TOWARDS THE DEVELOPMENT OF A MIXED REALITY HAPTIC TEMPORAL BONE SURGICAL SIMULATION

by

Vivek Rampersad

A Thesis submitted to the Faculty of Graduate Studies of

the University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Mechanical and Manufacturing Engineering

University of Manitoba

Winnipeg, Manitoba, Canada

Abstract

The temporal bone is an anatomically complex region within the skull. Current training for temporal bone surgery includes cadaveric, physical and virtual haptic simulations and apprenticeships. Cadavers are limited by low supply. Haptic devices are limited by their force and stiffness ratings and thus cannot adequately simulate rigid materials. Physical simulations excel at simulating stiff materials but do a poor job of soft tissue. The research objective was to develop a mixed reality (MR) temporal bone surgical haptic simulation. This novel concept would utilize physical models to simulate bone and haptic forces to simulate soft tissue.

A surgical drill was attached to a Quanser® High Definition Haptic Device™ (HD^2) via a clamp. An algorithm was implemented to simulate a force at the drill tip and to negate the weight of the clamp. This modified haptic system was interfaced to a temporal bone haptic simulation. Haptic chatter unique to the modified haptic system was observed and low-pass filters were used to mitigate this issue.

Due to the poor positional accuracy of the HD^2, MR simulation was not achieved. However, VR haptic simulation was achieved. Six expert surgeons were recruited to investigate the following questions: "What is the impact of different haptic hardware on surgical realism?" and "Would end users prefer a surgical drill over a standard haptic manipulandum?" Three cases were compared: a Phantom Omni®, a standard HD^2 and a modified HD^2 with attached drill.

Expert surgeons rated the standard HD^2 and Phantom Omni equivalently whilst preferring the modified HD^2 with attached drill. Though the modified HD^2 scored higher in all categories only “Acoustics” and “Overall Appreciation” displayed statistical significance. This implies that drill acoustics is critical for realism.

Acknowledgments

I would like to thank Dr Bertram Unger for his technical knowledge, thesis corrections and for his time and guidance. It was an honor to work with someone who is an expert in both robotic haptics and medical research.

I would like to thank Dr Jordan Hochman for his useful and critical feedback on the evaluation of the haptic system. I would also like to thank Dr Hochman for his assistance lent in the design of the Evaluative Haptic Survey. I would also like to thank him in allowing me to witness temporal bone surgery.

I would like to thank Dr Jay Kraut for his advice and guidance on all matters related to C++. His work on the temporal bone virtual simulation is extraordinary and he has my deepest admiration and respect.

I would like to thank the examining committee for their valuable and insightful feedback.

I would like to thank Quanser for granting permission to reproduce their figures. All figures owned by Quanser are referenced to their source material. All other images are original creations or are the property of the Laboratory for Surgical Modeling Simulation and Robotics.

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 The Temporal Bone

The temporal bone is one of the most anatomically complex regions in the human body. It is a cranial bone which houses and protects the temporal lobes of the cerebrum as well as the organs of hearing. The temporal bone contains the region of the skull that is colloquially known as the temple [1-3]. Figure 1-1 shows the location of the temporal bone.

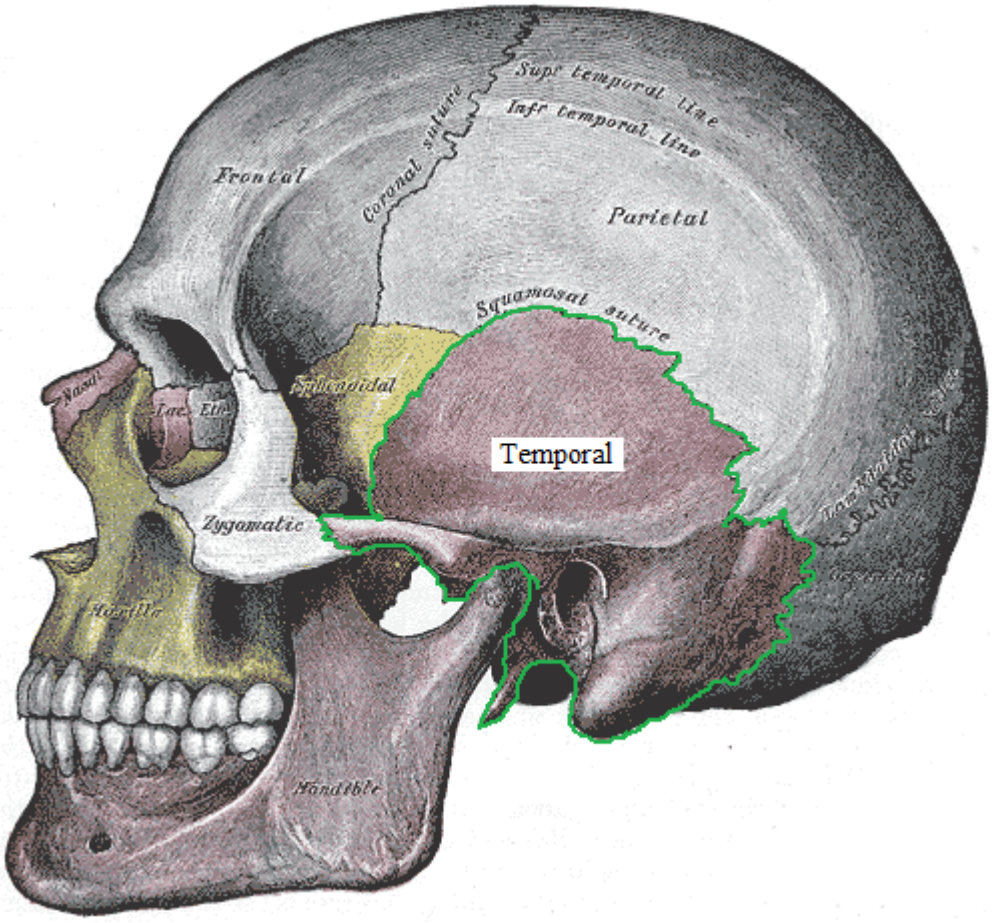

Figure 1-1: The temporal bone located within the human skull [4]

Figure 1-2 shows the detailed outer surface of the temporal bone, whilst Figure 1-3 shows the inner features of the temporal bone.

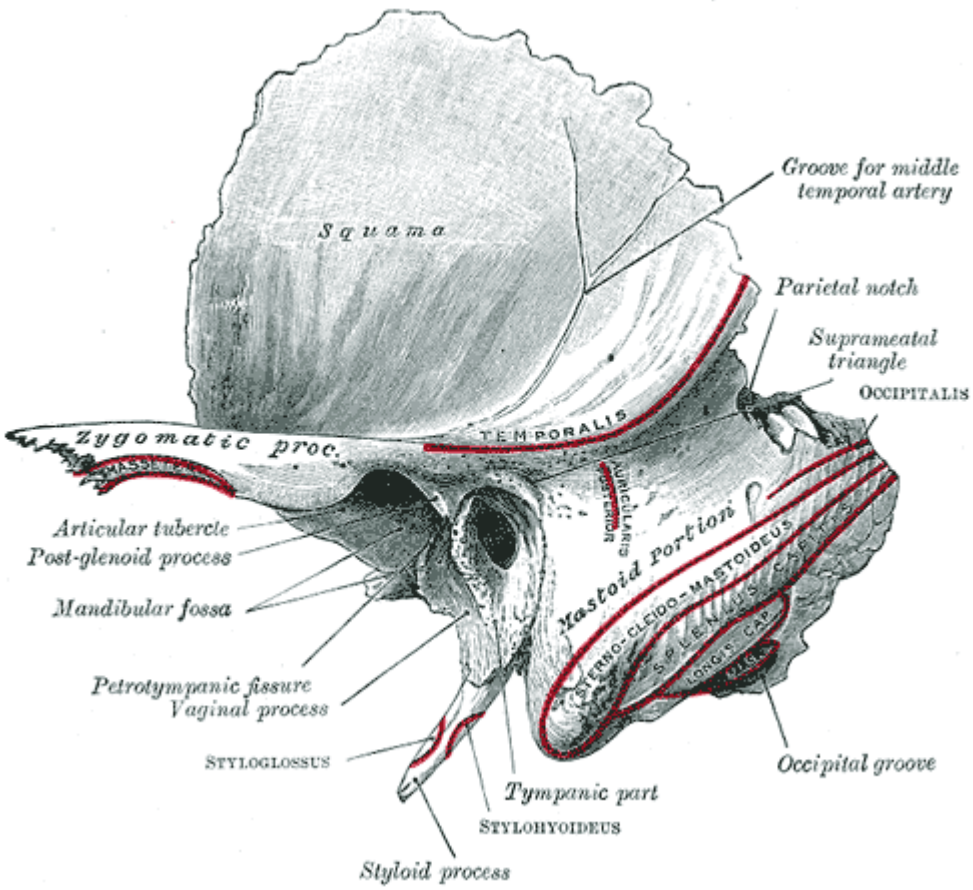

Figure 1-2: Detailed outer surface of left temporal bone [5]

Figure 1-2 shows that the temporal bone consists of five parts: the squama, petrous, mastoid, tympanic parts and the styloid process [5]. The bone itself is comprised of two kinds of tissue; one that is dense in texture and is referred to as cortical bone or compact bone, and the other that consists of slender fibers and lamellae and is referred to as spongy bone, trabecular bone or cancellous tissue. The compact tissue is always on exterior of the bone and encapsulates trabecular tissue which makes up the interior of the bone [6].

A very important and unique feature of the temporal bone is the presence of air cells. These are located in the mastoid part of the bone and are sometimes referred to as mastoid cells. These air

pockets vary in number, size and shape and their purpose is to help drain the middle ear. This internal feature is shown in Figure 1-3 [5].

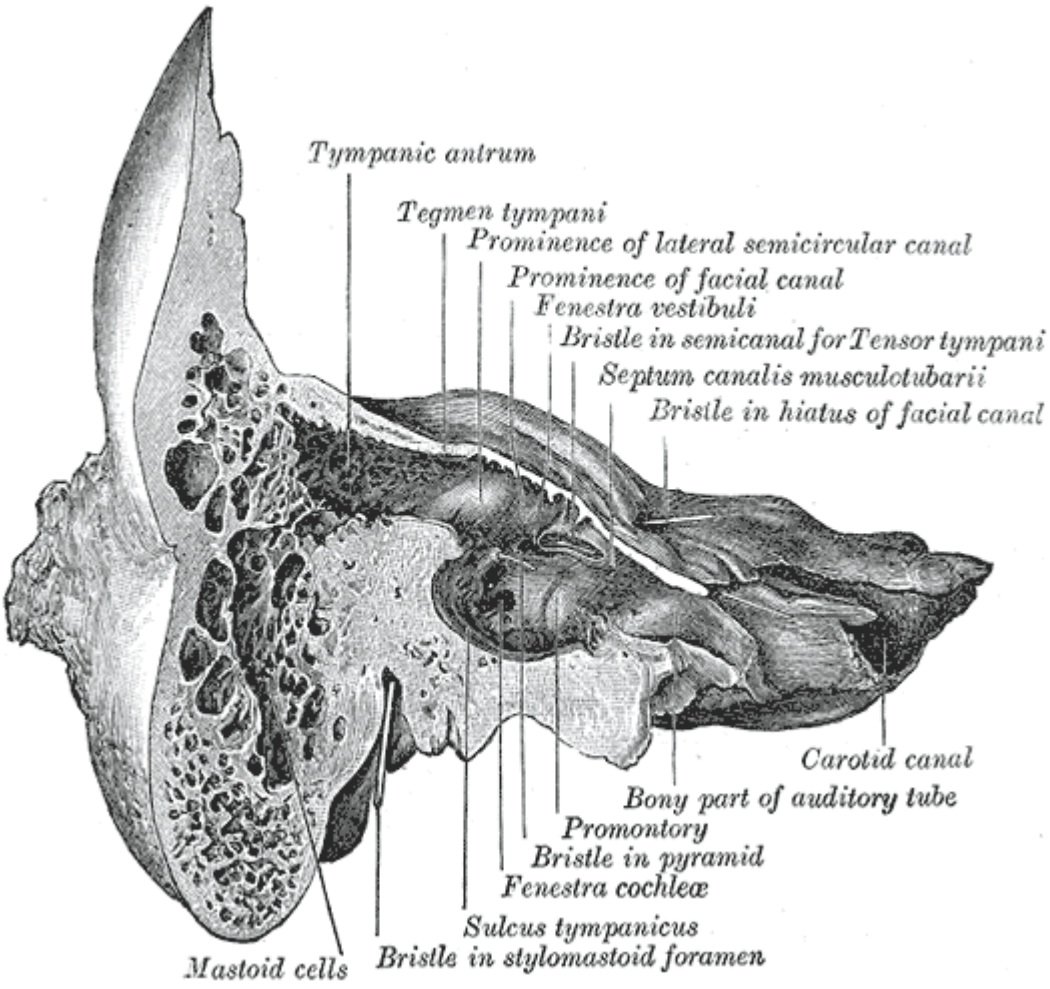

Figure 1-3: Coronal section of right temporal bone [5]

The temporal region is of particular importance to both otolaryngologists and neurosurgeons. Surgery is often necessary to correct hearing problems, balance problems or to remove cysts or tumors [1, 3].

Surgery in this region is challenging as it contains critical nerves, veins and arteries, sensitive hearing organs of the inner and middle ear and the temporal lobe. Any errors in drilling can result in partial paralysis, hearing loss, exsanguination or damage to brain matter. The risk is further amplified since this region has high anatomical variation [1, 3].

Surgeons must therefore have very specialized knowledge, extensive experience and a very high level of technical skill.

1.2 Current Training Methods

To train surgical residents effectively, four methods are currently employed at the Health Sciences Center (Winnipeg, MB). These are:

1. Operative Apprenticeship
2. Cadaveric Specimens
3. Physical Simulations
4. Virtual Simulations

Operative apprenticeships or surgical rotations entail learning by shadowing of experienced surgeons and by practicing on patients under strict supervision. This is one of the primary methods by which surgical residents gain experience and knowledge.

Due to the complex anatomy of the temporal region, specialized knowledge and precise motor skills are required. Therefore, other training methods are utilized to reduce risks to patients and to give residents the opportunity to safely develop precise drilling and motor skills.

The use of cadavers provides the most realistic training; second only to performing an operation on a living patient. However, the use of cadavers is constrained by legal issues, significant costs and limited supply. These constraints necessitate the use of physical and virtual simulations.

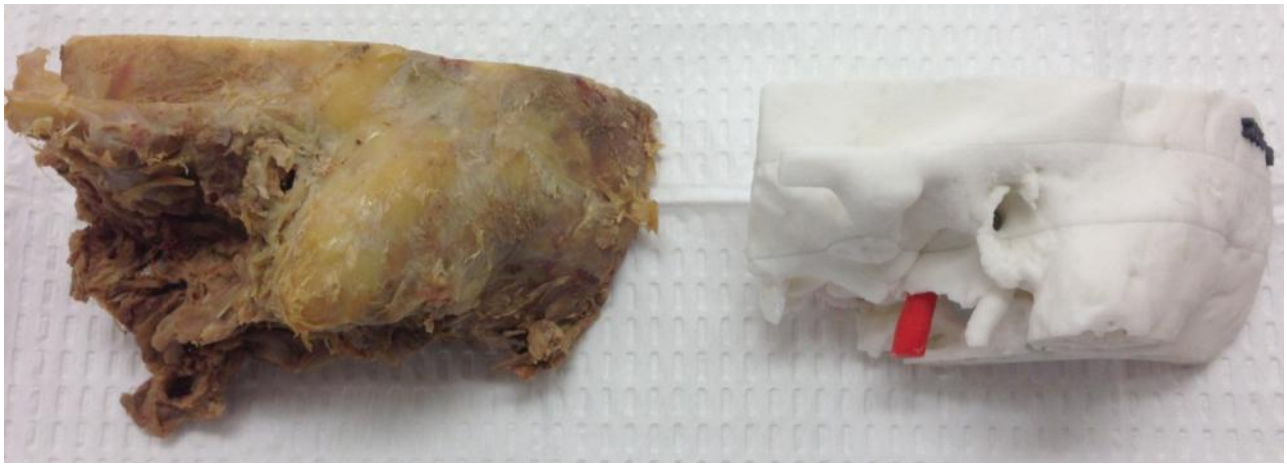

Figure 1-4: Cadaveric specimen alongside printed physical model

Physical simulations employ a surgical drill and a rapid-prototyped temporal bone. A Zprinter®650 (3D Systems, Rock Hill, SC) is used to create the rapid prototype model. These 3D models have been shown to be an excellent replica of a human temporal bone accurately representing both the internal void spaces and the mechanical bone properties [7]. Physical simulations therefore exhibit excellent drill character and do an excellent job of simulating human bone [7].

Physical simulation however has a few drawbacks:

- 3D printing is expensive
- The current 3D printing method cannot simulate the feel of soft tissue
- Significant dust is created during dissection

Figure 1-5 shows the simulation of a mastoidectomy procedure utilizing the novel rapid prototype bone.

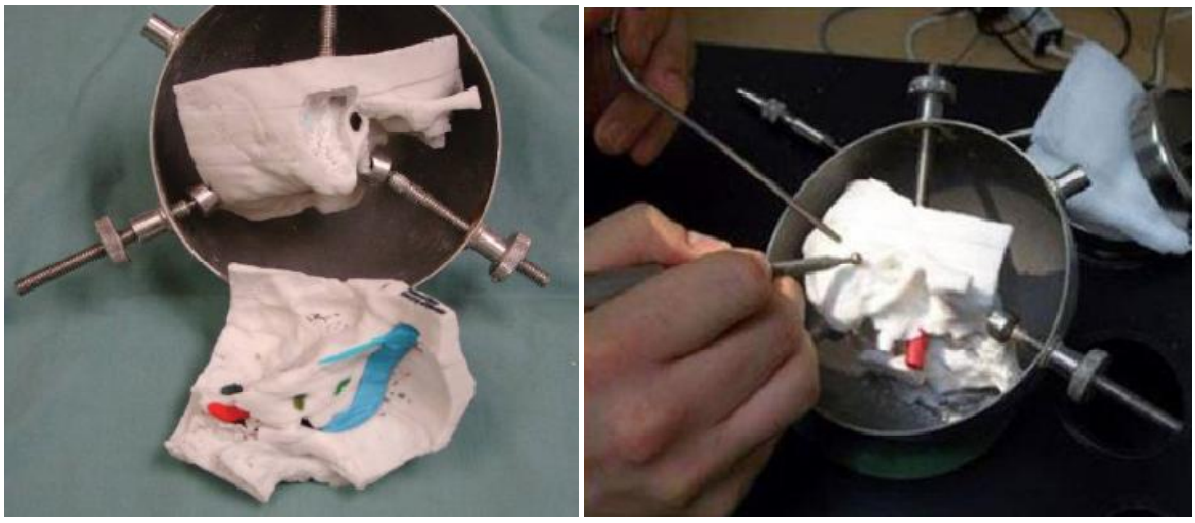

Figure 1-5: Physical simulation of temporal bone surgery [4]

Virtual haptic simulation utilizes a Geomagic® Touch™ haptic device (Geomagic, Research Triangle Park, NC) and an in-house C++ simulation developed by the Laboratory for Surgical Modeling Simulation and Robotics [5].

The pros of the virtual haptic simulation are:

- Excellent modelling of soft tissue forces
- Ability to switch between different temporal bones and cases
- Ability to create virtual teaching tools such as save, undo, zoom, modify transparency and replays

Due to size and torque limitations of back-drivable DC motors, haptic devices are severely limited by force and stiffness output. Due to these relatively low max force and stiffness ratings, haptic devices are unable to realistically simulate rigid materials such as bone. Haptic systems are therefore inferior to physical models in the simulation of drill character and bone interaction.

The Geomagic® Touch™ was formerly known as the Phantom Omni®. This thesis refers to the device by its original name for convenience. Figure 1-6 shows the virtual haptic simulation and the currently used haptic device.

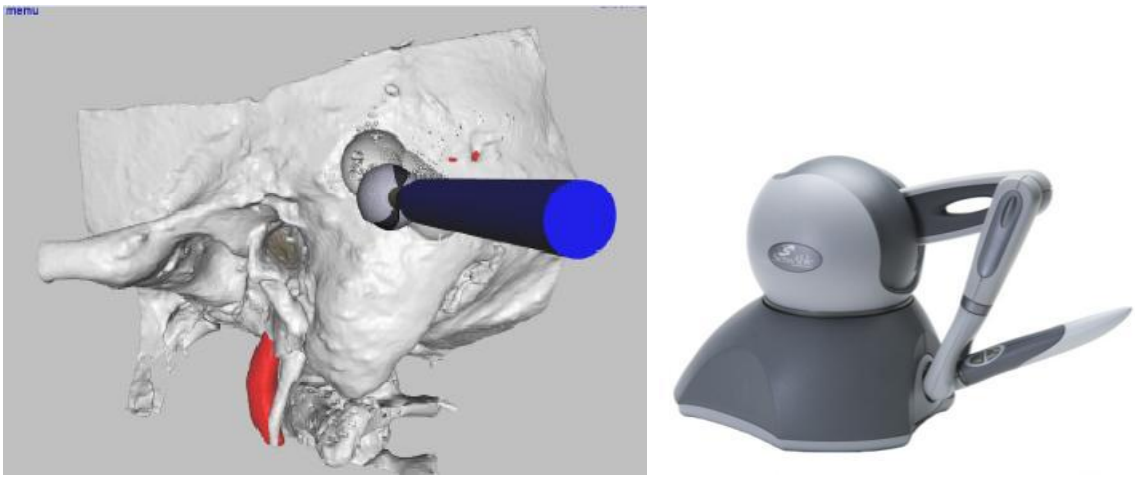

Figure 1-6: Virtual haptic simulation and the Phantom Omni® haptic device [8]

1.3 Motivation

Simulators are convenient tools to facilitate learning and training. Of the simulators reported in otology research, the majority of them are temporal bone drilling [9-13]. All otology simulators fall into three categories [9-13]:

1. Virtual – in which the anatomy is viewable on a computer screen [14-17]
2. Virtual with Haptic – tactile sensory feedback accompanies the computer image [8, 18-20]
3. Physical – this includes cadavers as well as 3D models [7, 21-24]

3D printed physical models accurately portray the look and feel of real bone and exhibit near identical drill character [7]. However, these 3D models give poor tactile feedback for soft tissue.

Due to the constraints of haptic technology, virtual haptic simulations do not realistically simulate rigid materials such as bone.

In summary, currently available temporal bone simulations are limited in realism and are therefore not very effective. According to surgeons, realism is related to the feel of the drill in addition to corresponding auditory and visual cues. How can these training methods be improved?

1.4 Research Objectives

The goal of this research endeavor is to design, develop and validate a mixed-reality (MR) haptic temporal bone surgical simulator. A MR haptic simulation is defined as a hybrid simulation which combines features from both physical simulations and virtual haptic simulations. To date, this has not yet been achieved in the field of otology and has not been reported in the literature [9-13, 25]. Achieving MR haptic simulation would not only improve medical training and learning, but would also advance the field of haptic research.

To achieve this goal, a surgical drill must be attached a haptic device and the 3D printed model and the virtual temporal bone model must be precisely aligned. The printed model would simulate rigid bone, whilst soft tissues would be simulated via haptic forces.

Therefore, a mixed reality surgical simulation would combine the separate strengths of both virtual simulations and physical simulations while negating their individual weaknesses. However attempting MR simulation may introduce new and complex representational and implantation problems.

This thesis seeks to answer three research questions:

1. Attaching and integrating a surgical instrument to a haptic device is a significant modification. What are the potential problems that may arise? Would this have any impact on the performance, stability and usability of the VR haptic simulation?
2. Does the addition of a surgical drill improve the realism of a temporal bone virtual haptic simulation? Would expert surgeons prefer a virtual haptic simulation with the default end effector or would they prefer to hold a real vibrating drill when performing virtual haptic surgeries?
3. Is there a discernible difference in end user experience if the same virtual haptic surgical simulation program is run on different haptic devices?

1.5 Prior Work

The work presented in this thesis builds upon prior work. This section acknowledges the contributions of former and current members of the Laboratory for Surgical Modeling Simulation and Robotics Lab.

