NORTHWESTERN UNIVERSITY

The Cobotic Hand Controller: Design, Control and Analysis of a Novel Haptic Display

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Mechanical Engineering

By

Eric L. Faulring

EVANSTON, ILLINOIS

December 2005

ABSTRACT

The Cobotic Hand Controller: Design, Control and Analysis of a Novel Haptic Display

Eric L. Faulring

Haptic displays utilizing cobotic technology are able to simulate virtual environments with high dynamic range. The use of nonholonomic, continuously variable transmissions allows for extremely high stiffness in directions that would violate a virtual constraint. At the same time, smooth, unimpeded motion is permitted tangential to virtual constraints and in open space.

This thesis introduces a novel six-degree-of-freedom cobot, utilizing six continuously variable rotational-to-linear transmissions connected in parallel to a common power cylinder. A controller is developed to best utilize the control freedom of this redundantly actuated cobot. Kinematics and dynamics of the Hand Controller are derived, along with extensive characterization of its overall control architecture and resulting performance. An admittance device framework for physically accurate haptic display of bilateral constraints and inertia is developed and examples provided.

This thesis also provides an extensive analysis of the rotational-to-linear dry-friction rolling contact cobotic transmissions utilized in the Cobotic Hand Controller. The elastic, inelastic and shear losses at the contact patch are characterized and a bond graph is provided. In addition, losses due to overhead of the cobotic system are analyzed and a power efficiency comparison is made between cobotic and conventional system architectures. An understanding of the implications of cobotic transmission design on the performance of a high-degree-of-freedom haptic display is provided.

Acknowledgements

I sincerely thank my advisors Ed Colgate and Michael Peshkin for sharing with me their wealth of engineering and analysis skills as teachers, committee members and colleagues, and for always challenging me to improve my own understanding of a problem.

I thank Kevin Lynch for his guidance as a teacher, committee member, and geometry guru, and for his attention to Chapter 8.

I thank Kornel Ehmann for his guidance as a teacher and as committee member.

I acknowledge the many contributions of collaborators Young Soo Park, Hyosig Kang and Tom Ewing at Argonne National Labs.

I thank Professor Sam Colgate for some ingenious design suggestions.

I thank fellow lab mates Brian DeJong, Tom Worsnopp, Songho Kim, Mark Salada, Traveler Hauptman, Jim Lipsey, David Weir, Jay Bernheisel and Kyle Reed for numerous discussions about design, programming and algorithm issues.

I thank master machinists Richard Dojutrek and Robert Taglia for teaching me everything I know about machining.

I acknowledge the support of the Department of Energy, grant number DE-FG07-01ER63288, for funding the development of the Cobotic Hand Controller.

I acknowledge the support the American Society of Engineering Education's National Defense Science and Engineering Graduate Fellowship, the Walter P. Murphy Fellowship, and the Royal E. Cabell Fellowship.

Preface

I dedicate this work to:

My parents for teaching me the virtues of pride and perfection in whatever I do, and for enabling me to pursue whatever I wanted in life.

My wife, for encouraging me to obtain a graduate degree, for supporting me through all of our schooling and for making the world fun to explore.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

The Cobotic Hand Controller was developed in order to provide Argonne National Laboratory and the Department of Energy with a superior teleoperation master controller that excelled at rendering virtual fixtures and would be useful for decontamination and decommissioning work. In this chapter, we discuss the above application and other uses for a high fidelity haptic display. We review previously existing haptic displays and introduce cobotics as a novel solution. In addition, we review the development of cobots, the modelling of cobotic transmissions, cobot control algorithms and the genesis of haptic display algorithms in general.

1.1. Introduction to haptics

There are an increasing number of virtual environment and teleoperation based tasks in which a high fidelity haptic master[1] controller is desired. These include interaction with computer aided design models, flight simulators, the control of a slave robot in medical surgeries, micro/nano-manipulation, undersea salvage, regular maintenance of nuclear plants and other hazardous environments, and decontamination and decommissioning of many chemical and nuclear facilities. The execution of these

[1] We will use the terms *master* and *slave* to define the manipulandum held by the operator and the remote manipulator, respectively.

tasks by a remote operator is affected by his/her level of telepresence and the transparency of the master-slave relationship [154]. This illusion of presence is enhanced by audio, visual and haptic cues. While visual cues are certainly mandatory, and audio cues beneficial at times, haptic cues can significantly improve the flow of information from the slave site to the remote operator for many tasks requiring dexterity. Haptic cues are either impedances or admittances - relationships between motion and force or vice-versa, that an operator encounters when interacting with a display.

1.1.1. Motivation for the Cobotic Hand Controller

The specific application for the development of the master hand controller introduced here is the teleoperation of the Dual Arm Work Platform (DAWP) at Argonne National Laboratory [102, 134]. The DAWP contains two six-degree-of-freedom Schilling Manipulators, several tools and seven cameras. The DAWP can be lowered into a nuclear reactor environment to perform disassembly. The tool inventory consists of a portable band saw, circular saw, pry bar, and wrench. The manipulators are hydraulically actuated, serial link robots, which utilize a gripper as a seventh degree of freedom that can attach rigidly to specially outfitted tools.

The existing master controllers for the DAWP consist of two *mini-master* manipulators, which share the same kinematics as the slave manipulators but are completely passive devices, equipped only with encoders for use as input devices. For the DAWP operator, 90 percent of his/her time is spent in alignment procedures, often after slipping off the head of a bolt or out of a cut in progress [49]. The development of the

Cobotic Hand Controller at Northwestern seeks to improve an operator's execution of manual tasks while a collaborative sister project at Argonne National Laboratory seeks to automate certain tasks through fusion of sensor information [137].

1.1.2. Haptic cues and virtual fixtures

One of the key improvements the Cobotic Hand Controller can provide to DAWP operation is the implementation of virtual surfaces, or virtual constraints on motion, as suggested by Rosenberg [144], Arai [6], Joly and Andriot [88] and Abbott et al. [1, 2]. While this could be done at the slave side in the existing system, an active master allows for the reproduction of these constraints at the master and greatly reduces operator fatigue while increasing efficiency by eliminating unneeded motions in six-space. Thus, if the operator is using a saw and constrains the motion of the saw to the plane of the blade at the slave, he/she will feel these same constraints at the master.

Physical cues provided by the master manipulandum or haptic display, such as feedback from the remote site or assistive constraints on motion, improve operator performance and efficiency for many tasks requiring dexterity. Consider a surgeon remotely controlling a slave robot's movement of a scalpel. In order to execute a precise incision, it may be desirable for the motion of the scalpel to be constrained to a straight path at a certain depth, filtering out tremor and preventing damage to tissues beneath the incision or along either side. Such constraints can vastly simplify execution of a six-degree-of-freedom task in a teleoperation setting. Force feedback can also be used to inform the surgeon as to what impedance the scalpel is encountering. The interactions or constraint forces imposed on the scalpel in the remote environment are relayed back to the operator via the haptic display.

1.1.3. On transparency and perception

Haptic displays provide both sensory feedback that coincides with visual and auditory cues, and also entirely haptic cues known as virtual fixtures. Rosenberg defined virtual fixtures as "abstract perceptual information overlaid on top of the reflected sensory feedback from a telepresence work site or virtual environment..." [144]. In order for the virtual fixtures to be interpreted properly, Rosenberg introduced the notion of *design for perception* rather than *design for reality.* Rosenberg suggested that designers take a step back from the traditional goal of reproducing a physical reality, and instead attempt to produce the relevant perceptual reality, which may in fact be easier and more effectively implemented. Conversely, Colgate and Brown noted that the concept of transparency places focus on mimicking the governing state equations of physical systems and not on obeying underlying physical laws such as conservation of energy, which often lead to a lack of transparency when not obeyed [37]. The two views are not necessarily in conflict. Rosenberg focused on the ability to interpret cues whereas Colgate and Brown focused on perceiving immersion in a virtual reality scenario.

Although the designer of a haptic display is concerned with the user's proper perception of feedback, we approach the design of our haptic display and our various simulations from the perspective of Colgate and Brown. We do not attempt to produce any purely perceptual cues, rather we attempt to make our simulations energetically and physically accurate, believing that to do otherwise leads to strongly deleterious perceptual cues that disturb immersion. Nevertheless, we remain cognizant of the user's perception of any undesirable properties of the simulation or the display's inability to convey a desired impedance.

1.1.3.1. High stiffness. Our most fundamental premise of perception is that constraint surfaces should be energetically passive and very rigid. Thus a control system must render a very stiff impedance, or relationship between motion and force. A stiff impedance is extremely difficult to implement with discrete sampling in time and space. This is particularly true for impedance displays.[2] Hannaford [74] and Brown and Colgate [23] address methods for increasing coupled stability and for guaranteeing passivity between a virtual environment and/or a teleoperator and the master device by proposing the bilateral impedance formulation and virtual-coupling methods respectively. Hannaford recognizes that control of constraints with impedance displays must be stable in the face of the bilateral flow of power and information [75]. The beauty of admittance displays, such as the device introduced in this thesis, is that the control of impedance in the constrained direction is irrelevant and simply provided for by the non-backdrivable display without the need for any feedback control-loop.

[2] *Impedance* displays sense motion and render force, while *admittance* displays sense force and render motion. Impedance displays are highly backdrivable while admittance displays are not.

While rendering rigid constraints is a bilateral scenario for impedance displays, it is merely a unilateral trajectory tracking scenario for admittance displays.

1.1.3.2. High bandwidth. Secondary to the user's ability to perceive constraints on motion, we desire that the user perceive a high dynamic range. From an ergonomic standpoint, a haptic display must not tire the user. Therefore free motion directions must appear completely unimpeded. The user should not feel encumbered by an overly large inertia or by deleterious viscous effects. Users of force-reflecting telemanipulators often complain of fatigue, even when the haptic cues are for their benefit. Thus we adopt another constraint on our haptic display - it must either have inherently low inertia or be able to simulate it. Control of inertial properties is a bilateral process for admittance displays, and the requirements of passivity place a lower limit on the impedance that can be simulated. This is unlike impedance displays that can render ultra-low impedance by merely turning actuators off (although impedance displays can not render an impedance lower than their own inertia and backdrivability). Our admittance display, introduced in this thesis, struggles to render small inertias (lower impedances) but excels at rigid constraints.

1.1.3.3. Masked inertia. The haptic virtual world includes the probe, stylus or tool that the operator is remotely manipulating which is the analog of the haptic display end-effector he/she is actually in contact with. The designer of such a simulation may desire that the operator experience not only the dimensions of the virtual tool through its simulated collisions, but also experience its mass and inertial properties, along with its viscous drag and attractions or repulsion from virtual environment elements. Khatib et al. recognize this fact explicitly, stating that “for haptic interaction to seem realistic to a user, the virtual object should exhibit the same simulated physical properties as the real object, including the dynamics of rigid and articulated bodies, along with such mutual influences as those created by the impact forces during contact...” [97]. Users of a teleoperator typically manipulate a master-manipulandum that may be quite different from the slave’s implement or end-effector in the remote environment. In addition, the apparent inertia of the master-manipulandum end-effector changes with configuration of the haptic display mechanism while that of the virtual tool does not. In order to maintain the operator’s sense of telepresence, we believe that the master-manipulandum’s apparent inertia should be controlled to appear like that of the virtual tool.

It has been the practice of the haptics community, often utilizing impedance displays,[3] to control forces along the contact normals with a constraint, and allow (not control based on a dynamic model) motion orthogonal to these surface normals. In this work, we instead disallow motion along the contact normals and control (not simply allow) motion orthogonal to them based on a dynamic model.

[3] Perhaps because of the predominance of impedance displays and the lack of devices equipped with load cells, very little literature has addressed the haptic display of inertial systems. The vast majority of haptic devices are impedance displays (e.g., the ubiquitous Phantom developed by Massie and Salisbury [112]) - they sense motions applied by users and control force and torque in response. We are not saying that impedance displays cannot simulate inertia, only that it is not commonly done and is difficult to do with any level of fidelity in the absence of a load cell.

1.2. Existing haptic display architectures

While today's impedance and admittance displays may both be used to simulate a wide range of mechanical behaviors, they excel in different areas due to the nature of their control and mechanical structures. If one does an extensive literature search of commercial and research devices, most serial link haptic displays have a maximum stable stiffness on the order of 1-5 N/mm and most parallel haptic displays an upper bound of 15-50 N/mm.[4] Impedance displays can have an unmasked inertia as low as 0.05 kg, while admittance displays have a minimum stable mass of 2-5 kg. Impedance displays are well-adapted to simulating low inertia, low damping environments, since they have low inertia and are highly backdrivable, but have difficulty rendering stiff constraints. On the other hand, admittance displays, which are highly geared and therefore non-backdrivable, are well-adapted to displaying rigid constraints, Rosenberg's so-called *virtual surfaces*, but struggle to simulate unencumbered motion [143, 144]. Unlike impedance displays, admittance displays must actively mask inertia and damping. Although controller complexity is increased, this allows for a higher level of fidelity. Another example of admittance display simulation fidelity is that the user can throw the end-effector and reasonably expect proper behavior even after the device has left his/her hand.

[4] 1 N/mm is generally accepted as the minimum stiffness required to convey the presence of a constraint. 10 N/mm is sufficient to convey the presence of a *hard* constraint.

1.2.1. Admittance displays

Early admittance displays include the haptic display control in the “what you see is what you can feel” visual/haptic registration techniques of Yokokohji et al. [169], the haptic display of the dynamics of virtual object manipulation by Yoshikawa et al. [172] and the use of accelerometers and strain gauges by Paines [136] in order to simulate inertia. Good reviews of admittance displays and their classification are found in Hayward and Astley [77] and Carignan [26]. The Haptic Master from FCS Control Systems is a very good recent implementation of the admittance paradigm [160]. The Virtuose sold by Haption is another admittance serial haptic display with three or six degrees of freedom [76]. The Steady Hand robot at Johns Hopkins is an admittance-controlled, experimental surgical master device [1]. The Cobotic Hand Controller introduced in this thesis is controlled as an admittance display. Although well-engineered admittance devices may have a higher dynamic range than impedance displays, they are rare due to cost and complexity. The required, accurate, multi-degree-of-freedom force sensors are very expensive. The admittance display may also need many more gears and bearings, and tight machining tolerances. Thus, due to cost concerns, the successful commercial haptic displays are often impedance devices with no more than three joint sensors and three actuators designed for applications requiring only three translational degrees of freedom.

1.2.2. Impedance displays

1.2.2.1. Cable driven systems. Most impedance devices utilize cables for power transmission, ideal for reducing mechanism inertia by removing the actuators from the links, for eliminating backlash, and for providing a capstan drive arrangement for ratio reduction. Salisbury et al. [149] developed the cable driven Whole Arm Manipulator (WAM) now sold in four and seven degree-of-freedom versions by Barrett Technology.[5] Massie and Salisbury [112] developed the PHANTOM haptic display, now sold by Sensable Technologies, which has three cable driven translational degrees of freedom and optional gear driven rotational stages. The Delta and Omega haptic displays sold by Force Dimension are three-degree-of-freedom parallel devices, with optional rotational stages [62]. Adams et al. developed the three-degree-of-freedom Cartesian haptic display Excalibur, which also utilizes cable transmissions [3]. Ellis developed a two-degree-of-freedom Cartesian mechanism driven by cables [51]. Burns designed a six-degree-of-freedom parallel-serial cable driven display that, if built, would have retained greater stiffness than the completely serial WAM [25].

1.2.2.2. Direct-drive displays. Others devices are direct-drive systems that eliminate compliant transmissions altogether, although often at the expense of large motors. Quanser commercialized the 3DOF Planar Pantograph [141] and 5DOF Haptic Wand [142] based on technology developed by Salcudean. Lee et al. analyzed a five-degree-of-freedom high bandwidth force-controlled device utilizing traction drives

[5] Bejczy and Salisbury developed one of the earliest teleoperation masters, the JPL Force Reflecting Hand Controller [9].

against five linearly moving rods [104]. Millman and Colgate designed a high stiffness four degree-of-freedom linkage/Cartesian driven display [119]. Adelstein and Rosen developed a high-bandwidth two-degree-of-freedom spherical linkage driven design [4]. Martin and Savall provide an excellent table of 31 different haptic displays and their capabilities [111].

1.2.2.3. Alternative drives. Nearly all haptic displays consist of linkages driven by electrical motors via cables or gears, with the exceptions of magnetic levitation, hydraulics and pneumatics. Kelley and Salcudean [95], Berkelman and Hollis [11], and Salcudean and Vlarr [148] have developed magnetically levitated mice and joysticks. Kazerooni's human power amplifier technology utilizes hydraulics [92]. The UTAH/M.I.T. dextrous hand by Jacobsen et al. [84, 85] and the Rutgers ankle by Girone and Burdea [68] utilize pneumatic actuators. In order to avoid the limited translational range of motion of magnetic levitation devices, the nonlinear control and leakage issues of hydraulics, and the compliance and stored energy issues of pneumatics, most designers of haptic displays attempt to create rigid linkages, coupled by transmissions with minimal backlash, to rotary electric motors.

1.2.3. Passive displays

Some haptic displays are devices that can render passive constraints but cannot move on their own. They are, however, able to guide or limit motion. This is the basic principle of the passive[6] cobot, but several other technologies are available as well. One way for the passive display of constraints is overrunning clutches as is employed in the Passive Arm with Dynamic Constraints (PADyC) device [45]. Controllable brakes utilized in the Passive Trajectory Enhancing Robot (P-TER) are also an option [16, 29]. The brake method dissipates energy and both the clutch and brake methods suffer from limited ability to convey a constraint direction smoothly. Goswami and Peshkin analyze the range of possible behaviors of devices with passive, programmable damping elements in the wrists [70]. However, as the degree of complexity - or capacity for immersion - of virtual reality grows with advances in computing, so to will the requirements on the design of haptic displays and passive devices may not be an adequate solution. Even two or three-degree-of-freedom unpowered cobots have significant friction, and undesirable configuration dependent inertias [168].

1.2.4. Other haptic display types

The devices introduced so far have been intended for gross kinesthetic manipulation in a wrist or arm sized workspace via a single finger or hand. Other displays exist that stimulate numerous other tactile and kinesthetic pathways. Immersion sells arguably one of the most scalable and complex haptic systems available. The CyberForce armature supports the CyberGrasp exoskeleton that fits over the CyberGlove which can contain the CyberTouch vibro-tactile feedback option. The CyberGrasp

[6] Unpowered or *passive* cobots require power to steer transmissions, but cannot cause the joints of the cobot to move. *Active* cobots not only have actuators to steer transmissions, but also have an actuator to provide power to move the links.

also establishes a paradigm shift, the user can grab onto a virtual tool, rather than exploring an environment with a fingertip or pen-type stylus that is represented in the virtual world either by a point or a small sphere. Since the user is grasping a real object in virtual-reality, six-degree-of-freedom rigid body manipulation is now implied, rather than simply stylus-type three dimensional exploration. Grasping of objects requires the display of rotation and the control of torques while exploration with a point only required the display of translation and forces. Intuitive Surgical and Immersion Medical make numerous surgical simulation or teleoperation systems that enable both hands of a user to collaborate on a surgical procedure. Other applications driving novel device design are fingertip displays, video games and locomotion displays. Fingertip displays include the texture display work of Minsky et al. [121], the vibro-tactile display of Howe et al. [80] and the slip display of Salada [147]. Immersion's gaming unit sells or licenses force feedback for all types of personal computer, television and arcade video games. Finally, Hollerbach et al. [79], Frisoli et al. [63] and Kazerooni et al. [92, 94] extend haptic displays to include locomotion devices, as well as upper and lower body exoskeletons.

1.3. Cobots - a novel solution

The word *cobot* is a concatenation of collaborative and robot, meaning shared control between human operator and computer.[7] The development of cobotic technology at the Laboratory for Intelligent Mechanical Systems (LIMS) was motivated by the desire of General Motors to have a load-bearing device that could provide passive guidance for material handling operations performed collaboratively with human operators [5]. While robots excel at performing low degree of freedom repetitive tasks and those tasks requiring significant effort, dextrous tasks such as tucking wires and insulation into place as doors, seats and dashboards are installed into automobiles, may require unreasonable computational and mechanical complexity. The LIMS solution was to design cobotic devices that control the relative velocities of their joints by modulating continuously variable transmissions (CVTs) with small *steering* actuators. Constraints in the transmissions cause cobots to have only a single mechanical instantaneous motion freedom, regardless of the dimension of their configuration space. The dynamics along this single instantaneous motion freedom defined by the CVTs are controlled via a single power injector in the case of the powered Cobotic Hand Controller, or by a human operator in the case of a passive cobot. CVTs utilize the nonholonomic constraints of steered wheels to relate the relative velocities of mechanism links.

[7] Cobots were initially called programmable constraint machines or nonholonomic haptic displays, but Brent Gillespie coined the winning selection *cobot* in a Laboratory for Intelligent Mechanical Systems (LIMS) contest for the name. The technology was first patented as *nonholonomic haptic display* [38] and then as *cobots* [39].

1.3.1. Survey of cobot prototypes

Cobotic CVTs that have been developed to relate two translational velocities, two rotational velocities, or a rotational velocity to a translational velocity, have been utilized in many prototype devices. The use and control of nonholonomic constraints as the basis for passive cobot technology is best summarized by Peshkin et al. [139] and Gillespie et al. [66]. Wannasuphoprasit et al. developed the Cartesian rail Unicycle Cobot and the Scooter Cobot, devices with two-translational and planar workspaces respectively [165]. Faulring modified Scooter into the Pallet Jack Cobot [54, 57]. Also in existence are the Arm Cobot and the 3DP Cobotic Manipulator which have three-translational workspaces, developed by Moore [125] and Wannasuphoprasit and Sirikasemsuk [166] respectively. Cobots utilizing a hemispherical workspace include the Extreme Joystick Cobot developed by Santos-Munné [151] and a cobot developed by Jabre and Gillespie [83]. Brokowski [20, 21] and Kim [99] performed experiments with the Box CVT testbed, originally constructed to relate the x-y motion of a rail-system. Wannasuphoprasit and Cheepsumol built a powered parallel cobotic platform utilizing two spherical CVTs [164]. Wannasuphoprasit and Chanphat utilized hydraulics to realize a passive haptic display [163]. Yambay [159], and then Worsnopp [167], developed UTLA, a two-link arm unicycle cobot, and analyzed the configuration dependent inertia of a passive two-link arm [168]. Chardon developed a vertical UTLA version, a powered two-link-cobot with dual wheels [28]. Emrich and Hodgson proposed an alternate CVT design (from that presented in this thesis) for a parallel six-degree-of-freedom cobot and analyzed a single leg prototype, although adapting this design to a powered device may be somewhat difficult [52]. Hodgson and Emrich also analyzed a different class of cobots termed *minimally constrained*, that have more than one instantaneous motion freedom [78]. Boy et al. [17, 18, 19] and Long et al. [107] analyzed the use of cobot algorithms for powered wheelchair systems.

1.3.2. Rolling contact CVTs outside cobotics

The attractive properties of transmissions based on rolling constraints led to the adoption of such mechanisms outside of the haptics community long before their use for cobots. Dry-friction rolling contact CVTs have been used intermittently in prototype automobiles and limited production vehicles since the full-toroidal CVT was patented in 1877 [81]. The late 1990s has seen Torotrak [64, 133] and NSK [82, 109, 123] culminate research into *traction*[8] fluid CVTs by successfully placing full-toroidal and half-toroidal traction based CVTs into full production vehicles. Goi et al. even analyzed the half-toroidal traction drive for helicopter applications due to the potential for significant noise reductions [69]. Fallbrook [140] is developing a variant of the spherical CVT for applications such as bicycle transmissions and the Milner CVT is yet another variant owned by Intersyn [120]. Sordalen et al. [157], Chung et al. [32, 33] and Nakamura et al. [132] introduced the use of spherical CVTs in a serial link nonholonomic manipulator. Kim et al. analyzed a variant of the spherical CVT [98]. Even a roller ball computer mouse is a simple cobot. Amacoil,

[8] *Friction* commonly refers to dry friction while *traction* refers to rolling contact mediated by a traction fluid.

a motion control company utilizes rotational-to-linear CVTs in their reciprocating motion products. Controllers for numerous types of nonholonomic mobile robots have been developed by researches such as Luo et al. [108], Sarkar et al. [152] and d'Andre-Novel et al. [42].

1.3.3. Smooth power transmission

The attractiveness of rolling contact CVTs is their smoothness of power transmission. Gear trains, timing belt transmissions, hydraulic and pneumatic systems as well as cable systems all have dissipative losses that result in heat and noise generation. In addition, stiction, friction, compliance and backlash in these transmissions add highly nonlinear dynamics to the mechanisms. Cobotic transmissions utilizing bearing quality steel components in dry-friction rolling contact have none of these nonlinearities. Haptic simulations have unusual realism when displayed on the Cobotic Hand Controller. The crisp distinction between free and forbidden directions of motion is a salient feature of cobots. This performance does not arise from elaborate control algorithms, but from the inherent physical characteristics of the device due to the utilization of nonholonomic rolling constraints in its transmissions.

Rolling constraints in the transmission elements, not electrical power, resist forces orthogonal to the current motion freedom. This leads to a natural stability when rendering virtual constraints, since the instabilities that plague conventional haptic displays, which arise from sampling in discrete time and space and exciting structural resonances, cannot impact any control loops in the constraint directions. While traditional admittance displays can impart rigid constraints, and impedance displays excel at low impedance during free motion, few mechanical architectures exhibit the dynamic range of impedances achievable with cobotic transmissions utilizing steel elements in rolling contact.

1.3.4. Additional benefits of cobotic architecture

We find that in addition to smoothness of power, cobotic transmission architecture yields several other benefits. Given a set of design criteria for a multi-degree-of-freedom mechanism, such as maximum flow, maximum effort and maximum power, we find that a cobot can meet these requirements with reduced numbers of high power actuators, reduced size requirements for those actuators and increased power efficiency relative to conventional actuation systems. Parallel cobot architectures require only one common element or power actuator for an unlimited number of degrees of freedom (Figure 1.1).[9]

The actuators that modulate the transmissions for each degree of freedom can be extremely small and low power, often an order of magnitude smaller than the single power actuator. The transmissions draw power from a single common element actuator as needed, thus reducing the weight and power requirements of the mechanism. Only one set of high power electronics and drive-train components are

[9] Cobots are characterized by the architecture of transmission arrangement which may be *serial* or *parallel*. Serial cobots have transmissions between successive links, that relate the relative velocity of each transmission's respective links, or link and ground for the base transmission. Parallel cobots relate all joint velocities through transmissions to the velocity of single common (power) element. Thus the velocity of adjacent joints of parallel cobots are referenced to one another via two transmissions and the common element, not a single transmission as with serial cobots.

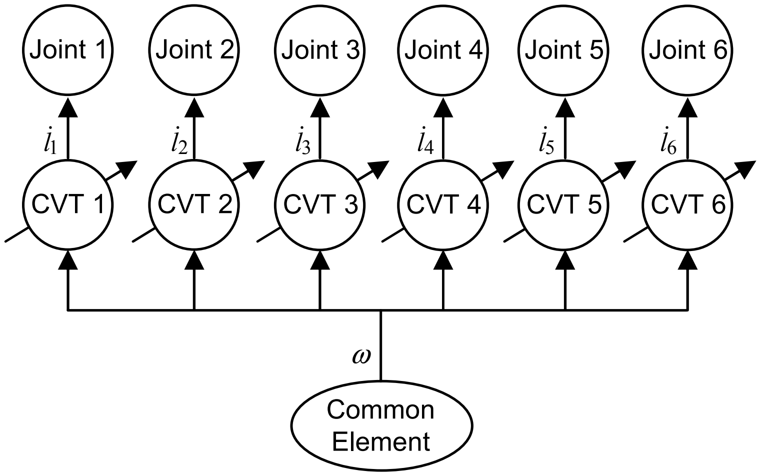


Figure 1.1. An example of parallel cobotic transmission architecture. While there are six joint speeds that must be controlled for the Cobotic Hand Controller to render a virtual environment, there are six CVTs and a common element that must be actuated. It is arbitrary at what speed to have the common element spinning since it is related via CVTs to the joints.

needed. Using a continuously variable cobotic transmission can eliminate the need to make compromises on output flow and effort, which are inherent in choosing a fixed transmission ratio, and also allow the common element actuator to be operated at an efficient speed nearly all of the time. In addition, the cobotic architecture allows for the ability to both clutch or decouple joints without any additional actuators beyond the low-power steering actuator for each CVT. With cobotic architecture, no electrical power is expended to resist forces in constrained directions.[10] Electrical power is spent only to provide effort along the current motion direction. Rolling constraints

[10] This is not strictly true if the common element is rotating, since some power is dissipated to elastic losses (lateral creep).

in the transmission elements, not electrical power, resist forces orthogonal to the current motion direction. Only joints involved in the current motion direction draw off power from the single common element actuator. Cobotic transmissions have a built in safety feature as well. Since they rely on frictional contacts to transmit power, the preload force at these contacts can be set to slip when a certain output force or joint acceleration is exceeded.

1.4. Review of cobot control

Cobot control strategies to date consist of two different control paradigms depending on whether freedom of motion or holonomic constraints are being simulated [165]. Cobots are either operated in *freemode*, where the intent of the operator in the full dimension of the task space is followed completely, or in *virtual-surface* mode, where the cobot confines the operator to travel within the configuration submanifold defined by a holonomic constraint.

1.4.1. Freemode algorithm

In freemode, a unicycle cobot endeavors to simulate a point mass $m_\perp = m_\|$. Therefore, in response to forces, $f_\perp$, normal to the current tangent, the cobot must steer at angular velocity, $\dot{\phi} = \frac{f_\perp}{um_\perp}$, in order to accelerate normal to the current tangent as a function of rolling speed, $u = r\dot{\theta}$. r is the wheel radius and θ the rolling angle. In response to forces along the current tangent, $f_\|$, the cobot must drive the wheel at angular acceleration, $\ddot{\theta} = \frac{f_\|}{rm_\|}$, so as to accelerate along the current tangent. In an unpowered cobot, the wheel is not driven and the operator provides the motive force. Thus for an unpowered cobot, the perceived mass, $m_{\|}$, in the tangent direction cannot be controlled, with the exception of parallel (i.e. redundant) unpowered cobots containing an equal number of CVTs as task space degrees of freedom [151].

1.4.1.1. Redundantly actuated cobots. A parallel cobot, with an n-degree-of-freedom configuration space, only requires that $n-1$ of n rolling contact transmissions be adjustable or continuously variable. The inclusion of n continuously variable transmissions in the Arm Cobot, the Extreme Joystick or the Hand Controller Cobot introduced in this thesis yields a control redundancy (Figure 1.1). This redundancy has been addressed previously. For example, in order to steer the three CVTs of the Arm Cobot, Moore [125] and Kim [99] take the desired ratio of task-space speed to cylinder speed, the desired ratio of task-space speed to drive roller speed, or simply a fixed cylinder speed as one piece of information, and the projection of the three-dimensional force vector orthogonal to the current tangent as two other pieces of information. Thus the three steering actuators of the Arm Cobot move due to two pieces of sensed information and one constraint. The single remaining mechanical degree of freedom moves due to the applied effort of the operator (in a passive cobot) or due to the applied effort of a motor driving the common element (in an active cobot).

The steering redundancy of the Extreme Joystick was to allow any dynamics (including the restoring force of springs) to be simulated via a passive device, provided the kinetic energy of the common element was non-zero, and essentially created a powered cobot. Santos-Munné described an *energy mode* for a passive cobot in which the common element is spinning, and contains kinetic energy a priori [151]. This kinetic energy is used to cancel out the dynamics of the cobot joints, and in his case, the gyroscopic effects of the spherical common element. Thus, the dissipation in the physical cobot determines how long the a priori kinetic energy exists to provide active effects. If the operator works with a virtual inertia more massive than the cobot itself, he/she could contribute to the kinetic energy of the common element and prolong the simulation. Moore [125] and Kim [99] also have a common energy storage element in the Arm Cobot but do not render any impedance properties other than inertia or rigid constraints in either unpowered or powered modes. They did not attempt to implement the a priori energy mode of Santos-Munné, as the dissipation in the Arm Cobot would have quickly absorbed any energy.

1.4.2. Virtual-surface algorithm

In the second control paradigm, that of travelling along a bilateral virtual surface (a holonomic or configuration constraint), the unicycle cobot wheel is steered in order to stay on and tangent to the configuration submanifold defined by the constraint. Steering control of the cobot wheel is based on the feedforward knowledge of the constraint surface curvature, κ_o, and on a feedback adjustment, $\delta\kappa$, composed of position and tangent errors from the desired position and from the tangent of the surface. Thus steering velocity is $\dot{\phi} = (\kappa_o + \delta\kappa)u$. The feedback adjustment, $\delta\kappa$, is necessary to compensate for creep and slip in transmissions that lead to departure from the desired constraint manifold. The wheel is not commanded to simply stay tangent to the surface since this is only a first order solution. Using the knowledge of the surface curvature yields a second order solution with reduced tracking error. Moore demonstrates the ability of the unpowered arm cobot to track one and two-dimensional bilateral surfaces in a three-translational configuration space [126].

1.4.2.1. Reference cobot algorithm. Gillespie et al. described a *reference cobot* method of incorporating the tracking of the closest point on the desired bilateral constraint surface into the control problem statement [66]. This algorithm is for passive cobots where the user moves the actual cobot, otherwise the actual cobot would be tracking the reference instead of the other way around. Gillespie's *reference cobot* method allows a reference cobot position to track the actual cobot position with specified linear dynamics inserted tangent and orthogonal to the constraints. Based on errors in position and tangent between the actual cobot and reference cobot, a linear controller drives a feedback linearized system (about the tracking error) and generates a control vector, U, containing acceleration commands tangent and normal to the constraints. In order to pull out the curvature component of the commanded acceleration normal to the constraints (utilized for steering the transmissions), the projection $(I_{nxn} - TT^T)U$ is utilized to project the control vector orthogonal to the current tangent T. Likewise, the dot product T^TU pulls out the control component parallel to the current tangent to yield an input to the reference cobot integrator. The integrated controls yield new desired positions and velocities that are used for feedback control at the next timestep. While $T^T U$ yields the motion of the reference cobot, the actual tangential acceleration is controlled by the operator.

1.4.3. Physically accurate display of inertia

Whether simulating freemode or a virtual-surface scenario, an algorithm must project the applied forces orthogonal to the current motion direction (and tangent to any virtual-surface) in order to determine the components that cause acceleration normal to the current motion direction. If we are simulating a configuration dependent inertia or have a task space that is not Cartesian, the projection of the applied force will need to be done via the Euler-Lagrange constrained dynamic equations, and not by a simple Euclidean projection or minimum-norm pseudo-inverse. The Euler-Lagrange equations project forces with respect to any inertia matrix, and handle coordinate representations of non-Euclidean topological spaces. This has *not* been discussed previously in the cobot literature. These issues did not arise for $\mathbb{R}^3$ cobots (the Arm Cobot) as only point mass inertias were simulated. These issues did manifest with $SE(2)$ cobots (Scooter) but were "managed" by using a length scale to convert to $\mathbb{R}^3$, a coordinate representation of $SE(2)$.

1.4.3.1. When a pseudo-inverse is acceptable. Another feature of the reference cobot algorithm, is that the integration takes place in a set of parametric coordinates for the constraint submanifolds so as to avoid drift off of the submanifolds.[11] When

[11] This submanifold can arise from embedding a line, plane or sphere in $\mathbb{R}^3$, for example, or from the configuration manifold of a virtual linkage with fewer degrees of freedom that the haptic display.

transferring desired motions to and from the full-dimension task space to the parametric space, a Jacobian and Hessian are required. This Jacobian is not square due to the dimensionality discrepancy, and therefore a Moore-Penrose pseudo-inverse is required. This pseudo-inverse obeys a minimum norm property for Euclidean spaces, but does not, in general, respect conservation of energy (it fails to do so for $SE(2)$ or $SE(3)$). However, as long as one utilizes the constrained Euler-Lagrange equations when computing desired motion from applied force, the pseudo-inverse required in the reference-cobot parametric integration scheme is merely a kinematic projection (change of coordinates) and works well since the dynamics have already been calculated.

1.4.4. Active dynamics - stiffness

It should be noted that no published work in the passive or active cobotic literature designs controllers explicitly for rendering impedance properties other than inertia or rigid constraints. A traditional passive cobot cannot restore its position after a spring has been compressed. Unilateral and bilateral surfaces are typically described as paths, with no spring or damper impedance properties. The exception to this is the *elastic path* mode developed by Long et al. in order to allow deviation from the programmed bi-lateral constraint [107], but this is for a passive cobot, and relies on the operator to provide power to move the cobot back to the surface. Santos-Munné describes the *energy storage* control mode for the Extreme Joystick that would allow for the simulation of any impedance, but this has not been implemented [151].

1.5. Review of cobot transmission modelling

In order to design the highest performance cobot, the dry-friction rolling contact transmission needs to be well understood. Many conflicting design parameters of the rolling contact elements impact the ability to modulate the transmission and drive power through the transmission. Understanding how design parameters such as component shapes and dimensions, material choices and loading conditions affect compliance and dissipation, leads to both mechanical and control choices to maximize the system's Z-width, or achievable stable impedance range [22]. Dissipation also plays a critical roll in the backdrivability of the cobot and in minimizing the power requirements for steering or for driving the comment element.

1.5.1. Motivation to include some damping

Colgate and Schenkel have recognized that physical viscous damping, be it fluid, electrical or mechanical, extends the stable Z-width of a haptic display provided that the actuators can drive through it [40]. Dioliati extends this to include Coulomb friction, which results in less conservative estimates of stable regimes [46, 47]. Mehling et al. [115, 116] and Brown [22] demonstrated these principles by showing the increase in Z-width with the addition of electrical damping and fluid damping respectively, to one-degree-of-freedom crank-type haptic displays. While the presence of physical damping is key, Colgate and Brown note that the limiting bounds on the range of virtual stiffness and virtual damping that can be stably simulated also grow with smaller sampling time and higher resolution encoders [37]. In addition, Brown and

Colgate note that Z-width is not only described by a maximum stiffness for a given damping, but also by a minimum displayable virtual mass [24].

1.5.2. Survey of modelling outside cobotics

Although the incorporation of damping into a haptic display may be useful, as designers, we would like to limit it so that too much electrical energy is not required to overcome it. The dissipative losses of dry rolling friction and lateral and longitudinal creeps have long been studied in the railroad and automotive industry. An early study of the rolling resistance of steel cylinders found that the rolling resistance increases both with load and with surface roughness [72]. Sharma and Reid compared the energy dissipated in rolling relative to sliding and found that rolling is energetically favorable [153]. Zaghzi et al. analyzed losses due to adhesion and the hysteretic compression of rolling of very compliant rubber [174]. Santos analyzed the material stresses, looking for insights as to failure modes during rolling contact [150]. The two authorities in the field of rolling contact elastic and inelastic losses are Johnson [87] and Kalker [89].

Sordalen et al. [157], Chung et al. [32, 33] and Nakamura et al. [132] discuss some of the difficult issues involved in matching preloads and coefficients of friction in friction drives rather than in traction fluid drive systems like those used in automotive transmissions. For comparison, Chung et al. quote traction drive systems as experiencing a coefficient of friction of 0.1. Nakamura et al. use a dry friction coefficient of 0.1 for steel on steel with 120 N loads between 10 mm wheels on a 19 mm sphere.

Moore operates a dry-friction spherical CVT testbed with loads up to 105 N with 12.7 mm wheels on a 50.8 mm sphere, and finds the steel on steel coefficient of friction to be 0.108 when using an expected high friction K-monel alloy [124]. In this thesis we assume a dry-friction coefficient of 0.12 and 250 N loads between 18 mm steel wheels on a 136 mm steel cylinder.

1.5.3. Survey of modelling within cobotics

1.5.3.1. Desired contact patch geometry. After experimentally analyzing dry-friction CVT mechanics and material properties, Bachman [7], Moore [124] and Jabre et al. [83] recommend circular contact patches in order to provide maximum sustainable forces while reducing steering torque. Bachman experiments with very compliant Rollerblade^{TM} wheels of all different profiles and durometer. He finds that increasing either the diameter of the wheels or the profile radius, both of which increase contact patch size, leads to an increased sustainable lateral[12] friction force for the same preload. Moore notes that the wheel itself should have a spherical profile in order to generate a circular contact patch and minimize steering torque. Moore also notes that the cylinder or sphere upon which the wheel rolls should have a larger radius to make the contact patch as circular as possible. Jabre utilizes finite elements code to estimate the performance benefits of changing material properties.

In addition, Moore notes that a key design concern is minimizing the ratio of rolling resistance to lateral force [124]. He states that a purely elastic material that

[12] The term *longitudinal* will refer to the heading of the wheel, while *lateral* is orthogonal to the heading.

has no rolling resistance is desirable in order to increase *constraint smoothness* in the parallel direction, and that one should choose steel rather than polyurethane. He also notes that increasing the sustainable lateral force, termed *constraint hardness*, will likely have more impact on the rolling resistance to lateral force ratio. This lateral force increase can be accomplished by increasing the friction coefficient and/or preload force. However, increased preload force would lead to additional wear and necessitate greater steering effort. Nevertheless, Moore experimentally concludes that a larger contact patch, particularly in the longitudinal direction, yields less transmission velocity error in response to tractive effort. While Moore desired to make the patch long in the longitudinal direction in order to sustain torques on the drive rollers of the spherical CVT, we desire wider contact patches to resist lateral creep in the rotational-to-linear CVT.

1.5.3.2. Kinematic-creep research. In the rotational-to-linear transmissions introduced in this thesis, only lateral forces exist at the contact patch. The axle of the wheel in the rotational-to-linear transmission is not coupled to any load. This is unlike the drive wheels of spherical CVTs whose axles are coupled to input and output shafts that transmit power. These drive wheels sustain longitudinal forces at their contact patches since they apply tractive effort to the central sphere. At the same time, these wheels experience tractive torques about their contact normals with the sphere as they are cornering as they move in an arc about the sphere. They have spin, or relative angular velocity, to the sphere beneath, even at a fixed transmission ratio. The rotational-to-linear transmission also has relative angular velocity between rolling elements, although only when the transmission ratio is changing. The simultaneous presence in the spherical CVT of these longitudinal forces and relative angular velocity introduces a dissipative phenomena called kinematic-creep that does *not* appear in the rotational-to-linear CVT. This phenomena is a combination of longitudinal creep and spin and was alluded to by Moore [124]. Quoting Gillespie et al. in regards to a wheel that is simultaneously undergoing tractive rolling and steering: "If a braking axial torque is applied (balanced by traction on the wheel opposite the direction of rolling), the point of rolling will migrate toward the center of rotation, whereas if an accelerating axial torque is applied (balanced by traction on the wheel in the direction of rolling), the rolling point will migrate away from the center of rotation. Any deviation of the rolling point away from the center of the contact produces a longitudinal creep..." [67].

Gillespie et al. addressed this phenomena with an analytical model of kinematic-creep between rigid bodies [67]. This model is called *complete-slip* in Johnson [87] and Kalker [89]. Gillespie et al. started a vein of research that attempts to characterize the impedance across a spherical CVT from input drive-shaft to output drive-shaft. Brokowski [20, 21] attempted to correlate the results of his extensive experiments with the Box CVT with Gillespie's model but was forced to scale a parameter by a factor of seven in order to get his data to match Gillespie's model, and even then it only matched in limited respects.[13] While Gillespie's model describes a real dissipative loss in the transmission, it is not the only dissipative loss, and thus efficiencies

[13] In a separate document from this thesis, we provide an analysis of the previous research on the spherical CVT, and make several suggestions for future analysis [56].

cannot be accurately predicted. In order to improve the match between data and model, Brokowski compensated for rolling friction at the steering and drive rollers. In addition, Kim noted that lateral creep at the steering rollers, not included in the kinematic-creep model for the drive rollers, contributed to the velocity mismatch [99]. Rolling friction and lateral creep models are the basis of our analysis of the rotational-to-linear CVT as they account for most of the power dissipation in the transmissions of the Cobotic Hand Controller.

1.6. Review of current dynamic simulation methods

Due to the nearly exclusive adoption of the impedance controlled device by researches and end-users alike, the haptics community has often neglected to represent inertial effects and to display smooth rigid bilateral constraints. Very rarely, in the literature outside of the cobot community, does one find fundamentally bilateral constraints. Dynamics are usually done for multiple bodies independently without the embedding of constraint equations. Most often only combinations, of unilateral contacts with a virtual environment or of forces due to potential or flow fields, are considered. In addition, the haptics community has been concerned with the operator's interaction with an environment in three dimensions. In this thesis we review the virtual environment simulation techniques and propose a controller that will allow for the proper representation of inertial dynamics both with and without bilateral constraints, in six dimensions via admittance type haptic displays. Our methods will be general to the simulation of both rigid bodies and linkages subject to holonomic and/or nonholonomic constraints.

1.6.1. Survey of traditional haptic algorithms

There are numerous haptic display algorithms for conveying a virtual environment to the user of a haptic display. A good survey is provided by Gillespie [65]. Two sources for simulation techniques are the haptics literature and the computer graphics/animation literature. In the former, popular approaches to handling constraints are the god-object tracker presented by Zilles and Salisbury [175] and the virtual-proxy algorithm of Ruspini and Khatib [146]. These algorithms track the location of a tool in the virtual world (e.g., the surgeon's scalpel), compute the points of interaction with that virtual world (e.g., a model of the patient) and ultimately compute contact forces due to inter-penetration depths. These forces are then used to compute a total reaction force to be displayed to the surgeon's master manipulandum. Significant in the present context, these algorithms do not normally allow for the control of inertial dynamics tangent to surfaces or while in unconstrained situations. The motion of the god-object or virtual-proxy along a constraint does not obey a dynamic model, but instead moves in response to the user's motion (this is the behavior of a conventionally controlled passive cobot). Initial algorithms employed gradient descent to track the closest active cells or locations on objects described by polyhedral or parametric models. These algorithms have evolved over time, with increasing environment complexity, dimensionality of display, number of objects in the environment, bandwidth of displays, impedance or admittance causality and desire for conveying physical reality.

Much more general algorithms for resolving dynamic collisions as well as the contact state between two rigid bodies have been developed by the computer graphics community. These methods compute constraint forces and impulses, and treat dynamics in all directions, normal as well as tangent [8, 145]. The graphics literature is interested in the point of contact, the dynamics of collisions (static momentum exchange) and even free-motion dynamics. Generally speaking though, these methods are not designed for *hard* real-time (where computation time is always less than the actual integration time step) or for interface to haptic devices.[14] The trajectory has been more important in the graphics literature than the response to user forces, and more often than not, the concern has been the dynamics of the objects in the environment rather than those of the tool. The operator simply feels the inertia of the manipulandum when unimpeded in the virtual environment since the force feedback algorithms are shut off.

1.6.2. Issues with topology

We address two avenues of simulation in this thesis that are largely unexplored in the haptics community - inertial dynamics and six-dimensional simulation routines. Six-dimensional routines are uncommon due to a lack of manipulandum degrees of freedom and mathematical complexity. Duffy notes that the notion of orthogonality

[14] Impulsive methods and event-based haptics have been used in attempts to make up for mechanical, electrical and computational device deficiencies and to address perceptual reality [101, 122, 148].

of forces and velocities via the usual inner product of $\mathbb{R}^6$ is not physically meaningful or coordinate invariant, and that the duality of forces and velocities has often been neglected in the literature [50]. Blajer [14], Liu and Li [106] and Yun and Sarkar [173] work to develop a unified method of combining holonomic and nonholonomic constraints with a matrix-math based geometric methodology for control and modelling of constrained dynamic systems. Constantinescu et al. [41] and Khatib [96] address the accurate dynamic display of forces for virtual linkages with fewer degrees of freedom than the robot manipulator. Their methods do treat forces, velocities and inertia properly. Integration of a simulation in the $SE(3)$ topology leads to complex issues as a consequence of the coupling of coordinates of the rigid-body orientation topology $SO(3)$ unlike in a $\mathbb{R}^n$ system where the coordinates can be integrated independently. Due to a lack of six-dimensional displays, very little mention or implementation of $SO(3)$ or $SE(3)$ integration issues is found in the haptics community, however, such considerations have been made in the mathematics community [13, 15, 34, 43, 129]. These integration difficulties also extend to the simulation of linkages, with topologies that combine translations (sliders) and/or rotations (revolute joints).

General, virtual environment simulation algorithms are provided by Gillespie [65], Lee et al. [105], Baraff [8], Berkelman et al. [12] and Son et al. [156]. Algorithms similar to those we present later are described in [106, 114, 160]. Nahvi et al. [131], utilizing the Sarcos arm and Vanderlinde et al. [160], using the FCS HapticMaster, accurately portray true dynamic models of inertia, but both are limited to three translational degrees of freedom and, thus, do not bridge the complex simulation of rigid-body inertial dynamics. Relatively recently, some haptic displays have been designed with active rotational degrees of freedom. Kim et al. [100] and Johnson and Willemson [86] utilize six-dimensional Phantoms and algorithms to compute the torques on a rigid body that are due to multiple intersections with an environment. They do not simulate inertial dynamics, partially due to lack of a load cell.

1.6.3. Admittance control

Nahvi and Vanderlinde implement admittance control, for which Yoshikawa et al. [171, 172] and Yokokohji et al. [169] are credited for early implementations. Carignan [26] notes that an earlier implementation of admittance control, *position-based impedance control*, was executed by Maples and Becker [110]. Kazerooni refers to other earlier methods of manipulator control as admittance control [91]. Many note that admittance display modes lead to trajectory control rather than force control [26, 35, 113]. Force, once integrated, becomes the input to a trajectory controller that subsequently utilizes position feedback. Carignan summarizes the genesis of the admittance control paradigm and provides a delineation of impedance and admittance control from an implementation standpoint [26]. The use of position feedback in addition to force feedback is his definition of admittance control. Many, including Lawrence et al. [103] and Lee et al. [104], have carried out pure closed loop force control while others, including Kazerooni [91, 93] and Bergamasco et al. [10], have implemented pure compliance control. Yoshikawa also provides an extensive review of the current state of force control research [170].

1.7. Summary of chapters

1.7.1. Focus 1: Device Design

In Chapter 2 we describe the design and kinematics of the Cobotic Hand Controller. Analysis of the workspace, backdrivability and apparent inertia, mechanism stiffness, and resonant modes of the Cobotic Hand Controller are described.

1.7.2. Focus 2: Control

In Chapter 3 we outline the structure of the Cobotic Hand Controller's control system. In Chapter 4 a novel *energy-tracking* cylinder controller is introduced to utilize the actuation redundancy of the Cobotic Hand Controller. In Chapter 5 we discuss the performance achieved with the above controllers in terms of motion control bandwidth, acceleration during unilateral impacts, trajectory tracking and impedance range.

1.7.3. Focus 3: Transmission Modelling

In Chapter 6 we present models of longitudinal and lateral creeps due to elastic behavior of the rolling bodies and of inelastic losses due to free-rolling friction. These are validated by experimentation with the Cobotic Hand Controller. In Chapter 7 we develop a model of the whole rotational-to-linear transmission system, provide a comprehensive bond graph for the system, and analyze the power efficiency of the system compared to an equivalent conventional system. An analysis is performed to determine the relative significance of the modelled power losses of the cobotic system.

1.7.4. Focus 4: Physics Simulation

In Chapter 8 we describe a virtual environment simulation format, common to admittance displays, for propagation in time of a physics simulation subject to holonomic and nonholonomic constraints. Three example constraint scenarios are presented. Results are provided for the these scenarios as implemented with the Cobotic Hand Controller.

1.7.5. Conclusions

Finally, in Chapter 9, we draw conclusions about the Cobotic Hand Controller, its control and performance, attempts to model it, and our virtual environment simulation method.

1.7.6. Summary of appendices

In Appendix A we provide a series of photographs of the Cobotic Hand Controller. In Appendix B we provide a list of vendors and part numbers for key components. In Appendix C we provide an accounting of the files comprising the software for the Cobotic Hand Controller. In Appendix D we analyze the pressure distributions at the contact patch and predict its geometry. In Appendix E we analyze the frequency response of the Cobotic Hand Controller's steering velocity plants via open-loop and closed-loop Bode plots. In Appendix F we analyze the frequency response of the Cobotic Hand Controller's cylinder velocity plant via open-loop and closed-loop Bode plots. In Appendix G we describe the method by which raw load-cell data is compensated for the weight of an end-effector, and transformed to the proper coordinate frame. In Appendix H we review the Euler angle parameterization used and in Appendix I the resulting inertia and Coriolis matrices. In Appendix J we provide the Pfaffian and parametric descriptions of the bilateral holonomic and nonholonomic constraint scenarios implemented with the Cobotic Hand Controller. In Appendix K we provide analytical expressions for the Jacobian and Hessian matrices characterizing the Cobotic Hand Controller's kinematics.

1.8. Publication note

Portions of the material in Chapter 2 were previously published in Faulring et al. [58]. Portions of the material in Chapters 3 and 5 will appear in Faulring et al. [60]. Portions of the material in Chapter 7 were previously published in Faulring et al. [59]. The material in Chapter 8 was previously published in Faulring et al. [61]. Descriptions of the use of the Cobotic Hand Controller as a teleoperation master for a Dual Arm Work Platform stand-in at Argonne National Laboratory are provided in Kang et al. [90], DeJong et al. [44] and Park et al. [138, 137]. Additional material not included in this thesis and otherwise unpublished includes an analysis of previous work and suggestions for future work with spherical cobotic transmissions [56].

Many pictures and videos of the Cobotic Hand Controller and its rendering of virtual environments are available at the Laboratory for Intelligent Mechanical Systems Web site at `http://lims.mech.northwestern.edu/projects/handcontroller/` [55].

CHAPTER 2

Focus 1: Design, Kinematics, Specifications

2.1. Design

The mechanical design of the Cobotic Hand Controller was carried out with the objective of producing a six-degree-of-freedom parallel cobot with high dynamic range. In order to make the Hand Controller a parallel cobot, we needed to incorporate a common cylinder and six wheels, at least five of which were steerable. The high dynamic range (i.e., high bandwidth) was needed in order to make the display of virtual surfaces for decontamination and decommissioning work effective. The constraints had to be convincingly rigid (both perceptively and from a functional standpoint) and the free-motion directions unimpeding and non-tiring. Other design objectives for the Cobotic Hand Controller included the incorporation of recommended characteristics for a high bandwidth force reflecting manipulandum: low inertia, low friction, high stiffness, backdrivability, zero backlash, gravitational counterbalancing, and output force capabilities matched to the human operator [119].

2.1.1. Geometry

The design of the six-degree-of-freedom Cobotic Hand Controller, shown in Figure 2.1, utilizes the kinematics of a parallel platform introduced by Merlet [117, 118].

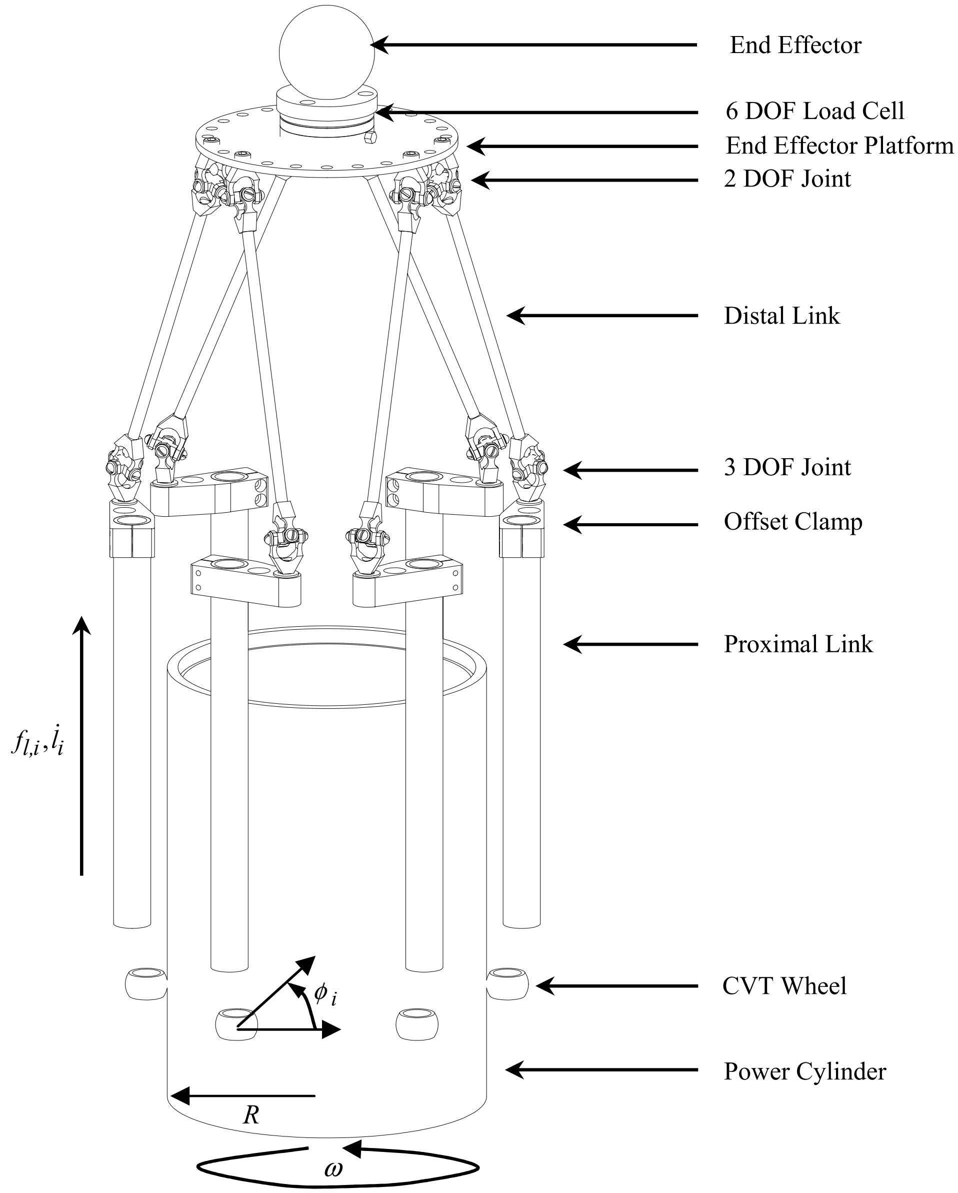


Figure 2.1. The kinematics of a Merlet-Cobotic parallel platform (not to scale - the proximal links and cylinder have been shrunk by about 40 percent in length relative to the distal links). This design consists of six linear actuators arrayed around a central power cylinder. Later figures detail the structure connecting the wheels to the proximal links.

The parallel platform portion of the geometry (i.e., everything but the cylinder and wheels) has also been used in an ophthalmic surgery robot developed by Grace [71] and, in a slightly modified form, in an industrial dextrous assembly robot called the Paradex [128]. The proximal links are coupled by three-degree-of-freedom universal joints to the distal links, and these in turn are coupled via two-degree-of-freedom universal joints to an end-effector platform. Here a force sensor is placed to determine the user's intent. Our addition to Merlet's kinematics has been to couple the six linear actuators to a central power cylinder through nonholonomic rolling constraints.

Linear actuation of the proximal links is achieved via a rotational-to-linear continuously variable transmission (CVT), namely a steered wheel. The angle of each wheel relates the linear velocity, $\dot{l}_i$, of each proximal link to the rotational velocity of the power cylinder. A linearly moving carriage, shown in Figure 2.2, couples each CVT wheel to each proximal link. When the wheels are steered such that their rolling axis is parallel to the power cylinder's ($\phi_i = 0$), a ratio $\dot{l}_i = -R\omega \tan\phi_i = 0$ is set. If the wheels are steered in either direction from $\phi_i = 0$, ratios between ± infinity can be achieved. In practice, wheel slip limits this range. It is also evident that turning all six wheels to $\phi_i = 0$ locks the six actuators, and turning them to $\phi_i = \pi/2$ completely decouples the actuators from the cylinder's velocity, although the cylinder would then be unable to turn.

As it was not a design goal to optimize the kinematic geometry for near-isotropic stiffness or certain workspace requirements, the robot was designed for some degree of kinematic flexibility. Thus the offset clamps (Figure 2.1) adjoining proximal and

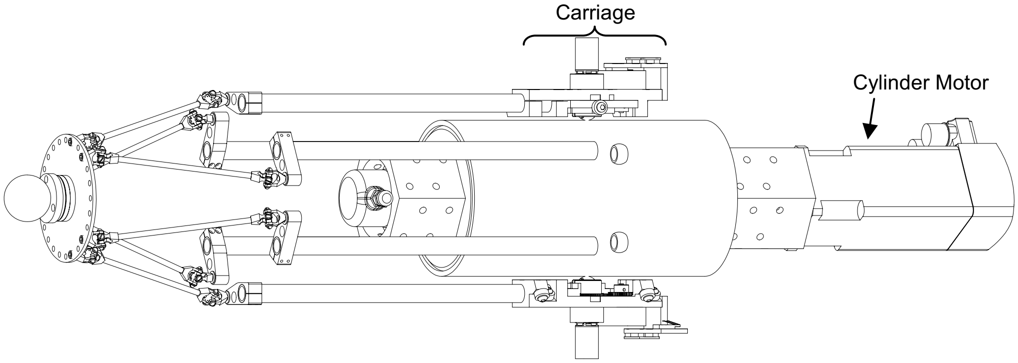


Figure 2.2. In this figure, the motor driving the cylinder is explicitly shown. Two of the steering wheels are exposed. Carriages relating two other wheels to their proximal links are visible.

distal links have two attachment points for the distal links and can be rotated about the proximal links. Rotating them inward yields a larger rotational workspace but reduced stiffness. The mounting positions of the distal universal joints to the end-effector plate are adjustable as well. In addition, the length of the distal links is easily changed as they are made of threaded rod that inserts directly into the universal joints. Dimensions of the current kinematic parameters are provided in Figure 2.3.

The universal joints (Figure 2.4) themselves are remarkable in that they exhibit continuous rotation even when coupling shafts at an angle of 87.5 degrees. This severe operation angle would be detrimental if they needed to transmit power, but here they need only to maintain a kinematic constraint. They were designed and built specifically for the Cobotic Hand Controller and each contains four preloaded radial bearings. These bearings are flanged, and have extended inner rings which fill

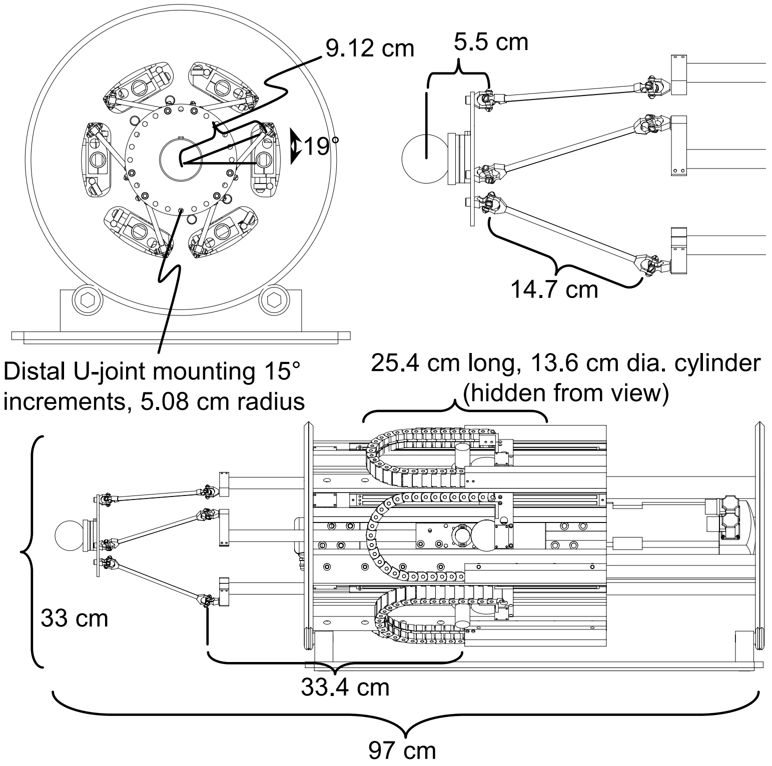

9.12 cm
5.5 cm
19°
14.7 cm
Distal U-joint mounting 15° increments, 5.08 cm radius
25.4 cm long, 13.6 cm dia. cylinder (hidden from view)
33 cm
33.4 cm
97 cm

Figure 2.3. Actual values of kinematic parameters.

