Robot Design Optimization with Haptic Interface Applications

By

Leo J. Stocco

B.A.Sc, The University of British Columbia, 1994

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
in
THE FACULTY OF GRADUATE STUDIES
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

We accept this thesis as conforming

to the required standards

- -

- -

- -

- -

- -

- -

THE UNIVERSITY OF BRITISH COLUMBIA

December 1999

Abstract

To meet the high performance demands of modern robot applications, design variables such as materials, geometry, actuators and sensors must be chosen for optimum performance. This thesis presents a new way of choosing design variables to tune the capabilities of a robot to the needs of an application. The associated proposals are demonstrated through the design of a haptic interface.

It is argued that isotropy over a given workspace is a good measure of design quality for many high performance applications. A new measure of isotropy, the Global Isotropy Index or GII, is presented which is computed from the singular values of a design matrix. To ensure that the singular values are meaningful, a technique is presented which normalizes and scales the design matrix. The technique removes all physical units from the design matrix and scales it to accommodate an application-dependent performance specification and non-homogeneous actuator capabilities. An algorithm has also been developed that solves for the design parameters that result in the optimum GII. It is so efficient that it can be used to compare the relative isotropies of different robot configurations.

Performance specifications for a haptic interface are taken from the literature and are augmented by two biomechanical studies. The values obtained are used by the proposed design procedure to select the best of three robots to be used as a haptic pen. The preferred candidate is a novel hybrid design that uses two 3-DOF pantographs to position the ends of a pen shaped end-effector. A prototype with passive roll about the pen axis is built and controlled to simulate three virtual environments including a virtual pencil, a virtual scalpel and a virtual excavator. Its performance characteristics are measured and are used to draw conclusions about the effectiveness of the design procedure.

Finally, a further performance improvement is sought via redundant actuation. It is shown that the motion range and force capabilities of a coarse-stage robot can be combined with the precision and high-acceleration of a fine-stage robot by connecting them in series and joining their end-effectors by a flexible coupling. A coarse-fine system such as this is expected to narrow the gap between achievable and ideal haptic interface performance.

Table of Contents

List of Tables

List of Figures

Dedication

To all who have stood on the other side of the net.

Acknowledgment

The contents of this thesis have been greatly inspired by my supervisor Dr. Tim Salcudean. I must also thank Dr. Farrokh Sassani my co-supervisor, Simon Backmann and Dave Fletcher for their contributions to the Twin-Pantograph haptic pen and Simon DiMaio for creating the graphical environments.

I must also thank the Natural Sciences and Engineering Research Council of Canada (NSERC), the Institute of Robotics and Intelligent Systems (IRIS/PRECARN) and the memory of JK Zee for funding this work.

“If I have seen further, it is by standing on the shoulders of giants”

- Sir Isaac Newton, 1676.

Chapter 1

Introduction

Modern robot applications such as haptic interfaces and surgical assistants make performance demands far beyond those of the assembly and repetitive task devices of the past. Specifications are no longer limited to just workspace, resolution and payload requirements but now include acceleration [64], mass [25] and stiffness [15] requirements, to name a few. Designing a robot to uphold a set of performance standards is complicated by the fact that the relationship between a robot's actuators and its end-effector varies with both position and direction. Only after minimizing this variation, or in other words maximizing isotropy, can one choose suitable components and design a controller. The greatest opportunity for improving isotropy is through geometric and actuator design parameter selection but making the best choice is no small task. A function must be defined that evaluates the performance of a device over a range of positions and directions and handles mixed physical units in a meaningful fashion. Many functions have been proposed to do this [37] but they are often inconclusive and handle mixed physical units somewhat arbitrarily [16]. Once a function has been defined, it can be used by an optimization algorithm to compare the performance of different robots and select design parameter values. The search space can be made finite through discretization but the order of the optimization problem is compounded by each geometric parameter and by each degree-of-freedom so even low dimensional problems can be extremely computationally demanding. While many efficient search methods exist, most are incompatible with minimax robot design problems whose objective functions are often non-linear, non-differentiable, non-convex and/or discontinuous. Descent algorithms [20] become trapped in local minima, stochastic approaches [68] have uncertain stopping criteria and the results of a global search become increasingly suspicious as the search resolution is decreased.

This thesis addresses many of the difficulties associated with robot design and demonstrates its proposals with haptic interface design examples. Its contributions include a condition index for evaluating robot isotropy, a method for normalizing physical units and scaling a design matrix to account for a task-dependent performance specification and non-homogeneous actuator capabilities, an optimization algorithm and design philosophy for improving isotropy through design parameter selection, a new hybrid manipulator and a proposal for a dual-stage coarse-fine system. All proposals are substantiated by design examples which include serial, parallel and hybrid manipulators with between two and six degrees of freedom.

The proposed condition index, the Global Isotropy Index or GII, summarizes the global performance of a robot throughout a pre-defined workspace by a scalar value. Unlike most global measures which compute the mean of a local measure, the GII computes worst-case performance so intermittent displays of poor performance are not tolerated. Since the GII is computed from the singular values of a design matrix, it becomes meaningless if the design matrix contains mixed physical units. To address this, a technique for scaling the design matrix is also proposed. The technique removes all physical units from the design matrix and introduces parameters into the condition index for specifying non-homogeneous task-space requirements and actuator capabilities. By fixing task-space requirements to match the direction dependent needs of the applications and treating the actuator scale factors as design variables, the method allows one to determine the geometry and actuators which best satisfy the needs of the application.

The proposed optimization algorithm, the culling algorithm, is a discrete branch-and-bound algorithm that is specifically designed to handle minimax problems such as those involving the GII. It is unaffected by nonlinear, non-differentiable, discontinuous, non-convex or unbounded objective functions and guarantees the same result as a global search but arrives at the solution orders of magnitude faster. It requires no estimation or integration of the objective function and is insensitive to initial conditions.

The above proposals are applied to the design of a haptic interface. An appropriate design criteria is motivated from the literature ([9] and [42] for example) and from a model of a human hand interacting with a haptic device. Biomechanical studies are also conducted to provide the target values required to normalize and scale the design matrix.

A new hybrid manipulator, the Twin-Pantograph, is introduced that provides six-degree-of-freedom motion with all but one actuator in the base. It has a larger workspace than a comparably sized parallel robot with similar stiffness and mass characteristics. It has a redundant actuator which eliminates a singular configuration and allows the controller to be decoupled to simplify computations and enable a higher control rate. The proposed design procedure is used to compare the Twin-Pantograph to two parallel robots in terms of their static force isotropy. Due to the superior results of the Twin-Pantograph, it is implemented as a haptic pen. The capabilities of the haptic pen are measured and the device is controlled to simulate three diverse virtual environments.

Although the prototype haptic pen satisfies many of the requirements that define ideal haptic performance [42], it does not have the low mass and high acceleration required for ideal free motion. These specifications can only be satisfied by a specialized device such as a maglev joystick [26]. Proposals have been made to extend the workspace of a high-fidelity but limited motion range device by mounting it on a coarse motion stage [36], [66]. Unfortunately, dual-stage systems such as this retain the static force

limitations of the fine-stage. It is proposed here that a coarse-fine system can be made to inherit both the large workspace and static force capabilities of the coarse stage and the high bandwidth of the fine stage by placing a flexible coupling in parallel with the coarse and fine stages.

This thesis is organized as follows. In Chapter 2, existing condition indices, normalization techniques, optimization algorithms, haptic interface performance specifications, haptic interface designs and coarse-fine systems are surveyed. A new condition index, presented in Chapter 3, is normalized and scaled in Chapter 4 and a new optimization algorithm is presented in Chapter 5. The design techniques are then applied to the design of a 6-DOF haptic pen. It starts in Chapter 6 with a discussion of haptic interface performance requirements and a description of biomechanical studies that were conducted to provide target values. In Chapter 7, a new hybrid manipulator called the Twin-Pantograph is introduced and is compared to two parallel platform manipulators in terms of their workspace sizes and static force isotropy. Chapter 8 describes the implementation of a haptic pen based on the Twin-Pantograph, records its capabilities and describes how it is controlled to simulate a variety of virtual environments. In Chapter 9, a proposal is made to further improve performance by redundantly connecting two manipulators in series via a flexible coupling. This is followed by conclusions and proposals for future work in Chapter 10. Appendix A contains the analytic equations of motion for the various robots that are used in the design examples throughout this thesis and Appendix B discusses discretization issues related to workspaces with mixed translational and rotational components.

The contributions made by this thesis are summarized as follows:

- a new condition index called the Global Isotropy Index which evaluates the isotropy of a design matrix throughout a pre-defined workspace.
- a method which removes all of the physical units from a design matrix by introducing parameters that describe maximum actuator and end-effector capabilities.
- a new optimization algorithm called the culling algorithm that solves discrete minimax or GII based design problems significantly faster than a global search.
- experimental data describing the force and velocity capabilities of the human hand which can be used to establish a design criteria for a haptic interface.
- a new hybrid robot design called the Twin-Pantograph.

- a design example that uses the described design techniques to compare two existing parallel platform robots to the Twin-Pantograph for use as a haptic pen. A prototype based on the Twin-Pantograph is built and controlled to simulate three diverse virtual environments.
- A proposal for combining the high bandwidth of a fine-stage robot with the large workspace and forces of a coarse-stage robot by connecting the two in series and joining their end-effectors with a flexible coupling.

Portions of this work have been presented at *The IEEE International Conference on Robotics and Automation* [74] [76], *The Winter Annual Meeting of the ASME, Sixth Annual Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems* [77], have been published in *Robotica, International Journal of Information, Education and Research in Robotics and Artificial Intelligence* [75], *The IEEE Transactions on Robotics and Automation* [78] and are under review by *The IEEE/ASME Transactions on Mechatronics*.

Chapter 2

Background

This chapter describes past proposals and past results related to robot design and haptic interfaces. The topics include condition indices in Section 2.1, normalization techniques in Section 2.2, optimization algorithms in Section 2.3, haptic interface performance specifications in Section 2.4, haptic interface designs in Section 2.5 and coarse-fine systems in Section 2.6.

2.1 Condition Indices

Most condition indices are computed from a design matrix such as the Jacobian [39] or mass matrix [47] to evaluate a performance attribute such as kinematic or inertial isotropy. Some of the more common condition indices are surveyed by Kim and Khosla in [37]. They include the manipulability measure M shown in equation (2.1) and the condition number k shown in equation (2.2) where $G(x,p)$ is the design matrix (e.g. Jacobian, mass matrix, etc.) evaluated at a position x for a design parameter p, λ_{min} and λ_{max} are the minimum and maximum eigenvalues of $G(x,p)$ and σ_{min} and σ_{max} are the minimum and maximum singular values of $G(x,p)$. The manipulability measure, originally proposed by Yoshikawa [90], is equivalent to the product of all singular values of $G(x,p)$ and is, therefore, proportional to the volume of the manipulability ellipsoid. The condition number k measures the variation in singular values (i.e. the roundness of the manipulability ellipsoid) at a position x. The larger the condition number, the greater the variation.

$$M = \sqrt{det(G(x,p)G^T(x,p))} \tag{2.1}$$

$$k = \sqrt{\frac{\lambda_{max}(G(x,p)G^T(x,p))}{\lambda_{min}(G(x,p)G^T(x,p))}} = \frac{\sigma_{max}(G(x,p))}{\sigma_{min}(G(x,p))} \tag{2.2}$$

Kim and Khosla [37] propose an isotropy measure Δ (2.3) which is the ratio between the manipulability measure and the average of all eigenvalues. The isotropy measure is a scale independent version of the manipulability measure.

$$\Delta = \frac{M}{mean(\lambda)} \tag{2.3}$$

Ma and Angeles [47] describe the dynamic condition index (DCI) μ (2.4) where W is a diagonal weighting matrix and d is a vector containing the upper triangular components of a difference matrix that is computed by subtracting the inertia matrix from its nearest isotropic neighbor (i.e a scaled identity matrix). The DCI measures both conditioning and decoupling of the manipulator dynamics.

$$\mu = \frac{1}{2} d^T W d \tag{2.4}$$

Other condition indices proposed by van den Doel and Pai [15] include the generalized Yoshikawa index Y (2.5) and the nonlinearity measure R (2.6) where η is a metric that measures the difference in position and orientation of the end-effector, g is a tensor valued performance field, h^{ij} is an element from the metric tensor on the joint space and R^k_{ijk} is an element from the Riemann tensor on the joint space. The generalized Yoshikawa index measures the amount of necessary robot motion for a given end-effector motion while the nonlinearity measure is a measure of error that would occur if a linear function were used to compute the kinematics of the device. Similar measures are also defined for constrained systems in [14].

$$Y = \sqrt{\frac{det(\eta)}{det(g)}} \tag{2.5}$$

$$R = h^{ij} R^k_{ijk} \tag{2.6}$$

The measures described in (2.1) through (2.6) are all local measures (i.e. evaluated at a single position) and one is usually interested in behaviour throughout a workspace. Various proposals have been made to remove configuration dependence from these measures. Ma and Angeles [47] design a Stewart Platform by solving for both the geometry and position that minimize the DCI. Their device is optimally conditioned at one position but arbitrarily conditioned at all others. This approach is best suited to a device that is only expected to navigate a small working volume.

Hayward *et al.* [25] plot the condition index of a planar 5-bar linkage over its workspace. This brute-force approach is difficult to automate and severely limits the range of design parameters that can be explored. Plotting is also impractical when a device has more than two degrees of freedom (see [41]).

The “Global Condition Index” (GCI) is proposed by Gosselin and Angeles [22] to overcome these problems. It is the integral of a local performance measure such as the condition number k over a workspace W scaled by the size of W. Because the GCI is an average value, it tolerates small regions of

poor behaviour and, therefore, may overrate a design that is unacceptable in practice. Computing the integral may also be difficult and time consuming for certain condition indices.

$$GCI = \int_W \frac{1}{k} dW \Big/ \int_W dW \tag{2.7}$$

Global measures that are derived from local measures also have another problem. Information is often lost when a local measure is computed. For example, scale information is discarded when the condition number is computed so changes in the magnitudes of singular values at different positions go undetected by a plotting or an averaging approach. Hayward *et al.* [25] deal with this problem by following up a condition number optimization with a manipulability check but iterative strategies such as this do not always lead to the global optimum.

Park [55] proposes a measure of global dexterity that computes the average distortion between two Riemannian manifolds with curvatures representing the joint-space and task-space volumes of the robot. Although this does not depend on any local measure, it is an average value and, therefore, tolerates intermittent displays of poor behaviour.

2.2 Normalization Techniques

Methods for handling non-uniform workspace dimensions have been suggested by Gosselin [21], Tandirci *et al.* [82], Zanganeh and Angeles [92], Ma and Angeles [46], Ranjbaran *et al.* [61] and Doty *et al.* [16]. They address the problem pointed out by Lipkin and Duffy [44] that a measure such as the condition number of the Jacobian matrix is of little practical significance in the presence of non-uniform physical units. This occurs when a robot can both translate and rotate its end-effector or when it contains both rotary and prismatic actuators. To accommodate this, Gosselin [21] defines a new Jacobian that transforms actuator velocities into the linear velocities of two points on the end-effector but it is not discussed how one should choose these points. Tandirci *et al.* [82] normalize the Jacobian by scaling all translational elements by a Characteristic Length or CL. The CL that produces the best performance measure is dubbed the “Natural Length” (NL) by Ma and Angeles [46] and is used for design optimization. When the NL of a platform manipulator is not derivable, it is approximated by the average platform radius. Angeles [3] calculates the NL for a serial manipulator by averaging the distances between the operating point and all active joint axes while Ranjbaran *et al.* [61] find a serial manipulator’s NL by making it a free design parameter. Doty *et al.* [16] propose a method of inverting non-square matrices with mixed physical units so that the solution is both unit and frame invariant. The method achieves physical unit consistency but does not differentiate between quantities with similar units but dissimilar magnitudes.

2.3 Optimization Algorithms

Maximizing a condition index inside a workspace is a minimax optimization problem. There are many existing optimization algorithms that can be nested to solve minimax problems. They include iterative approaches such as the Newton-Raphson method [20], gradient descents such as the Fletcher-Reeves method of conjugate gradients [19], non-gradient methods such as Powell's method of conjugate directions [59] and quasi-gradient/direction methods such as the Davidon-Fletcher-Powell method of the variable metric [12]. These methods are local methods and are only useful for finding local minima. To obtain the global minimum, methods have been developed that repeatedly invoke a local routine to find all local minima inside a constraint boundary. Examples include Torn's clustering method [79], systematic methods [7] which look for both local minima and decomposition points that may lead to other local minima, Multi-start [57] which randomly selects initial conditions until the quantity of identified local minima exceeds a Bayesian estimate, Rock and Roll [45] which rocks the landscape (objective function) and rolls the solution into the nearest valley and a tunneling algorithm [88] which tunnels horizontally through the landscape in search of a valley where it can descend to the local minima and continue tunnelling from there.

Other global search methods do not rely on local minimization techniques but look for the global minimum directly. Branch-and-Bound algorithms recursively subdivide the search space and formulate upper and lower bounding functions for each region. When the lower bound of one region is above the upper bound of another, that first region is discarded. Many proposals ([57] for example) have been made regarding ways of subdividing search spaces (branching) and bounding objective functions with various properties (i.e. concavity, continuity, smoothness) to improve convergence time.

Statistical or Bayesian/sampling methods perform sequential objective function evaluations using the parameter with the greatest probability of improving the result. There are many versions including Direct, Stuckman's, Mockus's, Perttunen's, Zilinskas's, Shaltenis's and Dzemyda's which are discussed in [32] [79] [80]. Other methods emulate natural processes such as simulated annealing [68] [79] which emulates how a metal cools (i.e. anneals) to reach a state of minimum energy, simulated diffusion [38] which emulates the diffusion of particles in a cooling metal and genetic algorithms [23] [73] which treat parameters as members of an evolving species where only the strongest (most favourable) survive and generate offspring which undergo random probabilistic mutation. Both statistical and natural process algorithms traverse the feasible region with the intent of improving the objective function until a stopping criteria is satisfied. They offer no guarantee of global optimality but rather a measure of confidence that a better function value does not exist elsewhere in the search domain.

Monte-Carlo [79] [93] methods perform uniform random searches with a known probability of convergence to a global optimum. Due to their probabilistic nature, a global optimum is not guaranteed. Neural networks [53] describe an optimization problem using a physical analog circuit. Parameters are adjusted using variable resistors, discrete switches and controlled source gains and the solution appears as a node voltage in the circuit. Neural networks can solve optimization problems in real time since the settling time of the analog components is very short but they are very difficult to implement, particularly when the objective function is complicated.

Some algorithms perform minimax optimization directly such as linear and nonlinear programming approaches which use discrete semi-infinite parameter spaces and gradients [54], methods which solve the unconstrained concave dual to the problem using descents [83], methods which integrate the objective function over the semi-infinite parameters and optimize the average value [31] and tree-searches such as game theory [5]. It is not always possible to apply these algorithms to robot design due to the properties of the objective functions (i.e. nonlinear, discontinuous, non-differentiable, etc.). Some that have been applied to robot design include the complex method [22], DBCOMF (IMSLTM) [47], least squares [46], Pattern Search [71], Rosen's gradient projection method [56], DEMOCRAT [49], and global searches [25].

2.4 Haptic Interface Performance Specifications

A haptic interface is a device that enables the exchange of tactile or kinesthetic information between a human and a computer. It is accomplished through the use of a robot that acts as both an input and an output device [24].

It has been argued that the range of impedances that can be emulated by a teleoperation master only needs to span what can be felt by a human hand [4], [18], [25], [70], [89]. This range is still being defined by ongoing human-factors and biomechanical studies [28], [42], [70], [89]. Meanwhile, specifications for a "universal" force-reflecting hand controller have been suggested by Brooks [4], Fischer [18], Hayward [24] and Sharpe [70] who base their recommendations on surveys of telerobotic experts and literature, the relative merits of existing teleoperation systems, mathematical models and human biomechanical experiments. The quantitative capabilities of the human hand are defined to some extent from prior work. Specifications from experience with existing haptic interfaces and biomechanical studies directed at defining human motor capabilities are summarized in Table 2.1 for the hand motions shown in Figure 2.1.

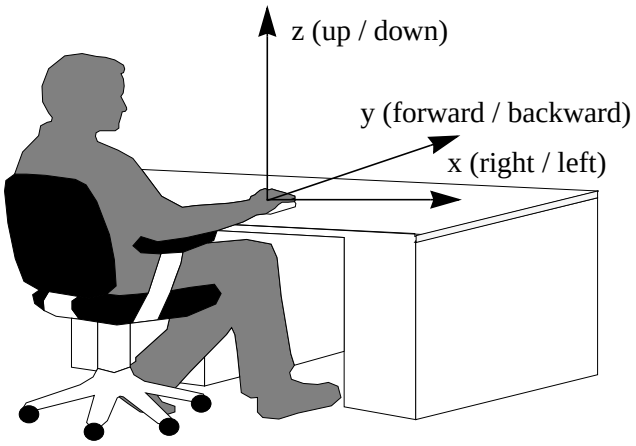


Figure 2.1: Directions of Human Hand Motion

Table 2.1: Cited Haptic Interface Performance Specifications

Characteristic	Cited Specification			
Translation Range (mm)	[18] : 6.7 (x, y, z) [A,B] [29] : 300 (x, y, z) [C] [34] : 17 (x, y) [C] [34] : 40 (x, y) [B] [40] : 200 (x, y, z) [C]		[48] : 250 x 170 x 80 (x, y, z) [C] [51] : 200 x 200 x 850 (x, y, z) [C] [60] : 160 x 100 (x, y) [C] [65][66]: 9 (x, y, z) [C] [84] : 130-195 x 100-150 (x, y) [A]	
Rotation Range (deg)	[6] : +99˚/ -90˚ (x) [A] [6] : +113˚/ -77˚ (y) [A] [6] : +47˚/ -27˚ (z) [A] [17] : ±84˚ x ±30˚ (x, z) [A]		[18] : ±30˚ (x, y, z) [A,B] [40] : ±86˚ (z) [C] [51] : ±45˚ (z) [C] [65][66]: ±7˚ (x, y, z) [C]	
Position Bandwidth (Hz)	[4] : 3.9-9.7 [B]	[18] : 50 [A,B]	[70] : 5-10 [A]	[89] : 70 [A]
Force Capability (N)	[18] : 100 [A,B] [29] : 22.6 [C] [30] : 0.5 [C]	[34] : 6.3 [C] [40] : 12 [C] [48] : 10 [C]	[51] : 44.48 [A] [60] : 10 [C]	[65][66]: 60 [C] [87] : 151-463 [A]
Torque Capability (Nm)	[1] : 1.26 [A] [2] : 9.925 [A]	[29] : 0.294 [C] [40] : 0.4 [C]	[51] : 1.356 [A] [65][66]: 6 [C]	[87] : 0.658 [A]
Force/Torque Bandwidth (Hz)	[4] : 320 [A] [25] : 300 [B]	[28] : 450 [A] [34]: 1000 [C]	[65][66]: >1000 [C] [70] : 300 [A]	[85] : 100 [A]
Max. Velocity (m/s)	[4] : 1.1 [B]	[18] : 1 [A,B]		
Max. Acceleration (m/s^2)	[4] : 12.2 [B]	[18] : 9.81 [A,B]	[65][66]: 90 [C]	
Min. Natural Freq. (Hz)	[51] : 100 [C]			

A. Based on human factors study.
B. Based on practical experience with an existing haptic interface.
C. Reported capability of an existing haptic interface.

Lawrence and Chapel [42] define ideal haptic behaviour by an upper and a lower bound on the available impedance range of a device. The high impedance lower bound corresponds to the minimum stiffness required to counteract a reasonable maximum hand force while the low impedance upper bound corresponds to the maximum impedance that is too small for a human to detect. Both bounds are constant

up until *20 rad/s* where the low frequency upper bound increases by *40 dB/decade* corresponding to the typical response of a system with significant mass such as a human hand (see Figure 2.2) where impedance is defined in equation (2.8) as a function of force *F* and position *X(jω)*. Any haptic devices that meet these standards are said to be ideal and are, therefore, equivalent since any capabilities outside of those defined are beyond the scope of human perception.

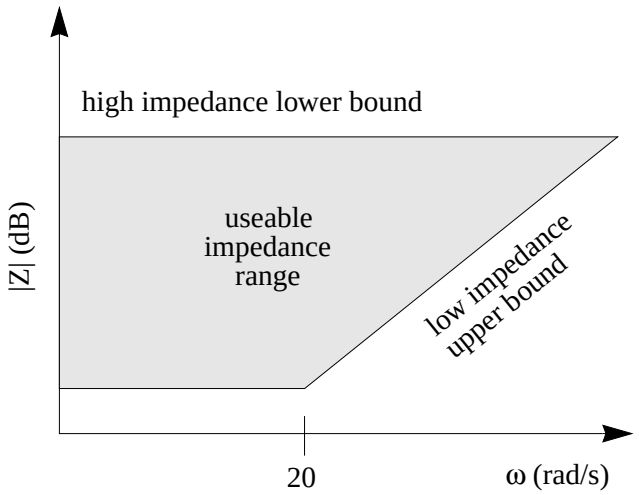


Figure 2.2: Practical Haptic Interface Impedance Range

$$|Z(j\omega)| = \frac{F}{|X(j\omega)|} \tag{2.8}$$

There have been many proposals on how to widen a device's impedance range. Lawrence and Chapel [42] propose lowering effective mass while Hayward *et al.* [25], Massie and Salisbury [48] and Ma and Angeles [47] make an argument for reducing variations in mass. Kurtz and Hayward [41] and Zanganeh and Angeles [92] argue the merits of an isotropic Jacobian while Colgate and Schenkel [9] suggest increasing stability by adding physical damping.

2.5 Haptic Interface Designs

Many attempts have been made to design a realistic force-reflecting master hand controller. In [8] a 6-DOF device combining three 2-DOF linkages is presented. Iwata [29] built a 9-DOF device that provides 6-DOF motion to the hand and 1-DOF motion to 3 sets of fingers as well as a 6-DOF haptic pen [30] positioned by two 3-DOF manipulators. In [10] a 6-DOF joystick with three parallel pantograph linkages is described while in [89] a hand controller using three prismatic actuators for translation and three rotary actuators for orientation is presented. In [48] a haptic probe with three active translational degrees of freedom and three passive rotational degrees of freedom is described while a 4-DOF device using only rotary actuators is

presented in [40]. A 4-DOF (3 translational, 1 rotational) joystick is described in [51] and a 2-DOF five-bar linkage with a horizontal planar workspace is optimized in [25]. A 2-DOF planar positioning device with linear coil/magnet actuators is presented in [34] while Vertut [85] provides a historical survey of earlier hand controllers, articulated arms, and exoskeletons.

Although each of the large motion range devices mentioned above can satisfy some of the requirements defined in Table 2.1, none are able to provide a sufficient force/torque bandwidth. There is, however, a less conventional device that can. It uses Lorentz magnetic levitation (maglev) [26] to eliminate all physical connections between the end-effector and ground. The advantages of using such a device in the design of a force-feedback hand controller have been shown in [65]. They include small mass, high frequency response, backdriveability, and low friction. Preliminary results with a teleoperation system using maglev wrists for the master and slave have shown good transparency for small motions [66]. The main drawback of the device is its small motion range. It was, therefore, suggested that the maglev device be mounted on a conventional 6-DOF motion stage to provide coarse positioning [66], [86].

2.6 Coarse-Fine Systems

Dual-stage, coarse-fine motion devices have been proposed before as a means of improving dynamic performance [35] and stability [69] for demanding robotic applications. Examples include the ARTISAN robot of Khatib and Roth [36] which combines a coarse stage 7-DOF serial arm with a fine-stage 3-DOF parallel wrist and the 2-DOF planar device of Yoshikawa *et al.* [91] which combines a flexible elbow manipulator with a rigid micro-manipulator. The problem with series coupled coarse-fine systems is that they are force limited to the capabilities of the fine-stage device. Morrell and Salisbury overcome this problem by connecting the devices in parallel via a flexible coupling [52]. This is shown to combine the high bandwidth performance of the fine-stage with the static force capabilities of the coarse-stage but limits the motion range of the device to that of the fine-stage. In Chapter 9, a series coarse-fine system with a flexible coupling is proposed to combine the force magnitude and motion range of the coarse stage with the high frequency performance of the fine stage.

Chapter 3

The Global Isotropy Index (GII)

3.1 Introduction

Isotropy is defined in [3] to represent equal properties in all directions. Much past work has focussed on improving the isotropy of the Jacobian or mass matrix to minimize wasted energy [3], to improve accuracy (i.e. invertability of the kinematic equations for control) [21], [22], [39], to optimally utilize the actuators [25] and to provide robust state-space representation, decoupled inertia torques and weak joint coupling for advanced control and dynamic simulation [47]. Existing isotropy indices can be classified as either local (e.g. the condition number [37]) or global measures (e.g. the GCI [22]). Local measures are position dependent whereas global measures are workspace inclusive. Manipulators that are designed to be locally isotropic may not exhibit similar levels of isotropy throughout their workspaces. Global measures are much more rigorous but they are often computed from either the average value [22] or the variation [25] of a local measure. This can produce misleading information since local measures often discard scale information [25] and average values hide intermittent displays of poor performance. In this chapter a new global measure called the Global Isotropy Index (GII) is introduced. Unlike most existing global measures, the GII is a worst-case performance measure. It retains scale information and is intolerant of poor performance anywhere in the workspace.

This chapter is organized as follows. In Section 3.2 shortcomings with existing condition indices are used to motivate the definition of a new global performance measure. In Section 3.3 the global isotropy index is described. In Section 3.4 some concluding remarks are made.

3.2 Shortcomings of Existing Condition Indices

Many relationships have been used in the past to quantify robot performance. They include, but are not limited to, the Jacobian $J(x)$ (3.1) that relates actuator rates $\dot{q}$ to end-effector velocity v at a workspace point x, its transpose (3.2) which relates end-effector force/torque f to actuator force/torque τ and the impedance $Z(s,x)$ (3.3) present at a robot's end-effector which may contain mass, stiffness and damping terms and is a function of the Laplace transform variable, s.

$$v = J(x)\dot{q} \tag{3.1}$$

$$\tau = J^T(x) f \tag{3.2}$$

$$f = Z(s, x)v \tag{3.3}$$

Relationships (3.1) to (3.3) are matrix transformations that are functions of x (i.e. position dependent) and are non-diagonal (i.e. direction dependent) in general. Minimizing the non-uniformity of these dependencies is often the primary goal in robot design optimizations. An argument is made later in this thesis (see Chapter 6) that supports maximizing the isotropy of the Jacobian matrix (3.2) for a haptic interface. For consistency, the Jacobian is also used in the examples presented in this chapter but the concepts are not limited in any way to the Jacobian. They can be applied to any design matrix such as the mass matrix which is demonstrated in Chapter 5.

While there are many ways to measure isotropy, a number of which are discussed in [37], perhaps the most common is the condition number which describes worst-case behaviour at a position. Using the force / torque transformation in (3.2) for example, if one were to plot all actuator torques that produce an end-effector force of unit magnitude and arbitrary direction, a joint-space ellipse would result. The lengths of the major and minor axis of the ellipse correspond to the maximum $\tilde{\sigma}(J(p, x))$ and minimum $\varsigma(J(p, x))$ singular values of $J^T(x)$ which is a function of both position x and geometry p. This relationship is shown in Figure 3.2 for the planar elbow manipulator in Figure 3.1. The robot has the geometry ($l_1=5$, $l_2=4$) and can apply forces in all directions but its workspace is limited to the horizontal trajectory $x \in \{-xmax, xmax\}$ at $y=2$.

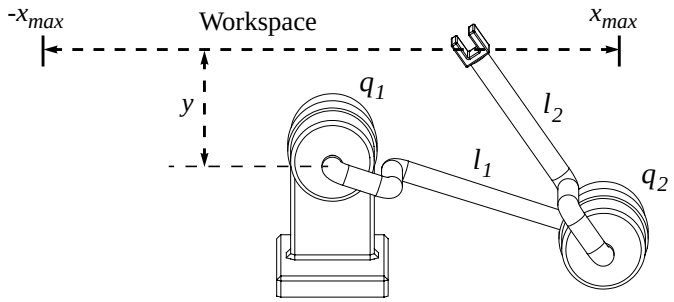


Figure 3.1: Constrained Planar Elbow Manipulator

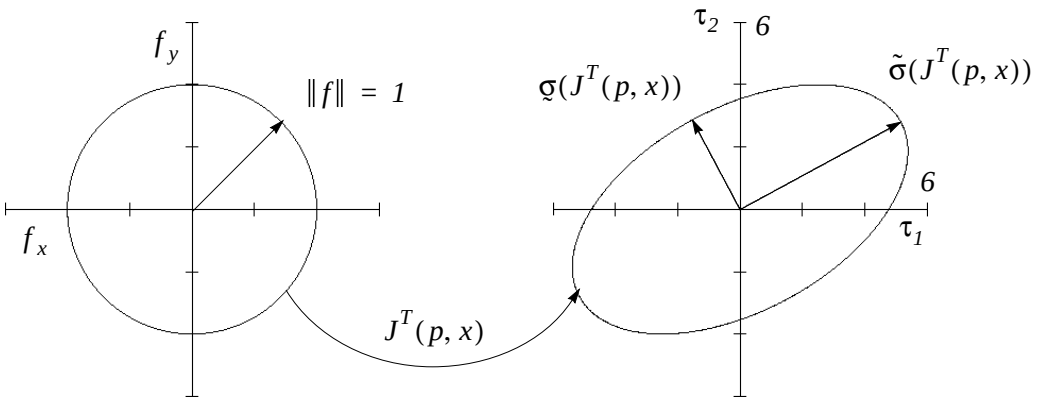


Figure 3.2: Torque Ellipse at x=5

The condition number of the Jacobian $\kappa(p, x)$ (3.4) compares the highest and lowest effective transmission ratios occurring in all directions by the minimum and maximum singular values of the Jacobian. For consistency, accuracy, direction independence and maximum distance from kinematic singularities, this ratio should be as close as possible to unity. The condition number approaches infinity as the robot nears a singular position and has a value of one when the robot is perfectly isotropic.

$$\kappa(p, x) = \frac{\tilde{\sigma}(J(p, x))}{\underline{\sigma}(J(p, x))} \tag{3.4}$$

Because the Jacobian is computed at a position x, the condition number is a local measure and manipulators that are designed to be isotropic at individual positions may not exhibit similar levels of isotropy throughout their workspaces. The condition number also only measures the shape (roundness) of the ellipse but discards information related to size. Since the shape represents directional isotropy and the size represents the average transmission ratio or positional isotropy, both are important. In Figure 3.3, torque ellipses are computed for the robot in Figure 3.1 at three different positions. Notice that the ellipses at $x=0$ and $x=\pm 5$ are similar in shape (i.e. 20% difference in the condition numbers) but are much different in size (i.e. 54% difference in the average singular values). Therefore, the manipulator is capable of producing forces that are an average of 54% larger at $x=0$ than it can at $x=\pm 5$. This is hardly uniform behaviour but it results in deceptively favourable condition numbers.

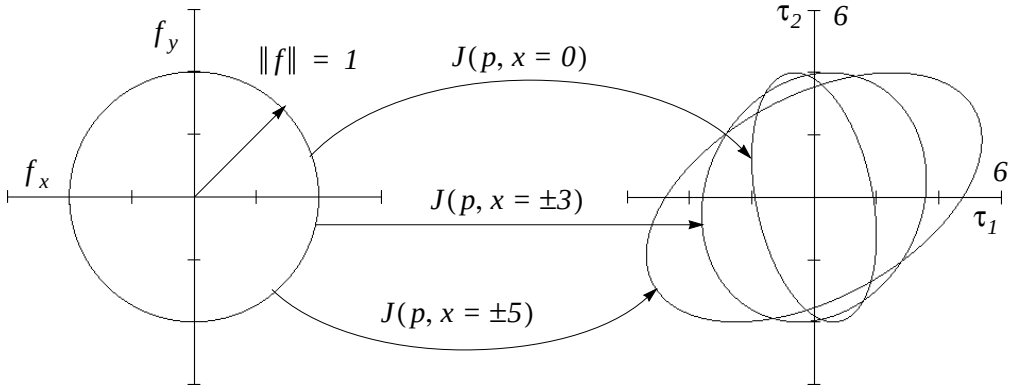


Figure 3.3: Force/Torque Transformation

In the past [25], configuration dependence has been checked by consulting the products of singular values (3.5) throughout the workspace as a secondary criteria.

$$m(p, x) = \varsigma(J(p, x)) \ \tilde{\sigma}(J(p, x)) \tag{3.5}$$

3.3 Description of the Global Isotropy Index

Here, a secondary, local measure is made to be unnecessary by introducing the Global Isotropy Index or GII which compares the largest and smallest singular values in the entire workspace (3.6). The GII is essentially a global or workspace inclusive version of the condition number. However, for convenience it is defined as the minimum singular value over the maximum singular value rather than the other way around so that perfect isotropy is assigned a value of *1* and singular behaviour is assigned a value of *0* (not ∞).

$$GII(p) = \min_{x_0, x_1 \in W} \frac{\varsigma(J(p, x_0))}{\tilde{\sigma}(J(p, x_1))} \tag{3.6}$$

Since the GII evaluates a robot design by the bounds on its singular values and not by an average value, it does not tolerate intermittent displays of poor performance. It also takes into account scale information since this information is reflected by the singular value bounds.

Consider again the elbow manipulator in Figure 3.1. Local actuator torque ellipses are computed at all values of *x* ranging from *-5* to *+5* and are superimposed upon one another in Figure 3.4. The GII is the ratio of the radius of the largest circle contained in all of these ellipses to the radius of the smallest circle containing all of these ellipses. The thinner the resulting torus is, the larger (i.e more favourable) the GII will be.

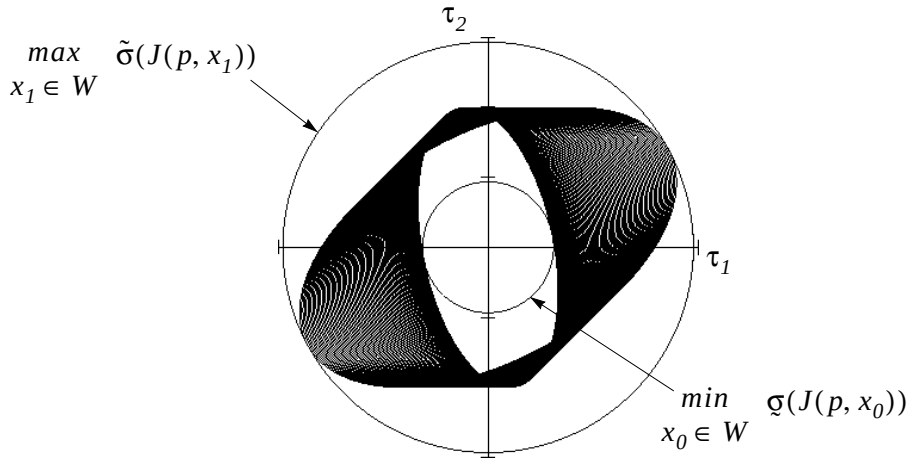


Figure 3.4: Force/Torque Ellipses and GII

A GII of *1* indicates that all singular values are identical throughout the workspace and that the mechanism is not only directionally isotropic at each position in its workspace, but also that its behaviour is consistent among all positions in its workspace. An optimally isotropic robot design parameter *p** is, therefore, one that maximizes the value of the GII as in (3.7).

$$p^* = \arg\max_{p \in P} GII(p) \tag{3.7}$$

Although the GII is presented in (3.6) using a Jacobian matrix, it can be similarly applied to any design matrix. For example, the GII of a mass matrix *D(p,x)* is shown in (3.8).

$$GII(p) = \min_{x_0, x_1 \in W} \frac{\underline{\sigma}(D(p, x_0))}{\tilde{\sigma}(D(p, x_1))} \tag{3.8}$$

3.4 Concluding Remarks

A new global measure called the GII has been described which compares the bounds on all singular values in the workspace. It can be used to measure the isotropy of the Jacobian matrix, the mass matrix, or any other design matrix of interest. A mechanism with a favourable GII will behave consistently in all directions and that behaviour will not change as the mechanism is moved around its workspace. The GII assigns a value of *1* to a device which is perfectly isotropic and a value of *0* to a device that is singular anywhere in its workspace.

