12-2006

Nonlinear Control Techniques for Robot Manipulators

Nitendra Nath
Clemson University, nnath@clemson.edu

NONLINEAR CONTROL TECHNIQUES FOR ROBOT MANIPULATORS

A Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Electrical Engineering

---

by
Nitendra Nath
December 2006

Accepted by:
Dr. Darren Dawson, Committee Chair
Dr. Ian Walker
Dr. Timothy Burg

ABSTRACT

This Masters Thesis describes the design and implementation of control strategies for the following topics of research: *i)* Whole Arm Grasping Control for Redundant Robot Manipulators, *ii)* Neural Network Grasping Controller for Continuum Robots and, *iii)* Coordination Control for Haptic and Teleoperator Systems.

An approach to whole arm grasping of objects using redundant robot manipulators is presented. A kinematic control which facilitates the encoding of both the end-effector position, as well as body self-motion positioning information as a desired trajectory signal for the manipulator joints is developed. A joint space controller which provides asymptotic tracking of the encoded desired trajectory in the presence of system uncertainties is presented. Experimental results for a planar, three link configuration of the Barrett whole arm manipulator are provided to illustrate the validity of the approach.

Continuum or hyper-redundant robots are robots which exhibit behavior similar to biological trunks, tentacles and snakes. Unlike traditional robots, continuum robot manipulators do not have rigid joints, hence the manipulators are compliant, extremely dexterous, and capable of dynamic, adaptive manipulation in unstructured environments; however, the development of high-performance control algorithms for these manipulators is a challenging problem. In this paper, we present an approach to whole arm grasping control for continuum robots. The grasping controller is developed in two stages; high level path planning for the grasping objective, and a low level joint controller using a neural network feedforward component to compensate for dynamic uncertainties. These techniques are used to enable whole arm grasping without using contact force measurements and without

using a dynamic model of the continuum robot. Experimental results using the OCTARM, a soft continuum robotic manipulator are included to illustrate the efficacy of the proposed low level joint controller.

Two controllers are developed for nonlinear haptic and teleoperator systems for coordination of the master and slave systems. The first controller is proven to yield a semi-global asymptotic result in the presence of parametric uncertainty in the master and the slave dynamic models provided the user and the environmental input forces are measurable. The second controller yields a global asymptotic result despite unmeasurable user and environmental input forces provided the dynamic models of the master and slave systems are known. These controllers rely on a transformation and a flexible target system to allow the master system's impedance to be easily adjusted so that it matches a desired target system. This work also offers a structure to encode a velocity field assist mechanism to provide the user help in controlling the slave system in completing a pre-defined contour following task. For each controller, Lyapunov-based techniques are used to prove that both controllers provide passive coordination of the haptic/teleoperator system when the velocity field assist mechanism is disabled. When the velocity field assist mechanism is enabled, the analysis proves the coordination of the haptic/teleoperator system. Experiment results are presented for both controllers.

DEDICATION

This thesis is dedicated to my parents whose love and encouragement has made it possible.

ACKNOWLEDGMENTS

I would like to take this opportunity to thank all those who helped me in my pursuit of a Masters degree. I express deep gratitude to my advisor, Dr. Darren Dawson for his support, guidance and for motivating me to strive for higher levels of excellence in my work. I would also like to thank my committee members, Dr. Ian Walker and Dr. Timothy Burg for all their technical guidance.

I would also like to thank my colleague and friend, Ninad Pradhan for his support and encouragement during the course of my experimental work. Special thanks go to the people at the Controls and Robotics group, especially Enver Tatlicioglu for his invaluable help and assistance. His support in time of need was greatly appreciated.

David Braganza deserves special credit for guiding me in almost every project I was associated with. His patience and exemplary working habits have motivated me throughout my research work in Clemson.

Special thanks to Gauri Phadke for her altruistic support and presence, during last few months, that played a significant role in the completion of my work.

Thanks also go out to my roommates, past and present, Shyam Panyam, Rohit Jagnani, Sourabh Patki, Parth Bhavsar, Satyen Raina and all my friends at Clemson whose support helped me concentrate on my work. Finally, this acknowledgment would be incomplete without thanking my parents, my sister and my brother-in-law whose motivation and support at all times, has made me what I am.

TABLE OF CONTENTS

Table of Contents (Continued)

Page

LIST OF TABLES

LIST OF FIGURES

List of Figures (Continued)

CHAPTER 1

INTRODUCTION

Whole Arm Grasping Control for Redundant Robot Manipulators

One of the advantages of redundant robot manipulators is their ability to perform whole arm grasping of objects. Whole arm manipulation [18] is the term used to describe the ability of the manipulator to grasp an object with its entire body (or arm), as compared to fingertip grasping performed by traditional robotic grippers and hands. Whole arm grasping can be performed by allowing the robot manipulator to make contact with the object in a snake or tentacle like manner, using portions of the manipulator itself to wrap around the object and grasp it. The equivalent whole hand and whole finger grasping techniques have been studied in [11] and [17], respectively. Whole arm grasping is also known by the equivalent expressions "power grasping" ([12] and [24]) or "enveloping grasping" [21]. Recently in [16], the authors presented experimental results which demonstrated whole hand grasping with a 12 degree-of-freedom (DOF) robotic hand. However, there has been very little experimental work reported on whole arm grasping with redundant robot manipulators. Specifically, one of the few results in the literature is given in [5] where whole arm grasping with a 30 DOF robotic arm was demonstrated. Shape control is another technique being studied for the control of whole arm grasps. In [13], the author proposed an impedance control based approach for controlling the shape of the entire arm of a redundant manipulator. Whole arm grasping has a number of useful properties as noted by [1], [12], [18] and others. The authors of [12] point out that distribution of contact points enables

increased load capacity. The ability to use the entire body of the manipulator for grasping also allows objects of various dimensions to be grasped [18]. These capabilities can be used in many applications, including, search and rescue, underwater and space exploration.

Traditional robotic grasping control can be broadly classified into two main categories [15]. The first category, known as the geometrical planning based approach, requires the object model and the constraint forces to be known *a priori* (e.g. [1] and [19]). Here, the grasping contact points are pre-planned and the desired constraint force for each contact point are assumed to be known. The grasping control system then moves the hand/arm along a pre-determined trajectory and force feedback (force sensors on the hand/arm) is used to control the interaction forces. The second category for robot grasping control is the sensory approach, where the object model is unknown and the grasping controller relies on tactile force-feedback data. In this sensory based approach, it is often assumed that the arm has a sensory "skin" for force measurements [2]. The arm/hand must either start off close to the object to be grasped, or with all contact points touching the object. Then, the grasp controller positions and re-positions the arm to minimize an error function in an attempt to optimize the grasp configuration [16].

The techniques described above require either that the geometry of the object and the constraint forces be known *a priori* [19], or that the contact forces be measurable using some type of force sensor [1], [2] and [15]. When extending the traditional approaches (i.e., fingertip grasping) to whole-arm grasping, the previously mentioned requirements might not be easily met due to the increased number of contact points and the large number of grasping configurations possible [16]. Motivated by the need to have a whole arm grasping controller which does not require the constraint forces to be known *a priori* while also eliminating the requirement for contact force sensing (due to the inherent inaccuracy/noise

in measurement), a grasping controller for redundant robot manipulators is designed which requires only the object geometry to be known *a priori*. In addition, the proposed controller does not require the exact dynamic model for the robot manipulator or the contact forces. This paradigm makes the whole arm grasping technique easily extendable to various manipulator systems.

Roughly speaking, the whole arm grasping objective is achieved by integrating the path planner and the controller such that two tasks, robot end-effector positioning and robot body self-motion positioning, are accomplished simultaneously. The end-effector positioning controller forces the end-effector to follow a path around the object which in turn, forces the robot's body to wrap itself around the object to be grasped. The body self-motion positioning controller "repels" the body of the manipulator away from the object while the end-effector moves around the object. This control-induced repulsion-like property facilitates object avoidance as well as removes the "slack" from the robot body as the robot begins to move into the grasping position. When all possible slack is removed, the manipulators body makes contact with the object, hence, completing the whole arm grasp of the object.

To facilitate the explanation of the proposed whole arm grasping control design, we first develop a Lyapunov-based kinematic control. The kinematic control input is then passed through a desired trajectory filter which produces a desired, joint level trajectory. The smooth control strategy developed in [22] is then utilized for the joint space controller since it provides asymptotic tracking of the desired trajectory in the presence of dynamic uncertainties.

Neural Network Grasping Controller for Continuum Robots

Continuum or hyper-redundant robot manipulators are manipulators which exhibit behavior similar to biological trunks, tentacles and snakes [1] and [2]. Unlike traditional rigid link robots, continuum robot manipulators do not have rigid joints, also the increased number of degrees of freedom give the manipulator some very useful properties. The manipulators are flexible, compliant, extremely dexterous and capable of dynamic adaptive manipulation in unstructured environments. Due to these inherent properties of soft continuum robot manipulators, they are uniquely suited to perform whole arm grasping. Whole arm manipulation [18] is the term used to describe the ability of the manipulator to grasp an object using its entire body, as compared to fingertip grasping performed by traditional robotic grippers and hands. Whole arm grasping is performed by allowing the robot manipulator to make contact with the object in a snake or tentacle-like manner, using portions of the manipulator itself to wrap around the object and grasp it. Several researchers including [18], [12] and others have pointed out the advantages of whole arm grasping, some of which are, increased load capacity and the ability to grasp objects of various dimensions. These capabilities can be used in many applications, including search and rescue, underwater and space exploration. The grasping techniques presented in most of the previous work requires either that the geometry of the object and the constraint forces to be known *a priori* [19], or that the contact forces be measurable using some type of force sensor [1], [16].

The development of high performance model based control algorithms for continuum manipulators is a challenging problem for several reasons. For example, since the arms must be modeled as continuous curves, the kinematic and dynamic models are difficult to derive. Also the manipulator body is soft and flexible which makes accurate control

difficult. Several researchers have proposed kinematic control techniques for continuum manipulators, for example see [8], and the references therein. A set-point controller for a variable length manipulator based on an artificial potential function method which does not require the dynamic model was presented in [9]. Various other techniques have been suggested for tracking control of continuum manipulators. In [10], sliding mode and impedance control techniques for hyper-flexible manipulators were presented. In [11], the authors present a trajectory tracking control of snake robots based on its dynamic model. In [12], a fuzzy control method was presented. Shape tracking control, where the manipulator follows a desired shape prescribed by a time-varying spatial curve was considered by [13]. All of the aforementioned techniques but [9] require the dynamic model of the manipulator be known.

In this paper, a path planner is presented for whole arm grasping which does not require the constraint forces to be known *a priori* and also eliminates the requirement for contact force sensing. The path planner is then fused with a low level joint controller that uses a neural network feedforward component to compensate for the unknown dynamics of the continuum robot manipulator. Both the planner and controller are designed such that force measurements are not required. The design of the neural network is based on the augmented back propagation algorithm [14]. Specifically, the neural network is used to compensate for the nonlinear uncertain dynamics of the continuum robot manipulator while a nonlinear feedback controller [22] is used to provide semi-global asymptotic tracking. The advantage of the proposed control scheme compared to previous works is that semi-global asymptotic tracking can be proved, whereas most previous results for neural network control of robot manipulators [16] – [18] only prove ultimate boundedness of the tracking error.

The remainder of the paper is organized as follows, in section II, the kinematic and dynamic models for the continuum robot are presented. In section III, the high level path planning for whole arm grasping is presented. In section IV, the low level joint control objective is defined, and the design of the low level controller with the neural network feedforward component is presented. To demonstrate the performance of the proposed controller with the neural network feedforward component, the controller was tested on the OCTARM, a soft continuum robotic manipulator for a sinusoidal trajectory. Experimental results both with and without the neural network feedforward component are presented in section V to illustrate the effectiveness of the proposed control strategy.

Coordination Control for Haptic and Teleoperator Systems

For the purposes of this research, the following definitions are made. A teleoperator system enables a user to execute a remote task with an output system (i.e., a slave system) operating in a physical environment by manipulating an input system (i.e., a joystick or a master system) while providing feedback on the input system. A haptic system is similar to a teleoperator system with the exception that the slave system operates in a virtual environment. Some common application areas for teleoperator and haptic systems include handling hazardous materials, maneuvering mobile robots, underwater vehicles, and microsurgery in either a physical or a virtual environment. The operator's ability to accurately complete these tasks is affected by the transparency of the teleoperator or haptic system. Tactile and force feedback from the system controller along with assistive mechanisms greatly increase the user's performance in completing the desired task [50]. Tactile and force feedback provides the user of the system with a sense of feel or sense of *telepresence* [71] of

what the slave system is experiencing in either a physical or a virtual environment. Assistive mechanisms can be integrated into the system controller in various ways. One example, which will be discussed further in subsequent sections of this paper, is the encoding of a tracking objective in the master system that assists the user in completing a pre-defined task (i.e., consider a teleoperator grinding application where the remote user controls the slave system to track a repeated circular path to complete the desired task).

Both the teleoperator and/or haptic problem are theoretically challenging due to issues that impact the user's ability to impart a desired motion on the remote environment while maintaining a sense of feel through the system controller. This problem is further complicated due to the fact that master system apparent inertia is normally very different than that of the slave system that is operating in the remote environment, be it physical or virtual. If the apparent inertia of the master system could be adjusted by the system controller to appear like that of the slave systems, the operator's sense of *telepresence* would be achieved, hence, increasing the user's ability to operate the slave system. To address the above control objective, commercially available haptic systems come in two distinct classes: impedance controlled devices, and admittance controlled devices [74]. Both classes have advantages/disadvantages depending on the application, see [50] and [74] for more details.

The focus of some of the previous teleoperator system research has been to achieve ideal transparency between the environment and the user. In [52], Hannaford modeled the teleoperator system as a two-port network where an estimate of the impedance of the slave system is required to achieve transparency. In [48], *a priori* knowledge of the environmental inputs to the slave system is required to achieve the transparency control objective. Controllers aiming at low-frequency transparency were suggested in [51], [56] and [70]. Frequency-based control designs given in [48], [51], [52], [56] and [70] are for linear

teleoperator systems. The concept of the four-channel architecture, which assumes knowledge of system impedances was introduced by the authors of [56] and [76]. To overcome parametric uncertainties, common in teleoperator systems, adaptive controllers were developed in [47], [54], [64], [68], [72] and [77].

Other research has focused on maintaining safe and stable operation of the teleoperator system through passivity concepts. In [44], Anderson and Spong transformed the time delay problem of the teleoperator system into a transmission line problem and presented a controller for the communication circuit that guarantees passivity of the teleoperator system independent of time delay present in the communication block. In [65], Niemeyer and Slotine extended the results in [44], and introduced wave-variables formulation to represent transmission delays, which results in a new configuration for force-reflecting teleoperation. These results were then extended to solve the position tracking problem where [46] and [66] provided a solution when the time delay is constant and [45] provided a solution when the time delay is time-varying. In [59], a passive decomposition for linear dynamically similar systems is introduced. In [57], Lee and Li extended these results to define a nonlinear decomposition which achieves passivity of the master and the slave robots by decomposing the closed-loop teleoperator system into two sub-systems. The reader is referred to [58], [60], and [61] for improvements of passive decomposition. In [62] and [63], Lee suggested a controller for a master and multiple cooperative slave robots over a communication network in the presence of a time delay. In [53], Hannaford and Ryu proposed a passivity based model-insensitive approach that measures the total energy of the system and damps excess energy by injecting a variable damping, which was then extended in [69].

In this paper, the work in [9] is extended so that it is applicable for the control of both teleoperator and haptic systems. Two controllers are developed for nonlinear haptic and teleoperator systems that target coordination of the master and slave. The first controller is proven to yield a semi-global asymptotic result in the presence of parametric uncertainty in the master and slave dynamic models provided the user and environmental input forces are measurable; henceforth, referred to as the MIF, (measurable input force) controller. The second controller yields a global asymptotic result despite unmeasurable user and environmental input forces (UMIF) provided the dynamic models of the master and slave systems are known. This paper differs from [9] in that the transformation and target system development are both modified to allow the master system's impedance, felt by the user, to be adjusted so that it closely matches that of a desired target system operating in a remote environment. This work also provides the encoding of a velocity field assist mechanism to provide the user help in controlling the slave system in completing a pre-defined contour following task. To achieve these control objectives, a continuous nonlinear integral feedback controller/observer (see [67] and [75]) is exploited to compensate for the lack of master and slave dynamics information or user and environmental force measurements. For each controller, Lyapunov-based techniques are used to prove that the controller development implements a stable coordinated haptic/teleoperator system with the optional assist mechanism enabled. When this mechanism is disabled, the subsequent analysis proves the controller development implements a stable passively coordinated haptic/teleoperator system. The passivity objective is motivated to ensure the safety of the user and the environment when in contact with the haptic/teleoperator system.

CHAPTER 2

WHOLE ARM GRASPING CONTROL FOR REDUNDANT ROBOT MANIPULATORS

Kinematic and Dynamic Models

In this section the kinematic and dynamic models for an *n*-joint ($n \geq 6$), revolute, direct drive robot manipulator are presented. The subsequent development is based on these models.

Kinematic Model

Denavit-Hartenberg based forward kinematic model for an *n*-segment redundant manipulator can be developed as follows

$$x_n = f_n(q) \tag{2.1}$$

where $x_n(t) \in \Re^p$ represents the robot end-effector's task-space vector, $q(t) \in \Re^n$ denotes the joint position, and $f_n(q) \in \Re^p$ denotes the forward kinematics of the manipulator. The velocity kinematics for the manipulator can be developed as follows

$$\dot{x}_n = J_n(q)\dot{q}(t) \tag{2.2}$$

where $\dot{x}_n(t) \in \Re^p$ represents the task-space velocity, $\dot{q}(t) \in \Re^n$ denotes the joint velocity, and $J_n(q) \triangleq \frac{\partial f_n(q)}{\partial q} \in \Re^{p \times n}$ denotes the manipulator Jacobian.

Dynamic Model

The dynamic model for an *n*-joint ($n \geq 6$), revolute, direct drive robot manipulator is described by the following expression [6]

$$M(q)\ddot{q} + N(q,\dot{q}) + F_e(q,\dot{q}) = \tau \tag{2.3}$$

where $M(q) \in \Re^{n\times n}$ represents the inertia effects, $N(q,\dot{q}) \in \Re^n$ represents the remaining dynamic terms, such as the centripetal-Coriolis effects, gravitational effects, and frictional effects, $F_e(q,\dot{q}) \in \Re^n$ represents the contact forces placed on the robot manipulator by the environment, $\tau(t) \in \Re^n$ represents the input torque vector. The subsequent development is based on the assumptions that $q(t)$ and $\dot{q}(t)$ are measurable, $M(q)$, $N(q,\dot{q})$ and $F_e(q,\dot{q})$ are unknown, second order differentiable, functions of $q(t)$ and $\dot{q}(t)$, and the following property holds [6],

Property 1: The inertia matrix $M(q)$ is symmetric and positive-definite, and satisfies the following inequalities

$$m_1\|\xi\|^2 \leq \xi^T M(q)\xi \leq m_2\|\xi\|^2 \qquad \forall \xi \in \Re^n \tag{2.4}$$

where $m_1, m_2 \in \Re$ are positive constants, and $\|\cdot\|$ denotes the standard Euclidean norm.

Remark 1: Since this development is only concerned with revolute robot manipulators, the kinematic and dynamic terms denoted by $M(q)$, $N(q,\dot{q})$ *and* $J(q)$*, are assumed to be bounded for all possible* $q(t)$ *(i.e., these kinematic and dynamic terms only depend on* $q(t)$ *as arguments of trigonometric functions).*

Grasping with Kinematic Control

To facilitate the kinematic control development, the pseudo-inverse of $J_n(q)$ denoted by $J_n^+(q) \in \Re^{n \times p}$, is defined as follows

$$J_n^+ \triangleq J_n^T (J_n J_n^T)^{-1} \tag{2.5}$$

where $J_n^+(q)$ satisfies the following equality

$$J_n J_n^+ = I_p \tag{2.6}$$

where $I_p \in \Re^{p \times p}$ is the standard identity matrix. As shown in [14], the pseudo-inverse defined by (2.5) satisfies the Moore-Penrose conditions given below

$$\begin{aligned} J_n J_n^+ J_n &= J_n & J_n^+ J_n J_n^+ &= J_n^+ \\ (J_n^+ J_n)^T &= J_n^+ J_n & (J_n J_n^+)^T &= J_n J_n^+ \end{aligned} \tag{2.7}$$

In addition to the above properties, the matrix $(I_n - J_n^+ J_n)$ satisfies the following useful properties

$$\begin{aligned} (I_n - J_n^+ J_n)(I_n - J_n^+ J_n) &= I_n - J_n^+ J_n \\ (I_n - J_n^+ J_n)^T &= I_n - J_n^+ J_n \\ J_n (I_n - J_n^+ J_n) &= 0 \\ (I_n - J_n^+ J_n) J_n^+ &= 0 \end{aligned} \tag{2.8}$$

where $I_n \in \Re^{n \times n}$ is the standard identity matrix.

Remark 2: During the control development, the assumption is made that the minimum singular value of the manipulator Jacobian, denoted by σ_m *is greater than a known, small positive constant* $\delta > 0$*, such that* $\max\{\|J_n^+(q)\|\}$ *is known a priori and all kinematic singularities are always avoided.*

Typically in the robotics literature, when a kinematic control is designed, $\dot{q}(t)$ is taken to be the control input. A joint space controller must then be used to ensure that the actual robot joint angles track this reference trajectory. Following this paradigm, the kinematic controller is first designed as follows

$$\dot{q}(t) \triangleq J_n^+ U_e + (I_n - J_n^+ J_n) U_m \tag{2.9}$$

where $U_e(t) \in \Re^p$ is the *end-effector positioning* controller, and $U_m(t) \in \Re^n$ is the *robot body self-motion* controller. In the subsequent sub-sections, the design of the robot *end-effector positioning* controller $U_e(t)$ and the *robot body self-motion* controller $U_m(t)$ will be discussed in detail.

End-Effector Positioning

The objective of the *end-effector positioning* controller is to force the end-effector to track a desired trajectory that encompasses the surface of the object to be grasped. For this type of problem, instead of a time based trajectory, a velocity field control (VFC) is utilized because it more effectively penalizes the end-effector for leaving the contour ([4], [7] and [8]). The VFC will also not exhibit the *radial reduction* phenomenon which is common with traditional control methods ([4] and [7]). For example, when the object to be grasped is circular, the velocity field generates a desired trajectory that forces the end-effector to spiral inwards, toward and around the surface of the object.

Remark 3: A velocity field specifies a desired velocity $\dot{x}_d(t)$ *at each displacement position* $x_n(t)$ *on the task space of the system [7]. In [7] and [8], the authors provide specific information about the construction of velocity fields. See [4] and [10] for details of circular velocity fields.*

The *end-effector positioning* controller $U_e(t) \in \Re^p$ is designed as follows

$$U_e \triangleq \vartheta(x_n) + K_e e + k_n \left\| \frac{\partial V(x_d)}{\partial x_d} \right\|^2 \rho^2(x_n, x_d) e \tag{2.10}$$

where $\vartheta(x_n) \in \Re^p$ is a task-space velocity field, $K_e \in \Re^{p \times p}$ is a positive definite diagonal gain matrix, $k_n \in \Re^+$ is a scalar gain parameter, $e(t) \in \Re^p$ is the error between the desired and actual task space position and is defined as follows

$$e \triangleq x_d - x_n, \tag{2.11}$$

where $x_d(t) \in \Re^p$ is the desired task-space position, and $x_n(t)$ was introduced in (2.1). In (2.10), $V(x_d) \in \Re$ is a first order differentiable, nonnegative function and $\rho(.) \in \Re$ is a known positive function that is assumed to be bounded provided $x_n(t)$ and $x_d(t)$ are bounded. For details on how to construct $\rho(x_n, x_d)$ for a specific application, the reader is referred to [10].

For the whole arm grasping control objective, the desired task space velocity trajectory is defined as

$$\dot{x}_d(t) \triangleq \vartheta(x_n) \tag{2.12}$$

where $\vartheta(x_n)$ is the velocity field generated by the task-space position $x_n(t)$. The velocity tracking error signal can be derived by taking the first derivative of (2.11) and using (2.12), we have

$$\dot{e} = \vartheta(x_n) - \dot{x}_n \tag{2.13}$$

After utilizing (2.2), the expression in (2.13) can be written as follows

$$\dot{e} = \vartheta(x_n) - J_n \dot{q} \tag{2.14}$$

After utilizing (2.9), the expression in (2.14) can be written as follows

$$\dot{e} = \vartheta(x_n) - U_e \tag{2.15}$$

where (2.6) and (2.8) has been used. After substituting for $U_e(t)$, as defined in (2.10) can be written as follows

$$\dot{e} = -K_e e - k_n \left\| \frac{\partial V(x_d)}{\partial x_d} \right\|^2 \rho^2(x_n, x_d) e \tag{2.16}$$

Theorem 1: The control law described by (2.10) guarantees that $e(t)$, $\dot{e}(t)$ and $U_e(t) \in \mathsf{L}_\infty$ and $\|e(t)\| \to 0$ as $t \to \infty$.

Proof: See [10], which contains a similar result.

The result of Theorem 1 proves that $\|e(t)\| \to 0$ as $t \to \infty$ and that $U_e(t) \in \mathsf{L}_\infty$. Thus, the control law defined in (2.10) guarantees that the manipulators end-effector follows the desired contour while also ensuring that all signals remain bounded. If the controller defined in (2.10) is used alone (i.e. $\dot{q}(t) = J_n^+ U_e$), the joint space desired trajectory that is tracked may take a path such that the end-effector and body of the manipulator may make contact with the object while the end-effector tries to follow the contour of the object to be grasped. Since this is an undesirable effect, the *robot body self-motion positioning* controller is designed in such a manner that provides object avoidance as the body of the manipulator moves around the object to be grasped.

Body Self-Motion Positioning

The objective of the *body self-motion positioning* controller is to "repel" the end-effector and body of the manipulator away from the object to be grasped, while the end-effector moves around the object. This control-induced repulsion-like property not only facilitates obstacle avoidance but also removes the "slack" from the body of the manipulator as the robot moves into the grasping position. When all possible slack is removed, the manipulator body makes contact with the object, hence completing the whole arm grasp of the object. Following this line of reasoning, the *body self-motion positioning* controller $U_m(t) \in \Re^n$ in (2.10), is designed as follows

$$U_m \triangleq -k_m [J_s (I_n - J_n^+ J_n)]^T y_a \tag{2.17}$$

where $k_m \in \Re^+$ is a control gain, $J_s \in \Re^{1\times n}$ is a subsequently designed Jacobian-like vector, $I_n \in \Re^{n\times b}$ was defined in (2.8), and $y_a(t) \in \Re$ is an auxiliary scalar signal which is yet to be defined. The signal $y_a(t)$ encodes the geometric information about the object's surface and how it relates to the manipulator's joint positions in an effort to keep the body of the manipulator away from the object. See [20] for details of a general auxiliary signal for self-motion control of a redundant robot manipulator.

For whole arm grasping, a specific auxiliary signal $y_a(t)$ is designed as follows

$$y_a \triangleq \sum_{i=1}^{n} h_{ai}(x_i) \tag{2.18}$$

where *n* is the number of joints of the redundant manipulator, $x_i = \left[\bar{x}_{i1}\ \bar{x}_{i2} \ldots \bar{x}_{ip}\right]^T \in \Re^p$ is the Euclidean-space coordinate for the i^{th} joint, and $h_{ai}(x_i) \in \Re$ is the repulsion function for the i^{th} joint that encodes the geometric information about the surface of the object with

respect to the i^{th} joint's Euclidean position. The repulsion function $h_{ai}(x_i)$ is defined as follows

$$h_{ai}(x_i) = k_{hi} \exp(-\alpha_i \beta_i^2(x_i)), \qquad \forall i = 1,...,n \tag{2.19}$$

where k_{hi}, $\alpha_i \in \Re^+$ are constants, and $\beta_i(x_i) \in \Re$ is the joint specific geometric function. The function $\beta_i(x_i)$ should be designed to be positive when the manipulator is not touching the object as well as that $\beta_i(x_i) \in L_\infty$, if $x_i(t) \in L_\infty$. For example, given a spherical object in three dimensional Euclidean-space, $\beta_i(x_i)$ could be defined as follows

$$\beta_i(x_i) \triangleq (\overline{x}_{i1} - \overline{x}_{c1})^2 + (\overline{x}_{i2} - \overline{x}_{c2})^2 + (\overline{x}_{i3} - \overline{x}_{c3})^2 - r_o^2$$

where $\overline{x}_{c1}, \overline{x}_{c2}, \overline{x}_{c3}, r_o \in \Re$ are the Euclidean coordinates of the center of the spherical object and its radius, respectively.

To determine the dynamics of $y_a(t)$, the time derivative of (2.18) is taken, and can be written as follows

$$\dot{y}_a = J_s \dot{q} \tag{2.20}$$

where a Jacobian-type vector $J_s(t) \in \Re^{1 \times n}$ is defined as follows

$$J_s = \frac{\partial y_a(x_1, x_2, ..., x_n)}{\partial [x_1^T, x_2^T, x_3^T]} \begin{bmatrix} J_1 \\ . \\ J_n \end{bmatrix} \tag{2.21}$$

where $\dot{x}_i = J_i \dot{q}$ and $J_i \in \Re^{p \times n}$ is the Jacobian matrix relating the joint velocities and the Euclidean velocities for the i^{th} joint. Using (2.9) and substituting for $\dot{q}(t)$ in (2.20), the expression for $\dot{y}_a(t)$ can be written as follows

$$\dot{y}_a = J_s J_n^+ U_e + J_s (I_n - J_n^+ J_n) U_m \tag{2.22}$$

After substituting for $U_m(t)$ as defined in (2.17), $\dot{y}_a(t)$ of (2.22) can be further expressed as

$$\dot{y}_a = -k_m \left\| J_s (I_n - J_n^+ J_n \right\|^2 y_a + J_s J_n^+ U_e \tag{2.23}$$

Theorem 2: The control law described by (2.17) guarantees that $\dot{y}_a(t)$ is practically regulated (i.e., ultimately bounded) in the following sense

$$|y_a(t)| \le \sqrt{|y_a(t_0)|^2 \exp(-2\mu t) + \frac{\omega}{\mu}} \tag{2.24}$$

provided the following sufficient conditions are true

$$\left\| J_s (I_n - J^+ J \right\|^2 > \bar{\delta} \tag{2.25}$$

and

$$k_m > \frac{1}{\bar{\delta}\delta_2} \tag{2.26}$$

where $\omega, \mu, \bar{\delta}, \delta_2 \in \Re$ are positive constants.