The chosen haptic device is the High Definition Haptic Device^{TM} (or HD^2) produced by Quanser® Inc., Markham, Ontario. A mechanical clamp allowing a surgical drill to be attached below the HD^2 end-effector handle was previously designed and built by the lab [25]. Figure 1-7 shows the CAD design and Figure 1-8 shows the surgical drill attached to the HD^2 haptic system.

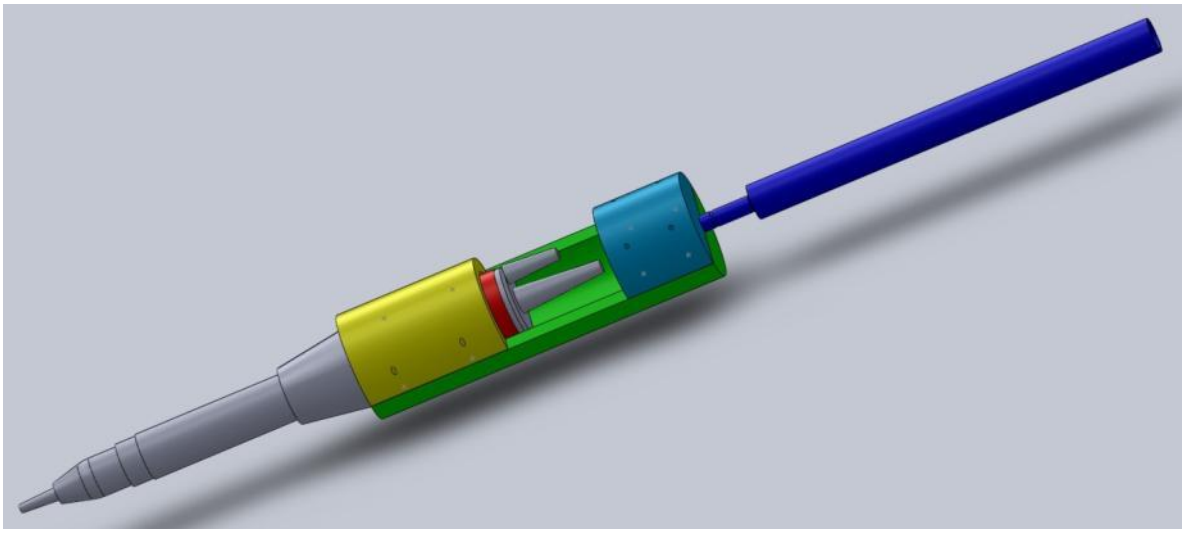

Figure 1-7: Mechanical clamp used to attach the surgical drill below the handle [25]

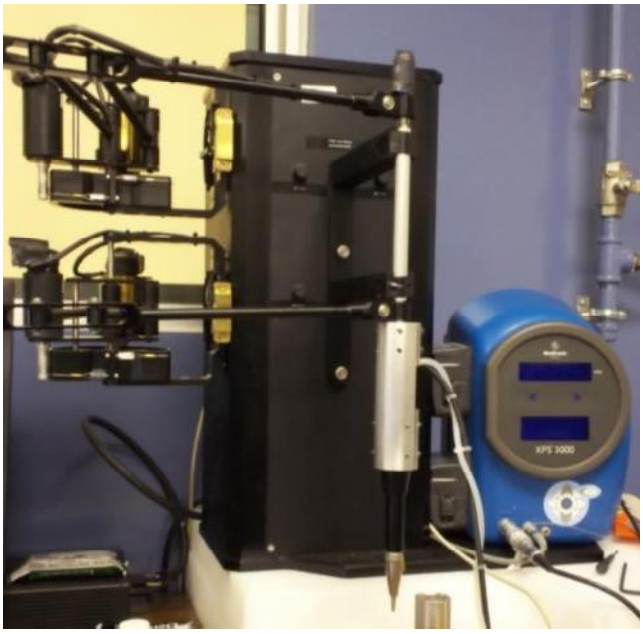

Figure 1-8: Surgical drill attached to the HD[2]

A C++ multi-threaded, windows-based, temporal bone virtual haptic simulation was previously developed. A system for segmenting patient anatomy and creating realistic virtual models and 3D rapid prototypes was also developed by the Laboratory for Surgical Modeling, Simulation and Robotics [7, 8]. Figure 1-9 shows the slicing of the virtual temporal bone model in preparation for 3D printing.

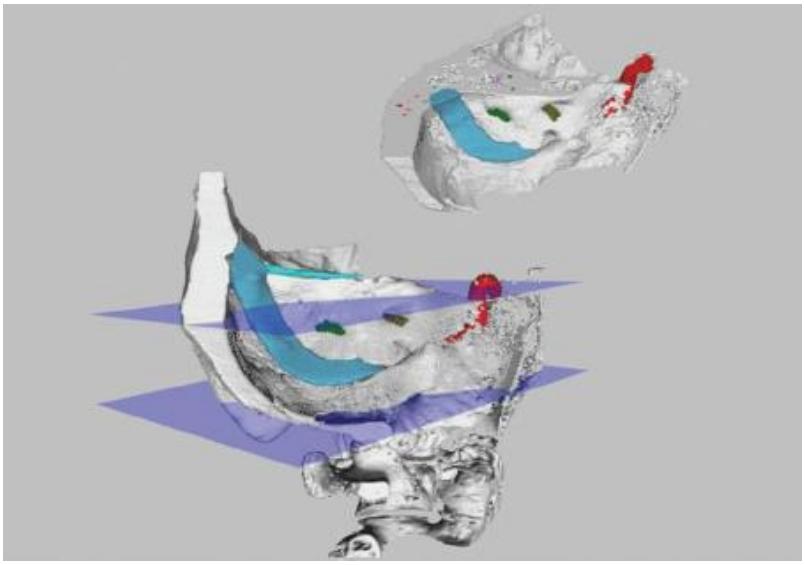

Figure 1-9: Slicing of anatomy to allow for effective 3D printing [7]

1.6 Methodology and Thesis Outline

To accomplish our research goals, several tasks were deemed essential. These tasks are presented here in chronological order.

The very first task was to understand fully the workings of the chosen haptic device. An overview of the High Definition Haptic Device is given in Chapter 2 of this thesis. Hardware specifications are included in "Appendix A – HD2 Detailed Technical Specifications". Details of how the Simulink® driver works including the governing kinematic equations are presented in "Appendix C – Driver Details". Chapter 2 also covers the errors found in the Simulink® drivers and briefly touches on how MR haptic simulation would be accomplished via fiducial registration.

From understanding the kinematics of the High Definition Haptic Device, an equation was derived to transmit the haptic interaction from the default handle center to any point along the line of the end effector wand. This equation was used to simulate the haptic force at the tip of the attached surgical drill. This equation was also used to negate the weight of the mechanical attachment used to clamp the drill below the haptic end effector wand. This work is presented in Chapter 3. Also included in Chapter 3 is the calibration of initial joint angles as well as the motor voltage bias correction. Chapter 3 also covers interfacing the HD^2 to the C++ temporal bone haptic simulation and the integration of the surgical drill state into the C++ simulation.

Two tangential objectives were also researched. How would one go about designing a testing rig for force validation testing of the High Definition Haptic Device? An adjustable clamp and platform was designed which rigidly attached to a force and torque sensor. The clamp dimensions were chosen so that the clamp would be able to grip a small spherical object such as a drill burr whilst still being able to clamp the upper and lower arms of the haptic device as well as the upper and lower ends of the haptic end effector. A second objective was investigating the suitability of using the Quarc Visualization interface for the generation of graphics in temporal bone haptic simulation. These two objectives did not directly contribute towards achieving VR haptic simulation on the HD^2 and are thus included in "Appendix B – Related Technical Work".

Once the HD^2 and the C++ simulation were successfully integrated, haptic instability was observed when the drill tip interacted with the surface of the virtual bone. The haptic chatter was

examined, traced and corrected using low-pass filters. This work is presented in Chapter 4. At this point, a working VR haptic simulation ran robustly on the modified HD^2 system. However, MR haptic simulation could not be attained due to a substantial position error inherent to the HD^2 system.

Chapter 5 describes a study evaluating the temporal bone virtual haptic simulation on three separate hardware configurations: the Phantom Omni, the standard HD^2 and the HD^2 with an attached surgical drill. The chapter presents the research and design methodology, the preparatory engineering work required, the results, analysis and recommendations for future studies. The evaluative study seeks to answer the two following questions:

1) Would an attached surgical drill be preferred by expert surgeons and will it improve surgical realism?
2) Would running the identical software simulation on different haptic hardware result in different learning experiences

Chapter 6 presents the conclusions, contributions obtained from this research and briefly presents suggested future works.

2 Technical Background and HD^2 Limitations

This chapter gives an overview of haptics technology. It also gives a description of the chosen haptic hardware and a brief overview of its driver software. Limitations of the HD^2 and discovered driver errors are also presented in this chapter. The method of aligning real and virtual models, fiducial registration, is also presented.

2.1 Haptics Overview

Haptics refers to sensing or manipulation through touch [26]. The term had been used by psychologists in the early 1900's for studies involving humans touching real objects [26]. Today the definition of haptics has expanded to include, "all aspects of information acquisition and object manipulation through touch by humans, machines, or a combination of the two; and the environments can be real, virtual or teleoperated" [26].

Figure 2-1 below gives an overview of how the current temporal bone virtual reality surgical simulation works.

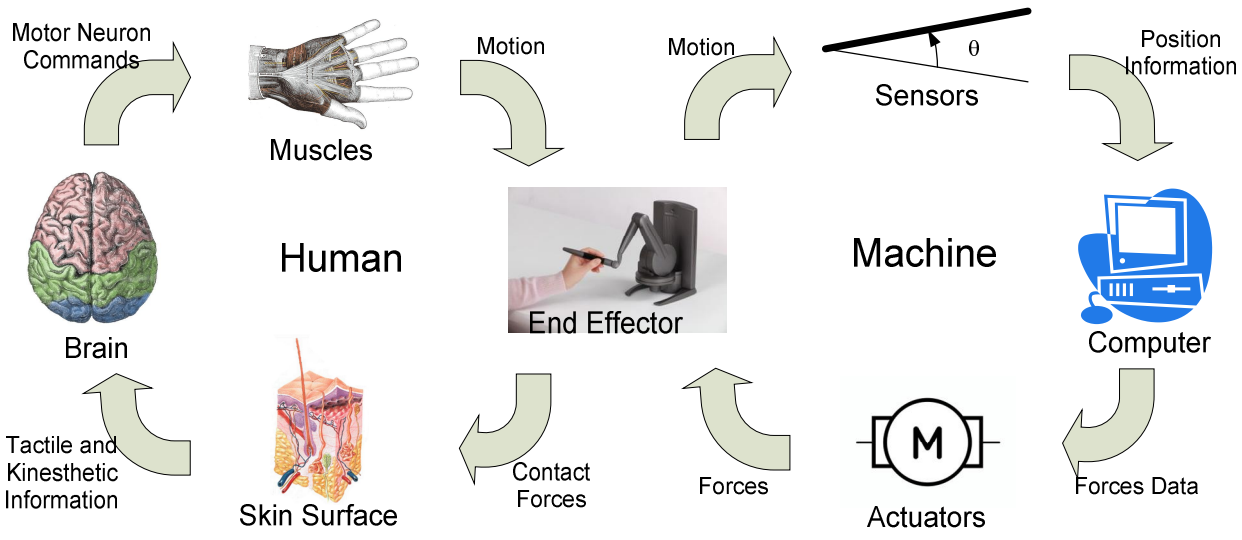

Figure 2-1: An overview of haptics [27]

Many commercial haptic devices can be described as specialized robotic manipulators. Both the HD^2 and the Phantom Omni are robots which operate in force control mode. A human user is able to move the robot's end effector in three dimensional space [26, 27]. This 3D spatial co-ordinate can be read by a computer. This is done via sensors located on the haptic robot. Typically, encoders are on each joint, thus allowing the end position to be calculated [28]. A computer system which calculates the end-effector end position also checks this position against the location of virtual scene objects [26, 27]. This computer system will often have a graphical interface showing the relative position of end-effector to the virtual object [8, 26].

If the end position intersects with a virtual object, a force will be commanded in the opposing motion [24]. The force is a function of the penetration distance of the end-effector into the virtual object and may also take into account the current end effector velocity [8]. A force at the end-effector is created by having several joint motors actuate in synchrony [27].

The human user is able to sense forces, vibrations and other tactile feedback via sensory nerves in their skin [26, 27]. This feedback travels along neurons to the brain, where it is processed [26, 27]. The human brain will then decide where to move the hand (and the end-effector) in response to this sensory disturbance [26].

Both the human and machine systems have sensors, processors and actuators [27]. These two systems working together allow complicated virtual reality haptic tasks to be practiced or performed.

2.2 HD^2 – Description & Specifications

The High Definition Haptic Device or HD^2 is a haptic device designed and manufactured by Quanser Inc (Markham, Ontario). The device is shown in Figure 2-2 below.

Mechanically it is two 3-DoF linkages connected in parallel via a pair of universal joints. The 6 motors are capstan driven and work together to produce force feedback to the user in 5-DoF. Seven high resolution encoders allow motion to be captured for 6 DoF [28, 29].

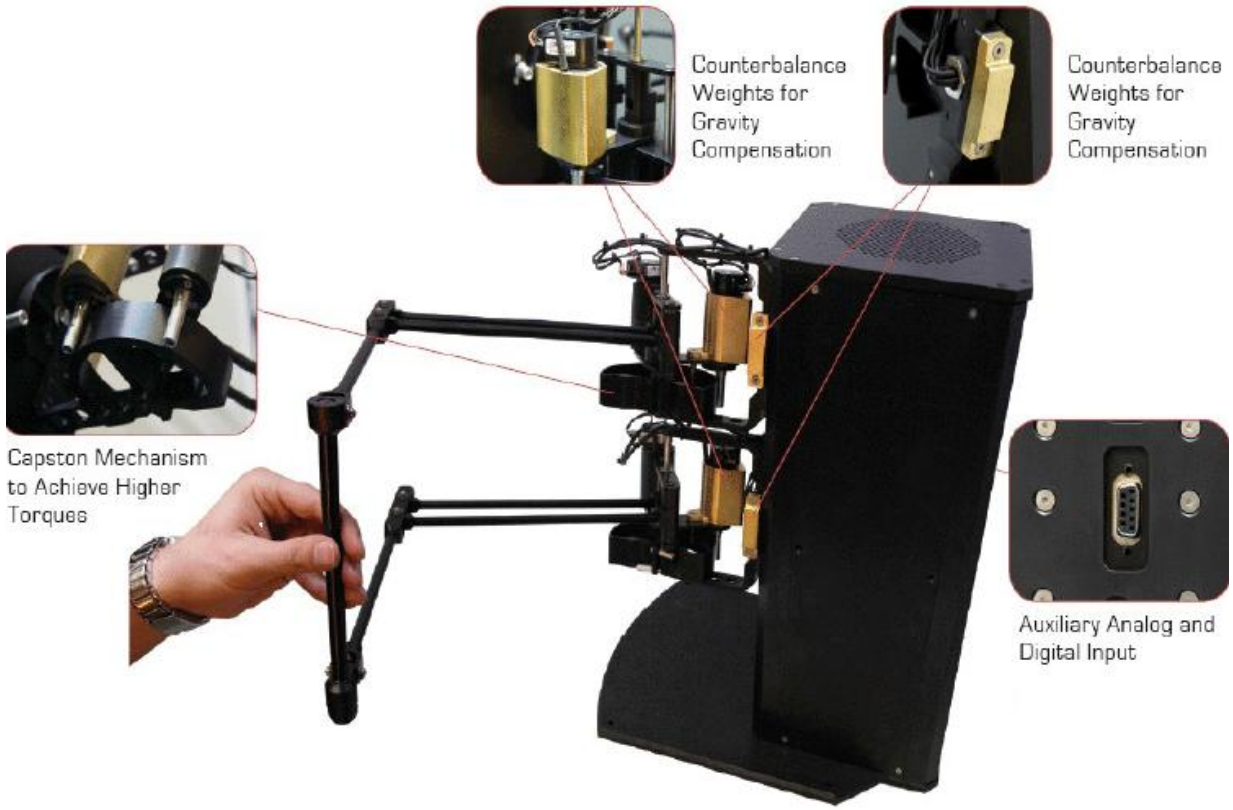

Figure 2-2: Features of the HD^2 [28]

The important specifications of the device are shown in Table 1 and Table 2. The complete specifications can be found in Appendix A – HD2 Detailed Technical Specifications.

Table 1: HD^2 Specifications [29]

	X	Y	Z	Roll	Pitch	Yaw
Workspace	800 mm	250 mm	350 mm	180°	180°	Continuous
Tip inertia	300g	300g	300g	2.29 g.m^3	2.29 g.m^3	0.79 g.m^3
Back drive frictions	0.353 N	0.353 N	0.353 N	61.755 N/mm	61.755 N/mm	0.5 N/mm
Max Force/ Torque	19.71 N	19.71 N	13.94 N	1.72 N.m	1.72 N.m	1.72 N.m
Cont Force/ Torque	10.48 N	10.48 N	7.67 N	0.948 N.m	0.948 N.m	0.948 N.m
Position Resolution	0.051 mm	0.051 mm	0.051 mm	0.033°	0.033°	0.088°
Stiffness at 10 kHz	3000 N/m	3000 N/m	3000 N/m	3.4 N.m/rad	3.4 N.m/rad	0.05 N.m/rad
Dimensions [H × W × L]	0.53 m × 0.3 m × 0.5 m					
Total Mass	22 kg					

Table 2: Link Lengths of the HD^2 [28]

Linkage	Length [m]
Motor Arm Length	0.280
Forearm Length	0.290
End-Effector Handle	0.175

2.3 HD^2 Driver Overview

The HD^2 driver runs at 1000 Hz and is coded in Simulink. This section gives a simplified overview of the driver's operation. A detailed and complete overview of the Simulink driver architecture can be found in Appendix C – Driver Details.

Since the HD^2 device is a robot operating in force control mode, the main inputs to the driver are a force vector, [F_x F_y F_z] and a torque vector [T_x T_y T_z]. The force is generated at the center of the end-effector handle and the torque vector is defined about this center position [30].

The main outputs from the driver are position, [x y z] and orientation angles, θ_x and θ_y [30]. The coordinate system for the base frame of the robot is shown in Figure 2-3.

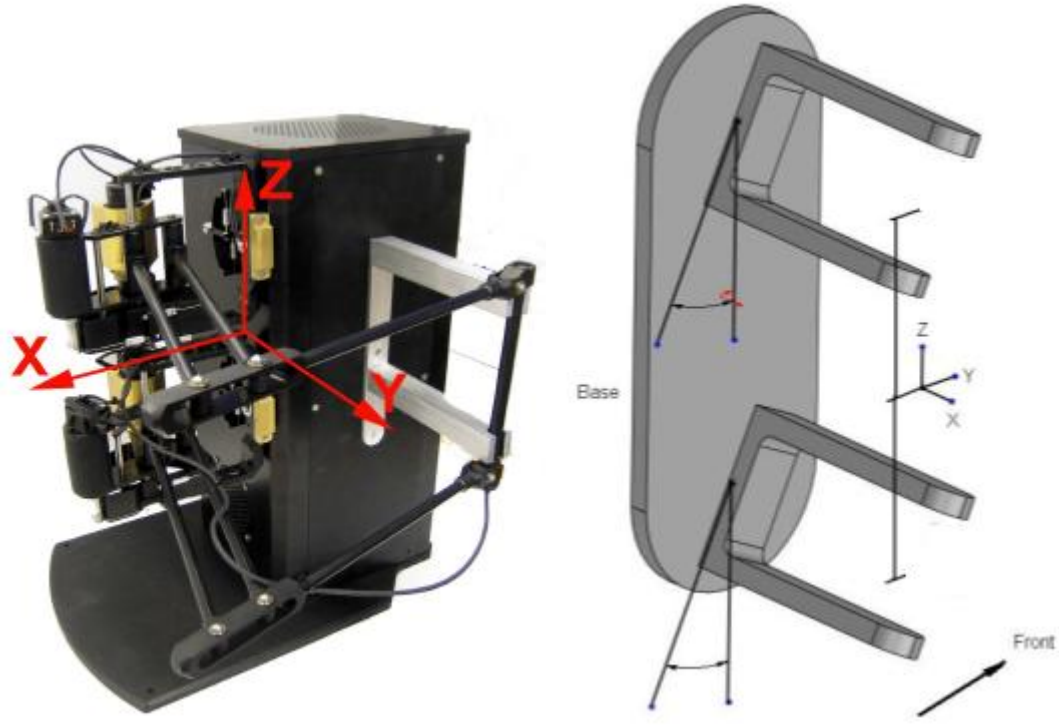

Figure 2-3: The base co-ordinate system of the HD^2 [28, 31]

The driver treats the parallel arms as separate. The end positions of the upper and lower arms are found independently of one another and thus the driver calculates two separate Jacobian matrices.

Since the haptic force is generated at the center of the haptic handle, the commanded force is split evenly between the upper and lower arms.

$$F_{upper} = F_{lower} = \frac{1}{2} F_{commanded} \tag{2-1}$$

For the case of simulating a torque about the handle center, this is accomplished by decoupling the torque vector into parallel and perpendicular components. The parallel torque is simulated using the motor located on the end effector handle. For the perpendicular torque component, this is simulated by using couple forces; i.e. having a pair of equal but opposite forces located at the upper and lower ends of the handle.

2.3.1 Forward Kinematics and the Jacobian Matrix

The HD^2 comprises of two parallel arms each made up of three linkages. Link 1 is connected to the base or the main body of the HD^2. The ends of link 3 for both the upper and lower arms are connected to the end-effector haptic wand via universal joints. The linkage labels are illustrated in Figure 2-4.

For the latest driver version, the link angles are set to be zero at the "home position". The "home position" is defined as when links 2 and 3 point in the Y axis direction as shown in Figure 2-5. It is important to note that this Y-axis oriented position is theoretical as the linkages 2 and 3 can never be parallel due to mechanical constraints. It is also important to note that θ_2 and θ_3 is defined with respect to global Y-axis direction and not to the previous link.