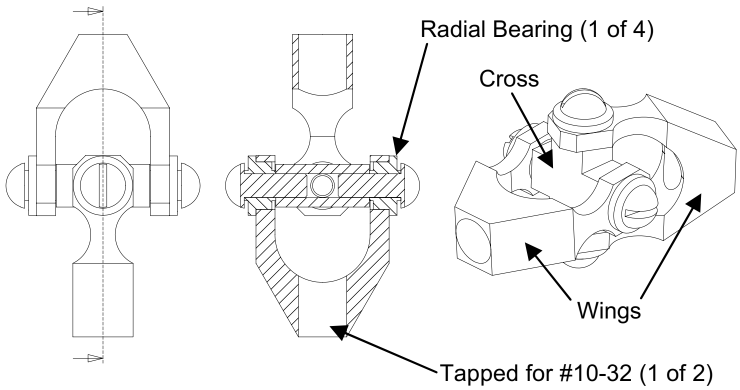


Figure 2.4. Home-made universal joints used to relate the proximal and distal links, and the distal links to the end-effector platform.

the gaps between the screw heads and cross. The bearing flanges slightly compress the wings and thus load the radial bearings.

As shown in Figure 2.5, the two ends of the device are capped by endplates which sit in v-groove rollers. Thus the whole device can be rotated and fixed by a locking pin at increments of 30 degrees for maintenance or kinematic purposes. Each of the carriages can be removed independently if the proximal-distal offset clamp is detached. Wire management guides all wiring (not shown) through the rear endplate. The whole cobot can be turned upright and operated with the cylinder oriented vertical, although significant power (and a fraction of the preload at the wheel) would be utilized to move the joints against gravity.

2.1.2. Six actuator assemblies

The parallel nature of the Merlet-Cobotic mechanism allows for six identical actuator assemblies. As shown in Figure 2.5, there are six equally spaced proximal links and actuator assemblies. These assemblies, depicted in Figure 2.6 and detailed in Figures 2.7-2.11, are bolted to a central core, detailed in Figure 2.12.

All structural components are machined from aluminum with the exception of the proximal links. These are 15.875 mm diameter ceramic tubes chosen for their high strength to weight ratio and because of the confines of the space available. The ceramic tubes at their current length provide 24 cm of workspace along the axis of the cylinder. The upper limit of the workspace is limited by the cylinder's 25 cm

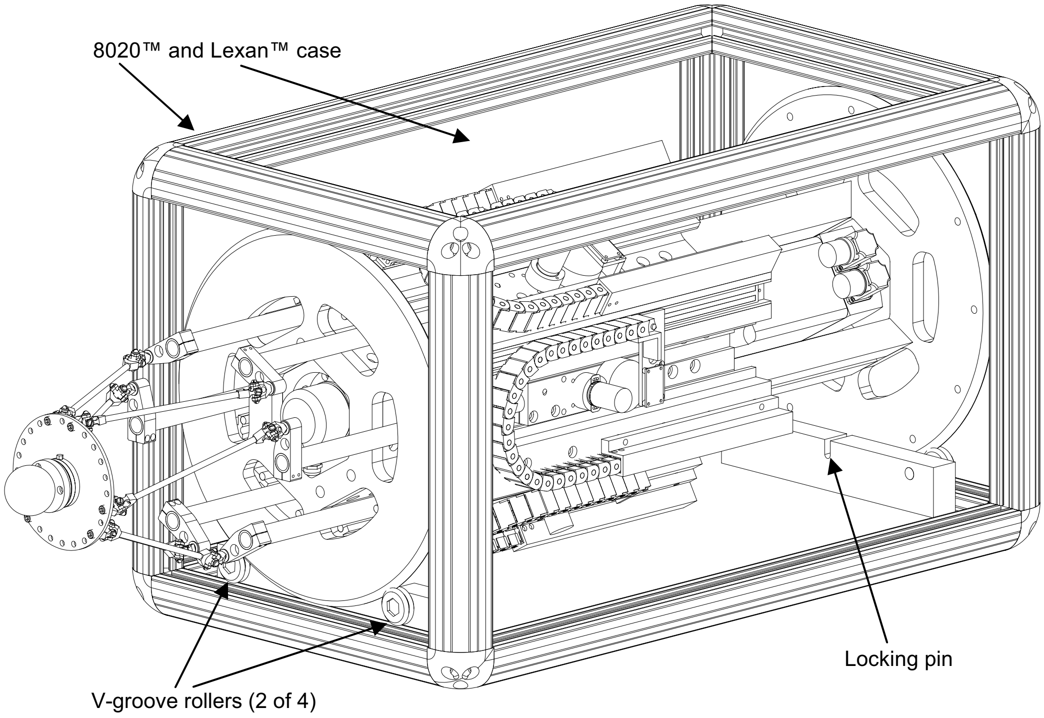


Figure 2.5. A computer aided design rendering of the cobot in its protective case. The cobot is cradled in v-groove rollers to allow easy repositioning of the device for maintenance or kinematic purposes.

length. The moving portion of each joint assembly (depicted in Figure 2.7) has mass $m_l = 0.9$ kg.

A conductive-plastic linear potentiometer (see Figure 2.8) was chosen as a continuous linear sensor over numerous digital incremental options due to its lightweight untethered wiper, as well as for the ability to perform analog differentiation of its output in order to obtain a high resolution velocity signal. Although the circuitry and code were developed to interpret this analog differentiated signal, the signal-to-noise

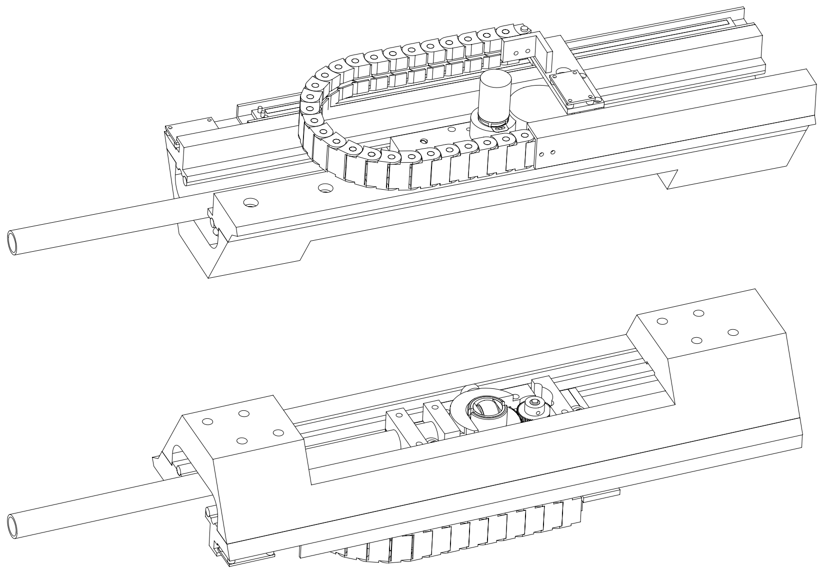


Figure 2.6. Top and bottom views of a linear actuation assembly.

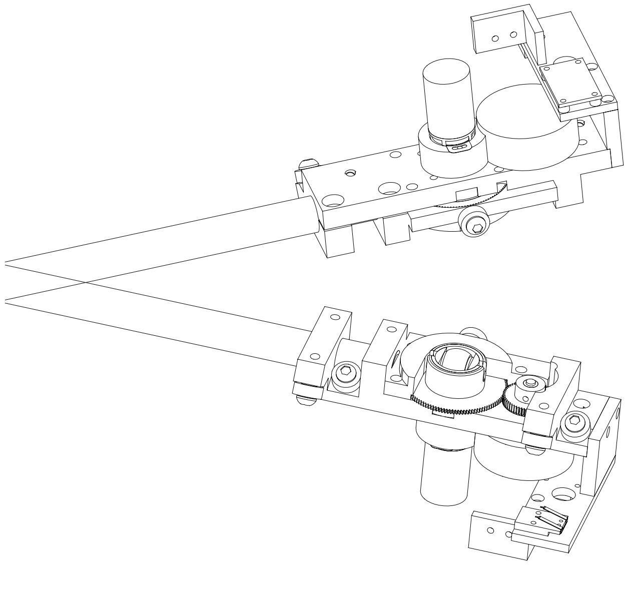


Figure 2.7. Top and bottom views of a carriage and proximal link, which make up the moving portion of the linear actuation assembly.

ratio is such that the finite-differentiated and digitally filtered position signal yields equally good velocity. Ultimately, the linear velocity used is implied from the steering angle of the CVTs rather than the analog differentiated or discretely differentiated linear position signal.

Figure 2.9 details the linear guideway chosen. It was designed to minimize the friction and construction tolerances required for the linear guideway. In addition, we desired to locate the CVT wheel, which is preloaded against the power cylinder, between the two halves of the linear bearing assembly in order to avoid the generation of significant moments. The resulting design utilizes two guide rods and five rollers, four of which are aligned against one guide rod, the fifth against a second guide rod.[1] The sixth point of contact, which constrains the carriage to one or zero degrees of freedom depending on the steering angle, is provided by the cobotic steering wheel.

There are several key advantages of this design. Notably, the guideway does not over-constrain the linear motion of the carriage, and operates without binding (locking up) even when the two guide rods are skew or bent by significant loading. More importantly, the guideway only works when preloaded by the CVT wheel. As the preload at the wheel is increased, the ability of the linear guideway to resist wrenches on the proximal link increases. This effect coincides with the desire for the wheel to provide higher constraint forces. An adjustable preload is provided by placing approximately 25 Belleville disc springs (see Figure 2.10), in some combination of parallel or

[1] While our rollers are radial bearings on button head bolts, no support of axial loading is required and ideally one would utilize needle-roller cam-follower studs in order to increase load capacity and to reduce bearing play.

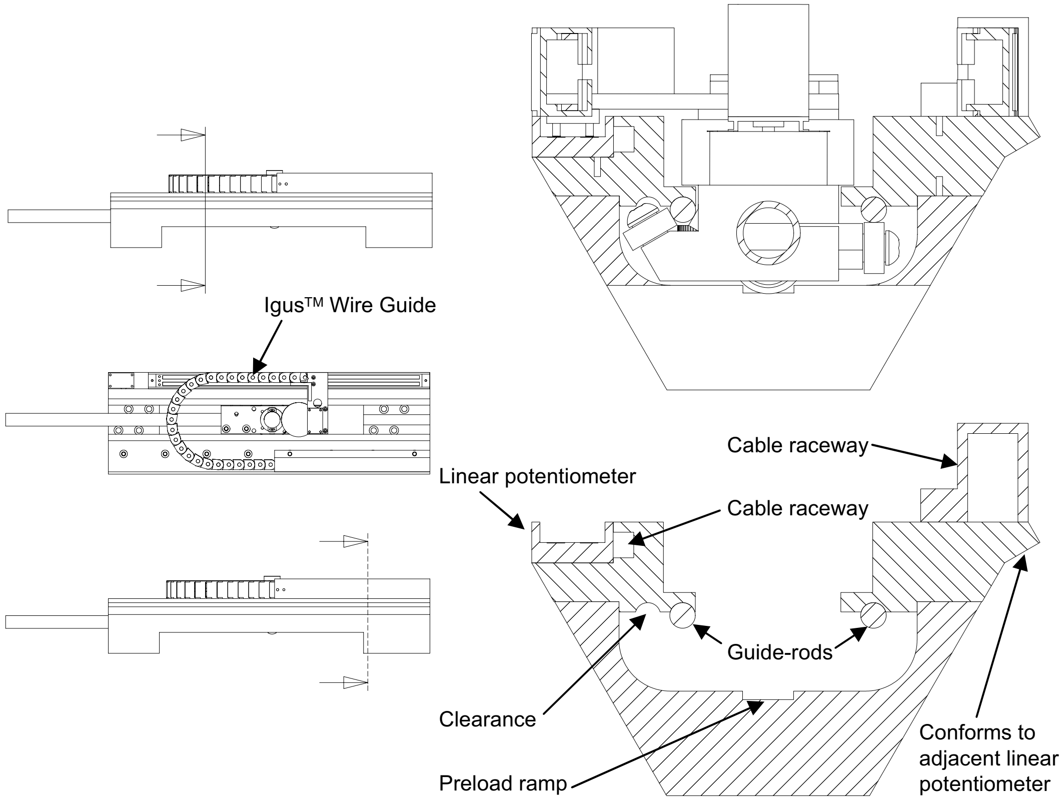


Figure 2.8. Shown is one of six identical actuator assemblies. The proximal link is grounded to a carriage on a linear guideway. An Igus^{TM} flexible wire guide manages wiring for the steering motor and encoder. A ramp allows the carriage to be inserted between the guide-rods and cylinder, with the spacing decreasing gradually as the CVT wheel approaches the cylinder. This allows the application of a preload force by compressing springs within the steering bell (see Figure 2.10).

series, between the yoke of the CVT wheel and its housing. The spring-constant for this set of Bellevilles needs to be such that as the carriage travels from one end of the cylinder to another, changes in the length of the spring do not significantly alter the

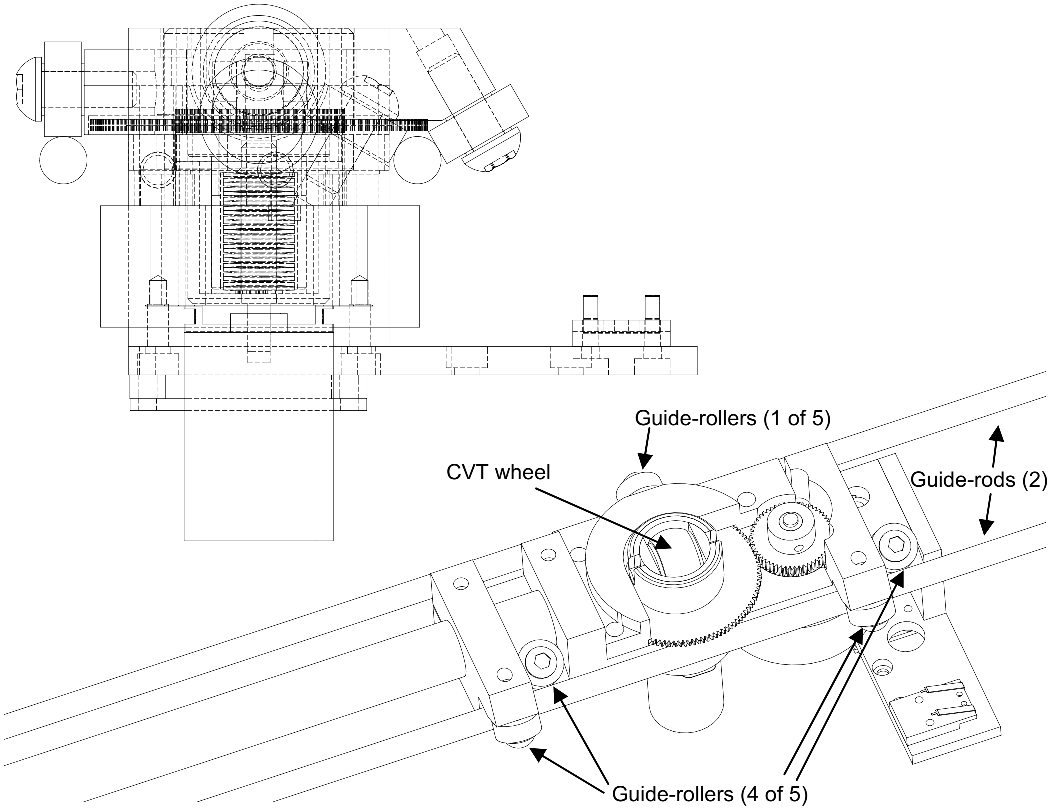


Figure 2.9. Removing the supporting structure of the linear actuation assembly in Figure 2.6 reveals steel guide rods for linear motion. The linear bearing consists of five cam-follower studs, two guide-rods and the CVT wheel.

preload, since the preload will affect the dynamics of steering, linear motion control and cylinder control. Also mounted on the carriage is an optical encoder, a steering motor coupled via gears to the steering bell and a wiper for a linear potentiometer (see Figure 2.11).

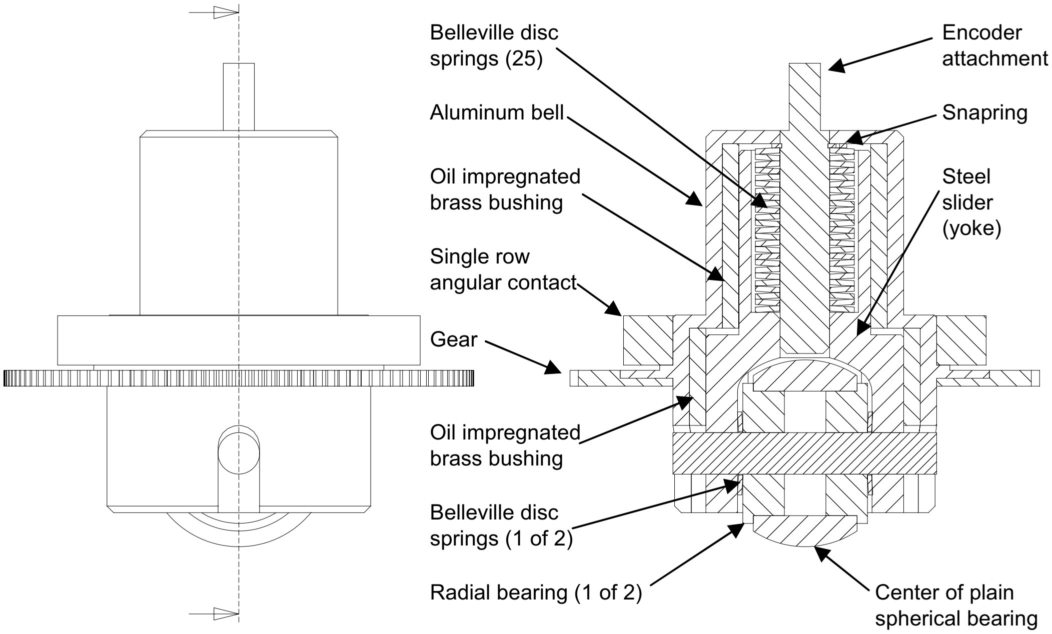

Belleville disc springs (25)
Aluminum bell
Oil impregnated brass bushing
Single row angular contact
Gear
Oil impregnated brass bushing
Belleville disc springs (1 of 2)
Radial bearing (1 of 2)
Encoder attachment
Snapring
Steel slider (yoke)
Center of plain spherical bearing

Figure 2.10. Steering bell features. The yoke supports the CVT wheel axle and is able to slide freely within the bell, guided by the brass bushings normal to the cylinder. If the distance between the linear guideway and cylinder changes over the stroke length, the Belleville springs absorb the change in position of the yoke while maintaining a preload. The wheel axle intersects the bell and causes the wheel to steer as the bell is driven by a gear pressed around it.

Designing wheels for use in cobots has always been problematic. Conflicting design goals when choosing wheel materials has limited wheel performance. It is desired that the wheels provide a high transverse frictional force with minimal preload, yet it is also desired that the wheels have low steering friction in order to allow for smaller steering actuators and higher bandwidth of control. It is also desired that the wheels have low rolling friction and little dissipation in order to provide for backdrivability

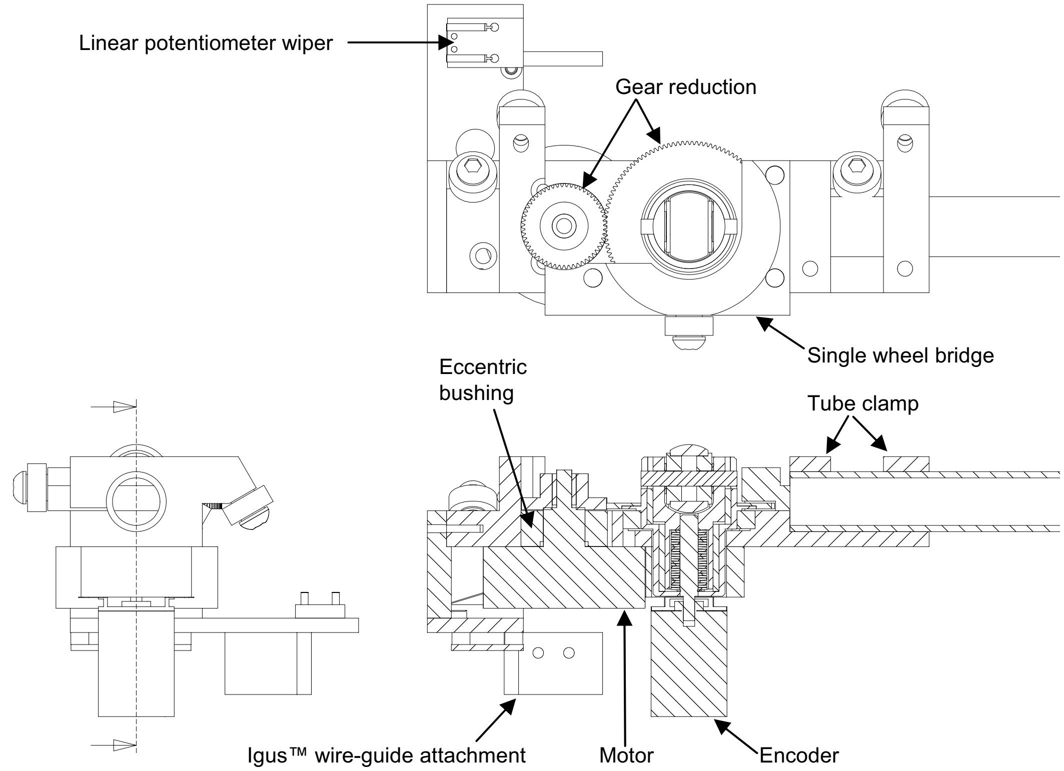


Figure 2.11. Carriage features. Each carriage relates a CVT wheel to a proximal link. It houses the steering motor which drives the steering bell assembly via a single-stage gear pair. An eccentric bushing allows fine adjustment of the inter-gear spacing.

and a reduced power requirement for powered cobots. In addition, the wheel should not have any compliance transverse to the rolling direction.

Previous cobots have typically utilized polyurethane Rollerblade^{TM} wheels (75 mm in diameter) in order to obtain the necessary transverse coefficient of friction. However, this was not an available option for this device since the wheels are 18 mm in diameter. Also, the linear guideways required the existence of a high preload (unnecessary for the high friction polyurethane wheels). With these two constraints in mind, steel wheels were chosen to run against a steel power cylinder even though the coefficient of friction of steel on steel is an order of magnitude less than that of polyurethane on steel. Depending on performance needs (transverse friction or resisting of wrenches on the guideway), more or less preload can be utilized. Currently the preload P is set to around 250 N, with a maximum design limit of 300 N. The CVT wheels have a spherical profile and are the centers of plain spherical bearings with a hardness of Rockwell C 58. We find the rolling coefficient of friction (for support of lateral forces), μ, between the steel wheels and steel cylinder to be around 0.12. This is the point at which lateral creep breaks down into gross slip.

2.1.3. Power cylinder

As shown in Figure 2.12, the power cylinder is located between two mating blocks. The steel cylinder is 25 cm in length, 13.64 cm in diameter, and has a 6.25 mm wall thickness. The cylinder shell has been welded to its end-caps, and these to the shaft. The cylinder/end-caps/shaft were then hardened to Rockwell C 59.8 and cylindrically ground between centers to a 12 μinch finish. This surface yields no cylinder wear and insignificant wheel wear after a quick break-in period. After a few hours of use, the wheels, originally with a black-oxide coating, have a shiny flat strip 880 μm across. Even after 9 months of intermittent use in the lab, the strip is no larger than 910 μm across, which amounts to a total of 11 μm of wear off the radius of the wheel (see Figure 6.2 for a sketch of this wear geometry).

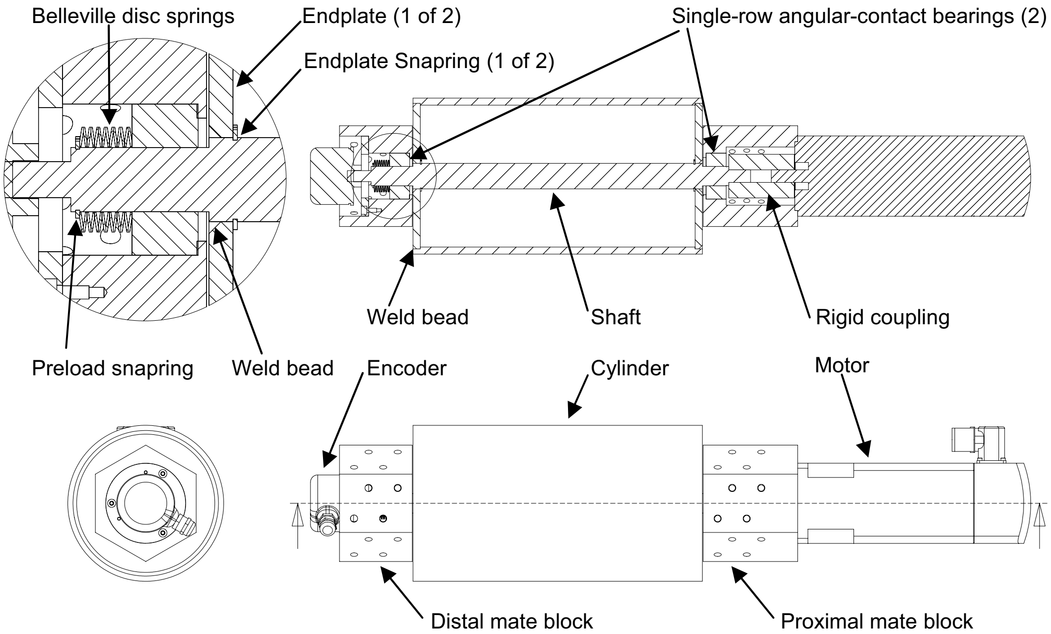


Figure 2.12. A power motor drives the power cylinder. The cylinder shell is welded to its endplates and these to the shaft. A series of Belleville disc springs load the cylinder bearings.

Each of the six linear actuator assemblies bolt to the mating blocks. Also connected to the blocks are the power motor and a high resolution encoder. The motor is connected via a rigid coupling to the cylinder. A flexible coupling was originally present, but later removed to avoid unwanted resonances. A large 1400 watt motor was chosen as it was readily available in the lab and had sufficient torque to operate without gearing. The original goal was to mitigate backlash, thus allowing smooth operation including reversal of direction, and to allow backdrivability if the system was ever operated passively. Ultimately, control algorithms were never implemented to take advantage of this feature, and preload and speed limitations have only allowed us to draw about 60 watts of mechanical power from the cylinder, five percent of its capacity. Assuming we had a 70 percent efficient gear-train between a much smaller motor and the cylinder, a 72 watt cylinder motor would have sufficed.

2.1.4. Sensor and actuator specifications

Table 2.1 summarizes the specifications of the sensors and actuators. All motors are brushless DC operated in torque (current) mode. The linear position and force are recorded via 16 bit ADC boards. The linear position signal is differentiated in analog with a two-pole differentiator followed by a Maxim^{TM} MAX7401 eight-pole Bessel filter-on-chip with programmable cutoff. With this cutoff set at 1000 Hz, and the resulting analog voltage discretized by a 13-bit ADC, linear velocity resolution of 1.0 mm/s is achieved for the proximal links at 1000 Hz.[2] Additional vendor and part number information is given in Appendix B.

2.1.5. Electronics enclosure

In order to ensure reliability and portability[3], all the necessary electronics, including the cylinder motor and steering motor amplifiers and power supplies, the control computer, the main circuit board and the force sensor conditioning box were placed

[2] Although this elaborate scheme eliminates the need to differentiate and filter in software, doing just that with the 16-bit position signal from the potentiometer yields a velocity signal with equivalent resolution and time delay to the hardware version.

[3] The Cobotic Hand Controller was taken to Argonne National Laboratory twice, in order to integrate it with the teleoperation slave system.

Table 2.1. Sensor and actuator specifications.

Sensor	**Resolution**	**Linearity**	**Range**
Power Cyl Enc	140,000 cnts/rev	NA	2π
CVT Encoders	40,960 cnts/rev	NA	2π
Linear Pots	16 bit ADC (5μm)	1/2000	300 mm
Force-Torque	16 bit ADC (2.5 mN)	1/3000 (±5 N), 1/70 (±40 N)	±40 N, ±2 Nm
Actuator	**Peak Torque**	**Cont Torque**	**Wattage**
Power Cyl Motor	13 Nm	3.7 Nm	1400[a]
CVT Motors	260 mNm	54 mNm	30

[a] This oversized motor was chosen as it was readily available and eliminated the need for gearing and the associated backlash and nonbackdrivability. Ultimately control modes never took advantage of these features, and only 60 mechanical Watts have ever been asked of the cylinder motor.

in a single *black box* measuring 41x46x53 cm. A single AC110V power cord provides electrical power for the whole system. External jacks are present for steering motors (x6), steering encoders (x6), linear potentiometers (x6), a cylinder motor, a low-resolution cylinder encoder, a high-resolution cylinder encoder, a force sensor and a dead-man's/E-stop switch. There are also jacks for a keyboard, mouse, monitor and ethernet cord. Switches independently turn on cooling fans, the computer, the steering motor amplifiers, and the cylinder motor amplifier.

2.1.6. Software

A 1.53 GHz AMD standard personal computer running QNX 6.2 operates the control system. An oscillator and counter, on one of three data acquisition boards, is used to generate hardware interrupts at approximately 2000 Hz, to which all data acquisition and output is latched electronically.[4]

A *c*-code program was developed to run the cobot from a QNX platform. Appendix C outlines the lines of code required for each portion of the code. The graphical interface was text based, utilizing the *curses* library. Critical sensor and algorithm states are portrayed on the screen for general system health and/or debugging purposes. A scrollable window displays all program messages. Single key-stroke commands are used to invoke mode changes, data-logging and parameter adjustments. The program allows the cobot to serve as a master manipulator for teleoperation, virtual-reality and pure haptic scenarios. It was used as a driver file to allow the Cobotic Hand Controller to operate transparently as a proxy of the teleoperation interface and virtual fixture workstation of DeJong et al. [44, 90, 137, 138]. Another application of the program was a virtual-reality demonstration in which the user could see the virtual tool and virtual environment as rendered on a Windows platform with OpenGL (also developed by DeJong). Finally, the program was most often used for rendering of simple haptic constraint scenarios and the testing of various control algorithms. The program is modular such that adding a new controller or constraint scenario only involves a few lines of code.

[4] Board IO and algorithms that run at the full 2000 Hz take about 60 μs and 220 μs respectively. Writing data to disk, network communication and updating the GUI are performed at lower rates, and are lower priority threads.

2.2. Kinematics

Let us define as the forward kinematics of the Cobotic Hand Controller as the functions, $\vartheta(l)$, that take us from joint space coordinates, l, to task space coordinates, x.

$$x = \vartheta(l) \tag{2.1}$$

In this section we demonstrate how to evaluate the inverse and the forward kinematics of the Cobotic Hand Controller.

2.2.1. Inverse kinematics

The kinematic diagram shown in Figure 2.13 shows the relationship between the joint locations, $l_i, i = 1 \ldots 6$, and the task frame coordinates, $x_j, j = 1 \ldots 6$ of the manipulandum frame M. The manipulandum coordinates are given by three translational coordinates (x_1, x_2, x_3) and three Euler angles (x_4, x_5, x_6). We have chosen an Euler angle set such that all singularities are outside of the workspace for our specific manipulandum. Euler angles allow us to work in generalized coordinates rather than with the special orthogonal group $SO(3)$. Only one of the six joint coordinates, l_i, is shown in Figure 2.13. Also note that c_i is the fixed length of the link connecting the i'th distal and proximal universal joints. The task frame X is located at a convenient position along the cylinder axis with $\hat{x}_3$ aligned with the linear motion of the joints,

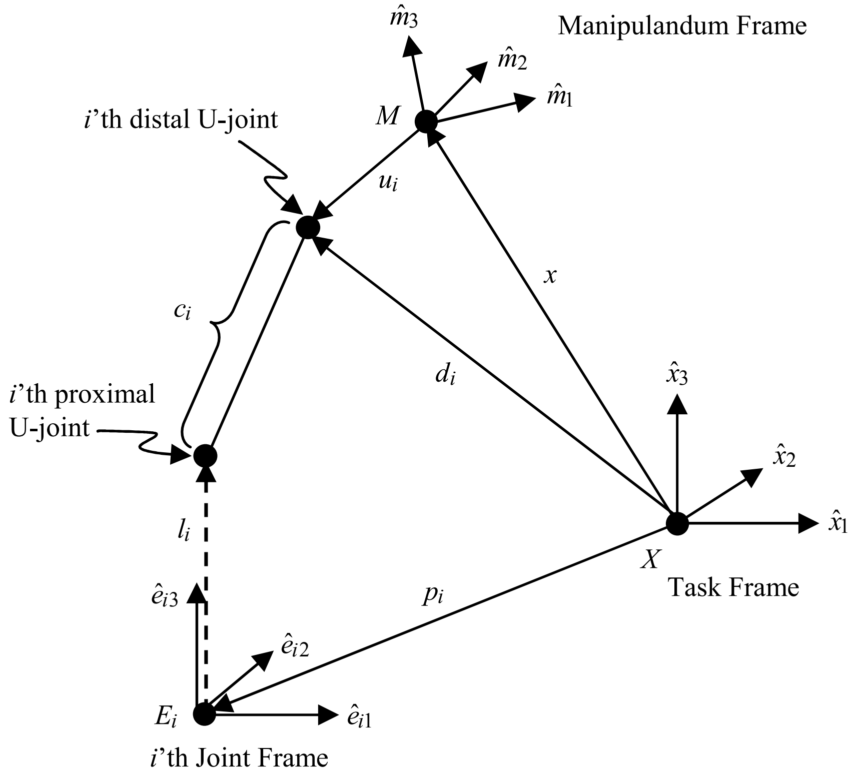


Figure 2.13. Kinematic variables. Distal link lengths c_i are fixed values. Distal universal-joints are fixed in frame M at positions u_i. The manipulandum (end-effector) is located at position and orientation x in task space frame X. Proximal joint frames E_i are fixed in task frame X. The dotted arrow shows the linear motion l_i of the i'th joint.

$\hat{e}_{i3}$. The kinematic frames and vectors are overlaid on a sketch of the Cobotic Hand Controller in Figure 2.14.

Inverse kinematics are trivially deduced, relating an end point position, x, in task frame coordinates to the joint space extensions, l. An equation for the known fixed

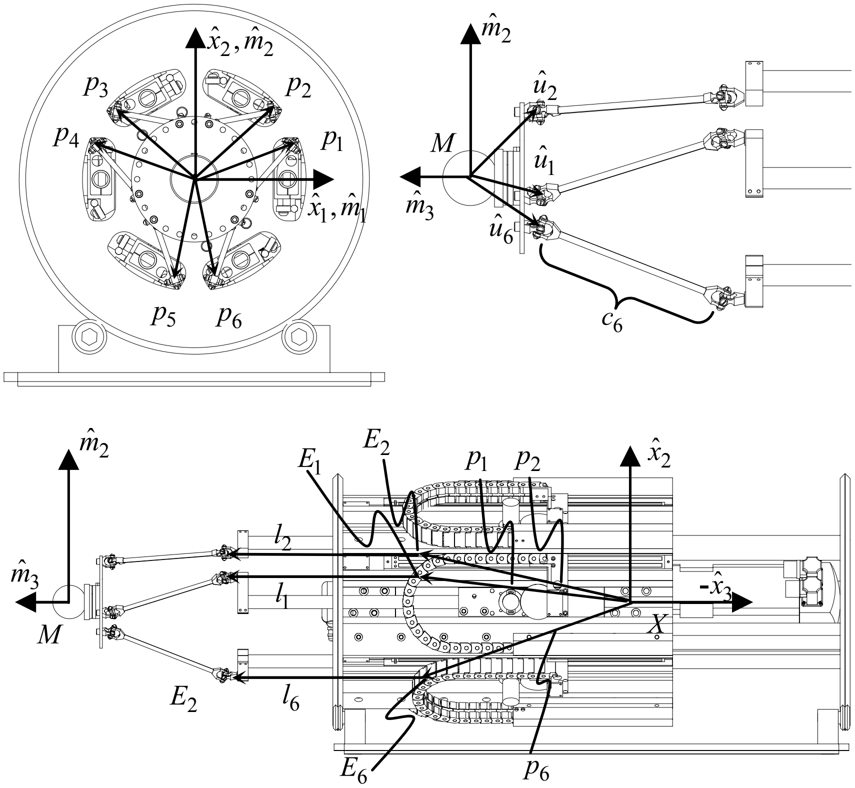


Figure 2.14. The frames and vectors of Figure 2.13 overlaid on schematics of the Cobotic Hand Controller.

length c_i of the proximal link is employed. Solving for joint positions $l_i\hat{e}_{i3}$ we obtain:

$$l_i\hat{e}_3 = d_i\hat{x}_3 - p_i\hat{x}_3 - \sqrt{c_i^2 - (d_i\hat{x}_1 - p_i\hat{x}_1)^2 - (d_i\hat{x}_2 - p_i\hat{x}_2)^2} \tag{2.2}$$

This is equivalent to $l_i = \vartheta_i^{-1}(x)$. The $\hat{}$ notation indicates a frame basis (unit) vector and implies a dot product with the preceding vector, thus pulling out the component in that direction.

Equation 2.3 depicts how the task frame locations of the distal universal joints, d_i, are found by rotating the manipulandum frame coordinates of the distal universal joints, u_i, by Euler angles x_4, x_5 and x_6 and translating this result by $[x_1, x_2, x_3]^T$. The Euler angle description and rotation matrix $R(x_4, x_5, x_6)$ from manipulandum frame M to task frame X is given in Appendix H.

$$d_i = \begin{bmatrix} & x_1 \\ R(x_4, x_5, x_6) & x_2 \\ & x_3 \end{bmatrix} \begin{bmatrix} u_i \\ 1 \end{bmatrix} \tag{2.3}$$

The constant vectors, p_i, give the locations of the proximal universal-joint frames, E_i, in the task frame X. The extension of the proximal universal-joints from their home positions in the $\hat{e}_{i1} - \hat{e}_{i2}$ plane are l_i in the $\hat{e}_{i3}$ direction.

The Jacobian $J(l) = \frac{\partial \vartheta(l)}{\partial l}$ relates motion in joint space, $\dot{l}$, to motion in task space $\dot{x}$ (Equation 2.4).

$$\dot{x} = \ J(l)\dot{l} \qquad J(l) = \tfrac{\partial \vartheta(l)}{\partial l} \tag{2.4}$$

While we do not have an analytical expression for $J(l)$, its inverse $J^{-1}(x) = \frac{\partial \vartheta^{-1}(x)}{\partial x}$ is easily established from the expressions $l_i(x) = \vartheta_i^{-1}(x)$, and relates velocities $\dot{l}$ and $\dot{x}$ (Equation 2.5).

$$\dot{l} = J^{-1}(x)\dot{x} \qquad J_{ij}^{-1}(x) = \frac{\partial \vartheta_i^{-1}(x)}{x_j} \tag{2.5}$$

A series of six, 6×6 Hessian matrices $H_i^{-1}(x), i = 1 \ldots 6$, can also be established that expose individual joint accelerations $\ddot{l}_i$ from general task space accelerations $\ddot{x}$.[5]

$$H_{i,jk}^{-1}(x) = \frac{\partial^2 \vartheta_i^{-1}(x)}{\partial x_j \partial x_k} \tag{2.6}$$

$$\ddot{l}_i = \sum_{j=1}^{6} J_{ij}^{-1}(x)\ddot{x}_j + \sum_{j=1}^{6}\sum_{k=1}^{6} H_{i,jk}^{-1}(x)\dot{x}_k\dot{x}_j$$

2.2.2. Forward kinematics

For the general case of a six-degree-of-freedom parallel manipulator, if no pairings (intersections of universal joints) exist at the platform or base, a closed form analytical solution is not available for the forward kinematics. In fact, twelve solutions are possible for task space coordinates for a given set of joint coordinates without using any heuristics about collisions or range of motion.

In practice, a Newton-Raphson iterative scheme, Algorithm 1, is used to compute the task space coordinates, x, given measured joint coordinates, l, and an initial estimate for the task space coordinates, x_o. This algorithm yields the actual task space coordinates from which to compute the Jacobian and Hessian. Currently, only one iteration is needed at each control interval in order to converge to a reasonable residual, a consequence of the 2000 Hz rate of execution of the algorithm relative to the bandwidth of actuation.[6]

[5] At runtime the inverse Jacobian (36 terms) and the Hessians (216 terms, 90 of which are unique and non-zero) are computed analytically at each time step. The analytical expressions for these are provided in Appendix K. The Jacobian $J = (J^{-1}(x))^{-1}$ is also computed via a numerical routine. It is needed to map forces from task to joint space in order to compute a feedforward cylinder torque.

[6] It will become apparent that knowledge of the actual task space coordinates is unnecessary and the Newton-Raphson scheme is not needed. The Cobotic Hand Controller tracks a desired trajectory

Algorithm 1 Newton-Raphson algorithm for forward kinematics

```
Require: $tol$
Require: $x_o$ initialized to a reasonable guess
Require: Set $x \rightarrow x_o$
repeat
    Measure $l'$
    Set $tol \rightarrow |\delta x|$
    while $|\delta x| \geq tol$ do
        Evaluate $\tilde{l}(x) = \vartheta^{-1}(x)$ and $\tilde{J}^{-1}(x) = \frac{\partial \vartheta^{-1}(x)}{\partial x}$
        Evaluate $\tilde{J}(x) = (\tilde{J}^{-1}(x))^{-1}$ via a numerical inverse
        Evaluate $\delta x = \tilde{J}(x)(l' - \tilde{l}(x))$
        Update $x = x + \delta x$
    end while
    Delay for haptic timestep $dt$
until haptic simulation ends
```

2.2.3. Joint to cylinder kinematics

The body fixed frame N of the i'th wheel moves along the cylinder parallel to $\hat{e}_{i3}$ and $\hat{x}_3$ (Figure 2.15). Basis directions $\hat{n}_{i3}$, $\hat{n}_{i2}$ and $\hat{n}_{i1}$ point normal to the cylinder, along the rolling direction, and lateral to the rolling direction, respectively. $\hat{e}_{i3}$ and therefore $\dot{l}_i$ lie in the $\hat{n}_{i1} - \hat{n}_{i2}$ plane, with $\hat{n}_{i1}$ rotated from $\hat{e}_{i3}$ about $\hat{n}_{i3}$ by the angle ϕ_i. The cylinder of radius R has angular velocity, ω, and is driven by torque, τ_c, about the $\hat{x}_3$ axis. The wheels of radii, r_i, have rolling angular velocities, $\dot{\theta}_i$, and are steered by torques, $\tau_{b,i}$, about the $\hat{n}_{i3}$ axis.

in task space and we map this desired position, velocity and acceleration to joint space. This is done by utilizing a Jacobian and Hessian computed from the *desired* task space location. Then our feedback control is implemented in joint space. Thus we do not need to map the *actual* joint space location to task space via Newton-Raphson.

Equation 2.7 relates input and output flows for each rotational-to-linear transmission.

$$\frac{\dot{l}_i}{\omega} = -R\tan(\phi_i) \tag{2.7}$$

Here we have neglected flow losses due to elastic creeps in the transmission which are modelled in Section 6.2. The input and output efforts of these transmissions are related by Equation 2.8.

$$\frac{f_{l,i}}{\tau_c} = \frac{1}{R\tan(\phi_i)} = -\frac{f_{w,i}}{\tau_c} \tag{2.8}$$

$f_{l,i}$ is the external load applied to the joint along the $\hat{e}_{i,3}$ direction while $f_{w,i}$ is the net force applied by the cylinder on the wheel (after compensating for joint friction and inertia) along the $\hat{e}_{i,3}$ direction. Here we have neglected the effort losses due to rolling friction in the transmission and the CVT wheel axle bearings which are modelled in Sections 6.3 and 6.4 respectively.

2.3. Dynamics

2.3.1. Apparent inertia from cylinder's perspective

In order to provide a feedforward torque for the power cylinder, we compute the hand controller's dynamics from the cylinder's perspective. We equate the applied power of the cylinder motor, $\tau_c\omega$, and of the user, $f_l^T\dot{l} = ((J^{-1}(x))^{-T}f_x)^T\dot{l}$, with the power

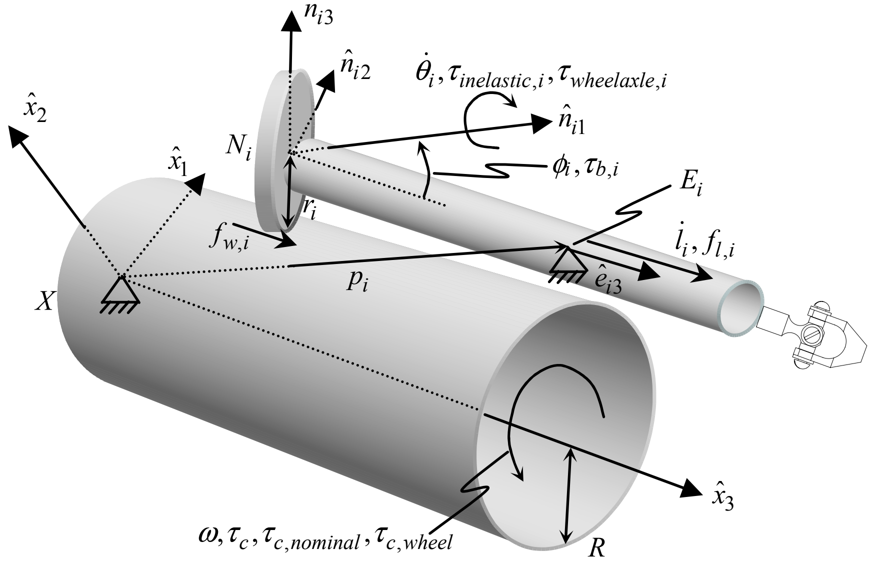


Figure 2.15. Definition of flow and effort variables in the joint, CVT and cylinder frames. Note that $\hat{p}_i$ is a constant vector, locating the origin of frame E_i (from which joint location l_i is measured) in frame X. Power $f_{l,i}\dot{l}_i$ is the output of the joint. Force $f_{w,i}$ of the cylinder on the wheel in the e_{i3} direction has an opposite sign from $f_{l,i}$ in a static situation, and the two differ in magnitude due to the joint inertia and joint friction. Nominally, the CVT wheels can not sustain torques about the $\hat{n}_{i1}$ axes since their axles are not coupled to a load, but rolling friction and bearing friction require that they be driven by torques, $\tau_{inelastic,i} + \tau_{wheelaxle,i}$. The cylinder frame torque required for these losses is $\tau_{c,wheel}$. Thus, power, $(\tau_c - \tau_{c,wheel})\omega$ (we consider only one joint here), gets through the transmission to the joint. Torque $\tau_{c,nominal}$ is required to combat friction in the cylinder bearings, and is the torque required to drive the cylinder in the absence of any CVT wheels.

gained by the cylinder inertia and joint masses, $\omega J_c \dot{\omega} + \dot{l}^T (I_{6\times6} m_l)\ddot{l}$ (Equation 2.9).

$$\tau_c \omega + ((J^{-1}(x))^{-T} f_x)^T \dot{l} = \omega J_c \dot{\omega} + \dot{l}^T (I_{6\times6} m_l)\ddot{l} \qquad (2.9)$$

The joints each have mass, m_l, while the cylinder has inertia, J_c. The task frame force applied by the user is f_x. Rolling friction of the wheel-cylinder interface, lateral creep and linear guideway friction could all be included in order to more accurately estimate the power cylinder torque, although we only focus on the inertial terms here for brevity. While $(J^{-1}(x))^{-T}$ could be simplified to $J^T(x)$, we leave explicit the fact that we have an analytical expression for $J^{-1}(x)$ and must invert this numerically before transposing.

The vectors $\dot{l} = -R\omega \tan(\phi)$ and $\ddot{l} = -R\dot{\omega} \tan(\phi) - R\omega \sec^2(\phi)\dot{\phi}$ are simplified via the vector expressions $\Upsilon = -R \tan \phi$ and $\Xi = -R \sec^2(\phi)$, and ω is eliminated from Equation 2.9 to yield Equation 2.10.

$$\tau_c + ((J^{-1}(x))^{-T} f_x)^T \Upsilon = (J_c + \Upsilon^T (I_{6\times6} m_l)\Upsilon)\dot{\omega} + \omega(\Xi^T (I_{6\times6} m_l)\Upsilon)^T \dot{\phi} \tag{2.10}$$

The cobot has the scalar inertia $(J_c + \Upsilon^T (I_{6\times6} m_l)\Upsilon)$ and the 1x6 Coriolis vector $\omega(\Xi^T (I_{6\times6} m_l)\Upsilon)^T$. The cylinder motor sees an inertial torque, $(J_c + \Upsilon^T (I_{6\times6} m_l)\Upsilon)\dot{\omega}$, when attempting to accelerate the cylinder. Likewise, the cylinder motor sees a Coriolis and centripetal torque, $\omega(\Xi^T (I_{6\times6} m_l)\Upsilon)^T \dot{\phi}$, when the transmissions are modulated. The cylinder also combats a user applied torque of $((J^{-1}(x))^{-T} f_x)^T \Upsilon$.

We could also include the computed torque term for the acceleration of the end effector platform inertia in addition to that of the joints and cylinder, however, we ignore this term since it is a comparatively small mass and involves complex kinematics.

2.3.2. Apparent inertia from operator's perspective

If the dynamics are considered from the user's six-dimensional perspective rather than the one-dimensional perspective of the cylinder, we cannot solve for the apparent inertia since it will be infinite in the constrained directions. This is manifest in the requirement of an inverse of the singular matrix $\Upsilon\Upsilon^T$.[7] Nevertheless, we can approximate the configuration-dependent minimum inertia from the operator's perspective. The most unencumbered the cobot can feel like in the translational direction along the cylinder's axis is $6m_l$, the sum of the six joints. To feel any lighter would require a powered cobot or stored energy, particularly if we consider friction in the cobot. Otherwise, in an attempt to feel light, the wheels would steer to $\pm\frac{\pi}{2}$ and decouple the joints, hindering the cobot's ability to render a haptic simulation.

Continuing to isolate the apparent inertia of the cobot while avoiding the inversion of the singular matrix, $\Upsilon\Upsilon^T$, we develop the following analysis: consider that we can operate the parallel cobot in a mode where we desire to passively simulate the virtual inertia, $M(x)$. Simulation of damping is trivially incorporated here as well. Specifying a desired $M(x)$ is the closest we can get to the apparent inertia of the cobot as seen by the operator. Due to the passive nature of the proposed system, we desire that the power added to $M(x)$ to be equal to the power added to the inertia of the cylinder and joints (less the power dissipated by friction in the cobot, although we do not

[7] Constantinescu et al. provide an excellent literature review and analysis of how to choose an appropriate pseudo-inverse or projection routine when simulating infinite inertia along constrained directions utilizing an impedance device [41].

include this friction here for brevity) as described by Equation 2.11.

$$\dot{x}^T M(x)\ddot{x} = \dot{l}^T (I_{6\times 6} m_l)\ddot{l} + \omega J_c \dot{\omega} \tag{2.11}$$

2.3.2.1. Control of tangential dynamics with a passive cobot. This section elaborates on Equation 2.11 and is purely tangential and speculative. It outlines Algorithm 2, a method for passive haptic display of tangential dynamics with a redundant cobot. Santos-Munné covers similar thoughts in more detail [151].

Given that we know $\dot{l} = J^{-1}(x)\dot{x}$ and $\ddot{l} = J^{-1}(x)\ddot{x} + \dot{x}^T H^{-1}(x)\dot{x}$ from Equations 2.5 and 2.7, Equation 2.12 solves for cylinder acceleration, $\dot{\omega}$, the cylinder acceleration required to simulate inertia $M(x)$.

$$\dot{\omega} = \frac{\dot{x}^T M(x)\ddot{x} - \left(J^{-1}(x)\dot{x}\right)^T (I_{6\times 6} m_l)\left(J^{-1}(x)\ddot{x} + \dot{x}^T H^{-1}(x)\dot{x}\right)}{J_c \omega} \tag{2.12}$$

$\ddot{x}$ is computed by applying measured operator forces to a physics model of $M(x)$, and $\dot{x}$ is computed by integrating the desired acceleration $\ddot{x}$.[8] $J^{-1}(x)$ and $H^{-1}(x)$ are computed from the desired position, x, obtained by integration of $\ddot{x}$. ω is the only measured quantity in Equation 2.12 (although $\ddot{x}$, and therefore $\dot{x}$ and x utilize the measured operator interaction force).

The steering velocities (Equation 2.13) required to simulate inertia $M(x)$ are computed via the time derivative of Equation 2.7.

$$\dot{\phi} = -\frac{\ddot{l} + R\dot{\omega}\tan(\phi)}{R\omega \sec^2(\phi)} \tag{2.13}$$

[8] Chapter 8 contains much more information on our simulation methods.

Accelerations $\ddot{l} = J^{-1}(x)\ddot{x} + \dot{x}^T H^{-1}(x)\dot{x}$ are derived from the physics model and $\dot{\omega}$ is derived from Equation 2.12. ϕ and ω are measured parameters. We have an obvious singularity in this controller when $\omega = 0$. The singularity is approached whenever power needs to flow to the joint inertias, but the cylinder has no kinetic energy from which to draw upon. As we will see in Chapter 4, we discard the class of cylinder controllers that ever allow $\omega = 0$.[9] Feedback control is a very difficult concept in this context, particularly as the cylinder speed approaches zero. The dual desires to emulate the simulated mass and yet keep the transmissions away from $\phi = \frac{\pi}{2}$ conflict. However, this might be a good starting point for analysis of a controller that requires the least amount of cylinder torque.

Algorithm 2 Control of tangential dynamics with a passive cobot

```
Require: Timestep $T$
Ensure: Kinematics from task to joint space $J^{-1}(x)$ and $H^{-1}(x)$
Ensure: Desired inertial properties $M(x)$
repeat
    Measure $\omega$, $\phi$ and $f_x$
    Solve dynamic equations for $\ddot{x} = M^{-1}(x) f_x$
    Integrate to yield $\dot{x}$ and $x$
    Solve Equation 2.12 for $\dot{\omega}$
    Check that required cylinder torque $\tau_c \approx 0$ via Equation 2.10
    Evaluate $\ddot{l} = J^{-1}(x)\ddot{x} + \dot{x}^T H^{-1}(x)\dot{x}$
    Solve Equation 2.13 for $\dot{\phi}$
    Utilize closed-loop steering velocity controllers
    Delay for haptic timestep $T$
until haptic simulation ends
```

[9] This is acceptable for us since we only operate the Cobotic Hand Controller with the cylinder powered.

2.4. Backdrivability

In order to backdrive the Cobotic Hand Controller, an operator has to overcome joint guideway friction, rolling friction at the wheel-cylinder interface and friction in the wheel and cylinder bearings. The joint frame force due to guideway friction, $c_{d,l}P\text{sgn}(\dot{l})$, is established in Section 2.4.1. P is the preload and $c_{d,l}$ is the Coulomb friction coefficient describing joint guideway friction. The joint frame force needed to backdrive the CVT wheels is $\frac{\tau_{inelastic}+\tau_{wheelaxle}}{r\sin(\phi)}$ for each wheel, where r is the radius of a CVT wheel, $\tau_{inelastic}$ the CVT wheel frame rolling friction from inelastic losses at the wheel-cylinder interface (see Section 6.3), and $\tau_{wheelaxle}$ the CVT wheel frame friction in the wheel axle bearings (see Section 6.4). The joint frame force needed to backdrive the cylinder bearings is $\frac{\tau_{c,nominal}}{R\tan(\phi)}$, where $\tau_{c,nominal}$ is found in Section 6.5. If all the wheels are steered at $\phi = \frac{\pi}{2}$ such that the cylinder does not spin, an operator would have to apply effort $6(c_{d,l}P\text{sgn}(\dot{l}) + \frac{\tau_{inelastic}+\tau_{wheelaxle}}{r\sin(\phi)}) = 6(0.84+0.11) = 5.7$ N in order to backdrive the six joints. If the wheels were steered at $\phi = \frac{\pi}{4}$, an operator would have to apply $6\left(c_{d,l}P\text{sgn}(\dot{l}) + \frac{\tau_{inelastic}+\tau_{wheelaxle}}{r\sin(\phi)}\right) + \frac{\tau_{c,nominal}}{R\tan(\phi)} = 6(0.84+0.16)+1.23 = 7.23$ N of effort.

2.4.1. Linear guideway friction experiment

2.4.1.1. Purpose. To characterize the friction force of the linear guideways in order to estimate the power efficiency of the cobot and the required cylinder torque.

2.4.1.2. Methods. With all six transmissions present, a single joint is commanded to execute a sinusoidal motion at low frequency. The cylinder torque is recorded during this motion, and is in excess of the torque required for zero linear motion. This excess cylinder torque, multiplied by its angular velocity, yields the power dissipated by joint guideway friction. This excess rotational power can be integrated over time and divided by the linear distance travelled during that same time to yield an average Coulombic joint friction force. Negligible power goes to counteract the inertia of the joint due to the low speeds at which the experiment is performed. Likewise, the steering angle $\phi \approx 0$, due to the low joint speeds relative to the cylinder speed, avoids confounding variations in rolling dissipation which are a function of $\sec(\phi)$.

2.4.1.3. Results. The experiment was performed for all six joints (one at a time) and yielded an average value of 0.84 N for each guideway (Figure 2.16). For a 250 N preload on the guideways, this yields a dynamic coefficient of friction of $c_{d,l} = 0.0034$, which purely describes the 5 guide-rollers of the linear guideway, and not the CVT wheel.

2.4.1.4. Discussion. A Coulomb friction model of joint guideway friction force, $c_{d,l}P\text{sgn}(\dot{l})$, is utilized to characterize the backdrivability of the Cobotic Hand Controller's joints. Variability in the measured friction force for each joint may be due to varying preloads among joints due to machining and assembly tolerances.

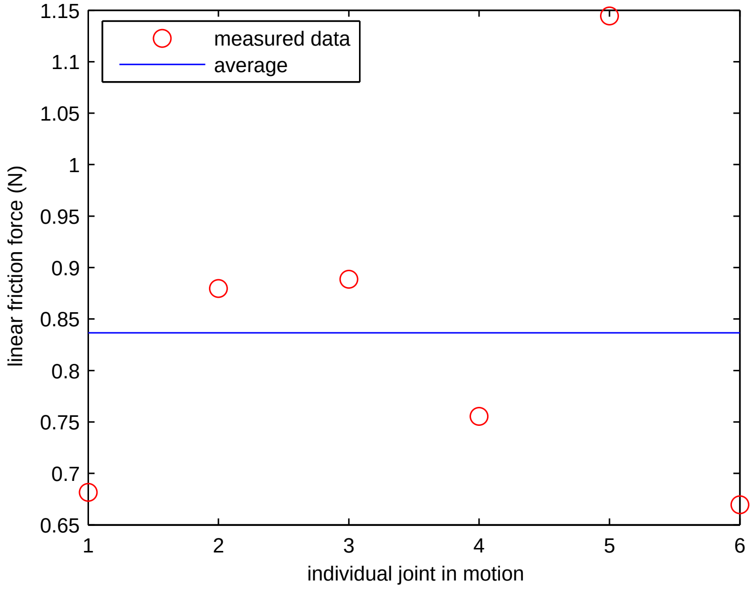


Figure 2.16. Determination of linear guideway friction. Each joint was tested individually.

2.5. Workspace analysis

Situations that limit the workspace of the Cobotic Hand Controller are distal-joint to distal-link collisions, distal-link to platform singularities (occurring when the link lies parallel to the platform), distal-link to proximal link singularities (occurring when the links are orthogonal), proximal link stroke range, and universal joint operation range. These limits are from a collision and singularity standpoint only, and do not take into account manipulability or stiffness. Projections are needed in order to portray the six-dimensional workspace in two or three dimensional plots. We follow Wang [162] and Chuckpaiwong [31] who give excellent workspace analyses

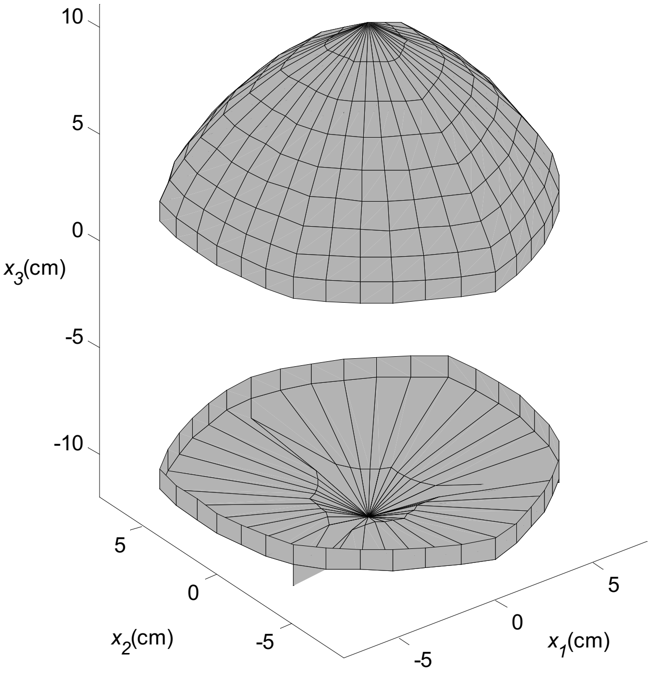


Figure 2.17. Translational workspace without allowing rotation. The workspace is best approximated by an 8 cm radius hemisphere stacked on top of a 16 cm diameter, 13 cm long cylinder. Thus the workspace has a relatively flat bottom and a domed top. The three-fold symmetry of the proximal-distal link connection points is apparent in the grooves on the bottom, and in the slightly hexagonal shape of the cylinder and dome.

of the Paradex manipulator, a device with similar kinematics to the Cobotic Hand Controller. In Figure 2.17 we show the translational workspace of the Cobotic Hand Controller provided no rotations are permitted. In Figure 2.18 we show the rotational workspace of the Cobotic Hand Controller provided no translations are permitted. In Figure 2.19 we show the translational workspace throughout which ±15 degrees of

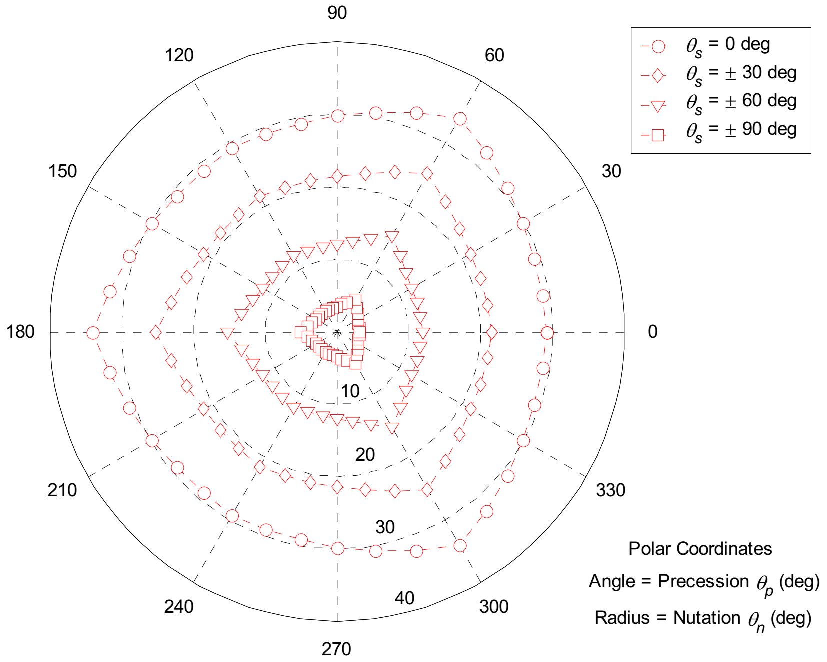


Figure 2.18. Rotation workspace without allowing translation. Nutation θ_n is the angle by which the end-effector is tilted away from the axis of the central cylinder. Precession θ_p is the direction in which the nutation occurred. Spin θ_s is the spin of the end-effector about the nutated and precessed axis. The various contours on the polar plot are for different fixed amounts of spin, with the radius of the contour indicating the amount of nutation, and the angle of the contour the precession. If we desire spin $\theta_s = \pm 90$ degrees, the amount of nutation allowed is only 2-3 degrees, depending on direction of precession. If we only desire spin $\theta_s = \pm 30$ degrees, the amount of nutation allowed is 22-25 degrees, depending on direction of precession.

orientation are allowed. These figures are for the current kinematic configuration of the Cobotic Hand Controller detailed in Figure 2.3.

