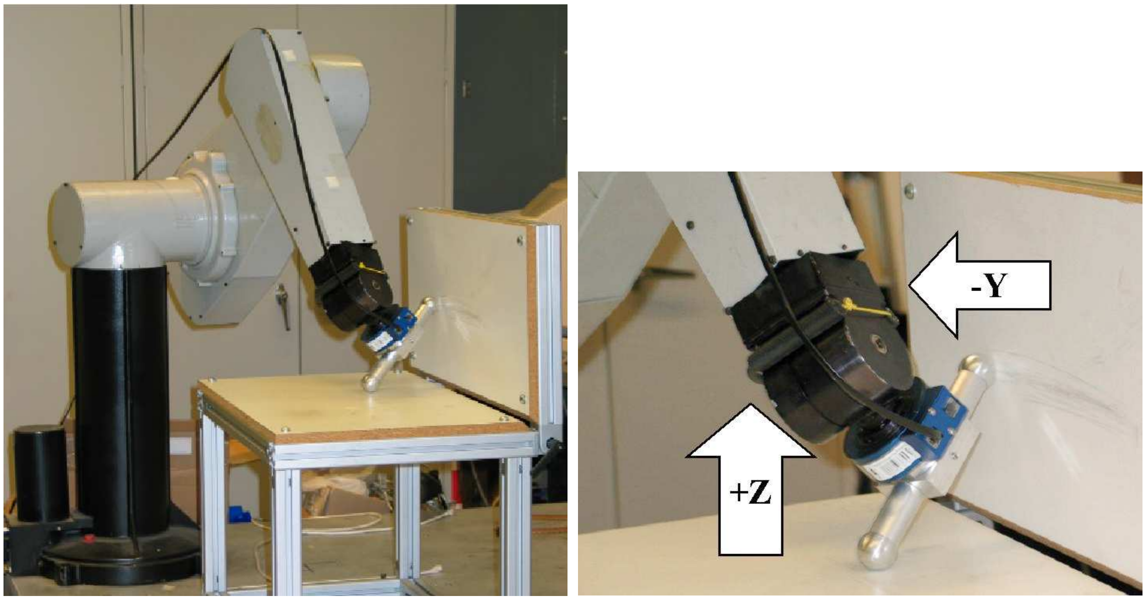
The multi-link multi-contact formulation was verified by experiments using a PUMA560 robot. The first set of experiments was designed to control the end-effector in multiple contacts with the environment. The second set of experiments was designed to control the robot in multiple contacts over multiple links.

The PUMA560 manipulator was controlled through a TRC205 amplifier package from Mark V Automation Corporation. This amplifier package was connected to a PC, running a QNX real-time operating system. The servo program in QNX commanded torques to the robot through this setup. The servo rate was set to 500 Hz. The PUMA560 robot has 6 DOF, including 3 revolute joints at the wrist. A JR3 force sensor was mounted at the wrist to measure the contact forces at the end-effector. The force sensor measures forces and moments in all three dimensions.

4.4.1 Multiple contacts at the end-effector

System setup

The system setup in Figure 4.2 consists of a PUMA robot, a table and a vertical board. The vertical board makes a 90 deg angle with the table. As can be seen in Figure 4.2, the end-effector has two rigid bars. Each tip of the rigid bars makes contact with a table and a vertical board, respectively. Step functions are commanded



(a) (b)

Figure 4.2: **System setup for multi-contact at the end-effector.**

for two contact forces. The motion in the other direction tracks a sinusoidal input while the orientation of the end-effector is controlled to maintain fixed orientation.

Experimental results

The motion control is designed to be critically damped with a natural frequency $\omega_n = 30 rad/sec$. The full state feedback gain L_r for the force control is set to have a time constant of 0.02 *sec*. The input time delay, T_d, is 3 sampling periods. Due to this delay, the system state, $x_{a,k}$, has 6 states that are the contact force, the derivative of the contact force, 3 delayed inputs, and the active state, p_k. The system noise matrix is $Q_k = 10^{-3} I_{6\times6}$ and $P_0 = 10^{-3} I_{6\times6}$. The measurement noise R_k is calculated on-line using the previous 50 samples.

Figures 4.5 and 4.6 show the contact forces and the estimated forces over time. The translational motion in the x direction is plotted in Figure 4.7. In the range of 160-190 *sec*, the step functions between $10N$ and $15N$ have been commanded while the orientation and translational motion in the x direction were commanded to maintain the starting position. The square functions were simultaneously commanded for both contacts with the table and the vertical board. The operational point of the end-effector (the wrist point) was commanded to follow a sinusoidal trajectory starting at 190 *sec* in the x direction. The same square functions were commanded to the contact force controllers while the robot was in motion.

The contact force follows the commanded force with the designed time constant. Figure 4.8 shows the force variance for each contact. The variance is about 0.6 in a static situation, increasing to about 100.0 when the manipulator moves. Although the noise characteristic varies a great deal the on-line noise estimation is able to properly adapt the control such that the contact forces are not significantly affected by the motion. The starting motion of the end-effector disturbs both contact forces significantly at about 195, 205, and 217 *sec*.

The pre-estimated value of $k_{s,i} = 6,000N/m$ was used for the stiffnesses of both the table and the vertical board. These values were evaluated off-line by measuring the displacement and the contact forces at the very low contact force range (0-5 N). As the contact force increases, the environment stiffness also increases. The stiffness of the table is approximately more than $50,000$ N/m based on the computation using its material property and geometry.

It should be mentioned that on-line noise variance estimation plays a critical role in the control performance. Without on-line estimation, the control only works if the robot does not move. The motion generates too much noise in the force data, leading the system to instability. The high-pass filter in the variance computation prevents the system responses being slow due to disturbances. These disturbances occur at the beginning of the robot motion and at the step commands.

4.4.2 Multiple contacts over multiple links

System setup

The experimental setup for multi-link multi-contact is shown in Figures 4.3 and 4.4. The first contact was established at the third link and the control point was chosen to be the contact point. The desired contact force direction was normal to the link, i.e. X_3 direction in Figure 4.4 (a). A JR3 force sensor was mounted on the contact

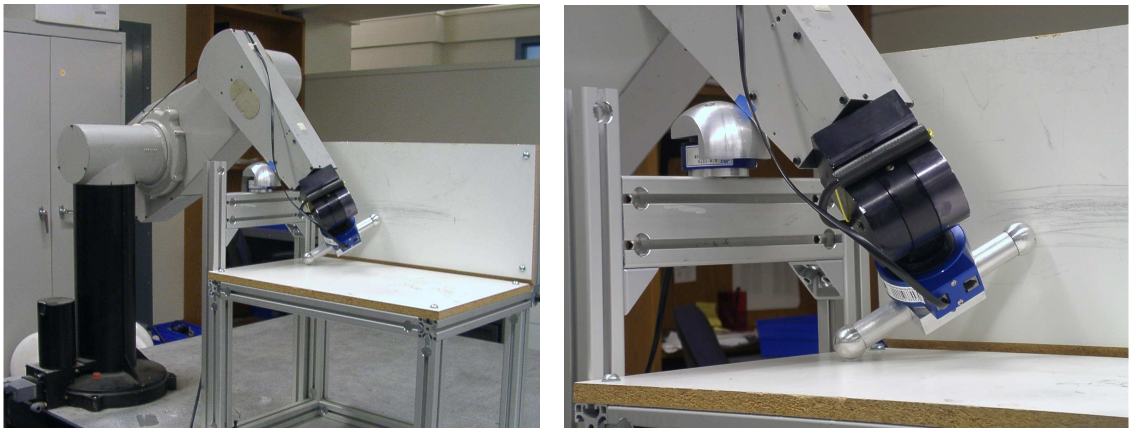
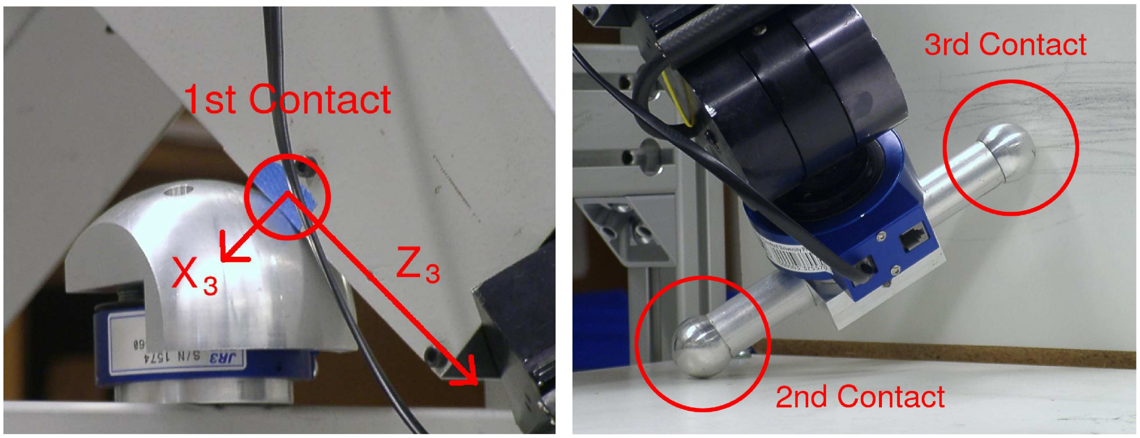


Figure 4.3: **PUMA560 in three contacts**

environment since it was difficult to be mounted on the link of the robot. The contact force was computed by projecting the measured contact force to the normal direction of the contact link. The second and third contacts were at the end-effector; one contact with the horizontal table and the other with the vertical rigid board in Figure 4.4 (b). The control point of the end-effector was chosen to be the wrist point and another JR3 force sensor was mounted on the wrist.

The contact environments were an aluminium frame, a wooden table and a wooden vertical board. Consequently, they were near rigid contacts. However, the linkage between the table and the vertical board had some flexibility. The system stiffnesses of the three contacts were pre-estimated at low contact forces as $k_{s,1} = 6,000N/m$ for the third link contact, $k_{s,2} = 6,000N/m$ for the end-effector contact with the table, and $k_{s,3} = 3,000N/m$ for the end-effector contact with the vertical board. The actual stiffnesses of all three contacts were infinite at high contact forces. These estimates of the stiffness were used in the Kalman filter.

The motion control for the first control point was designed to hold the position along the third link, i.e. Z_3 in Figure 4.4 (a). The orientation of the end-effector was controlled through null space control so that 2 remaining DOF could be controlled after controlling 3 contact forces and one motion direction. Two sets of experiments were conducted with and without motion command in the null space.



(a)

(b)

Figure 4.4: **System setup for multi-link multi-contact.** (a) The contact of the third link (b) Two contacts at the end-effector.

Static contact experiment (Figures 4.9-4.11)

While three contact forces were controlled, the motion control was commanded to hold the current configuration. Since all three contacts were very stiff, the motion of the robot was very small during the experiment. When one of the desired contact forces was commanded with step functions, the other desired contact forces were controlled to be maintained at the same value.

Although the effect of one contact force control on the others was not completely eliminated, contact force control was successfully accomplished. The settling time of the step response was longer than the designed value (0.23 seconds) mainly due to the interaction with the other contact force controls. At the time of a step command, the contact force control corresponding to the step command created disturbance to the other contact forces, whose controller, then, compensated the disturbance. In the process all three contact forces affected each other since the model of the manipulator and the contact environment was not perfect.

Moving contact experiment (Figures 4.12-4.14)

Three contact forces were controlled to follow step commands from 5 to 15 N and the displacement along the third link was commanded to maintain the position. Concurrently, the desired orientation of the end-effector was designed to rotate around the 4th joint of the robot, i.e. the first joint of the wrist. Therefore, the second and third contact points moved along the table and vertical board correspondingly. The first contact at the third link also moved in the direction that was perpendicular to the direction along the link and the contact direction. That is, the contact point moved in the Y_3 direction in Figure 4.4 (a).

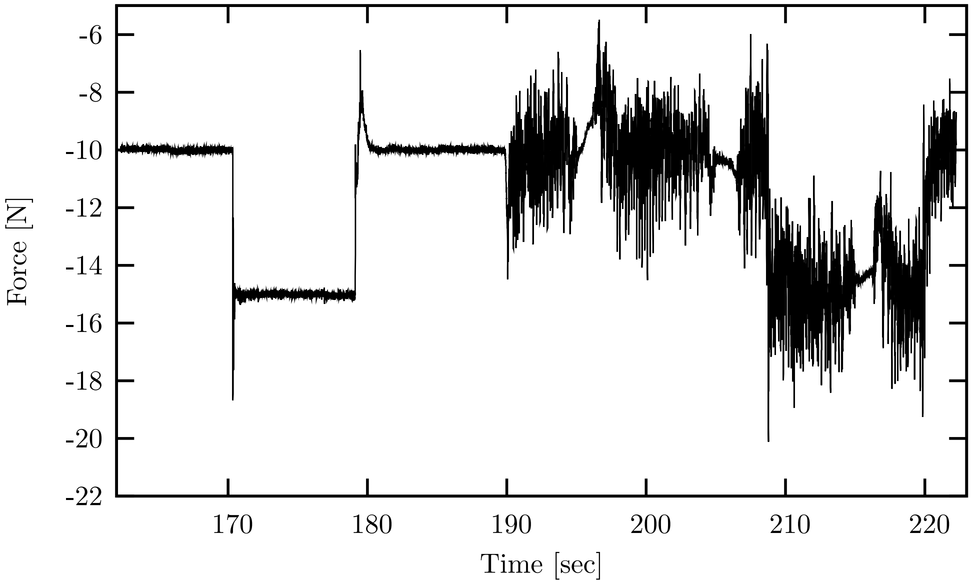
All three contact forces were affected by the wrist motion mainly due to the contact surface characteristics and an imperfect kinematic model of the contact. When the contact started moving, greater disturbance was produced to contact forces due to the static friction of the contacts.

Although the measurement characteristic changed from static contact to moving contact, the robust force controller with on-line variance estimation maintained the contact forces while rejecting the disturbances.

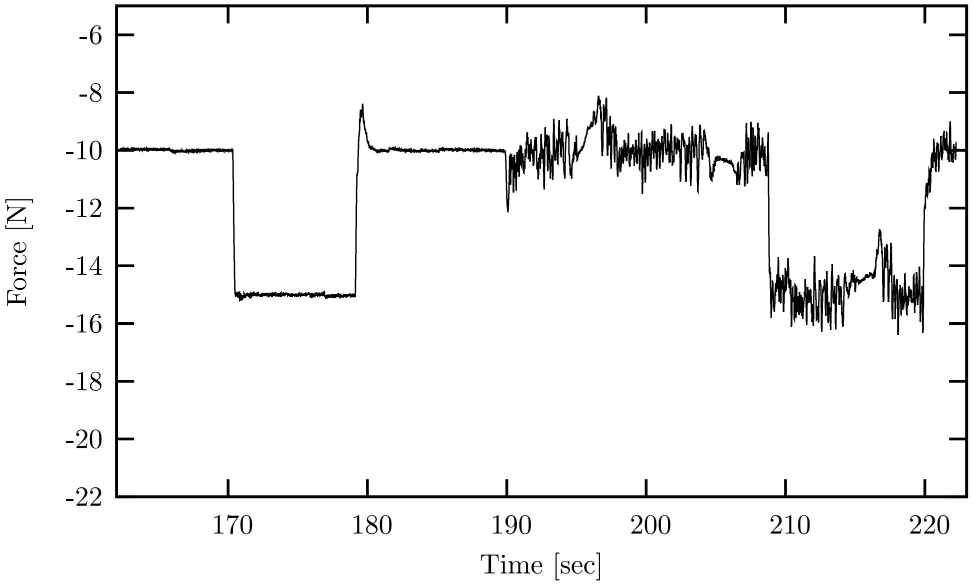
4.5 Conclusion

A framework for multi-link multi-contact compliant motion control is presented in this chapter. The complex contact compliant task is implemented in the operational space formulation utilizing a multi-contact model and contact environment model. The operational space coordinates are defined to be the composition of the displacements in the contact normal spaces. To effectively deal with modeling errors, full state feedback with an AOB is applied. Motion is controlled in the null space of the contact force space.

The experimental results with a PUMA560 manipulator demonstrate the successful implementation of this framework. The first experiment was conducted when the end-effector was in contact with the environment at two points. The second experiment was conducted when three points were in contact; one with the third link and two with the end-effector of the PUMA560 manipulator. Static and moving contact experiments show the high performance of the multi-link multi-contact force control framework, even in the presence of varying contact characteristics and disturbance from the motion of the manipulator.