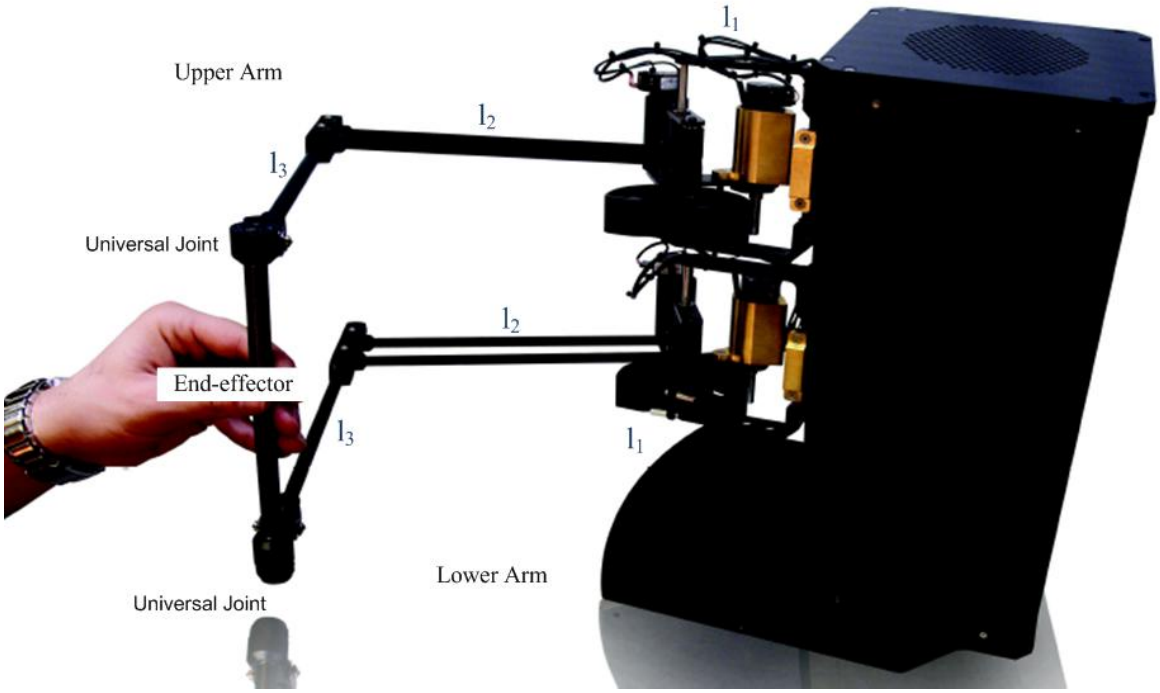

Figure 2-4: Figure showing the labelled linkages of the HD[2] [28]

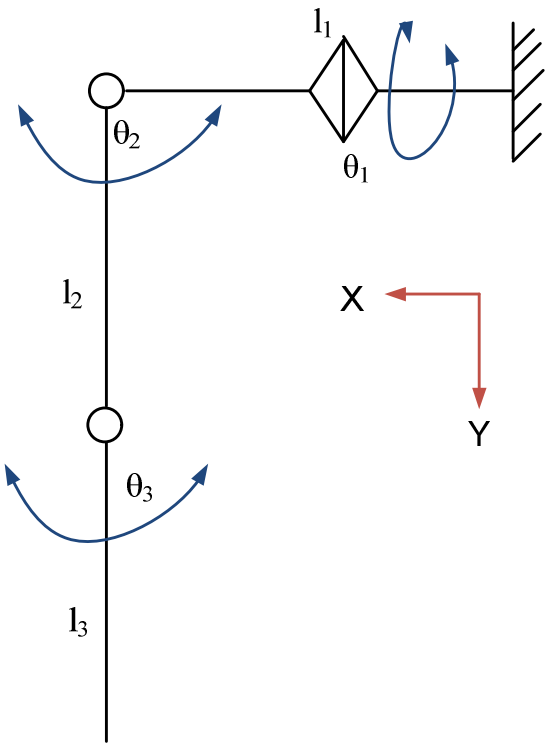

Figure 2-5: Theoretical home position of upper and lower arms (top view)

Based on geometry, the forward kinematic equations for both the upper and lower arms are defined as shown in Equations (2-2), (2-3) and (2-4). Equation (2-4) takes into account the Z offset of the upper and lower arm from the base frame.

Equations (2-5) to (2-7) give the formulation of the Jacobian matrix and the joint torques for the upper or lower arm.

$$x = l_1 - l_2 \sin(\theta_2) - l_3 \sin(\theta_3) \tag{2-2}$$

$$y = l_2 \cos(\theta_1)\cos(\theta_2) + l_3 \cos(\theta_1)\cos(\theta_3) \tag{2-3}$$

$$z = l_2 \sin(\theta_1)\cos(\theta_2) + l_3 \sin(\theta_1)\cos(\theta_3) \pm h/2 \tag{2-4}$$

$$J = \begin{bmatrix} \partial x/\partial\theta_1 & \partial x/\partial\theta_2 & \partial x/\partial\theta_3 \\ \partial y/\partial\theta_1 & \partial y/\partial\theta_2 & \partial y/\partial\theta_3 \\ \partial z/\partial\theta_1 & \partial z/\partial\theta_2 & \partial z/\partial\theta_3 \end{bmatrix} \tag{2-5}$$

$$J = \begin{bmatrix} 0 & -L2\, c_{\theta 2} & -L3\, c_{\theta 3} \\ -L2\, s_{\theta 1} c_{\theta 2} - L3\, s_{\theta 1} c_{\theta 3} & -L2\, c_{\theta 1} s_{\theta 2} & -L3\, c_{\theta 1} s_{\theta 3} \\ L2\, c_{\theta 1} c_{\theta 2} + L3\, c_{\theta 1} c_{\theta 3} & -L2\, s_{\theta 1} s_{\theta 2} & -L3\, s_{\theta 1} s_{\theta 3} \end{bmatrix} \tag{2-6}$$

$$\tau = J^T \, {}^0_e F \tag{2-7}$$

Where,

J is the Jacobian matrix

F is the desired force at the end of the Link 3 with respect to global co-ordinates

τ is a 3x1 matrix of Joint Torques for Links 1, 2 and 3

Based on the required torques at each joint, a proportional voltage signal is fed to each motor. All six motors actuate in sync to create the sensation of a force at the end-effector handle.

It is important to note that Equations (2-2), (2-3) and (2-4) give the position at the end of the upper and lower arms; this is not equivalent to the position of the end-effector handle. A 22mm offset link which acts as a universal joint attaches the upper and lower ends of the handle to the ends of the upper and lower arms.

To find the end position, the local x and y rotation angles of the two offset links are needed. However as there are no encoders on these links, these four angles must be calculated. At the time of writing this thesis there were two versions of the driver software.

The original driver attempts to iteratively deduce these four angles using a Newton Method Iteration [28]. (See Appendix Section C.2.2 Forward Kinematics of the Original Driver.) Whereas the updated driver does a direct calculation for these angles based on the assumption that the end-effector handle is parallel to the line joining the end points of the upper and lower arms. This is covered in Appendix Section C.3.6 Finding the Handle Center.

2.4 Rectified System Errors

In the process of documenting and understanding the HD^2 driver, system errors were found. This section presents errors that were corrected.

2.4.1 Motor Bias Correction

It was found that when the system had power, but the commanded force and torque was set to zero, a small force and torque can be felt in the end effector handle. This indicates that when the commanded voltage is zero, the actual voltage output from the amplifying circuitry was not zero. This problem is discussed in Section 3.2. This issue was corrected by adding very small voltage biases to the motor voltage control signals.

2.4.2 Decomposing Torque into Couple Forces

For the original HD^2 driver, the math that performed the decomposition of commanded torque into the required upper and lower forces was flawed. This error is explained in Appendix Section C.2.3.

We corrected the math error, as presented in Section 3.3 using a computationally efficient algorithm. Subsequent driver updates from Quanser also addressed this problem (Appendix C.3.2). Since update rates were considered essential to haptic fidelity, and Quansar's updated

proved less computationally efficient than our own, we ultimately used our in-house algorithm for the drive.

2.5 End-effector Position Error

A severe limitation of the chosen haptic device was found late in development. Once full VR temporal bone surgical simulation was achieved with the modified HD^2 system (see Sections 3.7 and 4.5), the next step was to achieve mixed reality simulation. A method called fiducial registration was employed to accurately align real and virtual models with one another. This method is briefly described in Section 2.6.

While this method was able to accurately align virtual and real models when used with the Phantom Omni haptic device, alignment proved to be impossible with Quanser's HD^2. Though the misalignment error cannot be accurately measured, its range is approximately 5 – 20mm depending on drill position and orientation. This value was estimated using a ruler to measure the difference in position between the drill tip on the real model vs. its position on the virtual model displayed on the computer screen.

This misalignment error was investigated and it was found that the HD^2 system itself was inherently inaccurate; specifically the calculation of the end-effector position. Due to the complexity of independently measuring the end-effector co-ordinate position, calculated handle length was used as a surrogate error metric.

The HD^2 end-effector handle has a length of 175 mm. Since both the original and updated drivers report the upper and lower positions of the end effector wand, the length between the upper and lower position can be calculated and compared to the absolute length of 175 mm. For a perfect system, the difference between the calculated handle length and the actual absolute length should be very close to zero.

For both the original and updated driver, it was found that this error varies between ±5mm depending on position and orientation. This inaccuracy is illustrated in Figure 2-6. The sample data was captured from a trial run of the virtual reality temporal bone surgical simulation using the modified HD^2 system.

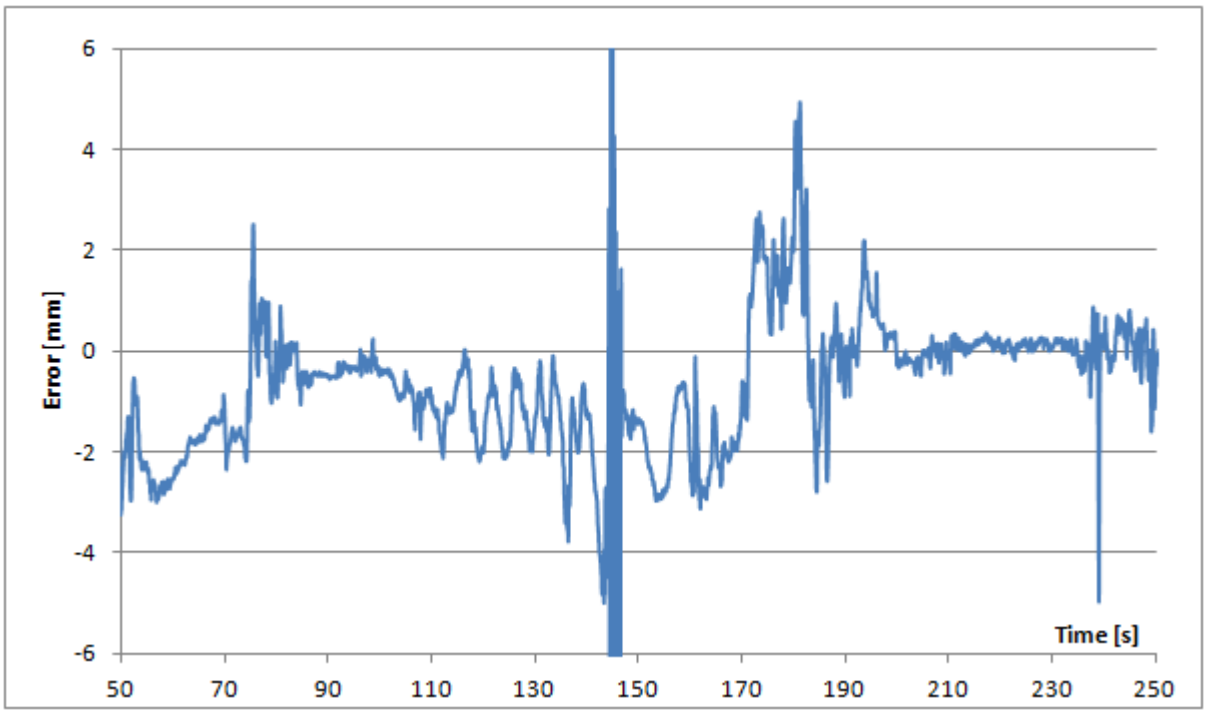

Figure 2-6: Handle length error vs. time

It was confirmed that this positional error was not limited to our HD^2 unit alone. Quanser ran their HD^2 close to the home position and produced Figure 2-6 [32]. Figure 2-6 shows a handle error in the range of -3mm to +4mm.

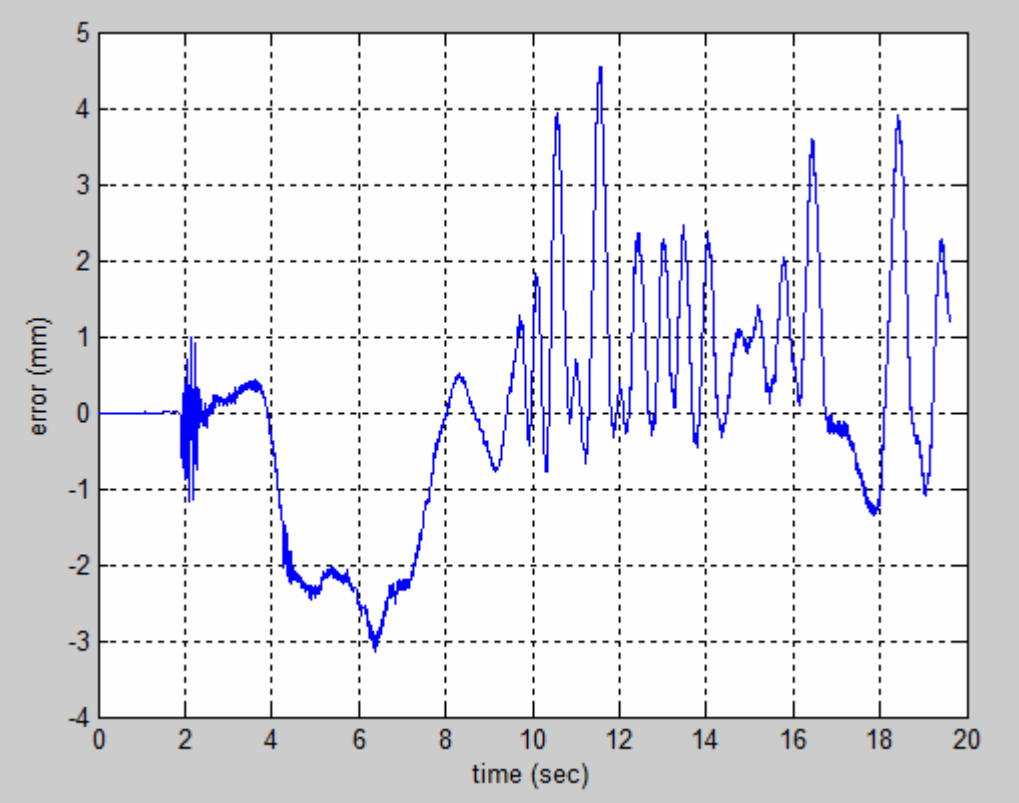

Figure 2-7: Handle length error reported by Quanser [32]

Therefore it can be concluded that the advertised accuracy specification of 0.05mm is highly over-estimated. (See Section 2.2, Table 1: HD2 Specifications [29].)

The misalignment between virtual and real objects is as high as 20mm whilst the typical handle length error appears to be typically ≤ 5mm. This additional inaccuracy is explained as follows:

1. Fiducial registration relies on mapping and linking three pairs of points between the real and virtual world. The final position uncertainty can therefore be as high as 3 times the uncertainty of a single position point.
2. The handle error metric represents the uncertainty of the upper and lower handle positions. What the simulation actually uses is the drill tip position. Due to the mechanical setup, the uncertainty of the drill tip position is magnified as is shown in Figure 2-7 below.

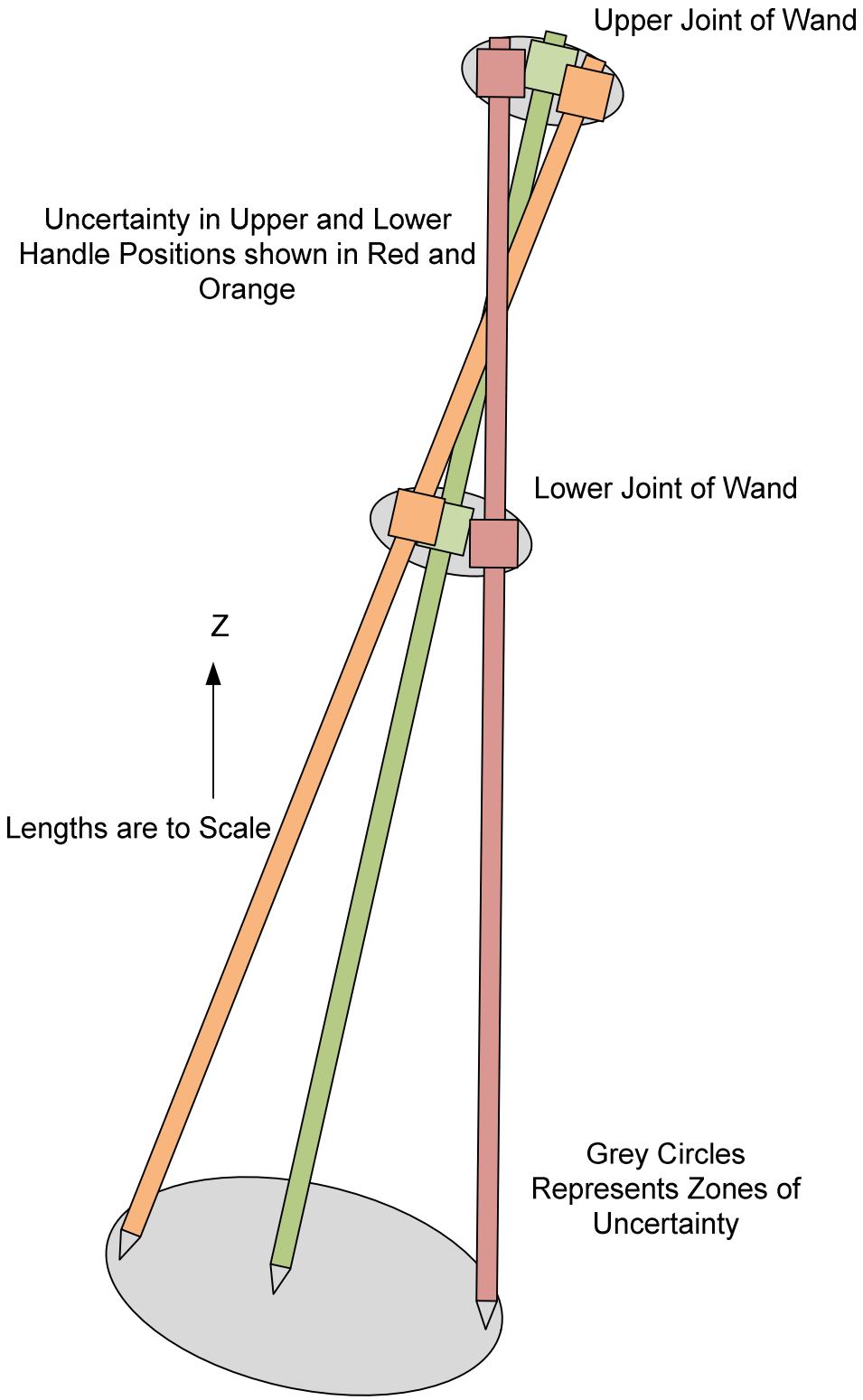

Figure 2-8: Magnification of handle position error due to mechanical configuration

Therefore, relatively small errors in the upper and lower joint locations result in a much larger position error at the drill tip.

2.5.1 Explaining the Inaccuracy of the HD^2 End-Effector Position

The upper and lower joint positions of the HD^2 handle are not precisely calculated. This is due to numerous reasons:

1. There are no encoders located on the handle's universal joints. Therefore simple forward kinematic calculations to determine the upper and lower handle positions is impossible
2. Even though a symbolic formulation of the end effector position exists, its application has not been successfully realized outside of simulation [32, 33].
3. For the updated drivers, the method of calibrating the initial joint angles is not ideal and thus the initial angles may not be accurate (see Section 3.1). For the original driver, the initial angles assumption may also be incorrect [32].
4. The forward kinematic calculations for both the original driver and updated driver are imperfect:
    a. For the original driver, the roll and pitch angles of the upper and lower joints are iteratively calculated using the Netwon's method iteration. Therefore, this method, even if accurate, will never give the position in real time. This is discussed further in Appendix Section C.2.2.
    b. For the updated driver, the roll and pitch angles of the universal joints are not iteratively calculated. An assumption is made however, that the handle is parallel to the end points of the upper and lower arms before the universal joints. (See Appendix Section C.3.2.) An elementary inspection of the HD^2 can show that this assumption is true if and only if $\theta_x = 0°$. Therefore, the forward kinematic calculation is a based on a crude approximation.
5. To a lesser extent, manufacturing tolerances and flexing (non-rigidity) of robotic links may also contribute to the system's inaccuracy [32]. Robotic manipulator kinematics is based on the assumption that each link in the system is a perfect rigid body [34]. If this assumption is invalid, Forward Kinematic calculations will not be accurate.

Correcting this inherent position error was deemed to be outside the scope of this thesis.

2.6 Fiducial Registration

A fiducial or a fiducial marker is an identifying mark such as a dot, line or point which is placed on a real object which also appears on an image, for use as a point of reference or measure [35, 36].

To allow real and virtual temporal bone models to be aligned with one another, three markers were added to both the 3D printed model and to the virtual bone model. Three points are enough to fully constrain and align the position and orientation of the virtual model to the real physical model [36]. Figure 2-9 shows the fiducial markers on the printed and virtual models respectively.

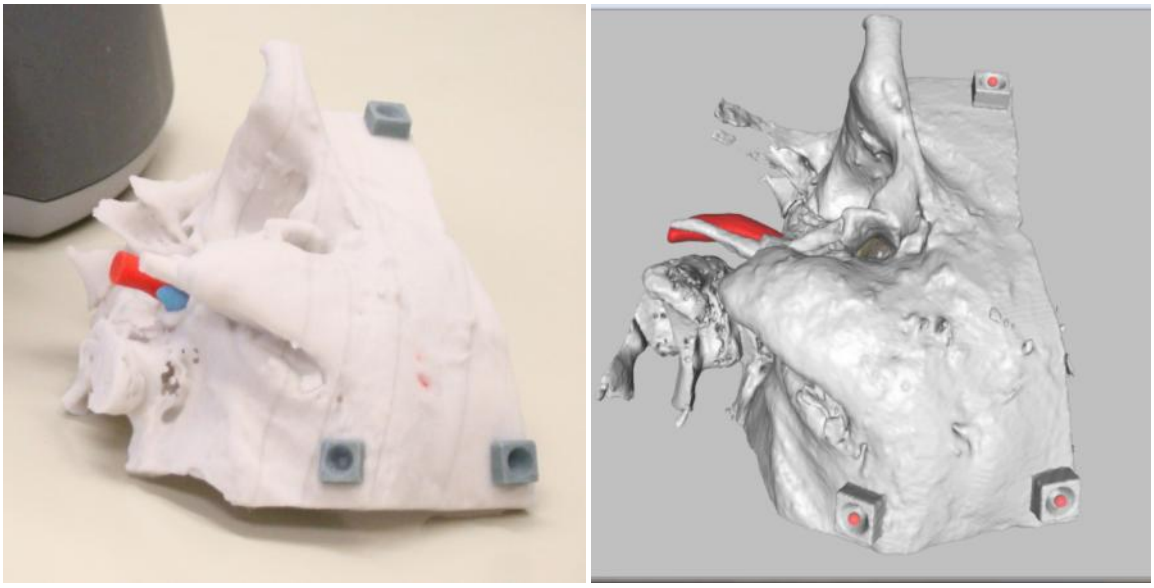

Figure 2-9: Fiducial markers on printed temporal bone model (left) and on the virtual model (right)

Using the haptic end-effector tip, the physical markers must be individually paired to the corresponding virtual markers. Once this is done for all three markers, the real and virtual models are aligned using an Iterative Closest Points algorithm. This is an algorithm which

compares two point clouds and returns the rigid transformation, (rotation matrix and translation vector) that best aligns the point clouds [37]. This is illustrated in Figure 2-10 below.

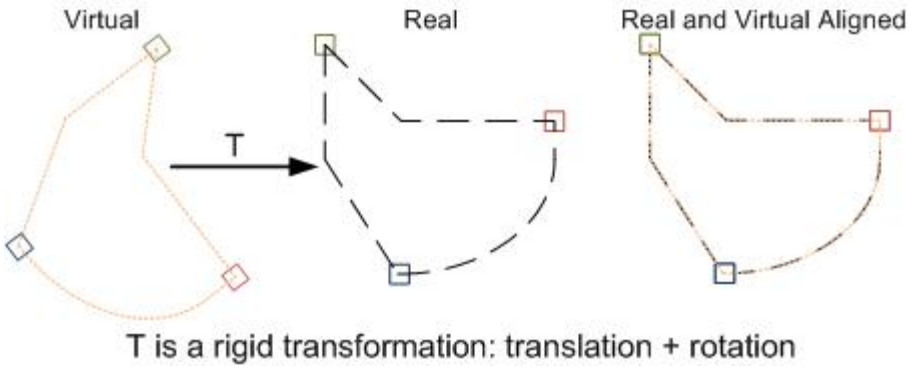

Figure 2-10: Aligning real and virtual models using rigid transforms

Unfortunately, due to its inherent position error of the High Definition Haptic Device, accurately aligning real and virtual models was not possible. Though virtual reality haptic simulation was ultimately achieved (see Sections 3.7 and 4.5), mixed reality haptic simulation was not possible.

2.7 Chapter Summary

This chapter broadly describes how haptic systems work. It presents the technical specifications of the HD^2 system and gives an overview on the workings of the driver software. A detailed presentation of the Simulink driver controls and kinematics is given in Appendix C.