In the following chapter, a method is proposed that removes the physical units from the design matrix that is used to compute the GII. It introduces task-space parameters into the GII so that it reflects the needs of an application and it introduces actuator design parameters into the GII to improve the performance of a design based on GII maximization.

Chapter 4

A Task-Dependent Design Matrix Normalizing Technique

4.1 Introduction

The GII, presented in Chapter 3, uses the singular values of a design matrix to evaluate robot isotropy. For singular values to be meaningful, the physical units of all matrix elements must be uniform [44]. This is not the case with robots that are capable of both linear and angular motion and/or combine rotary and prismatic actuators. These devices have a Jacobian (mass matrix, Coriolis matrix, etc.) that contains mixed physical units. Scale factors such as the characteristic length [82] and natural length [46] have been used in the past to deal with mixed physical units but the values assigned are somewhat arbitrary [16]. Furthermore, isotropy, as it is traditionally defined (see [3]), may not be an appropriate goal for some robot applications. Although it may be desirable for a robot's behaviour to be consistent over a range of positions, it may not be desirable for that performance to be homogeneous in all directions. For example, a legged robot can support its own weight with minimal energy when its legs are near singular positions.

In this chapter, a method is proposed for normalizing a design matrix so that all physical units are eliminated. A physical interpretation of the scale factors shows that they represent relative task-space and joint-space performance values. Task-space values are fixed to assign non-homogeneous performance goals for different directions of motion and joint-space values are used as design parameters to select actuator sizes for improved isotropy. The technique is demonstrated by a number of design examples which are carried out using the optimization algorithm described in Chapter 5.

This chapter is organized as follows. In Section 4.2 the design matrix normalization technique is described. In Section 4.3 the ability of the task-space scaling matrix to assign a direction-dependent performance specification to the design matrix is demonstrated using robot design examples. In Section 4.4 the ability of the joint-space scaling matrix to improve performance through actuator scaling is demonstrated using robot design examples. In Section 4.5 some concluding remarks are made.

4.2 Specifying a Direction-Dependent Performance Objective

Lipkin and Duffy [44] point out that the condition number of a design matrix has little meaning when the matrix has non-uniform physical units. This occurs when a robot can both translate and rotate its end-effector or when it contains both rotary and prismatic actuators. Furthermore, even if physical units are

uniform, the singular values only evaluate the uniformity of actuator responsibilities for a task-space event of unit magnitude and arbitrary direction. They do not address the more general case of non-uniform actuator capabilities and/or task-space responsibilities. To remove this limitation, a more general description of desired robot performance is formulated and a conformity measure is derived.

A common robot design criteria is isotropy of end-point forces. For the sake of simplicity, consider a robot example where only forces f (i.e. no torques) are produced at the end-effector and only torques τ are generated by the actuators (i.e. all rotary actuators) so that physical units are homogeneous. The more general case will be considered later. A robot is said to have an isotropic force profile at a position if the length of the joint-space torque vector (RMS value of all joint torques) is constant for any task-space force of unit magnitude. A perfectly isotropic force profile is, therefore, illustrated by the mapping shown in Figure 4.1a. The ratio of singular values of a robot's Jacobian $J(x)$ at a position x describes how closely $J(x)$ approximates the transformation shown in Figure 4.1a.

Figure 4.1a represents an ideal isotropic mapping only if the intended use of the manipulator demands forces of equal magnitude in all directions. For some applications, it may be preferable that the robot be capable of larger forces along one axis than it is along another (e.g. a device affected by gravity). The ideal force/torque transformation for a device with non-homogeneous task-space force requirements would map an ellipse in task-space into a circle in joint-space. A mapping such as this would suggest that the kinematic chain has a mechanical advantage along the direction corresponding to the major axis of the task-space ellipse. Of course, the axes of the desired task-space force ellipse may not align with the coordinate system of the design matrix task-space variables resulting in a desired transformation shown in Figure 4.1b.

Figure 4.1b is represents an ideal isotropic mapping only if the actuators have uniform torque capabilities. If the actuators have different torque capabilities, the stronger of the two would be under-utilized. Figure 4.1b would, therefore, be undesirable for most serial manipulators since they typically use actuators of various sizes. Full actuator utilization would require that the task-space ellipse be mapped into a joint-space ellipse as shown in Figure 4.1c, where the major axis of the joint-space ellipse aligns with the axis of the stronger actuator. Note that as long as there is no cross-coupling between actuators, the axes of the joint-space torque ellipse always align with the joint-space coordinate frame. Figure 4.1c, therefore, illustrates a general representation of an ideal task-space to joint-space mapping given variations in both task-space requirements along different directions and individual actuator capabilities. Note that a similar argument is easily applied to non-uniform velocity, acceleration, resolution, and other task and joint-space quantities.

Conformance of a robot's Jacobian to a desired mapping between two ellipsoids such as that shown in Figure 4.1c cannot be determined by singular values alone. If, however, a new transformation matrix $\hat{J}^{-T}(x)$ is derived that transforms fractions of maximum values (i.e. Δf, $\Delta\tau$) rather than the actual values themselves (i.e. f, τ), even the more general representation of desired isotropic behaviour such as that shown in Figure 4.1c takes on the familiar form shown in Figure 4.1d since fractions of maximum values are always unity bounded. The GII of the transformation in Figure 4.1d is easily computed from the minimum and maximum singular values in the workspace of the scaled Jacobian $\hat{J}^{-T}(x)$.

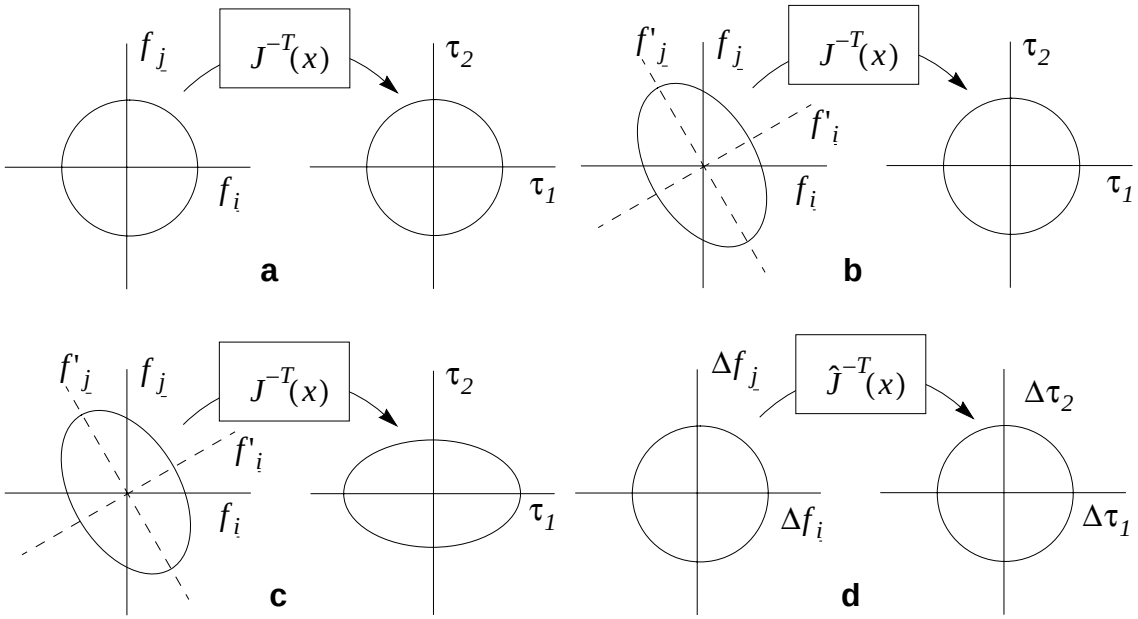


Figure 4.1: Desired Force/Torque Transformations

The normalized transformation matrix $\hat{J}(x)$ is computed for a parallel manipulator by representing task-space force f and joint-space torque τ as fractions of their maximum values. This is shown in (4.2) and (4.3) where S'_T and S_J are diagonal scaling matrices with maximum values along the diagonal, S'_R is a task-space rotation matrix which rotates the desired force ellipse's axes into the design matrix's task-space coordinate frame, and Δf and $\Delta\tau$ are vectors of unity bounded fractions of maximum values. Equations (4.2) and (4.3) are substituted into equation (4.1) in (4.4) which is rearranged in (4.5) and (4.6) to arrive at the normalized transformation matrix $\hat{J}(x)$ in (4.7). Because all scaling matrices are extracted from the task and joint-space vectors (f and τ), S'_R, S'_T and S_T are $n \times n$ matrices where n is the number of active degrees of freedom and S_J is an $m \times m$ matrix where m is the number of actuators. This holds regardless of whether *J(x)* is square or whether the device is over or under-actuated.

$$f = J(x)^T \tau \tag{4.1}$$

$$f = S'_R S'_T \Delta f \tag{4.2}$$

$$\tau = S_J \Delta\tau \tag{4.3}$$

$$S'_R S'_T \Delta f = J(x)^T S_J \Delta\tau \tag{4.4}$$

$$\Delta f = S'^{-1}_T S'^{-1}_R J(x)^T S_J \Delta\tau = S^{-1}_T J(x)^T S_J \Delta\tau = \hat{J}(x)^T \Delta\tau \tag{4.5}$$

$$S_T = S'_R S'_T \tag{4.6}$$

$$\hat{J}(x) = S_J J(x) S^{-T}_T \tag{4.7}$$

For a two dimensional robot example, S'_R, S'_T, S_J, Δf and $\Delta\tau$ are expanded in (4.8) through (4.12) where $\tilde{f}'_{\underline{i}}$ and $\tilde{f}'_{\underline{j}}$ are the maximum desired forces along the $\underline{i}'$ and $\underline{j}'$ axes which define a reference frame rotated α radians from the $\underline{i}$ and $\underline{j}$ axes, $\tilde{\tau}_1$ and $\tilde{\tau}_2$ are the maximum torque capabilities of actuators *1* and *2*, $\Delta f_{\underline{i}}$ and $\Delta f_{\underline{j}}$ are the fractions of maximum force along the $\underline{i}'$ and $\underline{j}'$ axes and $\Delta\tau_1$ and $\Delta\tau_2$ are the fractions of maximum torque of actuators *1* and *2*.

$$S'_R = \begin{bmatrix} cos(\alpha) & sin(\alpha) \\ -sin(\alpha) & cos(\alpha) \end{bmatrix} \tag{4.8}$$

$$S'_T = Diag\left[\tilde{f}'_{\underline{i}} \; \tilde{f}'_{\underline{j}}\right] \tag{4.9}$$

$$S_J = Diag\left[\tilde{\tau}_1 \; \tilde{\tau}_2\right] \tag{4.10}$$

$$\Delta f = \left[\Delta f'_{\underline{i}} \; \Delta f'_{\underline{j}}\right]^T \tag{4.11}$$

$$\Delta\tau = \left[\Delta\tau_1 \; \Delta\tau_2\right]^T \tag{4.12}$$

Note that S'_R may not be static as shown in (4.8). For example, it is configuration dependent if the task-space performance specifications are specified in end-effector coordinates. On the other hand, if the performance specifications are specified in base coordinates and $\alpha = 0$, S'_R is the identity and S_T is equal to S'_T (4.13) since $\tilde{f}'_{\underline{i}}$ and $\tilde{f}'_{\underline{j}}$ are equal to $\tilde{f}_{\underline{i}}$ and $\tilde{f}_{\underline{j}}$ (4.14):

$$S_T = Diag\left[\tilde{f}_{\underline{i}} \; \tilde{f}_{\underline{j}}\right] \tag{4.13}$$

$$\Delta f = \left[\Delta f_{\underline{i}} \; \Delta f_{\underline{j}}\right]^T \tag{4.14}$$

Since the homogeneous Jacobian $\hat{J}(x)$ (4.6) transforms fractions of maximum actuator force/torque values into fractions of maximum end-effector force/torque values, it is unit-free and easily adapted to the more general case where mixed physical units are originally present. Consider, for example, a 6-DOF manipulator with rotary odd numbered joints and prismatic even numbered joints. S'_R, S'_T, S_J, Δf and $\Delta\tau$ are shown in equations (4.15) through (4.19) where R is the 3×3 rotation matrix which rotates the desired force ellipse axes into the design matrix task-space coordinate frame. Note that desired torques specified about a point non-collocated with the reference end-point results in off-diagonal terms in S'_R (4.15).

$$S'_R = \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} \tag{4.15}$$

$$S'_T = Diag\left[\tilde{f}'_i \;\; \tilde{f}'_j \;\; \tilde{f}'_k \;\; \tilde{\tau}'_i \;\; \tilde{\tau}'_j \;\; \tilde{\tau}'_k\right] \tag{4.16}$$

$$S_J = Diag\left[\tilde{\tau}_1 \;\; \tilde{f}_2 \;\; \tilde{\tau}_3 \;\; \tilde{f}_4 \;\; \tilde{\tau}_5 \;\; \tilde{f}_6\right] \tag{4.17}$$

$$\Delta f = \left[\Delta f'_i \;\; \Delta f'_j \;\; \Delta f'_k \;\; \Delta\tau'_i \;\; \Delta\tau'_j \;\; \Delta\tau'_k\right]^T \tag{4.18}$$

$$\Delta\tau = \left[\Delta\tau_1 \;\; \Delta f_2 \;\; \Delta\tau_3 \;\; \Delta f_4 \;\; \Delta\tau_5 \;\; \Delta f_6\right]^T \tag{4.19}$$

Although the Jacobian is used for design examples presented here, the S'_R, S'_T and S_J matrices can be used to normalize and scale any transformation matrix such as (3.1) or (3.3) but S'_T and S_J will contain quantities other than forces and torques. For example, to normalize a mass matrix $D(x)$ (4.20) (i.e. $\hat{D}(x)$ (4.21)), S_J contains maximum end-effector forces/torques and S'_T contains maximum end-effector accelerations (4.22). Note that in (4.22), S_J contains maximum end-effector forces which are task-space quantities. Therefore, S_J is really a second task-space matrix. Its notation is maintained for the sake of simplicity. Similarly, to normalize the velocity Jacobian $\hat{J}(x)$ (3.1), S_J contains maximum joint rates while S'_T contains maximum end-effector velocities.

$$f = D(x)\ddot{x} \tag{4.20}$$

$$\Delta f = \hat{D}(x)\Delta\ddot{x} \tag{4.21}$$

$$\hat{D}(x) = S_J^{-1} D(x) S_T \tag{4.22}$$

Note that scalar multiples do not affect ratios of singular values so one element of S'_T and S_J can be factored out with the remaining elements representing relative values. Also note that these matrices are easily adapted to serial manipulators by rearranging their order (4.25) to account for the way the Jacobian is normally defined for a serial mechanism (4.23):

$$\tau = J(x)^T f \tag{4.23}$$

$$\Delta\tau = \hat{J}(x)^T \Delta f \tag{4.24}$$

$$\hat{J}(x) = S_T^T J(x) S_J^{-1} \tag{4.25}$$

4.3 A Method for Defining a Non-Homogeneous Performance Goal

By definition, the values along the diagonal of the task-space scaling matrix S_T describe the maximum capabilities of the robot end-effector along each direction of motion. By assigning these values to correspond to the requirements of the intended application, the scaled, normalized design matrix (e.g. $\hat{J}(x)$) will not be isotropic unless the requirements specified in the S_T matrix are met. Therefore, optimizing the GII of this design matrix will result in a robot design that is optimally suited to the specified non-homogeneous task-space performance goal.

Consider, for example, the three degree-of-freedom (3-DOF) parallel planar manipulator shown in Figure 4.2. This manipulator belongs to a large class of planar parallel manipulators that has been described by Merlet in [50]. The geometry of the device is described by five design parameters, l_1 through l_4 and θ_0 and the workspace of the device is constrained by two translational, $\langle x \rangle$, $\langle y \rangle$ and one rotational limit $\langle \theta \rangle$. Although additional geometric parameters exist for this robot, they are fixed (i.e. 120° separation between actuators on the platform and the base) to make parameter optimization more manageable. Computation of the Jacobian matrix (4.26) is shown in Appendix A.6.

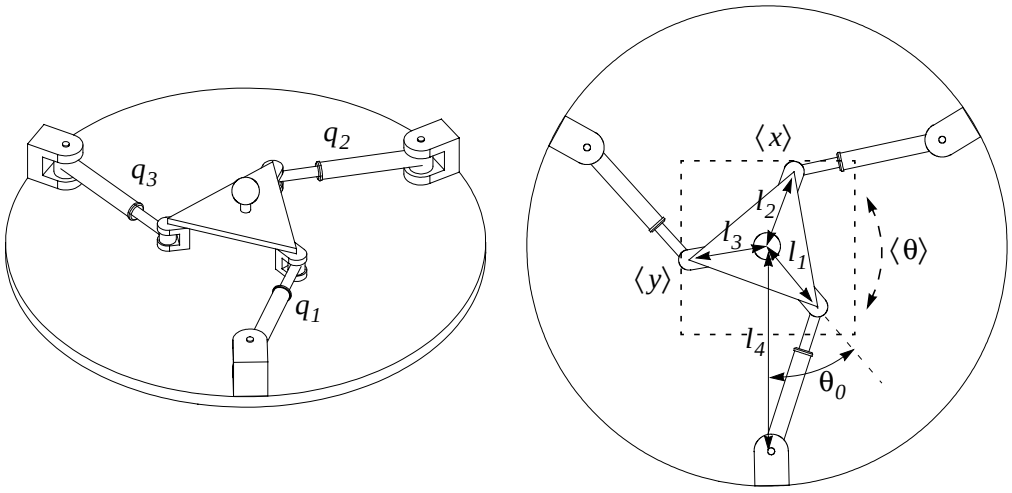


Figure 4.2: 3-DOF Planar Parallel Manipulator

$$\left[f_i \; f_j \; \tau_k\right]^T = J^T\left[f_1 \; f_2 \; f_3\right]^T \tag{4.26}$$

Since isotropy always improves when the radius of the base is increased, l_4 is fixed at *20cm*. The robot parameters l_1 through l_3 and θ_0 are found to optimize the GII inside a square workspace ($\langle x \rangle$, $\langle y \rangle$ = $\pm$*5cm*, $\langle \theta \rangle$ = $\pm$*30*˚). To observe the effect of different task-space requirements on the optimum geometry, the optimization is carried out many times using different values inside the S_T matrix. In all cases, the force requirements are kept equal in both directions while the torque requirements are varied. Since equal force requirements make S'_R inconsequential, it is set to the identity. Substituting $\tilde{f}_i = \tilde{f}_j = \tilde{f}_{i,j}$ into the maximum force/torque vector $\tilde{f}$ in (4.27) and dividing through by $\tilde{f}_{i,j}$ (ratios of singular values are invariant to scalar multiples) produces the S_T matrix shown in (4.28). For $\tilde{\tau}_k / \tilde{f}_{i,j}$ varied between *0.1cm* and *10.0cm*, the optimum geometries and GIIs are shown in Figure 4.3.

$$\tilde{f} = \left[\tilde{f}_i \; \tilde{f}_j \; \tilde{\tau}_k\right]^T \tag{4.27}$$

$$S_T = Diag\left[1 \; 1 \; \tilde{\tau}_k / \tilde{f}_{i,j}\right] \tag{4.28}$$

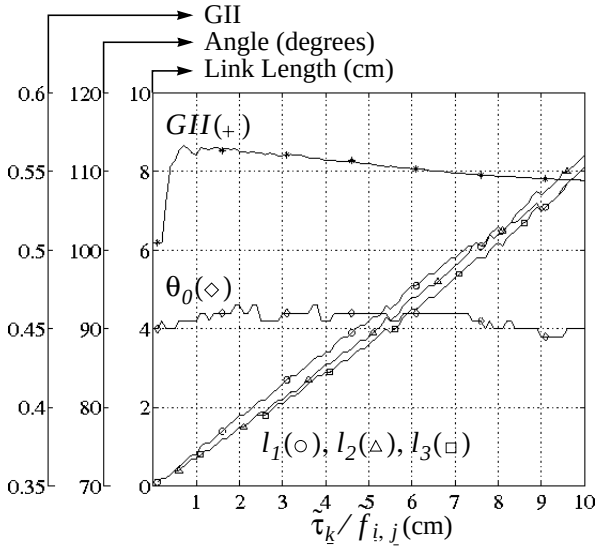


Figure 4.3: Optimum Parallel Manipulator Parameters

The optimum platform offset angle θ_0 stays relatively constant at approximately *90*˚ and the platform is nearly equilateral in all cases. Its size increases linearly with $\tilde{\tau}_k / \tilde{f}_{i,j}$ which is, on average, *1.27* times the mean platform radius. This is expected since a larger platform radius provides the actuators with additional leverage on the platform centre and increases the task-space torque capabilities $\tilde{\tau}_k / \tilde{f}_{i,j}$. Figure 4.3,

therefore, confirms that assigning values to the S_T matrix in accordance with a desired task-space performance specification produces a device that is particularly well suited to that performance specification.

Note that since S_T (4.28) places equal weighting upon all translational elements and all rotational elements, it is a special case that is mathematically equivalent to scaling by a Characteristic Length (CL) [82] equal to $\tilde{\tau}_{\underline{k}} / \tilde{f}_{i,\underline{j}}$. If one was to disregard the value of the scale factor (i.e. the CL or $\tilde{\tau}_{\underline{k}} / \tilde{f}_{i,\underline{j}}$), as done in the past, one would be led to conclude that the platform radius corresponding to the Natural Length or NL (CL of *0.7cm* which maximizes the GII in Figure 4.3) represents a globally optimum solution and that any two designs with similar GIIs should have similar performance characteristics. This obviously cannot be true since a platform radius of *5.3cm* is clearly capable of higher torques than a platform radius of *0.4cm*, but both produce the same GII. Thus, as expected, the optimum mechanism design varies significantly with the performance specification that is described by the scale factor(s) which, therefore, must not be ignored.

Since the S_T matrix contains a performance goal for every direction of motion, it can be used to assign dissimilar performance goals along all motion axes, not just those with dissimilar physical units. Consider, for example, an assembly robot that lifts a part onto a shaft and locks it in place. Translational force requirements are not homogeneous since horizontal positioning forces need only address the mass of the payload while vertical positioning forces must also overcome gravity. The torque requirements are different still and correspond to the torque needed to lock the part into place. An example S'_R matrix that accounts for an angle of $\alpha = 30^\circ$ between the manipulator's $\underline{j}$ axis and the real world's vertical ($\underline{j}$) axis is shown in (4.29) and an example S'_T matrix for the requirements in (4.30) through (4.32) is shown in (4.33) which are used to re-optimize the robot in Figure 4.2. Note that the S_T matrix (4.34) produced from S'_R (4.29) and S'_T (4.33) scales both translational and rotational elements and is not mathematically equivalent to multiplication by a CL.

$$S'_R = \begin{bmatrix} 0.866 & 0.5 & 0 \\ -0.5 & 0.866 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad for \quad \alpha = 30° \tag{4.29}$$

$$\tilde{f}'_{\underline{i}} = 5N \tag{4.30}$$

$$\tilde{f}'_{\underline{j}} = 25N \tag{4.31}$$

$$\tilde{\tau}'_{\underline{k}} = 50Ncm \tag{4.32}$$

$$S'_T = Diag\left[1 \;\; \tilde{f}'_{\underline{j}} / \tilde{f}'_{\underline{i}} \;\; \tilde{\tau}'_{\underline{k}} / \tilde{f}'_{\underline{i}}\right] = Diag\left[1 \;\; 5 \;\; 10cm\right] \tag{4.33}$$

$$S_T = \begin{bmatrix} 0.866 & 2.5 & 0 \\ -0.5 & 4.33 & 0 \\ 0 & 0 & 10cm \end{bmatrix} \tag{4.34}$$

The optimum robot geometry has $l_1 = 4.75$, $l_2 = 1.75$, $l_3 = 7.75$ and $\theta_0 = 77^\circ$ with a GII of *0.158*. Unlike the devices described in Figure 4.3, the optimum geometry for this application (shown in Figure 4.4a) has asymmetric platform dimensions. This solution is not obvious and illustrates the power of the proposed technique to tailor robot design parameters toward satisfying non-homogeneous task-space requirements. Note from Figure 4.4b that if the task-space requirements are mirrored across the manipulator's $\underline{j}$ axis ($\alpha = -30°$), the optimum geometry $l_1 = 4.75$, $l_2 = 7.75$, $l_3 = 1.75$ and $\theta_0 = -77^\circ$ produces an identical GII (*0.158*). This is the mirror image of the original solution, just as one would expect. Any other rotation angle, however, produces a completely different optimum geometry (e.g. $l_1 = 3.25$, $l_2 = 8.5$, $l_3 = 7.75$ and $\theta_0 = 97^\circ$, GII = *0.155* for $\alpha = 0^\circ$) as shown in Figure 4.4c.

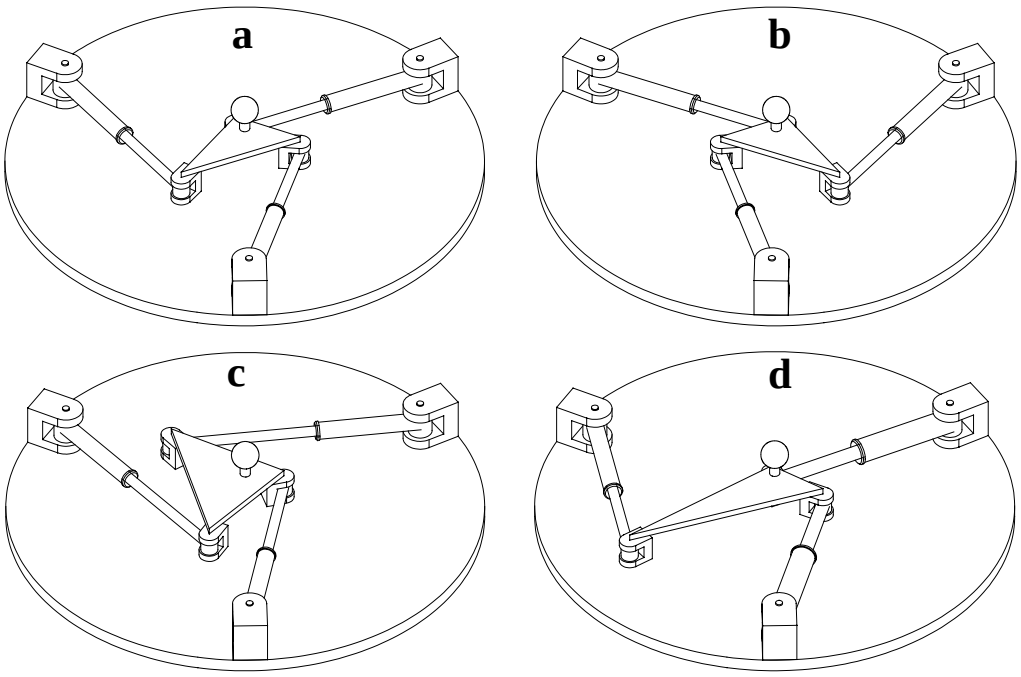


Figure 4.4: Asymmetric 3-DOF Planar Parallel Manipulators

4.4 A Method for Selecting Non-Homogeneous Actuators

By definition, the values along the diagonal of the joint-space scaling matrix S_J describe the maximum capabilities of the robot actuators. By making these values design variables, they can be chosen by the optimization algorithm to maximize isotropy.

To illustrate this, the robot design of Figure 4.2 is re-visited using the S_T matrix in (4.34) with S_J derived from the maximum actuator force vector $\tilde{\tau}$ shown in (4.35) where $\tilde{f}_1$ through $\tilde{f}_3$ are the maximum force capabilities of actuators *1* through *3*. Dividing by a constant $\tilde{f}_1$ produces the S_J matrix in (4.36). With $\tilde{f}_2/\tilde{f}_1$ and $\tilde{f}_3/\tilde{f}_1$ treated as free design parameters, an optimum geometry with $l_1 = 4.5$, $l_2 = 1.0$, $l_3 = 14.5$ and $\theta_0 = 64^\circ$, optimum actuator scale factors of $\tilde{f}_2/\tilde{f}_1 = 0.9$ and $\tilde{f}_3/\tilde{f}_1 = 0.5$, and a GII of *0.22* is obtained. Recall that the optimum GII was only *0.158* when the S_J matrix was not included (i.e. homogeneous actuators). A diagram of the optimized device with the proposed actuator scaling is shown in Figure 4.4d.

$$\tilde{\tau} = \left[\tilde{f}_1 \;\; \tilde{f}_2 \;\; \tilde{f}_3\right]^T \tag{4.35}$$

$$S_J = Diag\left[1 \;\; \tilde{f}_2/\tilde{f}_1 \;\; \tilde{f}_3/\tilde{f}_1\right] \tag{4.36}$$

Although non-homogeneous actuation improved the GII of the parallel manipulator by a substantial *39%*,

it is even more effective with serial manipulators. In practice, serial manipulators rarely have identical actuators throughout because torque requirements tend to diminish for actuators distal to the base. It is for this same reason that they often produce very poor condition numbers since computing a condition index using a design matrix that has not been scaled by an S_J matrix evaluates the device as though it is fitted with homogeneous actuators. Consider, for example, the 3-DOF planar serial manipulator shown in Figure 4.5 which has three design parameters, l_1 through l_3. Computation of the Jacobian matrix (4.37) is shown in Appendix A.9.

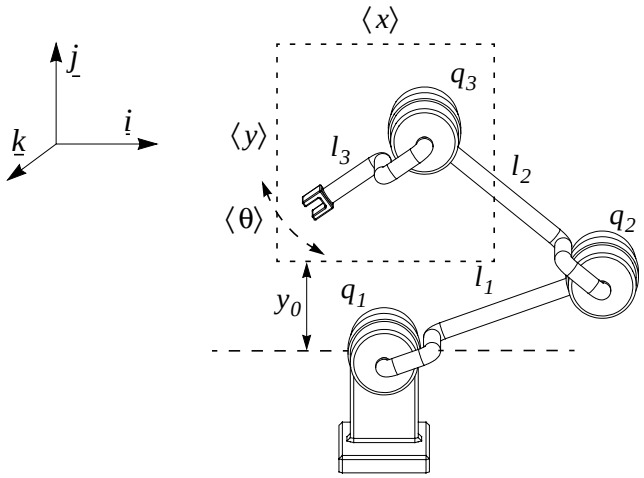


Figure 4.5: 3-DOF Planar RRR Serial Manipulator

$$\begin{bmatrix} \tau_1 & \tau_2 & \tau_3 \end{bmatrix}^T = J^T \begin{bmatrix} f_{\underline{i}} & f_{\underline{j}} & \tau_{\underline{k}} \end{bmatrix}^T \tag{4.37}$$

The workspace is centred a fixed distance y_0 of *5cm* above the base actuator q_1 and the robot parameters l_1 through l_3 are found to optimize the GII inside a square workspace ($\langle x \rangle$, $\langle y \rangle = \pm 5cm$) at any angle ($\langle \theta \rangle = \pm 180^\circ$). The force requirements along the $\underline{i}$ and $\underline{j}$ axes are kept equal ($\tilde{f}_{\underline{i}} = \tilde{f}_{\underline{j}} = \tilde{f}_{\underline{i},\underline{j}}$) while the torque requirements $\tilde{\tau}_{\underline{k}}$ are varied using the S_T matrix in (4.28). The results are shown in Figure 4.6.

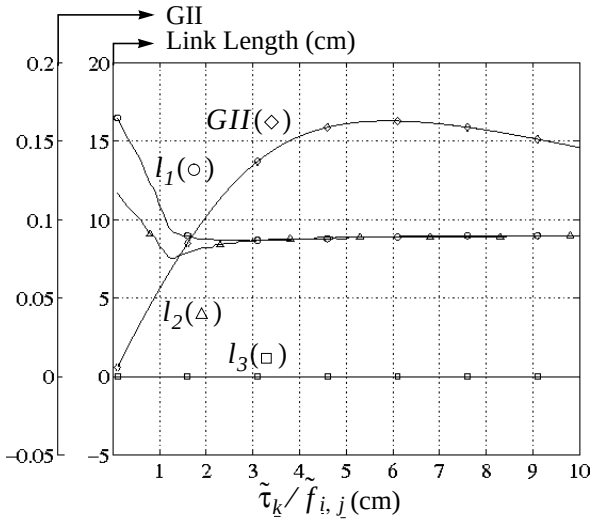


Figure 4.6: Optimum Parameters with Homogeneous Actuation

The GII in Figure 4.6 peaks at *0.163* and drops to *0.006* at low $\tilde{\tau}_k / \tilde{f}_{i,j}$ ratios. These results suggest that the serial device is unsuitable for applications with low torque requirements regardless of its geometry. These poor results, however, are not due to an intrinsic deficiency of the device but are largely due to inappropriate actuation. Performance is significantly improved by including the S_J matrix with free variables along the diagonal as shown in (4.38). Note that the S_J matrix in (4.38) is normalized with respect to $\tilde{\tau}_3$ since q_3 is likely to have the lowest torque requirements since it is farthest from the base.

$$S_J = Diag\left[\tilde{\tau}_1/\tilde{\tau}_3 \;\; \tilde{\tau}_2/\tilde{\tau}_3 \;\; 1\right] \tag{4.38}$$

Re-optimizing the serial robot with the two additional free parameters in (4.38) results in the designs presented in Figure 4.7. Notice the improvement in GII values which now vary from *0.17* to *0.28* with stronger actuators at the q_1 and q_2 joints. The non-homogeneously actuated device actually turns in its best results at low $\tilde{\tau}_k / \tilde{f}_{i,j}$ ratios with an up to *46*-fold improvement in its GII over its homogeneously actuated counterpart.

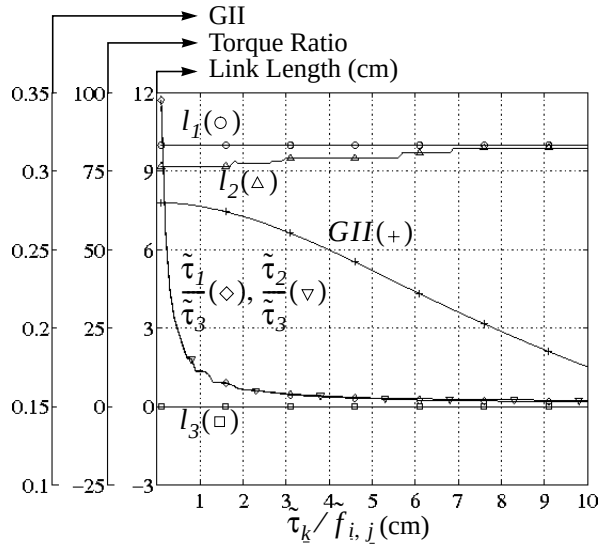


Figure 4.7: Optimum Parameters with Non-Homogeneous Actuation

Actuator scaling can cause physical dimensions (i.e. l_1 and l_2) to grow without bound since this is analogous to shrinking the workspace size which has a favourable effect on isotropy. This can be avoided by imposing physical constraints. In Figure 4.7, physical dimensions are limited to *10cm* so that a fair comparison can be made with Figure 4.6. In practice, one might choose to fix the actuator ratios in accordance with practical considerations such as availability, size or weight. For example, $\tilde{\tau}_2$ is fixed at twice $\tilde{\tau}_3$ and $\tilde{\tau}_1$ is fixed at twice $\tilde{\tau}_2$ by using the S_J matrix shown in (4.40). Using the non-homogenous task-space requirements described by the S_T matrix in (4.39), the optimum geometry has $l_1 = 9.6$, $l_2 = 6.6$, $l_3 = 0$ with a GII of *0.104*.

$$S_T = Diag\begin{bmatrix} 1 & 3 & 5 \end{bmatrix} \tag{4.39}$$

$$S_J = Diag\begin{bmatrix} 4 & 2 & 1 \end{bmatrix} \tag{4.40}$$

One might also fix both S_T and S_J when optimizing an impedance (3.3) or mass matrix (4.20) since both sides of the equation contain task-space quantities. Consider, for example, the 2-DOF planar serial manipulator shown in Figure 4.8 which has two design parameters, l_1 and l_2. The mass matrix $D(x)$ is computed in Appendix A.8 and is scaled using (4.41) and (4.42).

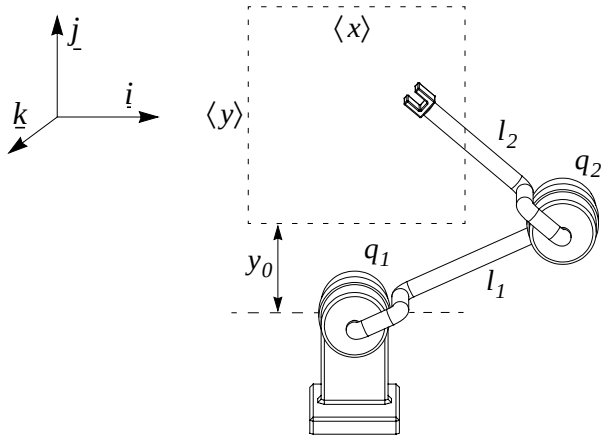


Figure 4.8: 2-DOF Planar RR Serial Manipulator

$$S_J = Diag\left[\tilde{f}_{\underline{i}}\ \tilde{f}_{\underline{j}}\right] \tag{4.41}$$

$$S_T = Diag\left[\tilde{\ddot{x}}_{\underline{i}}\ \tilde{\ddot{x}}_{\underline{j}}\right] \tag{4.42}$$

By assigning S_J to the identity, the desired effective masses along $\underline{i}$ and $\underline{j}$ can be specified by assigning S_T to the desired acceleration for an applied unit force. The results of three example optimizations are shown in Table 4.1 with $<x>$, $<y> = \pm 5cm$ and $y_0 = 5cm$. A linkage mass of *10 g/cm* and an actuator mass of *50 g* are used to compute *D*.

Table 4.1: Inertial Optimization Results for the Planar RR Manipulator

Desired Acceleration		Optimum		
i axis	j axis	l_1	l_2	GII
1.0	1.0	7.9	8.7	0.096
1.5	1.0	7.5	8.9	0.069
1.0	1.5	8.7	8.3	0.082

The resulting GIIs suggest that the device can be designed to present uniform inertia or to present a smaller inertia along $\underline{j}$ than along $\underline{i}$ but is not easily designed to present a smaller inertia along $\underline{i}$ than along $\underline{j}$.

Since the non-homogeneous actuation technique also removes the physical units from joint-space, it ensures a meaningful result even when a manipulator combines rotary and prismatic actuators. Consider, for example, the 3-DOF serial manipulator shown in Figure 4.9 which has only one geometric design parameter l_1. Since q_2 applies a force rather than a torque, it is optimized using the S_J matrix shown in (4.43). Computation of the Jacobian matrix (4.44) is shown in Appendix A.10.

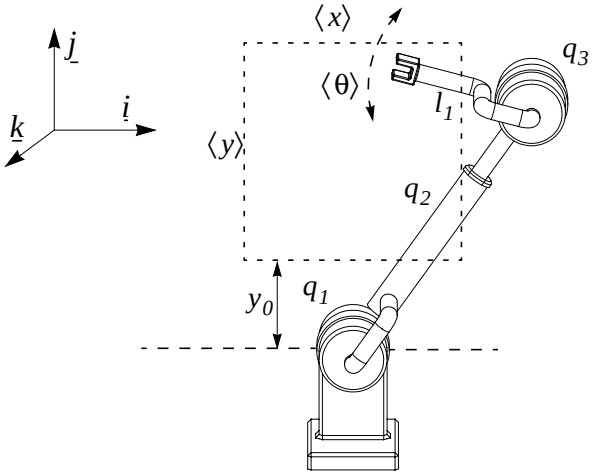


Figure 4.9: 3-DOF Planar RPR Serial Manipulator

$$S_J = Diag\left[\tilde{\tau}_1/\tilde{\tau}_3 \;\; \tilde{f}_2/\tilde{\tau}_3 \;\; 1\right] \tag{4.43}$$

$$\left[\tau_1 \;\; f_2 \;\; \tau_3\right]^T = J^T\left[f_{\underline{i}} \;\; f_{\underline{j}} \;\; \tau_{\underline{k}}\right]^T \tag{4.44}$$

For $\langle x\rangle$, $\langle y\rangle = \pm 5cm$, $\langle\theta\rangle = \pm 180^\circ$, $y_0 = 5cm$ and the force/torque requirements in (4.39), the optimum robot has $l_1 = 0cm$, $\tilde{\tau}_1/\tilde{\tau}_3 = 3.6$ and $\tilde{f}_2/\tilde{\tau}_3 = 0.9cm^{-1}$ with a GII of 0.203. This robot shows a significantly (95%) better GII than the homogeneous serial arm (Figure 4.5) but in practice may suffer from larger inertia due to its prismatic upper-arm.