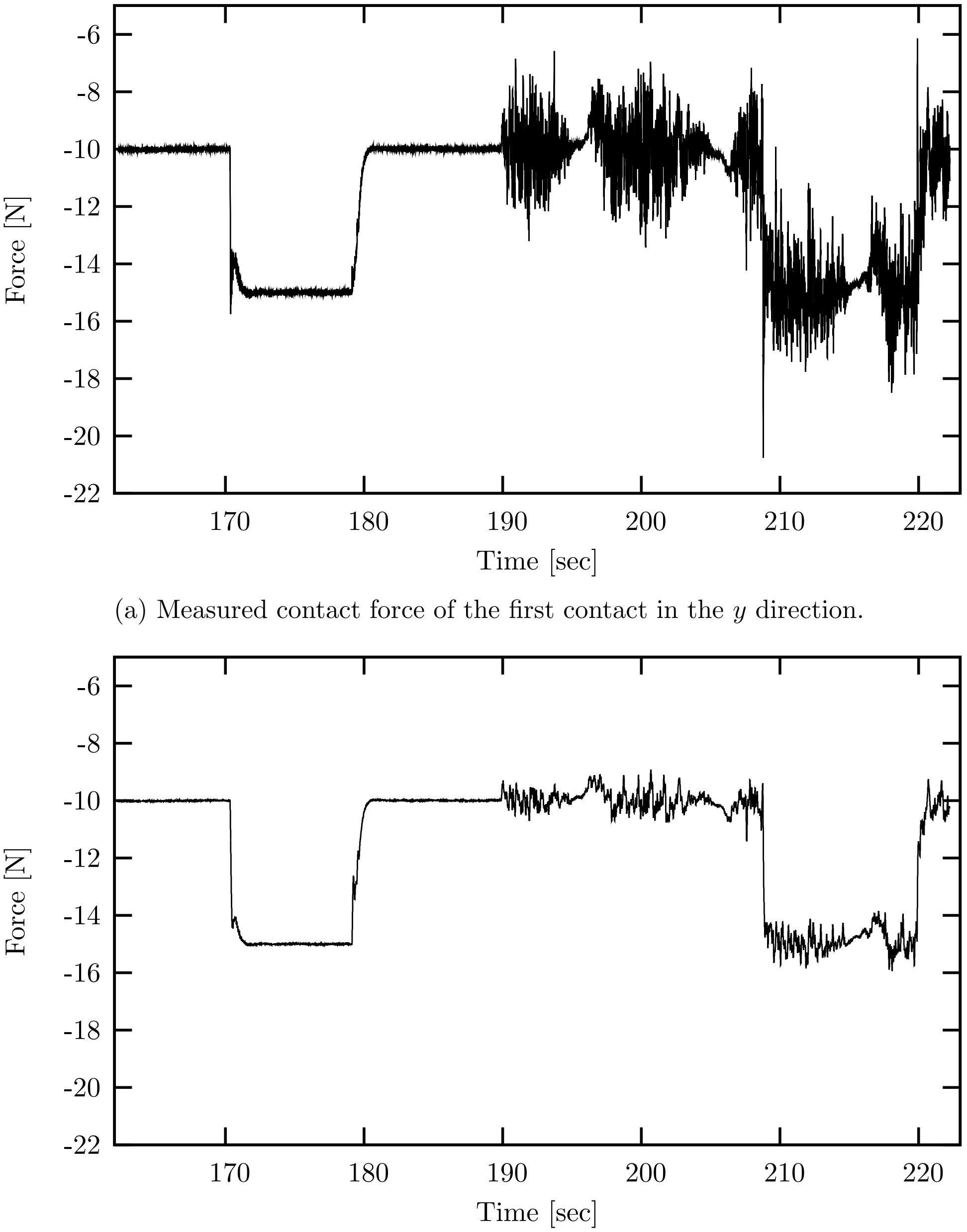


(a) Measured contact force of the first contact in the z direction.



(b) Estimated contact force of the first contact in the z direction.

Figure 4.5: **Multiple contacts at the end-effector.** Measured and estimated forces in contact with the table. The end-effector started moving at 190 *sec* in the x direction.



(a) Measured contact force of the first contact in the y direction.

(b) Estimated contact force of the first contact in the y direction.

Figure 4.6: **Multiple contacts at the end-effector.** Measured and estimated forces in contact with the vertical board. The end-effector started moving at 190 *sec* in the x direction.

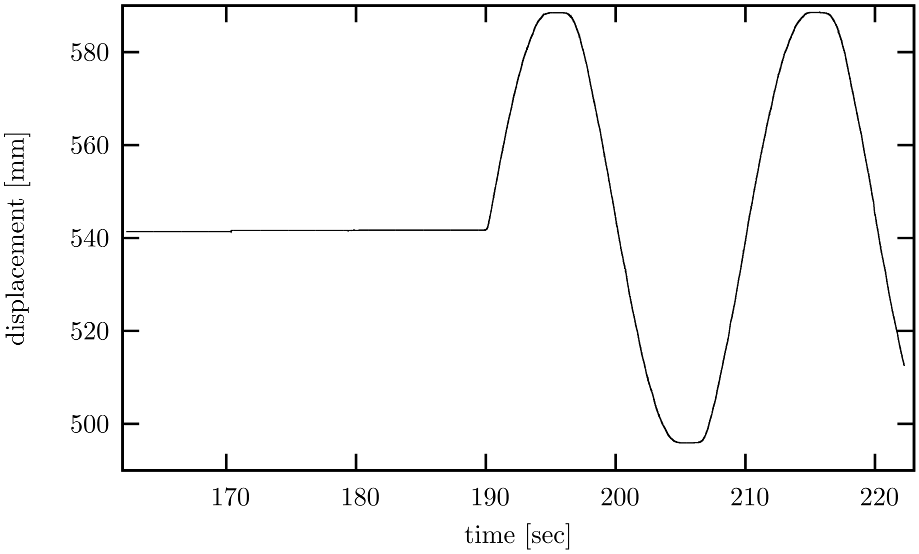
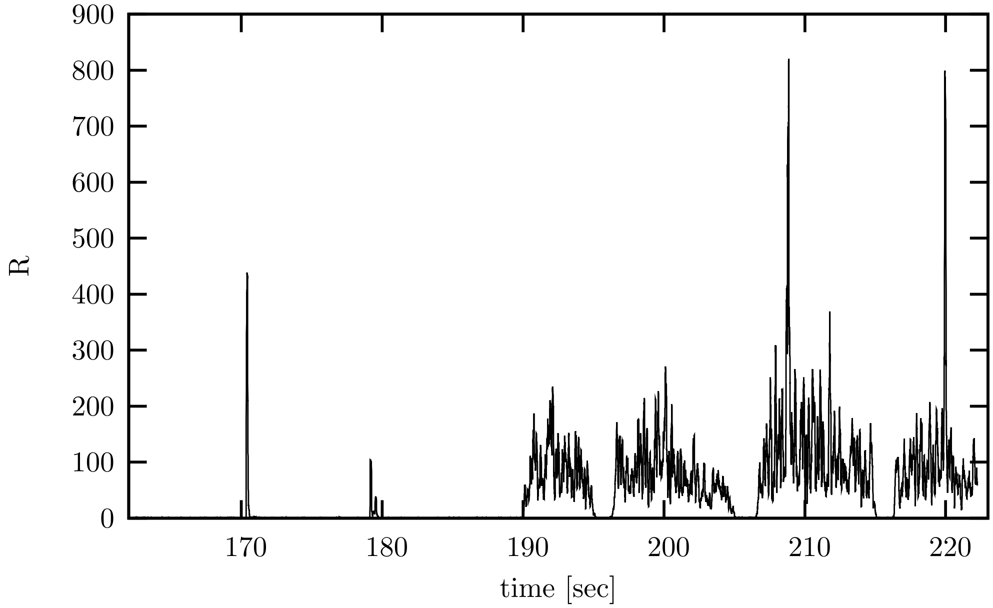
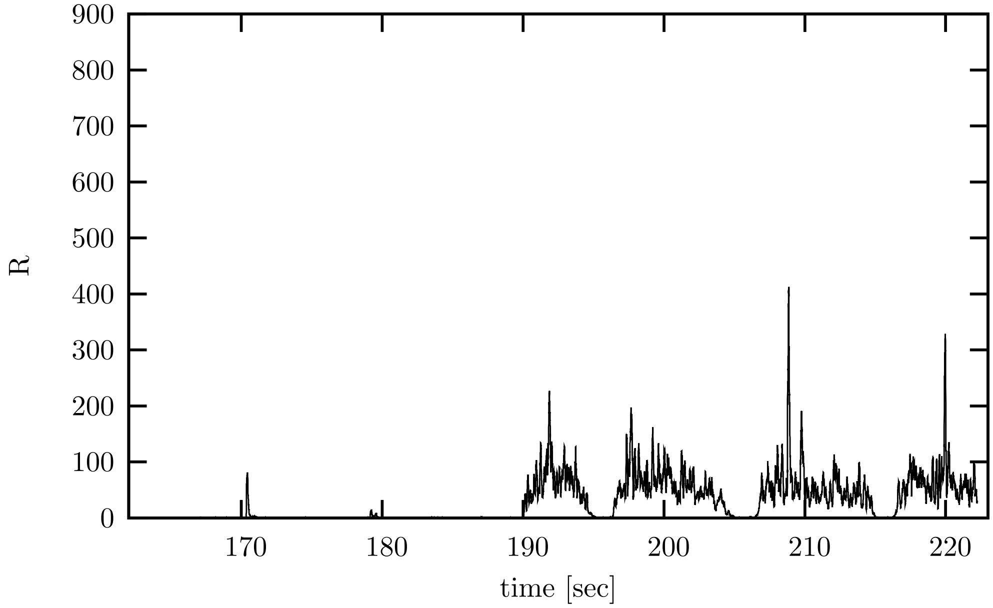


Figure 4.7: **Multiple contacts at the end-effector.** End-effector translational motion in the x direction.

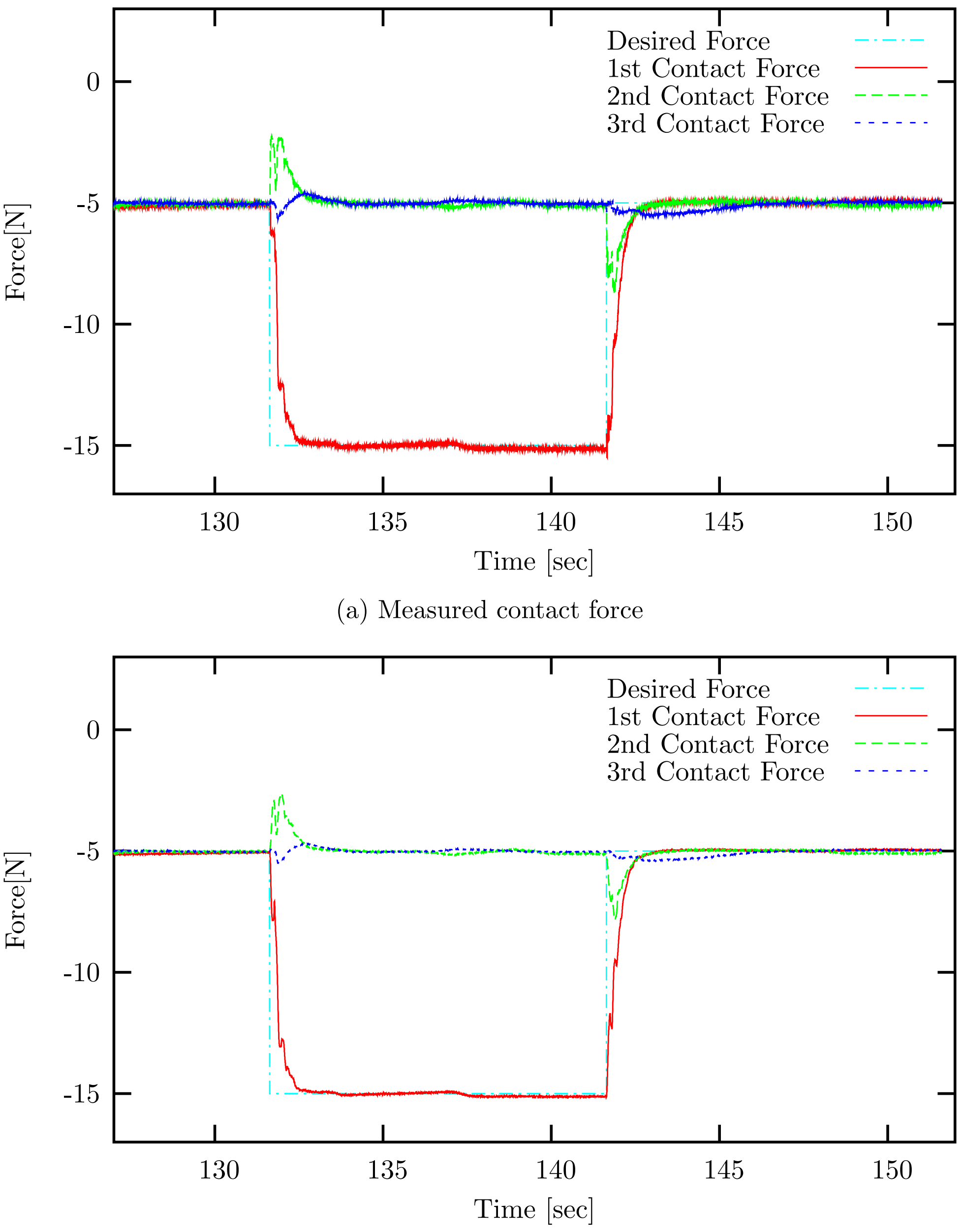


(a) Noise Covariance Estimation for the first contact force.



(a) Noise Covariance Estimation for the second contact force.

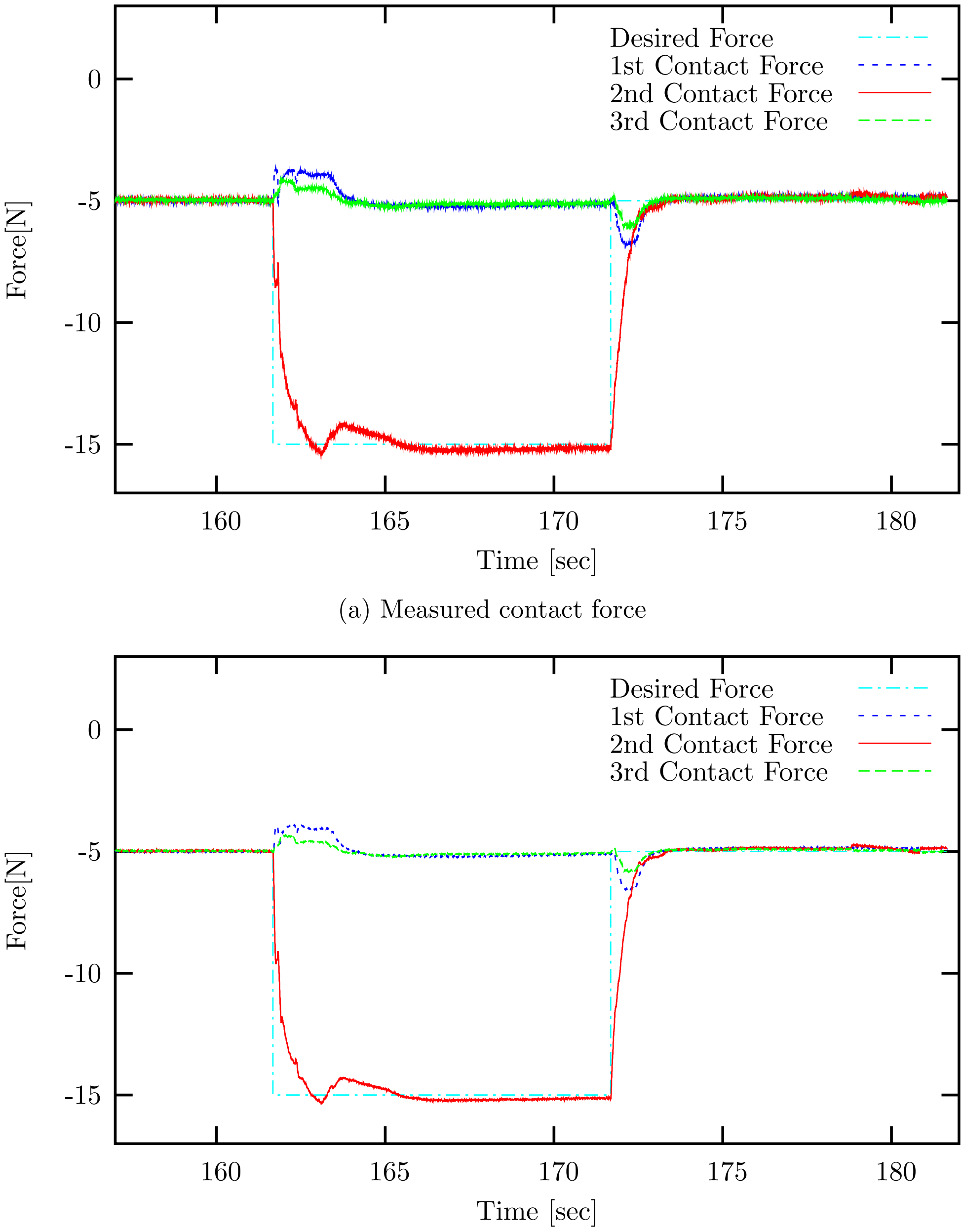
Figure 4.8: **Multiple contacts at the end-effector.** Noise variance estimations. The end-effector started moving at 190 *sec* in the *x* direction.



