

Tool-Based Haptic Interaction with Dynamic Physical Simulations
using Lorentz Magnetic Levitation

by

Peter John Berkelman

A Dissertation Presented to the
FACULTY OF THE Robotics Ph.D. Program
CARNEGIE MELLON UNIVERSITY

In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Robotics)

June 1999

Copyright 1999 Peter John Berkelman

Acknowledgements

The years I've been in this program have gone by faster than I could have imagined. I feel lucky and grateful to have had the opportunity to be a part of the Robotics Institute, the Microdynamic Systems Laboratory, and this maglev haptic interaction project. Though there were some stressful and frustrating times along the way, there was never any doubt that I was in the ideal environment for me to study and do research work towards a doctorate.

The Robotics Institute Ph.D. Program treats students better than any graduate program I know. The Microdynamic Systems Laboratory has been an excellent resource of equipment, people and ideas. The whole robotics community at CMU is full of stunningly smart, creative, experienced, and helpful people. The ones most responsible for helping me and this project to get to this point are listed below.

My advisor Ralph Hollis developed the first Lorentz levitation devices with Tim Salcudean and proposed the use of Lorentz levitation for haptic interaction. David Baraff developed the physically-based simulation system used in this project. Al Rizzi and Yasuyoshi Yokokohji provided additional technical advice in the lab.

Zack Butler performed early characterization of the large position sensing photodiodes and designed the sensor circuitry and assemblies. Stella Yu formulated the iterative kinematics solution method used. Our 1996 summer student, Chris Donohue, worked on selection, acquisition and assembly of the enclosure, VMEbus, and power supplies. Todd Okimoto assisted with assembly and testing of the prototype Lorentz force actuator.

This project was supported by National Science Foundation Grant No. IRI-9420869.

Contents

Acknowledgements	ii
List Of Tables	ix
List Of Figures	x
Abstract	xiii
1 Introduction	1
1.1 Applications of Haptic Interfaces	2
1.1.1 CAD Simulations	3
1.1.2 Medical Simulations	3
1.1.3 Entertainment	3
1.1.4 Teleoperation	3
1.1.5 Other Applications	4
1.2 Haptic Interaction Issues	4
1.2.1 Degrees of Freedom	5
1.2.2 Haptic Environment Emulation	6
1.3 Lorentz Magnetic Levitation	7
1.4 Outline	7
I Magnetic Levitation Haptic Interface Device	9
2 CMU Maglev Haptic Device Design	10
2.1 Design Goals	10
2.2 Configuration	12
2.2.1 Flotor and Stator	12
2.2.2 Handles	15
2.3 Cable and Linkage Haptic Interface Devices	17
2.4 Other Magnetic Levitation Haptic Devices	19
2.4.1 Magic Wrist	20
2.4.2 UBC Wrist	20
2.4.3 UBC Powermouse	20
3 Actuation	23

3.1	Actuator Configuration	24
3.1.1	Hemisphere Sizes	25
3.1.2	Outer Shell Thickness	27
3.1.3	Actuator Configuration Considerations	28
3.1.4	New Selected Actuator Configuration	32
3.2	Single Actuator	35
3.2.1	Coils	35
3.2.2	Finite Element Analysis Models	36
3.2.3	Magnet Assembly Design	37
3.2.4	FEA Modeling Results	39
3.2.5	Design Verification	40
3.3	Comments	42
4	Position Sensing	45
4.1	Sensor Assemblies	45
4.2	Sensor Calibration	48
4.2.1	Calibration Lookup Table	48
4.2.2	Lookup Table Interpolation	50
4.2.3	Lookup Table Accuracy	51
4.3	Sensing Kinematics	51
4.3.1	Inverse Kinematics	53
4.3.2	Forward Kinematics	54
4.3.3	Inverse Jacobian Lookup Table Method	54
4.3.4	Reduced Order Beam Length Method	55
4.4	Comments	59
5	Control	60
5.1	Control Hardware	60
5.1.1	Control Processors	61
5.1.2	Analog I/O	62
5.1.3	Current Amplifiers	62
5.1.4	Signal Filters	62
5.2	Control Software	62
5.2.1	Digital Filtering	62
5.2.2	Spatial Rotation Conversions	63
5.2.3	PD Control	64
5.2.4	Motion Bounds	64
5.2.5	Multiprocessing	65
5.2.6	Levitation Liftoff and Landing	65
5.2.7	Environment Emulation	66
6	Fabrication	67
6.1	Flotor	67
6.2	Stator	69
6.3	Enclosure	70
6.4	Comments	71

7	Performance Testing and Results	73
7.1	Performance Limits	73
7.1.1	Maximum Control Rate	73
7.1.2	Maximum Forces and Torques	74
7.1.3	Flotor Inertia	75
7.1.4	Power Consumption	75
7.1.5	Sensor Noise and Position Resolution	76
7.1.6	Stiffness Range	78
7.2	Response Bandwidths	78
7.2.1	Test Setup	78
7.2.2	Amplifier Current and Force Bandwidths	79
7.2.3	Position Bandwidths	79
7.3	Discussion	81
II	Interactive Simulations	87
8	Haptic Simulation and Display	88
8.1	Modularity	89
8.2	Simulation and Device Correspondence	90
8.3	Intermediate Representation	91
8.4	Virtual Coupling	91
8.5	Contact Point Intermediate Representation	92
8.6	Simulation and Controller Integration Summary	93
9	Modeling and Simulation for Haptic Interaction	95
9.1	Physical Dynamics Simulation	96
9.1.1	Dynamic Simulation Methods	96
9.1.2	Dynamic Simulation for Haptic Interaction	97
9.2	Friction	98
9.2.1	Viscosity	98
9.2.2	Coulomb Friction	99
9.3	Texture	99
9.3.1	Texture from Facets	100
9.3.2	Fine Generated Texture	101
9.3.3	Texture Issues	101
10	Local Simulations	103
10.1	Basic Rigid-Body Simulations	103
10.1.1	Interface and Control Method	104
10.1.2	Cube-In-Cube	105
10.1.3	Peg-in-Hole	108
10.2	Surface Characteristics	109
10.2.1	Coulomb Friction	109
10.2.2	Texture	110
10.2.3	Impact	111

10.3	Experimental Results	111
10.3.1	Cubes	111
10.3.2	Surface Characteristics	113
10.4	Discussion	115
11	Haptic User Interface	116
11.1	User Interface Control Panel	116
11.2	Graphical Display	117
11.3	Mouse Interface	119
11.4	Viewpoint Motion	119
11.5	Simulation	120
11.5.1	Integration Method	120
11.5.2	Motion Recovery	120
11.5.3	Print Operating Information	120
11.5.4	Other Simulation Parameters	121
11.6	Tool	121
11.6.1	Shape	121
11.6.2	Actions	123
11.6.3	Material Properties	123
11.7	Virtual Coupling Parameters	123
11.8	Control Modes	123
11.8.1	Rate Control Mode	124
11.8.2	Automatic View Control	124
11.8.3	Tool Coordinate Mode	124
11.9	Mapping Motion between Device and Simulation	124
11.10	Comments	125
11.10.1	Interfaces for Different Users	126
11.10.2	Interface Controls on Handle	126
12	Virtual Coupling	128
12.1	Physical Simulation	128
12.2	Coupled Simulation and Control	129
12.2.1	Sample Tasks	131
12.3	Sample Experimental Results	131
12.3.1	Tool Colliding and Sliding on Floor	132
12.3.2	Peg in Hole Insertion	133
12.4	Summary and Conclusions	138
13	Contact Point Intermediate Representation	140
13.1	Configuration	141
13.1.1	CORIOLIS Simulation	141
13.1.2	Communication	142
13.2	Controller	142
13.2.1	Contact Point Intermediate Representation Control Law	143
13.2.2	Modifications for Stability	144
13.2.3	Added Friction	145

13.3	Experimental Results	145
13.3.1	Tool Colliding and Sliding on Floor	146
13.3.2	Peg in Hole Insertion	146
13.4	Discussion	146
13.4.1	Comparison to Virtual Coupling Interface Method	153
13.4.2	Further Potential Improvements	153
14	Conclusion	155
14.1	Summary of System Operation	155
14.1.1	Maglev Device Operation	155
14.1.2	Interaction with Simulation	156
14.2	Device Contributions	157
14.2.1	Design	157
14.2.2	Evaluation	157
14.2.3	Interactive Haptic Simulation Testbed	157
14.3	Simulation and Interface Contributions	158
14.3.1	System Integration	158
14.3.2	Interactive Demonstrations	158
14.3.3	Haptic User Interface Features	159
14.3.4	Application Interface	159
14.4	Future Directions for Research and Development	159
14.4.1	Psychophysical Research	159
14.4.2	Applications	159
14.4.3	Simulation Complexity and Realism	160
	Reference List	161
	Appendix A	
	Magnetic Levitation Device Jacobian	166
	Appendix B	
	Kinematic Solution	170
	B.1 Initial Estimate	170
	B.2 Iterative Improvement	170
	Appendix C	
	Design and Performance Parameters	172
	C.1 Design	172
	C.1.1 Motion Range	172
	C.1.2 Flotor	172
	C.1.3 Coils	173
	C.1.4 Sensors	173
	C.2 Dynamic Performance	173
	C.2.1 Force and Acceleration Limits	173
	C.2.2 Impedance Range	174
	C.2.3 Power Consumption	174

C.2.4 Control Bandwidths	174
------------------------------------	-----

Appendix D

Physical Simulation Software	175
D.1 Contact Constraint and Collision Forces	175
D.2 Collision Impulses	176
D.3 Collision Detection	177

List Of Tables

2.1	Published Haptic Interface Device Parameters	13
3.1	Coil Positions on Flotor Hemisphere	33
3.2	Finite Element Analysis Runtime Information	38
3.3	Lorentz Forces from Coil Currents in FEA Models	40
10.1	Cube in Cube Experiment Parameters	111
10.2	Surface Simulation Experiment Parameters	113
12.1	Environment and Controller Parameters for Peg in Hole Insertion Trial . . .	133

List Of Figures

1.1	Using a Desktop Maglev Haptic Interface Device with a Graphical Display	2
2.1	CMU Magnetic Levitation Haptic Interface in use with a Graphical Workstation	11
2.2	Device Cabinet	14
2.3	Hand Operation of the Haptic Device	15
2.4	Cutaway View of the Haptic Device	16
2.5	Hemispherical Flotor Shell with Coils, LEDs, and Handle	17
2.6	Sample Handles for Haptic Interface Device	18
2.7	PHANToM Haptic Interface Device	19
2.8	Magic Wrist	21
2.9	UBC Wrist	21
2.10	UBC Powermouse	21
3.1	Lorentz Force Actuation	24
3.2	Flotor and Stator Radii with Flotor in Center Position	26
3.3	Flotor and Stator Radii with Flotor at Translation Limit	26
3.4	Test Response with No Magnetic Field, 0.78 mm Al Plate	29
3.5	Test Response with No Magnetic Field, 1.56 mm Al Plate	29
3.6	Test Response Across Long Magnet Axis, 0.78 mm Al Plate	29
3.7	Test Response Across Long Magnet Axis, 1.56 mm Al Plate	29
3.8	Test Response Across Short Magnet Axis, 0.78 mm Al plate	29
3.9	Test Response Across Short Magnet Axis, 1.56 mm Al plate	29
3.10	New Flotor Coil Configuration	33
3.11	Configuration of Six Actuators	34
3.12	Single Actuator with Magnet Assemblies and Suspended Oval Coil	36
3.13	FEA 1/4 Model of Magnet Assembly and Coil in Control Volume	37
3.14	Coil and Magnet Shapes Projected onto a Plane	39
3.15	FEA Magnetic Field in Gap, 20 mm magnets	41
3.16	FEA Magnetic Field in Gap, 25 mm magnets	41
3.17	FEA Magnetic Field in Gap, 30 mm magnets	41
3.18	FEA Magnetic Field in Gap, 30 and 45 mm magnets	41
3.19	FEA Predicted Magnetic Field in Gap	43
3.20	Measured Magnetic Field in Gap	43
3.21	Test Actuator	44
4.1	Sensor configuration and Coordinate Frames	46

4.2	Sensor Housing Assembly	46
4.3	UDT SC-50D Position Sensitive Photodiode, Rear View	47
4.4	LED Position Calibration Grid	49
4.5	Sensor Signal Distortion	49
4.6	Interpolated Lookup Table Error	52
4.7	Vector Geometry For Forward Kinematics Solution	56
4.8	Convergence of LED-sensor Vector Lengths Solution	58
5.1	Hardware System Components in Device Enclosure	61
6.1	Fabricated Flotor	69
6.2	New Fabricated Lorentz Levitation Haptic Device	70
7.1	Sensor Signal with Clamped Flotor	77
7.2	Calculated Position with Clamped Flotor	77
7.3	Sensor Signal with Levitated Flotor	77
7.4	Calculated Position with Levitated Flotor	77
7.5	Amplifier Current Frequency Response	80
7.6	Force at Handle Frequency Response	80
7.7	Closed-Loop Position Response with Copper Coil Flotor at 650 Hz Control Rate	82
7.8	Closed-Loop Position Response with Copper Coil Flotor at 1.3 kHz Control Rate	83
7.9	Closed-Loop Position Response with Aluminum Coil Flotor at 650 Hz Control Rate	84
7.10	Closed-Loop Position Response with Aluminum Coil Flotor at 1.3 kHz Control Rate	85
8.1	Potential Contacts Aligning a Cube	93
9.1	Sample Display of Physically-Based Simulation	98
9.2	Texture modelling methods (a) Sandpaper system (b) surface grooves (c) stochastic	100
10.1	Single Vertex-Face Contact	106
10.2	Single Edge-Edge Contact	106
10.3	Multiple Contacts	106
10.4	Force and Torque Feedback from Contact Point	107
10.5	Local Cube-in-Cube Simulation	107
10.6	Local Peg-in-Hole Simulation	107
10.7	Surface Characteristic Simulation	110
10.8	Position and Force Data from Cube-in-Cube Simulation	112
10.9	Position and Force Data from Surface Simulation	114
11.1	User Interface for Haptic Interaction with Dynamic Physical Simulations	118
11.2	Haptic and Visual Interface System	119
11.3	Haptic Tool Selection in Coriolis Simulations	122

12.1 Haptic and Visual Interface System	129
12.2 Virtual Coupling of Simulation and Device	129
12.3 Block Manipulation Task	132
12.4 Peg-in-Hole	133
12.5 Key and Lock	133
12.6 Position and Force Data from Impact and Sliding on Floor	134
12.7 Position and Rotation Data from Peg in Hole Insertion	135
12.8 Force and Torque Data from Peg in Hole Insertion	136
13.1 Schematic of Contact Point Intermediate Representation Interface Configuration	142
13.2 Feedback Force and Torque from Single Intermediate Representation Contact Point	143
13.3 Position and Force Data from Impact and Sliding Motions with Contact Point Intermediate Representation	147
13.4 Position and Force Data from Impact and Sliding Motions with Contact Point Intermediate Representation and Local Friction	148
13.5 Position and Rotation Data from Peg in Hole Insertion with Contact Point Intermediate Representation	149
13.6 Force and Torque Data from Peg in Hole Insertion with Contact Point Intermediate Representation	150
13.7 Position and Rotation Data from Peg in Hole Insertion with Contact Point Intermediate Representation and Local Friction	151
13.8 Force and Torque Data from Peg in Hole Insertion with Contact Point Intermediate Representation and Local Friction	152
B.1 Vector Geometry For Inverse Kinematics Solution	171

Abstract

For my robotics dissertation I have designed and built a high-performance 6 degree of freedom magnetic levitation haptic interface device and integrated its operation for use with realistic, detailed, graphically displayed three-dimensional simulated rigid-body physical environments.

In tool-based haptic interaction, the user feels and interacts with the simulated environment through a rigid tool of a given shape rather than directly with the hand and fingers. Consequently, a tool-based haptic interaction device only needs to control the dynamics of the handle part grasped by the user, rather than stimulate the user's skin, joints, or muscles directly. Tool-based human tasks such as cutting, pushing, screwing, probing, and insertion can all be simulated with the new device.

An ideal haptic interface would enable simulated virtual objects to be sensed and manipulated in exactly the same natural and intuitive manner as real physical objects. Psychophysical studies have shown that the haptic sense of the human hand can detect surface variations down to the micron level and at a rate of at least several hundred Hertz [1, 2]. To perfectly emulate the experience of handling real objects, the haptic interface system would have to reproduce object dynamics with the same high level of detail and responsiveness. To approach this level of performance requires a device with stiff and lightweight moving parts, powerful and responsive actuators, high resolution sensors, and a fast, low latency control system.

Lorentz levitation technology [3] is especially well suited to high-performance tool-based haptic interaction because it provides motion and force feedback in 6 DOF with high control bandwidths and sensitivity by using non-contact frictionless actuation and position sensing. The motion range of the new magnetic levitation haptic interaction device is ± 12.5 mm in translation and $\pm 7.5^\circ$ in rotation to accommodate fingertip motions for fine haptic tasks. The device has a closed-loop, small-amplitude position bandwidth of at least 100 Hz in each translation and rotation axis, a maximum stiffness of 25.0 N/mm, and a position resolution of 5-10 μm .

The haptic interface device and control software has been integrated with a physically-based realtime dynamic simulation of rigid objects and the surface effects of texture and friction. The integration methods developed and implemented maximize the dynamic performance of the device and the haptic realism of interaction for the user when the processing power and communication bandwidth and latency of the local controller is limited. Additional user interface features such as rate-based control and viewpoint tracking modes, and variable scaling and offset gains have been implemented to enable the user to move the virtual tool easily over arbitrarily large distances and rotations in the simulated environment using the fingertip motion range of the haptic device. The effectiveness of the haptic interface system has been demonstrated with interactive simulations of sample tasks.

Chapter 1

Introduction

This research explores the new issues in 6 degrees of freedom [DOF] haptic interaction which arise from the high sensitivity and response bandwidths which can be realized using Lorentz magnetic levitation. For this dissertation, a new magnetic levitation haptic interface device was designed and built to provide high resolution and control bandwidths, and new control and modelling methods were developed and implemented to provide realistic haptic user interaction with simulated environments using the magnetic levitation device.

Haptic sensing refers to the sensing modalities of the hand, or more generally the sense of touch or feeling. The haptic sense is a combination of tactile sensing, which is local sensation from nerve receptors in the skin, and kinesthetic sensing, which is due to internal distributed sensation in the joints and muscles. Comprehensive surveys of psychophysical, perceptual, sensitivity and bandwidth issues in human haptic sensing are given by Shimoga [1] and Cholewiak and Collins [2].

Humans use dexterous motion of the hand together with the sense of touch and feel to gain information about the dynamics and surface characteristics of our environment when we grasp, squeeze, push, pick up, manipulate, or touch the surface of objects. The synthesis of haptic sensing with dexterous motion is *haptic exploration* or *haptic interaction*.

To enable haptic interaction with a virtual environment, a device is needed which can reproduce the haptically sensed characteristics of objects such as shape and stiffness and preferably more subtle characteristics such as surface friction and texture. The realism of the haptic interaction is determined by how well the speed, resolution, and sensitivity of the haptic interface device duplicates the characteristics of the simulated environment, up to the limits of human hand sensitivity.

The development and widespread availability of faster computer processing, cheaper memory, and improved algorithms make it possible to simulate more and more complex dynamic physical environments in real time while modelling collision and friction between

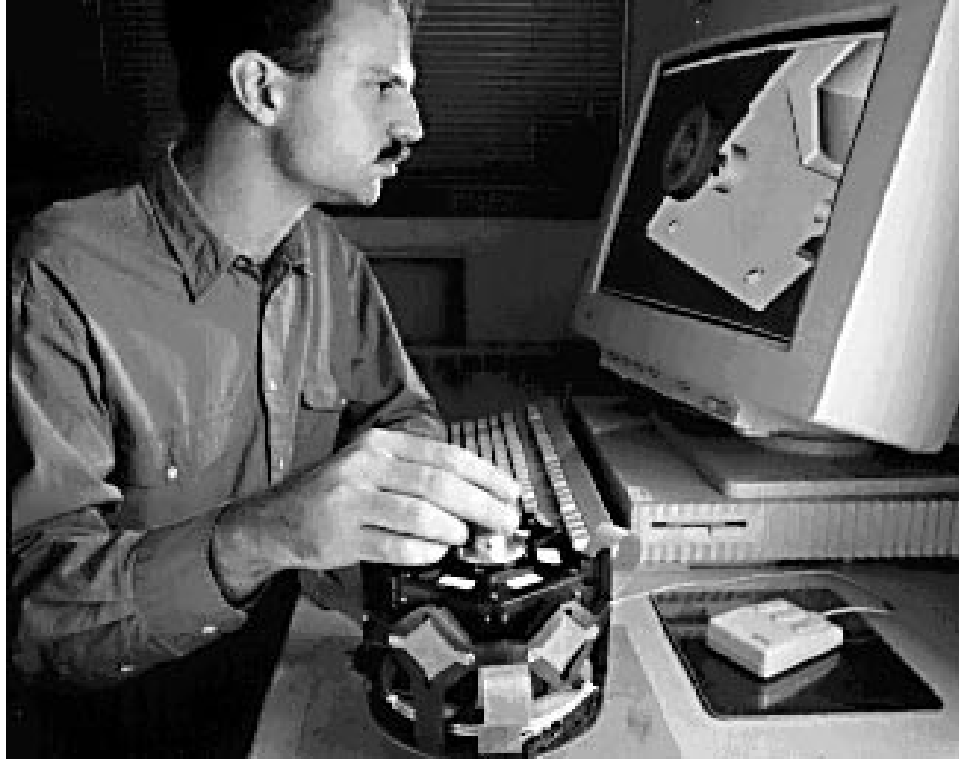


Figure 1.1: Using a Desktop Maglev Haptic Interface Device with a Graphical Display

multiple objects. A high-performance haptic interface device, fully integrated with a graphical display and a physical simulation as pictured in the example of Fig. 1.1 could give its user convincing interaction with a realistic environment. Comprehensive surveys in the current state-of-the-art in haptic interfaces are given in [4] and [5].

1.1 Applications of Haptic Interfaces

The main potential applications of haptic interaction are in the areas of CAD, teleoperation, biomolecular analysis, medical simulation, and entertainment. A haptic interface to a CAD system would enable a user to directly feel subtleties of the fit, surface finish, and inertia of modelled parts. Haptic feedback in teleoperation lets the user directly feel the interaction of a remote machine with its environment as it performs some task. Haptic interaction with medical simulations would allow a surgeon in training to realistically feel and manipulate body tissues. Additional possible applications arise from haptic exploration of any other data in three-dimensional space. A user could move in and feel any arbitrary properties representable in a vector field, such as fluid flow, pressure, magnetic field, or any potential

field gradient. The examples of multidimensional data sets shown graphically in Tufte's *Envisioning Information*[6] could similarly be represented haptically.

1.1.1 CAD Simulations

Haptic interaction can benefit CAD applications by giving the designer an immediate, physical feel for any parts which have been modelled. Physical characteristics such as the shape, surface texture and friction, stiffness, or the inertia of a modelled part would be displayed haptically to complement the user's visual perception of the part graphically displayed on a monitor. The fit and ease of assembly of parts in a complex mechanism could be felt directly before an actual prototype is fabricated. Part interferences that would be difficult to see on a flat CAD display or drawing would be obvious to the haptic interface user. The designer of a tool could get the actual feel of the tool during use before it was fabricated.

1.1.2 Medical Simulations

In medical applications, haptic interaction can add to the realism and effectiveness of simulated surgical or other medical procedures for training. The haptic interface user would be able to feel different simulated bodily tissues as they are operated upon such as the feel of muscle, tendon, and bone as they are cut with a scalpel or the feel of a displaced or broken bone as it is set into place. Microsurgery and minimally invasive surgical techniques could also benefit from additional haptic feedback built into the tools during the surgical procedures.

1.1.3 Entertainment

Rudimentary haptic interface devices are already currently commercially available in the mass market for video games. The Microsoft Sidewinder ProTM is a force-reflecting 3 DOF joystick and several game controllers for Nintendo and other video game consoles generate some impact or vibration feedback for the user. More demanding users may desire more sophisticated haptic interfaces to add more realistic interaction in the video game.

1.1.4 Teleoperation

Haptic feedback can be used to assist an operator for local control of a remotely located vehicle or machine. Scaling of forces and distances can be useful for teleoperation. Teleoperation with scaled haptic feedback has been applied in systems from giant earth-moving equipment [7] to microsurgery [8] and approaching molecular micromanipulation [9].

1.1.5 Other Applications

Haptic feedback has been applied also to biomolecular analysis. An operator can haptically manipulate modelled molecules to feel how they react and dock with other molecules. Extensive development by Brooks' group at UNC Chapel Hill has been done in this area [10].

Haptic interface devices could develop into a useful communication interface for the deaf and/or blind. The motion trajectory, direction, impedance or other behavior of the device could be translated to text for communication [3]. With a haptic interface, two-way communication can be done with a single device. Furthermore, haptic devices could convey meaning through means such as gesture, shape, texture, and rhythm in a continuously variable range rather than only by discrete text or other symbolic information that would be represented in a medium such as Braille writing.

1.2 Haptic Interaction Issues

The haptic interaction device performance criteria considered to be most important for this project are the sensitivity or position resolution, responsiveness or control bandwidths, and impedance range. There is an important distinction to be made between force and position bandwidths, where the force control bandwidth of a device determines the maximum frequency at which device can generate desired forces and the position bandwidth is the maximum frequency that the motion of the device can follow. For a force-actuated device with position feedback, the force bandwidth will be much greater since it depends only on the time constants of the actuators and any backlash in the transmissions or joints, but the position bandwidth is limited by the sensor bandwidths, the control rate and gains, and the inertia of the moving parts.

Ease of operation is an important consideration for a haptic device that may be used for extended periods of time or by untrained operators. Some regard should be made to basic ergonomic principles in the device design. Another consideration is the generality of the device and simulation environments.

With existing technology, no single haptic interface device can satisfy all combinations of tasks, environments, and applications. Our magnetic levitation haptic interface device aims to achieve high bandwidth, high resolution interaction that is practical and easy to use from the desk by the computer. A magnetic levitation haptic interface device would be applicable to fine dextrous manipulation tasks performed with the fingertips but would not be as suitable for other tasks requiring a larger range of motion.

Performance tradeoffs and acceptable impedance bounds for free and constrained motion were established experimentally in [11]. Mechanical control issues of haptic interface devices such as impedance range, control bandwidth and stability have been examined by Colgate *et al.* [12]. One of the main conclusions of this study was the overwhelming benefit of passive damping in the device for stable control over a wide range of device impedance.

1.2.1 Degrees of Freedom

Brooks' group at the University of North Carolina established Project GROPE for evaluation of spatial object placement and molecular docking tasks in virtual physical environments using 6 DOF haptic interfaces [10]. In their experiments, user performance was approximately doubled with the addition of force feedback. One of the observations from this research was that the manipulator arm haptic interfaces they were using were marginally adequate and noted that mechanical backlash, static friction, and other motion problems were "very troublesome". In their experience, a haptic device using finger and hand motions would be preferable to arm motions since the relative sensitivity of the hand and fingers is at least as great as the arm and hand, and finger-hand motions are less tiring.

With a 6 DOF device, the user can both locate and orient virtual objects with a single interaction in an intuitive, direct way which is not possible with any combination of lower dimensionality devices. In [13], Waters and Wang used 6 DOF input devices to manipulate objects and interact with a synthetic 3-D environment and concluded that the tasks considered would be very difficult with lower dimensionality devices.

A compact, inexpensive 6 DOF input device called Magellan or the Space Mouse has recently been widely commercialized by the German Aerospace Research Establishment and Logitech for robot control and 3-D CAD interaction[14]. The user guide reports that "flying an object in 6 DOF is done intuitively without any strain". It was found that users preferred to manipulate a small rounded puck shape with the fingertips than to grasp a large ball with the entire hand, since fingertip manipulation is more sensitive and less tiring. This device does not provide any force feedback to the user, however, and is therefore not a haptic interface device.

Taken together, the conclusions of the task device interaction studies described above suggest that an ideal device would fulfill the following criteria:

- passive damping
- finger-hand motion
- negligible backlash and static friction

- 6 DOF of motion and force feedback

The maglev haptic interface device described in Part I meets all the above criteria. Its control bandwidths and impedance range are maximized by minimizing the device inertia and maximizing the control rate.

1.2.2 Haptic Environment Emulation

An important research issue is how to haptically display modelled environments. Objects in the simulated environment have a given shape and stiffness or elasticity and may also have more detailed surface features such as textures and friction.

To enable the user to feel the objects in the simulation, collision detection must be performed and forces on the haptic device must be generated from the object models in a physically realistic way. Surface interpenetration with stiffness and damping or potential field feedback can be used to generate forces from environment surfaces and parameters. The operation of the simulation and device controller must remain stable at all times. The computational burden of collision detection and generation of haptic forces must be handled in an efficient way.

One example of haptic representation errors that may occur in a discretized time simulation is the surface “pop-through” effect [15, 16, 17]. Fast-moving small objects may completely pass through thin objects during one update interval without registering any opposing barrier force at all. For object interpenetrations near edges, the system must determine which face adjoining the edge should be used to calculate the force feedback magnitude and direction to avoid the user sensation of unnaturally tunneling through from one face to the other during motions near sharp edges.

Virtual coupling [16] is a method of compliantly coupling the haptic device with a tool representation in the simulation environment. Stability can be easily preserved in this method by modulating the stiffness and damping of the coupling. Development and operation of the haptic device and the simulation can be done independently to simplify the system.

Intermediate representation [18] is a method to simplify the calculation of haptic interaction forces. A local subset of the simulation environment is used to calculate interaction forces so that the forces can be calculated and generated at a faster update rate.

The modelling and display methods used must be appropriate to the scale of the surface features in the environment. As shape features increase in number and decrease in size, it becomes more effective to model them in the aggregate as a texture feature. As the texture

becomes finer to the point that individual facets cannot be distinguished, the texture can be represented as a combination of friction parameters only.

Haptic texture was first effectively displayed in the simple 2-D “sandpaper system” developed by Minsky[19]. Siira and Pai have developed and implemented a stochastic representation of haptic texture [20].

1.3 Lorentz Magnetic Levitation

To stably levitate a free object in space, a magnetic levitation device must have at least six actuators to control the six DOF of the levitated object. A Lorentz force actuator consists of a fixed magnet assembly and a current-carrying wire coil. The area of the coil that is in the magnetic field experiences a generated force

$$\mathbf{f} = \oint \mathbf{B} \times I d\mathbf{l}$$

where \mathbf{l} is the length of the wire in the coil, \mathbf{B} is the magnetic field, and I is the current in the wire. The actuation of the maglev device is analogous to a Stewart platform since all the actuators operate in parallel between a fixed base and a single moving platform. To provide position error feedback to the control system, non-contact position sensors are needed. Position sensing has been done using LED markers and position sensing photodiodes. In the rest of this document, the levitated part of the device will be referred to as the *flotor* and the stationary base as the *stator*.

Each magnetic levitation device has one moving part which is levitated by the interaction between magnetic fields and current-carrying coils embedded in the moving body. The only physical connection between the levitated body and the base is through thin, flexible, lightweight cables which carry sensor signals or LED marker wiring and the actuator coil currents. This structure and means of actuation eliminates problems such as friction and hysteresis which are inherent in manipulable force-reflecting systems such as robotic master arms and linkages. Furthermore, the levitation device provides a high degree of position and force accuracy and enables superior control bandwidths.

1.4 Outline

The remainder of this dissertation is divided into two parts. Part I describes hardware and control aspects including the design, function, fabrication, performance testing and results for the new haptic interface device. Part II describes methods and implementation of the integration of the interface device with physical simulations.

Chapter 2 in Part I introduces the general design of the new magnetic levitation haptic interface device including the motivation and performance goals for the new device, reviews the effectiveness of several representative haptic interface devices using actuated linkages and cables, and describes the other existing magnetic levitation haptic devices. The next two chapters cover the development and design of the actuation and position sensing subsystems of the device. Chapter 5 describes the fabrication methods used to build the device to the desired specifications at a reasonable cost and Chapter 6 describes the control system and methods for levitating and controlling the impedance and motion of the handle of the haptic interface device. Performance testing and results of the magnetic levitation device are presented in Chapter 7.

Chapter 8 in Part II begins with an overview of current research and development of dynamic rigid-body and surface characteristic simulation methods for haptic interaction and provides an introduction to goals of and approaches to haptic interaction with dynamic physical environments simulated on the host workstation, but with limitations on the communication bandwidth and latency between the workstation and the magnetic levitation device controller. Chapter 9 describes the CORIOLIS software package used for physical simulation and the methods used for haptic display of surface friction and texture. Chapter 10 describes haptic interaction simulations and provides experimental results for environments simulated on the local device control processor using limited computing power, and using the host workstation for graphical display only. In Chapter 11, haptic user interface issues are examined, and the added user interface features developed for interaction with dynamic environment simulations using the magnetic levitation haptic interface device are described.

Chapters 12 and 13 describe the methods and implementations of integrating the device control with a physical simulation, first by using a single virtual coupling between the levitated handle and the virtual tool in the simulation, and then by calculating and communicating the location of each vertex contact point in the simulation individually to determine the control impedance of the levitated handle. Chapter 14 reviews the contributions of the new magnetic levitation haptic interface device and the interactive simulation work described and suggests directions for the future development of the magnetic levitation haptic interface device and haptic interactive simulations.

Part I

Magnetic Levitation Haptic Interface Device

Chapter 2

CMU Maglev Haptic Device Design

The new device described here is the first Lorentz maglev device that was specifically designed for haptic interaction. The device in operation is shown in Fig. 2.1. Previous research using magnetic levitation fine motion devices demonstrated the feasibility of Lorentz magnetic levitation for haptic interaction, but also revealed the shortcomings of these positioning devices for haptics. The new design was completed by carefully evaluating desired characteristics for the device and finding a balance among conflicting requirements. The Pro/Engineer parametric solid modelling CAD package was used throughout the design process for visualization, ease of modifications, calculation of part clearances, and preparation of part drawings for fabrication.

The defining features of the new maglev device design are its large, decoupled motion range and the placement of the handle at the center of rotation of the flotor. The motion range in translation of the new device is at least 2.5 times greater in all directions for a workspace volume over 15 times greater than the previous Lorentz magnetic levitation devices [21, 22] described in Section 2.4, and the rotation range is approximately twice that of the other devices. The hemispherical shape of the device flotor resulted in the large motion ranges, the handle at the center, minimal mass, efficient high force and torque capability, and good heat dissipation. Further details of the new magnetic levitation device are available in a design paper [23].

2.1 Design Goals

The small, coupled ranges of translation and rotation in the Magic wrist [21] and UBC wrist [24] (<10 mm and $<10^\circ$) described in Section 2.4 severely limit the user's sensation of interaction and haptic shape perception in the simulated environment. The squat cylindrical shape of both devices makes them awkward to grasp and manipulate, which reduces the

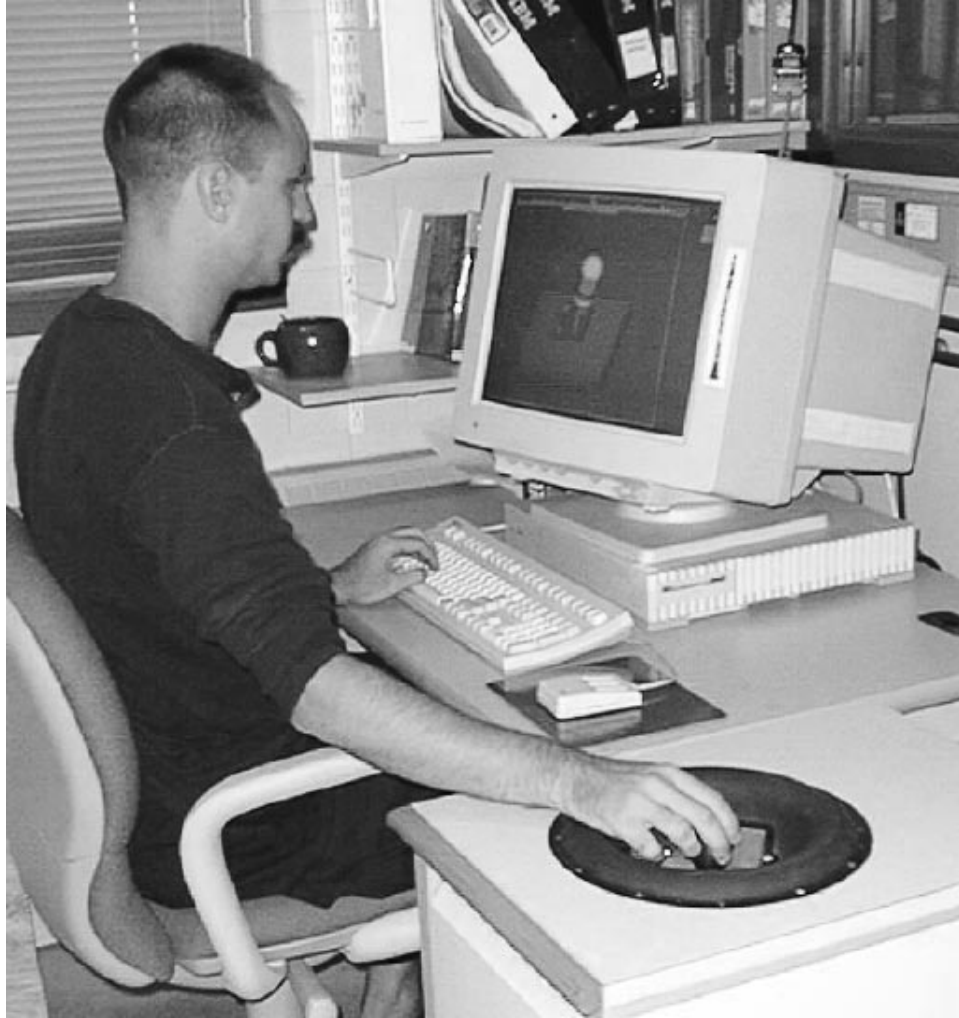


Figure 2.1: CMU Magnetic Levitation Haptic Interface in use with a Graphical Workstation

effectiveness of the haptic interaction. Ideally, the flotor should be grasped at the center of a rounded shape, so that the range of rotation is equal in all directions and nearly decoupled from translation. The range of motion should be comparable to the range of motion of a user's fingertips with the wrist resting on a table top.

Although the control bandwidths of the maglev devices are an improvement over most linkage devices, the bandwidths could be increased by increasing the controller sample rate, increasing the actuation forces, and reducing the inertia of the flotor. Furthermore, the complexity and number of parts in the completed device could be reduced.

Practicality of fabrication and use were also primary considerations in the design of the new maglev haptic device. The device should not be prohibitively expensive or difficult to produce and additional devices of the same design should be significantly cheaper and

easier to fabricate, so that some economy of scale could be realized if multiple devices are produced in the future. The number of parts should be minimized and the simplest fabrication processes possible should be used which can produce the necessary geometry and tolerances.

For practicality of use, the device should be portable, with the flotor and stator, current amplifiers, analog i/o, processing and power supplies all contained in a single wheeled cabinet. The top of the cabinet and the handle of the maglev device should be at a height so that it can be used comfortably for long periods and so it can be positioned next to a graphics workstation as an integrated visual and haptic interface. The handle should be easy to grasp and manipulate.

2.2 Configuration

To meet the described goals for a magnetic levitation haptic interface, a new device was designed for haptic interaction. The new haptic magnetic levitation device was designed specifically for haptic interaction, with a larger range of motion and more comfortable form for fingertip manipulation, to enable high-performance haptic interaction with a physically-based simulated environment with added effects of friction and texture. The new device significantly increases the force and motion ranges and bandwidths and provides a more ergonomic handle for the user to manipulate while retaining the force and position sensitivity of the older devices. The expected performance parameters of the device are given in Table 2.1. A comfortable hand grip is located at the center of rotation of the flotor and the translation range is at least ± 12.5 mm in all directions with at least $\pm 7^\circ$ of rotation to accommodate the typical fingertip motions for common tool handling tasks. Since the purpose of the device is fine haptic interaction using the fingertips, a motion range sufficient for arm motion is not necessary. The complete system including the device, amplifiers, power supplies, control processors, network communication, and analog I/O is contained in a single desktop-height cabinet enclosure that can be easily rolled up next to a desk and used with a graphical workstation.

2.2.1 Flotor and Stator

The new device flotor is a hemispherical shell with a handle at its center. With this shape, the flotor can be rotated about its center without colliding with the stator. The ranges of rotation and translation are nearly independent and equal in all directions. The user can reach in and grasp a handle located at its center. The flotor coils, the free space around

Device Type	DOF	Force Range	Motion Range	Bandwidth	Resolution
Texas 9-string[25]	6	43.4 N, 4.8 N-m	450 mm	<10 Hz pos.	NA
SPIDAR II [26]	6-12	4 N/cable	300 mm	30 Hz force	0.5 mm
NWU Platform [27]	4	45 N, 1.35 N-m	200x80 mm	<100 Hz	0.05°
U. Tsukuba Pen [28]	6	5 N	440 mm	<10 Hz pos.	2 mm
PER-Force [29]	6	53 N	100 mm, 90-180°	100 Hz	7.5 μ m
PHANToM [30]	3	8.5 N	130x180x250 mm	800 Hz force	0.03 mm
HapticMaster [31]	6	69 N, 0.5 N-m	400 mm	50 Hz force	0.4 mm
Magic Wrist [21] & UBC Wrist	6	20N	9 mm and 6°	15-50 Hz pos. 3 kHz force	< 5 μ m
New Maglev Device	6	60 N, 3 N-m	> 25 mm and 20°	> 100 Hz pos.	5 μ m

Table 2.1: Published Haptic Interface Device Parameters

the flotor, and the magnet assemblies in the stator or base all conform to the spherically curved shape.

The entire device is embedded in a compact desk-height enclosure with the flotor handle and the top rim of the stator at the level of the desktop surface. The device cabinet is shown in Fig. 2.2. This configuration allows the user’s wrist and forearm to rest on the stator rim and desktop while the haptic device handle is manipulated with the fingertips. This position is more comfortable and less tiring for manipulation and results in more sensitive fingertip control for the user since the weight of the hand and forearm is supported by the stator. The position of the user’s hand during operation of the haptic device is shown in Fig. 2.3.

In order to create a device with the desired motion range and hemispherical shape, the actuator configuration, magnet assemblies, and position sensing system were made with a completely new design distinctively different from the previous magnetic levitation devices from IBM and the University of British Columbia described in Section 2.4. Fig. 2.4 shows a cutaway representation of the device showing the principal parts including the levitated flotor, the stator enclosure, magnet and sensor assemblies, and the tool handle.

The flotor should have minimal levitated mass but high stiffness for the best performance. The main body of the flotor is a thin hemispherical aluminum shell with large oval cutouts for the actuator coils. The coils fit together in a densely packed configuration to maximize the area on the flotor used to generate actuation forces. LEDs for position sensing fit snugly between the coils. The coils are wound from ribbon wire on spherical forms to fit the curvature of the flotor. The design of the flotor assembly with the hemispherical shell, actuator coils, LEDs, and handle is shown in Fig. 2.5. Although the flotor mass is concentrated in the hemispherical shell, it is still easy to manipulate with one hand since the total mass is low and the weight is supported by actuation forces during use.

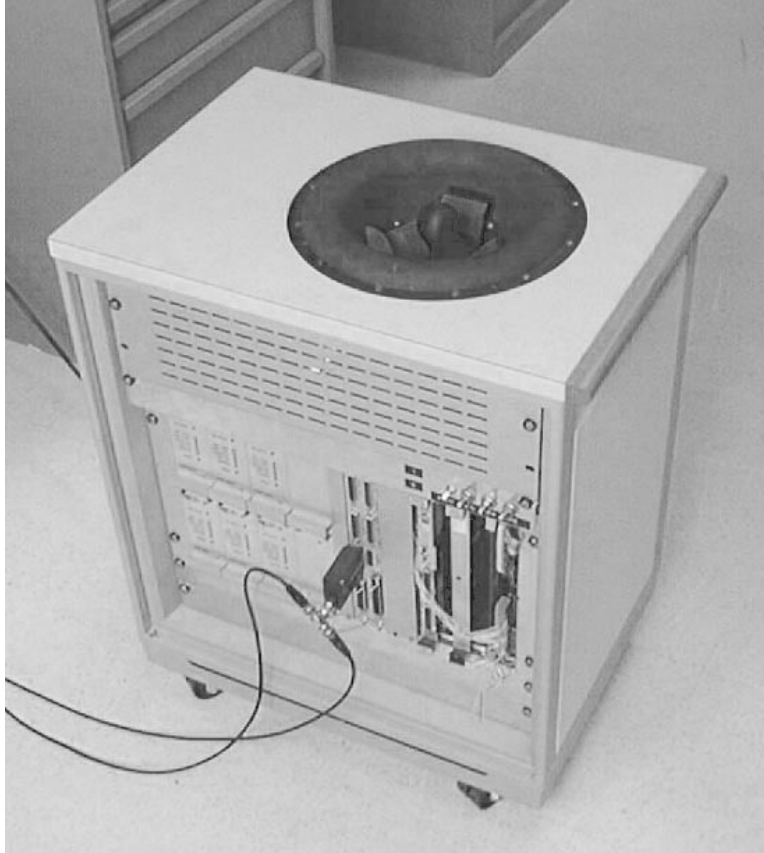


Figure 2.2: Device Cabinet

One drawback of Lorentz levitation actuation is that the range of motion of the device is limited by the width of the air gap in the magnetic assemblies. As a result, the 25-30 mm range of motion of this device is smaller than most other haptic interface devices. Since the aim of this device is to enable fine, dextrous, local haptic interaction using the fingertips rather than with large-scale arm motions, translation beyond typical fingertip motion for common tool-based tasks is less critical than the sensitivity and control bandwidths of the device.

Although the expanded motion range of the new device is large enough for fingertip motion, its utility would be severely limited if the user could only move the virtual tool over the same range in the simulated environment. To enable the user to move the virtual tool over larger distances in the simulated environment, variable scaling and offsets, and position-tracking and rate-based control modes are made available to the user during interaction. These additional control modes are described with the user interface features in Section 11.8.

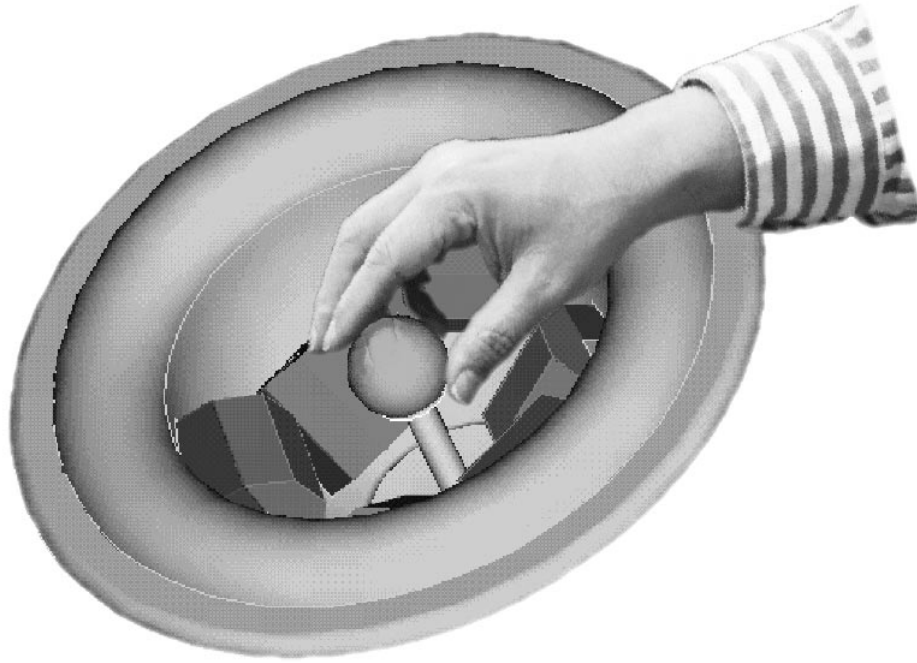


Figure 2.3: Hand Operation of the Haptic Device

2.2.2 Handles

Depending on the tasks to be simulated, other handle shapes such as a stylus, joystick, or T-grip may be more suitable than the ball grip as shown in Fig. 2.6. To enable the use of different handles, the handle mount was designed to accommodate replaceable handles. For the best correspondence between the device and the simulation, the tool handle shown in the virtual environment should be the same shape as the handle currently installed on the device flotor. A T-grip handle was found to provide a good combination of manipulability for sensitive or powerful force and motion for generic tasks. The T-grip is the easiest shape for the user to grasp and manipulate in all directions in both translation and rotation.

A number of additional handles have been built and used for suitability and ease of use with various different haptic interaction tasks. The different handle shapes and their advantages and disadvantages are listed below:

ball handle: symmetrical, easy to grip, compact, good for fine fingertip motion, but difficult for the user to apply high torques.

T-grip: best for large forces and torques, comfortable hand position, but not as good for dextrous fine motions.

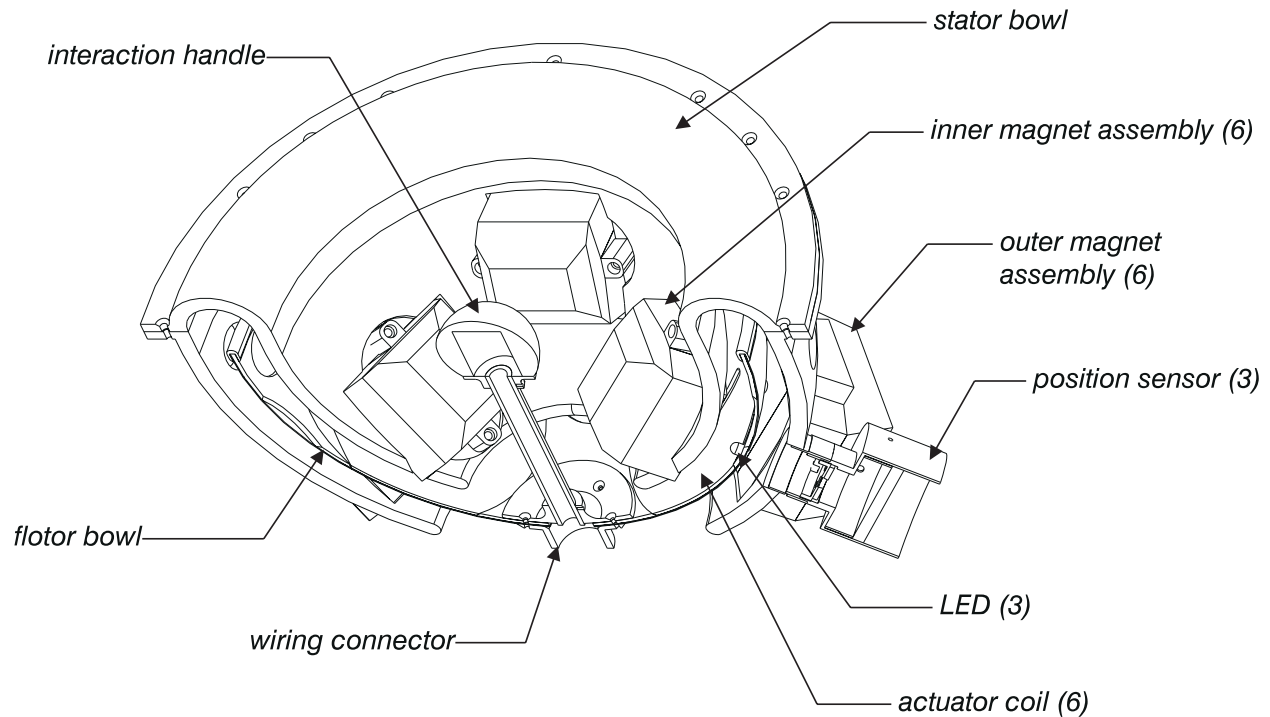


Figure 2.4: Cutaway View of the Haptic Device

joystick: familiar, responsive, accommodates added buttons or switches

stylus: thin pen shape is very good for fine tool manipulation, thicker handle is good for typical hand shop tools such a screwdriver or wrench; much more difficult to apply high torques in all directions and difficult to manipulate large inertias.