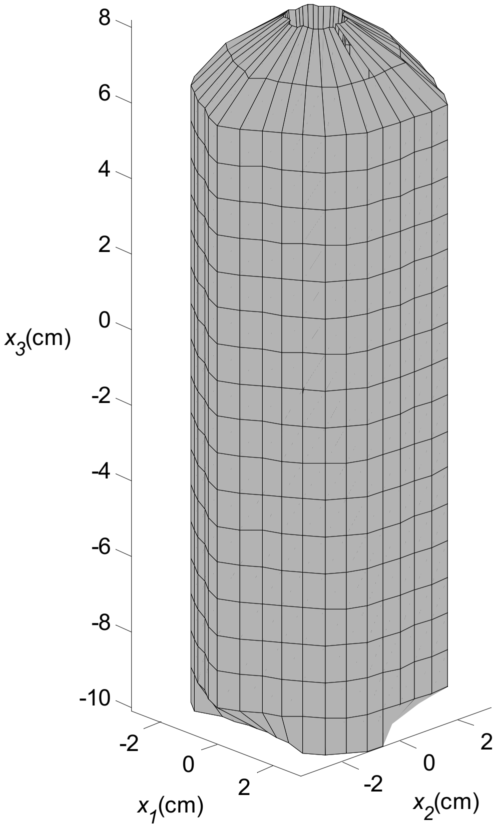


Figure 2.19. Translational workspace given a mandatory rotational workspace of spin $\theta_s = \pm 15$ degrees, nutation $\theta_n = 15$ degrees, in any direction of precession $0 \leq \theta_p \leq 2\pi$. The workspace is a 17 cm long by 6 cm diameter cylinder.

2.6. Mechanism stiffness

Structural stiffness of the Cobotic Hand Controller in x_3 translation is approximately 400 kN/m throughout the workspace. Structural stiffness in x_1 and x_2 ranges from 50 kN/m in the center of the workspace to 20 kN/m away from the x_3-axis and rotated at extreme angles. The stiffness goes to zero in one or more degrees of freedom when the Jacobian becomes singular. These measurements were made by loading the platform with a spring-scale and measuring deflections with a dial indicator.

2.7. Vibration mode analysis experiment

2.7.1. Purpose

To predict the potential bandwidth of the Cobotic Hand Controller by examining the resonant modes of its mechanical linkages.

2.7.2. Methods

In order to gain some understanding of the resonant modes of the long cantilevered beams of the proximal and distal links, we choose to analyze the frequency content of the acceleration of the end-effector due to impulses delivered from a hammer. The blows were delivered to the end-effector proximal to the load cell. The load cell, in conjunction with a known mass attached distal to it, was used as an accelerometer. Three force channels were sampled at 2000 Hz for ten seconds during which 9 hammer strikes occurred. Three strikes were along the x_1 axis, three along the x_2 axis and three along the x_3 axis. Data for each force channel was divided into 39 segments of 512 samples, and the spectrum was taken via FFT in Matlab for each segment. Subsequently, these 39 spectra were averaged. This was done for all three force channels.

2.7.3. Results

The acceleration time traces for the impulse responses are shown in Figure 2.20. The frequency spectra are shown in Figure 2.21.

2.7.4. Discussion

The natural frequency for a 0.5 Kg end-effector with 400 kN/m stiffness is 142 Hz $\approx \frac{\sqrt{400000/0.5}}{2\pi}$. The natural frequency for a 0.5 Kg end-effector with 50 kN/m stiffness is 50 Hz $\approx \frac{\sqrt{50000/0.5}}{2\pi}$. The experimentally measured spectra are consistent with earlier reported stiffness and mass measurements.

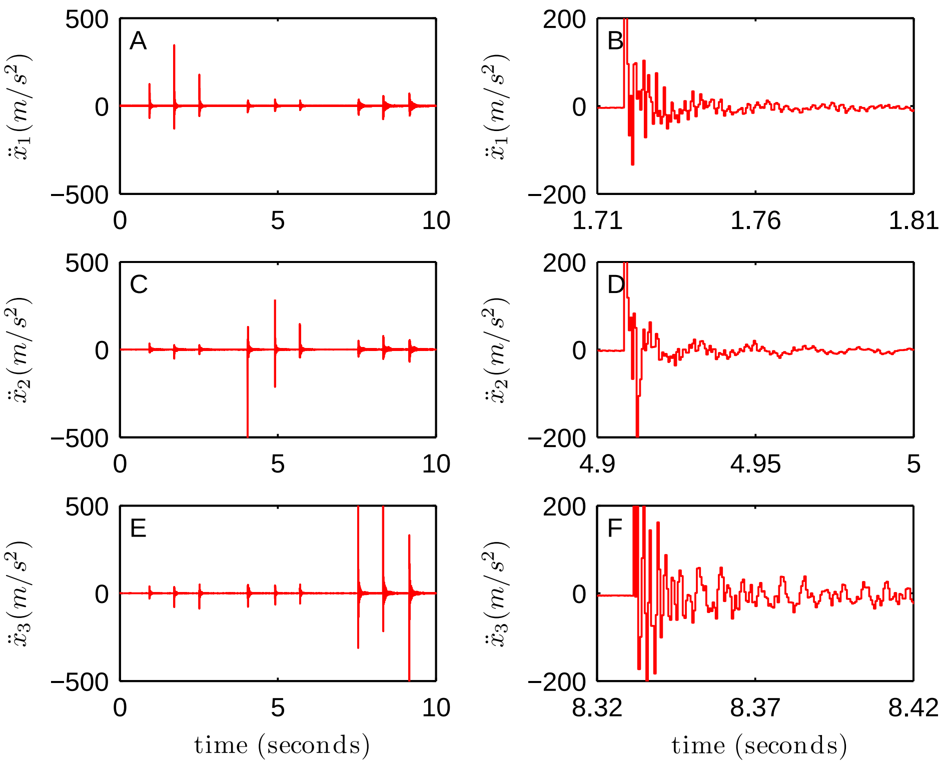


Figure 2.20. Acceleration of the Merlet platform portion of the cobot in response to hammer strikes. Plots A, C and E depict the time traces for nine hammer strikes, three in the positive x_1, followed by three in the positive x_2 and three in the positive x_3 directions. The x_1 and x_2 axes are orthogonal to the cylinder axis while the x_3 axis is parallel to the cylinder. Plots B, D and F show the resonances for a single hammer strike.

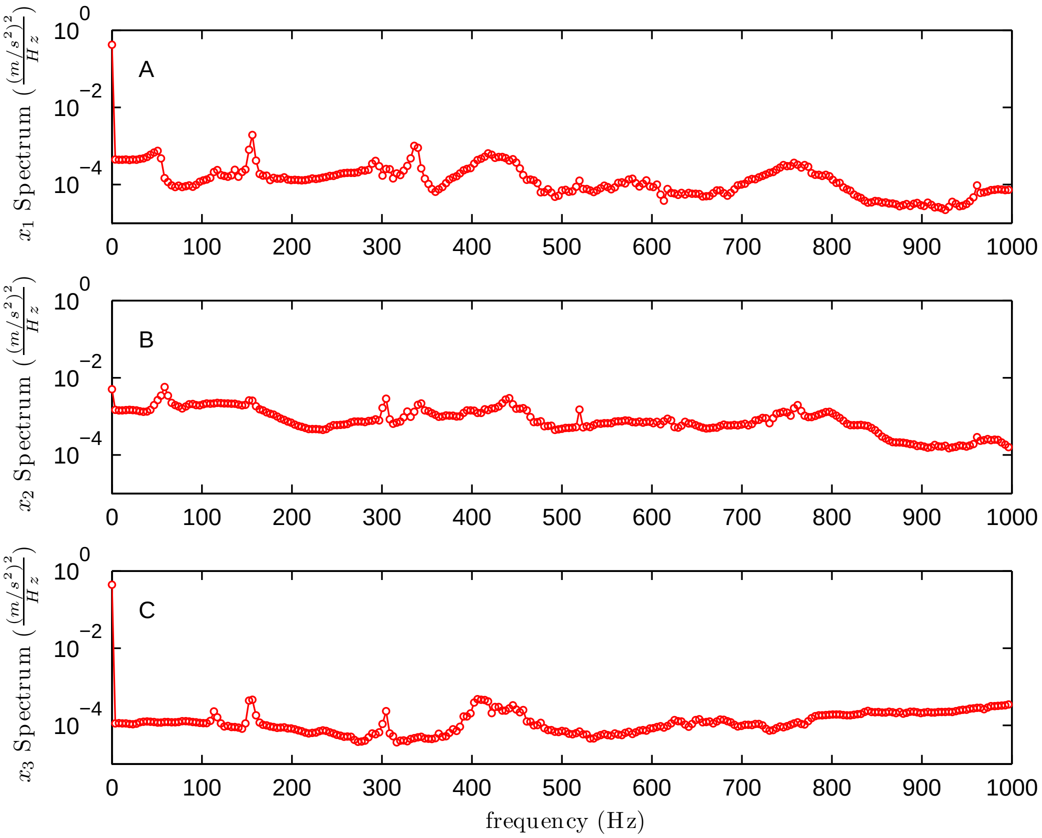


Figure 2.21. Frequency content of the Merlet platform in response to an impulse. As seen in Subplots A and B, the Cobotic Hand Controller's first resonant mode orthogonal to the x_3 axis is around 60 Hz. The following resonant modes are at 120-150 Hz (present in all directions) and 300 Hz. In Subplot C, the lack of the 60 Hz mode shows the extremely high stiffness along the axis of the Cobotic Hand Controller.

CHAPTER 3

Focus 2a: Hand Controller Control Architecture

3.1. Overview

The overall control scheme for the six-degree-of-freedom Cobotic Hand Controller is diagramed in Figure 3.1. This scheme consists of a six-degree-of-freedom dynamics simulation or *physics model* that computes desired accelerations in response to measured interaction forces with the human operator. The six-degree-of-freedom dynamics simulation is the subject of Chapter 8. The dynamics simulation contains an integrator and, therefore, is capable of feeding forward position, velocity and acceleration commands to the cobot, which executes motion control at the joint level. The cobot plant contains a single common power cylinder plant and six, rotational-to-linear plants, each comprised of a steering plant and a linear position sensor. The cobot renders motion to the human operator, and receives a force in response.

Notation for Figure 3.1 and the remainder of this chapter is summarized in Tables 3.1-3.5. The $'$ notation indicates a zero order held or sampled or measured signal.

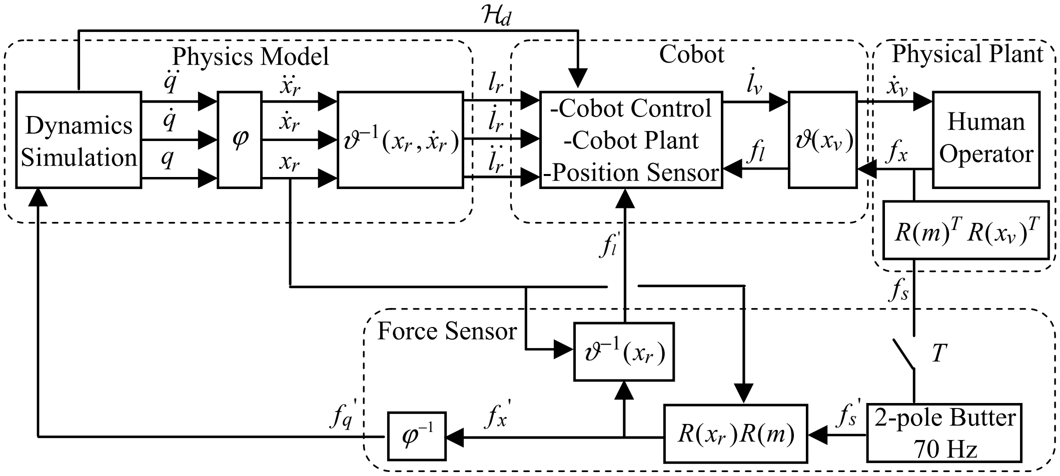


Figure 3.1. The overall control diagram including kinematics. φ is identity for our implementation since our virtual environment and task frames are aligned.

Table 3.1. Kinematics notation.

Variable	Definition
q, $\dot{q}$, $\ddot{q}$	virtual environment state
x_r, $\dot{x}_r$, $\ddot{x}_r$	reference (desired) task space state
x_v, $\dot{x}_v$	actual task space state
x_v'	measured task space position (via Newton-Raphson)
m	location of force sensor frame S in end-effector frame M
f_x, f_x'	actual and measured task frame force
f_s, f_s'	actual and measured sensor frame force
f_q'	measured virtual environment frame force
φ	kinematics from task to virtual environment
ϑ	kinematics from joint to task
R	3x3 rotation matrix

Table 3.2. Miscellaneous notation.

Variable	Definition
T, ZOH	sample rate interval and zero order hold
s, z	continuous and discrete time variables
k_p, k_i, k_d	proportional, integral and derivative gains
2-pole Butter	software Butterworth filter

Table 3.3. Joint notation.

Variable	Definition
m_l	joint mass
l_v, $\dot{l}_v$	actual joint state
l'_v, $\dot{l}'_v$	measured joint state
$\ddot{l}_d$	desired joint accelerations
f_w, f'_w	actual and estimated force on the wheel
f_l, f'_l	actual and measured joint frame force
l_r, $\dot{l}_r$, $\ddot{l}_r$	reference joint state

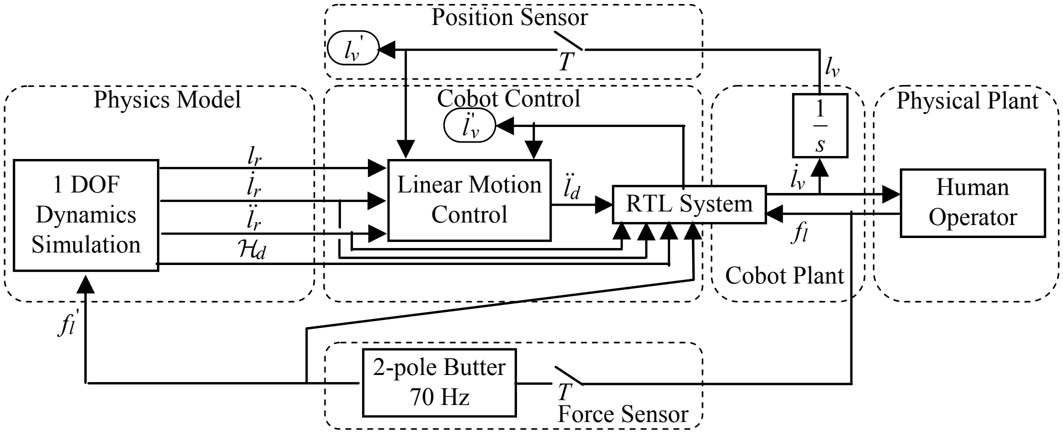


Figure 3.2. The overall control diagram without kinematics (for a single leg).

Table 3.4. Cylinder notation.

Variable	Definition
R	cylinder radius
ω, ω', ω_d	actual and measured and desired cylinder speed
J_c, $J_{m,c}$	inertia of cylinder and cylinder motor
τ_c	cylinder torque delivered to CVT
$\tau_{c,nominal}$	cylinder torque required to combat cylinder bearing friction
$\tau'_{m,c}$, $\tau_{m,c}$	commanded and actual torque applied by cylinder motor
$\tau_{c,ff}$	feedforward cylinder torque
$\mathcal{H}_d$	desired total energy of virtual environment

Table 3.5. Steering notation.

Variable	Definition
ϕ, ϕ'	actual and measured steering angle
$\dot{\phi}$, $\dot{\phi}'$, $\dot{\phi}_d$	actual, measured and desired steering velocity
J_b	inertia of steering bell
τ_b	torque applied to steering bell
$\tau'_{m,s}$, $\tau_{m,s}$	commanded and actual torque applied by steering motor
$\dot{W}_{shear}$	actual power dissipation due to shear at the contact patch
$\dot{W}_{bell}$	actual power dissipation due to friction in the bell bearing

We simplify Figure 3.1 by removing the kinematics so that only a single joint system is considered, as shown in Figure 3.2. Here we divide the cobot block of Figure 3.1 into *cobot control*, *cobot plant*, and *position sensor* blocks. The signals l'_v

and $\dot{l}'_v$ are terminated at squarish elliptical ports for future output to another diagram. Throughout the remainder of this chapter, we break down Figure 3.2 into increasingly smaller elements and analyze them in detail.

3.2. Joint motion control

The *linear motion control* block, enlarged in Figure 3.3, is a feedback controller to assure that the measured joint position l'_v tracks the output of the *physics model*, l_r. This consists of a proportional plus integral plus derivative (PID) feedback controller and a feedforward component. The output of this controller is a desired joint acceleration, $\ddot{l}_d$. Even in the full six-degree-of-freedom plant, feedback on motion control is still implemented at the joint level. This makes the tuning of gains simple by rendering the tuning of disparate rotational and translational feedback gains unnecessary. A disadvantage of feedback in joint space is that the dynamic response in task space is no longer linear, after the kinematics of the universal joints and distal links. While Gillespie et al. [66] seek generality by developing a cobot virtual-surface controller in task space, we have gone ahead and moved the feedback control into joint space for our device.

In Figure 3.4 we expand the *rotational-to-linear* (RTL) block that spans the *cobot control* and *cobot plant* blocks in Figure 3.2. The RTL system consists of elements termed *RTL control* and *RTL plant*. The controllers and plants of the steering and cylinder systems span the boundary between these elements. The cylinder and steering plants act on the *transmission plant*, which in turn outputs motion $\dot{l}_v$ in response

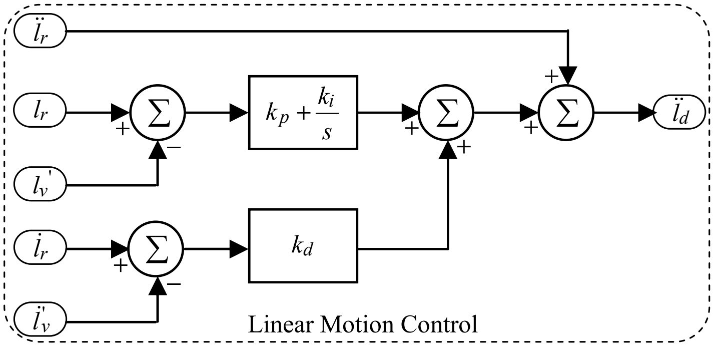


Figure 3.3. Motion control is placed at the level of the individual joints, and consists of a feedback component as well as a feedforward component.

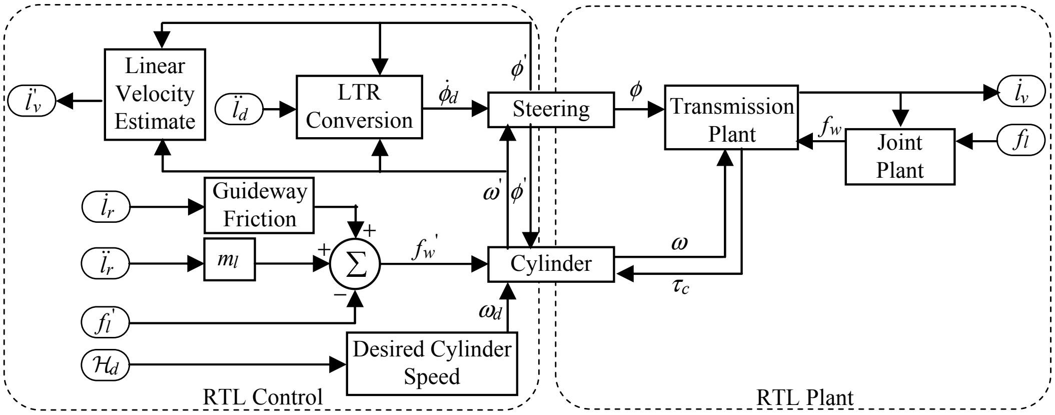


Figure 3.4. The rotational-to-linear system consists of the cylinder and steering plants, the transmission and joint plants and several controllers.

to the *joint plant's* mediation of force f_l into f_w.[1] The estimated linear output force of the wheel, f_w', is computed via a model of the joint plant and is used for feedforward control of the cylinder torque. f_w', is the sum of the reference model inertial force

[1] Chapter 6 develops a model of the transmission plant, and bond graphs are provided for both the transmission plant and joint plant in Chapter 7.

of the joint mass, $m_l\ddot{l}_r$, the reference model guideway friction force, $c_{d,l}P\text{sgn}(\dot{l}_r)$, and the measured force, f'_l, applied by the operator. The modelled and actual joints obey the following respectively (Equation 3.1).

$$\begin{aligned} f'_w &= m_l\ddot{l}_r + c_{d,l}P\text{sgn}(\dot{l}_r) - f'_l \\ f_w &= m_l\ddot{l}_v + c_{d,l}P\text{sgn}(\dot{l}_v) - f_l \end{aligned} \tag{3.1}$$

The *linear velocity estimate* block computes linear velocity $\dot{l}'_v$ utilizing current cylinder speed and steering angle rather than by differentiating linear position. The *desired cylinder speed* block computes a desired cylinder speed, ω_d, based on the desired total energy, $\mathcal{H}_d$, computed from the physics model. This block is described in greater detail in Chapter 4.

3.3. Linear-to-rotational conversion

The *linear-to-rotational (LTR) conversion* in Figure 3.4 is the method by which desired accelerations, $\ddot{l}_d$, are turned into desired steering velocities, $\dot{\phi}_d$. In Equation 3.2, we relate the total energy of the virtual environment, $\mathcal{H}$, to the cylinder surface speed, $R\omega$, via ratio k.

$$k = \frac{\mathcal{H}}{\omega R} \tag{3.2}$$

This ratio has units (kg)(m)/(sec), and solves the actuation redundancy of our six-degree-of-freedom device with seven actuators. This is elaborated on in much greater detail in Chapter 4. We require that individual legs abide by Equation 3.3.

$$\tan\phi_i = -\frac{\dot{l}_i}{R\omega} \tag{3.3}$$

Combining Equations 3.2 and 3.3 allows us to remove ω (Equation 3.4).

$$\tan\phi_i = -\frac{k\dot{l}_i}{\mathcal{H}} \tag{3.4}$$

Note that without Equation 3.2 for k, Equation 3.3 alone represents six equations in seven unknown velocities. Differentiating Equation 3.4 with respect to time yields six steering velocity equations (Equation 3.5).[2]

$$\dot{\phi}_i = \frac{\dot{\mathcal{H}}(k\dot{l}_i) - \mathcal{H}(k\ddot{l}_i + \dot{k}\dot{l}_i)}{\mathcal{H}^2 \sec^2\phi_i} \tag{3.5}$$

During actual implementation, the $\dot{\mathcal{H}}$ and $\dot{k}$ terms are neglected. This has the effect of providing some damping in the system by not providing a perfect feedforward model for steering of the wheels.[3] The joint space feedback compensates for any inaccuracies (albeit with delay) in the generation of steering commands due to the leaving out of terms.

[2] Note that the task to joint Jacobians and Hessians are absent from the present discussion, but are required since the desired joint accelerations and velocities, $\ddot{l}$ and $\dot{l}$, come from a physics simulation performed in virtual environment space ($\dot{q}$ and $\ddot{q}$), and must be mapped to first the task space ($\dot{x}$ and $\ddot{x}$) and then to joint space ($\dot{l}$ and $\ddot{l}$).

[3] Neither the rate of change of the desired ratio nor the desired acceleration of the power wheel are found in the constant ratio controllers of Moore [125] and Kim [99].

As an alternative to Equation 3.5 we differentiate Equation 3.3 and obtain Equation 3.6, having set $\dot{\omega} = 0$ rather than $\dot{\mathcal{H}} = 0$ and $\dot{k} = 0$.

$$\dot{\phi}_i = -\frac{\ddot{l}_i}{R\omega \sec^2 \phi_i} \tag{3.6}$$

3.4. Steering

The *steering system* is divided into *steering control* and *steering plant* elements as shown in Figure 3.5. Included in the plant model are the bell inertia, J_b, and the dissipation of the bell bearing, $\dot{W}_{bell}$. Also included is the dissipation at the contact patch, $\dot{W}_{shear}$. The transfer function $\tau_b(s)/\tau_{m,s}(s)$ describes the relation between steering motor torque generated by electrical current, and torque actually applied to the steering bell. This transfer function includes the inertia of the steering motor, and a gearing resonance in our implementation. This is described in greater detail in Appendix E, where an analysis of the performance of the steering controller is found, and in Chapter 7, where a bond graph of the steering plant is provided.

3.5. Linear slip heuristic

On the *steering control* side we have a proportional plus integral (PI) steering velocity controller. This controller utilizes an estimate of the steering velocity, $\dot{\phi}'$, obtained by finite-differentiation of the measured angle, ϕ'. The *linear slip heuristic* pulls the commanded torque, $\tau'_{m,s}$, to zero if the estimate of the steering velocity, $\dot{\phi}'$, exceeds the steering velocity, $\dot{\phi}_{max}(\phi)$, above which slip of the wheel will result due

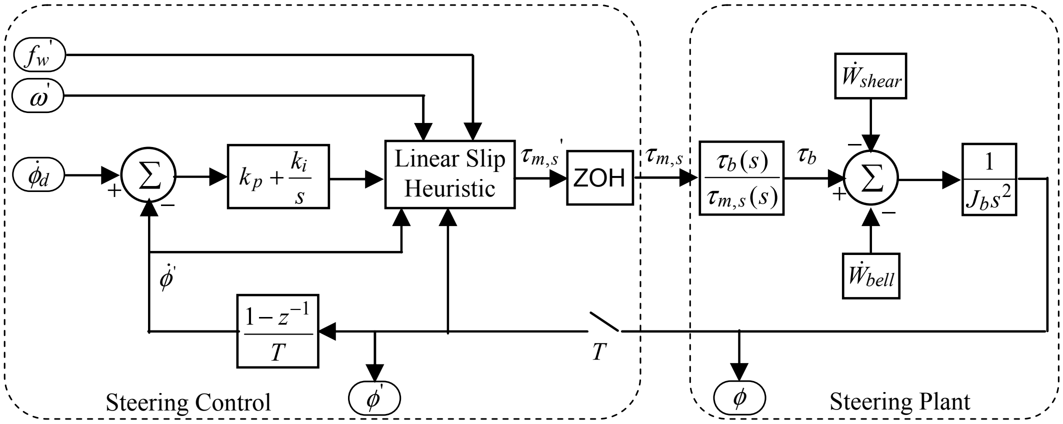


Figure 3.5. Steering system.

to an attempt to accelerate or decelerate the joint mass too quickly (Equation 3.7).

$$\dot{\phi}_{max}(\phi) = -\frac{\varepsilon\mu P}{m_l R\omega \sec^3(\phi)} + \frac{c_{d,l} P \operatorname{sgn}(\dot{l})}{m_l R\omega \sec^2(\phi)} - \frac{f_l}{m_l R\omega \sec^2(\phi)} \tag{3.7}$$

$\dot{\phi}$ is nominally computed from a desired acceleration, $\ddot{l}$ (Equation 3.6), but can be related to the forces on the wheel in order to determine $\dot{\phi}_{max}(\phi)$. For a known carriage mass, m_l, we need to determine the maximum acceleration that the wheel can sustain, $\ddot{l}$, given the available forces from the user, f_l, and from the wheel, f_w. We solve Equation 3.1 for $\ddot{l}$ and enter the result into Equation 3.6. f_w is replaced with $\varepsilon\frac{\mu P}{\sec(\phi)}$, where ε is some fraction of the friction force such that we do not approach gross slip. Dividing by $\sec(\phi)$ converts between the lateral force on the wheel and the output force of the wheel, f_w, in the direction of the joint.

This has been implemented with $\varepsilon = 0.5$ and is successful at preventing slip-based instabilities. The guideway friction force ($c_{d,l}P\text{sgn}(\dot{l})$) and operator force ($f_l$) components are ignored in the implementation since these loads are small relative to the inertial loads. At ($\phi = 0, \omega = 5, \epsilon = 0.5$) the inertial term yields a maximum steering velocity of 48 rads/s. At ($\phi = \frac{\pi}{6}, \omega = 10, \epsilon = 0.5$) the inertial term yields a maximum steering velocity of 16 rads/s.

3.6. Cylinder

The *cylinder system* shown in Figure 3.6 is also part of the RTL system. The cylinder system consists of a *cylinder controller* and a *cylinder plant.* The cylinder and cylinder motor inertias, J_c and $J_{m,c}$, are acted on by the cylinder motor torque, $\tau_{m,c}$, less the nominal torque spent to combat bearing friction, $\tau_{c,nominal}$, and the torque passed on to the transmission, τ_c. Here we have left out any gear ratio between the cylinder and cylinder motor so as to match our implementation. In general, a gear ratio would be included as is done in the bond graph of a cylinder plant provided in Chapter 7.

The cylinder controller consists of feedback and feedforward components. The proportional plus integral (PI) feedback controller compensates for friction, $\tau_{c,nominal}$, and the cylinder and motor inertias, $J_c + J_{m,c}$. The feedforward term estimates the disturbances, τ_c, and can be computed from Equation 2.10, our model of the Cobotic Hand Controller's dynamics from the cylinder's perspective. In the current implementation we compute the feedforward cylinder torque, $\tau_{c,ff}$, as the sum of the

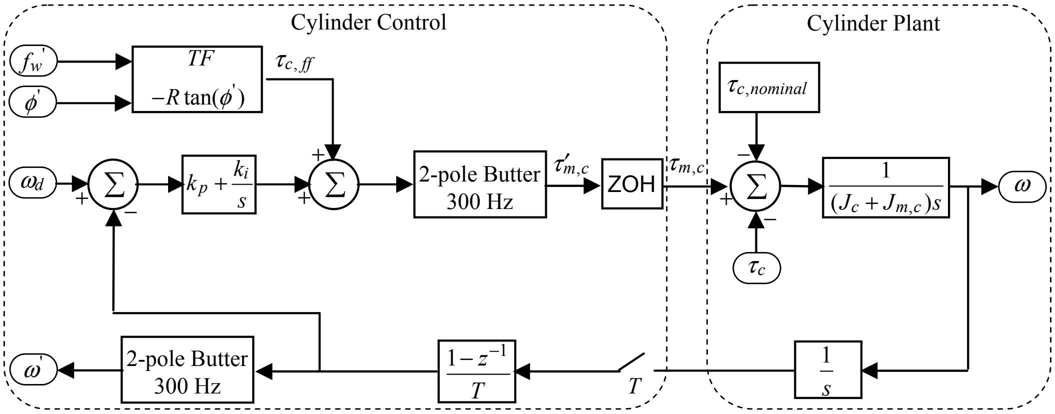


Figure 3.6. The cylinder system consists of a plant and controller.

wheel forces, f_w, that are modulated to the cylinder torque via the transmissions (Equation 3.8).

$$\tau_{c,ff} = -\sum_{i=1}^{6} f'_{w,i} R \tan(\phi_i) \tag{3.8}$$

This accounts for all portions of Equation 2.10, except for the cylinder inertia, which we leave out to provide some damping and to prevent unstable behavior. Also left out of the estimate $\tau_{c,ff}$ (and from Equation 2.10), are the losses in the CVT. However, at times when the PI feedback controller alone is not adequate, the dominant disturbance is from the joint inertias, for which we have provided feedforward compensation.

3.7. Redundant sensor information

In Figure 3.7, we tie together the l'_v, $\dot{l}'_v$ and q ports that did not lead anywhere else in this control framework. Although we chose to do our feedback control in joint

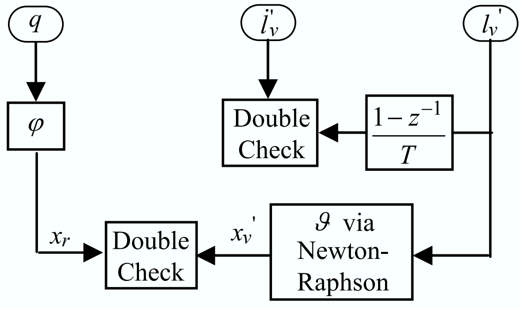


Figure 3.7. We have redundant methods to realize position and velocity.

space, we might want to check on our performance in task space. We can compare the reference position, x_r, to the actual measured position, x_v', as computed via the Newton-Raphson routine. We can also evaluate the validity of using cylinder speed and steering angle to compute linear speed, $\dot{l}_v'$ (Equation 2.7), by comparing it to the differentiated version of the linear potentiometer signal, $\left(\frac{1-z^{-1}}{T}\right) l_v'$.

Depending on the steering angle of the transmission, the joint velocity resolution, as computed from the steering angle and cylinder speed, ranges from far superior to approximately equivalent to the velocity signal obtained from just the linear potentiometer. Thus, a high-resolution joint position sensor may not be necessary in order for cobots to accurately determine their velocity. However, imperfections (elastic lateral creeps) in the transmission cause a slight phase lag of the steering angle/cylinder speed estimate of joint velocity relative to the linear potentiometer version, but typically less than that induced by filtering the linear potentiometer signal.

The joint velocity resolution for the potentiometer determined velocity is $\delta\dot{l} = 0.5$ mm/s at 100 Hz. The joint velocity resolution of the CVT determined velocity is $\delta\dot{l} = R\omega\tan(\delta\phi) \approx R\omega(\delta\phi) = 0.05$ mm/s. This is evaluated at cylinder speed, $\omega = 5$ rads/s, cylinder radius, $R = 0.0682$ m, and steering encoder resolution, $\delta\phi$, the resolution of a 40,960 count encoder in radians. Resolution of the cylinder speed, $\delta\omega$, also affects the joint velocity resolution,[4] although its contribution is dependent on the steering angle, varying from as much as $\delta\dot{l} = R\delta\omega\tan(\phi) = 0.3$ mm/s at $\phi = 60$ degrees, down to as little as $\delta\dot{l} = R\delta\omega\tan(\phi) = 0.0$ mm/s at $\phi = 0$ degrees. These are evaluated for $\delta\omega = 0.0025$ rads/s, the resolution of a 144,000 count encoder at 300 Hz.

[4] Although with a more heavily geared cylinder and thus better control of ω, the cylinder would introduce little or no error regardless of angle.

CHAPTER 4

Focus 2b: Redundancy Resolution

4.1. Background

The Cobotic Hand Controller utilizes a parallel architecture that contains a control redundancy. It has more actuators than task-space degrees of freedom. When the operator for a passive parallel cobot or the power motor driving the common element for an active parallel cobot is included, there are $n+1$ actuators for a n-degree-of-freedom task space (Figure 1.1), resulting in a control redundancy. It is arbitrary what speed the common element should be operating at. Moore [125] and Kim [99] have addressed this control redundancy via different methods with the powered and unpowered versions of the Arm Cobot. Moore implements a controller that keeps the common element speed at some ratio (*constant ratio*) to the Euclidean norm of task space velocity. Kim implements a controller that fixes the speed of the common element (*constant velocity*). Santos-Munné also describes utilizing the control redundancy of the passive Extreme Joystick Cobot to control the apparent inertia along the free motion direction (*constant inertia*), similar to the method we discussed in Section 2.3.2.1 [151]. Here we propose a controller that varies the common element speed with respect to the total energy of the virtual environment (*energy tracking*),[1] but never allows the common element speed to fall below a minimum threshold.

4.2. Motivation

We desired to operate the Cobotic Hand Controller with the cylinder spinning at some nominal minimum speed, which accelerates as the end-effector moves faster. There are several reasons for spinning at a nominal speed and, thus, preventing the cylinder from ever coming to rest. A resting cylinder is a control singularity where the current direction of the end-effector's single motion freedom and its speed along that single motion freedom are no longer defined by the geometry of the CVTs and cylinder. Also, near-zero speeds involve stiction and gross nonlinearities in rolling friction. These constraints preclude reversing the cylinder direction since a reversal would necessitate a zero speed condition.

It is also necessary to operate the cylinder in such a manner as to avoid large transmission ratios. We could simply run a fixed velocity controller with a high velocity, but this would result in increased wear of components.[2] In addition, high rolling velocity would cause significant mechanical vibrations and noise, even in the tightest tolerance system. A control scheme that will allow the cylinder speed to vary, but never approach zero, is necessary.

[1] Originally we tried using the joint kinetic energy (actual energy - not virtual), but this did not work well near kinematic singularities of the Merlet platform and neglected the potential energy stored in the springs in the virtual environment. We could have incorporated the virtual joint potential energy due to the springs, but we still would have suffered from the singularity issues.

[2] Although steering wear would decrease with higher common element velocity, rolling wear would increase.

We also took into consideration the requirements of a system that has a topology other than $\mathbb{R}^3$ or $SE(2)$. As described in Chapter 1, Moore fixed the ratio between the Euclidean norm of end-effector velocity and power wheel speed [124]. Kim fixed the ratio between Euclidean norm of drive-roller velocity and the power wheel speed [99]. Moore's method does not scale to six-degrees of freedom as there is no Euclidean velocity metric for $SE(3)$. Kim's method, when scaled to a six-degree-of freedom mechanism, falters since the motion of the proximal joints (equivalent to his choice of drive-roller speeds) may not affect the Euclidean norm in a desirable manner due to the Merlet platform kinematics.

4.3. Other considerations

Other criteria considered in the development of this *energy tracking* controller but not necessarily directly addressed, were the kinetic energy exchange from joints to power cylinder, how steering angle and cylinder speed affect acceleration/deceleration capacity, electrical power requirements, achievable bandwidth in tangent and normal directions of motion, and haptic transparency at times of free motion, constrained motion and unilateral impacts. The Cobotic Hand Controller should dissipate joint kinetic energy not with actuator torques, but by the speed at which it can transfer kinetic energy into the power cylinder from the proximal and distal links by CVT steering. In this way, the power cylinder, or *momentum sink*, provides a non-electrical means of absorbing a large amount of energy from the user and joints during virtual wall impacts.[3]

In order to accelerate and decelerate maximally, we need both adequate cylinder speed (more is better) and small enough angles to assure that the lateral friction force of the wheel is aligned along the joint in order to yield maximum acceleration ability for the joint. Thus, we need to increase the speed of the cylinder as we approach a virtual environment unilateral impact in order to have small angles and adequate cylinder speed for the required accelerations of the virtual environment. Also, in virtual environment free space, adequate nominal cylinder speed is required to allow for voluntary operator accelerations of the virtual environment mass.

We also considered that in general, if we increased the cylinder speed prior to impact we could significantly reduce constraint penetration since the CVTs would have less excursion distance required to completely turn off any flow. However, as the CVTs are driven by a control system with linear dynamics, the rise time of the steering angles (from the instant a unilateral constraint is arrived at) is the same regardless of the distance they have to turn.

For smooth static surfaces, it may be desired to operate with the cylinder moving slowly relative to end-effector speed, or even rotating at a constant velocity, and in doing so utilize a wide range of steering angles available to the CVTs. When rendering

[3] However, counter to the hypothesis that kinetic energy transfers from the joints to cylinder at impact, we find that the cylinder tends to slow down slightly during the deceleration phase of a virtual environment impact (see Figure 5.3).

rough surfaces, it may be desirable to spin the cylinder faster in order to elicit larger joint motions with less CVT steering.

From an electrical power efficiency standpoint, it would be advantageous to minimize frictional dissipation in the system when accelerating, by exchanging stored energy from the cylinder to the linear joints, and to preserve energy when decelerating, by transferring it from the joints to the cylinder. Any work done by the operator should also be captured and stored as an increase in cylinder energy. If a ratio type controller is chosen the cylinder could be allowed to coast if its speed is ever higher than desired, and friction could be used rather than electrical power in order to slow the cylinder.[4]

One can also consider increasing frictional dissipation in order to meet stability/passivity requirements or to aid in deceleration. Maximizing or minimizing existing frictional dissipation requires optimization of the choice of steering angles and cylinder speed.

4.4. Total energy function

We propose a method where a virtual environment energy function is used to determine a desired cylinder speed.[5] This method will work well with the $SE(3)$ or $\mathbb{R}^6$ topology of the Cobotic Hand Controller. Since the method involves the virtual

[4] Many variants of this have been tested and all lead to some undesirable oscillations as the cylinder controller turns on and off or strengthens and weakens.

[5] Relating the cylinder's energy (rather than speed) to the joint-space energy, would render the direction of the cylinder ambiguous. However, if you assume that the absolute value of cylinder speed can never drop below a threshold you can avoid this issue.

environment, it will provide a stable and objective method of determining the desired cylinder speed, independent of the specific haptic display kinematics. While one could relate the cylinder speed to the kinetic energy of the joints, the velocities of the joints may change rather abruptly due the to scaling and sensitivity of the Jacobian. Since each joint velocity would be squared in order to compute kinetic energy, we could ignore their contribution to the cylinder speed at low joint speeds but not at higher speeds. Near joint singularities this would cause an undesirable acceleration of the cylinder.

Our total energy, $\mathcal{H}$, as shown in Equation 4.1, is the sum of the virtual environment kinetic energy and any potential energy stored in virtual environment springs.

$$\mathcal{H} = \frac{1}{2}\left(\dot{q}^T M(q)\dot{q} + k_t\left(p_o - [q_1, q_2, q_3]^T\right)^T\left(p_o - [q_1, q_2, q_3]^T\right) + k_r\theta^2\right) \tag{4.1}$$

Translational and rotational spring constants, k_t and k_r, along with translational spring origin, p_o, and axis-angle deviation, θ, are defined in Section 8.4. q and $\dot{q}$ indicate that the desired virtual environment state, and not the measured state are used. Table 4.1 summarizes notation introduced in this chapter.

The relationship between cylinder speed and this total energy function can be defined such that higher cylinder speeds (and therefore small steering angles) are required as we move faster in free-space. Upon hitting a virtual environment constraint, the virtual kinetic energy of $M(q)$ will be transferred to virtual potential energy of the springs k_t and k_r, and, the total energy will not change abruptly. Smaller steering angles due to high cylinder speeds will allow for the greatest deceleration/acceleration

Table 4.1. Redundancy algorithm notation.

Variable	Definition
$\mathcal{H}$	virtual environment total energy (desired mass but measured position and velocity)
$\mathcal{H}_d$	desired virtual environment total energy (desired mass, position and velocity)
$\dot{\mathcal{H}}_d$	desired rate of change of virtual environment total energy
ω_d, $\dot{\omega}_d$	desired cylinder velocity and acceleration
k, k_d	actual and desired ratio between $\mathcal{H}$ and $R\omega$
$\dot{k}_d$	desired rate of change of ratio k
$\Phi(k)$	relationship between k and ω
f_{cyl}	bandwidth of the cylinder velocity control in hertz
r_c	characteristic length scale (composite radius of proximal and distal universal-joints)

capability during the impact event. Damping inside the virtual constraint will gradually reduce the total energy and, therefore, the desired cylinder speed. The rate of change of the total energy, $\dot{\mathcal{H}}$, involves desired accelerations, $\ddot{q}$, and is required for computing steering velocities (recall Equation 3.5).

4.5. Desired ratio/cylinder speed trajectory

We now need to determine if $k = \frac{\mathcal{H}}{R\omega}$ (Equation 3.2) is to be a constant or if it is to be a function of the virtual environment total energy, $\mathcal{H}$. Since we want to prevent the cylinder from coming to rest while still allowing changes in cylinder speed, we specify a relationship where cylinder surface speed, $R\omega$, and desired ratio k are both dependent on $\mathcal{H}$. Thus for a given energy level $\mathcal{H}$, we desire both the cylinder surface speed $R\omega$ and ratio k to adjust. For this purpose, we define a space with abscissa k

and ordinate $R\omega$ (Figure 4.1). Iso-contours of energy in this space are $\mathcal{H} = R\omega k$. $R\omega$ and k are dependent upon $\mathcal{H} = R\omega k$ via a trajectory Φ as given by Equation 4.2.

$$R\omega = \Phi(k) \tag{4.2}$$

$\Phi(k)$ contains a minimum value for $R\omega$ and is the path we proceed along across the level curves of $\mathcal{H}$. The suggested minimum value for $R\omega$ is $\sigma = \frac{\mu P}{|M(q)|}\left(\frac{1}{f_{cyl}}\right)$ where f_{cyl} is the bandwidth of the cylinder speed controller in Hertz and $M(q)$ the inertia matrix of the simulated rigid body or mechanism in the virtual environment. $|M(q)|$ for a rigid body needs to incorporate both translational mass, m_t, and rotational inertia, J_r, via a metric such as $\frac{1}{2}\left(m_t + \frac{\text{trace}(J_r)}{{r_c}^2}\right)$, where r_c is a characteristic length scale (e.g., the radius of the end-effector platform or the radius of the proximal universal joints from the cylinder axis).

The need for a minimum speed arises from the need to anticipate rapid accelerations caused by human intent and the need to keep the cylinder away from $\omega = 0$. Thus, for a given virtual environment mass and preload force availability, we need a certain level of cylinder speed to keep the required steering velocity (and steering excursion) within reason.

When we have a virtual environment model running, the vectors q, $\dot{q}$ and $\ddot{q}$ will be computed. From these vectors we can evaluate the desired $\mathcal{H}_d$ and $\dot{\mathcal{H}}_d$. Subsequently, we can solve for the desired k (Equation 4.3).

$$\mathcal{H} = kR\omega = k\Phi(k) \tag{4.3}$$

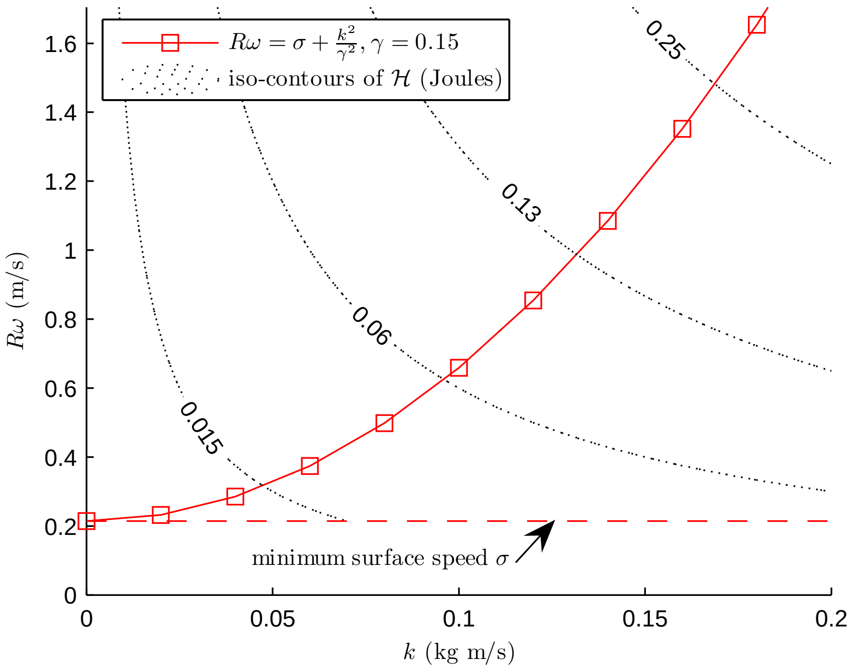


Figure 4.1. A desired trajectory relating parameters k and $R\omega$. We tend to find the best performance when the trajectory heads orthogonal to the iso-contours of total energy, $\mathcal{H} = R\omega k$, and curves upward to avoid requiring high transmission ratios. The orthogonal trajectory avoids requiring either the cylinder or transmissions to adjust too rapidly as $\mathcal{H}$ varies.

The functions Φ may be difficult to solve for k given $\mathcal{H}$, but a simple binary search can efficiently find the solution of a monotonically increasing function. k can then be substituted back in, yielding the desired cylinder speed, ω_d, and desired ratio, k_d.

We can also differentiate for the rate of change of the total energy with respect to time (Equation 4.4).

$$\dot{\mathcal{H}} = k\frac{\partial(R\omega)}{\partial t} + R\omega\frac{\partial k}{\partial t} = k\frac{\partial\Phi(k)}{\partial k}\dot{k} + R\omega\dot{k} \tag{4.4}$$

Equation 4.4 can be solved for $\dot{k}_d$ (Equation 4.5).

$$\dot{k} = \frac{\dot{\mathcal{H}}}{k\frac{\partial\Phi(k)}{\partial k} + R\omega} \tag{4.5}$$

$\dot{k}_d$ can then be used to solve for cylinder acceleration, $\dot{\omega}_d$.

Now we can evaluate the steering velocities using Equation 3.5 in terms of k_d, $\dot{k}_d$, $\mathcal{H}_d$ and $\dot{\mathcal{H}}_d$. As we mentioned earlier, Equation 3.5 is simplified to Equation 3.6 by dropping the excitatory $\dot{\mathcal{H}}_d$ and $\dot{k}_d$ terms (they tend to make the model of the system a bit to perfect and therefor “alive”). We also neglect to use $\dot{\omega}_d$ for the cylinder computed-torque feedforward controller, thereby providing a bit of damping. The cylinder controller does utilize ω_d for the set-point of the cylinder velocity feedback controller.

In Figure 4.1 we plot a proposed trajectory, Φ, in the space of cylinder surface speed $R\omega$ and ratio k. σ indicates the minimum cylinder surface speed and a scale parameter, γ, is utilized. The iso-contour curves of $\mathcal{H}$ are also depicted.

4.6. Free motion performance experiment

4.6.1. Purpose

To analyze the performance of the proposed energy-tracking cylinder speed controller during manipulation with an unimpeded mass-damper virtual environment.

4.6.2. Methods

We manipulate a 2.0 kg mass in a 0.1 (N)/(m/s) damping environment. All three translations are allowed and rotation is disallowed.

4.6.3. Results

In Figures 4.2 and 4.3 we record the energy-tracking controller's performance during manipulation of this mass-damper system. In Subplot A of Figure 4.3, the system was accelerated from a state of rest to 0.04 Joules, which is 0.2 m/s for a 2.0 kg mass, and then decelerated back to rest. In Subplot B the cylinder speed increases and subsequently decreases in surface speed, from the nominal minimum of $\frac{\mu P}{|M(q)|}\left(\frac{1}{f_{cyl}}\right) \approx$ 0.214 m/s to a maximum of 0.478 m/s.

4.6.4. Discussion

Although the actual cylinder speed lags the desired cylinder speed, the actual energy (computed from measured joint speeds) leads desired energy due to the slight lead nature (gain greater than unity) of our joint motion controller (a Bode plot of this

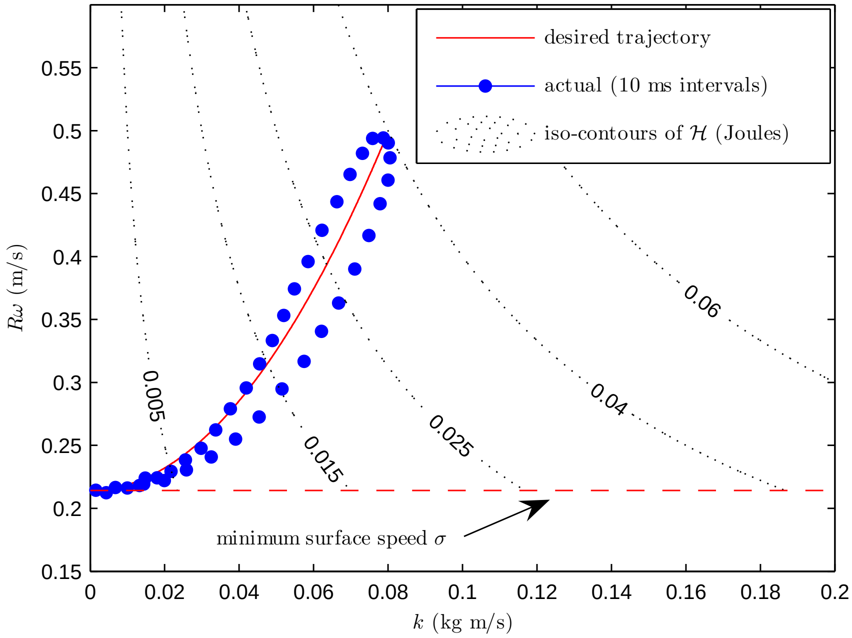


Figure 4.2. Desired and actual cylinder surface speed and ratio during interaction with a mass-damper virtual environment.

controller is provided in Figure 5.4). Recall that while we use potentiometer derived joint velocity ($\frac{1-z^{-1}}{T}l_v'$) to calculate the actual total energy for plotting purposes in Figures 4.2 and 4.3, the joint velocity computed from steering angle and cylinder speed ($R\omega' \tan\phi'$) is used to perform joint motion control. This measure of joint velocity leads the potentiometer derived velocity due to elastic creep in the transmission. In Figure 4.2 we can see that the lag of the cylinder control and lead of the linear motion control cause our motion to depart from the desired trajectory at non-zero energies. While the actual trajectory may depart significantly from the desired trajectory, the time scale during which it deviates is very small.

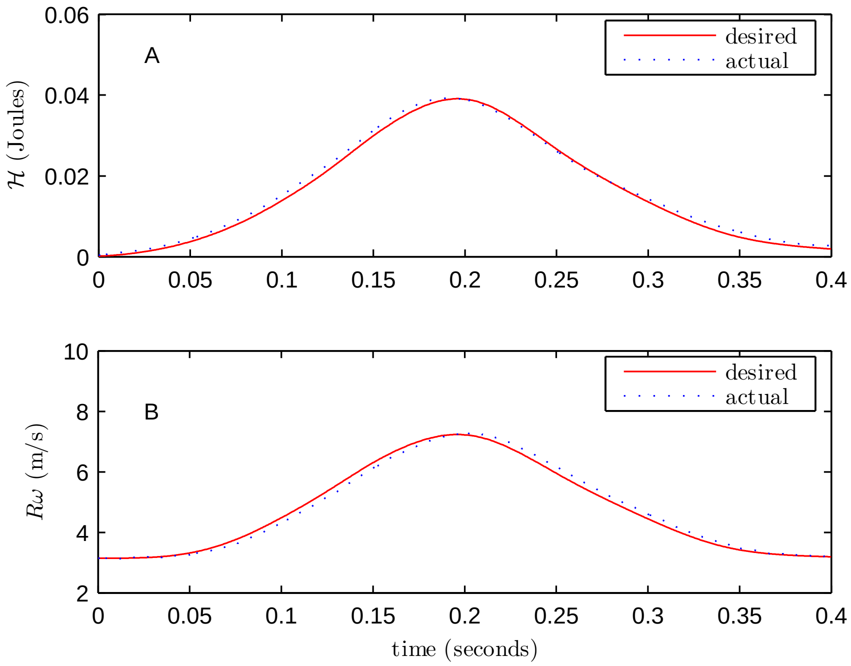


Figure 4.3. A. Desired and actual total energy of mass-damper virtual environment. The desired total energy is computed from the desired inertia, positions and velocities. The actual total energy is computed from the desired inertia but the actual (measured) positions and velocities. The lead nature of our joint motion controller causes the lead of the actual total energy relative to the desired total energy. B. The desired and actual cylinder speeds.

4.7. Unilateral impact performance experiment

4.7.1. Purpose

To analyze the performance of the proposed energy-tracking cylinder speed controller during manipulation in a virtual environment containing unilateral constraints.

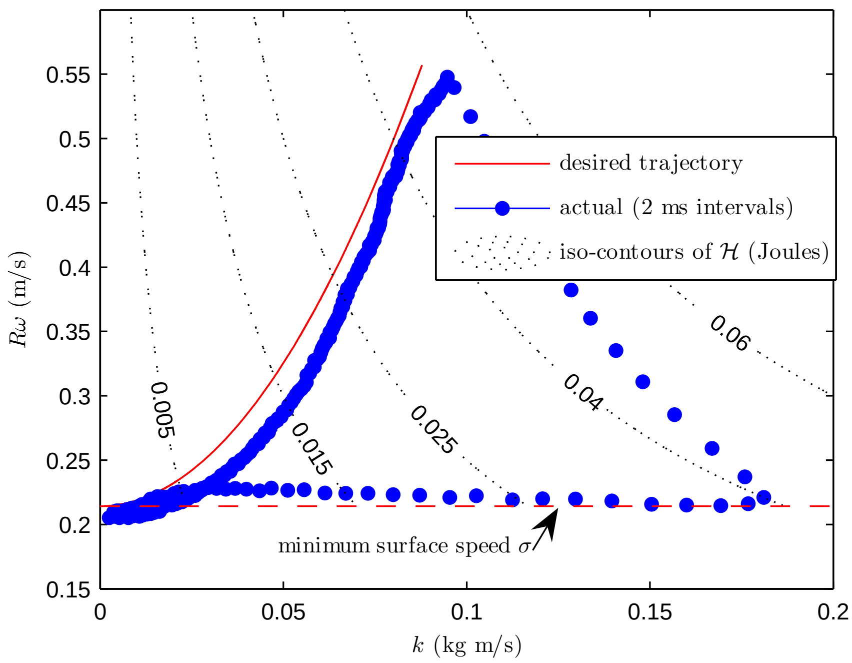


Figure 4.4. Desired and actual cylinder surface speed and ratio during a unilateral collision with a dissipative wall.

4.7.2. Methods

We again manipulate a 2.0 kg mass in a 0.1 (N)/(m/s) damping environment while disallowing rotations. The environment is bounded by unilateral constraints characterized by 10,000 N/m, 400 (N)/(m/s) walls.

4.7.3. Results

In Figures 4.4 and 4.5 we record the energy-tracking controller's performance during an impact with these dissipative unilateral constraints.

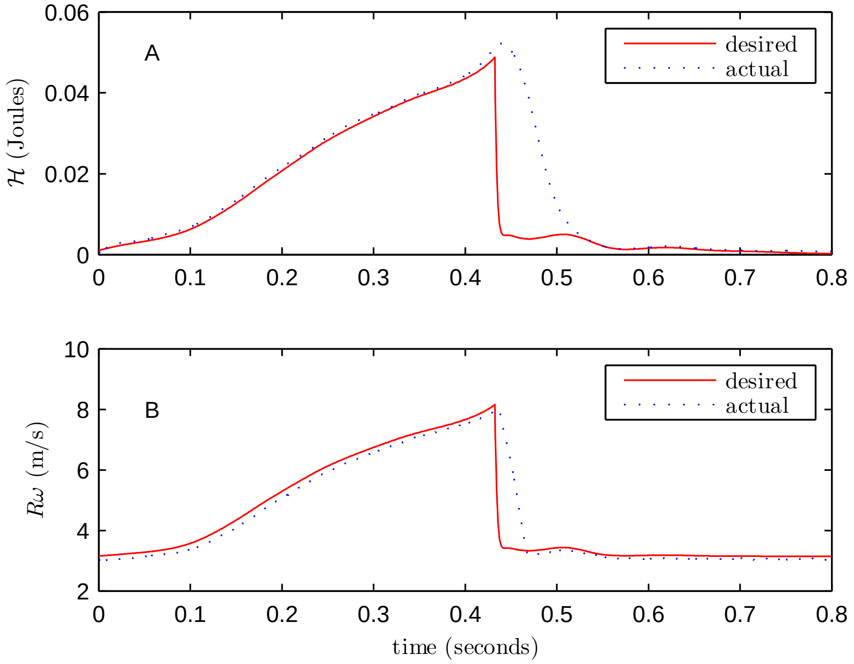


Figure 4.5. A. Desired and actual (measured) total energy of virtual environment during a unilateral collision with a dissipative wall. The desired total energy is computed from the desired inertia, positions and velocities. The actual total energy is computed from the desired inertia but the actual (measured) positions and velocities. B. Desired and actual cylinder surface speed.

4.7.4. Discussion

The desired trajectory in Figure 4.4 is followed relatively well until the constraint is encountered. When the constraint is encountered, the cylinder is able to reduce speed much quicker than the joints, which require time for the wheels to steer (thus the high actual (measured) energy as the joints are still in motion). Even as the wheels steer, creep and slip occur since the wheels incur heavy loads decelerating the joint masses and resisting the operator applied forces. Again, note the timescale for which the actual trajectory deviates from the desired trajectory. Most of this departure is due to the fact that the cobot system is not capable of decelerating as quickly as is called for by the virtual environment scenario. This limitation is discussed in greater detail in Section 5.2, where the deceleration capability of the energy-tracking controller is compared to that of a constant cylinder speed controller.

4.8. Conclusion

We have implemented a controller that addresses several cobotic architecture concerns by appropriately utilizing the inherent control redundancy of a parallel cobot. The energy-tracking controller maintains a slow nominal cylinder speed that prevents excessive vibration and wear, but is still adequate to provide the necessary joint accelerations given the relative magnitudes of voluntary operator forces and a virtual environment mass. We will also show that the controller performs just as well as a fixed-speed controller of higher nominal cylinder velocity (Section 5.2). The controller succeeds in achieving high cylinder speeds at times of impact with unilateral constraints when small steering angles are needed to sustain the forces required for high joint accelerations. The high cylinder speed and resulting small steering angles reduce the lateral loads on the wheel, decreasing the probability of slip.

CHAPTER 5

Focus 2c: Performance

In this chapter we quantify the motion control capabilities of the Cobotic Hand Controller as an admittance display. We demonstrate its ability to faithfully render a desired admittance, the bandwidth of its motion control, how different control algorithms affect faithful rendering of programmed dynamics during unilateral impacts, and the virtual environment impedance ranges it can stably render while interacting with various real impedances.

5.1. Nominal admittance range experiment

5.1.1. Purpose

As described in Chapter 3, the Cobotic Hand Controller conveys a virtual environment by establishing and rendering an admittance, or a relationship between measured force and executed motion. In this experiment, we seek to establish the ability of the hardware and software to render a desired admittance, $\frac{\dot{x}(s)}{f(s)}$, across a range of frequencies.

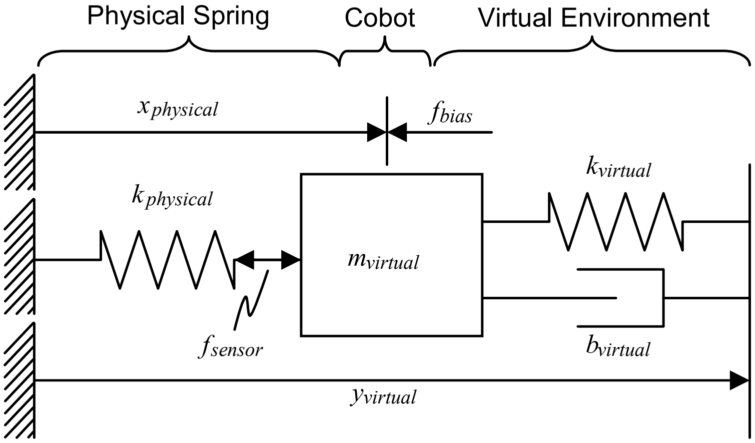


Figure 5.1. Schematic of admittance display system. The virtual environment gives the desired impedance, or interaction behavior of the cobot with the physical environment.

5.1.2. Methods

Figure 5.1 depicts the method by which the data is obtained for this experiment. The virtual environment of the system consists of a virtual spring, $k_{virtual}$, a virtual damper, $b_{virtual}$, and a virtual mass, $m_{virtual}$. A virtual bias force, f_{bias}, drives the virtual system and the actual cobot a distance, $x_{physical}$, to compress the real spring, $k_{physical}$, thereby generating load, f_{sensor}, at the load cell. The simulation is oriented along the axis of the cylinder, rendering the motion of all six joints identical.

For the experiment, the desired system properties are $k_{virtual} = 10,000$ N/m, $b_{virtual} = 200$ Nm/s and $m_{virtual} = 2$ kg (these are common values for the implementation of unilateral constraints with the display, and we know them to be stable for interaction with $k_{physical} = 1000$ N/m). The origin of $k_{virtual}$, $y_{virtual}$, is placed such that f_{sensor} read 5 N when f_{bias} is 0 N. This is achieved by driving $x_{physical}$ at 5 cm/s towards $k_{physical}$ until f_{sensor} reads 5 N. Then f_{bias} is varied sinusoidally with mean 5 N and amplitude of 1.5 N at 33 frequencies spaced logarithmically between 1 and 100 Hz. Data is recorded for 0.25 seconds (500 samples) at each frequency, and a time-domain analysis obtains the amplitude and phase relationship between $\dot{x}_{physical}$ and f_{sensor}. The analysis uses a Matlab algorithm *sinefit.m* developed by Dou and Chen that performs two-stage linear least squares fitting to extract amplitude and phase between two signals [48].

5.1.3. Results

Figure 5.2 depicts $\frac{\dot{x}_{physical}(s)}{f_{sensor}(s)}$ as desired and as determined experimentally.