(a) Measured contact force

(b) Estimated contact force from Kalman filter

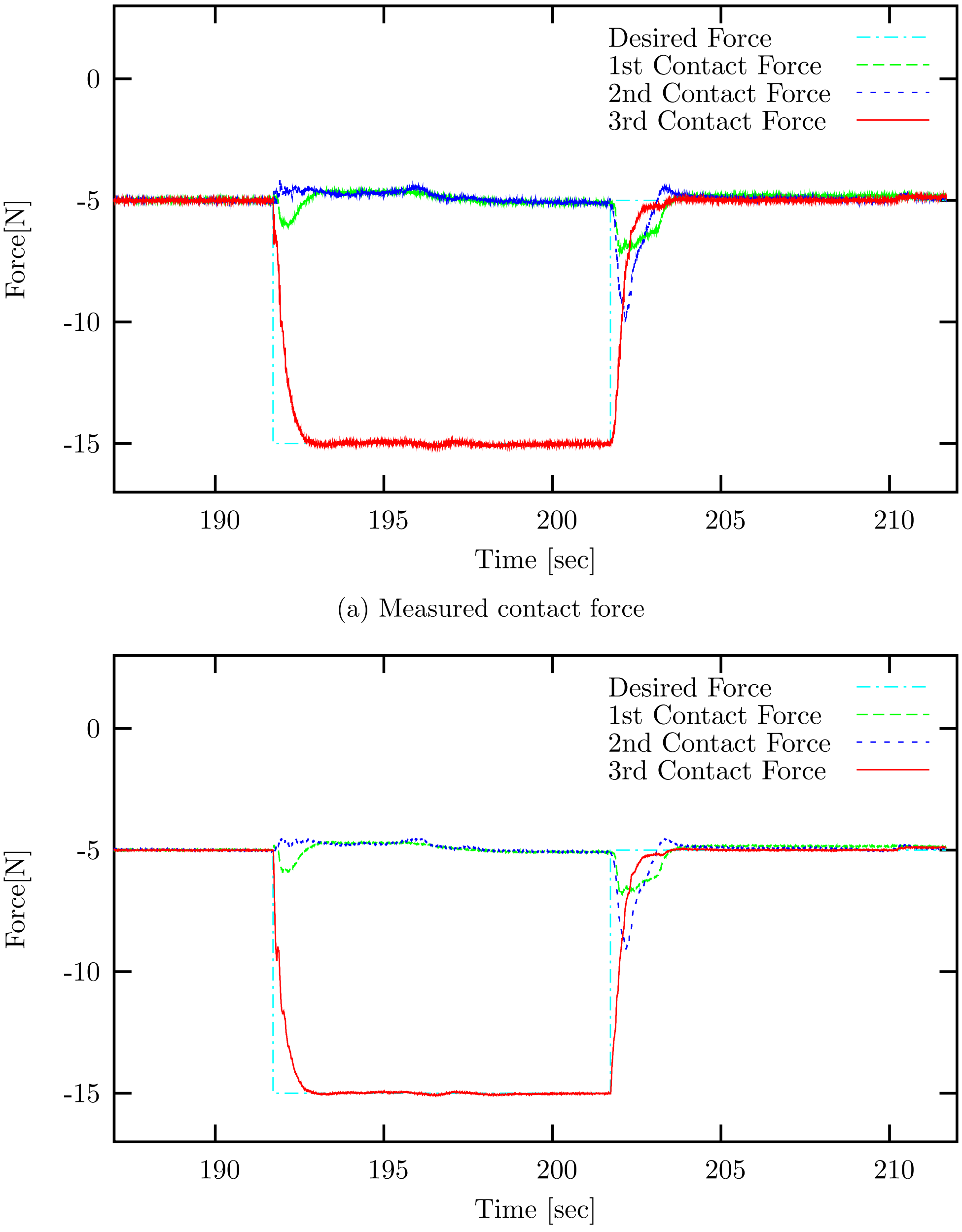
Figure 4.9: **Multiple contacts over multiple links - static contact experiment.** Step response of the third link contact (1st contact).



(a) Measured contact force

(b) Estimated contact force from Kalman filter

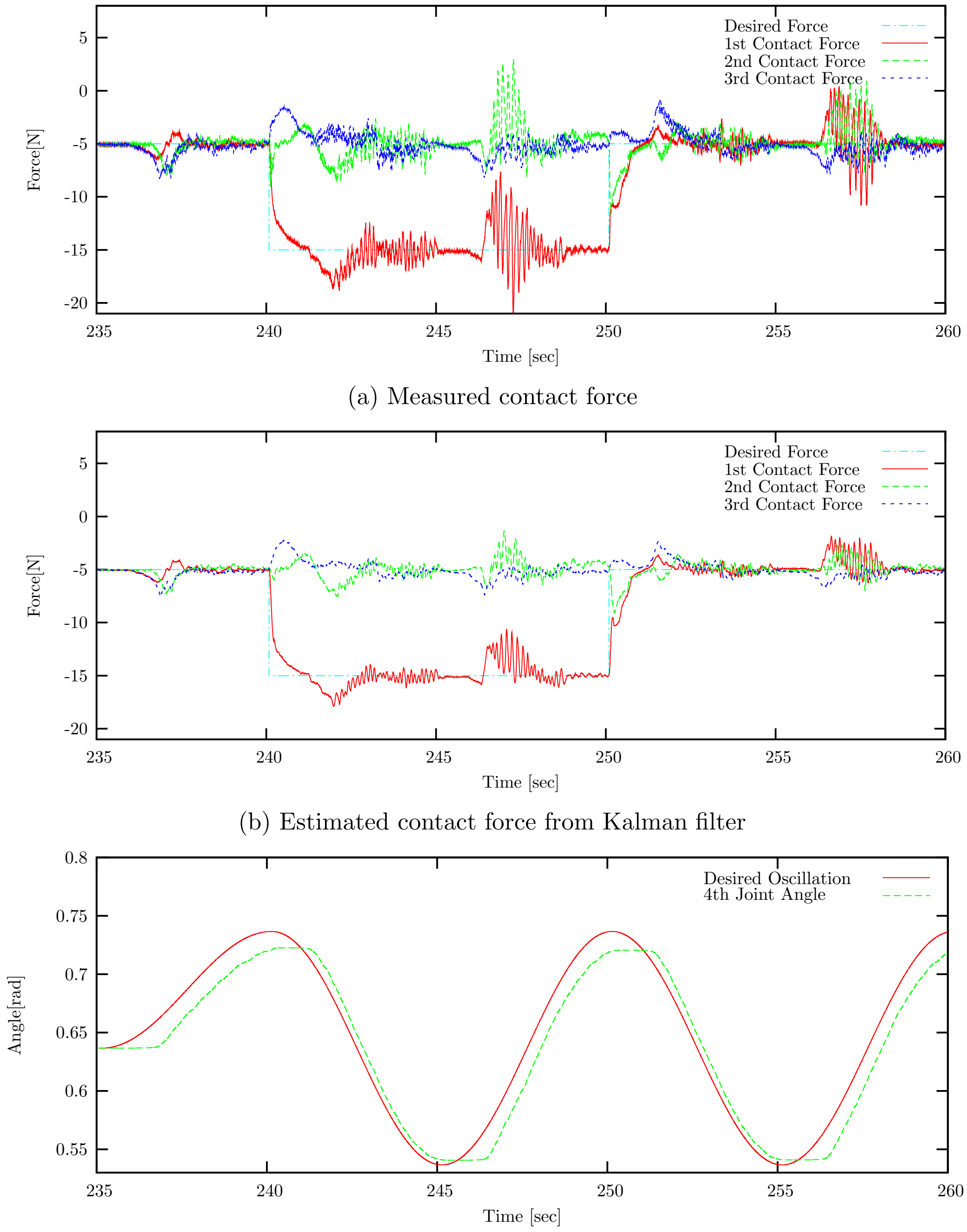
Figure 4.10: **Multiple contacts over multiple links - static contact experiment.** Step response of the end-effector contact with the table (2nd contact).



(a) Measured contact force

(b) Estimated contact force from Kalman filter

Figure 4.11: **Multiple contacts over multiple links - static contact experiment.** Step response of the end-effector contact with the vertical board (3rd contact).

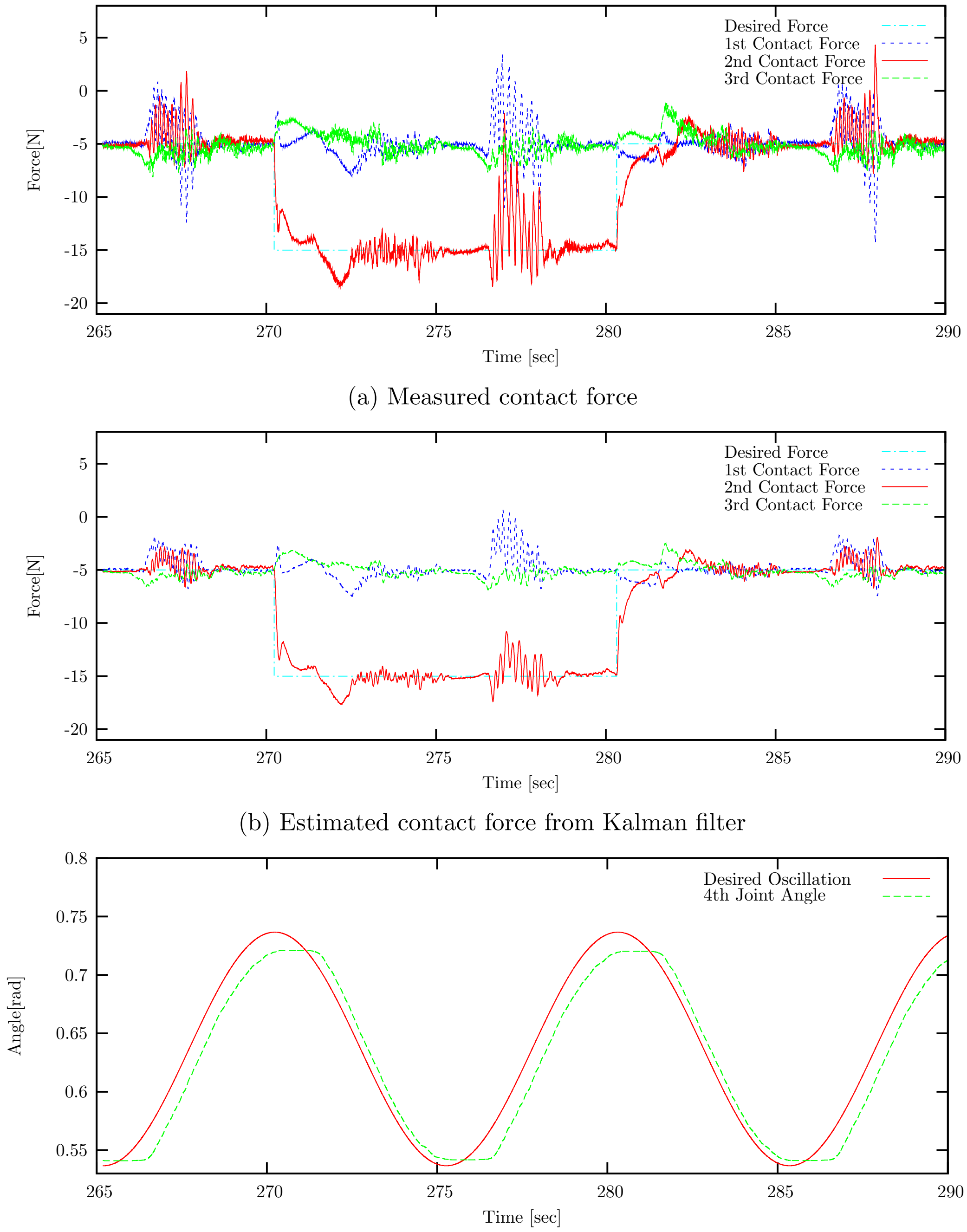


(a) Measured contact force

(b) Estimated contact force from Kalman filter

(c) 4th joint motion

Figure 4.12: **Multiple contacts over multiple links - moving contact experiment.** Step response of the third link contact (1st contact).

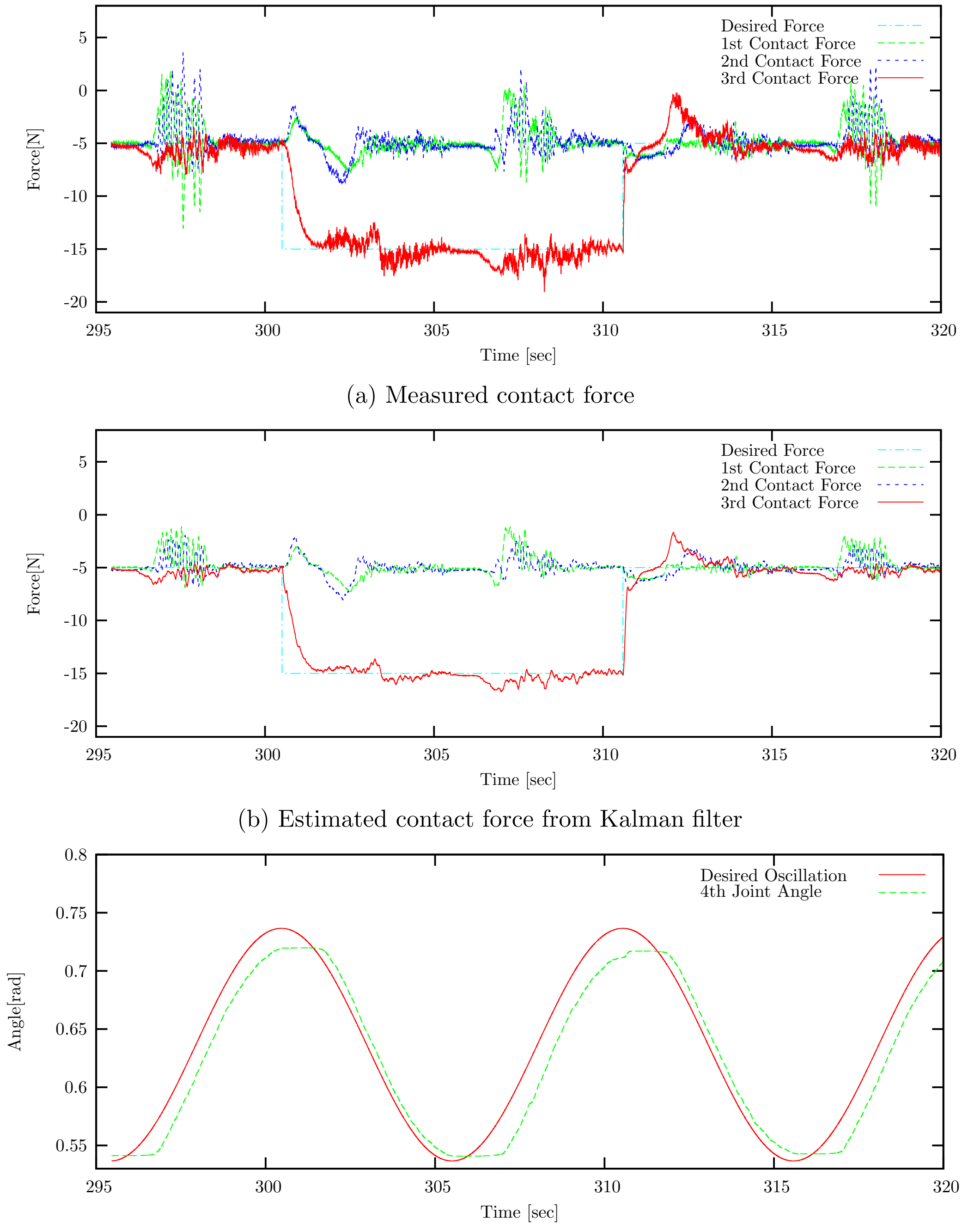


(a) Measured contact force

(b) Estimated contact force from Kalman filter

(c) 4th joint motion

Figure 4.13: **Multiple contacts over multiple links - moving contact experiment.** Step response of the end-effector contact with the table (2nd contact).



(a) Measured contact force

(b) Estimated contact force from Kalman filter

(c) 4th joint motion

Figure 4.14: **Multiple contacts over multiple links - moving contact experiment.** Step response of the end-effector contact with the vertical board (3rd contact).