Various buttons or switches could also be added to the handles in the future so that the user can easily switch between different operating modes such as position or rate-based control, or initiate some action in the simulated environment such as grasping and release, using the same hand which is gripping the haptic handle.

A compact 6 DOF force sensor such as the miniature models commercially available¹ could also be added in between the device handle and the coil hemisphere in future work to enable inertial compensation, improve dynamic control accuracy, allow a wider variety of control algorithms, and provide information to the simulation.

¹ATI Industrial Automation, North Carolina

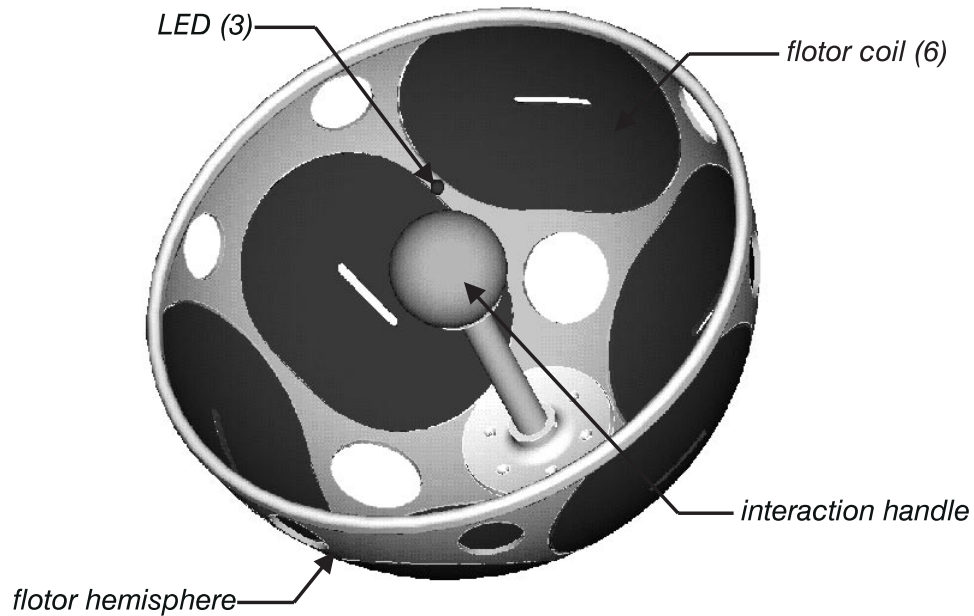


Figure 2.5: Hemispherical Flotor Shell with Coils, LEDs, and Handle

2.3 Cable and Linkage Haptic Interface Devices

Haptic interface devices can be classified according to whether they are body-referenced (fixed to the user) or ground-referenced (fixed to a stationary base), whether they provide tactile and/or force feedback, whether they enclose or are grasped by the user, whether they are operated by the entire arm, hand, or fingers only, and by how many degrees of freedom of motion and force feedback they provide to the user. The magnetic levitation haptic device developed in this work is a tool-based, ground-referenced, 6 DOF force feedback device.

The development of haptic interface devices began from large, heavy serial manipulators used as force-reflecting hand controller masters and has progressed to increasingly fast, lightweight, and sensitive exoskeleton, linkage, and tensioned cable devices. The main reported performance parameters of several existing force/kinesthesia based devices are summarized in Table 2.1.

Bergamasco and others [32] have developed force-reflecting exoskeleton systems; others are commercially available[33, 34]. Haptic exoskeletons may be either be supported by a fixed base or by the user's body only. Exoskeleton devices generally have the large inertia typical of serial linkage manipulators and fit may be a problem for users of different sizes.

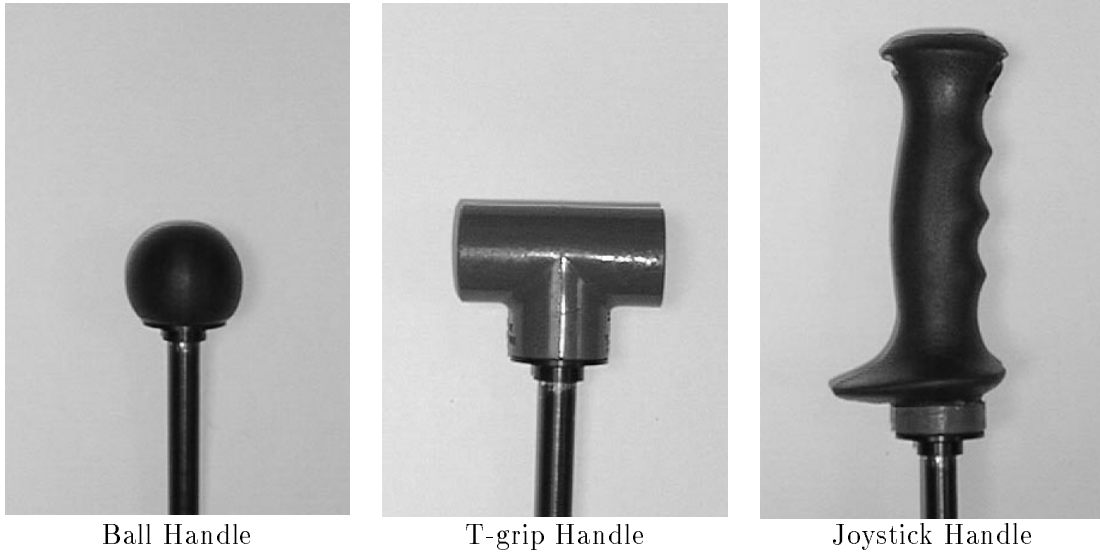


Figure 2.6: Sample Handles for Haptic Interface Device

Tensioned cable systems for haptic interaction have been developed at JPL [35], Tokyo Institute of Technology [26], and the University of Texas at Austin [25]. In these systems, a handle grasped by the user is supported from all directions by several actuated cables. The combined tension in the cables produces a net force and torque on the user's hand. The workspace of the device can be made very large while the actuated inertia remains small.

The Rutgers Master [36] developed by Burdea is a glove with pneumatic actuators between the palm and fingertips to emulate grasp forces. A 10 DOF enclosing glove-type mechanism with a fixed base was developed at the University of Tokyo [37].

A lightweight 3 DOF linkage called the PHANTOM [30], pictured in Fig. 2.7, was developed at MIT and is commercially produced by SensAble Technologies, Inc. This device has gained a measure of acceptance in the haptic research community and is probably the most widely used device in haptic research labs. The user interacts with this device through a fingertip thimble or grasped stylus. An optional encoder gimbal adds 3 DOF of orientation sensing but no feedback forces. The Pantograph, [38] developed at McGill University, is a small 2 DOF planar linkage with low inertias and reported to be capable of high bandwidths.

A 6 DOF platform called the HapticMaster was developed by Iwata at the University of Tsukuba and is commercially produced by Nissho Electronics[31]. The moving platform of this device is supported by three 3 DOF pantograph linkages, resulting in redundant actuation with 9 DOF. A compact pen-based 7 DOF linkage and tendon device is under development at McGill University which aims to provide dynamic response to 50-100 Hz, forces from 1 mN to 10 N, and variation in mechanical impedance of 1000:1 [39].

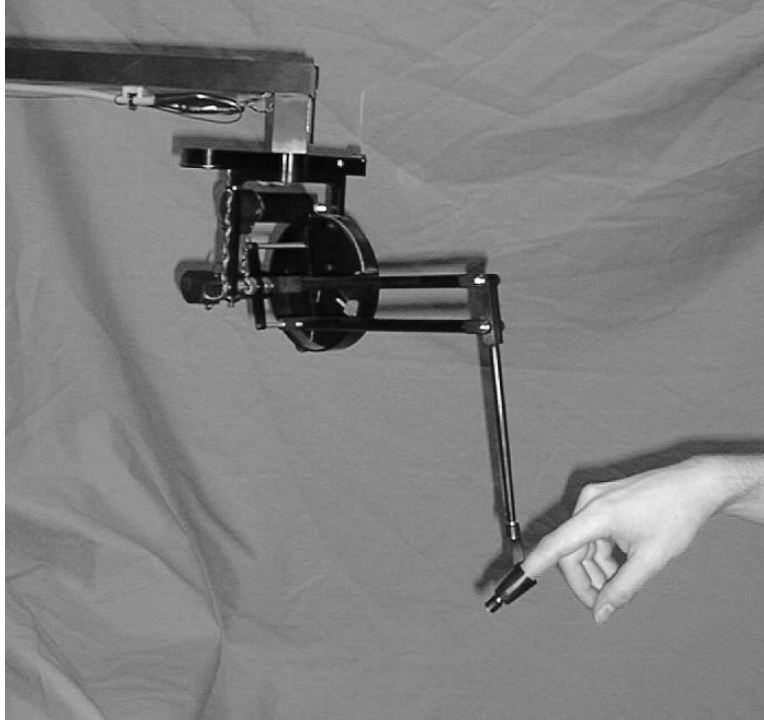


Figure 2.7: PHANToM Haptic Interface Device

Tactile haptic perception has been studied with various small devices attached on the hand which use vibration, heat, pin arrays, or pneumatic bladders to simulate tactile contact and pressure with virtual objects. Specialized force-reflecting devices have been made for simulations of specific medical procedures, such as laparoscopy [40], eye surgery [41], and other medical procedures [42].

2.4 Other Magnetic Levitation Haptic Devices

The haptic interaction approach used in this dissertation is Lorentz force magnetic levitation technology. The advantages of Lorentz magnetic levitation over conventional motors and linkages for haptic interaction device actuation are non-contact actuation, high force and motion control bandwidths, and 6 DOF in a compact device with only one moving part. This means of haptic interaction was previously demonstrated with the IBM Magic Wrist [21] and the UBC wrist [22]; another small desktop maglev haptic device has also been recently developed at UBC [43]. These devices are pictured in Figs. 2.8, 2.10, and 2.10.

2.4.1 Magic Wrist

The first Lorentz force magnetic levitation device was the *Magic Wrist* [21] developed at IBM T.J. Watson Research Center. This device was originally designed as a fine-motion robot wrist for coarse-fine manipulation and later adapted for use as a haptic interface for mechanism emulation, solid contacts, and texture and friction experiments [44]. The magnet assemblies in this device are arranged in inner and outer rings and the flotor coils are embedded in the sides of a hexagonal box. Three position sensing photodiodes on the inside of the flotor and three narrow-beam LEDs on the stator are used for position sensing. Its force, motion, and bandwidth parameters are also given in Table 2.1.

Due to the hexagonal shape of the flotor, the ranges of motion in translation and rotation are not independent. At the maximum translation the rotation range decreases to zero, and similarly there is no translation range at the maximum rotation because the edges and corners of the hexagonal box flotor collide with the magnet assemblies. These limitations of the Magic Wrist are not a problem when the wrist is carried by a 6 DOF robot arm, but in a haptic interface device a motion range under 10 mm may be a drawback for effective haptic interaction.

2.4.2 UBC Wrist

The UBC Wrist[22] is another 6 DOF Lorentz force magnetic levitation device that has been used as a teleoperation master, as a fine motion wrist, and for haptic interaction [45]. The UBC wrist is smaller and has a more compact arrangement of actuators but it has a similar cylindrical shape. Its inertia is 0.65 kg and position and force control bandwidths and ranges of force and motion are similar to the Magic Wrist.

Both the IBM magic wrist and the UBC wrist were designed as compact fine motion robot wrists for high position accuracy and sensitivity, so their limited motion range and non-ergonomic shape was not a consideration.

2.4.3 UBC Powermouse

Another magnetic levitation haptic device was recently developed at UBC by Salcudean *et al* and is called the Powermouse[43]. This device was designed for smaller size and lower cost and contains a novel geometry and optical position sensing setup using light curtains and linear position sensing photodiodes. The entire device and its enclosure is approximately shoebox-sized. The mass of the moving part is 260 grams and its motion range is ± 3 mm

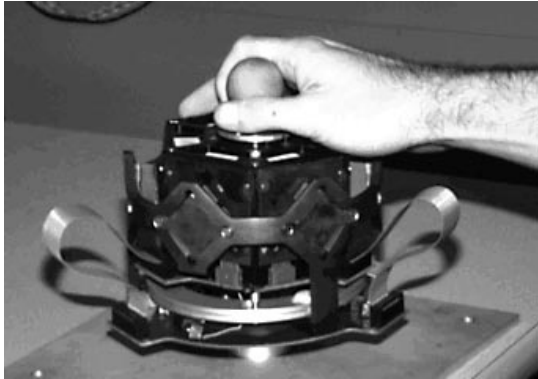


Figure 2.8: Magic Wrist

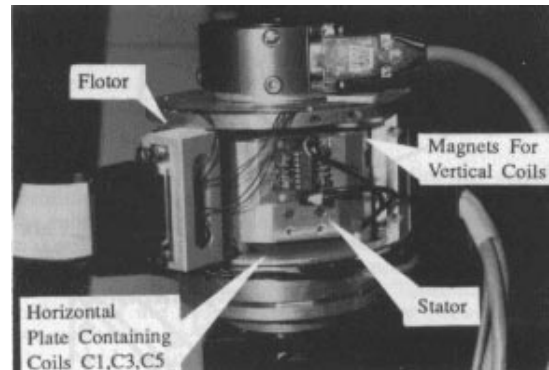


Figure 2.9: UBC Wrist

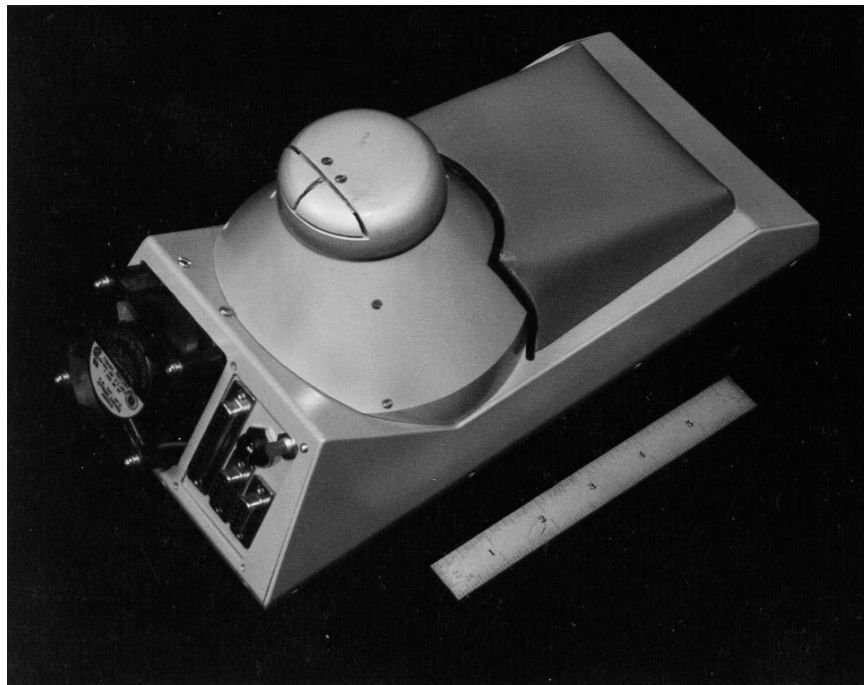


Figure 2.10: UBC Powermouse

and $\pm 5^\circ$. It can generate accelerations of over 10 g and only requires 1.6 W of power for levitation.

Chapter 3

Actuation

The force used for actuation in Lorentz force magnetic levitation is the electromagnetic force that is exerted on a wire of length l carrying an electrical current I in a magnetic field B , as expressed below:

$$\mathbf{f} = \oint \mathbf{B} \times I d\mathbf{l}. \quad (3.1)$$

In a uniform magnetic field, the generated force is independent of the position of the wire and Equation 3.1 simplifies to:

$$\mathbf{f} = \mathbf{B} \times (I\mathbf{l}). \quad (3.2)$$

A single Lorentz force actuator consists of two opposing fixed magnet assemblies and an oval wound coil suspended between them in the magnetic circuit air gaps. The Lorentz forces \mathbf{f} on the coil are generated in the two air gaps where the coil current loop I intersects the magnetic flux loop B and are perpendicular to both the current and the magnetic field vectors, as shown in Fig. 3.1. Both the electrical current and the magnetic field are in opposite directions in the two regions where the Lorentz force is generated, so the generated forces are in the same direction.

For stable levitation of a rigid body at least six actuators are necessary, since an unconstrained rigid body in space has a total of six degrees of freedom, three in translation and three in rotation. The Lorentz force actuators for the new magnetic levitation device are of the same type as the previous Magic Wrist and UBC Wrist maglev devices, but the design of the individual actuator magnet assemblies and coils and the arrangement of the six actuators in the device is unique to the new device in order to conform to the hemispherical flotor shape described in Chapter 2. The width of the field gap in the magnet assemblies is also several times greater than that of previous Lorentz magnetic levitation devices. To produce the actuator design and configuration for the new device, extensive geometric and magnetic field analysis was required.

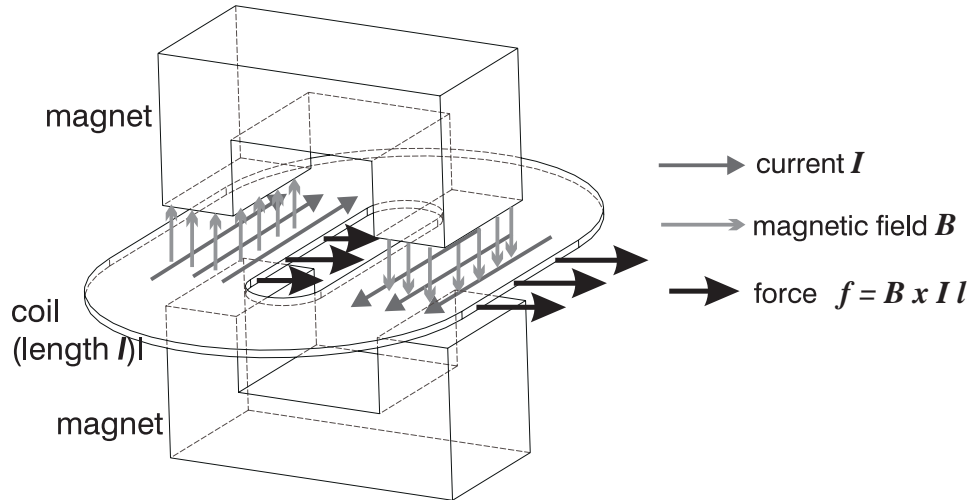


Figure 3.1: Lorentz Force Actuation

As with the previous Lorentz maglev devices, NdFeB alloy permanent magnets were used for the magnet assemblies. The current maglev device uses the highest $B_{max}H_{max}$ energy product magnets commercially available with a maximum energy product of 48 MGOe¹. The outermost aluminum shell covering the coils adds passive damping to the motion of the flotor due to eddy currents generated by motion through the strong magnetic fields in the stator magnet assembly gaps, which increases the range of stability of the controlled device. Testing procedures and results on the effects of eddy current damping in the outer hemisphere shell on the dynamics of the flotor are given in Section 3.1.2.

3.1 Actuator Configuration

A hemispherical shell was selected for the flotor shape in order to maximize the rotational range of the device and to enable the user to access a handle at the center of the flotor, as described in Chapter 2. The design of the overall size and shape of the magnetic levitation device and the arrangement of the actuators on the hemisphere developed from the constraint of the hemispherical flotor shape and maximizing the force and motion ranges of the device while minimizing the flotor mass.

¹Magnet Sales and Manufacturing, Inc., Culver City CA.

3.1.1 Hemisphere Sizes

The radius of the flotor hemisphere drives all the other radial dimensions of the device, since the coils must all fit together on the flotor shell and all other device components must fit in and around the flotor. The inner and outer stator shells enclose the flotor and hold the magnet assemblies and the separation between the stator shells determines the translational motion range of the flotor.

The hemisphere radius is limited due to mass and stiffness considerations for the flotor, and cost and safety considerations due to the sizes of the high energy magnets. Actuator forces are proportional to the area of the flotor coils, since the Lorentz force increases with the number of coil turns and the length of each turn in the magnetic field. Given a constant flotor thickness, the actuator forces and the flotor mass are both proportional to the flotor radius squared and the ratio of actuator forces to the flotor mass remains constant. A less massive flotor would have a smaller moment of inertia, however, and would be easier for the user to manipulate during unconstrained motion when there are no feedback forces to be applied from a haptic interaction simulation and the haptic tool is in free space. A smaller radius flotor shell would also be stiffer than a larger one of equal thickness. Also, the required volume of the actuator magnets scales with the hemisphere radius cubed, so the cost of the required magnets and the difficulty of safely assembling them increases very rapidly for larger radii.

To determine the optimal radius for the flotor hemisphere, the following design parameters and assumptions were taken into account:

Design Parameter:	Value:
Translation range T	25 mm
Maximum flotor thickness F	4 mm
Maximum magnet thickness M	30 mm
Maximum iron pole piece thickness P	15 mm
Maximum flotor handle radius H	20 mm
Minimum clearance around handle for user hand C	23 mm

There must be enough space inside the flotor shell to accommodate half of the flotor thickness and translation range of the device, the inner stator bowl and its six magnet assemblies, the user's hand, the size of the flotor handle, and the range of motion of the

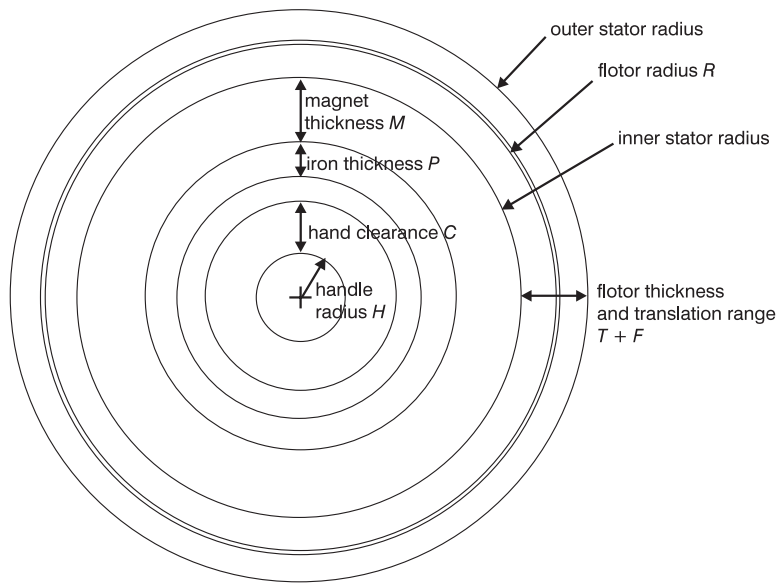


Figure 3.2: Flotor and Stator Radii with Flotor in Center Position

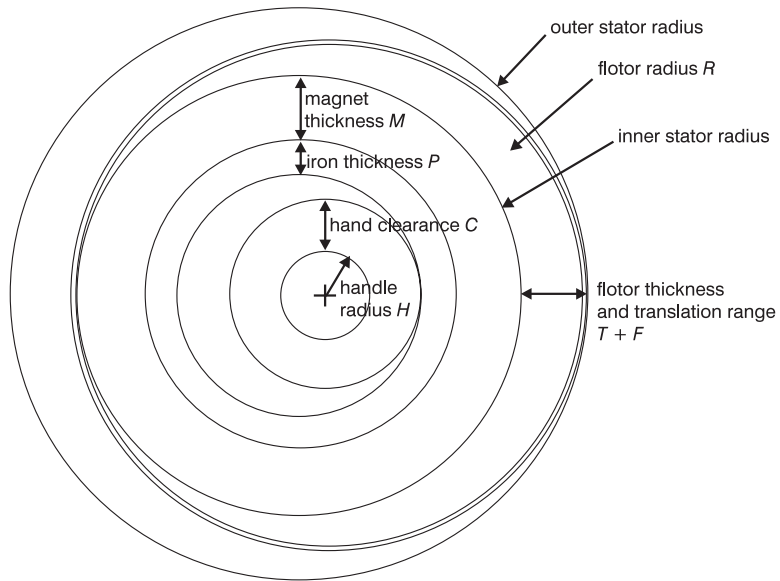


Figure 3.3: Flotor and Stator Radii with Flotor at Translation Limit

handle. By combining these constraints and dimensions together, the best flotor hemisphere radius

$$R = (T + F)/2 + M + P + T/2 + C + H \quad (3.3)$$

was determined to be 115 mm. The radius of the inner stator bowl surface facing the flotor is set to $R - (T + F)/2$ and rounded down 0.5 mm for 100 mm, and the outer stator surface facing in to the flotor is similarly set to 130 mm. The flotor shell radius of 115 mm results in large actuator surface areas and forces, yet allows the levitated mass to be easily manipulated and the entire device to be contained in a deskside cabinet enclosure. The flotor and stator radii and clearances are shown in Figs. 3.2 and 3.3.

3.1.2 Outer Shell Thickness

The magnetic levitation flotor hemisphere has a thin outer shell to hold the actuator coils in place and protect them from impacts with the magnets and stator bowl. If the shell is made of a conductive material, eddy currents will be generated when the shell moves in the magnetic field of actuator magnet assemblies, causing viscous damping.

Passive damping can significantly increase the impedance range of a haptic device, as mentioned by Hollis and Salcudean [21] for the design of the IBM Magic Wrist and shown by Colgate [12], so it is advantageous to design the flotor with a metallic shell which generates eddy current damping. Excessive damping would be detrimental to the dynamic response of the magnetic levitation device, however, and the flotor's resistance to motion would feel unnatural to the user when the haptic interface device is being used to move a simulated tool freely in space, although the effect of the passive damping could be reduced significantly when desired by applying negative active damping.

The outer flotor shell is made of aluminum due to considerations of material cost, ductility, and stiffness. To select the shell thickness that would provide the most suitable degree of eddy current damping, sample aluminum plates of different thicknesses were tested in motion in the actuator magnetic fields.

To test each aluminum plate, it was mounted on a linear bearing and coupled to a torsion spring and motor. The aluminum plate was positioned in the center of the gap between the magnet assemblies in a test actuator shown in Fig. 3.21. Each aluminum plate sample was at least as large as an actuator coil in order to capture all of the field in the magnet assembly gap. The plates were oscillated in both the long and short axis directions of the magnets. A random signal from a dynamic signal analyzer was used as the input for the motor amplifier and the motor shaft position was fed back to the dynamic signal analyzer.

The frequency response results for aluminum plates 0.78 and 1.56 mm thick, tested in both directions in the actuator magnetic fields and without the magnets are shown in Figs. 3.4 through 3.9. The masses of these two plates were 39.0 and 75.3 g respectively.

The effect of eddy current damping in the test setup is readily apparent from the frequency response plots for both aluminum plate setups. With no magnetic field the frequency response magnitudes show resonance peaks before dropping off, but when the plates are oscillated across the short axes of the magnet gaps the resonance peaks are almost completely damped out. The effect of the eddy current damping is less when the plates are oscillated along the long axis of the magnets because the gradient of the magnetic field is less.

Since the effect of the eddy current damping is quite significant even for the smaller tested thickness, a shell thickness of approximately 0.75 mm was selected for the flotor hemisphere, the thinnest possible shell thickness that could be spun into a hemisphere by our local supplier. To further diminish the effect of eddy current damping in the flotor shell, the thickness of the aluminum could be reduced by chemical etching or by using metal forming processes other than spinning, provided that the flotor stiffness is not reduced to the point that the resonant modes of the flotor hemisphere reduce the control bandwidths of the device.

3.1.3 Actuator Configuration Considerations

The forces and torques generated on the flotor when the actuator coils are positioned between their corresponding magnet assemblies can be expressed as $\mathbf{F} = \mathbf{A}\mathbf{I}$, where $\mathbf{F} = (f_x, f_y, f_z, \tau_x, \tau_y, \tau_z)$ is a 6-vector consisting of the forces and torques exerted on the flotor center, $\mathbf{I} = (I_1, \dots, I_6)^T$ a 6-vector of the coil currents, and \mathbf{A} is the matrix which maps currents to forces. \mathbf{A} can be obtained from the x, y and z components of the Lorentz forces from each coil according to Equation 4.2 and the x, y and z components of the torques from each coil combined into matrix form. The torques produced by each coil are given by

$$\boldsymbol{\tau}_i = \mathbf{c}_i \times \mathbf{f}_i, \quad (3.4)$$

where $\boldsymbol{\tau}_i$ is the torque caused by the force \mathbf{f}_i from coil i located at \mathbf{c}_i measured from the flotor center[21], so the combined matrix equation is:

$$[\mathbf{f} \ \boldsymbol{\tau}] = \begin{bmatrix} \mathbf{l}_1 \times \mathbf{B}_1 & \cdots & \mathbf{l}_6 \times \mathbf{B}_6 \\ \mathbf{c}_1 \times (\mathbf{l}_1 \times \mathbf{B}_1) & \cdots & \mathbf{c}_6 \times (\mathbf{l}_6 \times \mathbf{B}_6) \end{bmatrix} \begin{bmatrix} I_1 \\ \vdots \\ I_6 \end{bmatrix}. \quad (3.5)$$

The matrix \mathbf{A} determines the efficiency of force and torque generation in different directions. The top three rows determine generated forces and the bottom three rows determine

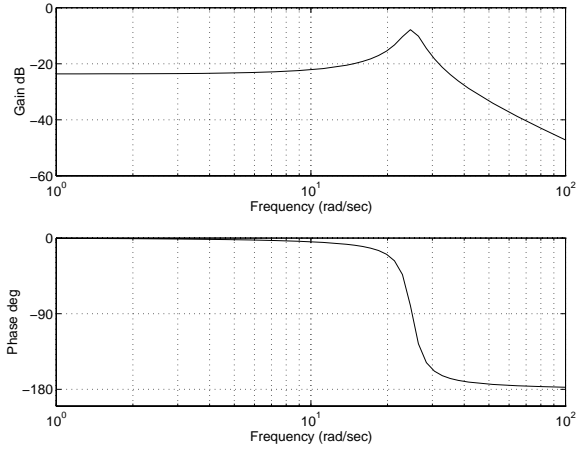


Figure 3.4: Test Response with No Magnetic Field, 0.78 mm Al Plate

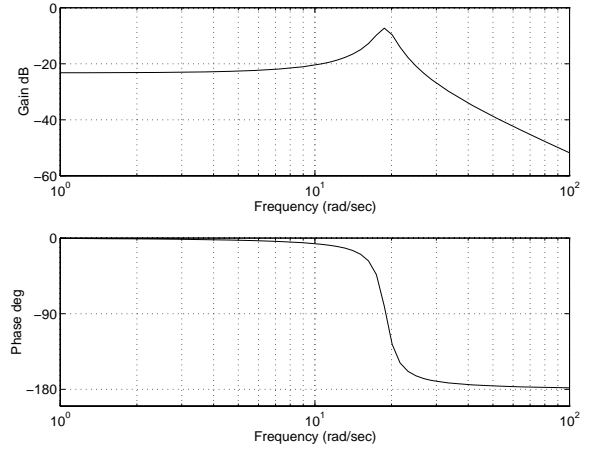


Figure 3.5: Test Response with No Magnetic Field, 1.56 mm Al Plate

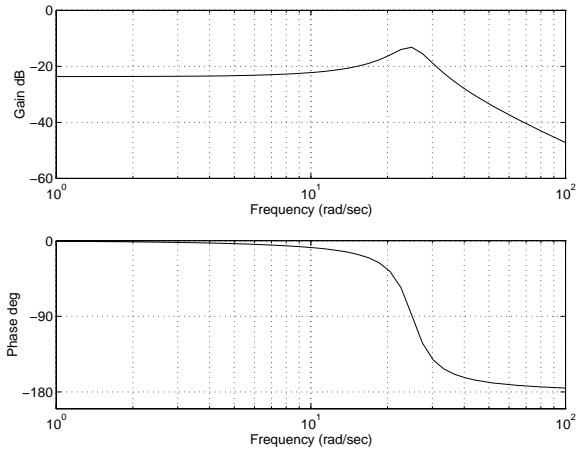


Figure 3.6: Test Response Across Long Magnet Axis, 0.78 mm Al Plate

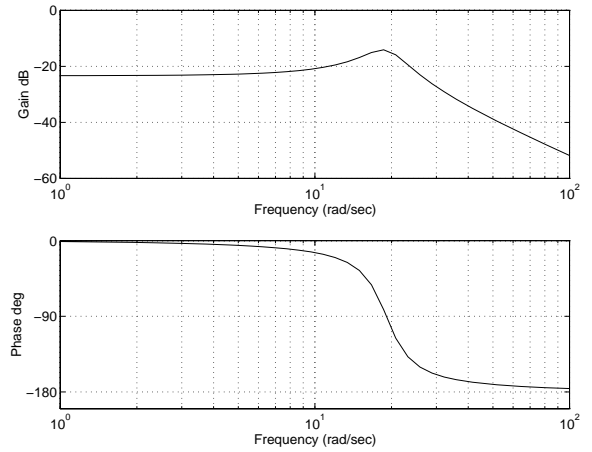


Figure 3.7: Test Response Across Long Magnet Axis, 1.56 mm Al Plate

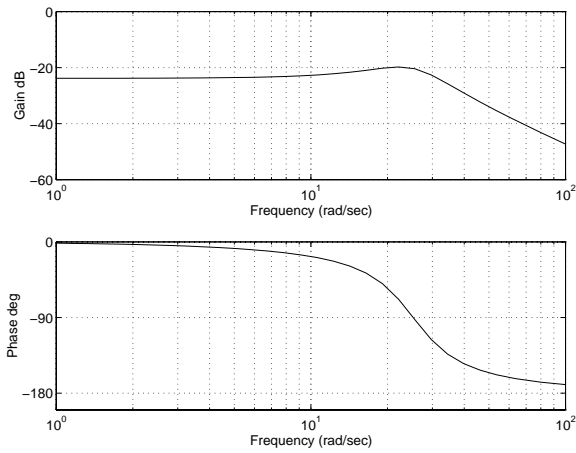


Figure 3.8: Test Response Across Short Magnet Axis, 0.78 mm Al plate

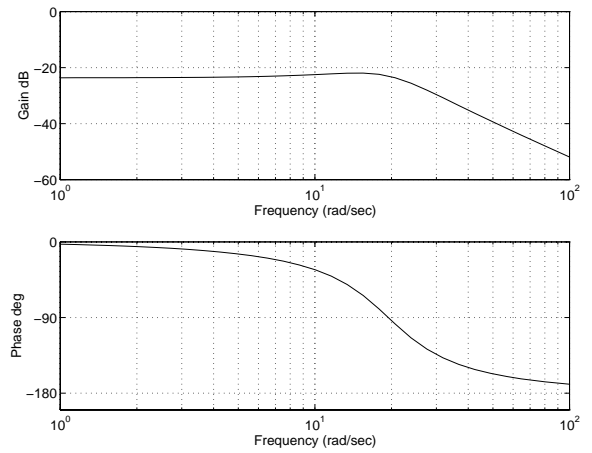


Figure 3.9: Test Response Across Short Magnet Axis, 1.56 mm Al plate

generated torques. It is analogous to the transposed Jacobian matrix and the manipulability matrix for manipulators introduced by Yoshikawa [46]. For the device to be able to generate forces and torques in all possible directions from a given coil configuration, \mathbf{A} must be nonsingular. Furthermore, the range of attainable forces and torques from a given maximum current for each coil describes ellipsoids whose principal axes are determined by the singular values of \mathbf{A} . If the coil configuration is radially symmetric about the vertical axis of the flotor, the singular values corresponding to the x and y translational axes will be the same and similarly the singular values for the x and y rotational axes will also be the same.

The relative force and torque generation efficiencies of different configurations of a set of six actuator coils can be compared by calculating the ratios of the elements of the singular value decomposition of the \mathbf{A} matrix calculated from each set of coils. The \mathbf{B} , \mathbf{C} , and \mathbf{H} vectors will be set to normalized unit lengths to calculate \mathbf{A} matrices to compare the merits of different coil configurations with all other factors being equal.

The most efficient coil configuration would be to locate the six coils on the six sides of a cube, with the long axes of the coils on each the the three opposing pairs of faces oriented in the three principal directions. This arrangement results in a normalized manipulability matrix of:

$$\mathbf{A}_{cube} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix}$$

and a singular value decomposition of

$$\mathbf{s}_{cube} = [\sqrt{2} \ \sqrt{2} \ \sqrt{2} \ \sqrt{2} \ \sqrt{2} \ \sqrt{2}].$$

This coil arrangement can generate forces and torques equally well in all directions, but the coils would not fit on a hemispherical shell and it would be very difficult conceive of a design to enable the user to reach between the coils to grasp and manipulate a handle in the center of the flotor. The UBC powermouse magnetic levitation haptic device actually does arrange the actuator coils on the faces of a cube, but the user grasps a handle located outside the flotor near one of the edges of the cube [43].

Another sample coil configuration is that of the Magic Wrist [21] with the coils on the sides of a hexagonal box. On the first Magic Wrist prototype, the long axes of the

coils alternated between horizontal and vertical orientations around the hexagonal box. Its normalized force manipulability matrix is shown below, where S is the sine function and C the cosine:

$$\mathbf{A}_{w1} = \begin{bmatrix} 0 & 0 & S(2\pi/3) & 0 & S(4\pi/3) & 0 \\ -1.0 & 0 & -C(2\pi/3) & 0 & -C(4\pi/3) & 0 \\ 0 & -1.0 & 0 & -1.0 & 0 & -1.0 \\ 0 & -S(\pi/3) & 0 & 0 & 0 & -S(5\pi/3) \\ 0 & C(\pi/3) & 0 & -1.0 & 0 & C(5\pi/3) \\ -1.0 & 0 & -1.0 & 0 & -1.0 & 0 \end{bmatrix}.$$

A “production” version of the Magic Wrist was also made in which the long axes of the coils on the hexagonal box alternated between $+45^\circ$ and -45° from the horizontal plane so that the currents to generate vertical forces would be distributed over all six coils rather than only three, thereby improving heat dissipation for vertical forces. The \mathbf{A} matrix for this configuration is:

$$\mathbf{A}_{w2} = \begin{bmatrix} 0 & -S(\pi/3)S(-\pi/4) & S(2\pi/3)S(\pi/4) & 0 & S(4\pi/3)S(\pi/4) & -S(5\pi/3)S(-\pi/4) \\ -S(\pi/4) & C(\pi/3)S(-\pi/4) & C(2\pi/3)S(\pi/4) & -S(-\pi/4) & -C(4\pi/3)S(\pi/4) & C(5\pi/3)S(-\pi/4) \\ -C(\pi/4) & C(-\pi/4) & -C(\pi/4) & C(-\pi/4) & -C(\pi/4) & C(-\pi/4) \\ 0 & -S(\pi/3)S(-\pi/4) & -S(2\pi/3)S(\pi/4) & 0 & -S(4\pi/3)S(\pi/4) & -S(5\pi/3)S(-\pi/4) \\ S(\pi/4) & C(\pi/3)S(-\pi/4) & -C(2\pi/3)S(\pi/4) & -S(-\pi/4) & C(4\pi/3)S(\pi/4) & C(5\pi/3)S(-\pi/4) \\ -C(\pi/4) & -C(-\pi/4) & -C(\pi/4) & -C(-\pi/4) & -C(\pi/4) & -C(-\pi/4) \end{bmatrix}.$$

The singular value decomposition of both of the Magic Wrist coil configuration \mathbf{A} matrices is the same:

$$\mathbf{s} = [1.7321 \ 1.7321 \ 1.2247 \ 1.2247 \ 1.2247 \ 1.2247].$$

This singular value decomposition shows that there are only small differences in force or torque generation efficiency in different directions, as the ratio of the smallest singular value to the largest is only $1.2247/1.7321$ or 71%.

To fit on a the flotor hemisphere, each coil must lie completely below the hemisphere’s rim or equator. To account for the width of each coil, the candidate hemisphere coil configurations that were evaluated in the design process placed each coil center at at least 22.5° below the rim of the flotor hemisphere. For each of the arrangements described below, the coils are equally spaced around the circumference of the hemisphere and the long axes of the coils alternate between horizontal (latitude) and vertical (longitude) orientations. The singular values obtained for each configuration are given with an evaluation of each configuration.

Configuration 1: Each coil center located 45° below the hemisphere equator rim:

$$\mathbf{s} = [2.0908 \ 2.0908 \ 1.2247 \ 1.2247 \ 0.3587 \ 0.3587].$$

This configuration is inefficient for vertical forces and for torques about x and y .

Configuration 2: Horizontal coils 22.5° below the rim and vertical 45° below:

$$\mathbf{s} = [1.9081 \ 1.9081 \ 1.6002 \ 1.224 \ 0.573 \ 0.5734].$$

Reasonably efficient configuration in all directions; coils can be packed together well on the flotor hemisphere.

Configuration 3: Horizontal coils at 22.5° and below the rim and vertical coils 60° below:

$$\mathbf{s} = [2.0247 \ 2.0247 \ 1.6002 \ 0.8660 \ 0.4953 \ 0.4953].$$

Coils pack together even better but configuration is less efficient and balanced, and magnets near hemisphere pole may restrict motion of handle.

3.1.4 New Selected Actuator Configuration

The third hemisphere coil configuration described in the previous section was selected since it enables using almost the entire area of the flotor hemisphere for actuation, and the ratios between the singular values of the \mathbf{A} matrix described previously are not greater than 2:1, so the configuration can generate forces and torques reasonably well in all directions. Actually, it is advantageous to have a actuator configuration with greater force capabilities in the vertical direction in order to be able to better support the weight of the flotor and the user's hand.

This tightly packed configuration results in actuation force efficiency, large ranges of translation and orientation, and low inertia for a magnetic levitation device. The hemisphere area is used as much as possible for actuation, in order to maximize the force to inertia ratio of the device and consequently its motion bandwidth.

The actuator magnets and coils are arranged with three actuators next to the hemisphere rim centered at 22.5° below the hemisphere rim and the other three centered at 45° below the rim and rotated by 90° . The positions and orientations of the six coil centers on the hemisphere are listed in Table 3.1 and pictured in Fig. 3.10, where θ is the angular position from the x axis along the rim, ϕ is the angle down from the hemisphere rim, and ψ is the angle between the long axis of the coil and the hemisphere rim. The configuration of the

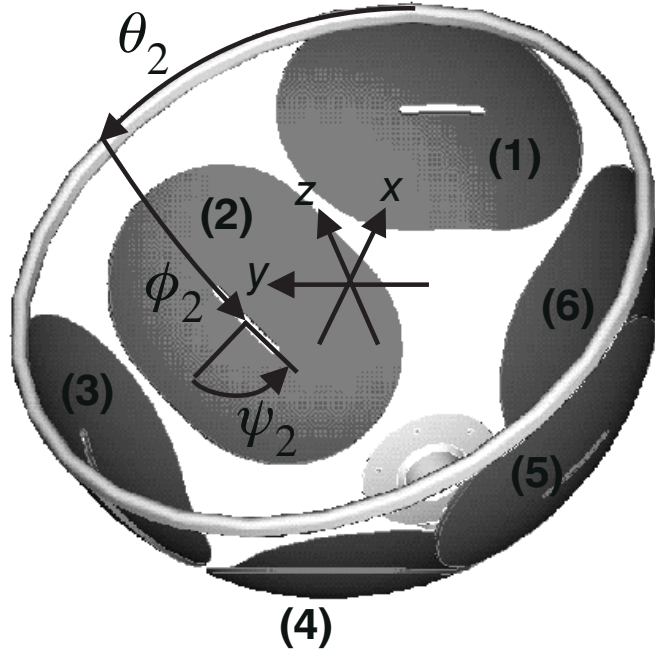


Figure 3.10: New Flotor Coil Configuration

Coil	Theta	Phi	Psi
1	0°	22.5°	0°
2	60°	45°	90°
3	120°	22.5°	0°
4	180°	45°	90°
5	240°	22.5°	0°
6	300°	45°	90°

Table 3.1: Coil Positions on Flotor Hemisphere

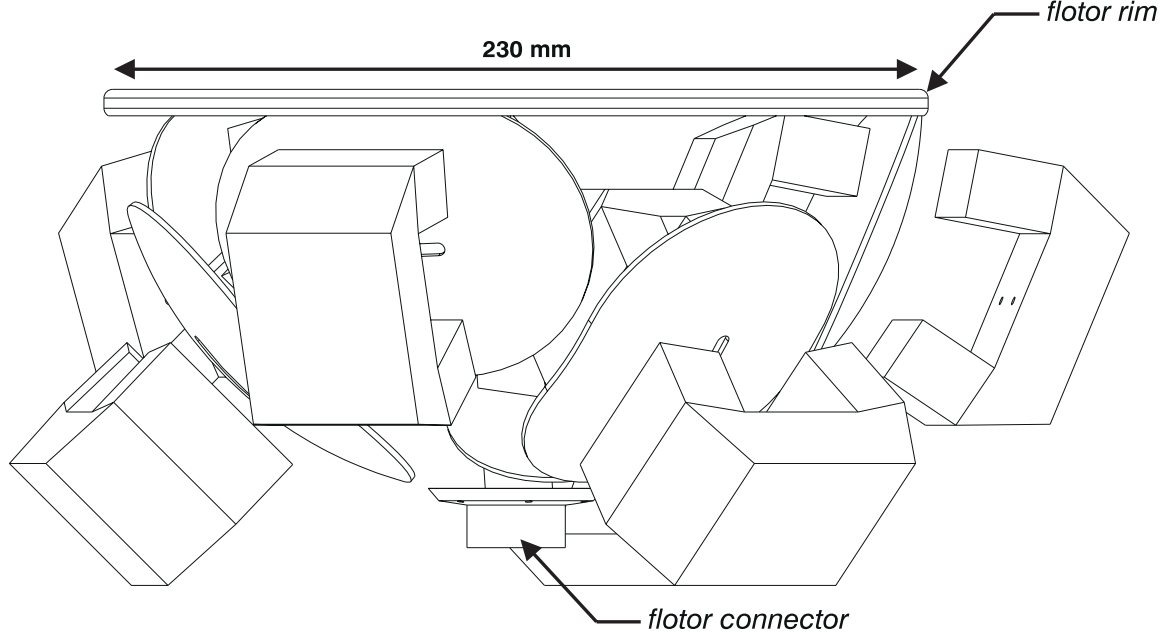


Figure 3.11: Configuration of Six Actuators

six actuators for the new device including the magnet assemblies is shown in Fig. 3.11. The flotor hemisphere rim and the connector at the hemisphere pole are also shown for clarity.

The described actuator configuration results in the transform matrix shown below when the actual values for the magnetic field \mathbf{B}_i , coil locations \mathbf{C}_i , and coil wire lengths \mathbf{l}_i from the next section are substituted for the normalized values used previously:

$$\mathbf{A}_{new} = [7.2 \ 7.2 \ 7.2 \ 0.83 \ 0.83 \ 0.83] \times \begin{bmatrix} -S(-\pi/8) & -S(\pi/3) & -C(2\pi/3)S(-\pi/8) & 0 & -C(4\pi/3)S(-\pi/8) & -S(5\pi/3) \\ 0 & C(\pi/3) & -S(2\pi/3)S(-\pi/8) & -1 & -S(4\pi/3)S(-\pi/8) & C(5\pi/3) \\ C(-\pi/8) & 0 & C(-\pi/8) & 0 & C(-\pi/8) & 0 \\ 0 & -C(\pi/3)S(-\pi/4) & S(2\pi/3) & S(-\pi/4) & -S(4\pi/3) & -C(5\pi/3)S(-\pi/4) \\ -1 & -S(\pi/3)S(-\pi/4) & C(2\pi/3) & 0 & C(4\pi/3) & -S(5\pi/3)S(-\pi/4) \\ 0 & -S(-\pi/4) & 0 & -S(-\pi/4) & 0 & -S(-\pi/4) \end{bmatrix},$$

which evaluates to the following values:

$$\mathbf{A}_{new} = \begin{bmatrix} 2.7553 & -6.2354 & -1.3777 & 0 & -1.3777 & 6.2354 \\ 0 & 3.6000 & 2.3862 & -7.2 & -2.3862 & 3.6 \\ 6.6519 & 0 & 6.6519 & 0 & 6.6519 & 0 \\ 0 & 0.2927 & 0.7171 & -0.5855 & -0.7171 & 0.2927 \\ -0.8280 & 0.5070 & 0.4140 & 0 & 0.4140 & -0.5070 \\ 0 & 0.5855 & 0 & 0.5855 & 0 & 0.5855 \end{bmatrix},$$

and

$$\mathbf{s} = [11.5215 \ 9.4984 \ 9.4984 \ 1.0141 \ 0.6867 \ 0.6867].$$

This \mathbf{s} vector indicates that the maximum horizontal forces are 82% of the maximum vertical force and that the maximum torques about the horizontal axes are 67% of the maximum torque about the vertical axis.

3.2 Single Actuator

The individual actuators for the magnetic levitation device must be shaped to all fit together in the flotor and stator bowls. The magnet assemblies must be designed to generate magnetic fields as large and uniform as possible in the desired areas of the coils for actuation. A 3-D finite element analysis software package² was used for the design of the new magnetic assemblies.

The inner and outer magnet assemblies in the new actuators have different widths and the coil is spherically curved in order to conform to the overall hemispherical shape of the new device. Due to the curvature of the coil and the tapered shape of the magnet assemblies, the forces generated from the two sides of the coil are not in line with each other, although the vertical components of the generated forces cancel each other out. As a result, the net force generated from this actuator is slightly smaller than the force that would be generated from straight vertical magnetic fields and a perfectly flat coil of equal area, although the torque generated around the hemisphere center is greater than the one that would be generated from a flat coil.

3.2.1 Coils

Each oval-shaped actuator coil is curved to conform to the flotor shell and spans a $45^\circ \times 61^\circ$ solid angle on the hemisphere. The six oval-shaped flat actuator coils together occupy approximately two-thirds of the flotor area. The actuator coils for the maglev device were wound from ribbon wire with approximately 310 turns for each coil. Ribbon wire with a cross section of 0.12×1.84 mm was used so that the resulting wound coils would have the current density sufficient to provide high desired actuation forces and the electrical impedance suited for the current amplifiers used.

The first flotor for this magnetic levitation device was made using copper ribbon wire and had a mass of 850 grams including the flotor handle. The resistance of each copper

²Maxwell 3-D, Ansoft, Pittsburgh PA.

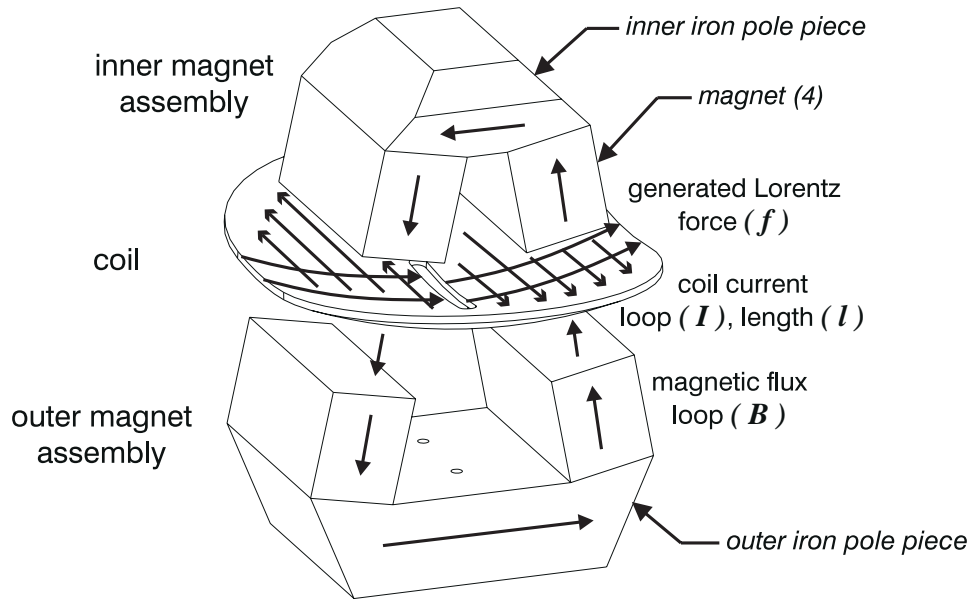


Figure 3.12: Single Actuator with Magnet Assemblies and Suspended Oval Coil

coil was approximately 5.5Ω with an inductance of approximately 0.3 mH . To reduce the flotor mass, a second flotor was made from a set of coils wound from aluminum ribbon wire. The resulting flotor mass was approximately 600 grams and the resistance of each coil was approximately 8Ω .