Proof: See [20], which has a similar result.

Remark 4: From (2.18) and (2.19), it is clear that $0 < y_a(t) \le \sum_{i=1}^{n} k_{hi}$ *and that as* $\beta_i(\cdot)$ *increases,* $h_{ai}(t)$ *decreases, and hence,* $y_a(t)$ *decreases. In addition each* $\beta_i(\cdot)$ *is designed such that* $\beta_i(\cdot) > 0$ *if the manipulators links are outside the object. From (2.24), it can be shown that the initial conditions of the manipulator and the bounding constants can be selected such that* $y_a(t) \le \sum_{i=1}^{n} k_{hi}$ *, hence, it is clear from (2.18) and (2.19) that* $\beta_i(t) > 0 \forall t$.

The result of Theorem 2 illustrates that the repulsive term $y_a(t)$ can be bounded by an exponentially decreasing function. This means that when all the manipulators links are in contact with the object the auxiliary repulsion function $y_a(t)$ will approach a constant value

($\sum_{i=1}^{n} k_{hi}$), hence $\beta_i(t) \approx 0$. Interestingly, as the slack in the robot body is removed, the effect of the control term $U_m(t)$ is automatically reduced. This is because as the manipulator links make contact with the object, the number of redundant degrees of freedom available to accomplish the task space objective reduces. As a consequence, the self-motion component of the control input becomes almost zero (i.e., the null space projection $\left\|(I_n - J_n^+ J_n)\right\|$ approaches zero), and hence, (2.25) is no longer satisfied.

Grasping with Dynamic Control

In the previous section, a kinematic control development was presented which enabled the whole arm grasping objective to be encoded as a desired trajectory signal which can be fed to the subsequently designed joint space tracking controller. This desired trajectory signal encodes information from the two auxiliary controllers, the *end-effector positioning* controller, and the *body self-motion positioning* controller. In the subsequent section, the kinematic control will be utilized to generate a bounded desired joint trajectory such that its higher order derivatives are also bounded.

Desired Trajectory Generator

Traditionally for torque based control, the desired trajectory and its higher order derivatives are required for the control implementation. It is assumed that the desired trajectory and its higher order derivatives are always bounded for this problem to be tractable. In this section, a desired trajectory filter which generates bounded desired joint space trajectories for the joint space tracking controller is provided. The structure of the

desired trajectory generator is motivated by the choice of the joint space controller [22], which is a continuous, nonlinear integral feedback controller and requires the desired trajectory to be bounded upto its fourth derivative. This controller was selected because of its ability to meet the tracking objective in the presence of system uncertainties (i.e. uncertainty in the robot dynamic model and unmeasurable contact forces).

To ensure that the desired joint space velocity trajectory is bounded, we could use the following expression

$$\dot{q}_d(t) \triangleq sat \text{ (RHS of (2.9))} \tag{2.27}$$

where RHS denotes the right hand side of the equation, $sat(\xi) \in \Re^n$ is defined as $sat(\xi) = [sat(\xi_1), sat(\xi_2), ..., sat(\xi_n)]^T \; \forall \xi = [\xi_1, \xi_2, ..., \xi_n]^T \in \Re^n$ where $sat(\xi_i) \in \Re$ $\forall i = 1, ..., n$ is the following saturation function

$$sat(\xi_i) = \begin{cases} -\xi_{\min} & if \quad \xi_i \le -\xi_{\min} \\ \xi_i & if \quad \xi_i \le -\xi_{\min} \quad or \quad \xi_i < \xi_{\max} \\ \xi_{\max} & if \quad \xi_i \ge \xi_{\max} \end{cases}$$

where $\xi_{\min}, \xi_{\max} \in \Re^+$ are constants. If (2.27) is used to generate the desired trajectory, we cannot prove that $q_d(t)$ is bounded, so we could use the following filtering operation

$$q_d(s) \triangleq \frac{1}{\left(\frac{s}{\varepsilon} + 1\right)} sat \text{ (RHS of (2.9))} \tag{2.28}$$

where $s \in \mathbb{C}$ is the standard Laplace variable, and $\varepsilon \in \Re^+$ is an integration constant selected very close to zero. However, in the case of (2.28), we cannot prove that the higher order

derivatives of $q_d(t)$ will remain bounded. So the desired trajectory $q_d(t)$ for the manipulator joint angles are generated by the following expression

$$q_d(s) \triangleq \frac{1}{\left(\frac{s}{\varepsilon}+1\right)\left(\frac{s}{\kappa}+1\right)^3} sat \text{ (RHS of (2.9))} \tag{2.29}$$

where $\kappa \in \Re^+$ is an integration constant selected to be very large. From (2.29), it is clear that $q_d(t), \dot{q}_d(t), \ddot{q}_d(t), \dddot{q}_d(t)$, and $\ddddot{q}_d(t) \in L_\infty$.

Control Objective

The objective of the closed-loop system is to ensure asymptotic tracking between the manipulator and the desired trajectory in the sense that

$$q(t) \to q_d(t) \text{ as } t \to \infty \tag{2.30}$$

where $q_d(t) \in \Re^n$ is obtained from (2.29). To quantify the control objective, an error signal $e_1(t) \in \Re^n$ is defined as follows

$$e_1 \triangleq q_d - q \tag{2.31}$$

Furthermore, a tracking error signal $e_2(t) \in \Re^n$ is defined as follows

$$e_2 \triangleq \dot{e}_1 + \gamma_1 e_1 \tag{2.32}$$

where $\gamma_1 \in \Re^+$ is a control gain.

Control Law

Since the robot dynamic model is a nonlinear uncertain multi-input multi-output system, the strategy developed in [22] can be used for the continuous joint space controller. The control objective of (2.30) can be met with the following controller [22]

$$\tau \triangleq (K_s + I_n)\left[e_2(t) - e_2(t_0) + \gamma_2 \int_{t_0}^{t} e_2(\tau) d\tau \right] + \int_{t_0}^{t} \left[\Gamma \operatorname{sgn}(e_2(\tau)) \right] \, d\tau \qquad (2.33)$$

where $\tau(t) \in \Re^n$ is the control input defined in (2.3), $K_s, \Gamma \in \Re^{n \times n}$ are positive diagonal control gain matrices, and $\operatorname{sgn}(\cdot) \in \Re^n$ denotes the vector signum function defined as $\operatorname{sgn}(\xi) = [\operatorname{sgn}(\xi_1), \operatorname{sgn}(\xi_2), ..., \operatorname{sgn}(\xi_n)]^T \quad \forall \xi = [\xi_1, \xi_2, ..., \xi_n]^T \in \Re^n$. The controller presented in (2.33), provides asymptotic convergence of the joint tracking error, i.e. $\|e_1(t)\| \to 0$ as $t \to \infty$. For a detailed analysis of the controller the reader is referred to [22].

Remark 5: The trajectory generator defined in (2.29) generates a filtered version of (2.9). This filtered signal is used as a desired trajectory for the joint space controller defined in (2.33). The joint space controller (2.33), forces the actual robot joint angles to track the filtered desired trajectory of (2.29). However, we cannot show that the actual robot joint velocities track the kinematic velocity signal defined in (2.9). Thus, the results of Theorem 1 and Theorem 2 may not be technically valid, however, the validity of the approach is illustrated through experimental results which will be presented in a future paper.

Experimental Results

The proposed controller was implemented on three links of the Barrett whole arm manipulator (WAM). The WAM is a seven degree of freedom (d.o.f.), highly dexterous and back-drivable robotic manipulator. To simplify the controller implantation, four joints of the

robot were locked at fixed angles and the remaining links of the manipulator were used as a three d.o.f. planar robot manipulator. Figure 2.1, shows the experimental setup for a circular object to be grasped.

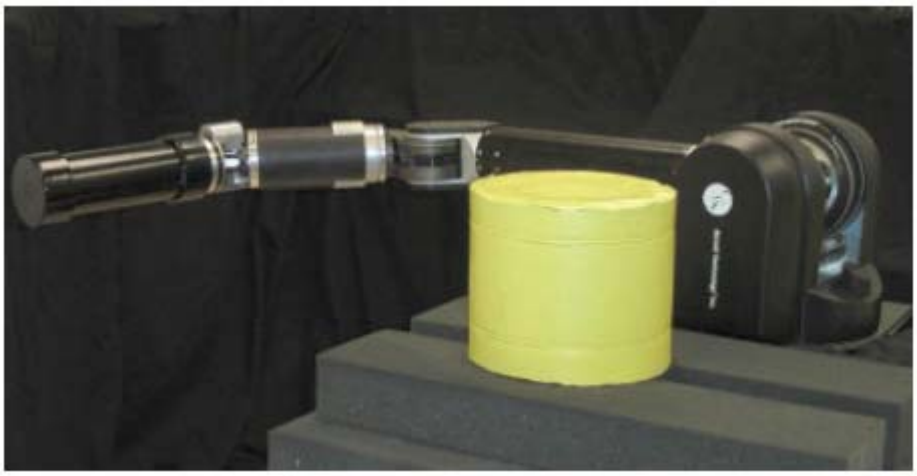

Figure 2.1: Experiment setup showing the Barrett Whole Arm Manipulator and object to be grasped

The control algorithm was written in "C++" and hosted on an AMD Athlon 1.2 GHz PC running the QNX 6.2.1 real-time operating system. Data logging and on line gain tuning was performed using Qmotor 3.0 control software [9]. Data acquisition and control implementation was performed at a frequency of 1.0 [kHz] using the ServoToGo I/O board. Joint positions were measured using the optical encoders located at the motor shaft of each axis. Joint velocity measurements were obtained using a filtered backwards difference algorithm.

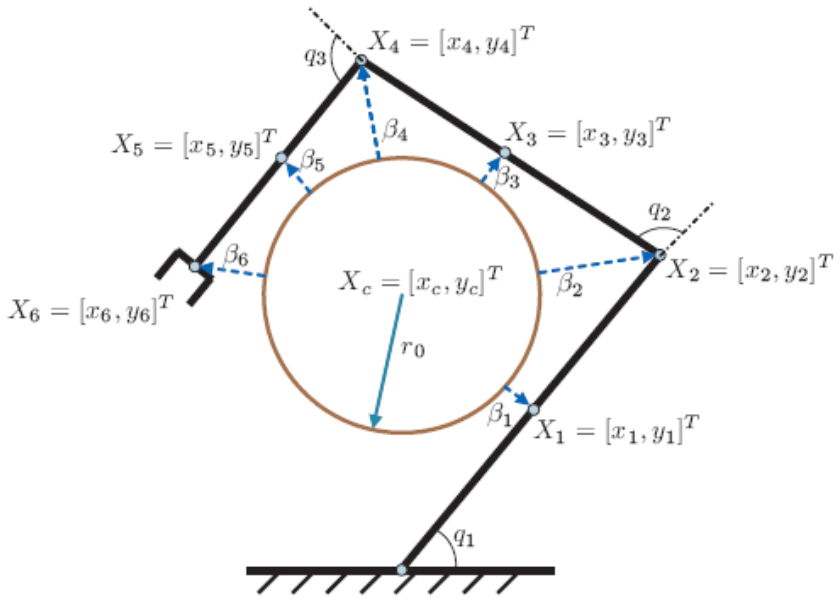

Figure 2.2: Planar configuration for the three link robot with a circular object

Refer to Figure 2.2 for explanation of the notions used in this section. $X_c = [x_c, y_c]^T \in \Re^2$ represents the co-ordinates of the center of the object and $r_o \in \Re$ represents the object radius. We define the task space variable for each of the three links as $X_i = [x_i, y_i]^T \in \Re^2 \quad \forall i = 1,2,...,6.$ The joint angles for the three links are represented by $q = [q_1, q_2, q_3]^T \in \Re^3$. The object specific functions defined for each of the three links and the mid-points of the three links are defined as $\beta_1(\cdot), \beta_2(\cdot), \beta_3(\cdot) \in \Re.$

The object specific functions for this planar application were defined as follows

$$\beta_i(X_i) \triangleq (x_i - x_c)^2 + (y_i - y_c)^2 - r_o^2 \quad \forall i = 1,...,6. \tag{2.34}$$

The following task-space velocity field for a planar, circular contour was utilized [4]

$$\dot{X}_d = \vartheta(X_6) = -2K(X_6) f(X_6) \begin{bmatrix} (x_6 - x_c) \\ (y_6 - y_c) \end{bmatrix} + 2c(X_6) \begin{bmatrix} -(y_6 - y_c) \\ (x_6 - x_c) \end{bmatrix} \tag{2.35}$$

where the functions $f(X_6)$, $K(X_6)$, and $c(X_6) \in \Re$ are defined according to [4] as follows

$$\begin{aligned} f(X_6) &= (x_6 - x_c)^2 + (y_6 - y_c)^2 - r_o^2 \\ K(X_6) &= \frac{k_0}{\sqrt{f^2(X_6)}\left\| \frac{\partial f(X_6)}{\partial X_6} \right\| + \varepsilon_0} \\ c(X_6) &= \frac{c_0 \exp\left(-\mu_0 \sqrt{f^2(X_6)}\right)}{\left\| \frac{\partial f(X_6)}{\partial X_6} \right\|} \end{aligned} \tag{2.36}$$

In (2.36), the constant parameters were selected as $\varepsilon_0 = 0.005$ [m^3], $\mu_0 = 20$ [m^{-1}], $k_0 = 0.1$ [ms^{-1}], and $c_0 = 0.1$ [ms^{-1}]. The desired position for the end-effector is $X_d = [x_d, y_d]^T \in \Re^2$. The controller defined in (2.10) was implemented with $e = X_d - X_6$, $V(X_d) \triangleq 4\|X_d\|^2$, and $\rho(\cdot) = 1$.

The initial joint angles were $q_1(0)$ = 98[deg], $q_2(0)$ = 45.8[deg], $q_3(0)$ = 31[deg], which corresponds to a position of $x_6(0)$ = 0.368 [m] $y_6(0)$ = -0.883 [m] for the end-effector in the task space. The position of the object center in the task space was found to be x_c = 0.307 [m], y_c = -0.117 [m], the radius of the circular object was found to be 0.12 [m]. To take into account the width of the manipulator arm, the radius of the object was set to r_0 = 0.16 [m] in the implementation of the repulsion function. The control gains were selected as follows

$$\begin{aligned} K_e = diag\{800,800\}, k_n = 1, k_m = 100, \\ k_h = \{0.001,0.01,0.01,0.05,8.5,8\}, \\ \alpha_i = \{3,3,3,3,5,5\}, K_s = diag\{16,9,6\}, \\ \beta = diag\{5,5,2\}, \gamma_1 = diag\{1,1,1\}, \\ \gamma_2 = diag\{2,2,2\}, \varepsilon = 0.01, \kappa = 500 \end{aligned}$$

Since the desired trajectory for the end-effector is a velocity field, it will continuously generate the trajectory. To stop the desired trajectory generation when all the links of the manipulator make contact with the object, the norm of the following vector $\beta(\cdot) = [\beta_1(\cdot), \beta_2(\cdot), ..., \beta_6(\cdot)] \in \Re^6$ was used. As the links of the manipulator move closer to the object boundary, $\|\beta(\cdot)\|$ approaches zero, and this gives an estimate of how close the manipulators links are to the object. For the experiment, we stop the trajectory generation by setting $\dot{X}_d = 0$ when $\|\beta(\cdot)\| \leq \eta_0$, where the constant $\eta_0 = 0.01$ was determined experimentally.

Remark 6: The value of $\|\beta(\cdot)\|$ at which we stop the generation of the desired trajectory is specific to a particular grasp configuration. It will change if the object is re- positioned in the task space. However, if we use a highly redundant robot arm which can wrap its entire body around the object, then $\|\beta(\cdot)\|$ will approach zero when the arm grasps the object, then $\|\beta(\cdot)\|$ will approach zero when the arm grasps the object, since the entire body of the arm will be in contact with the object.

Figure 2.3 shows the actual and desired joint angles. Figure 2.4 and Figure 2.5 show the joint space tracking error and the joint control torques respectively. Figure 2.6 shows the spatial position of each of the links and their mid-points as defined in Figure 2.2.

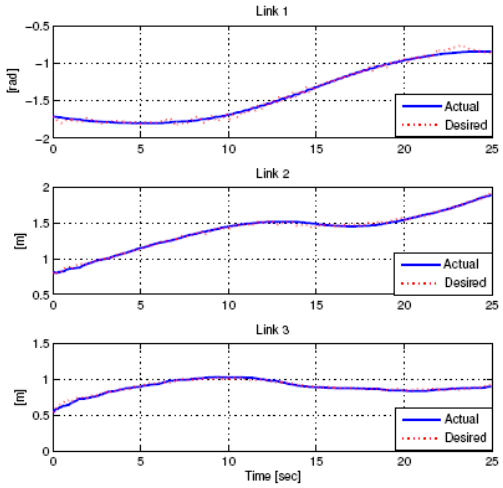

Figure 2.3: Desired joint angles $q_d(t)$ and actual joint angles $q(t)$

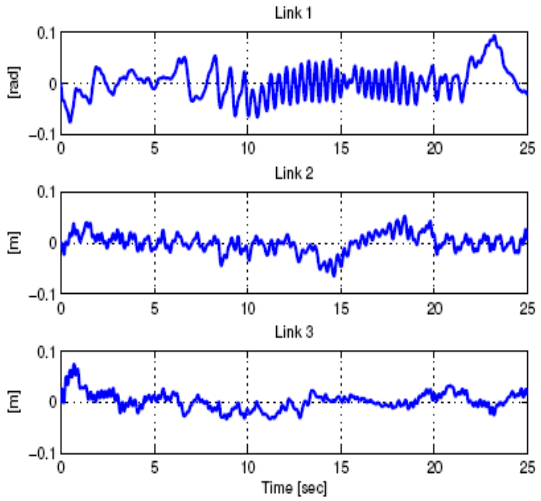

Figure 2.4: Joint space position tracking error $e_1(t)$

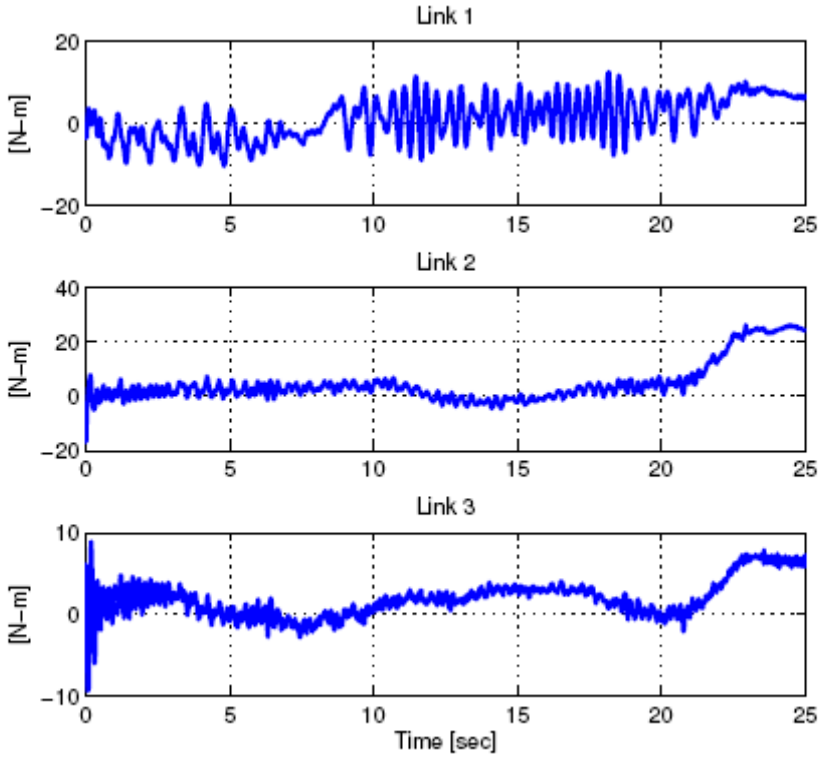

Figure 2.5: Joint space control torques $\tau(t)$

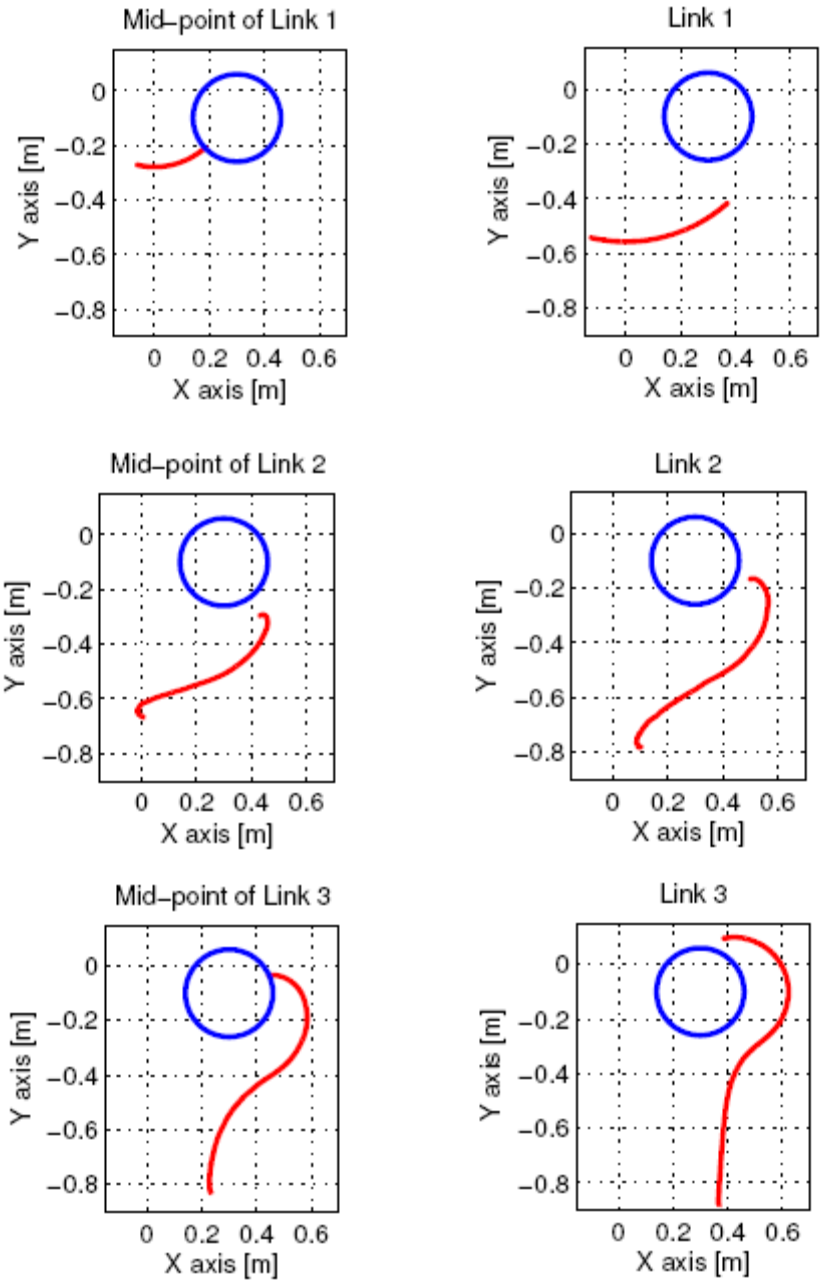

Figure 2.6: Spatial position $X_i \quad \forall i = 1,...,6$ (each link and mid-point of the link)

CHAPTER 3

NEURAL NETWORK GRASPING CONTROLLER FOR CONTINUUM ROBOTS

System Models

Kinematic Model

The forward kinematic model for an *n*-segment continuum robot can be developed as follows

$$x_n = f_n(q) \tag{3.1}$$

where $x_n(t) \in \Re^p$ represents the robot end-effector's task-space vector, $q(t) \in \Re^n$ denotes the joint position, and $f_n(q) \in \Re^p$ denotes the forward kinematics of the robot. Note for continuum robots, $q(t)$ is a vector of curvatures and extensions of the robot sections, for more information see [37]. The velocity kinematics for the robot can be developed as follows

$$\dot{x}_n = J_n(q)\dot{q}(t) \tag{3.2}$$

where $\dot{x}_n(t) \in \Re^p$ represents the task-space velocity, $\dot{q}(t) \in \Re^n$ denotes the joint velocity, and $J_n(q) \triangleq \dfrac{\partial f_n(q)}{\partial q} \in \Re^{p \times n}$ denotes the robot Jacobian. Recently in [37], the authors have developed a general method for determining the kinematics of continuum robots. This approach enables the Cartesian position and orientation of the end effector and the robot Jacobian to be computed in real-time.

Assumption 1: The kinematic model for the continuum robot is known, and all kinematic singularities are avoided such that J(q) is always non-singular.

Dynamic Model

From a review of the literature, it is evident that there have been few published results pertaining to the dynamic modeling of continuum robot arms. Some of the dynamic models which have been developed include [38], [39], where the planar model of the manipulator was considered, and [40], where the authors develop a 3D dynamic model for a constant length, non-extensible continuum manipulator. As such, the complete dynamic modeling of variable length continuum robot arms remains an open research area. In [40], the developed dynamic model was shown to satisfy the familiar property that the continuum manipulators inertia matrix is symmetric and positive definite. In this development, we will assume that the dynamic model of a 9-joint continuum robot manipulator can be described by the following expression

$$M(q)\ddot{q} + N(q,\dot{q}) = u \tag{3.3}$$

where $M(q) \in \Re^{9\times9}$ represents the inertia matrix, $N(q,\dot{q}) \in \Re^{9}$ represents the remaining dynamic terms, $u(t) \in \Re^{9}$ represents the input torque vector, and $q(t), \dot{q}(t), \ddot{q}(t) \in \Re^{9}$ represent the joint position, velocity and acceleration respectively.

The subsequent development is based on the following assumptions

Assumption 2: The manipulators joint position $q(t)$ *and joint velocity* $\dot{q}(t)$ *are measurable.*

Assumption 3: The dynamic terms denoted by M(q) and $N(q,\dot{q})$ *are unknown nonlinear functions of q(t) and* $\dot{q}(t)$ *which are second order differentiable and satisfy the following properties*

$$M(\cdot), \dot{M}(\cdot), \ddot{M}(\cdot) \in \mathcal{L}_\infty \;\; if \;\; q(t), \dot{q}(t), \ddot{q}(t) \in \mathcal{L}_\infty \tag{3.4}$$

$$N(\cdot), \dot{N}(\cdot), \ddot{N}(\cdot) \in \mathcal{L}_\infty \; if \;\; q(t), \dot{q}(t), \ddot{q}(t), \dddot{q}(t) \in \mathcal{L}_\infty \tag{3.5}$$

Assumption 4: The inertia matrix M(q) is symmetric and positive-definite, and satisfies the following inequalities

$$m_1 \|\xi\|^2 \le \xi^T M(q) \xi \le m_2 \|\xi\|^2 \quad \forall \xi \in \Re^9 \tag{3.6}$$

where $m_1, m_2 \in \Re$ *are positive constants, and* $\|\cdot\|$ *denotes the standard Euclidean norm.*

High Level Path Planning

Whole arm grasping can be achieved by integrating the path planner and the controller such that two tasks, robot *end-effector positioning* and robot *body self-motion positioning*, are accomplished simultaneously [3]. The *end-effector positioning* controller forces the end-effector to follow a path around the object which in turn, forces the robot's body to wrap itself around the object to be grasped. The *body self-motion positioning* controller "repels" the body of the manipulator away from the object while the end-effector moves around the object.

Kinematic Planning

To facilitate the kinematic planning, the pseudo-inverse of the manipulator Jacobian denoted by $J_n^+(q) \in \Re^{n \times p}$ is defined as follows

$$J_n^+ \triangleq J_n^T (J_n J_n^T)^{-1} \tag{3.7}$$

where $J_n^+(q)$ satisfies the following equality

$$J_n J_n^+ = I_p \tag{3.8}$$

where $I_p \in \Re^{p\times p}$ is the standard identity matrix. As shown in [14], the pseudo-inverse defined by (3.7) satisfies the Moore-Penrose conditions given below

$$J_n J_n^+ J_n = J_n \qquad J_n^+ J_n J_n^+ = J_n^+$$

$$(J_n^+ J_n)^T = J_n^+ J_n \qquad (J_n J_n^+)^T = J_n J_n^+ \tag{3.9}$$

In addition to the above properties, the matrix $(I_n - J_n^+ J_n)$ satisfies the following useful properties

$$(I_n - J_n^+ J_n)(I_n - J_n^+ J_n) = I_n - J_n^+ J_n$$

$$(I_n - J_n^+ J_n)^T = I_n - J_n^+ J_n$$

$$J_n(I_n - J_n^+ J_n) = 0 \tag{3.10}$$

$$(I_n - J_n^+ J_n)J_n^+ = 0$$

where $I_n \in \Re^{n\times n}$ is the standard identity matrix.