5.1.4. Discussion

We see that the cobot is faithful in its rendering of the desired admittance up until around 50 Hz, where it begins to gain in magnitude and lose in phase. Although structural resonances and the slight lead nature of our joint motion controller may account for some of this discrepancy, the desired joint velocity of the experiment results in a joint amplitude of less than 1 linear position sensor increment for frequencies greater than 60 Hz. In other words, the amplitude of f_{bias} is inadequate to move $m_{virtual}$ a measurable distance at high frequencies. In order to compensate we could increase the amplitude of f_{bias}, however the steering actuator torque saturates.

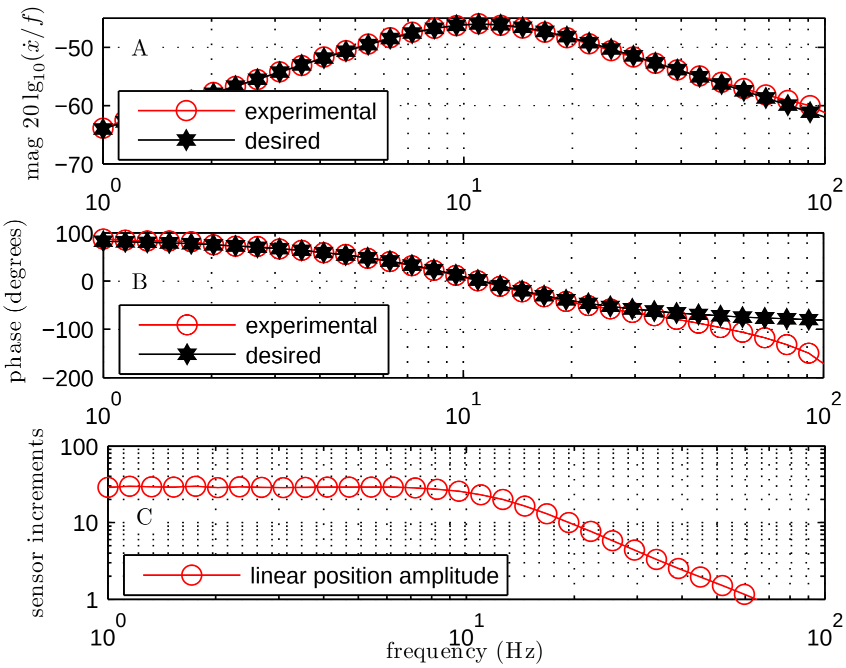


Figure 5.2. Desired and experimentally rendered admittance, $\frac{\dot{x}_{physical}}{f_{sensor}}$, for a 10,000 N/m spring, a 200 Nm/s damper, and a 2 kg mass. This demonstrates the cobot's ability to render a desired relationship between force and motion while in contact with a 1000 N/m spring. A. Magnitude of $\frac{\dot{x}_{physical}}{f_{sensor}}$. B. Phase of $\frac{\dot{x}_{physical}}{f_{sensor}}$. C. Amplitude of joint motion in terms of sensor increments during the experiment.

5.2. Acceleration capability experiment

5.2.1. Purpose

While the nominal admittance range experiment provides a good benchmark of frequency based performance, we would like to establish performance in the event that high accelerations are requested regardless of frequency content. In this experiment we investigate the ability of the cobot to render unilateral constraints.

5.2.2. Methods

Consider a virtual environment mass of 2 kg moving toward a virtual wall characterized by stiffness and damping, 10,000 N/m and 400 Ns/m, respectively. The experiment is executed by the operator holding the end-effector loosely with all five fingers as he/she bangs it sequentially into the front and back of a virtual box. Attention is given to obtaining approximately the same pre-impact velocity of 0.25 m/s at each impact.

5.2.3. Results

In Figure 5.3, Subplots A and E, we show end-effector position during virtual environment unilateral impacts for both fixed-speed and energy-tracking cylinder controllers. In Subplots B and F, we focus in on one impact in order to see the actual penetration versus the desired penetration. Note that the desired trajectory penetrates the wall only about 1.0 mm while the actual penetration is 3-4 mm. In Subplots C and G the position error is reported. In Subplots D and H the cylinder speed is reported.

5.2.4. Discussion

Cylinder speed dips slightly during impacts when the fixed-speed controller is used. When the energy-tracking controller is used, cylinder speed rises with kinetic energy

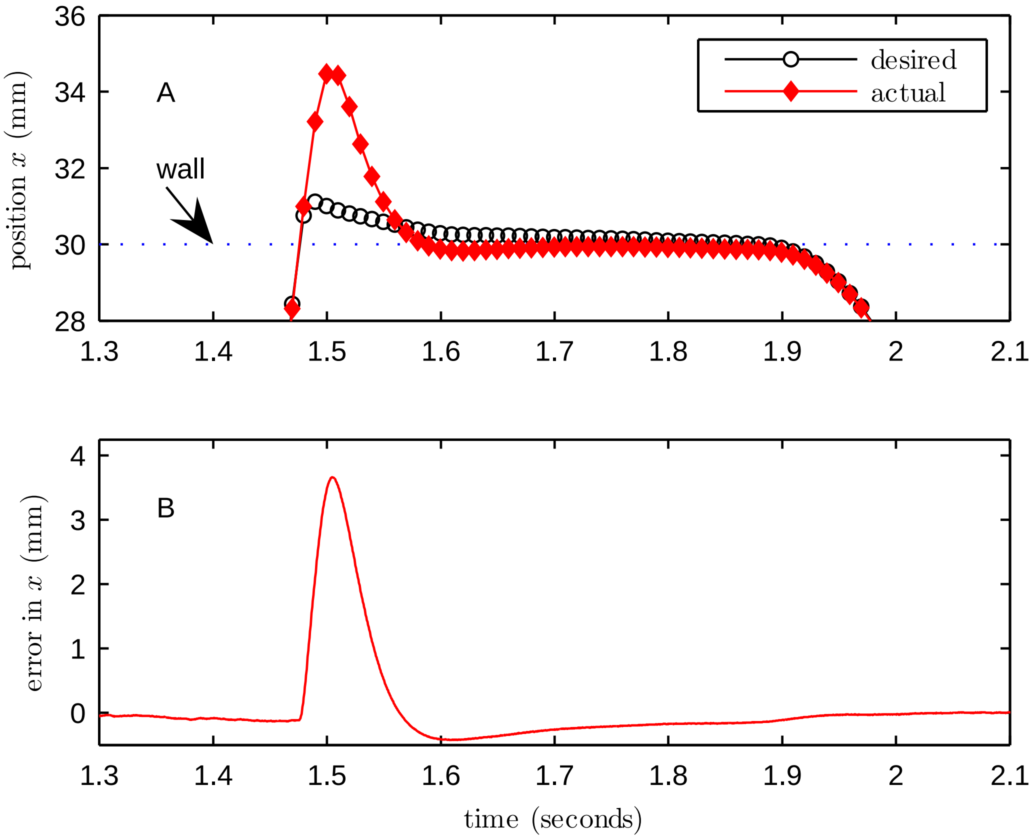


Figure 5.3. On the left is fixed cylinder speed and on the right is the variable cylinder speed algorithm. The user is moving a ball between walls located at ±30 mm.

as the mass is accelerated towards the wall and, during the impact, the cylinder speed reduces to the nominal minimum, without dipping below the minimum, as the virtual damper characterizing the wall gradually dissipates all the energy.

The walls are initially soft (the 3-4 mm penetration rather than the desired 1 mm penetration) but harden quickly as the user is pushed back to the wall, even when applying a significant load. The resulting haptic perception is of a soft wall that hardens quickly after impact. This contrasts with impedance displays that use impulses to create the perception of a hard initial contact, and are then unable to sustain high forces [148].

The simulation demands about 34 m/s^2 acceleration while only a 7.4 m/s^2 acceleration is delivered by the cobot. Since the cobot is capable of approximately 30 m/s^2 provided the wheels are at $\phi = 0$ (and allowed 15 m/s^2 for $\varepsilon = 0.5$ in the linear slip prevention heuristic of Equation 3.7), there is some margin for controller development. We would like to increase the gains in order to be able to achieve the maximum allowed acceleration of 15 m/s^2, however this leads to excitation of structural dynamics and to wheel slip. Nevertheless, achieving 7.4 m/s^2 out of a limit of 15 m/s^2, is perhaps as best as can be expected given the numerous controllers and plants involved.

Two common methods for extending the perceived Z-width of a display are event-based [101] and impulsive [122, 148] haptics. Impulsive haptics can not be used for this system, since we can only achieve a limited acceleration and applying an impulse to the cylinder or steering motors results in slip. This slip limits the achievable accelerations of the Cobotic Hand Controller by causing a positive feedback loop. As slip occurs, more steering is requested which only compounds the issue. Event-based feedforward routines might do well, ignoring both the resonance of structural dynamics and the nonlinear disaster of creep/slip of the rolling wheel, by simply playing back a steering trajectory independent of joint motion control.

5.3. Joint motion control experiment

5.3.1. Purpose

Here we examine the frequency response of the joint motion control, $\frac{\dot{l}_v(s)}{\dot{l}_r(s)}$, in order to better understand this subsystem of the admittance plant, $\frac{\dot{x}(s)}{f(s)}$. We also examine the impact[1] of our method of measuring $\dot{l}_v$ (via CVT steering angle and cylinder velocity) and our choice of cylinder speed controller on the frequency response of the joint motion control, $\frac{\dot{l}_v(s)}{\dot{l}_r(s)}$.

5.3.2. Methods

We now drive the system with a motion input rather than a force input. Data is taken without the end-effector in contact with anything since force information is not used. The six joint systems are driven via commanding a desired sinusoidal position, velocity and acceleration for 0.25 seconds (500 samples) at each of 33 frequencies spaced logarithmically from 1 to 100 Hz. The amplitudes are all for $\ddot{l}_r = 1.0$ m/s^2. The frequency response of all six joints are recorded and averaged.

This experiment is run four times, with three different methods of measuring $\dot{l}$ and two different cylinder speed controllers. $\dot{l}_v$ is measured via finite differentiation of the linear potentiometer ($\frac{1-z^{-1}}{T} l_v$), via CVT angle and desired cylinder velocity ($-R\omega_d \tan\phi$), and via CVT angle and actual cylinder velocity ($-R\omega \tan\phi$). Both fixed-speed and energy-tracking cylinder controllers are also utilized. The simulated

[1] $\dot{l}_v$ is required for the derivative term of the PID motion controller that compares $\dot{l}_v$ with $\dot{l}_r$.

virtual environment is a 2.0 kg mass to provide the energy-tracking controller a total energy from which to compute a desired speed.

5.3.3. Results

In Figure 5.4 we plot the closed-loop motion bandwidth of the joints. We plot a predicted response based on our model of the steering plant dynamics (Appendix E) and joint motion control gains. Along with this predicted response are four measured responses, each for different cylinder controllers and methods of utilizing sensor information. The most excitatory (gain greater than unity) response is when linear velocity is measured via $(\frac{1-z^{-1}}{T}l_v)$ (Condition 1). This excitation is also coupled with a loss of phase. Measuring the joint velocity via $(-R\omega_d \tan\phi)$ with a fixed-speed cylinder controller leads to much better response (Condition 2). Measuring the joint velocity with $(-R\omega \tan\phi)$ with a fixed-speed cylinder controller (Condition 3) leads to a slightly more attenuated response than in Condition 2. Finally, measuring the joint velocity with $(-R\omega \tan\phi)$ with a energy-tracking cylinder controller (Condition 4) leads to a response that is marginally more attenuated than that of Condition 3.

5.3.4. Discussion

Ideally (if the actual transmission exactly obeys $\dot{l} = -R\omega\tan(\phi)$), the open-loop frequency response of joint-motion (the LTR conversion and steering plant without the linear motion controller - see Section 3.2 for this terminology) would exactly match the frequency response of the CVT steering velocity controllers as shown in Figure

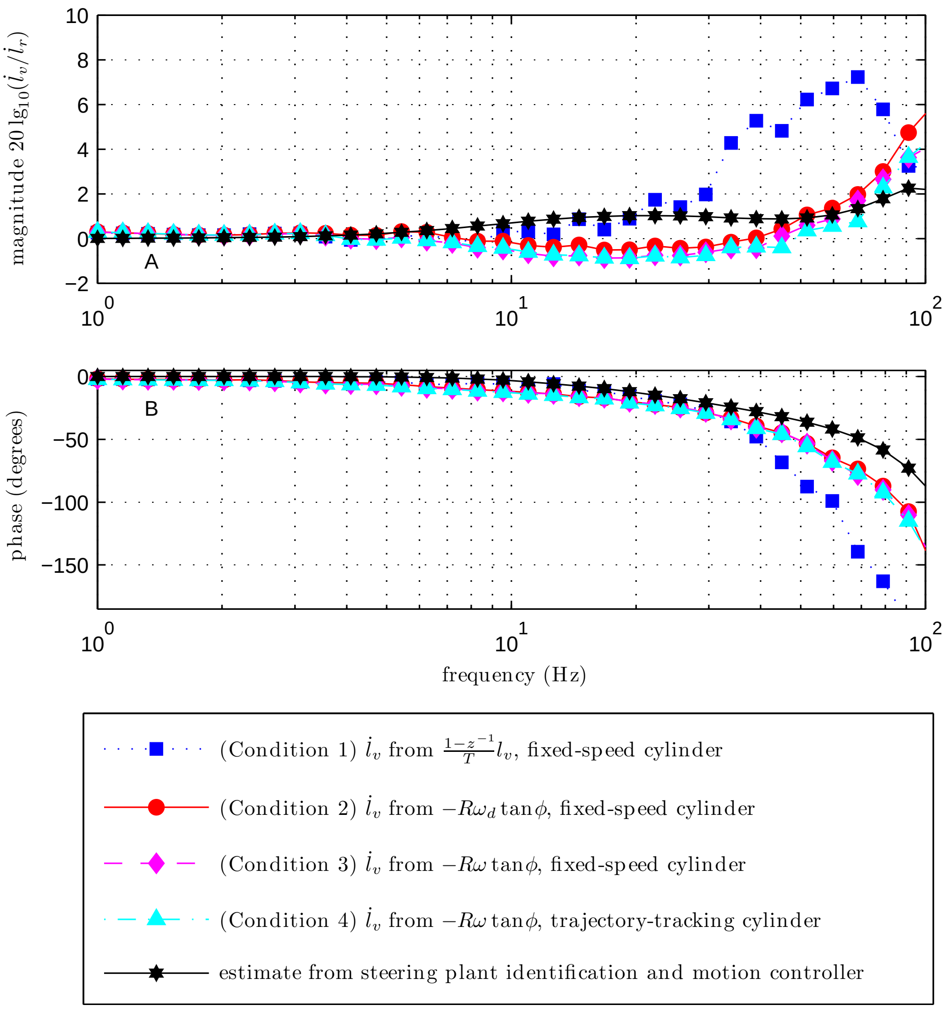


Figure 5.4. Closed loop joint motion control frequency response for various measures of joint velocity utilized for feedback, and for various cylinder controllers.

E.2. There the response was a flat line at -3 dB that rolled off at 40 dB/decade starting at 100 Hz. Since the actual transmission does not exactly obey ($\dot{l} = -R\omega \tan(\phi)$) and the cylinder introduces some unwanted dynamics, a PID joint motion controller with a feedforward component was added (Figure 3.3). This controller pulls up the gain to 0 dB, and compensates for slip, creep and measurement and actuation errors in the steering plant.

The set of gains deemed most useful for this controller yielded the predicted and measured responses depicted in Figure 5.4. The predicted and experimental responses all have a gain greater than unity as 100 Hz is approached. Condition 1 led to the most unwanted amplification as it uses an electrically noisy analog signal that has been software differentiated. Conditions 2, 3 and 4 led to unwanted amplification since they utilize an estimated velocity signal that is out of phase with the true motion of the joint, as a consequence of slip and creep in the transmission. Conditions 2, 3 and 4 all yielded equivalent responses, and 3 and 4 were then utilized most often for haptic display. Condition 1 severely limited the natural frequency of a virtual environment impedance that can be stably simulated, since it has gain greater than unity starting near 10 Hz. Unfortunately, Conditions 2, 3 and 4 are attenuated from 10 to 40 Hz by as much as 1 dB. This led to some of the sluggishness we saw in Section 5.2, but did not lead to instability. Conditions 2, 3 and 4 have gain greater than unity above 40 Hz, and we will show in Experiment 5.5 that the natural frequency of stable virtual environments is limited to 40 Hz.

For all intents and purposes, the fixed-speed versus energy-tracking cylinder controllers have not affected the frequency response, since the test takes place at near zero joint motion where very little change occurs in the virtual environment total energy and in the desired cylinder speed. While doing this experiment at different cylinder speeds would require more or less steering torque and steering excursion, the magnitude and phase plots would be identical if the steering actuators do not saturate.

5.4. Trajectory tracking experiment

5.4.1. Purpose

As another demonstration of motion control capabilities, we examine the ability of the manipulator to track a trajectory in task space.

5.4.2. Methods

In order to demonstrate the ability of the Cobotic Hand Controller to autonomously track a trajectory, we command it to follow a spiral in the plane orthogonal to the cylinder axis. No rotation or x_3 translation is required. The desired position spirals outward for one revolution, at which point a step change to a larger circular trajectory occurs. The appropriate desired position, velocity, and acceleration information is commanded to the motion control system. The force sensor is not used for this experiment. A virtual mass of 2.0 kg is used in conjunction with the desired velocity so that the energy-tracking controller can adjust the cylinder speed appropriately.

5.4.3. Results

Figure 5.5 shows the desired and actual trajectory of the cobot when the energy-tracking cylinder controller is used. Figure 5.6 shows the translational error between

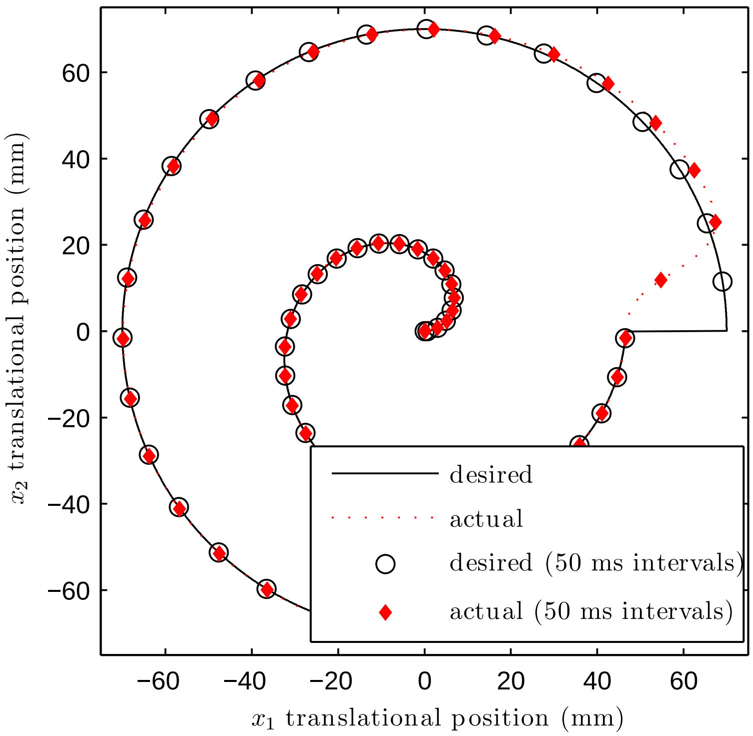


Figure 5.5. Tracking a spiral and circle trajectory. The experiment starts at the center and spirals outward for 1.6 seconds before abruptly (discontinuously) switching to a circle for another 1.1 seconds. The maximum speed required is 0.28 m/s.

desired and measured position for the energy-tracking and fixed-speed cylinder controllers in Subplots A and B, respectively. For both, the steady-state error is about 0.3 mm for the velocities incurred and the ninety-five percent rise-time for the step is about 0.1 seconds.

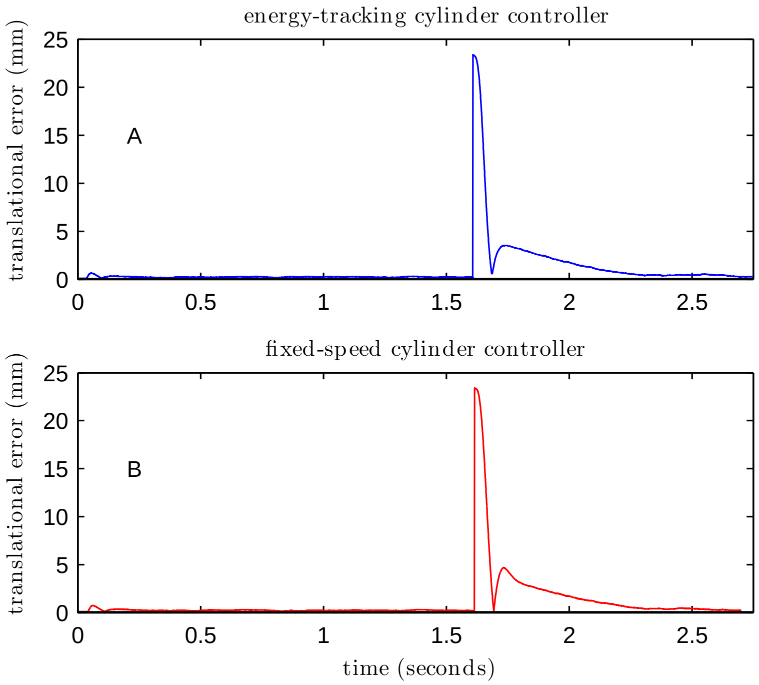


Figure 5.6. Both the energy-tracking and fixed-speed cylinder controllers have nearly identical steady-state tracking errors and rise-time characteristics.

5.4.4. Discussion

The fixed-speed and energy-tracking cylinder controllers both have nearly identical disturbance and steady-state characteristics as would be expected from their identical frequency responses in Experiment 5.3. While the ninety-five percent rise-time of 0.1 seconds may seem slow for a system with 40 Hz capability, the maximum steering velocity heuristic and creep/slip in the transmissions are limiting the acceleration for this abrupt step input (which nominally asks for discontinuous acceleration).

5.5. Impedance range experiment

5.5.1. Purpose

We establish the range of virtual admittances or impedances that the cobot can stably render depending on the real mechanical impedance the end-effector contacts.

5.5.2. Methods

This is similar to the protocol outlined in Figure 5.1 for Section 5.1. Similar testing protocols are implemented by others for determining the stable impedance range of a haptic display [36, 51, 53, 127]. This test was done with all six legs intact. The test protocol required the end-effector to be brought into contact with the real spring (moving along the axis of the cylinder at 5 cm/s until a 5 N load was detected. After a 0.25 second delay, the virtual environment simulation was started and a stability metric applied. The simulation consisted of a virtual spring, $k_{virtual}$, with zero point at the 5 N load point of the real spring, $k_{physical}$, a virtual mass, $m_{virtual}$, and a virtual damper, $b_{virtual}$. f_{bias} is zero in this experiment.

The stability algorithm is as follows: After the 0.25 second delay, the simulation is started[2] and let run for time $4\tau = 8\frac{m_{virtual}}{b_{virtual}}$, or a minimum of 0.25 seconds to a maximum of 2.5 seconds. τ is the rise time of a second order system, and 4τ is the two percent settling time criterion. After the first 4τ, metrics are recorded during the second and third 4τ intervals, for time t from $4\tau < t < 8\tau$ and $8\tau < t < 12\tau$,

[2] Overall, the simulation is not started at an equilibria - there is a small twitch as the simulation settles from the initial 5N compression point of the actual spring to the true steady-state equilibrium point.

respectively. In order for a trial to be deemed stable, the standard deviation of measured position during the third interval must be less than 30 μm. Since this is a highly nonlinear plant dependent on initial conditions, discrete sampling, etcetera, we repeated the test for each set of conditions several times.

We chose a mass, and then explored the damping-stiffness plane at that mass, using a grid of logarithmically spaced values. We started with the minimum damping value and minimum stiffness value, and gradually increased stiffness. Values for $k_{virtual}$ and $b_{virtual}$ are logarithmically distributed with 5 points per decade. Whenever a single unstable trial was found, we repeated the trial. If it was again unstable, we reduced the stiffness one level and tried again. Once we found a stiffness value that was stable in 4 out 5 trials, we reduced the stiffness and tried for stability in 5 out 5 trials. If we get 4 out of 5 trials, and then 5 out of 5 trials stable at the next lower level, that lower stiffness level was recorded as the maximum stable point for the given damping level. Otherwise, we searched for 5 out of 5 stable trials at a lower stiffness. In effect we tried to find a boundary of stiffness with a certain level of confidence as shown qualitatively in Figure 5.7. Once a stable stiffness value is found at a given damping, damping is raised by one level. However, stiffness need not be set back to zero, and the previous stable stiffness value can be used as a starting point for the new damping value.

We tested 4 different virtual masses over a range of virtual damping and virtual stiffness. In addition, the whole experiment was performed with three different physical springs, $k_{physical}$. The first spring was a 1000 N/m die spring, approximately 30

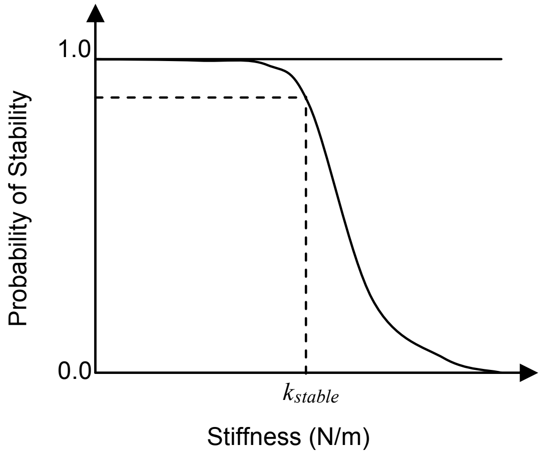


Figure 5.7. Probability of a trial yielding a stable result.

cm long and 3.5 cm in diameter, which was slid onto a cylinder for support as it lay horizontally. The second spring was a 3000 N/m die spring, also 30 cm long and 3.5 cm in diameter, and again slid onto a cylinder for support as it lay horizontally. The third was a 6000 N/m piece of 12 mm thick, 40 durometer polyurethane.

The experiment was performed with the fixed-speed cylinder controller set at 5.0 rads/s, and with the maximum steering velocity heuristic actively limiting steering torque if $\dot{\phi}_{max}$ was to be exceeded (see Section 3.5).

5.5.3. Results

In Figures 5.8 and 5.9 we plot the results of the Z-width exploration tests. Figure 5.8 shows the stable virtual stiffness/damping regimes for the three physical springs examined. Figure 5.9 plots this stability regime in a natural-frequency versus damping-ratio coordinate system, eliminating the need for separate curves for each virtual mass tested.

5.5.4. Discussion

Regardless of the impedance, $k_{physical}$, that the end-effector contacted, increasing the mass, $m_{virtual}$, always led to higher stable stiffness, $k_{virtual}$, and to a larger range of stable values for damping, $b_{virtual}$. Increasing the impedance, $k_{physical}$, from a 1000 N/m to a 3000 N/m die spring led to a reduced natural frequency that could be stably rendered (from 32 Hz down to 22 Hz). However, with the additional damping of the polyurethane, the 6000 N/m 40 durometer pad was as stable as the 1000 N/m die spring.

For a small virtual mass, the resulting high accelerations caused torque saturation of the steering control which led to instability. The high desired joint accelerations that result from using a small mass also led to slip based instabilities that escaped the steering velocity heuristic.

For a small mass and the 3000 N/m spring we heard an audible high frequency buzz, even though our algorithm proclaimed it stable.

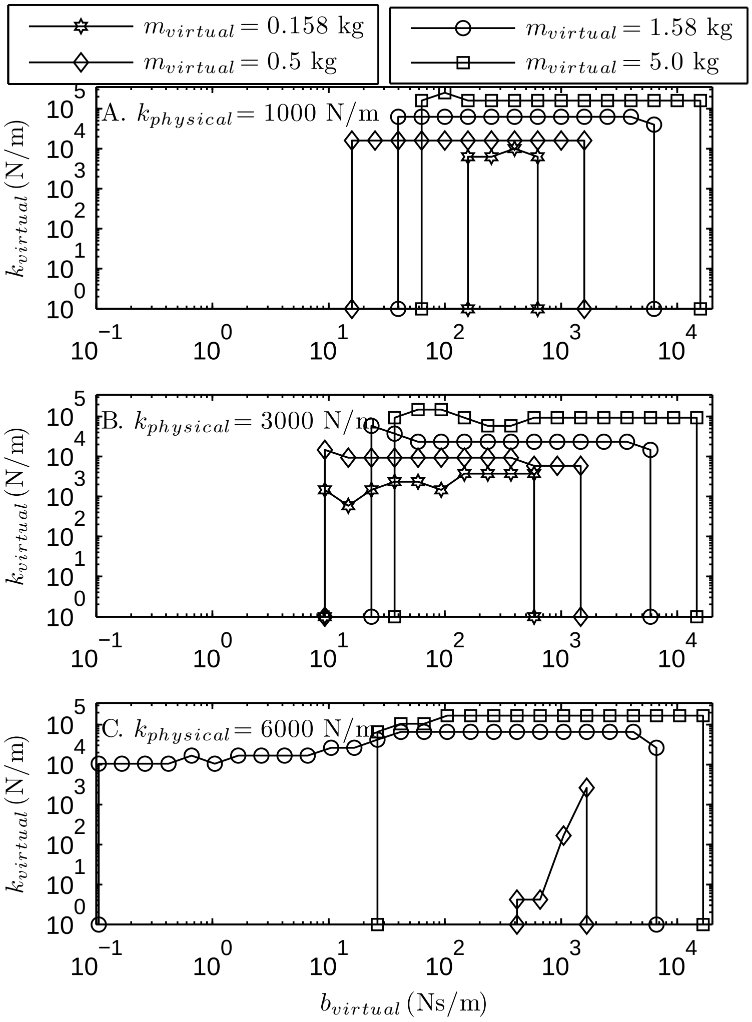


Figure 5.8. Stability regime in the virtual stiffness and virtual damping plane for various virtual inertias. A. Physical impedance $k_{physical}$=1000 N/m die spring. B. Physical impedance $k_{physical}$=3000 N/m die spring. C. Physical impedance $k_{physical}$=6000 N/m 40 durometer polyurethane pad. There were no stable points for $m_{virtual}$ = 0.158 kg with the polyurethane.

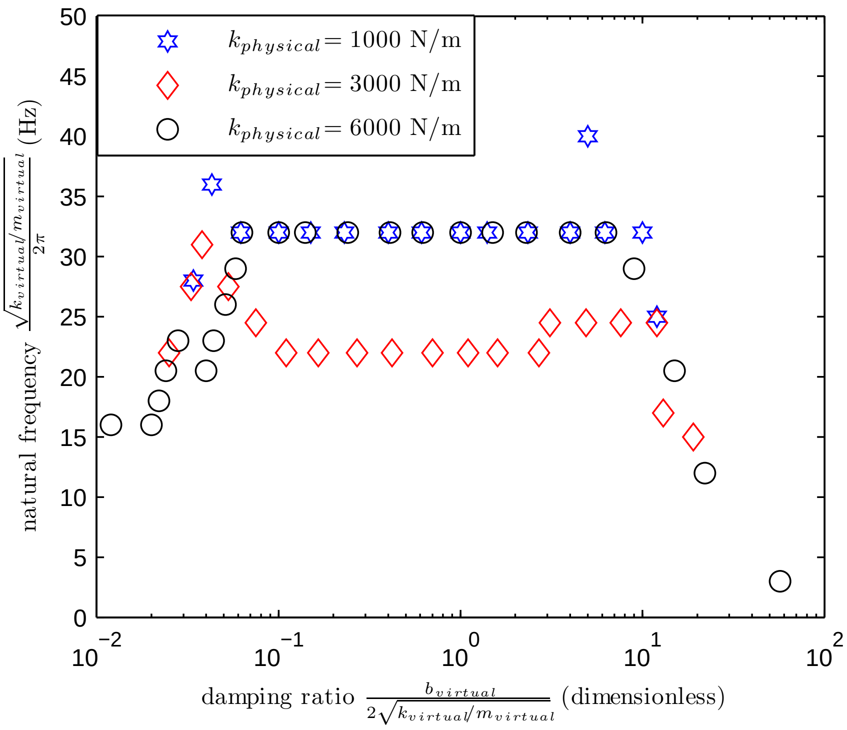


Figure 5.9. Stable regime in natural frequency and damping ratio plane for various physical impedances $k_{physical}$. Switching from a 1000 N/m die spring to a 3000 N/m die spring decreases the range of frequencies that can be stably rendered. However, increasing to an even stiffer 6000 N/m polyurethane pad results in an increase in the stable range due to the presence of significant damping in the pad relative to the die springs.

We could also plot the impedance range for the energy-tracking cylinder controller, however, since the joint velocities are not high in this experiment, the controller would not vary the cylinder speed and would become essentially a fixed-speed controller.

CHAPTER 6

Focus 3a: Reduction element losses

Even when fixed at a certain angle, the rolling wheel is not 100 percent efficient at transmitting power between the cylinder and joint. In this chapter we analyze inefficiencies arising from elastic material properties in the form of lateral creep, from inelastic material properties in the form of free-rolling friction, and from friction in the CVT wheel axle bearings. The three methods are first discussed independently, and then combined, in order to predict the efficiency of the rolling contact reduction element. We then compare these predictions with experimental data from the Cobotic Hand Controller.

The modelling within this chapter is a steady-state analysis of the transmission, ($\dot{\phi} = 0$, $\dot{\omega} = 0$), with static loading conditions and constant cylinder velocity. In general, no method except a full numerical method (not discussed in this thesis) is applicable for analyzing dynamic rolling scenarios of elastic bodies where the rate of change of material strain is nonzero. By dynamic we mean $\dot{\omega} \neq 0$, or $\dot{\phi} \neq 0$, or changing loading conditions. Linear-elastic media models such as those found in [87] and [89] only apply well for small relative angular velocities that are constant ($\dot{\phi} \approx 0$, $\ddot{\phi} = 0$). Another issue that limits linear-elastic media models to $\dot{\phi} \approx 0$ is that longitudinal and lateral creeps cannot be accurately combined at high rates of relative angular velocity.

6.1. Background material

6.1.1. Coordinate frames

Shown in Figure 6.1, Subplot B, is wheel W rolling on cylinder C where the shaded area represents a contact patch that is much wider in the lateral direction. The origin O of the wheel-fixed frame N, moves at $\dot{l}$ relative to cylinder C. The $\hat{n}_1$ direction points along the wheel axle, $\hat{n}_2$ along the wheel heading and the $\hat{n}_3$ direction is always normal to the cylinder surface. Wheel W has radius, r, and rolling angular velocity, $\dot{\theta}$. Torque $\tau_{wheelaxle}$ combats friction in the wheel axle and torque $\tau_{inelastic}$ combats inelastic behavior at the wheel-cylinder interface. Cylinder C has radius R and angular velocity ω. The cylinder requires torque $\tau_{c,nominal}$ to combat friction in its own bearings. The cylinder delivers torque, τ_c, to the transmission. A portion of this, $\tau_{c,wheel}$, is lost to transmission inefficiencies, and the remainder flows through the transmission to the joint. The steering angle of wheel W is ϕ and the steering angular velocity is $\dot{\phi}$. Torque τ_b steers the wheel assembly. The lateral half-width of the contact patch is a, in the $\hat{n}_1$ direction, and the longitudinal half-width is b, in the $\hat{n}_2$ direction. The wheel fixed N-frame location of arbitrary point Q in the contact patch is given by x, in the $\hat{n}_1$ direction, and y, in the $\hat{n}_2$ direction. The notation introduced in this figure and other notation for this chapter are summarized in Table 6.1.

Table 6.1. Reduction element losses notation.

Variable	**Definition**
C, W	cylinder and wheel objects
O	origin of wheel-fixed frame N
Q	arbitrary point in wheel-fixed frame N
x, y	$\hat{n}_1$ and $\hat{n}_2$ coordinates in frame N
a, b	lateral ($\hat{n}_1$) and longitudinal ($\hat{n}_2$) contact patch half-widths
u_a, u_b	lateral ($\hat{n}_1$) and longitudinal ($\hat{n}_2$) creeps of wheel relative to cylinder
f_a, f_b	lateral ($\hat{n}_1$) and longitudinal ($\hat{n}_2$) forces from cylinder on wheel
ξ, ζ	lateral ($\hat{n}_1$) and longitudinal ($\hat{n}_2$) dimensionless creeps (e.g., $\xi = \frac{u_a}{r\dot{\theta}}$)
$\phi_g(f_w)$	angle at which gross slip occurs for transmission output force f_w
ϕ_s	slip angle description for lateral creep
ψ_{kj}	spin definition of Kalker and Johnson
ψ_g	spin definition of Gillespie
ν	Poisson's ratio
G	combined shear modulus of two bodies in rolling contact
C_{ij}	nondimensional creep coefficients tabulated from numerical models
α_{fr}	hysteresis loss factor characterizing rolling friction
$\dot{W}_{elastic}$	power loss due to lateral creep
$\dot{W}_{inelastic}$	power loss due to rolling friction
$\dot{W}_{wheelaxle}$	power loss due to axle bearing of the CVT wheel
$\dot{W}_{strain}$	rate of strain work done at wheel/cylinder interface
$\tau_{c,nominal}$	nominal torque required to turn cylinder in absence of any wheels
τ_c	cylinder torque delivered to transmission, including $\tau_{c,wheel}$
$\tau_{c,wheel}$	cylinder torque per wheel required to combat $\tau_{inelastic}$ and $\tau_{wheelaxle}$
$\tau_{inelastic}$	rolling friction torque on wheel due to inelastic rolling contact
$\tau_{wheelaxle}$	rolling friction torque on wheel due to CVT wheel axle bearing friction

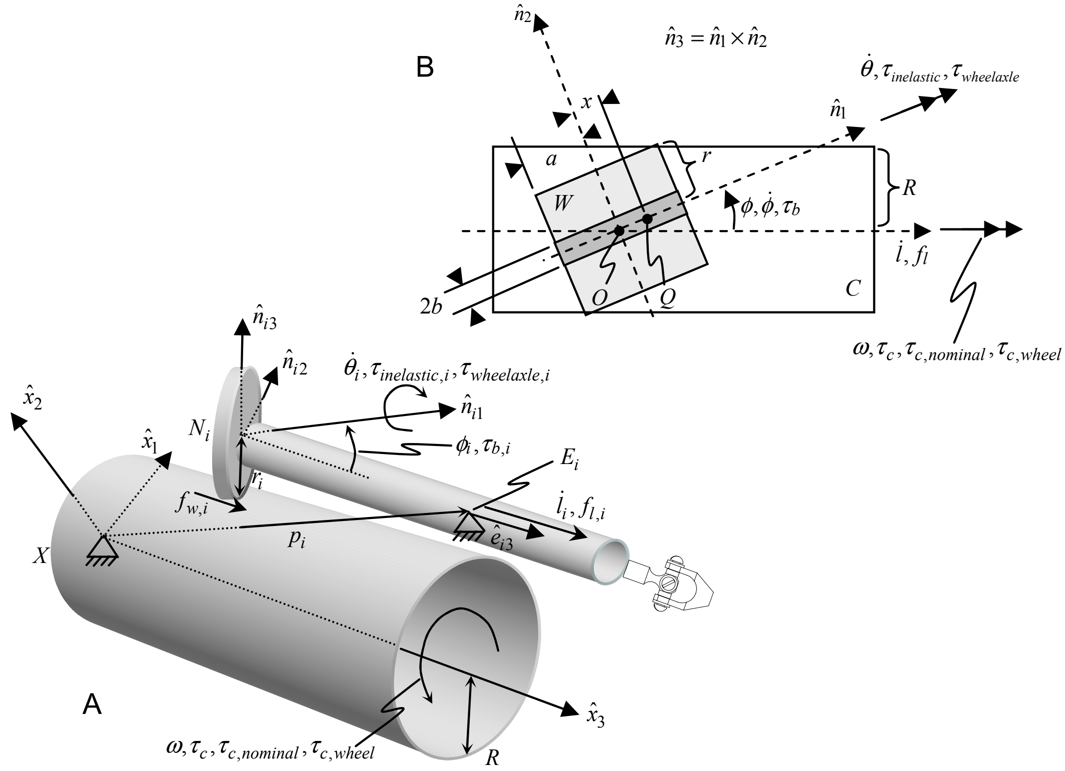

$\hat{n}_2$
$\hat{n}_3 = \hat{n}_1 \times \hat{n}_2$
B
$\dot{\theta}, \tau_{inelastic}, \tau_{wheelaxle}$
x
$\hat{n}_1$
a
r
R
W
$\phi, \dot{\phi}, \tau_b$
$\dot{l}, f_l$
$2b$
O
Q
C
$\omega, \tau_c, \tau_{c,nominal}, \tau_{c,wheel}$
$\hat{n}_{i3}$
$\hat{n}_{i2}$
$\dot{\theta}_i, \tau_{inelastic,i}, \tau_{wheelaxle,i}$
$\hat{n}_{i1}$
$\hat{x}_2$
$\hat{x}_1$
N_i
$\phi_i, \tau_{b,i}$
E_i
r_i
$f_{w,i}$
$\dot{l}_i, f_{l,i}$
p_i
$\hat{e}_{i3}$
X
A
$\hat{x}_3$
$\omega, \tau_c, \tau_{c,nominal}, \tau_{c,wheel}$
R

Figure 6.1. A. Reproduction of Figure 2.15. B. Notation for the geometry of the contact patch is introduced here. While the contact patch is shown as a rectangle, we also model it as an ellipse or line contact for various purposes. The wheel fixed frame N moves relative to the cylinder at linear velocity $\dot{l}$ and angular velocity $\dot{\phi}$.

6.1.2. Contact patch shape

Initially, a new wheel for the Cobotic Hand Controller is perfectly spherical. Since it is much smaller than the cylinder ($\frac{r}{R} \ll 1$), the contact patch will be nearly a circle ($\frac{a}{b} \approx 1$). The surface pressure will be greatest at the center of this contact patch, but the greatest shear stress (the actual failure mode of material if high enough) of 300

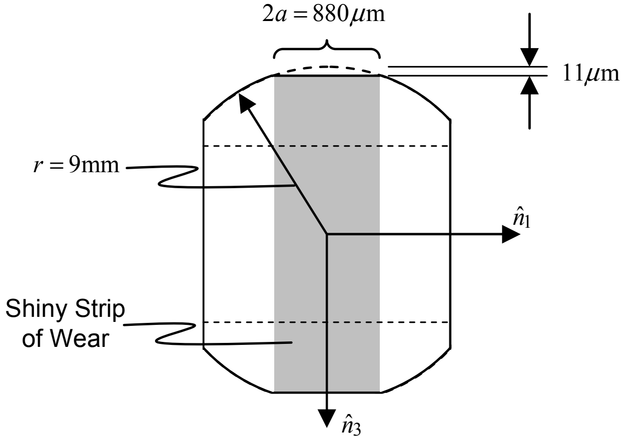


Figure 6.2. A sketch of a worn wheel - not to scale. A shiny strip of width $2a = 880\mu$m appears quickly, and the radius of the spherical wheel is reduced by 11μm. No significant wear occurs after the initial break-in period.

MPa, is at a depth of 56 μm in the center of the contact patch.[1] This shear stress is both below the strength of the material, and not at a location where we see failure. Even once the wheels have worn a flat with half-width $a = 440\mu$m, only a depth of 11μm has been physically worn away (see Figure 6.2). We do not expect failure at the surface due to compressive loading at all since P=250 N is a small load for a ball of this size of bearing hardness.

However, the simple Hertzian contact calculations have not included the stresses introduced by steering (i.e. when $\dot{\phi} \neq 0$). Although our wheels are bearing grade components, they experience a harsher environment than is typical for bearings. While

[1] The calculations for Hertzian contact of two elliptical bodies are quite lengthy. A Java applet at the Tribology Laboratory of the University of Florida (`http://grove.ufl.edu/~wgsawyer/Laboratory/Software/Elliptical/EllipticalContacts.HTML`) [135], based on equations of Hamrock and Brewe [73], makes these calculations simple.

gear teeth and some bearings experience surface shearing action, it is not concentrated at a point contact. It is our hypothesis that the steering action, in addition to the pressure due to load P, causes the wheel's shear yield stress to be exceeded at the surface of the wheel. As the wear and the resulting shape of the wheel progresses, the combination of loadings eventually falls below the wheel's shear yield stress, and wear ceases to occur.

In Figure 6.1, the contact patch is depicted as rectangular with lateral half-width, a, and longitudinal half-width, b. Assuming a rectangular contact patch between two parallel cylinders, and having measured the half-width a of a broken-in wheel, we compute our eccentricity $\frac{b}{a} = 0.37$ from Appendix D, Figure D.3. However, if the wheel is steered away from $\phi = 0$, the contact patch shape and the pressure distribution changes (approaching that of two orthogonal cylinders) and approximately resembles that of a virgin sphere on a plane, elliptical rather than rectangular.

6.1.3. Forces at contact patch

Figure 6.3 diagrams the forces acting at the wheel-cylinder interface. The wheel sustains a lateral force of $f_a = f_w \sec\phi$ along the $\hat{n}_1$ direction (Equation 6.1). This lateral force ultimately arises from the joint force, $f_w = m_l\ddot{l} + c_{d,l}P\text{sgn}(\dot{l}) - f_l$, which is opposed by a force arising from cylinder torque, $f_w = -\frac{\tau_c - \tau_{c,wheel}}{R\tan\phi}$. The wheel cannot sustain a longitudinal force, f_b, along the $\hat{n}_2$ direction since no torque is applied to the wheel's axle in order to prevent angular acceleration (except for friction within the bearings).

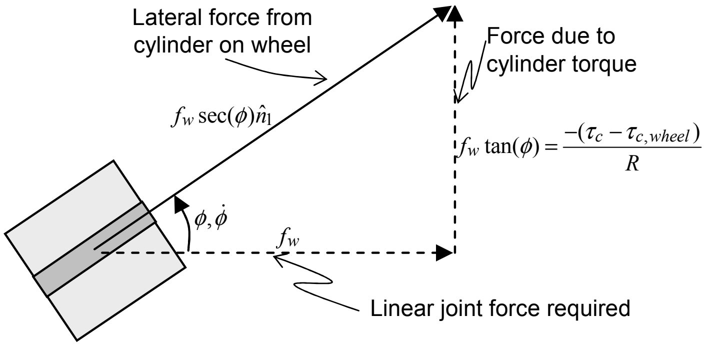

Lateral force from cylinder on wheel
Force due to cylinder torque
$f_w \sec(\phi)\hat{n}_1$
$f_w \tan(\phi) = \frac{-(\tau_c - \tau_{c,wheel})}{R}$
$\phi, \dot{\phi}$
f_w
Linear joint force required

Figure 6.3. Forces from cylinder acting on the wheel at the contact patch.

$$\begin{aligned} f_a\hat{n}_1 &= f_w \sec(\phi) \\ f_b\hat{n}_2 &= 0 \end{aligned} \tag{6.1}$$

In addition, when $\dot{\phi} \neq 0$, a torque is applied on the wheel by the cylinder about the $\hat{n}_3$ axis. This torque is always zero for Chapter 6 where $\dot{\phi} = 0$, although it is present when the wheel is steering. These loading conditions are very different from the drive rollers of the spherical CVT analyzed by Gillespie [67], Brokowski [20] and Kim [99]. In their investigations they modelled tractive effort[2], the longitudinal forces present at the contact patches of the drive rollers. The drive rollers of the spherical CVT do not sustain lateral forces. In addition to tractive effort, the drive rollers of the spherical CVT sustain a continuous torque about the contact normal since they are spinning relative to the sphere underneath. Conversely, the steering

[2] Here we delineate *tractive-rolling* from *free-rolling*. The drive rollers of the spherical CVT experience *tractive-rolling* as the axial torques on these wheels are the inputs and outputs of the spherical CVT. The Cobotic Hand Controller's wheels and the steering rollers of the Spherical CVT experience only *free-rolling*, as no axial torques are applied.

rollers of the spherical CVT do not experience this spin or sustain tractive rolling, but instead sustain lateral forces in order to keep the rotational axis of the sphere positioned. Thus the steering rollers of the spherical CVT are under very much the same loading conditions as the wheels of the rotational-to-linear CVTs of the Cobotic Hand Controller.

The static angle ϕ_g, at which the rotational-to-linear CVT will enter gross slip for a given output force f_w, can be computed by comparing the lateral force $f_a = f_w \sec(\phi)$ to the preload μP (Equation 6.2).

$$\phi_g = \operatorname{arcsec}\left(\frac{\mu P}{f_w}\right) \tag{6.2}$$

It is evident from Equation 6.2 that $\phi_g = 0$ will be the angle for which the system can output the greatest force f_w and therefore produce the greatest acceleration of the joint mass.

6.1.4. Longitudinal creep definition

If we had a tractive effort, f_b, in the rolling direction, between an elastic wheel and/or substrate, a longitudinal creep, ζ, the ratio between the creep velocity u_b and the rolling velocity $|r\dot{\theta}|$, would result (Equation 6.3).

$$\zeta = \frac{u_b}{|r\dot{\theta}|} \tag{6.3}$$

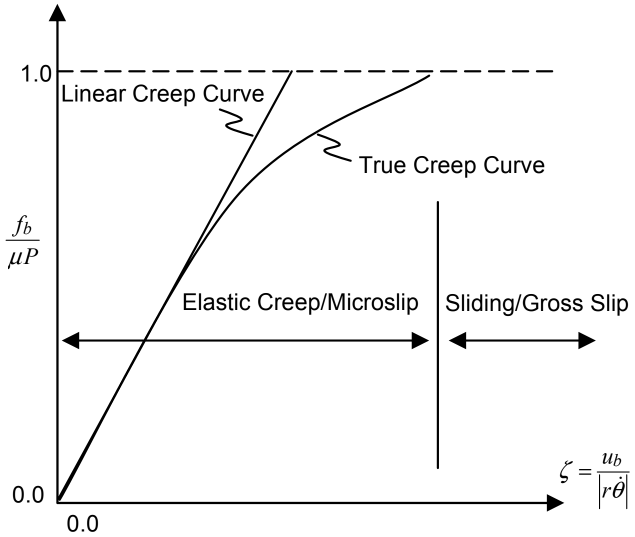


Figure 6.4. Typical longitudinal creep curves, including the true phenomena, and a linear approximation to the phenomena.

The creep coefficients are essentially the fraction of power passing through a contact patch that will be dissipated. A typical longitudinal creep curve is portrayed in Figure 6.4 along with a curve predicted by the *linear creep* model. This curve is reflected through the origin if the tractive effort is braking rather than driving. This longitudinal creep curve is independent of any spin between the two rolling bodies (this will not be the case for the lateral creep curve).

6.1.5. Lateral creep definition

Lateral creep, ξ is the ratio between lateral creep velocity u_a and rolling velocity $|r\dot{\theta}|$ (Equation 6.4).

$$\xi = \frac{u_a}{|r\dot{\theta}|} \tag{6.4}$$

Lateral forces, f_a, result in much more complex creep phenomena, particularly in the presence of spin ($\dot{\phi} \neq 0$). For example, a cornering car will actually creep towards the center of rotation of the car, even in the presence of large centripetal forces outward. This is termed camber thrust. In Figure 6.5 we can see that the *true creep curve* for lateral creep actually shifts left from the origin and changes slope in the presence of spin. The linear models of lateral creep predict the change in slope, but not the leftward shift. With $\dot{\phi} = 0$, the *linear creep curve* matches the true creep curve nicely. However, when spin is introduced ($\dot{\phi} \neq 0$), camber thrust and thus the lateral creep velocity, u_a, are not predicted by the linear creep models.

The leftward shift of the true creep curve (camber thrust) reaches a maximum for a certain spin value (dependent on eccentricity), and then decreases, once again aligning with the linear creep curve for high spin velocities. This leftward shift is minimized by reducing the ratio b/a, effectively stiffening the wheel against camber thrust due to spin.

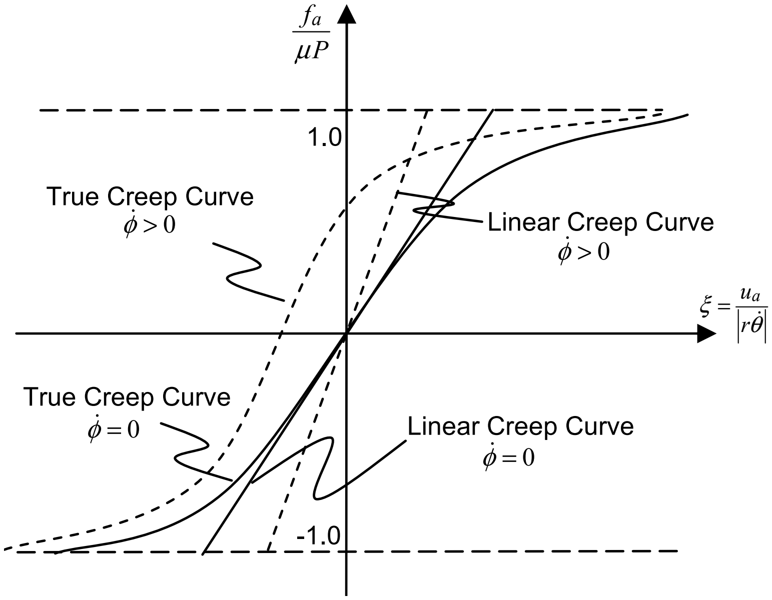


Figure 6.5. Typical lateral creep curves with and without steering velocity. As steering velocity increases, the creep curve steepens and shifts leftward. The leftward shift of the creep curve is a phenomena known as camber thrust.

6.2. Elastic losses

6.2.1. Background

Table 6.2 summarizes the applicable models for elliptical contact patches depending on the presence of lateral and/or longitudinal loads, and the magnitude of spin, ψ_{kj}. $\psi_{kj} = \frac{\dot{\phi}(ab)^{1/2}}{|r\dot{\theta}|}$ is a nondimensional expression for spin found in Kalker [89] and

Table 6.2. Applicable regimes of rolling contact models for elliptical contact patches.[a]

	$\psi_{kj} = 0$	$\psi_{kj} \approx 0$	$\psi_{kj} \gg 0$
$f_a = 0,\ f_b \neq 0$	longitudinal creep	kinematic-creep or complete-slip	kinematic-creep or complete-slip
$f_a \neq 0,\ f_b = 0$	lateral creep	numerical simulation only due to camber thrust	asymptotes back to lateral creep as $\dot{\phi}$ increases
$f_a \neq 0,\ f_b \neq 0$	linear creep	linear creep	numerical simulation only

[a] The first row applies to the drive rollers of the spherical CVT, the second row to the rotational-to-linear CVT and the steering rollers of the spherical CVT, and the final row to a general wheel, experiencing both lateral and longitudinal loading. Eccentricity of the contact patch and the magnitude of the friction coefficient is used to decide the threshold of spin at which to switch from the second to third columns.

Johnson [87].[3] The contact patch geometry and friction coefficient determine whether the case for $\psi_{kj} \approx 0$ or $\psi_{kj} \gg 0$ is relevant.[4]

If we consider an elliptical patch, the theory of Vermeulen and Johnson [161] (Johnson Equations 8.45 and 8.46 [87], Kalker Sections 2.2.2 and 5.2.1 [89]) serves as the basis of all applicable algorithms including the *linear creep* model we employ. Their theory, supported by experiment, predicts an elliptical subset of the elliptical contact patch to be in adhesion, while the rest is in slip. Those portions in adhesion

[3] Gillespie et al. define another spin, $\psi_g = \frac{\dot{\phi} a}{|r\dot{\theta}|}$ [67]. This only involves one of the contact patch half-widths since his model assumes a line contact.

[4] If spin is absent and the ratio b/a is such that the contact patch is nearly rectangular, Carter's [27] theory for the contact of two cylinders would also apply (Johnson, Equations 8.26 and 8.31 [87]).

are characterized by static friction, and those in slip by kinetic friction. The equations are complicated, particularly when improved upon by others, and have certain terms numerically evaluated and tabulated for a range of contact patch eccentricities and Poisson's ratios.

6.2.2. The linear creep model

Equation 6.5 presents the *linear* creep equations in terms of nondimensional creep coefficients C_{11}, C_{22} and C_{23}, derived from a numerical model and tabulated by Kalker.

$$\begin{aligned} \zeta &= -\frac{1}{C_{11}Gab} f_b \\ \xi &= -\frac{1}{C_{22}Gab} \left(f_a + C_{23}G(ab)^{3/2}\psi_{kj} \right) \end{aligned} \tag{6.5}$$

$G = (2(G_1^{-1} + G_2^{-1}))^{-1}$ is the combined shear modulus of elasticity for the two bodies in rolling contact. Although Carter's or Vermeulen and Johnson's complete theories closely approximate the true creep curve since they are nonlinear, the tabulated versions shown in Equation 6.5 are only a first order *linear* approximation. These linearized results are accurate for small to no spin and for forces less than approximately 60 percent of the slip force (refer to the creep curves in Figures 6.4 and 6.5). The linear models do not predict gross slip when the preload is reached, and, therefore, will under-estimate the losses due to slip. The term linear has a double meaning here. While the model is a linear approximation to the complete numerical solution, longitudinal and lateral effects can be computed separately and then combined linearly (this is not an issue for us since we only have lateral loading). In addition, this approximate model is capable of predicting the steepening of the lateral creep curve with spin (note the dependence of Equation 6.5 on ψ_{kj}), but does not predict the leftward shift due to camber thrust.

6.2.3. Creeps for the Cobotic Hand Controller

For $b/a = 0.37$ and for Poison's ratio $\nu = 0.3$, the creep coefficients we use are $C_{11} = 3.70$, $C_{22} = 2.87$ and $C_{23} = 0.80$ (Kalker Table E3 [89] or Johnson Appendix 5 [87]). We select a combined shear modulus of rigidity, $G = 78$ GPa, along with contact patch dimensions $a = 440$ μm and $b = 163$ μm. In the absence of spin or longitudinal loading, Equation 6.5 simplifies to Equation 6.6.

$$\begin{aligned} \zeta &= 0 \\ \xi &= -\frac{f_a}{C_{22}Gab} = -6.23 \times 10^{-5} f_a \end{aligned} \tag{6.6}$$

Using Equations 6.4 and 6.3, the slip velocities of the wheel relative to the cylinder are given by Equation 6.7.

$$\begin{aligned} u_b &= 0 \\ u_a &= -\frac{f_a}{C_{22}Gab}|r\dot{\theta}| \end{aligned} \tag{6.7}$$

Recall that creeps are a ratio of slip velocity to baseline longitudinal rolling velocity. Therefore, the total power dissipated is the product of the slip velocities in the longitudinal and lateral directions, $u_b = \zeta|r\dot{\theta}|$ and $u_a = \xi|r\dot{\theta}|$, with the applied forces of the cylinder on the wheel in those directions, f_b and f_a. The power dissipated due to elastic based creeps for the Cobotic Hand Controller will only contain a lateral creep component and is expressed in Equation 6.8.

$$\dot{W}_{elastic} = u_a f_a = \frac{\sec^3(\phi) R|\omega| f_w^2}{C_{22}Gab} \tag{6.8}$$

In Equation 6.8 we make use of the fact that $f_a = \sec(\phi) f_w$ and $R\omega = -r\dot{\theta}\cos(\phi)$. We remove a minus sign since we choose to have the dissipation of power be a positive quantity for a damper.

6.3. Inelastic losses

Power is also dissipated due to the inelastic compression and restitution of the wheel and cylinder as they roll together. This is known as free-rolling friction. The rate of strain work, $\dot{W}_{strain}$, done to the wheel and cylinder at the contact patch can be computed from an integral of the pressure distribution as shown in Equation 6.9 (Johnson pg 285) [87].

$$\dot{W}_{strain} = \dot{\theta}\int_0^b p(y) y dy = \frac{2Pb\dot{\theta}}{3\pi} \tag{6.9}$$

We assume that we have two cylinders, loaded together via preload P, with a thin rectangular contact of longitudinal half-width b, whose pressure distribution $p(y)$

in the longitudinal direction is given by Equation D.2. We use the two cylinder assumption (and rectangular contact patch approximation), since it is relatively true when $\phi = 0$ and inelastic losses (our present concern) are large relative to power throughput of the transmission. The power dissipated to these inelastic losses during free-rolling of a wheel, $\dot{W}_{inelastic}$, is the fraction, α_{fr}, of the rate of strain work, $\dot{W}_{strain}$, which is dissipated by hysteresis (Equation 6.10).

$$\dot{W}_{inelastic} = \alpha_{fr} \dot{W}_{strain} \tag{6.10}$$

α_{fr} is a material dependent coefficient of restitution, called the hysteresis loss factor, which relates the expansive work regained to the compressive work spent. α_{fr} is not a constant for a given material, since it increases with strain (b/r) and varies with rolling history. In our case, strain is fixed due to the constant preload and to the steady-state rolling history, and a fixed value for α_{fr} can be utilized.

Since the $\dot{W}_{inelastic}$ occurs at rolling speed $\dot{\theta}$, a rolling torque about the CVT wheel axis, $\tau_{inelastic}$, can be computed that represents the effort required to keep the wheel rolling due to inelastic losses (Equation 6.11).

$$\tau_{inelastic} = -\alpha_{fr} \mathrm{sgn}(\theta) \frac{2Pb}{3\pi} \tag{6.11}$$

The power dissipated by the inelastic rolling is given in terms of cylinder speed by Equation 6.12.

$$\dot{W}_{inelastic} = -\tau_{inelastic} \dot{\theta} = \alpha_{fr} |\omega| \frac{2Pb}{3\pi} \sec(\phi) \frac{R}{r} \tag{6.12}$$

$\dot{W}_{inelastic}$ is dependent on pre-load P but independent of loading f_a and f_b. Also note that Equation 6.12 is linear in cylinder speed rather than quadratic, yielding a constant torque Coulombic effect rather than viscous behavior.

6.4. CVT wheel axle bearing friction

In Figure 2.10, two flanged radial bearings are shown that sustain the radial preload on the CVT wheel, P, and the axial load $f_w \sec\phi$ (lateral load on the wheel) due to output effort of the joint. These bearings produce significant frictional dissipation that can be shown to be largely dependent on radial loading (preload P), and rolling speed, but relatively independent of axial loading. We utilize a *frictional moment and power loss* calculator provided by bearing manufacturer SKF at `http://www.skf.com/portal/skf/home/products?maincatalogue=1&newlink=first&lang=en` [155] in order to determine the axle bearing friction portion of rolling friction torque on the CVT wheel, $\tau_{wheelaxle}$.

We examine SKF bearing number 604, which is almost identical to those used for the CVT wheels, for several operating conditions and plot the results in Figure 6.6.[5] When no axial (lateral) forces are present ($f_a = f_w \sec\phi = 0$), $\tau_{wheelaxle} = 0.97$ mNm. If any axial (lateral) forces are present ($f_a = f_w \sec\phi \neq 0$), $\tau_{wheelaxle} = 1.24$ mNm. The resulting dissipated power, $\dot{W}_{wheelaxle}$, is computed via Equation 6.13 to be 36.6

[5] The loading conditions are halved before being entered into the model, and the results of the model doubled before being plotted in Figure 6.6, as the model is for a single one of the pair of radial bearings that each CVT wheel contains.