3.2.2 Finite Element Analysis Models

Magnet assembly design was carried out using the Maxwell-3D electromagnetic field finite element analysis software package from Ansoft. Full three-dimensional FEA models were used rather than a simplified 2D one due to the high energy product of the permanent magnets and the large size of the air gap relative to the size of the magnets, which may result in large stray fringing fields around the magnet assemblies.

To calculate the magnetic fields around a magnet assembly, the analysis software calculates an initial mesh of polyhedron volumes and then makes multiple passes to adaptively refine the mesh in locations where the field gradients are large. Particular areas of interest can be selected by the user for additional mesh refinement. The software can also calculate resulting electromagnetic forces on different elements in the simulation. Model shapes and material properties can be formulated by the user or selected from standard libraries. One set of sample execution runtime data is shown shown in Table 3.2.

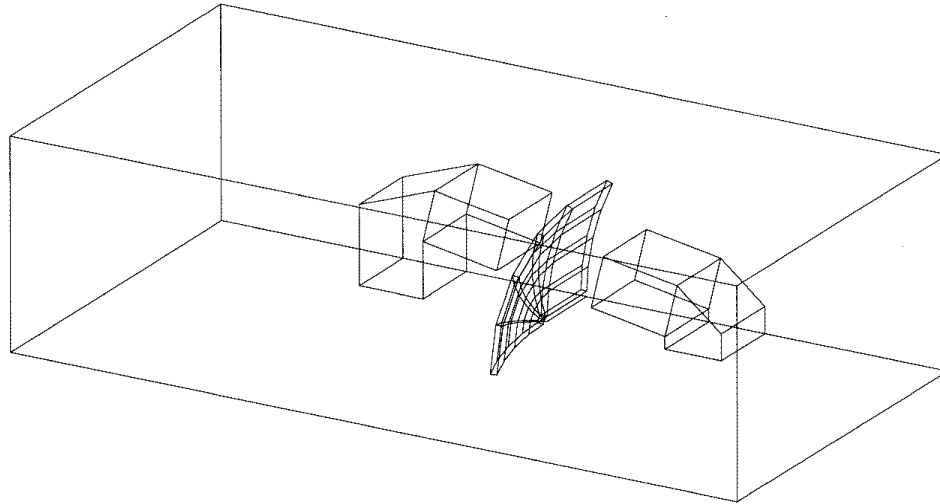


Figure 3.13: FEA 1/4 Model of Magnet Assembly and Coil in Control Volume

An example FEA model used for analysis of magnetic fields and Lorentz forces is shown in Fig. 3.13. The model contains one-quarter of the actual magnet assembly; the assembly and fields are mirrored across the rear and bottom sides of the volume shown in the figure. The boundary conditions of the model volume are specified so that the magnetic field lines run normal when they intersect planes of mirror symmetry and run parallel to all other boundary planes. Results of the finite element analysis for sample magnet actuator designs are given in Section 3.2.4.

3.2.3 Magnet Assembly Design

The approximate size and shape parameters of the magnet assemblies such as the taper angle, separation, and cross-section of the magnet face were determined from the size of the flotor hemisphere coils and the desired motion range of the device. The magnet assemblies are tapered so that the magnet assemblies on the inner stator can fit together more closely, but the outside magnet assemblies can be made larger so that the fringing magnetic fields are reduced. The separation distance and taper angle between the two pairs of magnets in each assembly was determined by the size of the actuator coils and the range of motion of the flotor. The arrangement of the coil and magnet cross-section shapes are shown in Fig. 3.14 as they would appear projected onto a flat surface, since the coil actually conforms to the shape of the flotor hemisphere. The air gap between the inner and outer magnet

Maxwell 3D Field Simulator Version 3.0.12 beginning m25mm_c on
 trombone.msl.ri.cmu.edu at 11/19/95 23:08:32

Command/File	Real Time	Cpu Time	Mem/File Size	Number of Elements
SEEDED MESH				
mesh_seed	00:00:34	00:00:33	3231K	2554 tetrahedra

Finished m25mm_c on trombone.msl.ri.cmu.edu in 00:00:35 at 11/19/95 23:09:07

Maxwell 3D Field Simulator Version 3.0.12 beginning m25mm_c on
 trombone.msl.ri.cmu.edu at 11/19/95 23:09:53

Command/File	Real Time	Cpu Time	Mem/File Size	Number of Elements
ADAPTIVE PASS 1				
adapt	00:01:12	00:01:03	10269K	2554 tetrahedra
ADAPTIVE PASS 2				
mesh_adapt	00:00:31	00:00:29	4034K	3619 tetrahedra
adapt	00:20:49	00:20:06	14245K	3619 tetrahedra
ADAPTIVE PASS 3				
mesh_adapt	00:00:26	00:00:24	5039K	5091 tetrahedra
adapt	00:03:27	00:03:15	18441K	5091 tetrahedra
ADAPTIVE PASS 4				
mesh_adapt	00:00:35	00:00:34	7864K	7152 tetrahedra
adapt	00:07:06	00:06:54	25143K	7152 tetrahedra
Solution process is completed successfully				

Finished m25mm_c on trombone.msl.ri.cmu.edu in 00:35:04 at 11/19/95 23:44:57

Table 3.2: Finite Element Analysis Runtime Information

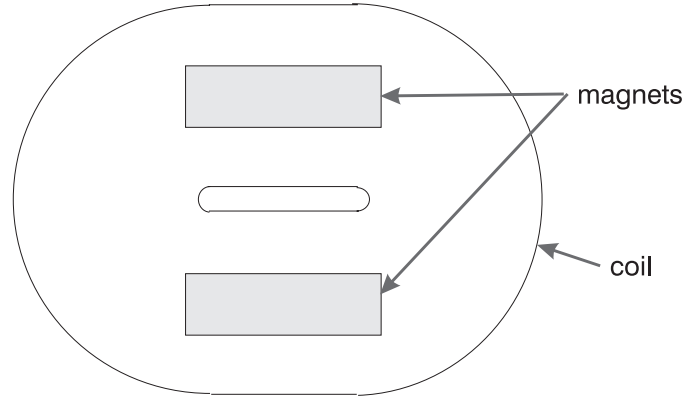


Figure 3.14: Coil and Magnet Shapes Projected onto a Plane

assemblies is 32 mm to accommodate the thickness of the flotor, its range of motion, and its curvature.

3.2.4 FEA Modeling Results

The exact shape and dimensions of the magnet assemblies were designed with the aid of the finite element analysis software. The goal of the design was to produce magnet assemblies with the highest and most uniform magnetic fields in the air gaps between the magnet faces and which would fit in the available space on the stator hemispheres. A high magnetic field in the air gaps enables high Lorentz forces to be produced on the flotor coils with low currents and heat dissipation, and a uniform magnetic field causes the Lorentz force to remain constant and independent of the position of the coil provided that the coil remains in the magnet gap.

Four different models were evaluated with different magnet thicknesses. Magnet thicknesses of 20 mm, 25 mm, and 30 mm were analyzed, also one model was analyzed with 30 mm magnets on the inside assembly and 45 mm magnets on the outside. Figs. 3.15-3.18 show the calculated magnetic fields in a plane through the center of the magnets. The magnetic field lines in the plots represent field magnitudes of 0.25 T to 0.75 T in increments of 0.025 T. The calculated forces generated on a coil in the center of the magnet assembly air gaps are shown in Table 3.3. The 1/4 model of the coil is modeled in two parts; the straight part is the part between the magnets and the curved part is the circular part on the ends of the coil.

From the data in Table 3.3, it can be seen that the 25 mm magnet thickness actually generates the greatest Lorentz force in the Z direction. The Lorentz forces in the X and Y directions would be canceled out in a full model of the actuator assembly due to the

Magnet Thickness:	20 mm magnets	25 mm	30 mm	30 and 45 mm
X force: straight	0.00784 N	-0.00093 N	0.00792 N	0.00843 N
curve	-0.346 N	-0.416 N	-0.389 N	-0.425 N
Y force: straight	0.801 N	0.917 N	0.811 N	1.15 N
curve	0.150 N	0.317 N	0.278 N	0.433 N
Z force: straight	2.97 N	3.09 N	2.96 N	3.13 N
curve	0.767 N	0.902 N	0.839 N	0.765 N
Total Z force:	3.74 N	3.99 N	3.80 N	3.90 N

Table 3.3: Lorentz Forces from Coil Currents in FEA Models

fourfold symmetry of the model. The magnet assembly model with 30 mm and 45 mm magnets generates nearly the same force as the 25 mm model, but requires much more magnet material. The additional magnet thickness in the models of Fig. 3.17 and Fig. 3.18 actually generate less force than the Fig. 3.16, since the larger magnets generate more magnetic field fringe leakage. The Fig. 3.16 model with the 25 mm magnets was selected for fabrication of the magnetic levitation device.

The magnet sizes in the final design are $20 \times 50 \times 25$ mm, to provide a field strength of 0.35 T at the center of the air gap using limited magnet material. A current of 2 A through each of the three rim coils when the flotor is in the center position was predicted to result in 60 N of vertical force.

Because of the large air gap in each magnet assembly, the magnetic field varies from 0.2 T at the edge of the gap, to 0.35 T at the center, to 0.6 T by the faces of the magnets, as seen in Fig. 4.5. Due to this variation, the Lorentz force obtained from the coil current is dependent on the position of the coil in the air gap. The force increases as the coil approaches the faces of the magnets and decreases as it moves sideways so that less of the coil is exposed to the areas of high magnetic field. The variation of force with position is averaged over the entire area of all six coils and is not significantly perceptible to the user, so it will be neglected during feedback control of the device.

3.2.5 Design Verification

The magnitude of the magnetic field in the the gap between the magnet assemblies in the plane parallel to and equidistant from the faces of the two magnets is shown in Figs. 3.19 and 3.20, where the plot shown in Fig. 3.19 is from the FEA analysis and Fig. 3.20 is from actual gaussmeter measured values from the test magnet assembly. Each contour plot shows

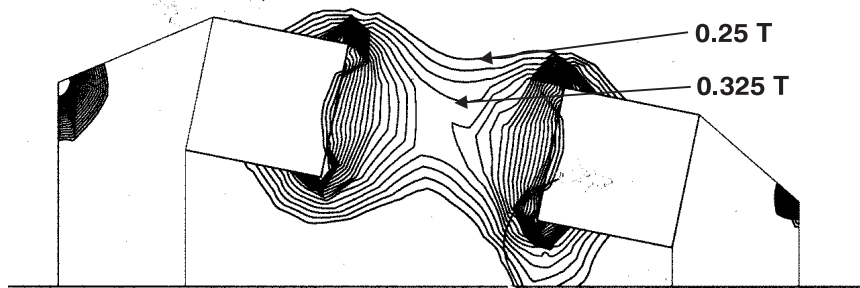


Figure 3.15: FEA Magnetic Field in Gap, 20 mm magnets

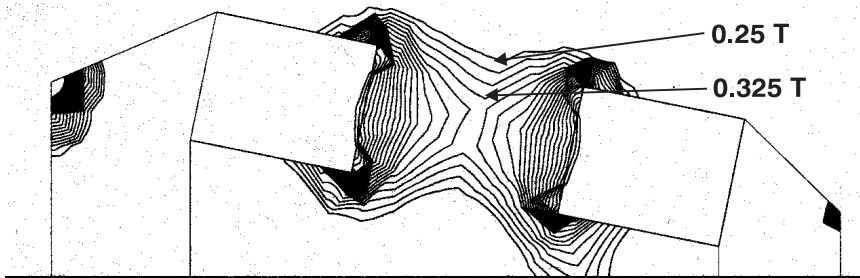


Figure 3.16: FEA Magnetic Field in Gap, 25 mm magnets

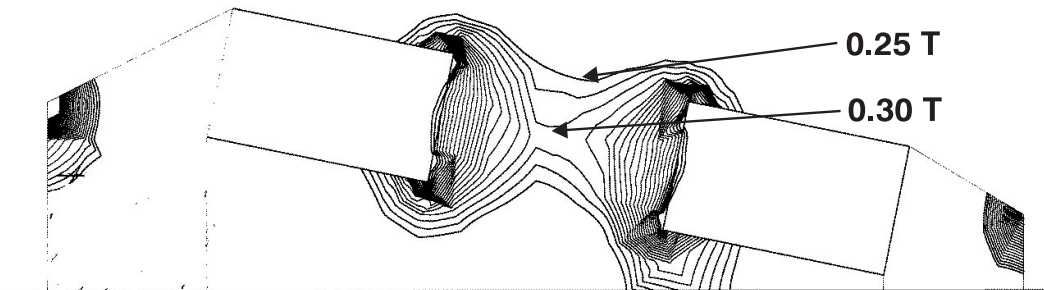


Figure 3.17: FEA Magnetic Field in Gap, 30 mm magnets

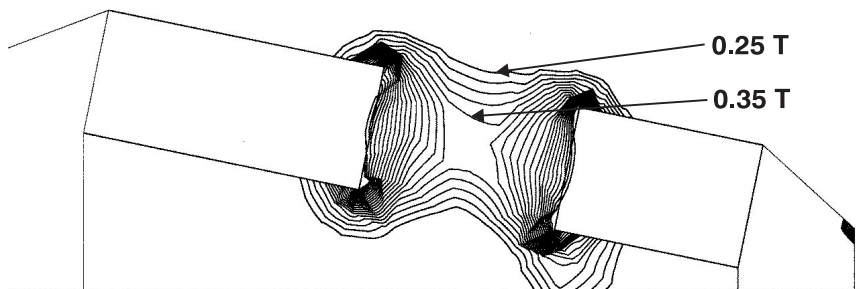


Figure 3.18: FEA Magnetic Field in Gap, 30 and 45 mm magnets

the magnetic field magnitude from the left half of a single magnet gap and the top, bottom, and left outlines of the plots correspond to the edges of the magnets.

The prototype test actuator pictured in Fig. 3.21 generated 7.2 N/A rather than 8.0 N/A as predicted by the FEA results with the sample coil centered in the magnetic field gap. The 10% difference may be attributable to FEA errors and a coil winding less dense than expected.

3.3 Comments

The steady-state vertical forces necessary to counteract the weight of the levitated flotor are distributed over the three coils next to the hemisphere rim so that the heat generated by the coil resistances is more easily dissipated. Heat dissipation in the actuator coils was a primary concern in the device design since the operating temperature of the coils is the limiting factor for the maximum steady-state forces exertable by the device. Due to this concern, the outer stator bowl was designed with air vents and thermistors were embedded in the first set of flotor coils so that coil temperatures could be monitored during operation.

During preliminary testing of the finished device with the first flotor, however, only 4.5 W of power was required to cancel gravity on the flotor and the coil temperatures only slightly increased above room temperature. Since heat dissipation is less critical than anticipated, aluminum ribbon coils have become a more attractive option. Although aluminum has 50% more resistivity than copper, its density is only 30% that of copper. Since most of the mass of the flotor is in the coils, an aluminum coil flotor has approximately half of the mass of the copper coil flotor, the power required to lift the flotor is reduced, and the motion bandwidth of the entire device is be considerably increased. If the maximum position control bandwidth of the device is proportional to $1/\sqrt{m}$, then the bandwidth would be increased by $\sqrt{2}$ or 41%.

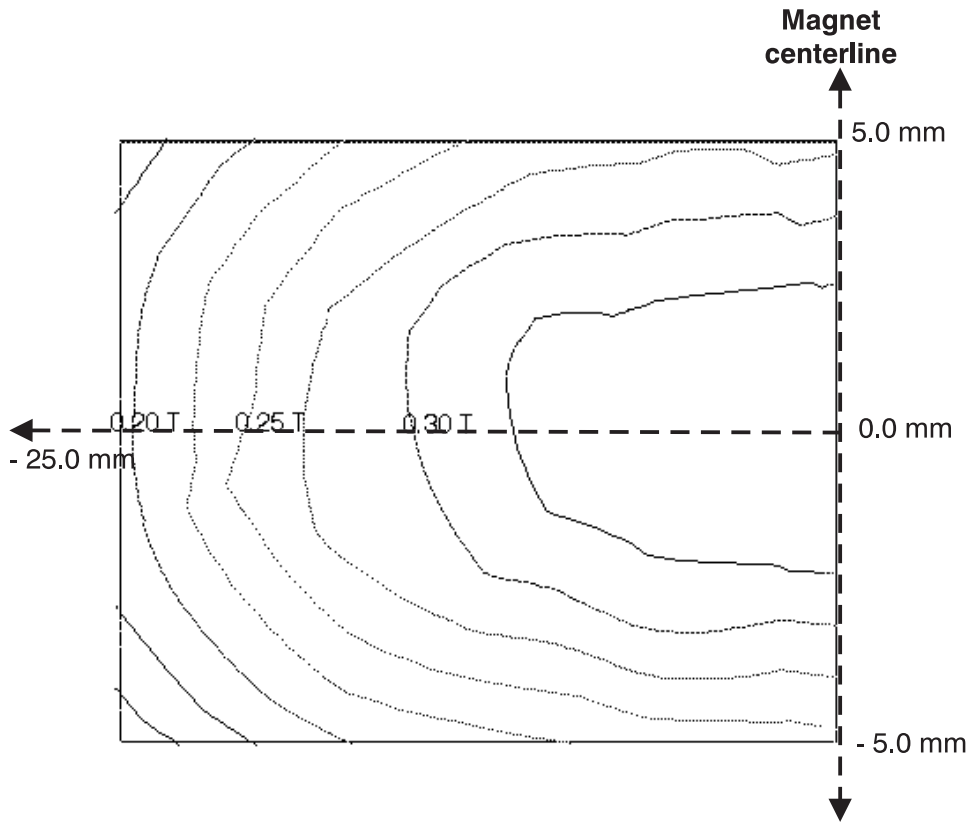


Figure 3.19: FEA Predicted Magnetic Field in Gap

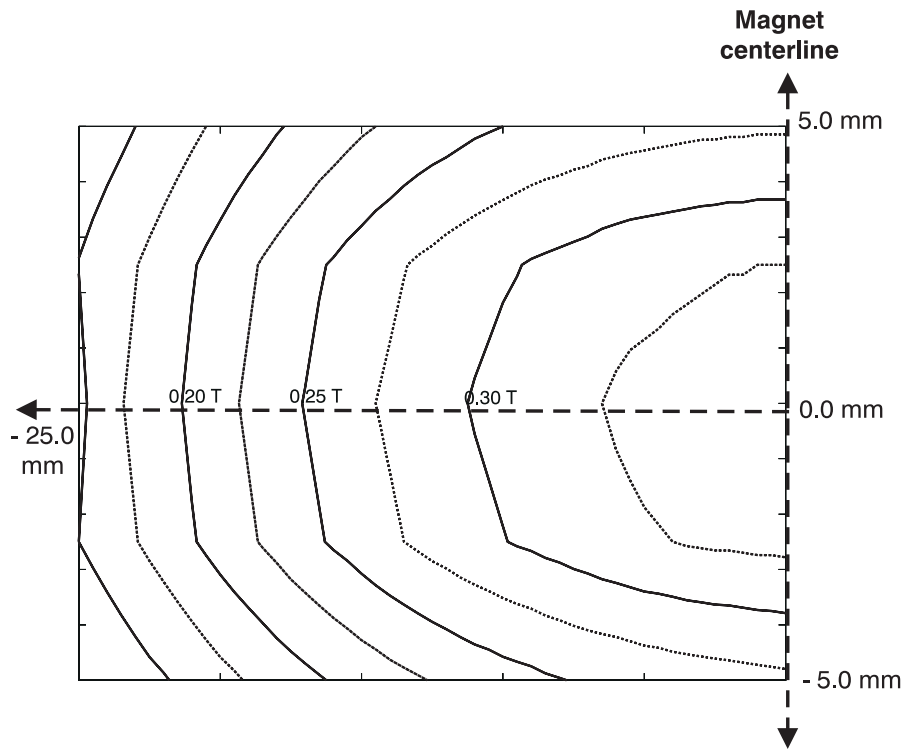


Figure 3.20: Measured Magnetic Field in Gap

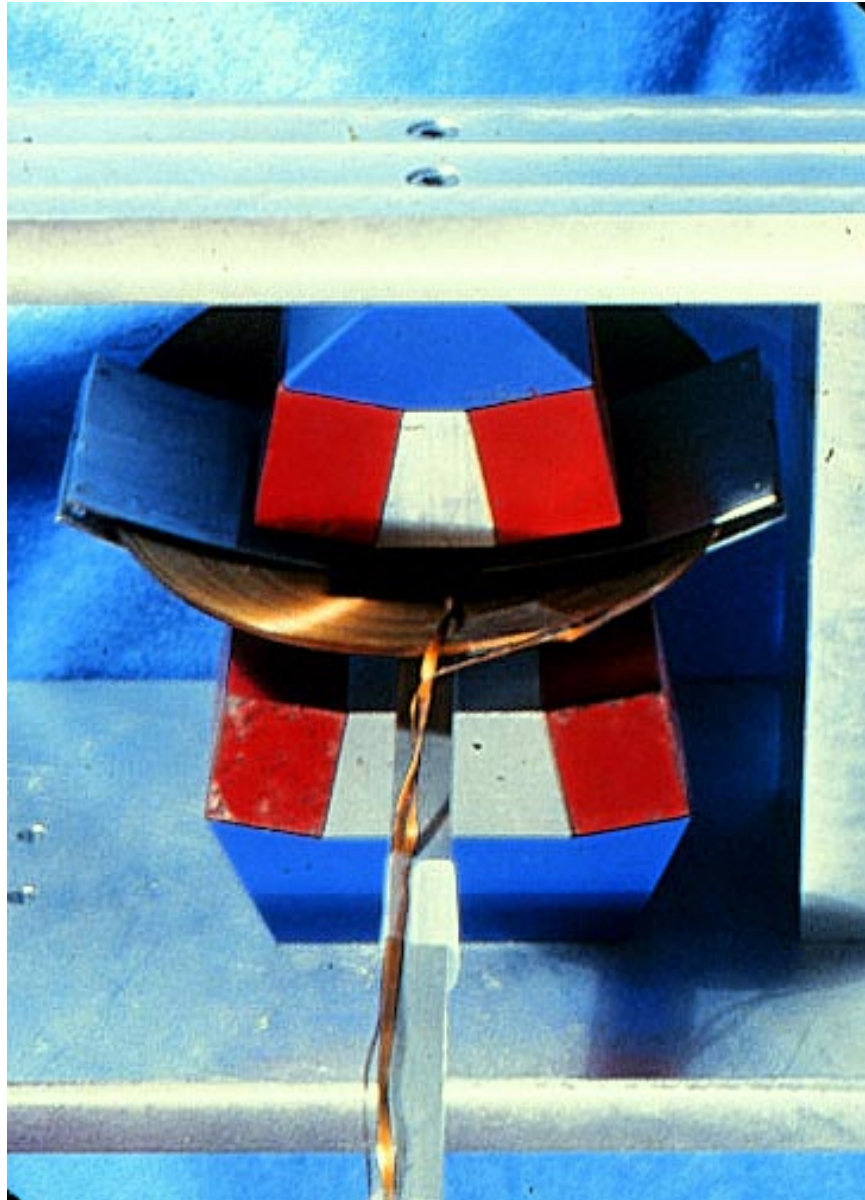


Figure 3.21: Test Actuator

Chapter 4

Position Sensing

Position sensing for the new magnetic levitation device, as with all previous Lorentz levitation devices, is optical and involves no contact between the device flotor and stator. On the new magnetic levitation haptic device, position is sensed by three lateral effect planar position sensitive photodiodes (PSDs) with lenses on the fixed outer stator which measure the positions of light spots from three LEDs mounted on the moving flotor. These sensors provide six independent variables (x and y on each sensor) which together determine the position and orientation of the flotor. The three sensors are equidistant from the stator center and mutually orthogonal to maximize position accuracy and simplify the geometric calculations. The sensor and LED configuration is shown in Fig. 4.1.

Due to the larger range of motion of the new device compared to the Magic Wrist and the UBC Wrist, a new position sensing scheme was designed. Instead of using narrow-beam LEDs shining directly onto the position sensing photodiodes, wide-beam LEDs on the moving flotor are imaged onto the photodiodes by fixed demagnifying lenses. Photodiodes with the size and signal resolution necessary to accommodate the range of motion of the new device with narrow-beam LEDs could not be obtained. Since the largest commercially available position sensitive photodiodes have an active area smaller than the range of motion of this device, direct sensing LED light beams was not possible.

4.1 Sensor Assemblies

Sensor assemblies containing a lens, sensor, housing and a sum-and-difference signal conditioning circuit to be fastened to the outer stator bowl were designed by Zack Butler. A lens demagnification ratio of 1:2.5 at the nominal flotor position maps the full range of motion of the LED onto the limited area of the sensor. The sensor assembly and the light path of the LED is shown in Fig. 4.2. The xy spot position on the sensor determines the angles

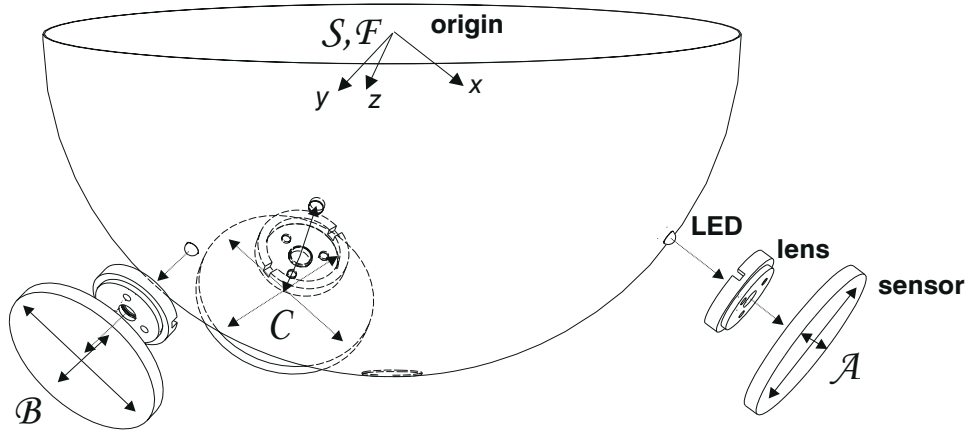


Figure 4.1: Sensor configuration and Coordinate Frames

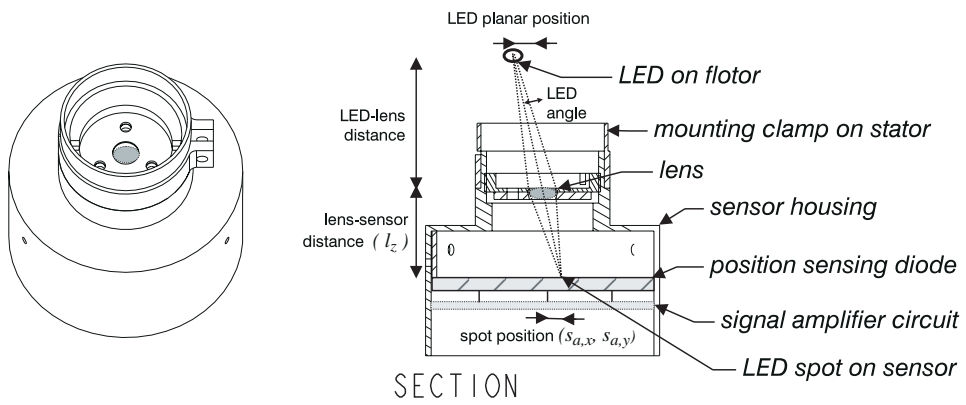


Figure 4.2: Sensor Housing Assembly

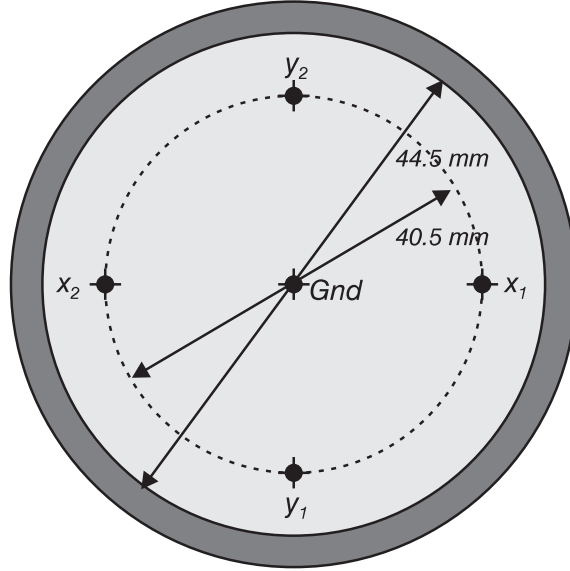


Figure 4.3: UDT SC-50D Position Sensitive Photodiode, Rear View

between the light path and the lens axis, but the LED-lens distances must be calculated from all the sensor signals in combination.

Planar position sensing photodiodes operate by a lateral photoelectric effect. In a tetralateral PSD, light striking the sensor surface generates electric charges which flow to four electrodes on the sides of the device. The charge flowing to each electrode is approximately inversely proportional to the distance from the electrode to the light spot. The position of the centroid of the light spot on the sensor can thus be approximately calculated as follows [47]:

$$\frac{x_{pos}}{L} \cong x_{signal} = \frac{x_1 - x_2}{x_1 + x_2}, \text{ and} \quad (4.1)$$

$$\frac{y_{pos}}{L} \cong y_{signal} = \frac{y_1 - y_2}{y_1 + y_2}, \quad (4.2)$$

where L is the distance from the center to each electrode and x_1 , x_2 , y_1 , and y_2 are the currents through each electrode.

The position sensitive photodiodes were obtained from UDT Sensors Inc. The active sensing area of the sensors is a 44.5 mm diameter circle and the sensing electrodes are positioned on a 40.5 mm diameter circle. The back of the sensor is shown in Fig. 4.3.

4.2 Sensor Calibration

The PSD output signals have significant nonlinearities near their edges and offsets which must be corrected to achieve accurate position sensing over the range of motion of the device. The combined effects of physical misalignment, optical misalignment, and sensor zero point offsets cause the x and y readings from each sensor to vary from zero when the flotor is in the center position. In addition, there is lens distortion and warping of the sensor signals which become more pronounced towards the edges of each sensor. The signal to noise ratio of each sensor is also critical to the positional accuracy.

4.2.1 Calibration Lookup Table

The sensors were calibrated by moving the LED using a computer controlled precision x-y stage¹ over a 20×20 grid of points spaced 2 mm apart, 15 mm above each of the three sensor assemblies and measuring the x and y sensor output at each point. The nonlinear distortion of sensor 1 and its lens assembly is shown in Figs. 4.4 and 4.5, where the planar grid of LED positions of Fig. 4.4 maps to the warped grid of sensor signal outputs shown in Fig. 4.5. The sensor nonlinearity at the extremes of this range of motion corresponds to a difference of approximately 5 mm in the LED planar position at the 15 mm height. The extreme warping at the corners of the sensor signal output grid may be due to several factors such as internal reflection of the LED light inside the sensor assembly, partial occlusion of the LED spot by the edge of the stator casting, or the light spot on the sensor partially spilling outside of the sensing area. In any case, flotor motion limits are imposed by the device controller to prevent the LED sensor light spots from moving into these areas, as described in Section 5.2.4.

The raw data from each sensor calibration procedure consists of a regular grid of LED position data mapped to a warped grid of sensor signal data. For practical use, this mapping must be inverted: a regular grid of sensor readings which maps to the corresponding LED positions in x and y is needed. The irregularly spaced sensor data is reinterpolated to create a grid of evenly spaced data to be used as a lookup table. Interpolation between the points of the sensor signal grid will then produce accurate, calibrated LED positions.

To quickly and accurately obtain LED spot positions given x and y sensor signals, a 2-D cubic reinterpolation was done on each set of calibration data to obtain an inverted mapping from a regular grid of sensor values to the corresponding inversely warped grid of LED spot

¹NEAT 310 Programmable Controller

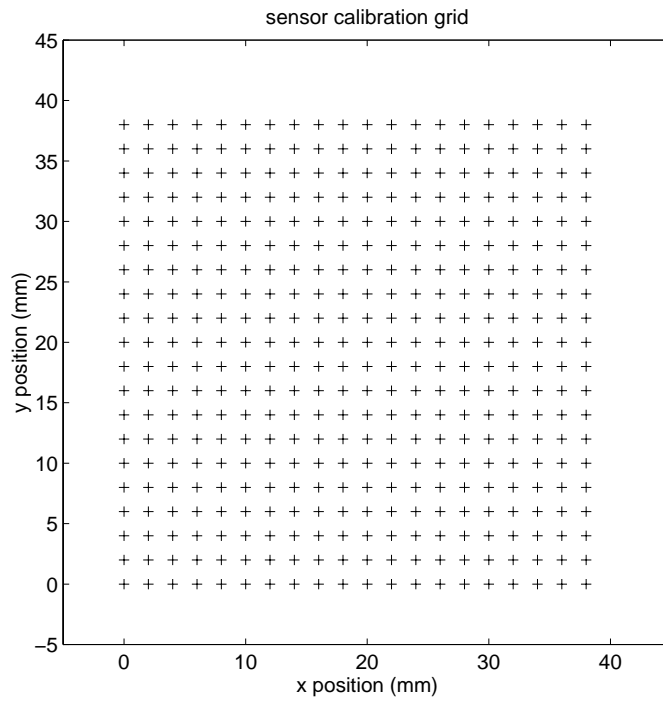


Figure 4.4: LED Position Calibration Grid

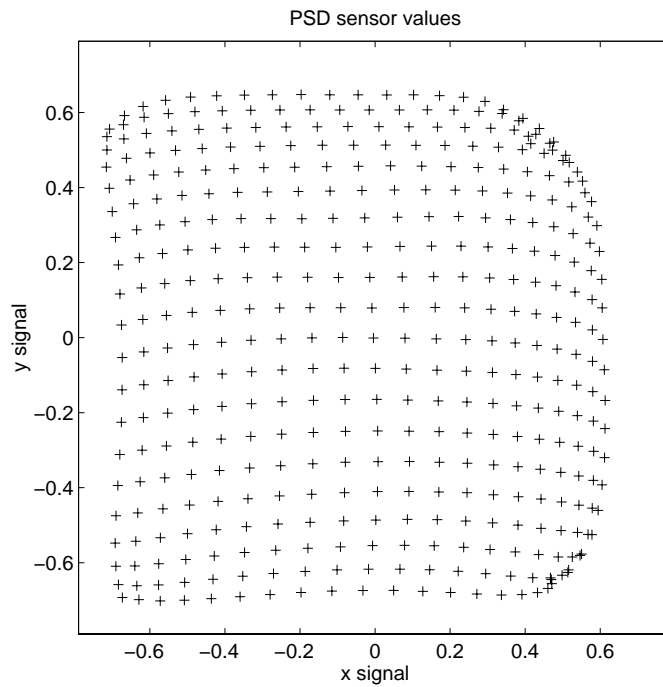


Figure 4.5: Sensor Signal Distortion

positions. The 2-D reinterpolation of the calibration data sets was done using the MATLAB `griddata` function, which uses an inverse distance 2-D reinterpolation method[48].

The reinterpolated data sets are used as lookup tables online during device operation to quickly calculate LED spot positions by 2-D linear interpolation at each sample period. Since the reinterpolated sensor values are spaced in a regular grid, the lookup tables can be indexed directly rather than with a bisecting search of indeterminate execution time. For each sensor two 2-D lookup tables are calculated, one for x values and one for y . The elements of the $m \times m$ lookup tables are referred to by $x_{i,j}$ and $y_{i,j}$ for $i < m$ and $j < m$. The indices of the lookup tables range from 0 to $m - 1$.

4.2.2 Lookup Table Interpolation

Since computation times are critical to the performance of the maglev device during operation, a fast and simple interpolation method is used to calculate corrected LED angles from the raw sensor signals using the calibration lookup tables. First, the indices i and j of the grid square in which the pair x_{raw}, y_{raw} falls are found, along with the x and y weights which correspond to the x, y position of the raw sensor data point within that grid square:

$$x_{lut} = (x_{raw} - x_{min})m, \quad (4.3)$$

$$i = \text{floor}(x_{lut}), \quad (4.4)$$

$$x_w = \text{fraction}(x_{lut}), \quad (4.5)$$

$$y_{lut} = (y_{raw} - y_{min})m, \quad (4.6)$$

$$j = \text{floor}(y_{lut}), \text{ and} \quad (4.7)$$

$$y_w = \text{fraction}(y_{lut}). \quad (4.8)$$

The lookup table indices i and j are given by the integer parts of x_{lut} and y_{lut} and the weights x_w and y_w are the fractional parts. The minimum x and y sensor signal values in the lookup tables are x_{min} and y_{min} respectively.

Then, the calibrated LED directions x_{cal} and y_{cal} are given by a simple 2-D weighted average of the lookup table values at each corner of the grid square determined by i and j :

$$x_{cal} = x_{i,j}(1 - x_w)(1 - y_w) \quad (4.9)$$

$$+ x_{i+1,j}x_w(1 - y_w)$$

$$+ x_{i,j+1}(1 - x_w)y_w$$

$$+ x_{i+1,j+1}x_wy_w, \text{ and}$$

$$y_{cal} = y_{i,j}(1 - x_w)(1 - y_w) \quad (4.10)$$

$$\begin{aligned}
&+ y_{i+1,j}x_w(1 - y_w) \\
&+ y_{i,j+1}(1 - x_w)y_w \\
&+ y_{i+1,j+1}x_wy_w.
\end{aligned}$$

This fast interpolation method is described in in *Numerical Recipes in C* [49] as bilinear 2-D interpolation. Higher order cubic or spline interpolations could be used instead for greater accuracy and smoothness of the interpolated output function, but the computation time required would increase geometrically with the order of interpolation. If greater accuracy is needed, the calibration raw data and the inverted lookup tables can be redone with more data points over the same area.

4.2.3 Lookup Table Accuracy

To estimate the accuracy of the generated lookup table, the original signal data from the sensor calibration procedure were substituted back into the interpolated lookup table procedure described above to calculate the error between the actual LED positions during calibration and the calculated LED positions from lookup table interpolation. The errors in x and y positions of the light spot over the complete area sensor area are shown in the surface mesh plots of Fig. 4.6. The interpolated lookup table procedure reduces the sensor signal distortion to under approximately 0.0015 mm within an approximately 8.0 mm radius circle around the sensor center. Beyond that radius the errors become much larger since the sensor distortion is much larger but the points in the interpolated table are still the same distance apart. The 8.0 mm sensor radius is sufficient to track the full motion range of the flotor in translation and rotation.

4.3 Sensing Kinematics

The coordinate frames referred to in the kinematics calculations are the fixed stator frame \mathcal{S} , the moving flotor frame \mathcal{F} , and sensor frames \mathcal{A} , \mathcal{B} , and \mathcal{C} at each lens. \mathcal{S} and \mathcal{F} are coincident when the flotor is at its home position, as in Fig. 4.1. Solving the position and orientation from the sensor data is referred to as the forward kinematics and calculation of the sensor data from the device position and orientation is the inverse or reverse kinematics. Whereas for a serial link manipulator, the forward kinematics is an easy and straightforward calculation and the inverse kinematics is more complicated, with this magnetic levitation device the opposite is true. Its kinematics calculations are more analogous to a parallel linkage or Stewart platform device rather than to a serial link manipulator.

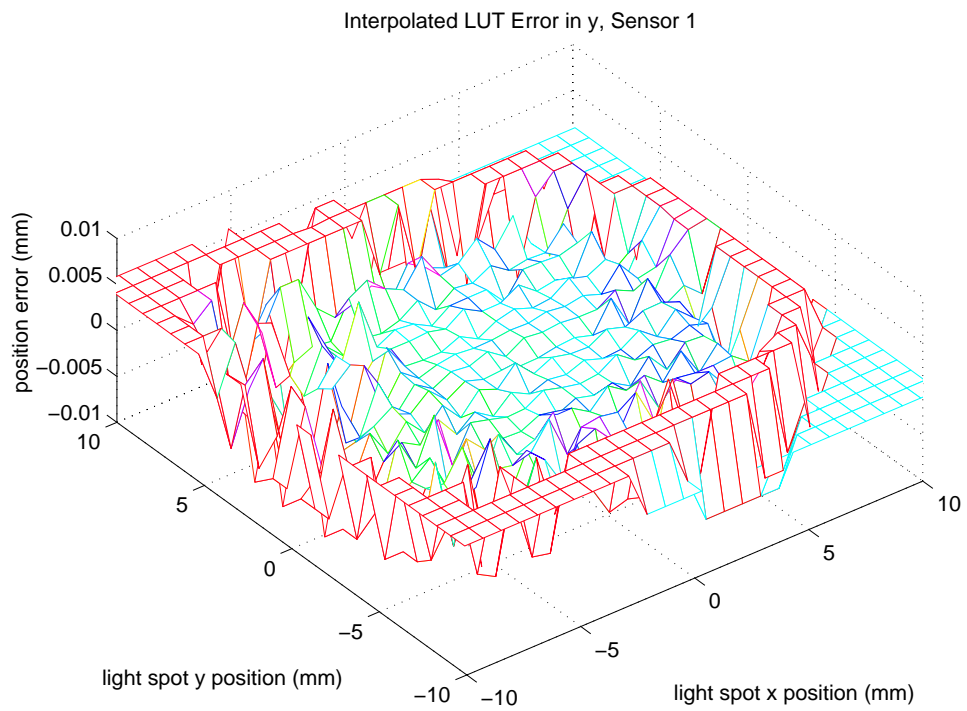
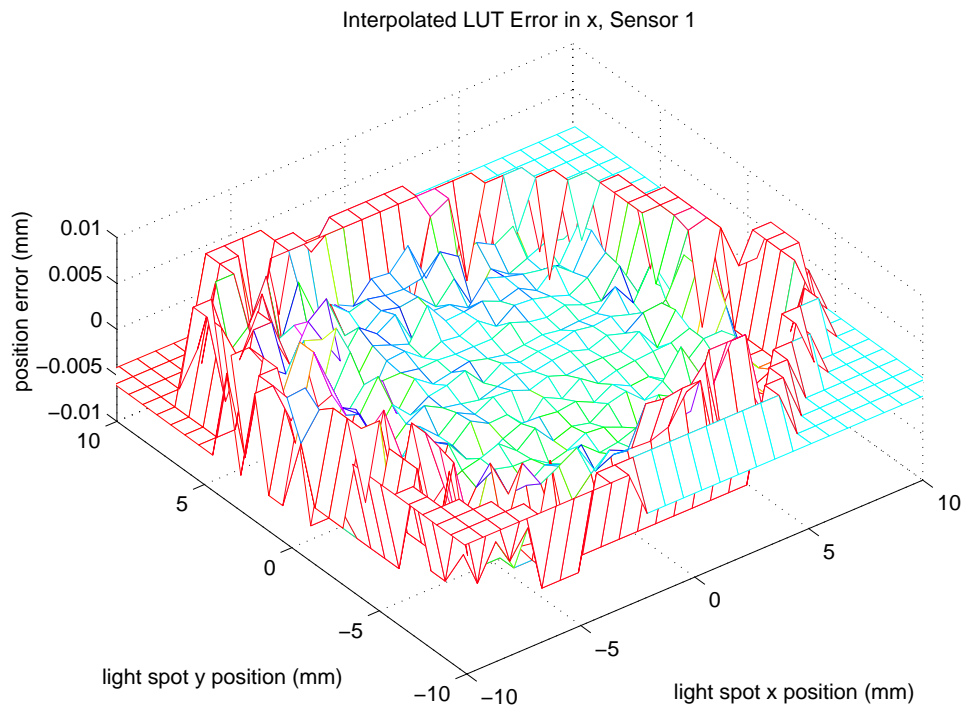


Figure 4.6: Interpolated Lookup Table Error

4.3.1 Inverse Kinematics

The inverse kinematics are computed by first finding the position of the LED in its corresponding lens frame with respect to the transformation matrix ${}^s_f\mathbf{T}$, which describes the flotor's motion. With the constant vector ${}^f\mathbf{A}$ (the location of LED \mathbf{a} with respect to the flotor frame) and the constant transformation matrix ${}^a_s\mathbf{T}$, this is given by

$${}^a\mathbf{A} = {}^a_s\mathbf{T} {}^s_f\mathbf{T} {}^f\mathbf{A}. \quad (4.11)$$

The light spot from the LED is imaged onto the sensor by the lens equation at a point given by:

$$\begin{bmatrix} s_{a,x} \\ s_{a,y} \end{bmatrix} = \begin{pmatrix} -l_z \\ {}^aA_z \end{pmatrix} \begin{bmatrix} {}^aA_x \\ {}^aA_y \end{bmatrix}, \quad (4.12)$$

where l_z is the distance from the lens to the sensor (13 mm).

The position and orientation of the flotor can be represented by a 6-element vector $[n_1\theta \ n_2\theta \ n_3\theta \ X \ Y \ Z]$, where n_1 , n_2 and n_3 represent a normalized rotation axis, θ the rotation angle about this axis, and X , Y , and Z the translational motion of the origin. The rotation angle θ can be extracted from this vector by calculating the norm of elements $n_1\theta$, $n_2\theta$, and $n_3\theta$. Creating ${}^s_f\mathbf{T}$ as a function of these variables and computing $s_{a,x}$ and $s_{a,y}$ from the equations above gives:

$$s_{a,x} = \frac{l_z l_l [n_1 n_3 (1 - \cos \theta) - n_2 \sin \theta] + Z}{l_l [n_1^2 + (1 - n_1^2) \cos \theta] + X + l_z - l_t}, \text{ and} \quad (4.13)$$

$$s_{a,y} = \frac{l_z l_l [n_1 n_2 (1 - \cos \theta) + n_3 \sin \theta] + Y}{l_l [n_1^2 + (1 - n_1^2) \cos \theta] + X + l_z - l_t}, \quad (4.14)$$

where l_l is the distance from the flotor origin to the LED (115 mm) and l_t the distance from the stator origin to the sensor (160 mm). Similar computations for the other two sensors give:

$$s_{b,x} = \frac{l_z l_l [n_1 n_2 (1 - \cos \theta) - n_3 \sin \theta] + X}{l_l [n_2^2 + (1 - n_2^2) \cos \theta] + Y + l_z - l_t}, \quad (4.15)$$

$$s_{b,y} = \frac{l_z l_l [n_2 n_3 (1 - \cos \theta) + n_1 \sin \theta] + Z}{l_l [n_2^2 + (1 - n_2^2) \cos \theta] + Y + l_z - l_t}, \quad (4.16)$$

$$s_{c,x} = \frac{l_z l_l [n_2 n_3 (1 - \cos \theta) - n_1 \sin \theta] + Y}{l_l [n_3^2 + (1 - n_3^2) \cos \theta] + Z + l_z - l_t}, \text{ and} \quad (4.17)$$

$$s_{c,y} = \frac{l_z l_l [n_1 n_3 (1 - \cos \theta) + n_2 \sin \theta] + X}{l_l [n_3^2 + (1 - n_3^2) \cos \theta] + Z + l_z - l_t}. \quad (4.18)$$

4.3.2 Forward Kinematics

No closed-form solution for inverting the sensor signal formulas could be obtained, so an iterative solution technique must be used. Two methods for solving the forward kinematics of the device were implemented; a method formulated by Zack Butler which uses a sixth-order Newton-Raphson root-finding method with a 3^6 entry lookup table of inverse Jacobian matrices computed beforehand [23], and a method by Stella Yu which reduces the sixth-order problem to a third-order problem with a rapidly converging iterative solution method [50].

Solving the forward kinematics of the magnetic levitation device occupies most of the processing time required for each control update, so the execution time of the kinematics solution is critical to the achievable control rate and the overall dynamic performance of the device. The lookup table method iterates faster, but more iterations are necessary than the rapidly converging reduced-order method to converge to within a given tolerance ϵ of the exact solution. Generally, the inverse Jacobian lookup table method and the reduced-order kinematic solution method required approximately the same amount of time. The reduced-order kinematics solution method was eventually selected as it was found to be more robust to sensor error and does not require the additional memory and initialization time to load the 3^6 Jacobian matrices of the other method.

4.3.3 Inverse Jacobian Lookup Table Method

The relations between the differential motions of a positioning device dx and its sensor signals dq are contained in the *Jacobian matrix* J , where

$$dx = J dq. \quad (4.19)$$

For the magnetic levitation device, the full Jacobian matrix can be expressed as

$$J = \begin{bmatrix} \frac{\delta(n_1\theta)}{\delta s_{a,x}} & \frac{\delta(n_1\theta)}{\delta s_{a,y}} & \frac{\delta(n_1\theta)}{\delta s_{b,x}} & \frac{\delta(n_1\theta)}{\delta s_{b,y}} & \frac{\delta(n_1\theta)}{\delta s_{c,x}} & \frac{\delta(n_1\theta)}{\delta s_{c,y}} \\ \frac{\delta(n_2\theta)}{\delta s_{a,x}} & \frac{\delta(n_2\theta)}{\delta s_{a,y}} & \frac{\delta(n_2\theta)}{\delta s_{b,x}} & \frac{\delta(n_2\theta)}{\delta s_{b,y}} & \frac{\delta(n_2\theta)}{\delta s_{c,x}} & \frac{\delta(n_2\theta)}{\delta s_{c,y}} \\ \frac{\delta(n_3\theta)}{\delta s_{a,x}} & \frac{\delta(n_3\theta)}{\delta s_{a,y}} & \frac{\delta(n_3\theta)}{\delta s_{b,x}} & \frac{\delta(n_3\theta)}{\delta s_{b,y}} & \frac{\delta(n_3\theta)}{\delta s_{c,x}} & \frac{\delta(n_3\theta)}{\delta s_{c,y}} \\ \frac{\delta X}{\delta s_{a,x}} & \frac{\delta X}{\delta s_{a,y}} & \frac{\delta X}{\delta s_{b,x}} & \frac{\delta X}{\delta s_{b,y}} & \frac{\delta X}{\delta s_{c,x}} & \frac{\delta X}{\delta s_{c,y}} \\ \frac{\delta Y}{\delta s_{a,x}} & \frac{\delta Y}{\delta s_{a,y}} & \frac{\delta Y}{\delta s_{b,x}} & \frac{\delta Y}{\delta s_{b,y}} & \frac{\delta Y}{\delta s_{c,x}} & \frac{\delta Y}{\delta s_{c,y}} \\ \frac{\delta Z}{\delta s_{a,x}} & \frac{\delta Z}{\delta s_{a,y}} & \frac{\delta Z}{\delta s_{b,x}} & \frac{\delta Z}{\delta s_{b,y}} & \frac{\delta Z}{\delta s_{c,x}} & \frac{\delta Z}{\delta s_{c,y}} \end{bmatrix}. \quad (4.20)$$

Computing this Jacobian matrix from the kinematics Equations 4.13 through 4.18 symbolically is straightforward but the resulting partial differential formulas are complicated and

would impose a prohibitively large computational burden on the control processor during operation. The complete Jacobian matrix for the magnetic levitation device is given in Appendix A.

The transformation from the sensor data to flotor position and orientation can be computed from the Jacobian matrix \mathbf{J} using a modified Newton-Raphson root-finding method in six dimensions, since the inverse kinematics from flotor position to sensor data is known analytically but there is no known closed-form solution to the forward kinematics. A six-dimensional lookup table of flotor vector with respect to sensor data at the desired accuracy would be prohibitively large. Newton-Raphson uses the numerical inverse of the Jacobian to determine the next step of the iteration with the update rule

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \mathbf{J}^{-1}[\mathbf{s} - \mathbf{f}(\mathbf{x}^{(i)})], \quad (4.21)$$

where \mathbf{s} is a vector of the known sensor values and $\mathbf{f}(\mathbf{x}^{(i)})$ is the predicted sensor values for the previous estimate of the flotor location. The system implemented for the haptic device cannot calculate the inverse Jacobians at run-time, since this is too computationally expensive, but instead will pick one out of a 3^6 entry lookup table which can be calculated beforehand. This slows the convergence slightly in terms of number of iterations required, but greatly increases the overall speed.

4.3.4 Reduced Order Beam Length Method

The fast numerical root-finding procedure formulated by Stella Yu using the motion history to obtain initial position guesses solves Cartesian position and orientation from the sensor signals in real time during device operation. Stella's method for calculating flotor position and orientation from sensor data is given in step-by-step detail in Appendix B.