Two major errors in the HD^2 system was identified and corrected; these are motor bias correction and the math involved in decomposing the commanded haptic torque into force couples. It was also discovered that the HD^2 is inherently inaccurate and that the manufacturer's specification of 0.05 mm position resolution is false. Using the haptic wand length as an error metric and comparing it to the calculated length between upper and lower end points it was found that there was an error of -5 to +5mm.

Correcting this positional error was deemed to be outside the scope of this MSc thesis. This positional error made accurate alignment between real and virtual models impossible and thus mixed reality haptic simulation could not be achieved with Quanser's High Definition Haptic Device.

3 Achieving HD² Virtual Reality Surgical Simulation

VR haptic simulation with the modified HD^2 system must first be achieved before mixed reality can be attempted. To achieve this more immediate goal, several steps had to be undertaken.

This chapter covers the calibration of initial joint angles, motor bias correction, the derivation and implementation of an algorithm which changes the haptic interaction point, weight cancellation of the drill-gripper attachment, interfacing the Simulink driver to the C++ temporal bone simulation and the integration of the otic drill state into the simulation.

During this time, research was also conducted into the viability of using "Quarc Visualization" for VR haptic temporal bone surgical simulation. "Quarc Visualization is a 3D graphics library from Quansar that works with Quarc entirely within the Simulink framework. Since this feature was not utilized and thus did not contribute towards the working VR haptic simulation on the modified HD^2 system, it is not covered in the main body of this thesis. However, the lessons learnt from working with "Quarc Visualization" are documented in Appendix B.

3.1 Calibration of Initial Joint Angles

Calibration is done automatically for both the original and updated driver and this takes place whenever the Quarc driver executable is started. The haptic manipulandum must be seated squarely in its supporting rack during start-up.

At the time of start-up, the encoder counts are zero and any positive or negative counts accrued are added to pre-set initial angles. For the updated 'HD^2 IO' Driver these initial angles had to be calibrated and set. A manual calibration Simulink program is shown in Figure 3-1 below.

The manual calibration program must be started with the haptic wand seated in its holder. Each joint is independently checked and set. When the joint is moved to its zero position, the angle output should read zero. For joint 1, this occurs when both Links 2 and 3 are in the X-Y plane. For Joint 2, this is when Link 2 points in the Y direction. For Joint 3, its zero position is also when Link 3 points in the Y direction. If these angles are not zero in their 'zero positions' their initial (reference bias) angles must be tweaked.

Using rulers and set squares, these initial angles were set as [2 -15 101 2 -15 101 0 0]. The first three numbers are the initial angles for the upper kinematic chain, and numbers 4 – 6 are the initial angles of the lower kinematic chain. Angles 7 and 8 refer to the scissors angle and Yaw angle of the wand. These last two angles are not important and are thus set to zero for their initial values.

Since this method of calibration relies on the human eye, it is not an accurate calibration method. This may be a contributing factor to the end-effector position error described in Section 2.5.

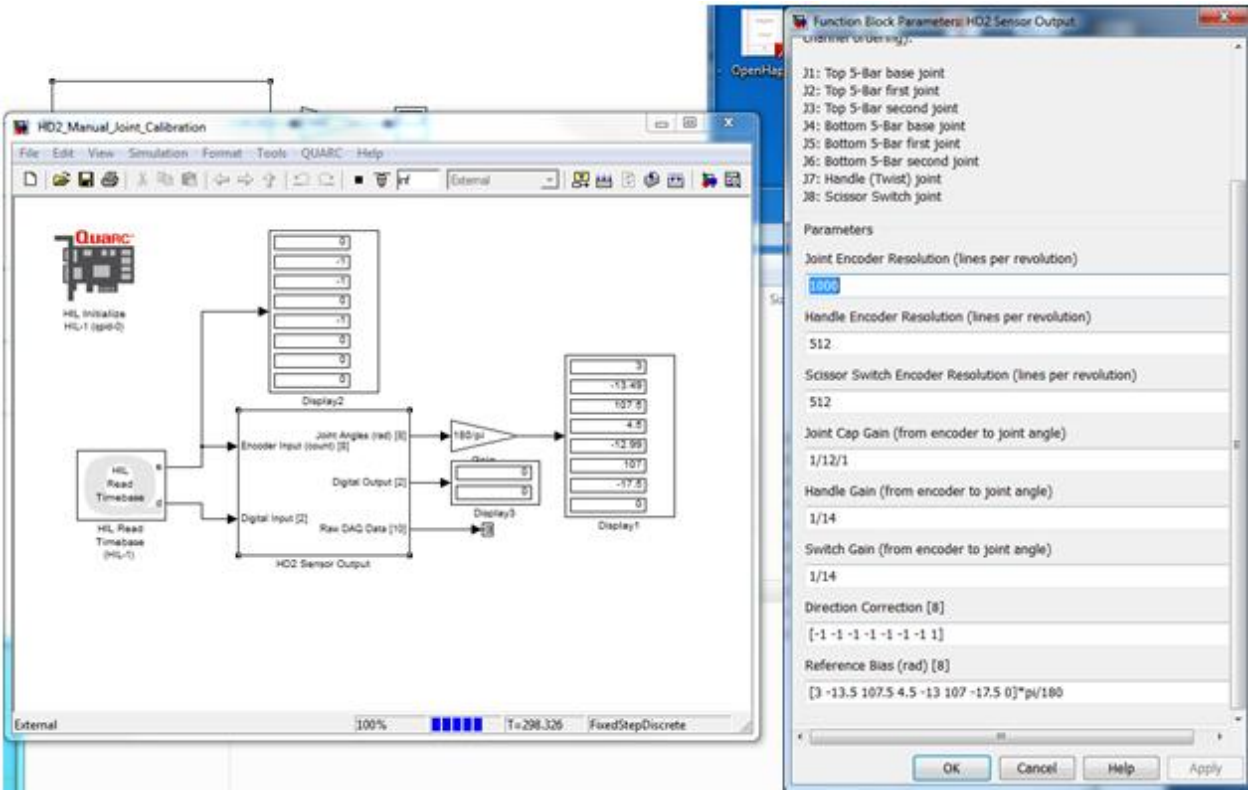

Figure 3-1: HD^2 manual joint calibration block

3.2 Motor Bias Correction

A small force and torque was observed and felt in the handle even when the commanded torques and forces were zero. Since the goal of the project was to accurately as possible simulate surgical forces, it was decided that this problem should be addressed.

This problem was corrected by adding very small joint torque biases into the motor control signal. These corrective biases were selected by incrementing the signal by minute amounts until the force was no longer felt in the end effector.

This correction was applied to both the original and updated driver blocks. Figure 3-2 shows the bias block which was added to the updated drivers.

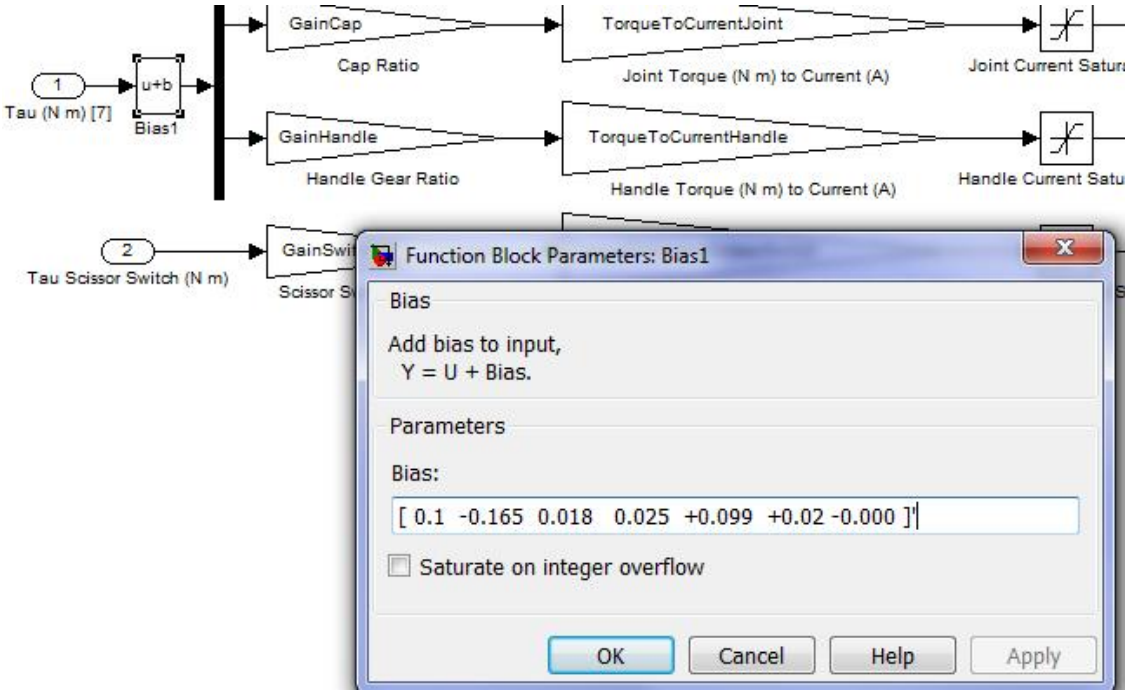

Figure 3-2: Motor bias correction

It is speculated that this problem is most likely the result of an internal electrical flaw as it indicates that when the commanded voltage is zero, the voltage output from the amplifying circuitry was not. There might be a floating ground or floating zero voltage error. It could be that the non-zero voltage might be the result of electrical noise originating from other circuitry within the HD^2. Since the capstan wire system multiplies the output torque by 12, even a relatively negligible voltage signal may result in an observable output.

3.3 Changing the Haptic Interaction Point

The default system allows the user to experience a torque and force at the center of the end-effector wand. If we wish to produce a force, *F*, at a distance from the handle center, *t*, then we must have a force, *F* at the center of the handle plus a pure torque. This required torque would have to be created via force couples and would be a function of wand orientation, the force, *F* and the distance, *t*. Figure 3-3 shows the vectors involved in this formulation.

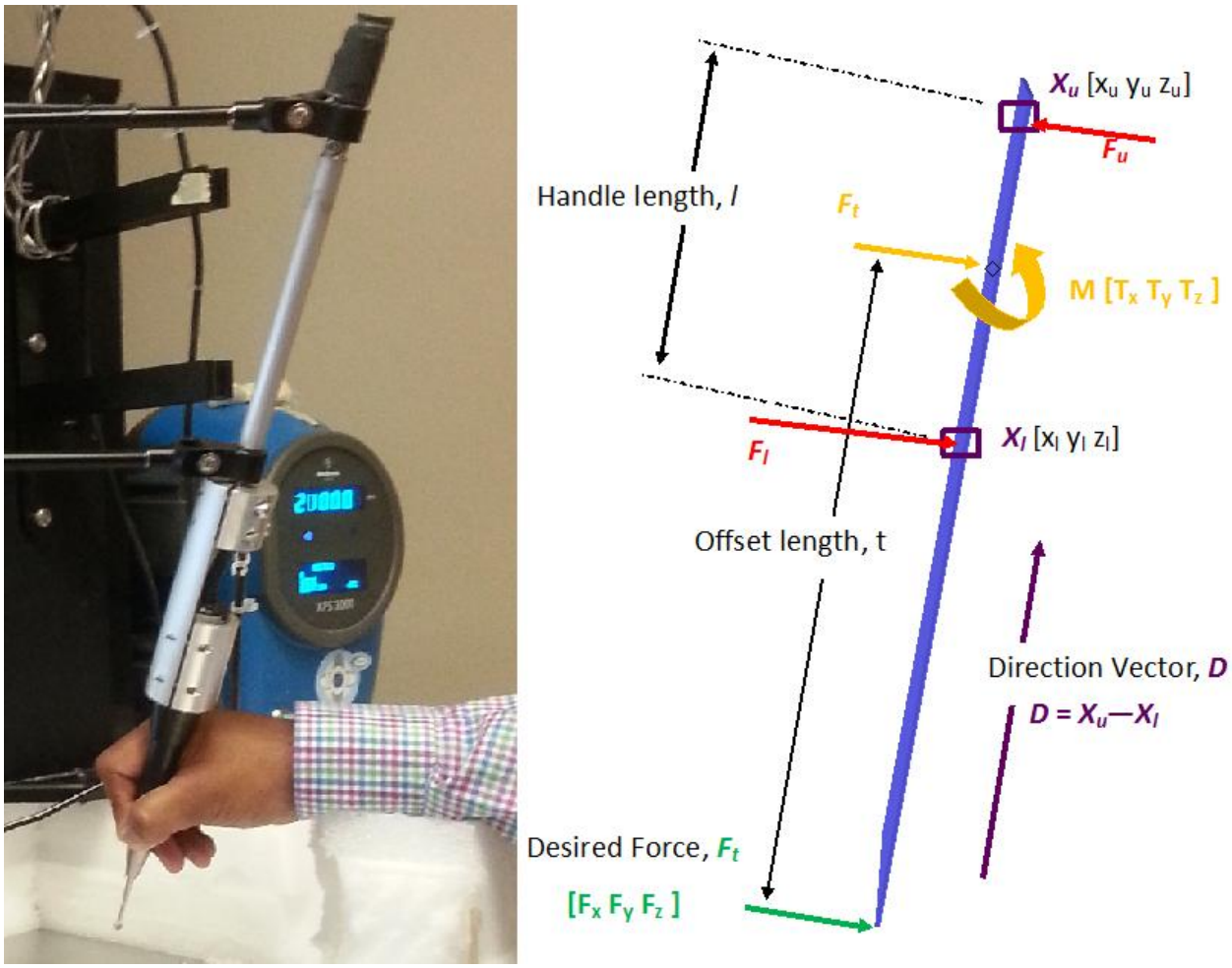

Figure 3-3: Changing the haptic interaction point of the HD^2

Decomposing the desired force and required couple torque between the upper and lower joints of the end-effector gives the following relationship:

The necessary force at the upper and lower joints is equal to the sum of half of the desired force plus the required couple force:

$$\overrightarrow{F_u} = \frac{\vec{F}}{2} + \overrightarrow{F_{UT}} \tag{3-1}$$

$$\overrightarrow{F_l} = \frac{\vec{F}}{2} + \overrightarrow{F_{LT}} \tag{3-2}$$

The required moment is:

$$\vec{T} = \left(\frac{-t}{l}\right)\vec{D} \times \vec{F} \tag{3-3}$$

Where, $\vec{D}$ is the vector:

$$\vec{D} = X_u - X_l \tag{3-4}$$

And where X_u and X_l are the upper and lower handle positions respectively.

For a given moment, the required couple forces, F_{UT} and F_{LT} are:

$$\overrightarrow{F_{UT}} = \left(\vec{T} \times \vec{D}\right)\frac{1}{|D|^2} = \frac{1}{l^2}\left(\vec{T} \times \vec{D}\right) \tag{3-5}$$

$$\overrightarrow{F_{LT}} = -\overrightarrow{F_{UT}} \tag{3-6}$$

Substituting Equation (3-3) into (3-5) gives:

$$\overrightarrow{F_{UT}} = \frac{1}{l^2}\left(\left(\frac{-t}{l}\vec{D} \times \vec{F}\right) \times \vec{D}\right) \tag{3-7}$$

Applying the triple cross product identity gives:

$$\overrightarrow{F_{UT}} = \frac{t}{l}\vec{F} - \frac{t}{l^3}(D \cdot F)\vec{D} \tag{3-8}$$

Therefore, to simulate a given force, F a distance, t away from the center, the required force at the upper and lower joints of the haptic wand is defined as:

$$\overrightarrow{F_u} = \frac{\vec{F}}{2} + \frac{t}{l}\vec{F} - \frac{t}{l^3}(D \cdot F)\vec{D} \tag{3-9}$$

$$\overrightarrow{F_l} = \frac{\vec{F}}{2} - \frac{t}{l}\vec{F} + \frac{t}{l^3}(D \cdot F)\vec{D} \tag{3-10}$$

Since this mathematical formula is general, it was also applied to cancel the weight of the drill gripper.

It is important to note that using Equation (3-8) is far more computationally efficient compared to the method employed in the updated HD^2 driver (see Appendix Section C.3).

3.4 Gripper Weight Compensation

Depending on the orientation of the HD^2 wand, the torque due to the weight of the gripper attachment will vary.

To cancel the perceived weight, we simply simulate a force, F_g at a point that is the center of gravity of the drill gripper. Note that F_g is set as a positive Z force equal to the weight of the gripper. If we know the distance between the wand center and the center of gravity, we can simply apply the formula presented in Section 3.3.

In the actual application, it was found that an additional Z force of around 1N was required at the handle center to keep the system weightless and balanced. The weight of the robotic linkages is not actively compensated for by the HD^2 driver. They are instead approximately mechanically balanced using counter weights attached to the motors [28]. With the addition of the gripper attachment, the system was no longer balanced and thus needed an extra 1N.

3.4.1 Driver Implementation – Original 'HD^2 IO' Block

A function block entitled, "F_upper_from_Ftip" was added to the original 'HD^2 IO' Simulink block. See Figure 3-4. This block calculates the couple forces required for both weight cancellation and for changing the haptic interaction point. This block utilizes Equations (3-6) and (3-8). Using these equations, this block is also capable of negating the weight of the otic drill.

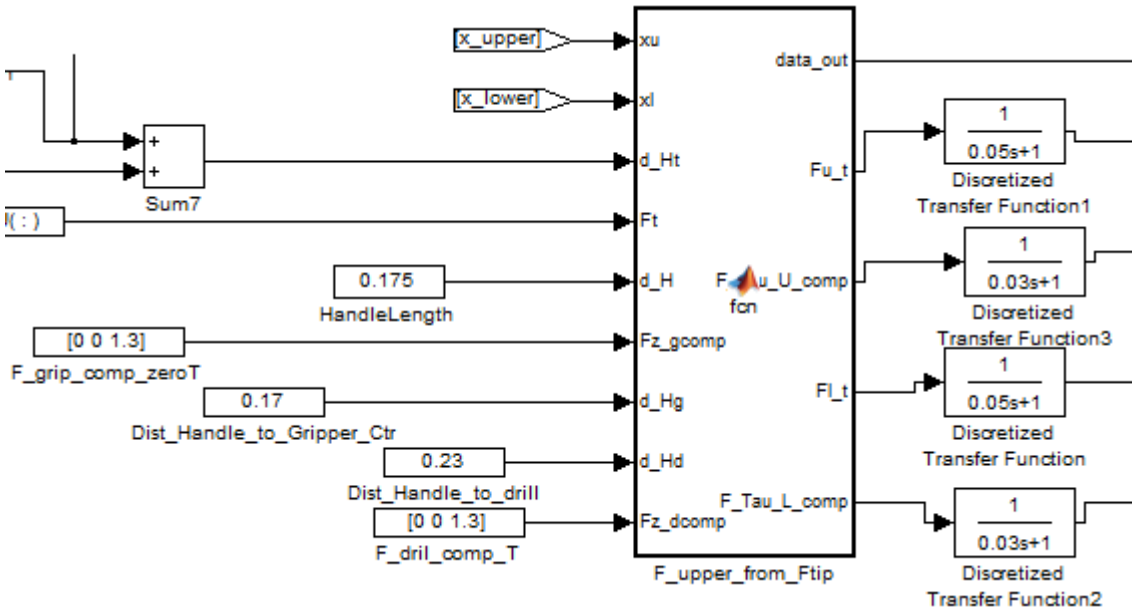

Figure 3-4: Changing haptic interaction point and applying weight cancellation of drill and gripper in the original driver

These generated couple forces are then added to the inputs of "5-bar Linkage Kinematics" block shown in Figure C-22 (Appendix Section C.2.6). In addition, a constant positive Z force must also be added to the handle center.

3.4.2 Driver Implementation – Updated 'HD^2 IO' Block

For simplicity, we wished to use the updated driver and to do so with minimal changes to it. To implement changing of the haptic point and the weight cancellation, while keeping changes superficial, it was decided to simply calculate the required moments and to add the results into the pre-existing moment/torque input signal. The required force for weight compensation was also added to the pre-existing force input signal.

A Simulink block called, 'Reqd_Couple_Moments' was added which calculates the required Moments for weight cancellation and for changing the haptic interaction point. This is shown in Figure 3-5.

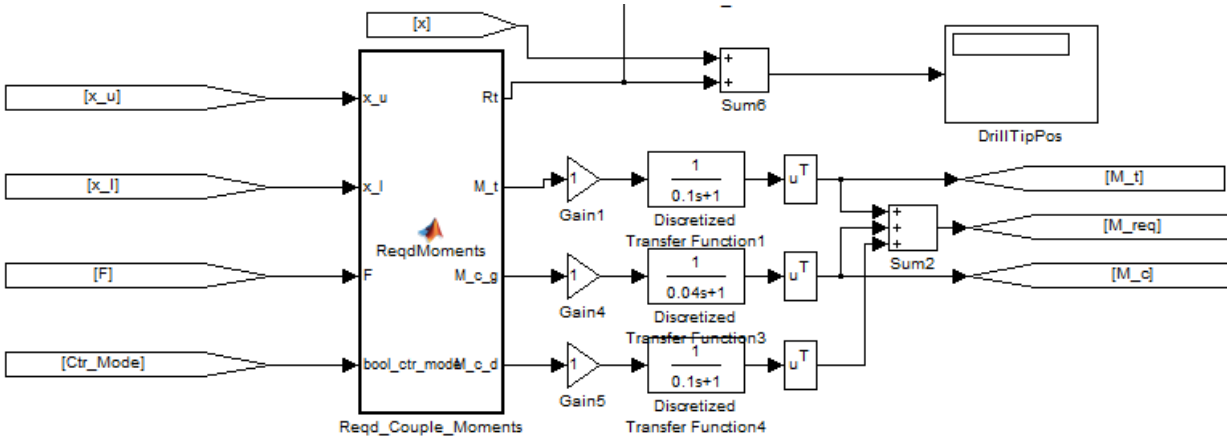

Figure 3-5: Implementing weight cancellation and changing of the haptic point in the updated driver

The inputs for this block are the upper and lower positions of the wand end-effector and the desired haptic force at the drill's tip. The block outputs the required torque for simulating the force at the drill tip as well as the torques required for compensating the weight of the mechanical clamp attachment and the weight of the surgical drill. These torques are added together and become the torque input for the 'HD^2 IO(PY)' block shown in Figure C-26 (Appendix Section C.3).

An additional block named, 'Angle Calculator' was also added to the update driver and is shown in Figure 3-6. This adds the older method (see Appendix Section C.2.5) of calculating the orientation angles to the new driver. This was done for compatibility with legacy applications within the driver and the already written C++ API. The HD^2 C++ API is discussed in Section 3.5.

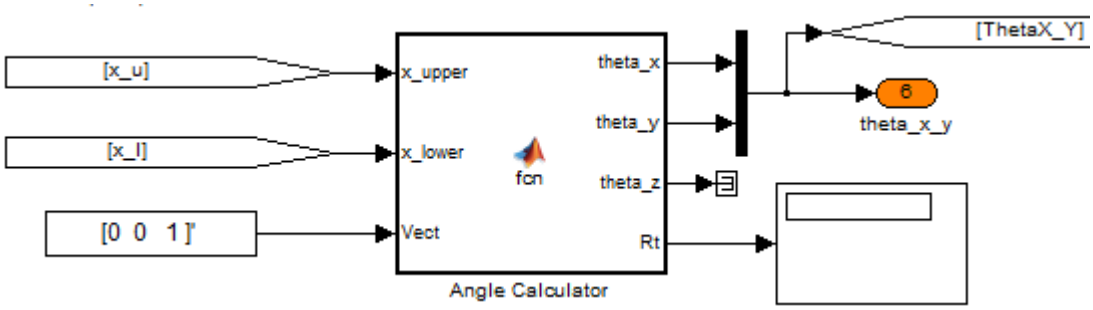

Figure 3-6: Calculating θ_x and θ_y

3.4.3 Weight Cancellation Performance

In several informal tests, carried out by numerous individuals, it was shown that the weight of the attached clamp was negated. Regardless of position and orientation, the haptic handle held its position and orientation. Figure 3-7 shows an example of one successful test in progress.