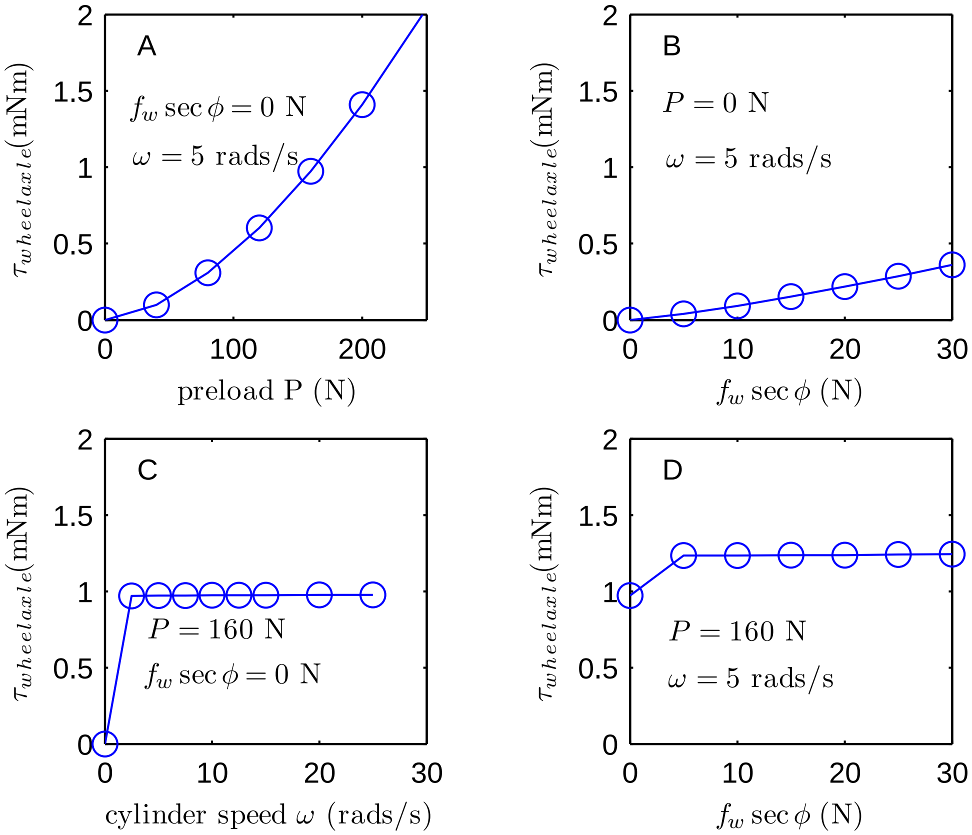


Figure 6.6. Investigation of the SKF bearing friction model allows us to approximate the bearing friction portion of rolling friction torque on the CVT wheel, $\tau_{wheelaxle}$, due to axial and radial loading on the bearing. A. We fix the axial load $f_w \sec\phi = 0$ N and the cylinder velocity $\omega = 5$ rads/s and find that the bearing friction is exponential in radial load P. B. Fixing radial load $P = 0$ N and cylinder velocity $\omega = 5$ rads/s we find that bearing friction is linear in the regime of axial loads that will be utilized. C. Fixing radial load $P = 160$ N (this is slightly less than the 250 N used elsewhere in the thesis, but is necessary to make this model align with the experimental data) and the axial load $f_w \sec\phi = 0$ N we find the bearing friction is relatively Coulombic (independent of cylinder speed). D. Operating at the radial load $P = 160$ N and the nominal cylinder speed $\omega = 5$ rads/s, the bearing friction torque is relatively independent of axial loading.

mW ($\omega = 5$ rads/s, $f_w \sec\phi = 0$ N) or 46.6 mW ($\omega = 5$ rads/s, $f_w \sec\phi = 10$ N).

$$\dot{W}_{wheelaxle} = -\tau_{wheelaxle}\dot{\theta} = \tau_{wheelaxle}\sec(\phi)\frac{R}{r}\omega \tag{6.13}$$

Like $\dot{W}_{inelastic}$, $\dot{W}_{wheelaxle}$ is linear in cylinder speed, as $\tau_{wheelaxle}$ is relatively independent of ω.

Before any power can be transferred to elastic creeps or to the joint, the cylinder must satisfy both the inelastic rolling friction and bearing friction by providing a torque, $\tau_{c,wheel}$, as defined in Equation 6.14.

$$\tau_{c,wheel} = -(\tau_{inelastic} + \tau_{wheelaxle})\sec\phi\frac{R}{r} \tag{6.14}$$

6.5. Rolling friction experiments

6.5.1. Purpose

To evaluate the magnitude of power dissipation composed both of inelastic losses at the contact patch ($\dot{W}_{inelastic}$) and of friction in the bearings for the axle of the wheel ($\dot{W}_{wheelaxle}$), and to determine whether the relationship between torque on the cylinder due to this dissipation and cylinder speed is predominantly Coulombic or viscous, by implementing the following two experiments.

6.5.2. Methods

In the first experiment we determine the average cylinder torque required to spin the cylinder at a constant speed of 5.0 rads/s for 10 revolutions of the cylinder, with the wheels steered to $\phi = 0$. This is done with six, then five, then four, then three, then two, then one and then with zero wheels present. A line is fit to the data, whose slope is the average cylinder torque $\tau_{c,wheel}$ required for each wheel, and the vertical-intercept of this line is the nominal cylinder torque, $\tau_{c,nominal}$, required to make the cylinder spin when no wheels are present.

In the second experiment, the cylinder torques required to drive a single wheel for cylinder speeds ranging from 0.25 rads/s to 25 rads/s are determined. The average cylinder torque for 10 revolutions of the cylinder at each speed is recorded.[6]

6.5.3. Results

From Figure 6.7 we find the nominal cylinder torque is $\tau_{c,nominal} = 0.0838$ Nm, and the average cylinder torque per wheel $\tau_{c,wheel}$ is 0.0077 Nm. Analyzing Figure 6.8, we find that the cylinder torque required to drive a single wheel only increases by a factor of 1.75, even when the cylinder speed is increased by two orders of magnitude from 0.25 rads/s to 25 rads/s.

6.5.4. Discussion

The first experiment (Figure 6.7) yields an average cylinder torque of $\tau_{c,wheel} = 7.7$ mNm per wheel for a 5 rads/s cylinder speed. This is equivalent to 38.5 mW of power dissipated per wheel. Since this power consists both of rolling friction due to inelastic behavior at the contact patch ($\dot{W}_{inelastic}$), and due to bearing friction ($\dot{W}_{wheelaxle}$), we

[6] This is not a constant during actual operation since the rigid coupling between the cylinder motor and cylinder kinks slightly. This effect is averaged out over time and does not affect our results.

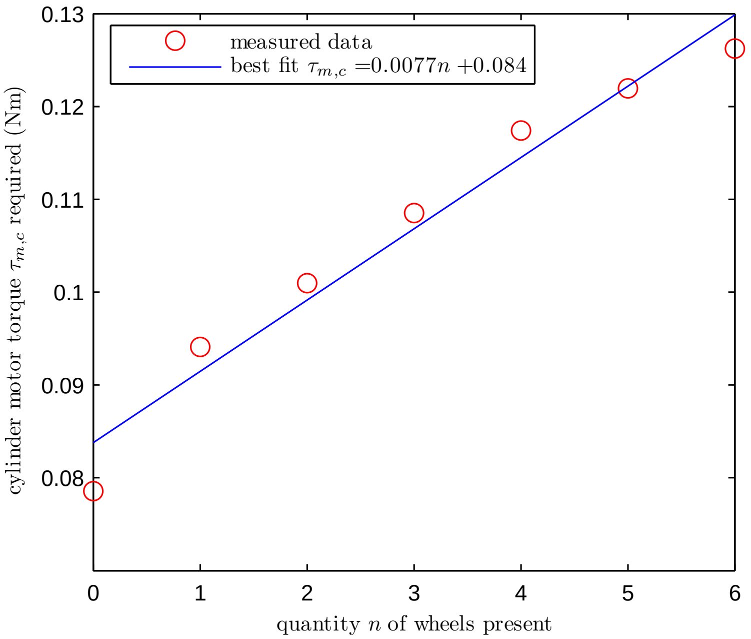


Figure 6.7. Determination of rolling friction due to inelastic dissipation at the contact patch, and friction in the CVT wheel axle bearings. This experiment is done at cylinder speed $\omega = 5$ rads/s. The vertical intercept indicates $\tau_{c,nominal}$, the cylinder torque required when no wheels are present. The slope indicates additional cylinder torque per wheel, $\frac{\tau_{c,wheel}}{n}$.

cannot isolate the two with this single experiment. However we can compare the power dissipation predictions of the analytical models for the two phenomena, presented in Sections 6.3 and 6.4. For the bearings of concern, and our estimate of experimental operating conditions (250 N, 5 rads/s), we obtain an estimated dissipation in the bearings of $\dot{W}_{wheelaxle} = 77$ mW, or twice the total measured value of $(\dot{W}_{wheelaxle} +$

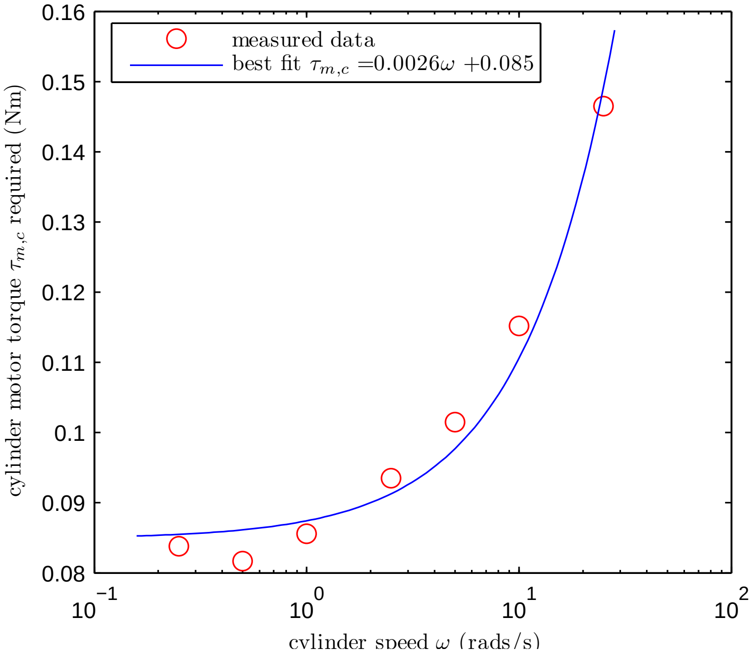


Figure 6.8. The cylinder motor torque, $\tau_{m,c}$, here a measure of $\tau_{c,wheel}+\tau_{c,nominal}$, is largely Coulombic with a very weak dependence on cylinder speed. This experiment is done with only one wheel present, steered to $\phi = 0$.

$\dot{W}_{inelastic}) = 38.5$ mW.[7] Reducing our estimated preload to 160 N as done in Section 6.4, we obtain an estimated dissipation of $\dot{W}_{wheelaxle} = 36.6$ mW, slightly less than the measured dissipation $(\dot{W}_{wheelaxle}+\dot{W}_{inelastic}) = 38.5$ mW. With $\alpha_{fr} = 0.01$, the power dissipated due to rolling friction is $\dot{W}_{inelastic} = 3.2$ mW as calculated by Equation 6.12 and the operating conditions (250 N, 5 rads/s). Johnson [87] suggests that $\alpha_{fr} \leq 0.01$, or in other words, metals stressed within the elastic limit are at least

[7] A combination of errors in parameters and the SKF model accuracy likely lead to the error, but we are not far off. We are much better than an order of magnitude.

99 percent efficient at rolling. Thus, the bearings are dissipating on the order of 10 times as much energy as the inelastic loss at the contact patch.

The second experiment (Figure 6.8) shows that the cylinder motor torque, $\tau_{m,c} = \tau_{c,wheel} + \tau_{c,nominal}$, due to $\dot{W}_{inelastic}$, $\dot{W}_{wheelaxle}$ and $\tau_{c,nominal}$, is only weakly dependent on cylinder speed, as is predicted by the models of $\dot{W}_{inelastic}$ and $\dot{W}_{wheelaxle}$. $\tau_{c,nominal}$ has not been modelled but we assume it will have a similar form as $\dot{W}_{wheelaxle}$.

6.6. Steady-state efficiency of rotational-to-linear reduction element

In this section we consider the efficiency of the rotational-to-linear transmission, by looking only at the losses of the rolling contact reduction element in the absence of steering action; i.e., losses of power due only to elastic, inelastic and CVT wheel axle bearing effects and not shearing losses. Linear guideway friction or inertial losses due to acceleration of loads are not considered either. We find an expression for the power efficiency of the rolling element at converting rotational power to linear power. We then experimentally validate this.

6.6.1. Theoretical efficiency

The total power loss of a rolling contact reduction element undergoing a constant force, constant velocity scenario is given by Equation 6.15.

$$\Delta\dot{W} = |\dot{W}_{elastic}| + |\dot{W}_{inelastic}| + |\dot{W}_{wheelaxle}| \tag{6.15}$$

The absolute values imply that all quantities are dissipative and make the following discussion more explicit. We are now in a position to predict efficiency as a function of power throughput. The total power throughput is given by Equation 6.16. Again, for the purposes of defining efficiency we take the absolute value of power flow through the transmission.

$$\dot{W} = f_w \dot{l} = |-f_w R\omega \tan(\phi)| \tag{6.16}$$

If we were to consider the efficiency of the whole cobotic system, and not just the reduction element, we would use f_l in Equation 6.16 for the total power throughput of the system. f_w, the force at the wheel, would still be used for the losses buried in $\dot{W}_{elastic}$. Theoretical efficiency, η_t, of the reduction element is given by Equation 6.17.

$$\eta_t = 1 - \frac{\Delta \dot{W}}{\dot{W}} = 1 - \frac{1}{C_{22}Gab}\left|\frac{\sec^2(\phi) f_w}{\sin(\phi)}\right| - \frac{\alpha_{fr} 2Pb}{3\pi r}\left|\frac{1}{\sin(\phi) f_w}\right| - \frac{1}{r}\left|\frac{\tau_{wheelaxle}}{\sin(\phi) f_w}\right| \tag{6.17}$$

Increasing f_w creates lateral creep efficiency losses but reduces the impact on efficiency of $\dot{W}_{inelastic}$ and $\dot{W}_{wheelaxle}$. Note that efficiency is independent of the magnitude of cylinder speed. Also note the singularity at $\phi = 0$, where efficiency must approach zero since no power would be flowing through the transmission but $\dot{W}_{inelastic}$ and $\dot{W}_{wheelaxle}$ would still be present.

6.7. Reduction element efficiency experiment

6.7.1. Purpose

To validate this theoretical model of reduction element efficiency by performing an experiment on a single joint of the Cobotic Hand Controller.

6.7.2. Methods

The experimental protocol to isolate the efficiency of the rolling contact reduction element is as follows: The steering plant fixes a transmission angle, ϕ, a priori. An effective reduction ratio, $\tan(\phi)$, between cylinder surface speed, $R\omega$, and joint speed, $\dot{l}$, is subsequently set. The efficiency of the rolling contact reduction element is defined as the mechanical power required to lift a known mass, m_k, via a pulley system, and to combat joint guideway friction (measured in Section 2.4.1) at constant velocity, divided by the mechanical power required to drive the reduction element. The total mechanical power entering the reduction element (including that which is dissipated and that which flows through) is $(\tau_c + \tau_{c,wheel})\omega$, where $\tau_c + \tau_{c,wheel} = \tau_{m,c} - \tau_{c,nominal}$, the commanded motor torque less the nominal torque required to make the cylinder spin in the absence of any wheels. Since the system is operating at a constant velocity, no mechanical power flows to the cylinder or carriage inertia and the steering plant is not in operation. The powers are integrated over time and computed as work done, which is useful since our linear and rotational sensors are position sensors. Thus the experimental efficiency of the reduction element, η_e, is computed by Equation 6.18.

$$\eta_e = \frac{(m_k g + c_{d,l} P \mathrm{sgn}(\dot{l})) \int \dot{l} dt}{(\tau_{m,c} - \tau_{c,nominal}) \int \omega dt} \tag{6.18}$$

The theoretical efficiency of the reduction element is computed via Equation 6.17.

6.7.3. Results

Figure 6.9 displays the theoretical predictions of efficiency and the experimental efficiency data for the Cobotic Hand Controller's rolling contact reduction elements. The experiment and theoretical model is applied to a large range of transmission ratios, $\tan(\phi)$. The theoretical values of Equation 6.17 are plotted for five values of the ratio of output force relative to preload force, $f_w/(\mu P)$, while the results of experimentation are provided at two values of this ratio. At 50 percent of peak output force, $f_w/(\mu P) = 0.5$, the transmission is 60 percent efficient even at a 90:1 gear ratio, and 98.5 percent efficient near a 1:1 gear ratio.

6.7.4. Discussion

The linear creep theory begins to fail, as it should, when lateral loading on the wheel, $f_a = f_w \sec(\phi)$ rises above 60 percent of the available friction force, μP. The experimental efficiencies fall away from the predicted values of the linear creep theory as gross-slip between the rolling elements is approached.

Experimental efficiencies are not reported for reduction ratios larger than $100:1$, since accurate measurements become a difficulty, although the device is capable of

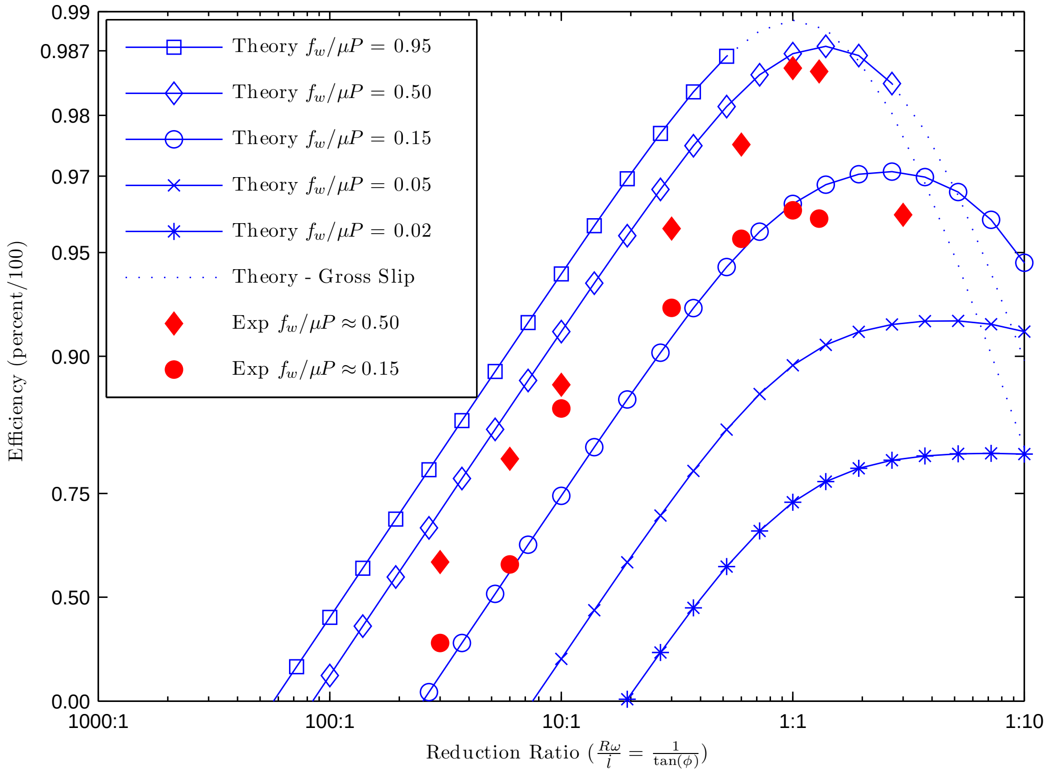


Figure 6.9. Theoretical and experimental efficiencies of the rolling contact reduction element of the cobotic transmission, operating at various transmission ratios and levels of maximum effort. The dotted lines indicate at what angle gross slip begins for a given $f_w/(\mu P)$.

rendering $\infty : 1$ ratios, or a completely clutched state. We can not collect data at a transmission ratio of $100 : 1$, since the steering angle can not be maintained accurately enough (or the encoder zeroed out precisely enough) which leads to inaccurate results. We are also unable to experiment with the highest possible loading condition, $f_w/(\mu P) = 0.95$, due to the unpredictable nature of friction near the breakaway force.

6.8. Slip-angles at zero power throughput

6.8.1. Theoretical slip angle

In this section we analyze the performance of the transmission when only the lateral elastic creep and not the inelastic losses were considered. Consider a joint commanded to have zero velocity, $\dot{l} = 0$, or equivalently, the transmission steered to $\phi = 0$. When a force f_w is then applied, the wheel creeps laterally, resulting in joint motion as if it were steered at *slip angle* ϕ_s.[8] The effective steering slip angle, ϕ_s (Equation 6.19), can be predicted by replacing $\dot{l}$ in $\phi = \operatorname{atan}(-\frac{\dot{l}}{R\omega})$ with the expression for lateral slip velocity, $u_a = \frac{f_w \sec(\phi)|r\dot{\theta}|}{C_{22}Gab}$, noting that $\sec(\phi) \approx 1$ and $r\dot{\theta} \approx R\omega$ for $\phi = 0$.

$$\phi_s = \operatorname{atan}\left(\frac{f_w}{C_{22}Gab}\right) \tag{6.19}$$

Note that the slip angle is independent of cylinder speed.

6.9. Slip angle experiment

6.9.1. Purpose

To experimentally isolate slip angles. The motivation was twofold. First we could compare the values we obtained with previous cobotic transmission analyses done with softer materials such as polyurethane Rollerblade^{TM} wheels. Second, we could develop an idea of how much steering correction would be needed to compensate for the lateral creep losses.

[8] Slip angles are introduced in the transmission analysis of Gillespie et al. [67].

6.9.2. Methods

Equation 6.19 can be solved for the theoretical value of the material and shape parameter, $(C_{22}Gab)_{thy} = \frac{f_w}{tan(\phi_s)}$. We determined the quantity, $(C_{22}Gab)_{exp}$, experimentally by measuring the force and slip angle for a single joint. The load cell was affixed to a single joint for these experiments. We commanded a feedback controller for the joint to provide zero output or $\dot{l} = 0$. As we gradually applied a force, f_l, by hand (while watching a readout), the joint feedback controller caused the transmission to steer into the force in order to combat the lateral creep. The force is then unloaded and applied in the opposite direction and, finally, brought back to zero. This full cycle takes about 30 seconds in order to approximate steady-state at each point. The force was filtered at 1 Hz and the joint friction force, $c_{d,l}P\text{sgn}(\dot{l})$, was removed to yield f_w. This was done for 5 different cylinder speeds with broken-in wheels with contact patch lateral half-width $a = 440\mu$m. Longitudinal half-width $b = 163\mu$m was obtained from the measured lateral half-width a via Figure D.3. Combined modulus of elasticity $G = 78$ GPa was given earlier in this chapter and creep coefficient $C_{22} = 2.87$ was obtained from tables in Kalker [89] or Johnson [87]. Thus $(C_{22}Gab)_{thy} = 1.6e + 004$ N.

6.9.3. Results

We used the *polyfit* command in Matlab to find the slope of the curves in Figure 6.10 between ±15 N, or about 50 percent of $f_w/(\mu P)$. This region of the creep curves is approximately linear and yields a value for $(C_{22}Gab)_{exp}$ for each cylinder speed.

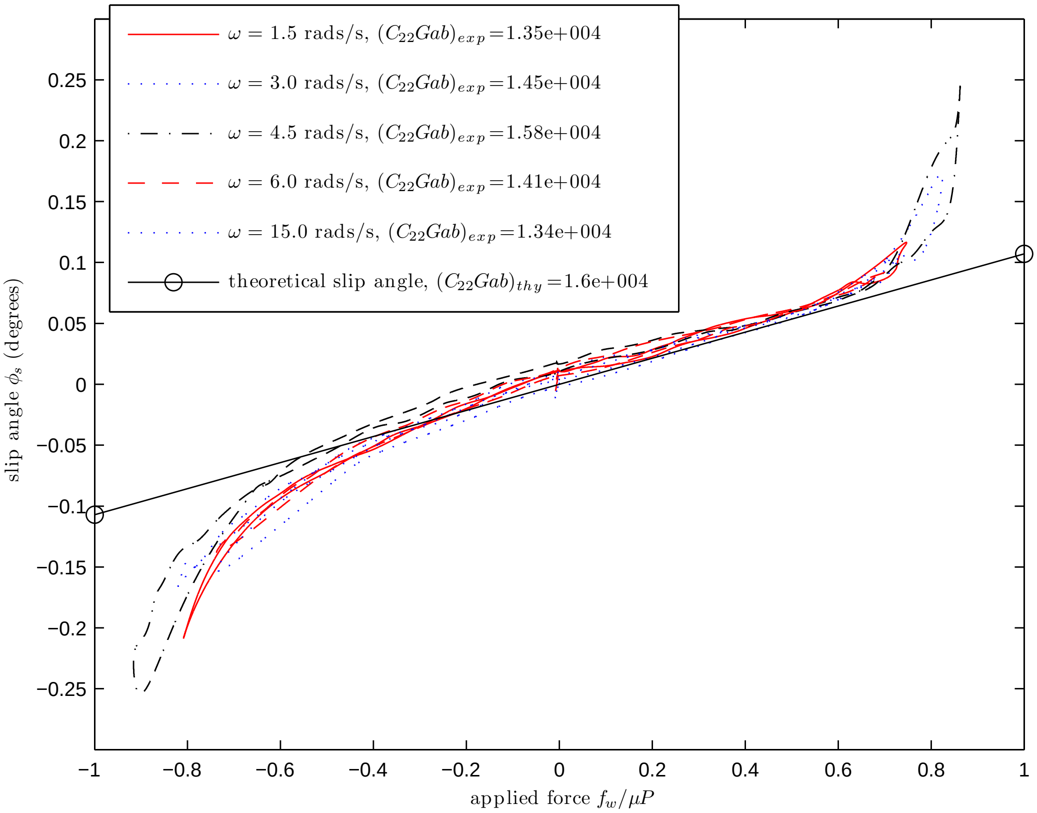


Figure 6.10. Here we compare the predicted and experiment slip angles, and values for the shape and material property factor $C_{22}Gab$. Experimental data was obtained after the wheels wore to an asymptotically stable half-width of about $a = 440\mu$m.

6.9.4. Discussion

The slip angles are only 0.2 degrees near $\frac{f_w}{\mu P} = 0.8$. This is much smaller than the as much as 6-10 degree angles seen with polyurethane Rollerblade^{TM} wheels [20, 21, 99]. By demonstrating that the slip angle of the steel-on-steel rotational-to-linear transmission is small, we prove that it is acceptable to use the CVT angle and cylinder speed to estimate linear velocity, as was done in the *linear velocity estimate* block of Figure 3.4.

The data in Figure 6.10 also indicates that the slip angle is independent of cylinder speed as is predicted by Equation 6.19.

Our experimentally determined shape and material factor, $(C_{22}Gab)_{exp}$, is about twenty percent lower than the predicted values. This was expected as vibrations and lubrication tend to increase the slip, and, therefore, reduce the experimental creep coefficient C_{22}. Some of the error may also be due to our approximation of the shape and dimensions of the contact patch.

Alternatively, we could have predicted the creep angles by measuring the change in applied torque to the cylinder as load is applied to the joint. This would have avoided the need to compensate for joint friction. However, our knowledge of the cylinder torque is not as accurate as the load cell's measurement of joint force.

The analysis contained within this chapter has been much improved in my ASME JMD publication.

CHAPTER 7

Focus 3b: Bond Graphs, Power Efficiency and Power Consumption

Having analyzed the rolling contact reduction element, we are now in a position to model and analyze the complete rotational-to-linear cobotic system. In this chapter we first develop bond graphs that describe all the power flows, both steady-state and dynamic, in the cobotic system. We then develop a bond graph for a hypothetical *conventional* system which uses a fixed gear ratio and a rotational electric motor. In Section 7.3 we provide a comparison of the efficiency of transmissions: the rolling contact reduction element as opposed to gears. In Section 7.4 we provide a comparison of the power efficiency of a cobotic system with that of a conventional system for constant power throughput.[1] In Section 7.5 we provide a comparison of the power efficiency of the two systems for sinusoidal power throughput across a range of frequencies. Finally, in Section 7.6 we evaluate the total Wattage dissipated in the Cobotic Hand Controller.

[1] Note the delineation between *transmission* and *system*. By *transmission* we mean just a rolling contact reduction element, or a pair of gears. By *system* we mean all components of an architecture, including motors, guideways and transmissions.

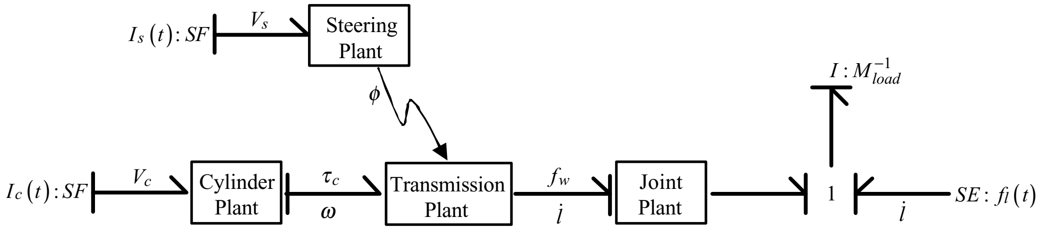


Figure 7.1. A general cobotic system is composed of four plants detailed in Figures 7.2 through 7.5. The plants allow two current sources and a human effort source to do work on a mass, M_{load}.

7.1. Cobotic system

Figure 7.1 depicts a bond graph describing a cobotic system consisting of a rotational-to-linear transmission. The top portion contains the steering plant, while the bottom portion contains the cylinder, transmission and joint plants. No controller elements are shown on this graph, only electrical and physical elements. The inputs to the bond graph are steering and cylinder motor currents I_s and I_c along with interaction force f_l. The output is the resulting motion of M_{load}. M_{load} is in addition to the nominal joint mass, m_l, depicted in the joint plant of Figure 7.5. The steering plant adjusts the modulus of the transformer in the transmission plant, but no power flows between the steering plant and transmission. Table 7.1 introduces any new notation used in Figures 7.1 through 7.5.

Moving from left to right in the steering plant (Figure 7.2), we see electrical current, I_s, driving gyrator, K_s, and Ohmic heating in motor resistance, R_s. After the gyrator creates rotational power, $\tau_{m,s}\dot{\tilde{\phi}}$, effort flows into the inertia of the steering

Table 7.1. Cobotic plant bond graph notation.

Variable	**Definition**
V_s	steering motor voltage
I_s	steering motor current
R_s	steering motor resistance
K_s	steering motor constant
n_s	gear ratio between steering motor and bell
$J_{m,s}$	inertia of steering motor
$c_{d,b}$	coefficient of steering bell bearing friction
μ_s	coefficient of contact patch steering (sliding) friction
$\dot{\tilde{\phi}}$	angular velocity of steering motor
$k_{gearing}$, $b_{gearing}$	stiffness and damping of steering plant transmission
V_c	cylinder motor voltage
I_c	cylinder motor current
R_c	cylinder motor resistance
K_c	cylinder motor constant
n_c	gear ratio between cylinder motor and cylinder
M_{load}	inertial load in addition to joint mass m_l
$\dot{W}_{linearguide}$	dissipation of linear guide friction

motor, $J_{m,s}$, and flow to the gear-train impedance described by stiffness, $k_{gearing}$, and damping, $b_{gearing}$. We will ignore this gear-train impedance for the remainder of Chapter 7 as it does not consume significant power. After this we have the actual transformer, $-n_s$, of the gearing followed by the final one-junction, driving the steering bell inertia, J_b, the dissipation of the bell bearing, $\dot{W}_{bell}$, and the dissipation of the contact patch, $\dot{W}_{shear}$.

The power dissipated due to shear at the contact patch is $\dot{W}_{shear} = \text{abs}(\dot{\phi})\frac{a\mu_s P}{2}$, if we assume a relatively uniform pressure distribution along a line contact. μ_s is a dynamic sliding coefficient of friction, different from the rolling coefficient of friction, μ. Assuming $\mu_s \approx \mu$, the required torque to combat this shear is $\frac{a\mu_s P}{2} = 0.0067$ Nm. We cannot experimentally isolate $\dot{W}_{shear}$ from $\dot{W}_{bell}$, but the sum of the two is measured to be 0.0167 Nm for a large range of $\dot{\phi}$. Removing the torque due to $\dot{W}_{shear}$ from this, the remaining 0.01 Nm must be due to the bell bearing dissipation, $\dot{W}_{bell}$. If we assume a Coulombic model for the bell bearing friction, $c_{d,b}P\text{sgn}(\dot{\phi})$, a dynamic coefficient of Coulombic bell friction $c_{d,b} = 40\mu\text{m}$ is found. We have given this coefficient units of distance since it relates a preload force to a torque.

Moving from left to right in the cylinder plant (Figure 7.3), we see electrical current, I_c, driving gyrator K_c and Ohmic heating in motor resistance, R_c. After the gyrator creates rotational power $-n_c\tau_{m,c}\omega$, power flows to the inertia of the cylinder motor, $J_{m,c}$, to the transformer, $-n_c$ (gearing between the cylinder motor and cylinder), and to any dissipation of this transformer and cylinder bearings, $\dot{W}_{nominal}$. The output of this transformer drives the cylinder inertia, J_c, and provides torque, τ_c, to the transmission plant at cylinder velocity, ω. In the current implementation of the Cobotic Hand Controller, $|n_c| = 1$. In general, $|n_c| \gg 1$ (on the order of 50) would be present to allow the cylinder to operate at a more efficient speed, and

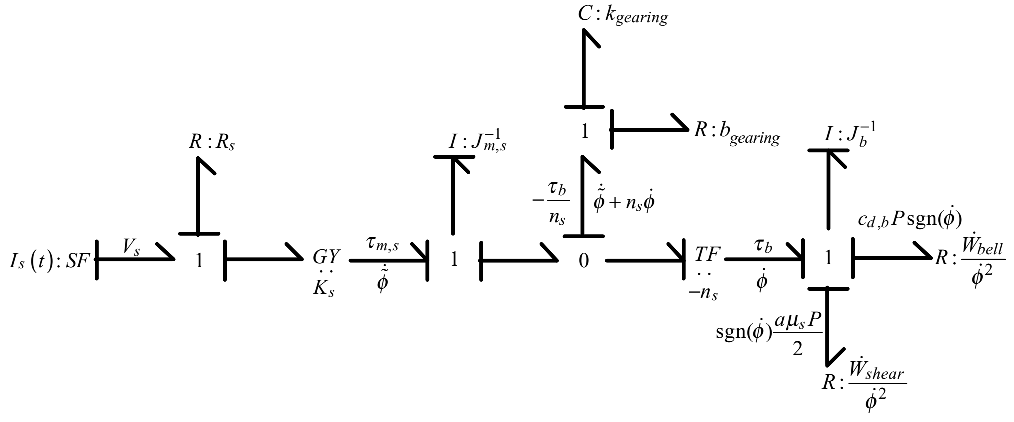

$C : k_{gearing}$
1
$R : b_{gearing}$
$R : R_s$
$I : J_{m,s}^{-1}$
$I : J_b^{-1}$
$-\frac{\tau_b}{n_s}$
$\dot{\hat{\phi}} + n_s\dot{\phi}$
$c_{d,b}P\,\mathrm{sgn}(\dot{\phi})$
$I_s(t) : SF$
V_s
1
GY
K_s
$\tau_{m,s}$
$\dot{\hat{\phi}}$
1
0
TF
$-n_s$
τ_b
$\dot{\phi}$
1
$R : \frac{\dot{W}_{bell}}{\dot{\phi}^2}$
$\mathrm{sgn}(\dot{\phi})\frac{a\mu_s P}{2}$
$R : \frac{\dot{W}_{shear}}{\dot{\phi}^2}$

Figure 7.2. Steering plant portion of the cobotic transmission system.

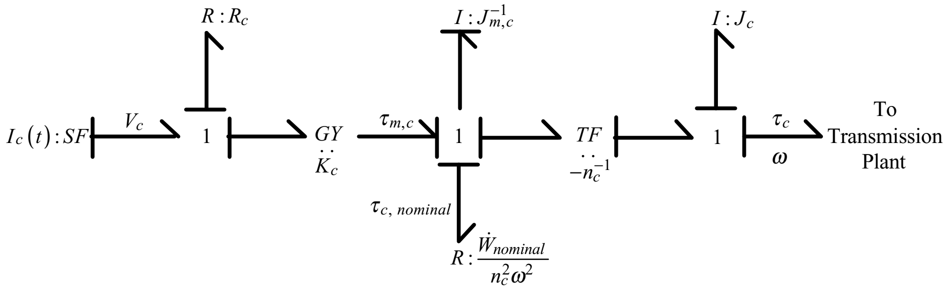


Figure 7.3. Cylinder plant portion of the cobotic transmission system.

significant losses due to $\dot{W}_{nominal}$ would be incurred. Recall that a large 1400 watt motor was chosen as it was readily available in the lab and had sufficient torque to operate without gearing. We do not model any compliance in the cylinder gear-train. Although this compliance could significantly impact controller dynamics, it will not dissipate significant energy so as to affect the present analysis of efficiency and power consumption.

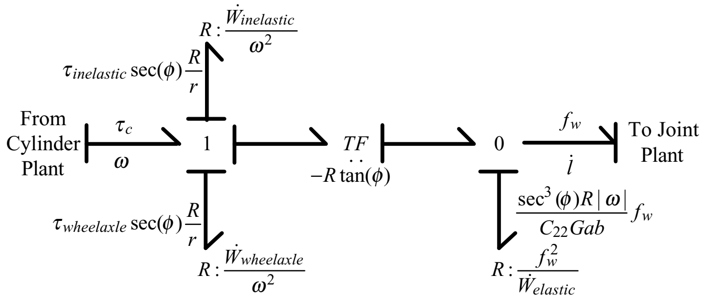

$R:\frac{\dot{W}_{inelastic}}{\omega^2}$
$\tau_{inelastic}\sec(\phi)\frac{R}{r}$
From Cylinder Plant
τ_c
ω
1
TF
$-R\tan(\phi)$
0
f_w
$\dot{l}$
To Joint Plant
$\frac{\sec^3(\phi)R|\omega|}{C_{22}Gab}f_w$
$R:\frac{f_w^2}{\dot{W}_{elastic}}$
$\tau_{wheelaxle}\sec(\phi)\frac{R}{r}$
$R:\frac{\dot{W}_{wheelaxle}}{\omega^2}$

Figure 7.4. Transmission plant portion of the cobotic transmission system.

The rolling contact reduction element is characterized in Figure 7.4. Power is lost to rolling-friction losses at the contact patch, $\dot{W}_{inelastic}$, and to the CVT wheel axle bearing losses, $\dot{W}_{wheelaxle}$, from the common flow junction. Following these losses is the transformer between rotational and linear motion, $-R\tan(\phi)$. This is followed by a common effort junction at which power, $\dot{W}_{elastic}$, is lost to elastic creep. The remaining power, $f_w\dot{l}$ flows to the joint plant.

Translational power $f_w\dot{l}$ flows into the joint plant depicted in Figure 7.5. This plant consists of a common flow junction that distributes power to the output loads, the joint inertia, m_l, and to the joint guideway friction, $\dot{W}_{linearguide}$.

7.2. Conventional system

In order to further the analysis of the rotational-to-linear cobotic transmission, we compare it to a hypothetical *conventional* electromechanical plant (see Figure 7.6). We define a conventional system as a rotational electrical motor coupled through a fixed-ratio gear-train, to a pulley or capstan drive, in order to impose a straight

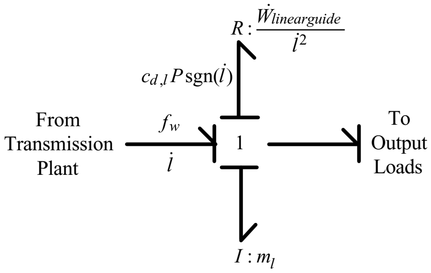

$R : \frac{\dot{W}_{linearguide}}{\dot{l}^2}$
$c_{d,l} P \operatorname{sgn}(\dot{l})$
From Transmission Plant
f_w
$\dot{l}$
1
To Output Loads
$I : m_l$

Figure 7.5. Joint plant portion of the cobotic transmission system.

Table 7.2. Conventional plant bond graph notation.

Variable	**Definition**
V_n	conventional motor voltage
I_n	conventional motor current
R_n	conventional motor resistance
K_n	conventional motor constant
n_n	gear ratio of conventional drive-train
$J_{convmotor}$	inertia of conventional system motor
$J_{gear,pulley,cable}$	inertia of the rest of the conventional system
τ_{gpc}	effort loss due to friction in conventional system elements
$\dot{W}_{gpc}$	power loss due to friction in conventional system elements
r_{pulley}	radius of pulley converting rotation to linear motion

line velocity on M_{load}. The conventional drive-train requires only a single motor to generate the same output as the rotational-to-linear cobotic system. Table 7.2 defines any new notation introduced in Figure 7.6.

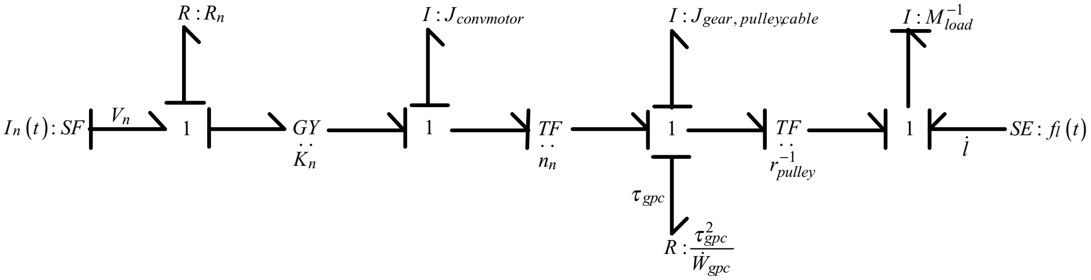


Figure 7.6. A conventional fixed-ratio rotary-electric to linear system.

Moving from left to right in the conventional system bond graph (Figure 7.6), the junctions represent power flow to the electrical resistance of the motor, the conversion of electrical power to mechanical power, losses due to inertia of the motor, a single stage gear reduction, n_n, friction and inertia due to gears, pulleys and cables in the system and, finally, the pulley radius transformer, r_{pulley}, that converts rotation to translation.

Figure 7.7 compares the cobotic and conventional systems. The output for each drive-train output is the flow, $\dot{l}$, of the mass, M_{load}. The conventional system requires electrical power, $I_n V_n$, while the cobotic system requires steering power, $I_s V_s$, and cylinder power, $I_c V_c$. Both systems allow an external force, f_l, to apply effort to the load.

7.3. Rolling-contact reduction element versus gears

We first compare the efficiency of the two transmissions before we compare the two complete systems. Figure 6.9 displays the theoretical predictions of efficiency and

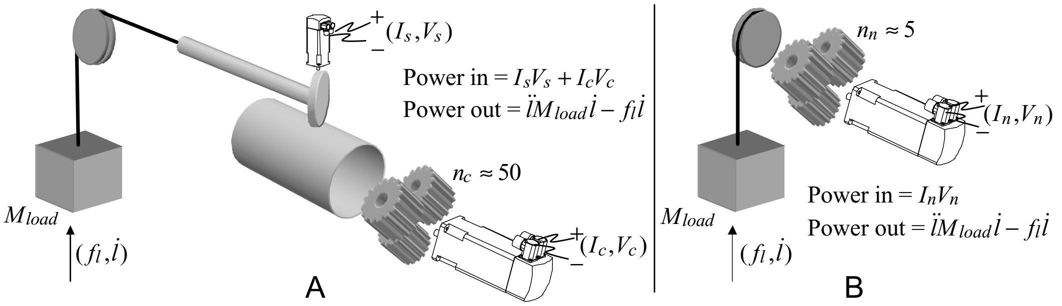


Figure 7.7. A. Cobotic architecture. B. Conventional architecture.

experimental efficiency data from the Cobotic Hand Controller for the rolling contact reduction element. We compare these efficiencies to other types of gear trains in Figure 7.8. Planetary gear trains used for low torque applications, harmonic drives, and worm gears, have similar efficiencies as the cobotic rolling contact reduction element at ratios of 50:1 or smaller. However, while we report cobotic efficiencies at 15, 50 and 95 percent of peak power throughput, the efficiencies for gears are conventionally reported at peak continuous power throughput where friction losses are smallest relative to the power throughput. Thus, in Figure 7.8, the cobotic reduction element may seem less inefficient at low power throughput relative to gears than it really is.

7.4. Analysis of static system efficiencies

In order to develop a fair comparison between the power efficiency of conventional and cobotic systems, we first define a common set of design goals. These goals are

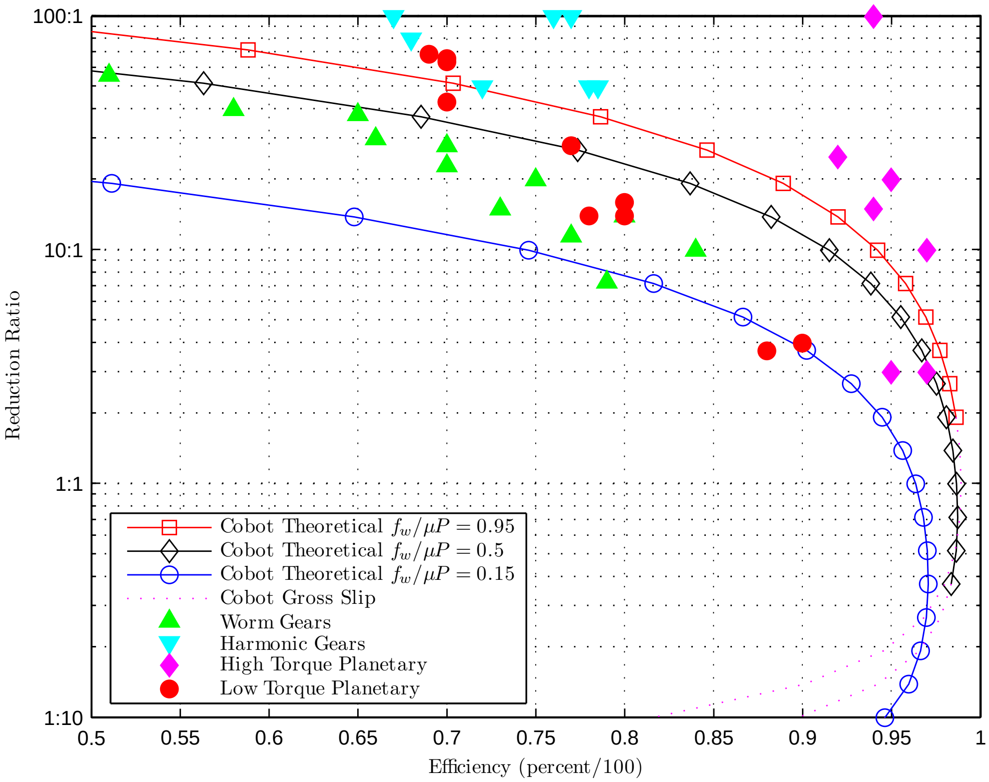


Figure 7.8. Efficiency of the rotational-to-linear rolling contact reduction element versus conventional gears sampled at random from the Internet. Only very expensive, very high torque planetary gear trains (not useful for prosthetics or lightweight applications), or single gear pairs can achieve efficiencies above 90 percent. The cobotic efficiencies are reported for various percentages of maximum power throughput $(\frac{f_w}{\mu P})$ at a given ratio. The cobotic efficiencies fall off steeply at large reduction ratios as rolling friction losses become large relative to power throughput.

a desired operating regime in the force, f_l, versus velocity, $\dot{l}$, plane. A designer of a linear actuation system will likely specify a maximum force required, a maximum velocity required and also a maximum power that is expected at any given time. A boundary in the force-velocity plane is then developed from these three specifications as illustrated in Figures 7.9 and 7.10. Given the desired operating regime for static power flows (constant velocity and constant force), conventional and cobotic systems are designed and evaluated in Sections 7.4.1 and 7.4.2 respectively.

7.4.1. Analysis of conventional system static efficiency

7.4.1.1. Purpose. To evaluate the efficiency and legal operating regime in the force-velocity plane for several conventional power-train designs of the framework outlined in Figure 7.6. The goal of this analysis is to determine the smallest motor with which the conventional architecture can meet the maximum force, maximum velocity and maximum power specifications.

7.4.1.2. Methods. A series of hypothetical systems, with the architecture outlined in Figure 7.6, are designed to meet these specifications. A Matlab simulation evaluates their legal operating regimes and power efficiencies. Efficiency is evaluated as mechanical power out divided by electrical power in: $|\frac{\ddot{l}M_{load}\dot{l}-f_l\dot{l}}{I_nV_n}|$. This simplifies to $|\frac{-f_l\dot{l}}{I_nV_n}|$, given that $\ddot{l}=0$ for a static power flow scenario. Losses include Ohmic heating of the motor windings and friction of the gear-train, pulley and cable. No power flows into inertial components when operating at constant force and constant velocity.

The legal operating regime is limited by the maximum continuous torque of the motor and the maximum velocity of the motor. Motor 1 has the properties of a Maxon^{TM} Re-max 29 brushed DC motor (22 watt, num. 226806)[2] and Motor 2 is a

[2] Although all motors in the Cobotic Hand Controller are brushless DC, we choose brushed DC here for ease of modelling.

version of this scaled up in strength (and weight and power consumption, etc.) by a factor of 2.7. Both have peak efficiencies of 87 percent. The four gearing options are all single stage planetary gear-heads with efficiency of 97 percent.

7.4.1.3. Results. In Figure 7.9 we show the performance capabilities of various conventional electro-mechanical drive-train designs (Motor 1 with Gearing 1, Motor 1 with Gearing 2, Motor 1 with Gearing 3 and Motor 2 with Gearing 4).

7.4.1.4. Discussion. Although Motor 1 is paired with several different gear ratios, it cannot achieve the maximum force and maximum velocity specifications simultaneously. It has no trouble developing the required power specification, but cannot do so across the range of operating conditions. The right hand boundary of the continuous operating regimes is limited by the continuous torque that a motor can develop without overheating.[3] In order to achieve the maximum force and velocity specifications for a single gear ratio, a much larger Motor 2 must be selected, which has more power capability than will ever be needed. Given that Motor 2's power capability is larger than needed, it never operates at maximum power, and therefore does not operate at high efficiency. Much of the electrical power is lost to resistive heating of the motor windings as it operates at inefficient speeds. Although the combination of Motor 2 and the Gearing 4 is capable of 85 percent power efficiency, it never exceeds 65 percent efficiency in our desired operating regime (note the efficiency contours in Figure 7.9).

[3] The sloping upper boundary is the maximum velocity that a motor can be driven at, given the operating voltage and applied torque. This sloping boundary would intersect the horizontal axis at the momentary stall torque achievable by the motor.

The analysis contained within this chapter has been much improved in my ASME JMD publication.

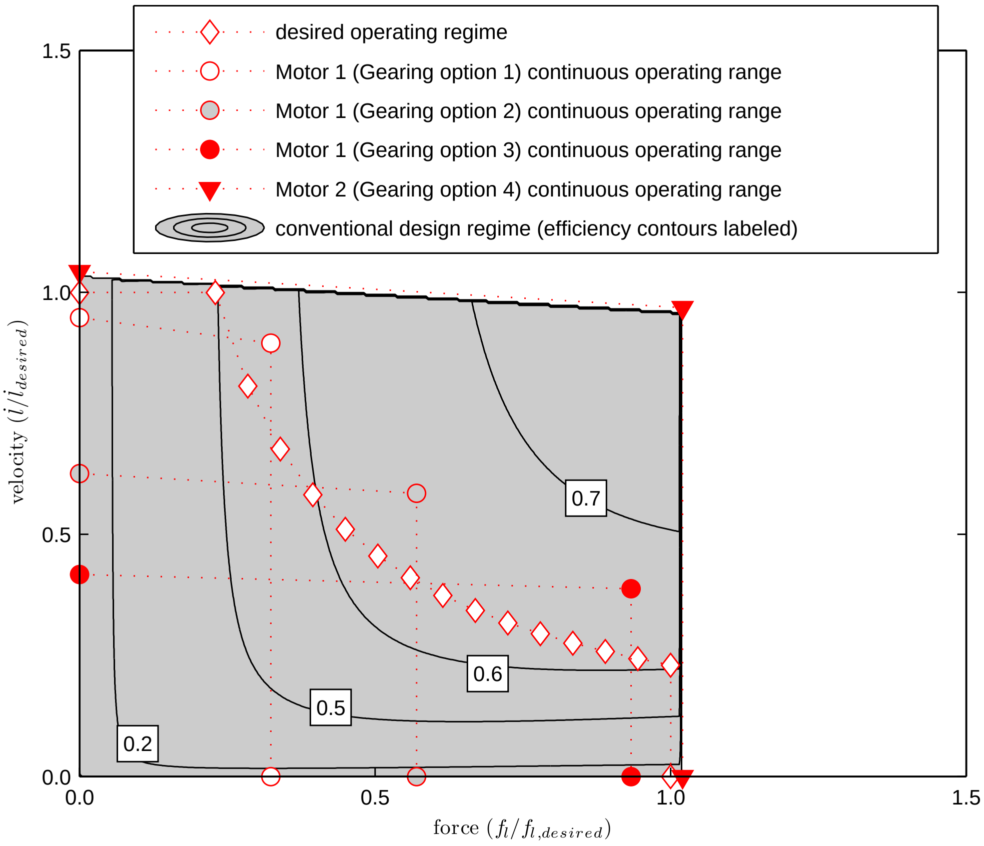


Figure 7.9. The ranges of operation of some conventional drive-train designs, and contours of power efficiency of the Motor 2 design. $\dot{l}_{desired}$ and $f_{l,desired}$ are the desired maximum velocity and force specifications. Motor 2 is 2.7 times larger than Motor 1.

7.4.2. Analysis of cobotic system static efficiency

7.4.2.1. Purpose. We evaluate the efficiency and legal operating regime in the force-velocity plane of a single leg of the Cobotic Hand Controller system as outlined in Figure 7.1.

7.4.2.2. Methods. In a static analysis of the cobotic drive-train, we are concerned with the efficiency of the transmission plant in the absence of any steering action. Thus a steering angle has been set and no electrical flow is required in order to maintain the angle of the CVT. Efficiency is evaluated as mechanical power out divided by electrical power in: $|\frac{\ddot{l}M_{load}\dot{l}-f_l\dot{l}}{I_sV_s+I_cV_c}|$. This simplifies to $|\frac{-f_l\dot{l}}{I_cV_c}|$, given that $\ddot{l}=0$ and $\dot{\phi}=0$ for a static power flow scenario. Losses consist of elastic creeps and inelastic rolling friction, Ohmic heating and frictional losses of the cylinder motor, frictional losses of the gear-set driving the cylinder, and friction of the linear guideway. No power flows into the inertias of the system since we are holding the velocity constant. Likewise, no power flows to the steering plant since the angle is fixed for this scenario.

The legal bounds of operation in the force velocity range are also computed. They are limited both by maximum cylinder motor torque and available friction force (preload). They are not limited by cylinder motor speed, since we use the fixed-speed cylinder controller set to $\omega = 14$ rads/s, and the cylinder motor is operating at its most efficient speed of 8040 RPM. This high operating speed of the cylinder motor is made possible by modifying the Cobotic Hand Controller to incorporate a gear ratio of $n_c = 57:1$, and thus we replace the 1400 Watt motor in our implementation with the 22 Watt *Motor 1* (also used in the hypothetical conventional system of Section 7.4.1). The gear-train, n_c, has a maximum efficiency of 80 percent, possible for a high end harmonic drive, or a high-torque planetary gear-set.

7.4.2.3. Results. A Matlab simulation evaluates the legal operating regime and power efficiency. Figure 7.10 displays static efficiency predictions of the cobotic system

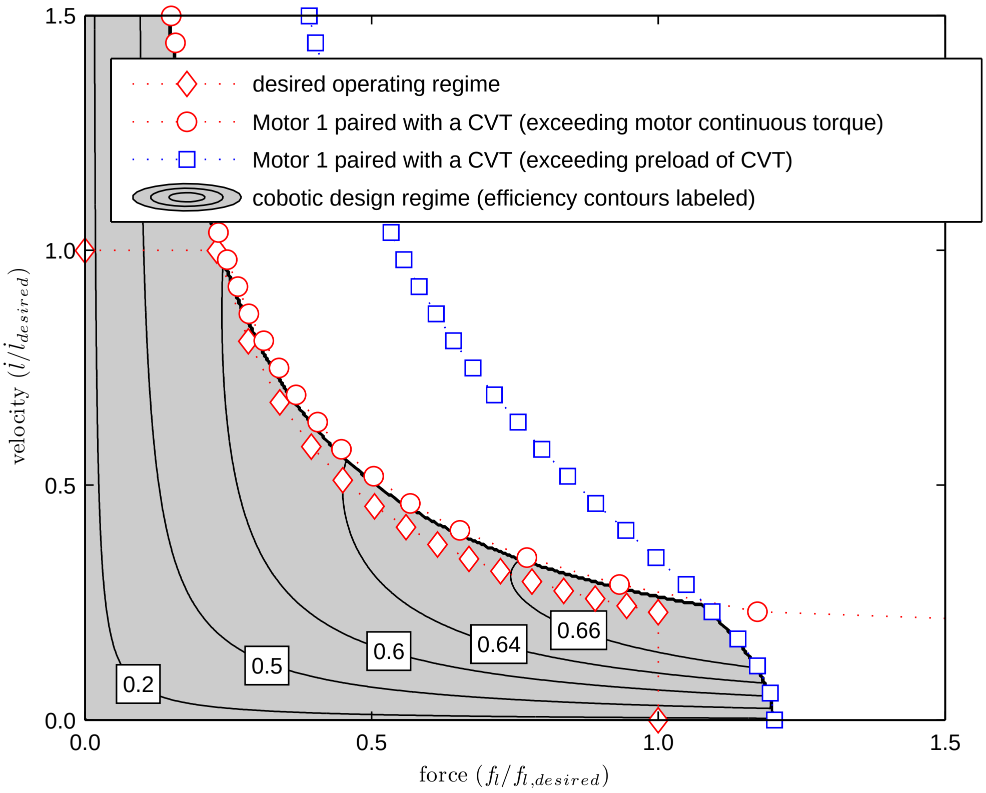


Figure 7.10. The range of operation of a cobotic drive-train using the same Motor 1 as the conventional drive-train design in Figure 7.9. The legal cobotic design regime is bounded where the required output force would cause the CVT wheel to slip, and where the required motor torque exceeds the motor's thermal limit. $\dot{l}_{desired}$ and $f_{l,desired}$ are the desired maximum velocity and force specifications.

defined in Figure 7.1. Note the much higher power efficiencies of the cobotic versus the conventional drive-train at high forces and low velocities. The velocity limit of Motor 1 is not an issue for the cobotic system since it is operating constantly at its most efficient speed.

7.4.2.4. Discussion. The smaller Motor 1, although insufficient to meet our performance criteria in the conventional drive-train, is sufficient in the cobotic drive-train design. The cobotic drive-train's continuously variable transmission allows the capture of maximum power output of Motor 1 across all output forces and velocities. In addition, higher power efficiencies are reached by the cobotic drive-train than the conventional drive-train at any given point, since the cobotic drive-train's motor is always operating at an efficient speed. Nevertheless, the significant gear reduction, which we have hypothetically placed between the cylinder motor and the cylinder limits the cobotic drive-train's efficiency to 70 percent (80 percent gearing efficiency and 87 percent efficient motor). We could have used a high-torque gear-set with higher efficiency (greater than 97 percent), that would be better suited to our huge cylinder motor, however a fairer comparison could be made to the conventional power-trains by including a low-torque gear-set with poorer efficiency.

7.5. Analysis of dynamic system efficiencies

7.5.1. Purpose

We evaluate the efficiency and legal operating regimes of the cobotic and conventional systems for the task of shaking mass M_{load} at various frequencies and amplitudes. This scenario will require additional power to steer the wheel or modulate the transmission of the cobotic system, and to accelerate and decelerate inertias in each system.

7.5.2. Methods

The space of dynamic operating conditions will be characterized by frequency, ϖ, and by a percentage of maximum power throughput. The maximum power throughput is the static maximum power specification from Figures 7.9 and 7.10. We now consider this continuous power an RMS power, $\dot{W}_{RMS}$. We evaluate the amplitude of $\dot{l}$, α, at which M_{load} must be shaken in order to achieve this power. An amplitude of motion, $\alpha = \sqrt{\frac{2\sqrt{2}\dot{W}_{rms}}{\varpi^3 M_{load}}}$ m/s, is computed. The RMS power is evaluated from instantaneous power, $\dot{W}(t) = \ddot{l}(t) M_{load} \dot{l}(t) = M_{load} \alpha^2 \varpi \sin(\varpi t) \cos(\varpi t)$. Each system is worked through one cycle of this motion and the desired output power is divided by the electrical power requirements to yield the dynamic power efficiency of the systems. Efficiency will now be computed as $\int |\frac{\ddot{l} M_{load} \dot{l}}{I_n V_n}| dt$ for the conventional system and as $\int |\frac{\ddot{l} M_{load} \dot{l}}{I_s V_s + I_c V_c}| dt$ for the cobotic system. No external forces f_l are present for this scenario. We assume that the cylinder is driven at constant speed. Again, Matlab simulations for the systems depicted in 7.1 and 7.6 are used to evaluate the legal operating regimes and power efficiencies.

7.5.3. Results

Figure 7.11 depicts a comparison of the conventional and cobotic system efficiencies across a range of frequencies and across a range of fraction of maximum specified power throughput.

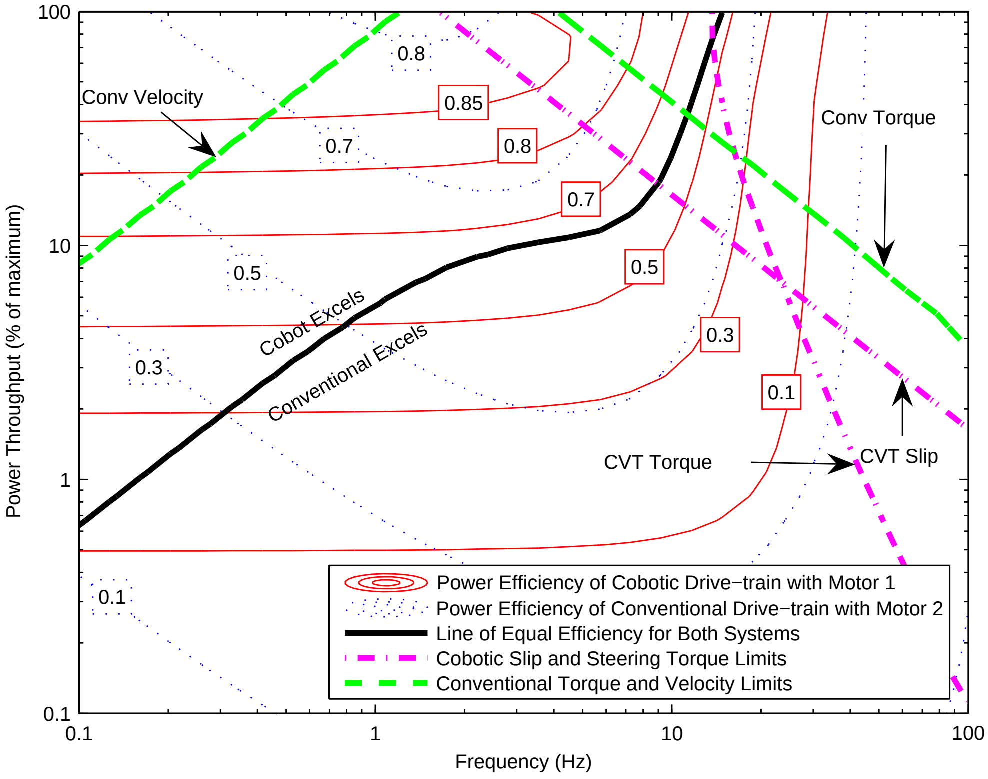


Figure 7.11. Comparison of the power efficiency contours of cobotic drive-trains and conventional drive-trains at driving a mass sinusoidally. Neither system can work in the upper right portion of the plot and only the cobotic system can work in the upper left portion.

7.5.4. Discussion

Both drive-trains can achieve higher efficiencies in the dynamic case than the static loading scenario. Frictional losses do not detract from dynamic efficiency since friction helps the system decelerate the load in the same manner the acceleration phase is hindered. Therefore, although the cobotic system must modulate a steering plant,

The analysis contained within this chapter has been much improved in my ASME JMD publication.

and has an 80 percent efficient gear-set between the cylinder motor and cylinder, it is able to achieve 85 percent efficiencies for sinusoidal motion.

The analysis here compares our best attempts at designing two different architecture systems to meet the same specifications. While the results are certainly not quantitatively precise, they have important qualitative implications. In general, Figure 7.11 portrays the cobotic drive-train as having higher power efficiency than the conventional drive-train at less than 10 Hertz and greater than 10 percent of the maximum power throughput. In this regime, the cobot is losing the majority of its power to the steering plant, and the conventional drive-train is losing most of its power to electrical resistance. At the mid-range frequencies of voluntary human motion (one to ten hertz), the two drive-train types have relatively similar power efficiencies, even with the additional expenditure of modulating the steering angle by the cobot. Both systems show increasing efficiency with increasing power throughput since many of their losses are due to Coulombic friction or Ohmic heating of motor windings. Both systems also exhibit decreasing efficiency at high frequencies since, in addition to accelerating and decelerating the load, inertias in the drive-trains must also be accelerated and decelerated.

There is much room for improvement of the dynamic efficiency in our current cobotic design. This can be accomplished by reducing the rotational inertia of the steering plant, and by reducing the mass of the linearly moving joint plant.

The analysis contained within this chapter has been much improved in my ASME JMD publication.