We can consider a vector from each LED to the lens of its sensor. Given the direction and magnitude of each vector, the position and orientation of the flotor are uniquely determined. Since the sensors are fixed, the vectors determine the positions of the LEDs. The flotor center is a known constant distance above the plane determined by the three LEDs, and the normalized vectors from the flotor center to each LED make up a rotation matrix.

The directions of the three LED-sensor vectors are given by the sensor signals and the fixed distance between each lens and its sensor. Only the magnitudes of the three vectors are unknown, so the solution of the position and orientation of the flotor can be reduced to a third-order problem.

Let the LED-sensor unit vector directions be denoted as $\mathbf{b}_1, \mathbf{b}_2$, and \mathbf{b}_3 and the magnitudes as u_1, u_2 , and u_3 . The sensor positions are given by $\mathbf{S}_1, \mathbf{S}_2$, and \mathbf{S}_3 and the LED

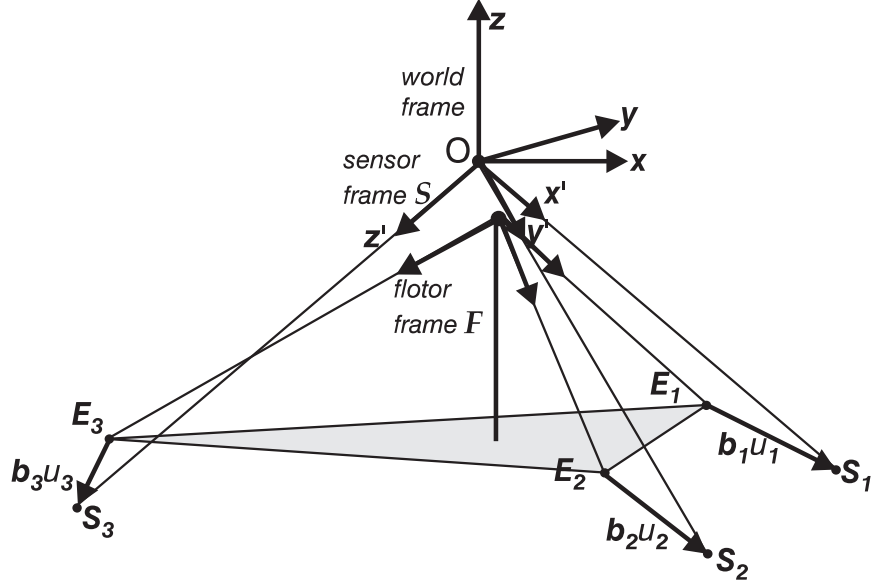


Figure 4.7: Vector Geometry For Forward Kinematics Solution

positions are \mathbf{E}_1 , \mathbf{E}_2 , and \mathbf{E}_3 . The locations of the points and vectors in this configuration are shown in Fig. 4.7.

For $i, j = 1, 2, 3$ and $i \neq j$ it can be seen from the figure that

$$\mathbf{E}_i = \mathbf{S}_i - \mathbf{b}_i u_i, \text{ and} \quad (4.22)$$

$$|\mathbf{E}_i - \mathbf{E}_j|^2 = a^2, \quad (4.23)$$

where length of each side of the triangle determined by the three LEDs is a . Substituting for each \mathbf{E}_i and \mathbf{E}_j in Equations 4.22 and 4.23 the following system of three quadratic equations with three unknowns can be obtained:

$$u_1^2 + 2g_3 u_1 u_2 + u_2^2 + 2f_{12} u_1 + 2f_{21} u_2 + d^2 = 0, \quad (4.24)$$

$$u_2^2 + 2g_1 u_2 u_3 + u_3^2 + 2f_{23} u_2 + 2f_{32} u_3 + d^2 = 0, \text{ and} \quad (4.25)$$

$$u_3^2 + 2g_2 u_3 u_1 + u_1^2 + 2f_{31} u_3 + 2f_{13} u_1 + d^2 = 0, \quad (4.26)$$

where

$$g_k = -\mathbf{b}_i \cdot \mathbf{b}_j, \quad (4.27)$$

$$f_{ij} = (\mathbf{S}_i - \mathbf{S}_j) \cdot \mathbf{b}_i, \quad (4.28)$$

and

$$d^2 = (\mathbf{S}_i - \mathbf{S}_j)^2 - a^2 > 0. \quad (4.29)$$

Although the forward kinematics for the maglev device has now been simplified to a third-order problem, a closed-form solution still could not be obtained. To obtain a solution, a two-stage numerical method is used, first obtaining an approximate solution by differentiating the equations to update u_i using the previous solution values, then improving the accuracy of the approximation by an iterative Newton gradient method.

Differentiating Equations 4.24, 4.25, and 4.26 produces the following:

$$\begin{aligned} [u_1(t) + g_3(t)u_2(t) + f_{12}(t)]du_1(t) + [u_2(t) + g_3(t)u_1(t) + f_{21}(t)]du_2(t) \\ = -u_1(t)u_2(t)dg_3(t) - u_1(t)df_{12}(t) - u_2(t)df_{21}(t), \end{aligned} \quad (4.30)$$

$$\begin{aligned} [u_2(t) + g_1(t)u_3(t) + f_{23}(t)]du_2(t) + [u_3(t) + g_1(t)u_2(t) + f_{32}(t)]du_3(t) \\ = -u_2(t)u_3(t)dg_1(t) - u_2(t)df_{23}(t) - u_3(t)df_{32}(t), \end{aligned} \quad (4.31)$$

$$\begin{aligned} [u_3(t) + g_2(t)u_1(t) + f_{31}(t)]du_3(t) + [u_1(t) + g_2(t)u_3(t) + f_{13}(t)]du_1(t) \\ = -u_3(t)u_1(t)dg_2(t) - u_3(t)df_{31}(t) - u_1(t)df_{13}(t), \end{aligned} \quad (4.32)$$

where

$$d\mathbf{b}_i(t) = \Delta\mathbf{b}_i(t) = \mathbf{b}_i(t + \Delta t) - \mathbf{b}_i(t), \quad (4.33)$$

$$d\mathbf{f}_{ij}(t) = \Delta\mathbf{f}_{ij}(t) = \mathbf{f}_{ij}(t + \Delta t) - \mathbf{f}_{ij}(t), \quad (4.34)$$

and

$$d\mathbf{g}_i(t) = \Delta\mathbf{g}_i(t) = \mathbf{g}_i(t + \Delta t) - \mathbf{g}_i(t). \quad (4.35)$$

These linear equations can be put into a matrix equation of the form $\mathbf{A}\mathbf{x} = \mathbf{y}$ with the unknown 3-vector $d\mathbf{u}$ as \mathbf{x} . This equation is solved by the efficient LU decomposition method (described in *Numerical Recipes in C* [49]) to produce the approximate solution

$$\mathbf{u}(t + \Delta t) \approx \mathbf{u}(t) + d\mathbf{u}(t). \quad (4.36)$$

This approximate solution is iteratively improved by resubstituting the approximation $\mathbf{u}(t + \Delta t)$ into the matrix \mathbf{A} and calculating the resulting error in \mathbf{y} , $\delta\mathbf{y}$. Then LU decomposition is used again to solve the equation $\mathbf{A}\delta\mathbf{x} = \delta\mathbf{y}$ and improve the solution estimate $\mathbf{u}(t + \Delta t)$. This step can be repeated as desired until the errors are within a given tolerance ϵ .

Yu's formulation and analysis of this forward kinematics solution method includes a simulator to demonstrate the performance of the iterative kinematic solution. The simulation of the flotor motion performs a random walk of the flotor position and orientation, assuming a ± 12.5 mm motion range, a $\pm 7^\circ$ rotation range, and a maximum variation between samples of 1.0%. The results of a 1000-sample simulation are shown in Fig. 4.8. The

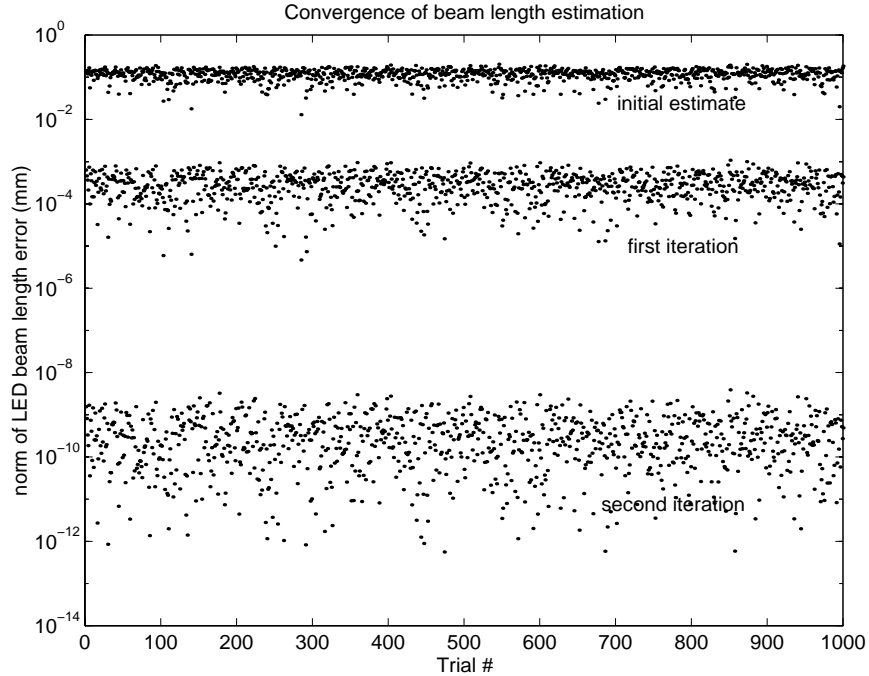


Figure 4.8: Convergence of LED-sensor Vector Lengths Solution

position estimation errors in the plot are given as the norm of the beam length errors, since this iterative method solves for the lengths of the light paths from each LED to its position sensor lens. The beam length error is an upper bound to the actual flotor position error. In this simulation, the initial estimate of the beam lengths is always within approximately 0.1 mm of the true lengths, the first iterative refinement of the solution is within approximately 0.001 mm, and the next iteration is within approximately 10^{-9} mm.

In the actual implementation the algorithm is performed twice at each cycle, once to obtain the initial estimate and then iterated once, so the “first iteration” values from the simulation test would be used, with an expected accuracy of 0.001 mm in the absence of any sensor errors. With a fixed number of iterations on each update, the execution time can be predetermined, simplifying the realtime programming of the controller.

The complexity and the computation required for this iterative solution is reduced if the sensor coordinate frame from Fig. 5.7 is used for calculations instead of the “world” frame used in the controller, since the sensor lens axes are coincident with the principal sensor frame axes. Since the origins of the sensor frame and the world frame are always coincident, to transform the flotor position vector and rotation matrix from the sensor coordinate frame to the world coordinate frame only the rotation matrix from one frame to the other is needed.

Each axis of the sensor frame forms an angle of $-\phi$ with the xy plane of the world frame, where $\phi = 35.26^\circ$. The projections of the x', y', z' sensor frame axes onto the xy plane of the world frame form angles of θ , $(\theta - 2\pi/3)$, and $(\theta + 2\pi/3)$ with the world x axis respectively, where $\theta = 30.5^\circ$. By projecting each unit axis of the sensor frame into the world frame, the rotation matrix transform from the sensor frame to the world frame can be calculated as follows:

$$R_s^w = \begin{bmatrix} \cos(\phi) \cos(\theta) & \cos(\phi) \cos(\theta - 2\pi/3) & \cos(\phi) \cos(\theta + 2\pi/3) \\ \cos(\phi) \sin(\theta) & \cos(\phi) \sin(\theta - 2\pi/3) & \cos(\phi) \sin(\theta + 2\pi/3) \\ -\sin(\phi) & -\sin(\phi) & -\sin(\phi) \end{bmatrix}. \quad (4.37)$$

4.4 Comments

The position sensing system design, calibration, and kinematic solution were performed with the aim of providing the most precise possible magnetic levitation flotor position and orientation measurements throughout the range of motion of the device at a sufficiently fast rate to enable a realtime controller to update at at least 1000 Hz.

For haptic interaction, the relative position or position sensing resolution is important rather than the absolute position accuracy. The dimension error tolerances of the device may be as large as 0.5 mm due to the difficulty of fabricating the complex geometry of the stator and flotor hemispheres. The position sensing resolution, however, is closer than $\pm 10 \mu\text{m}$, as experimentally shown in Section 7.1.5.

Chapter 5

Control

Feedback control in 6 DOF is necessary for stable magnetic levitation [3]. Gravity compensation can be done with a constant feedforward term in the controller.

The forces exerted *by* the user's hand have a relatively low bandwidth of 10-20 Hz while the forces exerted *on* the hand by the device may have frequencies up to half the control rate, so a slow lowpass filter or estimator is sufficient to track the hand motion disturbances. The motions and forces of the user's hand are treated as disturbances by the controller and state estimator.

The digital control and kinematics computations for the device are executed by a VME-bus 68060 single-board computer. Ethernet IP socket connections enable communication with a graphics workstation host. With this setup, a complex physical simulation may be executed on the workstation host while the on-board processors for the device handle only local kinematics and compliant control computations. The separation of the simulation and the device control processing allows the device to easily be connected to any different workstation executing a different simulation. A dedicated fast serial connection could be also be used as an option in the future instead of the Ethernet connection for faster, more reliable communication without network traffic interference.

5.1 Control Hardware

The components necessary for magnetic levitation are contained in a single enclosure with the maglev flotor and stator embedded in the top, amplifiers and processing, filtering and conversion electronics in the middle, and power supplies in the bottom. A VMEbus contains the control processor and analog-to-digital and digital-to-analog signal converters. Three

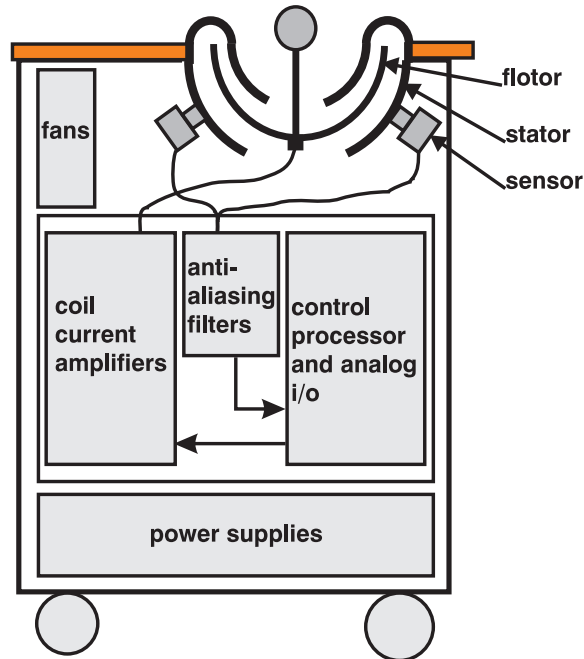


Figure 5.1: Hardware System Components in Device Enclosure

separate power supplies are used for the coil current amplifiers, the analog to digital converters, and the VMEbus and sensor electronics. The placement of the system components in the cabinet is pictured in Fig. 5.1.

5.1.1 Control Processors

The original control processor board for the maglev haptic device was a Motorola MVME162. The processor was later upgraded to MVME172 to enable faster control update rates and more complex local simulation of haptic environments. The control update rate of the Motorola MVME162 was limited to 1000 Hz while executing simple servo control and was reduced to 666 Hz while executing trajectories or multiple virtual barriers and communicating with the host workstation over an Ethernet network connection. The MVME172 can execute the device control at rates up to 1500 Hz. A realtime multitasking operating system¹ is used on the control processors.

¹VxWorks 5.3, Wind River Systems, Alameda CA

5.1.2 Analog I/O

16-bit analog to digital converters with conversion times of approximately 20 microseconds are used to digitize the position sensor signals. Since each of the three sensors outputs the sum and difference of the electrode currents for x and y directions, a total of 12 channels of A/D is needed. To reduce the total time required each cycle for analog to digital conversion, three converters are used in parallel, with one for each sensor.

A 12-bit digital to analog converter is used to send the computed coil currents from the controller to the six current amplifiers, one for each actuator coil. The remaining two channels of the 8-channel D/A converter are used to output realtime diagnostic information to an oscilloscope, if desired.

5.1.3 Current Amplifiers

PWM rack-mount servo amplifiers are used². A current limit of 4A per amplifier is imposed in the control software. A power transformer reduces 110V main current to 65V AC for two rectifier/filter cards to supply the 65V DC power supply for the amplifiers.

5.1.4 Signal Filters

Each of the 12 position sensor signals is passed through a lowpass analog antialiasing filter with a 1000 Hz cutoff frequency. The position signal cables are completely shielded and the sensor and filter electronics are inside aluminum enclosures to minimize electrical interference noise. The noise level of the sensor signals has been reduced to 2-3 units after the analog to digital conversion. The sensor noise level is shown in Section 7.1.5.

5.2 Control Software

The time required to calculate the position and orientation of the flotor in Cartesian space imposes the hardest limitation on the maximum cycle rate of the controller. Therefore, the speed of the kinematics solution is critical to the bandwidth performance of the device.

5.2.1 Digital Filtering

A second-order Butterworth filter is implemented in software. The cutoff rate of the filter is user selectable down to 1/5 of the actual sampling rate, or the unfiltered raw signals can be

²Copley Controls Corp., Waltham MA

used for control. The cutoff of the software filter can be set low for the most noise rejection or higher for greater control bandwidths. More noise rejection results in a smoother feel for the user during haptic interaction, but higher control bandwidths enable more realistic emulation of stiffer surfaces and fine shape and texture details. There is a tradeoff between noise rejection and faster control. High frequency resonances cannot be controlled if the cutoff frequency of the filter is too low. At high feedback gains, the filter cutoff frequency must be low enough to cancel most sensor noise, but not so low that uncontrollable high-frequency oscillations result in the actuator bowl.

5.2.2 Spatial Rotation Conversions

Representation of spatial rotations is not so straightforward as for translation due to the distinctive geometry of rotations in three-dimensional space. Spatial rotations are not commutative and it is impossible to represent rotations with only three variables without introducing singularities and distortions in measurement [51]. For these reasons, a number of different representations of rotation are maintained in the control routines.

The rotation matrix obtained by the forward kinematics calculation after each sample period is converted to the roll-pitch-yaw coordinate representation to be used for feedback control and is also converted to the axis-angle representation in order to apply the rotation limit described in Section 5.2.4. The roll-pitch-yaw specification describes a roll rotation r about the fixed x axis, followed by a pitch rotation p about the fixed y axis, followed by a yaw rotation y about the fixed z axis. The following conversion is used to obtain r , p , and y from the rotation matrix \mathbf{R} [51]:

$$r = \text{atan2}(R_{2,1}/\cos(p), R_{2,2}/\cos(p)), \quad (5.1)$$

$$p = \text{atan2}(-R_{2,0}, \sqrt{R_{0,0}^2 + R_{1,0}^2}), \quad (5.2)$$

$$y = \text{atan2}(R_{1,0}/\cos(p), R_{0,0}/\cos(p)), \quad (5.3)$$

where the $\text{atan2}(y, x)$ returns the correct solution of $\tan^{-1}(y/x)$ for each of the four quadrants of the xy plane. The roll, pitch, and yaw representation of spatial rotations has no singularities and minimal distortions for small rotations.

The axis-angle coordinate rotation specifies a single rotation θ in space about a unit vector $\hat{\mathbf{K}}$. The axis-angle representation $\theta, \hat{\mathbf{K}}$ is obtained from the rotation matrix as follows:

$$\theta = \arccos\left(\frac{R_{0,0} + R_{1,1} + R_{2,2} - 1}{2}\right), \quad (5.4)$$

$$\hat{K} = \frac{1}{2 \sin(\theta)} \begin{bmatrix} R_{2,1} - R_{1,2} \\ R_{0,2} - R_{2,0} \\ R_{1,0} - R_{0,1} \end{bmatrix}. \quad (5.5)$$

5.2.3 PD Control

The simplest effective means to control the levitated flotor is by proportional-derivative error feedback gains. Physical mechanisms and rigid contacts can easily be emulated with PD control since the stiffness of the environment is given by the proportional error gain and the damping or viscous friction by the derivative feedback. Since flotor velocities are not directly measured by sensors, the velocities must be calculated from the calculated position history.

A constant feedforward term can be included in the control of vertical translation to counteract the weight of the flotor. For accurate position control, the feedforward term should exactly cancel the constant control disturbance of gravity, but for haptic emulation of a tool, the simulation is more realistic if the user feels some weight in the interface device and the virtual tool can set down on a surface without drifting away into space.

It is easiest and most straightforward to represent the orientation of the flotor about a given center of compliance with the rotation angles roll, pitch, and yaw or $(\theta_x, \theta_y, \theta_z)$ as the three independent variables to be controlled. If small angles and rotational velocities are assumed, then the flotor dynamics can be modelled as 6 decoupled second order systems.

The PD control law used for each axis is:

$$f = -K_p(x_{pos} - x_{des}) - K_v((x_{pos} - x_{prev})r - v_{des}) + f_{ff}, \quad (5.6)$$

where f is the force to be generated by the actuator coils, r is the update rate, x_{pos} is the current axis position, x_{des} and v_{des} are the desired position and velocity, x_{prev} is the position from the previous update, and K_p and K_v are the proportional and velocity error gains. The feedforward term f_{ff} is generally used for gravity cancellation and can also be used for impacts and other forces in an environment simulation.

5.2.4 Motion Bounds

As the LED position markers on the flotor move increasingly far off the sensor lens axes, the amount of incident light collected by the lens and focused onto the position sensor decreases. Also, the nonlinearity of the position sensing photodiodes becomes much more pronounced as the incident light centroid moves closer to the edge of the sensor. Due to these two

phenomena, the resolution of position sensing becomes worse as the flotor translates and rotates away from the center position and towards its limits of motion. The translation of the flotor is physically limited by the magnet assemblies and stator bowls, but it is possible for the flotor to be rotated outside of the sensor ranges so that the LEDs cannot be seen by the position sensors. In the absence of position feedback, the flotor cannot be controlled.

To prevent the flotor from rotating past the sensor limits during operation, a stiff rotation limit is applied by the control software. The limit of rotation depends on the distance that the flotor is off center, since the motion of the position marker LEDs depends on both the translation and the rotation of the flotor. To allow the maximum safe rotation range throughout the translation range, the rotation limit θ_{lim} is calculated from the radial distance r of the flotor position from the center during operation:

$$\theta_{lim} = \theta_{center} - \frac{r}{r_{max}}(\theta_{center} - \theta_{edge}) \quad (5.7)$$

where θ_{center} is the maximum safe rotation when there is no flotor translation, θ_{edge} is the maximum rotation when the flotor is at the edge of its translation range, and r_{max} is the radial distance from the edge of the translation range to the center position.

5.2.5 Multiprocessing

A second processor board on the same VMEbus has been used for oversampling. During one control period, the sensor data can be sampled four times, averaged, and written to shared memory by the second board to reduce sensor noise. Higher stiffnesses and control bandwidths were found to be possible by using oversampling on a second control board, but the reliability of the system was substantially decreased due to the limited bandwidth of the VMEbus.

5.2.6 Levitation Liftoff and Landing

If the actuator amplifiers or control gains are abruptly switched on or off while the user is not grasping and supporting the flotor weight, the flotor is forced to make an abrupt transition between its rest position at the bottom of its motion range to its servo control position at the center. This sudden motion may cause the flotor shell to collide with the stator or the maximum safe rotation to be surpassed. To prevent these, software routines have been implemented to begin and end levitation in a stable and controlled manner.

To lift the flotor as smoothly as possible, it is first rotated to the zero rotation angle, then lifted to the zero position. At each of the two liftoff stages, the stiffness and damping control

gains are smoothly ramped from zero to a standard set of servo gains over approximately half a second. To land the flotor, the process is reversed.

5.2.7 Environment Emulation

To emulate dynamic interaction with solid objects in a simulated environment, the parameters of the controller must be continually updateable. For example, a fixed solid wall in the virtual environment can be represented by a unilateral constraint. The positions of the vertices of a virtual tool in the simulated environment would be determined by the position and orientation of the magnetic levitation flotor and whenever any vertex of the virtual tool penetrates the virtual wall, appropriate feedback forces and torques would be generated according to the depth of penetration and the stiffness and damping gains of the surface. Otherwise the flotor would be freely movable in all directions.

The motion of the magnetic levitation device flotor can also be coupled to an object in the simulation by introducing stiffness and damping gains in position and orientation between the flotor and the simulated object. Further details of the controller algorithms used to emulate haptic environments are described in Part II of this document.

Chapter 6

Fabrication

The spherical curvature, complex part geometry, and cost considerations posed various problems in fabrication. Many parts were machined by CMU machine shops from Pro/Engineer drawings, but several parts had to be fabricated by hand in the laboratory. All aluminum parts were black dye anodized to minimize LED reflections and protect the surface finish. Device assembly was done in the lab, requiring special care while handling the permanent magnets and other fragile parts.

6.1 Flotor

Two different flotors were fabricated for the magnetic levitation device. The first flotor was made with standard copper ribbon wire coils. After the first flotor was operational, it was found that the flotor temperature remained only slightly warmer than room temperature even during long periods of levitation. Since heat dissipation was not a problem, it was decided to fabricate a second flotor using aluminum ribbon wire coils instead of copper to reduce the flotor mass. The reduced flotor mass is easier for the user to manipulate and requires less actuation force to levitate, although the resistance of the aluminum coils is approximately 50% greater than the aluminum. The amount of material in the structure between the coils was also reduced in the second flotor to further reduce the mass.

The flotor coils were wound in the lab using a coil winder while carefully controlling the wire feed speed, angles, and tension by hand. Convex and concave curved Teflon forms were used to conform each coil to the correct spherical shape and clamp it during epoxy curing. Single component polyimide epoxy was used to set each coil together in its final shape¹.

¹Epotek, Billerica MA

Infrared LEDs with copper leads² were obtained to be used as position sensing markers. Copper leads must be used since the LEDs are in high magnetic fields and the ferrous content of typical LED leads causes very large magnetic force disturbances, reducing the stability and dynamic performance of the levitated flotor. The plastic domes on the LEDs were ground down to a flat surface just above the diode junction to make them into wide-angle point sources. The circumference of the LEDs were also ground down to a smaller diameter to fit between the coils in the flotor.

The handle parts, connector socket, and flotor rim were machined by the CMU Mechanical Engineering machine shop. The aluminum hemispheres were formed by an outside vendor³ by metal spinning on a steel mandrel. Metal spinning is a process for fabricating rotationally symmetric forms from thin sheets of metal by using a roller on a lever arm to push the metal onto a form or mandrel of the desired shape as it is spun.

The fastener, LED, and coil locations in the inner hemisphere were laid out in the lab using a (θ, ϕ) angular positioning setup, then fastener and LED holes were drilled, holes were punched for the connector and to reduce weight, and the large oval coil holes were traced and cut by hand using a power nibbler tool. The two hemispheres were bonded together with vibration damping viscoelastic polymer adhesive sheet⁴.

The coils were epoxied in place in the hemisphere with thermistors packed into their centers and the rim was attached to the hemisphere with thermally conductive epoxy. The LEDs were placed with cyanoacrylate adhesive. All flotor component wiring was done with flat ribbon wire to the connector at the pole of the hemisphere. The inside of the hemisphere was lined with adhesive bonded Teflon sheet to protect the wiring and coils from abrasion from contact with the inner stator bowl. For ease of assembly, the connector and its socket were attached to the hemisphere first, then the wiring pins were inserted into the connector from inside the hemisphere, then the handle parts were fastened to the hemisphere. The fabricated flotor is pictured in Fig. 6.1 before the Teflon lining was applied.

Thermistors were not used in the aluminum flotor since they were found to be unnecessary and cause slight magnetic disturbances due to ferrous material content.

²Hewlett-Packard HSDL-4220

³EH Schwab Co, Turtle Creek PA.

⁴3M



Figure 6.1: Fabricated Flotor

6.2 Stator

The aluminum braces which support the magnet assemblies and are bolted onto the stator bowls, the iron pole pieces in the magnet assemblies, and the sensor housing assemblies were machined from stock. The inner and outer stator bowls were first sandcasted⁵ to near-net-shape with mating surface and fastener details machined afterwards to reduce fabrication costs. The high-energy NdFeB magnets were obtained from Magnet Sales, Inc.

Stator assembly must be done carefully because of the very strong magnets that must be handled. First, the magnet assembly iron pieces were bolted to the aluminum braces. Then, each magnet was guided into its assembly. The subassembly must be rigidly clamped while one person wearing gloves carefully puts the magnet in place and the other slowly pulls out the wooden wedge separating the iron from the magnet. The inner stator magnet assemblies were then placed and bolted to the inner stator bowl while using cardboard, tape and bubble wrap to guide positioning and prevent the magnets from coming too close to each other. The magnetic field directions must be the same for each set of magnets. The outer stator bowl and the inner stator were then bolted together with the flotor between

⁵Laurel Aluminum Casting, Glassport PA



Figure 6.2: New Fabricated Lorentz Levitation Haptic Device

them. The outer stator magnet assemblies could then be placed by hand without too much difficulty. The sensor housings and upper sets of magnets must be inserted into place after the device is placed in the enclosure since they will not fit through the hole in the enclosure desktop. The fully fabricated device outside of the desktop enclosure is shown in Fig. 6.2.

6.3 Enclosure

A desktop height wheeled cabinet with a Eurocard rack was obtained from Schroff to contain all the power supplies, amplifiers, filter boards, control processing, and converters necessary for magnetic levitation. Rack-mounted PWM current amplifiers are used to power the amplifier coils. The control processing consists of a digital-to-analog and three analog-to-digital converter mezzanine cards on a VMEbus board with a single board computer on a VMEbus rack. Analog antialiasing filters were used for conditioning the sensor signals and all analog cables and electronics were shielded to reduce electrical interference noise. The original metal cabinet top was replaced with with a wood and formica top. The outer rim of the stator rests snugly on the edge of a circular hole in the enclosure top.

6.4 Comments

When the device assembly was first completed, an unforeseen problem with the first set of LEDs on the flotor became apparent: The casings of the original LEDs were slightly ferromagnetic, which resulted in additional attractive forces on the flotor as the LEDs approach the magnetic fields of the actuators. These force disturbances made the flotor motion very difficult to control stably and accurately. To eliminate this problem, non-ferromagnetic infrared LEDs were acquired and substituted for the original LEDs.

The thermistors in the first flotor were also found to be slightly ferromagnetic. Since heat dissipation was not a problem during operation of the first flotor, they were omitted from the second flotor made with aluminum coils.

Now that one magnetic levitation device of this design has been fabricated, additional devices of the same design could be produced much more quickly and at a reduced cost. Since the casting form for the stator bowls and the metal spinning mandrel for the flotor bowls have already been made, additional bowls could be produced quickly and the main time and cost to produce additional devices would be in machining the small parts of the device and the manual labor of coil winding, wiring, and assembly. A substantial economy of scale is achievable in the cost of the NdFeB magnets; the cost for each magnet was 40 percent less for magnets ordered in a set of 20 than for magnets in the first set of 4.

Some of the manual fabrication procedures required substantial trial and error, dexterity, and practice before they could be performed adequately, in particular the winding and epoxying of the actuator coils and mounting them in the flotor. The success rate of these procedures is much higher now that the difficult part of the learning curve has been overcome.

Although this magnetic levitation device is portable in that it is completely contained in a single wheeled cabinet, its bulk and weight make it difficult to be transported. The device cabinet is approximately 16 in. wide, 23 in. deep, and 28 in. high and weighs approximately 150 lbs. This size and weight could be substantially reduced in future models by minor design modifications and substitution of many of the enclosure components.

The stator castings alone weigh 10 lbs. The thickness of the stator bowl is much larger than necessary and could probably be reduced by half before introducing any significant material strain due to the attraction of each magnet assembly across the air gap. The Eurocard rack amplifiers and VMEbus components could be substituted with much smaller and cheaper parts. The VMEbus components could be replaced by a single card to be plugged in to the bus of the host computer or by a standalone embedded processor card

with mezzanine cards for analog I/O. The card rack PWM amplifiers could be replaced by H-bridge amplifiers and the 50 lb. amplifier power supply currently in use could certainly be replaced or eliminated.

With these component and design modifications, the weight of the entire device could probably be reduced by half. The current desktop height of the device would be maintained for ergonomic user operation, but the width of the device could be reduced somewhat and the depth also reduced to less than 18 in. The reduced size and weight would enable the device to be lifted by one person and transportable by car.

Chapter 7

Performance Testing and Results

Performance testing of the haptic magnetic levitation device was done to demonstrate the feasibility of the device for realistic haptic interaction. The critical performance parameters for fine, high-performance haptic interaction are the control bandwidths of the device and its sensitivity, or position resolution. Low inertia in the flotor enables the user to manipulate the attached handle more easily and feel computed interaction forces more distinctly. The maximum force and stiffness ranges of the device are also given. A range of performance tests were done for both the copper and aluminum coil floters. Tables of performance parameter data are given in Appendix C.

The significant differences between the copper and the aluminum coils are density and electrical resistance. Aluminum has 65% of the conductivity of copper but only 30% of the density of copper.

7.1 Performance Limits

The limits on performance of the magnetic levitation device with the copper coil flotor and the aluminum coil flotor were measured.

7.1.1 Maximum Control Rate

The maximum attainable sample and control rate of the present control system is 1.45 kHz. The maximum control rate is reduced if the processor must perform tasks in addition to servo control, such as logging data, performing collision detection, or communicating with the host workstation. This rate could be increased in the future by further upgrading the embedded control processor or by adding a fast floating-point digital signal processing board to the control system. The average execution time required on the present controller to perform the following tasks is shown below, with the remaining time on each cycle used

for host communication and other network and operating system tasks:

Operation	Time
Sensor Sampling and LUT Interpolation:	180 μ sec
Forward Kinematics:	360 μ sec
Control Law:	140 μ sec
Total:	680 μ sec

7.1.2 Maximum Forces and Torques

The main limitation on the forces obtainable with Lorentz actuation is the heat generated in the coils. With the hemispherical device design, the areas of the coils and flotor hemisphere are large and thin enough to quickly dissipate the heat from the coil currents so that the coils become only very slightly warmer than room temperature during steady-state operation. Since the heat dissipation of the coils does not impose such a significant force limitation for the desktop hemispherical device, the momentary peak forces and accelerations obtainable on the device are determined by the current limits of the actuator amplifiers.

The PWM current amplifiers used by the actuators have a 7 A current limit and each actuator generates 7.2 N/A when the coil is centered in the magnetic fields. The maximum peak forces and torques exertable by the device can then be calculated from the current-to-force transform matrix described in section 4.2. The resulting maximum forces and torques in each direction individually are:

Axis	Limit
Force in X :	64 N
Force in Y :	55 N
Force in Z :	140 N
Torque about X :	7.3 Nm
Torque about Y :	6.3 Nm
Torque about Z :	12.2 Nm

These maxima demonstrate that our device has more than adequate force and acceleration capabilities for any reasonable haptic interaction. Since such large forces and accelerations are not needed during typical operation, the maximum actuator currents are limited to 4 A in the control software.

7.1.3 Flotor Inertia

The flotor masses are 880 g for the copper coil flotor and 580 g for the aluminum coil flotor. The rotational inertia of the flotor bowl was calculated from the measured period of simple harmonic motion oscillations with the flotor attached to a torsional spring. The oscillation period is given by:

$$T = 2\pi\sqrt{\frac{I}{K}} \quad (7.1)$$

where T is the period of oscillation, K is the torsional spring stiffness, and I is the moment of inertia.

For a spherical shell with uniform density, the moment of inertia calculation

$$I = \int r^2 dm \quad (7.2)$$

with r is the distance from the axis of rotation of each mass particle dm , produces the formula

$$I = \frac{2Mr^2}{3} \quad (7.3)$$

with mass M and radius r . The mass distribution and moment of inertia of a hemispherical shell about its center point is the same as for a sphere and is the same for all principal axes.

The density of the magnetic levitation flotor hemispheres is not uniform, however, due to the distribution of coils and holes in the hemispheres. Compared to a uniform density shell, the measured moments of inertia were 8 percent greater for the copper coil flotor and 6 percent greater for the aluminum coil flotor.

Parameter	Copper Coil Flotor	Aluminum Coil Flotor
Mass	880 g	580 g
Calculated Moment of Inertia	0.00776 kg/m ²	0.00511 kg/m ²
Measured Moment of Intertia	0.00842 kg/m ²	0.00543 kg/m ²

7.1.4 Power Consumption

To levitate the flotor bowl, the vertical Lorentz force applied by the coil actuators must equal the weight of the flotor. Three of the six actuator coils on the flotor can generate forces with vertical components. The coils which generate vertical forces touch the flotor rim with their centers at 22.5° below the rim. If equal currents are applied to these coils the horizontal force vector components cancel out and the net force is purely vertical.

In the absence of force disturbances on the flotor, the feedforward currents cancel the flotor weight and the additional feedback currents necessary to stabilize the flotor in space

are negligible. The power consumption of the magnetic levitation device can be calculated by measuring the voltage drop across the three lifting coils and using the power consumption formula $P = V^2/R$ where V is the voltage drop over resistance R .

For the copper coil flotor, the voltage drop across each $6\ \Omega$ lifting coil was $3\ \text{V}$ for a power consumption of $1.5\ \text{W}$ each and a total of $4.5\ \text{W}$ required to levitate the $880\ \text{g}$ flotor. For the aluminum coil flotor, the voltage drop is $2.9\ \text{V}$ across three $10\ \Omega$ coils, dissipating $0.85\ \text{W}$ each, so $2.5\ \text{W}$ total is sufficient to levitate the $600\ \text{g}$ flotor. The aluminum coil flotor was also tested with an additional $1\ \text{kg}$ on the flotor to simulate a user's hand holding the handle and supported by the magnetic levitation device. With the added mass, the voltage drop was $7.3\ \text{V}$ over each lifting coil, dissipating $5.5\ \text{W}$ each for a total power consumption of $16.5\ \text{W}$.

The power for levitation generates resistive heating in the coils. At these power levels the heat is quickly dissipated from the flotor and the coils remain practically at room temperature during typical operation.

7.1.5 Sensor Noise and Position Resolution

The position sensor signals are digitized by 16-bit converters. The noise level of the converted signals after being passed through the analog antialiasing filters is less than ± 2 bits, for an analog noise level of less than $1\ \text{mV}$. Since the light spot position on the photodetector is given by the ratio of the electrode voltage difference to the sum in each direction, the position resolution of the sensor depends on the total amount of light hitting the sensor. If the LED is close to the sensor so that the sum signal is near saturation, the signal to noise ratio is 16000 ; if the flotor is at the edge of its translation and rotation ranges so that the LED is further away, the signal to noise ratio may be as low as 2500 . The average variation in the computed position in Cartesian coordinates due to sensor noise inside the $25\ \text{mm}$ and 15° motion range of the flotor ranges from ± 3 to $\pm 10\ \mu\text{m}$.

The noise level of one of the digitized sensor signals with the magnetic levitation flotor rigidly clamped in the center position is shown in Fig. 7.1, with an average variation in the digitized signal of only ± 1 to 2 units. This sensor noise level corresponds to an average variation in the calculated vertical position of less than $\pm 1\ \mu\text{m}$, as shown in Fig. 7.2.

When the flotor is levitated, the sensor noise causes added error in the controlled position. For the levitated flotor in the center position the average variation in the sensor signals is over ± 5 units and the average measured position error is approximately $\pm 4\ \mu\text{m}$, as shown in Figs. 7.3 and 7.4.

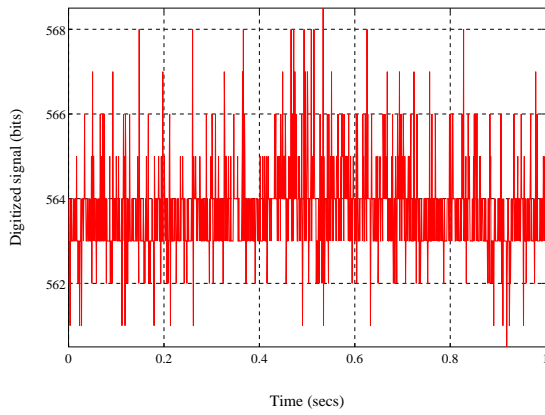


Figure 7.1: Sensor Signal with Clamped Flotor

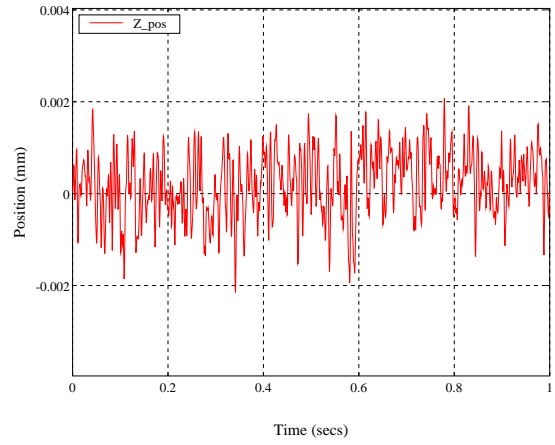


Figure 7.2: Calculated Position with Clamped Flotor

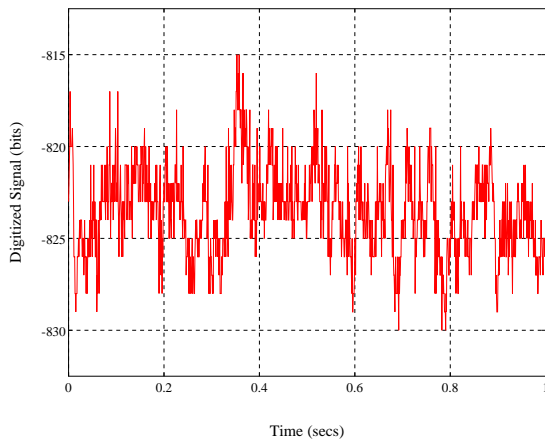


Figure 7.3: Sensor Signal with Levitated Flotor

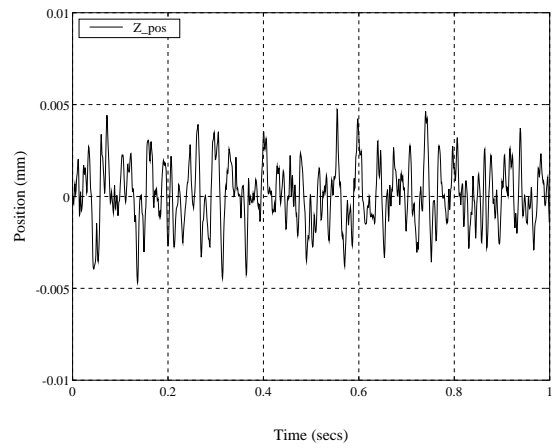


Figure 7.4: Calculated Position with Levitated Flotor

7.1.6 Stiffness Range

The maximum stiffnesses that have been achieved with the device to date are 25.0 N/mm in translation and 50.0 Nm/rad in rotation. During haptic interaction, this stiffness is used in unilateral constraints to give the user the impression of a rigid surface. Higher stiffnesses are achievable on and around the vertical axis due to greater passive damping with motion in that direction. During free motion the residual stiffness of the device due to the hanging coil current feed wires is approximately 0.005 N/mm, resulting in an maximum to minimum achievable impedance ratio of 5000:1. The stiffness of the device was measured from displacements using a millimeter scale and a handheld digital force gauge¹.

7.2 Response Bandwidths

The dynamic performance of the magnetic levitation device is determined by its frequency response bandwidths. Current, force, and position bandwidths are limited by the response of the amplifiers, the inductance of the actuator coils, the stiffness, inertia, and eddy current damping of the flotor hemisphere, the sensor signal noise, and the maximum update rate of the controller. Above a certain cutoff frequency, the magnitude of the system output of interest will tend to drop off or be increasingly attenuated as the input or command signal increases in frequency. The ratio of the input magnitude to the output magnitude and their phase differences is graphically shown in the Bode plot, where the response magnitude ratio is plotted in decibels on one plot and the phase difference is plotted in degrees against the input frequency on a logarithmic scale. Frequency response tests were performed on the magnetic levitation device for amplifier currents, vertical forces, and handle positions and orientations on each control axis.

7.2.1 Test Setup

A dynamic signal analyzer² was used to measure the frequency response of the coil current amplifiers, open-loop force generation at the device handle, and closed-loop position control. To measure the current bandwidth, the analyzer signal was the voltage drop across one of the actuator coils. For the force bandwidth, a load cell³ mounted between the flotor and a rigid platform was used. For the position bandwidth, the source signal from the signal

¹Model DFG-10, Chatillon, Greensboro NC

²Hewlett-Packard 3652A

³Entran ELF-TC500-20

analyzer was digitized to become the position control setpoint and the cartesian position calculated from the sensor signals was converted back to an analog signal for the analyzer.

The closed loop small-motion position control frequency response was measured with 0.2 mm and 0.002 radian peak-to-peak commands. The source signal for the frequency response was random noise in the 2-200 Hz range. Analog input and output channels were used to transmit the position command and response between the signal analyzer and the magnetic levitation device. The magnitude and phase of the frequency responses were calculated by the signal analyzer and plotted.

To determine the position control bandwidth of each axis of the device, the digital filter cutoffs and stiffness and damping gains were tuned to provide the largest possible ± 3 dB range of the frequency response magnitude. Frequency response testing was carried out on the vertical z axis and horizontal x axis in both translation and rotation, on both the aluminum and the copper coil fltors, and at control rates of 1.3 kHz and 650 Hz. The PWM current amplifiers used in the magnetic levitation device are rated for a 10 kHz bandwidth.

7.2.2 Amplifier Current and Force Bandwidths

The PWM current amplifiers have 10 kHz bandwidth rating, but frequency response testing with the actuator coil load showed a wide 6 dB resonance peak at approximately 600 Hz followed by a rolloff to -3 dB at approximately 6 kHz. The coil current frequency response plot is shown in Fig. 7.5. This amplifier current bandwidth is an upper limit for force and position bandwidths, since it is impossible to generate forces at frequencies higher than the actuator currents produced.

To measure the force bandwidth of the magnetic levitation device, the base on which the load cell is mounted must be as rigid as possible so that high frequency force responses can be accurately measured. Any vibration response in the load cell base platform will be reflected in the magnetic levitation Lorentz force reponse data. For this response measurement, the load cell was firmly attached to a stack of three 5 kg aluminum slabs to reduce transmission of vibrations. The force frequency response resonance at approximately 250 Hz is shown in the plots of Fig. 7.6. This 250 Hz resonance is concluded to be the resonant frequency of the fltor bowl.

7.2.3 Position Bandwidths

The closed loop position control frequency plots obtained during testing are shown in Figs. 7.7-7.10. The ± 3 dB bandwidth for each axis is greater at the faster control rate than

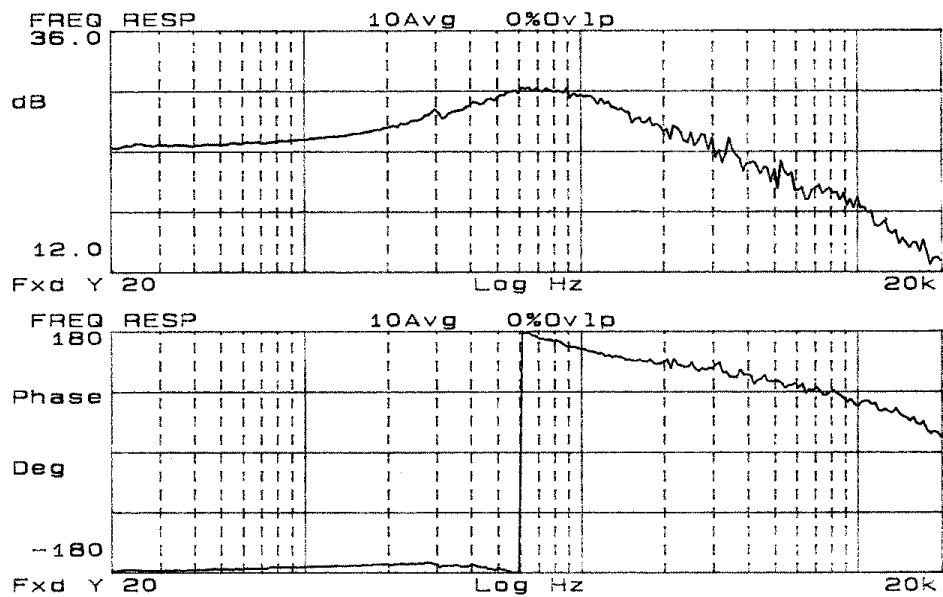


Figure 7.5: Amplifier Current Frequency Response

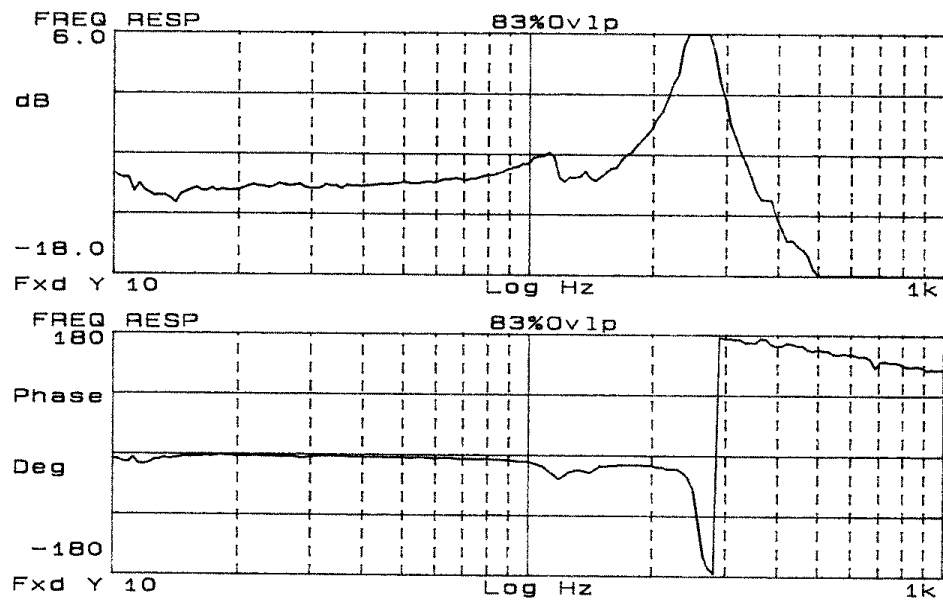


Figure 7.6: Force at Handle Frequency Response

the slower one, and greater for the aluminum coil flotor than for the copper coils. The resulting bandwidths for all tested degrees of freedom are in the neighborhood of 50 Hz at the half-speed control rate and 70 Hz at the fast control rate with the copper coil flotor, and 70 Hz range at the slow control rate and 100 Hz or greater at the fast control rate for the aluminum coil flotor. The bandwidths for vertical translation and rotation about the vertical axis are expected to be somewhat higher than for horizontal translations and rotations due to the symmetrical coil arrangement and mass distribution about the vertical axis.

7.3 Discussion

The performance of the aluminum coil flotor is superior to the copper coil flotor in the measures of

- lower weight,
- higher acceleration,
- lower typical power consumption,
- higher position frequency response bandwidths,

and the copper and aluminum coil flotors are equivalent in:

- maximum forces and torques,
- impedance range, and
- position sensitivity.