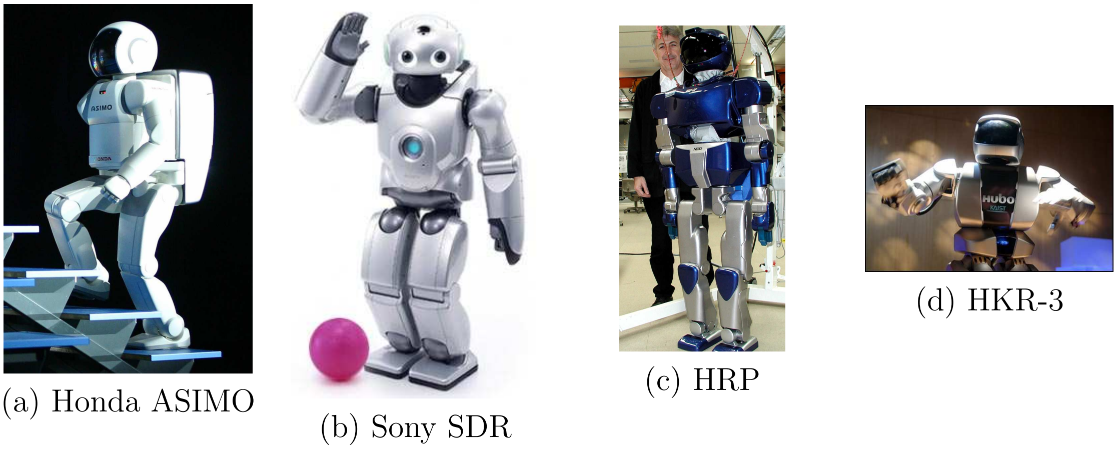
Chapter 5

Contact consistent control framework

Humanoid robotics is one of the most exciting new research areas in robotics. A humanoid robot is generally defined as a robotic system with human-like size and kinematic structure. That is, it has two legs, two arms and a head all connected by a torso.

As such, a humanoid robot could potentially accomplish tasks in the same manner as humans. Two legs offer biped locomotion, which is one of the most efficient means of locomotion, allowing the robot to move over different terrain (possibly with the help of two hands). Bi-manual manipulation provides dexterity in performing numerous everyday tasks. Therefore, given a choice of different types of robots for working in human environments, humanoid robots are the most attractive and promising.

Substantial research has been conducted in this field over the past two decades. This has led to the development of enhanced humanoid robots, such as Honda's ASIMO, Sony's SDR, HRP, and KHR-3 [23, 13, 35] (Figure 5.1). Walking is one of primary functions of any humanoid robot. In fact, research on biped walking predates humanoid robots (WABOT-1, Waseda University). Since the first successful walking humanoid system was presented by Honda in 1995, research in this area has garnered significant attention, and now it is common to see these humanoid systems walking. This was a very important development in the evolution of humanoid robots.



(a) Honda ASIMO

(b) Sony SDR

(c) HRP

(d) HKR-3

Figure 5.1: Humanoid robots

Humanoid systems have fundamentally different characteristics from conventional manipulators. Most significantly, they do not have fixed bases. This fact introduces intrinsic under-actuation of the system when there is no contact with the environment. Furthermore, humanoid systems have many degrees of freedom, i.e. many joints, whereas conventional manipulators are typically designed with 6 DOF or less. This fact derives from the desire to mimic human motion. Another distinguishing characteristic is that humanoid systems are in contact with the environment (usually with the ground) during most of their operation.

These characteristics have motivated different approaches for controlling humanoid robots compared with fixed base robots. The characteristics of high DOF systems often introduce approximations in dynamic and kinematic modeling. And many specific algorithms are proposed due to the non-fixed base characteristic. In general, The control strategies employed in these systems have been limited to specialized behaviors for desired tasks, especially walking [29, 50, 37, 44].

Walking control algorithms, in general, can be categorized into two groups [29]. The first category utilizes a zero moment point (ZMP) approach which requires precise knowledge of the dynamic and kinematic properties of the robot [65, 26, 28]. The second category utilizes an inverted pendulum model approach, which requires less knowledge because it uses an approximation of the system [55, 49, 62, 30]. The basic approach in these control schemes is the same in the sense that they generate desired trajectories for all joints and then try to follow those trajectories using feedback control. Because of this, it is not a simple procedure to integrate additional motion behaviors concurrent with the generated walking motion, although these control approaches have successfully implemented walking patterns in actual humanoid robots. To remedy this, a general control methodology is sought that integrates various whole body behaviors into a single dynamically compensated control structure.

The proposed approach in this thesis involves the *contact consistent control framework*. In this approach, full kinematic and dynamic models are used instead of approximate models. Typically, the kinematic model for a humanoid robot is constructed by defining the link in contact with the ground as the base link. However, this typically involves switching models between different contact configurations. To avoid switching models for different configurations, virtual joints and links are introduced to describe the kinematics and dynamics in a consistent way. The virtual joints consist of three prismatic joints and three rotational joints which connect to a physical link of the robot. This physical link can be a foot, the hip, the chest, or any other link. The mass/inertia properties for these virtual links are set to zero. In the control approach discussed here, the control torques for these virtual joints are selected to be zero. This approach provides a consistent way of representing the kinematics and dynamics of the system in all configurations.

Having a consistent representation, the dynamics of the robot in each contact state are computed including the dynamics of the environment. Given that, within the static friction and normal contact force/moment limits, the contact points or surfaces of the robot are moving together with the corresponding points or surfaces of the environment. We can treat these contact conditions as constraints. These constraints are incorporated into the system dynamics, so given torques at all the physical joints we can compute the resulting accelerations and contact forces.

Using this method, control of a humanoid robot can be accomplished using the derived joint space dynamics. However, designing a task or behavior in a different space is often more effective. This is referred to as operational space, which is formed by the composition of the variables to be controlled. The operational space coordinates may include positions/orientations of the hands, feet, the center of mass, or any other part of the robot. Of course, when using these coordinates, the operational space dynamics are necessary to control the tasks in the same way as the joint space dynamics are necessary to control the joints. Therefore, the joint space dynamics are projected into operational space to design the corresponding controller [63, 56].

Utilizing the operational space dynamics, the desired control forces in operational space are generated to command the robot. Unlike fixed base robots, however, the standard relationship between force and torque cannot be used due to the restriction that there are zero torques at the virtual joints. To compute the actual control torques needed to produce operational space forces, the relation from the command torque to the force, and its inverse, are derived. The torque is then computed using this inverse relationship. When the DOF of the control variables is smaller than that of the actuated joint space coordinates, there will be an infinite number of solutions. Although many solutions can be used, the solution associated with the dynamically consistent inverse is selected. This solution provides the control torques necessary to generate the control forces while minimizing motion in the rest of the robot.

As such, the constraint based approach assumes that all of the contacts are fixed contacts. The controller may choose solutions which violate the friction and normal contact force/moment conditions. Therefore, before the control torques are applied the expected contact forces must always be monitored and modified to maintain the current contact state. This procedure is especially important during fast motion.

In summary, the proposed approach models contacts using constraints to compute the current dynamics of the robot. These dynamic equations are utilized to create the commanding torques for desired motion in the operational space coordinates. This constraint based approach is referred to as *the contact consistent control framework* because the approach incorporates the contact dynamics. Therefore, without explicitly computing the desired contact forces to be generated or commanded, the whole body motion is controlled in a contact consistent manner.

While this approach offers a control structure for a single contact configuration, the actual tasks of the robot may involve more than one contact condition. For example, walking involves at least 3 contact conditions (one foot support and two foot support). The control strategy for these situations must be designed with special attention. One of the most important requirements is to design tasks with small impact during contact transitions. Also, there should not be discontinuous jumps in the commanding torques.

The contact consistent control framework was verified in an simulation environment. Various complex behaviors are demonstrated including walking, jumping, ladder climbing, and manipulation while walking.

5.1 System dynamics

Given a humanoid robot with k joints, the complete system has $n = k + 6$ degrees of freedom. This includes 6 DOF of rigid body motion relative to the ground. Three revolute joints and three prismatic joints can be assigned to any link of the system as *virtual joints* to accommodate this motion relative to the ground (Figure 5.2). The dynamic equations for the robot in free space are described by

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) = \Gamma, \tag{5.1}$$

where q is the $n \times 1$ vector of joint angles and Γ is the $n \times 1$ torque vector to the corresponding joints. The terms $A(q)$, $b(q, \dot{q})$, and $g(q)$ are the $n \times n$ joint space inertia matrix, the $n \times 1$ vector of Coriolis and Centrifugal forces, and the $n \times 1$ vector of gravity, respectively.

When the robot is in contact, the contact forces should be included in the dynamic equations (Figure 5.3).

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) + J_c^T f_c = \Gamma, \tag{5.2}$$

where f_c is the vector of the contact forces and moments and J_c is the Jacobian for the contact positions and orientations.

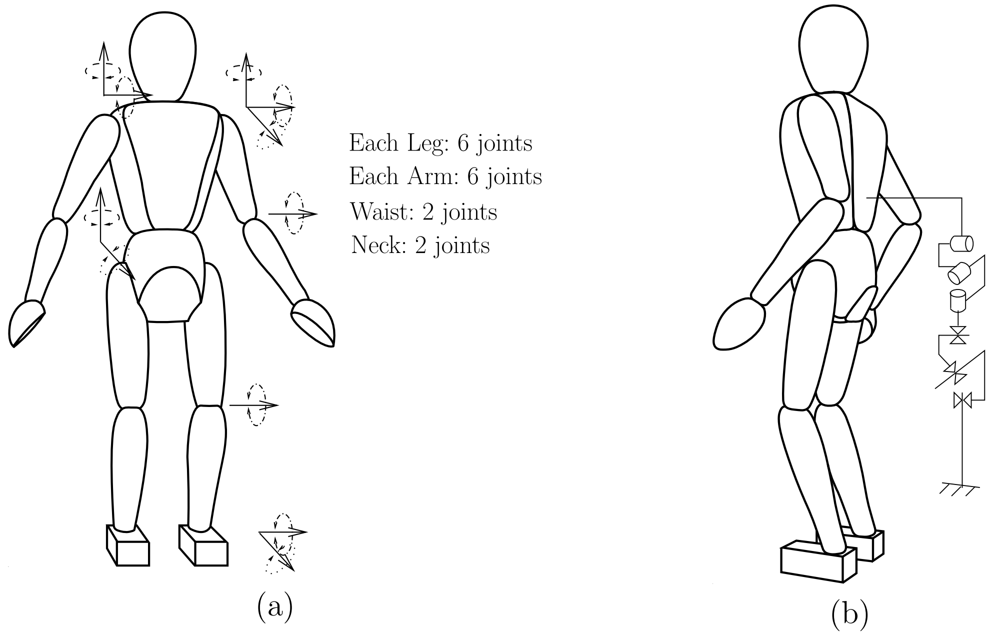


Figure 5.2: **Robot representation**. (a) joints (b) virtual links

5.1.1 Contact with the environment

Rigid body contact with the environment can be categorized as point, line, and plane contact. The contact forces/moments consist of contact normal forces/moments and tangential friction forces/moments. Contact on the robot and the environment can be considered constrained when the contact normal forces/moments between the robot and the environment are within certain limits and the tangential forces/moments due to friction do not exceed static friction values. That is, the contact point, line or plane is constrained to have the same position, velocity and acceleration as those of the contacted environment. These conditions or boundaries required to maintain the contacts will be referred as *contact conditions*.

The specific contact conditions are shown in Figure 5.4. In Figure 5.4 (a), the force from the robot to the environment in the normal direction must be negative and the magnitude of the tangential force must be less than $\mu_{static}F_z$ to maintain the contact. In addition to the conditions for the point contact case, the line contact case requires that the magnitude of the applied normal moment in the x direction, M_x, be less than $|F_z| \times \frac{l_y}{2}$ and that the magnitude of the applied moment in the z

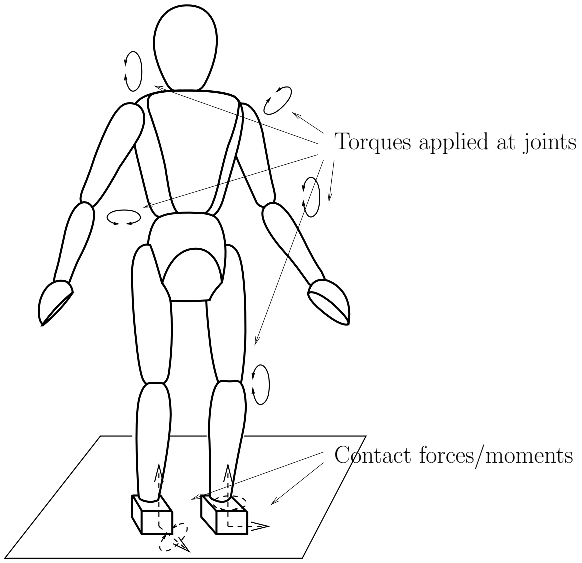


Figure 5.3: **Joint torques and contact forces**

direction, M_z, be less then $\mu'_{static}|F_z|$ (Figure 5.4 (b)). In addition to the conditions for the line contact case, the plane contact case requires that the magnitude of the applied normal moment in the y direction, M_y, be less than $|F_z| \times \frac{l_x}{2}$ (Figure 5.4 (c)).

To describe the contact dynamics, the Jacobian corresponding the contact has to be defined. For point contact, the Jacobian for contact is defined as the Jacobian of the position of the contact point; 3 degrees of freedom are constrained. For line contact, the Jacobian for contact is defined as the Jacobian of the position/orientation of the geometrical center of the contact line excluding the orientation about the contact line; 5 degrees of freedom are constrained. For plane contact, the Jacobian for contact is defined as the Jacobian of the position/orientation of the geometrical center of the contact plane; 6 degrees of freedom are constrained.

The contact Jacobian, J_c is defined as

$$\dot{x}_c = J_c \dot{q} \tag{5.3}$$

and

$$\ddot{x}_c = J_c \ddot{q} + \dot{J}_c \dot{q}. \tag{5.4}$$

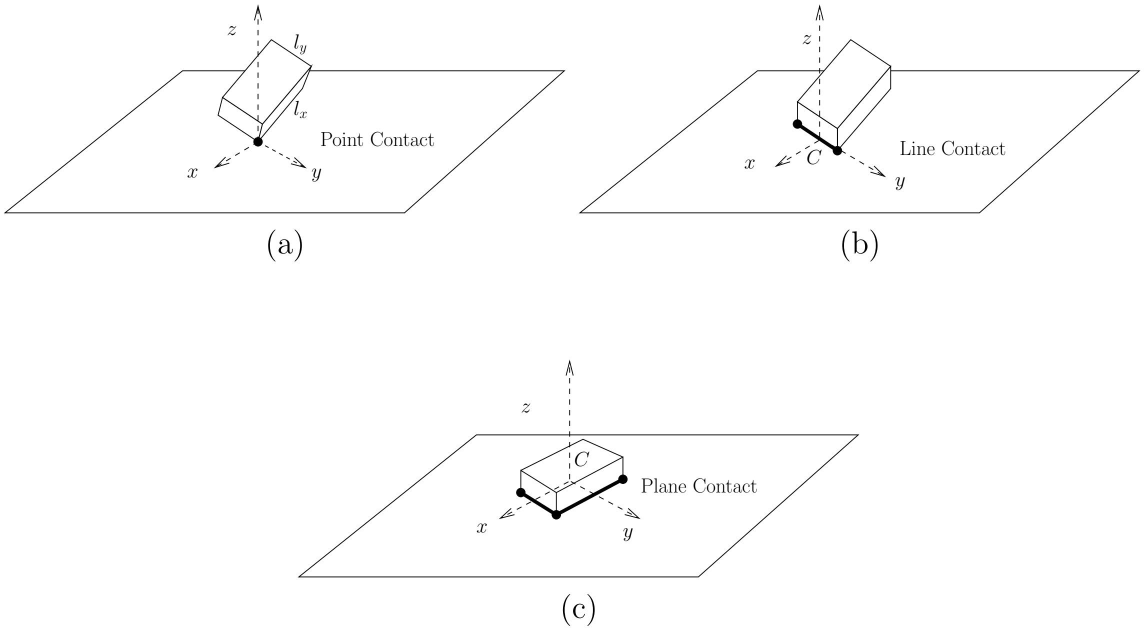


Figure 5.4: **Contact Conditions**. (a) Point contact can be maintained if $F_z < 0, \sqrt{F_x^2 + F_y^2} < \mu_s$, and $|F_x| < \mu_s F_z$, where $f_c = [F_x \quad F_y \quad F_z]^T$. (b) Line contact can be maintained if $|M_x| < |F_z|\frac{l_y}{2}$, and $|M_z| < |F_z|\mu_s'$, additional to the conditions in (a), where $f_c = [F_x \quad F_y \quad F_z \quad M_x \quad M_z]^T$. (c) Plane contact can be maintained if $|M_y| < |F_z|\frac{l_x}{2}$ in addition to the conditions in (b). The terms μ_s and μ_s' are the static friction coefficient for forces and moments, respectively, where $f_c = [F_x \quad F_y \quad F_z \quad M_x \quad M_y \quad M_z]^T$.