7.6. Analysis of relative significance of cobot system losses

7.6.1. Purpose

To discern the relative contributions of individual components to power dissipation.

7.6.2. Methods

Tables 7.3 and 7.4 utilize parameter values, as specified in Table 7.5, and operating conditions, as specified in Table 7.6, to compute their example power losses. The parameter values are all representative of the actual Cobotic Hand Controller. The operating conditions represent a moderately aggressive scenario of a user manipulating the end-effector in a mass-damper virtual environment with a energy-tracking cylinder controller.

7.6.3. Results

Table 7.3 summarizes methods by which power is dissipated in the reduction element, both at the contact patch and in the CVT wheel axle bearings. These methods include lateral creep, rolling friction, steering friction and the CVT wheel axle bearings. Table 7.4 summarizes steering bell, carriage and cylinder losses. These losses include bell friction, bell inertia, linear guideway friction, carriage inertia, cylinder friction and cylinder inertia. An expression for the *power loss* of each method is provided along with an expression for the operating *region* in which the effect is relevant. Expressions are also provided for the *material* property ratio and the *geometry* ratio that should be reduced in order to minimize power consumption. The *contact patch geometry* model, upon which the effect is based, is also indicated. The material property and geometry ratios may contain variables that are not present in the power loss equations since simplifications and substitutions have been made.

7.6.4. Discussion

The device has 1257 mW of dissipation per joint in this scenario: 81 mW at the contact patch, 41 mW in the CVT wheel axle bearings, 53 mW from bell inertia and friction, 618 mW from joint inertia and friction and 464 mW of cylinder inertia and friction per joint.[4] This yields 7.54 Watts for all six joints. Not included in these tables are motor winding losses and Ohmic heating.

[4] As indicated in Table 7.4, the total cylinder inertia and friction terms were divided by 6 to compare properly on a per joint basis.

Table 7.3. Reduction element design chart.[a]

	Lateral Creep	Inelastic Friction	Wheel Bearing Friction	Steering Friction
contact patch geometry	ellipse	rectangular	NA	line
power loss	$\frac{f_w^2 \sec^3(\phi) R \lvert\omega\rvert}{C_{22} G a b}$	$\text{abs}(\omega) \frac{\alpha_{fr} 2 P b}{3\pi} \sec(\phi) \frac{R}{r}$	$\tau_{wheelaxle} \sec(\phi) \frac{R}{r} \omega$	$\text{abs}(\dot{\phi}) \frac{a \mu_s P}{2}$
example	44 mW	4 mW	41 mW	33 mW
region	$f_w^2 \sec^3(\phi) \omega$	$\sec(\phi)\omega$	$\sec(\phi)\omega$	$\dot{\phi}$
material	$\frac{\mu}{G}$	$\frac{\alpha_{fr}}{\mu}$	$\frac{\tau_{wheelaxle}}{\mu}$	$\frac{\mu_s}{\mu}$
geometry	$\frac{b}{a}$	$\frac{b}{a}$	r_a	a

[a] All power losses are specified for a single reduction element. Bearing friction $\tau_{wheelaxle}$ is computed from [155], and is largely a function of P and r_a (r_a is the radius of the bearing). Here we have included contact patch steering friction as a loss of the reduction element, while in the $\dot{\phi} = 0$ analysis of Chapter 6 we neglected this.

Table 7.4. Bell, carriage and cylinder design chart.[a]

	Bell Friction	Bell Inertia	Joint Friction	Joint Inertia	Cylinder Friction	Cylinder Inertia
power loss	$c_{d,b} P \text{sgn}(\dot{\phi}) \dot{\phi}$	$J_b \ddot{\phi} \dot{\phi}$	$c_{d,l} P \text{sgn}(\dot{l}) \dot{l}$	$m_l \ddot{l} \dot{l}$	$\frac{1}{6} \tau_{c,nominal} \text{sgn}(\omega) \omega$	$\frac{1}{6} J_c \dot{\omega} \omega$
example	50 mW	3 mW	168 mW	450 mW	70 mW	394 mW
region	$\dot{\phi}$	$\ddot{\phi}\dot{\phi}$	$\dot{l}$	$\ddot{l}\dot{l}$	ω	$\dot{\omega}\omega$
material	$\frac{1}{\mu}$	$\frac{1}{\rho}$	$\frac{1}{\mu}$	$\frac{1}{\rho}$	$\tau_{c,nominal}$	$\frac{1}{\rho}$
geometry	r_b	r_b^2	NA	m_l	NA	R^2

[a] All power losses are specified for a single transmission. The cylinder terms have been divided by 6 for comparison with the joint and bell terms. Here ρ is the specific strength of the materials, and r_b is a length scale factor for the bell about the steering axis.

Table 7.5. Parameter values for example power losses in Tables 7.3 and 7.4.[a]

Design Parameter	Value	Units
α_{fr}	0.1175	NA
a	440	μm
b	163	μm
$\mu \approx \mu_s$	0.12	NA
P	250	N
$c_{d,l}$	0.0034	NA
$c_{d,b}$	40	μm
C_{22}	2.87	NA
G	78	GPa
m_l	1	kg
J_b	0.00001	kgm^2
J_c	0.0236	kgm^2
R	0.0682	m
r	0.009	m

[a] Values for shear modulus of elasticity E and specific strength ρ are not given since they are not required for computation of the power losses.

Table 7.6. Operating conditions for example power losses of Tables 7.3 and 7.4.[a]

Operating Condition	Value	Units
ω	5	rads/s
$\dot{\omega}$	20	rads/s ^{2}
ϕ	0.53	radians
$\dot{\phi}$	5	rads/s
$\ddot{\phi}$	50	rads/s ^{2}
$\dot{l}$	0.2	m/s
$\ddot{l}$	2.3	m/s ^{2}
f_l	0	N
f_w	3.14	N

[a] This list contains several redundant conditions, however, all are stated for clarity.

CHAPTER 8

Focus 4: Simulation of a Dynamic Haptic Environment

When controlling a passive cobot interacting with a haptic environment, researchers have typically thought in terms of the extreme cobotic behaviors, *freemode* and *virtual-surface* mode, which are implemented quite differently on a passive cobot. With regards to the powered and parallel (redundantly actuated) Cobotic Hand Controller, the delineations between *freemode* and *virtual-surface* mode are blurred, and a completely different strategy can be adopted. The virtual environment description need not be set up to pull out motion commands strictly orthogonal to the current motion freedom (from which to take steering action). While the passive cobot has no control authority over haptic display of dynamics along its current motion freedom, the active Cobotic Hand Controller does. A more general strategy can be used which commands desired accelerations in all directions. In this chapter we develop such a strategy, for the general admittance controlled device, capable of displaying the proper dynamics for any desired inertia matrix, and rendering holonomic as well as nonholonomic constraints. No special adaptations need be made to utilize this algorithm with a cobotic display.

8.1. Haptic system framework

A teleoperation master or virtual environment haptic interface ideally conveys a simulated impedance to the user, in addition to any constraints on motion or configuration. For example, during a teleoperation task in which a surgeon uses a scalpel to make an incision, the impedance is that of the inertia of the scalpel sliding through the stiffness and damping of skin, subject to any artificially imposed motion or configuration constraints, to aid the surgeon in his/her task.

Figure 8.1 illustrates terminology we use to describe the operation of a haptic system. We term the physical device manipulated by the user the *manipulandum*, with task space coordinates x and unmasked inertia matrix $M_m(x)$. As in Chapter 2, coordinates x are the $\mathbb{R}^6$ coordinate representation of the $SE(3)$ workspace of the manipulandum. The *virtual tool* in the virtual environment has coordinates q which are of dimension of the inertia matrix of the virtual tool, $M(q)$ (e.g., 6 for a rigid body or 2 for a $2R$ manipulator). The virtual environment coordinates, q, are related to the coordinates of the *reference manipulandum*, x_r, by the kinematics described in Equation 8.1. Due to control errors, the actual (i.e. measured) manipulandum coordinates, x, may not precisely coincide with the reference manipulandum coordinates, x_r.

$$x_r = \varphi(q) \tag{8.1}$$

Virtual environment notation is summarized in Tables 8.1 through 8.4.

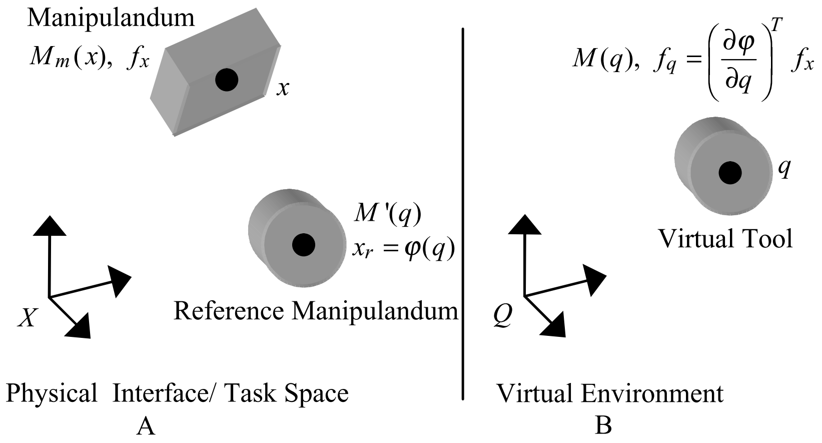


Figure 8.1. Physical interface / task space and virtual environment frames of reference. Our algorithm does not address redundant virtual linkages so $\dim(Q) \leq \dim(X)$. The task space version of $M(q)$, is $M'(q) = (\partial\varphi/\partial q)^{-T} M(q)(\partial\varphi/\partial q)^{-1}$, the operational space inertia [96].

The reference manipulandum moves according to a physics simulation propagated in the virtual environment coordinates, and always exactly satisfies the virtual holonomic constraints (Equation 8.2) and the virtual nonholonomic constraints (Equation 8.3).

$$H'(x) = 0 \tag{8.2}$$

$$A'(x)\dot{x} = 0 \tag{8.3}$$

The holonomic constraints can be differentiated and included in the $A'(x)$ matrix and we do so for the remainder of this work. However, creep resulting from integration errors must be avoided. A parametric method (later discussed in greater detail) is employed in order to keep the holonomic constraints satisfied.

The dynamic equations will be written and solved in the coordinates q, so the Pfaffian constraints are projected to this subspace (Equations 8.4 and 8.5). These constraints were written much more intuitively for rigid body motion in task space rather than in terms of a reduced set of virtual environment coordinates (e.g., a designer would use task space coordinates to constrain the end-point of a virtual linkage to move in a circle, and not the native joint angles of the linkage).

$$H'(x) = 0 \Rightarrow H'(\varphi(q)) = 0 \Rightarrow H(q) = 0 \tag{8.4}$$

$$A'(x)\dot{x} = 0 \Rightarrow A'(\varphi(q))\left(\frac{\partial \varphi}{\partial q}\right)\dot{q} = 0 \Rightarrow A(q)\dot{q} = 0 \tag{8.5}$$

Any rows of $H(q)$ and $A(q)$ that have all zeros after the projection can be removed.

For our example of the surgeon, we could employ both holonomic and nonholonomic constraints to limit the depth of the scalpel's incision and the direction of its motion, respectively. The holonomic constraint reduces both the number of available motion freedoms and the dimension of the configuration space. The nonholonomic constraint also reduces the number of motion freedoms but does not reduce the dimension of the configuration space.

Table 8.1. Virtual environment inertial notation.

Variable	Definition
$M_m(x)$	unmasked inertia matrix of manipulandum
$M(q)$	desired $n \times n$ inertia matrix of virtual environment object
$M'(q)$	desired 6×6 inertia matrix in task space
m_t	translational mass of rigid body virtual environment object
J_r	rotational inertia matrix of rigid body virtual environment object
$C(q, \dot{q})$	Coriolis matrix of $M(q)$
$\Gamma_{ij}^k(q)$	Christoffel symbols of $M(q)$

Table 8.2. Virtual environment constraint notation.

Variable	Definition
$H(q)$, $H'(x)$	holonomic constraints
$h_i(q)$	the rows of $H(q)$
$A(q)$, $A'(x)$	nonholonomic and differentiated holonomic constraints altogether
$a_i(q)$	the rows of $A(q)$
c_i	constant of integration for holonomic (integrable) constraints
λ	Lagrange multipliers (constraint force magnitudes)

The control scheme described in this chapter can be summarized as follows: The user applies a generalized force, f_x, to the manipulandum, which is sensed by a six-degree-of-freedom load cell. This force is transformed by the kinematics φ to a force

Table 8.3. Virtual environment external force notation.

Variable	Definition
τ	external(non-inertial and non-constraint) forces on $M(q)$
τ_d	generalized force of virtual damper
b_t, b_r	translational and rotational damping coefficients
${}^bv$, ${}^b\omega$	body fixed translational and rotational velocities
τ_s	generalized force of virtual spring
k_t, k_r	translational and rotational spring constants
p_o, R_o	translational and rotational spring origins
K, θ	axis angle description of orientation
$g(q)$	generalized force of gravity or potential fields

Table 8.4. Virtual environment submanifold notation.

Variable	Definition
n	dimension of system (length of coordinate vector q)
m	number of constraints (rows in $A(q)$)
b	number of holonomic constraints (rows in $H(q)$)
$n-(m-b)$	dimension of submanifold (parametric space)
Z	submanifold frame
z	coordinates of the submanifold
$x=\psi(z)$	n functions mapping frame Z to frame X

acting on the virtual tool, f_q.

$$f_q = \left(\frac{\partial\varphi}{\partial q}\right)^T f_x \tag{8.6}$$

Assuming that x is approximately x_r (one of the purposes of the controller is to keep the manipulandum close to the reference manipulandum), the generalized force f_q is applied to the virtual tool, along with any other virtual external forces due to springs, dampers, and gravity. These forces may be defined in task space and projected to the virtual environment dynamics via $\left(\frac{\partial \varphi}{\partial q}\right)^T$, or may be defined locally in virtual environment coordinates. The acceleration of the virtual tool is then calculated and integrated forward. The total acceleration is transformed by the kinematics φ to an acceleration of the manipulandum, $\ddot{x}$. This acceleration is a feedforward term applied to the reference manipulandum. In addition to this feedforward acceleration, a (hopefully small) feedback acceleration is applied to compensate for small position and velocity errors between the manipulandum and the reference manipulandum. The result is a realistic display of the constrained dynamics of the virtual tool.

Since the haptic display is executing motion, it is imposing forces on itself through any end-effector distal to the load cell. The weight and inertial effects of the mass of this distal object should be counteracted, such that the forces between the object and environment or operator are properly conveyed. The implementation of this with the Cobotic hand Controller is discusses in Appendix G.

8.2. Euler-Lagrange formulation

The Euler-Lagrange dynamic equations, including holonomic and/or nonholonomic constraints, are presented in Equations 8.7 and 8.8.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = \tau + A(q)^T\lambda \tag{8.7}$$

$$A(q)\dot{q} = 0 \qquad (8.8)$$

These equations are expressed in terms of n generalized coordinates q. $M(q)$ is an $n \times n$ symmetric positive definite inertia matrix. The Coriolis forces of the virtual rigid body or linkage are represented by the vector $C(q, \dot{q})\dot{q}$. $A(q)$ is an $m \times n$ matrix of Pfaffian constraints, either holonomic and/or nonholonomic, and λ is the vector of Lagrange multipliers representing the m constraint force magnitudes. The rows of $A(q)$ are the constraint force directions. External forces $\tau = f_q - \tau_d - \tau_s - g(q)$ are composed of user interaction forces, f_q, forces of virtual springs, τ_s, and forces of virtual dampers, τ_d. Generalized user interaction forces are measured via a force torque sensor in an admittance-type haptic display. The springs and dampers may be defined in task space and the resulting forces mapped to the virtual environment via $\left(\frac{\partial \varphi}{\partial q}\right)^T$, or can be defined natively in virtual environment coordinates (e.g., damping applied to the end-point of a linkage would likely be applied in task space, while damping at the joints of a linkage would be defined in the virtual environment coordinates of that linkage). Virtual gravity and potential fields are represented by $g(q)$.

Equations 8.7 and 8.8 apply for any simulated mechanical device. For the particularly interesting case of a rigid-body virtual tool, generalized coordinates can be given by three translational coordinates (q_1, q_2, q_3) and three Euler angles (q_4, q_5, q_6). The mapping $x = \varphi(q)$ will be identity for this example if we also describe the task space of our display by three translational coordinates (x_1, x_2, x_3) and three Euler angles (x_4, x_5, x_6) of the same parameterization. We have chosen an Euler angle set such that all singularities are outside of the workspace for our specific manipulandum. Euler angles allow us to work with generalized coordinates in $\mathbb{R}^6$ rather than with the special Euclidean group $SE(3)$ for a rigid body. $M(q)$is obtained from the system kinetic energy, $T(q,\dot{q})$, as shown in Equation 8.9.

$$M_{ij}(q) = \frac{\partial^2 T(q,\dot{q})}{\partial\dot{q}_i \partial\dot{q}_j} \qquad i,j = 1\dots n \tag{8.9}$$

$C(q,\dot{q})$ is obtained from the Christoffel symbols, $\Gamma^k_{ij}(q)$, as outlined in Equations 8.10 and 8.11 [30].

$$C(q,\dot{q})_{ij} = \sum_{k=1}^{n} \Gamma^k_{ij}(q)\dot{q}_k \tag{8.10}$$

$$\Gamma^k_{ij}(q) = \frac{1}{2}\left(\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} - \frac{\partial M_{kj}(q)}{\partial q_i}\right) \qquad i,j,k = 1\dots n \tag{8.11}$$

The inertia and Coriolis matrices for our specific choice of Euler angles and inertial properties are given in Appendix I.

8.3. Incorporation of nonholonomic and holonomic constraints

Given n generalized coordinates and m constraints, the tangent space at a given configuration has $n - m$ motion freedoms. If the rows of $A(q)$, $a_i(q)$, $i = 1...m$, can be represented by $\partial h_i/\partial q = a_i(q)$, for some real-valued functions $h_i(q)$, the constraint is said to be holonomic or integrable and can be written as $h_i(q) = c_i$.

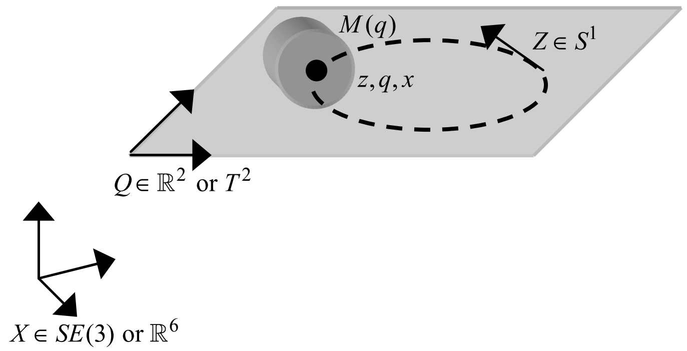

$M(q)$
$Z \in S^1$
z, q, x
$Q \in \mathbb{R}^2$ or T^2
$X \in SE(3)$ or $\mathbb{R}^6$

Figure 8.2. The embedding of Q and Z in X illustrated through an example. A two-dimensional virtual environment frame Q is embedded in the $SE(3)$ or $\mathbb{R}^6$ task space, X, of our haptic display. Frame Q is that of a Cartesian ($\mathbb{R}^2$) or two-rotational (T^2) mechanism. The end-point is limited to travel on a circular path, the one-dimensional (S^1) submanifold Z, due to a holonomic constraint defined in task space. The virtual mechanism has native inertia $M(q)$ and is located at coordinates, z, q and x in the various spaces.

If b of the m constraints in $A(q)$ correspond to holonomic constraints, there exists an $(n-b)$-dimensional integrable submanifold Z of reachable configurations. While nonholonomic constraints reduce the apparent motion freedoms, they do not decrease the dimensionality of the reachable configuration submanifold. The coordinates $z_j, j = 1...(n-b)$ parameterize the configuration space of the constrained system, a submanifold of the ambient space X. n functions $x = \psi(z)$ define the parametric description. As we define the Pfaffian constraints intuitively in task space as $A'(x)$, followed by a mapping to virtual environment space as $A(q)$, we also define the parametric description for the holonomic constraint surface in terms of task space coordinates x. These various spaces are illustrated in Figure 8.2.

8.4. Representation of damping and stiffness in $SE(3)$

Returning to the example of simulating a rigid body, the generalized damping wrench, τ_d, is computed in the body frame by using Equation 8.12.

$$(8.12) \qquad {}^b\tau_d = \begin{bmatrix} b_t I_{3\times3} & 0 \\ 0 & b_r I_{3\times3} \end{bmatrix} \begin{bmatrix} {}^b v \\ {}^b\omega \end{bmatrix}$$

b_t and b_r are the translational and rotational damping coefficients respectively. The body frame translational and angular velocities, ${}^b v$ and ${}^b\omega$, respectively, are computed as indicated in Appendix H.

The generalized spring wrench, τ_s, is computed in the body frame by using Equation 8.13.[1]

$$(8.13) \qquad {}^b\tau_s = -\begin{bmatrix} k_t I_{3\times3} & 0 \\ 0 & k_r \theta I_{3\times3} \end{bmatrix} \begin{bmatrix} R^T\left(p_o - [q_1, q_2, q_3]^T\right) \\ K \end{bmatrix}$$

k_t and k_r are the translational and rotational spring constants, respectively. The body frame spring origin, $R(q)^T\left(p_o - [q_1, q_2, q_3]^T\right)$, is computed from the world frame translational spring origin, p_o, where the rotation matrix R, given in Appendix H, transforms from body frame to world frame via three generalized Euler angle coordinates (q_4, q_5, q_6). K and θ are the axis-angle representation of the current orientation, where the axis and angle are expressed from the body frame to the spring origin. McNeely et al. also use the axis-angle formulation for rotational springs [114].

[1] Note that the bilateral constraints to be rendered here are considered infinitely rigid in the virtual environment, and are not described via springs or dampers. Springs can impact motion and forces in the free directions though.

The axis-angle representation is computed as shown in Equations 8.14 and 8.15 for $\Delta R = R^T R_o$, where R_o is the nominal spring orientation [30].

$$\theta = \arccos\left(\frac{\text{trace}(\Delta R) - 1}{2}\right) \tag{8.14}$$

$$K = \frac{1}{2\sin(\theta)} \begin{bmatrix} \Delta R_{32} - \Delta R_{23} \\ \Delta R_{13} - \Delta R_{31} \\ \Delta R_{21} - \Delta R_{12} \end{bmatrix} \tag{8.15}$$

8.5. Parametric formulation for integration

Several steps are required for propagating the physics simulation of a virtual tool that is subject to constraints. The input to the simulation is the current state of the virtual tool, q and $\dot{q}$, as well as forces, f_q, applied by an operator. The output is the resulting acceleration, $\ddot{q}$, based on the simulated inertia, damping, springs, gravity and constraints. This acceleration is then integrated in a manner consistent with the configuration submanifold specified by holonomic constraints, and becomes the new state of the physics simulation.

In order to solve for the acceleration, $\ddot{q}$, we first evaluate the Lagrange multipliers, λ, which are isolated by first differentiating $A(q)\dot{q} = 0$ and then inserting it into the dynamic equations for $\ddot{q}$. This yields Equation 8.16.

$$\lambda = (AM^{-1}A^T)^{-1}\left(-\dot{A}\dot{q} + AM^{-1}\left(C(q,\dot{q})\dot{q} - \tau\right)\right) \tag{8.16}$$

Subsequently we solve for the complete acceleration of the virtual tool (Equation 8.17).

$$\ddot{q} = M(q)^{-1}\left(\tau + A(q)^T\lambda - C(q,\dot{q})\dot{q}\right) \tag{8.17}$$

Plugging Equation 8.16 into Equation 8.17 yields Equation 8.18.

$$\begin{aligned} \ddot{q} &= -\tilde{A}\dot{A}\dot{q} + M^{-1}P_u(\tau - C\dot{q}) \\ \tilde{A} &= M^{-1}A^T(AM^{-1}A^T)^{-1} \\ P_u &= I_{n\times n} - A^T(AM^{-1}A^T)^{-1}AM^{-1} \end{aligned} \tag{8.18}$$

The projection $-\tilde{A}\dot{A}\dot{q}$ yields the accelerations due to constraint forces. The projection matrix P_u, as defined by Choset et al. [30], projects generalized forces to those that do work on the system (forces in the unconstrained directions).[2] Thus the projections $-\tilde{A}\dot{A}\dot{q}$ and $M^{-1}P_u(\tau - C\dot{q})$ yield the accelerations in the constrained and unconstrained directions, respectively. The reference acceleration, $\ddot{q}$, can now be integrated to yield $\dot{q}$ and q. The integration method must keep the virtual tool on the constraint submanifold. We suggest a parametric approach in order to effectively integrate the equations of motion and to keep the reference position on the constraint submanifold. Generalized accelerations, $\ddot{q}$, are related to device accelerations $\ddot{x_r}$, via

[2] Khatib [96] and Liu and Li [106] are also excellent sources for more on these projection operations.

the kinematics φ (Equation 8.19).

$$\ddot{x}_r = \frac{\partial \varphi}{\partial q}\ddot{q} + \dot{q}^T \frac{\partial^2 \varphi}{\partial q^2}\dot{q} \tag{8.19}$$

These in turn are related to parametric accelerations, $\ddot{z}$, via the kinematics given in Equation 8.20.

$$\ddot{x}_r = \frac{\partial \psi}{\partial z}\ddot{z} + \dot{z}^T \frac{\partial^2 \psi}{\partial z^2}\dot{z} \tag{8.20}$$

In order to integrate the parameters z from desired accelerations $\ddot{q}$, Equations 8.20 and 8.19 can be combined to yield Equation 8.21.

$$\ddot{z} = \left(\frac{\partial \psi}{\partial z}\right)^{\dagger} \left(\left(\frac{\partial \varphi}{\partial q}\ddot{q} + \dot{q}^T \frac{\partial^2 \varphi}{\partial q^2}\dot{q} \right) - \dot{z}^T \left(\frac{\partial^2 \psi}{\partial z^2} \right) \dot{z} \right) \tag{8.21}$$

Here $(\partial \psi / \partial z)^{\dagger} = \left((\partial \psi / \partial z)^T (\partial \psi / \partial z) \right)^{-1} (\partial \psi / \partial z)^T$ is the Moore-Penrose form of the pseudo-inverse. The pseudo-inverse is merely affecting a change of coordinates, or a kinematic projection in this case. Numerical integration of $\ddot{z}$ will yield a reference point on the configuration submanifold that is necessary in the next section for the computation of feedback terms. This projection to a parametric space and subsequent integration is essentially the *reference-cobot* method of Gillespie et al. [66].

The method of integration should be chosen carefully to avoid numerical problems over time, but our key concern in this paper is the instantaneous constrained dynamics. Also, although dynamics are implemented for a rigid body, we have chosen coordinates and not a Lie group representation (which would have utilized the special orthogonal group for rotations). Our choice of coordinates leads to integration problems near singularities. However, we have avoided much more complex integration issues for the implicit $SO(3)$ representation of orientation.[3] Euler angle coordinate representation singularities are not an issue here since they can be placed outside of the workspace for our specific six-degree-of-freedom device.

8.5.1. Upright rolling disk example

In order to demonstrate a physics simulation consisting of both holonomic and nonholonomic constraints, we consider a disk with inertia $M(q)$ as our rigid body in $SE(3)$, confined to rolling upright on a plane without slipping. Figures 8.3 and 8.4 define the translational coordinates (q_1, q_2, q_3) and rotational coordinates (q_4, q_5, q_6)[4] we have chosen to represent the disk. s indicates the generalized coordinate inertial frame, and b the body frame of the disk. Restricting a disk of radius r_d to roll on the plane $q_3 = 0$ and stand upright yields the holonomic constraints in Equations 8.22 and 8.23.

$$q_3 = r_d \rightarrow \dot{q}_3 = 0 \tag{8.22}$$

[3] This thesis does not deal with stabilization of constraints/integration issues since we mostly avoid both by using coordinates and parameters. Holonomic constraint drift is not possible with our methodology. Stabilization of integration of constraint equations without necessarily utilizing a parametric description for integration is addressed by [13, 15, 43, 129, 130]

[4] These are not the choice of Euler angles used for the actual implementation of the kinematics φ on the Cobotic Hand Controller, but they greatly simplify the constraint equations for the example given here.

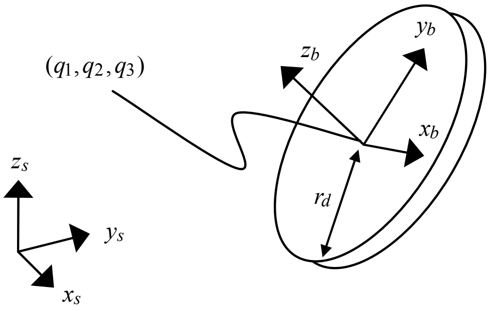


Figure 8.3. Definition of translational coordinates. Subscripts s and b indicate the spatial and body frame axes. Since the virtual tool is a rigid body, frame Q and task space frame X are equivalent for this example.

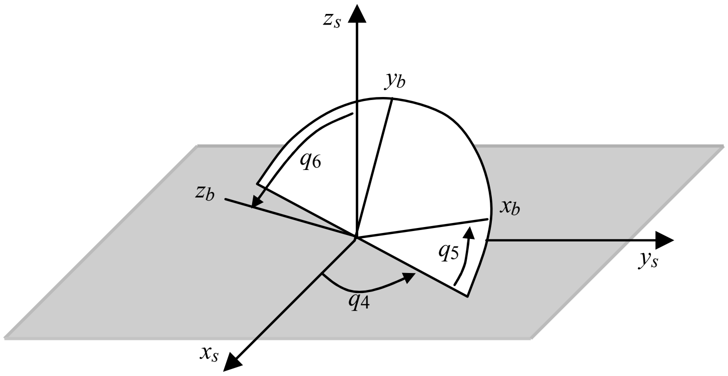


Figure 8.4. Definition of rotational coordinates.

$$q_6 = \frac{\pi}{2} \rightarrow \dot{q}_6 = 0 \tag{8.23}$$

Assuming the holonomic constraints are satisfied, imposing the no-slip rolling constraint yields the nonholonomic constraints represented by Equations 8.24 and 8.25.

$$r_d \dot{q}_5 \cos(q_4) + \dot{q}_1 = 0 \tag{8.24}$$

$$r_d\dot{q}_5\sin(q_4) + \dot{q}_2 = 0 \tag{8.25}$$

The constraint matrix is give by Equation 8.26.

$$A(q) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & r_d\cos(q_4) & 0 \\ 0 & 1 & 0 & 0 & r_d\sin(q_4) & 0 \end{bmatrix} \tag{8.26}$$

Since the virtual environment inertia for this example is that of a rigid body, our mapping from virtual environment space to task space, $x_r = \varphi(q)$, is identity. Thus the constraint descriptions $A'(x)$ and $A(q)$ are identical, and we have written them directly in terms of q. The reachable configuration submanifold is now four-dimensional and can be parameterized as described by Equation 8.27.

$$x_r = \psi(z) = \begin{bmatrix} z_1 \\ z_2 \\ r_d \\ z_3 \\ z_4 \\ \frac{\pi}{2} \end{bmatrix} \tag{8.27}$$

The inertia matrix $M(q)$ is given in Equation 8.9. Given a generalized user applied force, f_q, and current state, q and $\dot{q}$, we compute the acceleration of the virtual tool, $\ddot{q}$. Since we have the current z and $\dot{z}$, we can use Equation 8.21 to project the acceleration of the virtual tool, $\ddot{q}$, to the parametric acceleration, $\ddot{z}$. Integrating yields new values of $\dot{z}$ and z from which we can then compute new values $x_r = \psi(z)$ and

$\dot{x}_r = (\partial\psi/\partial z)\dot{z}$, and from these new values of $q = \varphi^{-1}(x_r)$ and $\dot{q} = (\partial\varphi/\partial q)^{\dagger}\dot{x}_r$. These terms will be used in the development of feedback controllers in the next section.

8.6. Feedforward and feedback components

As shown in Figure 8.1, the actual manipulandum position, x, likely does not correspond exactly with the reference manipulandum position, x_r. Thus a controller is needed to make the actual manipulandum track the reference manipulandum, which is exactly tracking the dynamics simulation in the virtual environment.

The feedforward acceleration of the manipulandum, $\ddot{x}_{ff}$, is equal to the complete acceleration of the reference manipulandum, $\ddot{x}_r$. A proportional plus integral plus derivative (PID) feedback controller for $\ddot{x}_{fb}$ will take the form described by Equation 8.28.

$$\ddot{x}_{fb} = K_p e + K_i \int e dt + K_d \dot{e} \tag{8.28}$$

$e = (x_r - x)$ is the tracking error of the actual manipulandum relative to the reference manipulandum state of the physics simulation. K_p, K_i and K_d are feedback gain matrices.[5] Summing the feedforward and feedback accelerations of the manipulandum yields the total acceleration command for the actual manipulandum (Equation 8.29).

$$\ddot{x} = \ddot{x}_{ff} + \ddot{x}_{fb} \tag{8.29}$$

[5] In Chapter 3 we performed the feedback in joint space on a joint by joint basis rather than in task space coordinates. Reference motions in task space were mapped to desired joint motions and these were compared with actual joint motion. The PID gains consisted of a set of 3 scalars for each joint.

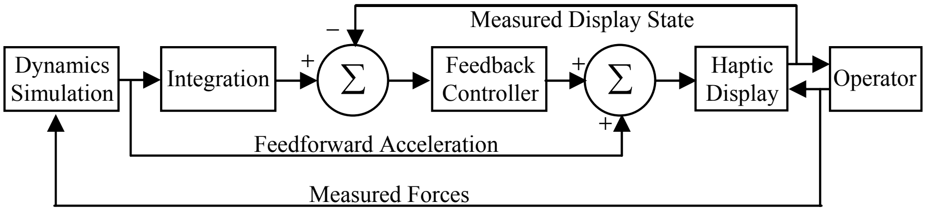


Figure 8.5. A block diagram of an admittance-type haptic control system.

Device specific kinematics convert $\ddot{x}$ into joint level commands ($\ddot{l}$ via Equation 2.7) and these into steering commands for cobots ($\dot{\phi}$ via Equation 3.6).

Algorithm 3 provides a summary of the virtual-environment simulation method. It includes the application of the Euler-Lagrange equations, the Pfaffian description of motion constraints, the parametric description of holonomic constraints and the integration and feedback control of motion. A block diagram of the admittance-type haptic control system described here is shown in Figure 8.5. Implicit in this diagram is the need to cancel the static and dynamic effects of an end-effector mass that is distal to the force sensor.

8.6.1. Rolling disk example, continued

Returning to the example of the disk (Section 8.5.1), we now see that the reference disk state, $x_r = \psi(z)$ and $\dot{x}_r = (\partial\psi/\partial z)\dot{z}$, can be utilized to generate feedback accelerations both in the constrained directions and in the free directions. The feedback terms attempt to keep the velocity vector of the virtual disk to purely rolling in the plane and

Algorithm 3 Virtual environment algorithm

```
Require: Timestep $T$
Require: Initial parametric coordinates $z_o$
Require: Kinematics from virtual environment to task $x_r = \varphi(q)$
Require: Desired virtual environment inertial properties $M(q)$ and $C(q, \dot{q})$
Require: Pfaffian and parametric descriptions $A(q) = A'(\varphi(q))\frac{\partial\varphi(q)}{\partial q}$, $x_r = \psi(z)$
Require: Damping and stiffness descriptions $b_t$, $b_r$, $\tau_d$, $k_t$, $k_r$, $p_o$, $R_o$, $\tau_s$
Initialize $z \rightarrow z_o$, $q \rightarrow \varphi^{-1}(\psi(z_o))$, $\dot{z} \rightarrow 0$ and $\dot{q} \rightarrow 0$
repeat
    Measure $x$ and $\dot{x}$
    Measure $f_x$, Evaluate $f_q = \frac{\partial\varphi(q)}{\partial q}^T f_x$
    Evaluate $\tau_s$, $\tau_d$ and $g(q_r)$
    Evaluate $\tau = f_q - \tau_d - \tau_s - g(q_r)$
    Evaluate $M(q)$, $C(q, \dot{q})$, $A(q)$, $\dot{A}(q, \dot{q})$, $\tilde{A}(q)$ and $P_u(q)$
    Evaluate $\ddot{q} = -\tilde{A}\dot{A}\dot{q} + M^{-1}P_u(\tau - C\dot{q})$
    Evaluate $\ddot{x}_{ff} = \ddot{x}_r = \frac{\partial\varphi}{\partial q}\ddot{q} + \dot{q}^T\frac{\partial^2\varphi}{\partial q^2}\dot{q}$
    Evaluate $\ddot{z} = \left(\frac{\partial\psi}{\partial z}\right)^{\dagger}\left(\ddot{x}_r - \dot{z}^T\left(\frac{\partial^2\psi}{\partial z^2}\right)\dot{z}\right)$
    Integrate $z = z + \dot{z}T + \frac{1}{2}\ddot{z}T^2$ and $\dot{z} = \dot{z} + \ddot{z}T$
    Evaluate $x_r = \psi(z)$, $\dot{x}_r = \frac{\partial\psi}{\partial z}\dot{z}$, $e = x_r - x$ and $\dot{e} = \dot{x}_r - \dot{x}$
    Evaluate $q = \varphi^{-1}(x_r)$, $\dot{q} = \left(\frac{\partial\varphi}{\partial q}\right)^{\dagger}\dot{x}_r$
    Evaluate $\ddot{x}_{fb} = K_p e + K_i\int e dt + K_d\dot{e}$
    Evaluate $\ddot{x} = \ddot{x}_{ff} + \ddot{x}_{fb}$
    Delay for haptic timestep $T$
until haptic simulation ends
```

to obeying the no-slip rolling condition. Feedback terms will null out any deviations of the virtual disk from upright, and any deviations of the virtual disk from the plane. Feedback will also assist the feedforward terms in keeping the virtual disk spinning and rolling in pace with the reference disk.

8.6.2. Disk sliding on sphere example

As a second example of the algorithm presented here, consider a disk as shown in Figure 8.6, whose center must stay in contact with a sphere of radius r_s located at (s_1, s_2, s_3). The reference disk can move in two degrees of freedom about the sphere that it is constrained to, and can rotate about the contact normal axis. The reference disk, nominally residing in six-dimensional $SE(3)$, is given a set of generalized coordinates, q, in the same fashion as in Figures 8.3 and 8.4. In order to establish the constraint equations for this scenario, consider a rotation matrix $R(q_4, q_5, q_6)$,[6] composed of a rotation of q_4 about z_b, followed by a rotation of q_6 about the new x_b and followed by a rotation of q_5 about the new z_b. This matrix multiplied by the vector $(0, 0, r_s)^T$ in the body fixed frame will yield the current generalized coordinates (q_1, q_2, q_3) of the virtual tool less the location of the sphere (s_1, s_2, s_3) (Equation 8.30). The cells of the rotation matrix irrelevant to this example are not shown.

$$\begin{bmatrix} \dots & \dots & \sin(q_5)\sin(q_6) \\ \dots & \dots & \cos(q_5)\sin(q_6) \\ \dots & \dots & \cos(q_6) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ r_s \end{bmatrix} = \begin{bmatrix} q_1 - s_1 \\ q_2 - s_2 \\ q_3 - s_3 \end{bmatrix} \tag{8.30}$$

We can then pull out three holonomic constraint equations (Equations 8.31 through 8.33).

$$q_1 - r_s \sin(q_5)\sin(q_6) - s_1 = 0 \tag{8.31}$$

[6] Again, these are not the choice of Euler angles used for the actual implementation of the kinematics φ on the Cobotic Hand Controller, but they greatly simplify the constraint equations for the example given here.

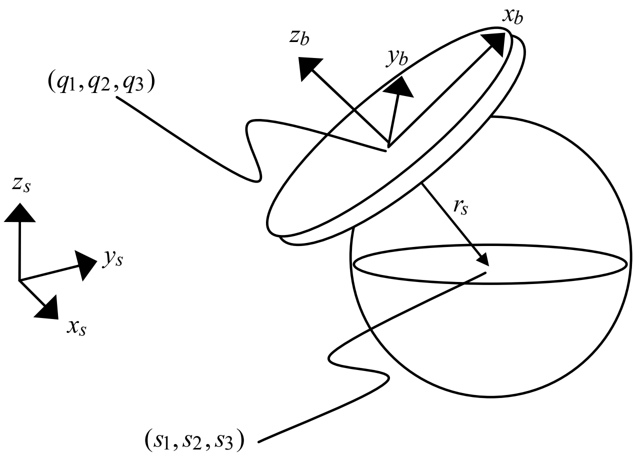


Figure 8.6. This virtual environment consists of a disk required to stay tangent to a sphere, but allowed to spin about the contact normal. Subscripts s and b indicate the spatial and body frame axes.

$$q_2 - r_s \cos(q_5)\sin(q_6) - s_2 = 0 \tag{8.32}$$

$$q_3 - r_s \cos(q_6) - s_3 = 0 \tag{8.33}$$

There is a lack of dependence on q_4, the allowed spin of the disk about the contact normal. These three holonomic constraint equations can be differentiated to yield Equation 8.34. Again, since the transformation φ between Q and X is identity for this example, we have written down $A(q)$ directly rather than $A'(x)$.

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & -r_s\cos(q_5)\sin(q_6) & -r_s\sin(q_5)\cos(q_6) \\ 0 & 1 & 0 & 0 & r_s\sin(q_5)\sin(q_6) & -r_s\cos(q_5)\cos(q_6) \\ 0 & 0 & 1 & 0 & 0 & r_s\sin(q_6) \end{bmatrix} \tag{8.34}$$

The reachable configuration submanifold is now three-dimensional and can be parameterized as described by Equation 8.35.

$$x_r = \psi(z) = \begin{bmatrix} s_1 + r_s \sin(z_2)\sin(z_3) \\ s_2 + r_s \cos(z_2)\sin(z_3) \\ s_3 + r_s \cos(z_3) \\ z_1 \\ z_2 \\ z_3 \end{bmatrix} \tag{8.35}$$

8.6.3. $2R$ manipulator constrained to a circle example

As a third example of the algorithm presented here, consider a $2R$ manipulator, with two links of length, l, and uniformly distributed mass, m, as shown in Figure 8.7. The kinetic energy of the manipulator may be written as the sum of the translational and rotational energy in both links, and the virtual environment space inertia matrix (Equation 8.36) can be computed using Equation 8.9.

$$M(q) = \begin{bmatrix} \frac{ml^2}{3}(5 + 3\cos(q_2)) & \frac{ml^2}{6}(2 + 3\cos(q_2)) \\ \frac{ml^2}{6}(2 + 3\cos(q_2)) & \frac{ml^2}{3} \end{bmatrix} \tag{8.36}$$

The virtual environment space Coriolis matrix (Equation 8.37) can be similarly computed from the kinetic energy via Equations 8.10 and 8.11.

$$C(q, \dot{q}) = \begin{bmatrix} -\frac{ml^2}{2}\sin(q_2)\dot{q}_2 & -\frac{ml^2}{2}\sin(q_2)(\dot{q}_1 + \dot{q}_2) \\ \frac{ml^2}{2}\sin(q_2)\dot{q}_1 & 0 \end{bmatrix} \tag{8.37}$$

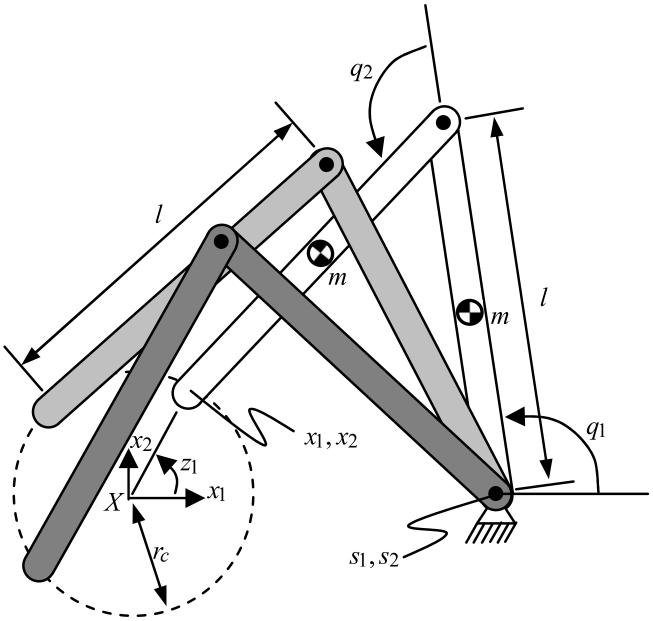


Figure 8.7. This virtual environment consists of a $2R$ manipulator, with base affixed in the X frame at (s_1, s_2), whose endpoint is constrained to move along a circle of radius r_c in task space.

For this example, the Jacobian $(\partial\varphi/\partial q)$ between virtual environment coordinates and task space coordinates, is no longer identity. Rather, the functions φ are given by Equation 8.38.

$$x = \varphi(q) = \begin{bmatrix} s_1 + l\cos(q_1) + l\cos(q_1 + q_2) \\ s_2 + l\sin(q_1) + l\sin(q_1 + q_2) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{8.38}$$

The Jacobian $(\partial\varphi/\partial q)$ will now be non-square, and will map the six-dimensional force and torques of the user in task space to the two torques on the virtual manipulator. Conversely, once the dynamics have been performed in the reduced set of coordinates, the kinematics φ^{-1} will project the manipulator motion up from virtual environment coordinates to task space coordinates.

In Figure 8.7, the end-point of the $2R$ manipulator is constrained to a circle of radius r_c. Equation 8.39 gives this holonomic constraint in task space coordinates. This may be written as a Pfaffian constraint (Equation 8.40) and mapped to virtual environment coordinates via $A(q) = A'(\varphi(q))(\partial\varphi/\partial q)$ (Equation 8.41).

$$x_1^2 + x_2^2 = r_c^2 \tag{8.39}$$

$$A'(x) = \left[\begin{array}{cccccc} 2x_1 & 2x_2 & 0 & 0 & 0 & 0 \end{array}\right] \tag{8.40}$$

$$A(q)^T = \left[\begin{array}{c} 2l\left(\left(\cos(q_1+q_2)+\cos(q_1)\right)s_2 - \left(\sin(q_1+q_2)+\sin(q_1)\right)s_1\right) \\ 2l\left(\cos(q_1+q_2)s_2 - \sin(q_1+q_2)s_1 - l\sin(q_2)\right) \end{array}\right] \tag{8.41}$$

The reachable configuration submanifold is one-dimensional and is parameterized via Equation 8.42.

$$x_r = \psi(z) = \begin{bmatrix} r_c \cos(z_1) \\ r_c \sin(z_1) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{8.42}$$

8.7. Experimentation with the Cobotic Hand Controller

The three examples from Sections 8.5.1, 8.6.2 and 8.6.3 were implemented on the Cobotic Hand Controller. The implementations used slightly different notation than described above, since a different Euler angle parameterization was used. The constraint equations and parametric descriptions for these examples and several others are described in Appendix J as implemented on the Cobotic Hand Controller.

8.7.1. Upright rolling disk example, continued

The six-centimeter-diameter disk in Figure 8.8 is in rolling contact with the plane. It is allowed to spin and roll, but is unable to slide sideways. It is able to move sideways by "parallel parking" motions, or by simply turning and driving in the desired direction. The simulated disk has mass 0.25 kg, all principal-axis inertias are 0.0025 (kg)(m)^2, translational damping is 1.0 (N)(s/m), and rotational damping is 0.1 (Nm)(s/rad). In Figure 8.9, several metrics of the implementation are reported. The error in the height of the disk and orientation of the disk are shown to be negligible and are on the order of the position resolution of the Cobotic Hand Controller. Also reported is the percent error in the rolling constraint, $0.01|v - r_d\omega|/(|v| + \epsilon)$, where $\epsilon = 0.001$ m/s, $r_d\omega = -(0.03)\dot{x}_5$ and $v = \dot{x}_1 \cos(x_4) + \dot{x}_2 \sin(x_4)$ is in m/s.

8.7.2. Disk sliding on a sphere example, continued

Figure 8.10 portrays a two-centimeter-diameter disk constrained to remain tangent to a ten-centimeter-diameter sphere, but permitted to rotate about its contact normal with the sphere. The disk has the same inertial properties as in the upright rolling disk example, although we have changed its dimensions. In Figure 8.11, several metrics of the implementation are reported. Also demonstrated is the ability of the Cobotic Hand Controller to impart rigid constraints (not described by a spring), since it suffers no significant position errors even when subjected to a 35 N load that is normal to the constraint surface.

8.7.3. $2R$ manipulator constrained to a circle example, continued

Figure 8.12 depicts experimental results of the implementation of the virtual $2R$ manipulator constrained to a 5 cm radius circle. The links each have length 15 cm, mass 2.0 kg, and damping at the joints of 0.01 Nm(s)/(rad). The base of the manipulator is 21 cm from the center of the circular constraint. The operator manipulates the virtual linkage for a few seconds and then releases it. The operator was attempting to maintain a constant velocity tangential to the constraint. This is difficult as it requires that the kinetic energy in the virtual manipulator rise and fall, and thus the user must alternately push and hold back the manipulator to maintain the constant tangential velocity. Once released, the kinetic energy stored in the linkage decreases monotonically due to damping at the joints, but the kinetic energy in each individual joint rises and falls.

8.8. Conclusions

The general algorithm developed here for admittance mode haptic display allows several firsts in cobotic haptic display. In addition to point masses, the cobot can now simulate the configuration-dependent inertia of rigid bodies and linkages, through the use of the Euler-Lagrange equations to relate forces and velocities with respect to the simulated inertia. The virtual-surface algorithm as previously posed did not support the representation of nonholonomic constraints. This was not seen as a limitation for the relatively low degree-of-freedom cobots prior to the Cobotic Hand Controller. While the algorithm developed in this Chapter still utilizes a parametric description of holonomic constraints for position feedback, the holonomic constraints are additionally written in a coordinate (not parametric) description, then differentiated with respect to time and included along side any nonholonomic constraints. Thus both holonomic and nonholonomic constraints share a Pfaffian description, are inserted as such into the Euler-Lagrange equations and limit the instantaneous motion freedoms of the cobot. Nonholonomic constraints can be thought of as constraints within the tangent space (motion freedoms) of the configuration (sub)manifold.

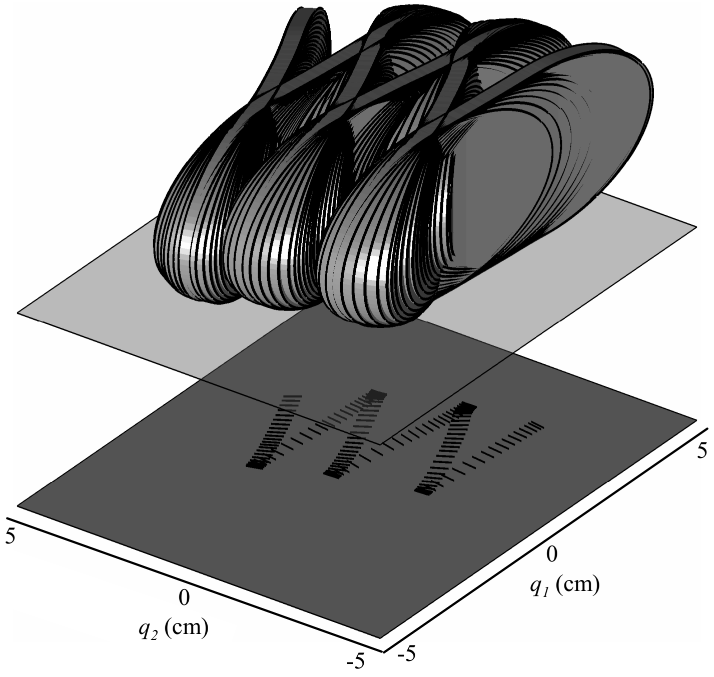


Figure 8.8. A disc shown at 0.1 second intervals in rolling contact with a planar surface. The shadow shows the line contact of the disc with the plane, or “tire tracks,” as the disk makes “parallel parking” motions.

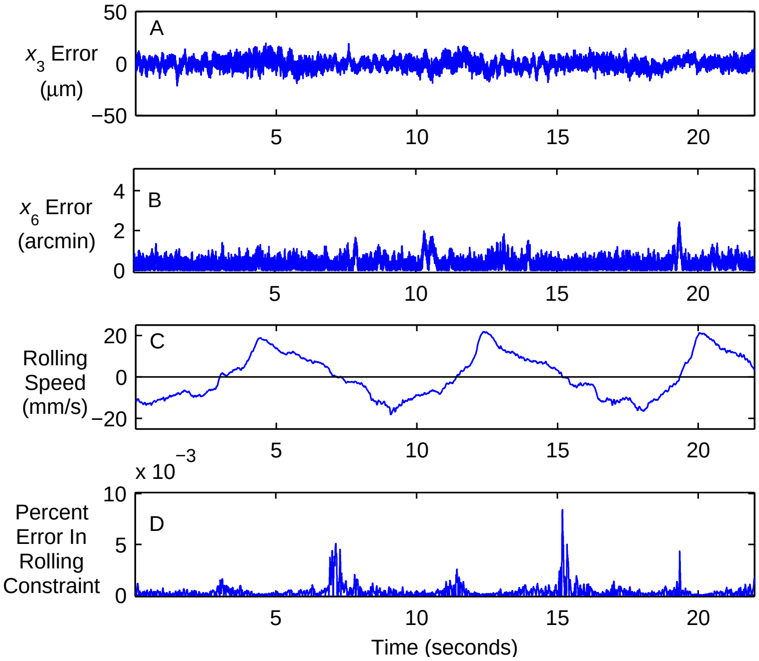


Figure 8.9. Experimental data recorded during the implementation of the upright rolling disk example with the Cobotic Hand Controller. All data are computed from the measured state of the display (x and $\dot{x}$). A) Error in the height of disk center. B) Error in the disk's upright orientation. The noise present in A and B is partially due to analog sensor noise and are nearly equivalent to what would be obtained when the device is not moving. C) Rolling velocity $v = \dot{x}_1 \cos(x_4) + \dot{x}_2 \sin(x_4)$ of the disk. D) The percent of error in the rolling constraint is $0.01|v - r_d\omega|/(|v| + \epsilon)$ where $r_d\omega = -(0.03)\dot{x}_5$, v is in m/s and $\epsilon = 0.001$ m/s. ϵ prevents this metric from reporting infinite error at near zero speed.

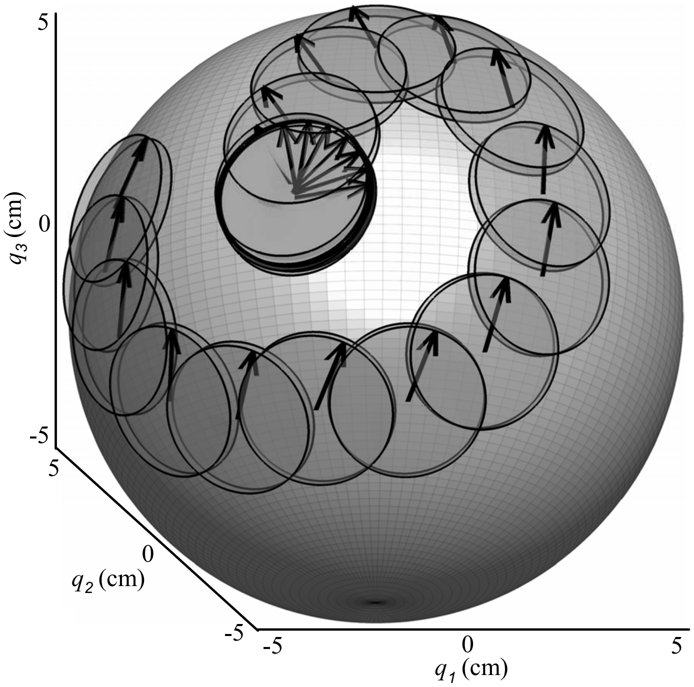


Figure 8.10. A simulated disk, shown at one second intervals, is constrained to track a virtual sphere while remaining tangent to it, but is permitted to rotate about the contact normal. The arrows indicate the orientation of the disk on the sphere.

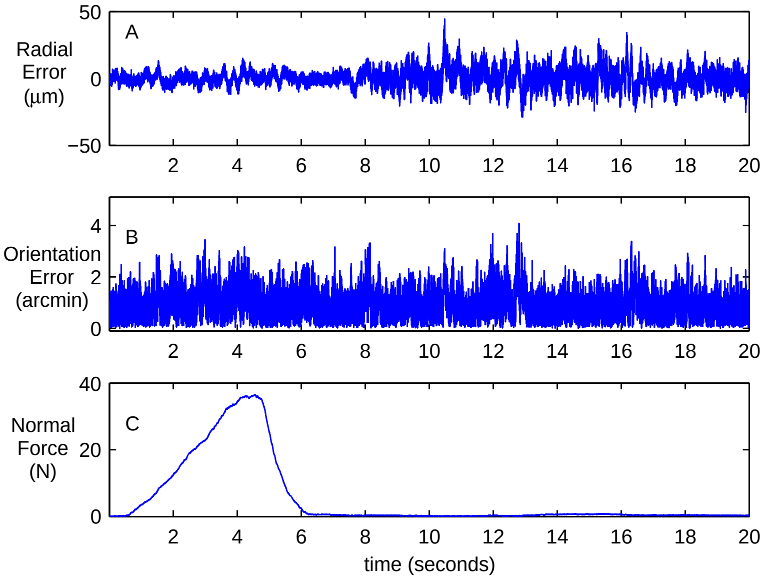


Figure 8.11. Experimental data recorded from an implementation of the disk sliding on sphere example with the Cobotic Hand Controller. All data are computed from the measured state of the display (x and $\dot{x}$). A) Translational deviation of the disk from the surface of the sphere. B) Error in the disk orientation as computed from the angle between the surface normal of the sphere and the surface normal of the disk at the point of contact. C) Normal force applied by the operator along the contact normal. The data indicate that the cobot is able to effectively ignore this large applied force, since no significant position errors occurred.

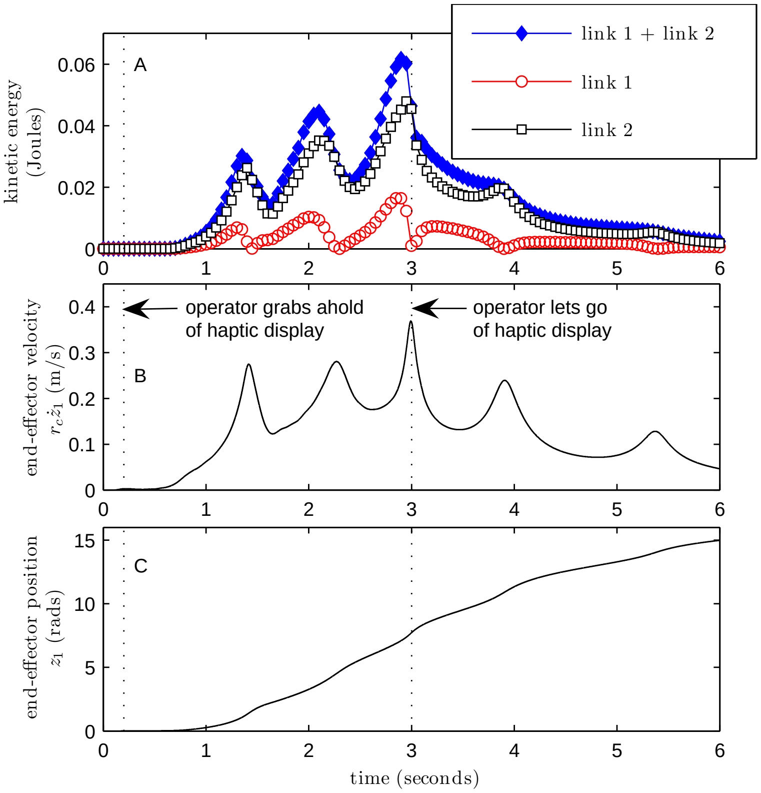


Figure 8.12. The operator grabs ahold of the end-effector and manipulates the virtual linkage. A. The total kinetic energy in the manipulator and the kinetic energy in the individual links. B. The speed of the endpoint tangential to the circular task space constraint. C. The position of the endpoint in parametric coordinate z ($z = 2\pi$ is one revolution about the constraint). Once the operator releases the linkage, the kinetic energy stored in the linkage decays monotonically due to virtual damping at the manipulator joints. Note that the kinetic energy of the base link (link 1) goes to zero each time it reverses direction. The kinetic energy of the distal link (link 2) never approaches zero as it has translational kinetic energy even when it has no rotational velocity. Note the increase in end-point speed at times 3.9 s and 5.4 s even while the total manipulator kinetic energy is decaying. When an operator attempts to maintain a steady speed along the constraint, they must alternately push and hold back the manipulator.

CHAPTER 9

Conclusions and Future Work

9.1. Design of the Cobotic Hand Controller

We have designed and built an active six-degree-of-freedom cobotic haptic display with force transmission capabilities exceeding 50 N, structural stiffness ranging from 20-400 kN/m, and a stable motion control bandwidth of 40 Hz. Based on our experience with haptic interface devices, the feel of this device is quite remarkable. The crisp distinction between free and forbidden directions of motion is striking. This performance arises not from elaborate control algorithms, but from the inherent physical characteristics of the device, the utilization of nonholonomic constraints. In order to render constraints on motion, the device controls the steering motion of transmissions rather than attempting to match an actuator torque to an operator applied effort. The Cobotic Hand Controller only requires a few watts of electrical power even while imparting high stiffness values and forces to the user,

A key design choice, which enabled the display of high stiffness values and crisp constraints, was the use of steel-on-steel precision-ground bearing quality components in dry-friction rolling contact. This choice yielded transmissions with high bandwidth, near-zero rolling friction, zero backlash and zero compliance.

A critical issue for steel-on-steel rolling contact systems is the impact of wear on performance. Wheels of cold-rolled steel or even 4140 alloy steel of hardness Rc 21 wear quickly and produce metal shavings. The wearing action also leads to an ever changing contact patch size, necessitating controller gain changes. In Chapter 2 we described the use of bearing quality hardened and ground components that virtually eliminated wear and effectively stabilized the dimensions of the contact patch. Further study should include an evaluation of the mechanism(s) by which the minimal wear occurs in order to improve the durability of the dry-friction rolling contact transmission. The state of stress is not simply a Hertzian contact situation, but includes additional surface shear loads due to steering.

The rolling contact transmissions of the Cobotic Hand Controller have very high preload forces relative to power throughput. This is due to the fact that cobotic devices tend to be backdrivable and, therefore, the rolling contacts of the transmissions are operated at relatively low velocities. Our use of steel-on-steel in the Cobotic Hand Controller, versus the historical use of polyurethane Rollerblade^{TM} wheels that required much lower preload forces, exacerbated this issue. In the current implementation, 250 N of preload is able to mediate RMS 10 Watts (shaking the mass of a leg), or 14 Watts peak, through a single transmission. When the transmissions steer away from a zero ratio, the lateral force on the wheel grows with $\sec(\phi)$, and thus large joint forces cannot be delivered at high velocities. This was addressed via our energy-tracking cylinder controller, where we increased cylinder speed as a function of total energy in the virtual environment. This had the effect of reducing steering angles when higher power throughput was expected.

Although the power throughput levels are relatively small compared to preload forces, an electrical motor must be operating at significant speeds in order to provide them. Even our large 1400 Watt motor is only capable of 3.7 Watts/(rad/s). While putting a gear ratio between the common element and power motor runs counter to the desire to be backdrivable, it may be necessary in order to reduce weight, to reduce electrical requirements, and to maintain safety. Gearing is needed to provide more torque at lower common element speeds. While greater speeds would allow our direct-drive implementation to deliver more mechanical power, they would result in excessive noise, vibration and wear.

A future design of the Cobotic Hand Controller should include some form of mechanical impedance to impede rolling contact induced vibrations from flowing out to the end-effector. These vibrations tend to excite the proximal and distal links, imparting a sense of "aliveness" that is somewhat distracting to the user. The slower speeds of the energy-tracking controller did help to reduce vibrations during times of low end-effector speed.

The size of the steering motors should be more carefully chosen in a future design. The peak velocity and peak torque of the steering motors in the current implementation exceed what is necessary to make the joints slip. While trying to maximize the bandwidth of our transmissions, we ignored the fact that the 200 rads/s steering velocities achievable by our steering plants exceeds the ability of our preload to accelerate our joint masses. By choosing a large steering motor, we exacerbated the problem because the mass of the motor is added to that of the joint. Larger gear ratios would allow us to combat the contact patch shear and bearing friction with a much smaller motor, and operate this motor at more efficient speeds. This would also reduce contributions of torque ripple to instability, although more backlash in the transmissions would need to be dealt with. Another solution would be anti-backlash gears or a cable drive for the steering plant reduction. This would allow for a higher ratio, yet would avoid the backlash that limits our steering bandwidth by exciting structural resonances.