4.5 Concluding Remarks

A method has been described that contradicts the claim that scaling matrices possess an arbitrary quality when used to address the unit inconsistency problem (i.e. Doty *et al.* [16]). It is shown that scaling is not arbitrary if the intended use is not arbitrary. When a specific performance goal exists, the choice of scale factors greatly affects the performance measure and, if chosen properly, can result in a drastic improvement in performance. The technique proposed here removes all physical units and allows one to specify a non-homogeneous performance goal for a device and solve for its optimum actuator sizes. This method is the first of its kind to simultaneously consider both geometric and actuator parameters in robot design optimization and tailor the design to a non-uniform performance specification. It can be applied to the GII or any other condition index that uses the singular values of a design matrix.

In the following chapter, a new optimization algorithm is presented. It is a discrete branch-and-bound optimization algorithm that is specifically designed to handle minimax and GII optimization problems.

Chapter 5

The Culling Optimization Algorithm

5.1 Introduction

The GII proposed in Chapter 3 and Chapter 4 describes how well a device and design parameter satisfies a performance criteria. Maximizing the GII, however, is a minimax optimization problem and few existing algorithms are equipped to solve minimax problems, especially ones that involves non-linear, non-differentiable, discontinuous and/or unbounded objective functions such as the condition number or GII. Descent algorithms [20] become trapped in local minima, stochastic approaches [68] have uncertain stopping criteria and the results of a global search become increasingly suspicious as the search resolution is decreased. In this chapter, a new optimization algorithm, the culling algorithm, is described which is specifically designed to handle minimax and GII optimization problems. It is a discrete optimization algorithm which guarantees the same result as a global search but arrives at it orders of magnitude faster. Unlike most optimization algorithms which look for parameters that improve the objective function, the culling algorithm looks for parameters with sub-optimal performance values and culls them from the search space until only the global optimum remains.

This chapter is organized as follows. In Section 5.2, two versions of the optimization algorithm are described, one for GII and another for minimax optimization problems. In Section 5.3 the algorithm is used to optimize and draw general conclusions regarding the kinematic and dynamic isotropy of a 2-DOF pantograph manipulator. In Section 5.4 the algorithm is used to kinematically optimize a 6-DOF Stewart platform. In Section 5.5 the efficiency and sensitivity to initial conditions of the algorithm are measured. In Section 5.7 some concluding remarks are made.

5.2 Description of the Culling Algorithm

The culling algorithm is a discrete optimization algorithm that belongs to the branch-and-bound family of optimization algorithms. It identifies non-optimal parameters and culls them from the search space until only the optimum remains. Two versions are presented for solving GII and minimax optimization problems. The GII culling algorithm optimizes the GII which is defined between *0* and *1* corresponding to poor and ideal performance respectively, over a workspace W which is a constrained set of configurations x for a parameter p. The optimization goal (3.7) is to find the parameter p^* with the best “worst-case”

behavior throughout the workspace *W*. The algorithm is described in (5.1) through (5.10) and uses the following notation:

List of Symbols

i	= looping index
P_i	= set of all parameters in parameter space
p_i	= design parameter
$\hat{p}_i$	= best known design parameter
W	= set of all positions in the workspace
x	= end-effector position
$\underset{\sim}{x}$	= position with the smallest singular value
$\tilde{x}$	= position with the largest singular value
$\underset{\sim}{\sigma}$	= minimum singular value at a position
$\tilde{\sigma}$	= maximum singular value at a position
$\underset{\sim}{\Sigma}_i : P_i \to \Re$	= minimum singular value upper bounding function
$\tilde{\Sigma}_i : P_i \to \Re$	= maximum singular value lower bounding function
s	= performance measure; either GII or k^{-1} as defined in equation (2.2)
$\hat{s}$	= performance measure of best known design parameter

GII Culling Algorithm

$$\text{Set} \quad i = 0, \quad \hat{s}_0 = 0 \tag{5.1}$$

$$\text{Set} \quad \left\{ \begin{matrix} \underset{\sim}{\Sigma}_0(p) = \infty \\ \tilde{\Sigma}_0(p) = 0 \end{matrix} \right\}; \ \forall \ p \in P_0 \tag{5.2}$$

$$\text{Choose} \quad (p_0 = \hat{p}_0) \in P_0 \tag{5.3}$$

REPEAT

$$\text{Find} \quad \underset{\sim}{x}_i = \underset{x \in W}{\arg min} \ \underset{\sim}{\sigma}(p_i, x), \quad \tilde{x}_i = \underset{x \in W}{\arg max} \ \tilde{\sigma}(p_i, x) \tag{5.4}$$

$$\begin{aligned} &\text{if} \ \left(\hat{s}_{i+1} = \frac{\underset{\sim}{\sigma}(p_i, \underset{\sim}{x}_i)}{\tilde{\sigma}(p_i, \tilde{x}_i)} \right) > \hat{s}_i; \ \hat{p}_{i+1} = p_i \\ &\text{otherwise} \qquad\qquad ; \ \hat{p}_{i+1} = \hat{p}_i, \hat{s}_{i+1} = \hat{s}_i \end{aligned} \tag{5.5}$$

$$\text{Set} \quad \left\{ \begin{matrix} \underset{\sim}{\Sigma}_{i+1}(p) = min \ \{\underset{\sim}{\Sigma}_i(p), \ \underset{\sim}{\sigma}(p, \underset{\sim}{x}_i)\} \\ \tilde{\Sigma}_{i+1}(p) = max \ \{\tilde{\Sigma}_i(p), \ \tilde{\sigma}(p, \tilde{x}_i)\} \end{matrix} \right\}; \ \forall \ p \in P_i \tag{5.6}$$

$$\text{Set} \quad P_{i+1} = \left\{ p \in P_i \ \middle| \ \frac{\underset{\sim}{\Sigma}_{i+1}(p)}{\tilde{\Sigma}_{i+1}(p)} > \hat{s}_{i+1} \right\} \tag{5.7}$$

$$\text{Choose} \quad p_{i+1} \in \underset{p \in P_{i+1}}{\arg max} \ \frac{\underset{\sim}{\Sigma}_{i+1}(p)}{\tilde{\Sigma}_{i+1}(p)} \tag{5.8}$$

$$i = i + 1 \tag{5.9}$$

$$\text{UNTIL} \quad \hat{p}_i = p_i \tag{5.10}$$

The algorithm starts with a looping index and best known performance measure of zero (5.1), optimistic bounding functions (5.2) and an initial p_0 that is chosen arbitrarily from P_0 (5.3). Minimum and maximum singular values are calculated for p_i at each x in W (5.4). If p_i produces a better GII than the best known parameter $\hat{p}_i$, p_i becomes the new best known parameter $\hat{p}_{i+1}$ and a new best known performance measure $\hat{s}_{i+1}$ is calculated (5.5). Singular values are calculated for each p in P_i at $\underset{\sim}{x}_i$ and $\tilde{x}_i$ and the corresponding upper $\underset{\sim}{\Sigma}_i(p)$ and lower $\tilde{\Sigma}_i(p)$ bounds are updated (5.6). Note that since all singular values are known for $\underset{\sim}{x}_i$ and $\tilde{x}_i$ from (5.4), bounding may be improved by replacing (5.6) with (5.11).

$$\text{Set} \quad \left\{ \begin{matrix} \underset{\sim}{\Sigma}_{i+1}(p) = min \ \{\underset{\sim}{\Sigma}_i(p), \ \underset{\sim}{\sigma}(p, \underset{\sim}{x}_i), \ \underset{\sim}{\sigma}(p, \tilde{x}_i)\} \\ \tilde{\Sigma}_{i+1}(p) = max \ \{\tilde{\Sigma}_i(p), \ \tilde{\sigma}(p, \underset{\sim}{x}_i), \ \tilde{\sigma}(p, \tilde{x}_i)\} \end{matrix} \right\}; \ \forall \ p \in P_i \tag{5.11}$$

Although any improvement from using (5.11) is attributable to good fortune, the computational cost is negligible and is, therefore, worthwhile. Also note that one or both updates in steps (5.6), (5.11) can be omitted for all p whose ratio of upper and lower bounds is already less (i.e. worse) than $\hat{s}_{i+1}$ since those p will be culled from P_i in (5.7). The p with the largest ratio of upper and lower bounds is chosen as the next candidate p_{i+1} (5.8). (5.4) through (5.9) are repeated until $\hat{p}_i$ is the only parameter left in P_i which conclusively identifies $\hat{p}_i$ as the global optimum (5.10).

Since a parameter is only removed from the search space after it has produced singular values with a ratio worse than that of another parameter $\hat{p}$ for which all singular values have been rigorously computed, the global optimum is guaranteed. Computational savings result from strategically exploring configurations which are likely to identify many parameters as sub-optimal. Expected efficiency, however, relies on the presumption that within a continuous, bounded range of parameters, many of them, particularly those in close proximity to each other, will exhibit similarly favourable or poor behaviour at common configurations. This is a presumption that holds well in robot design problems. Consider, for example, a robot that stretches to its reachable limit when visiting a position in the pre-defined workspace resulting in a minimum singular value of *0*. A small adjustment to one geometric parameter will usually only slightly affect the robot's reachable limit and it will continue to produce very large and/or small singular values at that position. It and all other neighbors of the original parameter are, therefore, likely candidates for being culled from the parameter space when evaluated at that position.

The GII culling algorithm is specifically designed for optimizing the GII but some design problems involve local measures and are of the form shown in (5.12). For example, Hayward *et al.* [25] use the objective function $s(p,x) = \sigma(D(p,x))/\tilde{\sigma}(D(p,x))$ to optimize the mass matrix *D(p,x)* of a planar pantograph haptic interface. Problems of this form can be solved using the minimax culling algorithm described in (5.13) through (5.22). Note that in the minimax version of the culling algorithm, Σ_i is an upper bound on the worst-case performance function *s(p,x)* and does not contain singular values explicitly.

$$p^* = \underset{p \in P_0}{\arg max} \; \underset{x \in W}{min} \; s(p,x) \tag{5.12}$$

Minimax Culling Algorithm

Set $i = 0, \quad \hat{s}_0 = 0$ (5.13)

Set $\underset{\sim}{\Sigma}_0(p) = 1; \ \forall \ p \in P_0$ (5.14)

Choose $(p_0 = \hat{p}_0) \in P_0$ (5.15)

REPEAT

Find $\underset{\sim}{x}_i = \underset{x \in W}{\arg min} \ s(p_i, x)$ (5.16)

if $(\hat{s}_{i+1} = s(p_i, \underset{\sim}{x}_i)) > \hat{s}_i; \ \hat{p}_{i+1} = p_i$
otherwise $; \ \hat{p}_{i+1} = \hat{p}_i, \hat{s}_{i+1} = \hat{s}_i$ (5.17)

Set $\underset{\sim}{\Sigma}_{i+1}(p) = min \ \{\underset{\sim}{\Sigma}_i(p), \ s(p, \underset{\sim}{x}_i)\}; \ \forall \ p \in P_i$ (5.18)

Set $P_{i+1} = \{p \in P_i \mid \underset{\sim}{\Sigma}_{i+1}(p) > \hat{s}_{i+1}\}$ (5.19)

Choose $p_{i+1} \in \underset{p \in P_{i+1}}{\arg max} \ \underset{\sim}{\Sigma}_{i+1}(p)$ (5.20)

$i = i + 1$ (5.21)

UNTIL $\hat{p}_i = p_i$ (5.22)

The minimax culling algorithm is illustrated by a step-by-step example using the planar elbow manipulator of Figure 3.1. The performance index is the ratio of singular values of the Jacobian matrix $s(p, x) = \underset{\sim}{\sigma}(J(p, x))/\tilde{\sigma}(J(p, x))$ and the parameter space is reduced to a single dimension by calculating the minimum forearm length l_2 from (5.23) which ensures that the boundaries of the usable and reachable workspaces shown in Figure 5.1 are separated by a minimum safety margin of length K. In other words, l_2 is chosen such that $\{k_1, k_2\} \geq K$.

$$l_2 = max(\left\| \sqrt{x_{max}^2 + y^2} - l_1 \right\|, \|y - l_1\|) + K \tag{5.23}$$

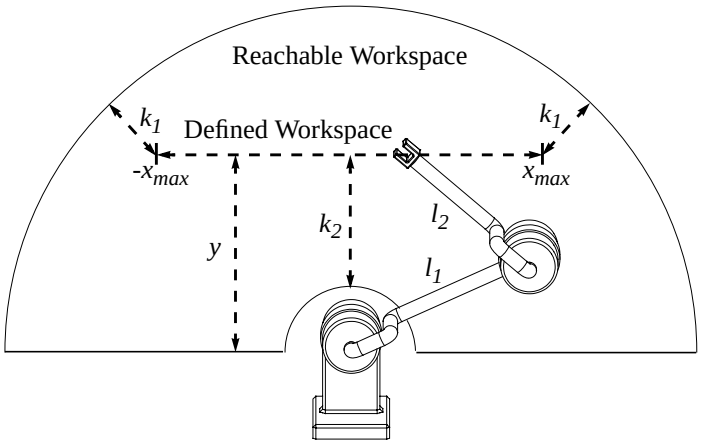


Figure 5.1: Workspace of a Planar Elbow Manipulator

The performance index is non-linear, non-differentiable and contains local minima and maxima in both its operation and parameter spaces which, in this case, can be verified by brute force since the problem has only one workspace dimension (x) and one design parameter (l_1). The performance index $s(l_1,x)$ is plotted against x and l_1 in Figure 5.2 with $x_{max}=5$, $y=2$, $K = 0.4$ and $2 \le l_1 \le 8$.

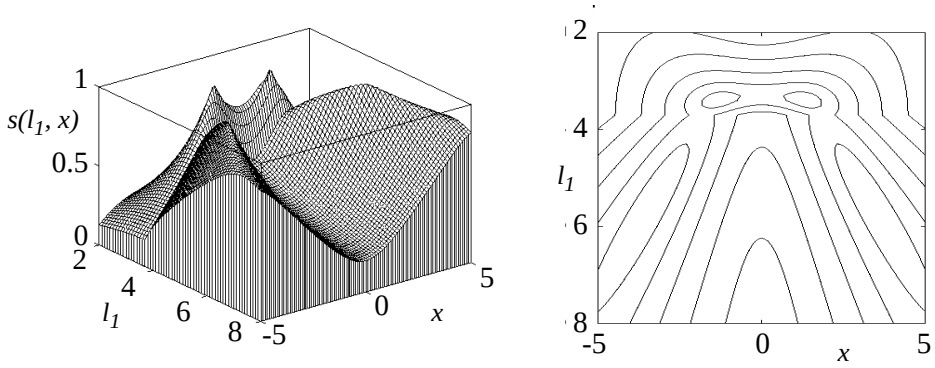


Figure 5.2: Surface and Contour Plots of Dexterity

An intermediate value of l_1=6 is picked as the initial best known parameter $\hat{p}_0$ (5.15). The workspace of l_1=6 is searched to find that x_0=0 minimizes $s(l_1,x)$ with a value of $s(6,0)=0.28$ (5.16) and the condition index of the best known parameter is updated (5.17). The parameter space at x_0=0 is searched, the upper bound $\tilde{\Sigma}_1$ is updated for each parameter (5.18), (5.24) and all sub-optimal parameters are culled (shaded regions in Figure 5.3) from the parameter space (5.19). l_1=3.3 has the largest upper bound and is the next candidate p_1 (5.20).

$$\tilde{\Sigma}_1(l_1) = s(l_1, 0) \tag{5.24}$$

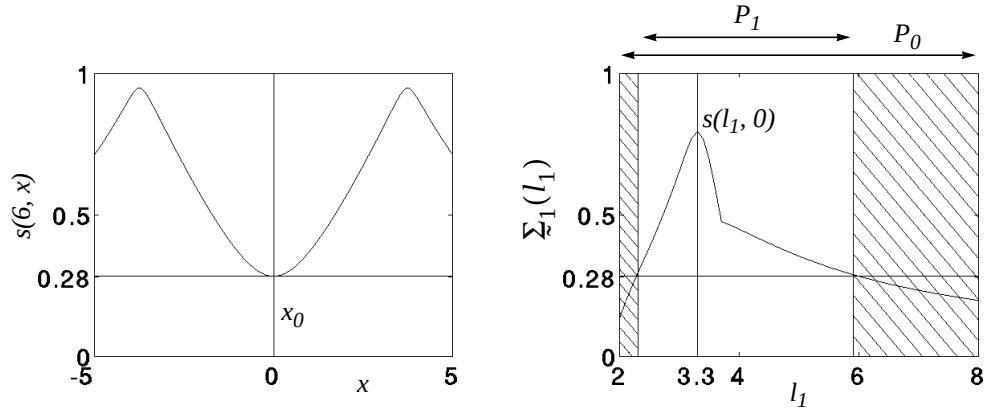


Figure 5.3: First Culling of Non-Optimal Geometries

The workspace of l_1=3.3 is searched to find that x_1=–5 minimizes $s(l_1,x)$ with a value of $s(3.3,-5)=0.16$ (5.16) so l_1=6 is still the best known parameter (5.17). The parameter space $P_1=\{2.3...5.9\}$ at x_1=–5 is searched, the upper bound $\tilde{\Sigma}_2$ is updated for each parameter (5.18), (5.25) and all sub-optimal parameters are culled from the parameter space (5.19). l_1=4.5 has the largest upper bound and is the next candidate p_2 (5.20) (see Figure 5.4).

$$\tilde{\Sigma}_2(l_1) = min\{\tilde{\Sigma}_1(l_1), s(l_1, -5)\} = min\{s(l_1, 0), s(l_1, -5)\} \tag{5.25}$$

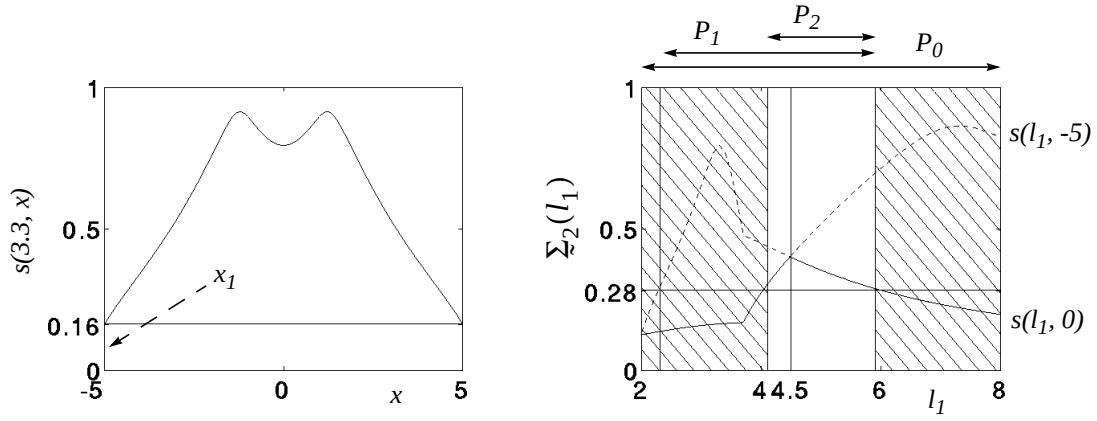


Figure 5.4: Second Culling of Non-Optimal Geometries

The workspace of l_1=4.5 is searched to find that x_2=0 minimizes $s(l_1,x)$ with a value of $s(4.5,0)=0.4$ (5.16) making it the new best known parameter (5.17). All remaining parameters left in the parameter space $P_2=\{4.1...5.9\}$ have upper bounds that are below this value and are culled from the parameter space (5.18), (5.19). Only l_1=4.5 remains and is, therefore, the global optimum p^* (5.22). This is verified by performing

an exhaustive search and plotting the worst-case performance of each geometry. The optimum geometry is shown to be l_1=*4.5* in Figure 5.5.

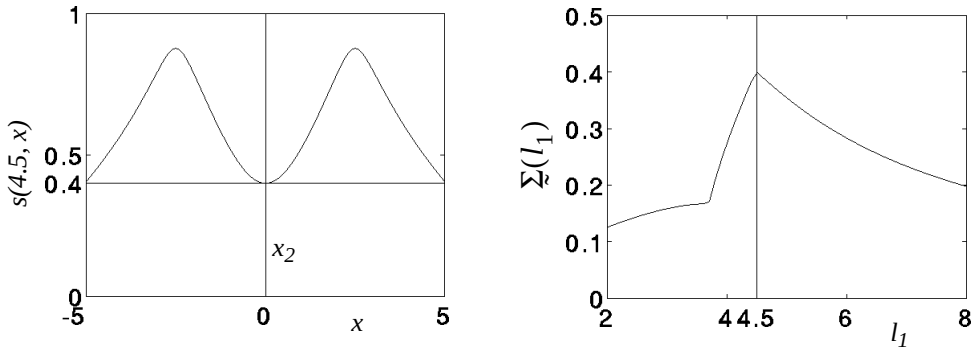


Figure 5.5: Final Culling and the Optimum Solution

The culling algorithm belongs to the branch-and-bound family of optimization algorithms but is unconventional in that it is strictly geared toward solving minimax problems. It performs all of its bounding through explicit function evaluations and, therefore, does not require any worst-case estimates of the objective function. Each time the condition index of a candidate parameter is rigorously computed, the value is used to push up the lower bound on the performance index of the optimum parameter and each time the condition index is computed for a parameter at a new position, the value is used to push down the upper bound on the performance index of that parameter. If the lower bound on the optimal performance index exceeds the upper bound on the performance index of any parameter, that parameter is culled from the parameter space. This is how the culling algorithm performs bounding. It performs branching by alternating between workspace searches for a single parameter and parameter space searches for a single position with the parameter and position chosen from the results of the previous iteration.

5.3 Design of a Planar Five-Bar Linkage

Karidis *et al.* [33] design a dynamically balanced five-bar linkage for micro-probing while Hayward *et al.* [25] optimize a five-bar linkage for use as a planar haptic interface. The culling algorithm and new definition of global isotropy are used to re-examine the five-bar linkage based haptic interface for both kinematic and dynamic conditioning. Since the performance of a robot is affected by many factors that include robot geometry, workspace size and workspace position, a general representation of a five-bar linkage with a square workspace is used which is shown in Figure 5.6. It contains design parameters for both the geometry of the robot and the position and orientation of the robot's workspace so that symmetry

and positioning guidelines can be determined to simplify future design optimizations involving a similar device and workspace. Computation of the Jacobian matrix (5.26) is shown in Appendix A.4.

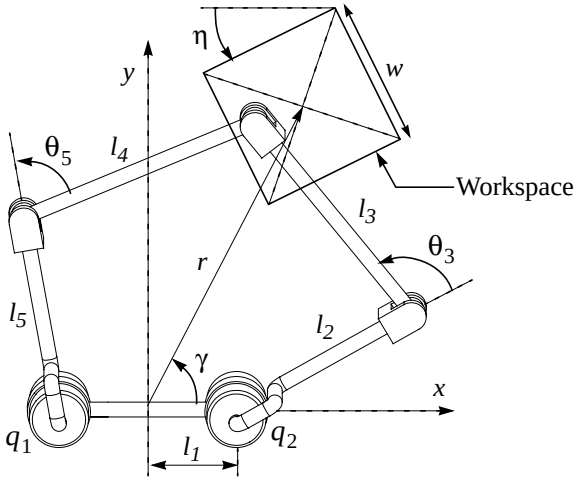


Figure 5.6: Five-Bar Linkage

$$\left[\dot{q}_1 \ \dot{q}_2\right]^T = J(p, x)\left[\dot{x} \ \dot{y}\right]^T \tag{5.26}$$

The kinematic GII is optimized with seven free design parameters (l_1 through l_5, γ and η) and with r and w fixed to avoid a trivial result since isotropy improves when $r \to \infty$ or as $w \to 0$. For $r=w=10$ and $\Delta w=0.1$ (Δw is the discrete sample spacing in the workspace) the search space and solution obtained by the GII culling algorithm are shown in Table 5.1.

Table 5.1: Design Parameter Search Space and Kinematic Optimum

Parameter	Min. Val.	Max. Val.	Resolution	Optimum
l_1	0	3	0.5	1.5
l_2	4	10	0.5	7.5
l_3	7	14	0.5	9.5
l_4	7	14	0.5	9.5
l_5	4	10	0.5	7.5
γ	0	$\pi/2$	$\pi/20$	$\pi/2$
η	0	$\pi/2$	$\pi/20$	0

The global optimum has left/right symmetry of both the robot and workspace. Future optimizations, therefore, need only consider three parameters (l_1, $l_{2,5}=l_2=l_5$ and $l_{3,4}=l_3=l_4$) and half of the workspace ($x \geq 0$). This simplifies the problem sufficiently to allow further generalization. Varying r while constraining the elbow angles $\theta_3, \theta_5 \in \{0...\pi\}$ shows the optimum posture for a range of r values where a negative value

of *r* corresponds to a workspace positioned below the line connecting joints $\{q_1, q_2\}$. Figure 5.7 shows the optimum geometry, GII and posture for $r \in \{-25...25\}$, $\gamma=\eta=\pi/2$ and the parameter space shown in Table 5.2. Note that the search space constrains the robot and workspace to a *30*x*30* square area.

Table 5.2: Reduced Design Parameter Search Space

Parameter	Min. Val.	Max. Val.	Resolution
l_1	0	15	0.2
$l_{2,5}$	5	30	0.2
$l_{3,4}$	5	30	0.2

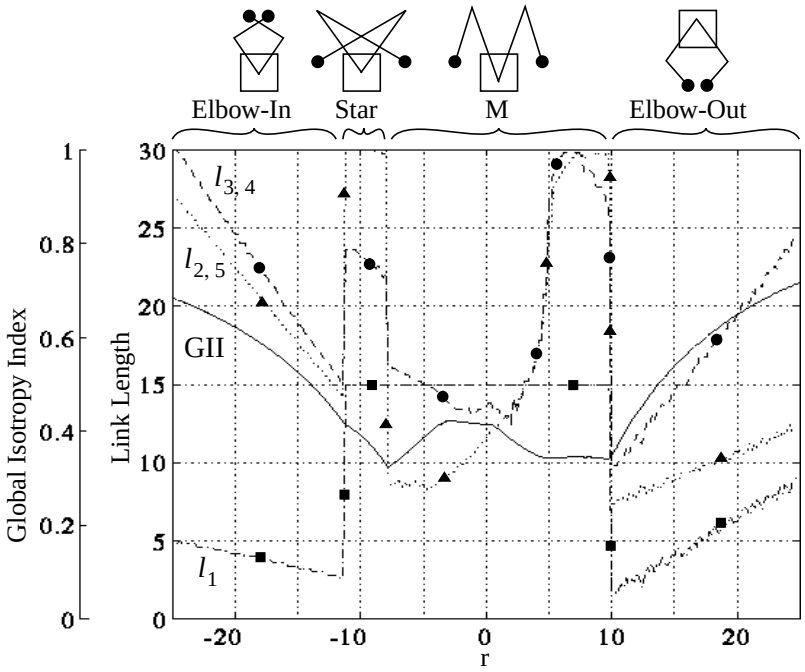


Figure 5.7: Optimum Parameters and Postures of a Five-Bar Linkage

The GII curve is non-smooth and the optimal parameter curves (l_1, $l_{2,5}$ & $l_{3,4}$) are discontinuous in *r*. Parametric discontinuities occur at the intersections of optimum GII curves for different postures. Consider the region around *r=10*. The optimum GII of the "M" posture is relatively level while the optimum GII of the elbow-out posture increases with *r*. When the curves intersect, the optimum posture switches from "M" to elbow-out and the parametric curves experience a jump. There are clearly two viable ranges for *r*. Values of $r \in \{-3...1\}$ are acceptable in which case the "M" posture is best with the workspace positioned between the actuators. Magnitudes greater than 10 are also acceptable in which case the elbow-out posture is preferred. While elbow-in achieves GIIs similar to elbow-out for similar magnitudes of *r* it requires longer

physical link lengths ($l_{2,5}$ & $l_{3,4}$). The inertial implications of this distinguish elbow-out as the better posture. For all other values (i.e. $r \in \{\{-12...-3\},\{1...10\}\}$), the optimal postures combine long link lengths with poor GIIs and should be avoided.

For an inertial optimization, it is assumed that the device is held with a light fingertip grip so hand inertia is neglected. It is also assumed that inertia is dominated by the linkages which are made from circular cross-section 2024-T4 aluminum tubing. The mass matrix is computed as shown in Appendix A.5. It is debatable whether it is preferable to optimize the mass matrix for isotropy or scale. Since isotropic mass is not particularly important if the magnitude is small, maximum mass is minimized by considering the normalized performance index involving only the maximum singular value of the mass matrix shown in (5.27). Since this measure can be computed locally, the minimax culling algorithm is used.

$$s(p, x) = \frac{1}{1 + \tilde{\sigma}(D(p, x))} \tag{5.27}$$

It was observed that the culling algorithm becomes significantly less efficient when the value obtained from the objective function does not change throughout large, connected portions of the workspace. This occurs when the desired workspace extends beyond the reachable workspace where $s(p,x)$ is equal to 0. Efficiency is improved by augmenting the condition index with a different function for unreachable points $s'(p,x)$ which assigns a value of 0 to a position on the boundary of the reachable workspace and a value of -1 to a position infinitely far from it (5.28). The augmented condition index $s''(p,x)$ (5.29) is still a normalized index since its magnitude is less than or equal to unity (5.30) with positive values (5.27) for reachable positions and negative values (5.28) for unreachable positions.

$$s'(p, x) = \frac{1}{1 + \text{distance from w/s}} - 1 \tag{5.28}$$

$$s''(p, x) = \begin{cases} s(p, x) & \text{if position is reachable} \\ s'(p, x) & \text{otherwise} \end{cases} \tag{5.29}$$

$$s''(p, x) \in \{-1 \ldots 1\} \tag{5.30}$$

Since $s''(p,x)$ can have either a positive or a negative value, the initialization step (5.13) must be replaced with (5.31) but the remainder of the algorithm is compatible.

$$\text{Set} \quad i = 0, \quad \hat{s}_0 = -1 \tag{5.31}$$

Two optimizations are performed, both with $w=10$, $\Delta w=0.1$ and $\omega=200\pi$ (100 Hz) but searching two different parameter spaces. The first optimization (parameter space A) attempts to find a design that is both

dynamically and kinematically favourable by picking the dynamic optimum from the kinematic optima described in Figure 5.7. There is only one free parameter *r* while all other geometric parameters (l_1, $l_{2,5}$ & $l_{3,4}$) are chosen as a function of *r* from Figure 5.7. Although this parameter space is very small, it ensures a kinematic GII that is no less that *0.3*. The second optimization (parameter space B) includes all geometric combinations searched during the kinematic optimizations so all four geometric design parameters l_1, $l_{2,5}$, $l_{3,4}$ and *r* are free. Note that although a trivial result is obtained when *r* is free during kinematic optimization, this does not occur during dynamic optimization. The two parameter spaces and optimum solutions are shown in Table 5.3.

Table 5.3: Design Parameter Search Space and Inertial Optimum

Parameter Space A				
Parameter	Min. Val.	Max. Val.	Resolution	Optimum
l_1	from Figure 5.7			1.6
$l_{2,5}$	from Figure 5.7			7.6
$l_{3,4}$	from Figure 5.7			9.8
r	-25	25	0.2	10.4

Parameter Space B				
Parameter	Min. Val.	Max. Val.	Resolution	Optimum
l_1	0	15	0.2	0.4
$l_{2,5}$	5	30	0.2	7.2
$l_{3,4}$	5	30	0.2	8.2
r	-25	25	0.2	-8.6

Parameter space A has a narrow scope but guarantees a kinematically favourable solution. Parameter space B ignores kinematic conditioning but results in the dynamic global optimum. In order to decide which solution is most favourable overall, the two are compared by the sensitivity analysis in Table 5.4. Since both solutions have similar dynamic performance but solution A has significantly better kinematic performance, solution A (l_1*=1.6*, $l_{2,5}$*=7.6*, $l_{3,4}$*=9.8*, *r=10.4*) is concluded to be the best overall design.

Table 5.4: Sensitivity Analysis of Kinematic and Inertial Conditioning

	Kinematic GII		Maximum Inertia Index	
	Value	% Drop	Value	% Drop
Solution A	0.366	n/a	0.896	-3.6%
Solution B	0.246	-32.8%	0.929	n/a

Special purpose robots such as haptic interfaces are hampered by large changes is singular values as well as by sudden changes in singular values. Smoothness is checked as a secondary measure for the optimal design by plotting the minimum and maximum singular values of both the Jacobian and mass matrix over the workspace. As seen in Figure 5.8, they are both smooth and even have regions of perfect local isotropy (i.e. intersection of minimum and maximum singular values).

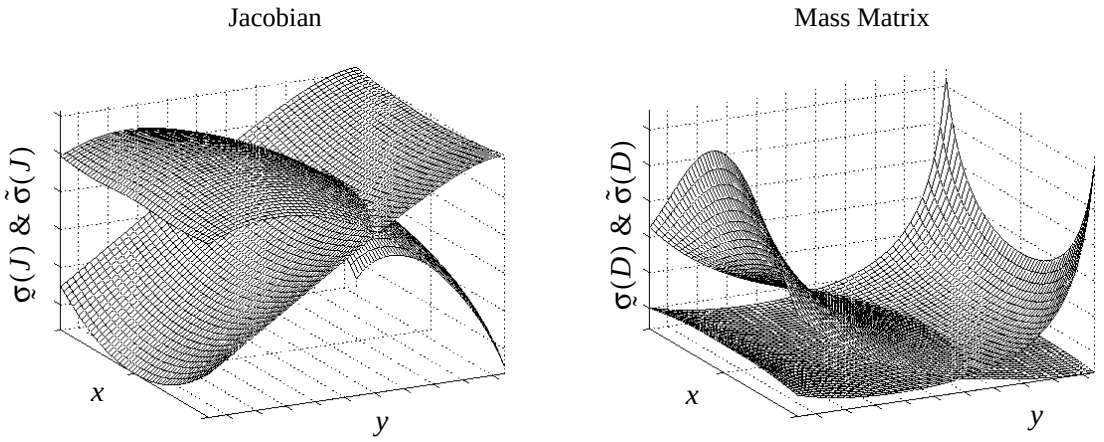


Figure 5.8: Singular Value Plots for the Jacobian and Mass Matrix

5.4 Design of a 6-DOF Stewart Platform

The GII culling algorithm is next used to attempt a higher dimensional design optimization involving the 6-DOF Stewart Platform manipulator shown in Figure 5.9.

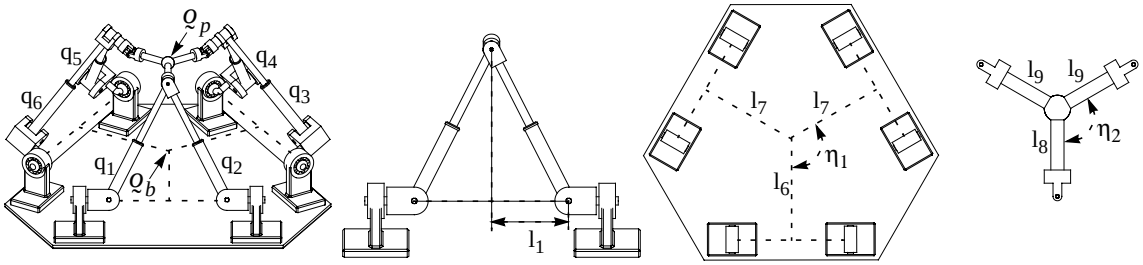


Figure 5.9: 6-DOF Stewart Platform

The manipulator is designed by exploring combinations of the design parameters l_1, l_6, l_7, $\eta_1=\eta_2=\eta$ and the ratio $l_6/l_8=l_7/l_9=L$. The workspace is a cubic volume with all sides of length *10 cm* centred *25 cm* above the centre of the base. The range of orientations includes a solid angle of *30˚* traced by the $\underline{k}$ axis of the platform coordinate frame combined with all rotations of up to *30˚* about that axis (see Appendix B).

Symmetry of the workspace about the $\underline{jk}$ plane is used to reduce the number of geometric robot parameters by imposing the same symmetry upon the robot. The distances between the centre of the platform and the left and right pairs of actuators are equal (l_7) and the platform is shaped similarly to the base but is scaled by the ratio L. For the purposes of this example, the physical units of the Jacobian are normalized using a reasonable task-space torque/force scale factor of *12cm* in the S_T matrix. The kinematic GII is optimized using the discrete parameter space shown in Table 5.5 and the discrete workspace shown in Table 5.6.

Table 5.5: Design Parameter Search Space for the Stewart Platform

Parameter	Minimum	Maximum	Resolution	Optimum
l_1	1	20	0.5	16.5
l_6	1	20	0.5	10.0
l_7	1	20	0.5	10.0
L	0.5	1.5	0.1	0.7
η	110°	130°	1°	118

Table 5.6: Workspace of the Stewart Platform

Dimension	Minimum	Maximum	Resolution	Total
Translation				
i axis	0	5	0.625	9
j axis	-5	5	0.625	17
k axis	-5	5	0.625	17
Rotation				
i, j axes	Uniformly Sampled Solid Angle			168
k axis	-30˚	30˚	5	13

The optimum parameters shown in Table 5.5 produce a GII value of *0.281*. To see if the result is sensitive to sample spacing, the discretization resolution is halved for all parameters and workspace dimensions and the optimization is repeated. A GII of 0.285 results from the parameters l_1*=17*, l_6=l_7*=10*, η=120˚ and L=0.7. Since the GII fluctuates by only 1.4% and the optimum parameters differ by a maximum of only 3%, it is concluded that the original resolution is adequate and that little improvement can be expected in terms of a better (parameter space resolution) or more trustworthy (workspace resolution) solution from reducing the sample spacing.

5.5 Efficiency of the Culling Algorithm

The culling algorithm belongs to the branch and bound family of optimization algorithms where all bounds are determined by explicit function evaluations. It avoids redundant evaluations by eliminating parameters that are shown to be sub-optimal and, therefore, always converges to a global optimum within the discretized parameter space. Each loop iteration removes at least one parameter from contention so the number of potential loop iterations is bounded by the dimension of the parameter space and the stopping criterion is always satisfied in finite time. A worst case scenario of no culling whatsoever results in an exhaustive global search. As with most optimization algorithms, efficiency depends on the objective function and initial conditions. While the algorithm makes no efficiency guarantees, experience with robot design problems has consistently shown dramatic improvement over a global search with low sensitivity to initial conditions. Table 5.7 compares the number of objective function evaluations performed by the culling algorithm to those required by a global search for the optimizations described in this paper. Notice that efficiency tends to increase with larger parameter spaces.

Table 5.7: Comparison of Computational Effort for Culling and Global Search

Optimization	Workspace Size	Param Space Size	Global Search : Culling Ratio
Table 5.1	1.02×10^{4}	2.66×10^{7}	3670 : 1
Table 5.2	5151	1.21×10^{6}	1910 : 1[A]
Table 5.3 A	5151	251	82 : 1
Table 5.3 B	5151	3.03×10^{8}	4440 : 1
Table 5.5	5.68×10^{6}	1.37×10^{7}	1.79×10^{6} : 1

[A.] *Typical ratio for each r value*

The algorithm is demonstrated to be effective at solving both modest (i.e. few dimensions) and complex (i.e. many dimensions) problems. An example involving the Stewart Platform with a workspace containing over *5 million* elements and a parameter space containing over *13 million* elements resulted in an effort reduction of over *6* orders of magnitude over a global search. This calculation which took less than *6* hours to solve on a Silicon Graphics O^2 workstation using culling is estimated to require over *1,200* years to solve by a global search using a similar machine. By looking at the severity of culling during each loop iteration, we can identify where most of the computation effort takes place. The Stewart Platform optimization described in Table 5.5 and Table 5.6 was solved in *5* loop iterations. Table 5.8 shows for each loop iteration the number of parameters initially present, the number of parameters culled, the maximum number of objective function evaluations required to update the singular value upper and lower bounds of all remaining parameters and the number of objective function evaluations required to search all

workspace locations of the candidate parameter. Note that because the GII culling algorithm was used, up to two parameter evaluations (two different workspace locations for each parameter) can occur during each loop iteration.