With the contact Jacobian, the dynamics of the system (5.2) are projected into the contact space by pre-multiplying by $J_c A^{-1}$,

$$\ddot{x}_c + \eta_c + \zeta_c + \Upsilon_c f_c = J_c A^{-1} \Gamma, \tag{5.5}$$

where

$$\Upsilon_c = J_c A^{-1} J_c^T \tag{5.6}$$

$$\eta_c = J_c A^{-1} b(q, \dot{q}) - \dot{J}_c \dot{q} \tag{5.7}$$

$$\zeta_c = J_c A^{-1} g(q). \tag{5.8}$$

The term Υ_c is the inverse of the inertia matrix in contact space. When Υ_c is invertible,

$$\Lambda_c \ddot{x}_c + \mu_c + p_c + f_c = \bar{J}_c^T \Gamma, \tag{5.9}$$

where

$$\begin{aligned} \Lambda_c &= \Upsilon_c^{-1} && (5.10) \\ \mu_c &= \Lambda_c \eta_c && (5.11) \\ p_c &= \Lambda_c \zeta_c && (5.12) \\ \bar{J}_c^T &= \Lambda_c J_c A^{-1}. && (5.13) \end{aligned}$$

The matrix Λ_c is the inertia matrix in the contact space and $\bar{J}_c$ is the dynamically consistent inverse of J_c. The term μ_c is the projection of the Coriolis/centrifugal forces at the contact, and p_c is the projection of the gravity forces at the contact.

5.1.2 Constrained dynamics of the system

Equation (5.2) describes the robot dynamics with the external forces, f_c. However, to utilize the contact forces, they need to be expressed in terms of the commanding torques and the dynamic parameters. This is possible in special cases when the dynamics of the contact environment are known. One of these cases is when the robot is in contact with the *ground* (e.g., standing, walking, or running).

While the contact conditions are met and the environment is stationary and rigid (e.g., the ground), we have $\ddot{x}_c = 0$. Therefore,

$$f_c = \bar{J}_c^T \Gamma - \mu_c - p_c. \tag{5.14}$$

Now, the equations of motion of the whole system can be rewritten by substituting f_c in Equation (5.14) into Equation (5.2).

$$A(q)\ddot{q} + b(q, \dot{q}) + g(q) - h_c(q, \dot{q}) = (I - P_c)\Gamma, \tag{5.15}$$

where

$$\begin{aligned} P_c &= J_c^T \bar{J}_c^T && (5.16) \\ h_c &= J_c^T(\mu_c + p_c). && (5.17) \end{aligned}$$

Note that the terms $-h_c(q, \dot{q})$ and $-P_c\Gamma$ in Equation (5.15) are the effective torques due to the contact forces. The input torque to the system is not $(I - P_c)\Gamma$ but Γ. Equation (5.14) and (5.15) completely describe the system in contact. Given Γ, q, and $\dot{q}$, the values of $\ddot{q}$ and f_c can be computed.

These equations can also be obtained by directly solving the equations of motion under the constraints of $\ddot{x}_c = J\ddot{q} + \dot{J}\dot{q} = 0$ [63]. Equation (5.14) and (5.15) are valid as long as Γ is chosen not to violate the contact conditions, which are determined by the friction coefficient and the geometry of the contacts as described in section 5.1.1.

5.1.3 Constrained dynamics in operational space

The operational space coordinates are the variables to be controlled for desired behaviors. Given the operational space coordinates, x, the corresponding Jacobian is defined as

$$\dot{x} = J\dot{q}. \quad (5.18)$$

The dynamic equations of motion in operational space coordinates, x, are obtained by pre-multiplying Equation (5.15) by JA^{-1}.

$$\ddot{x} + \eta(q, \dot{q}) + \zeta(q) = JA^{-1}N_c^T\Gamma, \quad (5.19)$$

where

$$\begin{aligned} N_c^T(q) =& I - P_c \\ \eta(q, \dot{q}) =& JA^{-1}N_c^T b - \dot{J}\dot{q} + JA^{-1}J_c^T\Lambda_c\dot{J}_c\dot{q} \quad (5.20) \\ \zeta(q) =& JA^{-1}N_c^T g. \end{aligned}$$

Using the torque and force relation, $\Gamma = J^T F$,

$$\ddot{x} + \eta(q, \dot{q}) + \zeta(q) = \Upsilon(q)F, \tag{5.21}$$

where

$$\Upsilon(q) = JA^{-1}N_c^T J^T. \tag{5.22}$$

In the case that $\Upsilon(q)$ is invertible,

$$\Lambda(q)\ddot{x} + \mu(q, \dot{q}) + p(q) = F, \tag{5.23}$$

where

$$\begin{aligned}
\Lambda(q) &= \Upsilon(q)^{-1} = [JA^{-1}(I - P_c)J^T]^{-1} && (5.24)\\
\bar{J}^T &= \Lambda JA^{-1}N_c^T && (5.25)\\
\mu(q, \dot{q}) &= \bar{J}^T b(q, \dot{q}) \\
&\quad -\Lambda \dot{J}\dot{q} + \Lambda JA^{-1}J_c^T \Lambda_c \dot{J}_c \dot{q} && (5.26)\\
p(q) &= \bar{J}^T g(q) && (5.27)
\end{aligned}$$

Equation (5.23) describes the *constrained* dynamics in operational space coordinates. That is, contacts are treated as constraints and the dynamic equations account for the contact forces. The matrix $\Lambda(q)$ is the operational space inertia matrix. The term $\mu(q, \dot{q})$ is the Coriolis/centrifugal force in the operational space and the term $p(q)$ is the gravity force in operational space.

5.2 Control framework

The total number of DOF of the robot is n in free space. However, only k torques can be provided since there are k joints: Thus, The robot is an under-actuated system in free space. At most k DOF motion can be controlled since 6 DOF motion cannot be controlled such as the position of the center of mass point.

In a contact situation, such as standing, there are motion constraints. Therefore, the degrees of under-actuation for motion decreases. For one point contact, the contact position is constrained. Therefore, the number of DOF for motion is $n-3$. In the case of standing on one foot with plane-contact, 6 DOF of motion are constrained; thus, the motion of the system is not under-actuated. In other words, the motion of humanoid system can be fully controlled within the limit of the contact constraints.

Having motion constraints of more than 6 DOF is particularly interesting since the motion of the system is, now, over-actuated. Twelve DOF of motion are constrained when standing on two feet with plane contact. The total DOF of motion is now $n-12$ while there are $k = n-6$ actuators. This over-actuated system for motion has redundancy in the contact force space. In this case, there are 6 redundant actuations, which cause 6 DOF of redundancy in a 12 DOF contact force space. Therefore, while the motion is not disturbed, the contact forces can be controlled within the 6 DOF redundancy.

Note that the dynamic equations for a humanoid robot include virtual joints; thus, the joint torques, Γ, have zero components corresponding to the virtual joints. The superscript, 0, on the torque vector, Γ, will be used to denote that it has zero torques at the virtual joints. The actuated joint torques are denoted as the $k \times 1$ vector, Γ^k. We define a $k \times n$ selection matrix S^k such that

$$\Gamma^k = S^k \Gamma^0 \tag{5.28}$$

$$\Gamma^0 = (S^k)^T \Gamma^k. \tag{5.29}$$

In the case where the virtual joints are the first 6 joints,

$$S^k = [\ 0_{k\times 6} \quad I_{k\times k}\ ]. \tag{5.30}$$

This selection matrix can also be used to include real un-actuated joints.

We refer to Equation (5.15) as the *constrained dynamic equations of motion* in the sense that the contact dynamics is included in the system dynamics using constraints. Designing a control strategy based on the equations of motion utilizes not only the dynamics of the system but also the contact forces. Introducing the contact forces into the dynamic equation provides us the possible torque space to control the whole humanoid system.

5.2.1 Task control

The control torque for the desired task of a humanoid can be composed using the *constrained dynamic equations of motion* (5.15) within the contact conditions. If the number of constraints is greater or equal to the number of un-actuated joints, the constrained motion of the system will be fully controllable.

The dynamic equations can be projected to any coordinate space to be controlled, such as the center of mass, the hip orientation, the position of a certain point, etc. When we deal with many control points or coordinates, these can be concatenated into a single coordinate vector, x. Alternatively, the separate coordinates can be handled using priorities in a recursive way [3, 45, 59, 8, 34]. The recursive formulation is explained in detail in Appendix B.

Given x as the coordinate vector to be controlled the dynamics in the coordinates, x, are given by Equation (5.23). For a desired acceleration, f^*, the necessary force to be applied is

$$F = \Lambda f^* + \mu + p. \tag{5.31}$$

The control force, F, can be generated by the torque, $\Gamma = J^T F$, in the case of a fully actuated system. However, when virtual joints are used to describe the entire system this torque may not be applicable because no actuation is provided at those virtual joints.

Given the control torque, Γ, the force, F, is given by the relation, (Figure 5.5)

$$F = \bar{J}^T \Gamma. \tag{5.32}$$

Using a selection matrix when there are joints with no actuation,

$$\begin{aligned} F =& \bar{J}^T \Gamma^0 \\ =& \bar{J}^T (S^k)^T \Gamma^k. \end{aligned} \tag{5.33}$$

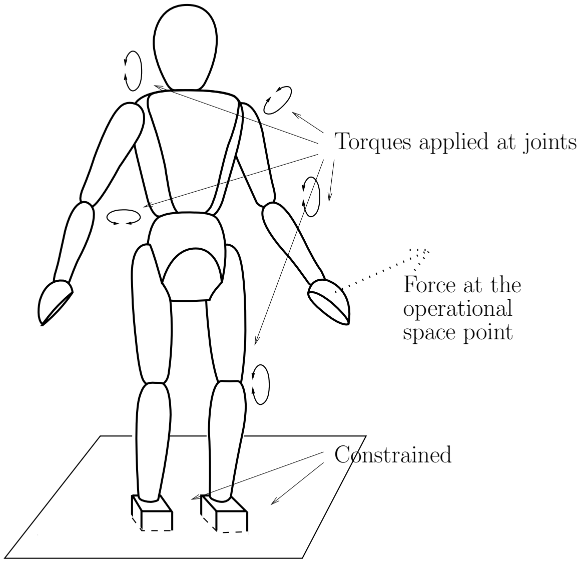

Torques applied at joints
Force at the operational space point
Constrained

Figure 5.5: **Representation of robot in contact as a constrained system**

Now, Γ^k must be chosen to produce the desired control force, F. If we denote m as the number of DOF of x then there are three cases: $m > k$, $m = k$, and $m < k$. In the case of $m > k$, F can not be produced as desired since there is an actuator deficiency. When $m = k$, Γ^k is uniquely determined if $\bar{J}^T(S^k)^T$ is invertible.

The most common and interesting situation in a high DOF system like a humanoid system is the case where $m < k$. In this case there are an infinite number of solutions for Γ^k that can achieve the desired force F. An intuitive way to resolve the redundancy is to minimize motion. That is, by minimizing the *acceleration energy* of the system, no unnecessary null-space motion will be produced. The *acceleration energy* is defined [5] as

$$E_a = \frac{1}{2}\ddot{\tilde{q}}^T A \ddot{\tilde{q}}, \tag{5.34}$$

where $\ddot{\tilde{q}}$ is the joint space acceleration induced by the control torque, i.e.,

$$\ddot{\tilde{q}} = A^{-1}(I - P_c)(S^k)^T \Gamma^k. \tag{5.35}$$

The *acceleration energy* can be expressed in terms of Γ^k as

$$E_a \quad = \quad \frac{1}{2}(\Gamma^k)^T W \Gamma^k, \tag{5.36}$$

where

$$\begin{aligned} W &= S^k(I - P_c)^T A^{-1}(I - P_c)(S^k)^T \\ &= S^k A^{-1}(I - P_c)(S^k)^T. \end{aligned} \tag{5.37}$$

The control torque, Γ^k, can be chosen as a solution to Equation (5.33) minimizing E_a. When W is rank-deficient, there are an infinite number of solutions that minimize the acceleration energy and produce the desired control force in operational space because there is redundancy in the contact force space. This can be resolved by specifying some of the contact forces or minimizing an additional quantity, e.g. the 2-norm of a torque vector. More details can be found in Appendix A.

We now define the matrix, $(J^k)^T$, as one of the generalized inverses of $\bar{J}^T(S^k)^T$, which minimizes E_a.

$$(J^k)^T = \overline{\bar{J}^T(S^k)^T}. \tag{5.38}$$

The commanding torque can then be expressed as

$$\begin{aligned} \Gamma^k &= (J^k)^T F \\ &= (J^k)^T \Lambda\{f^* + \mu + p\}. \end{aligned} \tag{5.39}$$

When $\bar{J}^T(S^k)^T$ is not singular, perfect estimates of all the system matrices will provide

$$\ddot{x} = f^*. \tag{5.40}$$

5.2.2 Null space projection matrix

The null-space projection matrix in the actual torque space is defined as

$$(N^k)^T = I - (J^k)^T \overline{(J^k)^T} \tag{5.41}$$

The overall torque to control the operational space coordinates in addition to the null-space tasks is then

$$\Gamma^k = (J^k)^T F + (N^k)^T \Gamma_0^k \tag{5.42}$$

Note that no actuation of virtual joints reduced the control space of Γ. This fact results in $\bar{J}^T$ and J^T being replaced by $\bar{J}^T (S^k)^T$ and $(J^k)^T$, respectively. However, when all the joints are fully actuated, Equation (5.42) is simply equal to

$$\Gamma = J^T F + (N)^T \Gamma_0. \tag{5.43}$$

The torque, Γ^k, in Equation (5.42) will be simply referred as Γ_{task}^k in the following section.

5.2.3 Control of contact forces within the boundaries

The control torque for the desired task composed in the previous sections assumes that the contact forces are not limited. However, the contact forces are bounded within certain contact conditions. When they exceed one of the limits, the contact condition will change and the dynamics of the system will be altered. If we wish to preserve the contact state the contact forces generated by the control torque must be monitored and controlled to remain within the boundaries.

Given the composed task control torque, Γ_{task}^k, the resulting contact forces are computed from Equation (5.14).

$$f_{c,task} = \bar{J}_c^T (S^k)^T \Gamma_{task}^k - \mu_c - p_c. \tag{5.44}$$

Some of the contact forces may exceed the boundaries associated with the contact conditions. In this case we wish to add additional torques such that the resulting contact forces lie within the boundaries, i.e.,

$$\Gamma^k = \Gamma_{task}^k + \Gamma_{contact}^k. \tag{5.45}$$

Since the contact forces must be controlled to remain within the boundaries a selection matrix, S_c, can be used to select the components of the contact forces that exceed the limits.

$$\begin{aligned} f_{c,selected} &= S_c f_c \\ &= S_c f_{c,task} + S_c \bar{J}_c^T (S^k)^T \Gamma^k_{contact}. \end{aligned} \tag{5.46}$$

When some of the contact forces, $S_c f_c$, exceed the boundary values we set the desired values of those contact forces to be inside the boundary values. This ensures that the contact forces satisfy the contact conditions. Then,

$$\begin{aligned} \tilde{f}_{c,selected} &= f_{c,selected}|_{desired} - S_c f_{c,task} \\ &= S_c \bar{J}_c^T (S^k)^T \Gamma^k_{contact}. \end{aligned} \tag{5.47}$$

Having defined a matrix $(J_c^k)^T$ as

$$(J_c^k)^T = \overline{S_c \bar{J}_c^T (S^k)^T}, \tag{5.48}$$

The additional torque to control contact forces is given by

$$\Gamma^k_{contact} = (J_c^k)^T \tilde{f}_{c,selected}. \tag{5.49}$$

However, the torque for controlling contact forces, $\Gamma^k_{contact}$, affect the task control of the robot unless $\Gamma^k_{contact}$ is zero or in the null-space of W. This disturbance from $\Gamma^k_{contact}$ to the task control can be compensated for by applying additional torques in the null-space of this contact force control.

Also, note that in certain cases the contact forces can also be controlled without disturbing the task control. The matrix, W, in Equation (5.36) may be rank-deficient. The case of W being a rank-deficient matrix is that the system has more than 6 motion constraints, i.e., the motion of the humanoid robot is over-actuated. Therefore, there are remaining degrees of freedom in the torque space while generating the same motion. This torque space can be analyzed by performing *Singular Value Decomposition* of the symmetric matrix W.

$$\begin{aligned} W &= U\Sigma U^T && (5.50)\\ &= [u_1 u_2]\begin{bmatrix} \sigma_1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} u_1^T \\ u_2^T \end{bmatrix} && (5.51)\end{aligned}$$

The torques in the vector space spanned by the column vectors of u_2 will not change the *acceleration energy.* These torques do not produce any acceleration in the task coordinates, x, either. Therefore, the torques in the u_2 vector space can be used to modify *contact forces* without disturbing the task control.

5.3 Transition between contact states

Section 5.2 explained the control framework for task and contact forces in a given contact state such as one-foot plane contact or two-foot plane contact. However, realization of more complex behavior involves many behaviors in different contact states. For example, walking involves one foot contact and two foot contact. Therefore, transitions between them are necessary. This section explains how a smooth controlled transition in terms of joint torques can be implemented.