9.2. Control architecture

We have introduced many innovations to cobot control in our design of a control architecture for the Cobotic Hand Controller. We designed an admittance architecture for cobots, based on commanding acceleration rather than curvature. We chose to convey both holonomic and nonholonomic constraints via the constrained Euler-Lagrange equations, rather than constraint-normal based Euclidean projection methods. We simulated springs and dampers, gravity and friction in addition to masses. Our architecture also enabled us to accurately simulate the configuration dependent inertias that arise from any selected topology. We chose to perform our feedback motion control in joint space, both orthogonal and tangent to the instantaneous motion freedom of the cobot. We utilized steering velocity control rather than steering angle control. We developed a novel energy-tracking cylinder speed controller rather than utilizing a fixed-speed or constant-ratio controller. We designed a maximum steering velocity heuristic to prevent transmission slip. We utilized CVT angles and cylinder speed to determine joint space velocities rather than differentiating a joint position sensor.

We have shown that the Cobotic Hand Controller has a high dynamic range. Although the Cobotic Hand Controller is controlled as an admittance device, allowing motions based on the applied force, the cobot does not suffer from the high inertia, friction and backlash that normally exist in a highly geared admittance device. Therefore, it is not as limited as other admittance displays in the range of impedances it can convey. Depending on the impedance that the Cobotic Hand Controller is interacting with, the stable virtual impedances it can render are characterized by a top stable natural frequency between 15 and 40 Hz, with stable damping ratios between 0.03 and 20. The minimum useful impedance is around 2 kg with zero damping, although a “careful” user is capable of manipulating a 0.1 kg mass with zero damping. The 2 kg mass need not have any weight (zero virtual gravity), so the operator only feels the inertial forces which are small for low accelerations. The maximum unilateral stiffness we can expect to accurately render during the impact of a 2 kg virtual mass with an initial speed of 0.5 m/s, is on the order of 1 to 3 kN/m. If the stiffness were larger, the cobot would penetrate the constraint further than the reference due to the joint acceleration limit we have recognized.[1] The maximum unilateral stiffness of 1 to 3 kN/m is similar to the Phantom’s (a common haptic display from

[1] This limit applies only when the lateral forces on the cobot wheels are dominated by the acceleration of joint inertias rather than by user applied forces or gravity loads of the end-effector.

Sensable Technologies [158]) peak stiffness. However, after impact, the Cobotic Hand Controller is able to sustain much larger loads and accurately portray higher stiffness than most impedance displays, including the Phantom. If we focus on bilateral constraint performance, without the high frequency content and accelerations of an impact, the Cobotic Hand Controller is adept at rendering 50 kN/m stiffness levels.[2] Nevertheless, future cobot control improvements should strive to maximize the range of impedance by fully utilizing acceleration capability, particularly during unilateral impacts. As mentioned in Section 5.2, it may be useful to pursue an event-based solution to unilateral impacts by simply playing back a steering trajectory independent of joint motion control.

The 40 Hz bandwidth limitation[3] of the Cobotic Hand Controller is not due to structural resonance and actuator torque limits that commonly limit haptic displays. The stable impedance range of the Cobotic Hand Controller is limited by steering dynamics and transmission creep. Steering dynamics produce a phase lag relative to the virtual environment that tends to help filter out high frequency content from transmitting from the virtual environment to the proximal and distal links. However, creep, and, in the worst case, slip, occurs in the transmission when it attempts to accelerate/decelerate the inertia of a link. This slip causes a deviation between the reference and virtual positions, and large feedback errors are generated that require

[2] If the structural stiffness of the Merlet platform is taken into account, these 50 kN/m software levels may in actuality only be 20 to 40 kN/m, depending on configuration.

[3] The 40 Hz stable bandwidth of virtual environment impedance is slightly lower than the 60 Hz structural resonance of the Merlet platform. It is also below the 100 Hz gear-train resonance of the steering plants and below the 50 Hz point at which the closed loop joint motion control becomes unstable.

even larger steering velocities. In addition, the high frequency content of a slip occurrence will excite the structural resonances of the proximal and distal links. We believe that slip is the mechanism that ultimately limits the range of impedances that can be stably rendered. In analyzing the impedance range of a device like the Cobotic Hand Controller, it is important to determine if the simulation and physical environment (operator or otherwise) are soliciting an acceleration greater than the capacity of the display.[4]

9.3. Modelling

We have provided a thorough analysis of the rotational-to-linear cobotic transmission, describing all forms of dissipative losses at the contact patch and in the remainder of the cobotic architecture. The modelling of dissipative losses has helped to define key material parameters for cobot transmission design. The free-rolling friction coefficient, α_{fr}, should be minimized in order to reduce inelastic losses. The utilization of steel-on-steel and the resulting high preloads described here may have been too extreme since the CVT wheel axle bearings are a source of dissipation on the order of ten times larger than the inelastic rolling friction at the contact patch. Increasing the modulus of elasticity, G, would reduce lateral creep. Increasing the coefficient of friction, μ, reduces the required preload, and consequently reduces nearly all forms of dissipation, with the exception of lateral creep. Reducing bearing diameters, specifically the steering bell and CVT wheel axle bearings, may significantly

[4] Hayward and Astley note that while motion bandwidth is important, peak acceleration is a key system parameter, often not reported [77].

reduce frictional power dissipation. Finally, increasing the specific strength, ρ, of materials for the cylinder, bell, and carriage would reduce inertial losses.

We have also compared the performance and efficiency of cobotic actuation architectures to conventional actuation architectures. Through an extensive comparative analysis, a cobotic rotational-to-linear actuation system was shown to require a smaller power actuator and to have a comparable, if not higher, power efficiency than conventional electro-mechanical linear actuation systems for frequencies of voluntary human motion.

9.4. Dynamics simulation

We have outlined a virtual environment simulation and integration method for the haptic display of rigidly constrained dynamic systems. We derived a combined feedforward and feedback controller for the motion of the actual manipulandum relative to a reference manipulandum. In addition to point masses, the framework presented here can also be used to simulate the configuration-dependent inertia of rigid bodies and of linkages. Example constraint scenarios have been provided, including the first display of nonholonomic constraints with a cobot, and data derived from their implementation on the Cobotic Hand Controller are reported. We contend that the user feels both inertial and viscous forces that vary over a wide dynamic range. The proper coupling of translational and rotational dynamics in response to a generalized force, which is composed of both forces and torques, is conveyed, even in the presence of constraints on motion.

This haptic environment framework could be extended by examining its utility for rigid bilateral display of NURBS defined surfaces. However, an issue that arises with NURBS is the required switching between maps in an atlas of alternate parametric descriptions. Even a simple sphere described with our methods requires transitions between sets of parametric descriptions since it cannot be covered by a single two-coordinate map that can be integrated at all points. A formal method for switching between various bilateral constraint descriptions should be devised. In addition to more general methods for display of surfaces, research should evaluate the ability of cobots to convey surface properties such as texture and friction.

9.5. The future of cobotics

We envision cobotic continuously-variable transmissions as an enabling technology for haptics and prosthetics that will allow for increases in the dynamic range of these devices while simultaneously permitting reductions in actuator size and power requirements. Use of a CVT eliminates the need to make compromises on output flow and effort, which are inherent to choosing a fixed transmission ratio. The result would be a mechanism with enhanced dynamic range that extends continuously from a completely clutched state to a highly backdrivable state.

Three key requirements for robotic technologies used in prosthetics and rehabilitation are low weight, low power consumption and safety. We propose cobotic technology as a transmission architecture that can address all of these issues. Cobots allow for variable back-drivability, high efficiency, precise control of output force and velocity at low output speeds, and a single power actuator for multiple degrees of freedom without the need for brakes or clutches.

We have demonstrated the scalability of cobotic technologies to produce high degree-of-freedom, high bandwidth devices. Our broad analysis of dissipative losses and efficiency comparisons between cobots and conventional systems, our contribution of a novel solution to the control of a redundantly actuated device, and our development of a virtual environment simulation framework that allows for physically accurate display of inertia and constraints in any topology, will enable future design and control improvements.

References

[1] J.J. Abbott, G.D. Hager, and A.M. Okamura. Steady-hand teleoperation with virtual fixtures. In *12th IEEE International Workshop on Robot and Human Interactive Communication*, pages 145–151, 2003.

[2] J.J. Abbott and A.M. Okamura. Virtual fixture architectures for telemanipulation. In *IEEE International Conference on Robotics and Automation*, pages 2798–2805, Taipei, Taiwan, 2003.

[3] R. Adams, M. Moreyra, and B. Hannaford. Excalibur, a three-axis force display. In *ASME Winter Annual Meeting Haptics Symposium*, pages 465–474, Nashville, TN, 1999.

[4] B.D. Adelstein and M.J. Rosen. Design and implementation of a force reflecting manipulandum for manual control research. In *Advances in Robotics, ASME DSC*, volume 42, pages 1–12, Anaheim, CA, 1992.

[5] P. Akella, M. Peshkin, J.E. Colgate, W. Wannasuphoprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock. Cobots for the automobile assembly line. In *IEEE International Conference on Robotics and Automation*, pages 728–733, Detroit, MI, 1999.

[6] H. Arai. Controllability of a 3-DOF manipulator with a passive joint under a nonholonomic constraint. In *IEEE International Conference on Robotics and Automation*, pages 3707–3713, Minneapolis, MN, 1996.

[7] G.W. Bachman. *An experimental investigation of the cobot wheel contact patch.* Masters Thesis, Northwestern University, 1997.

[8] D. Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. *Computer Graphics*, 23(3):223–232, 1989.

[9] A.K. Bejczy and J.K. Salisbury. Controlling remote manipulators through kinesthetic coupling. *Computers in Mechanical Engineering*, 2(1):48–58, 1983.

[10] M. Bergamasco, B. Allotta, L. Bosio, L. Ferretti, G. Parrini, G.M. Prisco, F. Salsedo, and G. Sartini. An arm exoskeleton system for teleoperation and virtual environments applications. In *IEEE International Conference on Robotics and Automation*, pages 1449–1454, 1994.

[11] P.J. Berkelman and R.L. Hollis. Lorentz magnetic levitation for haptic interaction: Device design, performance, and integration with physical simulations. *The International Journal of Robotics Research*, 19(7):644–667, 2000.

[12] P.J. Berkelman, R.L. Hollis, and D. Baraff. Interaction with a realtime dynamic environment simulation using a magnetic levitation haptic interface device. In *IEEE International Conference on Robotics and Automation*, Detroit, MI, 1999.

[13] W. Blajer. A geometric unification of constrained system dynamics. *Multibody System Dynamics*, 1(1):3–21, 1997.

[14] W. Blajer. A geometrical interpretation and uniform matrix formulation of multibody system dynamics. *Journal of Applied Mathematics and Mechanics*, 81(4):247–259, 2001.

[15] W. Blajer. Elimination of constraint violation and accuracy aspects in numerical simulation of multibody systems. *Multibody System Dynamics*, 7(3):265–284, 2002.

[16] W.R. Book, R. Charles, H. Davis, and M. Gomes. The concept and implementation of a passive trajectory enhancing robot. In *ASME IMECE DSC*, volume 58, pages 633–638, 1996.

[17] E.S. Boy, E. Burdet, C.L. Teo, and J.E. Colgate. The learning cobot. In *ASME IMECE DSC*, New Orleans, LA, 2002.

[18] E.S. Boy, E. Burdet, C.L. Teo, and J.E. Colgate. Motion guidance experiments with Scooter cobot. In *11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 63–69, 2003.

[19] E.S. Boy, E.L. Teo, and E. Burdet. Collaborative wheelchair assistant. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 1511–1516, 2002.

[20] M.E. Brokowski. *Experimental characterization of a spherical continuously variable transmission.* Ph.D. Dissertation, Northwestern University, 2002.

[21] M.E. Brokowski, S. Kim, J. E. Colgate, R.B. Gillespie, and M. A. Peshkin. Toward improved CVTs: Theoretical and experimental results. In *ASME IMECE DSC*, New Orleans, LA, 2002.

[22] J.M. Brown. *A theoretical and experimental investigation into the factors affecting the Z-width of a haptic display.* Masters Thesis, Northwestern University, 1995.

[23] J.M. Brown and J.E. Colgate. Passive implementation of multibody simulations for haptic display. In *ASME IMECE DSC*, volume 61, pages 85–92, 1997.

[24] J.M. Brown and J.E. Colgate. Minimum mass for haptic display simulations. In *ASME IMECE DSC*, pages 249–256, 1998.

[25] D.T. Burns. *Design of a six degree of freedom haptic interface.* Masters Thesis, Northwestern University, 1996.

[26] C.R. Carignan and K.R. Cleary. Closed-loop force control for haptic simulation of virtual environments. *Haptics-e*, 1(2), 2000.

[27] F.W. Carter. On the action of a locomotive driving wheel. *Proceedings, Royal Society*, A112:151, 1926.

[28] M. Chardon. *The vertical planar cobot.* Masters Thesis, Northwestern University, 2005.

[29] R.A. Charles. *The development of the passive trajectory enhancing robot.* Masters Thesis, Georgia Institute of Technology, 1994.

[30] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrun. *Principles of robot motion: theory, algorithms, and implementations (Intelligent robotics and autonomous agents).* The MIT Press, Cambridge, MA, 2005.

[31] I. Chuckpaiwong. *Reflexive collision avoidance for a novel parallel manipulator.* Masters Thesis, Case Western Reserve University, 2001.

[32] W.J. Chung and Y. Nakamura. Design of the chained form manipulator. In *IEEE International Conference on Robotics and Automation*, pages 455–461, Albuquerque, NM, 1997.

[33] W.J. Chung, Y. Nakamura, and O.J. Sordalen. Prototyping a nonholonomic manipulator. In *IEEE International Conference on Robotics and Automation*, pages 2029–2036, Nagoya, Japan, 1995.

[34] M.B. Cline. *Rigid body simulation with contact and constraints.* Masters Thesis, University of British Columbia, 2002.

[35] C.L. Clover. A control-system architecture for robots used to simulate dynamic force and moment interaction between humans and virtual objects. *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, 29(4):481–493, 1999.

[36] J.E. Colgate. *The control of dynamically interacting systems.* Ph.D. Dissertation, Massachusetts Institute of Technology, 1988.

[37] J.E. Colgate and J.M. Brown. Factors affecting the Z-width of a haptic display. In *IEEE International Conference on Robotics and Automation*, pages 3205–3210, San Diego, CA, 1994.

[38] J.E. Colgate and M.A. Peshkin. U.S. Patent No. 5923139: Passive robotic constraint devices using nonholonomic transmission elements, July 1999.

[39] J.E. Colgate and M.A. Peshkin. U.S. Patent No. 5952796: Cobots, September 1999.

[40] J.E. Colgate and G. Schenkel. Passivity of a class of sampled-data systems: application to haptic interfaces. *Journal of Robotic Systems*, 14(1):37–47, 1997.

[41] D. Constantinescu, S.E. Salcudean, and E.A. Croft. Haptic rendering of topological constraints to users manipulating serial virtual linkages. In *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.

[42] B. d'Andrea Novel, G. Bastin, and G. Campion. Modelling and control of nonholonomic wheeled mobile robots. In *IEEE International Conference on*

Robotics and Automation, pages 1130–1135, Sacramento, CA, 1991.

[43] P. Dehombreux, O. Verlinden, and C. Conti. An implicit multistage integration method including projection for the numerical simulation of constrained multibody systems. *Multibody System Dynamics*, 1(4):405–424, 1997.

[44] B.P. DeJong, E.L. Faulring, J.E. Colgate, M.A. Peshkin, H. Kang, Y.S. Park, and T.F. Ewing. Lessons learned from a novel teleoperation testbed. Invited submission to *Industrial Robot*, September 2005.

[45] Y. Delnondedieu and J. Troccaz. PADyC: A passive arm with dynamic constraints; a prototype with two degrees of freedom. In *Medical Robotics and Computer Assisted Surgery*, pages 173–180, Baltimore, Maryland, 1995.

[46] N. Diolaiti, G. Niemeyer, F. Barbagli, and J.K. Salisbury. A criterion for the passivity of haptic devices. In *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.

[47] N. Diolaiti, G. Niemeyer, F. Barbagli, J.K. Salisbury, and C. Melchiorri. The effect of quantization and Coulomb friction on the stability of haptic rendering. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, 2005.

[48] H. Dou and Y. Chen. Optimal features extraction of noisy sinusoidal signals using two-stage linear least squares fitting. Technical report, Utah State University, Department of Electrical and Computer Engineering, 1998.

[49] J.V. Draper and L.M. Blair. Dual arm work platform performance estimates and telerobot task network simulation. In *American Nuclear Society Seventh Topical Meeting on Robotics and Remote Handling*, Savannah, GA, 1997.

[50] J. Duffy. The fallacy of the modern hybrid control theory that is based on "orthogonal complements" of twist and wrench spaces. *Journal of Robotic Systems*, 7(2):139–144, 1990.

[51] R.E. Ellis, O.M. Ismaeil, and M. Lipsett. Design and evaluation of a high-performance haptic interface. *Robotica*, 14(3):321–327, 1996.

[52] R. Emrich and A.J. Hodgson. A translational-to-rotational continuously variable transmission element for a parallel cobot. In *ASME IMECE DSC*, volume 69, pages 1285–1292, 2000.

[53] S.D. Eppinger. *Modelling robot dynamic performance for endpoint force control.* Ph.D. Dissertation, Massachusetts Institute of Technology, 1988.

[54] E.L. Faulring. *Human interaction with a pallet-jack cobot.* Masters Thesis, Northwestern University, 2002.

[55] E.L. Faulring. LIMS' Projects - Eric Faulring's Cobotic Hand Controller, October 2005. `http://lims.mech.northwestern.edu/projects/handcontroller/`.

[56] E.L. Faulring. A review of modelling and experimentation with the spherical CVT. Technical report, Northwestern University, 2005.

[57] E.L. Faulring and J.E. Colgate. Run-time three-dimensional blend-path generation for cobot constraint surfaces. In *ASME IMECE DSC*, New Orleans, LA, 2002.

[58] E.L. Faulring, J.E. Colgate, and M.A. Peshkin. A high performance 6-DOF haptic cobot. In *IEEE International Conference on Robotics and Automation*, pages 1980–1985, New Orleans, LA, 2004.

[59] E.L. Faulring, J.E. Colgate, and M.A. Peshkin. High performance cobotics. In *IEEE 9th International Conference on Rehabilitation Robotics*, Chicago, IL, 2005.

[60] E.L. Faulring, J.E. Colgate, and M.A. Peshkin. Control and performance of the rotational-to-linear cobotic transmission. Accepted to *IEEE 14th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, March 2006.

[61] E.L. Faulring, K.M. Lynch, J.E. Colgate, and M.A. Peshkin. Haptic interaction with constrained dynamic systems. In *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.

[62] Force Dimension. Delta Haptic Device, February 2004. `http://www.forcedimension.com/fd/avs/home/products/6-dof_delta/`.

[63] A. Frisoli, F. Rocchi, S. Marcheschi, A. Dettori, F. Salsedo, and M. Bergamasco. A new force-feedback arm exoskeleton for haptic interaction in virtual environments. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, 2005.

[64] R. Fuchs, Y. Hasuda, and I. James. Dynamic performance analysis of a full toroidal IVT: A theoretical approach. In *International Continuously Variable and Hybrid Transmission Congress*, 2004.

[65] R.B. Gillespie. Kane's equations for haptic display of multibody systems. *Haptics-e*, 3(2), 2003.

[66] R.B. Gillespie, J.E. Colgate, and M.A. Peshkin. A general framework for cobot control. *IEEE Transactions on Robotics and Automation*, 17(4):391–401, 2001.

[67] R.B. Gillespie, C.A. Moore, M.A. Peshkin, and J.E. Colgate. Kinematic creep in continuously variable transmissions: Traction drive mechanics for cobots. *Journal of Mechanical Design*, 124(4):713–722, 2002.

[68] M.J. Girone, G.C. Burdea, and M. Bouzit. The "Rutgers ankle" orthopedic rehabilitation interface. In *ASME Haptics Symposium*, volume 67, pages 305–312, 1999.

[69] T. Goi, K. Kawakami, E. Yamakawa, and H. Tanaka. Variable rotor speed transmission with high speed traction drive. In *AHS International Annual Forum*, pages 748–759, 1999.

[70] A. Goswami and M. Peshkin. Mechanically implementable accommodation matrices for passive force control. *International Journal of Robotics Research*, 18(8):830–844, 1999.

[71] K. Grace. *Kinematic design of an ophthalmic surgery robot and feature extracting bilateral manipulation.* Ph.D. Dissertation, Northwestern University, 1995.

[72] J. Halling. The relationship between surface texture and rolling resistance of steel. *British Journal of Applied Physics*, 9(11):421–428, 1958.

[73] B.J. Hamrock and D.E. Brewe. Simplified solution for stresses and deformation. *Journal of Lubrication Technology*, 105(2):171–177, 1983.

[74] B.A. Hannaford. A design framework for teleoperators with kinesthetic feedback. *IEEE Transactions on Robotics and Automation*, 5(4):426–434, 1989.

[75] B.A. Hannaford. Stability and performance tradeoffs in bi-lateral telemanipulation. In *IEEE International Conference on Robotics and Automation*, pages 1764–1767, Scottsdale, AZ, 1989.

[76] Haption. Virtuose 6D35-45, February 2004. `http://www.haption.com/v4/eng/index.htm`.

[77] V. Hayward and O.R. Astley. Performance measures for haptic interfaces. In *Robotics Research: The 7th International Symposium*, pages 195–207, 1996.

[78] A.J. Hodgson and R. Emrich. Control of minimally constrained cobots. *Journal of Robotic Systems*, 19(7):299–314, 2002.

[79] J. Hollerbach, D. Grow, and C. Parker. Developments in locomotion interfaces. In *IEEE 9th International Conference on Rehabilitation Robotics*, pages 522–525, 2005.

[80] R.D. Howe, D.A. Kontarinis, and W.J. Peine. Shape memory alloy actuator controller design for tactile displays. In *34th IEEE Conference on Decision and Control*, pages 3540–3544, New Orleans, LA, 1995.

[81] C.W. Hunt. U.S. Patent No. 197472: Counter-shaft for driving machinery, 1877.

[82] T. Imanishi and H. Machida. Development of the POWERTOROS UNIT half toroidal CVT(2): Comparison between half-toroidal and full-toroidal CVTs. *NSK Technical Journal Motion and Control*, (10):1–8, 2001.

[83] L. Jabre, R. McGrew, R.B. Gillespie, and P. Goleski. An assistive cobot for aid in self-care activities. In *IFAC Conference on Mechatronic Systems*, 2002.

[84] S.C. Jacobsen, E.K. Iversen, D.F. Knutti, R.T. Johnson, and K.B. Biggers. Design of the UTAH/M.I.T. dextrous hand. In *IEEE International Conference on Robotics and Automation*, pages 1520–1532, 1986.

[85] S.C. Jacobsen, J.E. Wood, D.F. Knutti, and K.B. Biggers. The UTAH/M.I.T. dextrous hand: Work in progress. *International Journal of Robotics Research*, 3(4):21–50, 1984.

[86] D.E. Johnson and P. Willemsen. Six degree-of-freedom haptic rendering of complex polygonal models. In *11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 229–235, 2003.

[87] K.L. Johnson. *Contact mechanics*. Cambridge University Press, Cambridge, U.K., 1985.

[88] L.D. Joly and C. Andriot. Imposing motion constraints to a force reflecting telerobot through real-time simulation of a virtual mechanism. In *IEEE International Conference on Robotics and Automation*, pages 357–362, 1995.

[89] J.J. Kalker. *Three-dimensional elastic bodies in rolling contact*. Kluwer Academic Publishers, Dordrecht, Boston, 1990.

[90] H. Kang, Y.S. Park, T.F. Ewing, E.L. Faulring, and J.E. Colgate. Visually and haptically augmented teleoperation in D&D tasks using virtual fixtures. In *ANS 10th International Conference on Robotics and Remote Systems for Hazardous Environments*, pages 466–471, Gainesville, FL, 2004.

[91] H. Kazerooni. Contact instability of the direct drive robot when constrained by a rigid environment. *IEEE Transactions on Automatic Control*, 35(6):710–714, 1990.

[92] H. Kazerooni. The human power amplifier technology at the University of California, Berkeley. In *ASME DSC*, pages 605–613, 1995.

[93] H. Kazerooni and M.G. Her. The dynamics and control of a haptic interface device. *IEEE Transactions on Robotics and Automation*, 10(4):453–464, 1994.

[94] H. Kazerooni, J.L. Racine, L. Huang, and R. Steger. On the control of the Berkeley lower extremity exoskeleton (BLEEX). In *IEEE International Conference on Robotics and Automation*, pages 4364–4371, 2005.

[95] A.J. Kelley and S.E. Salcudean. On the development of a force-feedback mouse and its integration into a graphical user interface. In *ASME IMECE DSC*, volume 55, pages 287–294, 1994.

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

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

[98] J. Kim, F.C. Park, Y. Park, and M. Shizuo. Design and analysis of a spherical continuously variable transmission. *Journal of Mechanical Design*, 124(1):21–29, 2002.

[99] S. Kim. *Control of the powered arm cobot and analysis of the rotational CVT*. Ph.D. Dissertation, Northwestern University, 2003.

[100] Y.J. Kim, M.A. Otaduy, M.C. Lin, and D. Manocha. Six-degree-of-freedom haptic display using localized contact computations. In *Tenth Symposium on Haptic Interfaces For Virtual Environment and Teleoperator Systems*, pages 209–216, Orlando, FL, 2002.

[101] K.J. Kuchenbecker, J. Fiene, and G. Niemeyer. Event-based haptics and acceleration matching: Portraying and assessing the realism of contact. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, 2005.

[102] Argonne National Laboratory. Dual arm work platform teleoperated robotics system. Technical Report DOE/EM-0389, Department of Energy, 1998.

[103] D.A. Lawrence, L.Y. Pao, M.A. Salada, and A.M. Dougherty. Quantitative experimental analysis of transparency and stability in haptic interfaces. In *ASME IMECE DSC*, volume 58, pages 441–449, Atlanta, GA, 1996.

[104] C.D. Lee, D.A. Lawrence, and L.Y. Pao. A high-bandwidth force-controlled haptic interface. In *Symposium on Haptic Interfaces for Teleoperation and Virtual Reality, ASME IMECE*, 2000.

[105] P.U. Lee, D.C. Ruspini, and O. Khatib. Dynamic simulation of interactive robotic environment. In *IEEE International Conference on Robotics and Automation*, pages 1147–1152, San Diego, CA, 1994.

[106] G. Liu and Z. Li. A unified geometric approach to modelling and control of constrained mechanical systems. *IEEE Transactions on Robotics and Automation*, 18(4):574–587, 2002.

[107] B. Long, B. Rebsamen, E. Burdet, H. Yu, and C.L. Teo. Elastic path controller for assistive devices. In *27th Annual International Conference of the IEEE EMBS*, Shanghai, China, 2005.

[108] Z.H. Luo, K. Machida, and M. Funaki. Design and control of a Cartesian nonholonomic robot. *Journal of Robotic Systems*, 15(2):85–95, 1998.

[109] H. Machida and Y. Murakami. Development of the POWERTOROS UNIT half toroidal CVT. *NSK Technical Journal Motion and Control*, (9):15–26, 2000.

[110] J. Maples and J. Becker. Experiments in force control of robotic manipulators. In *IEEE International Conference on Robotics and Automation*, pages 695–702, 1986.

[111] J. Martin and J. Savall. Mechanisms for haptic torque feedback. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, 2005.

[112] T.H. Massie and J.K. Salisbury. The PHANTOM haptic interface: A device for probing virtual objects. In *ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, volume DSC 55, pages 295–302, Chicago, IL, 1994.

[113] B.R. Mathewson. *Integration of force strategies and natural admittance control.* Masters Thesis, Case Western Reserve University, 1994.

[114] W.A. McNeely, K.D. Puterbaugh, and J.J. Troy. Six degree-of-freedom haptic rendering using voxel sampling. In *ACM SIGGRAPH*, pages 401–408, 1999.

[115] J.S. Mehling. *Electrical damping as a means for increasing the impedance range of haptic displays.* Masters Thesis, Northwestern University, 2004.

[116] J.S. Mehling, J.E. Colgate, and M.A. Peshkin. Increasing the impedance range of a haptic display by adding electrical damping. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, 2005.

[117] J-P. Merlet. U.S. Patent No. 5053687: Articulated device for use in particular in robotics, 1991.

[118] J.P. Merlet. Direct kinematics and assembly modes of parallel manipulators. *International Journal of Robotics Research*, 11(2):150–162, 1992.

[119] P.A. Millman and J.E. Colgate. Design of a four degree-of-freedom force reflecting manipulandum with a specified force-torque workspace. In *IEEE International Conference on Robotics and Automation*, pages 1488–1493, Sacramento, CA, 1991.

[120] P.J. Milner. U.S. Patent No. 6461268: Continuously variable transmission device, 2002.

[121] M. Minsky, M. Ouh-young, O. Steele, F.P. Brooks, and M. Behensky. Feeling and seeing: Issues in force display. In *Symposium on Interactive 3D Graphics*, pages 235–241, 1990.

[122] B. Mirtich and J. Canny. Impulse-based simulation of rigid bodies. In *Symposium on Interactive 3D Graphics*, pages 181–188, 1995.

[123] S. Miyata and H. Machida. Development of the half-toroidal CVT POWERTOROS UNIT (3): Development of the power-split system. *NSK Technical Journal Motion and Control*, (11):11–18, 2001.

[124] C.A. Moore. *Continuously variable transmission for serial link cobot architectures.* Masters Thesis, Northwestern University, 1997.

[125] C.A. Moore. *Design, construction, and control of a 3-revolute arm cobot.* Ph.D. Dissertation, Northwestern University, 2001.

[126] C.A. Moore, M.A. Peshkin, and J.E. Colgate. Cobot implementation of virtual paths and 3D virtual surfaces. *IEEE Transactions on Robotics and Automation*, 19(2):347–351, 2003.

[127] M. Moreyra and B. Hannaford. A practical measure of dynamic response of haptic devices. In *International Conference on Robotics and Automation*, pages 369–374, Leuven, Belgium, 1998.

[128] D.M. Morris. *Experiments in mechanical assembly using a novel parallel manipulator.* Masters Thesis, Case Western Reserve University, 2001.

[129] R. Muharliamov. On the equations of kinematics and dynamics of constrained mechanical systems. *Multibody System Dynamics*, 6(1):17–28, 2001.

[130] A. Muller and P. Maiber. A lie-group formulation of kinematics and dynamics of constrained MBS and its application to analytical mechanics. *Multibody System Dynamics*, 9(4):311–352, 2003.

[131] A. Nahvi, D.D. Nelson, J.M. Hollerbach, and D.E. Johnson. Haptic manipulation of virtual mechanisms from mechanical CAD designs. In *IEEE International Conference on Robotics and Automation*, pages 375–380, Leuven, Belgium, 1998.

[132] Y. Nakamura, W. Chung, and O.J. Sordalen. Design and control of the nonholonomic manipulator. *IEEE Transactions on Robotics and Automation*, 17(1):48–59, 2001.

[133] J.P. Newall, S. Cowperthwaite, M. Hough, and A.P. Lee. Efficiency modelling in the full toroidal variator: Investigation into optimization of EHL contact conditions to maximize contact efficiency. In *International Continuously Variable and Hybrid Transmission Congress*, 2004.

[134] M.W. Noakes, L.J. Love, and P.D. Lloyd. Telerobotic planning and control for DOE D&D operations. In *IEEE International Conference on Robotics and Automation*, pages 3485–3492, Washington D.C., 2002.

[135] Tribology Laboratory University of Florida. Elliptical Hertzian contact calculator, August 2005. `http://grove.ufl.edu/~wgsawyer/Laboratory/Software/Elliptical/EllipticalContacts.HTML`.

[136] J.D.B. Paines. *Optimization of manual control dynamics for space telemanipulation, impedance control of a force reflecting hand controller*. Masters Thesis, Massachusetts Institute of Technology, 1987.

[137] Y. Park, H. Kang, T.F. Ewing, E.L. Faulring, J.E. Colgate, and M.A. Peshkin. Semi-automatic teleoperation for D&D. In *ANS 10th International Conference on Robotics and Remote Systems for Hazardous Environments*, pages 283–290, Gainesville, FL, 2004.

[138] Y.S. Park, H. Kang, T.F. Ewing, E.L. Faulring, J.E. Colgate, and M.A. Peshkin. Enhanced teleoperation for D&D. In *IEEE International Conference on Robotics and Automation*, pages 3702–3707, New Orleans, LA, 2004.

[139] M.A. Peshkin, J.E. Colgate, W. Wannasuphoprasit, C.A. Moore, R.B. Gillespie, and P. Akella. Cobot architecture. *IEEE Transactions on Robotics and Automation*, 17(4):377–390, 2001.

[140] B. Pohl, M. Simister, R. Smithson, and D. Miller. Configuration analysis of a spherical traction drive CVT/IVT. In *International Continuously Variable and Hybrid Transmission Congress*, 2004.

[141] Quanser. 3 DOF Planar Pantograph, August 2005. `http://www.quanser.com/english/html/products/`.

[142] Quanser. 5 DOF Haptic Wand, August 2005. `http://www.quanser.com/english/html/products/`.

[143] L.B. Rosenberg. Virtual fixtures: perceptual tools for telerobotic manipulation. In *IEEE Virtual Reality Annual International Symposium*, pages 76–82, Seattle, WA, 1993.

[144] L.B. Rosenberg. *Virtual fixtures: Perceptual overlays enhance operator performance in telepresence tasks*. Ph.D. Dissertation, Stanford University, 1994.

[145] D. Ruspini and O. Khatib. A framework for multi-contact multi-body dynamic simulation and haptic display. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1322–1327, Takamatsu, Japan, 2000.

[146] D. Ruspini, K. Kolarov, and O. Khatib. The haptic display of complex graphical environments. In *24th Annual Conference on Computer Graphics and Interactive Techniques*, pages 345–352, 1997.

[147] M.A. Salada. *Fingertip haptics: Preliminary experiments on the perception of slip in haptic feedback*. Ph.D. Dissertation, Northwestern University, 2004.

[148] S.E. Salcudean and T. Vlaar. On the emulation of stiff walls and static friction with a magnetically levitated input-output device. *ASME Journal of Dynamics, Measurement and Control*, 119:127–132, 1997.

[149] K. Salisbury, B. Eberman, M. Levin, and W. Townsend. The design and control of an experimental whole-arm manipulator. In *The Fifth International Symposium on Robotics Research*, pages 233–241, 1990.

[150] F.C. Santos, A.A. Santos Jr., F. Bruni, and L.T. Santos. Evaluation of subsurface contact stresses in railroad wheels using an elastic half-space model. *Journal of the Brazilian Society of Mechanical Science and Engineering*, 26(4):420–429, 2004.

[151] J.J. Santos-Munné. *Extreme joystick: A cobot with stored energy.* Ph.D. Proposal, Mechanical Engineering Department, Northwestern University, 1997.

[152] N. Sarkar, X. Yun, and V. Kumar. Control of mechanical systems with rolling constraints. Application to dynamic control of mobile robots. *International Journal of Robotics Research*, 13(1):55–69, 1994.

[153] N.L. Sharma and D.D. Reid. Rolling as a frictional equilibrium of translation and rotation. *European Journal of Physics*, 20(3):129–136, 1999.

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

[155] SKF. Frictional moment - power loss calculations, October 2005. `http://www.skf.com/portal/skf/home/products?maincatalogue=1&newlink=first&lang=en`.

[156] W. Son, K. Kim, N.M. Amato, and J.C. Trinkle. A generalized framework for interactive dynamic simulation for multirigid bodies. *IEEE Transactions on Systems, Man and Cybernetics*, 34(2):912–924, 2004.

[157] O.J. Sordalen, Y. Nakamura, and W.J. Chung. Design of a nonholonomic manipulator. In *IEEE International Conference on Robotics and Automation*, pages 8–13, San Diego, CA, 1994.

[158] Sensable Technologies. PHANTOM Devices. `http://www.sensable.com/products/phantom_ghost/phantom.asp`.

[159] M.Y.Y. Valiente. *Design of a unicycle cobot.* Masters Thesis, Northwestern University, 2001.

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

[161] P.J. Vermeulen and K.L. Johnson. Contact of non-spherical elastic bodies transmitting tangential forces. *Journal of Applied Mechanics*, 31:338–340, 1964.

[162] Y. Wang. *Workspace analysis of a novel closed-chain manipulator*. Masters Thesis, Case Western Reserve University, 1999.

[163] W. Wannasuphoprasit and S. Chanphat. A novel fluid haptic interface. In *Industrial Technology, IEEE ICIT'02*, pages 359–364, Bangkok, Thailand, 2002.

[164] W. Wannasuphoprasit and T. Cheepsumol. Parallel CVT mechanisms. In *JSAE Annual Congress*, pages 15–18, Yokohama, Japan, 2004.

[165] W. Wannasuphoprasit, R.B. Gillespie, J.E. Colgate, and M.A. Peshkin. Cobot control. In *International Conference on Robotics and Automation*, pages 3571–3576, Albuquerque, NM, 1997.

[166] W. Wannasuphoprasit and S. Sirikasemsuk. On development of a 3D passive manipulator. In *Industrial Technology, IEEE ICIT'02*, pages 1285–1289, Bangkok, Thailand, 2002.

[167] T. Worsnopp. *Design of a unicycle cobot controller*. Masters Thesis, Northwestern University, 2003.

[168] T. Worsnopp, M. Peshkin, J.E. Colgate, and K. Lynch. Controlling the apparent inertia of passive human-interactive robots. In *IEEE International Conference on Robotics and Automation*, pages 1179–1184, New Orleans, LA.

[169] Y. Yokokohji, R.L. Hollis, and T. Kanade. What you can see is what you can feel. In *IEEE Virtual Reality Annual International Symposium*, pages 46–53, 1996.

[170] T. Yoshikawa. Force control of robot manipulators. In *IEEE International Conference on Robotics and Automation*, pages 220–226, San Francisco, CA, 2000.

[171] T. Yoshikawa and H. Ueda. Construction of virtual world using dynamics modules and interaction modules. In *IEEE International Conference on Robotics and Automation*, pages 2358–2364, Minneapolis, MN, 1996.

[172] T. Yoshikawa, Y. Yokokohji, T. Matsumoto, and X.Z. Zheng. Display of feel for the manipulation of dynamic virtual objects. 117:554–558, 1995.

[173] X. Yun and N. Sarkar. Unified formulation of robotic systems with holonomic and nonholonomic constraints. 14(4):640–650, 1998.

[174] N. Zaghzi, A. Carre, M.E.R. Shanahan, E. Papirer, and J. Schultz. A study of spontaneous rubber/metal adhesion. I. The rolling cylinder test. *Journal of Polymer Science: Part B: Polymer Physics*, 25(11):2393–2402, 1987.

[175] C.B. Zilles and J.K. Salisbury. A constraint-based god-object method for haptic display. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 146–151, 1995.

APPENDICES

A. Photographs of the Cobotic Hand Controller

The following are photographs of our implementation of the Cobotic Hand Controller.

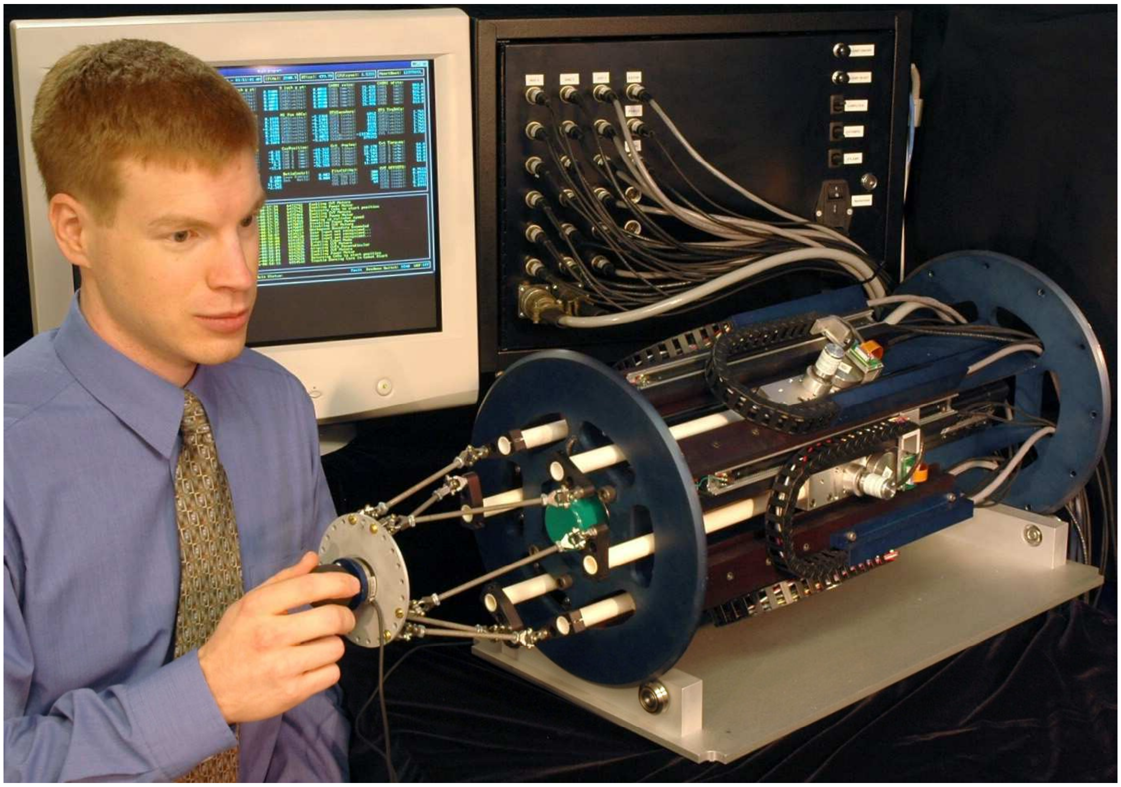


Figure A.1. The Cobotic Hand Controller.

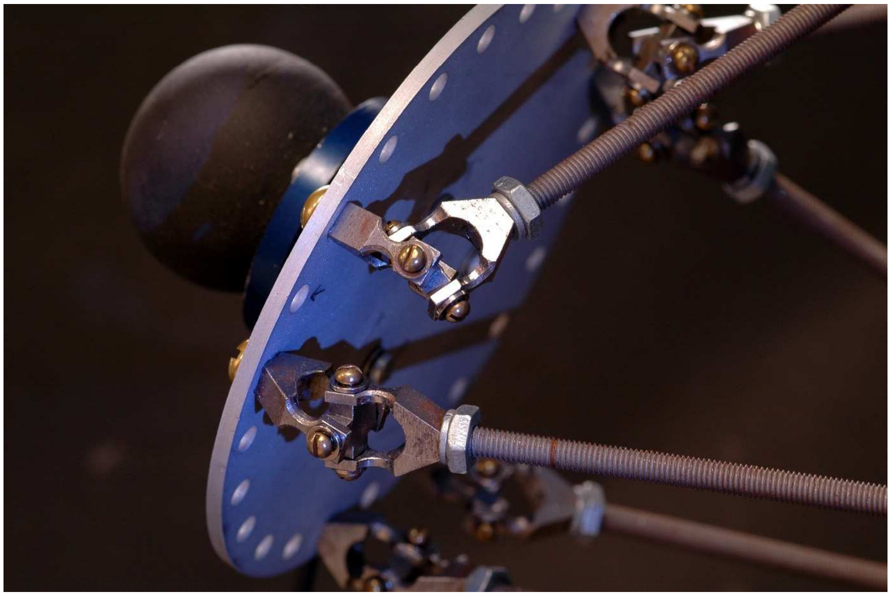


Figure A.2. The end-effector.

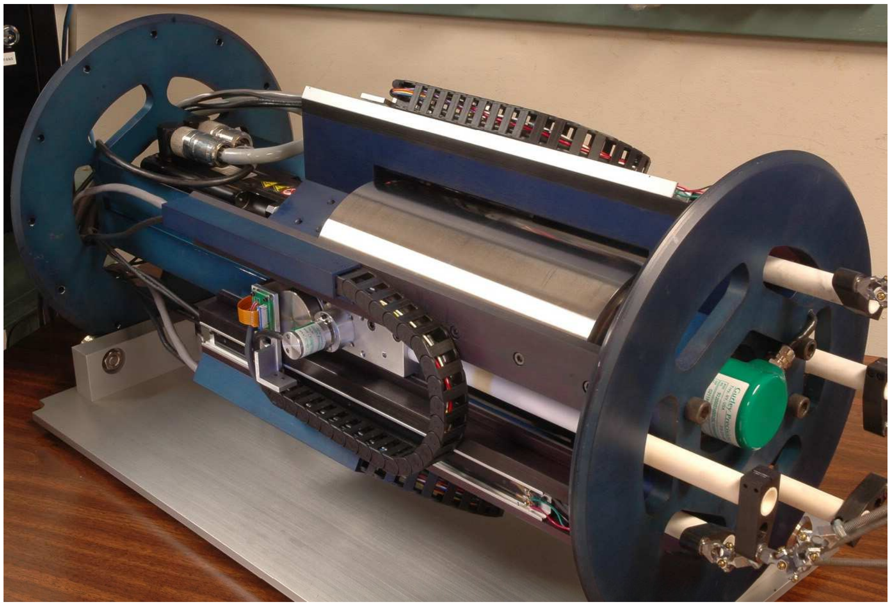


Figure A.3. View of the cylinder with one joint removed.

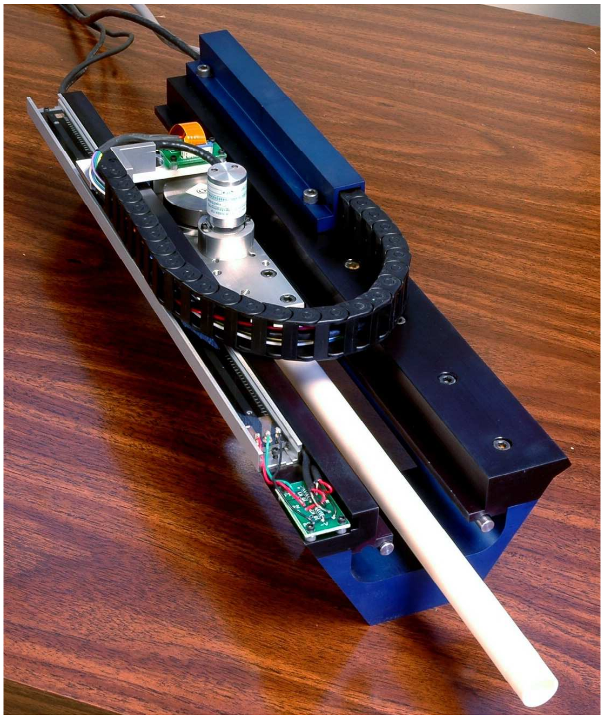


Figure A.4. Top of a single joint.

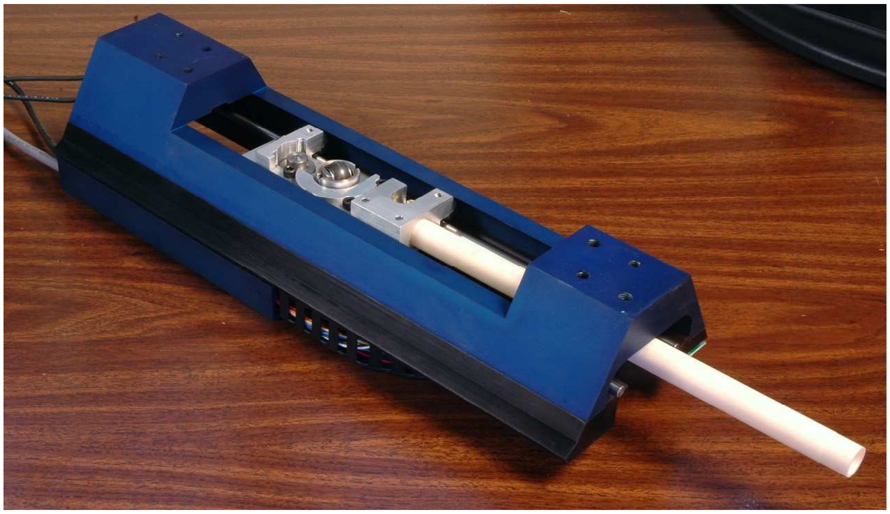


Figure A.5. Bottom of a single joint.

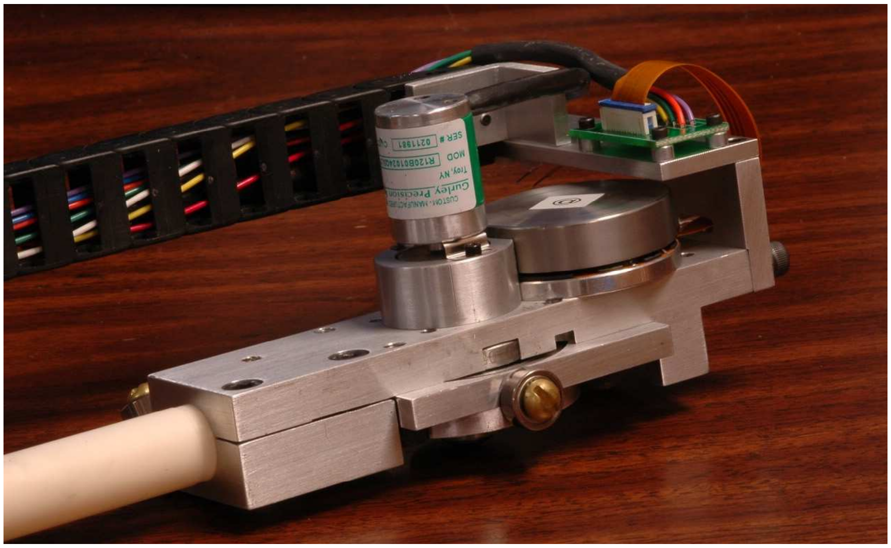


Figure A.6. Top of a single carriage.

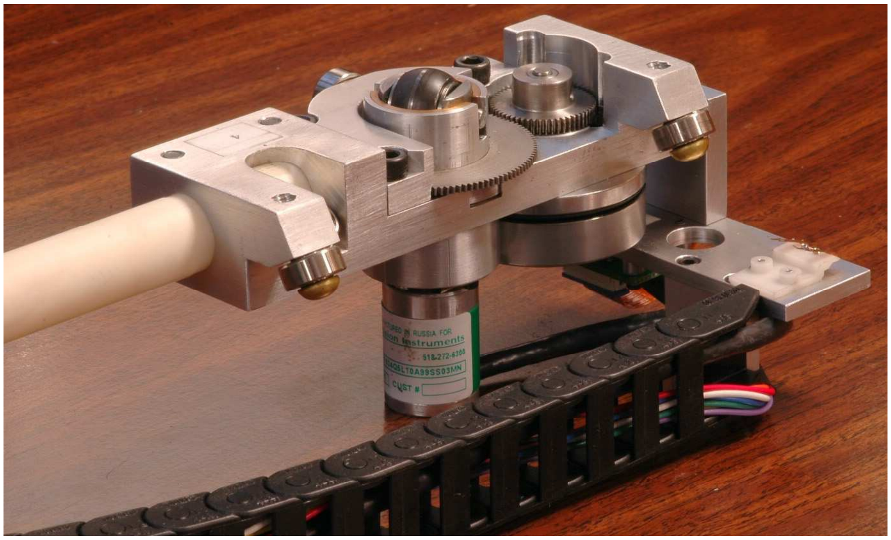


Figure A.7. Bottom of a single carriage.

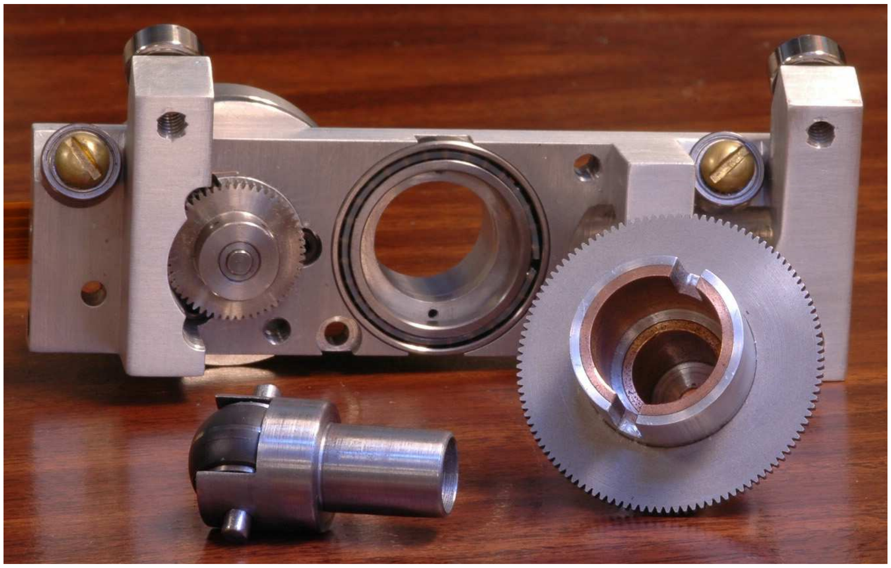


Figure A.8. Bell and carriage components.

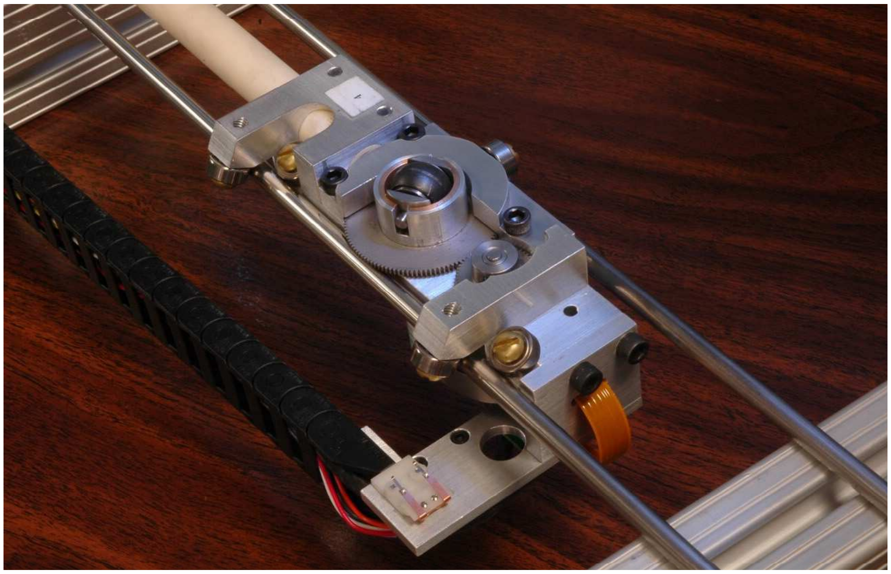


Figure A.9. Joint guideway.

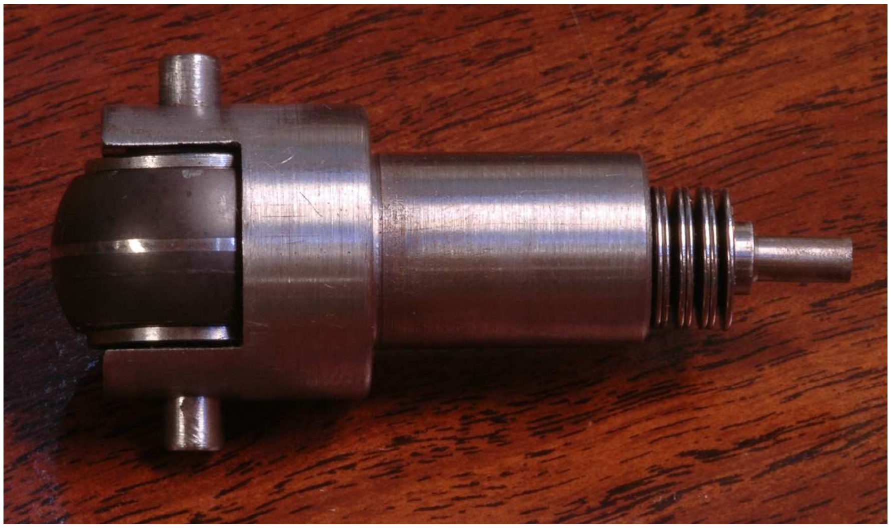


Figure A.10. Belleville springs for preload.

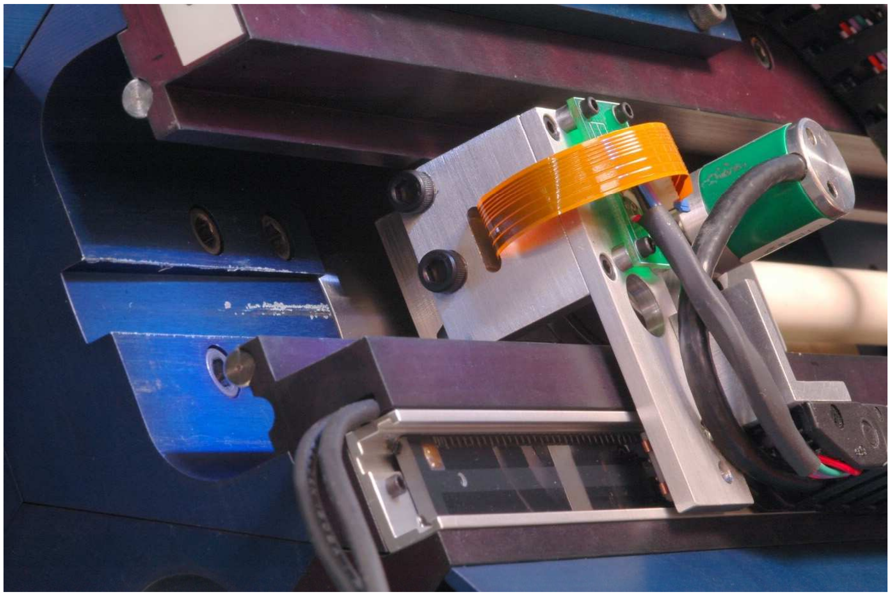


Figure A.11. Ramp for generating preload during insertion of carriage.

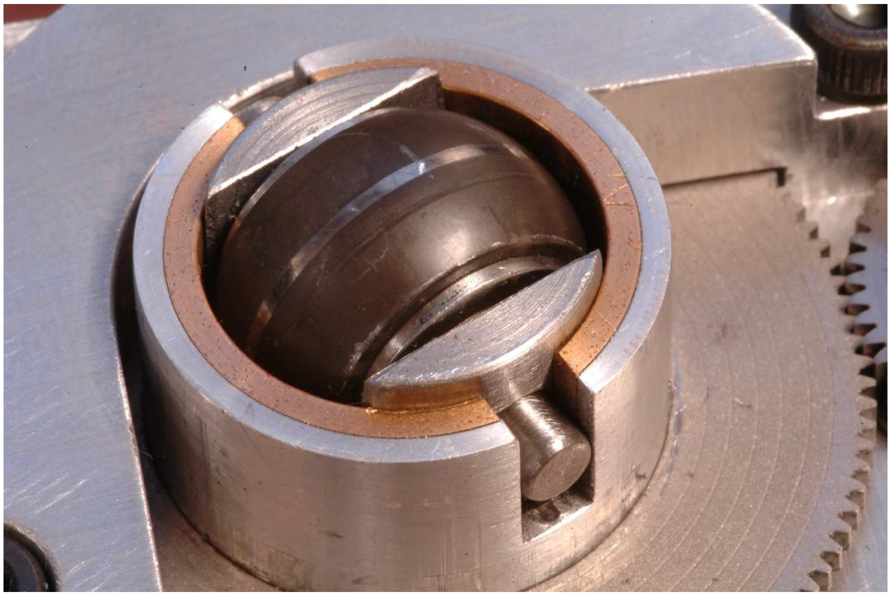


Figure A.12. Shiny patch on worn wheel.

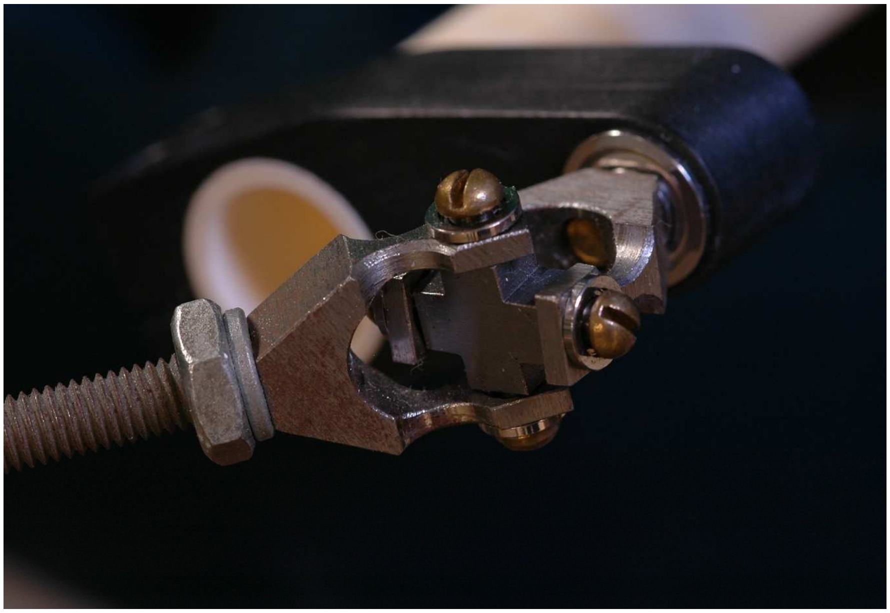


Figure A.13. A universal joint.

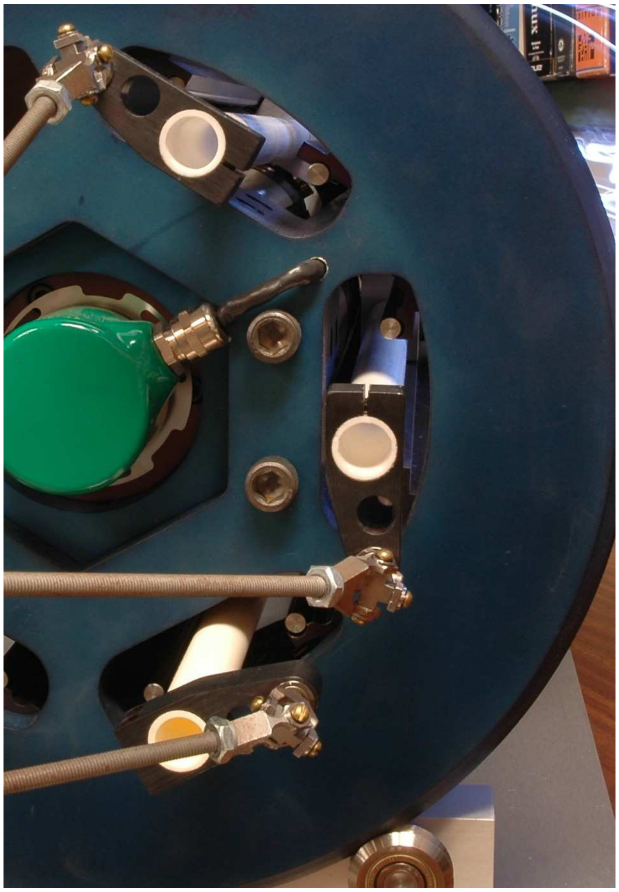


Figure A.14. View down the guideways.

B. Part Number/Vendor List

Table B.1. Part numbers, vendors and process specifications of critical components.

steering encoders Gurley R120B01024Q5L10A99SS03MN	**load cell** ATI Mini40	**wheels** SKF GE12E (approx RC58) from Bearing Headquarters
cylinder encoder Gurley R9220B03600Q5L10B99SS06EA	**force ADC board** NI PCI 6034E	**cylinder** 4140 cold-drawn seamless from Carbon and Alloy Steel Corp. hardened to RC60 by FPM Heat-treating 12 μinch cylindrically ground by Tru-Grind Inc.
power cylinder amplifier Moog T200-410	**linear pot ADC board** NI PCI 6034E	
power cylinder motor Moog G413-804	**linear potentiometers** ETI LCPL300 10K 0.05%	
steering motor amplifiers AMC B12A6	**STG Model II** amplifier enables motor DACs encoder counters interrupt generator	**high alumina ceramic tubes** McMaster-Carr 8746K22
steering motors Maxon EC45 (200142)		**QNX 6.2 computer** 1.53 GHz, 256 MB, 40 GB

C. Software Files

Table C.1. Software core files and their line counts.