Since there is no disadvantage to the use of aluminum coils in the flotor, aluminum coils are definitely preferable for this magnetic levitation device. Furthermore, since the maximum forces and torques are higher than most users can apply with the fingertips during haptic interaction and the average power consumption is quite low even with the higher resistances of the aluminum coils, further improvements in operational frequency response bandwidths could be achieved by using narrower aluminum coil ribbon wire for further reduction of the flotor mass. Less powerful magnets could also be used for the actuators to reduce the material cost of the magnetic levitation device, since the maximum forces and torques of the device could be reduced without significantly affecting the effectiveness of haptic interaction

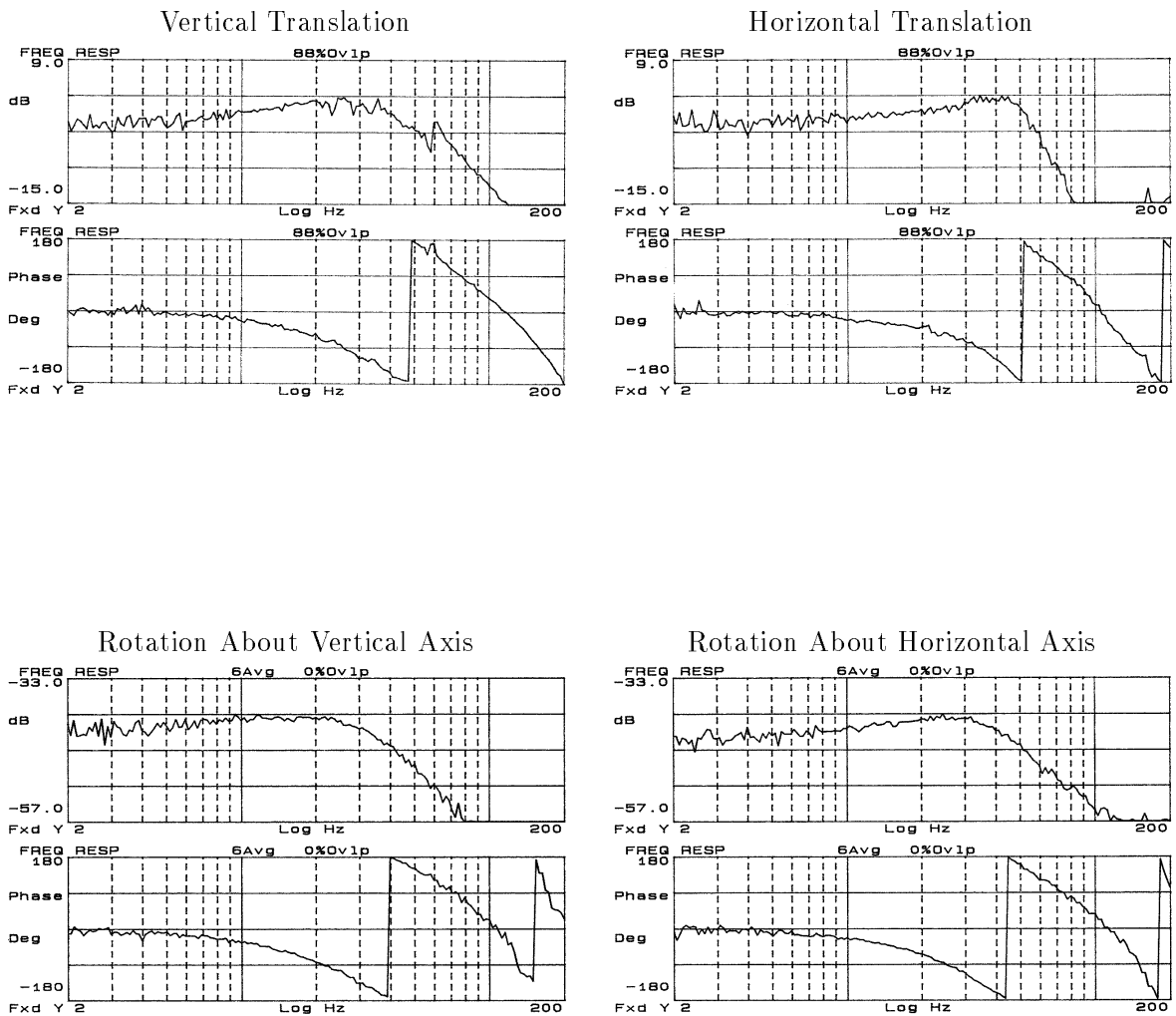


Figure 7.7: Closed-Loop Position Response with Copper Coil Flotor at 650 Hz Control Rate

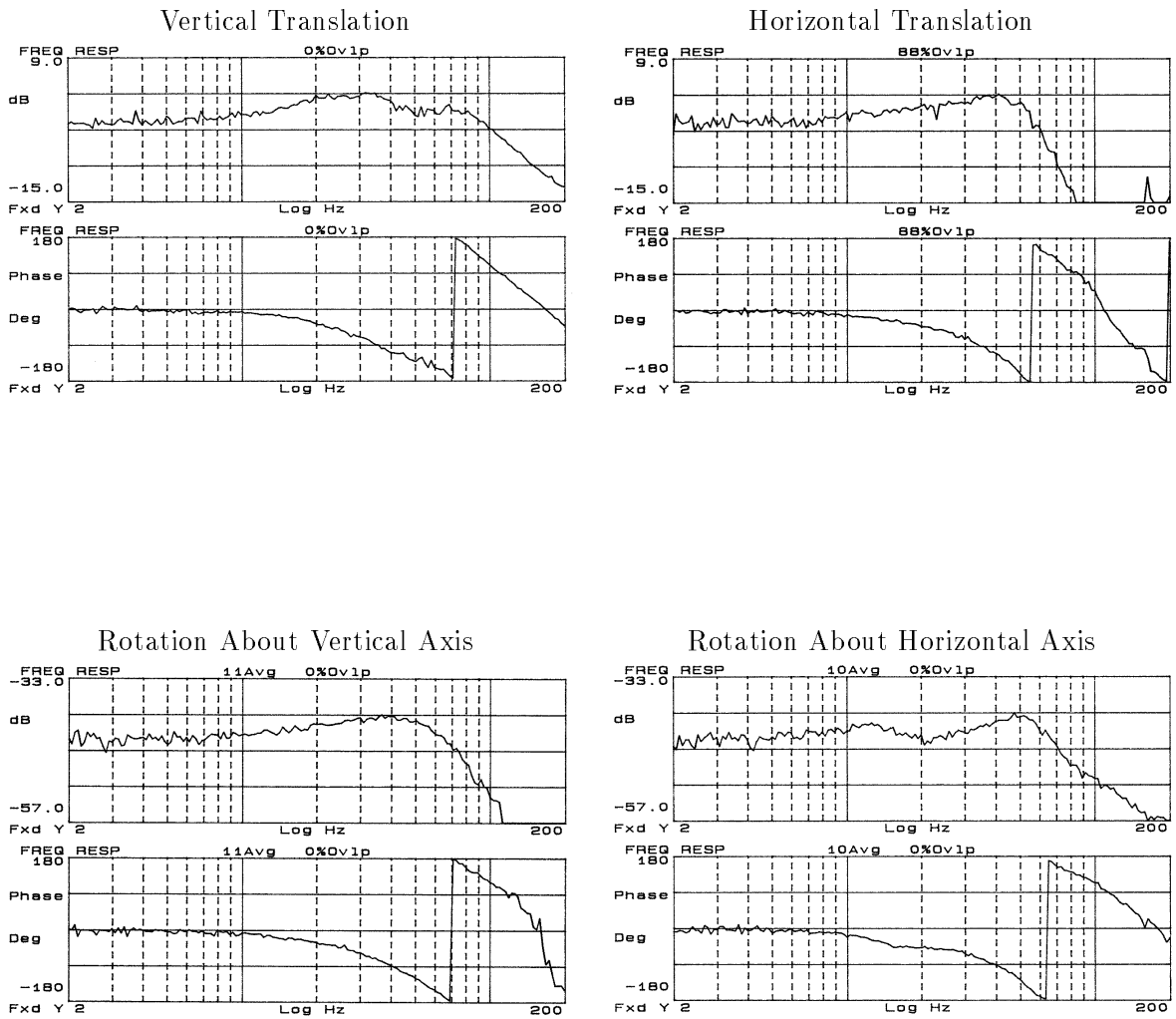


Figure 7.8: Closed-Loop Position Response with Copper Coil Flotor at 1.3 kHz Control Rate

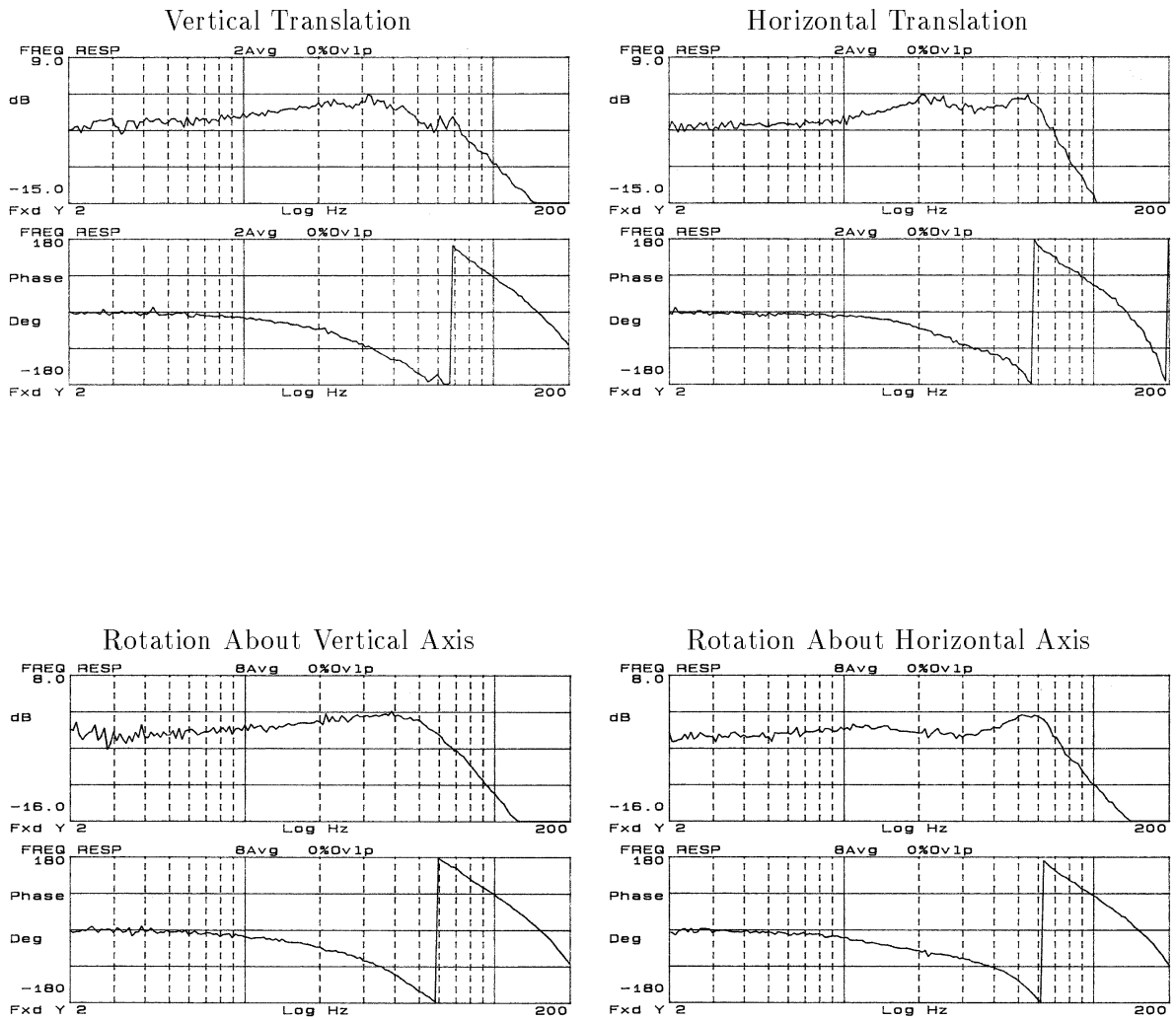


Figure 7.9: Closed-Loop Position Response with Aluminum Coil Flotor at 650 Hz Control Rate

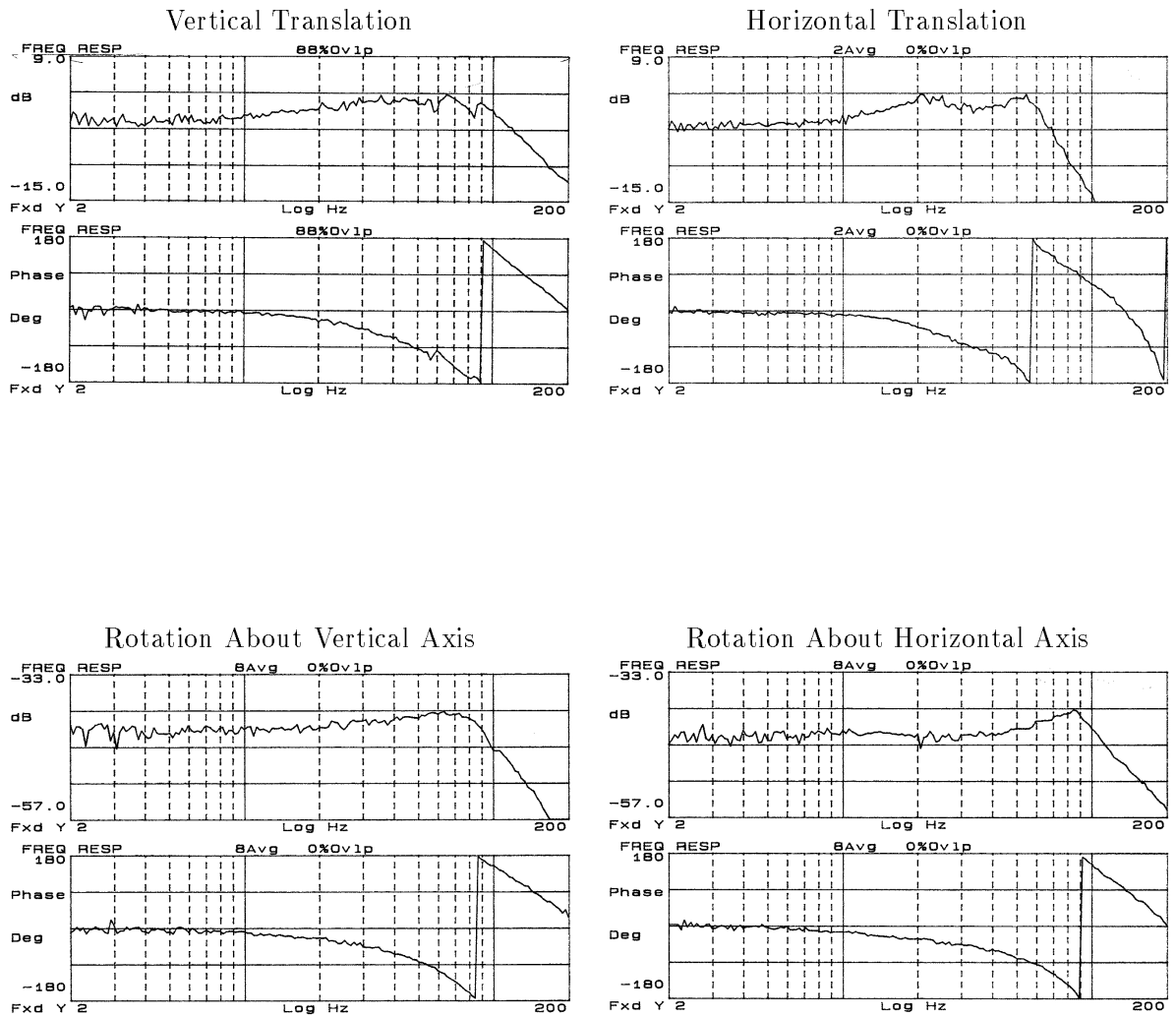


Figure 7.10: Closed-Loop Position Response with Aluminum Coil Flotor at 1.3 kHz Control Rate

using the device. To generate forces above the 250 Hz resonant frequency of the present aluminum coil flotor, the flotor would have to be made stiffer and/or lighter.

The aluminum flotor delivers position control frequency response bandwidths of at least 100 Hz in all the degrees of freedom of the magnetic levitation device with the 1.3 kHz servo control rate. Even at the 650 Hz control rate, the control bandwidths in all degrees of freedom are at least 70 Hz, which is still sufficiently high to provide convincing haptic contact and surface texture forces. Some further improvement in control bandwidths may be possible with a faster control processor and update rate, but the achievable control bandwidths have an upper bound determined by the sensor noise level, flotor inertia, and eddy current damping. At lower control rates the additional computation time available to the local control processor can be used for more sophisticated collision detection and environment simulation to provide more responsive haptic interaction.

The operation of this new magnetic levitation haptic interface device is part of the larger goal to realize high-performance, realistic haptic interaction with dynamic physical simulated environments. Since no other 6 DOF haptic interface device yet exists with the bandwidths of the new device, it enables evaluation of richer, more subtle methods of haptic interaction with dynamic physical simulations including sophisticated texture and friction surface effects. Noncontact actuation and position sensing eliminates the friction and other actuator nonlinearities of haptic interfaces which use conventional motors, linkages, and cables, and the new device also has a very large impedance range for haptic interaction.

A realistic, dynamic virtual environment, integrated with the magnetic levitation haptic interface device and a graphical display, has the potential to provide users with a very convincing experience of directly interacting with physical objects in a real environment.

Part II

Interactive Simulations

Chapter 8

Haptic Simulation and Display

The second part of my thesis work was to build an integrated hardware and software system to interact with realistic dynamic simulated environments using the magnetic levitation haptic interface device described in Part I as a virtual tool. Previous implementations of magnetic levitation haptic interaction with simulated environments [44, 45] were with simple, limited, static environments, in which only the magnetic levitation device flotor, corresponding to the haptic tool in the virtual environment, moved while the environment remained fixed. Although haptic interaction with a static environment is useful in itself to enable the user to feel the shape and surface characteristics of an object, the work to be described here goes further to provide interaction with a realistic physically simulated world in which objects dynamically react to forces acting on them due to collisions, friction, gravity or other force fields or point forces. Such a realistic, dynamic virtual environment, integrated with the haptic interface device controller described in Part I and a realistic graphical display, provides users with a convincing experience of directly interacting with physical objects in a real environment.

To realize this type of interaction, a simulation engine is needed to model the behavior of objects in the virtual world in real time. The haptic device controller has a limited computational speed and can only simulate simple environments locally. In the newly realized system, the simulation executes on a workstation and communicates with the maglev haptic device controller via an Ethernet network link. The simulation software must continually and smoothly handle collision detection, object motion mechanics, contact and constraint forces, and communication with the device controller. The device controller calculates and continually updates its force feedforward and impedance control parameters according to the contact state of the haptic tool in the simulated environment.

The software system has to generate haptic device forces from an environment model with the geometry of the virtual tool and other objects in the simulation, dynamic models

including masses and inertias of the objects, optional surface models of friction and texture, and the motion imposed on the haptic device handle by the user.

The effectiveness of the proposed haptic interaction system will depend on the system integration and virtual object control methods as much as on the dynamic performance of the device and the accuracy of the physical simulation. The intention of the integration methods is to make the haptic interface as transparent and direct as possible, so that the user can feel he or she is actually directly physically interacting with the simulated environment.

The effectiveness of a haptic interaction system is difficult to evaluate quantitatively since it depends not only on the accuracy with which the device reproduces the dynamics of a real physical system, but also on the subjective perception and neuromuscular properties of the user. It may be advantageous to emphasize or exaggerate some aspects of the physical simulation in order to compensate for other limitations in the simulation or haptic device and thereby improve the user's overall perception of the simulated environment. For example, increasing the force impulses to the haptic device due to object collisions in the simulated environment may compensate for limited stiffness in the haptic device. An accurate analysis of the effectiveness of a haptic interface system would include a user survey and results of task performance as well as test data from the device. Statistical surveys of the effectiveness and realism of simulated tasks using the haptic interaction system for a group of different representative users are planned in an upcoming project and are beyond the scope of the thesis described here.

8.1 Modularity

There are many benefits achieved by separating the haptic device controller and the simulation of the virtual environment. As mentioned previously, the computational power of the haptic device controller is limited and it is not able to simulate environments of any more than minimal complexity at a rate sufficient for haptic rendering. It is also easier to monitor and ensure the stability and reliability of the realtime haptic device controller if its interfaces and tasks are well-defined and consistent.

Separation of the device control and environment simulation enables the device to be interfaced to existing environment simulation systems which can operate at or near realtime rates while exchanging position and contact information with the device controller. More than one haptic device could be interfaced to a single simulated environment so that two-handed tool-based tasks could be simulated or so that multiple users could interact in

a shared environment. The environment simulation system could also be directly replaced by an actual manipulator which exchanged position information in the same format with the haptic device controller, to realize high-performance force reflecting teleoperation.

8.2 Simulation and Device Correspondence

The most critical problem that must be resolved to integrate physically-based modeling with the maglev haptic interface device is to maintain the correspondence between the simulated environment, the haptic interaction device, and the graphical display. The graphics display and the haptic interaction must be kept closely synchronized with minimal latency between them so that the environment seen in the display and felt through the device seems consistent and natural to the user. Perceptible differences in the feel and appearance of surface shapes or the timing of collision events seriously degrade the effectiveness of interaction with the simulated environment. To preserve consistency in the interaction system, the state information in the haptic device controller and the haptic tool object in the simulation must be mutually updated often enough that slight differences in timing, shape, and contact configurations will be virtually imperceptible to the user. For the best system performance, the simulation processing and the device control will be executed on separate processors and a local intermediate representation will be periodically generated by the simulation and communicated to the device controller to determine the device's spatial impedance at the much faster control cycle rate.

The simulation processing and device control must be decoupled because of their different performance requirements: The physical simulation requires very intensive processing and its update rate depends on the complexity of the environment and dynamic events occurring from moment to moment. A moderately complex simulation and graphical display executing on a SGI workstation updates at 20-30 Hz, which is sufficiently fast to appear almost continuous when graphically displayed to the user. The device controller must cycle at a much faster fixed sample rate, at least 500 Hz for convincing interaction with solid objects and preferably over 1000 Hz for realistic emulation of more finely detailed subtle surface characteristics. Rather than attempting to integrate intensive asynchronous computation, graphical display, I/O, and fast realtime sensing and control on a single processor, it is more practical and flexible to execute the realtime control task and analog I/O on a separate processor system.

Due to this decoupling, an intermediate dynamic representation will be used to achieve adequate performance. To keep the simulation and device control mutually consistent given

limited communication speed and bandwidths, the simulation will periodically send a description of the local environment to the device controller. The device controller will calculate the control gains, setpoints, and compliance frame variables for spatial compliance control from the local environmental description of motion constraints supplied by the simulation. Dynamic events can then be predicted by the device controller so that they occur nearly simultaneously in the simulation and the actual dynamic behavior of the haptic device. The device controller must also return the flotor motion and actuated interaction force and torque back to the simulation so that the simulation responds to the user motion and forces.

8.3 Intermediate Representation

Intermediate representation for virtual environment interaction was developed by Adachi *et al.* in [18]. Mark *et al.* and others describe several intermediate representations that they implemented, including plane-and-probe; friction and texture; and multiple planes, probes, and springs [52].

A physically-based dynamic simulation has been integrated with a PHANTOM 3 DOF point-based haptic interface device by Vedula with added Coulomb friction [53]. The most successful intermediate representation implemented consisted of a kinematic description of the object plane closest to the location of the probe tip in the virtual world including its updated position and the relative position and linear and angular velocity of the object's center of mass.

Previously developed intermediate representations [18, 52, 53] represent point force interactions only. For rigid-body 6 DOF haptic interaction, a more flexible and detailed intermediate representation is necessary. In this described research, I have developed and implemented an interface to a simulated environment by using a new representation model which generates and displays the complete force-torque wrenches resulting from all rigid body interactions.

8.4 Virtual Coupling

One way to interface a haptic device to a simulated environment is through a virtual coupling [16] between the tool in the simulation and the haptic interface device. A virtual spring and damper concisely transmits all force and motion information between the device and the simulation so that the haptic device user can feel the dynamics of the tool in the simulation.

The stability of the haptic device controller can be easily ensured by setting the gains of the virtual spring and damper. The drawback of the virtual coupling interface method is that the detail and responsiveness of the environment simulation is diminished, since sudden motions in the environment are filtered through the virtual spring and damper.

8.5 Contact Point Intermediate Representation

The simplest way to extend point-contact intermediate representations for rigid-body interaction would be to generate representations for multiple contact points as suggested in [52]. The combination of the point forces would produce a force-torque wrench on the haptic tool in the simulated environment and be felt by the user. The drawback of this method is that it only emulates contacts between vertices of the haptic tool and planar surfaces in the simulation; edge-to-edge contacts and contacts between haptic tool planes and environment vertices would not be detected and would not generate reaction forces.

To emulate real rigid-body haptic interaction, the multiple-point representation method of [52] must be extended to recognize edge-to-edge and tool-plane-to-world-vertex contact forces also. A full intermediate representation for rigid-body contacts may become rather large even for simple objects, as illustrated by the cube alignment task of Fig. 8.1. Small perturbations in the position of the cube shown may result in any of the following rigid-body contacts:

- Point **A** with Plane ***abcd***, ***acge***, and ***abfe***
- Point **B** with Plane ***abcd*** and ***abfe***
- Point **C** with Plane ***abcd*** and ***acge***
- Point **D** with Plane ***abcd***
- Point **E** with Plane ***acge*** and ***abfe***
- Point **F** with Plane ***abfe***
- Point **G** with Plane ***acge***
- Point **d** with Plane ***ABCD***
- Point **f** with Plane ***ABFE***
- Point **g** with Plane ***ACGE***
- Edge ***AB*** with Edge ***bd*** and ***bf***
- Edge ***AC*** with Edge ***cd*** and ***cg***
- Edge ***AE*** with Edge ***ef*** and ***eg***
- Edge ***BD*** with Edge ***bd*** and ***cd***
- Edge ***CD*** with Edge ***bd*** and ***cd***
- Edge ***BF*** with Edge ***bf*** and ***ef***

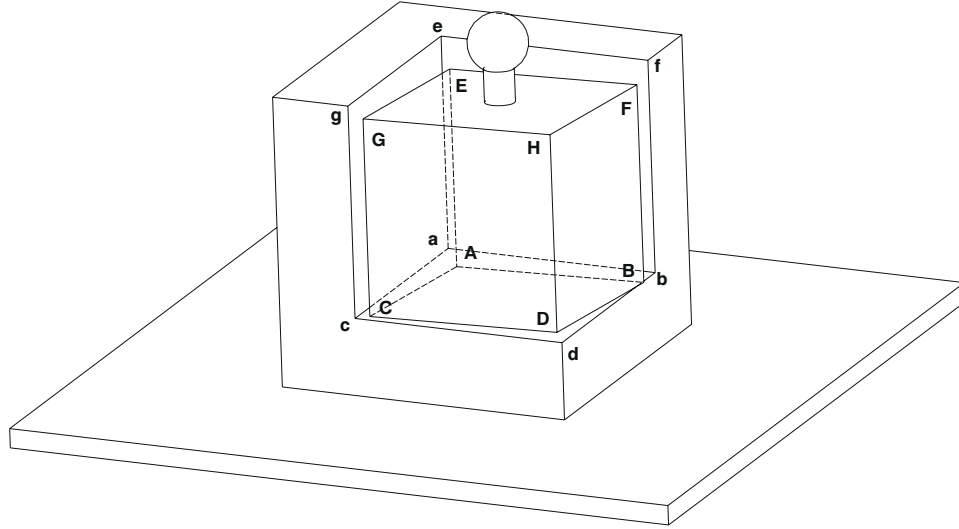


Figure 8.1: Potential Contacts Aligning a Cube

- Edge EF with Edge bf and ed
- Edge CG with Edge cg and eg
- Edge EG with Edge cg and eg

For the intermediate representation to correctly represent this local object configuration, it must contain the positions and types of all the potential local contacts, as well as the translational and angular velocities of the relevant surfaces, edges, and vertices. Due to the limitations in communication bandwidth and processing speed for realtime force interaction, it may be necessary to limit the number of vertices and edges of the modelled haptic tool.

8.6 Simulation and Controller Integration Summary

To summarize the requirements for integration of the physically-based simulation and the maglev device controller, the possible interactions between the simulation and the controller during free tool motion, tool-environment collisions, and constrained tool motion modes are described below:

Free tool motion: The physical simulation and the haptic controller are independent of each other in this mode. The user feels the actual inertia of the maglev flotor but the weight that is felt is dependent on the amount of feedforward gravity cancellation in the controller. The passive eddy current damping in the maglev device may be mostly eliminated if desired by adding negative damping in the control feedback. A collision

detection algorithm executing in the physical simulation will signal any change to the free motion of the maglev flotor.

Tool-environment collision: When a tool-to-environment collision occurs, a collision force-torque impulse wrench is generated by the physical simulation, which is dependent on the masses and moments of inertia of the tool and the collided object, their relative velocity, and a predefined coefficient of restitution ϵ where $0 < \epsilon < 1$. While the colliding tool and environment objects remain in contact in the simulation, the device controller must switch from free to constrained tool motion. The collision impulse vector from the simulation can be communicated to the device controller and reflected to the haptic interface user at the right instant by applying a force $\mathbf{F} = \mathbf{J}/T$ to the flotor for one sample period, where \mathbf{J} is the impulse vector and T is the sample period time. For the simple case where the flotor of mass m represents a point mass tool colliding with an stationary infinite-mass object, with a velocity v parallel to the surface normal, the collision impulse is $\mathbf{J} = -\epsilon m \mathbf{v}$.

Tool-environment resting contact and constrained motion: This is the most difficult case to emulate since combinations of contacts produce the impedance for the haptic device flotor at each controller sample period. To accomplish this, an intermediate representation data structure must be sent from the simulation to the device controller at each iteration of the simulation. The haptic device controller synthesizes the correct impedance gains and setpoints for the haptic device flotor to generate feedback forces at each sample period of the controller as different contacts are made and broken.

The three modes of interaction are not independent but may run concurrently in some dynamic conditions. For example, one corner of the haptic tool may be in resting contact with an object in the simulated environment when a different corner of the tool collides with an object. The remaining unconstrained degrees of freedom of the haptic tool and flotor will have free motion to pivot around the constrained corners.

Chapter 9

Modeling and Simulation for Haptic Interaction

To realize convincing, realistic haptic interaction with any modelled environment, the simulation of the environment should be as fast and detailed as possible to match the responsiveness of the magnetic levitation haptic device. The haptic perception capabilities of the hand still remain much greater than any haptic interface system can deliver, although the present system is a significant step towards that goal.

Control of the magnetic levitation device of Part I is based on a setpoint and stiffness and damping gains for each axis with added force feedforward, otherwise referred to as the proportional-derivative error [PD] control as described in Section 5.2.3. An interface of some kind is necessary to set the feedforward, setpoint, and gain control parameters from the environment simulation. A simple example of a control interface to a simulated environment is the unilateral constraint for a single point on the haptic tool contacting a solid wall. Both the stiffness and damping gains are set to zero when the point is on one side of the wall and set to stiff, stable gains if the point is inside the wall, with the position setpoint at the point on the wall closest to the tool point. For the haptic interaction system described here, the interface between the simulation and the device controller must account for rigid-body contacts between the tool and environment which may include multiple point, edge, and face contacts. The simulation to controller interface would further benefit with the addition of modelling and display of friction and texture.

The following sections describe the physically-based simulation software package used for the haptic interaction system, methods for interfacing the simulation with the magnetic levitation device, and methods for modeling and haptically displaying surface friction and texture.

9.1 Physical Dynamics Simulation

The CORIOLISTM 3-D physically-based dynamic simulation package developed by Baraff is used for the interactive simulation, an extension of the 2-D simulation described in [54]. This package calculates forces and motions of rigid bodies in space due to Newtonian mechanics, motion constraints, collisions, and friction in near-real time on the host workstation. The algorithms calculate object dynamics by numerical integration of the environment model differential equations at each timestep, accounting for instantaneous collisions and motion constraints due to resting or sliding contact [55, 56]. A sample display of Baraff’s real time physically-based simulations is shown in Fig. 9.1. The functionality and use of the CORIOLIS package is described in an API [57]. Added details of the methods used to calculate collision and constraint forces and perform efficient collision detection are given in Appendix D.

9.1.1 Dynamic Simulation Methods

Three types of motion dynamics must be modelled: free rigid-body dynamics, collisions, and constrained motion such as sliding or resting contact. Free rigid body dynamics are straightforward to calculate given each body’s state vector, initial conditions, and inertial parameters. Contact forces between objects are calculated from the relative velocities, the contact point, and a surface normal. Plane-to-plane and edge-to-plane contacts are modeled as multiple vertex-to-plane contacts. Vertex-to-vertex and vertex-to-edge contacts are degenerate and can be treated as vertex-plane contacts, since they can only occur instantaneously. For edge-to-edge contacts the surface normal is defined to be perpendicular to both edges.

The Newtonian free rigid-body dynamic state equations given below are integrated for each free body in the simulation:

$$\frac{d}{dt} \begin{pmatrix} \mathbf{x}(t) \\ \mathbf{R}(t) \\ \mathbf{P}(t) \\ \mathbf{L}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{v}(t) \\ \boldsymbol{\omega}(t)\mathbf{R}(t) \\ \mathbf{F}(t) \\ \boldsymbol{\tau}(t) \end{pmatrix}, \quad (9.1)$$

$$\mathbf{v}(t) = \frac{\mathbf{P}(t)}{M}, \quad \boldsymbol{\omega}(t) = \mathbf{I}(t)\mathbf{L}(t), \quad (9.2)$$

$$\mathbf{I}(t) = \mathbf{R}(t)\mathbf{I}_{body}\mathbf{R}(t)^T. \quad (9.3)$$

with position \mathbf{x} , rotation matrix \mathbf{R} , momentum \mathbf{P} , angular momentum \mathbf{L} , velocity \mathbf{v} , angular velocity $\boldsymbol{\omega}$, force \mathbf{F} , and torque $\boldsymbol{\tau}$. Body mass M is constant and inertia \mathbf{I} is a coordinate transform of constant inertia matrix \mathbf{I}_{body} .

In this simulation approach, objects are assumed to be perfectly rigid and non-interpenetrating. To prevent interpenetration when contacts occur, the necessary forces to prevent interpenetration are analytically computed and introduced into the simulation. In the case of collisions, the impulsive forces and accelerations are instantaneous. Since differential equation integration methods assume continuous dynamics, integration must be halted at the time of a collision and restarted again with new initial conditions.

Objects in prolonged contact with one another are subjected to only continuous forces and accelerations so discontinuities in the dynamics integration are unnecessary. Since the contact forces for multiple contacts may depend on each other, they must all be computed simultaneously. These forces must be solved using a quadratic program, however the parameters of the problem are such that this quadratic program can be solved efficiently.

Collision detection to prevent interpenetration can be a computationally intensive requirement of the physical simulation. A naive method of comparing points and faces at each time step would require $O(n^2)$ computation; efficient updating of a dependency tree structure can considerably reduce the computation required in practice. A 3-D hierarchical bounding box approach is reported to run essentially in $O(n + k)$ with k the number of pairwise bounding box overlaps [55].

9.1.2 Dynamic Simulation for Haptic Interaction

The CORIOLISTM simulation package was not originally intended for use with a haptic interface. The execution time of the simulation is non-deterministic and cannot strictly be said to be “realtime”. Any delays from the simulation or in communication are noticeable to the haptic interface user when the simulation is tightly coupled to the haptic device controller. In particular, if many collisions occur in the simulation, then the elapsed time in the simulation will drastically slow down and the tool in the haptic simulation will seem to be stuck or jammed.

Both the magnetic levitation haptic device controller and the simulation can be classified as impedance systems since they calculate force outputs from position inputs at each iteration cycle. Since each system expects position inputs from the other, they can not be interfaced directly. Some coupling interface is needed between the two subsystems.

The steps taken to mitigate the nondeterministic effects of the CORIOLISTM simulation and enable the physical simulation to be interfaced to the realtime magnetic levitation haptic

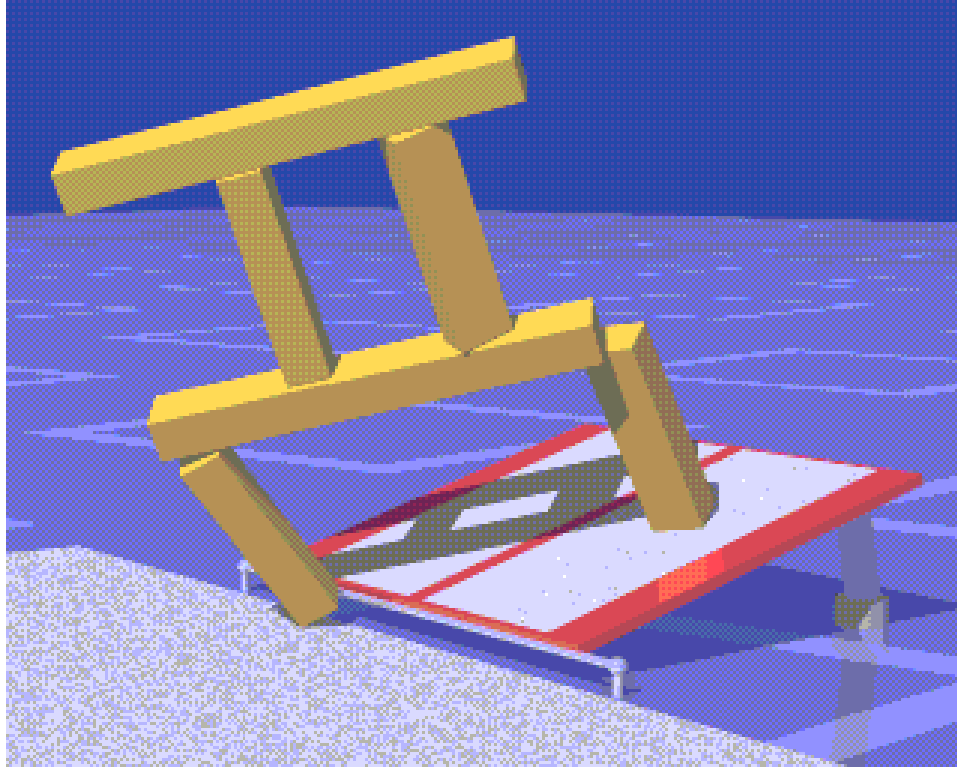


Figure 9.1: Sample Display of Physically-Based Simulation

device controller are explained in Chapters 12 and 13 which describe the development and implementation of the virtual coupling and the contact point intermediate representation interface methods.

9.2 Friction

Introduction of Coulomb friction into the non-penetrating rigid-body physically-based simulation could require exponential computation time, but a reformulation allowing additional introduction of non-impulsive contact forces enables the use of algorithms with polynomial time behavior [58].

9.2.1 Viscosity

Added viscous friction or damping in the maglev device dynamics is produced by the derivative gain in the controller applied to filtered or estimated velocities. Passive damping also exists in the device due to eddy currents in the flotor shell. It is possible to reduce the total damping of the device response by applying a negative damping gain, but this is not done

during haptic interaction to ensure stability and prevent resonances in the flotor from being excited.

Previous work has shown the damping control gain to be especially effective in emulating damped collisions, sliding on sticky or oily surfaces, and motion of objects submerged in viscous fluids, provided that the signal filters described in Section 5.2.1 are tuned to eliminate inaccuracies from sensor noise and quantization effects but still capture the motions of the user’s hand and fingers.

9.2.2 Coulomb Friction

The emulation of dry sliding or *Coulomb* friction is slightly more involved as it requires monitoring and updating a stick/slip condition. When a virtual tool object surface is in contact with another surface, the following algorithm determines motion on that surface [45].

During the sticking state, the control law used is:

$$f = -k_v \dot{x} - k_p(x_d - x). \quad (9.4)$$

If f exceeds a threshold force f_f , then the controller switches to the sliding state and uses the viscous only control law:

$$f = -k_v \dot{x}. \quad (9.5)$$

When f falls below f_f again, the position setpoint x_d is reset to x and the controller switches back to the sticking state. The sliding force threshold is determined by the defined coefficient of friction μ between the surfaces and the normal force between the objects f_n according to Coulomb friction:

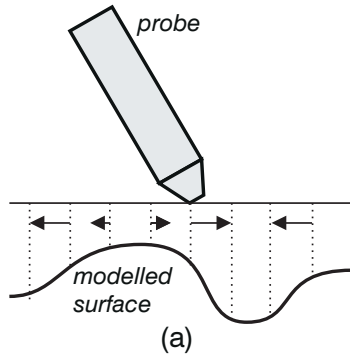
$$f_f = \mu f_n. \quad (9.6)$$

Both types of friction were previously demonstrated by the author [44] with a static environment using the IBM magic wrist. Implementation is straightforward and leads to a high degree of perceived realism in the haptic interaction. The integration of these friction emulation techniques with realtime physically-based modeling simulations is an important result of this research work.

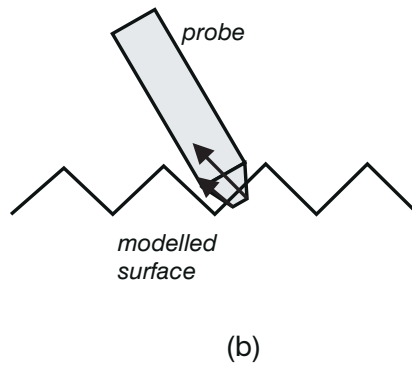
9.3 Texture

Convincing, realistic haptic display of texture was first demonstrated in the “sandpaper system” [19] with a 2-DOF force-reflecting joystick. Surface roughness was shown even

texture forces in plane
generated from depth map
gradient:



texture forces calculated from
rigid-body interpenetration:



texture forces randomly
generated from gaussian
statistical distribution:

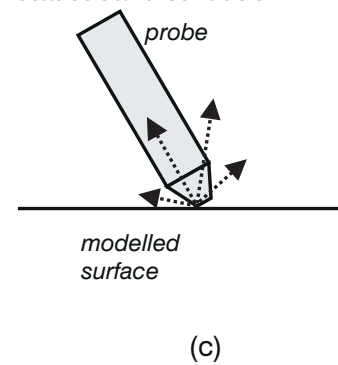


Figure 9.2: Texture modelling methods (a) Sandpaper system (b) surface grooves (c) stochastic

with only 2 DOF in the plane by generating forces proportional to the local gradient of a simulated bumpy surface as shown in Fig. 9.2(a).

9.3.1 Texture from Facets

Surface texture was emulated with the original Magic Wrist and the new magnetic levitation haptic device by defining a multifaceted surface and treating interaction with that surface in the same way as with other rigid surfaces [44]. A flat surface with parallel sawtooth grooves 200 μm wide and deep was defined as in Fig. 9.2(b). Contact with the angled walls of the grooves produces a reaction force normal to the surface proportional to the interpenetration distance. A disadvantage of this method is the “pop-through” effect: The reaction force is calculated by the distance between the actual position of the flotor and the closest surface point, so the assumed surface contact point can “pop-through” from one side of a peak to the other. This phenomenon can be avoided by tracking “ghost” points on the surface during the interpenetration so that the force direction does not abruptly switch from one surface to another in a nonphysical way[15, 16, 17].

The presence of the grooves could be felt by the user as long as the surface was sufficiently stiff. A buzzing sensation was felt as the flotor was moved perpendicular to the texture grooves, while the motion felt smooth parallel to the grooves. The addition of static friction improved the natural feel of the interaction.

9.3.2 Fine Generated Texture

For fine textures, pseudo-random generated force patterns can be superimposed on the contact forces, as in Fig. 9.2(c), to give the impression of surface textures instead of modelling surface features physically. A stochastic method of texture emulation for haptic interaction is presented and implemented in [20]. For texture emulation that is both realistic and efficient, the limitations of physical dynamics modelling, computational power, and human haptic sensing capabilities must be balanced against each other. Below a certain size of shape detail, the increase in the complexity of the physical model and the amount of necessary information transfer between the simulation and the controller becomes prohibitively large as it approaches the limits of human tool-based haptic perception.

9.3.3 Texture Issues

This surface texture simulation method for haptic display is analogous to 3-D graphics display methods: Larger geometric features are modelled with polygon meshes or spline curves, while fine texture patterns are directly mapped onto surfaces. For the best correspondence between the graphical display and haptic sensing of a virtual environment, the haptic texture can be either generated from the graphic texture or both the graphic and the haptic texture representations can be generated from an underlying surface roughness model in a manner similar to the “sandpaper system”.

Synthesis or coordination between friction and texture emulation can improve the effectiveness and realism of haptic interaction, as the same physical effects of surface variations and microcollisions cause both texture and friction. Coulomb friction itself is an approximate model which averages out the effects of microscopic surface variations. Massie defines a “haptic spectrum” where real-world surface variations are haptically perceived as shape, texture, or friction according to their scale, or “wavelength” [15].

Dynamic and static Coulomb surface friction as well as texture is emulated in [52] by Mark *et al.* by a surface populated by snags in a model similar to the bumpy surface texture model of [19]. In [20], friction and texture emulation forces are completely decomposed so that

$$\mathbf{F}_{contact} = \mathbf{F}_{constraint} + \mathbf{F}_{friction} + \mathbf{F}_{texture} \quad (9.7)$$

The haptic interaction system used decomposed models of texture and friction for increased flexibility and simplicity in surface modeling.

An interesting feature of the haptic interface is that exotic physical effects are possible. For example, directionally dependent viscosity can be easily obtained by using different velocity feedback gains in each control axis.

Chapter 10

Local Simulations

Several simple simulated environments were implemented to execute entirely on the haptic device controller processor. These interactive simulations demonstrate the effectiveness of the haptic interaction simulation and control methods without any delays or loss of bandwidth due to communication with a separate simulation operating at a slower update rate. For these simulations, the object dynamics and contact constraints are computed locally and the graphical workstation is used for only for rendering and display of the simulated environment. The position data is sent from the haptic device controller to the graphics workstation at approximately 50 Hz over an Ethernet network using a TCP/IP socket connection. The information flow over this socket connection is one way only. Since these simulations are completely independent of the host workstation, the reaction and feel of the haptic interface device is the same when there is no socket connection to the graphics workstation and the graphical display program is not executing. The graphical displays were generated using the Open Inventor graphical package on SGI workstations.

The environment is static. The virtual tool is the only object that moves in the local simulations.

The local implementation should provide the best responsiveness, bandwidth, sensitivity and stiffness, for applications but the complexity of the simulations is limited due to the limited processing speed and power of the realtime controller.

10.1 Basic Rigid-Body Simulations

The simple rigid-body local simulations demonstrate the capabilities of haptic interface system with 3-D rigid-body environments. To emulate point, edge, and face contacts, the complete 6 DOF of the haptic device are required to reflect forces and torques to the user. Simple tools and environments are demonstrated so as not to overburden the current

device controller and allow simulation and control update rates of 1000 Hz. The walls in the simulated environment are also aligned with world coordinate frame axes in order to minimize the contact calculations required.

10.1.1 Interface and Control Method

First, the positions of the haptic tool vertices are calculated from the position and orientation of the haptic interface device handle. The local rigid-body contact simulations use a block-shaped tool since the contact and feedback calculations are minimal. For a block-shaped haptic tool of dimensions $2w \times 2l \times 2h$ with the center of the block corresponding to the center of the haptic device handle, the positions of the vertices of the tool are given by

$$\begin{aligned}
\mathbf{v}_0 &= \mathbf{P} + w\hat{\mathbf{i}} + l\hat{\mathbf{j}} + h\hat{\mathbf{k}}, \\
\mathbf{v}_1 &= \mathbf{P} + w\hat{\mathbf{i}} + l\hat{\mathbf{j}} - h\hat{\mathbf{k}}, \\
\mathbf{v}_2 &= \mathbf{P} + w\hat{\mathbf{i}} - l\hat{\mathbf{j}} + h\hat{\mathbf{k}}, \\
\mathbf{v}_3 &= \mathbf{P} + w\hat{\mathbf{i}} - l\hat{\mathbf{j}} - h\hat{\mathbf{k}}, \\
\mathbf{v}_4 &= \mathbf{P} - w\hat{\mathbf{i}} + l\hat{\mathbf{j}} + h\hat{\mathbf{k}}, \\
\mathbf{v}_5 &= \mathbf{P} - w\hat{\mathbf{i}} + l\hat{\mathbf{j}} - h\hat{\mathbf{k}}, \\
\mathbf{v}_6 &= \mathbf{P} - w\hat{\mathbf{i}} - l\hat{\mathbf{j}} + h\hat{\mathbf{k}}, \\
\mathbf{v}_7 &= \mathbf{P} - w\hat{\mathbf{i}} - l\hat{\mathbf{j}} - h\hat{\mathbf{k}},
\end{aligned}$$

where the haptic device handle position is \mathbf{P} and the matrix rotation of the haptic device handle \mathbf{R} is partitioned as $[\hat{\mathbf{i}} \ \hat{\mathbf{j}} \ \hat{\mathbf{k}}]$.

Then, all possible combinations of vertices and faces are checked for interpenetration. If there are n vertices and m faces on the haptic tool and n' vertices and m' faces in the environment, then $nm + n'm'$ different combinations could need to be checked. In specific cases, many combinations can be eliminated due to the limited rotation of the magnetic levitation haptic interface device and the shapes of the simulated tool and environment. In a complex environment, bounding box and recursive tree-based algorithms can also eliminate many combinations in order to reduce computation. A single vertex-to-face contact interpenetration is shown in Fig. 10.1.

Edge to edge contacts must also be detected and their feedback forces and torques calculated. A pair of edges can be checked for contact by projecting both edge segments onto the two face planes adjoining one of the edges, solving for the intersection of the projected lines, and checking the endpoints of both edges. Due to interpenetration, the

edge segments do not actually intersect in space. A sample edge-to-edge contact is shown in Fig. 10.2.

For each calculated contact and interpenetration, the feedback force and torque contribution is calculated and added to the total force and torque to be generated by the haptic device. The feedback from multiple point contacts added together emulate line and face contacts, as shown in Fig. 10.3.

The feedback force and torque contributions from a single contact point are:

$$\mathbf{f}_{mp} = -\mathbf{K}_p \mathbf{x}_{err}, \text{ and} \quad (10.1)$$

$$\boldsymbol{\tau}_{mp} = (\mathbf{x}_{mp} - \mathbf{x}_{mc}) \times \mathbf{f}_{mp}, \quad (10.2)$$

where the stiffness feedback force \mathbf{f}_{mp} from a single point is the material stiffness proportional gain \mathbf{K}_p applied to the contact point interpenetration distance \mathbf{x}_{err} in the direction of the interpenetration. The feedback torque $\boldsymbol{\tau}_{mp}$ to be applied at the center \mathbf{x}_{mc} of the maglev handle from the contact point is the product of the contact point force vector \mathbf{f}_{mp} and the lever arm between the contact point and the body center, $\boldsymbol{\tau} = \mathbf{r} \times \mathbf{f}$. The \mathbf{r} and \mathbf{x}_{err} vectors to generate force and torque feedback from one interpenetration contact point is shown in Fig. 10.4.

Viscous or damping feedback must be generated whenever there is stiffness feedback for stability and to prevent excessively bouncy surfaces and surface chattering while the haptic device handle is held by the user. The damping feedback was applied to the translation axes normal to each interpenetration surface and rotation axes parallel to each surface according to the motion of the flotor center. Applying damping at the center of the tool is simpler than applying damping at each contact point and stability can be ensured more easily.

10.1.2 Cube-In-Cube

The cube-in-cube local simulation demonstration consists of a moving cube confined to a cubical box. Point, edge, and face contacts are possible between the inner and the outer cubes. No edge to edge contacts are possible in this environment, which greatly simplifies the contact calculations required for the simulation. Furthermore, there is no ambiguity as to which face is contacted when there is interpenetration near an edge or corner. There are no surfaces in the environment which the tool can “pop through”.

The graphical display is shown in Fig. 10.5. The inner moving cube is a solid colored block and the fixed outer cubical environment is shown in wireframe mode.