Table 5.8: Computational Effort in Each Culling Loop

Loop Number	Number of Parameters	Parameters Culled	Parameter Evaluations	Workspace Evaluations	Total Evaluations
0	13,702,689	12,498,014	27,405,378	5,680,584	33,085,962
1	1,204,675	1,204,049	2,409,350	5,680,584	8,089,934
2	626	353	1,252	5,680,584	5,681,836
3	273	271	546	5,680,584	5,681,130
4	2	1	4	5,680,584	5,680,588
5	1	0	0	0	0
Total	n/a	13,702,688	29,816,530	28,402,920	58,219,450

It is, however, possible to skip equations (5.6) and/or (5.18) so the number of total parameter space evaluations are bounded from below by the number of parameters in parameter space P_0. The minimum, maximum and actual number of calculations that took place are shown in Table 5.9.

Table 5.9: Number of Objective Function Evaluations

	Parameter Evaluations	Workspace Evaluations	Total Evaluations
Minimum	1.37×10^7	2.84×10^7	4.21×10^7
Maximum	2.98×10^7	2.84×10^7	5.82×10^7
Actual	1.52×10^7	2.84×10^7	4.36×10^7

Computational effort is shared evenly between parameter space and workspace searches. Parameter space searches leave little room for improvement since an average of only one or two positions are visited for each parameter but workspace searches might benefit from a different, perhaps stochastic, searching strategy. For example, workspace positions could be visited in steps (5.4) and (5.16) in a random order, stopping when $\hat{s}_{i+1} < \hat{s}_i$ which identifies p_i as suboptimal. If, however, this condition is never met, the workspace search can only be terminated after an exhaustive search has been completed so that the guarantee of global optimality is not compromised. Since the $\tilde{x}_i$ and/or $\underset{\sim}{x}_i$ produced from a truncated search are not expected to be as good as those produced by an exhaustive search, culling is likely to be less severe after the subsequent parameter space search but the net gain from truncating the workspace search may exceed the net loss from the reduction in parameter culling.

To determine sensitivity to initial conditions, the Stewart Platform optimization was repeated two more times, once with the initial condition set to the optimum solution from Table 5.5 and again with the initial condition set to the parameter farthest from the optimal solution (i.e. $l_1=1.0$, $l_6=l_7=20.0$, $\eta=130^\circ$ and $L=1.5$). The number of function evaluations resulting from each of the three trials is presented in Table 5.10 and shows that a good first guess reduced computational effort by 12.7% over the mean value. One way of obtaining a good first guess is to increase the sample spacing and perform a coarse optimization of the device. The optimization of the Stewart Platform with double the original sample spacing that was described earlier required 8.77×10^5 function evaluations to complete. Adding this initial investment to the required computational effort when starting from the optimal solution reduces its 12.7% gross effort reduction to a 9.9% net effort reduction over the mean value. Since the effort reduction from having a good starting point is only expected to be in the neighborhood of 10%, it is only practical to pursue when the problem is very large and is expected to take a long time to complete. Otherwise, it may be more practical to choose the starting point by making an educated guess at a favourable design parameter value.

Table 5.10: Sensitivity to Initial Conditions

Initial Conditions	Number of Evaluations	% Change From Mean
Arbitrary	4.36×10^7	+4.6%
Optimum	3.64×10^7	-12.7%
Farthest from Optimum	4.52×10^7	+8.4%
Mean	4.17×10^7	0%

Due to the low computational overhead of the culling algorithm, almost all of the processing power is consumed by objective function evaluations. The effort required to calculate a Jacobian and its singular values overwhelms the few conditional checks and assignment statements associated with the algorithm itself, especially in the case of a 6-DOF device. Since the culling algorithm performs blocks of function evaluations for large sets of parameters and positions where the order of evaluations is of no importance, the algorithm is easily adapted to machines with parallel processing capabilities. One could reasonably expect a near linear relationship between completion time and the number of available processors.

5.6 Assumptions and Limitations of the Culling Algorithm

Although the culling algorithm demonstrates high levels of efficiency when used to solve robot design problems, it is not without its limitations. As pointed out earlier, it does not make any performance guarantees. Its efficiency depends on the assumption that many design parameters will behave similarly at common configurations. For example, if one design parameter is known to perform poorly at a given

position in the workspace, it is expect that other parameters in the search domain will also behave poorly at that same position. This assumption tends to hold in robot design problems. A further assumption is that the performance associated with a design parameter is not expected to be uniform over a large continuous range of positions. As pointed out in Section 5.3, this is an assumption that breaks down when the desired workspace extends beyond the reachable limit of the robot and design parameter. However, this limitation is overcome by defining a piecewise objective function such as $s''(p,x)$ in equation (5.29) which uses different measures for evaluating reachable and unreachable (singular) positions. Note that both of these assumptions are necessary for the culling algorithm to be efficient but neither are necessary for it to work.

Another limitation of the culling algorithm is that it is only applicable to minimax design problems. It is not useful for designing a device with a degenerate workspace (i.e. comprised of just one position) since this reduces the culling algorithm to a global search. A further limitation of the culling algorithm is that it is a discrete algorithm so the real global optimum could be overlooked if the parameters and/or workspace are discretized too coarsely. Of course, the severity of this limitation is dependent on the number of parameters and the computing resources that are available.

5.7 Concluding Remarks

An optimization algorithm has been described which belongs to the branch-and-bound family of optimization algorithms and is specifically designed for GII and minimax optimization problems. It repeatedly uses the worst configuration of one parameter to eliminate others from contention until only the optimum remains. The approach guarantees convergence, finite time termination and a global result and can be used with objective functions that are nonlinear, non-differentiable, discontinuous, non-convex, or unbounded. Although it could theoretically perform the equivalent of a global search, it has consistently demonstrated substantial performance improvements over a global search when applied to practical robot design problems. One design example that took just under *6* hours to solve on a Silicon Graphics O^2 workstation using the culling algorithm is estimated to require over *1,200* years to solve by a global search on a similar machine. The culling algorithm, therefore, allows unsophisticated computer hardware to solve high dimensional problems that would otherwise be too computationally demanding to attempt.

The GII and culling algorithm have been presented as a means of optimizing a robot design for a given application. These tools are applied to the design of a haptic interface. In the following chapter, an appropriate design criteria for a haptic interface is established and tangible performance specifications are obtained through biomechanical studies.

Chapter 6

Haptic Interface Performance Requirements

6.1 Introduction

Haptic interfaces present a difficult mechanical design problem. It is just as important for a haptic device to be light and backdriveable as it is for it to be stiff and unyielding. This has led to much debate over what the most effective performance goal for a haptic interface is. Arguments have been made for kinematic [25] and inertial [48] isotropy, impedance [42], acceleration [64], stability [9] and so on. In this chapter, a number of performance objectives are discussed. Models describing the interaction between a human hand and a haptic device are used to motivate a design criteria for an impedance type haptic interface and biomechanical studies are conducted to obtain target values. These values are used in Chapter 7 to carry out a practical design example.

This chapter is organized as follows. In Section 6.2 system models are used to motivate a haptic interface design criteria. In Section 6.3 biomechanical experiments are conducted to obtain the static force and velocity capabilities of the human hand. In Section 6.4 some concluding remarks are made.

6.2 Performance Objectives

Many design specifications are relevant to the performance of a haptic interface. They include workspace size, position bandwidth, force magnitude, force bandwidth, velocity, acceleration, effective mass, accuracy, and so on. A survey of proposed values for many of these design specifications can be found in Section 2.4. Most of these values are derived from the inherent capabilities of a human hand since this is the environment of a haptic interface.

Physically, a human hand is comprised of a skeletal structure which presents a position dependent impedance Z_s and muscles which can either contract to apply a force f_h or stiffen to present an impedance Z_m to its environment. In Figure 6.1, the total hand impedance Z_h is the combination of voluntary muscular impedance Z_m and non-voluntary skeletal impedance Z_s. Its combined behaviour is adapted by the nervous system to simulate a force source when interacting with a stiff environment (e.g. an open-circuit), a velocity source when navigating a compliant environment (e.g. a short-circuit) and an impedance when restraining a non-passive environment (e.g. an active source).

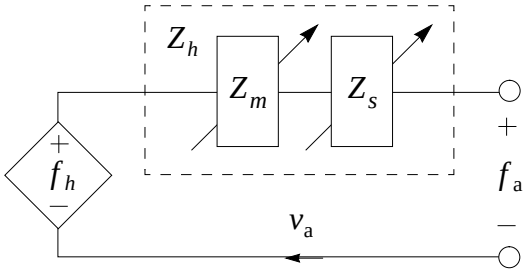


Figure 6.1: Human Hand Model

A haptic interface displays tactile information by presenting an impedance to the human hand. Of course, impedance must be varied in an intelligent fashion so the device includes active sources but to ensure stability, these sources can only be used to dissipate energy and must not supply any. Two types of devices can be used to implement an intelligent variable impedance, an impedance device and an admittance device. An impedance device is easily backdriveable in its passive state and typically uses direct drive or moderately geared actuators to adjust its effective impedance by applying a force based on position and velocity. An admittance device is the dual of an impedance device. It is not backdriveable in its passive state and typically uses highly geared actuators to adjusts its effective impedance by inducing motion based on applied force.

An impedance device senses position $s^{-1}v_a$, velocity v_a and possibly acceleration sv_a and responds with a force f_a comprised of an active force f_r from the motors and a passive force from the mechanical impedance $Z_r(s)$ of the device. The effective impedance experienced by the hand $Z_e(s)$ (6.1) may contain effective mass M_e, damping B_e and stiffness K_e terms (6.3) which are contributed in part by the mass M_r and damping B_r of the robot mechanism (6.2).

$$f_a = Z_r(s)v_a + f_r = Z_e(s)v_a \tag{6.1}$$

$$Z_r(s) = M_r s + B_r \tag{6.2}$$

$$Z_e(s) = M_e s + B_e + K_e s^{-1} \tag{6.3}$$

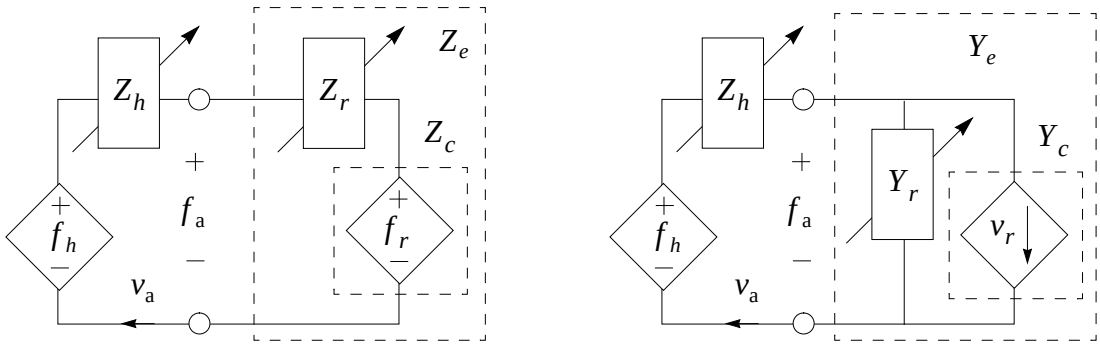
An admittance device senses applied force f_a and responds by inducing a velocity v_a which is comprised of both active velocity v_r from the motors and passive velocity due to the robot's non-zero mechanical admittance $Y_r(s)$. The total effective admittance experienced by the hand $Y_e(s)$ is shown in (6.4).

$$v_a = Y_r(s) f_a + v_r = Y_e(s) f_a \tag{6.4}$$

Combining the human hand model with the haptic device models results in the haptic systems shown in Figure 6.2 where $Z_c(s)$ in (6.5) and $Y_c(s)$ in (6.6) are the impedance and admittance simulated through active control.

$$Z_c(s) = v_r / f_a \tag{6.5}$$

$$Y_c(s) = f_r / v_a \tag{6.6}$$



Impedance System Admittance System

Figure 6.2: Haptic Systems

The high inertia and friction that exists in a highly geared admittance device inhibits high frequency transitions (i.e. switching between free motion and hard contact) and is difficult to actively compensate resulting in low transparency and sluggish free motion that can be tiring during prolonged use. Therefore, we are more interested here in design issues related to impedance devices.

As discussed by Colgate and Schenkel [9], it is difficult to actively compensate for the physical dynamics of a mechanical system without compromising stability. Therefore, a haptic device's minimum impedance is decreased by a reduction in mechanical impedance. Salcudean and Vlaar [64] also show that even when high static forces are not available, high stiffnesses can be simulated by inducing high acceleration. Therefore, it follows that both high and low impedance emulation are improved and the impedance range is widened when mechanical impedance is reduced. Reducing mechanical impedance can be done in a number of ways that do not necessarily involve kinematic design and the mass matrix. It may in fact be sufficient to use a parallel device, light weight materials, low friction joints and counterbalances so that kinematic design can be focussed on other criteria such as force.

From Figure 6.2, it is apparent that if Z_r is sufficiently small, an impedance device's performance is dictated by its force capabilities. Since actuators can always be scaled to meet any force magnitude requirements, it does not make much sense to maximize force through kinematic design. It is more

practical to focus instead on isotropy. Since a device is only as good as its worst-case performance inside its workspace, improving isotropy strengthens its weaknesses and allows smaller actuators to be used. This results in lower rotor inertia, consistently stiff virtual environments, improved compactness and lower cost. Force isotropy is achieved by making the Jacobian transpose $J^T(x)$ isotropic.

A similar argument can be made for measurement accuracy. Sensor resolution can always be increased to meet minimum standards but a device that is kinematically isotropic will tolerate coarser sensors and exhibit greater consistency with a smaller price tag and footprint. Since the Jacobian matrix $J(x)$ is a velocity transformation that also indicates proximity to singular positions, both position and velocity measurements are improved by making $J(x)$ isotropic, thereby improving both virtual stiffness and damping. Note that an isotropic Jacobian $J(x)$ does not imply an isotropic Jacobian transpose $J^T(x)$ since they are not scaled in the same way (see Chapter 4).

For the impedance device designed here, static force isotropy is used as the criteria for selecting geometric and actuator design parameters, kinematic isotropy is used as the criteria for selecting sensor resolutions and mechanical impedance is minimized by considering devices with primarily base mounted actuators and by using low impedance components as described above.

6.3 Biomechanical Experiments

For tangible static force requirements, the maximum force/torque capabilities of the human hand are needed. While existing data on maximum sustainable force is available in human-factors and haptic interface literature, it is not easily applied to haptics design. For example, in [1], [2] and [87] maximum forces and torques are measured using a hammer grip. This type of grip enables larger force capabilities but less precision than the pencil grip preferred for a desktop haptic device. Other maximum hand force/torque values can be found in [4] and [51] but they do not distinguish between capabilities in different directions. To obtain this information, a biomechanical experiment was conducted.

The experiment involved 20 participants (12 males, 8 females) ranging between 20-60 years and 45-90 Kg. Subjects included right and left handed people from a variety of disciplines including students, professors, engineers, secretaries and athletes. The test apparatus was an aluminum rod with the same diameter as a standard wooden pencil (*0.63cm*) fixed at each end to a 6-axis force/torque sensor. The rod was oriented vertically, covered with athletic tape to improve traction and mounted in front of an armrest to reduce the force contribution of the arm and shoulder. A photograph of the test apparatus is shown in Figure 6.3.

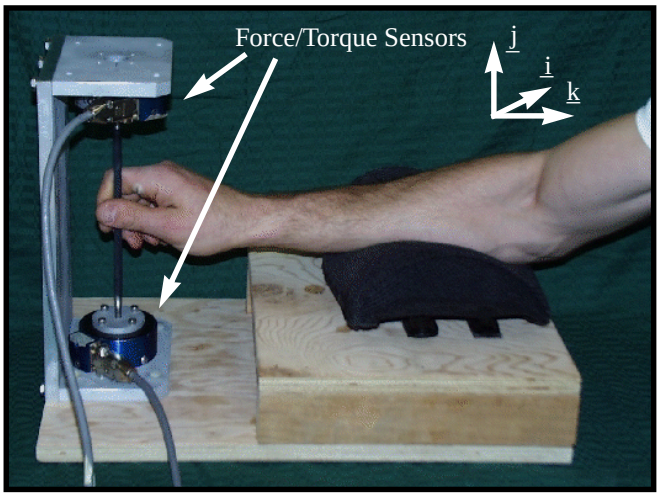


Figure 6.3: Force/Torque Test Apparatus

Subjects grasped the rod using a pencil grip and were instructed to push, pull and twist the rod in all six directions using a consistent amount of effort. They alternately applied static (push and hold) force/torque in both the positive and negative directions and dynamic (back and forth) force/torque along each of the 6 axes. A typical data set obtained from the force/torque sensors is shown in Figure 6.4 where *F* and *M* are forces and moments about axes $\underline{i}$, $\underline{j}$ and $\underline{k}$.

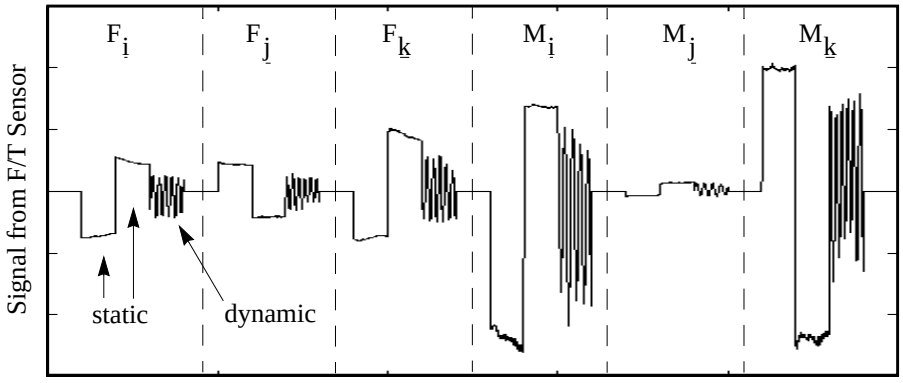


Figure 6.4: Typical Force/Torque Data

The task was performed twice by each subject, the first time with high exertion and the second time with medium exertion. Forces and torques were averaged by calculating the RMS values of a one second sampling period with forces measured in *N* and moments measured in *Ncm*. The highest and lowest 10% of

the RMS values were removed to account for experimental error and the remaining 80% were averaged. The mean and standard deviation for both the static and dynamic trials are shown in Figure 6.5.

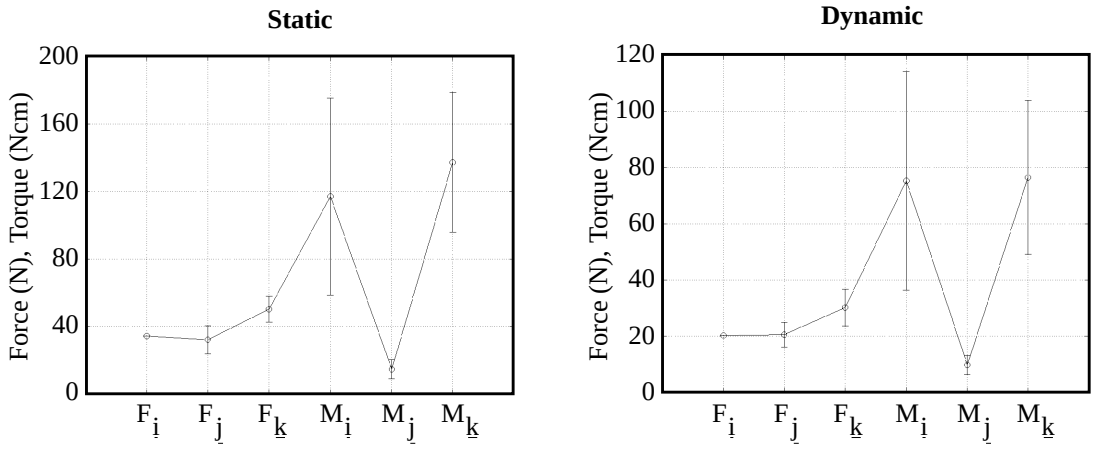


Figure 6.5: Force/Torque Test Values for High Exertion Trials

Since the relative results (normalized with respect to force about the $\underline{i}$ axis, $F_{\underline{i}}$) of the static and dynamic tests differ by less than 10% and have reasonable standard deviations, the data is assumed to be reliable. Averaging the static and dynamic values produces the relative values shown in Table 6.1. Note that the forces and torque that were measured during the medium exertion trials were approximately 50% smaller than those measured during the high exertion trials.

Table 6.1: Relative Force/Torque Capabilities of the Human Hand

	$\underline{i}$ axis	$\underline{j}$ axis	$\underline{k}$ axis
Force / $F_{\underline{i}}$	1.0	1.0	1.5
Torque / $F_{\underline{i}}$	3.6 cm	0.5 cm	3.9 cm

To obtain tangible position/velocity sensing requirements, the maximum velocity capabilities of the human hand were also identified. This involved a biomechanical experiment that measured the maximum achievable frequencies of periodic linear and angular motions of different amplitudes. Each experiment involved 10 participants (5 males, 5 females) from a distribution of ages, weights and professions similar to in the force/torque experiment. The test apparatus for the translational experiment was a pencil and paper with six rectangles drawn on it. The rectangles were *6cm* high and ranged in width from *0.15cm* to *16cm*. Subjects were asked to shade in the rectangles as fast as possible while their hand trajectory was recorded by a position tracking device (Bird^{TM} magnetic sensor). The test apparatus for the rotational experiment was a low friction potentiometer connected to a key shaped handle through a universal joint. The trajectory was recorded and displayed in real time on a CRT with visual upper and lower amplitude

bounds. Subjects were asked to oscillate between the bounds as quickly as possible and were free to orient the test apparatus to maximize comfort and speed. Photographs of the two test apparatus are shown in Figure 6.6 and Figure 6.7. Note that although the rotational test apparatus has a flat handle, it is held with a grip that is very similar to a pencil grip.

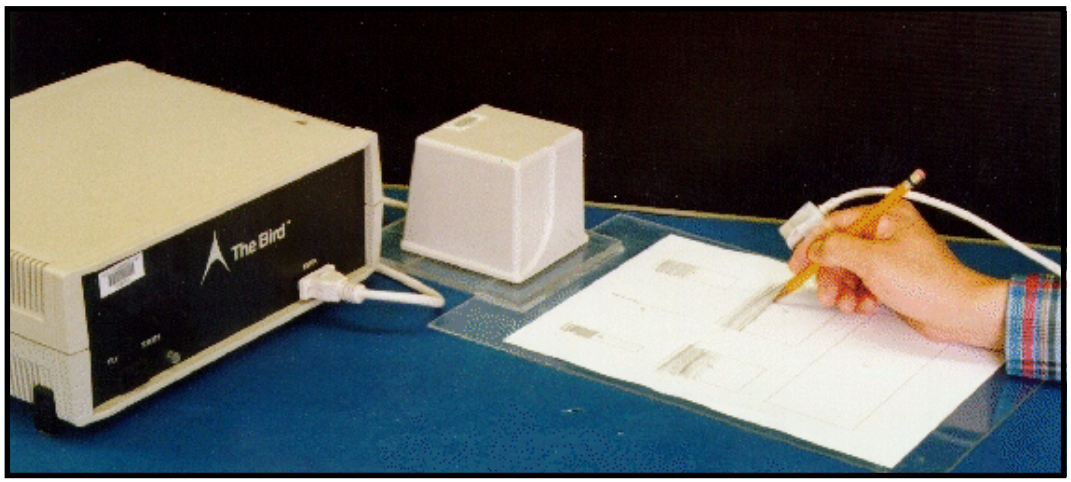


Figure 6.6: Translational Velocity Test Apparatus

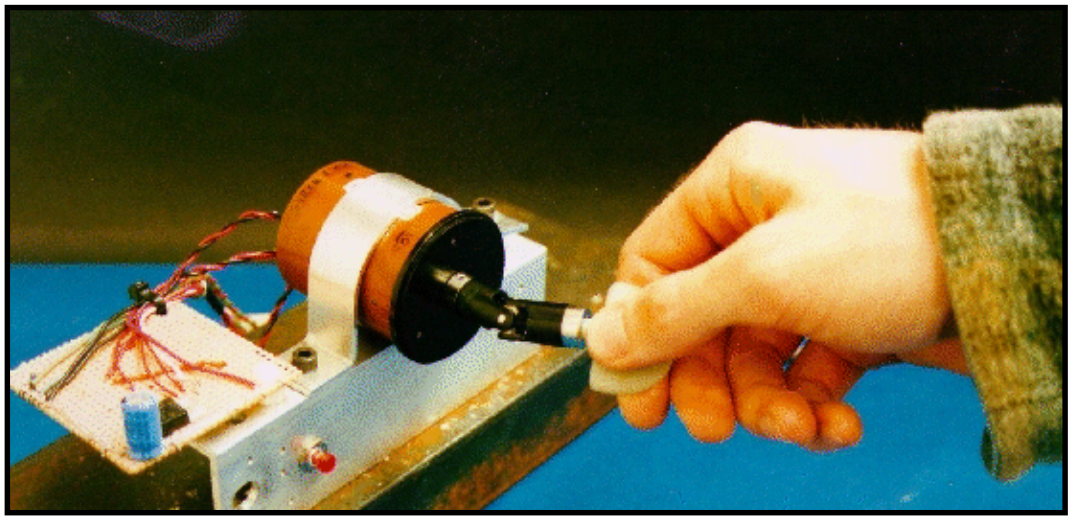


Figure 6.7: Rotational Velocity Test Apparatus

A typical data set obtained from one trial of the rotation experiment is shown in Figure 6.8. Translational data has a similar appearance and is not shown.

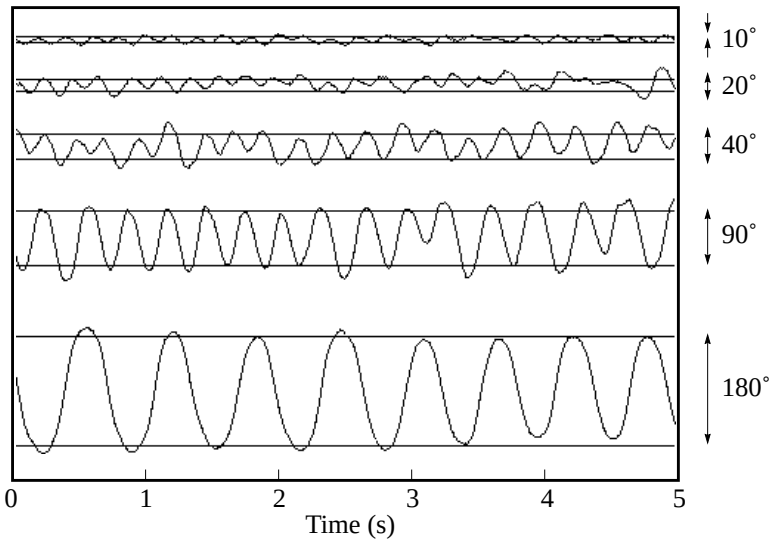


Figure 6.8: Typical Angular Velocity Data From One Trial

The trials were performed three times by each subject. The trajectories were analyzed to obtain the dominant frequency component and the results of all trials were averaged. Maximum velocity was computed by assuming a sinusoidal trajectory as shown in equations (6.7) through (6.9) and is plotted in Figure 6.9.

$$x = A\sin(\omega t) \tag{6.7}$$

$$\dot{x} = A\omega\cos(\omega t) \tag{6.8}$$

$$\|\dot{x}\| = A\omega \tag{6.9}$$

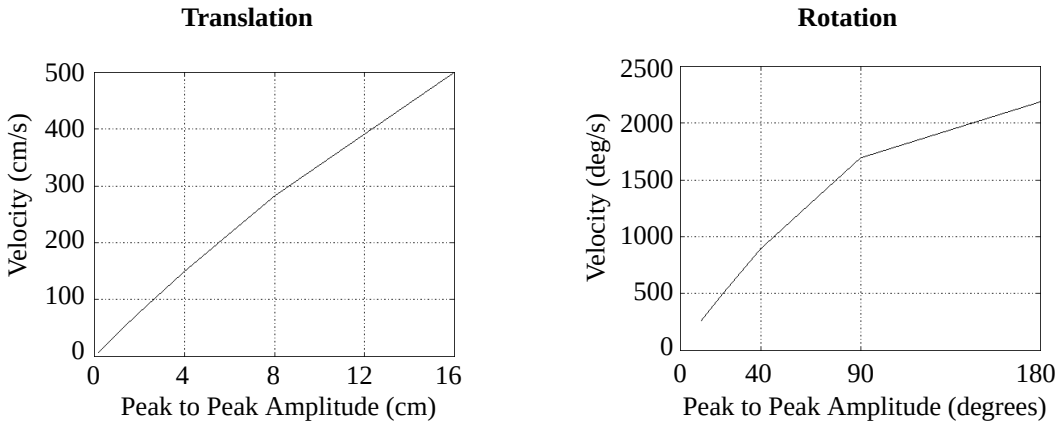


Figure 6.9: Velocity Test Results

Figure 6.9 provides the maximum speed that the human hand can achieve inside a workspace of up to *16cm* in translation and *180*˚ in rotation.

6.4 Concluding Remarks

In this chapter, it is reasoned that an impedance type haptic interface will perform best if it has low mechanical impedance and isotropic static force. It is proposed that impedance can be made sufficiently low by selecting appropriate components so that the kinematic design can be focussed on static force isotropy. It is also proposed that sensor resolutions can be chosen afterwards to obtain isotropic velocity sensing capabilities for wider stability margins. Biomechanical studies measuring the maximum force and velocity capabilities of the human hand are conducted to provide target values for a practical design example.

In the following chapter, this data is used in conjunction with the design procedure described in Chapter 3 through Chapter 5 to compare three candidate manipulators for use as a haptic pen. The preferred device is a novel hybrid manipulator called the Twin-Pantograph.

Chapter 7

Design Optimization of a Haptic Pen

7.1 Introduction

In Chapter 6 a design specification has been proposed for a haptic interface which includes low mechanical impedance and isotropic static forces. The low impedance requirement rules out the use of a serial manipulator but there are a number of parallel and hybrid devices that may be suitable. This chapter attempts to determine which one is the most suitable by comparing three candidates in terms of workspace size and static force isotropy using the design procedure described in Chapter 3 through Chapter 5. One of the candidates is a novel 6-DOF hybrid manipulator call the Twin-Pantograph that is introduced here. The Twin-Pantograph has many of the advantages of a parallel manipulator such as high stiffness and low mass but it also has a larger workspace and more isotropic static force capabilities. Therefore, it is selected as the basis for a prototype haptic pen. Since a sufficiently light actuator is currently unavailable for actuating roll about the pen axis, that degree-of-freedom is made passive and the prototype is re-designed for 5-DOF operation.

This chapter is organized as follows. In Section 7.2 the Twin-Pantograph is introduced and its performance is compared to two common parallel manipulators. In Section 7.3 the Twin-Pantographs is implemented as a haptic pen with passive roll about the pen axis and a redundant base actuator. In Section 7.4 some concluding remarks are made.

7.2 Device Comparison

The GII, scaling matrices, culling algorithm and workspace discretization technique are applied to the design of an impedance type haptic interface. Only robots with low effective mass (i.e. non-serial robots) are considered to ensure low mechanical impedance and design parameters are chosen to maximize static force isotropy. Three robots are considered including the Stewart Platform (see Figure 5.9), the Inoue Platform (see Figure 7.1) and a novel hybrid robot called the Twin-Pantograph (see Figure 7.2).

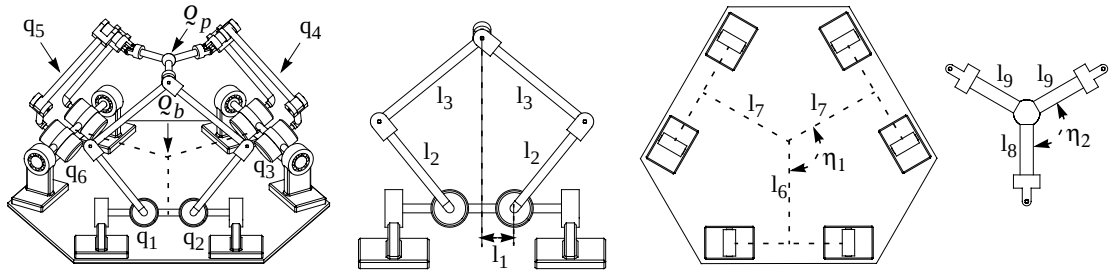


Figure 7.1: 6-DOF Inoue Platform

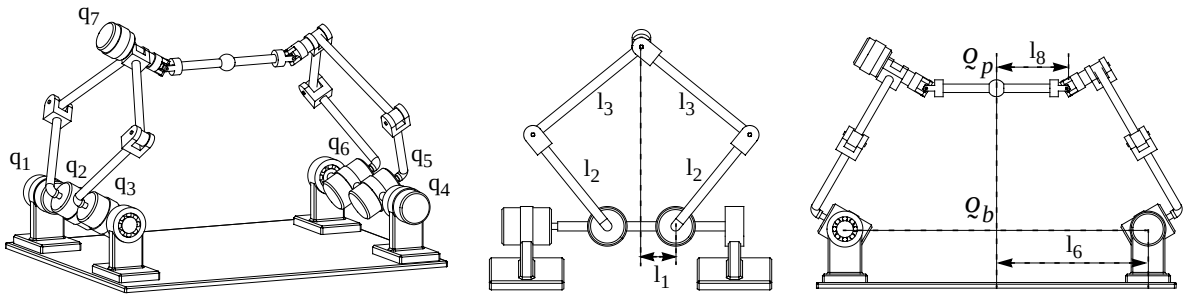


Figure 7.2: 6-DOF Twin-Pantograph

The Twin-Pantograph uses two 3-DOF 5-bar linkages that are actuated about their folding or waist joints (q_1, q_4) to provide five degrees of freedom (3 translation, 2 rotation) to a cylindrical end-effector. The sixth degree of freedom (roll) is provided by a series actuator (q_7) mounted to one of the 5-bar linkages. While many permutations of this device exist (e.g. the pantographs could be replaced by prismatic actuators, q_7 could be replaced by a lead-screw mechanism as in [30], etc.), only the version shown in Figure 7.2 is considered here.

The Twin-Pantograph is a 6-DOF robot with seven actuators. One waist actuator (q_1 or q_4) is redundant but eliminates a singularity that would occur when the platform axis lies in the plane of the five-bar linkage with a passive waist[1]. It can be shown that the only remaining singular configurations occur when (i) a pantograph is in a singular configuration (Figure 7.3a-c), (ii) the tip of a pantograph intersects its own waist axis (Figure 7.3c-d), (iii) a pantograph is at its workspace limit or (iv) the tips of both pantographs align with a forearm of a pantograph (i.e. a spherical wrist singularity). The pantograph singularities are, however, not particularly problematic since, by design, the singularity in Figure 7.3a is physically impossible if $l_1 > l_2$ as are the singularities in Figure 7.3b-d if $l_3 > (l_1 + l_2)$.

1. The same redundant actuation and sensing approach can be used with the Inoue platform.

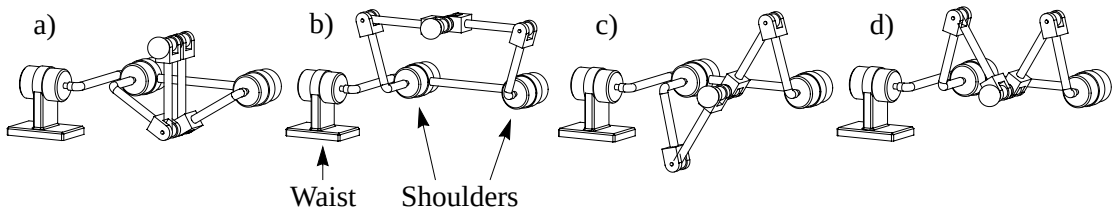


Figure 7.3: Singular Positions of a 3-DOF Five-Bar Linkage

Although the Twin-Pantograph uses a series actuator to roll its end-effector, the torque requirements of this actuator are small (see Table 6.1) so its inertial contribution is not expected to exceed that of the two additional base to platform links of the parallel platforms.

First, the semi-dextrous workspaces of the three devices are compared. A point belongs to such a semi-dextrous workspace if the platform origin o_p can be placed there and rolled, pitched and yawed ±30˚. For this comparison, the devices are given similar footprints and favourable geometries (i.e. similar forearm and upper arm lengths, singularity avoidance as described above and platforms half as large as the bases as suggested in [42]). The lengths *q* of the prismatic actuators are allowed to range between $9 \le q \le 18$ and 5-bar linkage geometries are chosen to provide a similar reach. The resulting semi-dextrous workspaces are shown in Figure 7.4a-c for the Stewart Platform, Inoue Platform and Twin-Pantograph respectively.

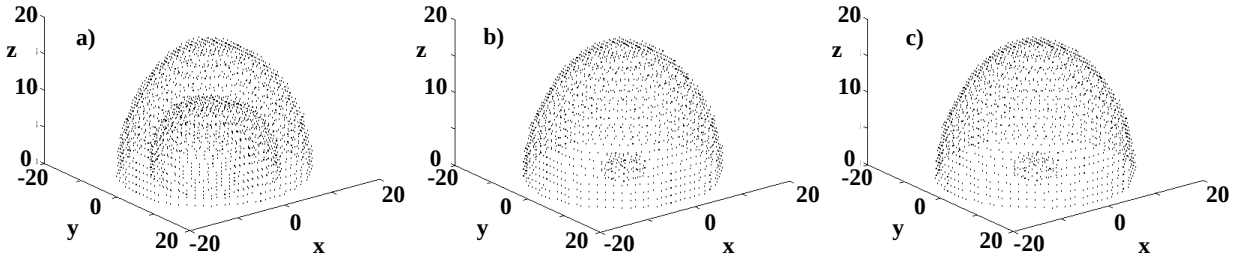


Figure 7.4: Semi-Dextrous Workspaces of 6-DOF Manipulators

The workspaces are all similar in size except that the Stewart Platform's has a large void in its centre due to the retraction limits of its prismatic actuators so its volume is much less than the others. If the U-joints between the 5-bar linkages and platforms of the Inoue Platform and Twin-Pantograph are constrained to ±85˚ to avoid a spherical wrist singularity, their workspaces are reduced to those shown in Figure 7.5a-b respectively. The Twin-Pantograph's workspace is clearly larger.

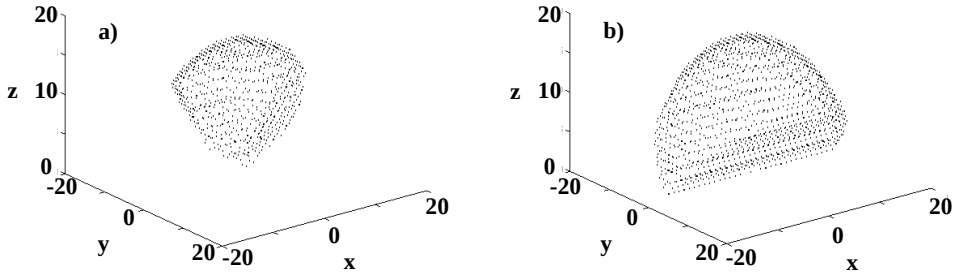


Figure 7.5: Semi-Dextrous Workspaces with U-Joint Bend Angle Constraint

Next, static force isotropy is compared by kinematically optimizing $J^T(x)$ for each device. For this comparison, the Twin-Pantograph's redundant (q_4) actuator is replaced by a passive joint to eliminate any unfair advantage that may result from redundant actuation so all three manipulators have square (i.e. 6×6) Jacobian matrices.

From Table 2.1, a reasonable workspace size for a haptic device is *16cm*(i-axis) × *10cm*(j,k-axes) × ±*45*˚. Since the parallel robots would have difficulty reaching a ±*45*˚ rotational workspace, it is reduced to ±*30*˚ for this comparison. The workspace centre is placed *25cm* (measured along the *k* axis) from the base origin o_b. This avoids a trivial result since isotropy always improves when the workspace is moved further from the base resulting in infinitely large robots. The 6-DOF workspaces are discretized as shown in Table 7.1. Since the parallel devices are symmetric about the *jk* plane, half of the cartesian workspace is redundant and will not affect the outcome. To eliminate unnecessary computations, the minimum position on the *i* axis is set to *0cm* so that the total number of discrete samples is *2,653,020*. The Twin-Pantograph, on the other hand, is not symmetric because of its wrist actuator q_7 which operates relative to a pantograph forearm so its entire workspace is considered.