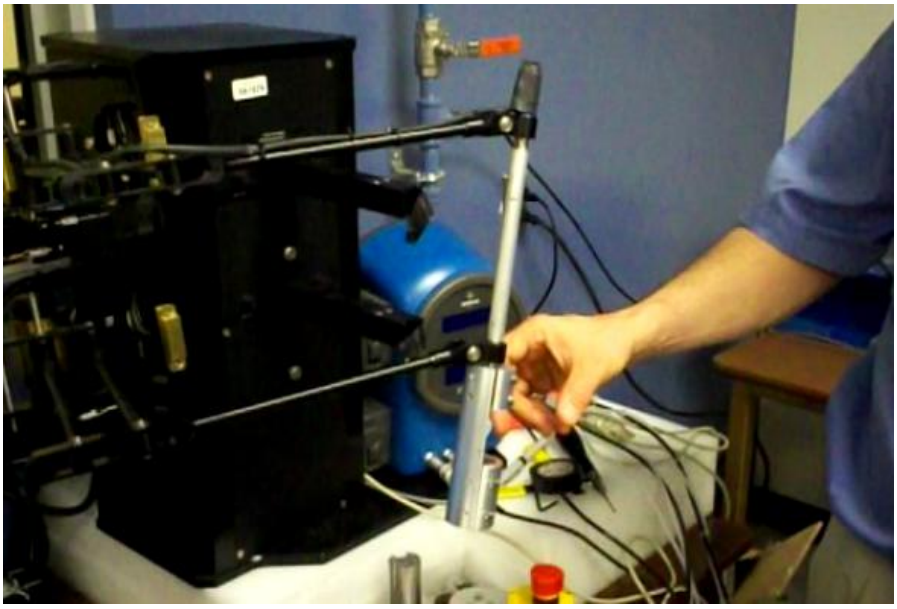

Figure 3-7: Testing the weight cancellation of the drill clamp

The following graphs show weight cancellation in action. The haptic handle was moved to various positions and orientations within the HD^2 workspace and released for a few seconds. Figure 3-8 gives the orientation angles θ_x and θ_y with respect to time. Figure 3-9 give X, Y and Z positions of the end-effector (handle center position).

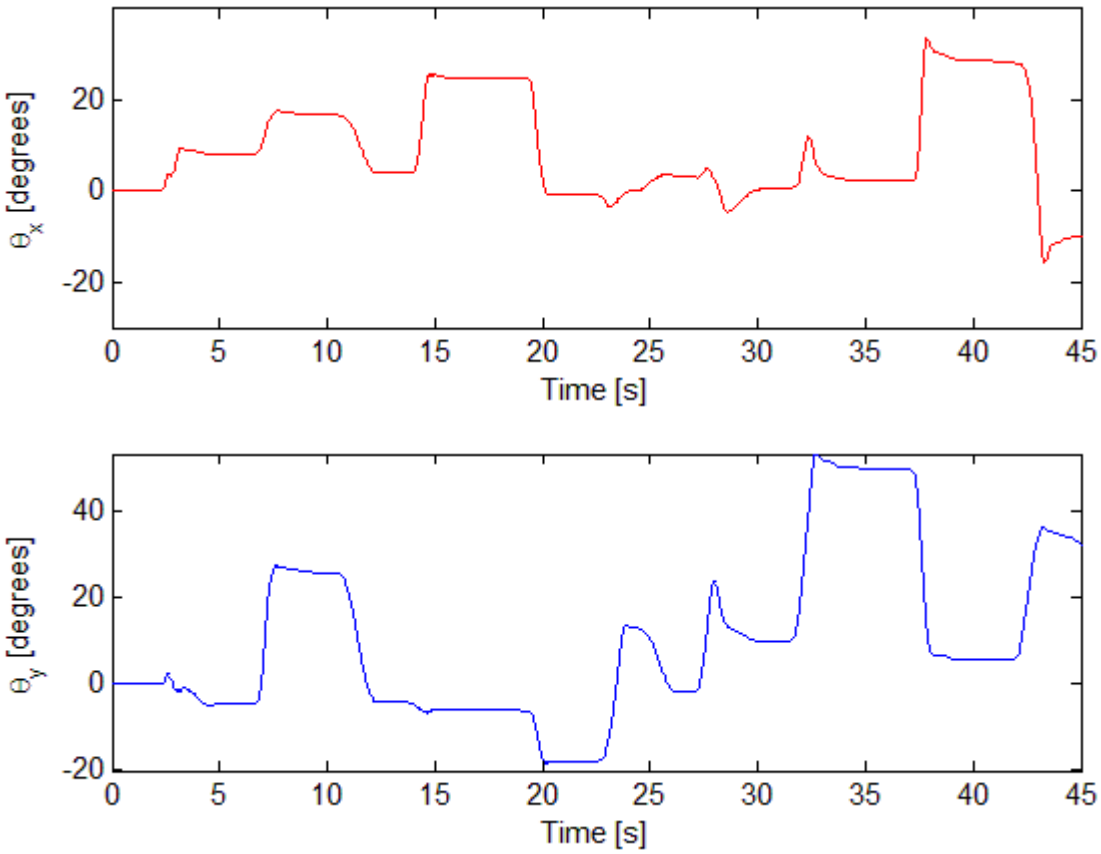

Figure 3-8: Clamp weight compensation – Orientation angles, θ_x, θ_y

All flat portions of the graphed profiles coincide with whenever the user hand releases the end-effector. It can be clearly shown that the weight cancellation performs satisfactorily as both position and orientation is held. The system has behaved nominally and within expectation consistently over a period exceeding one year.

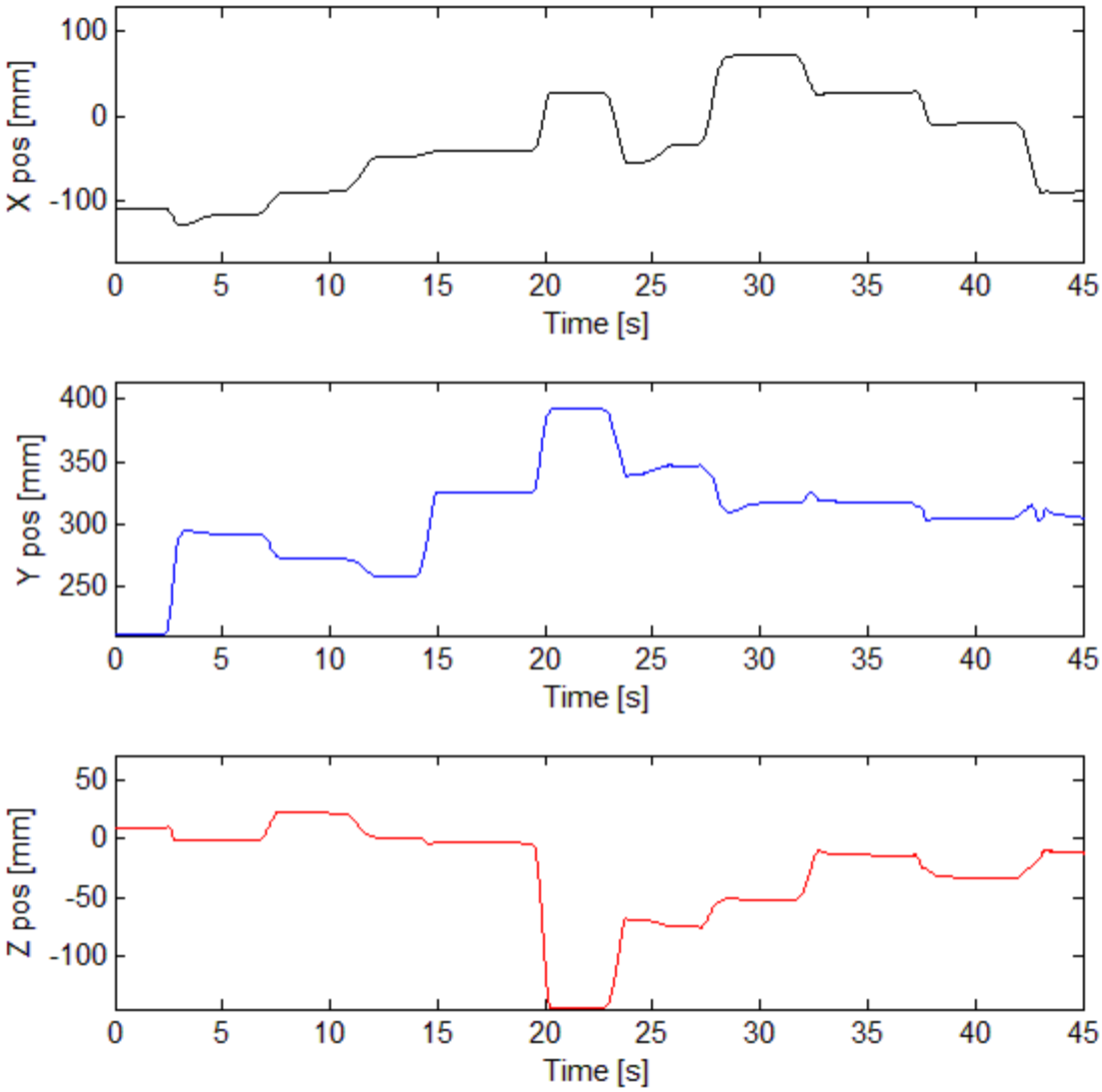

Figure 3-9: Weight compensation of drill clamp attachment– X, Y and Z positions

3.5 Interfacing the HD² System to the Temporal Bone Simulation

A C++ API thread was coded and added to the existing C++ Virtual Simulation. A simplified overview of the software architecture is shown in Figure 3-10 below. When the Simulink file is compiled, an executable is created. This executable communicates in real time with the HD^2 hardware and the C++ HD^2 API thread. The HD^2 C++ thread in turn communicates with the rest of the temporal bone simulation.

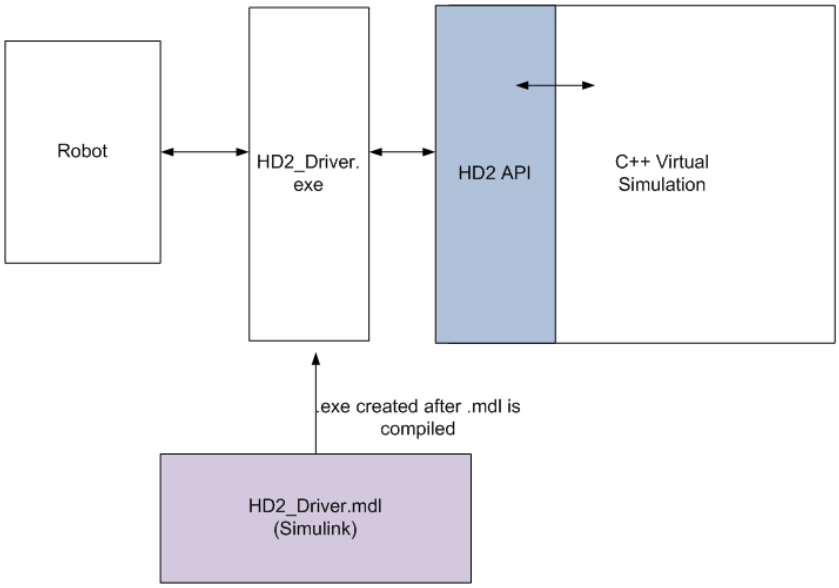

Figure 3-10: Simplified software architecture of the HD^2 and the surgical simulation

This API thread performs the task of being an information interface. Position data flows from the HD^2 hardware towards the Surgical Simulation, while Force Commands flow from the Surgical Simulation to the HD^2 hardware. A copy of HD2_Interface.cpp can be found in Appendix D – HD2 C++ API Code.

Since the Surgical Simulation was initially built around the Phantom Omni, the position and orientation of the HD^2 end effector must undergo a transformation so that the Surgical Simulation understands it. The different co-ordinate systems are shown in Figure 3-11.

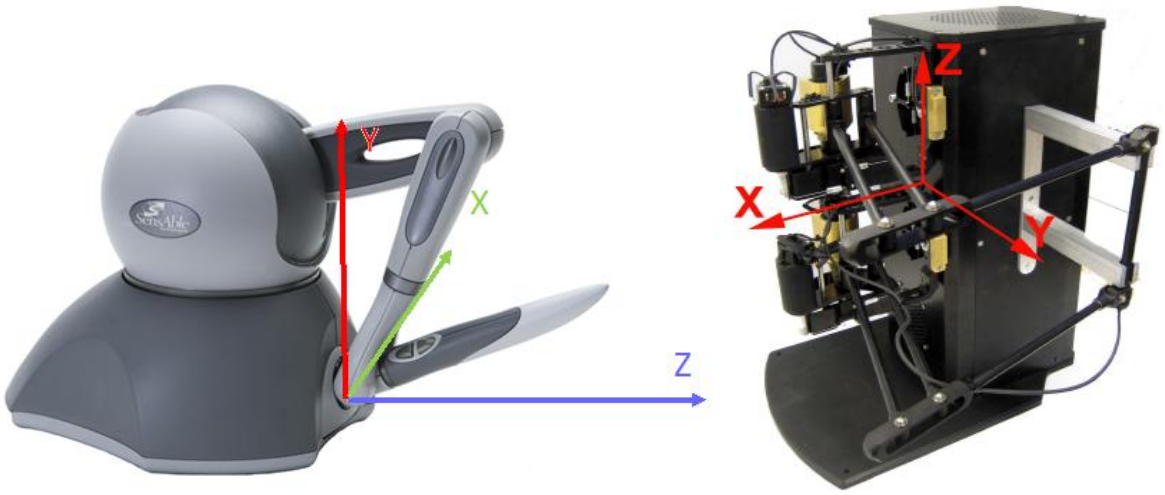

Figure 3-11: Reference frames of the Phantom Omni and HD^2

From Figure 3-11 it is evident that both haptic devices are to be used differently. For the Omni Phantom, the Z-axis is set up horizontally while the HD^2 is vertical. This means that drilling done by the Phantom Omni would be horizontal, while the HD^2 will be vertical. Using the z-axis as a reference point, it can be seen that a 180° rotation about the Z-axis will allow reference frames to match.

It is also important to note that the Phantom Omni workspace is close to its origin whereas, the HD^2 is not. Therefore an offset must be added to the X, Y and Z co-ordinates for the HD^2 to be in the same workspace within the temporal bone simulation.

Finally, the reference frame within the simulation is left-handed meaning that the z-axis is inversed. The software was coded to accept the transpose of the orientation matrix. Furthermore, X, Y and Z positions are all offset and then inversed.

Equations (3-11) to (3-14) show how the orientation matrix was derived. Equation (3-14) is implemented in the C++ HD^2 API thread. Recall from Appendix Section 2.5, the orientation matrix of the HD^2 handle is:

$$M_{HD^2} = Rot_X\, Rot_Y = \begin{bmatrix} c_{\theta y} & 0 & s_{\theta y} \\ s_{\theta x}s_{\theta y} & c_{\theta x} & -s_{\theta x}c_{\theta y} \\ -c_{\theta x}s_{\theta y} & s_{\theta x} & c_{\theta x}c_{\theta y} \end{bmatrix} \tag{3-11}$$

Converting this to the Phantom Omni reference frame, we get:

$$M_P = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta y} & 0 & s_{\theta y} \\ s_{\theta x} s_{\theta y} & c_{\theta x} & -s_{\theta x} c_{\theta y} \\ -c_{\theta x} s_{\theta y} & s_{\theta x} & c_{\theta x} c_{\theta y} \end{bmatrix} \tag{3-12}$$

$$M_P = \begin{bmatrix} -c_{\theta y} & 0 & -s_{\theta y} \\ -s_{\theta x} s_{\theta y} & -c_{\theta x} & s_{\theta x} c_{\theta y} \\ -c_{\theta x} s_{\theta y} & s_{\theta x} & c_{\theta x} c_{\theta y} \end{bmatrix} \tag{3-13}$$

Converting the matrix into the format required by the existing code we get:

$$M_{Sim} = {M_p}^T = \begin{bmatrix} -c_{\theta y} & -s_{\theta x} s_{\theta y} & -c_{\theta x} s_{\theta y} \\ 0 & -c_{\theta x} & s_{\theta x} \\ -s_{\theta y} & s_{\theta x} c_{\theta y} & c_{\theta x} c_{\theta y} \end{bmatrix} \tag{3-14}$$

This matrix is used in various downstream calculations within the C++ simulation including calculation of the haptic drill tip and in the rendering of the virtual drill.

At this point, the HD^2 was successfully integrated into the temporal bone simulation. However, haptic chatter was observed that made the system behave dangerously unstable. Applying a moving average in the form of a low-pass filter rectified the problematic condition. This problem is explained in detail in Chapter 4.

Once the problem with haptic chatter was resolved, the next step was to integrate a Medtronic surgical drill into the temporal bone simulation.

3.6 Integrating the Drill State into the Virtual Simulation

For a realistic simulation, it was crucial for the software to know the drill state in real time. The XPS 3000 drill system (Medtronic, Dublin, Ireland) is shown in Figure 3-12 below. Initial attempts to read state variables from the COM1 serial port on the back of the device proved futile.

Figure 3-12: Medtronic XPS 3000 unit and its COM port on its back

Therefore another means of reading the drill state had to be found. The drill pedal box was opened up and examined. It was discovered that a sliding potentiometer created a voltage signal with a range between 0V – 5V as the right pedal is stepped on. The signal being tapped was the blue and black wire is shown (circled in yellow) in Figure 3-13.

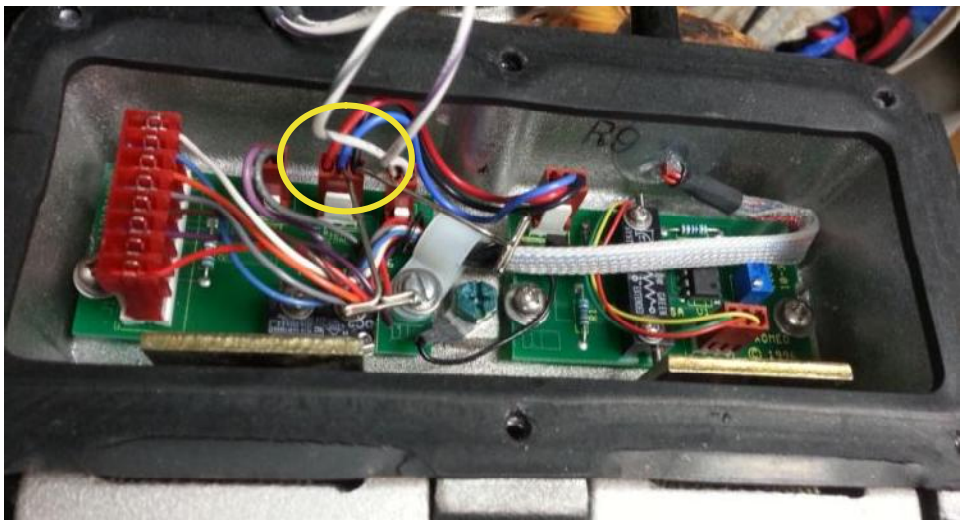

Figure 3-13: Inside the Medtronic step pedal box

Therefore, if this signal could be read into the HD^2 using either the auxiliary digital or analog input ports, the problem would be addressed. Figure 3-14 shows the locations of the analog and digital ports on the HD^2. Figure 3-15 shows the electrical schematic of both the digital and analog port.

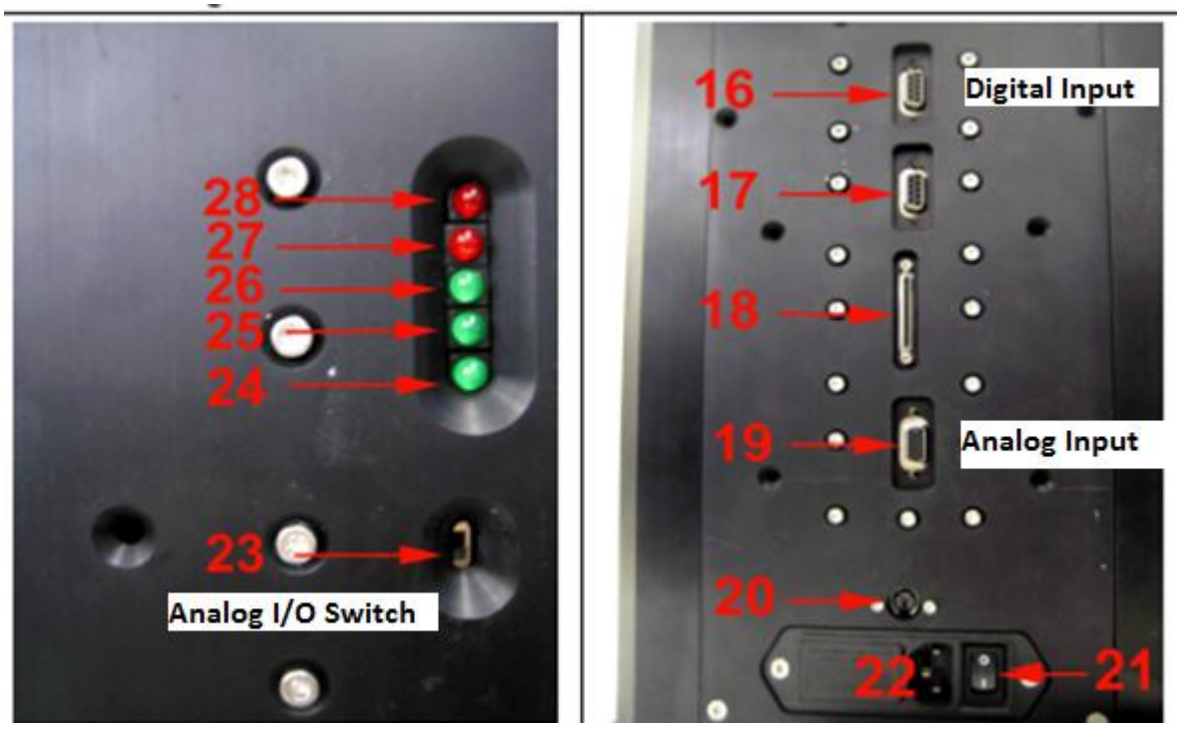

Figure 3-14: Back of HD^2 showing auxiliary analog and digital inputs [28]

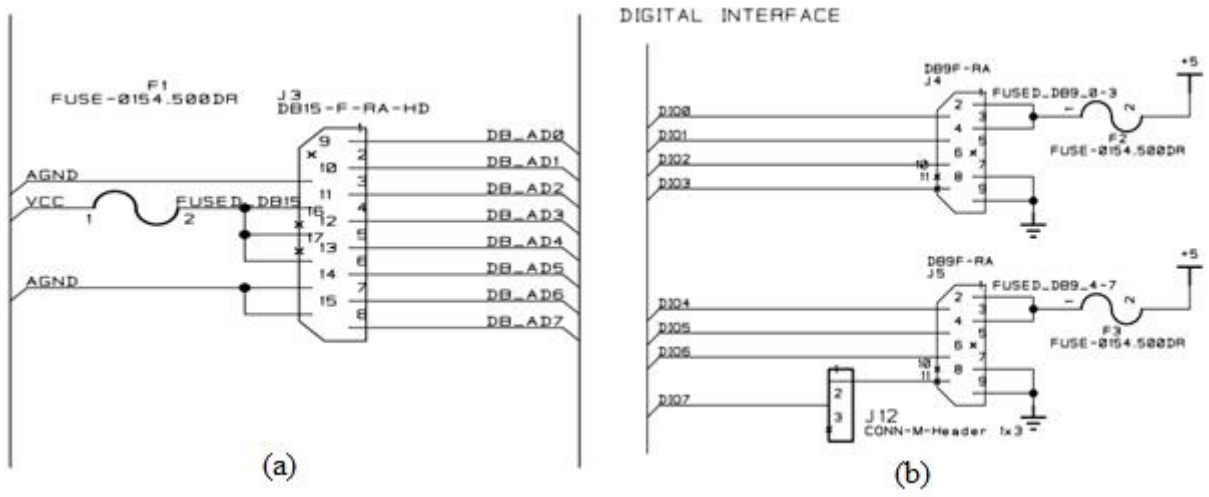

Figure 3-15: Schematic of (a) auxiliary analog input and (b) auxiliary digital input [32]

Originally, the step pedal was wired to the Digital I/O port. However it was decided to use the analog port due to its future potential. Currently, the temporal bone surgical simulation treats the drill state as a binary; either on or off. With an analog signal, it may be possible to simulate (for both force and animation) the effect of having the drill burr spin at variable speeds. This potential future work would definitely improve immersion and realism of the temporal bone haptic simulation.