Based on (3.2) and the above properties the kinematic planner, denoted by $U(t) \in \Re^n$, that enables the whole arm grasping objective is designed as follows

$$U(t) \triangleq J_n^+ U_e + (I_n - J_n^+ J_n)U_m \tag{3.11}$$

where $U_e(t) \in \Re^p$ is the *end-effector positioning* controller, and $U_m(t) \in \Re^n$ is the robot *body self-motion* controller. The objective of the *end-effector positioning* controller is to force the end-effector to track a desired trajectory encompassing the surface of the object to be grasped. We will utilize a task space velocity field, [8], for the end-effector positioning because it more effectively penalizes the end-effector for leaving the contour and does not exhibit the radial reduction phenomenon [8], [4]. For example, when the object to be grasped is circular, the

velocity field can be designed to generate a desired trajectory which forces the end-effector to spiral inwards, toward, and around the surface of the object.

The *end-effector positioning* controller is designed [3] as

$$U_e \triangleq \mathcal{G}(x_n) + K_e e + k_n \left\| \frac{\partial V(x_d)}{\partial x_d} \right\|^2 \rho^2(x_n, x_d) e \tag{3.12}$$

where $\mathcal{G}(x_n) \in \Re^p$ is a task-space velocity field which encircles the object to be grasped, $K_e \in \Re^{p \times p}$ is a positive definite diagonal gain matrix, $k_n \in \Re^+$ is a scalar gain parameter, $e(t) \in \Re^p$ is the error between the desired and actual task space position and is defined as follows

$$e \triangleq x_d - x_n, \tag{3.13}$$

where $x_d(t) \in \Re^p$ is the desired task-space position, and $x_n(t)$ was introduced in (3.1). In (3.12), $V(x_d) \in \Re$ is a first order differentiable, nonnegative function, and $\rho(\cdot) \in \Re$ is a known positive function that is assumed to be bounded provided $x_n(t)$ and $x_d(t)$ are bounded. The desired task space position $x_n(t)$. Refer to [3] for more details of this development.

The objective of the *body self-motion positioning* controller is to "repel" the end-effector and body of the manipulator away from the object to be grasped, while the end-effector moves around the object. This repulsion-like property facilitates obstacle avoidance and removes the "slack" from the body of the manipulator as the robot moves into the grasping position. Following this line of reasoning, the *body self-motion positioning* controller $U_m(t) \in \Re^n$ of (3.12) is designed [3] as follows

$$U_m \triangleq -k_m [J_s (I_n - J_n^+ J_n)]^T y_a \tag{3.15}$$

where $k_m \in \Re^+$ is a control gain, $J_s \in \Re^{1\times n}$ is a Jacobian-like vector, $I_n \in \Re^{n\times n}$ is the standard identity matrix, and $y_a(t) \in \Re$ is an auxiliary scalar signal encoding geometric information about the object's surface and it's relationship to the manipulator joint positions. This type of geometric encoding keeps the body of the manipulator away from the object during the initial phases of grasping.

For whole arm grasping, a specific auxiliary signal $y_a(t)$ is designed as follows

$$y_a \triangleq \sum_{i=1}^{m} h_{ai}(x_i) \tag{3.16}$$

where m is the number of sections of the redundant manipulator, $x_i = \left[\bar{x}_{i1}\ \bar{x}_{i2} \ldots \bar{x}_{ip}\right]^T \in \Re^p$ is the Euclidean-space coordinate for the i^{th} joint, and $h_{ai}(x_i) \in \Re$ is the repulsion function for the i^{th} joint that encodes the geometric information about the surface of the object with respect to the i^{th} joint's Euclidean position. The repulsion function $h_{ai}(x_i)$ is defined as follows

$$h_{ai}(x_i) = k_{hi} \exp\left(-\alpha_i \beta_i^2(x_i)\right), \qquad \forall i = 1,\ldots,m \tag{3.17}$$

where k_{hi}, $\alpha_i \in \Re^+$ are constants, and $\beta_i(x_i) \in \Re$ is the joint specific geometric function. The function $\beta_i(x_i)$ should be designed to be positive when the manipulator is not touching the object. For example, given a spherical object in three dimensional Euclidean-space, $\beta_i(x_i)$ could be defined as follows

$$\beta_i(x_i) \triangleq (\bar{x}_{i1} - \bar{x}_{c1})^2 + (\bar{x}_{i2} - \bar{x}_{c2})^2 + (\bar{x}_{i3} - \bar{x}_{c3})^2 - r_o^2$$

where $\bar{x}_{c1}, \bar{x}_{c2}, \bar{x}_{c3}, r_o \in \Re$ are the Euclidean coordinates of the center of the spherical object and its radius, respectively. For more details of this development refer to [3].

Fusing the Planner with the Controller

To fuse the high level path planner with the low level joint controller, we use a desired trajectory generator which ensures that the desired trajectories for the manipulator's joints are bounded. The structure of the desired trajectory generator is motivated by the choice of the low level controller (3.25), which requires the desired trajectory be bounded up to its fourth derivative.

To ensure that the desired joint space velocity trajectory is bounded, we could use the following expression

$$\dot{q}_d(t) \triangleq sat(U(t)) \tag{3.18}$$

where $U(t)$ was defined in (3.11), $\dot{q}_d(t) \in \Re^9$ are the desired joint velocities, $sat(\xi) \in \Re^n$ is defined as $sat(\xi) = \left[sat(\xi_1),...,sat(\xi_n)\right]^T \quad \forall \xi = \left[\xi_1,...,\xi_n\right]^T \in \Re^n$ where $sat(\xi_i) \in \Re \quad \forall i = 1,...,n$ is the following saturation function

$$sat(\xi_i) = \begin{cases} -\xi_{\min} & if \quad \xi_i \le -\xi_{\min} \\ \xi_i & if \quad \xi_i \le -\xi_{\min} \quad or \quad \xi_i < \xi_{\max} \\ \xi_{\max} & if \quad \xi_i \ge \xi_{\max} \end{cases}$$

where $\xi_{\min}, \xi_{\max} \in \Re^+$ are constants. If (3.18) is used to generate the desired trajectory, we cannot prove that the desired joint trajectory, $q_d(t) \in \Re^9$ is bounded, so we could use the following filtering operation

$$q_d(s) \triangleq \frac{1}{\left(\frac{s}{\varepsilon}+1\right)} sat(U(t)) \tag{3.19}$$

where $s \in \mathbb{C}$ is the standard Laplace variable, and $\varepsilon \in \Re^+$ is a small constant. However, in the case of (3.19), we cannot prove that the higher order derivatives of $q_d(t)$ will remain bounded. So the desired trajectory generated for $q_d(t)$ is modified further in the final form given by

$$q_d(s) \triangleq \frac{1}{\left(\frac{s}{\varepsilon}+1\right)\left(\frac{s}{\kappa}+1\right)^3} sat(U(t)) \tag{3.20}$$

where $\kappa \in \Re^+$ is large constant. From (3.20), it is clear that $q_d(t), \dot{q}_d(t), \ddot{q}_d(t), \dddot{q}_d(t),$ and $q_d(t) \in \mathsf{L}_\infty$.

Low Level Joint Control Objective

The low level control objective is to design a continuous controller which provides asymptotic tracking of the manipulator joint position and the desired trajectory in the sense that

$$q(t) \to q_d(t) \text{ as } t \to \infty. \tag{3.21}$$

To quantify the control objective, an error signal, denoted by $e_1(t) \in \Re^9$, is defined as follows

$$e_1 \triangleq q_d - q \tag{3.22}$$

Furthermore, an auxiliary tracking error signal $e_2(t) \in \Re^9$, is defined as follows

$$e_2(t) \triangleq \dot{e}_1 + \lambda_1 e_1 \tag{3.23}$$

where $\lambda_1 \in \Re^+$ is a control gain. For the closed loop error system development, we define a filtered tracking error signal $r(t) \in \Re^9$ as follows

$$r \triangleq \dot{e}_2 + \lambda_2 e_2 \tag{3.24}$$

where $\lambda_2 \in \Re^+$ is a control gain.

The dynamic model of the continuum robot is a nonlinear uncertain system; hence, the strategy developed by Xian et. al. [22] can be used for the joint level controller. This controller is chosen because it is continuous, it does not require the dynamic model of the manipulator or contact forces to be known and yet it provides semi-global asymptotic tracking. Specifically, the control objective described in (3.21) can be met with the following controller [22]

$$\begin{aligned} u(t) \triangleq (K_s + I)e_2(t) - (K_s + I)e_2(t_0) + \int_{t_0}^{t} \hat{f}(\tau)d\tau \\ + \int_{t_0}^{t} (\lambda_2(K_s + I)e_2(\tau) + \beta \operatorname{sgn}(e_2(\tau)))d\tau \end{aligned} \tag{3.25}$$

where $u(t) \in \Re^9$ is the control input defined in (3.3), $\lambda_2 \in \Re^+$ is a control gain $K_s, \beta \in \Re^{9\times 9}$ are positive definite diagonal control gain matrices, $\hat{f}(t) \in \Re^9$ is the neural network feedforward component and $\operatorname{sgn}(\cdot) : \Re^9 \mapsto \Re^9$ denotes the vector signum function defined as $\operatorname{sgn}(\xi) = \left[\operatorname{sgn}(\xi_1),...,\operatorname{sgn}(\xi_9)\right]^T \quad \forall \xi = \left[\xi 1,...,\xi 9\right]^T \in \Re^9$. The controller presented in (3.25), provides semi-global asymptotic convergence of the joint position tracking error, (i.e. $\|e1(t)\| \to 0$ as $t \to \infty$). For a detailed analysis of the controller the reader is referred to [22].

Remark 1: The design of the neural network feedforward component, $\hat{f}(t)$, is presented in the subsequent section. The only restriction imposed on the neural network component by the selection of the controller (3.25) is that $\hat{f}(t) \in \mathcal{L}_\infty$.

Neural Network Feedforward Design

The neural network feedforward component $\hat{f}(t) \in \Re^9$ is computed using a two layer network with *15* neurons. The number of neurons required for the system was determined experimentally by noting the performance achievable with a given number of neurons and increasing the number of neurons until satisfactory tracking performance was obtained. The weights are computed using a modified version of the back propagation algorithm presented in [33]. Given Remark 1, an important consideration regarding the design of the neural network feedforward component is that the output from the neural network must always be bounded (i.e. $\hat{f}(t) \in \mathcal{L}_\infty$). To this end the neural network component is defined as follows

$$\hat{f} = \hat{W}^T \bar{\sigma}\left(\hat{V}^T x\right) \tag{3.26}$$

where $\hat{W}(t) \in \Re^{15\times 9}$ and $\hat{V}(t) \in \Re^{37\times 15}$ are estimated weight matrices, and $x(t) \in \Re^{37}$ is the input vector to the neural network which is selected as

$$x = [1,\ q_d^T,\ \dot{q}_d^T,\ \ddot{q}_d^T,\ \ddot{\ddot{q}}_d^T]^T \tag{3.27}$$

where $q_d(t), \dot{q}_d(t), \ddot{q}_d(t), \ddot{\ddot{q}}_d(t)$ were previously defined. The vector activation function $\bar{\sigma}(\cdot) \in \Re^{15} \mapsto \Re^{15}$ is defined as follows

$$\bar{\sigma}(\omega) = [\sigma(\omega_1), \sigma(\omega_2), ..., \sigma(\omega_{15})]^T \tag{3.28}$$

is the sigmoid activation function defined as

$$\sigma(s) = \frac{1}{1+\exp(-s)} \tag{3.29}$$

The gradient of the vector activation function, denoted by $\bar{\sigma}'(\cdot) \in \Re^{15\times 15}$ can be expressed in closed form as follows, [33]

$$\bar{\sigma}(\omega)' = diag\{\bar{\sigma}(\omega)\}[I - diag\{\bar{\sigma}(\omega)\}] \tag{3.30}$$

If we were to design the weight update laws according to the augmented backpropagation algorithm [33], we would use the following update rule

$$\dot{\hat{W}} = -\kappa F \|r\| \hat{W} - F\bar{\sigma}'(\cdot)\hat{V}^T x r^T + F\bar{\sigma}(\cdot) r^T$$

$$\dot{\hat{V}} = -\kappa G \|r\| \hat{W} + Gx\left(\bar{\sigma}'^T(\cdot)\hat{W}r\right)^T$$

where $\kappa \in \Re^+$ is selected to be a small constant, $F \in \Re^{15\times 15}, G \in \Re^{37\times 37}$ are positive definite gain matrices, $x(t)$ is the input vector defined in (3.27), and $r(t)$ is the filtered tracking error signal defined in (3.24). Here, the filtered tracking error signal $r(t)$ is required in the update laws which requires the measurement of the manipulator joint acceleration, and hence, this is undesirable. To ensure that the weights generated from this law are bounded, and that joint acceleration measurements are not required, we redefine the update laws as follows

$$\dot{\hat{W}} = -\alpha_\omega \hat{W} + \gamma_1 \bar{\sigma}\left(\hat{V}^T x\right) sat\left(e_2 + \zeta\right)^T \tag{3.31}$$

$$\dot{\hat{V}} = -\alpha_\upsilon \hat{V} + \gamma_2 x\left[\bar{\sigma}'\left(\hat{V}^T x\right)\hat{W} sat\left(e2 + \zeta\right)\right]^T \tag{3.32}$$

where $\alpha_\upsilon, \alpha_\omega \in \Re^+$ are small constants, $\gamma_1, \gamma_2 \in \Re^+$ are control gains which effect the learning speed, the function $sat(\xi) \ : \ \Re^{15} \mapsto \Re^{15}$ was previously defined, and the auxiliary

signal $\zeta(t) \in \Re^9$ is a surrogate (i.e. a dirty derivative operation) for the signal $\dot{e}_2(t)$ which is defined as follows

$$\zeta = \frac{1}{\varepsilon}(e2 - \eta) \tag{3.33}$$

where $\varepsilon \in \Re^+$ is a small constant, and the signal $\eta(t) \in \Re^9$ is updated according to the following expression

$$\dot{\eta} = \frac{1}{\varepsilon}(e2 - \eta) \tag{3.34}$$

From equations (3.26)-(3.34) and the fact that the input vector to the neural network is bounded, it is easy to show that the weight matrices $\hat{W}(t)$ and $\hat{V}(t)$ are bounded, and hence, the output from the neural network, $\hat{f}(t)$, is bounded.

Experimental Results

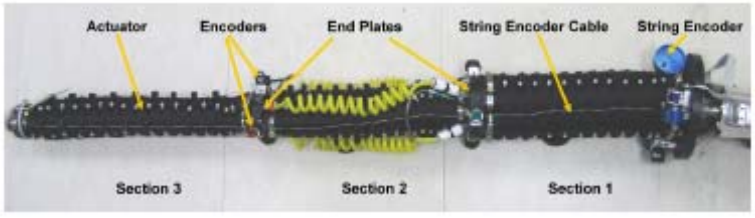

Figure 3.1: The OCTARM V.2 robotic manipulator

To verify the performance of the controller with the neural network feedforward component, the controller was implemented on the OCTARM continuum robot manipulator. In this section, we first provide a description of the OCTARM continuum robot manipulator, then experimental results are provided which demonstrate the

effectiveness of the neural network feedforward tracking controller. Research work for the OCTARM robotic manipulators is being conducted for the Soft Robot Manipulators and Manipulations project supported by the DARPA BIODYNOTICS program. This work is a multidisciplinary and multi-institutional effort, the reader is referred to [26], [41], [42], for more details of the project. The team members at Penn State University perform the mechanical design and construction of the arms while the team at Clemson University develops the electronics, kinematics and control systems.

Description of the OCTARM Manipulator

The OCTARM manipulator [42], [43] is a biologically inspired soft robot manipulator resembling a trunk or tentacle. The OCTARM is significantly more versatile and adaptable than conventional robotic manipulators, capable of adaptive and dynamic manipulation in unstructured environments. To provide the desired dexterity the OCTARM is constructed with high strain extensor air muscles called McKibben actuators, which are constructed by covering latex tubing with a double helical weave, plastic mesh sheath [43]. These actuators provide the large strength to weight ratio and strain required for soft robot manipulators.

The OCTARM is divided into three sections with each section capable of two axis bending and extension allowing nine total degrees of freedom. The arm is pneumatically actuated with a maximum pressure of 130 Psi through nine pressure control valves. To provide two-axis bending and extension, three control channels are selected for each section. Six actuators are used in each section one and two and three actuators are used in section three. The six actuator design has two actuators for each control channel and results in actuators located at a larger radius, corresponding to higher stiffness and load capacity.

Three closely-spaced actuators provide high curvature for the distal section. To provide torsional motion, the OCTARM has been fitted with a D.C. motor at the base. In OCTARM V.2 (see Figure 3.1), the D.C. motor is directly coupled to the end plate of the base section through a reduction gear mechanism. This arrangement of the base motor limits the maximum rotation to 180 degrees.

For closed loop control of the OCTARM manipulator accurate shape sensing is essential. The shape of the manipulator can be inferred by measuring the length of each of the actuators on the OCTARM. To measure the length of each actuator, three string encoders are used. Figure 3.1 shows the three section OCTARM V.2 with the string encoders attached to the base of section 1 and optical encoders located at the end plates of section 1 and 2. The cables from each of the string encoders run the entire length of the arm through the optical encoders at the lower sections, as seen in Figure 3.1. This configuration enables the length of each of the actuators on the OCTARM manipulator to be determined. To obtain actuator velocity measurements, a variable structure velocity observer is utilized (see [22]).

The design of these manipulators is constantly being refined to provide stronger actuators, additional sensory information, newer capabilities and eliminate some of the problems with the previous designs. With the OCTARM V.2, due to the arrangement of the string encoders at the base section and the optical encoders at the end plates of the distal sections, there were a number of protrusions on the surface of the arm limiting its grasping capabilities. Also the air tubes for the distal sections were coiled on the outer surface of the arm, again limiting its grasping capabilities. Another problem faced with OCTARM V.2 was slippage of the string encoder cable at the distal sections which caused a loss of calibration. To address these issues, a new prototype of the arm has been developed called OCTARM

VI (see Figure 3.2). OCTARM VI has also been fitted with a rotary union which has an electrical slip ring with 36 electrical connections and also provides 12 independent passages for pneumatic lines. This new rotary union enables continuous rotation of the base of the manipulator. Shape sensing with the string encoders has also been reconfigured in OCTARM VI. There are now nine string encoders arranged around the base section. New eyelets for guiding the encoder cables have also been developed to reduce friction. In addition, the electrical wiring for sensors and the air tubes for pneumatic channels have been enclosed inside the actuators, providing a clean exterior surface of the arm for grasping.

The robot control system consists of commercial off-the-shelf Pentium III EBX form-factor Single Board Computer (SBC) with two ServoToGo data acquisition boards which provide analog and digital I/O. The computer runs the QNXNeutrino real-time Operating System and QMotor [9] the in-house developed hard real-time control software for implementation of the control algorithms. Data acquisition and control implementation were performed at a frequency of 500 Hz.

Joint Trajectory Tracking Experiment Description

Preliminary experimental results were obtained using the OCTARM V.2 to demonstrate the effectiveness of the neural network feedforward control. To test the low level controller with the neural network component given in (3.25), a sinusoidal desired trajectory was selected for the actuator lengths. The three actuators on a section are *120* degrees apart mechanically, so the desired trajectory for each actuator in a section is shifted *120* degrees in phase from the trajectory of the previous actuator in that section. The trajectory for section i, where $i = 1,2,3$ represents the three sections of the OCTARM, was selected as follows

$$q_{dik} = l_{\min_i} + \left(1 - \exp(-0.5t)\right)\left[l_0 + 2\sin\left(0.0625\pi t + \frac{2}{3}\pi k \right)\right] \quad \forall k = 1,2,3$$

where $q_{di} = \left[q_{di1}, q_{di2}, q_{di3}\right] \in \Re^3$ represents the desired trajectory for the actuators on section *i*, $l_{\min_i} \in \Re$ represents the minimum lengths of the actuators on section *i* and l_0 = 7 [cm] is the initial extension of the actuators on section *i*. The initial extension was selected to keep the operating pressure close to the center of its operational range. The minimum and maximum lengths of the sections which correspond to the minimum pressure (0 psi) and maximum pressure (130 psi) respectively are physical limitations of the actuators and were found to be $l_{\min_1}$ = 22.9 [cm], $l_{\min_2}$ = 22.4 [cm], $l_{\min_3}$ = 27.9 [cm], $l_{\max_1}$ = 35.8 [cm], $l_{\max_2}$ = 37.1 [cm], $l_{\max_3}$ = 47.7 [cm].

The system gains which yielded satisfactory performance were determined by trial and error and were as follows

$$K_s = diag\{1,1,1,1,1,1,1,1,1\},$$
$$\beta = diag\{1,1,1,1,1,1,0.5,0.5,0.5\},$$
$$\lambda_1 = 1, \quad \lambda_2 = 1, \quad \gamma_1 = 10, \quad \gamma_2 = 500,$$
$$\alpha_\omega = 0.001, \quad \alpha_\upsilon = 0.001, \quad \varepsilon = 0.01.$$

There was no training period utilized to determine the initial values for the weight matrices $\hat{W}(t)$ and $\hat{V}(t)$, the matrices were initialized to zero.

Analysis of Results

To test the effectiveness of the neural network feedforward term, we compared the controller in (3.25) with a standard PID controller and the controller given in (3.25) without

the neural network component. To provide a means to quantify the performance of each controller, we compute the following measures

$$M_e \triangleq \int_{t_0}^{t} \left\| e_1(\tau) \right\|^2 d\tau \tag{3.35}$$

$$M_u \triangleq \int_{t_0}^{t} \left\| u(\tau) \right\|^2 d\tau \tag{3.36}$$

where $M_u(t)$ is a measure of the energy expended by the controller, and $M_e(t)$ is a measure of the magnitude of the tracking error over the period of operation of the system.

	PID	$u(t)$ **without** $\hat{f}(t)$	$u(t)$ **with** $\hat{f}(t)$
M_e	2.8357×10^3	256.2135	58.2223
M_u	1.3689×10^6	2.0459×10^6	2.069×10^6

TABLE 3.1: Comparison of energy measures for different controllers

The performance of the system was first tested without the neural network component. The control gains for the controller were adjusted till good performance was obtained. Figures (3.3, 3.4, 3.5) show the actual and desired joint trajectories, joint tracking error, and the input pressure for the controller without the neural network component. Next, the neural network feedforward was added to the controller, and the neural network weight update law gains were adjusted till best performance was obtained. Figures (3.6, 3.7, 3.8) show the actual and desired joint trajectories, joint tracking error, and the input pressure for the controller with the neural network feedforward component. It can be seen from Figure 3.7 that the tracking error with the neural network feedforward component settles out to ± 0.5 [cm].

To compare the controller performance with and without the neural network feedforward component, the energy measures were computed for the two configurations. The energy measures were also computed for a standard PID controller, these results are presented to show the performance improvement obtained by using the controller in (3.25). Table I, shows a comparison of the performance for the three controller configurations. It can clearly be seen from Table I that improved tracking performance is achieved by adding the neural network feedforward to the controller.

Figure 3.2: OCTARM VI grasping a ball

Grasping Experiment Description

The whole arm grasping experiment with the neural network based joint controller will be conducted with OCTARM VI in the upcoming months. Figure 3.2 shows the OCTARM VI grasping a circular object. The object specific functions for this planar application are defined as follows

$$\beta_i(X_i) \triangleq (x_i - x_c)^2 + (y_i - y_c)^2 - r_o^2 \quad \forall i = 1,..,6 \tag{3.37}$$

where $X_c = [x_c, y_c]^T \in \Re^2$ represents the co-ordinates of the center of the object, and $r_0 \in \Re$ represents the object radius. The task space variable for each of the three sections and the mid-point of the three sections are $X_i = [x_i, y_i]^T \in \Re^2 \quad \forall i = 1,...,6.$ The following task-space velocity field for a planar, circular contour will be utilized [4]

$$\dot{X}_d = \vartheta(X_6) = -2K(X_6) f(X_6) \begin{bmatrix} (x_6 - x_c) \\ (y_6 - y_c) \end{bmatrix} + 2c(X_6) \begin{bmatrix} -(y_6 - y_c) \\ (x_6 - x_c) \end{bmatrix} \tag{3.38}$$

where $\dot{X}_d \in \Re^2$ represents the desired task space velocity for the end-effector, $X_6 \in \Re^2$ represents the end-effector co-ordinate, and the functions $f(X_6), K(X_6),$ and $c(X_6) \in \Re$ are defined as in [4].

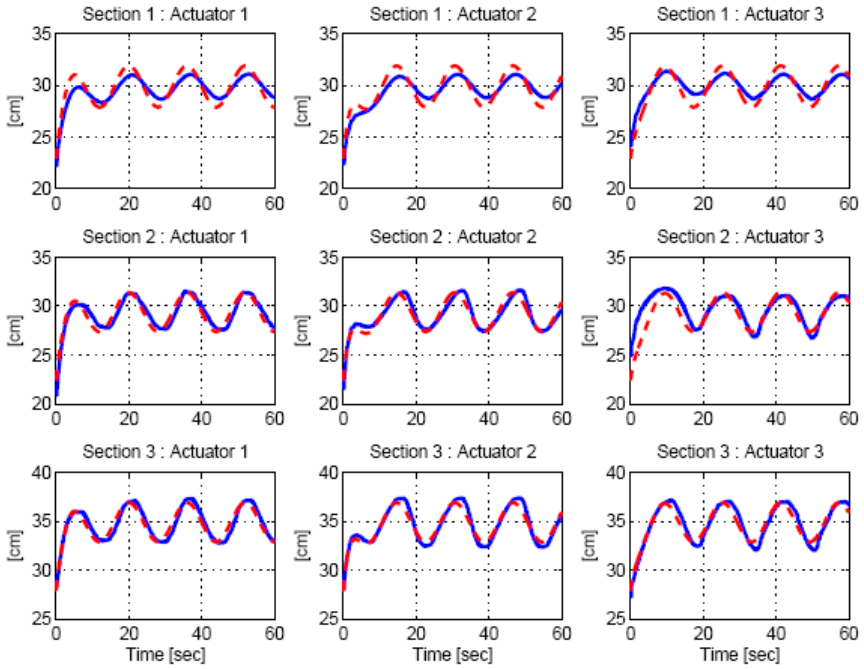

Figure 3.3: Actual and desired joint trajectory without neural network component, solid line represents the actual joint trajectory, dashed line represents the desired joint trajectory

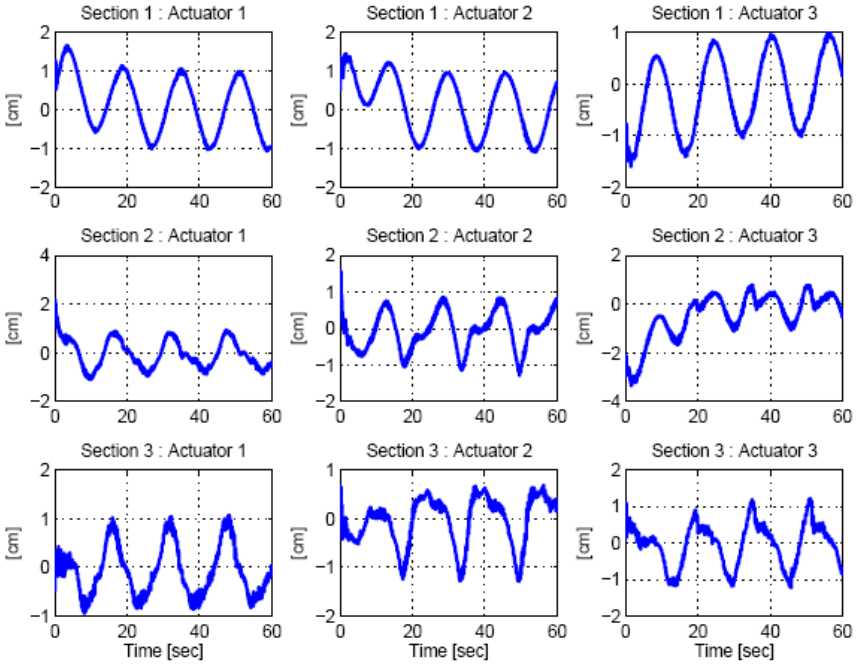

Figure 3.4: Tracking error without neural network component

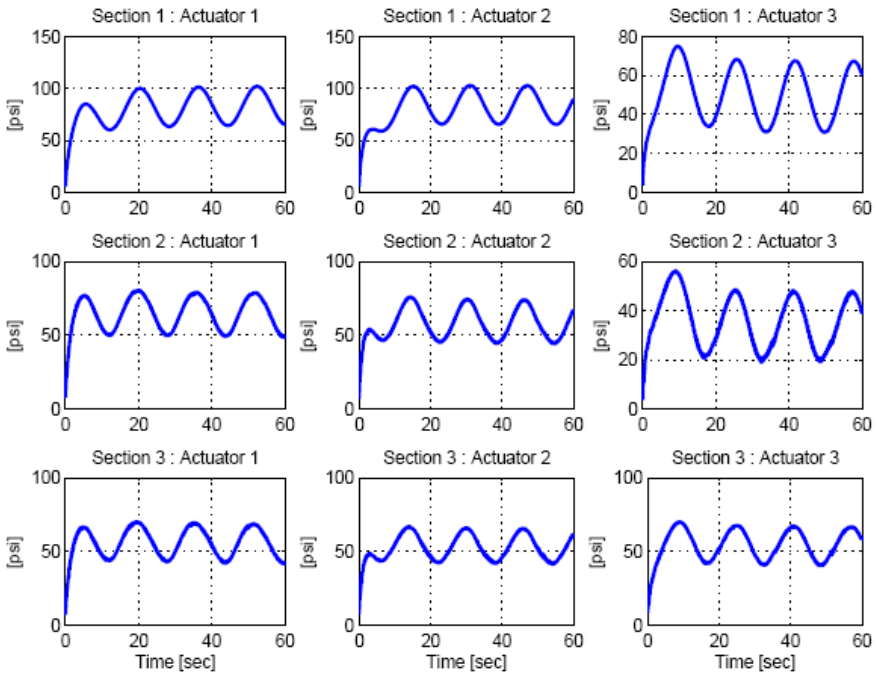

Figure 3.5: Control pressure without neural network component

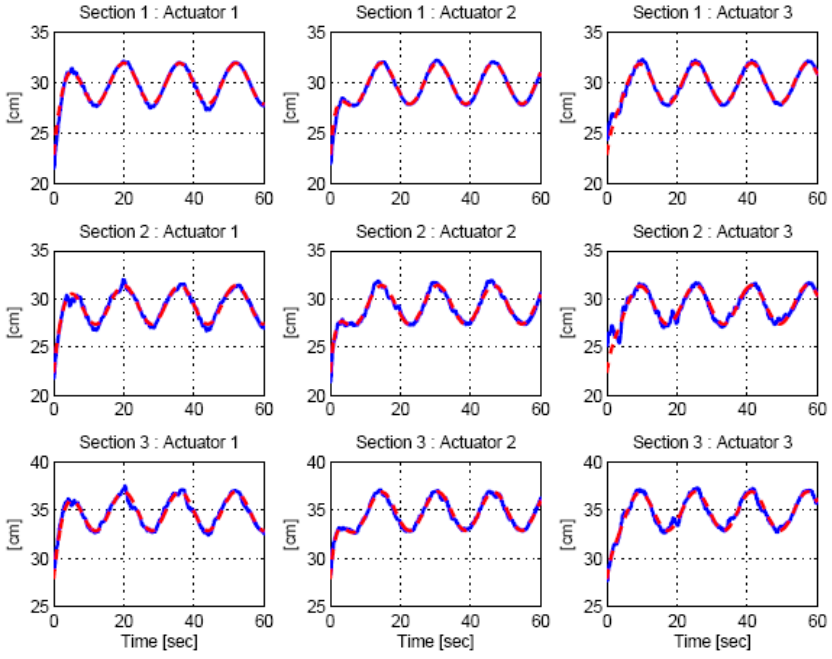

Figure 3.6: Actual and desired joint trajectory with neural network component, solid line represents the actual joint trajectory, dashed line represents the desired joint trajectory