Table 7.1: Discrete Workspaces of the 6-DOF Manipulators

Position				
Axis	Min	Max	Step	Total
i	-8 cm	8 cm	1.0 cm	17
j	-5 cm	5 cm	1.0 cm	11
k	-5 cm	5 cm	1.0 cm	11

Orientation				
Axis	Min	Max	Step	Total
i,k	-30˚	30˚	variable	168
j	-30˚	30˚	5˚	13
Total discrete samples			4,492,488	

The number of angular samples is computed from equation (B.3) using $k_i=17$, $k_j=k_k=11$ and $d_t=3$. For rotations about the *i* and *k* axes, $k_r=162$ for $d_r=2$ is rounded up to $K_r=168$ for $N=7$ from equation (B.4). For rotations about the *j* axis, $k_r=13$ for $d_r=1$.

Although additional geometric parameters to those shown in Figure 5.9, Figure 7.1 and Figure 7.2 could be defined, some symmetries are introduced to keep the number of parameters manageable and all devices are allotted two actuator scale factors, Q_1 and Q_2. For the Stewart and Inoue Platforms, Q_1 is applied to q_3 and q_6 and Q_2 is applied to q_4 and q_5 since the devices are symmetric about the *jk* plane but not about the *ik* plane. For the Twin-Pantograph, Q_1 is applied to q_1 and Q_2 is applied to q_7 since the waist (q_1), wrist (q_7) and shoulder (q_2, q_3, q_5, q_6) joints have the most dissimilar torque demands. The resulting joint-space scaling matrices S_J are shown in (7.1) for the Twin-Pantograph and in (7.2) for the parallel platforms.

$$S_J = Diag([Q_1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; Q_2]) \tag{7.1}$$

$$S_J = Diag([1 \;\; 1 \;\; Q_1 \;\; Q_2 \;\; Q_2 \;\; Q_1]) \tag{7.2}$$

Since a haptic interface must counteract hand forces, the maximum force/torque capabilities reported in Table 6.1 are used in the S_T matrix (7.3). Since these forces are relative to the subject's hand, the Jacobian is expressed in the end-effector coordinate frame prior to scaling.

$$S_T = Diag([1 \;\; 1 \;\; 1.5 \;\; 3.6 \;\; 0.5 \;\; 3.9]) \tag{7.3}$$

Due to practical limits (processing speed of a SGI O^2 running IRIX) the design parameter space is limited to 50 million elements. It is, therefore, possible to search a broader range or use finer sample spacing when there are fewer design parameters. Since the Twin-Pantograph has 7 free design parameters (l_1 - l_4, l_6, Q_1, Q_2), the Stewart Platform has 9 (l_1, l_4 - l_7, η_1, η_2, Q_1, Q_2) and the Inoue Platform has 11 (l_1 - l_7, η_1, η_2, Q_1, Q_2), search ranges and sample spacings are assigned accordingly. The parameter spaces and optimal solutions are shown in Table 7.2 through Table 7.4.

Table 7.2: Design Parameter Search Space for the Twin-Pantograph

Param	Min	Max	Step	Total	Optimum
l_1	0.5	5	0.5	10	3.0
l_2	10.0	25.0	1.0	16	16.0
l_3	15.0	30.0	1.0	16	22.0
l_4	1.0	10.0	1.0	10	2.0
l_6	0.5	6.0	0.5	12	3.0
Q_1	0.6	2.0	0.1	15	1.9
Q_2	0.005	0.05	0.005	10	0.03
Total discrete samples				46,080,000	
GII of optimum solution				0.327	

Table 7.3: Design Parameter Search Space for the Stewart Platform

Param	Min	Max	Step	Total	Optimum
l_1	4.0	16.0	2.0	7	10.0
l_4	2.0	14.0	2.0	7	6.0
l_5	2.0	14.0	2.0	7	6.0
l_6	1.0	7.0	1.0	7	3.0
l_7	1.0	7.0	1.0	7	4.0
η_1	110˚	140˚	5˚	7	120˚
η_2	140˚	175˚	5˚	8	170˚
Q_1	0.3	1.5	0.2	7	0.5
Q_2	0.3	1.5	0.2	7	0.9
Total discrete samples				46,118,408	
GII of optimum solution				0.096	

Table 7.4: Design Parameter Search Space for the Inoue Platform

Param	Min	Max	Step	Total	Optimum
l_1	1.0	5.0	1.0	5	2.0
l_2	12.0	22.0	2.0	6	16.0
l_3	20.0	30.0	2.0	6	24.0
l_4	2.0	8.0	2.0	4	2.0
l_5	2.0	8.0	2.0	4	2.0
l_6	1.0	5.0	1.0	5	2.0
l_7	1.0	5.0	1.0	5	4.0
η_1	100˚	140˚	10˚	5	120˚
η_2	130˚	170˚	10˚	5	170˚
Q_1	0.5	1.5	0.25	5	0.75
Q_2	0.5	1.5	0.25	5	0.75
Total discrete samples				45,000,000	
GII of optimum solution				0.215	

The optimum GIIs of the Twin-Pantograph, Stewart Platform and Inoue Platform are *0.327*, *0.096* and *0.215* respectively. The static force GII of the Twin-Pantograph is *3.4* and *1.5* times larger than the GIIs of the parallel platforms. The Twin-Pantograph is, therefore, considerably more isotropic than the parallel platforms in terms of the static force requirements of a haptic interface.

7.3 Design of a Haptic Pen

Since the Twin-Pantograph has a larger semi-dextrous workspace and can produce more isotropic static forces than the parallel platforms, it is the most viable candidate. As predicted, the serial wrist actuator should be very small (i.e. *1/63* the torque of the waist actuator) according to Table 7.2. A light-weight, actuator is under development for this purpose but in the meantime, this joint is made passive. This slightly simplified device (5 active DOF, 1 passive DOF) lends itself well to applications which do not rely on reaction torques from axial rotations such as a surgical probe or an excavator bucket.

To fully exploit the device's motion range, the rotational workspace is extended to ±45˚ and to further enhance its static force isotropy, the redundant waist actuator (q_4) is reintroduced. With only 5 active degrees-of-freedom, the Twin-Pantograph is symmetric about both the *jk* plane and the *ik* plane so only one quarter of its translational workspace needs to be considered to rigorously evaluate its performance. This allows the workspace to be discretized more finely as shown in Table 7.5.

Table 7.5: Discrete Workspace of the 5-DOF Twin-Pantograph

Position				
Axis	Min.	Max.	Step	Total
i	0 cm	8 cm	0.5 cm	17
j	0 cm	5 cm	0.5 cm	11
k	-5 cm	5 cm	0.5 cm	21

Orientation				
Axis	Min.	Max.	Step	Total
i,k	-45˚	45˚	n/a	617
Total discrete samples			2,422,959	

Since a singularity is eliminated by introducing the redundant waist actuator, isotropy always improves with increased base length (l_4) since this increases the distance between the workspace and the actuators. Therefore, l_4 is fixed at *8.0 cm* to keep the height of the device reasonable while providing adequate clearance for the shoulder actuators. The resulting device has 5 free design parameters which are shown with the optimal solution in Table 7.6. Note that l_1 is assigned a minimum value of *2.4 cm* to account for the width of the shoulder actuators and l_6 is assigned a minimum length of *7.0 cm* so that the end-effector can be held comfortably by a human hand.

Table 7.6: Design Parameter Search Space for the 5-DOF Twin-Pantograph

Dim.	Min.	Max.	Step	Total	Optimum
l_1	2.4	7.0	0.2	24	2.4
l_2	10.0	30.0	0.5	41	17.0
l_3	15.0	35.0	0.5	41	23.0
l_6	7.0	11.0	0.2	21	7.0
Q_1	0.5	3.0	0.1	26	1.6
Total discrete samples				22,027,824	
GII of optimum solution				0.234	

The workspace position (i.e. *25cm* away from the base) was chosen somewhat arbitrarily. Increasing this distance is known to improve isotropy but also increases the mass and footprint of the device. To see how performance is affected by workspace position, the device is optimized for different workspace distances ranging between *15cm* and *40cm*. The results are shown in Table 7.7.

Table 7.7: Optimum Geometries and GIIs with Different Workspace Distances

Dist.	l_1	l_2	l_3	l_6	Q_1	GII
15 cm	2.4	13.0	16.0	7.0	1.6	0.089
20 cm	2.4	14.5	20.0	7.0	1.7	0.182
25 cm	2.4	17.0	23.0	7.0	1.6	0.234
30 cm	3.6	19.5	27.0	7.0	1.6	0.262
35 cm	3.6	22.5	30.5	7.0	1.6	0.280
40 cm	4.0	25.5	34.5	7.0	1.6	0.292

The position/velocity sensing isotropy is considered next by selecting optimum sensor resolutions for the designs in Table 7.7. To do this, it is assumed that to maintain a given velocity signal quality, the required position sensor resolution is inversely proportional to the maximum joint velocity. Therefore, the S_J matrix in (7.4) is used to find the optimal position sensor resolution ratio between the shoulder and waist joints (R_w/R_s) and the S_T matrix is assigned to the maximum velocity capabilities of the human hand as in (7.5). From Figure 6.9, a human hand can produce a maximum linear velocity of *500cm/s* inside a *16cm* workspace and maximum angular velocity of *30rad/s* inside a $\pm 45^{\circ}$workspace. Since the subjects in the velocity experiments were allowed to choose the direction that maximized their performance, this data is applied equally to all directions (7.6), (7.7) producing the S_T matrix shown in (7.8).

$$S_J = Diag\left(\left[1 \;\; \frac{R_w}{R_s} \;\; \frac{R_w}{R_s} \;\; 1 \;\; \frac{R_w}{R_s} \;\; \frac{R_w}{R_s}\right]\right) \tag{7.4}$$

$$S_T = Diag\left(\left[1 \;\; \frac{v_j}{v_i} \;\; \frac{v_k}{v_i} \;\; \frac{\omega_i}{v_i} \;\; \frac{\omega_k}{v_i}\right]\right) \tag{7.5}$$

$$v_i = v_j = v_k = 500 \;\; cm/s \tag{7.6}$$

$$\omega_i = \omega_k = 30 \;\; rad/s \tag{7.7}$$

$$S_T = Diag([1 \;\; 1 \;\; 1 \;\; 0.06 \;\; 0.06]) \tag{7.8}$$

Scaling the Jacobian *J(x)* in (7.9) by S_J in (7.4) and S_T in (7.8) produces the normalized, scaled Jacobian $\hat{J}(x)$ in (7.10).

$$\dot{q} = J(x)\dot{x} \tag{7.9}$$

$$\hat{J}(x) = S_J^{-1} J(x) S_T \tag{7.10}$$

Using $\hat{J}(x)$ (7.10), the optimal encoder resolutions and resultant velocity GIIs for the designs in Table 7.7 are shown in Table 7.8 and the GIIs for both force and velocity are plotted in Figure 7.6.

Table 7.8: Optimum Position Sensor Resolution Ratios

	15 cm	20 cm	25 cm	30 cm	35 cm	40 cm
R_W/R_S	1.63	1.67	1.63	1.64	1.62	1.59
GII	0.105	0.186	0.232	0.269	0.285	0.303

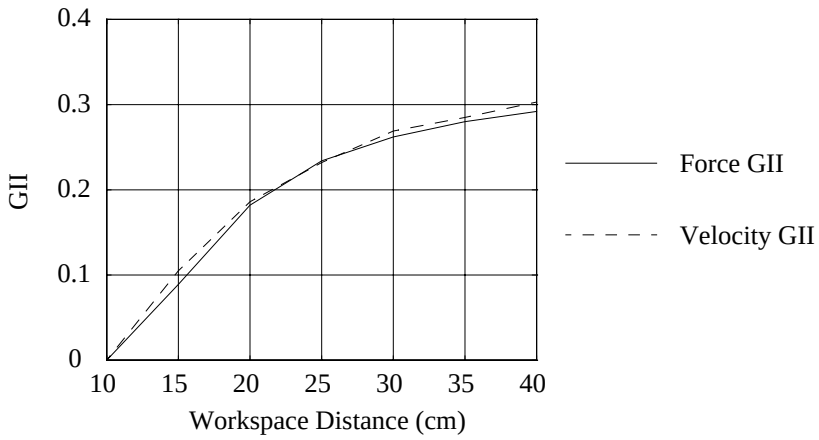


Figure 7.6: Plot of GII vs. Workspace Distance

Figure 7.6 shows a declining improvement in both the force and velocity GII when the workspace is moved further from the base. Performance is substantially degraded when the workspace is placed less than *20cm* from the base and improvements become marginal when it is placed more than *25cm* from the base. A design based upon a workspace placement between *20-25cm*, therefore appears to optimally trade-off performance with compactness. Note that regardless of workspace position, the waist actuators should have approximately *1.6* times the torque and *1.6* times the sensor resolution of the shoulder actuators.

7.4 Concluding Remarks

A new 6-DOF hybrid manipulator called the Twin-Pantograph has been introduced. It is stiff and light like a parallel manipulator but has a much larger workspace, particularly in rotation. All singularities can be eliminated from its reachable workspace by introducing a redundant base actuator and respecting some geometric constraints.

The design procedure described in Chapter 3 through Chapter 5 has been used to show that the Twin-Pantograph also has more isotropic static force capabilities than a Stewart or Inoue platform and is,

therefore, chosen as the basis for the prototype haptic pen. The prototype is designed with passive roll about the pen axis since a suitable actuator for this particular degree-of-freedom is still under development.

In the next chapter, the 5-DOF Twin-Pantograph haptic pen is evaluated and controlled to mimic three different virtual environments.

Chapter 8

Description, Evaluation and Control of a Haptic Pen

8.1 Introduction

A 5-DOF Twin-Pantograph haptic was designed in Chapter 7. In this chapter, all of the physical aspects of the prototype system are described. This includes the materials used to build it, the computer hardware and electronics used to control it, and the graphics system used to display the virtual scene. It also presents a table of measured performance benchmarks that can be used to compare the device to existing devices such as the SensAble PHANToM [48], describes the control algorithm that was used to implement the virtual impedances and shows how the system has been used to emulate three virtual environments including a virtual pencil, a virtual scalpel and a virtual excavator.

This chapter is organized as follows. Section 8.2 describes the hardware which includes the materials and components used to construct the haptic pen and the electronic and computer hardware used to control it. In Section 8.3 performance benchmarks of the haptic pen are measured. In Section 8.4 the control algorithm is described. In Section 8.5 the simulated virtual environments are described. In Section 8.6 a hard contact environment is used to evaluate the stability of the system. In Section 8.7 some concluding remarks are made.

8.2 Hardware

The haptic pen is oriented so that its *j* axis is vertical. This enhances both ergonomics and versatility since humans are accustomed to holding pens and pen-like instruments in a near vertical fashion. A photograph and a schematic of the haptic pen are shown in Figure 8.1.

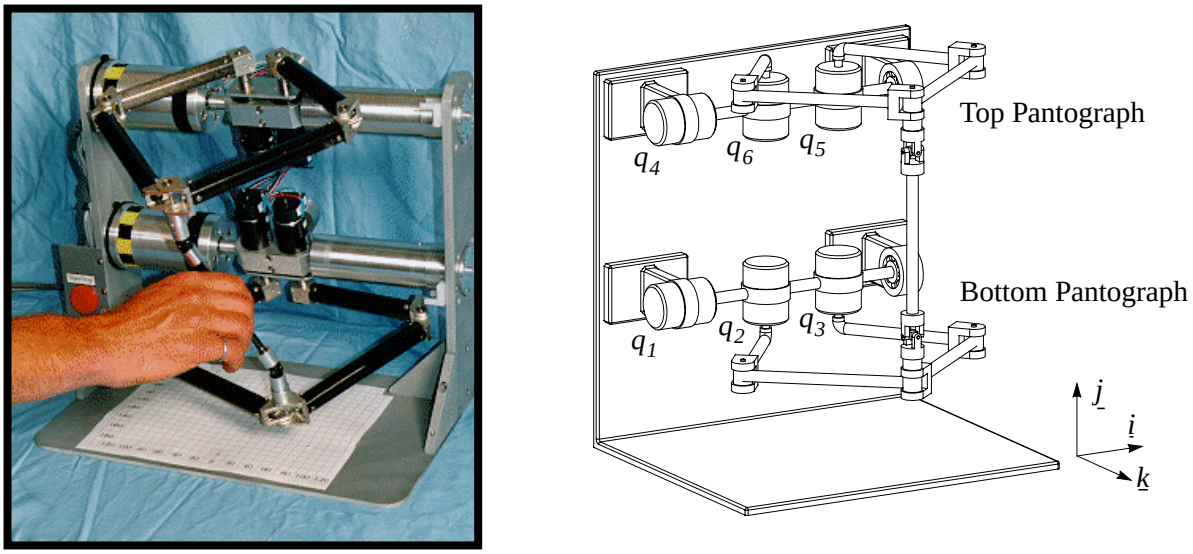


Figure 8.1: The Twin-Pantograph Haptic Pen

In accordance with Chapter 6, the device is constructed to be as light as possible. The 5-bar linkage arms are cut from *0.82"* diameter carbon fibre tubing and the end-effector is cut from *0.28"* diameter carbon fibre tubing. The clevises are machined from magnesium and all other components (i.e. frame, shoulder motor clamps) are machined from aluminum. Stainless steel counterbalances, mounted behind the shoulder motors, are adjusted to oppose gravity as much as possible without causing the device to "fall up" at any position inside its workspace.

The device is made as friction and backlash free as possible by using direct drive motors (i.e. no reduction mechanisms) and roller bearings in nearly all passive joints including the elbow joints and the axial joints that interface the universal joints to the 5-bar linkages and end-effector. Each universal joint is equipped with a passive axial joint on each side to triple its maximum useable bend angle from *30*˚ to *90*˚. The universal joints themselves account for the only plain bearings on the robot. They are made from Delrin to provide negligible friction and backlash.

Two different actuators are used on the robot. Maxon 90W rare earth magnet motors are used at the shoulder joints and Maxon 80W ferrite magnet motors are used at the waist joints. The 80W motors have approximately twice the stall torque as the 90W motors and approximately *20* times the rotor inertia. Since two shoulder motors ride on each waist motor, the large rotor inertia of the waist motors has little impact on system performance but their higher torque satisfies the design specification of Chapter 7.

Position sensing is handled using *1000* window (i.e. *4000 counts/rev*) US Digital optical encoders at each active joint. Although the design optimization suggests a need for higher resolution position sensors at the waist joints, this was not economically viable for the prototype. Although both torque and sensing resolution could be scaled at the waist joints by adding reduction mechanisms, the associated increase in inertia, friction and backlash is expected to result in a net performance loss.

Encoder data is decoded by a custom built VME decoder board, read by an XVME 200 D/D converter and processed by a Themis SPARC 5/64 CPU. Analog control signals are output by an XVME 542 D/A converter into a *6* channel linear amplifier which is fed by a *1000W* (*40V, 25A*) power supply. A photograph of the Twin-Pantograph haptic pen, VME cage, linear amplifier and power supply is shown in Figure 8.2.

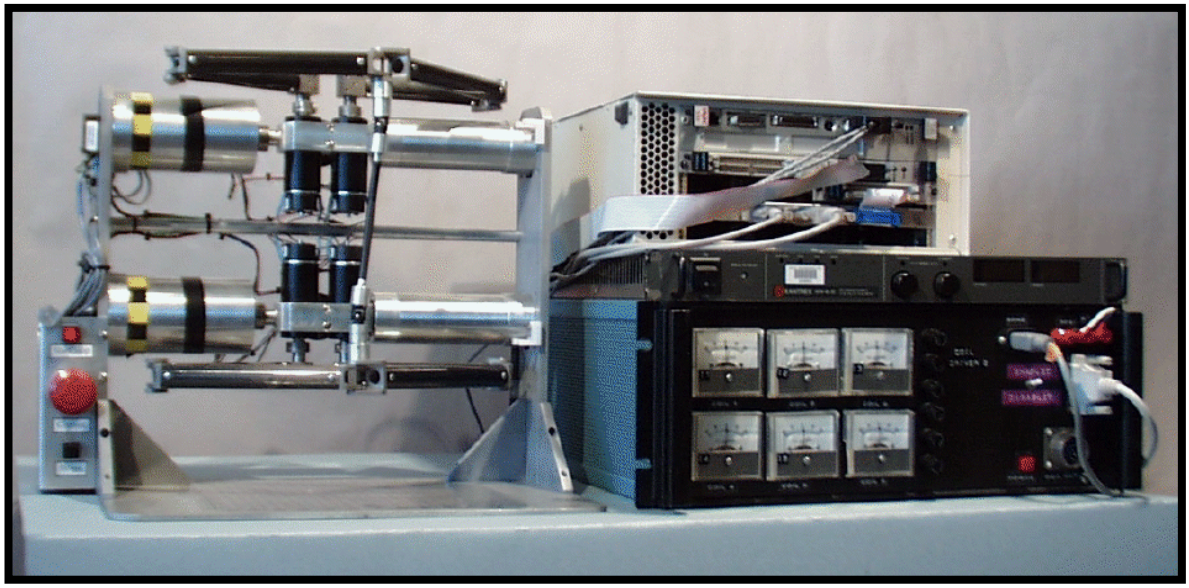


Figure 8.2: The Twin-Pantograph and Associated Hardware

8.2.1 Optical Decoder Board

An optical decoder board was built to provide an economical means of interpreting the encoder data. It uses HP HCTL 2016 optical decoder chips to accumulate the transitions of six incremental optical encoders. The data is latched so that the positions of all six encoders are acquired at the same instant and clocked out serially to reduce the number of required digital input channels. The serial data of all six encoders are read in parallel so a total of six digital input channels are required. The board is controlled by a 4-state state machine using two D flip-flops so only two digital output channels are required. One control signal advanced the state while the other shifts out the serial data. To calibrate the system, both outputs are

held high to reset the state machine and counters. A block diagram of the digital circuit is shown in Figure 8.3. A timing diagram for controlling the decoder board and reading in the encoder data is shown in Figure 8.4.

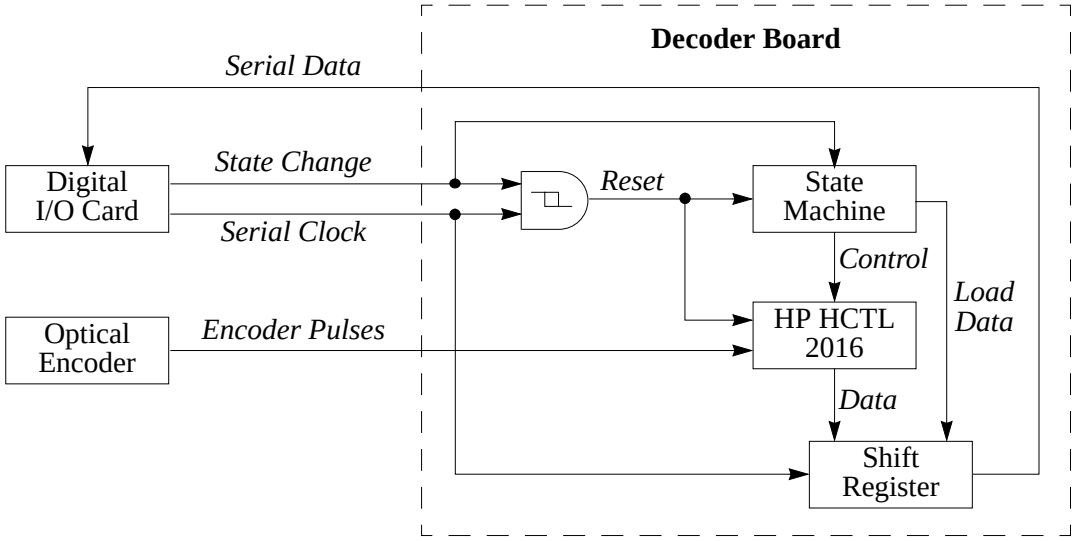


Figure 8.3: Optical Decoder Circuit Block Diagram

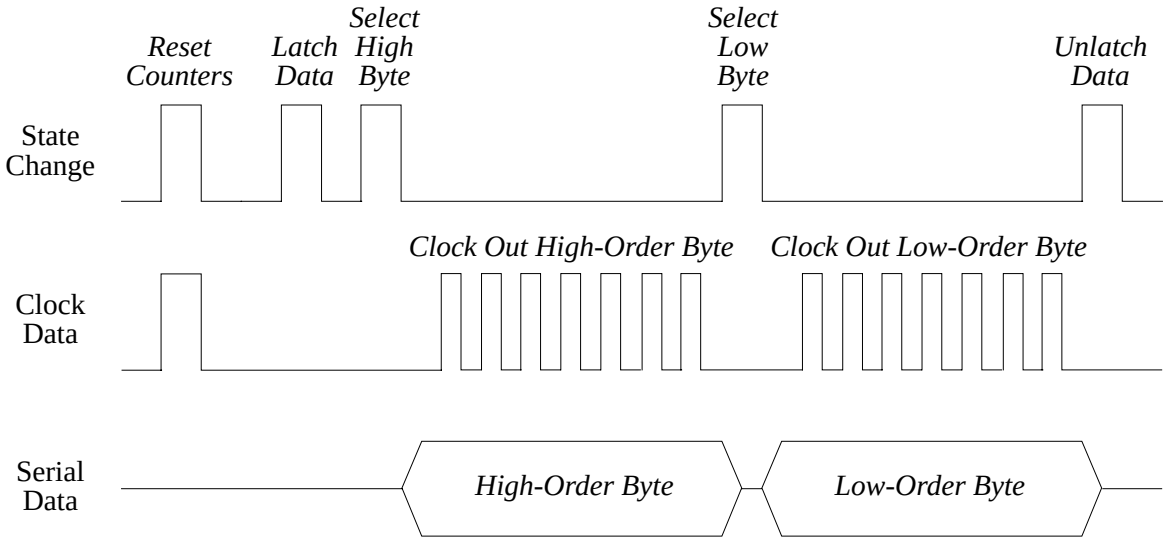


Figure 8.4: Timing Diagram for Encoder Data Acquisition

A photograph of the six channel optical decoder board is shown in Figure 8.5 which was designed to be mounted in a VME chassis.

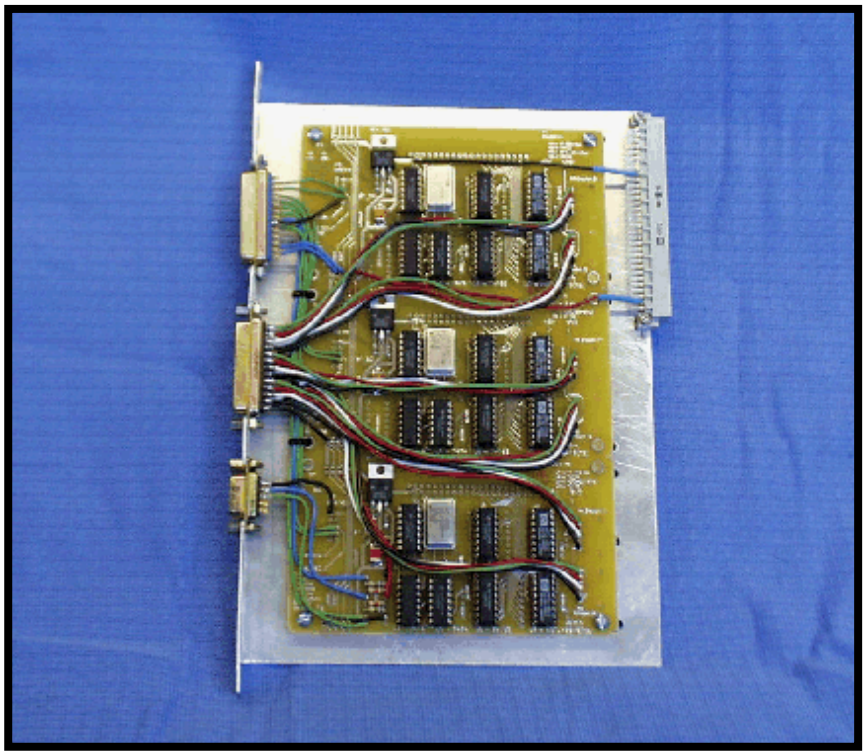


Figure 8.5: The 6-Channel Optical Decoder Board

8.3 Performance

A mass matrix (8.1) is computed numerically for the Twin-Pantograph haptic pen at its home position (i.e. x,y=0, z=20*cm*, zero rotation) using Pro/MECHANICA™. Its value is shown in (8.2).

$$\begin{bmatrix} f(N) \\ \tau(Ncm) \end{bmatrix} = D \begin{bmatrix} \ddot{x}(cm/s^2) \\ \dot{\omega}(rad/s^2) \end{bmatrix} \tag{8.1}$$

$$D = \begin{bmatrix} 190 & 0.197 & -0.179 & 0.333 & 1.52 \\ 0.197 & 226 & -0.035 & -346 & -0.515 \\ -0.179 & -0.035 & 156 & -0.642 & -0.335 \\ 0.333 & -346 & -0.642 & 10300 & 12 \\ 1.52 & -0.515 & -0.335 & 12 & 12600 \end{bmatrix} \times 10^{-5} \tag{8.2}$$

The robot specifications in Table 8.1 are either measured directly or derived from the mass and Jacobian matrices and the actuator and sensor specifications. All values correspond to the robot at its home position except for unbalanced weight which is a workspace inclusive range. The stiffness and damping coefficients are obtained using a tuned PD controller at a control frequency of 1*KHz* with the end-point velocity computed from low-pass filtered finite difference position readings using the filter described in [81]. The minimum force is the force required to overcome static friction and the dynamic range is the ratio between the minimum and peak force/torque capabilities.

Table 8.1: Haptic Pen Performance Specifications

	Translation Axis			Rotation Axis	
	x	y	z	x	z
Workspace	±6cm	±3.75cm	±3.75cm	±45˚	±45˚
Spatial Resolution	142µm	314µm	175µm	0.122˚	0.099˚
Minimum F/T	0.022N	0.045N	0.023N	0.19Ncm	0.18Ncm
Continuous F/T	5.0N	3.3N	4.1N	34Ncm	41Ncm
Peak F/T	48N	21N	40N	324Ncm	396Ncm
Dynamic Range	2200:1	470:1	1700:1	1700:1	2200:1
Effective Mass	190g	226g	156g	10300gcm^2	12600gcm^2
Peak Acceleration	25.8G	9.5G	26.1G	3147s^{-2}	3143s^{-2}
Maximum Stiffness	16N/cm	12N/cm	13N/cm	874Ncm	1076Ncm
Maximum Damping	0.44Ns/cm	0.5Ns/cm	0.36Ns/cm	24Nscm	30Nscm
Unbalanced Weight	70g - 130g				

8.4 Control

The device is controlled by a VxWorks real-time system. The control software contains two loops, a fast loop that updates at *1000Hz* and a slow loop that updates at *200Hz*. The fast loop reads the encoder values, computes the direct kinematics and velocity, and uses this information to update the state of the virtual environment. The updated virtual environment computes a desired equivalent impedance Z_e (see the discussion on impedance devices in Chapter 6) which is used to transform the position and velocity into a desired reaction force. This force is converted into joint torques using J^{-T} which are applied to the actuators via a linear amplifier. The slow loop recalculates J^{-T} using the most recent position and communicates the updated position to the graphic display. Although the Sparc 5 CPU is actually under-utilized at these control frequencies, increasing the clock rate actually degrades performance because the discretization effect on the filtered finite-difference velocity signal becomes magnified [51].

Since the haptic pen has a redundant waist actuator, J^{-T} is a pseudo-inverse which has an infinite number of solutions. The choice of J^{-T} determines how the load is distributed between the redundant waist actuators. Here, the load is distributed equally by converting the force and torque at the pen centre into an equivalent force at each end of the pen (i.e. each pantograph). Inverse Jacobian matrices are computed for both the top J_T^{-T} and bottom J_B^{-T} pantograph and are used to transform pantograph forces into actuator torques (see Figure 8.6). This approach has the additional advantage that it is much easier to compute a 3×3 inverse Jacobian for each pantograph than it is to compute one 6×5 pseudo-inverse Jacobian for the entire robot. The computational savings makes it possible to use a control frequency with a wider stability margin.

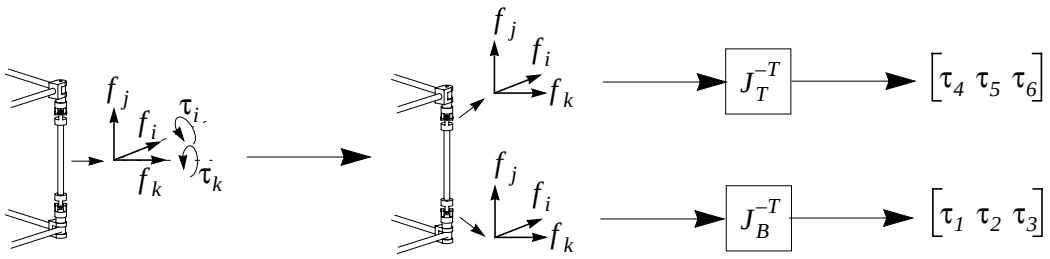


Figure 8.6: Decoupled Force Control of Pantographs

The control algorithm uses PD+B control where B is a braking pulse [64] that is applied upon contact with a stiff environment and is proportional to end point velocity. The braking pulse applies the necessary force to bring the end-effector to a halt in a single control cycle. Since vertical forces are primarily induced by the waist motors while horizontal forces are primarily induced by the shoulder motors, different controller gains are used along the vertical and horizontal axes to account for mechanical differences to make the device feel homogeneous.

A simple gravity compensation algorithm acts as an active counterbalance to make the device feel virtually weightless. It is comprised of forces along the *j* and *k* axes. Since effective weight of the device increases as it is stretched out (i.e. moved in the positive *k* direction), the force along the *j* axis is increased linearly to oppose it. Similarly, since the pen tends to fall in (i.e. in the negative *k* direction) when the pen is lifted and fall out (i.e. in the positive *k* direction) when the pen is lowered, force along *k* is also adjusted linearly with respect to *j*. To avoid erroneous torques on the pen, gravity compensation is computed individually for each pantograph while the gravity of the handle itself is neglected.

A block diagram of the control algorithm is presented in Figure 8.7 where f is the interaction force between the human and the robot, q is the joint angles, x is the position of the end-effector, $\dot{x}$ is the filtered finite-difference velocity of the end-effector, f_d is the desired reaction force, τ is the joint torques, V is the control voltages and I is the actuator currents.

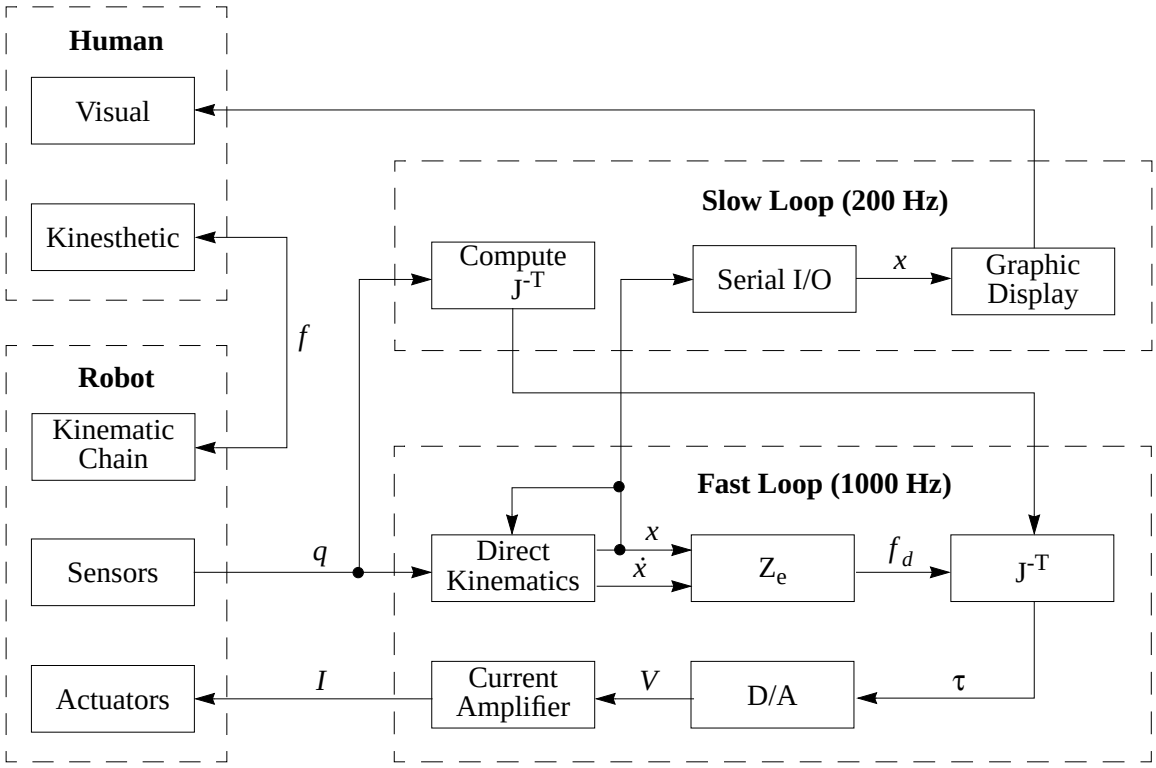


Figure 8.7: Block Diagram of Control Algorithm

The equivalent impedance block (Z_e) is expanded in Figure 8.8 where desired impedance is determined from the robot position x and the virtual environment being simulated, K_d is a damping scale factor, K_b is a braking scale factor, x_d is the desired position, x_e is the position error, $\dot{x}_e$ is the velocity error, $\dot{x}_b$ is the braking velocity, f_c is the contact force and f_g is the gravitational force.

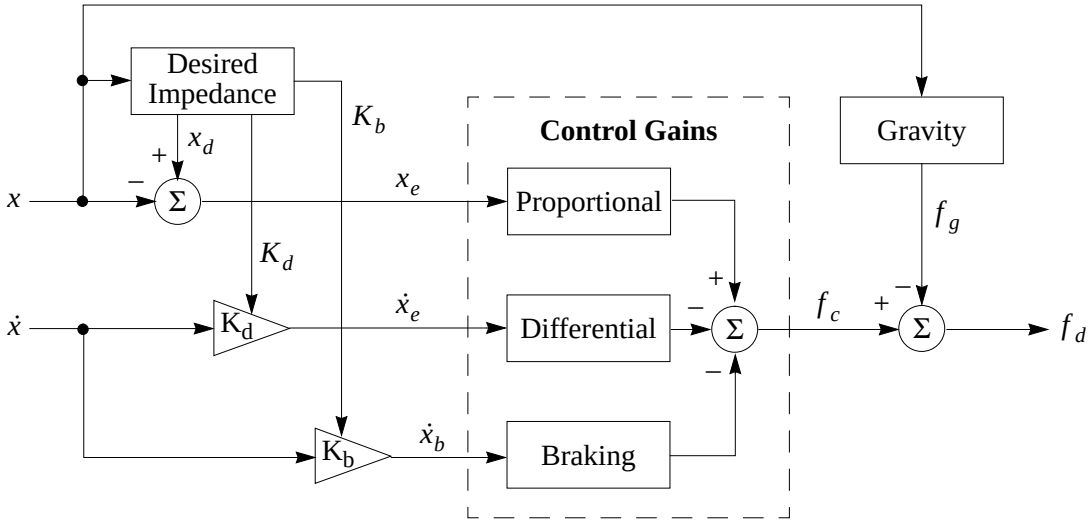


Figure 8.8: Details of Equivalent Impedance Block

8.5 Virtual Environments

The haptic pen is designed as a generic hand controller. Its is not designed for a specific application but rather, to address the capabilities of the operator. This makes it a suitable master for practically any virtual environment or tele-operation system. Its versatility is demonstrated by three diverse virtual reality simulations, a virtual pencil, a virtual scalpel and a virtual excavator. All virtual environments are implemented on an SGI O^2 graphics computer running IRIX.