It is important to note that the use of the analog I/O port required the flipping of the I/O switch which is labelled in Figure 3-14. The trade-off of this function is that the motor currents are no longer measured and reported to the HD^2 drivers [28, 32]. This is not a grave concern as it is shown in Section 4 that the motor amps operate well within the continuous limits during the virtual reality haptic simulation.

To allow the drill state to be passed into the C++ simulation, modifications were made to both the C++ code and the Simulink drivers.

3.7 Chapter Summary

This chapter presents the technical work required to prepare and integrate a standard HD^2 into a pre-existing temporal bone haptic simulation.

Initial angles in the updated drivers were calibrated and set. A non-zero motor bias was observed and corrected. A general mathematical formula was derived to create a force at any point along the line of the haptic handle. This formula was used to change the haptic interaction point from the center of the HD^2 wand to the tip of an attached drill. The formula was also applied to cancel the weight of an attached mechanical clamp. Numerous tests reveal that weight negation is performed satisfactorily. A C++ API was coded which allowed the HD^2 system to seamlessly integrate with an existing temporal bone simulation. The Medtronic surgical drill was fully integrated into the HD^2 temporal bone simulation system.

4 Rectifying Unstable Haptic Chatter

After the modified HD^2 was interfaced with the existing virtual reality haptic simulation, unstable and undesirable haptic chatter was observed. Haptic chatter is defined here as severe vibration whenever the drill tip touches or enters the surface of the virtual bone. This problem was so extreme that running the haptic simulation was not possible. This chapter describes the source of the problem and how it was rectified.

4.1 Observing Unstable Haptic Chatter

To find the root cause(s) of this haptic chatter, the Simulink driver was modified to output internal variables. The data that was recorded and scrutinized included:

- X, Y and Z position of the drill tip or the handle center
- Commanded force (F_x, F_y, F_z) from the temporal bone haptic simulation
- Upper forces due to commanded tip force (FuT_x, FuT_y,FuT_z)
- Upper forces due to gripper weight compensation (Fu_{cx}, Fu_{cy}, Fu_{cz})
- Total upper force (Fu_x, Fu_y, Fu_z)
- Lower forces due to commanded tip force (FlT_x, FlT_y,FlT_z)
- Lower forces due to gripper weight compensation (Fl_{cx}, Fl_{cy}, Fl_{cz})
- Total lower force (Fl_x, Fl_y, Fl_z)
- Measured motor current for the 6 main motors

To clarify, upper and lower forces refer to the forces that are generated at the upper and lower ends of the HD^2 handle. How these forces are calculated was covered in Sections 3.3 and 3.4. The position data and force data is given with respect to the HD^2 reference frame. (See Figure C-19 for reference.)

The following three figures present data captured during a case of unstable haptic chatter. Figure 4-1 gives position data and the commanded forces coming from the temporal bone simulation. The position data refers to the position of the drill tip. The commanded forces are given in X, Y

and Z components with respect to the global HD^2 reference frame and are to be simulated at the drill tip.

Figure 4-2 compares the upper and lower forces with respect to the commanded forces at the drill tip.

Figure 4-3 gives the current for the 6 main motors attached to the HD^2. The handle motor, 'Motor 7', is not used and is thus not reported.

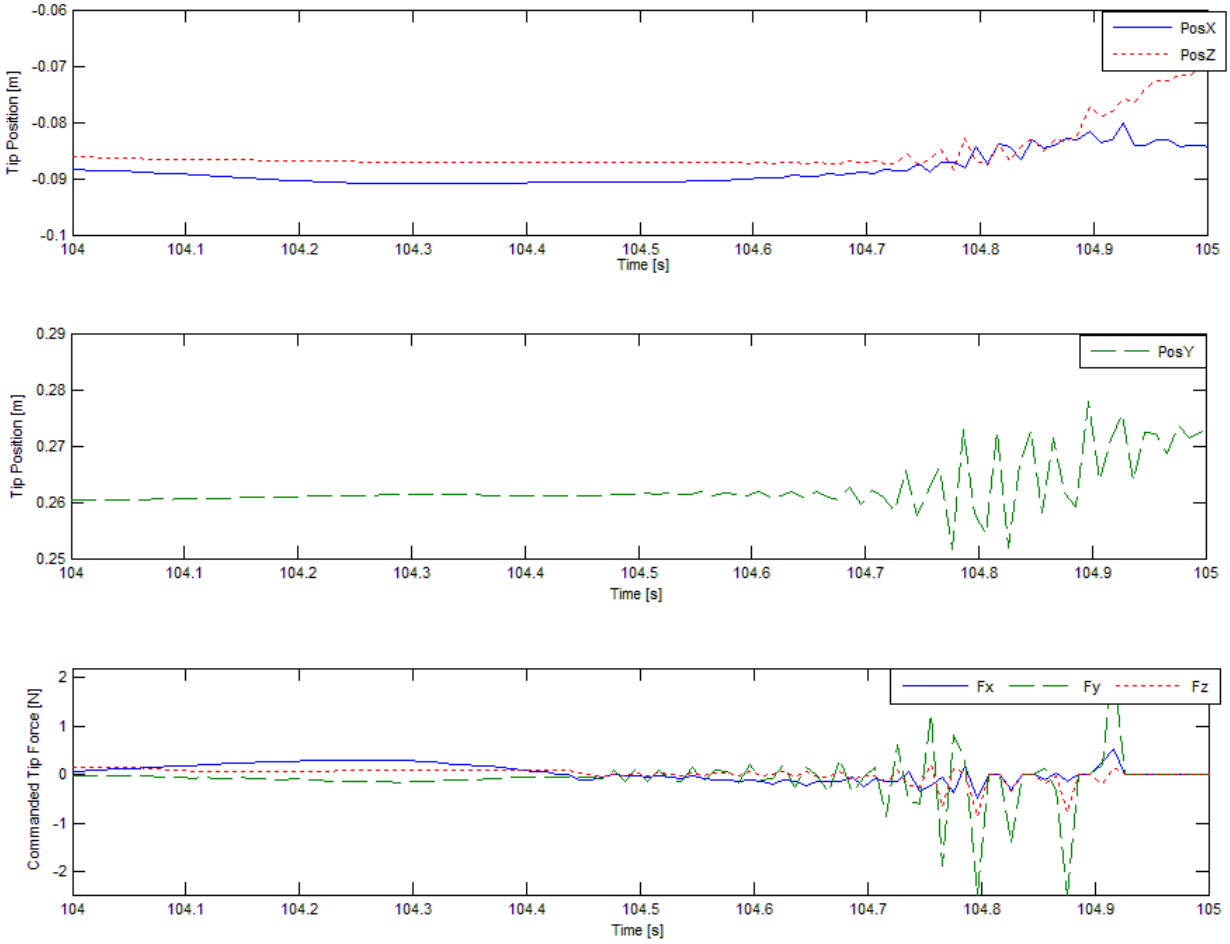

Figure 4-1: Instability case – position and commanded simulation forces

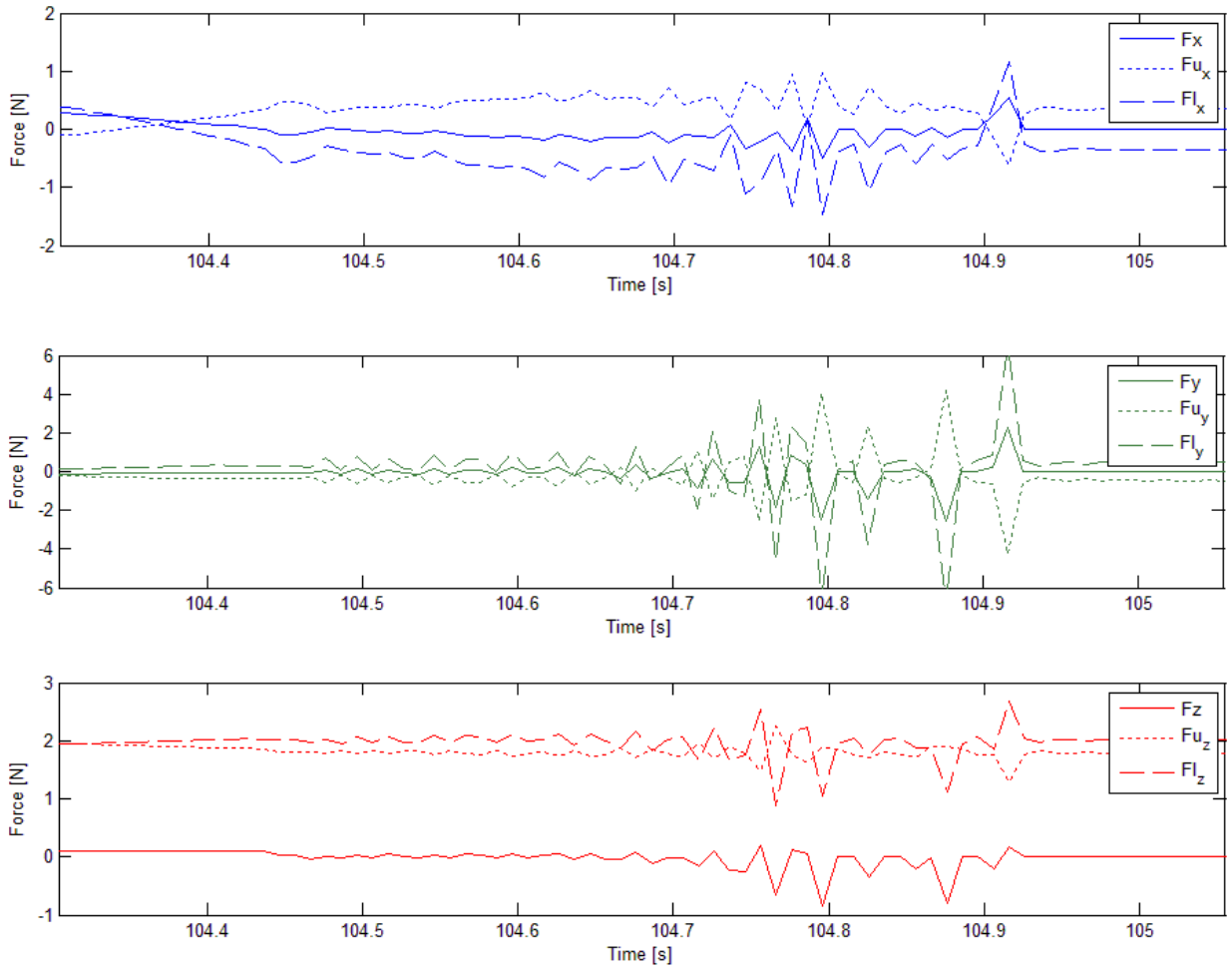

Figure 4-2: Instability case – upper and lower forces required in simulating commanded tip force

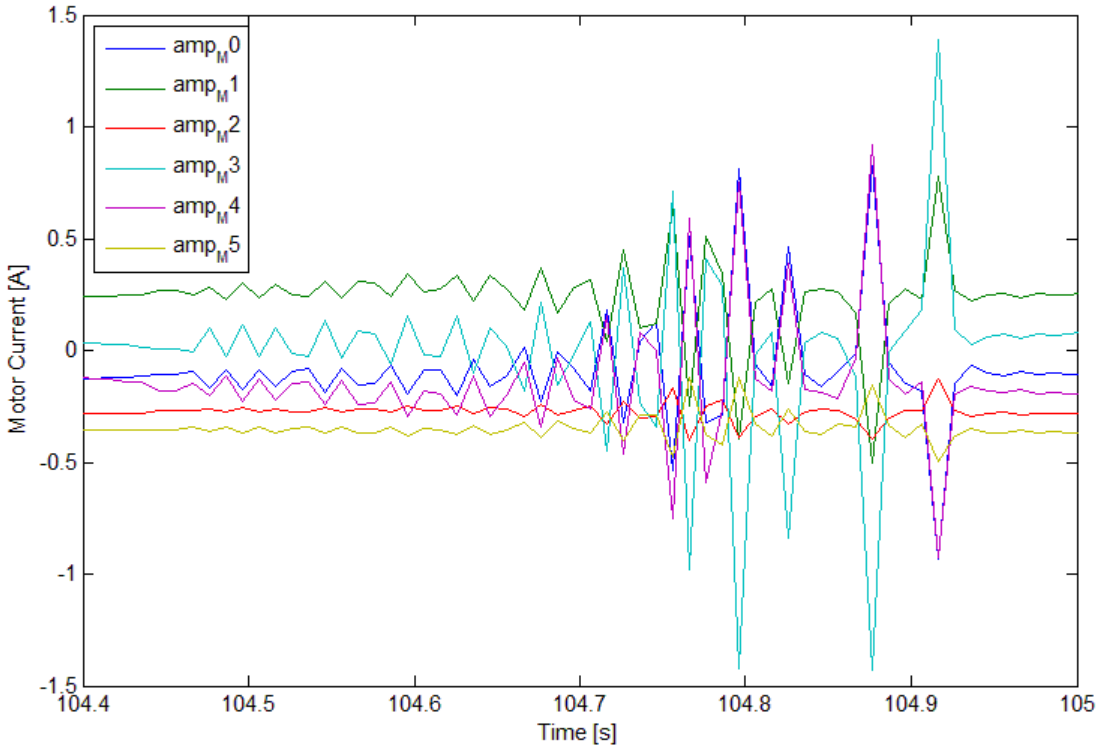

Figure 4-3: Instability case – motor currents

As can be seen in the above three figures, unstable haptic behaviour sets in very quickly; within 0.20 seconds of coming into contact with the virtual bone. Additionally, it can be seen that there is roughly 4 – 6 oscillations every 0.1 seconds. This puts the haptic chatter frequency in the range of 40 – 60 Hz. However, from these figures it was difficult to determine the source of the haptic chatter.

It can be concluded however that the instability is not due to motor saturation. The motors are all well within the continuous maximum limits of 3A and the peak limit of 5A [28]. (The full specifications of the HD^2 can be found in Appendix A – HD2 Detailed Technical Specifications.)

It is also important to note that the instability was observed well within the defined haptic workspace of the device. Therefore, undesirable haptic chatter cannot be attributed to workspace constraints.

4.2 Further Testing of the Modified HD² System

Since the source of haptic instability was not yet determined, further testing was carried out. From these tests the underlying problem was identified.

4.2.1 Running the Simulation with Forces Generated at the Handle Center

The temporal bone simulation was run with the force being generated at the default position, i.e. the center of the default HD^2 handle. It was found that the simulation behaved similarly to the Phantom Omni. Repeated trials revealed that the simulation performed stably and without issue.

Figure 4-4 shows the captured data for of one trial and shows the position and the commanded force in X, Y and Z.

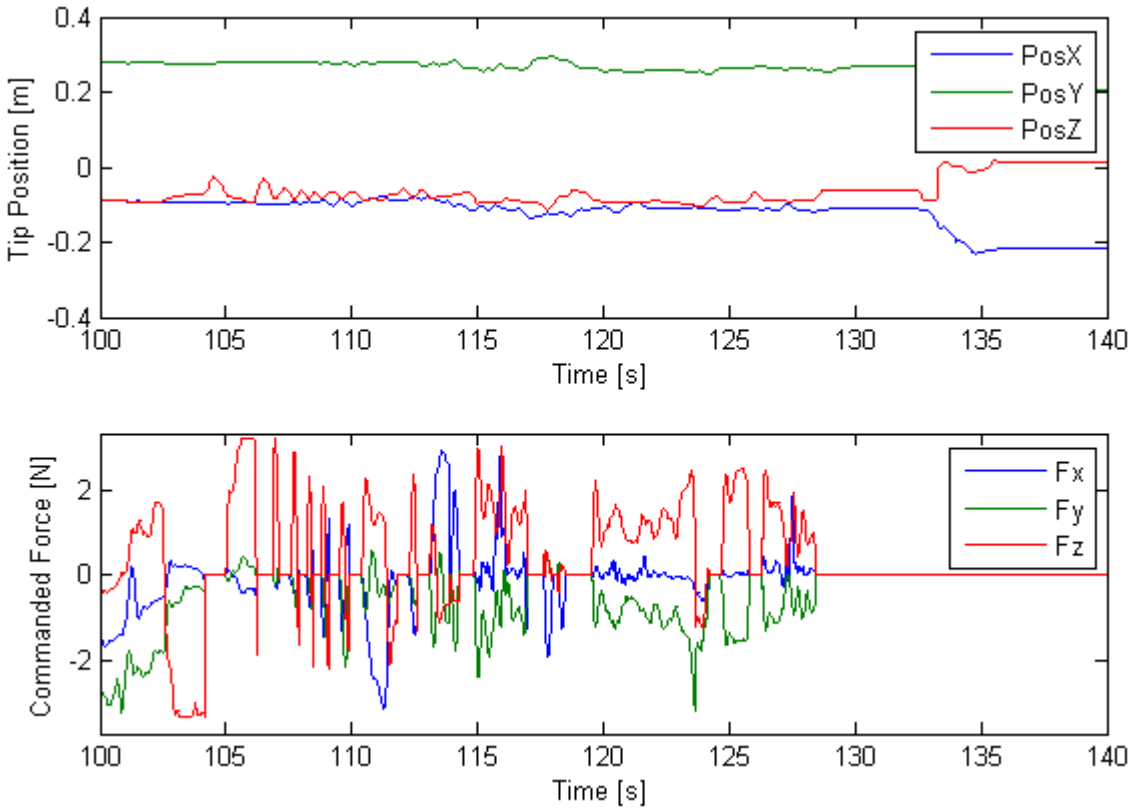

Figure 4-4: Stable simulation behavior observed when forces are generated at handle center

The temporal bone simulation was repeated under two more conditions: first with the gripper weight compensation activated and then with the haptic interaction point changed from the handle center to the drill tip. The weight compensation algorithm had no impact on the stability of the system.

However, re-enabling the haptic interaction point transformation algorithm reintroduced severe haptic chatter. This behavior was puzzling as the underlying math was sound. The problem therefore must be dynamic or situational.

4.2.2 The Effect of Square Wave Impulses on the Modified HD² System

It was decided to input square wave impulses of both force and torque for both the standard manipulandum configuration and for the modified attached drill configuration. Both configurations were subjected to square waves of various amplitudes and frequencies of F_x, F_y, F_z, T_x, T_y and T_z.

For the standard manipulandum case, nothing out of the ordinary was observed. This data is therefore not reported in this Thesis.

For the attached drill case, haptic chatter seen with the simulation was not observed. However, it provided a clear picture into the issue. Figure 4-5 below shows one case of a square wave force of +/- 1N in the Y axis simulated at the drill tip.

For clarity, X and Z forces are excluded from the plots. The plots give the upper and lower forces required to simulate the commanded 1N force at the drill tip. The first plot gives the upper and lower forces required assuming that the drill clamp attachment is weightless. The second plot shows the upper and lower forces required.

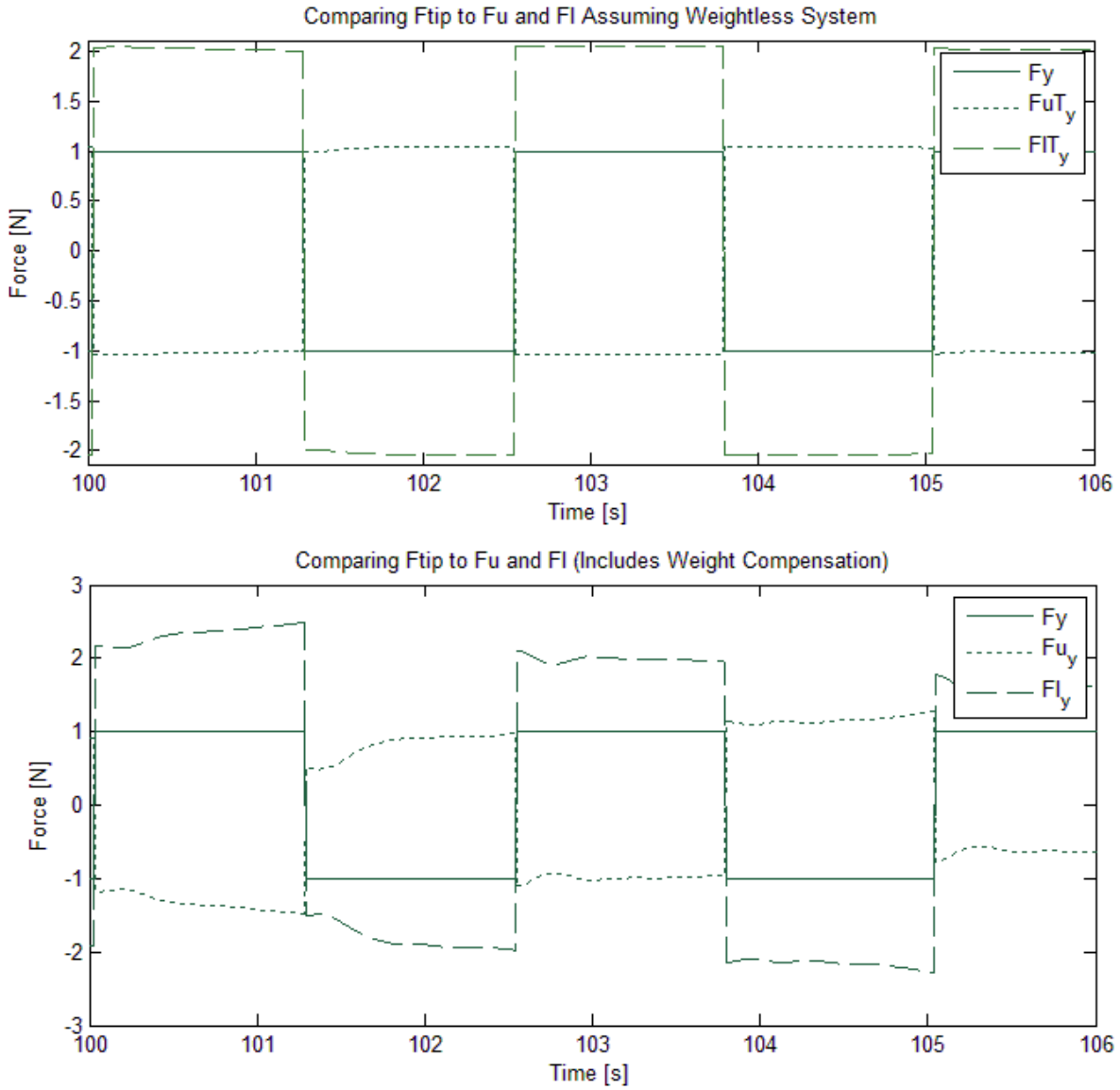

Figure 4-5: Square waves of 1N in the Y axis was created at the drill tip

These square wave tests revealed that simulating a force at the drill tip created an instantaneous torque about the default handle which significantly impacted the orientation of the drill. This jerking torque and the change of orientation are the contributors to the observed haptic chatter.

4.3 Explaining the Haptic Chatter

As shown in both Figure 4-5 and Figure 4-6, relatively large opposing forces are required at the upper and lower joints to simulate small forces at the drill tip. The upper and lower joint forces are calculated assuming the haptic handle and the drill clamp attachment is a rigid body in static equilibrium. How the couple forces are calculated is covered in Section 3.3. Since the driver runs at 1000Hz, this static calculation is performed every millisecond.

When the drill tip comes into contact with the virtual bone, relatively small X and Y drill tip forces result in significantly larger X or Y forces at the upper and lower joint handles. These large opposing forces result in the handle orientation being jerked into a new orientation.

If the drill tip is at the surface of the virtual bone, this sudden forced change in orientation often causes the drill tip to be pushed into a new area of virtual bone. See Figure 4-6. Due to the rough surface of the bone, valleys are common. Therefore, the drill tip will be pushed out of the right valley and into the left. This will result with an even larger reaction force in the opposite direction at the drill tip. This in turn results in larger opposing joint handle forces which will force the orientation in the opposite direction, ultimately pushing the drill tip into virtual bone once more.