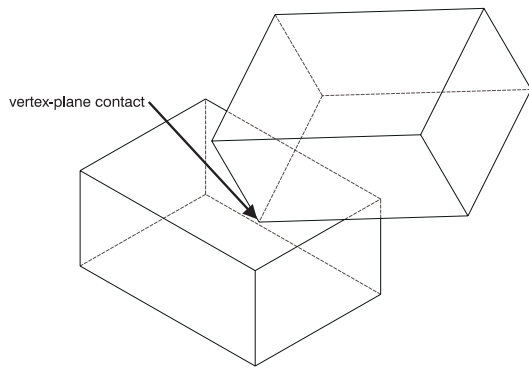


Figure 10.1: Single Vertex-Face Contact

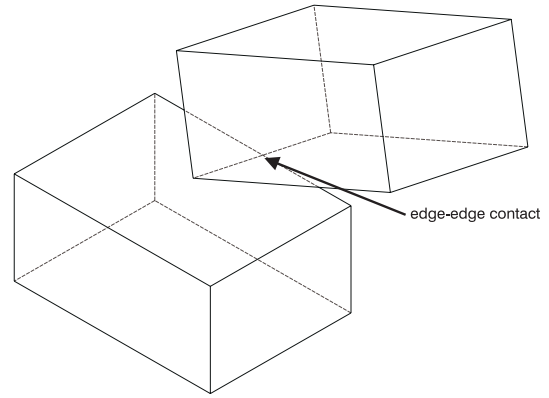


Figure 10.2: Single Edge-Edge Contact

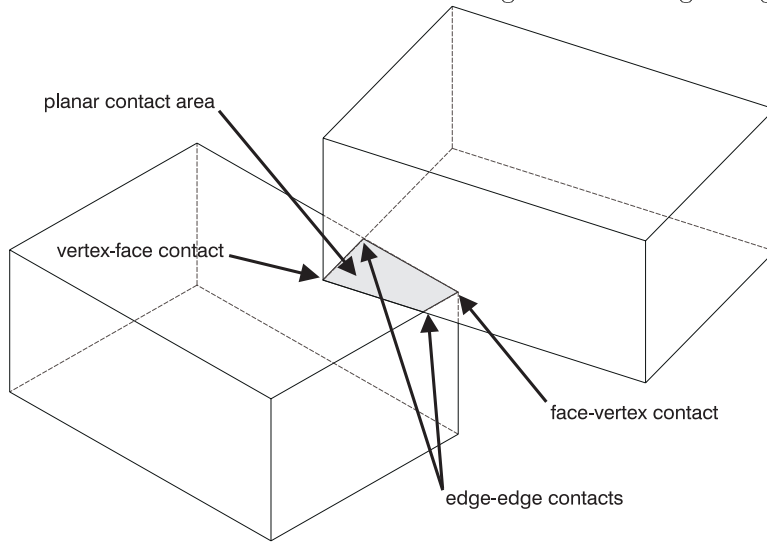


Figure 10.3: Multiple Contacts

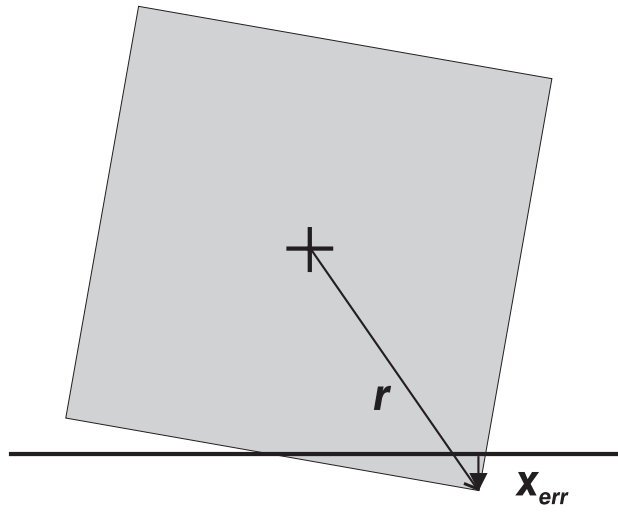


Figure 10.4: Force and Torque Feedback from Contact Point

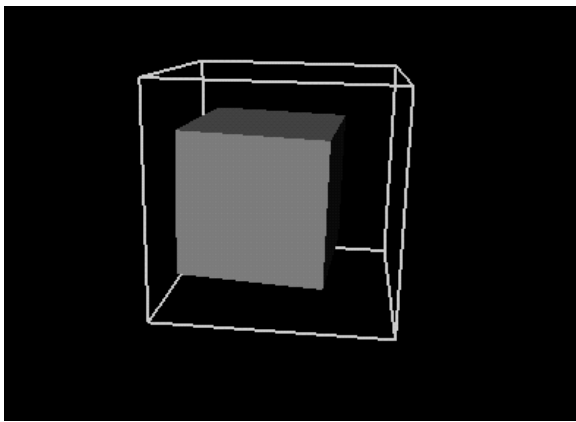


Figure 10.5: Local Cube-in-Cube Simulation

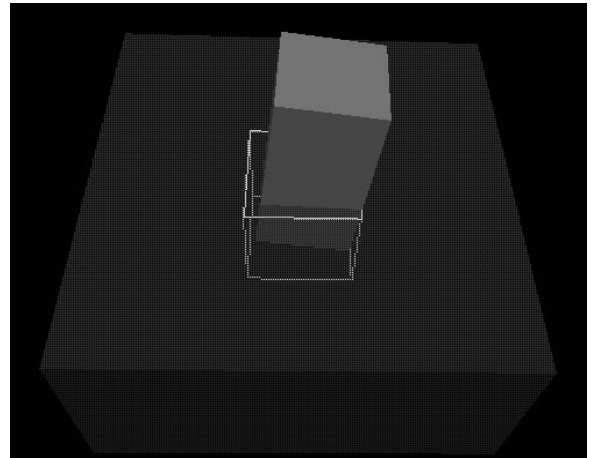


Figure 10.6: Local Peg-in-Hole Simulation

10.1.3 Peg-in-Hole

The peg-in-hole simulation is similar to the cube-in-cube simulation but also includes potential edge-to-edge contacts and “pop through” effects due to the convex edges in the environment formed by the top rim of the hole.

The possibility of edge-to-edge contacts and interpenetration must be checked between each of the peg edges and the four top rim edges of the hole. For each peg edge, if at least one endpoint of the edge is below the level of the hole, then both lines are projected onto the xy plane and their intersection is calculated. The hole rim edges are parallel to the x and y axes, so for a $2l \times 2l$ square hole centered on the origin and a peg edge segment between (x_0, y_0, z_0) and (x_1, y_1, z_1) , the two-point form of a straight line:

$$\frac{y - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1}, \quad (10.3)$$

can be used to calculate the x_{int} and y_{int} coordinates of the intersections of the peg and hole edges projected into the xy plane:

$$x_{int} = \pm l, \text{ and} \quad (10.4)$$

$$y_{int} = \frac{y_1 - y_0}{x_1 - x_0}(\pm l - x_0) + y_0 \quad (10.5)$$

for the edges of the hole at $y = \pm l$ and

$$x_{int} = \frac{x_1 - x_0}{y_1 - y_0}(\pm l - y_0) + x_0, \text{ and} \quad (10.6)$$

$$y_{int} = \pm l \quad (10.7)$$

for the edges at $x = \pm l$.

If the projected intersection point x_{int}, y_{int} lies between the endpoints of the current hole rim edge, then there is indeed contact of the peg and hole edges and interpenetration between their rigid-body polyhedron representations.

For edge-to-edge contacts between the rim of the hole and the bottom edges of the peg, the contact normal force is approximated to be vertical and the vertical interpenetration distance is used for surface stiffness feedback. A more physically exact representation would be to calculate the cross product of the two edge segment vectors to obtain a contact force direction unit vector orthogonal to both edge vectors. The vertical interpenetration in the approximate case is the distance between the top of the hole t and the intersection of the peg base edge and the wall of the hole. For the hole walls parallel to the xz plane, the vertical interpenetration distance d is

$$d = t - \left[\frac{z_1 - z_0}{x_1 - x_0} (\pm l - x_0) + z_0 \right] \quad (10.8)$$

and for the hole walls parallel to the yz plane the interpenetration is

$$d = t - \left[\frac{z_1 - z_0}{y_1 - y_0} (\pm l - y_0) + z_0 \right]. \quad (10.9)$$

For edge-to-edge contacts of the rim edges of the hole and the side edges of the peg, the contact normal force is approximated as horizontal and so the horizontal interpenetration is used for stiffness feedback. The horizontal interpenetration is found by calculating the intersection point of the peg edge with the plane of the top of the hole. For the hole edges parallel to the xz plane, the interpenetration d is

$$d = \pm \left[\frac{t - z_0}{z_1 - z_0} (x_1 - x_0) + x_0 \right] - l \quad (10.10)$$

and for the hole edges parallel to the yz plane,

$$d = \pm \left[\frac{t - z_0}{z_1 - z_0} (y_1 - y_0) + y_0 \right] - l. \quad (10.11)$$

The display for the peg-in-hole simulation is shown in Fig. 10.6. For added clarity, the base object around the hole was defined to be semitransparent and the edges of the hole are outlined in wireframe mode.

10.2 Surface Characteristics

The simulation environment in this demonstration consists of a single fixed flat surface. The haptic tool controlled by the haptic device is a pointed probe tip and the user can move the tip position anywhere above or on the simulated surface. Each quadrant of the surface has a different characteristic surface model. The surface models are (1) a frictionless and flat, (2) flat with friction, (3) textured with no friction, and (4) texture and friction together. For each model the texture and friction forces are added to the solid contact constraint forces to give the user the perception of surface characteristics. The graphical display for this simulation is shown in Fig. 10.2.

10.2.1 Coulomb Friction

All modeled surfaces must have some limited degree of damping or viscous friction in order to prevent chattering. Coulomb or dry friction is based on a different model which involves a stick-slip motion, however. The right half of the surface shown in the figure has added Coulomb friction.

In this basic stick-slip friction model, when the haptic device handle (and the probe tip in the simulation) contacts the friction surface, a position control setpoint is established

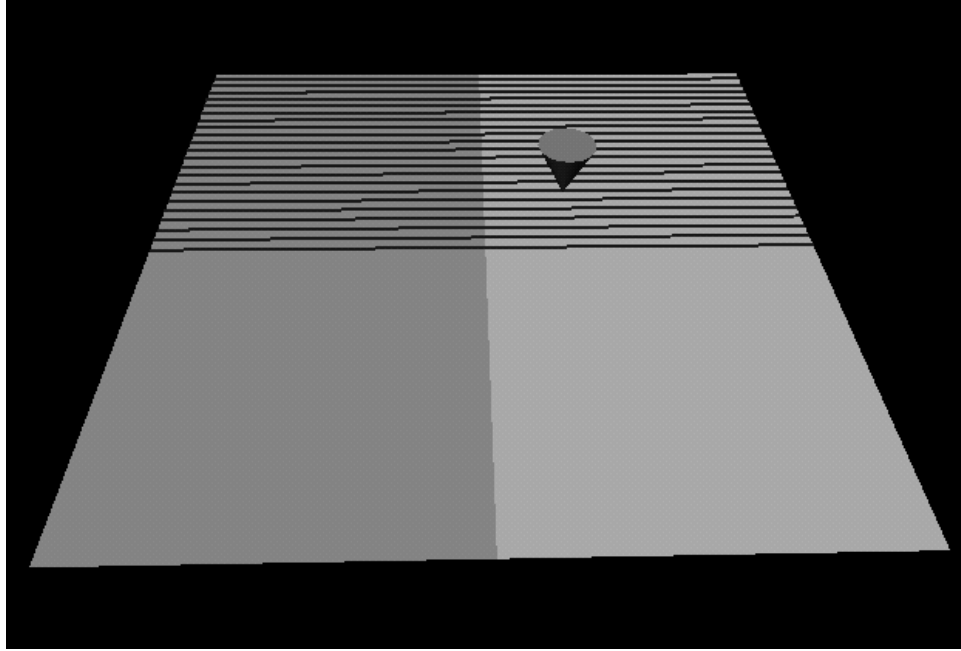


Figure 10.7: Surface Characteristic Simulation

on that surface at the current position. While the probe remains in contact with the surface, whenever the magnitude of the force on the probe tip parallel to the plane exceeds the normal force times the Coulomb friction coefficient μ , the position control setpoint is reestablished at the new probe position. For a surface parallel to the xy plane, the friction threshold is

$$f_x^2 + f_y^2 > (\mu f_z)^2. \quad (10.12)$$

10.2.2 Texture

The half of the surface farthest from the user in the figure is modeled with a faceted surface of grooves with a sawtooth profile. This texture simulation model was shown in Fig. 9.2(b). For grooves running parallel to the x axis with height h and width w , the added surface height z_s due to the texture at a given y is:

$$z_s(y) = \begin{cases} 2h \frac{y \bmod w}{w}, & \text{if } \frac{y \bmod w}{w} < 0.5 \\ 2h(1 - \frac{y \bmod w}{w}), & \text{otherwise} \end{cases} \quad (10.13)$$

and the added forces due to the texture model are:

$$f_z = K_p z_s, \text{ and} \quad (10.14)$$

Parameter	Value
Translation Gains:	5.0 N/mm Stiffness, 0.06 Ns/mm Damping
Rotation Gains:	25.0 Nm/rad Stiffness, 0.5 Nms/rad Damping
Force Feedforward:	5.0 N
Moving Cube Size:	10.0 mm
Fixed Box Size:	15.0 mm

Table 10.1: Cube in Cube Experiment Parameters

$$f_y = \begin{cases} K_p z_s, & \text{if } \frac{y \bmod w}{w} < 0.5 \\ -K_p z_s, & \text{otherwise} \end{cases} \quad (10.15)$$

with stiffness gain K_p .

10.2.3 Impact

The surface characteristic simulation includes an impact model. A force impulse \mathbf{f}_i proportional to the velocity normal to the surface \mathbf{v}_n is generated at the contact point when collision is detected,

$$\mathbf{f}_i = K \mathbf{v}_n, \quad (10.16)$$

where K is a proportionality constant. Due to the high control bandwidths of the magnetic levitation haptic device, impacts with the simulated surface actually produce audible clicks, just as actual impacts of a tool with a solid surface.

Audible impacts are also produced on the magnetic levitation device using just surface stiffness and damping if the gains are sufficiently high. The addition of the impact impulses reduces the interpenetration of the tool into the virtual surface, improving the stability of the device controller and generating a smoother, more realistic feel of solid surface impact to the user.

10.3 Experimental Results

The following sample experiments were run at a 1000 Hz control rate. The test data were recorded using a data logging software package¹ running on the realtime device controller.

10.3.1 Cubes

¹StethoScope, Real Time Innovations

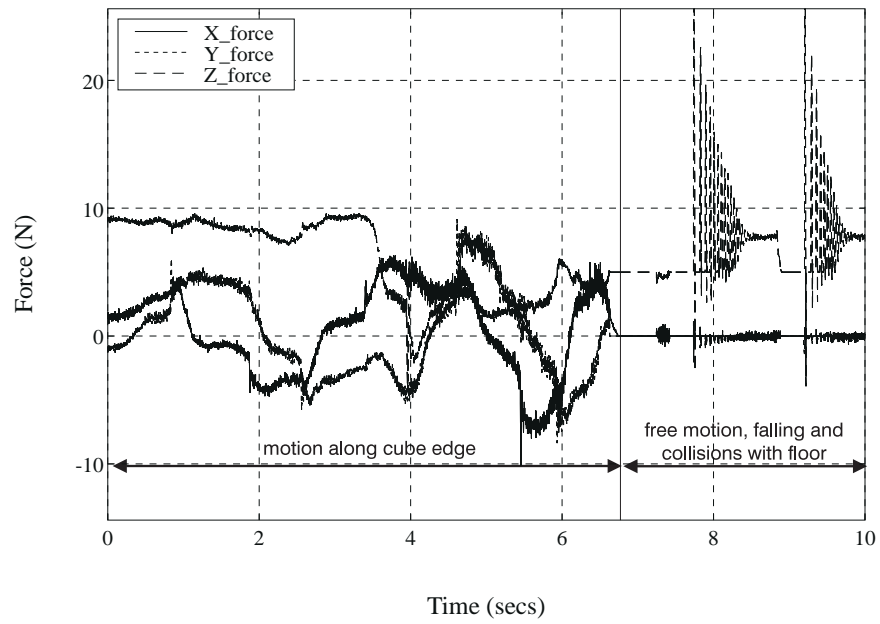
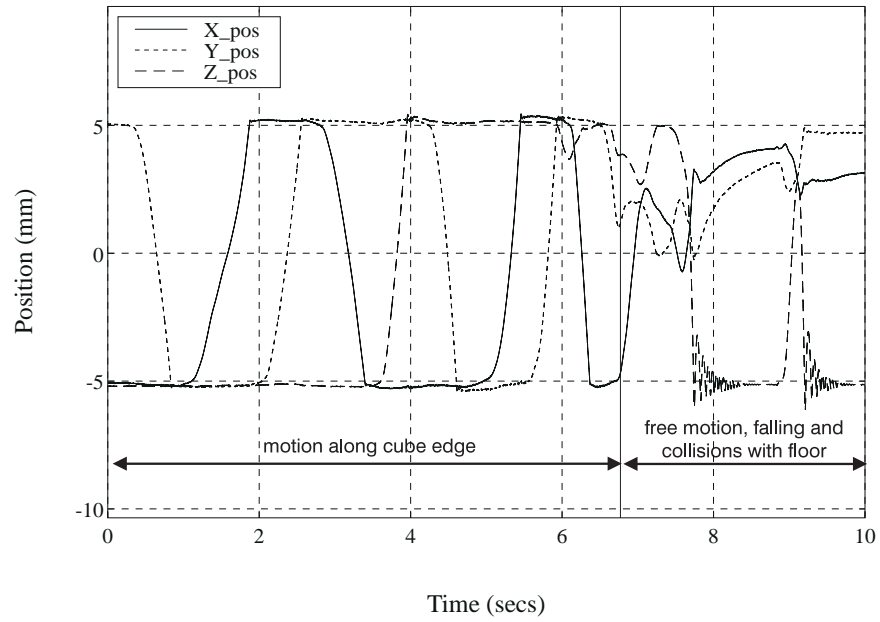


Figure 10.8: Position and Force Data from Cube-in-Cube Simulation

Parameter	Value
Translation Gains:	6.0 N/mm Stiffness, 0.04 Ns/mm Damping
Rotation Gains:	25.0 Nm/rad Stiffness, 0.5 Nms/rad Damping
Force Feedforward:	5.0 N
Coulomb friction coefficient:	0.3
Texture Groove Height:	0.25 mm
Texture Groove Width:	0.25 mm

Table 10.2: Surface Simulation Experiment Parameters

The parameters of a sample recorded cube-in-cube haptic interaction are given in Table 10.1. The position and force data are plotted in Fig. 10.8.

To obtain the sample data, the inner cube was moved along the bottom and top edges of the outer cube, then held in the center and dropped twice on the floor of the outer cube. Since there is 5 mm of clearance between the modelled cube and its environment, the position data range from -5 to 5 mm. Since the collision between the inner cube and the floor was very lightly damped, the cube bounces a few times at first and continued to oscillate in force until damped out. The force generated by the device during the experiment feels realistic and appears to be as expected, but added vibrations due to sensor noise and limited damping are apparent in the output data.

10.3.2 Surface Characteristics

To obtain the data of Fig. 10.9, the probe tip in the simulation was moved by the user in circles over each of the four quadrants of the environment surface in succession for approximately two seconds each. The simulation and control parameters are listed in Table 10.2.

The distinct surface characteristics of each of the four quadrants can be seen in the position and force data of Fig.10.9. For the first two seconds, both the position and force data are smooth, since the surface in the $(-x, -y)$ quadrant has neither texture nor friction.

For approximately the next two seconds, the probe was in the $(-x, +y)$ quadrant, with grooves aligned with the x axis. The x position and force data are still smooth due to the lack of friction, but the motion in the y direction is more jagged and flattens at the local maxima and minima due to the grooves in the surface resisting smooth motion. The force traces in y and z show high frequency vibrations due to the collisions with the groove surface facets and the vertical motion over the peaks and valleys of the sawtooth grooves can be seen in the z position data.

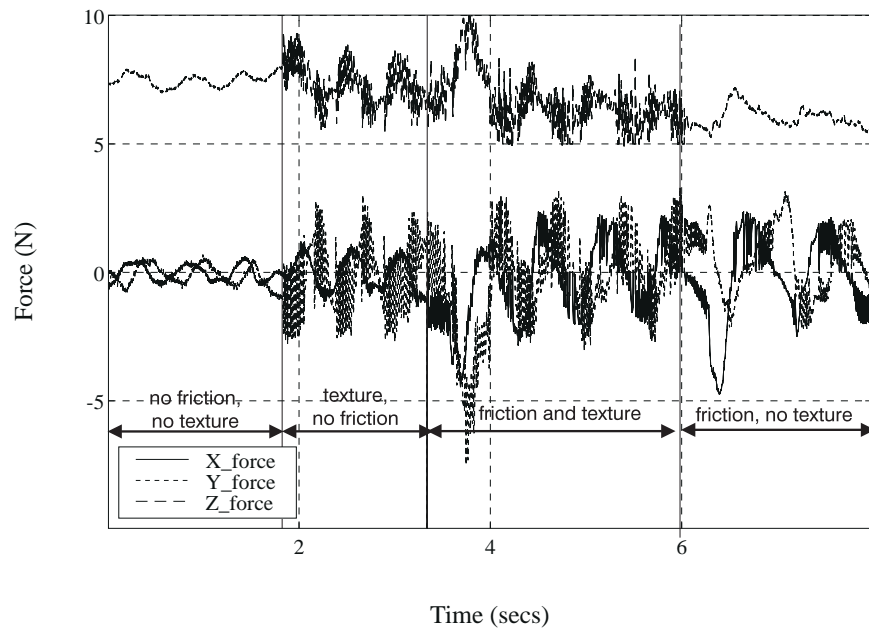
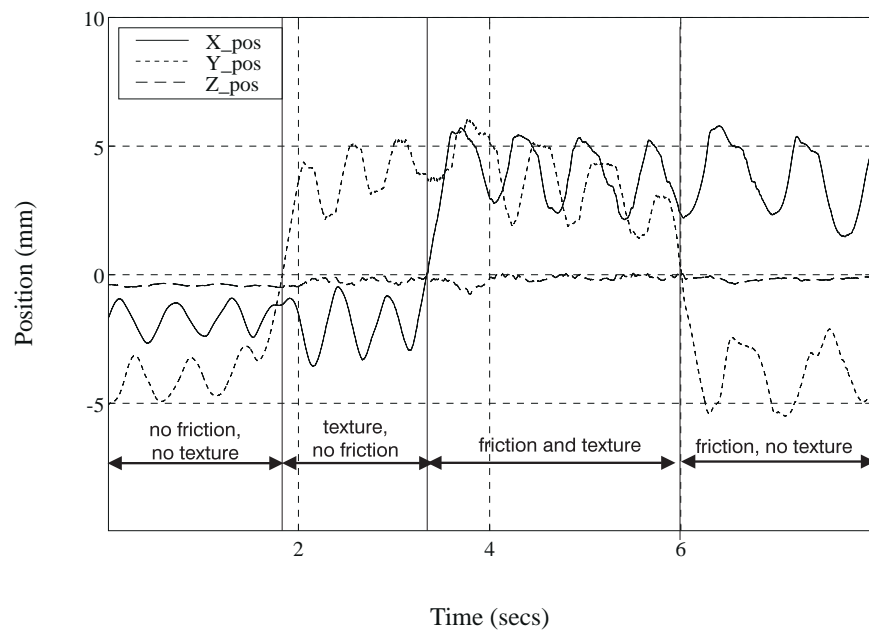


Figure 10.9: Position and Force Data from Surface Simulation

In the 3.5 to 6 second period, the probe was moved into the $(+x, +y)$ quadrant with both texture grooves and Coulomb friction. Here the x position and force data is somewhat less smooth due to the stick-slip of the friction model. The vertical motion of the probe tip over the grooves can still be seen in the z position data.

In the final period from 6 to 8 seconds the probe was in the $(+x, -y)$ quadrant, with Coulomb friction and a flat surface. The x and y position plots are less smooth than in the $(-x, -y)$ quadrant due to friction and vibrations can be seen in the x and y forces. The z position and force data are quite smooth since the surface is flat.

10.4 Discussion

Performing all the environment simulation calculations on the local control processor can provide the fastest possible performance response since the environment simulation is recalculated at every sample update of the controller and there is no delay due to communications with a simulation on a different processor. The complexity of the environment is limited by the local processor, however. The performance and response of the local simulations provide a point of comparison for the quality of haptic interaction with more complex simulations executed on the host workstation.

Use of the Open Inventor graphics modeling libraries enabled rapid, easy development of environments, but the dynamic haptic control simulations required extensive hand coding to calculate contact forces and tuning to realize stable, realistic haptic interaction. The control and modeling for each of the locally executed simulations had to be done specifically for each environment in order to achieve the fastest possible execution time and realize the best haptic performance. The gains and other parameters for the simulations were also individually tuned to provide the most realistic impression of haptic interaction. Specific programming for each different environment would be too time consuming and impractical for a general haptic interaction system, so it would be better to interface with general, existing environment simulation software if it is possible to realize comparable performance and response in the haptic interaction with a simulation which is separated from the device controller. Virtual coupling and intermediate representation methods were adapted to the magnetic levitation haptic interface device implemented with the aim of providing the response of the local simulations with a fully developed and general, dynamic, rigid-body, interactive simulation package.

Chapter 11

Haptic User Interface

In addition to the haptic interface device and the physical simulation, a graphical display and a means of easily modifying simulation environments, operational modes, and other parameters of the interface during operation are necessary to make a haptic interface system into a tool that is both versatile and easy to use.

Implementation of the graphical display of the simulated environment is straightforward on a graphics workstation. Simple, basic lighting and reflectance models were used for graphical rendering so that the display could operate at a reasonably smooth 20-30 Hz frame rate while allowing the simulation computations to run as fast as possible.

Various operational modes and parameters were implemented and included in the haptic interface system to provide the user with a degree of flexibility and customization in the system interaction and to overcome the limitations of the haptic device when compared to real world tool-based object manipulation. For example, a number of different modes and parameters can be used in combination to move the virtual tool over arbitrarily large distances and rotations in the simulated environment so that the size of the virtual environment is not limited to the range of motion of the maglev device.

11.1 User Interface Control Panel

A graphical user interface control panel was implemented during the course of the development of the haptic interface system so that operating parameters could be tuned and different environments and operational modes could be tested easily during operation. The same graphical user interface control panel allows all the operational modes and parameters to be set by simply pointing, clicking, and dragging with the workstation mouse, so it is also suitable to be used as a control interface for any end user.

The user interface control panel is shown in Fig. 11.1. It was designed and implemented using the Xforms graphical user interface toolkit free software package [59] and contains regular buttons, toggling “lightbuttons”, “radio” buttons for selecting among multiple options, and sliders. The control panel is divided into sections to control the following aspects of the haptic interface system:

- position of the viewpoint of the graphics window
- parameters, actions, and modes of the dynamic simulation
- selection of the virtual tool and its environments, tool actions, and setting tool material properties
- stiffness and damping gains for virtual coupling
- toggling control modes and setting their parameters
- motion scaling
- motion offsets

A message box at the bottom of the GUI window indicates when the rotation limit described in Section 5.2.4 is exceeded by the user.

11.2 Graphical Display

The dynamic simulation environment during haptic interaction is displayed using the OpenGL graphics library routines on a Silicon Graphics workstation or, equivalently, using the MesaGL package on a Sun Sparc workstation. A schematic representation of the intercommunication between the device, controller, simulation, and graphical display systems is shown in Fig. 11.2. The physical simulation supplies data for the graphical display in real time as well as two-way interactive communication with the haptic device controller.

Haptic interaction and 3-D graphics complement each other very well since the combined interaction system recreates human visual/haptic interaction with real objects. Since human haptic perception of object shapes is distorted and inaccurate depending on the hand and arm position [60] and visual cues are dominant over haptic perception [61], small position and force errors in the haptic device control should not be perceived by the user during interaction if object shapes are also graphically displayed.

The graphical representation of the simulated environment includes shadows vertically projected onto the floor from the objects in the simulation. The shadows were found by

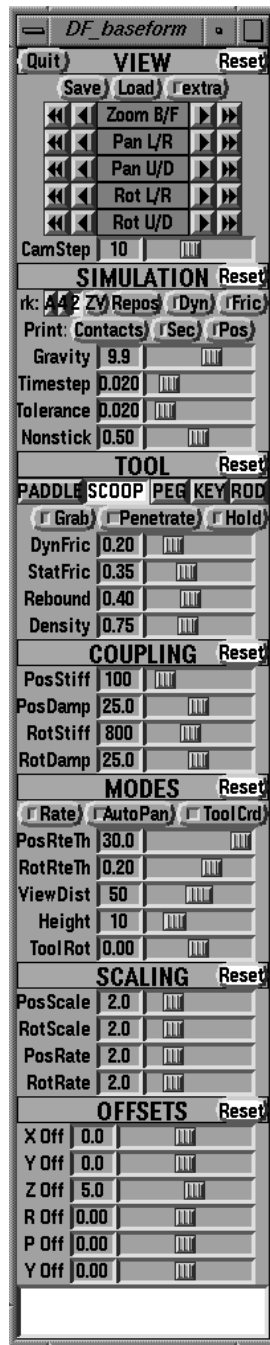


Figure 11.1: User Interface for Haptic Interaction with Dynamic Physical Simulations

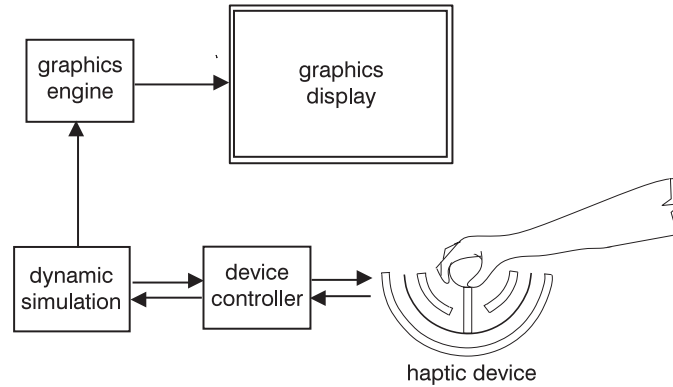


Figure 11.2: Haptic and Visual Interface System

various users to be useful visual cues to vertically align objects for tasks such as catching thrown objects and peg in hole insertion. The shadows compensate for the lack of depth perception while viewing a three dimensional environment on a flat monitor screen. The existing graphical display system could be extended into a true three dimensional stereo vision display system with the addition of LCD shuttered glasses synchronized with the refresh rate of the display monitor and graphical generation of separate images for each eye.

11.3 Mouse Interface

Clicking and dragging the mouse in the graphics display window creates a virtual spring in the simulated environment at the point on the object surface indicated by the mouse pointer. The virtual spring can then be used to pull objects around the simulated environment while the mouse button is held down. This interface feature was implemented by Sebastian Grassia and David Baraff for CORIOLIS simulations.

11.4 Viewpoint Motion

Viewpoint controls are provided to enable the user to zoom, rotate, and pan the view of the graphics window to display areas of interest in the simulated environment. For each of the five possible camera translations and rotations, four buttons are provided to command large or small steps in either direction. A slider is used to set the sizes of the camera motion steps. Specific viewpoints can also be loaded from or saved to files; a pop-up dialog box queries the filename. The button marked “extra” is not implemented.

11.5 Simulation

The controls in this section affect the operation of the simulation itself.

11.5.1 Integration Method

A set of three buttons selects the numerical integration method used in the simulation for modeling Newtonian rigid-body dynamics. The selections are the fifth-order Runge-Kutta integration method with adaptive timesteps, fourth-order Runge-Kutta, and second-order or midpoint Runge-Kutta. During typical operation, the 2nd-order Runge-Kutta integration method is selected for greatest speed, although the accuracy and stability of the other methods is greater.

11.5.2 Motion Recovery

The zero velocity button **ZV** sets the velocity of all the objects in the simulation to zero. This capability is useful if objects in the simulated environment are accidentally given a large velocity due to being struck or becoming suddenly unstuck. The button can stop the objects before they go out of reach or fall off the edge of the simulated world. A reposition button **Repos** sets all the objects in the simulation back to their initial positions with zero velocity. This button is the useful one when objects have already gone out of reach or off the edge of the world.

11.5.3 Print Operating Information

Three buttons cause runtime information of the simulation to be printed to the shell window. The **Contacts** button prints the current position and normals for all the points on the tool which are contacting other objects in the simulation. The tool contact data are used for the contact point intermediate representation described in Chapter 13.

Toggling the **Sec** button causes the real time and simulation time elapsed for each iteration of the simulation to be printed to the shell periodically, along with the average times and the number of elapsed simulation iterations and graphics frames. The real and simulated time intervals do not match exactly due to factors such as momentary high computational load, integration discontinuities due to object collisions, and nondeterministic delays due to other operating system processes and network communications.

The **Pos** button causes the current tool position and orientation to be printed to the shell periodically.

11.5.4 Other Simulation Parameters

The **Gravity** slider sets the gravity on the objects in the simulation. **Timestep** sets the attempted simulation time step per iteration.

Tolerance sets the tolerance for rigid-body contacts in simulation. Increasing the contact tolerance speeds up the simulation time slightly fewer calculation iterations are required to find contact points, but there is a corresponding loss of detail in the exact positions of the calculated contact points. For detailed environments and very small clearances, it would be preferable to set the tolerance lower.

The **Nonstick** parameter is a means to diminish the effects of the simulation drastically slowing down when multiple objects in the simulation are tightly wedged together, as described in Section 9.1. If the returned simulation time for one integration step T is less than the nonstick parameter N times the desired simulation timestep T_d ,

$$T < NT_d, \quad (11.1)$$

then the velocity of the virtual tool is set to zero, which tends to cause the simulation to run at a typical speed again without perceptibly changing the dynamics of the simulation.

11.6 Tool

This section of the graphical user interface control panel controls the parameters of the haptic tool in the simulation.

11.6.1 Shape

The topmost row of buttons in the **TOOL** section of the control panel is used to select one of the different tool shapes and environments that have been modelled for the physical simulation. The selection of tool shapes is pictured in Fig. 11.6.1 and described below:

PADDLE consists of handle attached to flat square panel

SCOOP handle attached to flat square panel turned up at back edge

PEG simple square peg for peg-in-hole simulation

KEY handle with attached tabs for key-in-lock simulation

ROD single thin stick to easily grab objects

For the peg tool, a set of holes of different sizes was created in the environment. The key tool has a simple keyhole and sliding bolt structure in its environment.

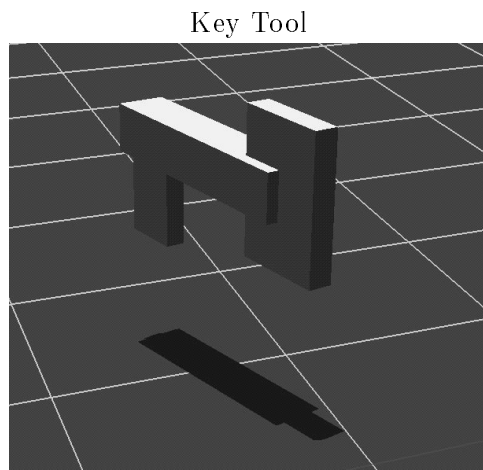
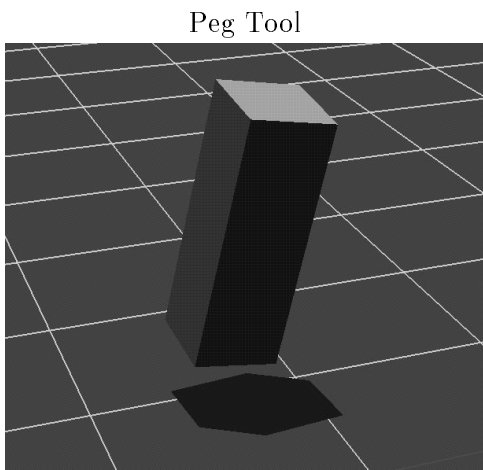
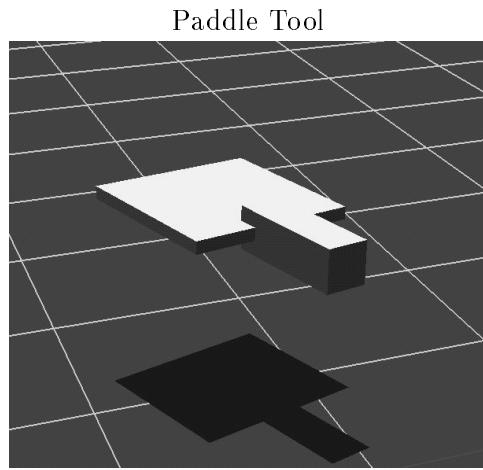
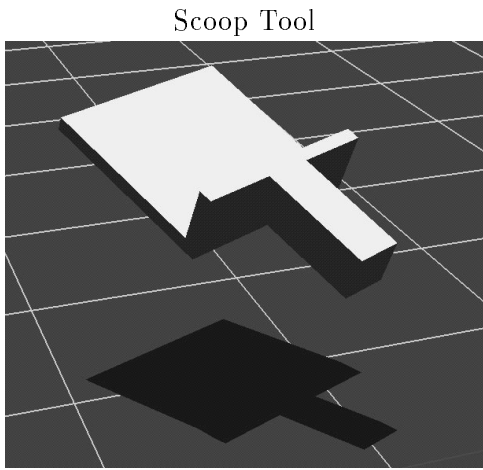


Figure 11.3: Haptic Tool Selection in Coriolis Simulations

11.6.2 Actions

Special added actions that can be performed with the tool in the simulated environment have been implemented for easier manipulation of other objects in the environment. When the **Grab** button is pushed, the next object that comes into contact with the tool becomes rigidly attached and can be moved around the environment as part of the tool. Pushing the button again releases the grabbed object. This feature of the interface enables the user to easily move and maneuver objects in the simulated environment from place to place.

The **Penetrate** button allows the tool to pass through other objects in the simulation, enabling the user to move the tool directly to other areas in the environment without having to move around any other objects in the way. The **Hold** button is currently unimplemented.

11.6.3 Material Properties

The material and surface properties of the selected tool can be set from the control panel. **DynFric** sets the dynamic friction coefficient and **StatFric** sets static friction for the Coulomb stick-slip sliding friction model in the simulation. The elasticity of the tool in collisions is set by the **Rebound** slider. **Density** sets the density of the tool material, which determines the mass of the different tools which can be selected in the simulation.

11.7 Virtual Coupling Parameters

The stiffness and damping of the virtual coupling interface between the haptic device and the tool in the simulation in translation and rotation are set by the four sliders in the **COUPLING** section. The virtual coupling implementation is described in detail in Chapter 12.

11.8 Control Modes

Three control mode options were implemented to assist tool motion and viewing in the simulated environment. The **Rate** mode, automatic view control **AutoPan** mode, and tool coordinate **ToolCrd** mode can all be set independently. When all three of these modes are enabled together, the result is a video game or flight simulator-like “fly-around” mode: The position and orientation of the haptic device handle determine the translational and rotational velocities of the tool in the simulation, the graphics viewpoint follows the tool around, and the coordinate frame of the haptic device remains aligned with the frame of the tool in the graphics window.

11.8.1 Rate Control Mode

In rate control mode, the position of the haptic device handle determines the velocity of the tool in the simulation. The zero offsets reflect the distance the tool has traveled while in rate mode. Rate control mode can be either manually selected by clicking the button on the user interface control panel window, or automatically invoked as the levitated handle approaches its motion limits. Two sliders specify the radial distance and angular rotation thresholds for the haptic device handle where rate control mode is started automatically.

11.8.2 Automatic View Control

The automatic viewpoint control mode tracks the position and orientation of the tool so that the zero position of the tool is centered in the display at a desired distance from the viewpoint and the major axis of the tool zero position is aligned with the viewing direction. Two sliders specify the graphics window viewpoint following distance behind the tool and the height of the viewpoint.

11.8.3 Tool Coordinate Mode

In the regular coordinate mode, the world coordinate frame of the haptic device is aligned with the simulated world coordinate frame with the x and y directions of the grid lines on the floor of the simulated environment corresponding to the right/left, forward/back motion of the haptic device handle. If the angle between the viewing axis and the simulation world y axis is less than approximately 30° , then the coordinate mapping from the haptic device to the simulated environment is still natural and intuitive for the user. If the viewpoint is rotated by larger angles, the mismatch between the view and device frames becomes confusing. Selection of the tool coordinate mode will keep the device frame oriented to the viewpoint frame rather than the simulation world frame.

11.9 Mapping Motion between Device and Simulation

A one-to-one mapping of the levitated handle motion to the motion of the selected tool in the simulated environment provides sensitive interaction for fine fingertip operations such as insertion, but does not enable the user to move the tool over larger distances in the simulated environment. Therefore, additional control modes have been added to the haptic user interface to enable the user to move and operate the haptic tool over arbitrarily large distances and rotations in the virtual environment in a natural, intuitive way. During

interaction the user can change the scaling factors and the zero position offsets between the levitated handle and the simulated environment, switch to rate control mode, and move the graphics viewpoint for displaying the simulation.

The actual mapping from the actual levitated handle position and orientation vectors \mathbf{x}_{device} and $\boldsymbol{\theta}_{device}$ to the simulation setpoint vectors \mathbf{x}_{setp} and $\boldsymbol{\theta}_{setp}$ is given by

$$\mathbf{x}_{setp} = \mathbf{x}_{offset} + \mathbf{x}_{scale}\mathbf{x}_{device}, \text{ and} \quad (11.2)$$

$$\boldsymbol{\theta}_{setp} = \boldsymbol{\theta}_{offset} + \boldsymbol{\theta}_{scale}\boldsymbol{\theta}_{device}. \quad (11.3)$$

The variable scaling factors \mathbf{x}_{scale} and $\boldsymbol{\theta}_{scale}$, and offsets \mathbf{x}_{offset} and $\boldsymbol{\theta}_{offset}$ for rotation and translation can be set manually by the user during operation with sliders in a graphical user interface control panel. Changing the translation scaling factor does not appear to affect the user's perception of haptic interaction with physical objects, but when the rotation scaling factor is increased and the tool is manipulated in tightly constrained environments such as the peg-in-hole, users have commented that the reaction torques feel unrealistic.

The offsets determine the position and orientation of the tool in the simulation when the haptic device handle is in the centered position. These offsets can be set so that the full sensitivity and motion range of the haptic interface device can be made available at any point in the simulated environment. The user can easily switch from making large motions across the entire simulation world to fine motions in one spot.

The sliders in the **SCALING** section set the \mathbf{x}_{scale} and $\boldsymbol{\theta}_{scale}$ parameters. The scaling gains for the rate control mode can also be set independently.

The set of six sliders in the **OFFSETS** section is used to set the offsets between the maglev device flotor and the haptic tool in the simulation. These offset values determine the position and orientation of the virtual tool in the simulated environment which corresponds to the center zero position and orientation of the flotor handle. Use of the offset sliders enables the user to quickly and easily move the virtual tool to any position and orientation in the simulated environment. The same result could also be obtained by switching into rate mode and moving the haptic device handle. The offsets are updated automatically when the rate control method is active.

11.10 Comments

The combination of different operating modes such as automatic view tracking, variable scaling and offsets, and rate-based control enables the user to move the virtual haptic tool easily over arbitrarily large distances in the virtual environment. When the virtual tool is

in the desired local neighborhood, the user can then switch back to a smaller scaling ratio between the virtual and actual forces and motions. The particular combination of control modes used to reposition the haptic tool is up to the user. The control panel of sliders and buttons in the graphics display allows the user to change and modify control modes during haptic interaction. It is hoped that use of the various haptic interface system user interface features will be as automatic for users as with a computer mouse.

11.10.1 Interfaces for Different Users

The user interface pictured in Fig. 11.1 provides user access to all the relevant features and parameters of the simulation and the interface to the haptic device. Depending on the particular simulated task and the experience of the user, most of the features of the current graphical user interface may not be necessary. A beginning user may desire a simple, intuitive interface, while a more experienced user would want finer control of the virtual tool for more sensitive tasks.

In the demonstrations of the haptic interface system so far, a wide range of differences in the use of the haptic simulation system has been observed, depending on the background and experience of the user. The device handle may be gripped in different ways and different approaches and strategies are used to perform tasks in the simulated environment. There is an initial short adjustment period for users: The first motions and actions performed with the device are tentative and uncertain, but the user's proficiency rapidly improves over the next few minutes of use. The short period required for the user to become accustomed to the capabilities of this haptic interaction system may be a significant advantage of the magnetic levitation device over large and awkward haptic devices and those which require gloves or exoskeletons.

11.10.2 Interface Controls on Handle

To use the present user interface control panel, the user must either release the magnetic levitation device handle to use the computer mouse, or operate the mouse and the magnetic levitation device in opposite hands. Either option is awkward and inconvenient. The user interface to the haptic interaction system would be improved if selected features and settings of the GUI control panel of Fig. 11.1 could be set directly by the user with the same hand which is grasping the haptic device handle.

Embedding some switches, buttons, or control knobs in the handle of the haptic device would enable the user to easily use the haptic interface and set the parameters of the interface simultaneously with the same hand. For example, it would be much more convenient to grab objects, select different tool shapes, switch to rate mode, or change the scaling parameters with the thumb or forefinger while grasping the haptic device handle with the same hand. A “deadman” switch on the handle could be used to prevent large forces or fast motions of the magnetic levitation flotor while it is not being held by the user. Commercial game joysticks are often designed this way with buttons or switches located on the joystick itself to be operated by the user’s thumb and forefinger while gripping the joystick.

The design of the connector and the handle socket of the magnetic levitation flotor accounts for potential future additions such as a force sensor, a 1-DOF gripper, or interface controls such as buttons, switches, and knobs. The device handle can be changed without difficulty as described in Section 2.2.3, the rod on which the handle is mounted is hollow to accommodate wiring from the flotor connector to the handle, and the round connector socket at the base of the magnetic levitation flotor has many extra pins available for future additional wiring connections. The design and implementation of any added controls on the device handle has been left to future development.

Chapter 12

Virtual Coupling

The idea of coupling a simulated object's motion with a haptic interface device by virtual stiffness and damping links was first mentioned by Colgate *et al*[16]. A generalization and stability analysis of virtual coupling has been given by Adams and Hannaford [62].

In this implementation, the physical simulation and control of the magnetic levitation device execute independently on separate processors. The simulation and controller exchange the current positions and orientations of the levitated handle and the virtual tool over a standard 10 Mb/s Ethernet 10BaseT network at the simulation update rate. The positions from each system act as impedance control setpoints for the other, with position error and velocity feedback on each system as a virtual spring and damper coupling between the two systems. The setpoints from the simulation are interpolated by the controller at the faster device control rate so that the user feels smooth sliding contacts without chattering from the slower updates of the simulation. The simple feedback coupling between the two systems enables the overall stiffness and stability of the combined system to be tuned easily and provides realistic haptic user interaction.

12.1 Physical Simulation

The CORIOLIStm simulation was tuned to run as fast as possible for realistic realtime interaction. The midpoint or second-order Runge-Kutta integration method was selected for speed and simplicity. A 100 Hz update rate was achieved for simulations with up to 10 polyhedral objects of 6-12 vertices each on an SGI Indigo 2 or Octane workstation. If the multiple rigid body contact states are sufficiently complex, such as when a chain of several objects collides nearly simultaneously, the simulation update rate occasionally slows down for one or two frames.

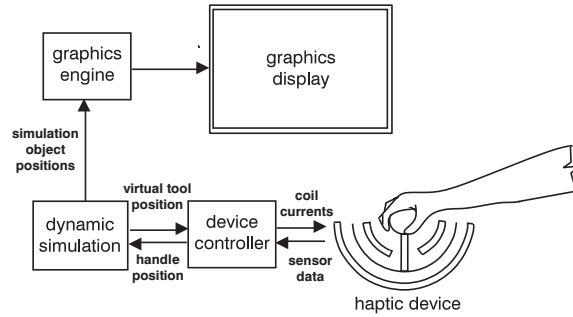


Figure 12.1: Haptic and Visual Interface System

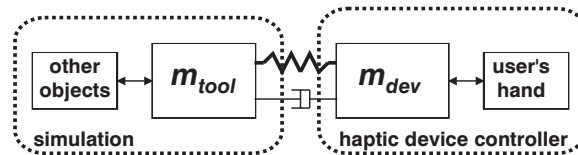


Figure 12.2: Virtual Coupling of Simulation and Device

The 3-D rendered graphics display is updated asynchronously in the background after the simulation dynamics are updated by an interval timer signal handler. The resulting frame rate of the graphics display is typically 15-30 Hz, depending on the complexity of the simulation and any other processes executing on the workstation.

12.2 Coupled Simulation and Control

The magnetic levitation haptic interface device and the realtime dynamic simulation system described in Chapter 9 have been integrated to enable the user of the haptic interface device to directly physically interact with the objects in the simulation. The simulation and the device controller can each operate independently and communicate using TCP/IP socket connections over a standard Ethernet network. A schematic representation of the intercommunication between the device, controller, simulation, and graphical display systems is shown in Fig. 12.1.

Interactive simulations have been implemented on the local control processor, as described in Chapter 10, but they are limited to static environments with fewer than 10 to 15 total vertices in the models of the tool and its environment. The present control processor is not sufficiently fast to perform dynamic simulation and collision detection on multiple moving objects while calculating the sensor kinematics and feedback control at a rate sufficient for stable stiff contacts.

The virtual coupling between the realtime simulation and the maglev haptic device controller as proposed by Colgate [16] is shown in Fig. 12.2. In our implementation, the present position and orientation vector of each system is periodically sent to the other system to act as its control setpoint. The stability and responsiveness of the simulation and device can be set by the spring and damper gains of the coupling as seen from either side. The generated forces on the maglev haptic device \mathbf{f}_{dev} and the virtual tool in the simulation \mathbf{f}_{tool} are given by:

$$\mathbf{f}_{dev} = \mathbf{f}_g + \mathbf{K}_p(\mathbf{x}_{tool} - \mathbf{x}_{dev}) + \mathbf{K}_v(\mathbf{x}_{dev} - \mathbf{x}_{devprev})\mathbf{r}, \text{ and} \quad (12.1)$$

$$\mathbf{f}_{tool} = \mathbf{f}_{other} + \mathbf{K}_{spring}(\mathbf{x}_{dev} - \mathbf{x}_{tool}) + \mathbf{K}_{damp}\mathbf{v}_{tool}. \quad (12.2)$$

where \mathbf{f}_g is gravity feedforward to reduce the weight of the levitated bowl, \mathbf{K} are the coupling gains, \mathbf{r} is the control rate of the device, and \mathbf{f}_{other} are the forces from the other objects in the simulation. When the virtual tool is not in contact with any other objects in the simulation, force feedback to the device is switched off. Realistic and stable performance for most task simulations has been obtained with the following coupling gains, where \mathbf{K}_{spring} and \mathbf{K}_{damp} are the gains from the simulation side and \mathbf{K}_p and \mathbf{K}_v are the maglev device control gains:

Gain	Position	Rotation
\mathbf{K}_{spring}	100 N/mm	800 Nm/rad
\mathbf{K}_{damp}	10 N/mm/s	20 Nm/rad/sec
\mathbf{K}_p	4.0 N/mm	25 Nm/rad
\mathbf{K}_v	0.1 N/mm/s	0.5 Nm/rad/sec

The time required to send a set of position setpoints from the simulation and receive a reply of setpoints from the control processor is generally approximately 1 millisecond. The simulated environment takes 10 milliseconds to respond to forces or motions exerted by the user, however, due to its 100 Hz update rate. The response delay due to this lag is perceived by the user as stickiness or sluggishness in haptic interaction.

Interpolation between the setpoints supplied by the simulation eliminates the jittering or chattering feel experienced by the user during sliding contacts between the simulated tool and other objects. The desired position setpoints \mathbf{x}_{goal} on the haptic device controller are interpolated from the last setpoint supplied by the simulation \mathbf{x}_{simnew} and the differences between the last setpoint and the previous one $\mathbf{x}_{simprev}$:

$$\mathbf{x}_{goal} = \mathbf{x}_{goal} + \frac{\mathbf{x}_{simnew} - \mathbf{x}_{simprev}}{\mathbf{T}_{avg}}, \quad (12.3)$$

where T_{avg} is the average of the last three simulation time intervals since there are occasional variations due to momentary overruns in the simulation calculations, network traffic, or other delays. This interpolation scheme is a first-order hold since the most recent sample and slope between samples are used to calculate the interpolated setpoints.

During the development and testing of the virtual coupling implementation in the haptic interaction system, it was found that the jitter due to the slower update rate of the physical environment simulation is eliminated when the simulation can update as fast as the control bandwidth. For simulations that can execute at 100 Hz on the host workstation, simulation setpoint interpolation was not used, since the position control bandwidths of all the axes of the magnetic levitation device are also approximately 100 Hz.

This same method of virtual coupling with interpolated setpoints can be used to add haptic interaction to any realtime simulation that calculates motions due to dynamic forces. Independent operation of the simulation and the haptic device controller significantly simplifies development, testing, and debugging of the integrated system.