8.5.1 Virtual Pencil

In the virtual pencil simulation, a wooden pencil is enclosed in a rigid bounding box. When the pencil lead contacts the box, hard contact is simulated by high PD gains and a braking pulse. A small damping force, proportional to the penetration depth, is also applied along the contact plane to simulate surface friction. The eraser end is treated similarly but there is no braking pulse, the PD gains are lower, and the planar friction force is higher to give the eraser a more rubbery feel. The two pantographs are controlled independently with the bottom pantograph simulating the pencil lead and the top pantograph simulating the eraser. This sacrifices some stiffness in exchange for reduced disturbance forces and a more realistic feel, particularly when one end of the pencil is held in contact while the other is swung freely about. Pencil marks are left and erased just as one would expect with a regular pencil and all marks are removed when a reset button is pressed. The reset button resides in the front left corner of the box. It is haptically rendered as a hard surface when contacted from the side and as a compliant linear spring when contacted from the top. A positive click, felt once the button has been fully depressed, is simulated by a discontinuous

decrease in the button's spring constant as described in [48]. A snapshot of the graphic environment[1] is shown in Figure 8.9.

8.5.2 Virtual Scalpel

The virtual scalpel simulation is much like the virtual pencil simulation except the pencil is replaced by a scalpel and a human brain protrudes from the floor of the bounding box. It demonstrates how such a device might be used as a surgical trainer. When the tip of the scalpel contacts the surface of the brain, it leaves cut marks which are erased by pressing the reset button which operates the same as in the virtual pencil simulation. Since a scalpel has no eraser, both ends experience hard contact with the bounding box. When the scalpel penetrates the brain tissue, a damping force is applied that is proportional to penetration depth. Damping is applied to both ends of the scalpel during cutting to impede both linear and angular motions. A snapshot of the graphic environment[1] is shown in Figure 8.9.

8.5.3 Virtual Excavator

In the virtual excavator simulation, the haptic pen manipulates the bucket of a virtual excavator. The centre of the pen is mapped to the teeth of the bucket and is free to move such that the kinematic constraints of the excavator are not violated. For example, rotations are not permitted about any axis in the plane of the boom. The motion range is also constrained to keep the haptic pen inside its workspace and to prevent anything out of the ordinary from happening in the simulation. For example, the excavator arm is kept from stretching beyond its reachable limit or from contracting so much that the boom crashes through the cabin. When the bucket is above ground level, damping is applied in all directions to slow down operator motion and present the sluggish, high impedance feel that one would expect when moving a heavy machine. When the bucket penetrates the ground, a small vertical impulse force is applied and digging interaction forces as described in [13] are presented thereafter. While in the ground, soil accumulation is computed as a function of trench depth and length. This value is used to apply a proportional downward force upon exiting the soil to simulate the weight of the load. When the bucket is subsequently tipped past a fixed dump angle, the vertical force is removed and a mound of soil is graphically rendered falling from the bucket to the ground. A snapshot of the graphic environment[1] is shown in Figure 8.9.

[1.] The graphical environments were created by a fellow graduate student, Simon P. DiMaio.

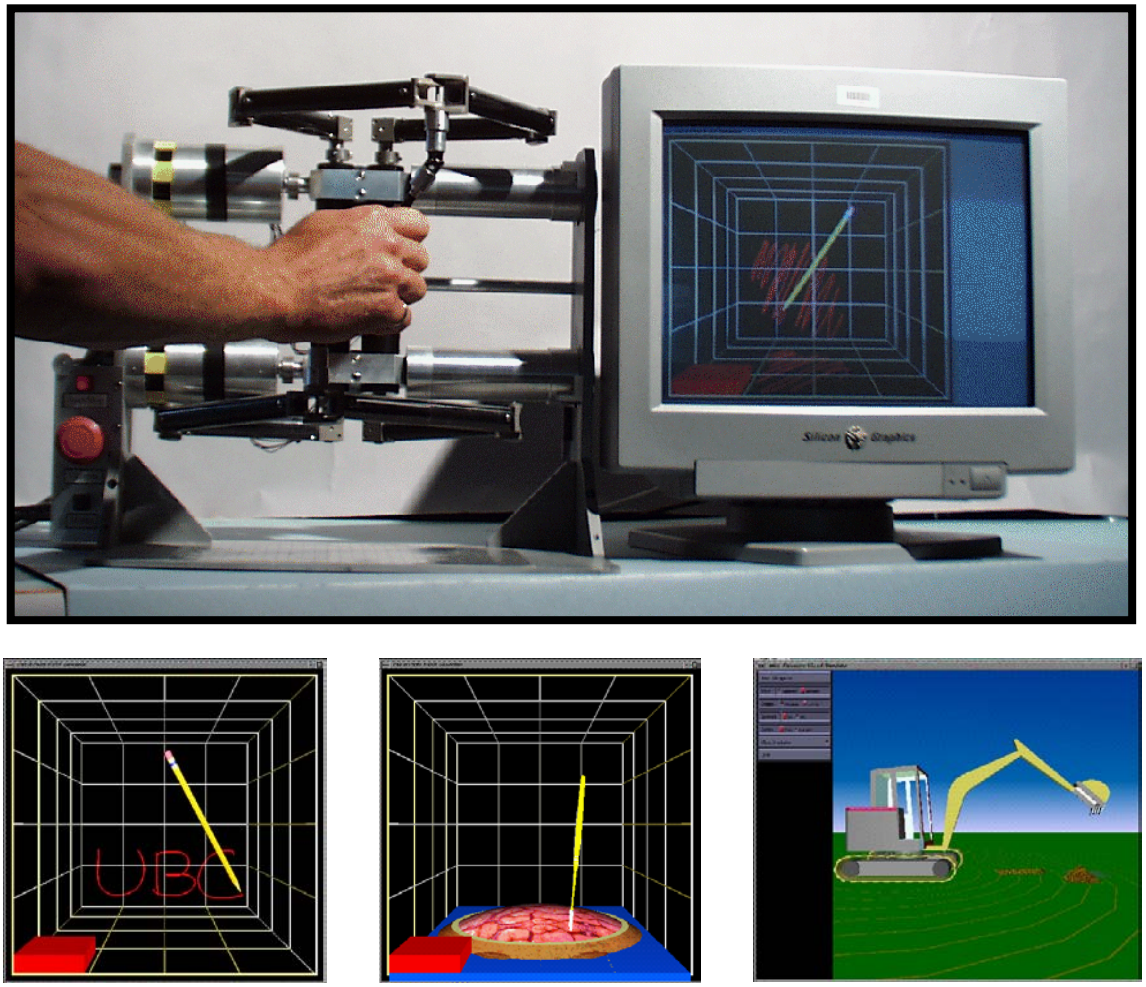


Figure 8.9: Graphical Environments

8.6 Stability

The system is very stable as can be seen in Figure 8.10 which shows motor currents and penetration depths when one end of the haptic pencil is manoeuvred about the bottom, back, left corner of the stiff virtual box. The stiff box is implemented using proportional gains of *3N/cm* (horizontal) and *2.1N/cm* (vertical), differential gains of *0.05Ns/cm* (horizontal) and *0.02Ns/cm* (vertical), and braking gains of *6Ns/cm* (horizontal) and *3Ns/cm* (vertical) at the contacting end of the haptic pencil. The top three traces are the control signals applied to the three lower actuators q_1 through q_3 (see Figure 8.1) and the bottom three traces are the penetration depths of the pencil tip into the virtual walls and floor. The pencil is lying flat on the floor without any human contact between t_0 and t_1, is held in free space between t_1 and t_2, is pressed into the walls, floor and adjoining corners between t_2 and t_3, is pressed into the corner adjoining all three surfaces between t_3 and t_4, is quickly knocked all round the corner between t_4 and t_5 and is returned to free space between t_5 and t_6. Note that between t_4 and t_5, rapid changes in current and penetration depth occur

because the operator is rapidly impacting three virtual surfaces but the system remains stable. Note the correspondence between penetration depths and motor currents and the absence of any extraneous oscillations. The trajectory of the pencil tip with respect to the corner of the box and the resultant reaction forces are shown in Figure 8.11.

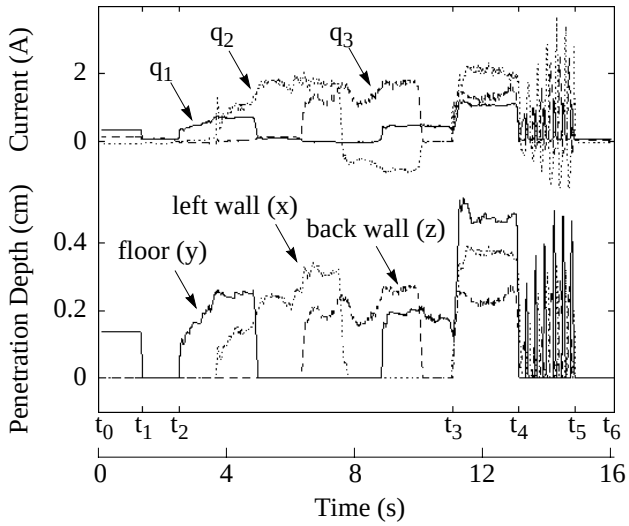


Figure 8.10: Motor Currents and Wall Penetration Depths

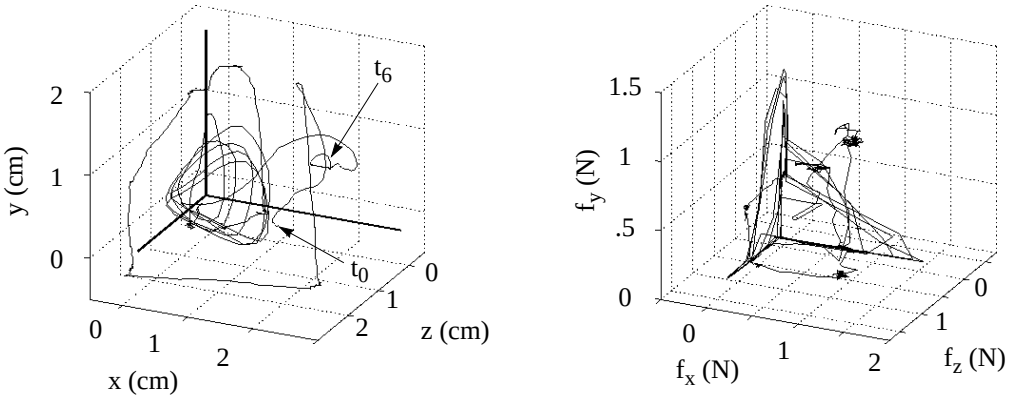


Figure 8.11: Tip Position and Reaction Force

8.7 Concluding Remarks

The physical components of the prototype haptic system have been described in detail. The haptic pen is constructed from light weight materials and low friction components to minimize mechanical impedance. Its measured performance capabilities are comparable to existing state-of-the-art haptic devices such as the SensAble PHANToM [48] even through the Twin-Pantograph haptic pen has two additional degrees-of-

freedom. The general applicability of the device is demonstrated by simulating diverse virtual environments that include an ultra light pencil and a heavy duty excavator. The pencil simulation is used to observe system stability during hard contact simulations. These combined results confirm that the design procedure was successful at producing a high performance yet stable device.

In the following chapter, a proposal for a further improvement to haptic performance using redundant actuation is investigated. It involves extending the workspace of a high-performance fine stage robot by mounting it on a coarse stage robot and joining the two end-effectors with a flexible coupling.

Chapter 9

A Dual-Stage Coarse-Fine Haptic Interface

9.1 Introduction

Although the haptic pen can emulate a wide range of impedances, it is not ideal as defined by Lawrence and Chapel [42]. It is incapable of producing static forces large enough to counteract a sustained, maximum hand force. This requires the high gear ratios of an admittance device. It is also incapable of generating forces at the top of the frequency spectrum of human sensory capabilities (*500-1000 Hz*). This requires the low mass and friction of a device such as a maglev joystick [26]. To obtain performance levels beyond those of a conventional robot, it was proposed in [26], [65] that the workspace of a maglev joystick could be expanded by transporting it on a coarse motion stage. Unfortunately, such a device would be force magnitude limited by the maglev. In this chapter, it is shown that the force limitation can be removed by including a parallel flexible coupling between the coarse and fine stages. This results in a device with both the force bandwidth of the maglev and the motion range and force magnitude of a conventional admittance device. A redundant device such as this may possibly fulfill the requirements that define ideal haptic performance.

This chapter is organized as follows. In Section 9.2 the characteristics of the maglev joystick are cited. In Section 9.3 a proposal for a series coarse-fine system with a parallel compliant coupling is described. In Section 9.4 some concluding remarks are made.

9.2 Characteristics of a Maglev Joystick

The performance figures for a maglev joystick reported in [65], [66] are shown in Table 9.1. These figures do, however, vary with the size of the specific device, with the force to mass ratio deteriorating slightly with scale [67]. The consensus figures (Table 2.1) for what is deemed an adequate workspace clearly exceed what can be achieved with a single-stage maglev device since it would be extremely difficult, if not impossible, to increase the workspace by an order of magnitude. Although the continuous maximum forces generated by the maglev device only marginally satisfy the quoted requirements, the frequency response specifications (assuming they apply to the motion travel accommodated by maglev devices) exceed them.

Table 9.1: Maglev Joystick Performance Specifications

Performance Specification	Value
Diameter	13 cm
Height	11 cm
Nominal Motion Range (Translation)	± 4.5 mm
Nominal Motion Range (Rotation)	± 7˚
Closed-Loop Bandwidth (Translation)	> 30 Hz
Closed-Loop Bandwidth (Rotation)	> 15 Hz
Maximum Impulse Force[A,B]	60N
Maximum Impulse Torque[A,B]	6 Nm
Maximum Continuous Force[A,B]	18 N
Maximum Continuous Torque[A,B]	0.6 Nm
Force Bandwidth[C]	> 1000 Hz
Force Resolution[D]	0.1 N
Maximum Acceleration[A]	90 m/s^2
Flotor Mass	0.65 Kg
Position Resolution	1 µm

A. Along z-axis; roughly 30% smaller along x,y axes
B. Coil amplifier dependent
C. Based on coil time constant
D. Digital-to-analog converter dependent

Although the limited continuous force capabilities of a maglev device may reduce the maximum static stiffness it can emulate, its ability to emulate contact with stiff walls is enhanced by its high acceleration. This is demonstrated in [63] where ten subjects were asked to move a maglev joystick against a virtual wall and press a button upon detecting it. The surface model was a spring with stiffnesses ranging from *1-10 N/mm* and a removable braking force that brings the joystick to rest within a single control sample upon penetration. This corresponds to accelerations of up to *90 m/s^2*. The results showed that subjects could locate the wall more accurately and with less penetration when the braking pulse was present and that performances did not improve significantly with stiffnesses above *6-7 N/mm*.

An analysis of the data using ANOVA tables and F-ratios showed that both increased stiffness and acceleration significantly decrease wall penetration and increase the accuracy with which the wall can be located, with minimal interaction between the two. Consequently for stiff wall emulation, the maximum force exerted by the haptic interface is no more important than its maximum acceleration. It is, however, shown in the next section that the coarse-fine approach used to enlarge the workspace of a maglev device can also be used to increase its maximum force by means of a compliant coupling.

9.3 A Parallel Compliant Coupling

Morrell & Salisbury [52] show that the high force resolution and bandwidth of a micro actuator acting on a load can be augmented by the high force capability of a macro actuator by connecting the two actuators via a flexible coupling. A drawing of a flexibly coupled parallel dual-stage system is shown in Figure 9.1, a model is shown in Figure 9.2 and equations for the force applied to the load and the impedance seen by the load are shown in (9.1) and (9.2) for the general case and in (9.6) through (9.9) for very stiff ($Z_t \rightarrow \infty$) and compliant ($Z_t \rightarrow 0$) couplings. In equations (9.1) through (9.9), the subscripts *c*, *f*, *t* and *l* denote coarse, fine, coupling and load components respectively and the subscript *e* denotes system equivalents. Both macro and micro actuators act in parallel with respect to a common mechanical ground. As opposed to the series macro-micro actuation schemes proposed in the past [35] [62] [63] [66] [69] [86] [91], the parallel combination enjoys the full static force capability of the macro actuator but because of the parallel connection, inherits the limited motion range of the micro actuator.

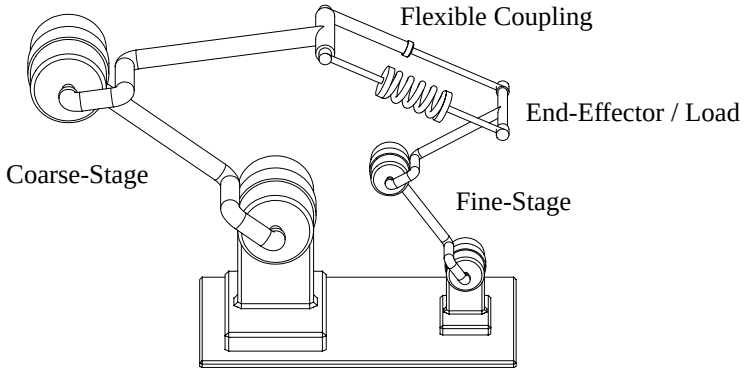


Figure 9.1: Parallel Dual-Stage Device with a Flexible Coupling

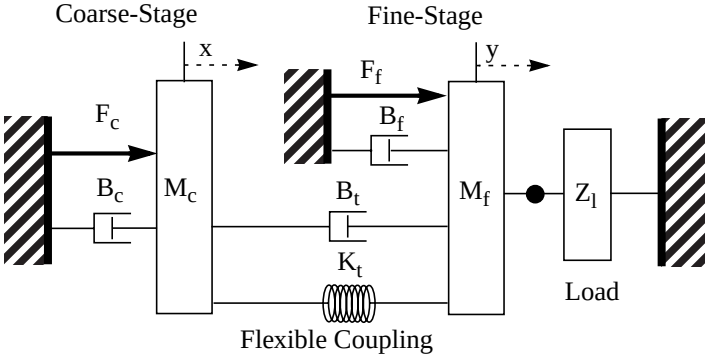


Figure 9.2: Model of a Parallel Dual-Stage Device

$$F_e = \frac{Z_t F_c}{Z_c + Z_t} + F_f \tag{9.1}$$

$$Z_e = \frac{Z_t Z_c}{Z_c + Z_t} + Z_f \tag{9.2}$$

$$Z_c = M_c s + B_c \tag{9.3}$$

$$Z_f = M_f s + B_f \tag{9.4}$$

$$Z_t = B_t + K_t s^{-1} \tag{9.5}$$

$$\lim_{Z_t \to 0} F_e = F_f \tag{9.6}$$

$$\lim_{Z_t \to \infty} F_e = F_c + F_f \tag{9.7}$$

$$\lim_{Z_t \to 0} Z_e = Z_f \tag{9.8}$$

$$\lim_{Z_t \to \infty} Z_e = Z_c + Z_f \tag{9.9}$$

While it is agreed that the actuators must be flexibly coupled in parallel to sum force contributions, it should be pointed out that they may be rigidly connected in series with little repercussion. This is done by mounting the base of the fine-stage on the end-effector of the coarse-stage and maintaining the flexible coupling between the two end-effectors. A drawing of the hybrid series/parallel dual-stage device is shown in Figure 9.3, a model is shown in Figure 9.4 and equations for the effective force and impedance seen by the load are shown in (9.10) through (9.15).

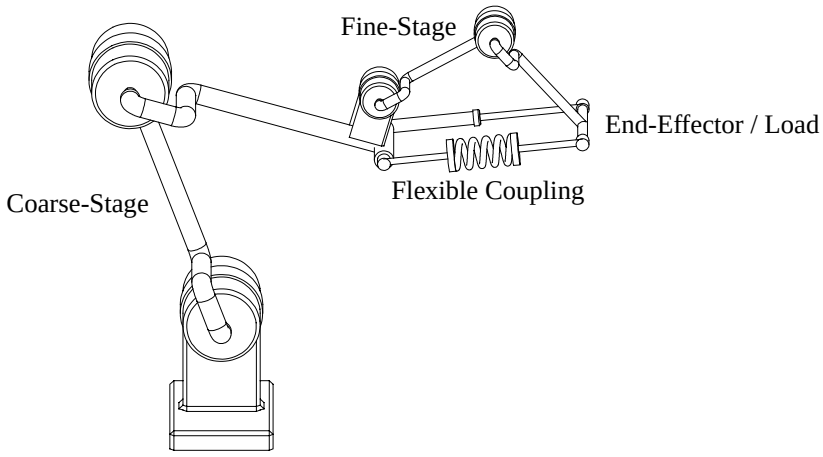


Figure 9.3: Hybrid Dual-Stage Device with a Flexible Coupling

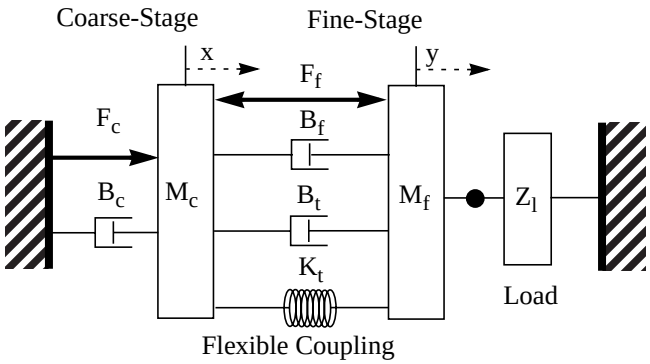


Figure 9.4: Model of a Hybrid Dual-Stage Device

$$F_e = \frac{(Z_t + B_f)F_c + Z_c F_f}{Z_c + Z_t + B_f} \tag{9.10}$$

$$Z_e = \frac{(Z_t + B_f)Z_c}{Z_c + Z_t + B_f} + Z_f \tag{9.11}$$

Assuming $B_f \ll |Z_c|$

$$\lim_{Z_t \to 0} F_e = F_f \tag{9.12}$$

$$\lim_{Z_t \to \infty} F_e = F_c \tag{9.13}$$

$$\lim_{Z_t \to 0} Z_e = Z_f \tag{9.14}$$

$$\lim_{Z_t \to \infty} Z_e = Z_c + Z_f \tag{9.15}$$

Just as with the parallel device described in [52], the force capability and internal impedance of the dual-stage device mimic that of the fine-stage for a compliant coupling and approach that of the coarse-stage as the coupling is made stiffer. The difference is that the workspace size of the hybrid device is dictated by the coarse-stage which must, however, bear the additional payload of the fine-stage.

For the hybrid device, the effect of coupling impedance on the maximum force magnitude and mechanical impedance magnitude presented to the load over a range of frequencies is illustrated in Figure 9.5 and Figure 9.6 respectively. The coupling impedance $B_t + K_t s^{-1}$ is selected by varying K_t and selecting B_t such that the coupling supporting the fine-stage mass is critically damped (i.e. $\zeta=1$).

$$\frac{1}{s^2 M_f + sB_t + K_t} = \frac{1}{s^2 + 2\zeta\omega_0 s + \omega_0^2} = \frac{1}{s^2 + 2\omega_0 s + \omega_0^2} \tag{9.16}$$

F_f, M_f and B_f refer to the fine-stage force, mass and damping respectively and are assigned values similar to those of the maglev device described in Table 9.1. F_c, M_c and B_c refer to the coarse-stage force, mass and damping respectively and are assigned the typical values $F_c = 10F_f$, $M_c = 50M_f$ and $B_c = 1000B_f$.

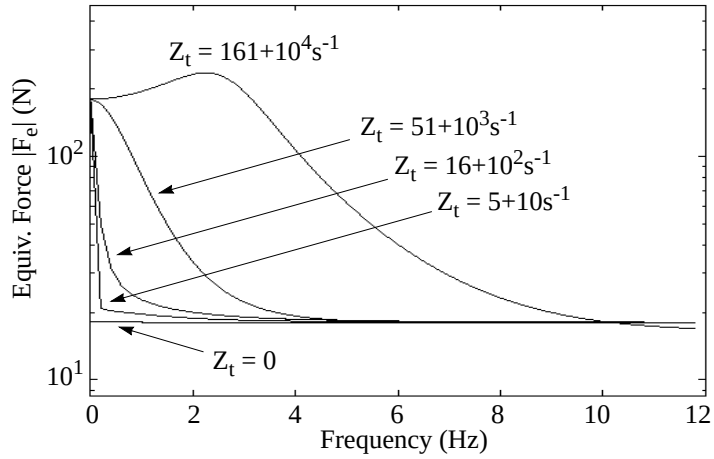


Figure 9.5: Equivalent Force

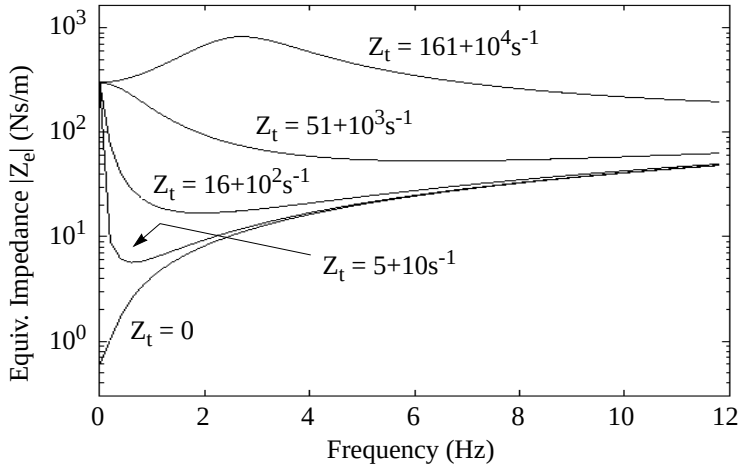


Figure 9.6: Equivalent Impedance

Figure 9.5 and Figure 9.6 show that the coupled dual-stage device has the low mechanical impedance and force capabilities of the fine stage device when the coupling impedance is low and the high mechanical impedance and force capabilities of the coarse stage device when the coupling impedance is high. This allows one to optimally trade off the characteristics of the coarse-stage and the fine-stage by choosing the compliance of the flexible coupling. One could also introduce a stiffening coupling that is very compliant when the fine-stage is near its home position and becomes increasingly rigid as the fine-stage approaches its workspace limit. Consider, for example, a dual-stage system consisting of a geared Twin-Pantograph

coarse-stage and a maglev fine-stage with a piece of light foam rubber mounted between the flotor and stator of the maglev. The material would provide minimal resistance when fully extended and would stiffen as it is compressed. The resulting device would supply the high fidelity small-signal performance of the maglev when small forces are applied that do not displace the maglev flotor much from its centre position as well as the large workspace and force of the Twin-Pantograph when larger forces are applied which hold the flotor off-centre and compress the material.

9.4 Concluding Remarks

A dual-stage device has been proposed which combines the high force magnitude and large workspace of a coarse-stage robot such as a Twin-Pantograph with the high force and acceleration bandwidth of a fine-stage robot such as a maglev joystick. It involves mounting the two devices rigidly in series and flexibly (i.e. using a flexible coupling) in parallel. It is shown that the dual-stage device inherits the characteristics of the coarse-stage when a stiff coupling is used and the characteristics of the fine-stage when a compliant coupling is used. Therefore, if a stiffening coupling is used, the device will behave like the coarse-stage when large forces are applied and like the fine-stage when small forces are applied, thereby automatically assuming the necessary characteristics to accommodate a dynamic environment.

In the final chapter, the contributions of this thesis are summarized and some future work is proposed.

Chapter 10

Conclusions

This chapter summarizes the contributions made in this thesis and suggests some areas for further research.

10.1 Contributions

The contributions are summarized as follows.

10.1.1 The Global Isotropy Index

A new condition index called the Global Isotropy Index (GII) is introduced. The GII is a global measure of isotropy throughout a device's workspace. Unlike existing local measures, the GII retains scale information and does not need to be checked by a secondary measure. Unlike existing global measures, the GII is a worst-case measure that does not tolerate intermittent displays of poor performance anywhere in the workspace.

10.1.2 Non-Homogeneous Design and Actuator Specifications

A method for normalizing and scaling a design matrix is introduced. A task space matrix S_T is used to fix the demands of the application while a joint space matrix S_J holds design variables which are used to select actuator scale factors. This method removes all physical units from the design matrix and skews the condition index so that it evaluates performance according to a non-uniform task-space specification. It also allows the robot geometry and actuator sizes to be chosen simultaneously for improved performance, particularly with serial mechanisms.

10.1.3 The Culling Optimization Algorithm

A new optimization algorithm is introduced. It is a discrete branch-and-bound algorithm for solving GII and minimax problems that is unhampered by many of the difficulties associated with existing optimization algorithms. It can be applied to any objective function (e.g. no estimation or integration required), guarantees a global optimum, is insensitive to initial conditions and is unhampered by nonlinear, non-differentiable, discontinuous, non-convex, or unbounded objective functions. The algorithm is also very easy to implement.

10.1.4 Haptic Interface Performance Specifications

A haptic interface performance specification has been derived from the observed shortcomings of earlier haptic interface designs and a model of a human hand interacting with a haptic interface. Biomechanical studies were then conducted to provide target values pertaining to the above criteria. The studies obtained the maximum force/torque and velocity capabilities of a human hand holding a pen shaped instrument.

10.1.5 The Twin-Pantograph Hybrid Manipulator

A novel 6-DOF hybrid robot manipulator called the Twin-Pantograph is introduced. Its equations of motion are used to show that its reachable workspace can be made free of singular positions if a redundant actuator is included and some geometric constraints are observed. A 5-DOF version is also introduced that has all of its motors in the base like a parallel device but has a much larger workspace and better static force isotropy.

10.1.6 The Twin-Pantograph Haptic Pen

A 5-DOF haptic pen is built and controlled in three virtual environments which include a virtual pencil, a virtual scalpel and a virtual excavator. These diverse environments demonstrate the scope of the device and its capability to simulate a wide range of impedances. Its measured performance benchmarks are found to rival a current state-of-the-art mechanism (the SensAble PHANToM [48]) even through the Twin-Pantograph haptic pen has two additional degrees of freedom in rotation. Lastly, it is shown that stability is maintained by considering a hard contact situation involving three intersecting surfaces. The results confirm that the proposed design procedure was able to create a high performance yet stable device.

10.1.7 A Coarse-Fine Haptic Interface

A proposal is made for a device that could potentially support an even wider virtual impedance range than the haptic pen. It is comprised of a coarse-stage admittance device connected in series with a fine-stage impedance device whose end-effectors are joined in parallel by a flexible coupling. It is shown that the stiffness of the coupling determines which performance characteristics the dual-stage device inherits. It is suggested that by using a stiffening coupling, the dual-stage device may actually share the high performance characteristics (i.e. large workspace, large force magnitude, high acceleration/bandwidth) of both devices.

10.1.8 Robot Design Examples

Many robot design examples are presented which include RR, RRR and RPR serial manipulators, a planar 3-DOF parallel manipulator, a planar 5-bar linkage, a Stewart Platform, an Inoue Platform and a Twin-Pantograph. A variety of aspects are considered including compactness, workspace size, kinematic (force & velocity) isotropy, dynamic isotropy and singular value smoothness. Some widely applicable results come from these examples such as the kinematic optimality of a symmetric 5-bar linkage and the optimal postures associated with different workspace positions.

10.2 Future Work

Proposals for future work are summarized as follows.

10.2.1 An Experimental Dual Stage System

The theoretical work regarding the flexibly coupled dual-stage coarse-fine robot was not substantiated by a physical design example. To complete this work, a 1-DOF dual-stage system is planned to demonstrate the correspondence between theoretical and actual results. This is to be followed by a full scale, 6-DOF, Twin-Pantograph/Maglev wrist coarse-fine haptic interface.

10.2.2 Alternative Applications of the Culling Algorithm

Presently, the culling algorithm has only been applied to robot design problems. However, robot design is not the only field where minimax problems are prevalent. They are common in other fields such as filter design. It is likely that the culling algorithm will be just as effective at solving these other types of problems as they are at solving robot design problems.

Bibliography

[1] S.K. Adams, P.J. Peterson, "Maximum Voluntary Hand Grip Torque for Circular Electrical Connectors", *Human Factors*, v. 30, no. 6, pp. 733-745, 1988.

[2] P.A. Anderson, C.E. Chanoski, D.L. Devan, B.L. McMahon, E.P. Whelan, "Normative Study of Grip and Wrist Flexion Strength Employing a BTE Work Simulator", *The Journal of Hand Surgery*, v. 15A, no. 3, pp.420-425, May 1990.

[3] J. Angeles, "Kinematic Isotropy in Humans and Machines", *Proceedings of IFToMM 9th World Congress on Theory of Machines & Mechanisms* (Milan, Italy), V. 1, pp. XLII-XLIX, Aug. 29 - Sept. 2, 1995.

[4] T.L. Brooks, "Telerobotic Response Requirements", *Proceedings of IEEE International Conference on Systems, Man and Cybernetics* (Los Angeles, California), pp. 113-120, Nov. 4-7, 1990.

[5] M.S. Campbell, T.A. Marsland, "A Comparison of Minimax Tree Search Algorithms", *Artificial Intelligence*, v. 20, pp. 347-367, 1983.

[6] D.B. Chaffin, G.B.J. Andersson, "Occupational Biomechanics", John Wiley & Sons, 1991.

[7] H.D. Chiang, C.C. Chu, "A Systematic Search Method for Obtaining Multiple Local Optimal Solutions of Nonlinear Programming Problems", *Proceedings of IEEE ANNPS 93 Second International Forum on Neural Networks to Power Systems* (Yokohama, Japan), pp. 447-450, Apr. 19-22, 1993.

[8] K. Cleary, T. Brooks, "Kinematic Analysis of a Novel 6-DOF Parallel Manipulator", *Proceedings of 1993 IEEE International Conference on Robotics and Automation* (Atlanta, Georgia), pp. 708-713, May 2-6, 1993.

[9] J.E. Colgate, G.G. Schenkel, "Passivity of a Class of Sampled-Data Systems: Application to Haptic Interfaces", *Journal of Robotic Systems*, v. 14, no. 1, pp. 37-47, Jan. 1997.

[10] C.L. Collins, G.L. Long, "The Singularity Analysis of an In-Parallel Hand Controller for Force-Reflected Teleoperation", *IEEE Transactions on Robotics and Automation*, v. 11, no. 5, pp. 661-669, Oct. 1995.

[11] J.J. Craig, "Introduction to Robotics Mechanics and Control", 2nd Ed., Addison-Wesley, 1989.

[12] W.C. Davidon, "Variable Metric Method for Minimization", Argonne National Laboratory, ANL-5990 Rev., University of Chicago, 1959.

[13] S.P. DiMaio, S.E. Salcudean, C. Reboulet, S. Tafazoli, K. Hashtrudi-Zaad, "A Virtual Excavator for

Controller Development and Evaluation", *Proceedings of IEEE International Conference on Robotics and Automation* (Leuven, Belgium), pp. 52-58, May. 16-23, 1998.

[14] K. van den Doel, D.K. Pai, "Performance Measures for Constrained Systems", *IEEE Transactions on Robotics and Automation*, v. 13, no. 2, pp. 278-289 Apr. 1997.

[15] K. van den Doel, D.K. Pai, "Performance Measures for Robot Manipulators: A Unified Approach", *The International Journal of Robotics Research*, v. 15, no. 1, pp. 92-111, Feb. 1996.

[16] K. Doty, C. Melchiorri, C. Bonevento, "A Theory of Generalized Inverses Applied to Robotics", *The International Journal of Robotics Research*, v. 12, no. 1, pp. 1-19, Feb. 1993.

[17] D. Dowson, V. Wright, "An Introduction to the Bio-Mechanics of Joints and Joint Replacement", *Mechanical Engineering Publications*, 1982.

[18] P. Fischer, R. Daniel, K.V. Siva, "Specification and Design of Input Devices for Teleoperation", *Proceedings of IEEE International Conference on Robotics and Automation* (Cincinnati, Ohio), pp. 540-545, May 13-18, 1990.

[19] R. Fletcher, C.M. Reeves, "Function Minimization by Conjugate Gradients", *Computer Journal*, v. 7, no. 2, pp. 149-154, 1964.

[20] R.L. Fox, "Optimization Methods for Engineering Design", Addison-Wesley, Reading, Massachusetts, 1971.

[21] C. Gosselin, "Dexterity Indices for Planar and Spatial Robotic Manipulators", *Proceedings of IEEE International Conference on Robotics and Automation* (Cincinnati, Ohio), pp. 650-655, May 13-18, 1990.

[22] C. Gosselin, J. Angeles, "A Global Performance Index for the Kinematic Optimization of Robot Manipulators", *Transaction of the ASME, Journal of Mechanical Design*, v. 113, pp. 220-226, Sept. 1991.

[23] G.M. Griner, "A Comparison of Simulated Evolution and Genetic Evolution Performance", *Proceedings of First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence* (Orlando, Florida), pp. 374-378, Jun. 27-29, 1994.

[24] V. Hayward, O. Astley, "Performance Measures for Haptic Interfaces", *Proceedings of* 7^{th} *International Symposium on Robotics Research* (Herrshing, Germany), Oct. 21-24, 1995, Springer Verlag, pp. 195-207, 1996.

[25] V. Hayward, J. Choksi, G. Lanvin, C. Ramstein, "Design and Multi-Objective Optimization of a Linkage for a Haptic Interface", *Proceedings of ARK '94, 4th International Workshop on Advances in Robot Kinematics* (Ljubliana, Slovenia), pp. 352-359, Jun. 1994,

[26] R.L. Hollis, S.E. Salcudean, P.A. Allen, “A Six Degree-of-Freedom Magnetically Levitated Variable Compliance fine Motion Wrist: Design, Modelling and Control”, *IEEE Transactions on Robotics and Automation*, v. 7, pp. 320-332, Jun. 1991.

[27] J.B. Hunt, “Dynamic Vibration Absorbers”, *Mechanical Engineering Publications Ltd.*, 1979.

[28] I. Hunter, L. Jones, T. Doukoglou, S. Lafontaine, P. Hunter, M. Sagar, “Ophthalmic Microsurgical Robot and Surgical Simulator”, *Telemanipulator and Telepresence Technologies*, v. 2351, pp. 184-190, 1994.

[29] H. Iwata, “Artificial Reality with Force-Feedback: Development of Desktop Virtual Space with Compact Master Manipulator”, *SIGGRAPH* (Dallas, USA), v. 24, no. 4, pp. 165-170, Aug. 6-10, 1990.

[30] H. Iwata, “Pen-based Haptic Virtual Environment”, *IEEE International Symposium on Virtual Reality* (Seattle, Washington), pp. 287-292, 1993.

[31] D.C. Jiang, K.L. Teo, W.Y. Yan, “A New Computational Method for the Functional Inequality Constrained Minimax Optimization Problem”, *Proceedings of the 34th IEEE Conference on Decision & Control* (New Orleans, Los Angeles), pp. 2310-2315, Dec. 13-15, 1995.

[32] D.R. Jones, C.D. Perttunen, B.E. Stuckman, “Global Optimization: Beyond the Lipschitzian Model”, *Proceedings of IEEE International Conference on Systems, Man and Cybernetics* (Chicago, Illinois), pp. 566-570, Oct. 18-21, 1992.

[33] J.P. Karidis, G. McVicker, J.P. Pawletko, L.C. Zai, M. Goldowsky, R.E. Brown, R.R. Comulada, “The Hummingbird Minipositioner - Providing Three-Axis Motion at 50 G’s With Low Reactions”, *Proceedings of IEEE International Conference on Robotics and Automation* (Nice, France), pp. 685-692, May 10-15, 1992.

[34] A.J. Kelley, S.E. Salcudean, “The Development of a Force-Feedback Mouse and its Integration into a Graphical User Interface”, *Proceedings of the International Mechanical Engineering Congress and Exposition* (Chicago, Illinois), DSC-Vol. 55-1, pp. 287-294, Nov. 6-11, 1994.

[35] O. Khatib, “Inertial Properties in Robotic Manipulation: An Object-Level Framework”, *The International Journal of Robotics Research*, v. 14, no. 1, pp.19-36, Feb. 1995.

[36] O. Khatib, B. Roth, “New Robot Mechanisms for New Robot Capabilities”, *Proceedings of IEEE/RSJ International Workshop on Intelligent Robots and Systems* (Osaka, Japan), pp. 44-49, Nov. 3-5, 1991.