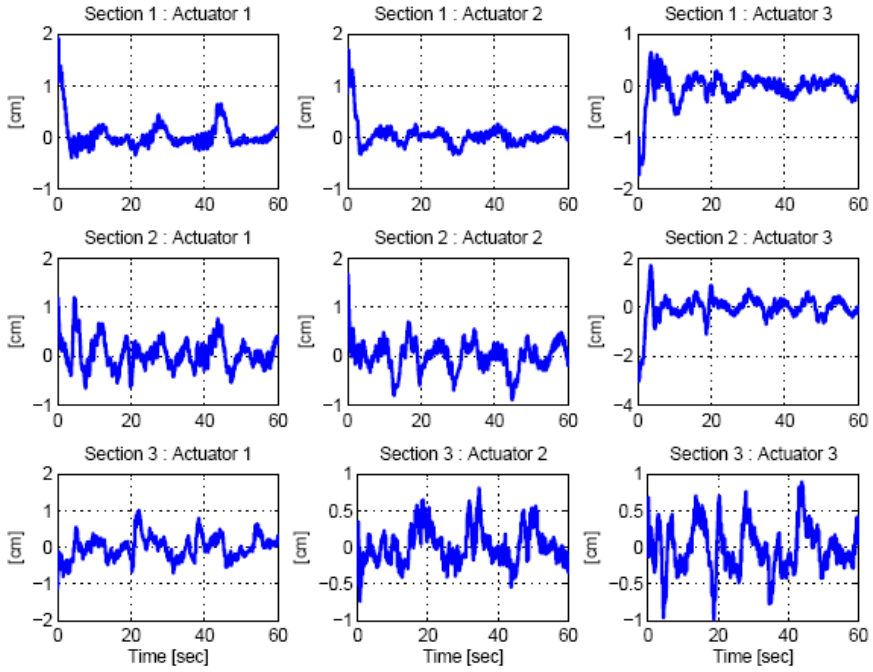

Figure 3.7: Tracking error with neural network

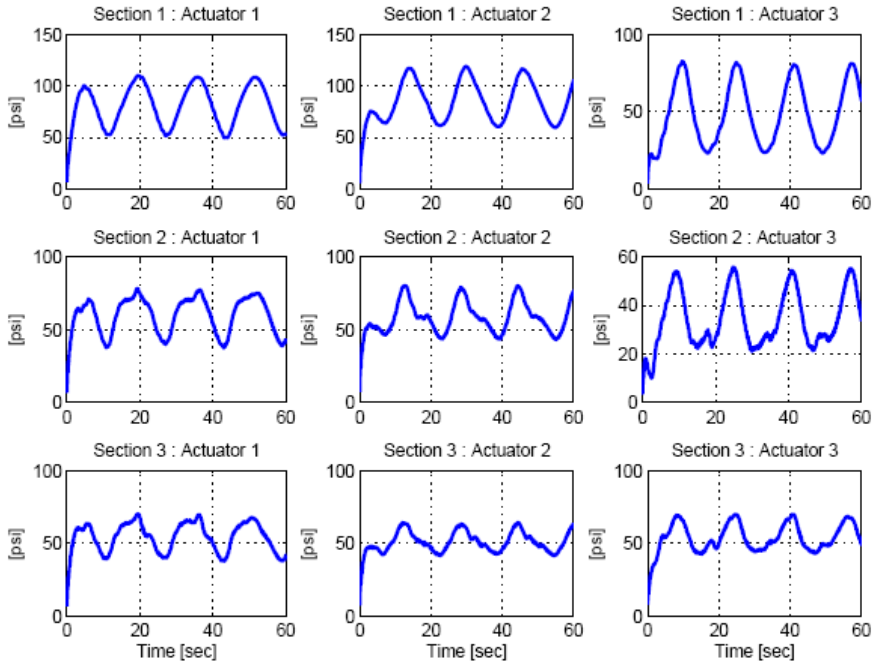

Figure 3.8: Control pressure with neural network component

CHAPTER 4

COORDINATION CONTROL FOR HAPTIC AND TELEOPERATOR SYSTEMS

System Model

The mathematical model for a 2*n*-DOF nonlinear haptic/teleoperator system consisting of a revolute *n*-DOF master and a revolute *n*-DOF slave system are assumed to have the following forms

$$M_1(x_m)\ddot{x}_m + N_1(x_m, \dot{x}_m) = T_1 + F_H \tag{4.1}$$

$$M_2(x_s)\ddot{x}_s + N_2(x_s, \dot{x}_s) = T_2 + F_E \tag{4.2}$$

In (4.1) and (4.2), $x_m(t), \dot{x}_m(t), \ddot{x}_m(t) \in \Re^n$ denote the task-space position, velocity, and acceleration for the master system and $x_s(t), \dot{x}_s(t), \ddot{x}_s(t)$ denote the task-space position, velocity, and acceleration for the slave system, $M_1(x_m), M_2(x_s) \in \Re^{n\times n}$ represent the inertia effects, $N_1(x_m, \dot{x}_m), N_2(x_s, \dot{x}_s) \in \Re^n$ represent other dynamic effects, $T_1(t), T_2(t) \in \Re^n$ represent the control input vectors, $F_H(t) \in \Re^n$ represents the user input force, and $F_E(t) \in \Re^n$ represents the input force from the physical or virtual environment. End-effector positions $x_m(t)$ and $x_s(t)$ can be decomposed as follows

$$x_m \triangleq \begin{bmatrix} x_{mp}^T & x_{mr}^T \end{bmatrix}^T \qquad\qquad x_s \triangleq \begin{bmatrix} x_{sp}^T & x_{sr}^T \end{bmatrix}^T$$

where $x_{mp}(t), x_{sp}(t) \in \Re^p$ represent position vectors and $x_{mr}(t), x_{sr}(t) \in \Re^r$ represent orientation angle vectors, where the integers *p* and *r* satisfy $p + r = n$. The subsequent

development utilizes the property that the master and slave inertia matrices are positive definite, symmetric and satisfies the following inequalities [6].

$$m_{1i}\left\|\xi\right\|^2 \le \xi^T M_i(\cdot)\xi \le m_{2i}\left\|\xi\right\|^2 \quad \forall \xi \in \Re^n \tag{4.3}$$

where $m_{1i}, m_{2i} \in \Re,\ \ i = 1,2$ are positive constants, and $\left\|\cdot\right\|$ denotes the Euclidean norm. To achieve the control objectives, the subsequent development is derived based on the assumption that $x_m(t), x_s(t), \dot{x}_m(t), \dot{x}_s(t)$ are measurable, and $M_i(\cdot), N_i(\cdot)$ are second order differentiable for $i = 1,2$.

Assumption 1: The user input force and the environmental force along with their first and second time derivatives, $F_H(t), \dot{F}_H(t), \ddot{F}_H(t), F_E(t), \dot{F}_E(t)$ and $\ddot{F}_E(t)$ are bounded (see [57] and [59] for the precedence of this type of assumption).

MIF Control Development

For the MIF controller development, the following analysis will prove a semi-global asymptotic result despite parametric uncertainty in the master and slave system dynamic models provided the user and the physical or virtual environmental input forces are measurable. It should be noted that for many types of virtual slave systems, the dynamic model of the virtual slave is known *a priori*; however; unstructured uncertainties in the dynamic model are common for teleoperator slave systems.

Control Objective and Model Transformation

A control objective for haptic and teleoperator systems is to ensure the coordination between the master and the slave systems and to meet the tracking objective in the following sense

$$x_s(t) \to x_m(t) \quad \text{as} \quad t \to \infty \tag{4.4}$$

$$x_m(t) \to \xi_d(t) \quad \text{as} \quad t \to \infty \tag{4.5}$$

where $\xi_d(t) \in \Re^n$ is a subsequently designed desired trajectory. Another sub-control objective is to guarantee that the closed-loop system remains passive with respect to the user and the physical/virtual environmental power in the following sense [57]

$$\int_{t0}^{t} \left(\dot{x}_m^T(\tau) F_H(\tau) + \dot{x}_s^T(\tau) F_E(\tau) \right) d\tau \geq -c_1^2 \tag{4.6}$$

where $c_1 \in \Re$ is a bounding constant. The passivity objective is motivated to ensure the safety of the user and the physical environment [57]. The final objective is that all signals are required to remain bounded within the closed-loop system. It should be noted that, the passivity objective is not met when the subsequently presented user assist mechanism is enabled.

To facilitate the subsequent development, an invertible transformation is defined that encodes the control objectives as follows

$$x \triangleq S \begin{bmatrix} x_m^T & x_s^T \end{bmatrix}^T \tag{4.7}$$

where $x(t) \in \Re^{2n}$ and $S \in \Re^{2n \times 2n}$ is defined as follows

$$S \triangleq \begin{bmatrix} I_n & 0_{n \times n} \\ I_n & -I_n \end{bmatrix} \tag{4.8}$$

where $I_n \in \Re^{n\times n}$ denotes the identity matrix, $0_{n\times n} \in \Re^{n\times n}$ denotes a matrix of zeros, and it is noted that $S^{-1} = S$. After utilizing the transformation defined in (4.7), the dynamic models of the haptic/teleoperator systems given in (4.1) and (4.2) can be combined as follows

$$\bar{M}\ddot{x} + \bar{N} = \bar{T} + \bar{F} \tag{4.9}$$

where $\bar{N}(x,\dot{x}), \bar{T}(t), \bar{F}(t) \in \Re^{2n}$ and $\bar{M}(x) \in \Re^{2n\times 2n}$ are defined as follows

$$\bar{M} \triangleq S^{-T}\begin{bmatrix} M_1 & 0_{n\times n} \\ 0_{n\times n} & M_2 \end{bmatrix} S^{-1} \tag{4.10}$$

$$\bar{N} \triangleq S^{-T}\begin{bmatrix} N_1^T & N_2^T \end{bmatrix}^T \tag{4.11}$$

$$\bar{T} \triangleq S^{-T}\begin{bmatrix} T_1^T & T_2^T \end{bmatrix}^T \tag{4.12}$$

$$\bar{F} \triangleq S^{-T}\begin{bmatrix} F_H^T & F_E^T \end{bmatrix}^T \tag{4.13}$$

The subsequent development utilizes the property that $\bar{M}(x)$ is positive definite, symmetric and satisfies the following inequalities [6]

$$\bar{m}_1 \|\xi\|^2 \le \xi^T \bar{M}(x)\xi \le \bar{m}_2 \|\xi\|^2 \quad \forall \xi \in \Re^{2n} \tag{4.14}$$

where $\bar{m}_1, \bar{m}_2 \in \Re$ are positive constants. By utilizing the assumption that $M_i(\cdot), N_i(\cdot)$ are second order differentiable for $i = 1,2$, it is clear that are also second order differentiable.

To facilitate the development of the error system, the filtered tracking error signal, denoted by $r(t) \in \Re^{2n}$, is defined as follows

$$r \triangleq \dot{e}_2 + \alpha_1 e_2 \tag{4.15}$$

where $e_2(t) \in \Re^{2n}$ is defined as follows

$$e_2 \triangleq \dot{e}_1 + \alpha_2 e_1 \tag{4.16}$$

where $\alpha_1, \alpha_2 \in \Re$ are positive control gains, and $e_1 \in \Re^{2n}$ is defined as follows

$$e_1 \triangleq x_d - x \tag{4.17}$$

The error signal $e_1(t)$ can be decomposed as follows

$$e_1 \triangleq \begin{bmatrix} e_{11}^T & e_{12}^T \end{bmatrix}^T \tag{4.18}$$

where $e_{11}(t) \in \Re^n$ represents the master system tracking error, and $e_{12}(t) \in \Re^{2n}$ represents the coordination error. In (4.17), $x_d(t) \in \Re^{2n}$ is defined as follows

$$x_d \triangleq \begin{bmatrix} \xi_d^T & 0_n^T \end{bmatrix}^T \tag{4.19}$$

where $0_n \in \Re^n$ denotes a vector of zeros. Based on the definition of $x_d(t)$ in (4.7) and $e_1(t)$ in (4.17), it is clear that if $\|e_1(t)\| \to 0$ then $x_s(t) \to x_m(t)$ and $x_m(t) \to \xi_d(t)$.

The desired trajectory $\xi_d(t)$ introduced in (4.5) is generated by the following second-order coupled dynamic target system

$$\dot{\xi}_d = \gamma \begin{bmatrix} \varphi^T(\xi_p) & 0_r^T \end{bmatrix}^T + \eta_d \tag{4.20}$$

$$M_T \dot{\eta}_d + B_T \eta_d + K_T \lambda_d = F \tag{4.21}$$

where $\eta_d(t) \in \Re^n$ is an auxiliary filter signal, $M_T, B_T, K_T \in \Re^{n \times n}$ are constant positive definite, diagonal matrices, $\varphi(\cdot) \in \Re^{n \times n}$ is a velocity field function [8] that encodes the user assist mechanism, $0_r \in \Re^r$ denotes a vector of zeros, γ is a constant gain that is either 0 or 1. It should be noted that, when $\gamma = 0$, the user assist mechanism is disabled, and when $\gamma = 1$, then the user assist mechanism is enabled. In (4.21), $F(t) \in \Re^n$ and $\lambda_d(t) \in \Re^n$ are defined as follows

$$F \triangleq F_H + F_E \tag{4.22}$$

$$\lambda_d \triangleq \xi_d - \gamma \left[\int_{t_0}^{t} \varphi^T \left(\xi_p(\tau) \right) d\tau \quad 0_r^T \right]^T \tag{4.23}$$

The desired trajectory, $\xi_d(t)$, is generated by the differential equation of (4.20), and can be decomposed as follows

$$\xi_d \triangleq \left[\xi_p^T \quad \xi_r^T \right]^T \tag{4.24}$$

where $\xi_p(t) \in \Re^p$ represents a position vector, and $\xi_r(t) \in \Re^r$ represents an orientation angle vector.

Remark 1: *Velocity fields have been utilized in previous control literature, see [4] and [8] for their definition and application. The velocity field function in (4.20) is integrated to assist the user in executing a remote task (i.e., tracking a circular contour). It is assumed that the velocity field function is designed such that* $\varphi(\cdot), \dot{\varphi}(\cdot), \ddot{\varphi}(\cdot)$ *and* $\dddot{\varphi}(\cdot)$ *are bounded provided that their arguments are bounded.*

Remark 2: *The velocity field function* $\varphi(\cdot)$ *is assumed to be designed such that, from (4.20), if* $\eta_d(t) \in \mathcal{L}_\infty$ *then* $\xi_d(t), \dot{\xi}_d(t) \in \mathcal{L}_\infty$. *Based on this assumption and the analysis in Appendix A, it is easy to show that all signals in dynamic target system given in (4.20) and (4.21) are bounded, and that the higher order derivatives are also bounded.*

Remark 3: *It should be noted that, when the user assist mechanism is disabled, (i.e.,* $\gamma = 0$*) the target system defined by (4.20) and (4.21), becomes a standard impedance model as follows*

$$M_T \ddot{\xi}_d + B_T \dot{\xi}_d + K_T \xi_d = F \tag{4.25}$$

Closed-Loop Error System

Based on the assumption that the user forces $F_H(t)$, and the physical/virtual environmental forces $F_E(t)$ are measurable, the control input $\overline{T}(t)$ of (4.9) is designed as follows

$$\overline{T} \triangleq \overline{u} - \overline{F} \tag{4.26}$$

where $\overline{u}(t) \in \Re^{2n}$ is a subsequently designed auxiliary control input. Substituting (4.26) into (4.9) results in the following simplified dynamic system

$$\overline{M}\ddot{x} + \overline{N} = \overline{u} \tag{4.27}$$

After taking the time derivative of (4.15) and premultiplying by $\overline{M}(x)$, the following expression can be derived

$$\overline{M}\dot{r} = \overline{M}\ddot{x}_d + \dot{\overline{M}}\dot{x} + \dot{\overline{N}} - \dot{\overline{u}} + \alpha_2 \overline{M}\dot{e}_1 + \alpha_1 \overline{M}\dot{e}_2 \tag{4.28}$$

where (4.16), (4.17), and the time derivative of (4.27) were utilized. To facilitate the subsequent analysis, the expression in (4.28) can be arranged as follows

$$\overline{M}\dot{r} = \tilde{N} + N_d - e_2 - \dot{\overline{u}} - \frac{1}{2}\dot{\overline{M}}r \tag{4.29}$$

where $\tilde{N}(x, \dot{x}, \ddot{x}, t) \in \Re^{2n}$ is defined as follows

$$\tilde{N} \triangleq N - N_d \tag{4.30}$$

where $N(x, \dot{x}, \ddot{x}, t) \in \Re^{2n}$ is defined as follows

$$N \triangleq \overline{M}\ddot{x}_d + \dot{\overline{M}}\dot{x} + \alpha_2 \overline{M}\dot{e}_1 + \alpha_1 \overline{M}\dot{e}_2 + e_2 + \dot{\overline{N}} + \frac{1}{2}\dot{\overline{M}}r \tag{4.31}$$

and $N_d(t) \in \Re^{2n}$ is defined as follows

$$\begin{aligned} N_d &\triangleq N\Big|_{x=x_d, \dot{x}=\dot{x}_d, \ddot{x}=\ddot{x}_d} \\ &= \overline{M}(x_d)\ddot{x}_d + \dot{\overline{M}}(x_d, \dot{x}_d)\ddot{x}_d + \dot{\overline{N}}(x_d, \dot{x}_d, \ddot{x}_d) \end{aligned} \tag{4.32}$$

Remark 4: *After utilizing (4.19), (4.32) and the fact that we show in Appendix A, then* $\left\|N_d(t)\right\|$ *and* $\left\|\dot{N}_d(t)\right\|$ *can be upper bounded as follows*

$$\left\|N_d(t)\right\| \leq \varsigma_1 \qquad \left\|\dot{N}_d(t)\right\| \leq \varsigma_2 \tag{4.33}$$

where $\varsigma_1, \varsigma_2 \in \Re$ are known positive constants.

To achieve the stated control objectives, the auxiliary control input $\overline{u}(t)$ introduced in (4.26) is designed as follows

$$\overline{u} \triangleq \left(k_s+1\right)\left[e_2(t)-e_2(t_0)+\alpha_1\int_{t_0}^{t} e_2(\tau)d\tau\right]+\left(\beta_1+\beta_2\right)\int_{t_0}^{t}\operatorname{sgn}\left(e2(\tau)\right)d\tau \tag{4.34}$$

where $k_s, \beta_1, \beta_2 \in \Re$ are positive control gains, and $\operatorname{sgn}(\cdot)$ denotes the vector signum function. The term $e_2(t_0)$ in (4.34) is used to ensure that $\overline{u}(t_0) = 0_{2n}$, where $0_{2n} \in \Re^{2n}$ denotes a vector of zeros. The time derivative of (4.34) is obtained as follows

$$\dot{\overline{u}} = \left(k_s+1\right)r+\left(\beta_1+\beta_2\right)\operatorname{sgn}(e_2) \tag{4.35}$$

where (4.15) was utilized. Substituting (4.35) into (4.29) results in the following closed-loop error system

$$\overline{M}\dot{r} = -\left(k_s+1\right)r-\left(\beta_1+\beta_2\right)\operatorname{sgn}(e_2)+\tilde{N}+N_d-e_2-\frac{1}{2}\dot{\overline{M}}r \tag{4.36}$$

Stability Analysis

Theorem 1: *The controller given in (4.26) and (4.34) guarantees that all the system signals are bounded under the closed-loop operation and that coordination between the master and the slave systems, and the tracking objective are met in the sense that*

$$x_s(t) \to x_m(t) \quad \text{as} \quad t \to \infty \tag{4.37}$$

$$x_m(t) \to \xi_d(t) \quad \text{as } t \to \infty \tag{4.38}$$

provided the control gain β_1 *introduced in (4.34) is selected to satisfy the following sufficient condition*

$$\beta_1 > \varsigma_1 + \frac{1}{\alpha 1}\varsigma_2 \tag{4.39}$$

where ς_1 *and* ς_2 *were introduced in (4.33), the control gains* α_1 *and* α_2 *are selected greater than 2, and* k_s *is selected sufficiently large relative to the system's initial conditions.*

Proof. See Appendix B.

Theorem 2: *The controller given in (4.26) and (4.34) guarantees that the closed-loop system is passive with respect to the user and the physical/virtual environmental power when the user assist mechanism is disabled (i.e.,* $\gamma = 0$*).*

Proof. See Appendix C.

UMIF Control Development

For the UMIF controller development, the following analysis will prove a global asymptotic result despite unmeasurable user and environmental input forces provided the dynamic models of the master and slave systems are known. Assumption 1 is also utilized for the subsequent development. It should be noted that, for many types of virtual slave systems, the virtual environmental forces are measurable; however, the user input force may not be measurable.

Control Objective and Model Transformation

A control objective for haptic and teleoperator systems is to guarantee coordination between the master and the slave systems and to meet the tracking objective in the following sense

$$x_s(t) \to x_m(t) \quad \text{as} \quad t \to \infty \tag{4.40}$$

$$x_m(t) \to \xi_d(t) \quad \text{as} \quad t \to \infty \tag{4.41}$$

where $\xi_1(t) \in \Re^n$ is a subsequently designed desired trajectory. Another sub-control objective is to guarantee that the system remains passive with respect to the user and the environmental power as in (4.6). It should be noted that the passivity objective is not met when the user assist mechanism is enabled. The final objective is that all signals are required to remain bounded within the closed-loop system.

To facilitate the subsequent development, an invertible transformation is defined that encodes the control objectives as follows

$$x \triangleq S \begin{bmatrix} x_m \\ x_s \end{bmatrix} + \begin{bmatrix} 0_n \\ \xi_2 \end{bmatrix} \tag{4.42}$$

where $\xi_2(t) \in \Re^n$ is a subsequently defined auxiliary trajectory, $S \in \Re^{2n \times 2n}$ was defined in (4.8), and $x(t) \in \Re^{2n}$. After utilizing the transformation defined in (4.42), the dynamic models of the haptic/teleoperator system given in (4.1) and (4.2) can be combined as follows

$$\bar{M}\ddot{x} - \bar{M} \begin{bmatrix} 0_n \\ \ddot{\xi}_2 \end{bmatrix} + \bar{N} = \bar{T} + \bar{F} \tag{4.43}$$

where $\bar{M}(x)$, $\bar{N}(x, \dot{x})$, $\bar{T}(t)$, and $\bar{F}(t)$ were defined in (4.10)-(4.13).

The filtered tracking error signal denoted by $r(t) \in \Re^{2n}$ is defined as follows

$$r \triangleq \dot{e}_2 + e_2 \tag{4.44}$$

where $e_2(t) \in \Re^{2n}$ is defined as follows

$$e2 \triangleq \bar{M}(\dot{e}_1 + \alpha e_1) \tag{4.45}$$

where $\alpha \in \Re$ is a positive control gain, and $e_1(t) \in \Re^{2n}$ is defined as follows

$$e_1 \triangleq \xi_d - x \tag{4.46}$$

where $\xi_d(t)$ is a subsequently defined desired trajectory. The error signal $e_1(t)$ can be decomposed as follows

$$e_1 \triangleq \begin{bmatrix} e_{11}^T & e_{12}^T \end{bmatrix}^T \tag{4.47}$$

where $e_{11}(t) \in \Re^n$ represents the master system tracking error, and $e_{12}(t) \in \Re^n$ represents the coordination error.

To compensate for the unmeasurable user and physical/virtual environmental forces, a nonlinear force observer is designed subsequently. This nonlinear observer is utilized in driving the target system, thus requiring a 2*n*-dimensional system. As a result of this fact, the desired trajectory, defined as $\xi_d(t) \in \Re^{2n}$, is generated by the following second-order coupled dynamic target system

$$\dot{\xi}_d = \gamma \begin{bmatrix} \varphi^T(\xi_{1p}) & 0_s^T \end{bmatrix}^T + \eta_d \tag{4.48}$$

$$M_T \dot{\eta}_d + B_T \eta_d + K_T \lambda_d = \left(\bar{M} M_T^{-1}\right)^{-1} \hat{F} \tag{4.49}$$

where $\eta_d(t) \in \Re^{2n}$ is an auxiliary filter signal, $\bar{M}(x)$ was defined in (4.10), M_T, B_T and $K_T \in \Re^{2n \times 2n}$ represent constant, positive definite, diagonal matrices, $\hat{F}(t) \in \Re^{2n}$ is a subsequently designed nonlinear observer, $\varphi(\cdot) \in \Re^p$ was introduced in Section 4.1, $0_s \in \Re^s$ denotes a vector of zeros where $s + p = 2n$ and γ is a constant gain that is either 0 or 1. It

should be noted that, when $\gamma = 0$, the user assist mechanism is disabled, and when $\gamma = 1$, then the user assist mechanism is enabled. In (4.49), the term $\lambda_d(t) \in \Re^{2n}$ is defined as follows

$$\lambda_d \triangleq \xi_d - \gamma \left[\int_{t_0}^{t} \varphi^T \left(\xi_{1p}(\tau) \right) d\tau \quad 0_s^T \right]^T \tag{4.50}$$

where $\xi_d(t) \triangleq \left[\xi_1^T \quad \xi_2^T \right]^T$ is generated by the differential equation given in (4.48) where $\xi_1(t), \xi_2(t) \in \Re^n$. The desired trajectory for the master system denoted by $\xi_1(t)$, can be decomposed as follows

$$\xi_1 \triangleq \left[\xi_{1p}^T \quad \xi_{1r}^T \right]^T \tag{4.51}$$

where $\xi_{1p}(t) \in \Re^p$ represents a position vector, and $\xi_{1r}(t) \in \Re^r$ represents an orientation angle vector.