This behaviour can best be described as a type of 'positional-dependent, self-perpetuating, haptic chatter'. As stated in Section 4.2.1 this unstable behavior is not observed when the force is generated at the center of the haptic wand. This chattering behavior is also not observed when the Phantom Omni haptic device interfaces with the temporal bone haptic simulation. The haptic chatter is a by-product of the drill gripper design which places the haptic interaction point too far below the default haptic handle.

The observed haptic instability is due to large instantaneous couple forces at the upper and lower joints of the default haptic handle. These couple forces are required to simulate relatively small forces at the drill tip. Ideally, the drill clamp attachment and haptic handle should be redesigned such that the drill is either between the upper and lower joints or parallel to it so that the drill tip would be just below the lower end joint. This redesign would eliminate the need for large couple forces, which would in turn, eliminate the haptic chattering. However, this recommendation was deferred to future work.

An alternate solution which utilized the existing drill attachment was implemented. To correct for the unstable haptic chatter, a moving average in the form of a first order low-pass filter was applied to required force and couple forces.

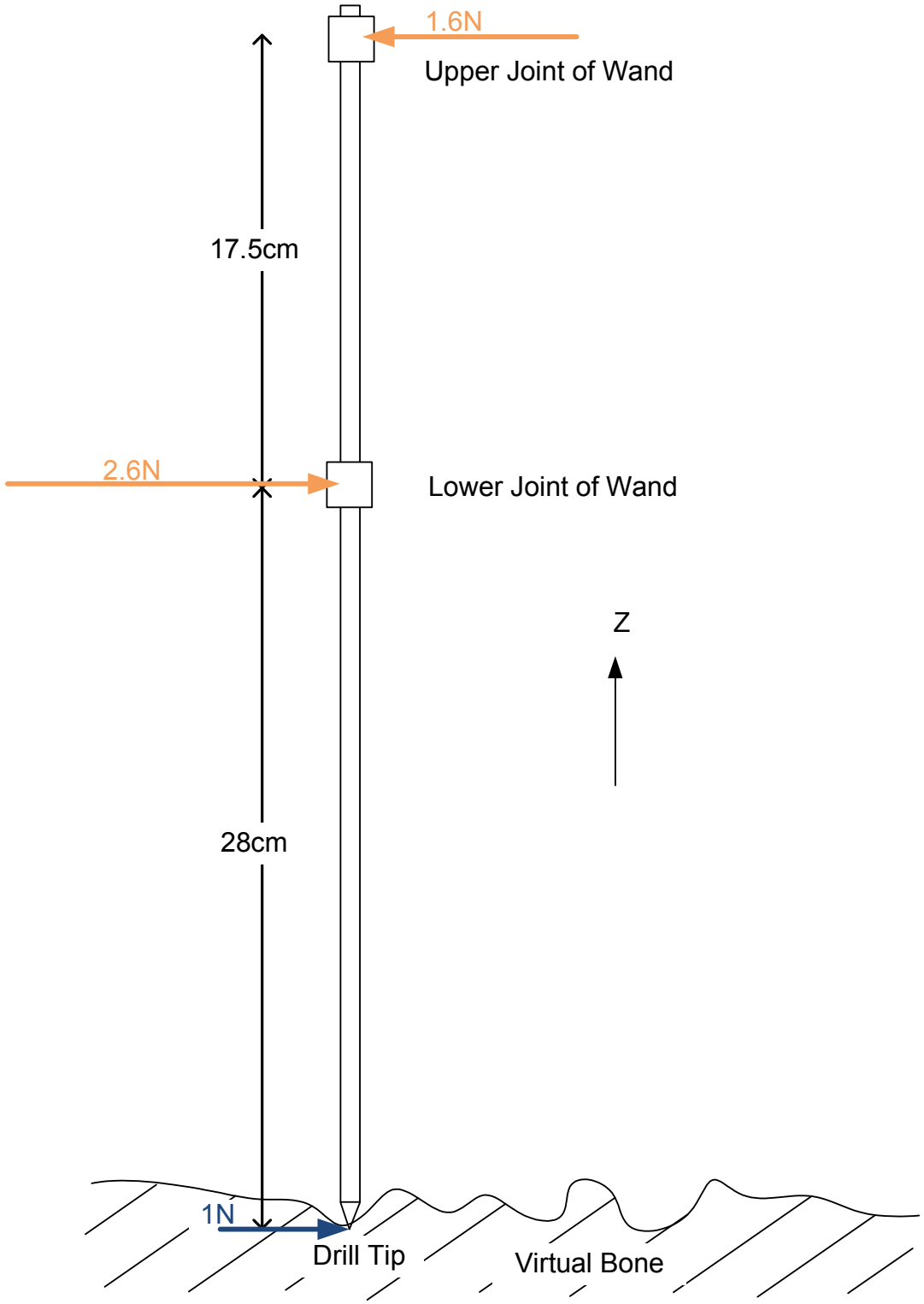

Figure 4-6: Simplified example of upper and lower forces required to simulate a force at the drill tip

4.4 Applying a Low-Pass Filter to Correct Haptic Chatter

Originally the intent was to apply a moving average to reduce and smooth the observed spikes of the produced couple forces. Since a simple moving average block was no longer part of the Simulink library, a low-pass filter was used to perform the same function.

A low-pass filter allows frequencies less than the cut-off frequency to pass whilst attenuating all frequencies that are higher. A first-order low-pass filter is defined as [38, 39]:

$$\frac{Output}{Input} = K\frac{1}{1+s\tau} \tag{4-1}$$

Where, τ is the time filter time constant in seconds [s] and K is the filter pass-band gain. For our application, K must be equal to 1.

The cut-off frequency [Hz] is given by [38, 39]:

$$f_c = \frac{1}{2\pi\tau} \tag{4-2}$$

A first order low pass filter was utilized because they are easy to implement, and for a given cut-off frequency, the introduced time-response lag is smaller compared to higher order filters.

A low pass filter was added to the HD^2 driver to condition the couple force signals. It was found that a τ value of 0.05s eliminated all observed haptic chattering. This value coincides with a corner frequency of 3.18Hz. A bode plot of this low pass filter is shown in Figure 4-7. From the previous section, it was shown that the haptic chapter occurs within a frequency range of about 40 – 60 Hz. As illustrated on Figure 4-7, the gain at 50Hz is very close to zero (0.0645).

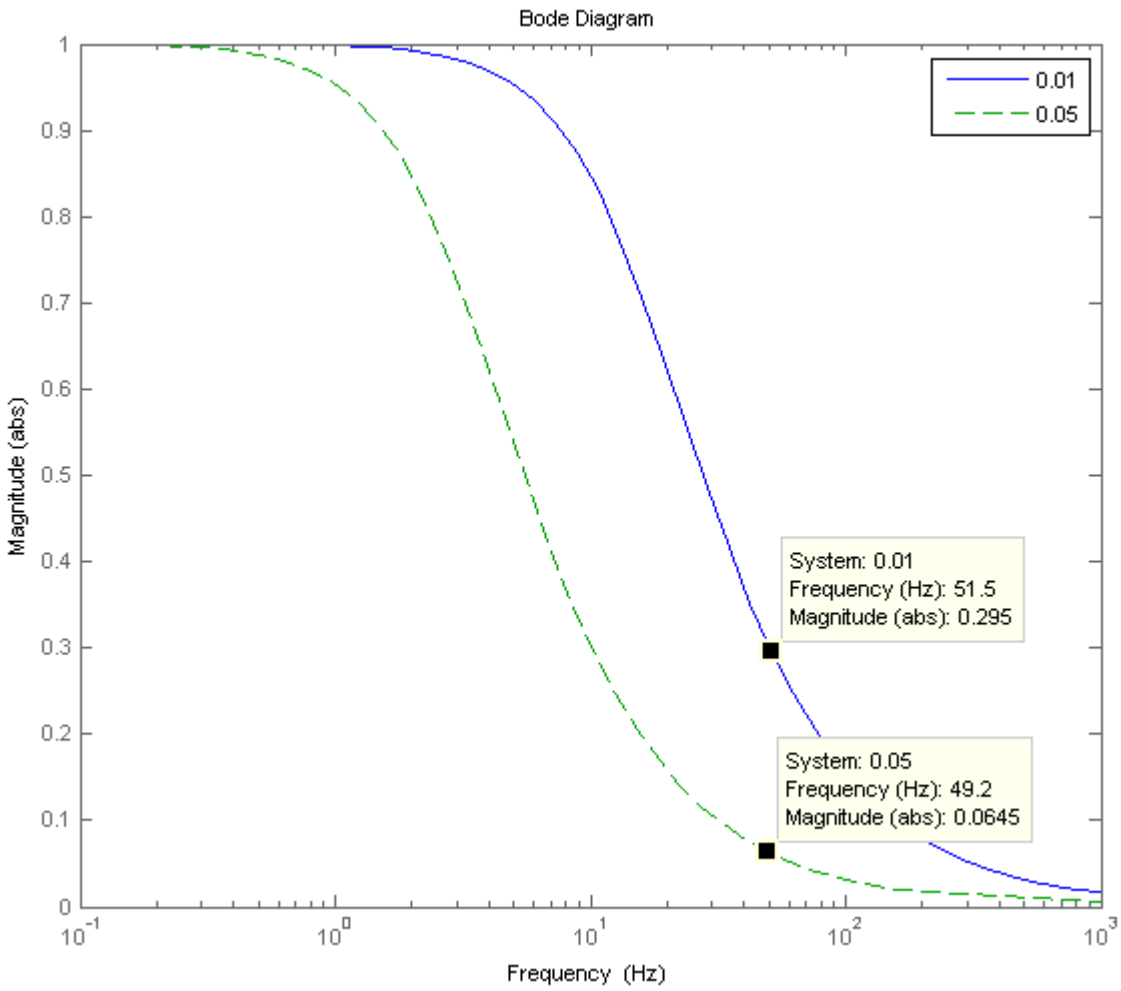

Figure 4-7: Selecting the τ for the first order low past filter

An additional low pass filter was applied to the commanded force signal. The HD^2 documentation recommends that the motor control signals should be band limited to 50Hz [28]. To meet this requirement, a 1st order low pass filter with a τ of 0.005s and corner frequency 31.8Hz was implemented.

Figure 4-8 and Figure 4-9 show two separate stable cases of a temporal bone haptic simulation after the implementation of filtering. Figure 4-8 shows the upper and lower forces and Figure 4-9 shows the drawn amps of the HD^2 motors. As can be seen in the two Figures below, the oscillatory and spiking behaviour is gone.

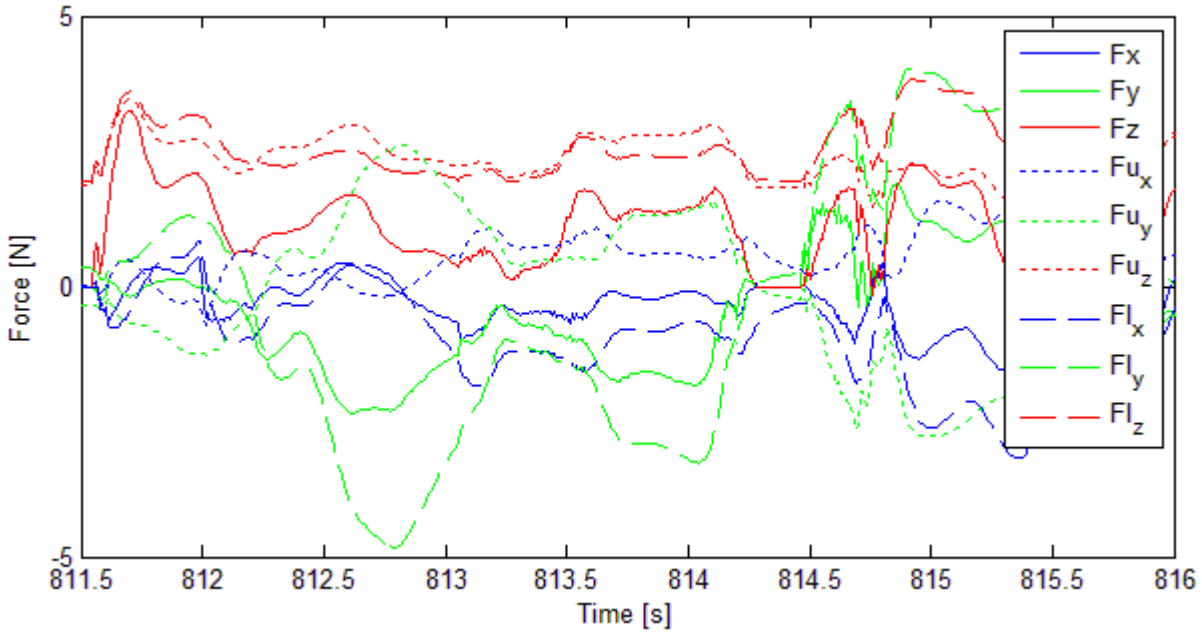

Figure 4-8: Stable case showing commanded haptic force and the corresponding upper and lower joint forces

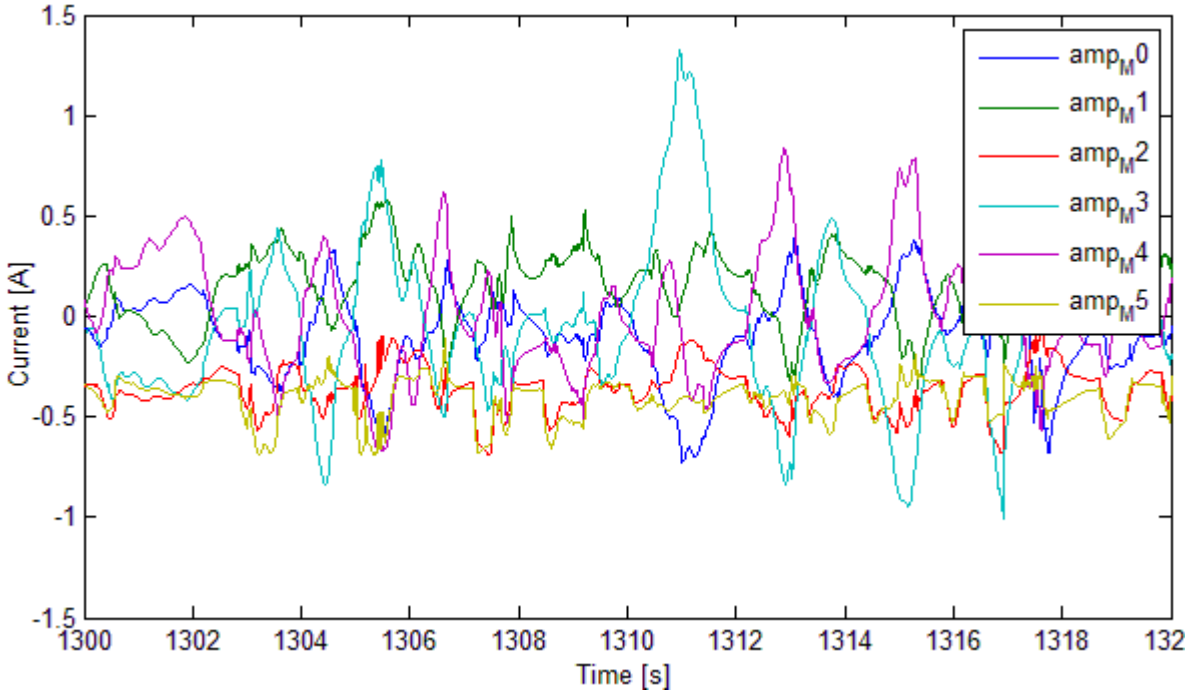

Figure 4-9: Stable case showing motor amps after low pass filter implementation

4.4.1 Human Tactile Response Characteristics

The human hand can sense and perceive four types of tactile information at the following maximum bandwidths [40]:

1. Compressive Stress: 10 Hz
2. Skin Motion Stimulus: 20 – 30 Hz
3. Vibration: 50 – 400 Hz
4. Skin stretch: (very low frequency)

Of these four, skin motion stimulus and compressive stress is the most important for virtual haptic simulation. From the previous section, two LP filters were applied:

Table 3: Low Pass Filter Characteristics

	Time Constant, τ [s]	Cut-off Frequency, f_c [Hz]
Force Signals	0.005	31.8
Couple Forces	0.05	3.18

Since the force that is perceived is comprised of two components, it is difficult to ascertain, the reduction in tactile bandwidth. Ideally, the resulting cut-off frequencies of both the force and the couple forces low-pass filters should both be greater than 30Hz.

However, the simulation with the attached drill feels comparable to the Phantom Omni case and to the HD^2 with the default manipulandum case. Any reduction in tactile feedback was not noticed by human users. There was no perceivable lag introduced by the addition of the low pass filters defined in Table 3. Therefore, it can be surmised that perhaps the bandwidth of the force signal is of far greater tactile importance than the bandwidth of the torque (force couple) signal.

Since the haptic simulation was now stable and human users were not able to detect any reduction in realism or performance, the implemented low pass filters were deemed sufficient. Attempts to further optimize the system by testing higher order low pass filters were deferred to future work.

With a working virtual reality haptic simulation, the next steps were integrating the drill state into the virtual simulation (Section 3.6) and attempting mixed reality haptic simulation (Sections 2.5 and 2.6).

4.5 Chapter Summary

Due to the mechanical design, large opposing couple forces are required to produce relatively smaller forces at the drill tip. These spiking couple forces resulted in instable haptic behavior. Implementing a pair of first order low-pass filters successfully removed the haptic chatter.

Any loss in haptic tactile bandwidth due to low-pass filter implementation was not detected by human users. Also it was shown that the HD^2 virtual haptic simulation operates well within the maximum continuous motor amp rating.

The current length from the handle center to the drill tip is far too long. This long offset reduces the overall efficiency of the system and makes it more prone to instable chattering. Therefore, there is justification in redesigning how the HD^2 holds a surgical drill. There is also potential for optimizing the filter design. Both these tasks are recommended as future work.

In summary, the temporal bone virtual haptic simulation works fully with the HD^2; both with the attached drill and without it. The next steps that were carried out were integrating the drill state into the virtual simulation (Section 3.6) and attempting mixed reality haptic simulation (Sections 2.5 and 2.6).

Due to the inherent positional inaccuracy of the HD^2 (covered in Section 2.5), mixed reality haptic simulation was not possible with the Quanser High Definition Haptic Device. The next chapter presents an evaluative study which compares the virtual reality haptic simulation on three hardware cases: the Phantom Omni haptic device, the standard HD^2 and the modified HD^2 with attached drill.

5 Evaluation of Various Haptic Manipulandum and Devices in Temporal Bone Simulation

Evaluative testing was conducted to study the perceived impact of using different haptic hardware with the temporal bone haptic simulation. Though it was determined that the HD^2 does not have the expected positional precision, a human user in the loop can substantially compensate for the lack of precision. End users naturally move the end effector to match the intended trajectory as shown on the virtual display. (Section 2.5 covers the HD^2 inaccuracy in detail.)

For Virtual Reality applications, the impact of HD^2 system inaccuracy appears to be unnoticeable to most end users as there are HD^2 VR simulations and tele-presence applications which produce favourable experiences [41–43].

5.1 Overview of Conducted End-User Tests

Temporal bone drilling procedures can be simulated in a variety of ways, including the use of cadavers, rapid-prototyped models and virtual haptic models. This study focused exclusively on virtual haptic simulations. Virtual haptic simulations are composed of a software and hardware component. The objective of these tests was to study how different haptic hardware is perceived by the end user and how it impacts the software temporal bone simulation effectiveness as a teaching tool.

The specific hypotheses being tested were:

1. Identical software running on different hardware platforms will result in a different learning experience during temporal bone simulation. Specifically, we compare the HD^2 and the Phantom Omni hardware platforms to each other.
2. Drilling with the HD^2 using an otic drill as the manipulandum provides a superior learning experience than drilling with only a standard manipulandum

Specifically, three cases were examined. Participants were asked to perform virtual temporal bone haptic mastoidectomy using the:

1. Phantom Omni haptic device
2. Quanser High Definition Haptic Device (HD^2) with default manipulandum
3. Modified HD^2 with attached otic drill

The same C++ temporal bone surgical haptic simulation software was used on all three hardware configurations.

This research endeavour was granted ethics approval by the Research Ethics Board and is filed under, REB number H2014:354.

5.2 Research and Design Methodology

This section covers the background engineering work required in preparation of the tests, participant criteria and sample size and describes the survey instrument used in data collection.

5.2.1 Preparatory Engineering Work

The three test cases operate using distinctly different hardware. Therefore the first step in preparing for these tests was to optimize the C++ temporal bone simulation and the HD^2 Simulink API for each case.

It is important to note that these changes, while necessary, did not significantly alter the simulation program nor does it modify the existing haptic control algorithms.

5.2.1.1 Optimizing Simulation Parameters

To simplify the optimization process, the most critical parameters impacting the simulation of drill forces were identified. These parameters are detailed in Table 4.

Table 4: Critical Simulation Parameters

float maxforce	This is a value which limits the maximum X, Y and Z forces. The default value was 3.3N to accommodate the limitations of the Phantom Omni haptic device. This float is defined in HapticControl.cpp of the C++ simulation.
bool m_bForceAveraging	A Boolean flag which can be changed by the user via the drill settings menu when the C++ simulation is running. If true, the force is averaged over a set number of force values. It was found that if this Boolean was false, there was an increased possibility of unwanted haptic vibration for the Phantom Omni case.
int m_nForceAveraging	This is an *int* which determines over how many force values is the force averaged before it is output to the haptic device. Low values result in "crisp" forces which clearly demark the boundary between air and virtual bone. However, low values increase the probability of unwanted haptic vibration. Very high values result in the virtual bone feeling soft or spongy. This *int* only impacts the simulation if *m_bForceAveraging* is True.
bool m_bPositionAveraging	A Boolean flag which determines if the haptic end position (virtual burr position) is averaged or not.
int m_nPositionAveraging	An *int* which determines how many position values are averaged before it is outputted into the rest of the haptic program.
float m_PIDKConstants	This value refers to the bone stiffness, K_P. The haptic force is determined by: $$\vec{F}_{PID} = K_P\vec{x} \quad (5\text{-}1)$$ Where, $\vec{x}$ is the *Desired Vector*: $$\vec{x} = DesiredLocation - HapticLocation \quad (5\text{-}2)$$

	Where, *HapticLocation* and *DesiredLocation* are defined in Figure 5-1. The position vectors are defined in mm. Therefore, as Kp, increases, so too does the produced haptic force (within the limits of *maxforce*). Therefore, the bone feels stiffer when K_P is higher. It is important to note that this works similarly to a simple proportional controller. Though the simulation is coded to output a force based on a conventional PID controller, the integral and derivative gains are set to zero. Therefore, those terms are therefore excluded in the above equation.
float m_fDrillSpikingValue	This value is defined in HapticControl.cpp but can also be changed by the end user via the Drill Settings Menu. This value impacts the simulation when the drill is spinning and is contact with virtual bone. At a frequency of *m_nDrillSpikingFreq* (default value of 10Hz), a spike force is added to the commanded haptic force: $$\vec{F}_{spike} = SpikingValue \cdot {\vec{F}_{PID}}/{\left\|\vec{F}_{PID}\right\|} \quad (5\text{-}3)$$
float m_fAxisBuzzingValue	This parameter is defined in HapticControl.cpp but can also be changed by the end user via the Drill Settings Menu. When the step pedal is pressed, a force is produced in the direction of the Phantom Omni handle. The magnitude of this force is equal to: $$F_{Buzz} = BuzzValue \cdot Rand_{[0,1]} \quad (5\text{-}4)$$ This buzz force updates at the same speed as the haptic loop.
Low pass filter τ constant	This parameter exists within the HD^2 Simulink API and is discussed in Section 4.4. The LP filter can be thought of as a moving average. The larger the τ, the larger the averaging time. Spikes in HD^2 commanded forces and couple forces are smoothened with a larger τ.

Figure 5-1 shows the algorithm used by the haptic simulation to simulate the force on the drill tip. The 'locked location' is the last point in space whereby the drill burr does not cross into the virtual bone region. The 'haptic location' is the actual location corresponding to the physical end point of the haptic manipulandum. This position can be inside or outside the virtual bone.