[37] J-O. Kim, P.K. Khosla, “Dexterity Measures for Design and Control of Manipulators”, *Proceedings of IEEE/RSJ International Workshop on Intelligent Robots and Systems* (Osaka, Japan), pp. 758-763, Nov. 3-5, 1991.

[38] M. Kim, D. Yoon, S. Cha, J. Jin, S. Lim, K. Choi, “A Modified Simulated Diffusion Algorithm for Global Optimization of Model Parameters”, *Proceedings of the 1994 IEEE International Workshop on Numerical Modeling of Processes and Devices for Integrated Circuits* (Honolulu, Hawaii), Jun. 5-6, 1994.

[39] M.V. Kircanski, “Robotic Isotropy and Optimal Robot Design of Planar Manipulators”, *Proceedings of IEEE International Conference on Robotics and Automation* (San Diego, California), pp. 1100-1105, May 8-13, 1994.

[40] T. Kotoku, K. Komoriya, K. Tanie, “A Force Display System for Virtual Environments and its Evaluation”, *IEEE International Workshop on Robot and Human Communication* (Tokyo, Japan), pp. 246-251, Sept. 1-3, 1992.

[41] R. Kurtz, V. Hayward, “Multiple-Goal Kinematic Optimization of a Parallel Spherical Mechanism with Actuator Redundancy”, *IEEE Transactions on Robotics and Automation*, v. 8, Oct. 1992.

[42] D.A. Lawrence, J.D. Chapel, “Performance Trade-Offs for Hand Controller Design”, *Proceedings of IEEE International Conference on Robotics and Automation* (San Diego, California), pp. 3211-3216, May 8-13, 1994.

[43] S. Leguay-Durand, C. Reboulet, “Optimal Design of a Redundant Spherical Parallel Manipulator”, *Robotica, International Journal of Information, Education and Research in Robotics and Artificial Intelligence*, v. 15, pp. 399-405, part 4, Jul.-Aug. 1997.

[44] H. Lipkin, J. Duffy, “Hybrid Twist and Wrench Control for a Robotic Manipulator”, *Transactions of the ASME Journal of Mechanisms, Transmissions & Automation in Design*, v. 110, pp. 138-144, Jun. 1988.

[45] J.T-H. Lo, “A New Approach to Global Optimization and its Applications to Neural Networks”, *Proceedings of IEEE IJCNN International Joint Conference on Neural Networks* (Baltimore, Mariland), v. 4, pp, 600-605, Jun. 7-11, 1992.

[46] O. Ma, J. Angeles, “Optimum Architecture Design of Platform Manipulators”, 91 ICAR. *Proceedings of ICAR ‘91, 5th International Conference on Advanced Robotics. Robots in Unstructured Environments* (Pisa, Italy), pp. 1130-1135, Jun. 19-22, 1991.

[47] O. Ma, J. Angeles, “Optimum Design of Manipulators Under Dynamic Isotropy Conditions”, *Proceedings of IEEE International Conference on Robotics and Automation* (Atlanta, Georgia), pp. 470-475, May 2-6, 1993.

[48] T.H. Massie, J.K. Salisbury, “The PHANToM Haptic Interface: A Device for Probing Virtual Objects”, *Proceedings of the International Mechanical Engineering Congress and Exposition* (Chicago, USA), DSC-Vol. 55-1, pp. 295-301, Nov. 6-11, 1994.

[49] J-P. Merlet, “DEMOCRAT: A DEsign MethOdology for the Conception of Robot with parallel ArchiTecture”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, v. 3, pp. 1630-1636, Sept. 7-11, 1997.

[50] J-P. Merlet, “Direct Kinematics of Planar Parallel Manipulators”, *Proceedings of IEEE International Conference on Robotics and Automation* (Minneapolis, Minnesota), v. 4, pp. 3744-3749, Apr. 22-28, 1996.

[51] P.A. Millman, M. Stanley, J.E. Colgate, “Design of a High Performance Haptic Interface to Virtual Environments”, *IEEE International Symposium on Virtual Reality* (Seattle, Washington), pp. 216-222, 1993.

[52] J.B. Morrell, J.K Salisbury, “In Pursuit of Dynamic Range: Using Parallel Coupled Actuators to Overcome Hardware Limitations”, *Proceedings of Fourth International Symposium on Experimental Robotics* (Stanford, California), ISER’95, pp. 165-170, Jun. 30 - Jul. 2, 1995.

[53] J.C. Olivier, “Obtaining Optimum Minimum Points of Error Functions: A Neural Network Approach”, *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing*, v. 42, no. 11, Nov. 1995.

[54] E.R. Panier, A.L. Tits, “A Globally Convergent Algorithm with Adaptively Refined Discretization for Semi-Infinite Optimization Problems Arising in Engineering Design”, *IEEE Transactions on Automatic Control*, v. 34, no. 8, pp. 903-908, August 1989.

[55] F.C. Park, “Optimal Robot Design and Differential Geometry”, *Journal of Mechanical Design*, v. 117, pp. 87-92, Special Issue B, Jun. 1995.

[56] J-H Park, H. Asada, “Concurrent Design Optimization of Mechanical Structure and Control for High-Speed Robots”, *Transactions of the ASME, Journal of Dynamic Systems Measurement and Control*, v. 116, no. 3, pp. 344-356, Sept. 1994.

[57] A.T. Phillips, J.B. Rosen, “Computational Comparison of Two Methods for Constrained Global Optimization”, *Journal of Global Optimization*, v. 5, pp. 325-332, 1994.

[58] E.P. Popov, “Introduction to Mechanics of Solids”, Prentice-Hall Inc. 1968.

[59] M.J.D. Powell, “An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives”, *Computer Journal*, v. 7, no. 4, pp. 303-307, 1964.

[60] C. Ramstein, V. Hayward, “The PANTOGRAPH: A Large Workspace Haptic Device for a Multi-Modal Human-Computer Interaction”, *Proceedings of ACM/SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA), Apr. 1994.

[61] F. Ranjbaran, J. Angeles, M.A. Gonzalez-Palacios, R.V. Patel, “The Mechanical Design of a 7-Axes Manipulator with Kinematic Isotropy”, *Journal of Intelligent & Robotic Systems*, v. 14, no. 1, pp.

21-41, Sept. 1995.

[62] J.R. Sagli, O. Egeland, "Dynamic Coordination and Actuator Efficiency Using Momentum Control for Macro-Micro Manipulators", *Proceedings of IEEE International Conference on Robotics & Automation* (Sacramento, California), pp. 1201-1206, Apr. 9-11, 1991.

[63] S.E. Salcudean, T. Vlaar, "On the Emulation of Stiff Walls and Static Friction with a Magnetically Levitated Input-Output Device", *Proceedings of the International Mechanical Engineering Congress and Exposition* (Chicago, USA), DSC-Vol. 55-1, pp. 303-309, Nov. 6-11, 1994.

[64] S.E. Salcudean, T.D. Vlaar, "On the Emulation of Stiff Walls and Static Friction with a Magnetically Levitated Input/Output Device", *Transactions of the ASME, Journal of Dynamic Systems Measurement and Control*, v. 119, no. 1, pp. 127-132, Mar. 1997.

[65] S.E. Salcudean, N.M. Wong, R.L. Hollis, "A Force-Reflecting Teleoperation System with Magnetically Levitated Master and Wrist", *Proceedings of IEEE International Conference on Robotics and Automation* (Nice, France), May 10-15, 1992.

[66] S.E. Salcudean, N.M. Wong, "Coarse-Fine Motion Coordination and Control of a Teleoperation System with Magnetically Levitated Master and Wrist", *Third International Symposium on Experimental Robotics* (Kyoto, Japan), Oct. 28-30, 1993.

[67] S.E. Salcudean, J. Yan, "Towards a Force-Reflecting Motion-Scaling System for Microsurgery", *Proceedings of IEEE International Conference on Robotics and Automation* (San Diego, California), pp. 2296-2301, May 9-12, 1994.

[68] F. Schoen, "Stochastic Techniques for Global Optimization: A Survey of Recent Advances", *Journal of Global Optimization*, v. 1, pp. 207-228, 1991.

[69] A. Sharon, N. Hogan, D.E. Hardt, "High Bandwidth Force Regulation and Inertia Reduction Using a Macro/Micro Manipulator System", *Proceedings of IEEE International Conference on Robotics & Automation* (Philadelphia, USA), pp. 126-132, Apr. 24-29, 1988.

[70] J.E.E. Sharpe, "Technical and Human Operational Requirements for Skill Transfer in Teleoperations", *International Symposium on Teleoperation and Control* (Bristol, England), pp. 175-187, Jul. 1988.

[71] Z. Shiller, S. Sundar, "Design of Multi-Degree-of-Freedom Mechanisms for Optimal Dynamic Performance", *Journal of Mechanical Design*, v. 115, no. 2, pp. 199-206, Jun. 1993.

[72] M.W. Spong, M. Vidyasagar, "Robot Dynamics and Control", John Wiley & Sons, 1989.

[73] M. Srinivas, L.M. Patnaik, "Genetic Algorithms: A Survey", *Computer*, v. 27, Iss. 6, pp. 17-26, Jun. 1994.

[74] L. Stocco, S.E. Salcudean, "A Coarse-Fine Approach to Force-Reflecting Hand Controller Design", *Proceedings of IEEE International Conference on Robotics and Automation* (Minneapolis, Minnesota), v. 1, pp. 404-410, Apr. 22-28, 1996.

[75] L. Stocco, S.E. Salcudean, F. Sassani, "Fast Constrained Global Minimax Optimization of Robot Parameters", *Robotica, International Journal of Information, Education and Research in Robotics and Artificial Intelligence*, v. 16, pp. 595-605, 1998.

[76] L. Stocco, S.E. Salcudean, F. Sassani, "Matrix Normalization for Optimal Robot Design", *Proceedings of IEEE International Conference on Robotics and Automation* (Leuven, Belgium), pp. 1346-1351, May 16-23, 1998.

[77] L. Stocco, S.E. Salcudean, F. Sassani, "Mechanism Design for Global Isotropy with Applications to Haptic Interfaces", *Proceedings of ASME Winter Annual Meeting* (Dallas, Texas), v. 61, pp. 115-122, Nov. 15-21, 1997.

[78] L.J. Stocco, S.E. Salcudean, F. Sassani, "On the Use of Scaling Matrices for Task-Specific Robot Design", *IEEE Transactions on Robotics and Automation*, v. 15, no. 5, pp. 958-965, Oct. 1999.

[79] B. Stuckman, E. Easom, "A Comparison of Bayesian/Sampling Global Optimization Techniques", *IEEE Transactions on Systems, Man and Cybernetics*, v. 22, no. 5, pp. 1024-1032, Sept./Oct. 1992.

[80] B. Stuckman, G. Evans, M. Mollaghasemi, "Comparison of Global Search Methods for Design Optimization Using Simulation", *Proceedings of 1991 IEEE Winter Simulation Conference* (Phoenix, Arizona), pp. 937-944, Dec. 8-11, 1991.

[81] S. Tafazoli, "Identification of Frictional Effects and Structural Dynamics for Improved Control of Hydraulic Manipulators", *Ph.D. Thesis, Department. of Electrical and Computer Engineering, University of British Columbia*, Jan. 1997.

[82] M. Tandirci, J. Angeles, F. Ranjbaran, "The Characterstic Point and the Characteristic Length of Robotic Manipulators", *Proceedings of ASME 22nd Biennial Conference on Robotics, Spatial Mechanisms & Mechanical Systems* (Scotsdale, Arizona), v. 45, pp. 203-208, Sept. 13-16, 1992.

[83] M. Teboulle, "Nonlinear Perturbations for Linear Semi-Infinite Optimization Problems", *Proceedings of 29th IEEE Conference on Decision and Control* (Honolulu, Hawaii), pp. 2477-2478, Dec. 5-7, 1990.

[84] U. Tränkle, D. Deutschmann, "Factors Influencing Speed and Precision of Cursor Positioning Using a Mouse", *Ergonomics*, v. 34, no. 2, pp. 161-174, 1991.

[85] J. Vertut, "Control of Master Slave Manipulators and Force Feedback", *Proceedings of 1977 Joint Automatic Control Conference*, 1977.

[86] T. Vlaar, "Mechanism Emulation with a Magnetically Levitated Input/Output Device", *Master's*

Thesis, University of British Columbia, Dec. 1994.

[87] W.E. Woodson, “Human Factors Design Handbook”, McGraw Hill, 1981.

[88] Y. Yao, “Dynamic Tunneling Algorithm for Global Optimization”, *IEEE Transactions on Systems, Man, and Cybernetics*, v. 19, no. 5, Sept./Oct. 1989.

[89] H. Yokoi, J. Yamashita, Y. Fukui, M. Shimojo, “Development of 3D-Input Device for Virtual Surface Manipulation”, *IEEE International Workshop on robot and Human Communication* (Nagoya, Japan), pp. 134-139, Jul. 18-20, 1994.

[90] T. Yoshikawa, “Manipulability of Robotic Mechanisms”, *The International Journal of Robotics Research*, v. 4, no. 2, pp.3-9, Summer 1985.

[91] T. Yoshikawa, K. Harada, A. Matsumoto, “Hybrid Position/Force Control of Flexible-Macro/Rigid-Micro Manipulator Systems”, *IEEE Transactions on Robotics and Automation*, v. 12, no. 4, pp. 633-640, Aug. 1996.

[92] K.E. Zanganeh, J. Angeles, “Kinematic Isotropy and the Optimum Design of Parallel Manipulators”, *The International Journal of Robotics Research*, v. 16, no. 2, pp.185-197, Apr. 1997.

[93] Y. Zheng, W.L. Lewis, “Several Algorithms of Global Optimal Search”, *Advances in Engineering Software*, v. 21, pp. 87-98, 1994.

Appendix A

Equations of Motion

Using existing methods, the equations of motion (direct kinematics, inverse kinematics and Jacobian) of any non-redundant serial robot can be computed analytically. The same, however, cannot be said for parallel or hybrid robots. While the inverse kinematics of parallel robots can be computed analytically, the direct kinematics and Jacobian must often be solved numerically. To obtain analytic functions for robots that contain a parallel component, they are "serialized" by computing passive joint angles and/or rates from the active joint angles and/or rates that completes a serial path from the base to the end-effector. Then existing serial methods can be used to complete the kinematic equations.

A.1 Direct Kinematics of a 3-DOF 5-Bar Linkage

Because the five-bar linkage is a parallel device, its forward kinematics are difficult to compute. The task is further complicated by the 3-DOF hybrid version shown in Figure A.1. It can, however, be simplified by considering a serial analogy (also shown in Figure A.1) of the device which treats all passive joints as active and has an end-effector that coincides with the axis of the eliminated actuator q_2. The forward kinematics of the serial analogy are easily solved using serial techniques.

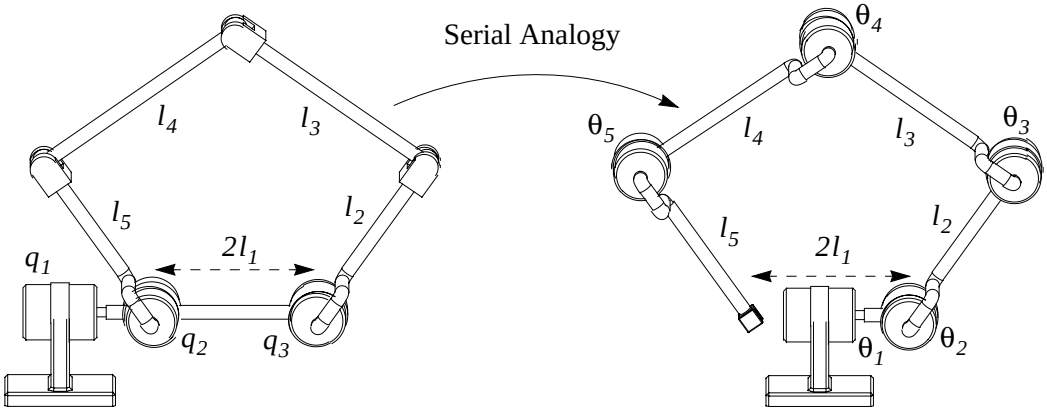


Figure A.1: Serial Analogy of a Five-Bar Linkage

All joint angles are set to zero and reference frames are assigned for the serial pseudo-manipulator in Figure A.2. Since the end effector of the 5-bar linkage is at the origin of frame 3, transformation matrices between the base to frame 3 are derived in equations (A.1) through (A.4).

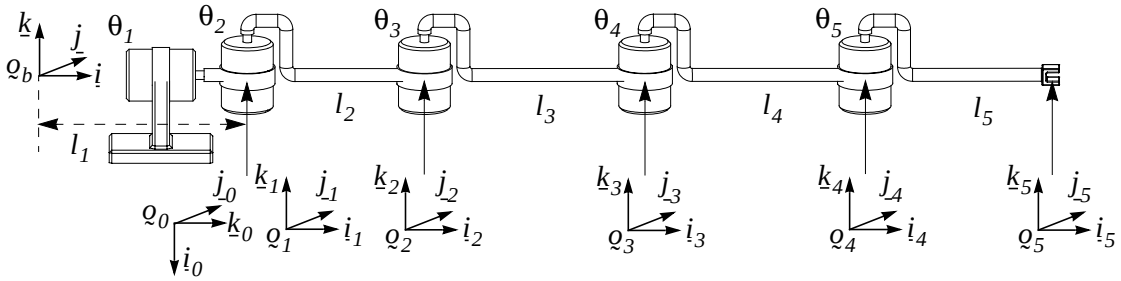


Figure A.2: Coordinate Frames of a Serial Pseudo-Manipulator

$$ {}^{b}T_0 = \begin{bmatrix} {}^{b}C_0 & l_1\hat{\underline{i}} \\ 0^T & 1 \end{bmatrix} \qquad {}^{b}C_0 = e^{\frac{\pi}{2}\underline{j}\times} \tag{A.1} $$

$$ {}^{0}T_1 = \begin{bmatrix} {}^{0}C_1 & 0 \\ 0^T & 1 \end{bmatrix} \qquad {}^{0}C_1 = e^{\theta_1\underline{k}\times}\ e^{-\frac{\pi}{2}\underline{j}\times} \tag{A.2} $$

$$ {}^{1}T_2 = \begin{bmatrix} {}^{1}C_2 & {}^{1}C_2 l_2\hat{\underline{i}} \\ 0^T & 1 \end{bmatrix} \qquad {}^{1}C_2 = e^{\theta_2\underline{k}\times} \tag{A.3} $$

$$ {}^{2}T_3 = \begin{bmatrix} {}^{2}C_3 & {}^{2}C_3 l_3\hat{\underline{i}} \\ 0^T & 1 \end{bmatrix} \qquad {}^{2}C_3 = e^{\theta_3\underline{k}\times} \tag{A.4} $$

Since θ_1 and θ_2 are known explicitly (A.5), (A.6), only θ_3 needs to be solved to complete the forward kinematics using (A.7).

$$ \theta_1 = q_1 \tag{A.5} $$

$$ \theta_2 = q_3 \tag{A.6} $$

$$ \begin{bmatrix} {}^{b}C_3 & {}^{b}\underline{o}_3 \\ 0^T & 1 \end{bmatrix} = {}^{b}T_3 = {}^{b}T_0\,{}^{0}T_1\,{}^{1}T_2\,{}^{2}T_3 \tag{A.7} $$

To solve θ_3 as a function of q_2 and q_3, the internal angles $\gamma_1 \rightarrow \gamma_8$ are defined in Figure A.3.

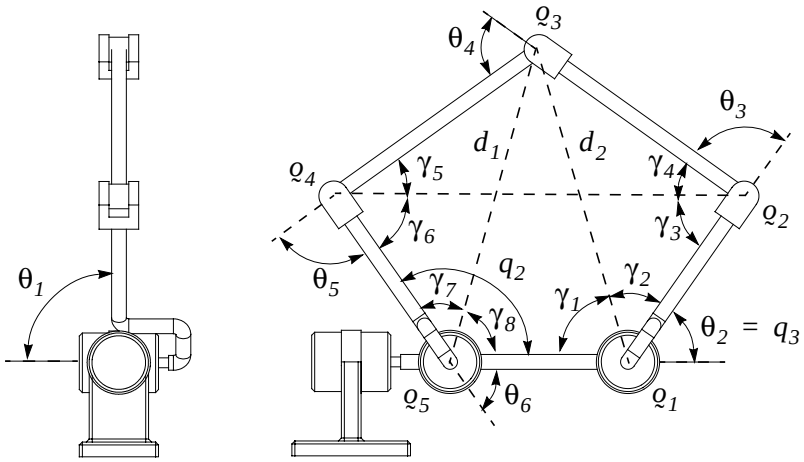


Figure A.3: Internal Angles of a 3-DOF 5-Bar Linkage

θ_3 is then easily computed from the geometry of the 5-bar linkage as shown in (A.8) - (A.12).

$$\theta_3 = \pi - \gamma_3 - \gamma_4 \tag{A.8}$$

$$\underline{\varrho}_{1,2} = \begin{bmatrix} l_2 cos(\theta_2) \\ l_2 sin(\theta_2) \end{bmatrix} \tag{A.9}$$

$$\underline{\varrho}_{4,2} = \underline{\varrho}_{1,2} + \begin{bmatrix} 2l_1 - l_5 cos(q_2) \\ -l_5 sin(q_2) \end{bmatrix} \tag{A.10}$$

$$\gamma_3 = 2\,atan\frac{\left\|\hat{\underline{\varrho}}_{4,2} - \hat{\underline{\varrho}}_{1,2}\right\|}{\left\|\hat{\underline{\varrho}}_{4,2} + \hat{\underline{\varrho}}_{1,2}\right\|} sgn\left(\theta_2 - atan2\left(\frac{\hat{\underline{\varrho}}_{4,2}{}^T \underline{j}}{\hat{\underline{\varrho}}_{4,2}{}^T \underline{i}}\right)\right) \tag{A.11}$$

$$\gamma_4 = \pi - 2\,atan\sqrt{\frac{(\|\underline{\varrho}_{4,2}\| + l_3)^2 - l_4^2}{l_4^2 - (\|\underline{\varrho}_{4,2}\| - l_3)^2}} \tag{A.12}$$

A.2 Inverse Kinematics of a 3-DOF 5-Bar Linkage

Because the 3-DOF 5-Bar linkage is a parallel device riding on a single serial actuator, its inverse kinematics are relatively easy to compute. Using $\underline{\varrho}_3$ (A.13) to derive distances d_1 (A.14) and d_2 (A.15), constants K_1 and K_2 are defined in (A.16) and (A.17) and plugged into equations (A.18) and (A.19) to solve for q_2 and q_3. Equations (A.18) and (A.19) are taken from [72] which describes a similar solution for the first joint angle of an elbow manipulator.

$$\underline{o}_3 = \begin{bmatrix} x_3 & y_3 & z_3 \end{bmatrix}^T \tag{A.13}$$

$$d_1 = \sqrt{(x_3 + l_1)^2 + {y_3}^2 + {z_3}^2} \tag{A.14}$$

$$d_2 = \sqrt{(x_3 - l_1)^2 + {y_3}^2 + {z_3}^2} \tag{A.15}$$

$$K_1 = \frac{d_1^2 - l_4^2 - l_5^2}{2 l_4 l_5} \tag{A.16}$$

$$K_2 = \frac{d_2^2 - l_2^2 - l_3^2}{2 l_2 l_3} \tag{A.17}$$

$$q_2 = atan2\left(\frac{\sqrt{{y_3}^2 + {z_3}^2}}{(x_3 + l_1)}\right) + atan2\frac{l_4 sin\left(atan2\frac{\sqrt{1 - K_1^2}}{K_1}\right)}{l_5 + l_4 cos\left(atan2\frac{\sqrt{1 - K_1^2}}{K_1}\right)} \tag{A.18}$$

$$q_3 = atan2\left(\frac{\sqrt{{y_3}^2 + {z_3}^2}}{(x_3 - l_1)}\right) - atan2\frac{l_3 sin\left(atan2\frac{\sqrt{1 - K_2^2}}{K_2}\right)}{l_2 + l_3 cos\left(atan2\frac{\sqrt{1 - K_2^2}}{K_2}\right)} \tag{A.19}$$

Then, q_1 is solved (A.21) from the projection of $\underline{o}_3 - \underline{o}_0$ onto the *x=0* plane (A.20).

$$\underline{v} = (\underline{j}\underline{j}^T + \underline{k}\underline{k}^T)(\underline{o}_3 - \underline{o}_0) \qquad \underline{j}\underline{j}^T + \underline{k}\underline{k}^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \equiv \textit{Projection onto x=0} \tag{A.20}$$

$$q_1 = 2 atan\frac{\left\|\hat{\underline{v}} - \underline{j}\right\|}{\left\|\hat{\underline{v}} + \underline{j}\right\|} sgn(\underline{o}_3^T \underline{k}) \tag{A.21}$$

A.3 Inverse Jacobian of a 3-DOF 5-Bar Linkage

The Jacobian matrix can also be computed using serial methods by converting the active joint rates ($\dot{q}_1 - \dot{q}_3$) into the joint rates ($\dot{\theta}_1 - \dot{\theta}_5$) of the pseudo-serial robot (Figure A.1). This results in a matrix that conforms to the definition of the Jacobian for a serial robot. Therefore, it corresponds to the inverse Jacobin in terms of parallel robot notation (A.22).

$$\dot{\underset{\sim}{\varrho}}_3 = J^{-1}\dot{q} \tag{A.22}$$

The Jacobian is computed in two parts. The first part is a transformation matrix *T* (A.23) that converts active joint rates into the joint rates of the first three joints of the pseudo-serial manipulator. The second part is the Jacobian matrix *J''* (A.26) of the first three joints of the pseudo-serial manipulator which is computed in (A.25) using common serial techniques.

$$\left[\dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3\right]^T = T\left[\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3\right]^T \tag{A.23}$$

$$\dot{\underset{\sim}{\varrho}}_3 = J^{-1}\left[\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3\right]^T = J''T\left[\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3\right]^T \tag{A.24}$$

$$J'' = \left[\underline{k}_0 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_0) \ \underline{k}_1 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_0) \ \underline{k}_1 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_2)\right] \tag{A.25}$$

The transformation matrix *T* (A.26) contains two functions, $V(q_2,q_3)$ and $V'(q_2,q_3)$ (A.27). *T* is easily inverted (A.28) as long as it has full rank (i.e. $V(q_2,q_3)\neq 0$).

$$\left[\dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3\right]^T = T\left[\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3\right]^T \tag{A.26}$$

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & V(q_2, q_3) & V'(q_2, q_3) \end{bmatrix} \tag{A.27}$$

$$T^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \dfrac{-V'(q_2, q_3)}{V(q_2, q_3)} & \dfrac{1}{V(q_2, q_3)} \\ 0 & 1 & 0 \end{bmatrix} \tag{A.28}$$

To calculate $V(q_2,q_3)$ and $V'(q_2,q_3)$ the 4×4 Jacobian matrix *J'* (A.34) for the redundant planar serial manipulator in Figure A.1 with θ_1 neglected ($\theta_1 = \dot{\theta}_1 = 0$) is computed in equation (A.35) where the positions $\underset{\sim}{\varrho}'_1$ through $\underset{\sim}{\varrho}'_5$ in the $\underline{ij}$ plane are computed using equations (A.29) through (A.33). Note that the inverse kinematics can be solved to obtain values for q_2 and q_3.

$$\underline{\varrho}'_1 = \underline{\varrho}_b + \begin{bmatrix} l_1 & 0 & 0 \end{bmatrix}^T \tag{A.29}$$

$$\underline{\varrho}'_2 = \underline{\varrho}'_1 + \begin{bmatrix} l_2 cos(q_3) & l_2 sin(q_3) & 0 \end{bmatrix}^T \tag{A.30}$$

$$\underline{\varrho}'_3 = \begin{bmatrix} \underline{i}^T \underline{\varrho}_3 & \| (\underline{j}\underline{j}^T + \underline{k}\underline{k}^T)\underline{\varrho}_3 \| & 0 \end{bmatrix}^T \tag{A.31}$$

$$\underline{\varrho}'_4 = \underline{\varrho}'_5 + \begin{bmatrix} l_5 cos(q_2) & l_5 sin(q_2) & 0 \end{bmatrix}^T \tag{A.32}$$

$$\underline{\varrho}'_5 = \underline{\varrho}_b - \begin{bmatrix} l_1 & 0 & 0 \end{bmatrix}^T \tag{A.33}$$

$$\begin{bmatrix} \dot{\underline{\varrho}}'_5 & \omega_5 \end{bmatrix}^T = J' \begin{bmatrix} \dot{\theta}_2 & \dot{\theta}_3 & \dot{\theta}_4 & \dot{\theta}_5 \end{bmatrix}^T \tag{A.34}$$

$$J' = \begin{bmatrix} \underline{k} \times (\underline{\varrho}'_5 - \underline{\varrho}'_1) & \underline{k} \times (\underline{\varrho}'_5 - \underline{\varrho}'_2) & \underline{k} \times (\underline{\varrho}'_5 - \underline{\varrho}'_3) & \underline{k} \times (\underline{\varrho}'_5 - \underline{\varrho}'_4) \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{A.35}$$

Applying the constraints $\dot{\underline{\varrho}}'_5 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, $\omega_5 = \dot{q}_2$, $\dot{\theta}_2 = \dot{q}_3$ and equation (A.36), which are intrinsic to the 5-Bar linkage, and expanding the remaining terms in (A.35) results in (A.37).

$$\underline{k} \times (\underline{\varrho}'_5 - \underline{\varrho}'_1) = \underline{k} \times (-2l_1 \underline{i}) = -2l_1 \underline{j} \tag{A.36}$$

$$\begin{bmatrix} 0 & -\underline{j}^T(\underline{\varrho}'_5 - \underline{\varrho}'_2) & -\underline{j}^T(\underline{\varrho}'_5 - \underline{\varrho}'_3) & -\underline{j}^T(\underline{\varrho}'_5 - \underline{\varrho}'_4) \\ -2l_1 & \underline{i}^T(\underline{\varrho}'_5 - \underline{\varrho}'_2) & \underline{i}^T(\underline{\varrho}'_5 - \underline{\varrho}'_3) & \underline{i}^T(\underline{\varrho}'_5 - \underline{\varrho}'_4) \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_3 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dot{q}_2 \end{bmatrix} \tag{A.37}$$

Equation (A.37) contains three linearly independent equations and three unknowns. Solving for $\dot{\theta}_3$ in terms of $\dot{q}_2$ and $\dot{q}_3$ (A.38) and rearranging it into the form (A.39) (equivalent to the third row of (A.27)) results in concise terms for $V(q_2,q_3)$ (A.40) and $V'(q_2,q_3)$ (A.41).

$$\dot{\theta}_3 = \frac{\underline{k}^T((\underline{\varrho}'_5 - \underline{\varrho}'_4) \times (\underline{\varrho}'_4 - \underline{\varrho}'_3))(\dot{q}_3 - \dot{q}_2) + 2l_1\underline{j}^T(\underline{\varrho}'_4 - \underline{\varrho}'_3)\dot{q}_3}{\underline{k}^T((\underline{\varrho}'_4 - \underline{\varrho}'_2) \times (\underline{\varrho}'_4 - \underline{\varrho}'_3))} \tag{A.38}$$

$$\dot{\theta}_3 = V(q_2, q_3)\dot{q}_2 + V'(q_2, q_3)\dot{q}_3 \tag{A.39}$$

$$V(q_2, q_3) = \frac{\underline{k}^T((\underline{\varrho}'_4 - \underline{\varrho}'_3) \times (\underline{\varrho}'_5 - \underline{\varrho}'_4))}{\underline{k}^T((\underline{\varrho}'_4 - \underline{\varrho}'_2) \times (\underline{\varrho}'_4 - \underline{\varrho}'_3))} \tag{A.40}$$

$$V'(q_2, q_3) = \frac{2l_1\underline{j}^T(\underline{\varrho}'_4 - \underline{\varrho}'_3)}{\underline{k}^T((\underline{\varrho}'_4 - \underline{\varrho}'_2) \times (\underline{\varrho}'_4 - \underline{\varrho}'_3))} - V(q_2, q_3) \tag{A.41}$$

A.4 Jacobian of a 2-DOF 5-Bar Linkage

The Jacobian of the 3-DOF 5-bar linkage is easily obtained by inverting the inverse Jacobian (A.22). It is, however, useful to have an analytic solution for the Jacobian of the 2-DOF 5-bar linkage which operates in the plane. Consider the 5-bar linkage in Figure A.3 where q_2 and q_3 are the only active joints (i.e. $q_1=0$). The Jacobian matrix is defined as in (A.42) with $\underline{\varrho}_3$ defined as in (A.43).

$$\begin{bmatrix}\dot{q}_2 & \dot{q}_3\end{bmatrix}^T = J\dot{\underline{\varrho}}_3 \tag{A.42}$$

$$\underline{\varrho}_3 = \begin{bmatrix}x_3 & y_3\end{bmatrix}^T \tag{A.43}$$

First, the inverse kinematics are solved to obtain values for q_2 and q_3. Then, by solving equations (A.46) and (A.47) by taking partial derivatives of equations (A.48) through (A.51), J (A.52) is obtained.

$$d_1 = \sqrt{(x_3 + l_1)^2 + y_3^2} \tag{A.44}$$

$$d_2 = \sqrt{(x_3 - l_1)^2 + y_3^2} \tag{A.45}$$

$$\dot{q}_2 = \dot{\gamma}_7 + \dot{\gamma}_8 \tag{A.46}$$

$$\dot{q}_3 = -\dot{\gamma}_1 - \dot{\gamma}_2 \tag{A.47}$$

$$\gamma_1 = atan\left(\frac{y_3}{l_1 - x_3}\right) \tag{A.48}$$

$$\gamma_2 = acos\left(\frac{l_2^2 - l_3^2 + d_2^2}{2 l_2 d_2}\right) \tag{A.49}$$

$$\gamma_7 = acos\left(\frac{l_5^2 - l_4^2 + d_1^2}{2 l_5 d_1}\right) \tag{A.50}$$

$$\gamma_8 = atan\left(\frac{y_3}{x_3 + l_1}\right) \tag{A.51}$$

$$J = \begin{bmatrix} \dfrac{-y_3}{d_1^2} + \dfrac{(l_5^2 - l_4^2 - d_1^2)(x_3 + l_1)}{d_1^2 \sqrt{4 l_5^2 d_1^2 - (l_5^2 - l_4^2 + d_1^2)^2}} & \dfrac{x_3 + l_1}{d_1^2} + \dfrac{y_3 (l_5^2 - l_4^2 - d_1^2)}{d_1^2 \sqrt{4 l_5^2 d_1^2 - (l_5^2 - l_4^2 + d_1^2)^2}} \\ \dfrac{-y_3}{d_2^2} - \dfrac{(l_2^2 - l_3^2 - d_2^2)(x_3 - l_1)}{d_2^2 \sqrt{4 l_2^2 d_2^2 - (l_2^2 - l_3^2 + d_2^2)^2}} & \dfrac{x_3 - l_1}{d_2^2} - \dfrac{y_3 (l_2^2 - l_3^2 - d_2^2)}{d_2^2 \sqrt{4 l_2^2 d_2^2 - (l_2^2 - l_3^2 + d_2^2)^2}} \end{bmatrix} \tag{A.52}$$

A.5 Mass Matrix of a 2-DOF 5-Bar Linkage

As described in (72), a robot's mass matrix can be interpretted as the position dependent transformation between end point velocity and kinetic energy (A.53).

$$KE = \frac{1}{2} \dot{o}_3^T D \dot{o}_3 \tag{A.53}$$

It is shown in (72) that the mass matrix *D'* for the planar 2-DOF serial elbow robot shown in Figure A.4 can be computed by (A.54) where *m* is the mass per unit length of the linkage bar stock and I_1 and I_2 are the longitudinal inertia of links l_1 and l_2 measured about axes perpendicular to the plane of motion and passing through the link centre. Note that (A.54) does not take into account actuator masses.

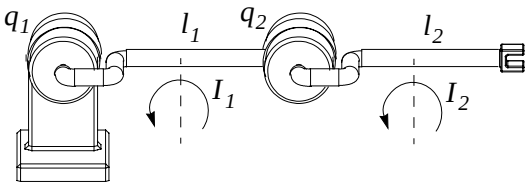


Figure A.4: 2-DOF Planar Serial Robot

$$D' = \begin{bmatrix} m\left(\frac{l_1^3 + l_2^3}{4} + l_1^2 l_2 + l_1 l_2^2 cos(q_2)\right) + I_1 + I_2 & m\left(\frac{l_2^3}{4} + \frac{l_1 l_2^2}{2} cos(q_2)\right) + I_2 \\ m\left(\frac{l_2^3}{4} + \frac{l_1 l_2^2}{2} cos(q_2)\right) + I_2 & \frac{m l_2^3}{4} + I_2 \end{bmatrix} \quad \text{(A.54)}$$

D' (A.54) can be used to compute the individual inertias of the left and right halves of a 5-bar linkage if the elbow joint rates are known. This requires a transformation *R* that computes joint rates $\dot{\theta}_3$ and $\dot{\theta}_5$ from $\dot{q}_2$ and $\dot{q}_3$ (from Figure A.3). Functions (A.40), (A.41) have already been derived for solving $\dot{\theta}_3$ in terms of q_2, q_3, $\dot{q}_2$ and $\dot{q}_3$. The same functions can be used to solve $\dot{\theta}_5$ by using the parameters from the mirror image of the 5-bar linkage. In other words, two new functions, *W* and *W'* are defined (A.55) which are the same as *V* and *V'* (A.39) except l_2 is interchanged with l_5, l_3 is interchanged with l_4, q_2 is replaced by π-q_3 and q_3 is replaced by π-q_2. Using *V, V', W* and *W', R* (A.56) can be computed from (A.57).

$$\dot{\theta}_5 = W'(\pi - q_3, \pi - q_2)\dot{q}_2 + W(\pi - q_3, \pi - q_2)\dot{q}_3 \quad \text{(A.55)}$$

$$\begin{bmatrix} \dot{\theta}_6 & \dot{\theta}_5 & \dot{\theta}_2 & \dot{\theta}_3 \end{bmatrix}^T = R \begin{bmatrix} \dot{q}_2 & \dot{q}_3 \end{bmatrix}^T \quad \text{(A.56)}$$

$$R = \begin{bmatrix} 1 & 0 \\ W'(\pi - q_3, \pi - q_2) & W(\pi - q_3, \pi - q_2) \\ 0 & 1 \\ V(q_2, q_3) & V'(q_2, q_3) \end{bmatrix} \quad \text{(A.57)}$$

The inertia matrix *D* is then computed from (A.58) where *D''* is *D'* with the 5-bar linkage geometric parameters l_5 and l_4 substituted for the serial parameters l_1 and l_2 respectively and internal angle θ_5 substituted for q_2 and *D'''* is *D'* with the 5-bar linkage geometric parameters l_2 and l_3 substituted for the serial parameters l_1 and l_2 respectively and internal angle θ_3 substituted for q_2.

$$D = J^T R^T \begin{bmatrix} D'' & 0 \\ 0 & D''' \end{bmatrix} RJ \tag{A.58}$$

The axial inertia I_A, longitudinal inertia I_L and mass per unit length m depend on the shape, dimensions and material used to make the links. For these, it is assumed that the links are constructed of tubing with a circular cross-section and wall thickness t. Geometric properties of the members are shown in Figure A.5.