In general the contact state of a robot cannot be known exactly. The state has to be estimated from sensors and models of the robot and environment. Given the estimated contact state, the corresponding dynamics will be used for control.

The change in the dynamics due to the change of the contact state is discrete. That is, the number of rows of the contact Jacobian, J_c, will decrease when the robot loses the contact or vise versa. A proposed strategy to deal with this discreteness is *a controlled contact transition strategy.* The strategy is to make the contact force or the *expected* contact force zero near the corresponding contact transition: Thus, there will be no abrupt changes in dynamic equations or control torque. This will generate the smooth transition in control torques between contact states. However, this strategy cannot guarantee a smooth transition in motion since our planned contact transition relies on the estimated contact states.

The contacts in transition are denoted with a subscript, *trans*, and the remaining contacts are denoted with a subscript, *0*. Equation (5.14) can be rewritten as

$$f_c = \bar{J}_c^T\{\Gamma - b(q,\dot{q}) - g(q)\}. \tag{5.52}$$

i.e.

$$\begin{aligned} J_c^T f_c &= J_c^T \bar{J}_c^T\{\Gamma - b(q,\dot{q}) - g(q)\} \\ &= P_c\{\Gamma - b(q,\dot{q}) - g(q)\}. \end{aligned} \tag{5.53}$$

The contact Jacobian, J_c, is decomposed into two parts as

$$\begin{aligned} J_c^T &= [J_{c,0}^T \quad J_{c,trans}^T] \\ \bar{J}_c^T &= \begin{bmatrix} \bar{J}_{c,0}^T \\ \bar{J}_{c,trans}^T \end{bmatrix}. \end{aligned} \tag{5.54}$$

Then,

$$P_c = P_{c,0} + P_{c,trans}, \tag{5.55}$$

where

$$\begin{aligned} P_{c,0} &= J_{c,0}^T \bar{J}_{c,0}^T \\ P_{c,trans} &= J_{c,trans}^T \bar{J}_{c,trans}^T, \end{aligned} \tag{5.56}$$

and

$$f_c = \begin{bmatrix} f_{c,0} \\ f_{c,trans} \end{bmatrix}, \tag{5.57}$$

where

$$\begin{aligned} f_{c,0} &= \bar{J}_{c,0}^T\{\Gamma - b(q,\dot{q}) - g(q)\} \\ f_{c,trans} &= \bar{J}_{c,trans}^T\{\Gamma - b(q,\dot{q}) - g(q)\}. \end{aligned} \tag{5.58}$$

The contact transition effect comes from $P_{c,trans}$. That is, even if we design a controller to generate a smooth trajectory of motion, the dynamics is not continuous due to the fact that P_c is discrete at the transition. This discrete dynamics create an abrupt changes of the commanding torques because the control framework considers the contact forces as internal forces. Therefore, the discreteness of the torques can be eliminated by controlling the contact forces to be continuous at the transition. We control the transient contact force, $f_{c,trans}$, to become smoothly zero from in-contact state to the contact transition. For the same reason, we control the *expected* contact force, $f_{c,trans}$ to become smoothly zero before the robot makes the contact. This strategy, in fact, controls the contact points to be compliant in the transition by limiting the contact forces.

5.4 Simulations

The proposed contact consistent control framework has been verified in the SAI simulation environment [32]. The dynamics engine in the SAI environment uses fast algorithms for dynamics and collision computations [7, 52]. It provides not only a simulation environment but also an interactive interface for the user. The red lines in all the figures of humanoid simulations (Figures 5.6-5.11) represent the contact forces simulated in SAI.

5.4.1 Standing on two feet

Standing on two feet involves many possible contact configurations which result in stable standing. One natural configuration involves plane contact on both feet. In this case, each foot has 6 motion constraints, resulting in a total of 12 constraints. These constraints create a contact force space of 12 degrees of freedom. Since the number of constraints exceeds the number of virtual joints (6), 6 DOF in the 12 DOF contact force space can be controlled without disturbing motion of the robot.

Utilizing this space, the moments acting on both feet can be set to zero, with the exception of the moment about the axis connecting the center of mass points of the two feet. This is a 4 DOF condition resembling natural bipedal standing, where the moments on the foot are only used for balance in the forward/backward direction.

The desired contact forces are composed from this 4 DOF condition and the contact conditions for maintaining current contacts.

Compensation for the gravity torques and adhering to the above conditions are sufficient to enable the robot to stand on two feet with plane contacts. This assumes that the starting configuration of the robot is statically balanced, i.e. the robot's center of mass is in between the feet. To maintain balance in the presence of disturbance, the minimal control required is control over the center of mass point. This can be realized by choosing the position of the robot's center of mass as the operational space coordinate.

Applying operational space control only for the center of mass creates compliant behavior for the robot in response to disturbances or external forces. When external forces are applied to a given link of the robot, the controller moves the other links to maintain the center of mass at the desired position rather than maintaining a fixed joint configuration. This kind of compliant behavior cannot be realized when all the joints of the robot are controlled to follow specific trajectories.

5.4.2 Walking

In order to realize a walking behavior, the control variables, i.e. operational space coordinates, are chosen as the center of mass position and the orientation of the head, chest, and hip. In the one foot support phase, the foot in free space is also controlled. The primary control variables for walking are the center of mass and individual foot positions. The other coordinates are chosen to maintain the desired posture of the robot while walking.

Figure 5.6 and 5.7 display snapshots and trajectories, respectively. Figure 5.7 shows how the center of mass and the feet are coordinated during walking. The gait cycle is designed first to produce the desired foot trajectories. Then, the desired center of mass motion is composed based on the foot motion. In the two foot support phase, the center of mass is controlled to move toward the foot which will be the support foot in the next phase. In the one foot support phase the foot in free space is controlled to move to the desired foot placement. In this phase, the center of mass

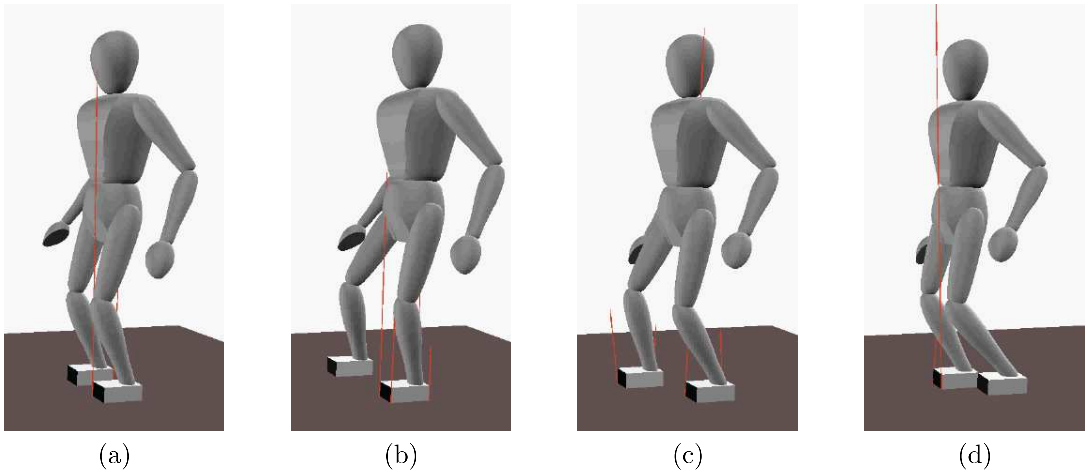


Figure 5.6: **Walking.** (a) Starting on two foot contact the center of mass is controlled to move toward the left foot. (b) Left foot supporting phase, where the right foot is controlled to move to the desired foot position. (c) Two foot supporting phase, where the center of mass is controlled to move toward the right foot. (d) Right foot supporting phase, where the left foot is controlled to move to the desired foot position.

position is also controlled to move toward the next supporting foot so that the robot moves its center of mass in advance before making two foot contact. This motion resembles human dynamic walking.

The dimension of the operational space is less than the total DOF of the system, leaving both arms without any specific controls. Therefore, the arms are compliant to disturbances during walking. Also, the controls for the posture, which include the orientation of head, chest, and hip, use very low gains such that the robot recovers its desired posture over time in the presence of disturbances, but is compliant at the instance of the disturbance. Therefore, it provides robustness to unexpected external forces imparted to the robot during walking.

5.4.3 Jumping

A jumping behavior is realized by controlling the same operational space coordinates as those in the *walking* behavior. The trajectory of the center of mass is designed such that it has a squatting phase, a leaping phase, a no-contact (airborne) phase, and a landing phase. The center of mass cannot be controlled when the robot is in the no-contact phase. However, once the feet have left the ground the system is controlled to generate a robust landing posture. Both feet are controlled to move to the expected landing position. The compliant behavior of the robot at the beginning of the landing is implemented by choosing low gains for all the controls of the operational space coordinates, particularly for the center of mass. Also, the contact forces applied to the environment are monitored and controlled to increase smoothly.

5.4.4 Climbing a ladder

The proposed contact consistent control framework can be applied to any robot link in contact. This is demonstrated by implementing a *ladder climbing* behavior (Figure 5.10). In this example, the hands are controlled to be in contact in order to maintain balance and also to control the ascension of the center of mass. A similar design procedure to that of the walking behavior is implemented to produce this complicated motion. The climbing cycle is produced by designing the foot and hand positions at the ladder contacts. The trajectory of the center of mass is chosen to maintain stable balance at all times.

5.4.5 Manipulation combined with walking

The final example involves the execution of a manipulation task while walking (Figure 5.11). In addition to walking, the hand is controlled to follow a trajectory in a contact plane and also to maintain contact forces in the normal direction of contact. This demonstrates the generality of the proposed control structure. That is, the proposed control framework enables us to implement hybrid motion/force control on humanoid robots using the operational space framework.

5.5 Conclusion

This chapter presents the complete dynamics of a humanoid system or, more generally, any robotic system in contact with the environment using virtual joints. The dynamic equations are derived by considering the system to be constrained by the contacts. The contact consistent dynamic formulation is essential for composing an appropriate control strategy for the robotic system.

The whole body control is accomplished by using a hybrid motion/force control approach in operational space. Motion design in operational space is more efficient and intuitive than in joint space because the operational space coordinates have physical meanings and their dimension is lower than that of joint space. The operational space coordinates may include the *center of mass* position as well as positions and orientations of a *foot*, the *hip*, and the *head*. The composition of control for additional behavior can be achieved by simply adding additional coordinates into the existing operational space coordinates. Since the control structure provides each coordinate with a decoupled second order linear system by compensating for the highly nonlinear dynamics, any linear control scheme can be applied to the control of each coordinate.

Additionally, control in operational space can use different gains for each coordinate. For example, high gains may be used for the center of mass control but low gains for chest orientation control to achieve good balancing characteristics. Therefore, complicated human behaviors can be designed by allowing for more possibilities in controller design. The simulation results demonstrate the characteristics of the proposed contact consistent control framework. Various human-like behaviors are designed and executed in simulation.

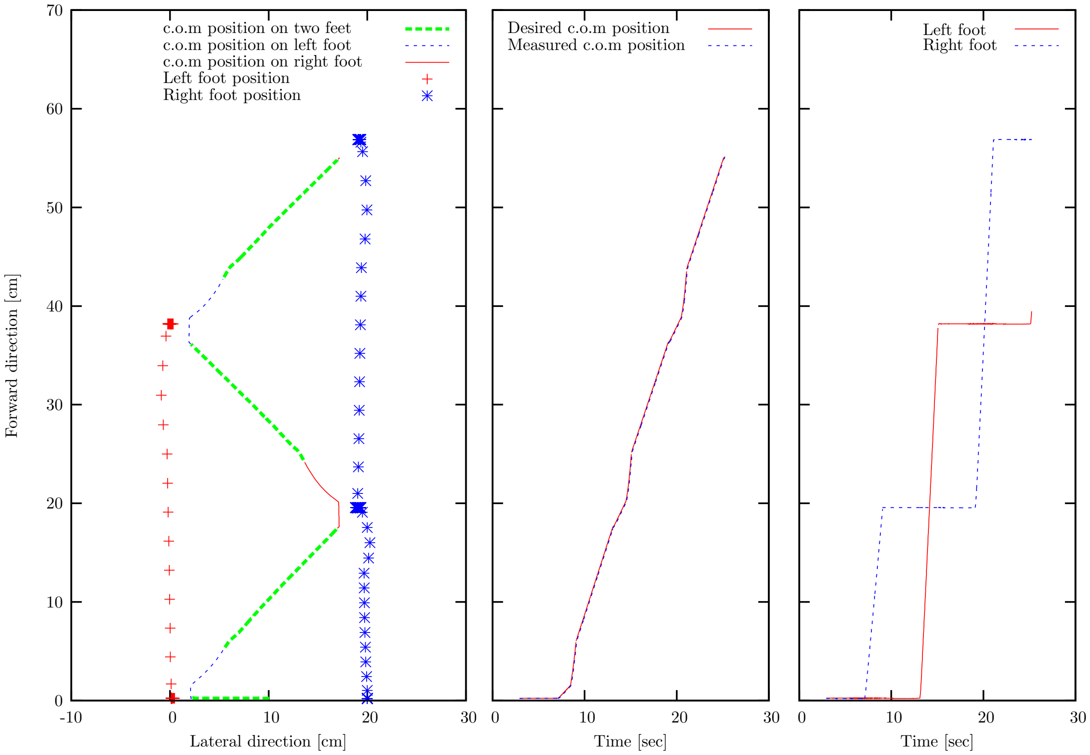


Figure 5.7: **Walking.** The robot starts from a standing configuration where both feet are side by side. The center of mass of the robot (green trajectory) is then moved over the left foot. The right foot is then moved forward (blue trajectory) while the left foot acts as support. In this phase, the center of mass is also controlled to move toward the right foot, resembling dynamic walking. The final phase involves transition between left and right foot contact. Both feet do not move but the center of mass moves toward the next support, in this case, the right foot. The gait cycle is then repeated.

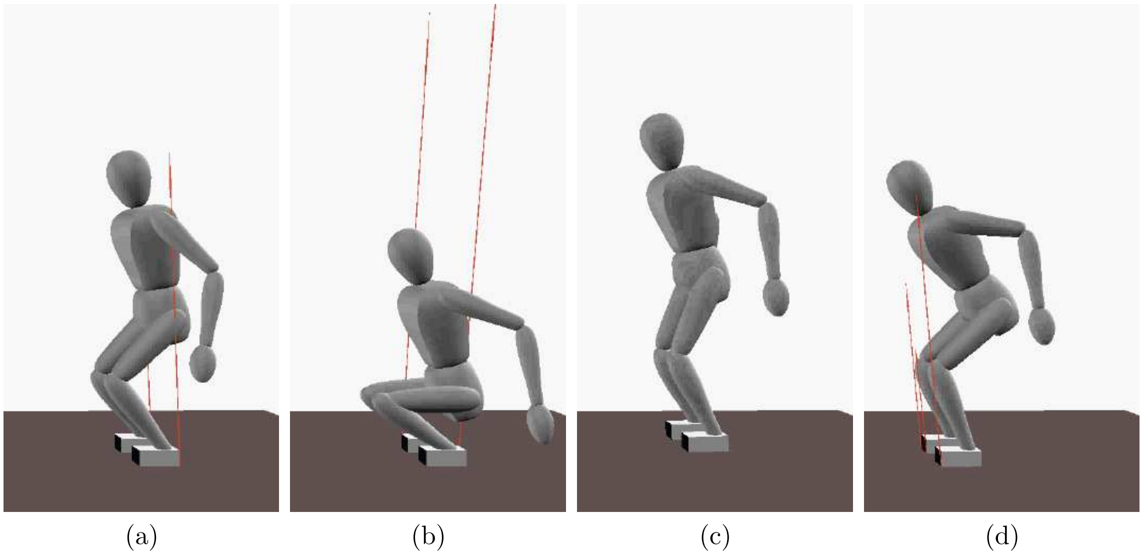


Figure 5.8: **Jumping.** (a) Squatting phase, where the robot is preparing to leap. (b) Leaping phase. (c) No contact, where the robot is preparing for landing. (d) Landing.