Remark 5: *The velocity field function* $\varphi(\cdot)$ *is assumed to be designed such that, from (4.48), if* $\eta_d(t) \in \mathcal{L}_\infty$ *then* $\xi_d(t), \dot{\xi}_d(t) \in \mathcal{L}_\infty$. *Subsequent analysis will prove that* $\hat{F}(t) \in \mathcal{L}_\infty$. *After utilizing these facts along with (4.14), the analysis in Appendix F proves that all signals in the dynamic target system given in (4.48) and (4.49) are bounded.*

Remark 6: *Although the desired trajectory dynamics defined in (4.48) and (4.49) generated a 2n-dimensional signal, it should be noted that the master system tracks an n-dimensional signal, denoted as* $\xi_1(t)$. *The use of a 2n-dimensional desired trajectory generator is a consequence of the fact that both the user input force and the physical/virtual environmental force are unmeasurable, and hence, a 2n-dimensional*

nonlinear force observer must be utilized to drive the target system as defined in (4.50). From the definition of the transformation and the error signal $e_1(t)$ *(see (4.42) and (4.46)), it is clear that additional set of desired trajectory dynamics, denoted by* $\xi_2(t)$*, are eliminated in the error system development.*

Remark 7: It should be noted that, when the user assist mechanism is disabled (i.e., $\gamma = 0$), then the target system defined by (4.48) and (4.49), becomes a standard impedance model described as follows

$$M_T \ddot{\xi}_d + B_T \dot{\xi}_d + K_T \xi_d = \left(\bar{M} M_T^{-1}\right)^{-1} \hat{F} \tag{4.52}$$

Closed-Loop Error System

To develop the closed-loop error system for $r(t)$, error system dynamics for $e_1(t)$ and $e_2(t)$ are derived first**.** After taking the second time derivative of (4.46) and premultiplying by $\bar{M}(x)$, the following expression can be derived

$$\begin{aligned} \bar{M}\ddot{e}_1 = \hat{F} - \left(\bar{M} M_T^{-1}\right)\left(B_T \eta_d + K_T \lambda_d\right) \\ - \bar{M}\begin{bmatrix} 0_n \\ \ddot{\xi}_2 \end{bmatrix} + \bar{N} - \bar{T} - \bar{F} \\ + \gamma \bar{M} \frac{d}{dt}\left(\begin{bmatrix} \varphi^T\left(\xi_{1p}\right) & 0_s^T \end{bmatrix}^T\right) \end{aligned} \tag{4.53}$$

where (4.43), (4.48) and (4.49) were utilized. Based on the assumption of exact model knowledge, the control input $\bar{T}(t)$ is designed as follows

$$\begin{aligned} \bar{T} \triangleq \bar{T}_1 - \left(\bar{M} M_T^{-1}\right)\left(B_T \eta_d + K_T \lambda_d\right) - \bar{M}\begin{bmatrix} 0_n \\ \ddot{\xi}_2 \end{bmatrix} \\ + \bar{N} + \gamma \bar{M}\left(\frac{d}{dt}\begin{bmatrix} \varphi^T\left(\xi_{1p}\right) & 0_s^T \end{bmatrix}^T\right) \end{aligned} \tag{4.54}$$

where $\overline{T}_1(t) \in \Re^{2n}$ is a subsequently designed auxiliary control input. Substituting (4.54) into (4.53) results in the following simplified expression

$$\overline{M}\ddot{e}_1 = \hat{F} - \overline{F} - \overline{T}_1 \tag{4.55}$$

The time derivative of $e_2(t)$ in (4.45) can be obtained as follows

$$\dot{e}_2 = \dot{\overline{M}}\dot{e}_1 + \alpha\dot{\overline{M}}e_1 + \alpha\overline{M}\dot{e}_1 + \hat{F} - \overline{F} - \overline{T}_1 \tag{4.56}$$

where (4.55) was utilized. Based on (4.56), the auxiliary control input $\overline{T}_1(t)$ is designed as follows

$$\overline{T}_1 \triangleq \dot{\overline{M}}\dot{e}_1 + \alpha\dot{\overline{M}}e_1 + \alpha\overline{M}\dot{e}_1 \tag{4.57}$$

After substituting (4.57) into (4.56), the following simplified expression is obtained

$$\dot{e}_2 = \hat{F} - \overline{F} \tag{4.58}$$

Taking the time derivative of (4.58) results in the following expression

$$\ddot{e}_2 = \dot{\hat{F}} - \dot{\overline{F}} \tag{4.59}$$

The error system dynamics for $r(t)$ can be derived by taking the time derivative of (4.44)

$$\dot{r} = r - e_2 + \dot{\hat{F}} - \dot{\overline{F}} \tag{4.60}$$

where (4.44) and (4.59) were utilized. To achieve the stated control objectives, the proportional-integral like nonlinear observer $\hat{F}(t)$ introduced in (4.49) is designed as follows

$$\begin{aligned} \hat{F} \triangleq & -(k_s+1)\left[e_2(t) - e_2(t_0) + \int_{t_0}^{t} e_2(\tau)d\tau\right] \\ & -(\beta_1+\beta_2)\int_{t_0}^{t} \operatorname{sgn}(e_2(\tau))d\tau \end{aligned} \tag{4.61}$$

where k_s, β_1, and $\beta_2 \in \Re$ are positive control gains. The term $e_2(t_0)$ is used to ensure that $\hat{F}(t_0) = 0_{2n}$. The time derivative of (4.61) is obtained as follows

$$\dot{\hat{F}} = -\left(k_s + 1\right) r - \left(\beta_1 + \beta_2\right) \text{sgn}\left(e_2\right) \tag{4.62}$$

where (4.44) was utilized. Substituting (4.62) into (4.60) results in the following closed-loop error system

$$\dot{r} = -e_2 - \dot{\tilde{F}} - k_s r - \left(\beta_1 + \beta_2\right) \text{sgn}\left(e_2\right) \tag{4.63}$$

Remark 8: After utilizing (4.13) and Assumption 1, *then* $\left\|\dot{\tilde{F}}(t)\right\|$ and $\left\|\ddot{\tilde{F}}(t)\right\|$ can be upper bounded as follows

$$\left\|\dot{\tilde{F}}(t)\right\| \leq \varsigma_3 \qquad \left\|\ddot{\tilde{F}}(t)\right\| \leq \varsigma_4 \tag{4.64}$$

where $\varsigma_3, \varsigma_4 \in \Re$ denote positive bounding constants.

Stability Analysis

Theorem 3: *The controller given in (4.54) and (4.57) guarantees that all signals are bounded under closed-loop operation and that coordination between the master and the slave systems, and the tracking objective are met in the sense that*

$$x_s(t) \to x_m(t) \quad \text{as} \quad t \to \infty \tag{4.65}$$

$$x_m(t) \to \xi_1(t) \quad \text{as} \quad t \to \infty \tag{4.66}$$

provided the control gain β_1 *introduced in (4.61) is selected to satisfy the sufficient condition*

$$\beta_1 > \varsigma_3 + \varsigma_4 \tag{4.67}$$

where ς_3 *and* ς_4 *were introduced in (4.64).*

Proof. See Appendix D.

Theorem 4: *The controller given in (4.54) and (4.57) guarantees that the haptic/teleoperator system is passive with respect to the user and the physical/virtual environmental power when the user assist mechanism is disabled (i.e.,* $\gamma = 0$ *).*

Proof. See Appendix E.

Experimental Results

Two different experiments were done to show the validity of the presented controllers. In both experimental setups, two links of the Barrett whole arm manipulator (WAM) were used as the master systems, while the slave systems were simulated on a PC. The WAM, which served as the master system, is a three-link, seven degree of freedom (d.o.f.), highly dexterous, back-drivable robot manipulator. For better representation of the presented user assist mechanism, the third link was removed that disabled the last three joints and the second link was replaced with a handle mounted on an ATI force/torque sensor to sense the applied user/human force. Two joints out of the remaining four joints were locked at fixed angles and the remaining links of the manipulator were used as a two d.o.f. planar robot manipulator. Figure 4.1, shows the experimental setup.

The control algorithm, written in "C++", was hosted on an AMD Athlon 1.2 GHz PC operating the QNX 6.2.1 RTP (Real Time Platform). This PC also served as the slave system. For real-time graphing, data logging, and on-line gain adjustment, Qmotor 3.0 [9] control software was utilized. A frequency of 1.0 [kHz] was utilized for data acquisition and control implementation by using the ServoToGo I/O board. Optical encoders, located at the motor shaft of each axis, were utilized to measure the joint positions.

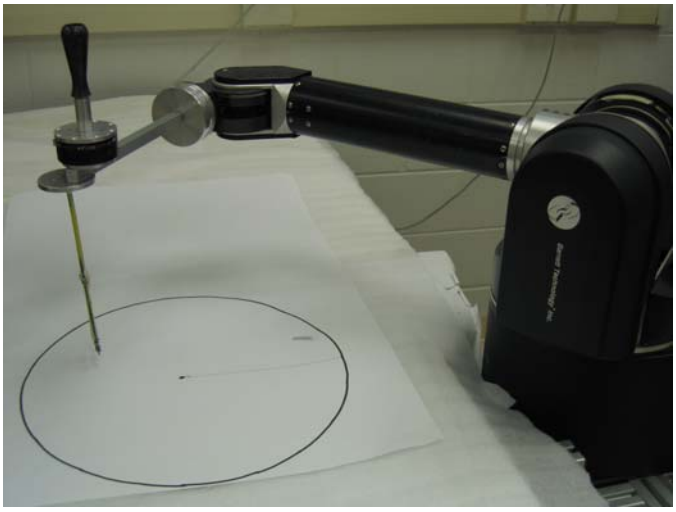

Figure 4.1: Barrett whole arm manipulator

Since there are no tachometers on WAM, a filtered backwards difference algorithm is utilized to measure the joint velocities. By utilizing a joint-space proportional derivative (PD) controller, the WAM was servoed to the following initial joint configuration for both experimental setups

$$q(t_0) = \begin{bmatrix} -45 & 90 & 90 & 100 \end{bmatrix}^T \text{ (in[deg])}. \tag{4.68}$$

Once the WAM was servoed to the initial joint configuration in (4.68), links 2 and 3 were locked, resulting in a planar configuration with links 1 and 4 (see Figure 4.1).

The forward kinematics and the Jacobian are given as follows

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix} = \begin{bmatrix} l_1 \cos(q_{i1}) + l_2 \cos(q_{i1} + q_{i4}) \\ l_1 \sin(q_{i1}) + l_2 \sin(q_{i1} + q_{i4}) \end{bmatrix} \tag{4.69}$$

$$J_i(q_i) = \begin{bmatrix} -l_1 \sin(q_{i1}) - l_4 \sin(q_{i1} + q_{i4}) & -l_4 \sin(q_{i1} + q_{i4}) \\ l_1 \cos(q_{i1}) + l_4 \cos(q_{i1} + q_{i4}) & l_4 \cos(q_{i1} + q_{i4}) \end{bmatrix} \tag{4.70}$$

where $l_1 = 0.558 \ [\text{m}]$ and $l_4 = 0.32 \ [\text{m}]$, and $i = 1$ denotes the master system and $i = 2$

denotes the slave system. The following dynamic model is utilized in both experiments to represent the WAM (*i.e.*, the master system) and the simulated slave system

$$\begin{aligned} M_i &= \begin{bmatrix} p_1 + 2p_2\sin(q_{i4}) & p_3 + p_2\sin(q_{i4}) \\ p_3 + p_2\sin(q_{i4}) & p_3 \end{bmatrix} \\ N_i &= \begin{bmatrix} -p_2\sin(q_{i4})\dot{q}_{i4} & -p_2\sin(q_{i4})(\dot{q}_{i1}+\dot{q}_{i4}) \\ p_2\sin(q_{i4})\dot{q}_{i1} & 0 \end{bmatrix}\begin{bmatrix} \dot{q}_{i1} \\ \dot{q}_{i4} \end{bmatrix} \\ &\quad + \begin{bmatrix} f_{d1} & 0 \\ 0 & f_{d2} \end{bmatrix}\begin{bmatrix} \dot{q}_{i1} \\ \dot{q}_{i4} \end{bmatrix} + \begin{bmatrix} f_{s1} & 0 \\ 0 & f_{s2} \end{bmatrix}\begin{bmatrix} \mathrm{sgn}(\dot{q}_{i1}) \\ \mathrm{sgn}(\dot{q}_{i4}) \end{bmatrix} \end{aligned} \tag{4.71}$$

where the mechanical parameters had been experimentally determined as $p_1 = 0.6786$ $\mathrm{kg}\cdot\mathrm{m}^2$, $p_2 = 0.1539$ $\mathrm{kg}\cdot\mathrm{m}^2$, $p_3 = 0.0668$ $\mathrm{kg}\cdot\mathrm{m}^2$, $f_{d1} = f_{d2} = 8$ $\mathrm{Nm}\cdot\mathrm{sec}$, and $f_{s1} = f_{s2} = 8$ Nm. It should be noted since a planar experimental setup is utilized, the gravitational effects are ignored.

MIF Controller Experiment Results

An experiment was performed for the MIF controller given in (4.26) and (4.34). The dynamic model introduced in (4.71) was utilized for both the master and slave systems. The target system, described by (4.20) and (4.21), is re-defined as follows

$$\dot{\xi}_p = \gamma\varphi\left(\xi_p\right) + \eta_d \tag{4.72}$$

$$M_T\dot{\eta}_d + B_T\eta_d + K_T\lambda_d = F_H + F_E \tag{4.73}$$

$$\lambda_d = \xi_p - \gamma\int_{t_0}^{t}\varphi^T\left(\xi_p(\tau)\right)d\tau \tag{4.74}$$

where $M_T = B_T = K_T = I_2$ with $I_2 \in \mathbb{R}^{2\times 2}$ being the standard identity matrix. The joint-space environmental input force, $F_E(t) \in \mathbb{R}^2$, was set equal to the following time-varying

signal

$$F_E = 0.1 M_2 \begin{bmatrix} -\sin(t) \\ -\cos(t) \end{bmatrix} \tag{4.75}$$

where $M_2(t) \in \mathbb{R}^{2\times 2}$ is defined in (4.71), and the joint-space user input force, $F_H(t) \in \mathbb{R}^2$, was read from the ATI force/torque sensor. The target system signals $\xi_p(t)$, $\eta_d(t)$, $\lambda_d(t) \in \mathbb{R}^2$ are of the following form

$$\xi_p = \begin{bmatrix} \xi_{px} \\ \xi_{py} \end{bmatrix} \quad , \quad \eta_d = \begin{bmatrix} \eta_{dx} \\ \eta_{dy} \end{bmatrix} \quad , \quad \lambda_d = \begin{bmatrix} \lambda_{dx} \\ \lambda_{dy} \end{bmatrix}.$$

The planar task-space velocity field function, denoted by $\varphi(\cdot) \in \mathbb{R}^2$, is defined as follows [2]

$$\begin{aligned} \varphi(\xi_p) \triangleq & -2K(\xi_p) f(\xi_p) \begin{bmatrix} \xi_{px} - r_x \\ \xi_{py} - r_y \end{bmatrix} \\ & + 2c(\xi_p) \begin{bmatrix} -(\xi_{py} - r_y) \\ \xi_{px} - r_x \end{bmatrix} \end{aligned} \tag{4.76}$$

where $r_x = 0.5$ [m] and $r_y = -0.2$ [m] denote the position of the circle center, and the functions $f(\cdot)$, $K(\cdot)$, $c(\cdot) \in \mathbb{R}$ are defined as follows

$$\begin{aligned} f(\xi_p) &\triangleq (\xi_{px} - r_x)^2 + (\xi_{py} - r_y)^2 - r_o^2 \\ K(\xi_p) &\triangleq k_o \left(\sqrt{f^2(\xi_p)} \left\| \frac{\partial f(\xi_p)}{\partial \xi_p} \right\| + \varepsilon \right)^{-1} \\ c(\xi_p) &\triangleq \frac{c_o \exp\left(-\mu \sqrt{f^2(\xi_p)} \right)}{\left\| \frac{\partial f(\xi_p)}{\partial \xi_p} \right\|}. \end{aligned} \tag{4.77}$$

In (4.77), $r_o = 0.2$ [m] denotes the circle radius, $k_o = 0.2$ [ms^{-1}], $\varepsilon = 0.005$ [m^3], $c_o = 0.2$ [ms^{-1}], and $\mu = 20$ [m^{-1}] were selected for this experiment. In (4.72) and

(4.74), γ is a constant gain that is either 0 or 1. It should be noted that, when $\gamma = 0$, the user assist mechanism is disabled, and when $\gamma = 1$, then the user assist mechanism is enabled. The controller gains were selected as $k_s = 10,\ \beta_1 + \beta_2 = 3,\ \alpha_1 = 1.5,$ and $\alpha_2 = 1.$

The desired end-effector position $\xi_p(t),$ when the user assist mechanism is disabled (*i.e.*, $\gamma = 0$) and when the user assist mechanism is enabled (*i.e.*, $\gamma = 1$), are presented in Figure 4.2. From Figure 4.2, it is clear that a circular desired trajectory is not created when the user assist mechanism is disabled. However, when the user assist mechanism is enabled (*i.e.*, $\gamma = 1$), then a circular desired trajectory can be created. The end-effector positions for the master and the slave systems are given in Figures 3 and 4, respectively.

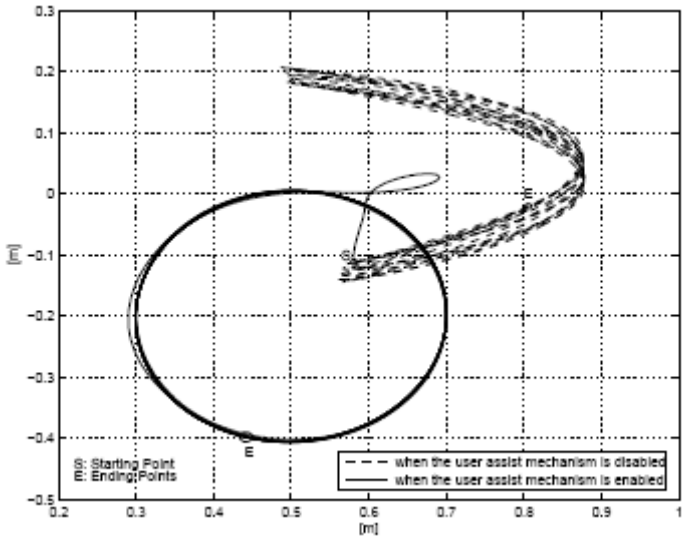

Figure 4.2: Desired end-effector position $\xi_p(t)$

The master system tracking error $e_{11}(t)$ and coordination error $e_{12}(t)$ are presented in Figures 4.5 and 4.6, respectively. From Figures 4.5 and 4.6, it is clear that tracking and coordination control objectives defined in (4.4) and (4.5), are met. The control inputs for the

master system $T_1(t)$ and the slave system $T_2(t)$ are provided in Figures 4.7 and 4.8, respectively

UMIF Controller Experiment Results

An experiment was performed for the UMIF controller given in (4.54) and (4.57). The dynamic model introduced in (4.71) was utilized for both the master and slave systems. The target system, described by (4.48) and (4.49), is re-defined as follows

$$\dot{\xi}_d = \gamma \left[\varphi^T(\xi_1) \quad 0_2^T \right]^T + \eta_d \tag{4.78}$$

$$M_T \dot{\eta}_d + B_T \eta_d + K_T \lambda_d = \left(\bar{M} M_T^{-1} \right)^{-1} \hat{F} \tag{4.79}$$

$$\lambda_d = \xi_d - \gamma \left[\int_{t_0}^{t} \varphi^T(\xi_1(\tau)) d\tau \quad 0_2^T \right]^T \tag{4.80}$$

where $M_T = 0.5 I_4$, $B_T = K_T = 8 I_4$ with $I_4 \in \mathbb{R}^{4\times4}$ being the standard identity matrix, and $\bar{M}(\cdot) \in \mathbb{R}^{4\times4}$ was found via (4.10), $\hat{F}(t) \in \mathbb{R}^4$ is the nonlinear observer, and $0_2 \in \mathbb{R}^2$ denotes a vector of zeros. The target system signals $\xi_d(t)$, $\eta_d(t)$, $\lambda_d(t) \in \mathbb{R}^4$ are of the following form

$$\xi_d = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \quad \eta_d = \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} \quad \lambda_d = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \tag{4.81}$$

where $\xi_i(t)$, $\eta_i(t)$, $\lambda_i(t) \in \mathbb{R}^2 \ \ \forall i = 1, 2$. The desired trajectory for the master system is denoted by $\xi_1(t)$ and can be decomposed as follows

$$\xi_1 \triangleq \left[\xi_{1x} \quad \xi_{1y} \right]^T \tag{4.82}$$

The velocity field function defined in (4.76) and (4.77), was utilized with the same parameters. In (4.78) and (4.80), γ is a constant gain that is either 0 or 1. It should be noted that, when $\gamma = 0$, the user assist mechanism is disabled, and when $\gamma = 1$, then the user assist mechanism is enabled. The controller gains are selected as $k_s = 20,\ \beta_1 + \beta_2 = 8,$ and $\alpha = 22$.

The desired end-effector position $\xi_1(t),$ when the user assist mechanism is disabled (*i.e.*, $\gamma = 0$) and when the user assist mechanism is enabled (*i.e.*, $\gamma = 1$) are presented in Figure 4.9. From Figure 4.9, it is clear that a circular desired trajectory is not created when the user assist mechanism is disabled. However, when the user assist mechanism is enabled (*i.e.*, $\gamma = 1$), then a circular desired trajectory can be created. The end-effector positions for the master and the slave systems are given in Figures 4.10 and 4.11, respectively. The master system tracking error $e_{11}(t)$ and the coordination error $e_{12}(t)$ are presented in Figures 4.12 and 4.13, respectively. From Figures 4.12 and 4.13, it is clear that tracking and coordination control objectives defined in (4.40) and (4.41), are met. The control inputs for the master system $T_1(t)$ and the slave system $T_2(t)$ are provided in Figures 4.14 and 4.15, respectively. The nonlinear force observer $\hat{F}(t)$ is presented in Figure 4.16.

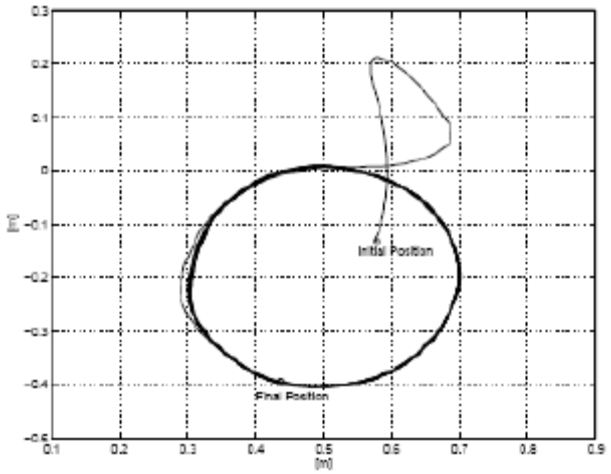

Figure 4.3: Master system end-effector position $x_m(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

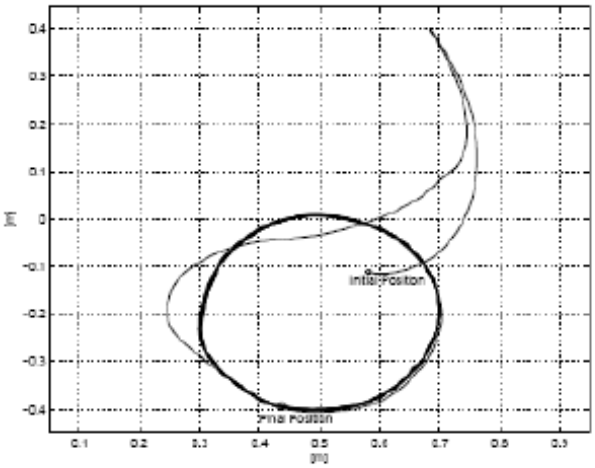

Figure 4.4: Slave system end-effector position $x_s(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

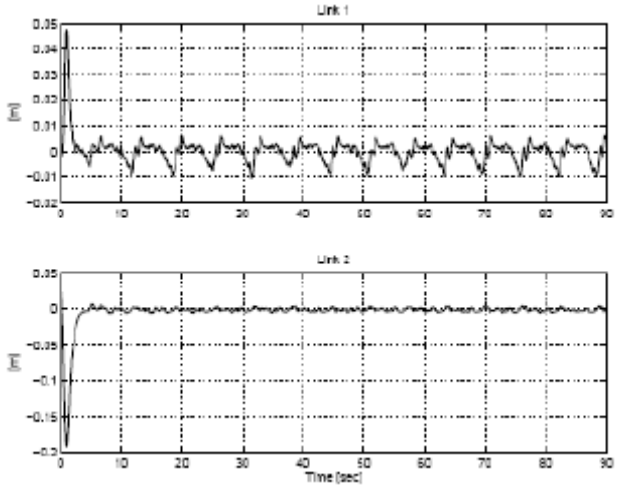

Figure 4.5: Master system tracking error $e_{11}(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

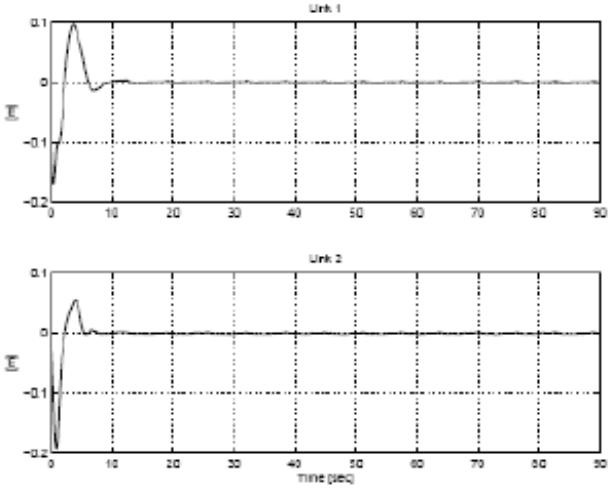

Figure 4.6: Coordination error $e_{12}(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

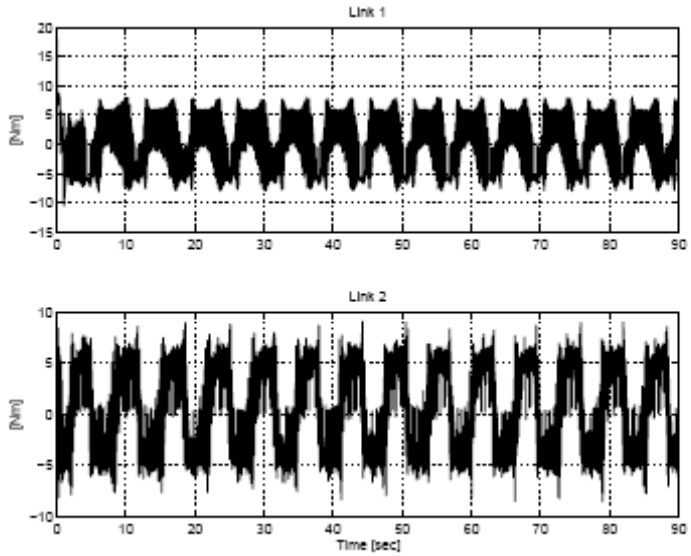

Figure 4.7: Control input for master system $T_1(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

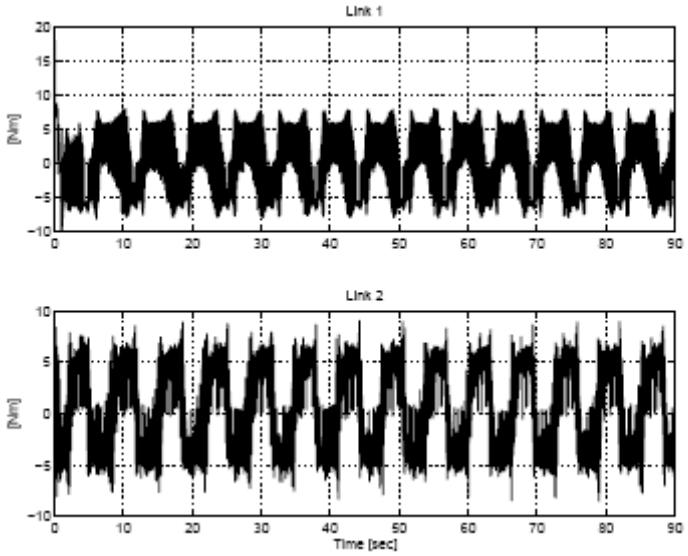

Figure 4.8: Control input for slave system $T_2(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

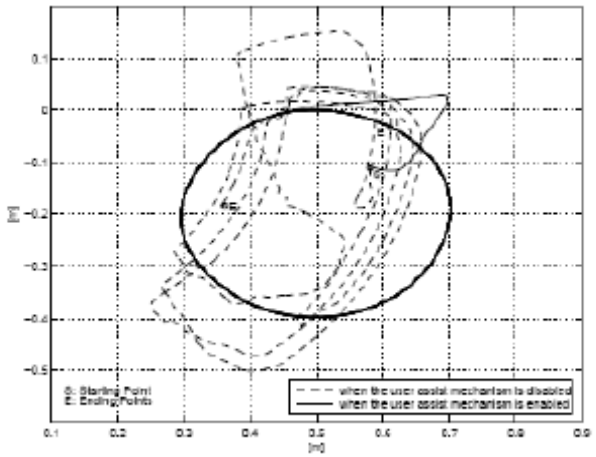

Figure 4.9: Desired end-effector position $\xi_1(t)$

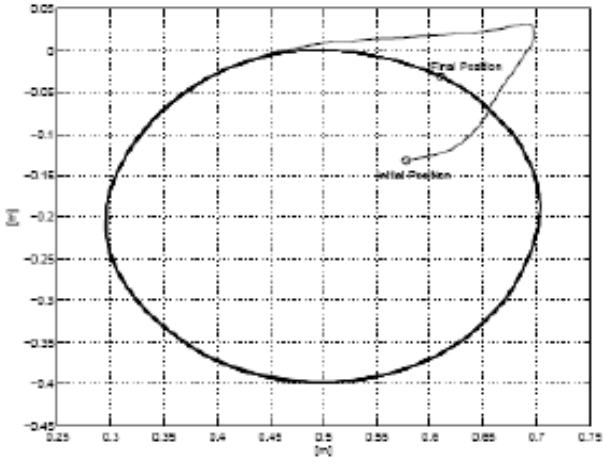

Figure 4.10: Master system end-effector position $x_m(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

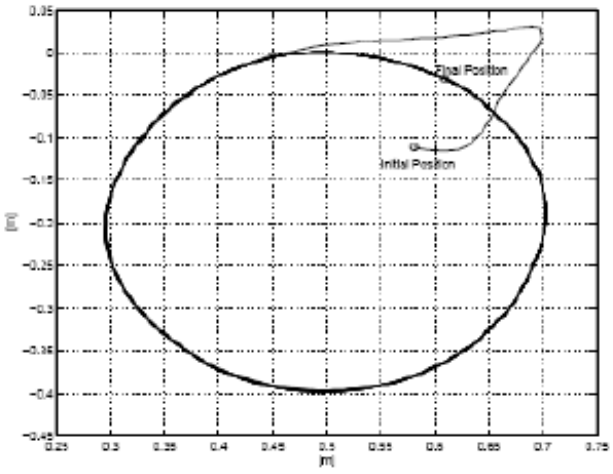

Figure 4.11: Slave system end-effector position $x_s(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