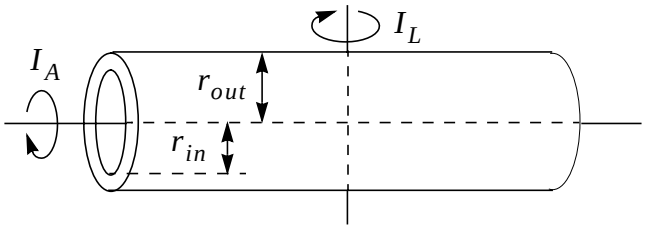


Figure A.5: Geometry and Inertia of a Linkage Member

The transverse inertia can then be computed from (A.59) where the average tube radius r is approximated by the mean value of r_{in} and r_{out} and l is the length of the tube.

$$I_L = \frac{mlr^2}{2} \tag{A.59}$$

The radius of the tube is fixed such that the first natural frequency of the links is bounded by ω. A conservative approach is used which considers the worst-case scenario of all four links (l_2 through l_5) rigidly connected in series to form a single cantilever beam (one end pinned, the other free). The equation for this is taken from [27] [58], shown in (A.60) and simplified by substituting in (A.61) where t is the thickiness of the tube wall (r_{out}-r_{in}).

$$\omega = \frac{3.52}{(l_2 + l_3 + l_4 + l_5)^2} \sqrt{\frac{EI_A}{A\rho}} = \frac{3.52r}{(l_2 + l_3 + l_4 + l_5)^2} \sqrt{\frac{E}{2\rho}} \tag{A.60}$$

$$\frac{I_A}{A} = \frac{\pi r^3 t}{2\pi r t} = \frac{r^2}{2} \tag{A.61}$$

Assuming 2024-T4 aluminum tubing is used, (A.62) is solved for r as a function of link length (l_2 through l_5) and natural frequency ω in (A.62).

$$For\ 2024\text{-}T4\ Aluminum: \quad \rho = 2.768\ (g/cm^3) \quad E = 2.758\times10^{11}\ (g/cms^2) \tag{A.62}$$

$$r = 1.27\times10^{-6}(l_2 + l_3 + l_4 + l_5)^2\omega \tag{A.63}$$

Mass per unit length m is then solved as a function of link length (l_2 through l_5), natural frequency ω and tube thickness t in (A.64).

$$m = \rho A = 2.768(2\pi rt) = 2.21\times10^{-5}(l_2 + l_3 + l_4 + l_5)^2\omega t\ (g/cm) \tag{A.64}$$

Equation (A.64) is substituted into (A.59) to compute the longitudinal inertias inside D" and D"' (A.58).

A.6 Kinematics of a 3-DOF Planar Parallel Manipulator

Due to their physical similarity, the kinematic equations of the 3-DOF planar parallel manipulator shown in Figure 4.2 can be derived using techniques similar to those used for solving the Stewart Platform. Using the origin and frame assignments shown in Figure A.6, the inverse kinematics are calculated from (A.65) and the Jacobian is calculated from (A.66).

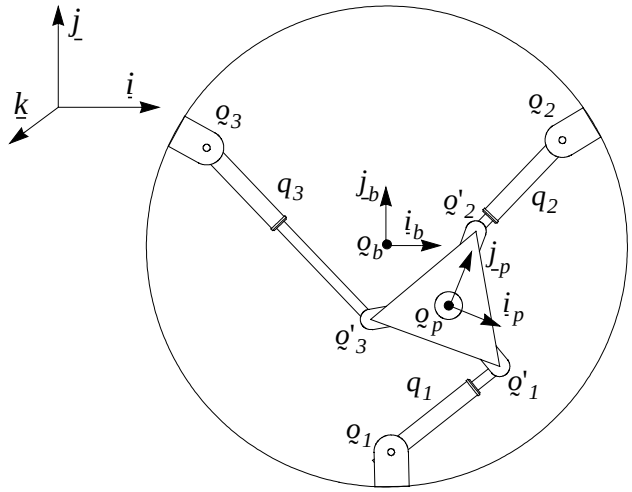


Figure A.6: Coordinate Frames of a Planar Parallel Manipulator

$$q_i = \left\| {}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_i + {}^bC_p\,{}^p\underset{\sim}{o}'_i \right\| \tag{A.65}$$

$$J = \begin{bmatrix} \frac{1}{q_1}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_1 + {}^bC_p\,{}^p\underset{\sim}{o}'_1)^T & \frac{1}{q_1}det\left[{}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_1 \quad {}^bC_p\,{}^p\underset{\sim}{o}'_1\right] \\ \frac{1}{q_2}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_2 + {}^bC_p\,{}^p\underset{\sim}{o}'_2)^T & \frac{1}{q_2}det\left[{}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_2 \quad {}^bC_p\,{}^p\underset{\sim}{o}'_2\right] \\ \frac{1}{q_3}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_3 + {}^bC_p\,{}^p\underset{\sim}{o}'_3)^T & \frac{1}{q_3}det\left[{}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_3 \quad {}^bC_p\,{}^p\underset{\sim}{o}'_3\right] \end{bmatrix} \tag{A.66}$$

A.7 Jacobian of a 2-DOF RR Serial Manipulator

The Jacobian of the planar serial manipulator in Figure 4.8 is computed in (A.67) using the origin and frame assignments shown in Figure A.7 where all joint angles are set to zero.

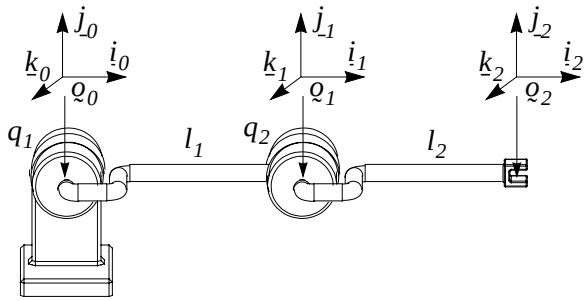


Figure A.7: Coordinate Frames of a 2-DOF Planar RR Serial Manipulator

$$J = \begin{bmatrix} \underset{\sim}{j}^T(\underset{\sim}{o}_0 - \underset{\sim}{o}_2) & \underset{\sim}{j}^T(\underset{\sim}{o}_1 - \underset{\sim}{o}_2) \\ \underset{\sim}{i}^T(\underset{\sim}{o}_2 - \underset{\sim}{o}_0) & \underset{\sim}{i}^T(\underset{\sim}{o}_2 - \underset{\sim}{o}_1) \end{bmatrix} \tag{A.67}$$

A.8 Simple Inertia Matrix of a 2-DOF Serial Manipulator

In [11], a mass matrix (A.68) is computed for the elbow manipulator in Figure A.7 where the total robot mass is approximated by two point masses m_1 and m_2 concentrated at points $\underset{\sim}{o}_1$ and $\underset{\sim}{o}_2$. Here, m_1 and m_2

are taken to be (A.69) and (A.70) where m is the mass per unit length of a linkage and m_q is the total mass of an actuator.

$$M = \begin{bmatrix} l_2^2 m_2 + 2 l_1 l_2 m_2 \cos(q_2) + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 \cos(q_2) \\ l_2^2 m_2 + l_1 l_2 m_2 \cos(q_2) & l_2^2 m_2 \end{bmatrix} \tag{A.68}$$

$$m_1 = m_q + m\frac{l_1 + l_2}{2} \tag{A.69}$$

$$m_2 = \frac{m l_2}{2} \tag{A.70}$$

The mass matrix (A.68) is describes the effective mass seen by the actuators (A.71). It is redefined in (A.72), (A.73) as the effective inertia D seen at the end-effector by assuming that the derivative of the Jacobian is small enough to be neglected.

$$\tau = M\ddot{q} \tag{A.71}$$

$$f = D\ddot{x} \tag{A.72}$$

$$D = J^{-T} M J^{-1} \tag{A.73}$$

A.9 Jacobian of a 3-DOF RRR Serial Manipulator

The Jacobian of the planar serial manipulator in Figure 4.5 is computed in (A.74) using the origin and frame assignments shown in Figure A.8 where all joint angles are set to zero.

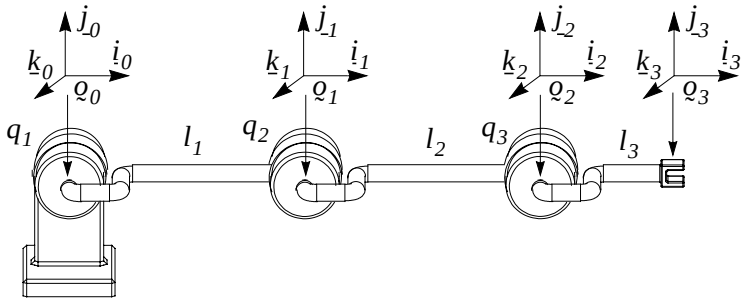


Figure A.8: Coordinate Frames of a 3-DOF Planar RRR Serial Manipulator

$$J = \begin{bmatrix} \underline{j}^T(\underline{o}_0 - \underline{o}_3) & \underline{j}^T(\underline{o}_1 - \underline{o}_3) & \underline{j}^T(\underline{o}_2 - \underline{o}_3) \\ \underline{i}^T(\underline{o}_3 - \underline{o}_0) & \underline{i}^T(\underline{o}_3 - \underline{o}_1) & \underline{i}^T(\underline{o}_3 - \underline{o}_2) \\ 1 & 1 & 1 \end{bmatrix} \tag{A.74}$$

A.10 Jacobian of a 3-DOF RPR Serial Manipulator

The Jacobian of the planar serial manipulator in Figure 4.9 is computed in (A.75) using the origin and frame assignments shown in Figure A.9 where all joint angles are set to zero.

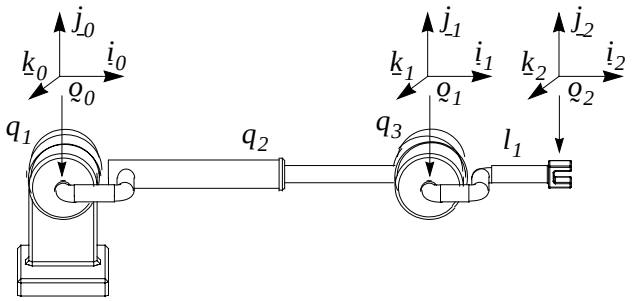


Figure A.9: Coordinate Frames of a 3-DOF Planar RPR Serial Manipulator

$$J = \begin{bmatrix} \underline{j}^T(\underline{o}_0 - \underline{o}_2) & \cos(q_1) & \underline{j}^T(\underline{o}_1 - \underline{o}_2) \\ \underline{i}^T(\underline{o}_2 - \underline{o}_0) & \sin(q_1) & \underline{i}^T(\underline{o}_2 - \underline{o}_1) \\ 1 & 0 & 1 \end{bmatrix} \tag{A.75}$$

A.11 Kinematics of the Stewart Platform

The method for computing the equations of motion for the Stewart Platform Figure 5.9 are well known. Using the origin assignments shown in Figure A.10, the inverse kinematics are shown in (A.76) and the Jacobian is shown in (A.77).

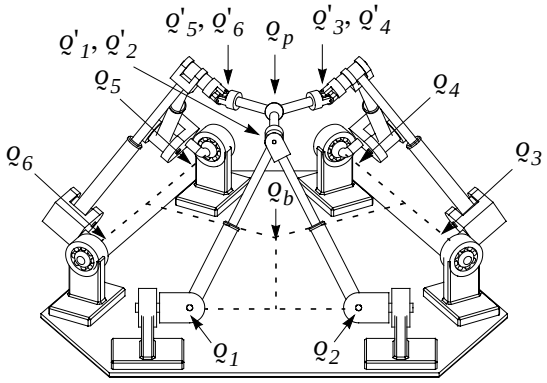


Figure A.10: Coordinate Origins of the Stewart Platform

$$q_i = \sqrt{\left\| {}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_i \right\|^2 + 2({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_i)^T {}^bC_p \, {}^p\underset{\sim}{o}'_i + \left\| {}^p\underset{\sim}{o}'_i \right\|^2} \tag{A.76}$$

$$J = \begin{bmatrix} \frac{1}{q_1}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_1 + {}^bC_p \, {}^p\underset{\sim}{o}'_1)^T & \frac{-1}{q_1}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_1) \times {}^bC_p \, {}^p\underset{\sim}{o}'_1)^T \\ \frac{1}{q_2}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_2 + {}^bC_p \, {}^p\underset{\sim}{o}'_2)^T & \frac{-1}{q_2}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_2) \times {}^bC_p \, {}^p\underset{\sim}{o}'_2)^T \\ \frac{1}{q_3}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_3 + {}^bC_p \, {}^p\underset{\sim}{o}'_3)^T & \frac{-1}{q_3}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_3) \times {}^bC_p \, {}^p\underset{\sim}{o}'_3)^T \\ \frac{1}{q_4}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_4 + {}^bC_p \, {}^p\underset{\sim}{o}'_4)^T & \frac{-1}{q_4}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_4) \times {}^bC_p \, {}^p\underset{\sim}{o}'_4)^T \\ \frac{1}{q_5}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_5 + {}^bC_p \, {}^p\underset{\sim}{o}'_5)^T & \frac{-1}{q_5}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_5) \times {}^bC_p \, {}^p\underset{\sim}{o}'_5)^T \\ \frac{1}{q_6}({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_6 + {}^bC_p \, {}^p\underset{\sim}{o}'_6)^T & \frac{-1}{q_6}(({}^b\underset{\sim}{o}_p - {}^b\underset{\sim}{o}_6) \times {}^bC_p \, {}^p\underset{\sim}{o}'_6)^T \end{bmatrix} \tag{A.77}$$

A.12 Kinematics of the Inoue Platform

The inverse kinematics of the Inoue Platform Figure 7.1 are performed by breaking the problem into three 3-DOF problems. First, the platform vertices $\underset{\sim}{o}'_1$, $\underset{\sim}{o}'_3$ and $\underset{\sim}{o}'_5$ shown in Figure A.11 are transformed into the points $\underset{\sim}{o}$, $\underset{\sim}{o}'$ and $\underset{\sim}{o}''$ using (A.78). The points $\underset{\sim}{o}$, $\underset{\sim}{o}'$ and $\underset{\sim}{o}''$ correspond to the three 3-DOF 5-bar linkage end-points for a device such as in Figure A.1.

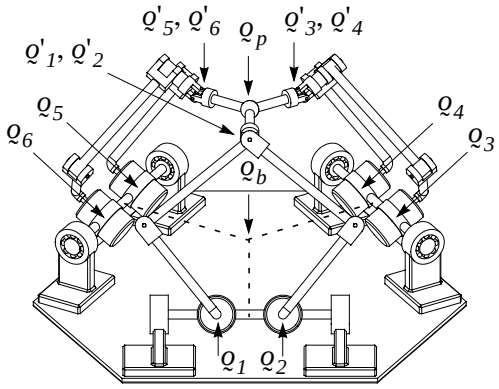


Figure A.11: Coordinate Origins of the Inoue Platform

$$\begin{bmatrix} \underset{\sim}{o} \\ \underset{\sim}{o}' \\ \underset{\sim}{o}'' \end{bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & e^{-\eta \underline{k}\times} & 0 \\ 0 & 0 & e^{\eta \underline{k}\times} \end{bmatrix} \begin{bmatrix} {}^b\underset{\sim}{o}_p + {}^bC_p\, {}^p\underset{\sim}{o}'_1 \\ {}^b\underset{\sim}{o}_p + {}^bC_p\, {}^p\underset{\sim}{o}'_3 \\ {}^b\underset{\sim}{o}_p + {}^bC_p\, {}^p\underset{\sim}{o}'_5 \end{bmatrix} + \begin{bmatrix} l_6\underline{j} \\ l_7\underline{j} \\ l_7\underline{j} \end{bmatrix} \tag{A.78}$$

Then, q_1 and q_2 are solved by substituting $\underset{\sim}{o}$ for $\underset{\sim}{o}_3$ in equations (A.18) and (A.19), q_3 and q_4 are solved by substituting $\underset{\sim}{o}'$ for $\underset{\sim}{o}_3$ in equations (A.18) and (A.19), and q_5 and q_6 are solved by substituting $\underset{\sim}{o}''$ for $\underset{\sim}{o}_3$ in equations (A.18) and (A.19).

The Jacobian of the Inoue Platform is solved in a similar fashion. First, the Jacobian of the Stewart Platform J_{Stew} (A.77) is computed to transform the linear ${}^b\dot{\underset{\sim}{o}}_p$ and angular ${}^b\underline{\omega}_p$ velocity of the platform into the rates of change of the dimensions d_1 and d_2 in Figure A.3. In equation (A.79), $\dot{d}_1$ and $\dot{d}_2$ correspond to the five-bar linkage comprising actuators q_1 and q_2, $\dot{d}'_1$ and $\dot{d}'_2$ correspond to the five-bar linkage comprising actuators q_3 and q_4, and $\dot{d}''_1$ and $\dot{d}''_2$ correspond to the five-bar linkage comprising actuators q_5 and q_6.

$$\begin{bmatrix} \dot{d}_1 & \dot{d}_2 & \dot{d}'_1 & \dot{d}'_2 & \dot{d}''_1 & \dot{d}''_2 \end{bmatrix}^T = J_{Stew} \begin{bmatrix} {}^b\dot{\underset{\sim}{o}}_p \\ {}^b\underline{\omega}_p \end{bmatrix} \tag{A.79}$$

Each $\dot{d}_1$, $\dot{d}_2$ pair is transformed into a corresponding planar velocity, $\dot{x}_{plane}$ and $\dot{y}_{plane}$, using the transformation T_1 (A.80) derived from the partial derivatives of (A.44) and (A.45) where *x*, *y*, and *z* are the components of $\underset{\sim}{o}$ and d_1 and d_2 are computed from (A.81) and (A.82).

$$\begin{bmatrix} \dot{x}_{plane} \\ \dot{y}_{plane} \end{bmatrix} = T_1 \begin{bmatrix} \dot{d}_1 \\ \dot{d}_2 \end{bmatrix} = \frac{1}{2l_1 y_{plane}} \begin{bmatrix} d_1 y_{plane} & -d_2 y_{plane} \\ d_1(l_1 - x_{plane}) & d_2(l_1 + x_{plane}) \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{d}_2 \end{bmatrix} \tag{A.80}$$

$$x_{plane} = x \tag{A.81}$$

$$y_{plane} = \sqrt{y^2 + z^2} \tag{A.82}$$

$$d_1 = \sqrt{(x_{plane} + l_1)^2 + y_{plane}^2} \tag{A.83}$$

$$d_2 = \sqrt{(x_{plane} - l_1)^2 + y_{plane}^2} \tag{A.84}$$

The planar velocity is then transformed into actuator velocities by substituting x_{plane} and y_{plane} for x_3 and y_3 into the equation for the Jacobian of a planar 5-bar linkage (A.52) to obtain transformation T_2 (A.85).

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = T_2 \begin{bmatrix} \dot{x}_{plane} \\ \dot{y}_{plane} \end{bmatrix} \tag{A.85}$$

Transformations T_1 (A.80) and T_2 (A.85) are combined to create transformation T_3 (A.86) which, when computed for all three 5-bar linkages (i.e. T_3, T_3' and T_3"), and combined with the Jacobian of the Stewart Platform (A.79) produces the Jacobian of the Inoue Platform (A.87).

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = T_2 T_1 \begin{bmatrix} \dot{d}_1 \\ \dot{d}_2 \end{bmatrix} = T_3 \begin{bmatrix} \dot{d}_1 \\ \dot{d}_2 \end{bmatrix} \tag{A.86}$$

$$J = \begin{bmatrix} T_3 & 0 & 0 \\ 0 & T_3' & 0 \\ 0 & 0 & T_3'' \end{bmatrix} J_{Stew} \tag{A.87}$$

A.13 Direct Kinematics of the Twin-Pantograph

To obtain analytic functions for the Twin-Pantograph Figure 7.2, it is divided into two halves which are "serialized" as shown in Figure A.12. The two serial robots have active joints labeled θ (A.89) and θ' (A.90). By using the angles and rates of the Twin-Pantograph's active joints q (A.88) which are defined in the range $-\pi \le \theta \le \pi$ to compute the angles and rates of the pseudo-active joints, serial techniques are again used to solve the kinematic equations.

$$q = \left[q_1\ q_2\ q_3\ q_4\ q_5\ q_6\ q_7\right]^T \tag{A.88}$$

$$\theta = \left[\theta_1\ \theta_2\ \theta_3\ \theta_4\ \theta_5\ \theta_6\right]^T \tag{A.89}$$

$$\theta' = \left[\theta'_1\ \theta'_2\ \theta'_3\right]^T \tag{A.90}$$

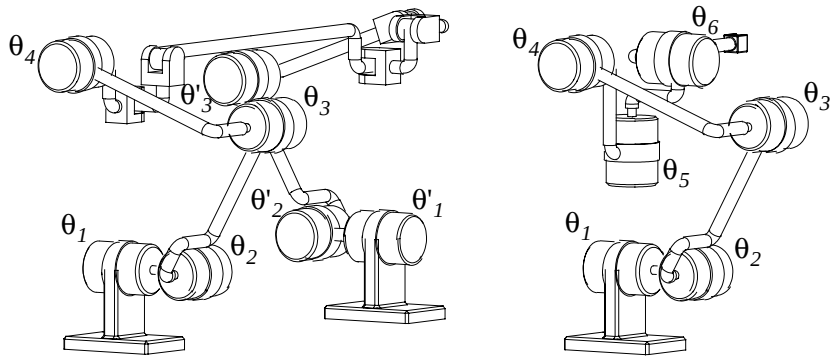


Figure A.12: Twin-Pantograph Schematic Diagram

The pseudo-active serial robots are shown with all joint angles set to zero and with reference frames in Figure A.13.

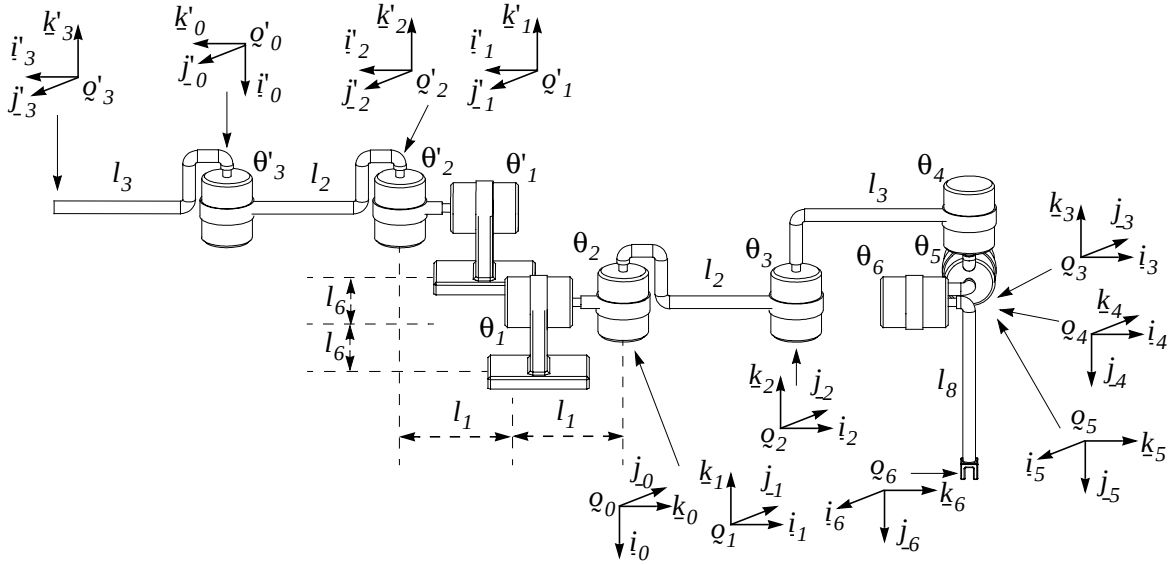


Figure A.13: Coordinate Frames of the Pseudo-Active Serial Robots

Using these frames, the homogeneous transformation matrices are defined in equations (A.91) through (A.95) where $T_{5\text{-}Bar}$ is the homogeneous transformation matrix of a 5-Bar Linkage defined in (A.7) which is computed in (A.1) through (A.12). In equation (A.92), ${}^{b}T'_3$ is obtained by substituting q_4 through q_6 for

q_1 through q_3 when computing $T_{5\text{-}Bar}$. The direct kinematics are solved by the transformation matrix ${}^{b}T_3$ (A.96).

$$ {}^{b}T_3 = \begin{bmatrix} I & -l_6\underline{j} \\ 0^T & 1 \end{bmatrix} T_{5\text{-}Bar} \tag{A.91} $$

$$ {}^{b}T'_3 = \begin{bmatrix} C' & 0 \\ 0^T & 1 \end{bmatrix} {}^{b}T_3 \qquad C' = e^{\pi\underline{k}\times} \tag{A.92} $$

$$ {}^{3}T_4 = \begin{bmatrix} {}^{3}C_4 & 0 \\ 0^T & 1 \end{bmatrix} \qquad {}^{3}C_4 = e^{\theta_4\underline{k}\times} e^{-\frac{\pi}{2}\underline{i}\times} \tag{A.93} $$

$$ {}^{4}T_5 = \begin{bmatrix} {}^{4}C_5 & 0 \\ 0^T & 1 \end{bmatrix} \qquad {}^{4}C_5 = e^{\theta_5\underline{k}\times} e^{\frac{\pi}{2}\underline{j}\times} \tag{A.94} $$

$$ {}^{5}T_6 = \begin{bmatrix} {}^{5}C_6 & {}^{5}C_6 l_8\underline{j} \\ 0^T & 1 \end{bmatrix} \qquad {}^{5}C_6 = e^{\theta_6\underline{k}\times} \tag{A.95} $$

$$ {}^{b}T_6 = {}^{b}T_3\,{}^{3}T_4\,{}^{4}T_5\,{}^{5}T_6 = \begin{bmatrix} {}^{b}C_6 & {}^{b}\underset{\sim}{o}_6 \\ 0^T & 1 \end{bmatrix} \tag{A.96} $$

Since $\theta_4=q_7$, only θ_5 and θ_6 must be derived. This is done by computing the platform end points ${}^{b}\underset{\sim}{o}_4$ and ${}^{b}\underset{\sim}{o}'_3$ using transformation matrices (A.97) and (A.98) so that the platform axis $\underline{j}_6$ can be found from (A.99).

$$ \begin{bmatrix} {}^{b}C_4 & {}^{b}\underset{\sim}{o}_4 \\ 0^T & 1 \end{bmatrix} = {}^{b}T_4 = {}^{b}T_3\,{}^{3}T_4 \tag{A.97} $$

$$ \begin{bmatrix} {}^{b}C'_3 & {}^{b}\underset{\sim}{o}'_3 \\ 0^T & 1 \end{bmatrix} = {}^{b}T'_3 \tag{A.98} $$

$$ \underline{j}_6 = \frac{\underset{\sim}{o}'_3 - \underset{\sim}{o}_4}{\left\| \underset{\sim}{o}'_3 - \underset{\sim}{o}_4 \right\|} \tag{A.99} $$

Joint angles θ_5 and θ_6 are then be calculated by performing the well known inverse kinematics of the spherical wrist (A.101) through (A.104). Note that due to the singularities of a spherical wrist, θ_5 and θ_6 can only be computed reliably when the angle between $\underline{j}_4$ and $\underline{j}_6$ is less than 90° (A.100).

$$\underline{j}_4^T \underline{j}_6 > 0 \tag{A.100}$$

$$\underline{k}_5 = \frac{\underline{j}_6 \times \underline{k}_4}{\left\| \underline{j}_6 \times \underline{k}_4 \right\|} \tag{A.101}$$

$$\theta_5 = 2\,atan \frac{\left\| \underline{k}_5 - \underline{i}_4 \right\|}{\left\| \underline{k}_5 + \underline{i}_4 \right\|} (sgn(\underline{k}_5^T \underline{j}_4)) \tag{A.102}$$

Once θ_5 has been determined, θ_6 is computed using (A.104) by obtaining $\underline{i}_5$ and $\underline{j}_5$ from ${}^bT_5$ (A.103).

$$\begin{bmatrix} {}^bC_5 & {}^b\underset{\sim}{o}_5 \\ 0^T & 1 \end{bmatrix} = {}^bT_4\,{}^4T_5 \tag{A.103}$$

$$\theta_6 = -2\,atan \frac{\left\| \underline{j}_6 - \underline{j}_5 \right\|}{\left\| \underline{j}_6 + \underline{j}_5 \right\|} sgn(\underline{j}_6^T \underline{i}_5) \tag{A.104}$$

With angles θ_5 and θ_6 known, the forward kinematics is computed using equation (A.96).

A.14 Inverse Kinematics of the Twin-Pantograph

The inverse kinematics are solved by first computing the end points $\underset{\sim}{o}_3$ and $\underset{\sim}{o}'_3$ of the five-bar linkages from (A.105) and (A.106).

$$\underset{\sim}{o}_3 = \underset{\sim}{o}_6 - l_8 C_6 \underline{j} + l_6 \underline{j} \tag{A.105}$$

$$\underset{\sim}{o}'_3 = e^{-\pi \underline{k} \times} \; (\underset{\sim}{o}_6 + l_8 C_6 \underline{j}) + l_6 \underline{j} \tag{A.106}$$

Joint angles q_1 through q_3 are computed from the inverse kinematics of a 5-Bar linkage, (A.13) through (A.22), and similarly for q_4 through q_6 by substituting $\underset{\sim}{o}'_3$ for $\underset{\sim}{o}_3$ in (A.13). With joint angles θ_1 through θ_3 known, transformation matrix ${}^bT_3$ (A.91) is computed from the forward kinematics and using the method for solving the inverse kinematics of a spherical wrist, q_7 is obtained from (A.107) and (A.108).

$$\hat{\underline{k}}_4 = \frac{\underline{k}_3 \times \underline{k}_6}{\left\| \underline{k}_3 \times \underline{k}_6 \right\|} \tag{A.107}$$

$$q_7 = -2\,atan \frac{\left\| \underline{j}_3 - \underline{k}_4 \right\|}{\left\| \underline{j}_3 + \underline{k}_4 \right\|} (sgn(\underline{i}_3^T \underline{k}_4)) \tag{A.108}$$

A.15 Jacobian of the Twin-Pantograph

The Jacobian transforms the linear and angular velocity of the end effector into the joint rates. This is done in a number of steps using the serial analogy described earlier. First the linear velocities of the two 5-bar Linkage end-points $\dot{\underset{\sim}{\varrho}}_3$ and $\dot{\underset{\sim}{\varrho}}'_3$ are computed using transformation T_1 (A.109), (A.110).

$$\begin{bmatrix} \dot{\underset{\sim}{\varrho}}_3 \\ \dot{\underset{\sim}{\varrho}}'_3 \\ \underline{\omega}_6 \end{bmatrix} = T_1 \begin{bmatrix} \dot{\underset{\sim}{\varrho}}_6 \\ \underline{\omega}_6 \end{bmatrix} \tag{A.109}$$

$$T_1 = \begin{bmatrix} I & l_8 \underline{j}_6 \times \\ I & -l_8 \underline{j}_6 \times \\ 0 & I \end{bmatrix} \tag{A.110}$$

Next, the 5-Bar Linkage end-point velocities are transformed into joint rates of spatial 3-DOF elbow robots using transformation T_2 described in (A.111) through (A.114). Note that T_3 (A.113) and T'_3 (A.114) are 3×3 serial Jacobian matrices for 3-DOF elbow robots and are easily inverted provided they are non-singular.

$$\begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 & \dot{\theta}'_1 & \dot{\theta}'_2 & \dot{\theta}'_3 & \underline{\omega}_6 \end{bmatrix}^T = T_2 \begin{bmatrix} \dot{\underset{\sim}{\varrho}}_3 & \dot{\underset{\sim}{\varrho}}'_3 & \underline{\omega}_6 \end{bmatrix}^T \tag{A.111}$$

$$T_2 = \begin{bmatrix} T_3 & 0 & 0 \\ 0 & T'_3 & 0 \\ 0 & 0 & I \end{bmatrix} \tag{A.112}$$

$$T_3 = \begin{bmatrix} \underline{k}_0 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_0) & \underline{k}_3 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_0) & \underline{k}_3 \times (\underset{\sim}{\varrho}_3 - \underset{\sim}{\varrho}_2) \end{bmatrix}^{-1} \tag{A.113}$$

$$T'_3 = \begin{bmatrix} \underline{k}'_0 \times (\underset{\sim}{\varrho}'_3 - \underset{\sim}{\varrho}'_0) & \underline{k}'_3 \times (\underset{\sim}{\varrho}'_3 - \underset{\sim}{\varrho}'_0) & \underline{k}'_3 \times (\underset{\sim}{\varrho}'_3 - \underset{\sim}{\varrho}'_2) \end{bmatrix}^{-1} \tag{A.114}$$

Next, the transformation matrix T_4 (A.115), (A.116) transforms the end-effector rotation rate from a rotation relative to frame C_0 to a rotation relative to frame C_3. This is accomplished using transformation T_5 (A.117) which computes the rotation matrix ${}^0\underline{\omega}_{3,0}$ of frame C_3 with respect to C_0 in C_0 as shown in (A.118).

$$\left[\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3\ \dot{\theta}'_1\ \dot{\theta}'_2\ \dot{\theta}'_3\ {}^3\underline{\omega}_{6,3}\right]^T = T_4\left[\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3\ \dot{\theta}'_1\ \dot{\theta}'_2\ \dot{\theta}'_3\ \underline{\omega}_6\right]^T \tag{A.115}$$

$$T_4 = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -T_5 & 0 & I \end{bmatrix} \tag{A.116}$$

$$T_5 = \left[\underline{k}_0\ \underline{k}_3\ \underline{k}_3\right] \tag{A.117}$$

$${}^3\underline{\omega}_{6,3} = {}^0\underline{\omega}_{6,0} - {}^0\underline{\omega}_{3,0} = \underline{\omega}_6 - T_5\left[\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3\right]^T \tag{A.118}$$

Next, the robot joint rates are computed from the serial elbow joint rates and the rotation vector ${}^0\underline{\omega}_{3,0}$ using transformation T_6 (A.119), (A.120). This is accomplished using transformations T_7 and T'_7 which are the transformations (A.28) defined as part of the inverse Jacobian matrix of a 5-Bar linkage. Transformation T_8 (A.121) is also used which is the inverse Jacobian of a spherical wrist.

$$\left[\dot{q}_1\ \dot{q}_2\ \dot{q}_3\ \dot{q}_4\ \dot{q}_5\ \dot{q}_6\ \dot{q}_7\right]^T = T_6\left[\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3\ \dot{\theta}'_1\ \dot{\theta}'_2\ \dot{\theta}'_3\ {}^3\underline{\omega}_{6,3}\right]^T \tag{A.119}$$

$$T_6 = \begin{bmatrix} T_7 & 0 & 0 \\ 0 & T'_7 & 0 \\ 0 & 0 & \underline{i}^T T_8 \end{bmatrix} \tag{A.120}$$

$$T_8 = \left[\underline{k}_3\ \underline{k}_4\ \underline{k}_5\right]^{-1} \tag{A.121}$$

$${}^3\underline{\omega}_{6,3} = \left[\underline{k}_3\ \underline{k}_4\ \underline{k}_5\right]\left[\dot{\theta}_4\ \dot{\theta}_5\ \dot{\theta}_6\right]^T \tag{A.122}$$

$$\dot{\theta}_4 = \underline{i}^T T_8\, {}^3\underline{\omega}_{6,3} \tag{A.123}$$

The Jacobian matrix is as shown in (A.124) and can be expanded as shown in (A.125).

$$J = T_6 T_4 T_2 T_1 \tag{A.124}$$

$$J = \begin{bmatrix} T_7 T_3 & T_7 T_3\ l_8 \underline{j}_6 \times \\ T'_7 T'_3 & -T'_7 T'_3\ l_8 \underline{j}_6 \times \\ -\underline{i}^T T_8 T_5 T_3 & \underline{i}^T T_8 (I - T_5 T_3) \end{bmatrix} \tag{A.125}$$

Appendix B

Workspace Discretization

B.1 Uniform Angular Sampling

A robot may navigate up to six degrees-of-freedom, three in translation and three in rotation. Any configuration belonging to a 6-DOF workspace can be specified by a position and an orientation vector and the extent of the workspace can be specified by a translational volume and a solid angle defined by maximum roll θ_0 and tip θ_1 angles as shown in Figure B.1.

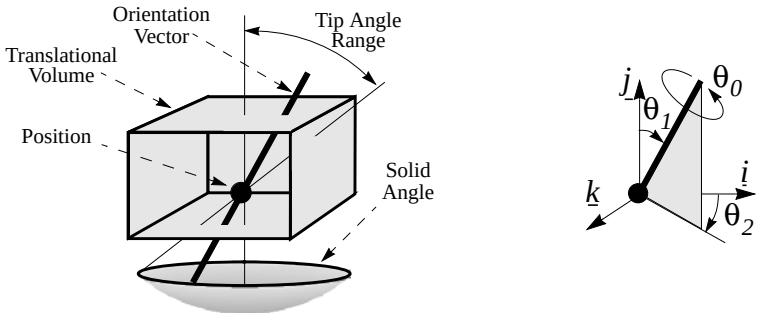


Figure B.1: 6-DOF Workspace

Defining and uniformly sampling a translational workspace is easy, particularly if the boundary is rectangular. It is, however, considerably more difficult to do this for a rotational workspace. Euler angles are not well suited to defining the boundary and result in non-uniform sample spacing [43] if they are discretized uniformly (see Figure B.3a). An axis-angle representation *R* shown in (B.1) which contains roll θ_0, tip θ_1 and sweep θ_2 angles is much better suited to defining the boundary. Maximum roll and tip values are assigned while sweep is unconstrained ($0 \leq \theta_2 < 2\pi$).

$$R = (\theta_0, \theta_1, \theta_2) \tag{B.1}$$

Unfortunately, sample spacing is still non-uniform if angles are discretized uniformly (see Figure B.3b). To homogenize sample spacing, θ_1 is discretized uniformly (i.e. constant $\Delta\theta_1$) but the sample spacing of θ_2

($\Delta\theta_2$) changes for each of the N θ_1 samples. The value of $\Delta\theta_2$ is computed in (B.2) for each θ_1 value. The terms in equation (B.2) are represented graphically in Figure B.2.

$$\Delta\theta_{2,n} = \frac{\Delta\theta_1}{\sin(n\Delta\theta_1)};\ n \in \{1, 2, 3, \ldots, N\} \tag{B.2}$$

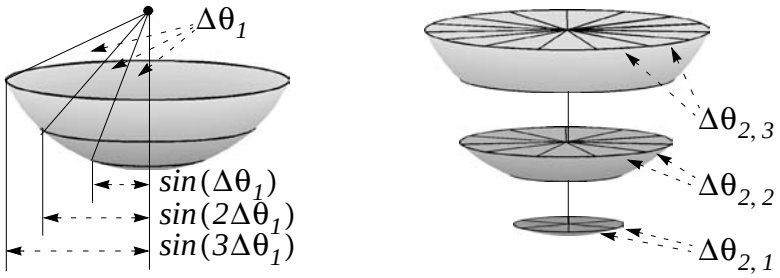


Figure B.2: Uniform Sampling Method

The three discretization methods are compared in Figure B.3 using a θ_1 of 45°. For each sample, a point is plotted on the solid angle defining the boundary of the angular workspace. Notice the workspace boundaries resulting from Euler angle and axis-angle methods as well as the uniform distribution resulting from non-uniform θ_2 sampling.

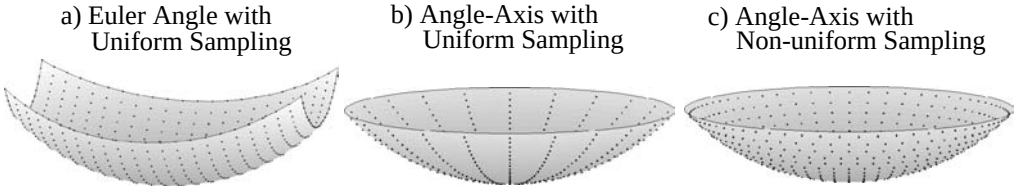


Figure B.3: Example Angular Discretizations

B.2 Selecting Sampling Resolutions

To ensure a fair split between translational (k_t) and rotational (k_r) samples, their geometric means are made the same by computing k_r from (B.3) where k_i, k_j and k_k are the number of samples along each translational axis and d_t and d_r are the number of axes in the translational and rotational workspaces.

$$k_r = (k_i k_j k_k)^{d_r/d_t} \tag{B.3}$$

N is then chosen such that K_r from (B.4) is as close as possible to k_r from (B.3) where $\bar{\theta}_1$ is the maximum tip angle defining the angular workspace boundary. Since the distance between θ_1 samples determines the distance between θ_2 samples and there must be an integer number N of θ_1 samples, any other number of total angular workspace samples is unachievable using the uniform sampling method described above. For example, it is found from (B.4) that $K_r=\{6,8,34,58,...\}$ for $N=\{1,2,3,4,...\}$ when $\bar{\theta}_1 = 45°$.

$$K_r = 1 + \sum_{n=1}^{N} floor\left(\frac{2\pi N}{\bar{\theta}_1} sin\left(\frac{n\bar{\theta}_1}{N}\right)\right) \tag{B.4}$$