12.2.1 Sample Tasks

General sample tasks have been programmed into the simulation to demonstrate rigid-body haptic tool manipulation. Each of these tasks requires 6-DOF haptic manipulation and feedback in the haptic interface device and could not be performed with a 3-DOF only point interaction device. The first general task world contains several free polyhedral objects and fixed walls, as pictured in Fig. 12.3. The user tool is a square scoop with a handle. The scoop can be used to feel, strike, push, pick up, or throw and catch other solid objects in the simulation while the user feels its dynamic response. The second task, shown in Fig. 12.4, demonstrates the simple classic peg-in-hole manipulation problem. The world contains only a fixed square hole and the user tool is a peg of a slightly smaller cross-section than the hole. The third task, in Fig. 12.5, is a variation of the peg-in-hole where the task setup includes a fixed keyhole, a movable bolt, and a key as the haptic user tool. The user can insert the key into the hole and rotate it to slide the bolt sideways.

12.3 Sample Experimental Results

Experimental data from sample task environments demonstrate the effectiveness of the integrated haptic interface system.

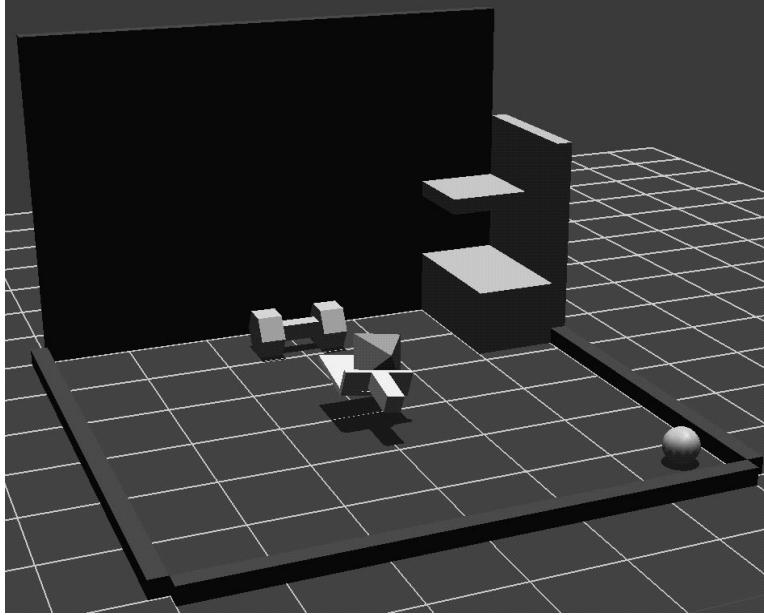


Figure 12.3: Block Manipulation Task

12.3.1 Tool Colliding and Sliding on Floor

The data plots in Fig. 12.6 were obtained from the magnetic levitation controller during typical user interaction with the haptic interface system. Rotation and torque data are not shown. The user brought the virtual scoop of Fig. 12.3 into contact with the environment floor while moving it in the positive x direction. The data show free tool motion until approximately 0.2 seconds, then vertical impact and sliding. The desired position setpoints in the controller are obtained from the virtual tool positions in the dynamic simulation.

In unconstrained axes, the desired setpoints lag behind the actual setpoints because there is no force feedback exerted by the system. Due to the slower update rate of the simulation, the desired position setpoint curves have a stair-step appearance. Since objects in the simulation cannot interpenetrate, the z setpoint is pinned to the floor level at -5 mm after impact, but the actual z position of the device handle is below this level due to the limited stiffness of the device controller. There is an inflection in the x position curve after the impact from the user's hand rebounding due to the friction in the simulation.

The force data plot shows the force commands exerted on the flotor by the Lorentz actuators. While the virtual tool is in free space, the user feels no resistance to motions other than the actual inertia and passive eddy current damping in the flotor. No Lorentz forces are generated in the xy horizontal plane and a 5 N force is generated upwards in z to partially cancel the weight of the flotor. When the virtual tool in the simulation contacts

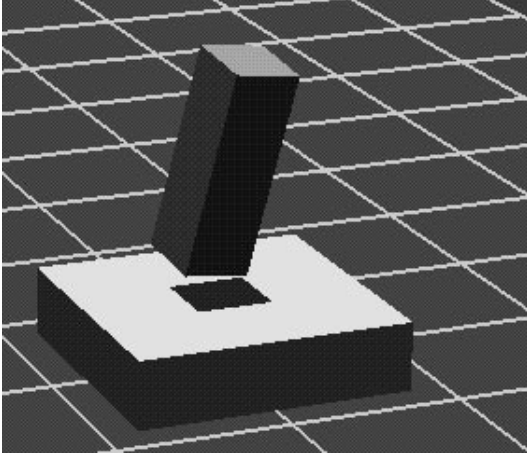


Figure 12.4: Peg-in-Hole

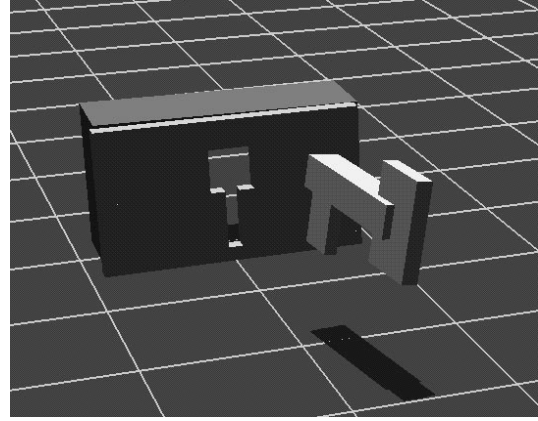


Figure 12.5: Key and Lock

Parameter	Value
Translation:	7.0 N/mm Stiffness, 0.06 Ns/mm Damping
Rotation:	25.0 Nm/rad Stiffness, 0.5 Nms/rad Damping
Force Feedforward:	5.0 N
Peg Size:	$2.0 \times 2.0 \times 6.0$ mm
Hole Top Level :	-8.0 mm
Hole Bottom Level:	-10.0 mm
Hole Cross-Section:	2.04×2.04 mm
Hole Center Location:	(4.0,0.0) mm

Table 12.1: Environment and Controller Parameters for Peg in Hole Insertion Trial

the floor, the force in y remains close to zero and a negative force in x is generated due to sliding friction modelled in the dynamic simulation. The force in z changes suddenly and rebounds at impact, then settles to support the weight of the flotor and additional disturbance forces from the user's hand.

12.3.2 Peg in Hole Insertion

A simulated square peg-in-hole insertion was performed using the virtual coupling haptic interaction system to obtain the position, rotation, force and torque data shown in Figs. 12.7 and 12.8. geometric parameters of the peg and hole task environment are given in Table 12.1.

In the haptic simulation, the peg was initially held in free space above the hole structure, then brought into contact with the flat surface next to the hole, slid over to the hole location, aligned with the hole orientation, and fully inserted until the bottom of the peg contacted

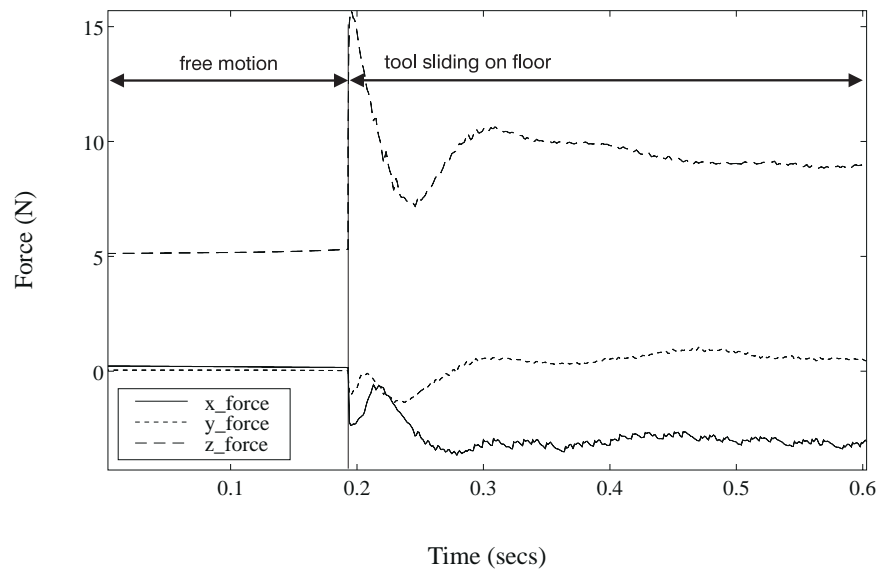
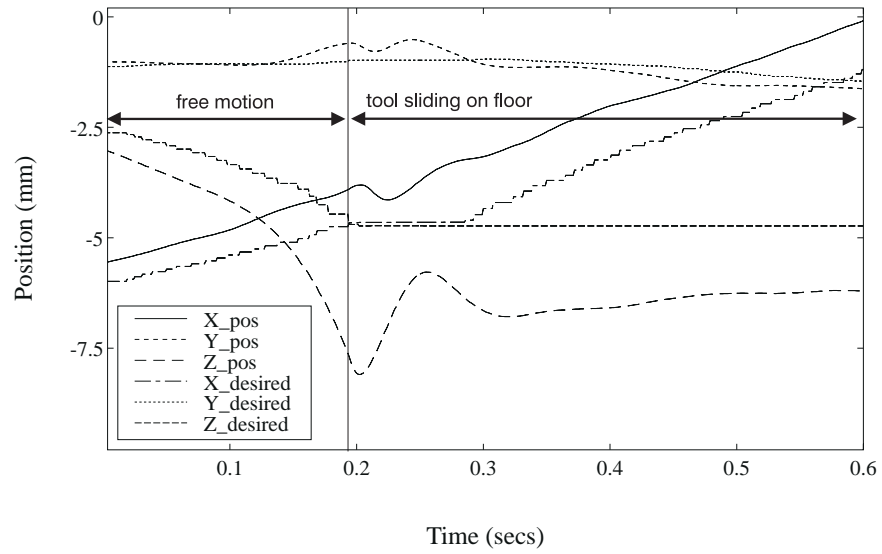


Figure 12.6: Position and Force Data from Impact and Sliding on Floor

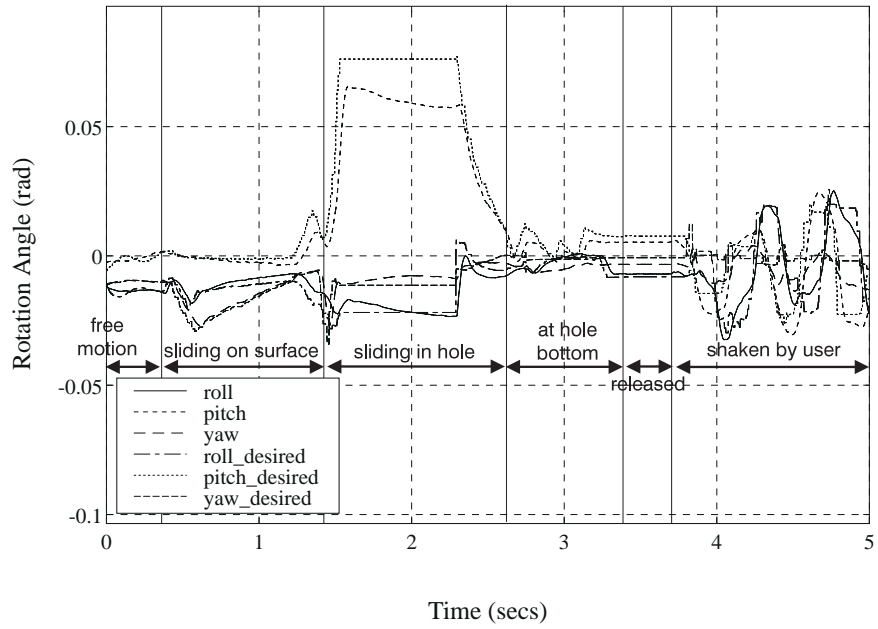
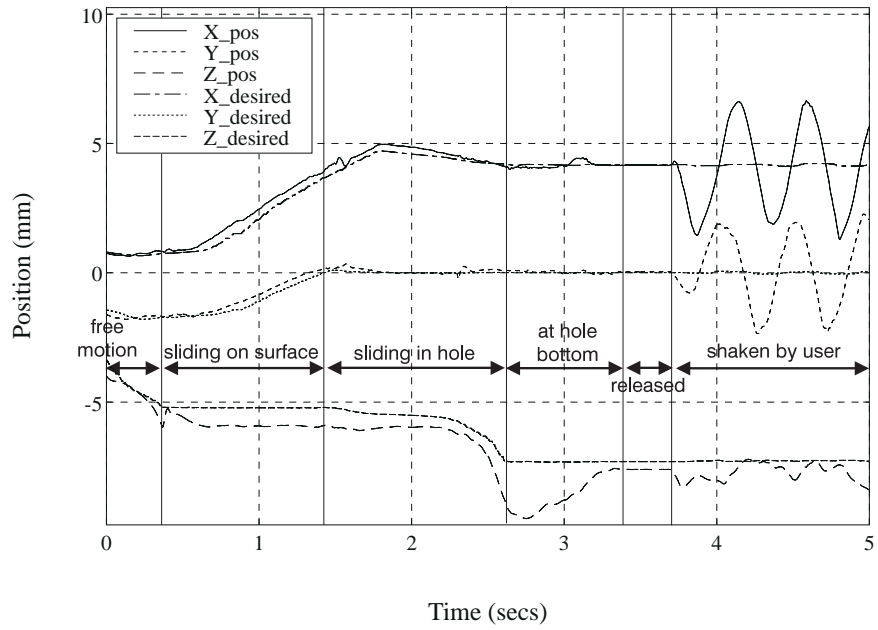


Figure 12.7: Position and Rotation Data from Peg in Hole Insertion

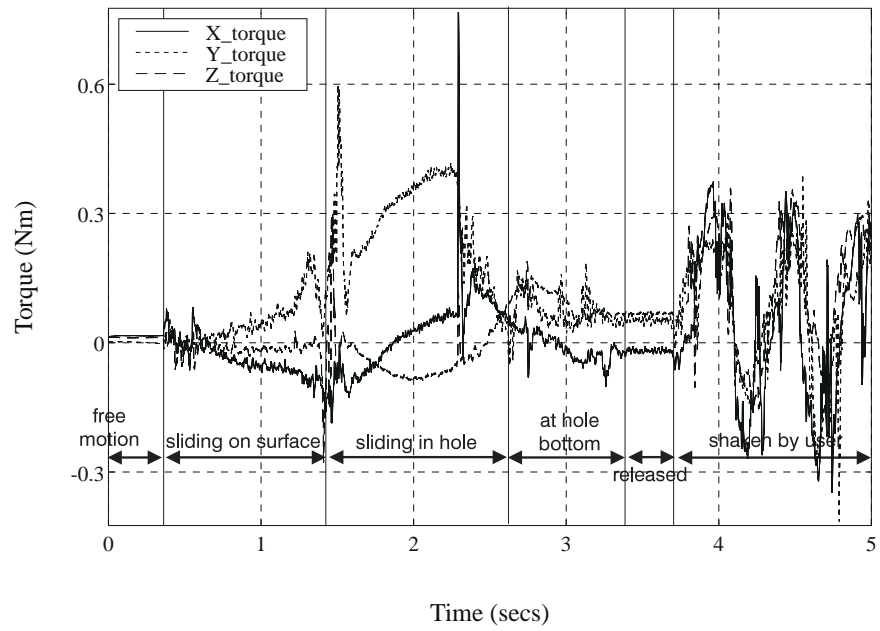
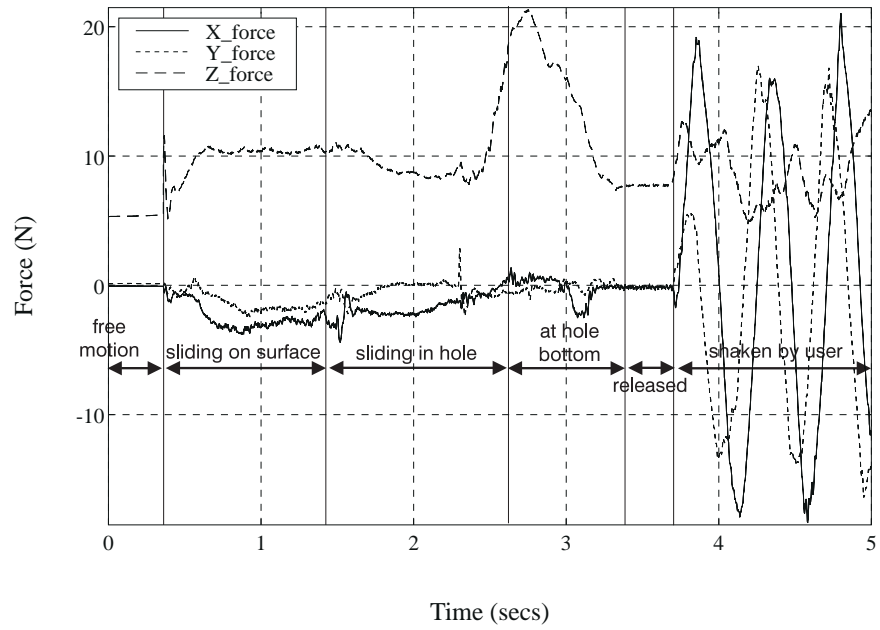


Figure 12.8: Force and Torque Data from Peg in Hole Insertion

the bottom of the hole. After insertion was complete, the handle of the haptic device was released, then regripped and pushed in all horizontal directions to demonstrate the translation and rotation constraints on the peg while inserted in the hole. Each of the stages of the peg-in-hole insertion can be seen in the sample data of Figs. 12.7 and 12.8.

For approximately the first 1/3 of a second, the peg is in free motion. There are no forces or torques generated by the haptic device except for a constant 5 N of vertical feedforward force to partially support the weight of the magnetic levitation device flotor, making it easier to be manipulated.

At the 0.3 second point, the peg contacted the surface next to the hole. The initial contact impulse can be seen in the *Z_force* line. The peg was then slid over the surface from approximately the $(x, y) = (1, -2)$ position to the $(x, y) = (4, 0)$ position, where the leading bottom edge of the collides with the corresponding edge of the hole, at slightly after 1.5 seconds, when a small collision impulse occurs in the *X_force* and a much larger impulse occurs in the *Y_torque*.

After the bottom of the peg collides with the edge of the hole, the top of the peg overrotates due to the momentum of the haptic device flotor and the delayed reaction of the user's hand. *X_pos* and *X_desired* overshoot the hole position and the *pitch_desired* angle increases to its maximum, which is a saturation point implemented in the virtual coupling in order to prevent the magnetic levitation haptic device from being torqued past its maximum rotation angle. By 2.3 seconds, the pitch angles are no longer saturated and the peg has been realigned. More force and torque impulses are generated as the peg aligns with the hole.

From 2.3 to 2.6 seconds, the peg slides down into the hole. *Z_desired* decreases from -5.5 to -7.0 mm. *Z_force* increases because of the friction and "stickiness" of the hole as the peg is sliding down and peaks after the peg reaches the bottom to counteract the force of the delayed reaction of the user's hand. *Z_pos* overshoots the level of the hole bottom as the user pushes the peg to interpenetrate the bottom of the hole.

From 3.3 to 3.7 seconds the user releases the haptic device handle. All the variables remain near zero except for *Z_force*, which now supports the entire weight of the magnetic levitation device flotor and handle. From 3.7 to 5 seconds, the user is pushing hard on the handle in different directions. *X_force* and *Y_force* oscillate through wide excursions, *X_desired* and *Y_desired* remain nearly constant due to the constraint of the hole on the peg, and *X_pos* and *Y_pos* oscillate due to the compliance of the virtual coupling.

Actual peg-in-hole experiments have been conducted by Whitney [63] using a peg supported by a remote center compliance. The peg-hole clearances, friction coefficients, and

the translational and rotational stiffnesses in the simulation and in Whitney's experiments are similar, differing by a factor of approximately two or less. The differences between the Whitney's experiments and the haptic simulation are that the simulated hole is shallow and not chamfered and the simulated hole and peg are square, while Whitney's peg and hole are round and much wider and deeper.

More significantly, the compliant support in Whitney's experiments is at a fixed point relative to the peg but in the simulation the peg is inserted by hand, so the orientation and the center of compliance of the user's hand grasping the peg may be anywhere and can change during the task according to the grasp and insertion strategy of the user. Furthermore, the insertion in the haptic simulation is not quasi-static since the response of the peg exhibits dynamic impacts and vibrations and is therefore dependent on simulated inertia and damping and not only geometry and stiffness.

Due to these differences in the insertion task, the experimental data from the haptic simulations and the actual peg-in-hole insertions cannot be directly compared. The torque and force levels of the simulation during the "sliding in hole" phase and the Whitney's experiment during the "one-point contact" are similar, however.

12.4 Summary and Conclusions

Tool-based haptic interaction using a magnetic levitation device can provide sensitive, high-bandwidth haptic interaction with a dynamic simulation. The virtual spring and damper coupling is a simple yet effective method for coupling a haptic interface device with a simulation to provide realistic, high-performance, 6-DOF haptic interaction. The three tasks set up for the simulation system demonstrate common haptic 6-DOF rigid body tasks that cannot be performed naturally using 3-DOF devices.

Separation of the simulation and the haptic interface simplifies development and testing and the overall system stiffness and stability can be tuned by adjusting the parameters of the virtual spring and dampers. Setpoint interpolation by the haptic device controller practically eliminates the feel of jitter during sliding contacts caused by the slower update rate of the simulation.

The flaws most noticeable by users in haptic interaction with dynamic physical simulations are brief periods when the simulation or the haptic device seem to freeze and not respond to the motions of the device handle. These periods occur whenever there is a delay in the simulation updates or the Ethernet communication between the haptic device controller and the simulation workstation. Delays can occur because the host workstation

operating system, the simulation software, and the communication interface are not strictly “hard” real time systems and complex events in the simulation, other processes on the workstation, or other traffic on the Ethernet link may cause the simulation software to miss its update deadlines. Elimination of these delays would require the simulation, its operating system, and all communications to be “hard” real time and guaranteed to meet all time deadlines without delays. This could not be done in our laboratory due to the limited computation speed of available realtime processors.

To further increase the fidelity of haptic interaction using the current hardware, the effects of communication latency and compliance between the simulation and the haptic device must be reduced. To accomplish this, all the simulated tool contact points can be sent to the device controller from the simulation instead of only its position. The feedback forces generated by the device would be calculated from each contact point and state rather than simply servoing to the interpolated setpoint. This multiple contact point intermediate representation integration is described in Chapter 13.

Chapter 13

Contact Point Intermediate Representation

An intermediate representation of a local region of a simulated environment can be used to generate feedback forces from a haptic device. The intermediate representation developed by Adachi [18] was extended by Mark *et al.* for more detailed models and situations [52], and Vedula has implemented an intermediate representation method to interface a PHANTOM haptic device with Baraff's CORIOLIS simulation for 2-D environments [53].

The second method implemented to combine the physical simulation and the magnetic levitation haptic device controller uses an intermediate representation based on multiple contact points and their normals. At each update of the simulation, a list of all the contact points between the haptic tool and other objects in the simulated environment are sent to the haptic device controller. The device controller then uses the contact point list and the device flotor position and orientation at each control cycle to calculate control forces based on the interpenetration at each contact point if the haptic tool position and orientation were coincident with the device handle position. The device also sends its current position and orientation back to the simulation.

The contact point intermediate representation can provide a better, more realistic performance response than the virtual coupling method because the impedance and contact state of the haptic device can change at every update of the controller according to the motion of the handle, rather than only when the position setpoints are updated by the simulation as in the virtual coupling case described in the previous chapter. Checking the interpenetration at each contact point at each update of the controller enables the controller to respond more quickly and appropriately to the user's hand motions since the contact state at each contact point can change at any controller update. The "stickiness" and sluggish feel of the virtual coupling method is eliminated and the user feels crisp, solid contacts as they are made and broken.

Direct feedthrough of contact forces generated in the CORIOLIS simulation would cause severe jitter leading to instability in the maglev haptic controller due to the drastic force discontinuities encountered in collisions. Also, the simulation cannot update fast enough for stable levitation and stiff contacts in the haptic interface device, even with simple environments and the fastest workstations available in our lab.

13.1 Configuration

The operation of the CORIOLIS simulations and the communication over an Ethernet TCP/IP socket is configured in the same way as for the virtual coupling method, except for the additional list of contact points and normals sent from the simulation on the workstation to the controller on the realtime processor. In the CORIOLIS simulations, objects are perfectly rigid and do not interpenetrate during normal operation, but contact points must interpenetrate in the intermediate representation on the haptic device controller side since the stiffness of contacts is limited by the maximum stiffness of the controller. The interpenetration at each contact point is calculated from the difference between the positions and orientations of the haptic tool in the simulation and the actual flotor handle of the magnetic levitation haptic device.

13.1.1 CORIOLIS Simulation

The operation of the CORIOLIS simulation does not require any modification since a list of contact points for the entire simulation is maintained and available to be read by other routines in the code according to the CORIOLIS API [57]. To model collisions and constraints of the dynamic system the simulation produces a list of all contact points and normal vectors in the body system at each iteration update.

To generate the intermediate representation for the haptic device controller and enable haptic interaction, each of the elements of the contact list is checked for contact with the virtual tool and a new list is compiled of only the contacts which involve the virtual tool. Each type of contact detected by the simulation, including face-to-vertex, vertex-to-face, edge-to-edge, sphere-to-point, sphere-to-edge, and sphere-to-face contacts is represented as a contact point and a unit surface normal vector.

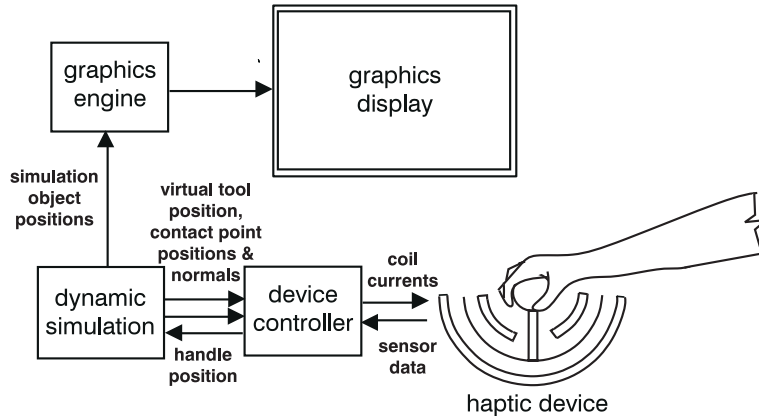


Figure 13.1: Schematic of Contact Point Intermediate Representation Interface Configuration

13.1.2 Communication

The simulation sends the list of contact points and normals along with the position and orientation vector of the virtual tool over the TCP/IP socket connection to the device controller. The magnetic levitation device controller sends back a vector of the position and orientation of the device handle, the same as in the virtual coupling case.

Since there may be many tool contact points when the tool is tightly constrained in a detailed environment simulation, much more data may need to be sent from the simulation to the controller. More time is required to be available for communication on both the controller and simulation sides of the socket so the update rates may need to be slower to avoid missing realtime update deadlines. The controller performs two socket reads; the first data set contains the position and orientation of the haptic tool and the number of contact points in the next data set so that the correct number of contact points and surface normals can be read quickly during the second socket read.

13.2 Controller

The haptic device controller must generate feedback at each control cycle from the intermediate representation list of contact points supplied from the simulation and from the motion of the haptic device handle. The objective of the controller is to provide a realistic response for haptic interaction as quickly and accurately as possible without causing instability.

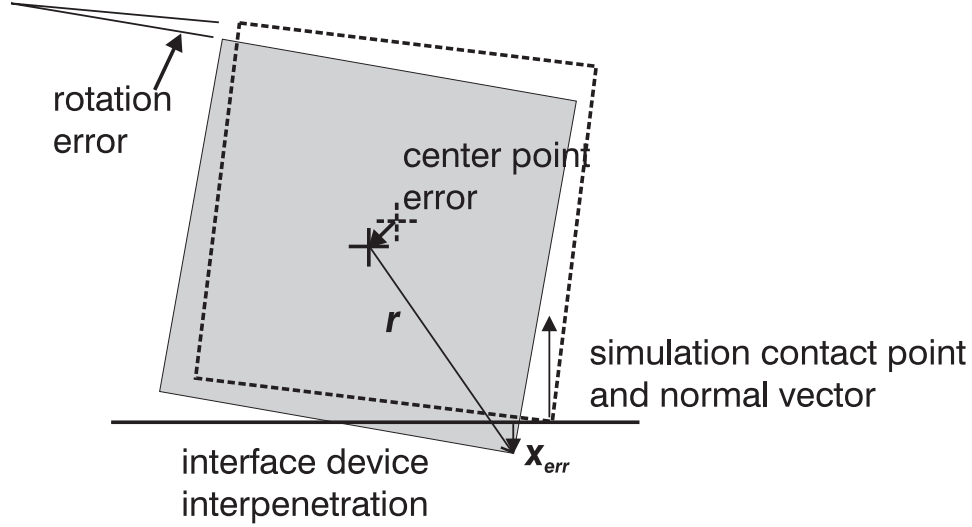


Figure 13.2: Feedback Force and Torque from Single Intermediate Representation Contact Point

13.2.1 Contact Point Intermediate Representation Control Law

The total force generated by the haptic device for its response is the sum of the forces from each contact point interpenetration. This control method is similar to the local hand-coded cube-in-cube and peg-in-hole simulations described in Chapter 10.

For each contact point in the intermediate representation:

$$\mathbf{x}_{mp} = \mathbf{x}_{mc} + \mathbf{R}_m(\mathbf{x}_{sp} - \mathbf{x}_{sc}), \text{ and} \quad (13.1)$$

$$\mathbf{x}_{err} = (\mathbf{x}_{mp} - \mathbf{x}_{sp}) \cdot \mathbf{N}_{sp}, \quad (13.2)$$

where \mathbf{x}_{mc} is the center position of the maglev device handle, \mathbf{R}_m is the 3×3 rotation matrix of the maglev handle, \mathbf{x}_{sp} is the contact point in the intermediate representation received from the simulation, \mathbf{x}_{sc} is the center point of the tool body in the intermediate representation. \mathbf{x}_{mp} represents the virtual position that the contact point on the tool in the simulation would have if the simulated tool had the position and the orientation of the maglev handle. \mathbf{x}_{err} is the interpenetration distance of the virtual contact point \mathbf{x}_{mp} , which is equivalent to the vector difference between the contact point in the simulation and the virtual contact point on the maglev handle projected onto the contact surface normal unit vector \mathbf{N}_{sp} from the intermediate representation.

An example of force and torque feedback from a single contact point is given in Fig. 13.2. The dotted line block represents the position and orientation of the tool in the simulation

while the shaded block would be the position and orientation of the tool superimposed onto the position and orientation of the haptic device handle.

The force and torque stiffness contribution from each contact point are:

$$\mathbf{f}_{mp} = -\mathbf{K}_p \mathbf{x}_{err} \mathbf{N}_{sp}, \text{ for } \mathbf{x}_{err} > 0, \quad (13.3)$$

$$\mathbf{f}_{mp} = 0, \text{ for } \mathbf{x}_{err} \leq 0, \text{ and} \quad (13.4)$$

$$\boldsymbol{\tau}_{mp} = (\mathbf{x}_{mp} - \mathbf{x}_{mc}) \times \mathbf{f}_{mp}, \quad (13.5)$$

where the stiffness feedback force \mathbf{f}_{mp} from a single point is the material stiffness proportional gain \mathbf{K}_p applied to the contact point interpenetration distance \mathbf{x}_{err} in the direction of the contact normal \mathbf{N}_{sp} . The feedback torque $\boldsymbol{\tau}_{mp}$ to be applied at the center \mathbf{x}_{mc} of the maglev handle from the contact point is the product of the contact point force vector \mathbf{f}_{mp} and the lever arm between the contact point and the body center, $\boldsymbol{\tau} = \mathbf{r} \times \mathbf{f}$.

If the interpenetration distance of the virtual contact point \mathbf{x}_{err} is less than zero, the maglev handle virtual contact point is no longer in contact with the other rigid body in the simulation and no stiffness feedback will be generated.

This will occur whenever the maglev device handle has been moved away from the contact surface since the last update of the simulation and intermediate representation. A major benefit of the contact point intermediate representation is that attractive forces between contacting bodies due to the delay in the simulation update and causing a “sticking” feeling during haptic interaction are never generated as with the simpler virtual coupling method of Chapter 11.

13.2.2 Modifications for Stability

With feedback from multiple contact points, it is more difficult to ensure that the control of the magnetic levitation haptic device will remain stable. In the virtual coupling case, the maximum stiffness along any axis is limited by the stiffness of the virtual coupling from the controller, but in the contact point intermediate representation the number of contact points is unlimited so the total contact stiffness along any axis may be higher than the limits of stability of the device controller. Clearly, some modification of the contact point intermediate representation control method is necessary to provide better stability in the controller.

During operation of the haptic interaction system using the described contact point intermediate representation it became apparent that the magnetic levitation haptic device control easily became unstable in rotation when the number of contact points was large and

the motion of the haptic tool was tightly constrained in many directions, such as during simulated peg-in-hole insertion.

A simple solution to the problem of instability in rotation is to use the virtual coupling method for control of rotation while using the contact point intermediate representation for control of translation. This hybrid of the contact point intermediate representation and virtual coupling for control is more easily stabilized over a larger range of environment stiffness and provides the crisp response of the contact point intermediate representation in translation. The sluggishness of the response in rotation alone due to the latency of the simulation was found to be not as perceptible to the user. The hybrid controller using a contact point intermediate representation for translation and virtual coupling for rotation was successfully implemented and found to be stable and provide realistic haptic interaction with the simulated environment.

13.2.3 Added Friction

Since the contact point intermediate representation control scheme eliminates the sluggishness and stickiness caused by the response lag of the simulation and only generates contact forces along the normal vector at each contact point, the environment contacted by the haptic tool feels frictionless to the user. Although the lack of friction enables the user to feel sharply defined details of the environment, the environment feels unnatural and it is difficult to manipulate objects which slide on the tool with no friction.

Some representation of friction must be added to the contact point intermediate representation controller to provide the user with a more realistic feel of the simulation environment. The simplest way to provide frictional resistance to motion during contacts is to add forces to servo towards the position setpoint received from the simulation while the tool is in contact with the environment. This method is similar to the stick-slip model of friction described in Section 9.2, but uses the tool position received from the simulation as the control setpoint rather than updating the setpoint when the static friction force threshold is exceeded. This approximation to sliding friction was implemented and does provide the desired friction forces during sliding contacts.

13.3 Experimental Results

The contact point intermediate representation and the modifications described in the previous section were tested with the same task simulation environments as the virtual coupling

setup in the previous chapter. The differences between the contact point intermediate representation and the virtual coupling that are most perceptible to the user are the lack of sluggishness and stickiness while the tool is in contact with other objects and a crisper response feel with the contact point intermediate representation.

13.3.1 Tool Colliding and Sliding on Floor

The plots of Figs. 13.3 and 13.4 were obtained by lowering the scoop tool onto the floor while moving it in the positive x direction. It is the same motion as was used in Section 12.3.1 for the virtual coupling case. The larger interpenetration of the device position is due to a higher vertical velocity imposed by the user before impact compared to the results of Section 12.3.1.

In Fig. 13.3, the “hybrid” control method was used where the contact point intermediate representation was used to generate the forces for position control and virtual coupling was used to generate torques to control rotation. No forces are generated in the x or y directions after contact with the floor since contacts are frictionless in this control method.

In Fig. 13.4, a simple approximation to Coulomb stick-slip friction has been added to the controller as described in Section 13.2.3. The friction forces generated in the x and y directions after contact is made can be seen in the plot.

13.3.2 Peg in Hole Insertion

Figs. 13.5 through 13.8 were generated from simulation of a peg-in-hole insertion task with the same tool and environment as described in Section 12.3.2 for virtual coupling control. The first two Figs. 13.5 and 13.6 were obtained using the hybrid controller with contact point intermediate representation for translation and virtual coupling for rotation and the Figs. 13.7 and 13.8 were obtained using the added friction approximation described in Section 13.2.3. The peg-in-hole task was much easier for the user in these experiments than for the experiments in Chapter 12 using only virtual coupling.

13.4 Discussion

The contact point intermediate representation method for integrating the control of the magnetic levitation haptic device with the state of the haptic tool in a physical simulation was made to be stable by using virtual coupling to control rotation.

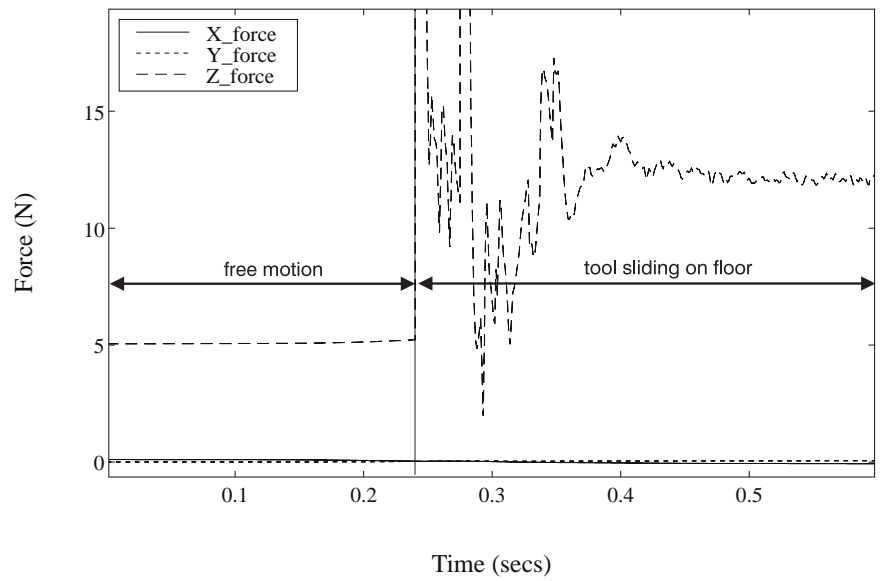
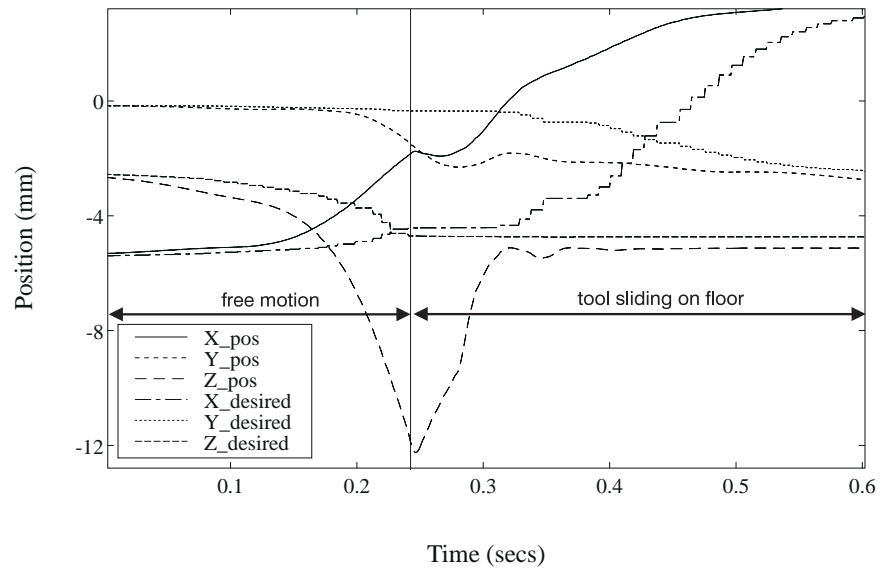


Figure 13.3: Position and Force Data from Impact and Sliding Motions with Contact Point Intermediate Representation

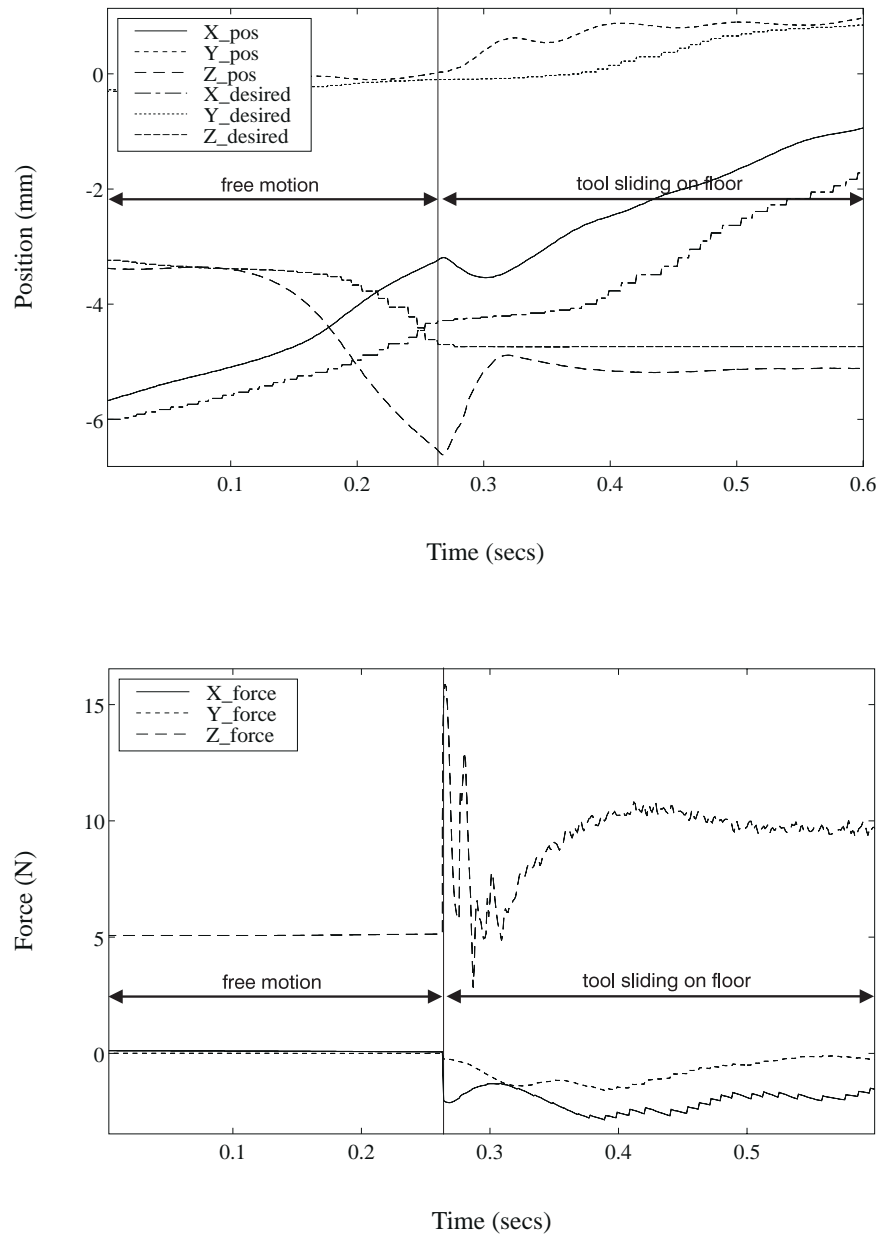


Figure 13.4: Position and Force Data from Impact and Sliding Motions with Contact Point Intermediate Representation and Local Friction

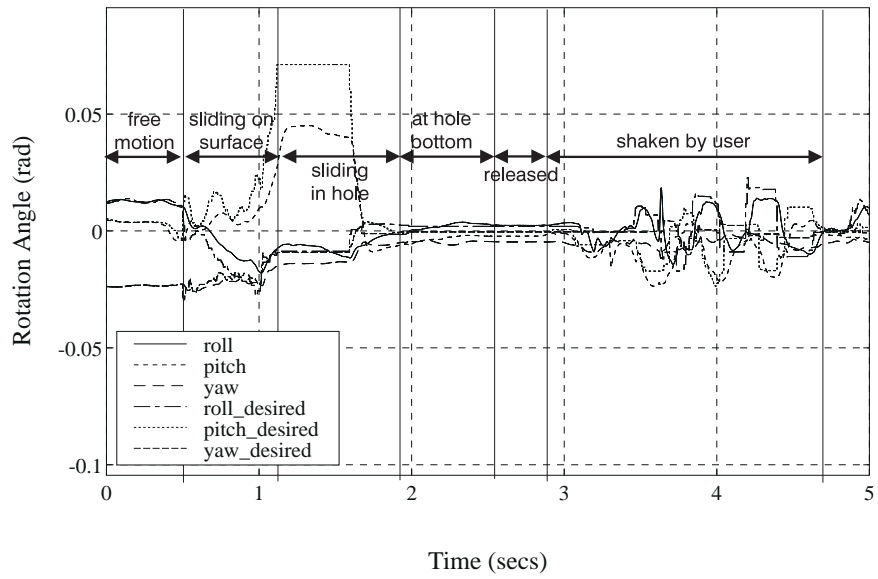
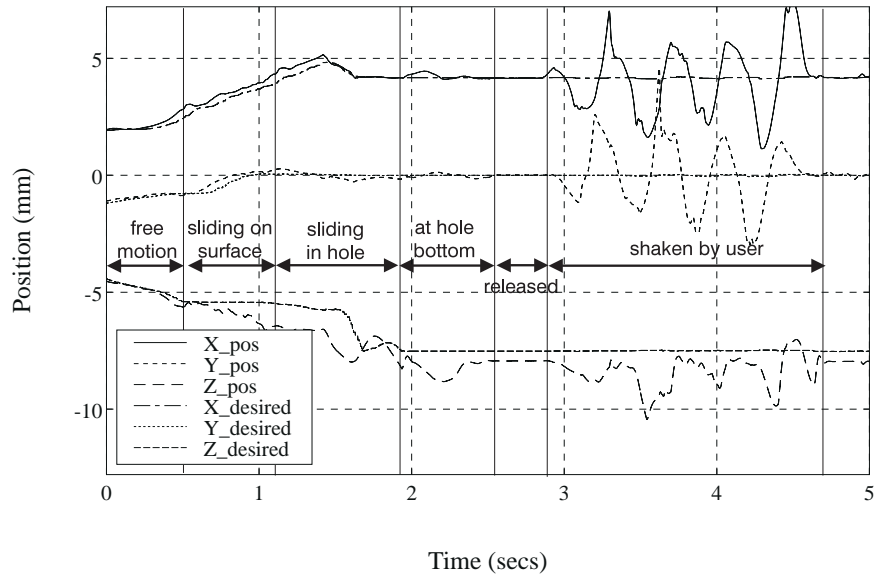


Figure 13.5: Position and Rotation Data from Peg in Hole Insertion with Contact Point Intermediate Representation

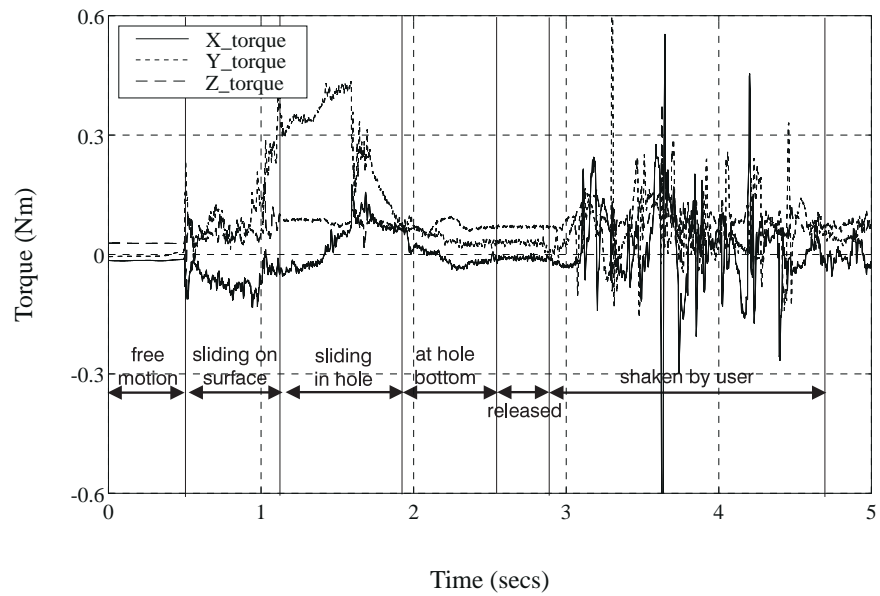
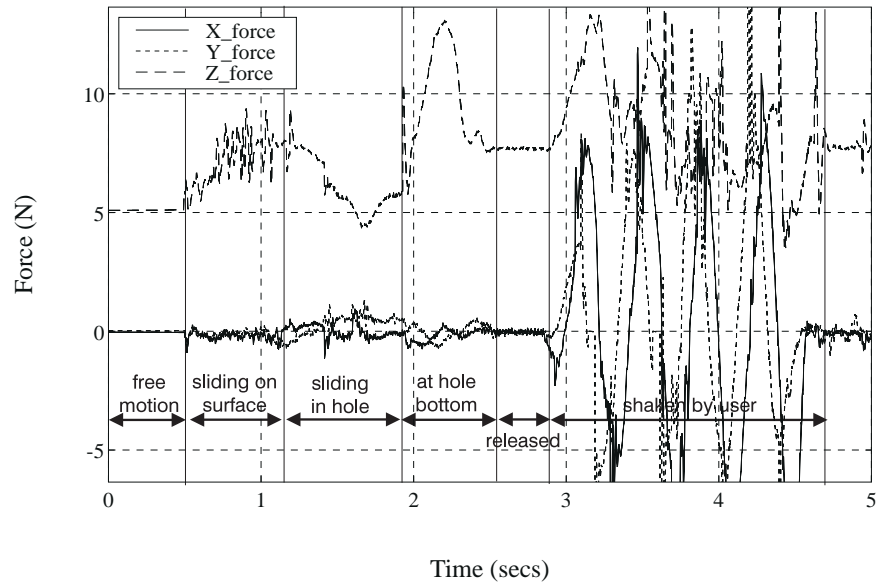


Figure 13.6: Force and Torque Data from Peg in Hole Insertion with Contact Point Intermediate Representation

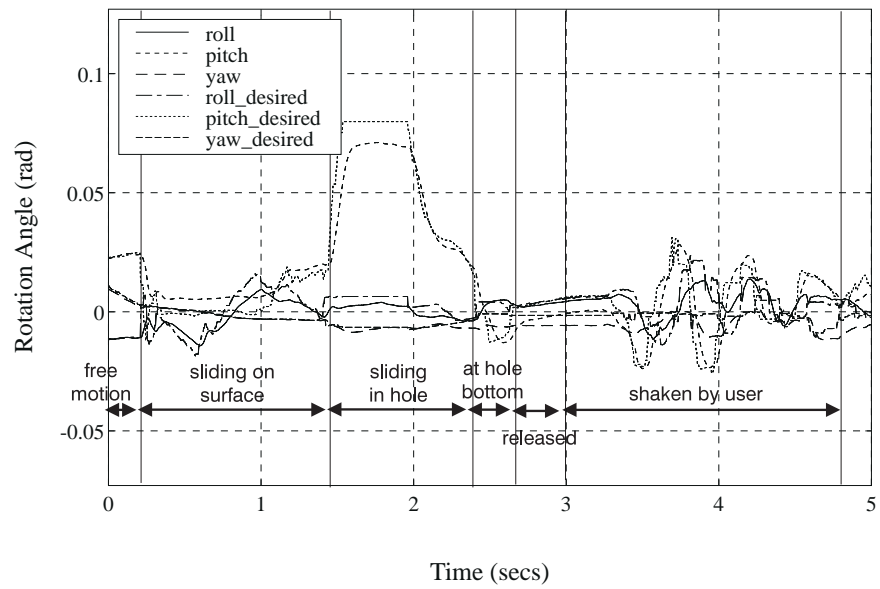
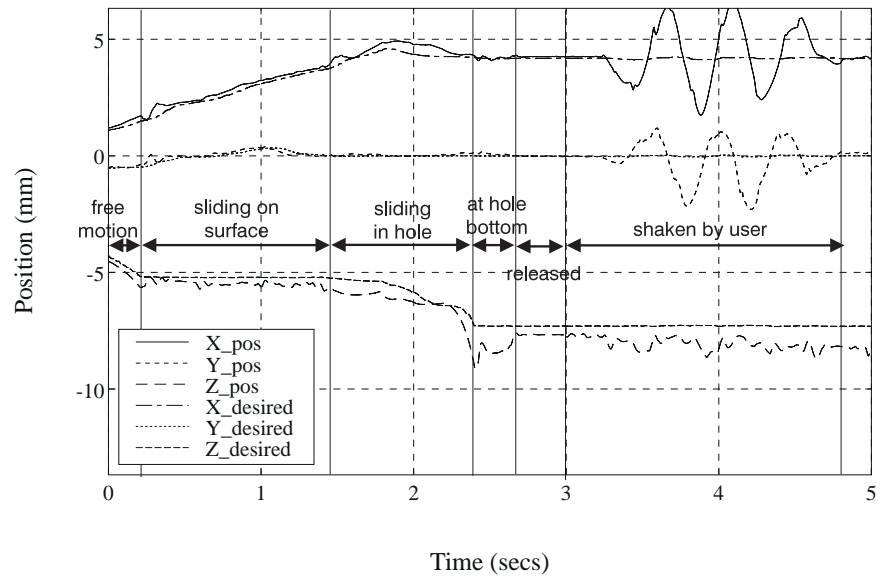


Figure 13.7: Position and Rotation Data from Peg in Hole Insertion with Contact Point Intermediate Representation and Local Friction

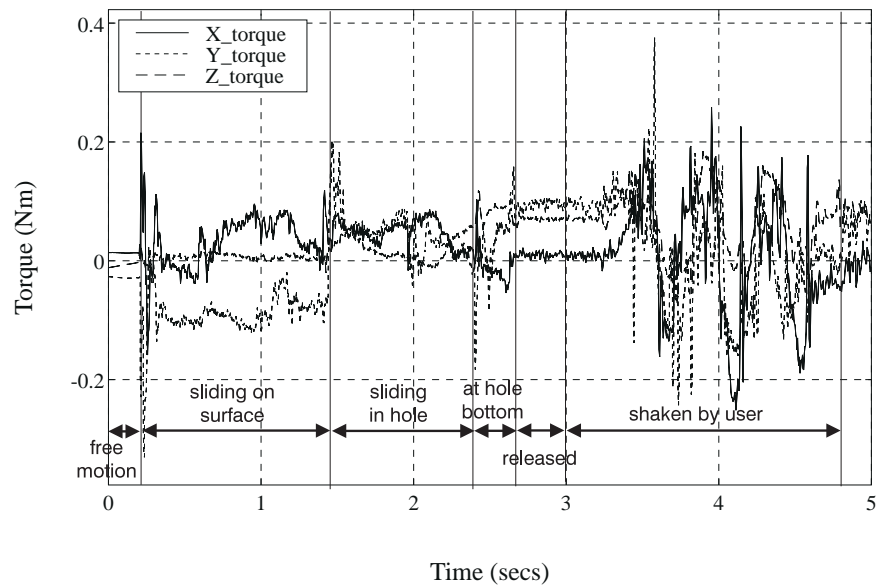
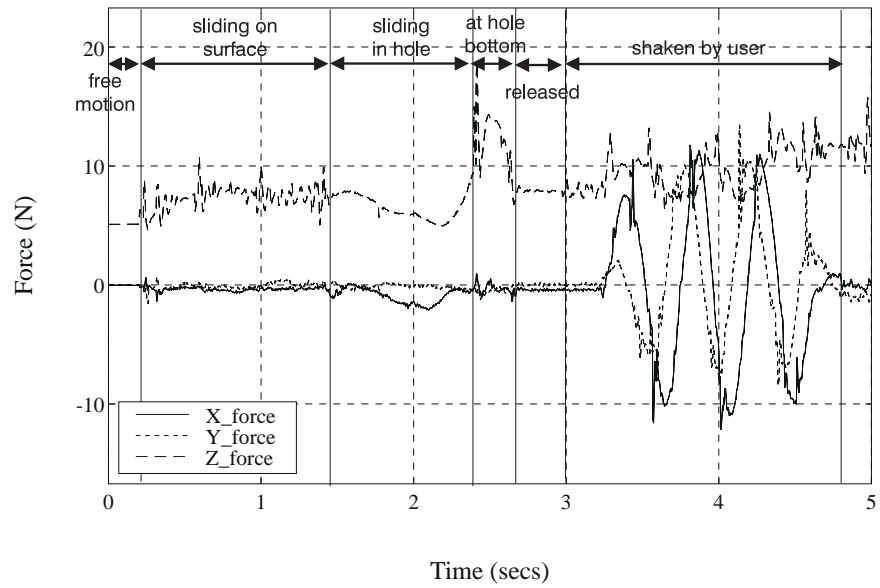


Figure 13.8: Force and Torque Data from Peg in Hole Insertion with Contact Point Intermediate Representation and Local Friction

13.4.1 Comparison to Virtual Coupling Interface Method

The operation of the physical simulation for the contact point intermediate representation is the same as for virtual coupling. In the case of the contact point intermediate representation, an additional routine compiles the list of haptic tool contact points and normals to be sent to the haptic device controller.