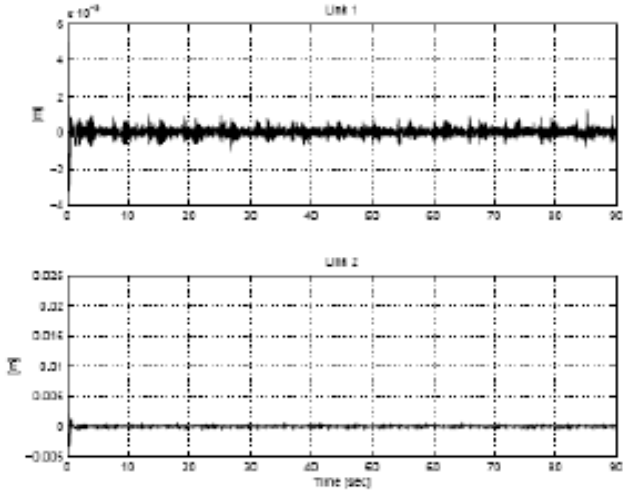

Figure 4.12: Master system tracking error $e_{11}(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

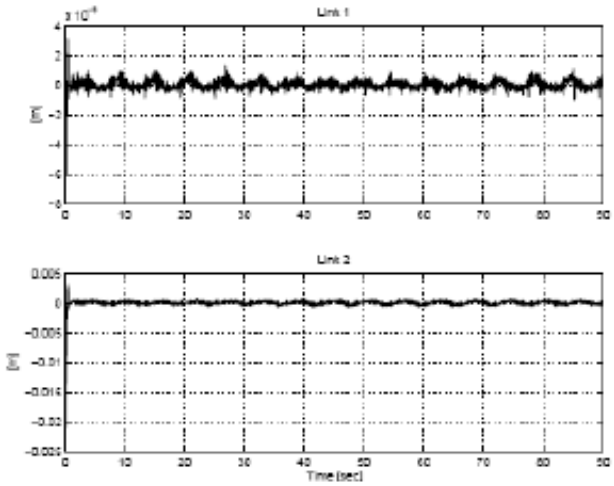

Figure 4.13: Coordination error $e_{12}(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

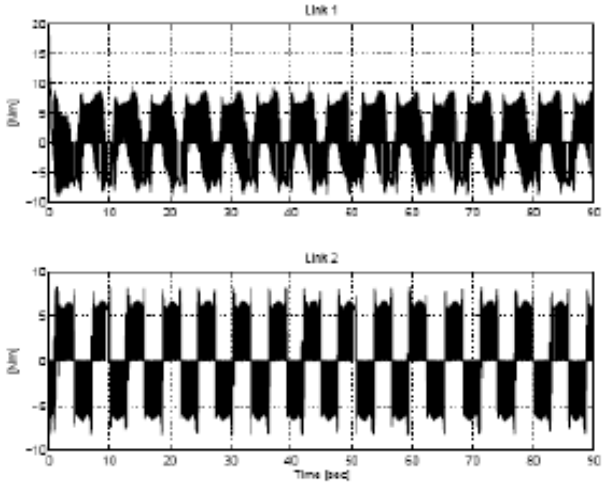

Figure 4.14: Control input for master system $T_1(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

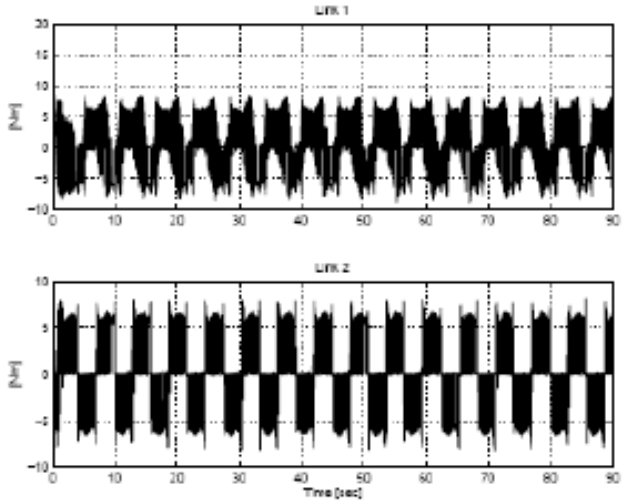

Figure 4.15: Control input for slave system $T_2(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

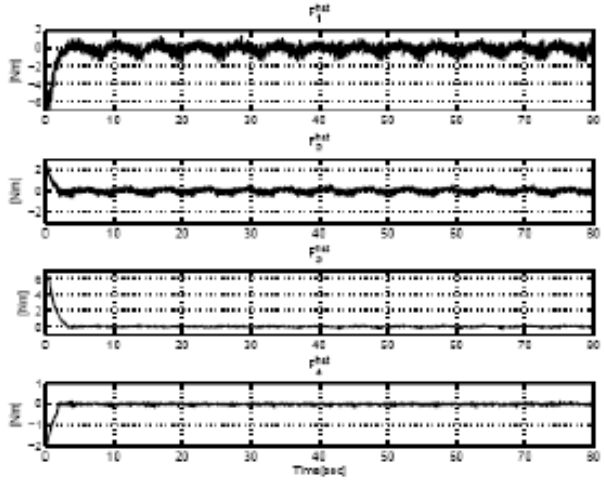

Figure 4.16: The output of the nonlinear force observer $\hat{F}(t)$ when the user assist mechanism is enabled (*i.e.,* $\gamma = 1$)

CHAPTER 5

CONCLUSION

In chapter 2, a whole arm grasping controller for redundant robot manipulators was presented. A kinematic control which enables *end-effector position* tracking information as well as *body self-motion positioning* control information to be encoded in the desired trajectory was developed. Then, a tracking controller, developed in [26], which forces the robot to track a desired trajectory in the presence of system uncertainties and unmeasurable contact forces was utilized. The controller provides asymptotic tracking which enables the whole arm grasping objective to be completed. Experimental results for a planar, three link configuration of the Barrett WAM are provided to demonstrate the controller performance. Future work will include applying the whole arm grasping technique to a hyper-redundant tentacle manipulator.

In chapter 3, a neural network controller has been presented for grasping control of a continuum robot manipulator. The feedforward neural network was used to compensate for the uncertain nonlinear dynamics of the continuum manipulator, while the nonlinear controller provides semi-global asymptotic convergence of the joint position tracking error. Experimental results for the OCTARM soft robotic manipulator operating along a sinusoidal trajectory were presented. A comparison of the tracking performance, both with and without the neural network feedforward component demonstrates the efficacy of the proposed neural network feedforward estimation technique. Future work will consist of testing the whole arm grasping controller with the neural network component on OCTARM VI.

In chapter 4, two controllers were developed for nonlinear haptic and teleoperator systems that target coordination of the master and slave. The first controller was proven to yield a semi-global asymptotic result in the presence of parametric uncertainty in the master and slave dynamic models provided the user and environmental input forces are measurable. The second controller was proven to yield a global asymptotic result despite unmeasurable user and environmental input forces provided the dynamic models of the master and slave are known. A transformation along with an adjustable target system were utilized that allows the master system's impedance to be adjusted so that matches a desired target system operating in a remote physical/virtual environment. This work also presented an optional strategy to encode a velocity field assist mechanism that provides the user of the system help in controlling the slave system in completing a pre-defined contour following task. For each controller, Lyapunov-based techniques were used to prove the control development implements a stable coordinated teleoperator/haptic system with a user assist mechanism. When the optional velocity field assist mechanism is disabled, the analysis proved the control development implements a stable passively coordinated teleoperator/haptic system. Experiment results demonstrated proof of concept for both controllers.

APPENDICES

Appendix A

MIF Desired Trajectory Stability Analysis

To prove that $\xi_d(t), \lambda_d(t), \eta_d(t), \dot{\eta}_d(t) \in \mathcal{L}_\infty$ let $V(t) \in \Re$ denote the following function

$$V \triangleq V_1 + V_2 \tag{A.1}$$

where $V_1(t) \in \Re$ denotes the following non-negative function

$$V_1 \triangleq \frac{1}{2}\eta_d^T M_T \eta_d + \frac{1}{2}\lambda_d^T K_T \lambda_d \tag{A.2}$$

where $\lambda_d(t), \eta_d(t), M_T$ and K_T were introduced in (4.21). The expression given in (A.2) can be lower bounded by the auxiliary function, $V_2(\overline{x}) \in \Re$, which is defined as follows

$$V_2 \triangleq 2\varepsilon \eta_d^T M_T \lambda_d \leq V_1 \tag{A.3}$$

where $\overline{x}(t) \in \Re^{2n}$ is defined as follows

$$\overline{x} \triangleq \begin{bmatrix} \lambda_d^T & \eta_d^T \end{bmatrix}^T \tag{A.4}$$

and $\varepsilon \in \Re$ is a positive bounding constant selected according to the following inequality

$$\varepsilon < \frac{\min\left\{\lambda_{\min}\left\{M_T\right\}, \lambda_{\min}\left\{K_T\right\}\right\}}{4\lambda_{\max}\left\{M_T\right\}} \tag{A.5}$$

where $\lambda_{\min}\{\cdot\}$ and $\lambda_{\max}\{\cdot\}$ denote the minimum and maximum eigenvalue of a matrix, respectively. From (A.3) it is clear that $V(t)$ is a non-negative function and bounded by the following inequalities

$$\underline{\lambda}_1 \left\|\overline{x}\right\|^2 \leq V\left(\overline{x}\right) \leq \overline{\lambda}_2 \left\|\overline{x}\right\|^2 \tag{A.6}$$

where $\bar{\lambda}_1, \bar{\lambda}_2 \in \Re$ are positive bounding constants defined as follows, provided that ε is selected according to (A.5)

$$\bar{\lambda}_1 \triangleq \frac{1}{2}\min\left\{\lambda_{\min}\left\{M_T\right\}, \lambda_{\min}\left\{K_T\right\}\right\} - 2\varepsilon\lambda_{\max}\left\{M_T\right\} \tag{A.7}$$

$$\bar{\lambda}_2 \triangleq \frac{1}{2}\max\left\{\lambda_{\max}\left\{M_T\right\}, \lambda_{\max}\left\{K_T\right\}\right\} - 2\varepsilon\lambda_{\max}\left\{M_T\right\}$$

To facilitate the subsequent analysis, the time derivative of (A.1) can be determined as follows

$$\dot{V} = \eta_d^T MT\dot{\eta}_d + \lambda_d^T K_T \dot{\lambda}_d + 2\varepsilon\dot{\eta}_d^T M_T \lambda_d + 2\varepsilon\eta_d^T M_T \dot{\lambda}_d \tag{A.8}$$

After utilizing (4.21) and the fact that $\eta_d(t) = \dot{\lambda}_d(t)$, the expression in (A.8) can be written as

$$\begin{aligned} \dot{V} = {} & \eta_d^T F - \eta_d^T B_T \eta_d + 2\varepsilon\lambda_d^T F - 2\varepsilon\lambda_d^T B_T \eta_d \\ & - 2\varepsilon\lambda_d^T K_T \lambda_d + 2\varepsilon\eta_d^T M_T \eta_d \end{aligned} \tag{A.9}$$

The right-hand side of (A.9) can be upper bounded as follows

$$\begin{aligned} \dot{V} \le {} & \frac{1}{\delta_1}\left\|\eta_d\right\|^2 + \delta_1\left\|F\right\|^2 - \lambda_{\min}\left\{B_T\right\}\left\|\eta_d\right\|^2 \\ & + 2\varepsilon\left[\delta_2\left\|\lambda_d\right\|^2 + \frac{1}{\delta_2}\left\|F\right\|^2\right] \\ & + 2\varepsilon\lambda_{\max}\left\{B_T\right\}\left[\delta_3\left\|\lambda_d\right\|^2 + \frac{1}{\delta_3}\left\|\eta_d\right\|^2\right] \\ & - 2\varepsilon\lambda_{\min}\left\{K_T\right\}\left\|\lambda_d\right\|^2 + 2\varepsilon\lambda_{\max}\left\{M_T\right\}\left\|\eta_d\right\|^2 \end{aligned} \tag{A.10}$$

where the following properties were utilized

$$
\begin{aligned}
\eta_d^T F &\leq \frac{1}{\delta_1}\|\eta_d\|^2 + \delta_1\|F\|^2 \\
-\eta_d^T B_T \eta_d &\leq -\lambda_{\min}\{B_T\}\|\eta_d\|^2 \\
\lambda_d^T F &\leq \delta_2\|\lambda_d\|^2 + \frac{1}{\delta_2}\|F\|^2 \\
-\lambda_d^T B_T \eta_d &\leq \lambda_{\max}\{B_T\}\left[\delta_3\|\lambda_d\|^2 + \frac{1}{\delta_3}\|\eta_d\|^2\right] \\
-\lambda_d^T K_T \lambda_d &\leq -\lambda_{\min}\{K_T\}\|\lambda_d\|^2 \\
\eta_d^T M_T \eta_d &\leq \lambda_{\max}\{M_T\}\|\eta_d\|^2
\end{aligned}
$$

where $\delta_1, \delta_2, \delta_3 \in \mathbb{R}$ are positive bounding constants.

The expression in (A.10) can be rearranged as follows

$$
\begin{aligned}
\dot{V} \leq & -(\lambda_{\min}\{B_T\} - \frac{1}{\delta_1} - \frac{2\varepsilon\lambda_{\max}\{B_T\}}{\delta_3} \\
& -2\varepsilon\lambda_{\max}\{M_T\})\|\eta_d\|^2 \\
& -2\varepsilon\left(\lambda_{\min}\{K_T\} - \delta_3\lambda_{\max}\{B_T\} - \delta_2\right)\|\lambda_d\|^2 \\
& +\left(\delta_1 + \frac{2\varepsilon}{\delta_2}\right)\|F\|^2 .
\end{aligned}
\tag{A.11}
$$

Provided that ε is selected to satisfy (A.5) and δ_1, δ_2, δ_3, M_T, B_T, K_T are selected to satisfy the following sufficient conditions

$$
\lambda_{\min}\{B_T\} > \frac{1}{\delta_1} + \frac{2\varepsilon\lambda_{\max}\{B_T\}}{\delta_3} + 2\varepsilon\lambda_{\max}\{M_T\} \tag{A.12}
$$

$$
\lambda_{\min}\{K_T\} > \delta_3\lambda_{\max}\{B_T\} + \delta_2 \tag{A.13}
$$

along with the Assumption 1, then the right-hand side of (A.11) can be upper bounded as follows

$$
\dot{V} \leq -\frac{\min\{\gamma_a, \gamma_b\}}{\bar{\lambda}_2}V + \varepsilon \tag{A.14}
$$

where (A.4) and (A.6) were utilized, and γ_a, γ_b, $\varepsilon \in \mathbb{R}$ denote positive bounding constants.

From (A.1) - (A.3), and (A.6), and the fact that $\mathrm{F}(t) \in \mathcal{L}_\infty$, the expression in (A.14) can be used with the result from [49] to prove that $\overline{x}(t)$, $\lambda_d(t)$, $\eta_d(t) \in \mathcal{L}_\infty$. By utilizing the fact that $\eta_d(t) \in \mathcal{L}_\infty$ along with (4.20) and Remark 2, it is clear that $\xi_d(t)$, $\dot{\xi}_d(t)$, $\varphi\left(\xi_p(t)\right) \in \mathcal{L}_\infty$. Based on (4.21), and the fact that $F(t) \in \mathcal{L}_\infty$ then $\dot{\eta}_d(t) \in \mathcal{L}_\infty$. After utilizing the above boundedness statements along with Remark 2 and the first time derivative of (4.20), it is clear that $\ddot{\xi}_d(t) \in \mathcal{L}_\infty$. The time derivative of (4.21) can be written as follows

$$M_T \ddot{\eta}_d + B_T \dot{\eta}_d + K_T \eta_d = \dot{F} \tag{A.15}$$

where the fact $\eta_d(t) = \dot{\lambda}_d(t)$ was utilized. After utilizing the fact that $\eta_d(t)$, $\dot{\eta}_d(t) \in \mathcal{L}_\infty$, and the assumption that $\dot{F}_H(t)$, $\dot{F}_E(t) \in \mathcal{L}_\infty$ along with (A.15), it is clear that $\ddot{\eta}_d(t) \in \mathcal{L}_\infty$. The second time derivative of (4.20) can be written as follows

$$\dddot{\xi}_d \triangleq \gamma \frac{d^2}{dt^2}\left(\begin{bmatrix} \varphi^T\left(\xi_p\right) & 0_r^T \end{bmatrix}^T\right) + \ddot{\eta}_d. \tag{A.16}$$

After utilizing the above boundedness statements and Remark 2 along with (A.16), then $\dddot{\xi}_d(t) \in \mathcal{L}_\infty$. The time derivative of (A.15) can be written as follows

$$M_T \dddot{\eta}_d + B_T \ddot{\eta}_d + K_T \dot{\eta}_d = \ddot{F} \tag{A.17}$$

After utilizing the fact that $\dot{\eta}_d(t)$, $\ddot{\eta}_d(t) \in \mathcal{L}_\infty$ and the assumption that $\ddot{F}_H(t)$, $\ddot{F}_E(t) \in \mathcal{L}_\infty$, from (A.17) it can be showed that $\dddot{\eta}_d(t) \in \mathcal{L}_\infty$. After taking time derivative of (A.16) and utilizing the facts that $\xi_d(t)$, $\dot{\xi}_d(t)$, $\ddot{\xi}_d(t)$, $\dddot{\xi}_d(t)$, $\dddot{\eta}_d(t) \in \mathcal{L}_\infty$, then it is clear that

$\dddot{\xi}_d(t) \in \mathcal{L}_\infty$. By utilizing the above boundedness statements along with (4.19), it is clear that $x_d(t)$, $\dot{x}_d(t)$, $\ddot{x}_d(t)$, $\dddot{x}_d(t)$, and $\ddddot{x}_d(t) \in \mathcal{L}_\infty$.

Appendix B

Proof of Theorem 1

Lemma 1 *Let the auxiliary functions* $L_1(t),\ L_2(t) \in \mathbb{R}$ *be defined as follows*

$$\begin{aligned} L_1 &\triangleq r^T\left(N_d - \beta_1 sgn(e_2)\right) \\ L_2 &\triangleq -\beta_2 \dot{e}_2^T sgn(e_2) \end{aligned} \tag{B.1}$$

where β_1 *and* β_2 *were introduced in (4.34). Provided that* β_1 *is selected to satisfy the following sufficient condition*

$$\beta_1 > \varsigma_1 + \frac{1}{\alpha_1}\varsigma_2 \tag{B.2}$$

where ς_1 *and* ς_2 *were introduced in (4.33), and* α_1 *was introduced in (4.15), then*

$$\int_{t_0}^{t} L_1(\tau)\,d\tau \le \xi_{b1} \qquad \int_{t_0}^{t} L_2(\tau)\,d\tau \le \xi_{b2} \tag{B.3}$$

where $\xi_{b1},\ \xi_{b2} \in \mathbb{R}$ *are positive constants defined as*

$$\begin{aligned} \xi_{b1} &\triangleq \beta_1 \sum_{i=1}^{2n} \left|e_{2i}(t_0)\right| - e_2^T(t_0) N_d(t_0) \\ \xi_{b2} &\triangleq \beta_2 \sum_{i=1}^{2n} \left|e_{2i}(t_0)\right|. \end{aligned} \tag{B.4}$$

Proof. After substituting (4.15) into $L_1(t)$ defined in (B.1) and then integrating in time, results in the following expression

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &= \alpha_1 \int_{t_0}^{t} e_2^T(\tau)\left[N_d(\tau) - \beta_1 sgn\left(e_2(\tau)\right)\right] d\tau \\ &+ \int_{t_0}^{t} \frac{de_2^T(\tau)}{d\tau} N_d(\tau)\,d\tau \\ &- \beta_1 \int_{t_0}^{t} \frac{de_2^T(\tau)}{d\tau} sgn\left(e_2(\tau)\right) d\tau.\end{aligned} \tag{B.5}$$

After integrating the second integral on the right side of (B.5) by parts and evaluating the last integral, the following expression is obtained

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &= \alpha_1 \int_{t_0}^{t} e_2^T \left(N_d - \frac{1}{\alpha_1}\frac{dN_d}{d\tau} \right. \\ &\left. - \beta_1 sgn\left(e_2\right)\right) d\tau + e_2^T(t) N_d(t) \\ &- \beta_1 \sum_{i=1}^{2n} \left|e_{2i}(t)\right| + \xi_{b1}.\end{aligned} \tag{B.6}$$

The right-hand side of (B.6) can be upper bounded as follows

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &\leqslant \alpha_1 \int_{t_0}^{t} \sum_{i=1}^{2n} \left|e_{2i}(\tau)\right| \left(\left|N_{d_i}(\tau)\right| \right. \\ &\left. + \frac{1}{\alpha_1}\left|\frac{dN_{d_i}(\tau)}{d\tau}\right| - \beta_1 \right) d\tau \\ &+ \sum_{i=1}^{2n} \left|e_{2i}(t)\right| \left(\left|N_{d_i}(t)\right| - \beta_1 \right) + \xi_{b1}.\end{aligned} \tag{B.7}$$

If β_1 is chosen according to (4.39), then the first inequality in (B.3) can be proven from (B.7). The second inequality in (B.3) can be obtained by integrating the expression for $L_2(t)$ defined in (B.1) as follows

$$\begin{aligned}\int_{t_0}^{t} L_2(\tau)\,d\tau &= -\beta_2 \int_{t_0}^{t} \dot{e}_2^T(\tau) sgn\left(e_2(\tau)\right) d\tau \\ &= \xi_{b2} - \beta_2 \sum_{i=1}^{2n} \left|e_{2i}(t)\right| \leq \xi_{b2}.\end{aligned} \tag{B.8}$$

The following is the proof of Theorem 1.

Proof. Let the auxiliary functions $P_1(t)$, $P_2(t) \in \mathbb{R}$ be defined as follows

$$P_1 \triangleq \xi_{b1} - \int_{t_0}^{t} L_1(\tau)\, d\tau \geq 0 \tag{B.9}$$

$$P_2 \triangleq \xi_{b2} - \int_{t_0}^{t} L_2(\tau)\, d\tau \geq 0 \tag{B.10}$$

where $L_1(t)$, $L_2(t)$, ξ_{b1} and ξ_{b2} were defined in Lemma 1. The proof of Lemma 1 ensures that $P_1(t)$ and $P_2(t)$ are non-negative. Let $V(y,t) \in \mathbb{R}$ denote the following non-negative function

$$V \triangleq \frac{1}{2} e_1^T e_1 + \frac{1}{2} e_2^T e_2 + \frac{1}{2} r^T \bar{M} r + P_1 + P_2 \tag{B.11}$$

where $y(t) \in \mathbb{R}^{6n+2}$ is defined as follows

$$y \triangleq \begin{bmatrix} z^T & \sqrt{P_1} & \sqrt{P_2} \end{bmatrix}^T \tag{B.12}$$

where $z(t) \in \mathbb{R}^{6n}$ is defined as follows

$$z \triangleq \begin{bmatrix} e_1^T & e_2^T & r^T \end{bmatrix}^T. \tag{B.13}$$

Because $\bar{M}(x)$ is assumed to be bounded as defined in (4.14), (B.11) is bounded as follows

$$W_1(y) \leq V(y,t) \leq W_2(y) \tag{B.14}$$

where $W_1(y)$, $W_2(y) \in \mathbb{R}$ are defined as

$$W_1(y) \triangleq \lambda_1 \left\| y(t) \right\|^2 \quad W_2(y) \triangleq \lambda_2 \left\| y(t) \right\|^2 \tag{B.15}$$

where $\lambda_1 \triangleq \frac{1}{2} \min\{1, \bar{m}_1\}$ and $\lambda_2 \triangleq \max\left\{1, \frac{1}{2}\bar{m}_2\right\}$.

After differentiating (B.11) in time, the following expression can be obtained

$$\begin{aligned}\dot{V} &= -\alpha_2 e_1^T e_1 - \alpha_1 e_2^T e_2 - r^T\left(k_s+1\right)r \\ &\quad + e_1^T e_2 + r^T\tilde{N} - r^T\beta_2 sgn\left(e_2\right) + \beta_2 \dot{e}_2^T sgn\left(e_2\right)\end{aligned} \tag{B.16}$$

where (4.15), (4.16), (4.36), (B.9), and (B.10) were utilized. To facilitate the subsequent analysis, the following inequality can be developed from (4.30) - (4.32) (see Appendix G)

$$\left\|\tilde{N}\left(\cdot\right)\right\| \le \rho\left(\left\|z\right\|\right)\left\|z\right\| \tag{B.17}$$

where $\rho\left(\cdot\right)$ is a positive, invertible bounding function that is non-decreasing in $\left\|z\right\|$. By utilizing (4.15), (B.17), and the triangle inequality, $\dot{V}\left(t\right)$ can be upper bounded as follows

$$\begin{aligned}\dot{V} &\le -\alpha_2 e_1^T e_1 - \alpha_1 e_2^T e_2 - r^T\left(k_s+1\right)r \\ &\quad + e_1^T e_1 + e_2^T e_2 + \rho\left(\left\|z\right\|\right)\left\|r\right\|\left\|z\right\| \\ &\quad - \alpha_1 e_2^T\beta_2 sgn\left(e_2\right).\end{aligned} \tag{B.18}$$

After utilizing (B.13), the right-hand side of (B.18) can be rearranged as follows

$$\begin{aligned}\dot{V} &\le -\lambda_3\left\|z\right\|^2 + \left[\rho\left(\left\|z\right\|\right)\left\|r\right\|\left\|z\right\| - k_s\left\|r\right\|^2\right] \\ &\quad - \alpha_1\beta_2\sum_{i=1}^{2n}\left|e_{2i}\right|\end{aligned} \tag{B.19}$$

where $\lambda_3 \triangleq \min\left\{\alpha_1 - 1, \alpha_2 - 1, 1\right\}$. Completing the squares on the bracketed term in (B.19), yeilds the follwing expression

$$\dot{V} \le -\left(\lambda_3 - \frac{\rho^2\left(\left\|z\right\|\right)}{4k_s}\right)\left\|z\right\|^2 - \alpha_1\beta_2\sum_{i=1}^{2n}\left|e_{2i}\right|. \tag{B.20}$$

Provided α_1 and α_2 are selected to be greater than 2 and k_s is selected according to the following sufficient condition

$$k_s \ge \frac{\rho^2\left(\left\|z\right\|\right)}{4\lambda_3} \text{ or } \left\|z\right\| \le \rho^{-1}\left(2\sqrt{k_s\lambda_3}\right) \tag{B.21}$$

then based on (B.20) the following inequality can be developed

$$\dot{V} \le W(y) - \alpha_1 \beta_2 \sum_{i=1}^{2n} |e_{2i}| \tag{B.22}$$

where $W(y) \in \mathbb{R}$ denotes the following non-positive function

$$W(y) \triangleq -\beta_0 \|z\|^2 \tag{B.23}$$

where $\beta_0 \in \mathbb{R}$ is a positive constant. From (B.11)-(B.15) and (B.20)-(B.23) the regions D and S can be defined as follows

$$\mathcal{D} \triangleq \left\{ y \in \mathbb{R}^{6n+2} \mid \|y\| < \rho^{-1}\left(2\sqrt{k_s \lambda_3}\right) \right\} \tag{B.24}$$

$$\mathcal{S} \triangleq \left\{ y \in \mathcal{D} \mid W_2(y) < \lambda_1 \left(\rho^{-1}\left(2\sqrt{k_s \lambda_3}\right) \right)^2 \right\}. \tag{B.25}$$

Note that the region of attraction in (B.25) can be made arbitrarily large to include any initial conditions by increasing the control gain k_s (i.e., a semi-global stability result). Specifically, (B.15) and (B.25) can be used to calculate the region of attraction as follows

$$\begin{aligned} W_2(y(t_0)) \quad &< \quad \lambda_1 \left(\rho^{-1}\left(2\sqrt{k_s \lambda_3}\right) \right)^2 \\ \Rightarrow \quad \|y(t_0)\| &< \sqrt{\frac{\lambda_1}{\lambda_2}} \rho^{-1}\left(2\sqrt{k_s \lambda_3}\right), \end{aligned} \tag{B.26}$$

which can be rearranged as

$$k_s \ge \frac{1}{4\lambda_3} \rho^2 \left(\sqrt{\frac{\lambda_2}{\lambda_1}} \|y(t_0)\| \right). \tag{B.27}$$

By utilizing (B.4), (B.12) and (B.13) the following explicit expression for $\|y(t_0)\|$ can be derived as follows

$$\begin{aligned} \left\| y(t_0) \right\|^2 &= \left\| e_1(t_0) \right\|^2 + \left\| e_2(t_0) \right\|^2 \\ &+ \left\| r(t_0) \right\|^2 + \xi_{b1} + \xi_{b2}. \end{aligned} \tag{B.28}$$

From (B.11), (B.22), (B.25)-(B.27), it is clear that $V(y,t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$; hence $e_1(t)$, $e_2(t)$, $r(t)$, $z(t)$, $y(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. From (B.22), it is easy to show that $e_2(t) \in \mathcal{L}_1$ $\forall y(t_0) \in \mathcal{S}$. The fact that $e_2(t) \in \mathcal{L}_1 \ \ \forall y(t_0) \in \mathcal{S}$ can be used along with (4.16) to determine that $e_1(t), \dot{e}_1(t) \in \mathcal{L}_1 \ \ \forall y(t_0) \in \mathcal{S}$. From (4.7), (4.17) and the fact that $x_d(t) \in \mathcal{L}_\infty$, it is clear that $x(t)$, $x_m(t)$, $x_s(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. From (4.15) and (4.16) it is also clear that $\dot{e}_2(t)$, $\dot{e}_1(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. Using these boundedness statements, from (4.35) it is clear that $\dot{\bar{u}}(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. Since $\ddot{e}_1(t) \in \mathcal{L}_\infty$, from the second time derivative of (4.17), and the fact that $\ddot{x}_d(t) \in \mathcal{L}_\infty$ along with (4.27), it is clear that $\bar{u}(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. The previous boundedness statements can be used along with (4.36), (B.17), and Remark 4 to prove that $\dot{r}(t) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in \mathcal{S}$. These bounding statements can be used along with the time derivative of (B.23) to prove that $\dot{W}(y(t)) \in \mathcal{L}_\infty \ \ \forall y(t_0) \in S$; hence, $W(y(t))$ is uniformly continuous. Standard signal chasing arguments can be used to prove that all remaining signals are bounded. A direct application of Theorem 8.4 in [55] can be used to prove that $\left\| z(t) \right\| \to 0$ as $t \to \infty \ \ \forall y(t_0) \in \mathcal{S}$. From (B.13), it is clear that $\left\| r(t) \right\| \to 0$ as $t \to \infty \ \ \forall y(t_0) \in \mathcal{S}$. Based on the definitions given in (4.15) and (4.16), standard linear analysis tools can be used to prove that if $\left\| r(t) \right\| \to 0$ then $\left\| \dot{e}_2(t) \right\|$, $\left\| e_2(t) \right\|$, $\left\| \dot{e}_1(t) \right\|$, $\left\| e_1(t) \right\| \to 0$ as $t \to \infty \ \ \forall y(t_0) \in \mathcal{S}$. Based on the definition of $x(t)$ in (4.7) and $e_1(t)$ in

(4.17), it is clear that if $\|e_1(t)\| \to 0$ then $x_s(t) \to x_m(t)$ and $x_m(t) \to \xi_d(t)$.