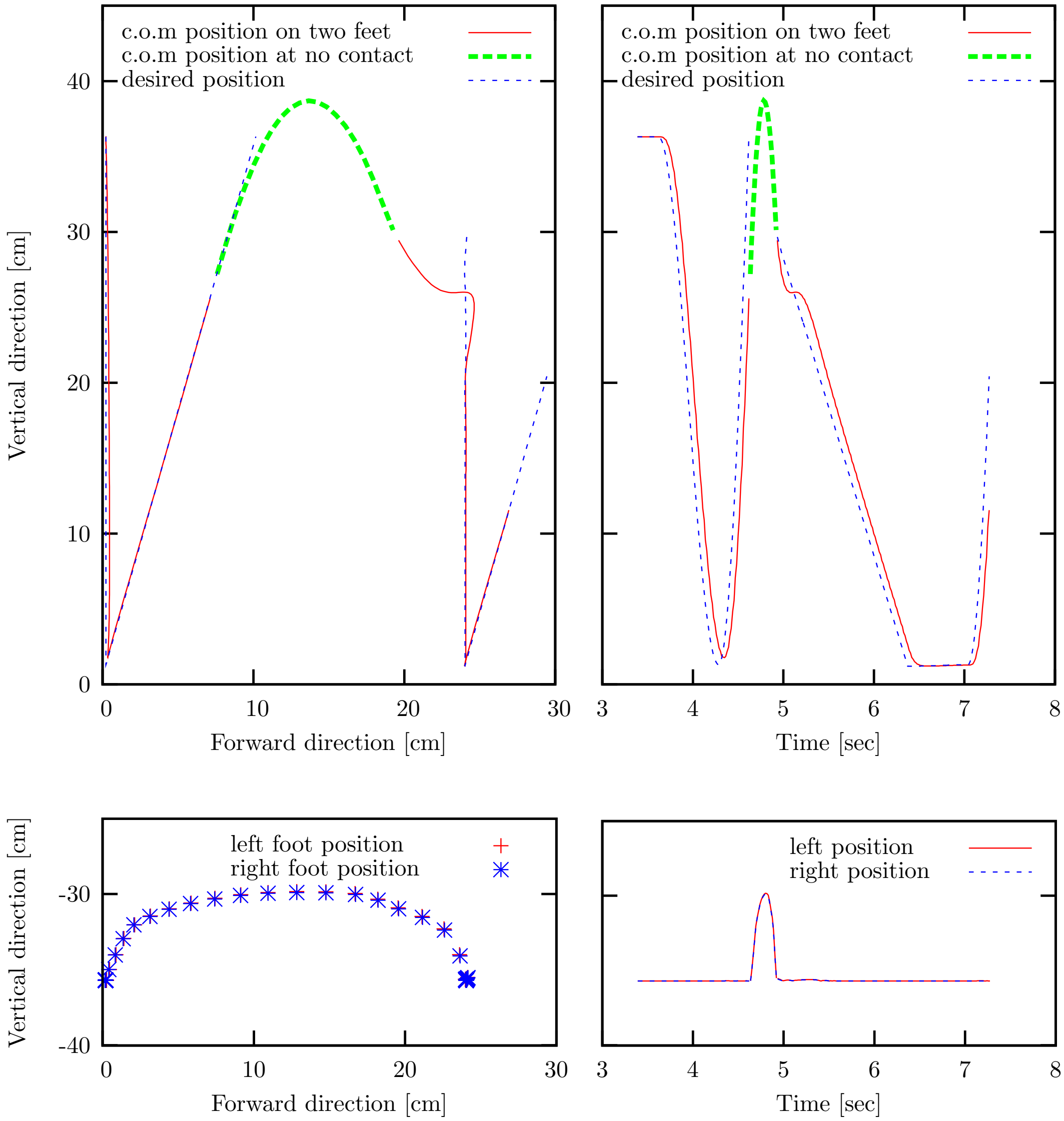


Figure 5.9: **Jumping.** The motion starts with a squatting phase, where the center of mass is controlled to move down. The leaping phase generates acceleration in the vertical direction. In the next phase the robot is airborne and the center of mass can no longer be controlled. The green line shows the parabolic trajectory of the center of mass in this phase. Both feet are controlled to move forward into a landing position. At the beginning of the landing phase, very small gains are set to control the center of mass such that the robot is compliant. This prevents bouncing on the ground due to the stiff motion of the body.

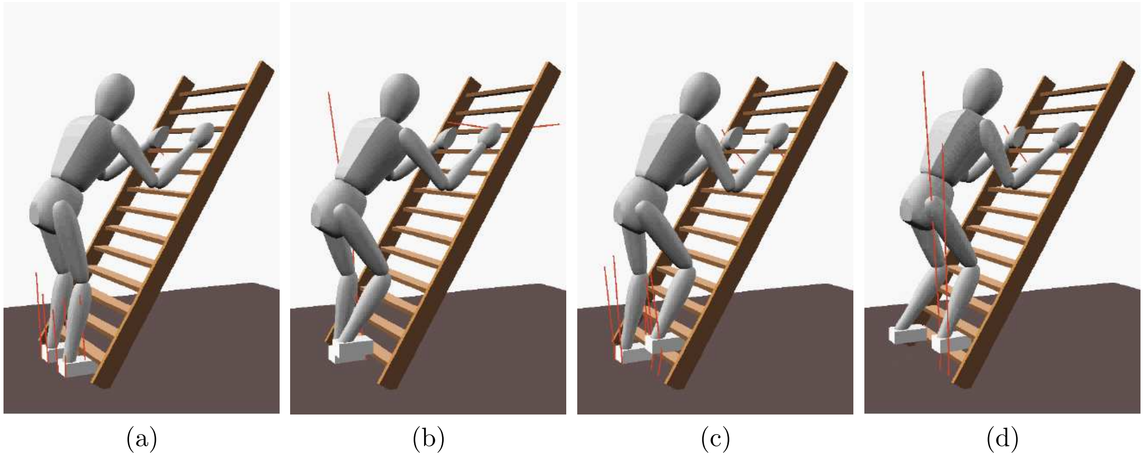


Figure 5.10: **Climbing a ladder.** (a) The robot begins to climb. It has contacts on both hands and feet. (b) The right foot is then controlled to move up one step. (c) Next, the center of mass is controlled to move to the right in order to maintain balance with two hands and the right foot. (d) The left foot is then controlled to move up one step.

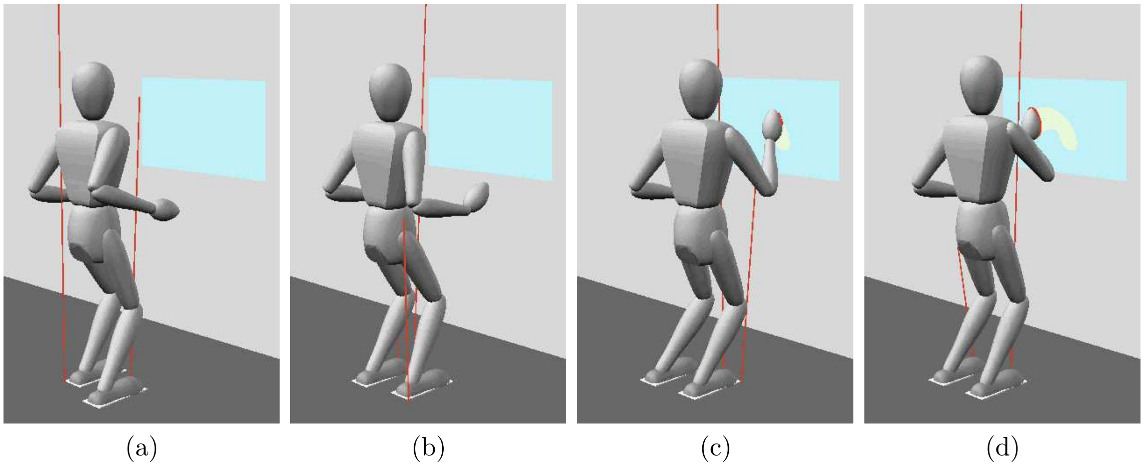


Figure 5.11: **Walking with manipulation.** (a) The robot walks to the window. (b) The right hand is then controlled to make contact with the window. (c) Next, the robot begins cleaning the window by applying a specified normal force and following a trajectory. (d) The robot is shown walking while controlling the hand motion.

Chapter 6

Conclusion

Control issues for robots in contact are addressed and control approaches are proposed in this thesis. In addressing complex control problems in multiple contact situations, one common feature is controlling contact force as well as the constrained motion in contact situations. This is particularly important not only to precisely control the contact force but also to maintain contacts if desired. Additionally, it is advantageous to have a control strategy decoupling the motion control from contact force control.

The contact force control approach in this thesis adapts Active Observers (AOB) into the operational space control framework to improve performance and consistency of the controller. The operational space control framework decouples the control structure for motion and contact force of the robot so that it provides more flexibility to implement various linear control schemes on the decoupled control structure. Among many linear control schemes, the AOB is chosen to deal with uncertainties in the system. The model reference approach implemented through AOB successfully realizes the robustness to unmodeled dynamics and parameter errors present in the model in addition to the performance improvement. This has been demonstrated through experiments using a PUMA560 manipulator and a DELTA haptic device. This force control approach is applied as a basis to haptic teleoperation and multicontact control.

A haptic teleoperation approach is developed based on the contact force control. Robustness is one of the most important factors in providing system stability and high fidelity force feedback to the user, particularly in teleoperation, where the robot is intended to interact with an object or an environment not planned a priori. The proposed teleoperation approach addresses the challenges of robustness and adaptiveness by integrating different components, such as a virtual spring, AOB contact force control, and the operational space control framework. This approach, in particular, compensates the dynamics of the robot using contact force control; thus, it overcomes the performance and stability issues on teleoperation systems consisting of kinematically and dynamically different master and slave systems. Stability is maintained while improving free space manipulability and contact force feedback by isolating the large slave dynamics. This has been demonstrated on a master-slave system consisting of PHANTOM haptic device and Stanford mobile platform (PUMA560 manipulator on a XR4000 mobile base). A moderate communication time-delay environment (a typical wireless network) was also tested to demonstrate the stability and performance characteristics of the system.

The need for operating robots in complex environment leads to the development of *a multi-link multi-contact control strategy.* This involves generalizing the idea of dividing the end-effector operational space into a general multi-contact force space and motion space. This generalization is realized by choosing control points at contact links and selecting the contact force spaces from each operational space at control points. The integrated operational space includes all these contact force spaces from different contact links. This approach decouples the contact force space from motion space in the whole system. Therefore, the unit mass systems for each contact are provided as individual controls. This approach takes into account the dynamics of the whole system in decoupling the force/motion control systems. The experimental setup uses the PUMA560 manipulator in contact with a table. Multi-link multi-contact controls were tested in the setup of two contacts at the end-effector and one contact at the third link. The results demonstrate the successful implementation of the control strategy in realizing simultaneous multiple contact force control over multiple links and motion control. The AOB contact force controller is implemented on this control strategy.

One robotic system that always involves a multicontact situation is a humanoid.

A humanoid robot is in contact with the ground to support its body. Therefore, multiple contacts over multiple links are present in natural configurations such as standing and walking. In addition to the contact on the foot links, the end-effector or the hand inevitably makes contact with the environment to execute useful tasks. In other situations, the hip link or elbow is likely to have contact with the environment. Addressing the control issues in this situation is not trivial and far more complicated than simple walking or free space motion of upper body. In an attempt to apply the multi-contact multi-link control framework to humanoids, *a contact consistent control framework* is developed to deal with a special robotic system - the non-fixed base robot. For control of these robots, contact with the environment is exploited to constrain the robot so that it obtains the ability of full control for the rest of the DOF. The control framework is composed based on the dynamics of the system, which include the contact dynamics. This approach enables a control algorithm to treat the contact forces as internal forces in the process of composing command torques. Therefore, it is not necessary to explicitly compute the desired contact forces for support or robot motion. However, there are boundaries that contact forces should remain within to maintain a current contact state. The contact forces are controlled to force the robot to remain in the desired contact state. The control uses operational space coordinates with special care taken for virtual joints. Therefore, the multi-link multi-contact control framework can be applied without any modification.

This control framework is verified in the SAI simulation environment. Demonstrated behaviors are standing, walking, jumping, climbing up a ladder, and manipulation with walking. Transition strategies between different contact states are also implemented to realize the series of behaviors in different contact states. The composition of these behaviors is done through designing trajectories in operational space coordinates such as the center of mass of the system, the hip position, a hand position, a foot position, etc. This design procedure is more intuitive than designing trajectories in joint space. The whole body control uses few necessary operational space coordinates for desired behaviors. The other remaining degrees of freedom are not specifically controlled; thus, the robot is compliant outside of the controlled coordinates. For example, the walking behavior requires only control of the feet and center of mass of the system. The other parts of the body such as the hands and the head can be left compliant if they are not specifically added to the operational space coordinates.

Future work

The contact force control framework developed in this thesis uses a reduced order model, particularly on the contact model. Having different assumptions or more sophisticated models may possibly improve the performance further. The model of the environment is chosen to have only stiffness since this captures the most important characteristic to be considered in the feedback controller. However, higher order models are also possible candidates. Techniques other than the Active Observer also should be investigated further. The stiffness estimation introduced in this thesis is mainly based on reasonable intuition and developed through experiments by trial and error. This is a good indication that more systematic and theoretical approaches could produce a better formulation and results. Many applications other than control issues can be investigated, such as identifying the characteristics of unknown objects.

In the contact consistent control framework for humanoids, only rigid environments have been considered in the dynamics. This is an approximation for relatively rigid contacts such as the ground although real environments always have dynamic properties such as compliance or inertia. Therefore, the control framework needs to be further extended for general contact environments. Although the basic concept of the formulation would be the same, dynamic decoupling is possibly complicated in the case of incorporating elasticity of the environment.

The current open loop control for contact force may not be sufficient when implementing the algorithm on real humanoid systems. Utilizing force sensors on the contact links, approaches that combine the contact consistent control framework with closed loop control for the supporting contacts may be required for better contact state stability.

The current formulation selects some contacts in constraining motion and obtains the dynamics. The other contacts are controlled through a motion/force control framework using the operational space control framework. This approach is yet to be generalized into one unifying formulation so that all the contacts would be treated in a consistent way.

This thesis has developed and investigated many important components on the subject of constrained motion control. It is believed that further development based on the current work will contribute to the general control framework for composing constrained motion in a complex multicontact environment for robotic systems.

Bibliography

[1] H. Ambrose and Z. Qu. Model reference robust control for mimo systems. In *Proc. of the American Control Conference*, pages 345–349, Albuquerque, New Mexico, 1997.

[2] R. J. Anderson and M. W. Spong. Bilateral control of teleoperators with time delay. *IEEE Transactions on Automatic Control*, 34:494–501, May 1989.

[3] P. Baerlocher and R. Boulic. Task-priority formulations for the kinematic control of highly redundant articulated structures. In *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 323–329, Victoria, B.C., Canada, 1998.

[4] H. Bruynincks, S. Demey, S. Dutré, and J. De Schutter. Kinematic models for model-based compliant motion in the presence of uncertainty. *Int. J. of Robotics Research*, 14(5):465–482, Oct 1995.

[5] H. Bruyninckx and O. Khatib. Gauss' principle and the dynamics of redundant and constrained manipulators. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 470–475, 2000.

[6] K. Burn, M. Short, and R. Bicker. Adaptive and nonlinear fuzzy force control techniques applied to robots operating in uncertain environments. *Journal of Robotic Systems*, 20(7):391–400, 2003.

[7] K. C. Chang and O. Khatib. Operational space dynamics: Efficient algorithms for modeling and control of branching mechanisms. In *Proc. of the Int. Conf. on Robotics and Automation*, 2000.

[8] S. Chiaverini. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *Int. J. on Robotics and Automation*, 13(3):398–410, June 1997.

[9] S. Chiaverini and L. Sciavicco. The parallel approach to force/position control of robotic manipulators. *IEEE Transactions on Robotics and Automation*, 9(4):361–373, August 1993.

[10] S. Chiaverini, B. Siciliano, and L. Villani. A survey of robot interaction control schemes with experimental comparison. *ASME Transactions on Mechatronics*, 4(3):273–285, September 1999.

[11] R. Cortesão. *Kalman Techniques for Intelligent Control Systems: Theory and Robotic Experiments*. PhD thesis, University of Coimbra, 2002.

[12] R. Cortesão, J. Park, and O. Khatib. Real-time adaptive control for haptic manipulation with active observers. In *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, Las Vegas, 2003.

[13] K. Daneko, F. Kanehiro, S. Kajita, K. Yokoyama, K. Akachi, T. Kawasaki, S. Ota, and T. Isozumi. Design of prototype humanoid robotics platform for hrp. In *Proc. of the Int. Conf. on Intelligent Robots and Systems*, pages 2431–2436, 2002.

[14] R. Daniel and P. McAree. Fundamental limits of performance for force reflecting teleoperation. *International Journal of Robotics Research*, 17(8):811–830, August 1998.

[15] J. C. Doyle and G. Stein. Robustness with observers. *IEEE Transactions on Automatic Control*, AC-24(4):607–611, August 1979.

[16] D. Erickson, M. Weber, and I. Sharf. Contact stiffness and damping estimaation for robotic systems. *International Journal of Robotics Research*, 22(1):41–57, 2003.

[17] R. Featherstone, S. S. Thiebaut, and O. Khatib. A general contact model for dynamically-decoupled force/motion control. In *Proc. of the Int. Conf. on Robots and Automation*, pages 3281–3286, 1999.

[18] G. F. Franklin, J. D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems, 4th edition.* Prentice Hall, 2002.