The contact point intermediate representation provides a more crisp and lively interaction and faster response to user motions compared to virtual coupling. There is no surface “stickiness” due to lags in communication and simulation updates. The peg-in-hole insertion task is much easier to perform for the user.

The device control based on the contact point intermediate representation is more complex than control based on virtual coupling. More communication bandwidth required to send the list of haptic tool contact points and normals from the simulation to the device controller. Stability of the controller is not ensured as it is in the virtual coupling case; some modification of the basic contact point intermediate representation control is necessary so that the controller will remain stable at the control rates of the system.

13.4.2 Further Potential Improvements

The contact point intermediate representation controller allows contact points between the tool and other objects in the simulation to be broken and reestablished at any cycle of the controller according to the motion of the device handle, but new contact points can only be introduced by the simulation at the slower simulation update rate. The implemented contact point intermediate representation cannot anticipate new contact points. As a result, new contact points tend to generate large, sudden impact forces since the device positions and orientations generally indicate some interpenetration already by the time the contact has been detected by the simulation. When the haptic device handle is being manipulated by the user, the haptic tool in the simulation can lag behind, since the stiffness coupling in the simulation must also be limited for stability of the simulation.

To prevent the sudden destabilizing impacts, the simulation would have to anticipate potential contact points and include their positions and normals in the intermediate representation so that the haptic device controller can respond to new contacts without delays. Anticipation of contact points would require modifications to the CORIOLIS simulation package, however. All potential contacts within a local neighborhood of the haptic tool would need to be detected and listed as well as the actual contacts in the simulation.

Better modeling of passivity conditions in the controller would be helpful to determine if it is possible to guarantee unconditional stability for the contact point intermediate representation while still providing realistic haptic interaction. A scheme of nonlinear, saturating damping and stiffness or combining the contributions of multiple contact points may improve stability without sacrificing performance response for realistic haptic interaction. The current solution using the hybrid controller with virtual coupling for rotation and the contact point intermediate representation for translation is *ad hoc* and not guaranteed to be stable; the stiffness and damping gains must be tuned to achieve stability for the motion constraints encountered in the simulated environment.

Chapter 14

Conclusion

The magnetic levitation haptic device of Part I and the simulation and integration software of Part II complement each other well since the 100 Hz update rate of the dynamic physical simulations is the approximate position control bandwidth of the haptic device. The simulation, interaction and control methods developed utilize the dynamic performance of the haptic device. The overall contribution of this work is the synthesis of a complete haptic interface system from the hardware design and all the control and interface software for it.

14.1 Summary of System Operation

The steps in the operation of the magnetic levitation haptic interface device controller during each sample and control cycle and of the host workstation during each update of the physical simulation are listed below.

14.1.1 Maglev Device Operation

In each controller cycle of the magnetic levitation haptic interface device, the controller waits for a timer interrupt and then the following steps are executed:

1. Sample sensor signals from planar position sensitive photodiodes.
2. Calculate sensor LED spot positions using calibrated lookup tables as described in Section 4.2.
3. Calculate flotor position from LED spot positions and previous position as described in Section 4.3.2.2.
4. Calculate forces for control, using the flotor position setpoints and PD gains as in Section 5.2.3. If the contact point intermediate representation of Chapter 13 is used

as an interface to the simulation, then the control forces are generated from the interpenetrations of all the contact points together.

5. Add any additional forces for rendering of surface friction and texture, as described in Chapter 9 and shown in Chapter 10.
6. Convert desired force and torque vector to vector of coil currents using \mathbf{A}^{-1} matrix from Section 3.1.4.
7. Output currents to amplifiers.

The local realtime device controller also executes network communications with the host workstation and a console shell task for the user to enter commands and configuration scripts to the controller. The console shell task and the servo control tasks were set to equal task priorities so that the controller was never noticeably delayed but shell commands still had a reasonable response time.

14.1.2 Interaction with Simulation

The host workstation executes the physical simulation as described in Section 9.1 and Appendix D and handles interaction with the magnetic levitation haptic device controller. A POSIX timer and signal handler is used to update the simulation at 100 Hz performing the following at each cycle:

1. Write haptic tool position to socket connection with maglev haptic device and haptic tool contact point locations and normals if contact point intermediate representation is in use.
2. Read device handle position from socket connection.
3. Advance simulation one timestep with added force on tool from virtual coupling with maglev haptic device handle position.
4. Compile list of tool contact locations and normals if contact point intermediate representation is in use.

The updating of the graphics display and execution of commands from the graphical user interface control panel are done by a loop outside of the timer signal handler. The execution rate of this loop depends on the complexity of the simulation and the number of contacts and collisions occurring in the simulation at each update.

14.2 Device Contributions

This new device is the first Lorentz magnetic levitation device that was specifically designed for haptic interaction. The measured control bandwidths for the device are the highest yet reported for any 6-DOF haptic interface device. Furthermore, the device dynamics are frictionless and the impedance range of the device in stiffness is greater than three orders of magnitude. The high performance of the device provides an opportunity for better evaluation of physical simulations, control algorithms, integration methods, and human perception for haptic interaction.

14.2.1 Design

The device was designed to be easily reproducible. The complete design schematics are available and the procedures followed at each stage of fabrication of the device are well documented. Since low cost fabrication methods and materials were used where possible, more devices of the same design can quickly and easily be fabricated for development of haptic interface applications such as CAD and medical simulations, and for future research in other areas such as teleoperation and psychophysical perception. The materials, fabrication procedures, and components in the design such as the large PSDs, high energy NdFeB magnets, and spherical wound coils are novel but not prohibitively expensive.

14.2.2 Evaluation

A thorough quantitative evaluation of the device performance parameters including the control bandwidths, accuracy, and ranges in both force and position has demonstrated the potential effectiveness of maglev haptic interaction in general and of the new hemispherical design in particular. The controlled bandwidths of the device were measured with a dynamic signal analyzer and a load meter. The relative positioning accuracy of the position sensing system was determined through the sensor calibration process and evaluation of the kinematics calculation algorithm and results. The power consumption of the device was calculated from resistance and voltage measurements.

14.2.3 Interactive Haptic Simulation Testbed

The magnetic levitation haptic interface device described here served as a testbed to investigate issues in high-performance haptic interaction with physical simulations. Since no other 6-DOF haptic interface device yet exists with the sensitivity and control bandwidths

of the new device, it enables evaluation of richer, more subtle methods of interaction and simulation. Newly developed simulation methods and their integration with the haptic device controller were described in Part II of this document.

A new research project in the Microdynamic Systems Laboratory involves using the magnetic levitation haptic interface device as a measurement tool for psychophysical perception with human test subjects. The motions, perceptions, and task execution times and effectiveness of different users will be tested while performing a peg-in-hole task with the maglev haptic device and a simulated environment, with the maglev haptic device for teleoperation, and with direct manipulation of a real peg into a real hole.

14.3 Simulation and Interface Contributions

Since haptic interface development is an emerging field of research, there is an opportunity to make a significant contribution by demonstrating a novel and fully operational system for 3-D haptic interaction. The main original software contributions of this work are in the integration of the technologies of realtime physical simulation and Lorentz magnetic levitation. The new magnetic levitation haptic device and the new methods and implementations of physical simulation and controller integration have together enabled convincingly realistic haptic interaction with sample task environments. Most of the integration methods and haptic rendering techniques developed are not specific to maglev actuation, but are applicable to any other type of high-performance haptic interface device.

14.3.1 System Integration

The methods used to integrate the simulation and the device control computations and to maintain correspondence between the two systems are also applicable to any other 6-DOF rigid-body device control interfaces. The intermediate representation developed to link the simulation at 100 Hz and the device controller at 1000 Hz is a new extension of the intermediate representations developed for point-based haptic interaction.

14.3.2 Interactive Demonstrations

Functioning demonstrations of different task simulation environments and integrated control implementations directly proves the feasibility of the methods used. Evaluation of the effectiveness and usability of the complete haptic interaction system described can serve as a benchmark for further improvement of the system and comparison with other systems.

14.3.3 Haptic User Interface Features

Haptic user interface features were implemented to enable the user to easily move the haptic tool in the simulation over arbitrarily large translations and rotations using the limited motion range of the magnetic levitation haptic device. The haptic user interface features include variable scaling and offsets, rate control mode, and viewpoint tracking modes.

14.3.4 Application Interface

The documentation of the complete system design enables more maglev haptic interfaces to be easily fabricated for CAD, medical, human factors applications. The interaction methods used are applicable to rigid-body tool teleoperation systems as well as simulations. The simulated tasks and environments set up for haptic interaction are generalizable to a wide range of specific tasks involving haptic sensing. Because the haptic interface device controller and the dynamic simulation run independently, it would be relatively easy and straightforward to implement force-reflecting teleoperation of another device or interface multiple haptic interface devices to the same physical simulation environment.

14.4 Future Directions for Research and Development

Since the field of haptic interface research is relatively new, there are many possible directions to improve the state of the art in haptic interaction. Haptic interaction using Lorentz magnetic levitation is well suited for development with task environments which involve detailed, sensitive fingertip motions.

14.4.1 Psychophysical Research

Since the magnetic levitation haptic device can reproduce the dynamic behavior of tools during execution of a manual task, it can be used to study human haptic perception and manipulation as described above in Section 14.2.3. This type of research is already underway in a project using the device to study user perception and task execution in simulation and teleoperation compared to direct object manipulation.

14.4.2 Applications

Specific application environments could be implemented for use with the magnetic levitation haptic device. Interfaces for CAD assembly, simulations of various medical procedures, or

other tasks which benefit from sensitive haptic interaction with small hand motions would be well suited to this haptic interface system.

14.4.3 Simulation Complexity and Realism

The complexity and update rates of the simulation environments and the device controller are limited by the computational speeds of the host workstation and the controller processor. Assuming that the current trends towards faster and cheaper computational processing continue, the speed of the simulation and controller processors will no longer be an issue in a few years and much more detailed simulation environments and more sophisticated haptic device control will be easily realized. These improvements to a haptic interface system will make it practical to simulate complex, realistic tasks rather than the sample demonstration tasks with limited numbers of vertices and objects such as the blocks world and peg-in-hole task environments.

Reference List

- [1] K. Shimoga, "A survey of perceptual feedback issues in dexterous telemanipulation: Part i. finger force feedback," in *Proceedings of the IEEE Virtual Reality Annual International Symposium*, (New York), pp. 263–270, January 1993.
- [2] R. Cholewiak and A. Collins, "Sensory and physiological bases of touch," in *The Psychology of Touch* (M. A. Heller and W. Schiff, eds.), Hillsdale, NJ: Lawrence Erlbaum Associates, 1991.
- [3] R. L. Hollis and S. E. Salcudean, "Lorentz levitation technology: a new approach to fine motion robotics, teleoperation, haptic interfaces, and vibration isolation," in *Proc. 6th Int'l Symposium on Robotics Research*, (Hidden Valley, PA), October 2-5 1993.
- [4] G. Burdea, *Force and Touch Feedback for Virtual Reality*. New York: Wiley and Sons, Inc., 1996.
- [5] N. Durlach and A. Mavos, eds., *Virtual Reality: Scientific and Technological Challenges*, ch. 4. National Academy Press, 1995.
- [6] E. R. Tufte, *Envisioning Information*. Cheshire, Conn.: Graphics Press, 1991.
- [7] N. Parker, S. Salcudean, and P. Lawrence, "Application of force feedback to heavy duty hydraulic machines," in *IEEE Int'l Conf. on Robotics and Automation*, (Atlanta), pp. 375–381, May 1993.
- [8] S. E. Salcudean and J. Yan, "Towards a force-reflecting motion-scaling system for microsurgery," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 2296–2301, May 1994.
- [9] R. Hollis, S. Salcudean, and D. Abraham, "Toward a tele-nanorobotic manipulation system with atomic scale force feedback and motion resolution," in *IEEE Micro Electro Mechanical Systems*, (Napa Valley, CA), February 1990.
- [10] F. Brooks, Jr., M. Ouh-Young, J. J. Batter, and P. Kilpatrick, "Project GROPE—haptic displays for scientific visualization," *Computer Graphics*, vol. 24(4), pp. 177–185, August, 1990.
- [11] D. A. Lawrence and J. D. Chapel, "Performance trade-offs for hand controller design," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 3211–3216, May 1994.

- [12] J. E. Colgate and J. M. Brown, "Factors affecting the Z-range of a haptic display," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 3205–3210, May 1994.
- [13] K. Waters and S. Wang, "A 3D interactive physically-based micro world," in *Proc. SPIE (Extracting meaning from complex data: processing, display, interaction)*, vol. 1259, (Santa Clara, CA), pp. 91–98, Feb. 14-16 1990.
- [14] Space Control GmbH/Logitech, Germany, *Magellan/Space Mouse User's Manual*, version 5.3 ed.
- [15] M. Bailey, D. Johnson, T. Massie, R. Taylor, and M. Yim, "Haptic explorations with the Phantom: Virtual touch through point interaction," in *So Real I Can Almost Touch It: The Use of Touch as an I/O Device for Graphics and Visualization*, New Orleans, Louisiana: Siggraph Course Notes 37, 1996.
- [16] J. Colgate, M. Stanley, and J. Brown, "Issues in the haptic display of tool use," in *Int'l Conf. on Intelligent Robots and Systems*, (Pittsburgh), August 1995.
- [17] C. Zilles and J. Salisbury, "A constraint-based god-object method for haptic display," in *Int'l Conf. on Intelligent Robots and Systems*, (Pittsburgh), August 1995.
- [18] Y. Adachi, T. Kumano, and K. Ogino, "Intermediate representation for stiff virtual objects," in *Proc. IEEE Virtual Reality Annual Intl. Symposium*, (Research Triangle Park, N. Carolina), pp. 203–210, March 1995.
- [19] M. Minsky, M. Ouh-young, O. Steele, F. P. Brooks, Jr., and M. Behensky, "Feeling and seeing: issues in force display," *Computer Graphics*, vol. 24(2), pp. 235–243, 1990.
- [20] K. Siira and D. Pai, "Haptic texturing: A stochastic approach," in *IEEE Int'l Conf. on Robotics and Automation*, (Minneapolis), pp. 557–562, April 1996.
- [21] R. L. Hollis, S. Salcudean, and A. P. Allan, "A six degree-of-freedom magnetically levitated variable compliance fine motion wrist: design, modeling, and control," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 320–332, June 1991.
- [22] S. Salcudean, N.M. Wong, and R.L. Hollis, "Design and control of a force-reflecting teleoperation system with magnetically levitated master and wrist," *IEEE Transactions on Robotics and Automation*, vol. 11, pp. 844–858, December 1995.
- [23] P. J. Berkelman, Z. J. Butler, and R. L. Hollis, "Design of a hemispherical magnetic levitation haptic interface device," in *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, (Atlanta), November 17-22 1996.
- [24] S. Salcudean and N. Wong, "Coarse-fine motion coordination and control of a teleoperation system with magnetically levitated master and wrist," in *Third International Symposium on Experimental Robotics*, (Kyoto, Japan), pp. 246–255, October 28-30 1993.

- [25] R. Lindemann and D. Tesar, "Construction and demonstration of a 9-string 6 dof force reflecting joystick for telerobotics," in *Proceedings of NASA International Conference on Space Telerobotics*, (NASA, Greenbelt, MD), pp. 55–63, Vol. 4 1989.
- [26] M. Sato, Y. Hirata, and H. Kawarada, "Space interface device for artificial reality—SPIDAR," *Syst. Comput. Jpn. (USA)*, vol. 23, no. 12, pp. 44–54, 1992.
- [27] P. Millman, M. Stanley, and E. Colgate, "Design of a high performance haptic interface to virtual environments," in *Proc. of IEEE Virtual Reality Annual International Symposium*, (New York), pp. 216–222, September 1993.
- [28] H. Iwata, "Pen-based haptic virtual environment," in *Proc. of IEEE Virtual Reality Annual International Symposium*, (New York), pp. 287–292, September 1993.
- [29] "Company brochure," Cybernet Systems Co., Ann Arbor, MI, 1995.
- [30] T. Massie and K. Salisbury, "The PHANTOM haptic interface: A device for probing virtual objects," in *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, (Chicago, Illinois), November 1994.
- [31] H. Iwata, "Artificial reality with force-feedback: development of desktop virtual space with compact master manipulator," *Computer Graphics*, vol. 24(4), pp. 165–170, 1990.
- [32] M. Bergamasco, B. Allota, L. Bosio, L. Ferretti, G. Parrini, G. M. Prisco, F. Salsedo, and G. Sartini, "An arm exoskeleton system for teleoperation and virtual environments applications," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 1449–1452, May 1994.
- [33] S. Jacobsen, F. Smith, D. Backman, and E. Iversen, "High performance, high dexterity, force reflective teleoperator II," in *ANS Topical Meeting on Robotics and Remote Systems*, (New York), February 1991.
- [34] B. Marcos, "Sensing, perception, and feedback for VR," in *VR Systems Fall 93 Conference*, (New York), SIG Advanced Applications, 1993.
- [35] M. L. Agronin, "The design of a nine-string six-degree-of-freedom force-feedback joystick for telemanipulation," in *Proc. NASA Workshop on Space Telerobotics*, pp. 341–348, 1987.
- [36] G. Burdea, J. Zhuang, E. Roskos, D. Silver, and N. Lagrana, "A portable dextrous master with force feedback," *Presence—Teleoperators and Virtual Environments*, vol. 1, pp. 18–27, March 1992.
- [37] H. Hashimoto, M. Buss, Y. Kunii, and F. Harashima, "Intelligent cooperative manipulation system using dynamic force simulator," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 2598–2603, May 1994.
- [38] V. Hayward, J. Choksi, G. Lanvin, and C. Ramstein, "Design and multi-objective optimization of a linkage for a haptic interface," in *ARK'94, 4th Int'l Workshop on Advances in Robot Kinematics*, (Ljubliana, Slovenia), June 1994.

- [39] V. Hayward, "Toward a seven axis haptic device," in *Int'l Conf. on Intelligent Robots and Systems*, (Pittsburgh), pp. 113–139, August 1995.
- [40] J. W. Hill, P. S. Green, J. F. Jensen, Y. Gorf, and A. S. Shah, "Telepresence surgery demonstration system," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 2302–2307, May 1994.
- [41] I. Hunter, T. Doukoglou, S. Lafontaine, P. Charette, L. Jones, M. Sagar, and P. Hunter, "A teleoperated microsurgical robot and associated virtual environment for eye surgery," *Presence*, vol. 2, no. 4, pp. 265–280, 1994.
- [42] S. Singh, M. Bostrom, D. Popa, and C. Wiley, "Design of an interactive lumbar puncture simulator with tactile feedback," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 1734–1739, May 1994.
- [43] S. Salcudean and N. Parker, "6-dof desk-top voice-coil joystick," in *International Mechanical Engineering Congress and Exposition*, (Dallas), November 1997.
- [44] P. J. Berkelman, R. L. Hollis, and S. E. Salcudean, "Interacting with virtual environments using a magnetic levitation haptic interface," in *Int'l Conf. on Intelligent Robots and Systems*, (Pittsburgh), August 1995.
- [45] S. Salcudean and T. Vlaar, "On the emulation of stiff walls and static friction with a magnetically levitated input-output device," in *International Mechanical Engineering Congress and Exposition*, (Chicago), pp. 303–309, November 1994.
- [46] T. Yoshikawa, "Manipulability of robotic mechanisms," *International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.
- [47] W. Light, *Non-Contact Optical Position Sensing Using Silicon Photodetectors*. United Detector Technology, April 1982.
- [48] D. T. Sandwell, "Biharmonic spline interpolation of geos-3 and seasat altimeter data," *Geophysical Research Letters*, vol. 2, pp. 139–142, 1987.
- [49] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. Cambridge University Press, 1988.
- [50] S. Yu, "Target location from three direction sensors." Microdynamic Systems Lab Report, August 1997.
- [51] J. J. Craig, *Introduction to Robotics: Mechanics and Control*. 2nd Edition: Addison-Wesley, 1989.
- [52] W. Mark, S. Randolph, M. Finch, J. V. Werth, and R. Taylor, "Adding force feedback to graphics systems," in *Computer Graphics (Proc. SIGGRAPH)*, pp. 447–452, 1996.
- [53] S. Vedula and D. Baraff, "Force feedback in interactive dynamic simulation," in *Proceedings of the First PHANToM User's Group Workshop*, (Dedham, MA), September 1996.

- [54] D. Baraff, "Interactive simulation of solid rigid bodies," *IEEE Computer Graphics and Applications*, vol. 15, pp. 63–75, 1995.
- [55] D. Baraff and A. Witkin, "Dynamic simulation of non-penetrating flexible bodies," in *Computer Graphics (Proc. SIGGRAPH)*, vol. 26, pp. 303–308, ACM, July 1992.
- [56] D. Baraff, "Issues in computing contact forces for non-penetrating rigid bodies," *Algorithmica*, vol. 10, pp. 292–352, 1993.
- [57] D. Baraff, *Coriolis v1.227 Documentation*, September 1997.
- [58] D. Baraff, "Coping with friction for non-penetrating rigid body simulation," in *Computer Graphics (Proc. SIGGRAPH)*, vol. 25, pp. 31–40, ACM, July 1991.
- [59] T. Zhao and M. Overmars, *Forms Library: A Graphical User Interface Toolkit for X*, v0.88 ed., November 1997.
- [60] E. D. Fasse and N. Hogan, "Quantitative measurement of haptic perception," in *IEEE Int'l Conf. on Robotics and Automation*, (San Diego), pp. 3199–3204, May 1994.
- [61] M. Srinivasan, G. Beauregard, and D. Brock, "The impact of visual information on the haptic perception of stiffness in virtual environments," in *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, (Atlanta), November 17-22 1996.
- [62] R. Adams and B. Hannaford, "A two-port framework for the design of unconditionally stable haptic interfaces," in accepted to *Int'l Conf. on Intelligent Robots and Systems*, (Victoria, B.C.), August 1998.
- [63] D. E. Whitney, "Quasi-static assembly of compliantly supported rigid parts," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 104, pp. 65–77, March 1982.
- [64] D. Baraff, "Fast contact force computation for nonpenetrating rigid bodies," in *Computer Graphics (Proc. SIGGRAPH)*, pp. 23–34, ACM, July 1994.
- [65] Y. Yokokohji, R. L. Hollis, and T. Kanade, "What you can see is what you can feel—development of a visual/haptic interface to virtual environment," in *Proc. of IEEE Virtual Reality Annual International Symposium*, (New York), pp. 46–53, April 1996.
- [66] A. Witkin *et al.*, "An introduction to physically based modeling," in *SIGGRAPH Course Notes*, 1994.
- [67] D. Baraff, "Curved surfaces and coherence for non-penetrating rigid body simulation," in *Computer Graphics (Proc. SIGGRAPH)*, vol. 24, pp. 19–28, ACM, August 1990.

Appendix A

Magnetic Levitation Device Jacobian

For the magnetic levitation flotor with cartesian position defined by $[X \ Y \ Z]$ and orientation defined by the angle-axis formulation with an angle θ of rotation about the unit vector axis defined by $[n_1 \ n_2 \ n_3]$, the reverse dynamics equations for the light spot locations on the three planar position sensing photodiodes is as shown below in equations A.1 through A.6.

$$s_{a,x} = \frac{l_z l_l [n_1 n_3 (1 - \cos \theta) - n_2 \sin \theta] + Z}{l_l [n_1^2 + (1 - n_1^2) \cos \theta] + X + l_z - l_t} \quad (\text{A.1})$$

$$s_{a,y} = \frac{l_z l_l [n_1 n_2 (1 - \cos \theta) + n_3 \sin \theta] + Y}{l_l [n_1^2 + (1 - n_1^2) \cos \theta] + X + l_z - l_t} \quad (\text{A.2})$$

$$s_{b,x} = \frac{l_z l_l [n_1 n_2 (1 - \cos \theta) - n_3 \sin \theta] + X}{l_l [n_2^2 + (1 - n_2^2) \cos \theta] + Y + l_z - l_t} \quad (\text{A.3})$$

$$s_{b,y} = \frac{l_z l_l [n_2 n_3 (1 - \cos \theta) + n_1 \sin \theta] + Z}{l_l [n_2^2 + (1 - n_2^2) \cos \theta] + Y + l_z - l_t} \quad (\text{A.4})$$

$$s_{c,x} = \frac{l_z l_l [n_2 n_3 (1 - \cos \theta) - n_1 \sin \theta] + Y}{l_l [n_3^2 + (1 - n_3^2) \cos \theta] + Z + l_z - l_t} \quad (\text{A.5})$$

$$s_{c,y} = \frac{l_z l_l [n_1 n_3 (1 - \cos \theta) + n_2 \sin \theta] + X}{l_l [n_3^2 + (1 - n_3^2) \cos \theta] + Z + l_z - l_t} \quad (\text{A.6})$$

where l_l is the distance from the flotor origin to the LED (115 mm) and l_t the distance from the stator origin to the sensor (160 mm).

The Jacobian matrix of a system of equations \mathbf{J} summarizes the differential relations between cartesian position and orientation \mathbf{x} and a set of sensor or actuator variables \mathbf{q} :

$$d\mathbf{x} = \mathbf{J} d\mathbf{q}. \quad (\text{A.7})$$

The elements of the Jacobian for the magnetic levitation haptic device and the sets of position and sensor variables are defined as:

$$J = \begin{bmatrix} \frac{\delta(n_1\theta)}{\delta s_{a,x}} & \frac{\delta(n_1\theta)}{\delta s_{a,y}} & \frac{\delta(n_1\theta)}{\delta s_{b,x}} & \frac{\delta(n_1\theta)}{\delta s_{b,y}} & \frac{\delta(n_1\theta)}{\delta s_{c,x}} & \frac{\delta(n_1\theta)}{\delta s_{c,y}} \\ \frac{\delta(n_2\theta)}{\delta s_{a,x}} & \frac{\delta(n_2\theta)}{\delta s_{a,y}} & \frac{\delta(n_2\theta)}{\delta s_{b,x}} & \frac{\delta(n_2\theta)}{\delta s_{b,y}} & \frac{\delta(n_2\theta)}{\delta s_{c,x}} & \frac{\delta(n_2\theta)}{\delta s_{c,y}} \\ \frac{\delta(n_3\theta)}{\delta s_{a,x}} & \frac{\delta(n_3\theta)}{\delta s_{a,y}} & \frac{\delta(n_3\theta)}{\delta s_{b,x}} & \frac{\delta(n_3\theta)}{\delta s_{b,y}} & \frac{\delta(n_3\theta)}{\delta s_{c,x}} & \frac{\delta(n_3\theta)}{\delta s_{c,y}} \\ \frac{\delta X}{\delta s_{a,x}} & \frac{\delta X}{\delta s_{a,y}} & \frac{\delta X}{\delta s_{b,x}} & \frac{\delta X}{\delta s_{b,y}} & \frac{\delta X}{\delta s_{c,x}} & \frac{\delta X}{\delta s_{c,y}} \\ \frac{\delta Y}{\delta s_{a,x}} & \frac{\delta Y}{\delta s_{a,y}} & \frac{\delta Y}{\delta s_{b,x}} & \frac{\delta Y}{\delta s_{b,y}} & \frac{\delta Y}{\delta s_{c,x}} & \frac{\delta Y}{\delta s_{c,y}} \\ \frac{\delta Z}{\delta s_{a,x}} & \frac{\delta Z}{\delta s_{a,y}} & \frac{\delta Z}{\delta s_{b,x}} & \frac{\delta Z}{\delta s_{b,y}} & \frac{\delta Z}{\delta s_{c,x}} & \frac{\delta Z}{\delta s_{c,y}} \end{bmatrix}. \quad (\text{A.8})$$

and can be calculated from Equations A.1 through A.6 to obtain:

$$\begin{aligned} J_{0,0} &= j_{am}l_1(a_{md}(n_3cbt - n_1^2n_3t_1 + n_1n_2t_2) - a_{xn}n_1(1 - n_1^2)t_4) \\ J_{0,1} &= j_{am}l_1(-a_{md}(n_1n_2n_3t_1 + n_2^2\cos(\theta) + sbt(1 - n_2^2)) + a_{xn}n_2j_{azn}) \\ J_{0,2} &= j_{am}l_1(a_{md}(n_1cbt - n_1n_3^2t_1 + n_2n_3t_2) + a_{xn}n_3j_{azn}) \\ J_{0,3} &= -a_{xn}j_{am} \\ J_{0,4} &= 0 \\ J_{0,5} &= j_{am}a_{md} \\ \\ J_{1,0} &= j_{am}l_1(a_{md}(n_2cbt - n_1^2n_2t_1 + n_1n_3t_3) - a_{yn}n_1(1 - n_1^2)t_4) \\ J_{1,1} &= j_{am}l_1(a_{md}(n_1cbt - n_2^2n_1t_1 + n_2n_3t_3) + a_{yn}n_2j_{azn}) \\ J_{1,2} &= j_{am}l_1(a_{md}(n_3^2\cos(\theta) + (1 - n_3^2)sbt - n_1n_2n_3t_1) + a_{yn}n_3j_{azn}) \\ J_{1,3} &= -a_{xn}j_{am} \\ J_{1,4} &= j_{am}a_{md} \\ J_{1,5} &= 0 \\ \\ J_{2,0} &= j_{bm}l_1(b_{md}(n_2cbt - n_1^2n_2t_1 + n_1n_3t_2) + b_{xn}n_1j_{bzn}) \\ J_{2,1} &= j_{bm}l_1(b_{md}(n_1cbt - n_2^2n_1t_1 + n_2n_3t_2) - b_{xn}n_2(1 - n_2^2)t_4) \\ J_{2,2} &= j_{bm}l_1(-b_{md}(n_1n_2n_3t_1 + n_3^2\cos(\theta) + (1 - n_3^2)sbt) + b_{xn}n_3j_{bzn}) \\ J_{2,3} &= j_{bm}b_{md} \\ J_{2,4} &= -j_{bm}b_{xn} \\ J_{2,5} &= 0 \end{aligned}$$

$$J_{3,0} = j_{bm}l_1(b_{md}(n_1^2 \cos(\theta) + (1 - n_1^2)sbt - n_1n_2n_3t_1) + b_{yn}n_1j_{bzn})$$

$$J_{3,1} = j_{bm}l_1(b_{md}(n_3cbt - n_2^2n_3t_1 + n_1n_2t_2) - b_{yn}n_2(1 - n_2^2)t_4)$$

$$J_{3,2} = j_{bm}l_1(b_{md}(n_2cbt - n_3^2n_2t_1 + n_1n_3t_2) + b_{yn}n_3j_{bzn})$$

$$J_{3,3} = 0$$

$$J_{3,4} = -j_{bm}b_{yn}$$

$$J_{3,5} = j_{bm}b_{md}$$

$$J_{4,0} = j_{cm}l_1(-c_{md}(n_1n_2n_3t_1 + n_1^2 \cos(\theta) + (1 - n_1^2)sbt) + c_{xn}n_1j_{czn})$$

$$J_{4,1} = j_{cm}l_1(c_{md}(n_3cbt - n_2^2n_3t_1 + n_1n_2t_3) + c_{xn}n_2j_{czn})$$

$$J_{4,2} = j_{cm}l_1(c_{md}(n_2cbt - n_3^2n_2t_1 + n_1n_3t_3) - c_{xn}n_3(1 - n_3^2)t_4)$$

$$J_{4,3} = 0$$

$$J_{4,4} = j_{cm}c_{md}$$

$$J_{4,5} = -j_{cm}c_{xn}$$

$$J_{5,0} = j_{cm}l_1(c_{md}(n_3cbt - n_1^2n_3t_1 - n_1n_2t_3) + c_{yn}n_1j_{czn})$$

$$J_{5,1} = j_{cm}l_1(c_{md}(n_2^2 \cos(\theta) + (1 - n_2^2)sbt - n_1n_2n_3t_1) + c_{yn}n_2j_{czn})$$

$$J_{5,2} = j_{cm}l_1(c_{md}(n_1cbt - n_3^2n_1t_1 - n_2n_3t_3) - c_{yn}n_3(1 - n_3^2)t_4)$$

$$J_{5,3} = j_{cm}c_{md}$$

$$J_{5,4} = 0$$

$$J_{5,5} = -j_{cm}c_{yn}$$

given the supplementary variables:

$$t_1 = \sin(\theta) + 2(1 - \cos(\theta))/\theta$$

$$t_2 = \sin(\theta)/\theta - \cos(\theta)$$

$$t_3 = (1 - \cos(\theta))/\theta - \sin(\theta)$$

$$t_4 = 2(1 - \cos(\theta))/\theta - \sin(\theta)$$

$$a_{md} = l_l(n_1^2 + (1 - n_1^2) \cos(\theta)) + X + l_e$$

$$a_{xn} = l_l(n_1n_3(1 - \cos(\theta)) - n_2 \sin(\theta) + Z$$

$$a_{yn} = l_l(n_1n_2(1 - \cos(\theta)) + n_3 \sin(\theta) + Y$$

$$b_{md} = l_l(n_2^2 + (1 - n_2^2) \cos(\theta)) + Y + l_e$$

$$\begin{aligned}
b_{xn} &= l_l(n_1 n_2(1 - \cos(\theta)) - n_3 \sin(\theta)) + X \\
b_{yn} &= l_l(n_2 n_3(1 - \cos(\theta)) + n_1 \sin(\theta)) + Z \\
c_{md} &= l_l(n_3^2 + (1 - n_3^2) \cos(\theta)) + Z + l_e \\
c_{xn} &= l_l(n_2 n_3(1 - \cos(\theta)) - n_1 \sin(\theta)) + Y \\
c_{yn} &= l_l(n_1 n_3(1 - \cos(\theta)) + n_2 \sin(\theta)) + X \\
j_{am} &= l_z/a_{md}^2 \\
j_{azn} &= (1 - n_1^2) \sin(\theta) + 2(1 - \cos(\theta)/\theta)n_1^2 \\
j_{bm} &= l_z/b_{md}^2 \\
j_{bzn} &= (1 - n_2^2) \sin(\theta) + 2(1 - \cos(\theta)/\theta)n_2^2 \\
j_{cm} &= l_z/c_{md}^2 \\
j_{czn} &= (1 - n_3^2) \sin(\theta) + 2(1 - \cos(\theta)/\theta)n_3^2
\end{aligned}$$

Appendix B

Kinematic Solution

The algorithm formulated by Stella Yu to calculate the haptic device position from the three position sensing photodiode signals [50] is summarized here. The vectors defining the sensor, LED marker, and device flotor positions are shown in Fig. B.

B.1 Initial Estimate

1. Calculate unit vectors $b_1(t + \Delta t), b_2(t + \Delta t), b_3(t + \Delta t)$ from the sensor signals.
2. For $i, j = 1, 2, 3$ $i \neq j$ and $k = 6 - i - j$ compute:

$$\begin{aligned}x_{ij} &= u_i(t) + g_k(t)u_j(t) + f_{ij}(t) \\f_{ij}(t + \Delta t) &= S_{ij}\dot{b}_i((t + \Delta t)) \\df_{ij}(t) &= f_{ij}(t + \Delta t) - f_{ij}(t) \\g_k(t + \Delta t) &= -b_i(t + \Delta t)\dot{b}_j(t + \Delta t) \\dg_k(t) &= g_{ij}(t + \Delta t) - g_{ij}(t) \\y_k(t) &= -u_i(t)u_j(t)dg_k(t) - u_i(t)df_{ij}(t) - u_j(t)df_{ij}(t)\end{aligned}$$

3. Form the matrix $J(u(t))$ and the vector $Y(u(t))$.
4. Solve $du(t)$ from the linear equation $J(u(t))\dot{du}(t) = Y(u(t))$.
5. Set $u(t + \Delta t) = u(t) + du(t)$.

B.2 Iterative Improvement

1. For $i, j = 1, 2, 3$ $i \neq j$ and $k = 6 - i - j$ compute:

$$x_{ij}(t + \Delta t) = u_i(t + \Delta t) + g_k(t + \Delta t)u_j(t + \Delta t) + f_{ij}(t + \Delta t).$$

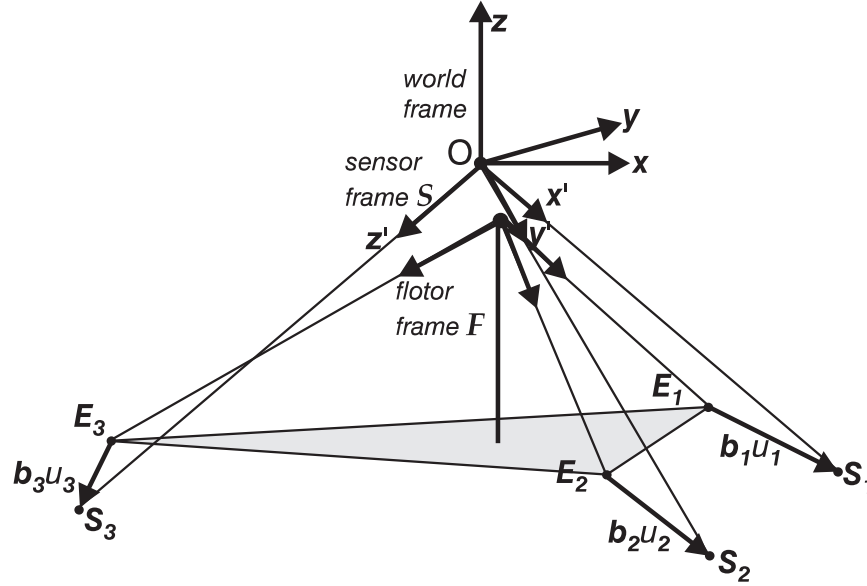


Figure B.1: Vector Geometry For Inverse Kinematics Solution

2. Form the matrix $J(u(t + \Delta t))$.
3. Compute $\delta\eta = \eta - \eta = \eta - F(u(t + \Delta t))$.
4. Solve $\delta u(t + \Delta t)$ from $J(u(t + \Delta t))\delta u(t + \Delta t) = \delta\eta$.
5. Set $u(t + \Delta t) = u(t + \Delta t) + \delta u(t + \Delta t)$.
6. Repeat iteration until $|\delta\eta| < \epsilon$.

To solve for a given 3-vector x in the linear equations $Ax = b$, the LU decomposition method given in *Numerical Recipes in C* [49] was used.

Appendix C

Design and Performance Parameters

C.1 Design

Physical parameters of the fabricated device and its components are listed below.

C.1.1 Motion Range

Translation: 25 mm
Rotation with centered flotor: 20°

These motion ranges are for the entire range of motion from one extreme to the other in any direction. The maximum excursion from the center in any direction is one half of the values given above.

C.1.2 Flotor

Hemisphere diameter: 230 mm
Coil shell thickness: 2 mm
Outer shell thickness: 0.6 mm

Two complete flotors for the magnetic levitation device were fabricated. The first flotor contains coils wound from copper ribbon wire and the second flotor has aluminum wire coils to reduce the mass of the levitated flotor.

	<u>Copper Coil Flotor</u>	<u>Aluminum Coil Flotor</u>
Flotor Mass:	880 g	580 g
Calculated Moment of Inertia:	0.00776 kg/m ²	0.00511 kg/m ²
Measured Moment of Inertia:	0.00842 kg/m ²	0.00543

The rotational inertia of the flotor bowl was calculated from the measured period of simple

harmonic motion oscillations with the flotor attached to a torsional spring. The calculated moment of inertia figures are based on uniform density in a thin hemisphere.

C.1.3 Coils

Wire Section:	1.8 x 0.1 mm
Turns:	300
Coil size:	75x120 mm

	<u>Copper</u>	<u>Aluminum</u>
Resistance:	6 Ω	10 Ω
Inductance:	0.3 mH	0.3 mH

Coil inductances were directly measured using a Hewlett-Packard 4192A LF Impedance Analyzer.

C.1.4 Sensors

Sensor Diameter:	35 mm
Sensor to Lens Distance:	13 mm
Lens to Hemisphere Center:	147 mm

The three position sensors are located on the outer stator hemisphere and point in mutually orthogonal directions. The position sensor and configuration geometry are shown in Figs. 4.1, 4.2, and 4.3.

C.2 Dynamic Performance

C.2.1 Force and Acceleration Limits

The limits for the current amplifiers which drive the flotor coils for actuation were set to 4.0 A. The force and acceleration limits of the device resulting from the amplifier current limits are shown below.

	<u>Force/Torque</u>	<u>Cu Coil Flotor Accel.</u>	<u>Al Coil Flotor Accel.</u>
Vertical Direction:	140 N	165 m/s ²	233 m/s ²
Horizontal Directions:	55 N	65 m/s ²	92 m/s ²
Vertical Rotation:	12.2 Nm	1449 rad/s ²	2247 rad/s ²
Horizontal Rotation:	6.3 Nm	748 rad/s ²	1160 rad/s ²

C.2.2 Impedance Range

The maximum stiffness given is the highest stiffness control gain that can be set for the magnetic levitation device at the 1500 Hz servo rate without causing instability or high frequency resonant vibrations of the flotor. The minimum stiffness is due to the approximate stiffness of the wiring to the flotor from the current amplifiers and LED drivers.

Maximum Stiffness:	25.0 N/mm
Minimum Stiffness:	0.005 N/mm
Impedance Ratio:	approximately 5000:1

C.2.3 Power Consumption

Copper Flotor:	4.5 W
Aluminum Flotor:	2.5 W
Aluminum Flotor with Additional 1 kg:	16.5 W

The power consumption during flotor levitation was calculated from the measured coil resistances and the voltage drops across each coil.

C.2.4 Control Bandwidths

The ± 3 dB closed loop position bandwidths given below were measured from the frequency response Bode plots of Figs. 7.7-7.10.

Control rate:	Copper Coil Flotor		Aluminum Coil Flotor	
	650 Hz	1300 Hz	650 Hz	1300 Hz
Vertical translation:	50 Hz	100 Hz	72 Hz	120 Hz
Horizontal translation:	60 Hz	60 Hz	70 Hz	100 Hz
Vertical axis rotation:	40 Hz	70 Hz	70 Hz	100 Hz
Horizontal axis rotation:	50 Hz	70 Hz	72 Hz	120 Hz

Appendix D

Physical Simulation Software

The CORIOLISTM software package, version 1.227 [57], from Baraff is used for the physical rigid-body simulations interfaced to the magnetic levitation haptic interface device. At every update of the physically-based simulation, the state of all the bodies in the system are advanced to the next time step according to the forces acting on them by numerical integration of Newtonian rigid-body dynamics. If there is any rigid-body interpenetration in the new system state, the time of the collision is determined and the system state is advanced to the time of the collision instead. Collision and constraint forces between objects in contact are then computed to advance the system state to the next time step. The simulation algorithms for calculating contact forces are described by Baraff [64] and the following summarization below is from Yokokohji [65].

D.1 Contact Constraint and Collision Forces

Suppose a rigid body is resting on another object with n contact points. For simplicity the frictionless case is considered. At the i -th contact point, a unit surface normal vector is defined such that the vector is directed outward from the surface. The i -th contact point acceleration, $\ddot{\mathbf{d}}_i$, which is the normal component of the translational acceleration of the object at the i -th contact point, can be expressed by the following equation:

$$\ddot{\mathbf{d}}_i = \mathbf{a}_{i1}\mathbf{f}_1 + \mathbf{a}_{i2}\mathbf{f}_2 + \cdots + \mathbf{a}_{in}\mathbf{f}_n + \mathbf{b}_i \quad (\text{D.1})$$

where \mathbf{f}_j denotes the magnitude of the j -th contact force, \mathbf{a}_{ij} is the coefficient representing the contribution of the j -th contact force to the i -th contact acceleration. \mathbf{b}_i is the term containing Coriolis and centrifugal forces and the external force.

For nonpenetrating rigid body motion, the following conditions must be satisfied:

$$\ddot{\mathbf{d}}_i \geq 0, \mathbf{f}_i \geq 0 \text{ and } \mathbf{f}_i \cdot \ddot{\mathbf{d}}_i = 0 \quad (\text{D.2})$$

Combining Eqs.(D.1) and (D.2) for all n contact points produces the system of equations:

$$\mathbf{A}\mathbf{f} + \mathbf{b} \geq 0, \quad (\text{D.3})$$

$$\mathbf{f} \geq 0 \text{ and } \mathbf{f}^T(\mathbf{A}\mathbf{f} + \mathbf{b}) = 0 \quad (\text{D.4})$$

The problem is to find all \mathbf{f}_i contact forces which satisfy Eqs.(D.3) and (D.4). This problem can be regarded as an optimization problem such as linear complementarity programming or quadratic programming. But solving such an optimization problem requires much computational effort and might not be adequate for the purpose of interactive simulation. Baraff [64] has proposed a fast algorithm to compute the contact forces by pivoting matrix \mathbf{A} . In the frictionless case, the algorithm is guaranteed to converge to the correct solution. The algorithm also works well in practice with surface friction.

D.2 Collision Impulses

Suppose that a rigid body object is colliding with another rigid object with m colliding points. Let \mathbf{v}_i^+ and \mathbf{v}_i^- denote normal components of the velocities after and before the collision at the i -th colliding point, respectively. Here \mathbf{v}_i^+ can be expressed by the following equation:

$$\mathbf{v}_i^+ = \mathbf{v}_i^- + \mathbf{a}_{i1}\mathbf{j}_1 + \mathbf{a}_{i2}\mathbf{j}_2 + \cdots + \mathbf{a}_{im}\mathbf{j}_m \quad (\text{D.5})$$

where \mathbf{j}_i denotes the impulse at the i -th colliding point, \mathbf{a}_{ij} is the coefficient representing the contribution of the j -th impulse to the i -th post-collision velocity. Newton's law of restitution says

$$\mathbf{v}_i^+ + \epsilon_i \mathbf{v}_i^- \geq 0 \quad (\text{D.6})$$

where ϵ_i denotes the coefficient of restitution at the i -th colliding point. The reason why we use “ \geq ” in Eq.(D.6) instead of “ $=$ ” is that there might be no impulse at the i -th colliding point but the object may be pushed away by the impulses at other colliding points.

For nonpenetrating rigid body collisions, the following conditions should be satisfied:

$$\mathbf{j}_i \geq 0 \text{ and } \mathbf{j}_i \cdot (\mathbf{v}_i^+ + \epsilon_i \mathbf{v}_i^-) = 0 \quad (\text{D.7})$$

Substituting eq.(D.5) into eq.(D.6), we get

$$\mathbf{a}_{i1}\mathbf{j}_1 + \mathbf{a}_{i2}\mathbf{j}_2 + \cdots + \mathbf{a}_{im}\mathbf{j}_m + \mathbf{v}_i^- + \epsilon_i \mathbf{v}_i^- \geq 0 \quad (\text{D.8})$$

Combining Eqs.(D.8) and (D.7) for all m colliding points, we get

$$\mathbf{A}\mathbf{j} + \mathbf{c} \geq 0 \tag{D.9}$$

$$\mathbf{j} \geq 0 \text{ and } \mathbf{j}^T(\mathbf{A}\mathbf{j} + \mathbf{c}) = 0 \tag{D.10}$$

The problem is to find the \mathbf{j}_i which satisfy Eqs.(D.9) and (D.10). Note that Eqs.(D.9) and (D.10) have the same forms as Eqs.(D.3) and (D.4). Therefore, the same algorithm can be used to find these impulse forces. Once we have obtained \mathbf{j}_i s, the system calculates the object velocities after the collision, resets the state variables, and restarts to solve the ODEs. More detail of the methods are given in [64, 66]

D.3 Collision Detection

The methods developed by Baraff for fast collision detection during physical simulation of rigid bodies [54, 67] include hierarchical bounding boxes and caching of results to exploit geometric coherence between simulation updates. Tables of bounding box overlaps along each dimension are updated at each simulation stem. The computations for updating the bounding box overlap tables and the actual interpenetration checking between polyhedra can be done essentially in linear time.