Appendix C

Proof of Theorem 2

Proof. Since the user assist mechanism is disabled (i.e., $\gamma = 0$), the target system defined in (4.20) and (4.21) can be simplified to (4.25). Let $V_p(t) \in \mathbb{R}$ denote the following non-negative function

$$V_p \triangleq \frac{1}{2}\dot{\xi}_d^T M_T \dot{\xi}_d + \frac{1}{2}\xi_d^T K_T \xi_d. \tag{C.1}$$

After differentiating (C.1) in time, the following simplified expression can be obtained

$$\dot{V}_p = \dot{\xi}_d^T F - \dot{\xi}_d^T B_T \dot{\xi}_d \tag{C.2}$$

where (4.25) was utilized. Based on the fact that B_T is a constant positive definite, diagonal matrix, the following inequality can be obtained

$$\dot{V}_p \leq \dot{\xi}_d^T F. \tag{C.3}$$

Integrating both sides of (C.3), results in the following inequality

$$-c_2 \leq V_p(t) - V_p(t_0) \leq \int_{t_0}^{t} \dot{\xi}_d^T(\sigma) F(\sigma) d\sigma \tag{C.4}$$

where $c_2 \in \mathbb{R}$ is a positive bounded constant (since $V_p(t)$ is bounded from the trajectory generation system in (4.25)).

By using the transformation in (4.7), the left-hand side of (4.6) can be expressed as

$$\begin{aligned} &\int_{t_0}^{t} \begin{bmatrix} \dot{x}_m^T(\tau) & \dot{x}_s^T(\tau) \end{bmatrix} \begin{bmatrix} F_H(\tau) \\ F_E(\tau) \end{bmatrix} d\tau \\ &= \int_{t_0}^{t} \dot{x}^T \bar{F} d\tau. \end{aligned} \tag{C.5}$$

By substituting the time derivative of (4.17) into (C.5), the following expression can be

obtained

$$\begin{aligned}\int_{t_0}^{t} \dot{x}^T(\tau)\bar{F}(\tau)d\tau &= \int_{t_0}^{t} \dot{\xi}_d^T(\tau)F(\tau)d\tau \\ &\quad -\int_{t_0}^{t} \dot{e}_1^T(\tau)\bar{F}(\tau)d\tau\end{aligned} \tag{C.6}$$

where (4.13),(4.19) and (4.22) were utilized. Based on (C.4), it is clear that $\int_{t_0}^{t} \dot{\xi}_d^T(\tau)F(\tau)d\tau$ is lower bounded by $-c_2$. The fact that $\dot{e}_1(t) \in \mathcal{L}_1$ (see the proof for Theorem 1) and the assumption that $\bar{F}(t) \in \mathcal{L}_\infty$ can be used to show that the second integral of (C.6) is bounded. Hence, these facts can be applied to (C.5) and (C.6) to prove that

$$\int_{t_0}^{t} \begin{bmatrix} \dot{x}_m^T(\tau) & \dot{x}_s^T(\tau) \end{bmatrix} \begin{bmatrix} F_H(\tau) \\ F_E(\tau) \end{bmatrix} d\tau \geq -c_3^2 \tag{C.7}$$

where $c_3 \in \mathbb{R}$ is a bounded constant.

Appendix D

Proof of Theorem 3

Lemma 2 *Let the auxiliary functions* $L_1(t), L_2(t) \in \mathbb{R}$ *be defined as follows*

$$\begin{aligned} L_1 &\triangleq -r^T\left(\dot{\bar{F}} + \beta_1 sgn(e_2)\right) \\ L_2 &\triangleq -\beta_2 \dot{e}_2^T sgn(e_2) \end{aligned} \tag{D.1}$$

where β_1 *and* β_2 *were introduced in (4.61). Provided that* β_1 *is selected to satisfy the following sufficient condition*

$$\beta_1 > \zeta_3 + \zeta_4, \tag{D.2}$$

where ζ_3 *and* ζ_4 *were introduced in (4.64), then*

$$\int_{t_0}^{t} L_1(\tau)\, d\tau \le \xi_{b1} \qquad \int_{t_0}^{t} L_2(\tau)\, d\tau \le \xi_{b2} \tag{D.3}$$

where ξ_{b1}, $\xi_{b2} \in \mathbb{R}$ *are positive constants defined as*

$$\begin{aligned} \xi_{b1} &\triangleq \beta_1 \sum_{i=1}^{2n} \left|e_{2i}(t_0)\right| - e_2^T(t_0)\left(-\dot{\bar{F}}(t_0)\right) \\ \xi_{b2} &\triangleq \beta_2 \sum_{i=1}^{2n} \left|e_{2i}(t_0)\right|. \end{aligned} \tag{D.4}$$

Proof. After substituting (4.44) into $L_1(t)$ defined in (D.1) and then integrating in time, results in the following expression

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &= \int_{t_0}^{t} e_2^T(\tau)\Big[-\dot{\bar{F}}(\tau) \\ &\quad -\beta_1 sgn\big(e_2(\tau)\big)\Big]d\tau \\ &\quad +\int_{t_0}^{t}\frac{de_2^T(\tau)}{d\tau}\Big(-\bar{\bar{F}}(\tau)\Big)d\tau \\ &\quad -\beta_1\int_{t_0}^{t}\frac{de_2^T(\tau)}{d\tau}sgn\big(e_2(\tau)\big)d\tau.\end{aligned} \tag{D.5}$$

After integrating the second integral on the right-hand side of (D.5) by parts and evaluating the last integral, the following expression is obtained

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &= \int_{t_0}^{t} e_2^T(\tau)\Big(-\dot{\bar{F}}(\tau)+\ddot{\bar{F}}(\tau) \\ &\quad -\beta_1 sgn\big(e_2(\tau)\big)\Big)d\tau \\ &\quad -e_2^T(t)\dot{\bar{F}}(t)-\beta_1\sum_{i=1}^{2n}\left|e_{2i}(t)\right|+\xi_{b1}.\end{aligned} \tag{D.6}$$

The right-hand side of (D.6) can be upper bounded as follows

$$\begin{aligned}\int_{t_0}^{t} L_1(\tau)\,d\tau &\leqslant \int_{t_0}^{t}\sum_{i=1}^{2n}\left|e_{2i}(\tau)\right|\Big(\left|\dot{\bar{F}}_i(\tau)\right| \\ &\quad +\left|\ddot{\bar{F}}_i(\tau)\right|-\beta_1\Big)d\tau \\ &\quad +\sum_{i=1}^{2n}\left|e_{2i}(t)\right|\Big(\left|\dot{\bar{F}}_i(t)\right|-\beta_1\Big)+\xi_{b1}.\end{aligned} \tag{D.7}$$

If β_1 is chosen to satisfy (D.2), then the first inequality in (D.3) can be proven from (D.7). The second inequality in (D.3) can be obtained by integrating $L_2(t)$, defined in (D.1) as follows

$$\begin{aligned}\int_{t_0}^{t} L_2(\tau)\,d\sigma &= -\beta_2\int_{t_0}^{t}\dot{e}_2^T(\tau)\,sgn\big(e_2(\tau)\big)d\tau \\ &= \xi_{b2}-\beta_2\sum_{i=1}^{2n}\left|e_{2i}(t)\right|\leq\xi_{b2}.\end{aligned} \tag{D.8}$$

The following is the proof of Theorem 3.

Proof. Let the auxiliary functions $P_1(t),\ P_2(t) \in \mathbb{R}$ be defined as follows

$$P_1 \triangleq \xi_{b1} - \int_{t_0}^{t} L_1(\tau)\, d\tau \geq 0 \tag{D.9}$$

$$P_2 \triangleq \xi_{b2} - \int_{t_0}^{t} L_2(\tau)\, d\tau \geq 0 \tag{D.10}$$

where $L_1(t),\ L_2(t),\ \xi_{b1}$ and ξ_{b2} were defined in Lemma 2. The proof of Lemma 2 ensures that $P_1(t)$ and $P_2(t)$ are non-negative. Let $V_1(y,t) \in \mathbb{R}$ denote the following non-negative function

$$V_1 \triangleq \frac{1}{2} e_2^T e_2 + \frac{1}{2} r^T r + P_1 + P_2 \tag{D.11}$$

where $y(t) \in \mathbb{R}^{4n+2}$ is defined as

$$y \triangleq \begin{bmatrix} e_2^T & r^T & \sqrt{P_1} & \sqrt{P_2} \end{bmatrix}^T. \tag{D.12}$$

Note that (D.11) is bounded by the following inequalities

$$W_3(y) \leq V_1(y,t) \leq W_4(y) \tag{D.13}$$

where $W_3(y),\ W_4(y) \in \mathbb{R}$ are defined as

$$W_3(y) = \lambda_4 \left\| y(t) \right\|^2 \qquad W_4(y) = \lambda_5 \left\| y(t) \right\|^2 \tag{D.14}$$

where $\lambda_4,\ \lambda_5 \in \mathbb{R}$ are positive bounding constants.

After differentiating (D.11) in time, results in the following expression

$$\dot{V}_1 = -e_2^T e_2 - k_s r^T r - \beta_2 e_2^T sgn(e_2) \tag{D.15}$$

where (4.44), (4.63), (D.9), and (D.10) were utilized. The expression in (D.15) can be

rewritten as

$$\dot{V}_1 = -\left\|e_2\right\|^2 - k_s \left\|r\right\|^2 - \beta_2 \sum_{i=1}^{2n} \left|e_{2i}\right|. \tag{D.16}$$

From (D.11) and (D.16), it is clear that $V_1(y,t) \in \mathcal{L}_\infty$; hence, $e_2(t) \in \mathcal{L}_\infty \cap \mathcal{L}_2 \cap \mathcal{L}_1$, $r(t) \in \mathcal{L}_\infty \cap \mathcal{L}_2$, and $y(t) \in \mathcal{L}_\infty$. Since $e_2(t)$, $r(t) \in \mathcal{L}_\infty$, then (4.44) and (4.64) can be used to prove that $\dot{e}_2(t)$, $\dot{\hat{F}}(t) \in \mathcal{L}_\infty$. Given that $e_2(t), r(t)$, $\dot{\hat{F}}(t) \in \mathcal{L}_\infty$ and the assumption that $\dot{\overline{F}}(t) \in \mathcal{L}_\infty$, (4.60) can be used to prove that $\dot{r}(t) \in \mathcal{L}_\infty$. Barbalat's Lemma can be utilized to prove

$$\left\|e_2(t)\right\|, \left\|r(t)\right\| \to 0 \quad as \quad t \to \infty. \tag{D.17}$$

From (4.44), (4.45), (D.7) and the fact that $\overline{M}(x) \in \mathcal{L}_\infty$, standard linear analysis arguments can be used to prove that $e_1(t)$, $\dot{e}_1(t)$, $\dot{e}_2(t) \in \mathcal{L}_\infty$ and $e_1(t)$, $\dot{e}_1(t) \in \mathcal{L}_1$, and that

$$\left\|e_1(t)\right\|, \left\|\dot{e}_1(t)\right\|, \left\|\dot{e}_2(t)\right\| \to 0 \quad as \quad t \to \infty. \tag{D.18}$$

By using the assumption that $\overline{F}(t) \in \mathcal{L}_\infty$ and the fact that $\dot{e}_2(t) \in \mathcal{L}_\infty$ from (4.58) it is clear that $\hat{F}(t) \in \mathcal{L}_\infty$. Since $\hat{F}(t) \in \mathcal{L}_\infty$, (4.49) and the proof in Appendix F can be used to prove that $\lambda_d(t)$, $\eta_d(t)$, $\dot{\eta}_d(t)$, $\xi_d(t)$, $\dot{\xi}_d(t)$, $\ddot{\xi}_d(t) \in \mathcal{L}_\infty$. Using these facts along with (4.42), (4.46) and their first time derivatives, it is clear that $x(t)$, $\dot{x}(t)$, $x_m(t)$, $\dot{x}_m(t)$, $x_s(t)$, $\dot{x}_s(t) \in \mathcal{L}_\infty$. Since $e_1(t)$, $\dot{e}_1(t)$, $\overline{M}(x)$, $\dot{\overline{M}}(x) \in \mathcal{L}_\infty$, it is clear from (4.57) that $\overline{T}_1(t) \in \mathcal{L}_\infty$, and using previously stated bounding properties, $\overline{T}(t) \in \mathcal{L}_\infty$. It is also possible to state that $\overline{T}_1(t) \in \mathcal{L}_1$, where (4.57) was utilized. Based on the definition of $x(t)$ in (4.42)

and the previously stated bounding properties, it is clear that $x_s(t) \to x_m(t)$ and $x_m(t) \to \xi_1(t)$. From these bounding statements and standard signal chasing arguments, all signals can be shown to be bounded.

Appendix E

Proof of Theorem 4

Proof. Since the user assist mechanism is disabled (i.e., $\gamma = 0$), the target system defined in (4.48) and (4.49) can be simplified to (4.52). To assist in the subsequent analysis, the following expression can be developed from integration by parts

$$\begin{aligned} \int_{t_0}^{t} \bar{M}\ddot{e}_1(\tau)\,d\tau &= \bar{M}\dot{e}_1(t) - \bar{M}\dot{e}_1(t_0) \\ &\quad - \int_{t_0}^{t} \dot{\bar{M}}\,\dot{e}_1(\tau)\,d\tau. \end{aligned} \tag{E.1}$$

Since $\bar{M}(x)$, $\dot{\bar{M}}(x)$, $\dot{e}_1(t) \in \mathcal{L}_\infty$, and $\dot{e}_1(t) \in \mathcal{L}_1$, then $\int_{t_0}^{t} \bar{M}\ddot{e}_1(\tau)\,d\tau \in \mathcal{L}_\infty$. After integrating (4.55) as follows

$$\int_{t_0}^{t} \tilde{F}(\tau)\,d\tau = -\int_{t_0}^{t} \bar{M}\ddot{e}_1(\tau)\,d\tau - \int_{t_0}^{t} \bar{T}_1(\tau)\,d\tau \tag{E.2}$$

and using the facts that $\bar{T}_1(t) \in \mathcal{L}_1$ (see proof of Theorem 3) and that $\int_{t_0}^{t} \bar{M}\ddot{e}_1(\tau)\,d\tau \in \mathcal{L}_\infty$, it is clear that $\tilde{F}(t) \in \mathcal{L}_1$, where $\tilde{F}(t) \in \mathbb{R}^{2n}$ is defined as follows

$$\tilde{F} \triangleq \bar{F} - \hat{F}. \tag{E.3}$$

The expression in (E.3) can be decomposed as $\tilde{F}(t) = \begin{bmatrix} \tilde{F}_1^T & \tilde{F}_2^T \end{bmatrix}^T$, where $\tilde{F}_1(t)$, $\tilde{F}_2(t) \in \mathbb{R}^n$. After utilizing the fact that $\hat{F}(t_0) = 0_{2n}$, the following can be derived

$$\hat{F}(t) = \int_{t_0}^{t} \dot{\hat{F}}(\tau)\,d\tau. \tag{E.4}$$

From the proof of Theorem 3 (see Appendix D), it is clear that $\hat{F}(t) \in L_\infty$, then from (D.4)

it is also clear that $\dot{\hat{F}}\ (t) \in \mathcal{L}_1$.

By using the transformation in (4.42), the passivity objective in (4.6) can be rewritten as follows

$$\begin{aligned} & \int_{t_0}^{t} \begin{bmatrix} \dot{x}_m^T(\tau) & \dot{x}_s^T(\tau) \end{bmatrix} \begin{bmatrix} F_H(\tau) \\ F_E(\tau) \end{bmatrix} d\tau \\ = & \int_{t_0}^{t} \dot{x}^T \bar{F} d\tau - \int_{t_0}^{t} \begin{bmatrix} 0_n^T & \dot{\xi}_2^T \end{bmatrix} \bar{F} d\tau. \end{aligned} \tag{E.5}$$

By utilizing (E.3) and the time derivative of (4.46), (E.5) can be rewritten as follows

$$\begin{aligned} & \int_{t_0}^{t} \dot{x}^T \bar{F} d\tau - \int_{t_0}^{t} \begin{bmatrix} 0_n^T & \dot{\xi}_2^T \end{bmatrix} \bar{F} d\tau \\ = & \int_{t_0}^{t} \dot{\xi}_1^T(\tau) \tilde{F}_1(\tau) d\tau + \int_{t_0}^{t} \dot{\xi}_1^T(\tau) \hat{F}_1(\tau) d\tau \\ & - \int_{t_0}^{t} \dot{e}_1^T(\tau) \bar{F}(\tau) d\tau. \end{aligned} \tag{E.6}$$

Following expression can be developed from integration by parts of the second integral at the right-hand side of (E.6)

$$\begin{aligned} & \int_{t_0}^{t} \dot{x}^T \bar{F} d\tau - \int_{t_0}^{t} \begin{bmatrix} 0_n^T & \dot{\xi}_2^T \end{bmatrix} \bar{F} d\tau \\ = & \int_{t_0}^{t} \dot{\xi}_1^T(\tau) \tilde{F}_1(\tau) d\tau - \int_{t_0}^{t} \xi_1^T(\tau) \dot{\hat{F}}_1(\tau) d\tau \\ & + \xi_1^T(t) \hat{F}_1(t) - \int_{t_0}^{t} \dot{e}_1^T(\tau) \bar{F}(\tau) d\tau \end{aligned} \tag{E.7}$$

where $\hat{F}(t_0) = 0_{2n}$ is both utilized. Since $\dot{\xi}_1(t) \in \mathcal{L}_\infty$ and $\tilde{F}(t) \in \mathcal{L}_1$, it is clear that the first integral expression in (E.7) is bounded and a lower negative bound exists. Since $\xi_1(t) \in \mathcal{L}_\infty$ and $\dot{\hat{F}}(t) \in \mathcal{L}_1$ it is clear that the second integral expression in (E.7) is bounded and a lower negative bound exists, and since $\xi_1(t), \hat{F}(t) \in \mathcal{L}_\infty$ then third expression is also bounded and a lower negative bound exists. Finally, because $e_1(t) \in \mathcal{L}_1$ and $\bar{F}(t) \in \mathcal{L}_\infty$, it is possible to

show that the last integral in (E.7) is also bounded and a lower negative bound exists. Hence, these facts can be applied to (E.5) to prove that

$$\int_{t_0}^{t} \begin{bmatrix} \dot{x}_m^T(\tau) & \dot{x}_s^T(\tau) \end{bmatrix} \begin{bmatrix} F_H(\tau) \\ F_E(\tau) \end{bmatrix} d\tau \geq -c_4^2 \tag{E.8}$$

where $c_4 \in \mathbb{R}$ is a bounded constant.

Appendix F

UMIF Desired Trajectory Stability Analysis

In the proof of Theorem 3 (see Appendix D), it is proven that $e_1(t), e_2(t), r(t), \hat{F}(t), \dot{\hat{F}}(t) \in \mathcal{L}_\infty$ as well as that $\|e_1(t)\|$, $\|e_2(t)\|$, and $\|r(t)\| \to 0$ as $t \to \infty$ regardless of whether or not $x(t)$, $\xi_d(t)$, $\lambda_d(t)$, $\eta_d(t)$, $\dot{\eta}_d(t) \in \mathcal{L}_\infty$. Therefore the fact that $\hat{F}(t) \in \mathcal{L}_\infty$ can be used in the subsequent analysis. To prove that $\lambda_d(t)$, $\eta_d(t) \in \mathcal{L}_\infty$, let $V(t) \in \mathbb{R}$ denote the following function

$$V \triangleq V_1 + V_2 \tag{F.1}$$

where $V_1(t) \in \mathbb{R}$ denotes the following non-negative function

$$V_1 \triangleq \frac{1}{2}\eta_d^T M_T \eta_d + \frac{1}{2}\lambda_d^T K_T \lambda_d \tag{F.2}$$

where $\lambda_d(t)$, $\eta_d(t)$, M_T and K_T were introduced in (4.49). The expression given in (F.2) can be lower bounded by the auxiliary function, $V_2(\bar{x}) \in \mathbb{R}$, defined as follows

$$V_2 \triangleq 2\varepsilon \eta_d^T M_T \lambda_d \le V_1 \tag{F.3}$$

where $\bar{x}(t) \in \mathbb{R}^{4n}$ is defined as

$$\bar{x} \triangleq [\lambda_d^T \quad \eta_d^T]^T \tag{F.4}$$

and $\varepsilon \in \mathbb{R}$ is a positive bounding constant selected according to the following inequality

$$\varepsilon < \frac{\min\{\lambda_{\min}\{M_T\}, \lambda_{\min}\{K_T\}\}}{4\lambda_{\max}\{M_T\}} \tag{F.5}$$

where $\lambda_{\min}\{\cdot\}$ and $\lambda_{\max}\{\cdot\}$ denote the minimum and maximum eigenvalue of a matrix,

respectively. From (F.3) it is clear that $V(t)$ is a non-negative function and bounded by the following inequalities

$$\underline{\lambda}_1 \|\overline{x}\|^2 \leq V(\overline{x}) \leq \overline{\lambda}_2 \|\overline{x}\|^2 \tag{F.6}$$

where $\underline{\lambda}_1,\ \overline{\lambda}_2 \in \mathbb{R}$ are positive constants defined as follows, provided that ε is selected according to (F.5)

$$\begin{aligned} \underline{\lambda}_1 &\triangleq \frac{1}{2}\min\left\{\lambda_{\min}\{M_T\}, \lambda_{\min}\{K_T\}\right\} \\ &\quad - 2\varepsilon\lambda_{\max}\{M_T\} \\ \overline{\lambda}_2 &\triangleq \frac{1}{2}\max\left\{\lambda_{\max}\{M_T\}, \lambda_{\max}\{K_T\}\right\} \\ &\quad + 2\varepsilon\lambda_{\max}\{M_T\}. \end{aligned} \tag{F.7}$$

To facilitate the subsequent analysis, the time derivative of (F.1) can be determined as follows

$$\begin{aligned} \dot{V} &= \eta_d^T M_T \dot{\eta}_d + \lambda_d^T K_T \dot{\lambda}_d \\ &\quad + 2\varepsilon\dot{\eta}_d^T M_T \lambda_d + 2\varepsilon\eta_d^T M_T \dot{\lambda}_d. \end{aligned} \tag{F.8}$$

After utilizing (4.49) and the fact that $\eta_d(t) = \dot{\lambda}_d(t)$, the expression in (F.8) can be written as

$$\begin{aligned} \dot{V} &= \eta_d^T \left(M_T \overline{M}^{-1}\right)\hat{F} - \eta_d^T B_T \eta_d + 2\varepsilon\lambda_d^T M_T \overline{M}^{-1}\hat{F} \\ &\quad - 2\varepsilon\lambda_d^T B_T \eta_d - 2\varepsilon\lambda_d^T K_T \lambda_d + 2\varepsilon\eta_d^T M_T \eta_d. \end{aligned} \tag{F.9}$$

The right-hand side of (F.9) can be upper bounded as follows

$$\begin{aligned}\dot{V} \leq\ & \xi_{\bar{m}}\lambda_{\max}\{M_T\}\left[\delta_1\|\eta_d\|^2+\frac{1}{\delta_1}\left\|\hat{F}\right\|^2\right] \\ & -\lambda_{\min}\{B_T\}\|\eta_d\|^2 \\ & +2\varepsilon\xi_{\bar{m}}\lambda_{\max}\{M_T\}\left[\delta_3\|\lambda_d\|^2+\frac{1}{\delta_3}\left\|\hat{F}\right\|^2\right] \\ & +2\varepsilon\lambda_{\max}\{B_T\}\left[\delta_2\|\lambda_d\|^2+\frac{1}{\delta_2}\|\eta_d\|^2\right] \\ & -2\varepsilon\lambda_{\min}\{K_T\}\|\lambda_d\|^2+2\varepsilon\lambda_{\max}\{M_T\}\|\eta_d\|^2\end{aligned} \tag{F.10}$$

where the following properties were utilized

$$\begin{aligned}\eta_d^T M_T\bar{M}^{-1}\hat{F} \leq\ & \xi_{\bar{m}}\lambda_{\max}\{M_T\}\left[\delta_1\|\eta_d\|^2\right. \\ & \left.+\frac{1}{\delta_1}\left\|\hat{F}\right\|^2\right]\end{aligned} \tag{F.11}$$

$$-\eta_d^T B_T\eta_d \leq -\lambda_{\min}\{B_T\}\|\eta_d\|^2 \tag{F.12}$$

$$2\varepsilon\lambda_d^T M_T\bar{M}^{-1}\hat{F} \leq 2\varepsilon\xi_{\bar{m}}\lambda_{\max}\{M_T\}\left[\delta_3\|\lambda_d\|^2+\frac{1}{\delta_3}\left\|\hat{F}\right\|^2\right] \tag{F.13}$$

$$-2\varepsilon\lambda_d^T B_T\eta_d \leq 2\varepsilon\lambda_{\max}\{B_T\}\left[\delta_2\|\lambda_d\|^2+\frac{1}{\delta_2}\|\eta_d\|^2\right] \tag{F.14}$$

$$-2\varepsilon\lambda_d^T K_T\lambda_d \leq -2\varepsilon\lambda_{\min}\{K_T\}\|\lambda_d\|^2 \tag{F.15}$$

$$2\varepsilon\eta_d^T M_T\eta_d \leq 2\varepsilon\lambda_{\max}\{M_T\}\|\eta_d\|^2 \tag{F.16}$$

where $\delta_1,\delta_2,\delta_3\in\mathbb{R}$ denote positive bounding constants and $\xi_{\bar{m}}\in\mathbb{R}$ denotes positive bounding constant defined as

$$\left\|\bar{M}^{-1}\right\|_\infty \leq \xi_{\bar{m}} \tag{F.17}$$

where $\left\|\bar{M}^{-1}\right\|_\infty$ denotes the induced infinity norm of the bounded matrix $\bar{M}^{-1}(x)$.

The expression in (F.10) can be rearranged as follows

$$\begin{aligned}\dot{V} \leq & -\Big(\lambda_{\min}\{B_T\} - \xi_{\bar{m}}\delta_1\lambda_{\max}\{M_T\} \\ & -\frac{2\varepsilon\lambda_{\max}\{B_T\}}{\delta_2} - 2\varepsilon\lambda_{\max}\{M_T\}\Big)\|\eta_d\|^2 \\ & -2\varepsilon\Big(\lambda_{\min}\{K_T\} - \delta_2\lambda_{\max}\{B_T\} \\ & -\xi_{\bar{m}}\delta_3\lambda_{\max}\{M_T\})\|\lambda_d\|^2 \\ & +\xi_{\bar{m}}\lambda_{\max}\{M_T\}\left(\frac{1}{\delta_1}+\frac{2\varepsilon}{\delta_3}\right)\|\hat{F}\|^2.\end{aligned} \tag{F.18}$$

Provided δ_1, δ_2, δ_3, M_T, B_T, K_T and ε are selected to satisfy (F.5) and the following sufficient conditions

$$\begin{aligned}\lambda_{\min}\{B_T\} &> \xi_{\bar{m}}\delta_1\lambda_{\max}\{M_T\} \\ &\quad + \frac{2\varepsilon\lambda_{\max}\{B_T\}}{\delta_2} + 2\varepsilon\lambda_{\max}\{M_T\} \\ \lambda_{\min}\{K_T\} &> \xi_{\bar{m}}\delta_3\lambda_{\max}\{M_T\} + \delta_2\lambda_{\max}\{B_T\}\end{aligned}$$

right-hand side of (F.18) can be upper bounded as follows

$$\dot{V} \leq -\frac{\min\{\gamma_a, \gamma_b\}}{\bar{\lambda}_2}V + \varepsilon \tag{F.19}$$

where (F.4) and (F.6) were utilized, and γ_a, γ_b, $\varepsilon \in \mathbb{R}$ denote positive bounding constants.