[19] E. Freund. The structure of decoupled nonlinear systems. *Int. J. Contr.*, 21(3):443–450, 1975.

[20] B. Hannaford. A design framework for teleoperators with kinesthetic feedback. *IEEE Transactions on Robotics and Automation*, 5:426–434, August 1989.

[21] B. Hannaford, L. Wood, D. A. McAffee, and H. Zak. Performance evaluation of a six-axix generalized force-reflecting teleoperator. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):620–633, May/June 1991.

[22] K. Hashtrudi-Zaad and S. E. Salcudean. Transparency in time-delayed systems and the effect of local force feedback for transparent teleoperation. *Int. J. on Robotics and Automation*, 18(1):108–114, February 2002.

[23] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of honda humanoid robot. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1321–1326, 1998.

[24] N. Hogan. Impedance control: An approach to manipulation, parts i-iii. *ASME J. Dynam. Syst., Meas., Contr.*, 107:1–24, 1985.

[25] R. Holmberg and O. Khatib. Development and control of a holonomic mobile robot for mobile manipulation tasks. *International Journal of Robotics Research*, 19(11):1066–1074, 2000.

[26] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie. Planning walking patterns for a biped robot. *IEEE Transactions on Robotics and Automation*, 17(3), June 2001.

[27] C. D. Johnson. Discrete-time disturbance-accommodating control theroy with applications to missile digital control. *Journal of Guidance and Control*, 4(2):116–125, 1980.

[28] S. Kagami, K. Nishiwaki, T. Kitagawa, T. Sugihiara, M. Inaba, and H. Inoue. A fast generation method of a dynamically stable humanoid robot trajectory with enhanced zmp constraint. In *Proc. of the Int. Conf. on Humanoid Robotics*, 2000.

[29] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1620–1626, 2003.

[30] S. Kajita and K. Tani. Study of dynamic biped locomotion on rugged terrain. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1405–1410, 1991.

[31] O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *Int. J. on Robotics and Automation*, 3(1):43–53, February 1987.

[32] O. Khatib, O. Brock, K. C. Chang, F. Conti, D. Ruspini, and L. Sentis. Robotics and interactive simulation. *Communications of the ACM*, 45(3):46–51, March 2002.

[33] O. Khatib and J. Burdick. Motion and force control of robot manipulators. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1381–1386, 1986.

[34] O. Khatib, L. Sentis, J. Park, and J. Warren. Whole-body dynamic behavior and control of human-like robots. *Int. J. of Humanoid Robotics*, 1(1):29–43, 2004.

[35] J. Kim, I. Park, J. Lee, M. Kim, B. Cho, and J. Oh. System design and dynamic walking of humanoid robot khr-2. In *Proc. of the Int. Conf. on Robotics and Automation*, 2005.

[36] W. S. Kim, B. Hannaford, and A. K. Bejczy. Force-reflecting and shared compliant control in operating telemanipulators with time delay. *Int. J. on Robotics and Automation*, 8:176–185, April 1992.

[37] A. D. Kuo. An optimal control model for analyzing human postural balance. *IEEE Transactions on Biomedical Engineering*, 42(1):87–101, Jun 1995.

[38] D. Lawrence. Stability and transparancy in bilateral teleoperation. *IEEE Trans. on Robotics and Automation*, 9(5):624–637, October 1993.

[39] H. Lipkin and J. Duffy. Hybrid twist and wrench control for a robotic manipulator. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 110(2):138–144, Jun 1988.

[40] Y. Liu, K. Kitagaki, T. Ogasawara, and S. Arimoto. Model-based adaptive hybrid control for manipulators under multiple geometric constraints. *IEEE Trans. on Control Systems Technology*, 7(1):97–109, January 1999.

[41] L. Love and W. Book. Environment estimation for enhanced impedance control. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1854–1859, 1995.

[42] M. Mason. Compliance and force control for computer controlled manipulators. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-11(6):418–432, June 1981.

[43] P. S. Maybeck. *Stochastic Models, Estimation and Control (Volume 2)*, volume 141-2 of *Mathematics In Science and Engineering.* Academic Press, 1982. (Edited by R. Bellman).

[44] Napoleon, S. Nakaura, and M. Sampei. Balance control analysis of humanoid robot. In *Proc. of the Int. Conf. on Intelligent Robots and Systems*, pages 2437–2442, 2002.

[45] D. N. Nenchev and Z. M. Sotirov. Dynamic task-priority allocation for kinematically redundant robotic mechanisms. In *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 518–524, 1994.

[46] G. Niemeyer and J. Slotine. Stable adaptive teleoperation. *IEEE Journal of Oceanic Engineering*, 16(1):152–162, January 1991.

[47] G. Niemeyer and J. Slotine. Telemanipulation with time delays. *The International Journal of Robotics Research*, 23:873–890, September 2004.

[48] J. Park, R. Cortesão, and O. Khatib. Multi-contact compliant motion control for robotic manipulators. In *Proc. of the Int. Conf. on Robotics and Automation*, New Orleans, U.S.A., 2004.

[49] J. Pratt, P. Dilworth, and G. Pratt. Virtual model control of a bipedal walking robot. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 193–198, 1997.

[50] J. Pratt and G. Pratt. Intuitive control of a planar biped walking robot. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 2014–2021, 1998.

[51] M. H. Raibert and J. J. Craig. Hybrid position/force control of manipulators. *ASME Journal of Dynamic Systems, Measurement and Control*, 103(2):126–133, June 1981.

[52] D. Ruspini and O. Khatib. Collision/contact models for dynamic simulation and haptic interaction. In *The 9th International Symposium of Robotics Research*, pages 185–195, Snowbird, USA, 1999.

[53] J. Ryu, D. Kwon, and B. Hannaford. Stable teleoperation with time-domain passivity control. *IEEE Transactions on Robotics and Automation*, 20(2):365–373, April 2004.

[54] J. K. Salisbury. Active stiffness control of a manipulator in cartesian coordinates. In *Proc. 19th Conf. Decision and Control*, pages 95–100, Albuquerque, NM, 1980.

[55] A. Sano and J. Furusho. Realization of natural dynamic walking using the angular momentum information. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1476–1481, 1990.

[56] V. De Sapio and O. Khatib. Operational space control of multibody systems with explicit holonomic constraints. In *Proc. of the Int. Conf. on Robotics and Automation*, 2005.

[57] J. De Schutter and H. Van Brussel. Compliant robot motion ii. a control approach based on external control loops. *International Journal of Robotics Research*, 7(4):18–33, 1988.

[58] H. Seraji and R. Colbaugh. Force tracking in impedance control. *International Journal of Robotics Research*, 16(1):97–117, 1997.

[59] B. Siciliano and J. E. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In *Proc. of the Int. Conf. on Advanced Robotics*, pages 1211–1216, 1991.

[60] B. Siciliano and L. Villani. *Robot Force Control.* The Kluwer International Series In Engineering and Computer Science. Kluwer Academic Publishers, 1999.

[61] S. Singh and D. Popa. An analysis of some fundamental problems in adaptive control of fooorce and impedance behavior: Theory and experiments. *IEEE Transactions on Robotics and Automation*, 11(6):912–921, 1995.

[62] T. Sugihara, Y. Nakamura, and H. Inoue. Realtime humanoid motion generation through zmp manipulation based on inverted pendulum control. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 1404–1409, 2002.

[63] F. E. Udwadia and R. E. Kalaba. *Analytical Dynamics: A New Approach.* Cambridge: Cambridge University Press, 1996.

[64] H. West and H. Asada. A method for the design of hybrid position/force controllers for manipulators contrained by contact with the environment. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 251–259, 1985.

[65] J. Yamaguchi, E. Soga, S. Inoue, and A. Takanishi. Development of a bipedal humanoid robot - control method of whole body cooperative dynamic biped

walking-. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 368–374, 1999.

[66] Y. Yokokohji and T. Yoshikawa. Bilateral control of master-slave manipulators for idela kinesthetic coupling. *IEEE Transactions on Robotics and Automation*, 10:605–620, October 1994.

[67] T. Yoshikawa. Force control of robot manipulators. In *Proc. of the Int. Conf. on Robotics and Automation*, pages 220–226, San Francisco, USA, 2000.

[68] W. Zhu and S. Salcudean. Stability guaranteed teleoperation: An adaptive motion/force control approach. *IEEE Trans. on Automatic Control*, 45(11):1951–1969, November 2000.

Appendix A

Under-constrained linear system

An under-constrained linear system of equations can be described as

$$y = Qx, \tag{A.1}$$

where $y \in R^m$, $x \in R^n$, Q is an $m \times n$ non-singular matrix and $n > m$. Note that only non-singular cases are considered in this Appendix. In accordance with Chapter 5, y is the desired force and x is the torque.

In general, there will be an infinite number of solutions to this problem, but a unique solution can be chosen by minimizing a weighted norm of x. If W is a positive definite matrix, then the weighted norm of x is defined by

$$||x||_W^2 =< x, x >_W = x^T W x, \tag{A.2}$$

and the unique solution is

$$x = W^{-1} Q^T (Q W^{-1} Q^T)^+ y. \tag{A.3}$$

However, we may also want to minimize $x^T W x$ when the weighting matrix, W, is positive semi-definite (as may occur in Equation (5.37)). In the case there are, once again, an infinite number of solutions to the system.

Solutions of the system can be described by using the *singular value decomposition*

(SVD) of Q.

$$Q = USV^T = [u_1 u_2] \begin{bmatrix} \sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_1^T \\ v_2^T \end{bmatrix}. \tag{A.4}$$

Based on the SVD, the *Least Squares* solution can be expressed as

$$\begin{aligned} x &= v_1 \sigma_1^{-1} u_1^T y + v_2 \alpha && \text{(A.5)} \\ &= \bar{x} + v_2 \alpha, && \text{(A.6)} \end{aligned}$$

where α can be any arbitrary vector. Using this solution, $x^T W x$ can be written as

$$x^T W x = (\bar{x} + v_2 \alpha)^T W (\bar{x} + v_2 \alpha). \tag{A.7}$$

To minimize $x^T W x$, we set the first derivative to zero:

$$\frac{\partial}{\partial \alpha}(x^T W x) = 0, \tag{A.8}$$

$$\Rightarrow v_2^T W (\bar{x} + v_2 \alpha) = 0. \tag{A.9}$$

Now, we solve the above equation for α:

$$\alpha = -(v_2^T W v_2)^+ v_2^T W \bar{x}. \tag{A.10}$$

Therefore,

$$x = \bar{x} - v_2 (v_2^T W v_2)^+ v_2^T W \bar{x}, \tag{A.11}$$

where

$$\bar{x} = v_1 \sigma_1^{-1} u_1^T y. \tag{A.12}$$

In a more compact form,

$$\begin{aligned} x &= \{(I - v_2 (v_2^T W v_2)^+ v_2^T W) v_1 \sigma_1^{-1} u_1^T\} y \\ &= Q^{W+} y. \end{aligned} \tag{A.13}$$

Note that the use of Moore-Penrose pseudo inverse on $v_2^T W v_2$ is only one of many possible solutions, and it minimizes the 2-norm in the vector space of v_2. Minimization of other quantities is possible since the choice does not affect the minimization of $x^T W x$. The inverse, Q^{W+}, is referred to the dynamically consistent inverse, $\bar{Q}$, in this thesis. Because $x^T W x$ represents the *acceleration* energy of the system, the resulting solution minimizes the null-space motion of the system.

Appendix B

Recursive control structure with priorities

When multiple variables are to be controlled simultaneously, all of them may be concatenated into a vector of coordinates, x. Alternatively, they can be divided into several sets of vectors so that the coordinates with higher priority are guaranteed to be controlled. This approach is based on a recursive structure for generating control torque input [3, 45, 59, 8, 34]. In this appendix we consider the recursive control structure that can be used to compose the control torque in the contact consistent control framework of Chapter 5. Note that the notation used here is based on that of Chapter 5.

If x_i is the set of coordinates with highest priority, the other set of coordinates, x_j, can be controlled using the rest of the DOF of the system but only after taking care of the control variable, x_i. Therefore, we want to have a control structure such that additional control torques for controlling x_j can be added to the control torques for x_i without disturbing the control for x_i. That is,

$$\Gamma^k = \Gamma_i^k + \Gamma_j^k. \tag{B.1}$$

Since Γ_j^k should not produce any acceleration of x_i, the null-space projection matrix, $(N_i^k)^T$, is used in composing the control torque, Γ_j^k. This guarantees that x_i is not disturbed by Γ_j^k (Equation (5.42)).

$$\begin{aligned} \Gamma_i^k &= (J_i^k)^T F_i, && \text{(B.2)} \\ \Gamma_j^k &= (N_i^k)^T \Gamma_{j,i}^k. && \text{(B.3)} \end{aligned}$$

Together, this gives:

$$\Gamma^k = (J_i^k)^T F_i + (N_i^k)^T \Gamma_{j,i}^k. \tag{B.4}$$

In other words, x_j is to be controlled in the null space of the control of x_i, so that it does not affect the control of x_i. Given the total control torque, Γ^k, the dynamic equation for x_j is obtained by pre-multiplying the *constrained dynamic equations* of the system, Equation (5.15), by $J_j A^{-1}$:

$$\ddot{x}_j + \eta_j + \zeta_j = J_j A^{-1} N_c^T (S^k)^T \Gamma_i^k + J_j A^{-1} N_c^T (S^k)^T (N_i^k)^T \Gamma_{j,i}^k, \tag{B.5}$$

where

$$\begin{aligned} \eta_j(q,\dot{q}) =& J_j A^{-1} N_c^T b - \dot{J}_j \dot{q} + J_j A^{-1} J_c^T \Lambda_c \dot{J}_c \dot{q} \\ \zeta_j(q) =& J_j A^{-1} N_c^T g \\ (N_i^k)^T =& I - (J_i^k)^T \overline{(J_i^k)^T}. \end{aligned} \tag{B.6}$$

This can also be described in the force space of the coordinates, x_j,

$$\Lambda_j \ddot{x}_j + \mu_j + p_j = \bar{J}_j^T (S^k)^T \Gamma_i^k + \bar{J}_j^T (S^k)^T (N_i^k)^T \Gamma_{j,i}^k, \tag{B.7}$$

where

$$\begin{aligned} \Upsilon_j(q) =& J_j A^{-1} N_c^T J_j^T \\ \Lambda_j(q) =& \Upsilon_j^{-1} \\ \mu_j(q,\dot{q}) =& \Lambda_j \eta_j \\ p_j(q) =& \Lambda_j \zeta_j \\ \bar{J}_j^T =& \Lambda_j J_j A^{-1} N_c^T. \end{aligned} \tag{B.8}$$

Now, by defining $(J^k_{j,i})^T$ as the dynamically consistent inverse of $\bar{J}^T_j(S^k)^T(N^k_i)^T$:

$$(J^k_{j,i})^T = \overline{\bar{J}^T_j(S^k)^T(N^k_i)^T}, \tag{B.9}$$

the torque command, $\Gamma^k_{j,i}$, can be composed to compensate for nonlinear dynamics of the system as follows:

$$\Gamma^k_{j,i} = (J^k_{j,i})^T\{\Lambda_j f^*_j + \mu_j + p_j - \bar{J}^T_j(S^k)^T\Gamma^k_i\}. \tag{B.10}$$

In summary, the choice of the control torque

$$\Gamma^k = (J^k_i)^T F_i + (N^k_i)^T(J^k_{j,i})^T F_j \tag{B.11}$$

with

$$\begin{aligned} F_i &= \Lambda_i f^*_i + \mu_i + p_i & \text{(B.12)}\\ F_j &= \Lambda_j f^*_j + \mu_j + p_j - \bar{J}^T_j(S^k)^T\Gamma^k_i & \text{(B.13)}\end{aligned}$$

provides the unit mass systems for both x_i and x_j. That is,

$$\begin{aligned} \ddot{x}_i &= f^*_i & \text{(B.14)}\\ \ddot{x}_j &= f^*_j. & \text{(B.15)}\end{aligned}$$

So far, we have assumed there is no singularity. However, the robot may be in a singular configuration caused by the conflicting goal between the tasks, x_i and x_j. In this case, part of x_j will not be controlled as desired since x_i has higher priority. The control structure for more than two sets of the operational space coordinates can be recursively constructed with the same procedure.

Although this recursive formulation is described using the notation for the contact consistent control framework, it is general enough to be applied to any robotic system. In the case of fixed base manipulators, the matrix, N^T_c, becomes an identity matrix. When there are no virtual joints or un-actuated joints, the selection matrix, S^k, will be an identity matrix and the superscript, k, may be dropped.

The presented formulation differs slightly from the task-posture decomposition control structure in Chapters 3 and 4. However, they are essentially the same formulation in providing unit mass systems by decoupling the highly nonlinear robotic system. The difference is that we have considered a more general approach in this Appendix, in the sense that we can command not only motion but also force on the coordinates if so desired. This force can be commanded by adding it to Equation (B.12) or (B.13).