Function	**C Files**	**Lines**	**Header Files**	**Lines**
UDP Communication	UDPpackage.c	462	UDPpackage.h	102
Linear Carriage Routines	car.c	267	car.h	159
Cobot Control	cobot.c	224	cobot.h	63
Overall Mode Control	command.c	437	command.h	232
Board Input/Output	control.c	195	control.h	74
Steering Control	cvt.c	294	cvt.h	158
Cylinder Control	cyl.c	189	cyl.h	137
Virtual Environment Runtime	dynamics.c	1585	dynamics.h	125
Virtual Environment Initialization	environment.c	283	environment.h	110
Math Library	ericmath.c	506	ericmath.h	75
Software Filter	filter.c	148	filter.h	45
Force Interpretation	force.c	201	force.h	89
Graphical Interface	gui.c	1489	gui.h	168
Initialization Routine	homing.c	859	homing.h	75
Kinematics	kinematics.c	598	kinematics.h	174
Data Recording	logging.c	100	logging.h	61
Thread Structure	main.c	851	mydefines.h	96
Numerical Recipe Format	nrutil.c	283	nrutil.h	79
Numerical Recipes Algorithms	recipes.c	155	recipes.h	38
Passing Info Between Threads	queue.c	64	queue.h	81
Servo-to-go Driver File	servotogo.c	1084	servotogo.h	262
National Instruments Driver File	adc6034e.cpp	712	adc6034e.h	76
PCI Memory Mapping	osiUserCode.cpp	108		
		11094		**2479**

D. Contact Patch Shape

When first installed, the principal radii of our wheels are equal, but quickly wear an easily measured flat spot of width $2a$. Consequently, the wheel is treated as a cylinder that has a thin rectangular contact with the main cylinder. The pressure distribution of a punch on a flat plate (given in Equation D.1 and Figure D.1) will approximately yield the pressure distribution between a finite length cylinder and an infinite cylinder, $p(x)$, along the line of contact when the rolling axes are parallel (Johnson Equation 2.64) [87].

$$p(x) = \frac{P}{2\pi a(a^2 - x^2)^{1/2}} \tag{D.1}$$

Note that the pressure grows very large near $\frac{x}{a} = \pm 1$. When the wheel is steered away from $\phi = 0$, the edge pressures are relieved since we actually have a wheel on a cylinder and not a punch on a flat plate, with a relatively uniform distribution along the thin rectangular contact. If we average the distribution over angle ϕ, an approximately uniform pressure distribution across the lateral axis of the contact patch results. Finally, assuming a known preload P is linearly distributed along the contact line, and having measured the lateral half-width a, the pressure distribution (Equation D.2) and the half-width b (Equation D.3) in the longitudinal direction can be calculated in order to get an estimate of the ratio of b to a. The Hertzian pressure distribution in the longitudinal direction (for two infinite length cylinders) is given

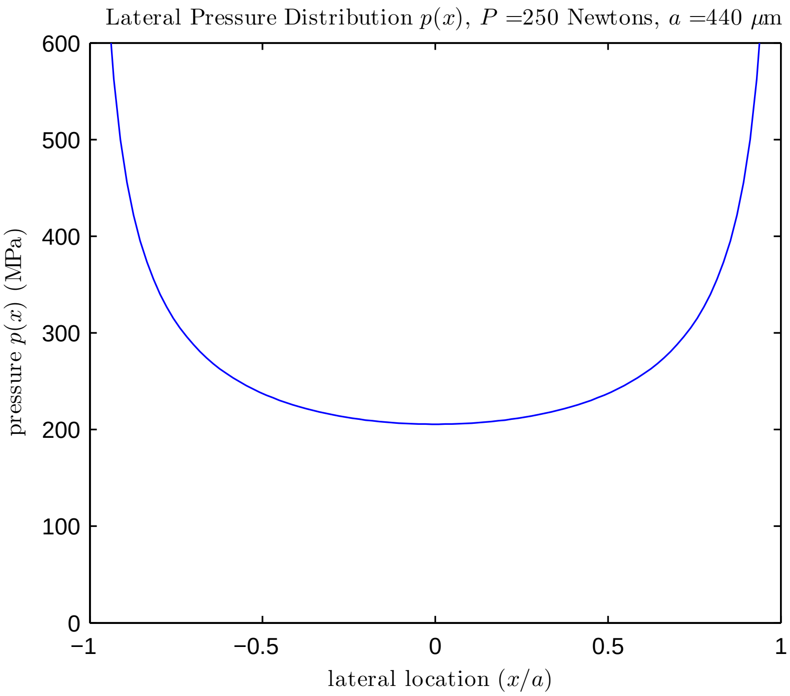


Figure D.1. Lateral pressure distribution for a rigid punch on a flat plate.

by Equation D.2 (Johnson Equation 4.44) [87].

$$p(y) = \left(\frac{P}{\pi a b^2}\right)\left(b^2 - y^2\right)^{1/2} \tag{D.2}$$

This pressure distribution is a semi-circle load with maximum pressure along the original line contact. The pressure goes to zero at $\frac{y}{b} = \pm 1$, and thus most of the load is concentrated along the lateral axis of the contact patch. Given a, and assuming that the contact is between two cylinders, Equation D.3 gives the relationship between

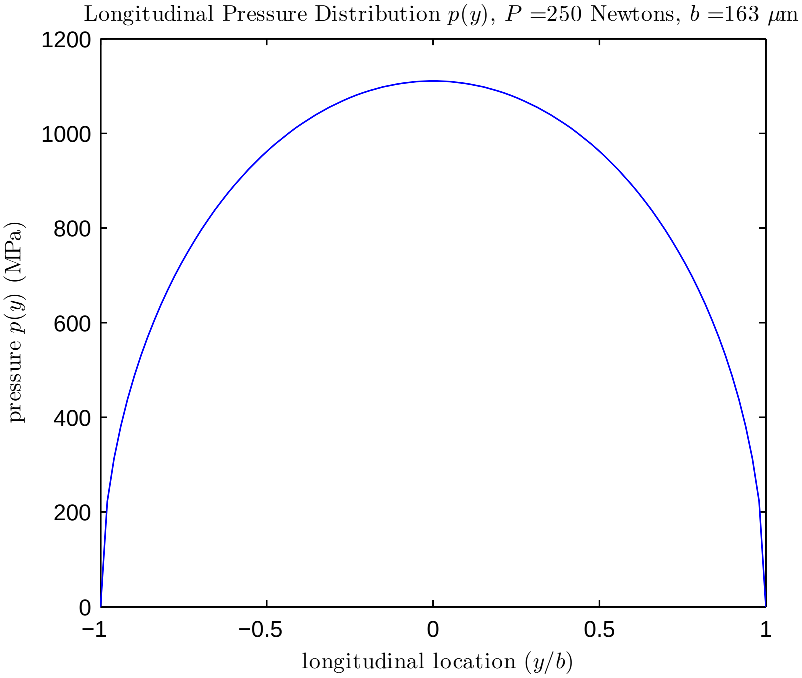


Figure D.2. Longitudinal pressure distribution for line contact between infinitely long cylinders.

b and a [87]. This relationship is depicted in Figure D.3.

$$b = \sqrt{\left(\frac{2P}{a\pi}\right)\left(\frac{2(1-\nu^2)}{E}\right)\left(\frac{1}{r}+\frac{1}{R}\right)^{-1}} \tag{D.3}$$

The wheel and cylinder radii are r and R, respectively, and ν and E are Poisson's ratio and the modulus of elasticity for the two cylinders. We make the simplifying assumption that the wheel is of the same material as the surface it is running on.

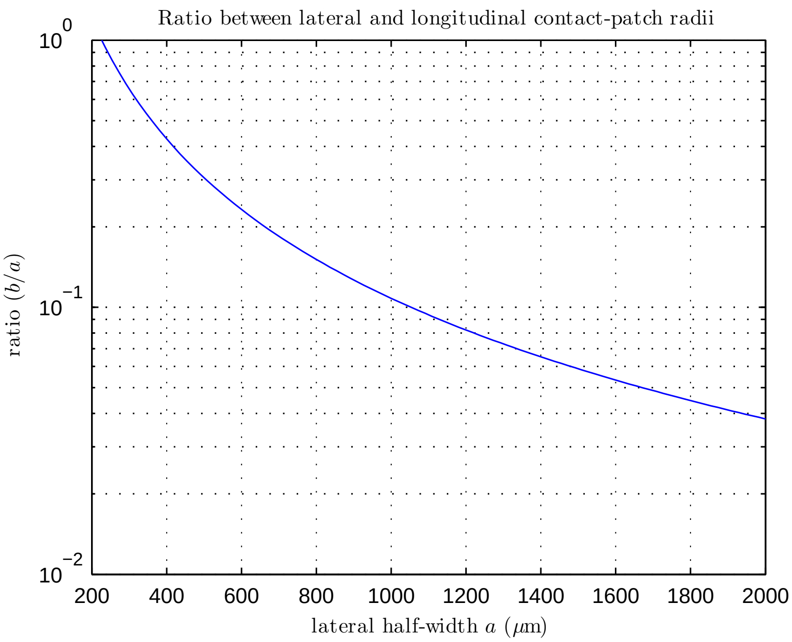


Figure D.3. The ratio b/a is small for large lateral contact-patch width a. The current lateral half-width of the Cobotic Hand Controller wheels is 440μm, yielding a ratio $b/a = 0.37$.

E. Steering Control

Figure E.1 depicts the open loop frequency response between steering motor torque and steering bell velocity. The data were collected by applying a sinusoidal torque at single frequencies, logarithmically spaced between 1 and 400 Hertz and sampled at 2000 Hertz. Inspection of the open loop frequency response reveals a resonance, evident since the plant rolls off at 40 db/decade rather than the 20 dB/decade expected for a simple mass-damper system. The steering plant diagramed in Figure 7.2 yields a relative order two transfer function.

In Figure E.1, Subplot A, the average, maximum and minimum magnitudes of open loop frequency response of the six steering plants is plotted. Subplot B verifies that adequate torque was applied during this sine-sweep in order to effect measurable motion at the encoder on the steering bell.

Figure E.2 depicts the closed loop frequency response between commanded and measured steering bell rotational velocity. Subplots A and B depict the magnitude and phase relationships for the six steering plants. Again, the average, maximum and minimum of the six plants are plotted at each frequency tested. In Subplot C, the velocity amplitude in terms of encoder counts per timestep becomes small as the frequency response rolls off. Attempts to increase the controller gain will lead to the resonance driving the frequency response above unity gain and to the deleterious effects of discrete sampling in time and space. The proportional plus integral controllers were individually tuned for each plant. All six steering plant closed loop responses are shown in Figure E.3. Although the magnitude plots are

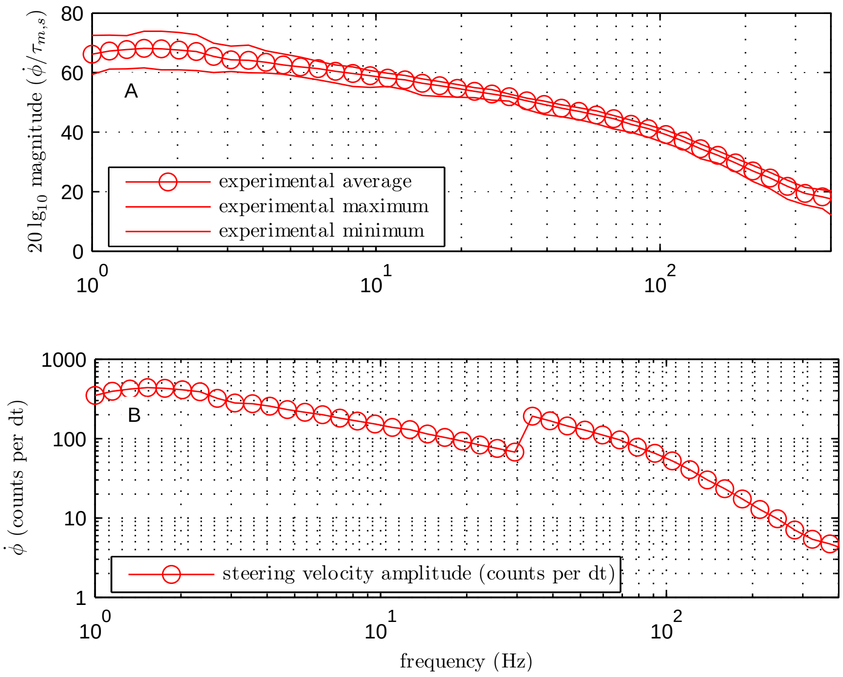


Figure E.1. A. Maximum, minimum and average experimental magnitude of open loop transfer function between steering motor torque and steering bell angular velocity. B. Steering bell encoder counts per timestep for the amplitude of motion resulting from the torque sinesweep. The dip is due to changing the amplitude of the torque input to maximize signal to noise and yet simultaneously avoid spinning the CVTs too fast at lower frequencies.

depressed (- 3 dB) from 1 to 100 Hertz, a resonance still occurs. The joint motion controller, which is wrapped around the steering controllers, ultimately compensates for this less than unity gain at low frequency.

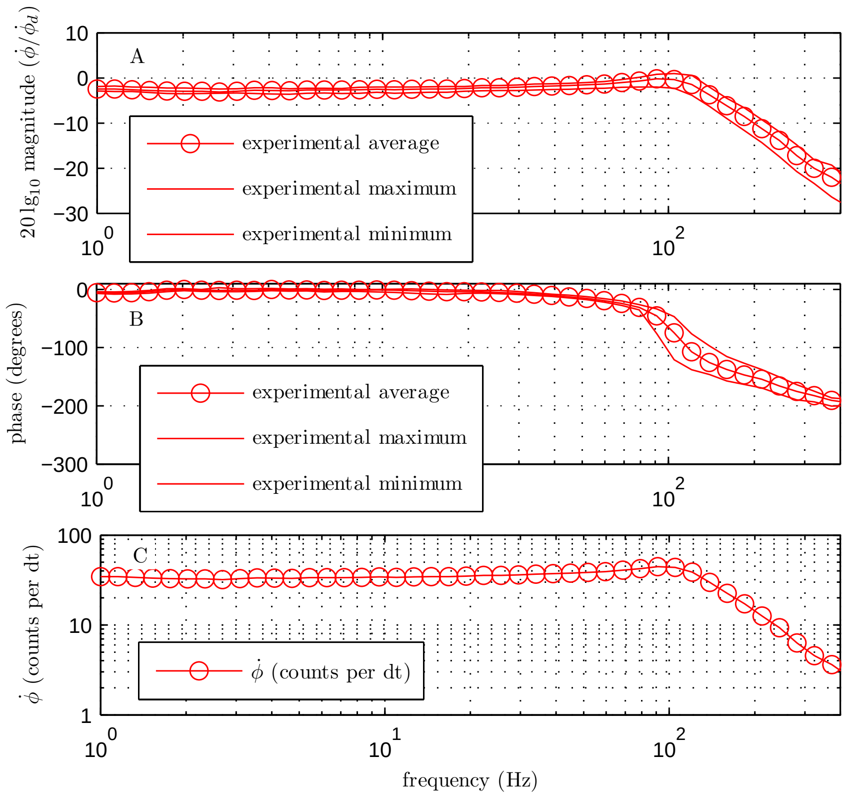


Figure E.2. A. Maximum, minimum and average magnitude of closed loop transfer function for steering velocity plant and controller. B. Maximum, minimum and average phase of closed loop transfer function for steering velocity plant and controller. C. Steering bell encoder counts per timestep for the resulting amplitude of motion: a verification of adequate signal-to-noise to evaluate frequency response.

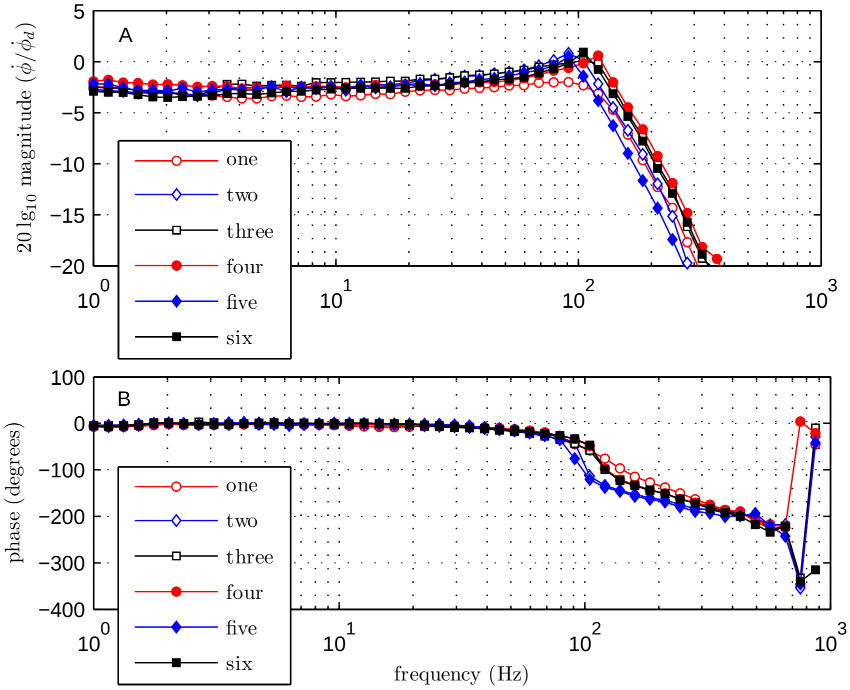


Figure E.3. A. Magnitude of individual steering velocity plant closed loop transfer functions. B. Phase of individual steering velocity plant closed loop transfer functions.

F. Cylinder Control

Figure F.1 depicts the open loop frequency response between cylinder motor torque and cylinder velocity measured at the cylinder encoder. The data was acquired by applying a sinusoidal torque at single frequencies, logarithmically spaced between 1 and 400 Hertz and sampled at 2000 Hertz. Several peculiar features are evident in the open loop plot. The frequency response deviates considerably from the simple mass-damper system that should have a non-resonant corner and role off at 20 dB/decade. Instead, at 3-4 Hz, we see a dip in the frequency-response attributed to the kinking of the rigid coupling between the cylinder and cylinder motor. This is not a linear effect that can be accurately predicted in a Bode diagram, and simply appears at a given frequency depending on the speed of the cylinder. The nominal corner of the mass-damper system is at approximately 10 Hertz. This represents the rolling friction of the wheels in conjunction with the inertia of the cylinder. As we have seen in Section 6.3, the rolling friction is more Coulombic than viscous, so again, a linear model will not work well. At around 100 Hz, the resonance (not kinking) of the rigid coupling, due to its stiffness relative to the geometric mean of the motor and cylinder inertias, is evident.

Figure F.2 depicts the closed loop frequency response between commanded and measured cylinder velocity. Subplots A and B depict the magnitude and phase relationships for the experimental data for the actual plant and controller. In Subplot C the velocity amplitude in terms of encoder counts per timestep becomes small as the frequency response rolls off. In conjunction with this, the frequency has increased,

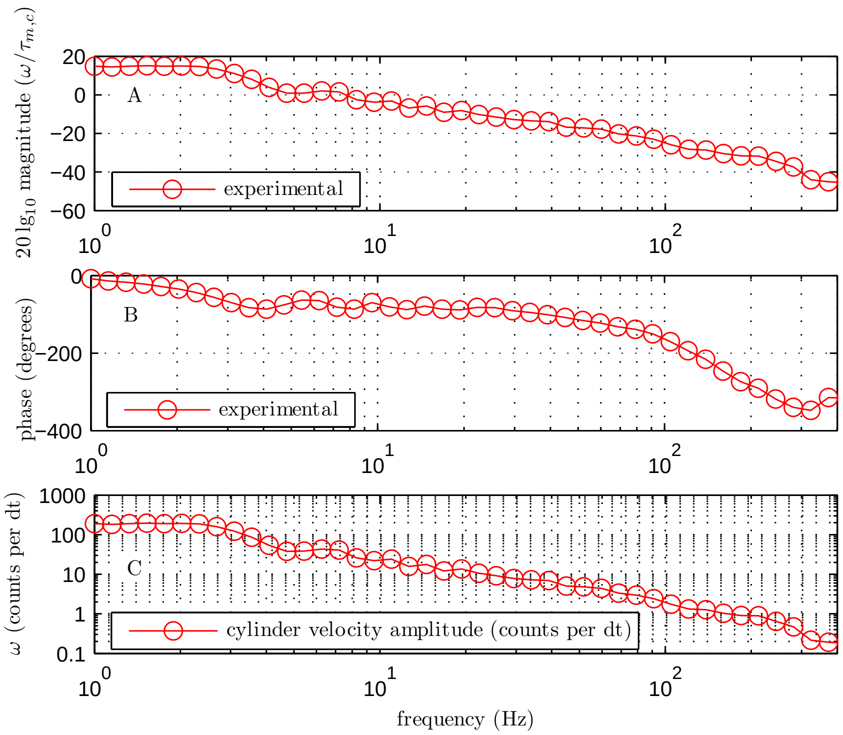


Figure F.1. A. Experimental magnitude of open loop transfer function between cylinder motor torque and cylinder angular velocity. B. Experimental phase of open loop transfer function between cylinder motor torque and cylinder angular velocity. C. Cylinder encoder counts per timestep for the amplitude of motion resulting from torque sine-sweep.

which would require more counts per timestep in order to accurately measure the signal. Attempts to increase the controller gain will lead to the deleterious effects of discrete sampling in time and space. Our simple proportional plus integral controller, as shown in Figure 3.6, keeps the experimental magnitude at unity up to 70 Hertz, although a phase lag of 90 degrees manifests by 50 Hertz.

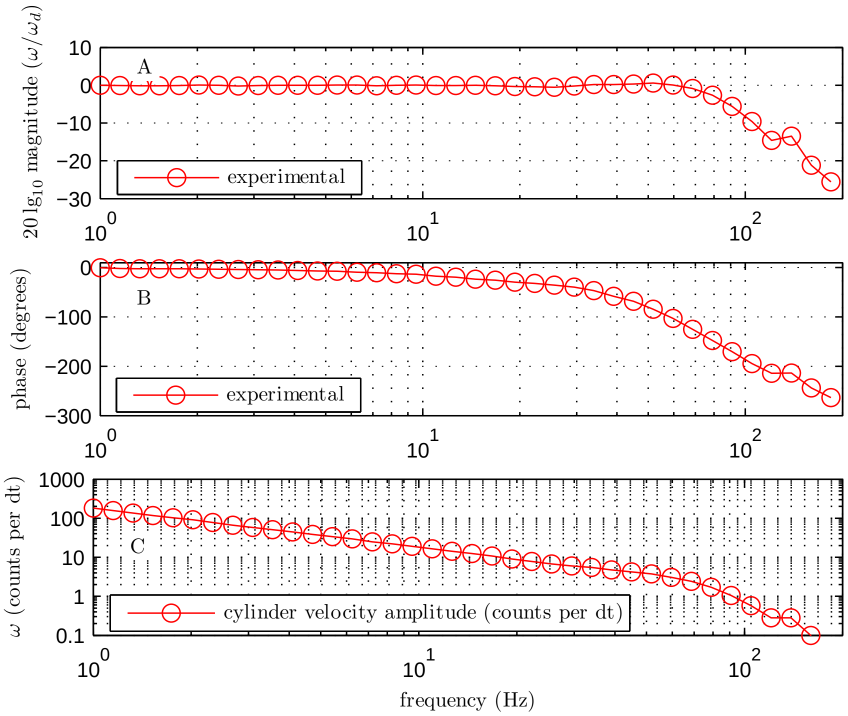


Figure F.2. A. Experimental magnitude of closed loop transfer function between desired cylinder velocity and actual cylinder angular velocity. B. Experimental phase of closed loop transfer function between desired cylinder velocity and actual cylinder angular velocity. C. Cylinder encoder counts per timestep for the amplitude of motion resulting from commanded velocity sine-sweep.

G. Load Cell

For the purpose of this section only, we break apart the generalized force, f_x, into translational and rotational components, $[t'^T, r'^T]^T$, where $t' = (f_{x1}, f_{x2}, f_{x3})$ are the three translational force components of the generalized force and $r' = (f_{x4}, f_{x5}, f_{x6})$ are the three rotational components of the generalized force.

G.1. Gravity compensation of induced forces

A known mass, m_g, distal to the load cell applies a task space force due to the acceleration of gravity, g. The translational components of this force are $t'_{x,g} = [0, -m_g g, 0]^T$. We can move $t'_{x,g}$ from the task frame X to the manipulandum frame M, using the rotation matrix $R(x)$ (Equation G.1).

$$t'_{M,g} = R(x)^T t'_{x,g} \tag{G.1}$$

This force can then be moved to the force sensor frame S from the manipulandum frame M by a second rotation matrix $R(m)$ where m are the coordinates of the sensor frame origin in the manipulandum frame (Equation G.2).

$$t'_{s,g} = R(m)^T t'_{M,g} \tag{G.2}$$

The nominal sensor bias, $t'_{s,o}$, and the gravity load bias, $t'_{s,g}$, are then removed from the raw reading, $t'_{s,r}$, to yield the gravity compensated sensor frame force t'_s (Equation

G.3).

$$t'_s = t'_{s,r} - t'_{s,o} - t'_{s,g} \tag{G.3}$$

Rotating back to task frame yields the compensated task frame force t'_x (Equation G.4).

$$t'_x = R(x)R'(m)(t'_{s,r} - t'_{s,o} - t'_{s,g}) \tag{G.4}$$

Sensor bias is established by putting the cobot in a configuration $x = 0$ where gravity load is trivially known, and applied force, t'_s, is zero since the operator is not touching the end-effector. Thus, $t'_{s,o} = t'_{s,r} - t'_{s,g}$ is established via a time average during an initialization routine.

G.2. Gravity compensation of induced torques

The compensated sensor frame torques, r'_s, must have the nominal sensor bias, $r_{s,o}$, removed, the torques due to distal mass gravity effects, $r'_{s,g}$, removed, and torques due to forces applied at the tool frame, $r'_{s,f}$, removed (Equations G.5 and G.6).

$$r'_{s,g} = v_{s,m} \times t'_{s,g} \tag{G.5}$$

$$r'_{s,f} = v_{s,g} \times t'_s \tag{G.6}$$

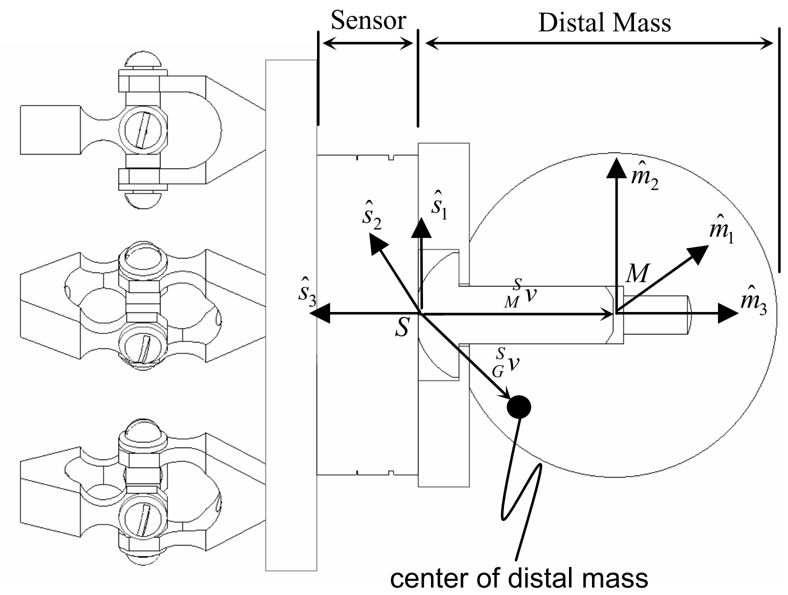


Figure G.1. Definition of force frame terminology.

$v_{s,g}$ is the sensor frame vector from the sensor frame to the center of mass of the distal load. $v_{s,m}$ is the sensor frame vector from the sensor frame to the manipulandum frame. The total compensated sensor frame torques are given by Equation G.7.

$$r'_s = r'_{s,r} - r'_{s,o} - r'_{s,g} - r'_{s,f} \tag{G.7}$$

We can then rotate this back, via the manipulandum frame, and then back to task frame to yield Equation G.8.

$$r'_x = R(x)R(m)r'_s \tag{G.8}$$

Sensor bias is established by putting the cobot in a configuration where the gravity load, $r'_{s,g}$, is trivially known, and applied torque, r'_s, and torque due to force, $r'_{s,f}$, are zero since the operator is not touching the end-effector. Thus, $r'_{s,o} = r'_{s,r} - t_{s,g}$ is established via a time average during an initialization routine.

G.3. Dynamic compensation

While we do compensate the load cell for the weight of the mass distal to it, we do not compensate for the inertial (dynamic) affects.

H. Euler Angles

Equations H.1-H.4 are utilized for the actual implementation of $R(q)$ (rotation from body to world in virtual environment) and $R(x)$ (rotation from cobot end-effector frame M to task space frame X). $R(q)$ consists of a rotation of q_6 about the q_3 body-fixed axis first, followed by a rotation of q_5 about the q_2 body-fixed axis, then followed by a rotation of q_4 about the q_1 body-fixed axis. The equivalent holds true for $R(x)$. Figure H.1 depicts the layout of frames X and Q relative to the Cobotic Hand Controller.

$$\text{(H.1)} \qquad R = \begin{bmatrix} cq_5cq_6 & -cq_5sq_6 & sq_5 \\ sq_4sq_5cq_6 + cq_4sq_6 & cq_4cq_6 - sq_4sq_5sq_6 & -sq_4cq_5 \\ -cq_4sq_5cq_6 + sq_4sq_6 & sq_4cq_6 + cq_4sq_5sq_6 & cq_4cq_5 \end{bmatrix}$$

The skew-symmetric matrix of body angular velocities is given by Equation H.2.

$$\text{(H.2)} \qquad R^T\dot{R} = \begin{bmatrix} 0 & -sq_5\dot{q}_4 - \dot{q}_6 & -cq_5sq_6\dot{q}_4 + cq_6\dot{q}_5 \\ sq_5\dot{q}_4 + \dot{q}_6 & 0 & -cq_5cq_6\dot{q}_4 - sq_6\dot{q}_5 \\ cq_5sq_6\dot{q}_4 - cq_6\dot{q}_5 & cq_5cq_6\dot{q}_4 + sq_6\dot{q}_5 & 0 \end{bmatrix}$$

The body frame angular velocities can be pulled out via Equation H.3.

$$\text{(H.3)} \qquad {}^b\hat{\omega} = R^T\dot{R} = \begin{bmatrix} 0 & -{}^b\omega_3 & {}^b\omega_2 \\ {}^b\omega_3 & 0 & -{}^b\omega_1 \\ -{}^b\omega_2 & {}^b\omega_1 & 0 \end{bmatrix}$$

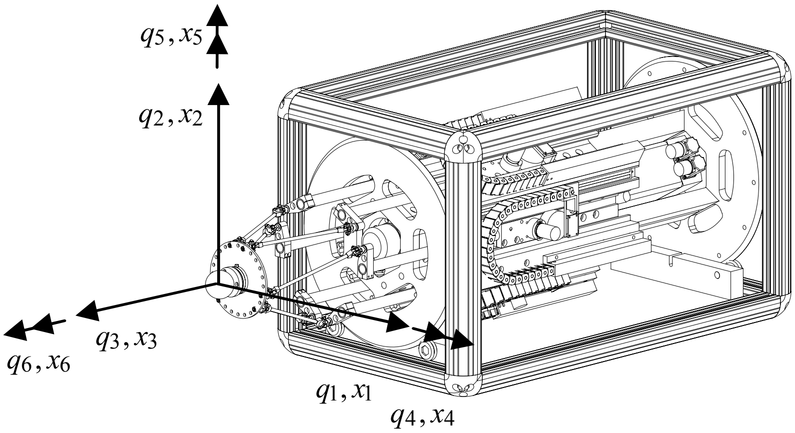


Figure H.1. The coordinates q and x describe positions in frames Q and X, whose origins lie at the center of the Cobotic Hand Controller's workspace (they are not end-effector fixed as might be implied from this schematic). For small deviation (in orientation) from the nominal position, the fourth, fifth and sixth coordinates are rotations about the first, second and third translational axes. Otherwise, they are an Euler angle parameterization.

The vector of the body frame translational velocities are given by Equation H.4.

$$ {}^{b}v = R^T \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \tag{H.4} $$

I. Inertia

For the virtual environment simulations of a rigid body (versus a linkage) the inertia matrix $M(q)$ and Coriolis matrix $C(q,\dot{q})$ are provided via Equations I.1 and I.2. These matrices are given in terms of the Euler angle parameterization as described in Appendix H. m_t is the translational inertia. For brevity here, the rotational inertia principle mass moments of the body frame inertia J_r are assumed identical $(J_{r,q_1q_1} = J_{r,q_2q_2} = J_{r,q_3q_3})$. Otherwise, Equations I.1 and I.2 are much more complex.

$$M(q) = \begin{bmatrix} m_t & 0 & 0 & 0 & 0 & 0 \\ 0 & m_t & 0 & 0 & 0 & 0 \\ 0 & 0 & m_t & 0 & 0 & 0 \\ 0 & 0 & 0 & J_{r,q_1q_1} & 0 & J_{r,q_1q_1}\sin(q_5) \\ 0 & 0 & 0 & 0 & J_{r,q_1q_1} & 0 \\ 0 & 0 & 0 & J_{r,q_1q_1}\sin(q_5) & 0 & J_{r,q_1q_1} \end{bmatrix} \tag{I.1}$$

$$C(q,\dot{q}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_6 & \frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_5 \\ 0 & 0 & 0 & -\frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_6 & 0 & -\frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_4 \\ 0 & 0 & 0 & \frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_5 & \frac{1}{2}J_{r,q_1q_1}\cos(q_5)\dot{q}_4 & 0 \end{bmatrix} \tag{I.2}$$

J. Constraints and Parameterizations Implemented

Several rigid body constraint scenarios in addition to those presented in Chapter 8, were also implemented. The Pfaffian constraints, $A(q)\dot{q} = 0$, and parametric descriptions, $x = \psi(z)$, of the examples in Chapter 8 and the additionally implemented bilateral constraint examples are reproduced in this appendix. Since virtual environment space Q and task space frame X are equivalent for simulation of a rigid body with the six-degree-of-freedom Hand Controller, $A'(x)$ and $A(q)$ are identical. The simulation of a linkage (i.e., where $A'(x)$ and $A(q)$ are not equivalent) is not reproduced here as it is fully defined as implemented on the Cobotic Hand Controller in the example of Section 8.6.3.

J.1. Translation but no rotation

$$A(q) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.1}$$

$$\dot{A}(q) = [0] \tag{J.2}$$

$$x = \psi(z) = [\ z_1 \ \ z_2 \ \ z_3 \ \ 0 \ \ 0 \ \ 0\ ]^T \tag{J.3}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{J.4}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.5}$$

J.2. Disk rolling and turning on $x_1 - x_3$ plane

A disk with radius r_d rolls and turns on the $x_1 - x_3$ plane. The special coordinate system utilized in Section 8.5.1 is replaced here by the common coordinate system of all implemented examples.

$$A(q) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & cq_6sq_4sq_5 + cq_4sq_6 & -cq_4cq_5cq_6 & cq_6sq_4 + cq_4sq_5sq_6 \\ 0 & 1 & 0 & r_dc\theta_dcq_5cq_6 & r_dc\theta_dsq_6 & 0 \\ 1 & 0 & 0 & -r_ds\theta_dcq_5cq_6 & -r_ds\theta_dsq_6 & 0 \end{bmatrix} \tag{J.6}$$

$\theta_d = \text{atan2}(cq_6sq_4sq_5 + cq_4sq_6, cq_5cq_6)$

$$\dot{A}(q) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s_2 & s_5 & s_6 \\ 0 & 0 & 0 & s_7 & s_8 & 0 \\ 0 & 0 & 0 & s_9 & s_{10} & 0 \end{bmatrix} \tag{J.7}$$

$s_1 = (cq_6sq_4sq_5 + cq_4sq_6)(-cq_6sq_5\dot{q}_5 - cq_5sq_6\dot{q}_6)$

$s_2 = cq_4cq_6sq_5\dot{q}_4 - sq_4sq_6\dot{q}_4 + cq_5cq_6sq_4\dot{q}_5 + cq_4cq_6\dot{q}_6 - sq_4dq_5sq_6\dot{q}_6$

$s_3 = (cq_5cq_6)^2 + (cq_6sq_4sq_5 + cq_4sq_6)^2$

$s_4 = (cq_5cq_6s_2 - s_1)/(s_3)$

$s_5 = cq_5cq_6sq_4\dot{q}_4 + cq_4cq_6sq_5\dot{q}_5 + cq_4cq_5sq_6\dot{q}_6$

$s_6 = cq_4cq_6\dot{q}_4 - sq_4sq_5sq_6\dot{q}_4 + cq_4cq_5sq_6\dot{q}_5 + cq_4cq_6sq_5\dot{q}_6 - sq_4sq_6\dot{q}_6$

$s_7 = -r_d(cq_5cq_6s_4s\theta_d + cq_6sq_5\dot{q}_5c\theta_d + cq_5sq_6\dot{q}_6c\theta_d)$

$s_8 = -r_d(sq_6s_4s\theta_d - cq_6\dot{q}_6c\theta_d)$

$$s_9 = -r_d(cq_5cq_6s_4c\theta_d + -cq_6sq_5\dot{q}_5s\theta_d - cq_5sq_6\dot{q}_6s\theta_d)$$

$$s_{10} = -r_d(sq_6s_4c\theta_d + cq_4\dot{q}_6s\theta_d)$$

$$x = \psi(z) = [\ z_1 \quad z_2 \quad -r_d \quad z_3 \quad \arcsin(\tan(z_3))\tan(z_4) \quad z_4\ ]^T \tag{J.8}$$

$$\frac{\partial\psi}{\partial z} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{\sec^2(z_3)\tan(z_4)}{\sqrt{1-\tan^2(z_3)\tan^2(z_4)}} & \frac{\tan(z_3)\sec^2(z_4)}{\sqrt{1-\tan^2(z_3)\tan^2(z_4)}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.9}$$

$$\begin{aligned} \frac{\partial^2 x_5}{\partial z_3\partial z_3} &= \tan(z_3)\sec^4(z_3)\tan(z_4)\sec^2(z_4)\frac{(1+2\cos(2z_3)+\cos(2z_4))}{2(1-\tan^2(z_3)\tan^2(z_4))^{\frac{3}{2}}} \\ \frac{\partial^2 x_5}{\partial z_3\partial z_4} = \frac{\partial^2 x_5}{\partial z_4\partial z_3} &= \frac{\sec^2(z_3)\sec^2(z_4)}{(1-\tan^2(z_3)\tan^2(z_4))^{\frac{3}{2}}} \\ \frac{\partial^2 x_5}{\partial z_4\partial z_4} &= \tan(z_3)\sec^2(z_3)\tan(z_4)\sec^4(z_4)\frac{(1+\cos(2z_3)+2\cos(2z_4))}{2(1-\tan^2(z_3)\tan^2(z_4))^{\frac{3}{2}}} \end{aligned} \tag{J.10}$$

J.3. $x_1 - x_2$ plane without in-plane rotation

$$A(q) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.11}$$

$$\dot{A}(q) = [0] \tag{J.12}$$

$$x = \psi(z) = [\ z_1 \quad z_2 \quad 0 \quad 0 \quad 0 \quad 0\ ]^T \tag{J.13}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{J.14}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.15}$$

J.4. $x_1 - x_2$ plane with in-plane rotation

(J.16)
$$A(q) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

(J.17)
$$\dot{A}(q) = [0]$$

(J.18)
$$x = \psi(z) = [\ z_1 \quad z_2 \quad 0 \quad 0 \quad 0 \quad z_3\ ]^T$$

(J.19)
$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(J.20)
$$\frac{\partial^2 \psi}{\partial z^2} = [0]$$

J.5. Screw motion along x_3 axis

A screw which has pitch ι is oriented along the x_3 axis.

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & \iota & 0 & 0 & -1 \end{bmatrix} \tag{J.21}$$

$$\dot{A}(q) = [0] \tag{J.22}$$

$$x = \psi(z) = [\ 0 \ \ 0 \ \ z_1 \ \ 0 \ \ 0 \ \ \iota z_1 \ ]^T \tag{J.23}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ \iota \end{bmatrix} \tag{J.24}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.25}$$

J.6. Translation along x_3 axis

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.26}$$

$$\dot{A}(q) = [0] \tag{J.27}$$

$$x = \psi(z) = [\ 0 \ \ 0 \ \ z_1 \ \ 0 \ \ 0 \ \ 0\ ]^T \tag{J.28}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{J.29}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.30}$$

J.7. Free motion

$$A(q) = [0] \tag{J.31}$$

$$\dot{A}(q) = [0] \tag{J.32}$$

$$x = \psi(z) = [\ z_1 \quad z_2 \quad z_3 \quad z_4 \quad z_5 \quad z_6\ ]^T \tag{J.33}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.34}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.35}$$

J.8. Rotation but no translation

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{J.36}$$

$$\dot{A}(q) = [0] \tag{J.37}$$

$$x = \psi(z) = [\ 0 \ \ 0 \ \ 0 \ \ z_1 \ \ z_2 \ \ z_3\ ]^T \tag{J.38}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{J.39}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.40}$$

J.9. Disk sliding and spinning on sphere

A disk is sliding along the surface of a sphere of radius r_s, and also spins about the contact normal. The special coordinate system utilized in Section 8.6.2 is replaced here by the common coordinate system of all implemented examples.

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & -r_s\cos(q_5) & 0 \\ 0 & 1 & 0 & r_s\cos(q_4)\cos(q_5) & -r_s\sin(q_4)\sin(q_5) & 0 \\ 0 & 0 & 1 & r_s\sin(q_4)\cos(q_5) & r_s\cos(q_4)\sin(q_5) & 0 \end{bmatrix} \tag{J.41}$$

$$\dot{A}(q) = \left[\begin{matrix} 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & -r_s\sin(q_4)\cos(q_5)\dot{q}_4 - r_s\cos(q_4)sin(q_5)\dot{q}_5 & \dots \\ 0 & 0 & 0 & r_s\cos(q_4)\cos(q_5)\dot{q}_4 - r_s\sin(q_4)sin(q_5)\dot{q}_5 & \dots \end{matrix}\right. \tag{J.42}$$

$$\left.\begin{matrix} \dots & r_s\sin(q_5)\dot{q}_5 & 0 \\ \dots & -r_s\cos(q_4)\sin(q_5)\dot{q}_4 - r_s\sin(q_4)cos(q_5)\dot{q}_5 & 0 \\ \dots & -r_s\sin(q_4)\sin(q_5)\dot{q}_4 + r_s\cos(q_4)cos(q_5)\dot{q}_5 & 0 \end{matrix}\right]$$

$$x = \psi(z) = [\ r_s\sin(z_2) \quad -r_s\sin(z_1)\cos(z_2) \quad r_s\cos(z_1)\cos(z_2) \quad z_1 \quad z_2 \quad z_3\ ]^T \tag{J.43}$$

$$\frac{\partial\psi}{\partial z} = \begin{bmatrix} -r_s\cos(z_1)\cos(z_2) & r_s\cos(z_2) & 0 \\ 0 & r_s\sin(z_1)\sin(z_2) & 0 \\ -r_s\sin(z_1)\cos(z_2) & -r_s\cos(z_1)\sin(z_2) & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{J.44}$$

$$\begin{aligned}
\frac{\partial^2 x_1}{\partial z_2 \partial z_2} &= -r_s \sin(z_2) \\
\frac{\partial^2 x_2}{\partial z_1 \partial z_1} = \frac{\partial^2 x_2}{\partial z_2 \partial z_2} &= r_s \sin(z_1) \cos(z_2) \\
\frac{\partial^2 x_2}{\partial z_1 \partial z_2} = \frac{\partial^2 x_2}{\partial z_2 \partial z_1} &= r_s \cos(z_1) \sin(z_2) \\
\frac{\partial^2 x_3}{\partial z_1 \partial z_1} = \frac{\partial^2 x_3}{\partial z_2 \partial z_2} &= -r_s \cos(z_1) \cos(z_2) \\
\frac{\partial^2 x_3}{\partial z_1 \partial z_2} = \frac{\partial^2 x_3}{\partial z_2 \partial z_1} &= r_s \sin(z_1) \sin(z_2)
\end{aligned} \tag{J.45}$$

J.10. Ball rolling and spinning on $x_1 - x_2$ plane

A ball of radius r_{rb} is rolling and spinning on the $x_1 - x_2$ plane.

$$A(q) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -r_{rb}\cos(q_4) & r_{rb}\sin(q_4)\cos(q_5) \\ 0 & 1 & 0 & r_{rb} & 0 & r_{rb}\sin(q_5) \end{bmatrix} \tag{J.46}$$

$$\dot{A}(q) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & r_{rb}\sin(q_4)\dot{q}_4 & r_{rb}\left(\cos(q_4)\cos(q_5)\dot{q}_4 - \sin(q_4)\sin(q_5)\dot{q}_5\right) \\ 0 & 0 & 0 & 0 & 0 & r_{rb}\cos(q_5)\dot{q}_5 \end{bmatrix} \tag{J.47}$$

$$x = \psi(z) = [\ z_1 \ \ z_2 \ \ 0 \ \ z_3 \ \ z_4 \ \ z_5\ ]^T \tag{J.48}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{J.49}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.50}$$

J.11. Disk rolling along x_1 axis

A disk of radius r_d is rolling along the x_1 axis.

$$A(q) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & r_d \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{J.51}$$

$$\dot{A}(q) = [0] \tag{J.52}$$

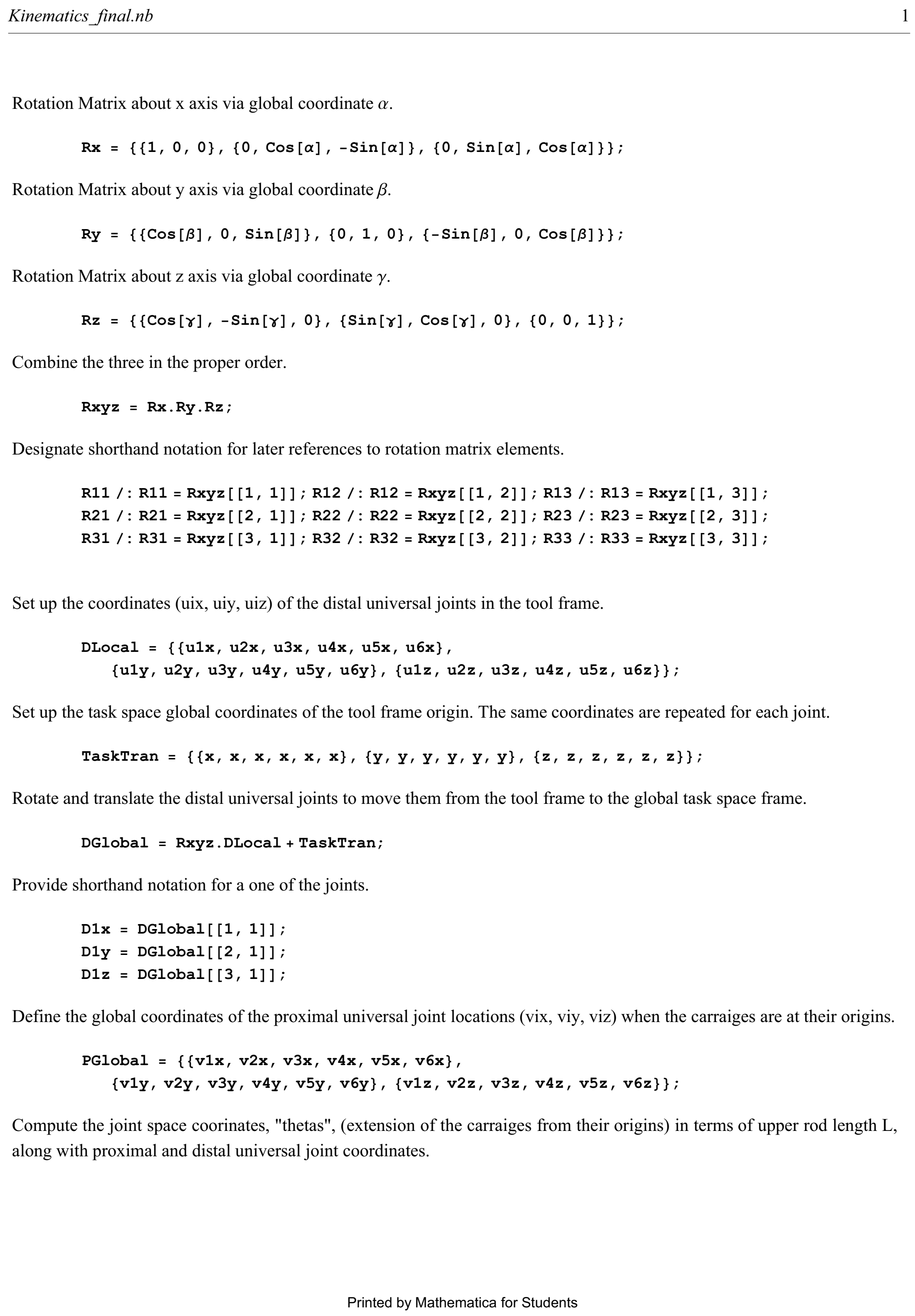
$$x = \psi(z) = [\ z_1 \quad 0 \quad 0 \quad 0 \quad 0 \quad -\tfrac{z_1}{r_d}\ ]^T \tag{J.53}$$

$$\frac{\partial \psi}{\partial z} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{r_d} \end{bmatrix} \tag{J.54}$$

$$\frac{\partial^2 \psi}{\partial z^2} = [0] \tag{J.55}$$

K. Jacobian and Hessians

The following seven pages provide the Mathematica printout of my analytical expressions for inverse Jacobian, $J^{-1}(x) = \frac{\partial \vartheta^{-1}(x)}{\partial x}$, and a series of six, 6×6 Hessian matrices $H_i^{-1}(x), i = 1 \ldots 6$. Great pains have been taken to find common terms and thus simplify both the coding and computation of the Jacobian and Hessian. The notation on the following pages is somewhere between the notation presented in this thesis, and that which appears in the software.



Rotation Matrix about x axis via global coordinate α.

```
Rx = {{1, 0, 0}, {0, Cos[α], -Sin[α]}, {0, Sin[α], Cos[α]}};
```

Rotation Matrix about y axis via global coordinate β.

```
Ry = {{Cos[β], 0, Sin[β]}, {0, 1, 0}, {-Sin[β], 0, Cos[β]}};
```

Rotation Matrix about z axis via global coordinate γ.

```
Rz = {{Cos[γ], -Sin[γ], 0}, {Sin[γ], Cos[γ], 0}, {0, 0, 1}};
```

Combine the three in the proper order.

```
Rxyz = Rx.Ry.Rz;
```

Designate shorthand notation for later references to rotation matrix elements.

```
R11 /: R11 = Rxyz[[1, 1]]; R12 /: R12 = Rxyz[[1, 2]]; R13 /: R13 = Rxyz[[1, 3]];
R21 /: R21 = Rxyz[[2, 1]]; R22 /: R22 = Rxyz[[2, 2]]; R23 /: R23 = Rxyz[[2, 3]];
R31 /: R31 = Rxyz[[3, 1]]; R32 /: R32 = Rxyz[[3, 2]]; R33 /: R33 = Rxyz[[3, 3]];
```

Set up the coordinates (uix, uiy, uiz) of the distal universal joints in the tool frame.

```
DLocal = {{u1x, u2x, u3x, u4x, u5x, u6x},
   {u1y, u2y, u3y, u4y, u5y, u6y}, {u1z, u2z, u3z, u4z, u5z, u6z}};
```

Set up the task space global coordinates of the tool frame origin. The same coordinates are repeated for each joint.

```
TaskTran = {{x, x, x, x, x, x}, {y, y, y, y, y, y}, {z, z, z, z, z, z}};
```

Rotate and translate the distal universal joints to move them from the tool frame to the global task space frame.

```
DGlobal = Rxyz.DLocal + TaskTran;
```

Provide shorthand notation for a one of the joints.

```
D1x = DGlobal[[1, 1]];
D1y = DGlobal[[2, 1]];
D1z = DGlobal[[3, 1]];
```

Define the global coordinates of the proximal universal joint locations (vix, viy, viz) when the carraiges are at their origins.

```
PGlobal = {{v1x, v2x, v3x, v4x, v5x, v6x},
   {v1y, v2y, v3y, v4y, v5y, v6y}, {v1z, v2z, v3z, v4z, v5z, v6z}};
```

Compute the joint space coorinates, "thetas", (extension of the carraiges from their origins) in terms of upper rod length L, along with proximal and distal universal joint coordinates.

```
thetas = DGlobal[[3]] - PGlobal[[3]] -
   √(L² - (DGlobal[[1]] - PGlobal[[1]])² - (DGlobal[[2]] - PGlobal[[2]])²) ;
```

Compute the Inverse Jacobian J^{-1} between task and joint space.

```
Jinverse = Transpose[
   {D[thetas, x], D[thetas, y], D[thetas, z], D[thetas, α], D[thetas, β], D[thetas, γ]}];
```

Now we attempt to simplify this expression for the Inverse Jacobian by pulling out common and repeated terms.
Here are 12 variables computed once for all Inverse Jacobian and Hessian Elements.

```
casb /: casb = Cos[α] Sin[β];

sasb /: sasb = Sin[α] Sin[β];

sbcg /: sbcg = Sin[β] Cos[γ] ;

sbsg /: sbsg = Sin[β] Sin[γ] ;

cacbcg /: cacbcg = Cos[α] Cos[β] Cos[γ];

cacbsg /: cacbsg = Cos[α] Cos[β] Sin[γ];

casbcg /: casbcg = Cos[α] Sin[β] Cos[γ] ;

casbsg /: casbsg = Cos[α] Sin[β] Sin[γ];

sacbcg /: sacbcg = Sin[α] Cos[β] Cos[γ] ;

sacbsg /: sacbsg = Sin[α] Cos[β] Sin[γ];

sasbcg /: sasbcg = Sin[α] Sin[β] Cos[γ];

sasbsg /: sasbsg = Sin[α] Sin[β] Sin[γ];
```

Here are 7 variables required FOR EACH ROW of the Inverse Jacobian. Since I will only demonstrate here the first row of the Inverse Jacobian, I give only the seven needed. The notation for j'th variable, j = 1:7, for the i'th row is risubj, uix, uiy, uiz, Dix, Diy, Diz.

```
r1sub1 /: r1sub1 = u1y (-R21) + u1x (R22);

r1sub2 /: r1sub2 = u1z Cos[β] - u1x (sbcg) + u1y (sbsg);

r1sub3 /: r1sub3 = -u1y (R11) - u1x (-R12);

r1sub4 /: r1sub4 = - (D1z - z);

r1sub5 /: r1sub5 =  u1x (sacbcg) + u1z (sasb) - u1y (sacbsg);

r1sub6 /: r1sub6 = - (D1x - x);

r1sub7 /: r1sub7 = u1z (-R23) - u1x (sasbcg) + u1y (sasbsg);
```

Finally, we detail the elements of the first row of the Inverse Jacobian in shorthand notation. The i'th row of the Inverse Jacobain has terms [(ric1 + ric1n/ric1d), (ric2 + ric2n/ric2d), (ric3 + ric3n/ric3d), (ric4 + ric4n/ric4d), (ric5 + ric5n/ric5d), (ric6 + ric6n/ric6d)]. Thus the remaining rows of the Inverse Jacobian can be computed with this same notation, provided additional sub variables are computed. Again, for additional rows of the Inverse Jacobian, note the notation mentioned for the 7 sub variables above (uix, uiy, uiz, Dix, Diy, Diz).

```
r1c1p /: r1c1p = 0;
r1c1n /: r1c1n = -v1x + D1x;
r1c1d /: r1c1d = √ (L^2 - r1c2n^2 - r1c1n^2) ;

r1c2p /: r1c2p = 0;
r1c2n /: r1c2n = -v1y + D1y;
r1c2d /: r1c2d = r1c1d;

r1c3p /: r1c3p = 0;
r1c3n /: r1c3n = 1;
r1c3d /: r1c3d = 1;

r1c4p /: r1c4p = D1y - y;
r1c4n /: r1c4n = (-r1c2n) (-r1sub4);
r1c4d /: r1c4d = r1c1d;

r1c5p /: r1c5p = -u1x (cacbcg) - u1z (casb) + u1y (cacbsg);
r1c5n /: r1c5n = Sin[α] (r1c2n) (-r1sub6) + (r1c1n) (r1sub2);
r1c5d /: r1c5d = r1c1d;

r1c6p /: r1c6p = u1y (-R31) + u1x (R32);
r1c6n /: r1c6n = (r1sub3) (r1c1n) + (r1sub1) (r1c2n);
r1c6d /: r1c6d = r1c1d;
```

Now we develope expressions for the Hessian matrices. There will be six Hessian matrices, a 6x6 Hessian for each row of the Inverse Jacobian. Since the shorthand expressions for each Hessian will be identical except for the row index, we need only develop expressions for one of the Hessians, or 36 of the 216 total Hessian terms.

I denote 10 more variables required FOR EACH Hessian. Since I will only demonstrate the first of six Hessians, I give only the ten for the Inverse Jacobian Row 1 Hessian. Again, for additional rows of the Inverse Jacobian, note the notation mentioned for the 7 sub variables above (uix, uiy, uiz, Dix, Diy, Diz).

```
r1sub8 /: r1sub8 = -u1z (R33) + u1x (casbcg) - u1y (casbsg);

r1sub9 /: r1sub9 = -u1y (sacbcg) - u1x (sacbsg);

r1sub10 /: r1sub10 = u1y (sbcg) + u1x (sbsg);

r1sub11 /: r1sub11 = u1y (cacbcg)  + u1x (cacbsg);

r1sub12 /: r1sub12 = -u1x (R11) + u1y (-R12);

r1sub13 /: r1sub13 = u1x (-R21) + u1y (-R22);

r1sub14 /: r1sub14 = u1x (-R31);
```

```
r1sub15 /: r1sub15 = u1y (-R32);

r1sub16 /: r1sub16 = r1c1d^3;

r1sub17 /: r1sub17 = r1c2n^2;
```

The first column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the first term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 1]], x];

D[Jinverse[[1, 1]], y];

D[Jinverse[[1, 1]], z];

D[Jinverse[[1, 1]], α];

D[Jinverse[[1, 1]], β];

D[Jinverse[[1, 1]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier. Here we establish 6 of the 216 terms of the Hessian in shorthand nottation. The notation is ricjdk, refering to row i of the Inverse Jacobian, column j of the Inverse Jacobian, and finally the partial of that Inverse Jacobian element with respect to task space global coordinate k.

```
r1c1dx /: r1c1dx = r1c1n^2 / r1sub16 + 1 / r1c1d;

r1c1dy /: r1c1dy = r1c1n * r1c2n / r1sub16;

r1c1dz /: r1c1dz = 0;

r1c1dα /: r1c1dα = r1c1n * r1sub4 * r1c2n / r1sub16;

r1c1dβ /: r1c1dβ = r1c1n * (r1sub2 * r1c1n + r1sub5 * r1c2n) / r1sub16 + r1sub2 / r1c1d;

r1c1dγ /: r1c1dγ = r1c1n * (r1sub3 * r1c1n + r1sub1 * r1c2n) / r1sub16 + r1sub3 / r1c1d;
```

The second column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the second term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 2]], x];

D[Jinverse[[1, 2]], y];

D[Jinverse[[1, 2]], z];

D[Jinverse[[1, 2]], α];

D[Jinverse[[1, 2]], β];

D[Jinverse[[1, 2]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier.

```
r1c2dx /: r1c2dx = r1c1dy;

r1c2dy /: r1c2dy = r1sub17 / r1sub16 + 1 / r1c1d;

r1c2dz /: r1c2dz = 0;

r1c2dα /: r1c2dα = sub4 * r1sub17 / r1sub16 + (r1sub4) / r1c1d;

r1c2dβ /: r1c2dβ = r1c2n * (r1sub2 * r1c1n + r1sub5 * r1c2n) / r1sub16 + r1sub5 / r1c1d;

r1c2dγ /: r1c2dγ = r1c2n * (r1sub3 * r1c1n + r1sub1 * r1c2n) / r1sub16 + r1sub1 / r1c1d;
```

The third column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the third term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 3]], x];

D[Jinverse[[1, 3]], y];

D[Jinverse[[1, 3]], z];

D[Jinverse[[1, 3]], α];

D[Jinverse[[1, 3]], β];

D[Jinverse[[1, 3]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier.

```
r1c3dx /: r1c3dx = 0;

r1c3dy /: r1c3dy = 0;

r1c3dz /: r1c3dz = 0;

r1c3dα /: r1c3dα = 0;

r1c3dβ /: r1c3dβ = 0;

r1c3dγ /: r1c3dγ = 0;
```

The fourth column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the fourth term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 4]], x];

D[Jinverse[[1, 4]], y];

D[Jinverse[[1, 4]], z];

D[Jinverse[[1, 4]], α];

D[Jinverse[[1, 4]], β];

D[Jinverse[[1, 4]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier.

```
r1c4dx /: r1c4dx = r1c1dα;

r1c4dy /: r1c4dy = r1c2dα;

r1c4dz /: r1c4dz = 0;

r1c4dα /: r1c4dα = r1sub4 + r1sub4^2 r1sub17 / r1sub16 + (r1sub4^2 - r1c2n * r1c4p) / r1c1d;

r1c4dβ /: r1c4dβ = r1sub5 + (r1sub4 * r1c2n) (r1c1n * r1sub2 + r1sub5 * r1c2n) / r1sub16 +
    (r1sub5 * r1sub4 - r1c5p * r1c2n) / r1c1d;

r1c4dγ /: r1c4dγ = r1sub1 + (r1sub4 * r1c2n) (r1c1n * r1sub3 + r1sub1 * r1c2n) / r1sub16 +
    (r1sub4 * r1sub1 - r1c6p * r1c2n) / r1c1d;
```

The fifth column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the fifth term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 5]], x];

D[Jinverse[[1, 5]], y];

D[Jinverse[[1, 5]], z];

D[Jinverse[[1, 5]], α];

D[Jinverse[[1, 5]], β];

D[Jinverse[[1, 5]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier.

```
r1c5dx /: r1c5dx = r1c1dβ;

r1c5dy /: r1c5dy = r1c2dβ;

r1c5dz /: r1c5dz = 0;

r1c5dα /: r1c5dα = r1c4dβ;

r1c5dβ /: r1c5dβ = r1sub8 + (r1c1n * r1sub2 + r1sub5 * r1c2n)^2 / r1sub16 +
    (r1c1n * r1sub6 + r1sub5^2 + r1sub2^2 + r1sub7 * r1c2n) / r1c1d;

r1c5dγ /: r1c5dγ =
  r1sub11 + (r1c1n * r1sub2 + r1sub5 * r1c2n) (r1c1n * r1sub3 + r1sub1 * r1c2n) / r1sub16 +
    (r1c1n * r1sub10 + r1sub3 * r1sub2 + r1sub5 * r1sub1 + r1sub9 * r1c2n) / r1c1d;
```

The sixth column of the Inverse Jacobian Row 1 Hessian is computed by taking partials w/r/t task space global coordinates of the sixth term of the first row of the Inverse Jacobian.

```
D[Jinverse[[1, 6]], x];

D[Jinverse[[1, 6]], y];
```

```
D[Jinverse[[1, 6]], z];

D[Jinverse[[1, 6]], α];

D[Jinverse[[1, 6]], β];

D[Jinverse[[1, 6]], γ];
```

Express these partials in terms of the Inverse Jacobian elements and "sub" expressions defined earlier.

```
r1c6dx /: r1c6dx = r1c1dγ;

r1c6dy /: r1c6dy = r1c2dγ;

r1c6dz /: r1c6dz = 0;

r1c6dα /: r1c6dα = r1c4dγ;

r1c6dβ /: r1c6dβ = r1c5dγ;

r1c6dγ /: r1c6dγ = r1sub14 + r1sub15 + (r1sub3 * r1c1n + r1sub1 * r1c2n)^2 / r1sub16 +
    (r1sub3^2 + r1c1n * r1sub12 + r1sub1^2 + r1c2n * r1sub13) / r1c1d;
```