From (F.1) - (F.3), and (F.6), and that $\hat{F}(t) \in \mathcal{L}_\infty$ (see Appendix D), the expression in (F.19) can be used with the result from [49] to prove that $\overline{x}(t)$, $\lambda_d(t)$, $\eta_d(t) \in \mathcal{L}_\infty$. Based on (4.49), and the fact that $\bar{M}^{-1}(x)$, $\hat{F}(t) \in \mathcal{L}_\infty$ then $\dot{\eta}_d(t) \in \mathcal{L}_\infty$. After utilizing the fact that $\eta_d(t)$, $\dot{\eta}_d(t) \in \mathcal{L}_\infty$ along with the Remark 2, then it is clear that $\xi_d(t)$, $\dot{\xi}_d(t)$, $\ddot{\xi}_d(t) \in \mathcal{L}_\infty$.

Appendix G

Upper Bound Development for MIF Analysis

To simplify the following derivations, (4.31) can be rewritten as follows

$$\begin{aligned} N &\triangleq N\left(x,\dot{x},\ddot{x},e_1,e_2,r,\dddot{x}_d\right) \\ &= \bar{M}\dddot{x}_d + \dot{\bar{M}}\,\ddot{x} + \dot{\bar{N}} + e_2 \\ &\quad + \bar{M}\left(\alpha_1+\alpha_2\right)r - \bar{M}\left(\alpha_1^2+\alpha_1\alpha_2+\alpha_2^2\right)e_2 \\ &\quad + \bar{M}\alpha_2^3 e_1 + \frac{1}{2}\dot{\bar{M}}\,r \end{aligned} \tag{G.1}$$

where (4.15) and (4.16) were both utilized. To facilitate the subsequent analysis, $N\left(x,\dot{x}_d,\ddot{x}_d,0,0,0,\dddot{x}_d\right)$, $N\left(x,\dot{x},\ddot{x}_d,0,0,0,\dddot{x}_d\right)$, $N\left(x,\dot{x},\ddot{x},0,0,0,\dddot{x}_d\right)$, $N\left(x,\dot{x},\ddot{x},e_1,0,0,\dddot{x}_d\right)$, and $N\left(x,\dot{x},\ddot{x},e_1,e_2,0,\dddot{x}_d\right)$ are added and subtracted to the right-hand side of (4.30) as follows

$$\begin{aligned} &\tilde{N} \\ &= \left[N\left(x,\dot{x}_d,\ddot{x}_d,0,0,0,\dddot{x}_d\right) - N_d\left(x_d,\dot{x}_d,\ddot{x}_d,0,0,0,\dddot{x}_d\right)\right] \\ &\quad + \left[N\left(x,\dot{x},\ddot{x}_d,0,0,0,\dddot{x}_d\right) - N\left(x,\dot{x}_d,\ddot{x}_d,0,0,0,\dddot{x}_d\right)\right] \\ &\quad + \left[N\left(x,\dot{x},\ddot{x},0,0,0,\dddot{x}_d\right) - N\left(x,\dot{x},\ddot{x}_d,0,0,0,\dddot{x}_d\right)\right] \\ &\quad + \left[N\left(x,\dot{x},\ddot{x},e_1,0,0,\dddot{x}_d\right) - N\left(x,\dot{x},\ddot{x},0,0,0,\dddot{x}_d\right)\right] \\ &\quad + \left[N\left(x,\dot{x},\ddot{x},e_1,e_2,0,\dddot{x}_d\right) - N\left(x,\dot{x},\ddot{x},e_1,0,0,\dddot{x}_d\right)\right] \\ &\quad + \left[N\left(x,\dot{x},\ddot{x},e_1,e_2,r,\dddot{x}_d\right) - N\left(x,\dot{x},\ddot{x},e_1,e_2,0,\dddot{x}_d\right)\right]. \end{aligned} \tag{G.2}$$

After applying the Mean Value Theorem to each bracketed term of (G.2), the following expression can be obtained

$$
\begin{aligned}
\tilde{N} \quad = \quad & \\
& \left.\frac{\partial N\left(\sigma_1, \dot{x}_d, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_1}\right|_{\sigma_1 = v_1} \left(x - x_d\right) \\
& + \left.\frac{\partial N\left(x, \sigma_2, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_2}\right|_{\sigma_2 = v_2} \left(\dot{x} - \dot{x}_d\right) \\
& + \left.\frac{\partial N\left(x, \dot{x}, \sigma_3, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_3}\right|_{\sigma_3 = v_3} \left(\ddot{x} - \ddot{x}_d\right) \\
& + \left.\frac{\partial N\left(x, \dot{x}, \ddot{x}, \sigma_4, 0, 0, \dddot{x}_d\right)}{\partial \sigma_4}\right|_{\sigma_4 = v_4} \left(e_1 - 0\right) \\
& + \left.\frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, \sigma_5, 0, \dddot{x}_d\right)}{\partial \sigma_5}\right|_{\sigma_5 = v_5} \left(e_2 - 0\right) \\
& + \left.\frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, e_2, \sigma_6, \dddot{x}_d\right)}{\partial \sigma_6}\right|_{\sigma_6 = v_6} \left(r - 0\right)
\end{aligned}
\tag{G.3}
$$

where $v_1 \in \left(x_d, x\right)$, $v_2 \in \left(\dot{x}_d, \dot{x}\right)$, $v_3 \in \left(\ddot{x}_d, \ddot{x}\right)$, $v_4 \in \left(0, e_1\right)$, $v_5 \in \left(0, e_2\right)$, and $v_6 \in \left(0, r\right)$. The right-hand side of (G.3) can be upper bounded as follows

$$
\begin{aligned}
\left\|\tilde{N}\right\| \;\leq\; & \\
& \left\| \left. \frac{\partial N\left(\sigma_1, \dot{x}_d, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_1} \right|_{\sigma_1 = v_1} \right\| \left\| e_1 \right\| \\
& + \left\| \left. \frac{\partial N\left(x, \sigma_2, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_2} \right|_{\sigma_2 = v_2} \right\| \left\| \dot{e}_1 \right\| \\
& + \left\| \left. \frac{\partial N\left(x, \dot{x}, \sigma_3, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_3} \right|_{\sigma_3 = v_3} \right\| \left\| \ddot{e}_1 \right\| \\
& + \left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, \sigma_4, 0, 0, \dddot{x}_d\right)}{\partial \sigma_4} \right|_{\sigma_4 = v_4} \right\| \left\| e_1 \right\| \\
& + \left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, \sigma_5, 0, \dddot{x}_d\right)}{\partial \sigma_5} \right|_{\sigma_5 = v_5} \right\| \left\| e_2 \right\| \\
& + \left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, e_2, \sigma_6, \dddot{x}_d\right)}{\partial \sigma_6} \right|_{\sigma_6 = v_6} \right\| \left\| r \right\|.
\end{aligned}
\tag{G.4}
$$

The partial derivatives in (G.3) can be calculated by using (G.1) as follows

$$
\begin{aligned}
\frac{\partial N\left(\sigma_1, \dot{x}_d, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_1} \;=\; & \frac{\partial \bar{M}\left(\sigma_1\right)}{\partial \sigma_1} \dddot{x}_d \\
& + \frac{\partial \dot{\bar{M}}\left(\sigma_1, \dot{x}_d\right)}{\partial \sigma_1} \ddot{x}_d \\
& + \frac{\partial \dot{\bar{N}}\left(\sigma_1, \dot{x}_d, \ddot{x}_d\right)}{\partial \sigma_1}
\end{aligned}
\tag{G.5}
$$

$$
\begin{aligned}
\frac{\partial N\left(x, \sigma_2, \ddot{x}_d, 0, 0, 0, \dddot{x}_d\right)}{\partial \sigma_2} \;=\; & \frac{\partial \dot{\bar{M}}\left(x, \sigma_2\right)}{\partial \sigma_2} \ddot{x}_d \\
& + \frac{\partial \dot{\bar{N}}\left(x, \sigma_2, \ddot{x}_d\right)}{\partial \sigma_2}
\end{aligned}
\tag{G.6}
$$

$$\frac{\partial N\left(x,\dot{x},\sigma_3,0,0,0,\dddot{x}_d\right)}{\partial\sigma_3} = \dot{\bar{M}}\left(x,\dot{x}\right) + \frac{\partial \dot{\bar{N}}\left(x,\dot{x},\sigma_3\right)}{\partial\sigma_3} \tag{G.7}$$

$$\frac{\partial N\left(x,\dot{x},\ddot{x},\sigma_4,0,0,\dddot{x}_d\right)}{\partial\sigma_4} = \alpha_2^3\bar{M}\left(x\right) \tag{G.8}$$

$$\frac{\partial N\left(x,\dot{x},\ddot{x},e_1,\sigma_5,0,\dddot{x}_d\right)}{\partial\sigma_5} = I_{2n} - \left(\alpha_1^2+\alpha_1\alpha_2+\alpha_2^2\right)\bar{M}\left(x\right) \tag{G.9}$$

$$\frac{\partial N\left(x,\dot{x},\ddot{x},e_1,e_2,\sigma_6,\dddot{x}_d\right)}{\partial\sigma_6} = \left(\alpha_1+\alpha_2\right)\bar{M}\left(x\right) + \frac{1}{2}\dot{\bar{M}}\left(x,\dot{x}\right) \tag{G.10}$$

where $I_{2n}\in\mathbb{R}^{2n\times 2n}$ denotes the identity matrix. By defining

$$\begin{aligned} v_1 &\triangleq x-\tau_1\left(x-x_d\right) & v_2 &\triangleq \dot{x}-\tau_2\left(\dot{x}-\dot{x}_d\right)\\ v_3 &\triangleq \ddot{x}-\tau_3\left(\ddot{x}-\ddot{x}_d\right) & v_4 &\triangleq e_1-\tau_4\left(e_1-0\right)\\ v_5 &\triangleq e_2-\tau_5\left(e_2-0\right) & v_6 &\triangleq r-\tau_6\left(r-0\right) \end{aligned}$$

where $\tau_i\in\left(0,1\right)$ $\forall$ i=1,2,...,6 and if the assumptions stated for the system model and the desired trajectory are met, then upper bounds for the right-hand sides of (G.5)-(G.10) can be rewritten as follows

$$\left\|\left.\frac{\partial N\left(x,\sigma_2,\ddot{x}_d,0,0,0,\dddot{x}_d\right)}{\partial\sigma_2}\right|_{\sigma_2=v_2}\right\| \leqslant \rho_2\left(x,\dot{x},\ddot{x}\right) \tag{G.11}$$

$$\left\|\left.\frac{\partial N\left(x,\dot{x},\sigma_3,0,0,0,\dddot{x}_d\right)}{\partial\sigma_3}\right|_{\sigma_3=v_3}\right\| \leqslant \rho_3\left(x,\dot{x}\right) \tag{G.12}$$

$$\left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, \sigma_4, 0, 0, \dddot{x}_d\right)}{\partial \sigma_4} \right|_{\sigma_4 = v_4} \right\| \leqslant \rho_4\left(x\right) \tag{G.13}$$

$$\left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, \sigma_5, 0, \dddot{x}_d\right)}{\partial \sigma_5} \right|_{\sigma_5 = v_5} \right\| \leqslant \rho_5\left(x\right) \tag{G.14}$$

$$\left\| \left. \frac{\partial N\left(x, \dot{x}, \ddot{x}, e_1, e_2, \sigma_6, \dddot{x}_d\right)}{\partial \sigma_6} \right|_{\sigma_6 = v6} \right\| \leqslant \rho_6\left(x, \dot{x}\right) \tag{G.15}$$

where $\rho_i\left(\cdot\right)$ $\forall$ i=1,2,...,6, are positive nondecreasing functions of $x(t), \dot{x}(t)$ and $\ddot{x}(t)$. After substiting (G.11)-(G.16) into (G.4), $\tilde{N}\left(\cdot\right)$ can be rewritten as

$$\begin{aligned} \tilde{N} \leq & \left[\rho_1\left(\|e_1\|, \|e_2\|, \|r\|\right) + \rho_4\left(\|e_1\|\right)\right]\|e_1\| \\ & + \rho_2\left(\|e_1\|, \|e_2\|, \|r\|\right)\|\dot{e}_1\| \\ & + \rho_3\left(\|e_1\|, \|e_2\|\right)\|\ddot{e}_1\| \\ & + \rho_5\left(\|e_1\|\right)\|e_2\| \\ & + \rho_6\left(\|e_1\|, \|e_2\|\right)\|r\| \end{aligned} \tag{G.17}$$

where (4.15) and (4.16) were utilized. The expressions in (4.15), (4.16) and (4.97) can be used to rewrite the upper bound for the right-hand side of (G.17) as in (B.17).

BIBLIOGRAPHY

[1] F. Asano, Z. Luo, M. Yamakita and S. Hosoe, "Dynamic Modeling and Control for Whole Body Manipulation," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Las Vegas, NV, 2003, pp. 3162-3167.

[2] F. Asano, Z. Luo, M. Yamakita, K. Tahara and S. Hosoe, "Bio-Mimetic Control for Whole Arm Cooperative Manipulation," *Proc. IEEE Int. Conf. on Systems Man and Cybernetics*, The Hague, The Netherlands, 2004, pp. 704-709.

[3] D. Braganza, M. McIntyre, D. Dawson and I. Walker, "Whole Arm Grasping Control for Redundant Robot Manipulators", Clemson University CRB Technical Report, CU/CRB/ 10/12/05 #1, http://www.ces.clemson.edu/ece/crb/publictn/tr.htm, October, 2005.

[4] I. Cervantes, R. Kelly, J. Alvarez-Ramirez, and J. Moreno, "A Robust Velocity Field Control," *IEEE Trans. on Control Systems Technology*, vol. 10, no. 6, pp. 888-894, 2002.

[5] G. Chirikjian and J. W. Burdick, "Design and Experiments with a 30 DOF Robot," *Proc. IEEE Int. Conf. on Robotics and Automation*, Atlanta, GA, 1993, pp. 113-119.

[6] F. L. Lewis, D. M. Dawson and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*, Marcel Dekker Publishing Co., New York, NY, 2003.

[7] J. Li and P. Y. Li, "Passive Velocity Field Control (PVFC) Approach to Robot Force Control and Contour Following," *Proc. of the Japan/USA Symposium on Flexible Automation*, Ann Arbor, Michigan, July 2000.

[8] P. Y. Li and R. Horowitz, "Passive Velocity Field Control of Mechanical Manipulators," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 4, pp. 751-763, 1999.

[9] M. Loffler, N. Costescu and D. M. Dawson, "QMotor 3.0 and the QMotor Robotic Toolkit – An Advanced PC-Based Real-Time Control Platform," *IEEE Control Systems Mag.*, vol 22, no. 3, pp. 12-26, (2002).

[10] M. L. McIntyre, W. Dixon, D. Dawson and B. Xian, "Adaptive Tracking Control of On-Line Path Planners: Velocity Fields and Navigation Functions," *Proc. of the American Controls Conference*, Portland, Oregon, June 2005, pp. 3168-3173.

[11] C. Melchiorri and G. Vassura, "Implementation of whole-hand manipulation capability in the UB hand system design," *Robotics Society of Japan - Advanced Robotics*, vol. 9, no. 5, pp. 547-560, 1995.

[12] K. Mirza and D. E. Orin, "Force Distribution for Power Grasp in the Digits system," *Proc. 8th CISM-IFToMM Symp. on Theory and Practice of Robots and Manipulators*, Cracow, Poland, 1990.

[13] H. Mochiyama, "Whole-Arm Impedance of a Serial-Chain Manipulator," *Proc. IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, 2001, pp. 2223-2228.

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

[15] R. Platt, A. H. Fagg and R. A. Grupen, "Nullspace Composition of Control Laws for Grasping," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002, pp. 1717-1723.

[16] R. Platt, A. H. Fagg and R. A. Grupen, "Extending Fingertip Grasping to Whole Body Grasping," *Proc. IEEE Int. Conf. on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 2677-2682.

[17] D. Reynaerts, H. Van Brussel, "Whole-finger Manipulation with a Two-fingered Robot Hand," *Robotics Society of Japan - Advanced Robotics*, vol. 9, no. 5, pp. 505-518, 1995.

[18] K. Salisbury, "Whole Arm Manipulation," Proc. 4th Int. Symposium Robotics Research, 1987, pp. 183-189.

[19] P. Song, M. Yashima and V. Kumar, "Dynamics and Control of Whole Arm Grasps," *Proc. IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, 2001, pp. 2229-2234.

[20] E. Tatlicioglu, M. L. McIntyre, D. Dawson and I. Walker, "Adaptive Nonlinear Tracking Control of Kinematically Redundant Robot Manipulators with Sub-Task Extensions" *Proc. IEEE Conf. on Decision and Control*, Seville, Spain, December 2005, pp. 4373 - 4378.

[21] J. C. Trinkle, J. M. Abel and R. P. Paul, "An Investigation of Frictionless Enveloping Grasping in the plane," *Int. J. Robot. Res.*, vol. 7, no. 3, pp. 33-51, 1988.

[22] B. Xian, D. M. Dawson, M. S. de. Queiroz and J. Chen, "A Continuous Asymptotic Tracking Control Strategy for Uncertain Nonlinear Systems," *IEEE Trans. on Automatic Control*, vol. 49, no. 7, pp. 1206 - 1211, 2004.

[23] A. Bicchi, "Hands for Dexterous Manipulation and Robust Grasping: A Difficult Road Toward Simplicity," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 6, pp. 652-662, 2000.

[24] M. J. Sheridan, S. C. Ahalt and D. E. Orin, "Fuzzy Control for Robotic Power Grasp," *Robotics Society of Japan - Advanced Robotics*, vol. 9, no. 5, pp. 535-546, 1995.

[25] G. Robinson and J. B. C. Davies, "Continuum robots – a state of the art," in *Proc. IEEE Int. Conf. Robot. Automat.,* Detroit, Michigan, USA, 1999, pp. 2849-2854.

[26] I. D. Walker, D. M. Dawson, T. Flash, F. Grasso, R. Hanlon, B. Hochner, W. Kier, C. D. Rahn, and Q. Zhang, "Continuum robot arms inspired by cephalopods," in *Proc. 2005 SPIE Conf. Unmanned Ground Vehicle Technology IV,* Orlando, Florida, USA, Mar. 2005, pp. 303-314.

[27] F. Matsuno and K. Suenga, "Control of redundant snake robot based on kinematic model," in *Proc. 41st SICE Annual Conference,* Osaka, Japan, 2002, pp. 1481-1486.

[28] M. Ivanescu, N. Popescu, and D. Popescu, "A variable length tentacle manipulator control system," in *Proc. IEEE Int Conf. Robot. Automat.,* Barcelona, Spain, 2005, pp. 3040-3045.

[29] T. Suzuki, K. Shintani, and H. Mochiyama, "Control methods of hyperflexible manipulators using their dynamical features," in *Proc. 41st SICE Annual Conference,* Osaka, Japan, 2002, pp. 1511-1516.

[30] F. Matsuno and H. Sato, "Trajectory tracking control of snake robots based on dynamic model," in *Proc. IEEE Int. Conf. Robot. Automat.,* Barcelona, Spain, 2005, pp. 3040-3045.

[31] M. Ivanescu, "Position dynamic control for a tentacle manipulator," in *Proc. IEEE Int. Conf. Robot. Automat.,* Washington, DC, 2002, pp. 1531-1538.

[32] H. Mochiyama, E. Shimemura, and H. Kobayashi, "Control of manipulators with hyper degrees of freedom: shape tracking using only joint angle information," *International Journal of Systems Science,* vol. 30, no. 1, pp. 77-85, 1999.

[33] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. London: Taylor and Francis, June 1999.

[34] C. Kwan, F. L. Lewis, and D. M. Dawson, "Robust neural-network control of rigid-link electrically driven robots," *IEEE Trans. Neural Networks,* vol. 9, no. 4, pp. 581-588, July 1998.

[35] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with gauranteed tracking performances," *IEEE Trans. Neural Networks,* vol. 7, no. 2, pp. 388-399, Mar. 1996.

[36] H. D. Patino, R. Carelli, and B. R. Kuchen, "Neural networks for asvanced control of robot manipulators," *IEEE Trans. Neural Networks,* vol. 13, no. 2, pp. 343-354, Mar. 2002.

[37] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," vol. 22, no. 1, pp. 43-55, Feb. 2006.

[38] I. Gravagne, C. Rahn, and I. Walker, "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME Trans. Mechatron.,* vol. 8, no. 2, pp. 299-307, June 2003.

[39] H. Mochiyama and T. Suzuki, "Kinematics and dynamics of a cable-like hyper-flexible manipulator," in *Proc. IEEE Int. Conf. Robot. Automat.,* Taipei, Taiwan, 2003, pp. 3672-3677.

[40] ——, "Dynamic modeling of a hyper-flexible manipulator," in *Proc. 41st SICE Annual Conference,* Osaka, Japan, 2002, pp. 1505-1510.

[41] W. McMahan, B. Jones, I. Walker, V. Chitrakaran, A. Seshadri, and D. Dawson, "Robotic manipulators inspired by cephalopod limbs," in *CDEN Symposium on Biomimicry, Bionics and Biomechanics,* Montreal, Canada, 2004, pp. 1-10.

[42] W. McMahan, V. Chitrakaran, M. Csencsits, D. M. Dawson, I. D. Walker, B. Jones, M. Pritts, D. Dienno, M. Grissom, and C. Rahn, "Field trials and testing of the octarm continuum manipulator," in *Proc. IEEE Int. Conf. Robot. Automat.,* Orlando, FL, 2006.

[43] M. B. Pritts and C. D. Rahn, "Design of an artificial muscle continuum robot," in *Proc. IEEE Int. Conf. Robot. Automat.,* New Orleans, LA, USA, 2004, pp. 4742-4746.

[44] R. J. Anderson and M. W. Spong, "Bilateral Control of Teleoperators with Time Delay," *IEEE Trans. on Automatic Control,* vol. 34, no. 5, pp. 494-501 (1989).

[45] N. Chopra, M. W. Spong, S. Hirche, and M. Buss, "Bilateral Teleoperation over the Internet: the Time Varying Delay Problem," *Proc. IEEE American Control Conference,* June 4-6, Denver, CO, 2003, pp. 155-160.

[46] N. Chopra, M. W. Spong, R. Ortega, and N. E. Barabanov, "On Position Tracking in Bilateral Teleoperation," *Proc. IEEE American Control Conference,* June 30-July 2, Boston, MA, 2004, pp. 5244-5249.

[47] N. Chopra, M. W. Spong, and R. Lozano, "Adaptive Coordination Control of Bilateral Teleoperators with Time Delay," *Proc. of IEEE Conference on Decision and Control,* December 14-17, Nassau, Bahamas, 2004, pp. 4540-4547.

[48] J. E. Colgate, "Robust Impedance Shaping Telemanipulation," *IEEE Trans. on Robotics and Automation,* vol. 9, no. 4, pp. 374-384 (1993).

[49] M. Corless and G. Leitmann, "Continuous State Feedback Guaranteeing Uniform Ultimate Boundedness for Uncertain Dynamic Systems," *IEEE Trans. on Automatic Control,* vol. 26, no. 5, pp. 1139-1144 (1981).

[50] E. L. Faulring, K. M. Lynch, J. E. Colgate and M. A. Peshkin, "Haptic Interaction With Constrained Dynamic Systems," *Proc. IEEE Int. Conf. on Robotics and Automation,* April 18-22, Barcelona, Spain, 2005, pp. 2458-2464.

[51] K. B. Fite, L. Shao, and M. Goldfarb, "Loop shaping for transparency and stability robustness in bilateral telemanipulation," *IEEE Trans. on Robotics and Automation,* vol. 20, no. 3, pp. 620-624 (2004).

[52] B. Hannaford, "A Design Framework for Teleoperators with Kinesthetic Feedback," *IEEE Trans. on Robotics and Automation,* vol. 5, no. 4, pp. 426-434 (1989).

[53] B. Hannaford and J. -H. Ryu, "Time domain passivity control of haptic interfaces," *IEEE Trans. on Robotics and Automation,* vol. 18, no. 1, pp. 1-10 (2002).

[54] K. Hashtrudi-Zaad and S. E. Salcudean, "Adaptive transparent impedance reflecting teleoperation," *Proc. IEEE Int. Conf. on Robotics and Automation,* April 22-28, Minneapolis, MN, 1996, pp. 1369-1374.

[55] H. Khalil, Nonlinear Systems, Third Edition, Upper Saddle River, NJ: Prentice-Hall, Inc., 2002.

[56] D. A. Lawrence, "Stability and Transparency in Bilateral Teleoperation," *IEEE Trans. on Robotics and Automation,* vol. 9, no. 5, pp. 624-637 (1993).

[57] D. Lee and P. Y. Li, "Passive Coordination Control of Nonlinear Bilateral Teleoperated Manipulators," *Proc. IEEE Int. Conf. Robotics and Automation,* May 11-15, Washington, DC, 2002, pp. 3278-3283.

[58] D. Lee and P. Y. Li, "Passive tool dynamics rendering for nonlinear bilateral teleoperated manipulators," *Proc. IEEE Int. Conf. Robotics and Automation,* May 11-15, Washington, DC, 2002, pp. 3284-3289.

[59] D. Lee and P. Y. Li, "Passive Bilateral Feedforward Control of Linear Dynamically Similar Teleoperated Manipulators," *IEEE Trans. on Robotics and Automation,* vol. 19, no. 3, pp. 443-456 (2003).

[60] D. Lee and P. Y. Li, "Toward robust passivity: a passive control implementation structure for mechanical teleoperators," *Proc. IEEE VR Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems,* March 22-23, Los Angeles, CA, 2003, pp. 132-139.

[61] D. Lee and P. Y. Li, "Passive bilateral control and tool dynamics rendering for nonlinear mechanical teleoperators," *IEEE Trans. on Robotics,* vol. 21, no. 5, pp. 936-951 (2005).

[62] D. Lee and M. W. Spong, "Bilateral Teleoperation of Multiple Cooperative Robots over Delayed Communication Networks: Theory," *Proc. of IEEE Int. Conf. Robotics and Automation,* April 18-22, Barcelona, Spain, 2005, pp. 360-365.

[63] D. Lee, O. Martinez-Palafox, and M. W. Spong, "Bilateral Teleoperation of Multiple Cooperative Robots over Delayed Communication Networks: Application," *Proc. of IEEE Int. Conf. Robotics and Automation,* April 18-22, Barcelona, Spain, 2005, pp. 366-371.

[64] H. K. Lee and M. J. Chung, "Adaptive controller of a master-slave system for transparent teleoperation," *Journal of Robotic Systems,* vol. 15, no. 8, pp. 465-475 (1998).

[65] M. McIntyre, W. Dixon, D. Dawson, and E. Tatlicioglu, "Passive Coordination of Nonlinear Bilateral Teleoperated Manipulators," *Robotica,* vol. 24, no. 4, pp. 463-476, 2006.

[66] G. Niemeyer, and J. -J. E. Slotine, "Stable Adaptive Teleoperation," *IEEE Journal of Oceanic Engineering,* vol. 16, no. 1, pp. 152-162 (1991).

[67] G. Niemeyer, and J. -J. E. Slotine, "Designing Force Reflecting Teleoperators with Large Time Delays to Appear as Virtual Tools," *Proc. IEEE Int. Conf. Robotics and Automation,* April 20-25, Albuquerque, NM, 1997, pp. 2212-2218.

[68] Z. Qu and J. -X. Xu, "Model-Based Learning Controls and Their Comparisons Using Lyapunov Direct Method," *Asian Journal of Control,* vol. 4, no. 1, pp. 99-110 (2002).

[69] J. H. Ryu and D. -S. Kwon, "A Novel Adaptive Bilateral Control Scheme Using Similar Closed-loop Dynamic Characteristics of Master/Slave Manipulators," *Journal of Robotic Systems,* vol. 18, no. 9, pp. 533-543, (2001).

[70] J. -H. Ryu, C. Preusche, B. Hannaford, and G. Hirzinger, "Time domain passivity control with reference energy following," *IEEE Trans. on Control Systems Technology,* vol. 13, no. 5, pp. 737-742 (2005).

[71] S. E. Salcudean, M. Zhu, W. -H. Zhu, and K. Hashtrudi-Zaad, "Transparent Bilateral Teleoperation Under Position and Rate Control," *Int. Journal of Robotics Research,* vol. 19, no. 12, pp. 1185-1202 (2000).

[72] T. B. Sheridan, Telerobotics, Automation, and Human Supervisory Control, Cambridge, MA: MIT Press, 1992.

[73] M. Shi, G. Tao, H. Liu, and J. Hunter Downs, "Adaptive control of teleoperation systems," *Proc. of IEEE Int. Conf. Decision and Control,* December 7-10, Phoenix, AZ, 1999, pp. 791-796.

[74] M. W. Spong, and M. Vidyasagar, Robot Dynamics and Control, New York, NY: John Wiley & Sons, Inc., 1989.

[75] R. Q. Van der Linde, P. Lammertse, E. Frederiksen, B. Ruiter, "The HapticMaster, a new high-performance haptic interface," *Proc. EuroHaptics,* Edinburgh, U.K., 2002.

[76] B. Xian, M. S. de Queiroz, and D. M. Dawson, "A Continuous Control Mechanism for Uncertain Nonlinear Systems," Optimal Control, Stabilization, and Nonsmooth Analysis, Lecture Notes in Control and Information Sciences, Heidelberg, Germany: Springer-Verlag, pp. 251-262, 2004.

[77] Y. Yokokohji and T. Yoshikawa, "Bilateral Control of Master-Slave Manipulators for Ideal Kinesthetic Coupling-Formulation and Experiment," *IEEE Trans. on Robotics and Automation,* vol. 10, no. 5, pp. 605-620 (1994).

[78] W. H. Zhu and S. E. Salcudean, "Stability Guaranteed Teleoperation: An Adaptive Motion/Force Control Approach," *IEEE Trans. on Automatic Control,* vol. 45, no. 11, pp. 1951-1969 (2000).