The 'desired next location' is what the user sees on the monitor display and it corresponds to a point just on the virtual bone surface that is closest to the 'haptic location'. Therefore, even if the drill tip is pushed beyond the bone's boundary, the virtual display helps give the illusion of it being a solid object. After each software cycle, the 'desired next location' becomes the new 'locked location'.

The 'desired vector' is the difference between the 'haptic location' and the 'desired next location'. The generated haptic force is proportional to the magnitude of the 'desired vector' by a magnitude of K_p as illustrated in Equation 5-1.

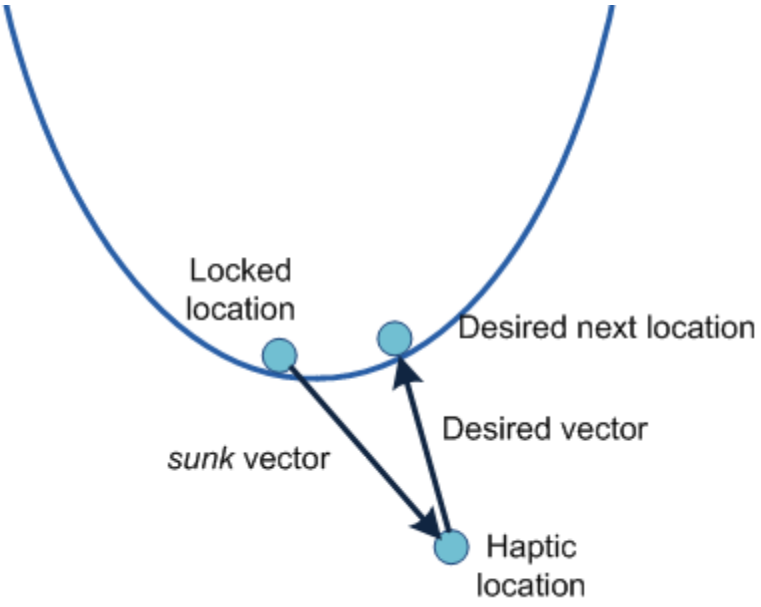

Figure 5-1: Haptic force generation via a sliding position locking algorithm [8, 44]

The critical simulation parameters were optimized by the MSc Candidate in consultation with an expert ENT (ears nose and throat) surgeon. The objective was to make each of the three cases feel as close as possible to reality. The result of this optimization exercise is shown in Table 5.

Table 5: Final Optimized Values of Critical Simulation Parameters

	Phantom Omni	HD^2 Alone	HD^2 with attached drill
maxforce	3.3	5.0	5.0
m_bForceAveraging	True	False	False
m_nForceAveraging	5	--	--
m_bPositionAveraging	False	False	True
m_nPositionAveraging	--	--	3
m_PIDKConstants	0.5	1.0	0.7
m_fDrillSpikingValue	0.2	0.35	0.2
m_fAxisBuzzingValue	0.1	0.1	0.0
LP Filter τ Constant	--	[0.02, 0.04]	[0.04, 0.1]

For the Phantom Omni case, force averaging is enabled whilst for the HD^2 cases it is disabled. With an *m_nForceAveraging* value of 5, there was no observed instability for the Phantom Omni case. Force averaging is turned off for the HD^2 cases because the force output is later filtered by the low pass filters within the HD^2 Simulink API.

Position averaging was not needed for the Phantom Omni and standard HD^2 cases and was thus left off. However, it was turned on for the HD^2 with drill case so as to mitigate the exacerbation of the HD^2 position error (described in Section 2.5), and to reduce the onset of haptic instability due to positional noise.

The maximum force for the Phantom Omni case was set at 3.3N because this was the maximum rated force for the device [45]. The maximum force was increased to 5N for the HD^2 cases to

take advantage of the device's higher force output specification [29]. It was hoped that having a higher maximum force output would make the temporal bone fill stiffer and thus closer to reality.

The parameter, *m_fDrillSpikingValue* dictates the magnitude of the 10Hz kickback force which occurs when the rotating virtual burr contacts virtual bone. Based on feedback from the expert ENT surgeon, it was deemed that 0.2 felt best for the Phantom Omni case and that 0.35 felt best for the standard HD^2 case. For the HD^2 with attached drill case, it was found that the system vibrated above 0.2 so it was left at this default value.

The variable, *m_fAxisBuzzingValue* was set at 0.1 for the standard HD^2 case and the Phantom Omni case. When the virtual burr is spinning, a random small force is output to the haptic manipulandum between 0 and 0.1 every 1ms so as to mimic the vibration felt by a spinning surgical drill. For the HD^2 with attached drill case, this variable was set to zero because it was not needed as there was a powered surgical drill attached to the haptic device.

Two values for τ are shown since a low pass filter is applied to both the commanded haptic force and to the required couple forces respectively. It is important to note that the constraining factor for setting the τ constants was haptic stability and not realism. The τ constants were the last values that were set. Ideally, significantly smaller τ values for the HD^2 cases were desired but were not possible.

The following section gives a brief description of how the haptic stiffness, K_p was chosen.

5.2.1.2 Real-world Bone Stiffness vs. Haptic Stiffness, K_p

The elastic modulus of cortical bone is approximately 3,000MPa [7]. This indicates that bone is a very stiff material that can withstand very large stresses with negligible deformation. Stiffness is a difficult property to estimate or measure for asymmetrical shapes as it is heavily dependent on geometry. To estimate the stiffness of human bone, we will use a simplified example utilising longitudinal stiffness.

Longitudinal stiffness can be defined as the force required per unit length of compression or extension. Defined in terms of elastic modulus, longitudinal stiffness, *S* is defined as [46]:

$$S = \frac{EA}{l} \tag{5-5}$$

where, E is the modulus of elasticity of the material, A is the cross sectional area and l is the length of the material [46].

The average thickness of the temporal bone varies from 4 mm to 8 mm [47]. Using an E of 3,000MPa, an A of 5mm × 10mm and an l of 5mm, the stiffness of temporal bone can be estimated to be about 30,000 N/mm.

There is no haptic system on the market capable of such high forces or stiffness; and such a theoretical haptic device would be dangerous to its end users. The maximum rated stiffness for the Phantom Omni haptic device and Quanser's High Definition Haptic Device is 1N/mm and 3N/mm respectively [29, 45].

The K_p values were set by an expert surgeon by drilling and trying different values in all three cases. It is guessed that the reason that the HD^2 cases required a slightly higher K_p was because the end user had to use their entire arm and drilled in the vertical Z axis. In contrast, the Phantom Omni haptic device, utilized the wrist and drilling was done in the horizontal Y axis. Therefore, one haptic device utilized a smaller and more precise muscle group, whereas another haptic device utilized a larger muscle group which was aided by gravity.

The difference between the K_p of the standard HD^2 case and the case with an attached drill may be attributed to the drill weight cancellation algorithm described in the following section.

5.2.1.3 Considerations for Drill Weight Cancellation

For the HD^2 case with the attached drill, special considerations had to be taken to deal with the weight of surgical drill. Section 3.4 discussed how the weight of the drill gripper was compensated for. To accurately simulate the forces a surgeon would feel, the weight of the surgical drill must be dynamically compensated for depending on the situation. This is best illustrated in Figure 5-2 below.

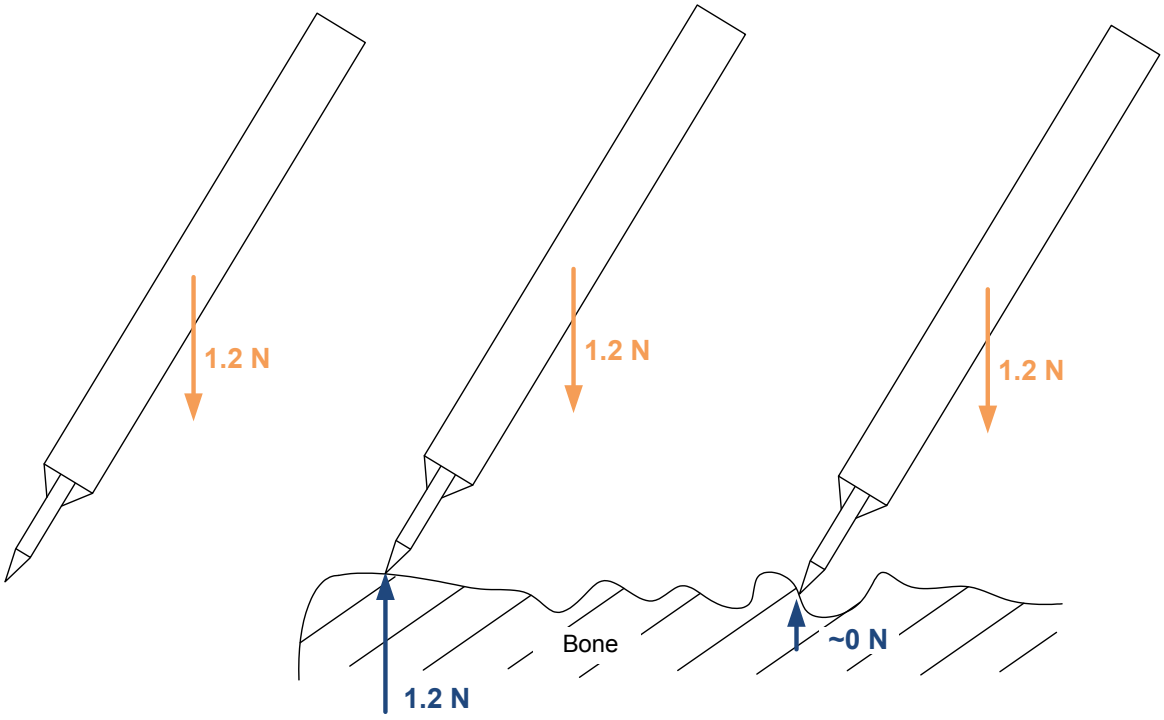

Figure 5-2: Free-body diagrams of the surgical drill

When the drill is not in contact with the virtual bone or tissue, the user would feel the weight of the drill. When the drill tip penetrates a virtual object with a predominantly negative Z sunken vector, the drill tip experiences a reaction force equal to the drill weight (in addition to the PID haptic force).

The logic for the reaction force due to drill weight on the drill tip is as follows:

```
%# F is the commanded haptic force
F_dr_cmp = [0 0 0]';

if F(3) > 0
    if F(3)> 0.15*norm(F)
        F_dr_cmp = [0 0 1.2]';
    end
end
```

This logic was implemented within the HD^2 Simulink API. F(3) refers to the F_Z component of the commanded haptic force, F. If F_Z is greater than zero and if $|F_Z|$ is significant compared to |F|, then the drill tip experiences an additional Z force equal to the drill weight. The constant 0.15 is used to determine if the $|F_Z|$ is significant in comparison to |F|.

A low-pass filter with a constant of $\tau = 0.09s$ was applied to this reaction force output to eliminate any potential vibration that may occur on the virtual bone surface.

Overall, the sensation felt from this algorithm is that on steep slopes, the weight of the drill is not compensated and the drill tip will naturally slide downwards. On horizontal surfaces the weight is compensated fully at the drill tip. On gentle slopes, the weight is fully compensated but if the user's hand is relaxed it may still slowly slide down the slope due to the F_{PID} algorithm shown in Figure 5-1 and Equation (5-1).

Expert users deemed that the haptic sensation created by this algorithm was agreeable and very close to reality. It feels closer to reality than either completely negating the drill weight or having the drill weight 'ON' all the time.

In retrospect, this can be further improved by having the weight reaction force normal to the slope of the virtual bone. This was not implemented at the time as the HD^2 Simulink API is not able to see the relevant data to ascertain the current slope of the virtual bone; it only receives force commands. One work around could be to assume that commanded force is normal to the virtual bone slope. The pre-requisites that F_Z be positive and significant should still remain.

5.2.1.4 Ergonomic and Workspace Considerations

For the HD^2 cases an initial offset had to be added to the haptic position. This was necessary as the C++ simulation is built around the Phantom Omni device which operates around [0, 0, 0] whereas for the HD^2 system, the origin is far outside its manoeuvrable workspace.

For the HD^2 wand and HD^2 with drill cases, the offset in meters was [-0.07, -0.27, 0.18] and [-0.07, -0.27, 0.368] respectively. Furthermore, the drill tip offset distance had to be set for each case:

Table 6: Drill Tip Offset (in mm)

	Phantom Omni	**HD^2**	**HD^2 with drill**
float m_DrillTipOffset	41.0	122.5	380.0

Care was taken to ensure that each setup was ergonomically comfortable for participants.

One advantage of the Phantom Omni haptic was that the device was designed to allow for comfortable small wrists movements. Furthermore, the simulation was designed about the origin position of the Phantom Omni device. Therefore no additional work was required for this case.

As the HD^2 has a much larger workspace, supporting blocks were fashioned out of polystyrene to support the participant's wrist. These are shown in Figure 5-3 and Figure 5-4. Without these blocks, the participants would have to hold their arms awkwardly in mid-air for sustained periods of time during evaluative tests.

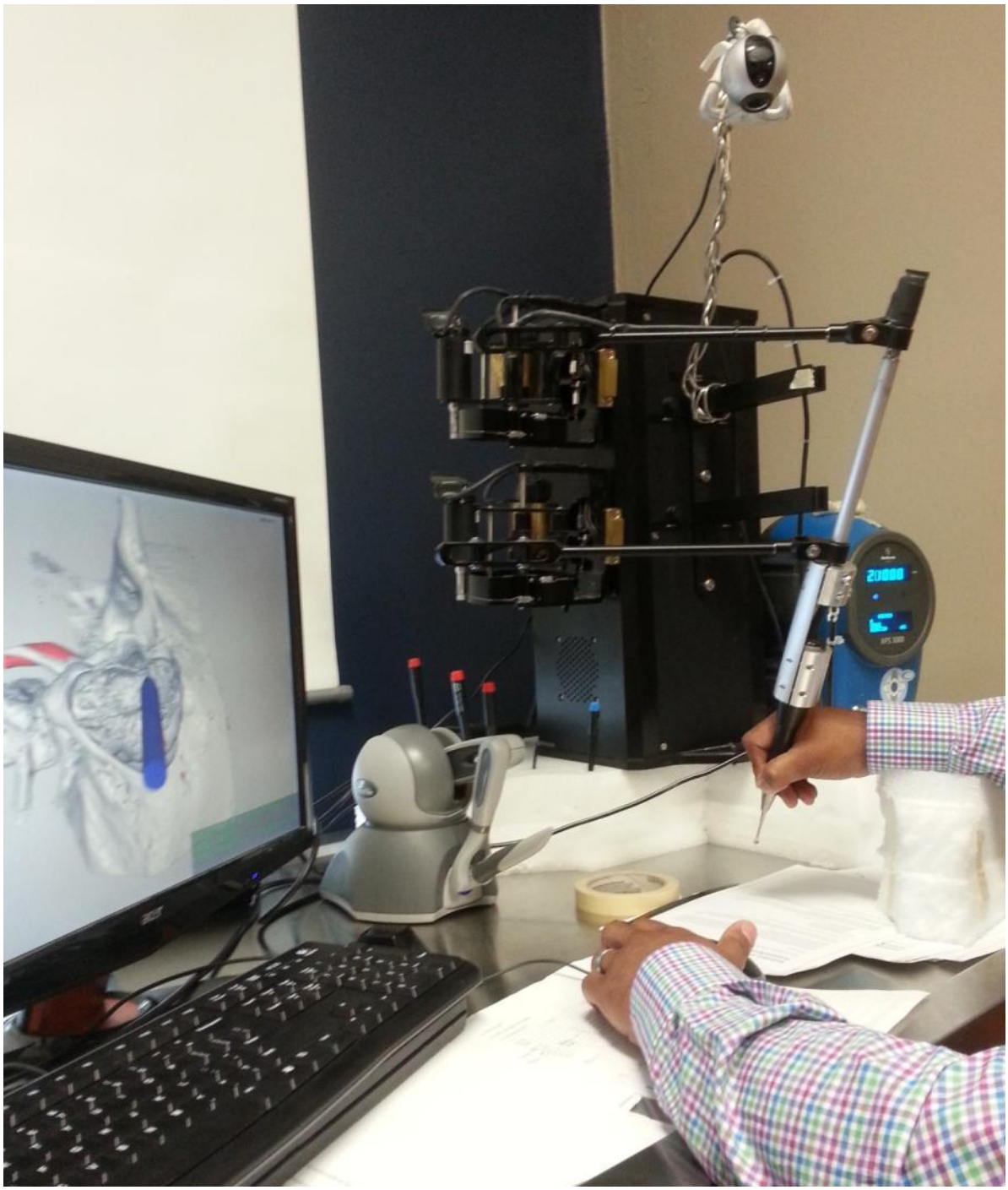

Figure 5-3: Supporting wrist block for HD^2 with surgical drill

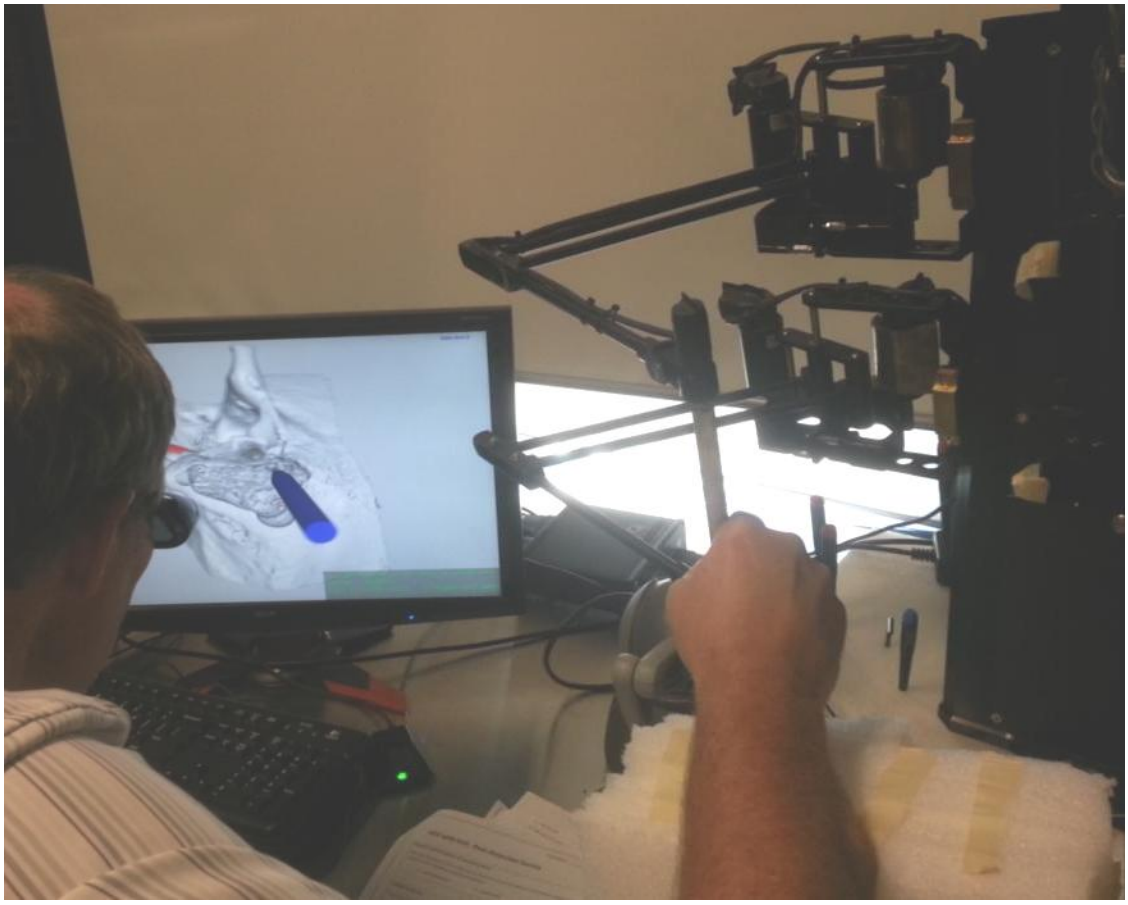

Figure 5-4: Polystyrene wrist supports for default HD^2 case

For the case with the default HD^2 handle, multiple polystyrene blocks had to be stacked to support the wrist. To ensure participant comfort, a chair with adjustable height was used for the tests.

This inconvenience was unavoidable since the HD^2 had to be elevated above the table to ensure clearance for the drill and its attachment (see Figure 5-3).

5.2.1.5 Recording of Simulation Data during Testing Evaluations

Simulation data is recorded by both the Simulink HD^2 API and by the C++ temporal bone simulation.

To record data within the C++ simulation, the Data Record box in the user menu must be checked. The HapticRecorder class of functions were modified to record the following data every 1ms:

- HD^2 time [s]
- C++ simulation time [s]
- Drill voltage [V]
- Drill step pedal boolean [0/1]
- Drill burr size [mm]
- Locked position [x, y, z]
- Actual haptic position (drill tip) [x, y, z]
- Haptic center [x, y, z]
- Haptic force [F_x, F_y, F_z]

The HD^2 API was modified to automatically record data for as long as the Quarc interface runs. When the Quarc executable is stopped, the recorded data is saved to a .MAT file. The following data is saved at a rate of 1000 Hz:

- Time [s]
- Handle error metric [m]
- HD^2 handle position [x, y, z]
- Handle orientation angle, θ_x [rad]
- Handle orientation angle, θ_y [rad]
- Joint angles, q [rad]
- Joint torques, τ [N.m]
- Step pedal voltage [V]

Though this data is not needed to answer the research questions proposed by the two Hypotheses, this data was deemed useful for future studies.

5.2.2 Participant Criteria and Sample Size

For this study, we required experienced temporal bone surgeons and otolaryngologist residents. Novices were not suitable for the chosen hypotheses. Since the goal of the study was to evaluate the existing haptic virtual simulations, we required individuals having extensive real life experience drilling the temporal bone. Due to the limited number of otolaryngologists in Manitoba, we were limited to a very small sample pool. Our goal was to meet the following numbers:

Senior ENT Surgeons: 6 – 8 participants
ENT Residents: 10 – 15 participants

All available ENT Surgeons participated giving a total number of 6. Testing utilizing ENT residents was deferred to future work.

5.2.3 Haptic Participant Evaluations and Data Collection

Participants were asked to perform several virtual procedures. The scheduled order of dissection was as follows:

Cortical mastoidectomy
Facial recess dissection
Labyrinthectomy
Translabyrinthine approach to the internal auditory canal

The above procedures were performed on each haptic setup for a total of three trials per participant. Each participant had 15 minutes to perform the virtual dissection on each haptic device. The order in which each participant used each haptic device was random and unique.

After each virtual dissection, the participant was asked to fill out an evaluative survey. Seven characteristics were evaluated using a 7 point Likert scale comparing the virtual haptic simulation to a cadaveric simulation, where the scale anchors are 1 is ‘Very dissimilar’, 4 is ‘Undecided’ and 7 is ‘Very similar’.

The evaluative categories were:

- Hardness of the cortical bone
- Hardness of the trabecular bone
- Vibrational properties/feel
- Acoustic properties/sound
- Drill skip
- Overall appreciation/similarity of bone
- Air-cell system

The 7 point Likert scale and these categories were chosen as they were utilized in previous studies conducted by the Surgical Modeling Simulation and Robotics Laboratory [48]. By choosing a 7 point scale once more, results can be compared across studies. Furthermore, these characteristics are considered important aspects of simulation realism, validity and relate to the surgeon's ability to perform [48].

After the participants completed and evaluated the three cases of virtual haptic temporal bone simulation, they were asked to rank the three simulations against each other; whereby 1 was the most preferred and 3 the least preferred. This forced ranking comparison was done with respect to the seven categories listed below:

- Overall realism
- Osseous realism
- Drilling realism
- Preferred for education
- Preferred for preoperative planning
- Ease of use
- Preferred simulation

These categories were chosen as they were deemed to be important or desirable qualities for temporal bone haptic surgical simulation [48]. A copy of the survey instrument is included in Appendix E – Survey Instrument.

5.3 Results

This section presents participant scores from the evaluative categories and participant results from the forced ranking. It also describes the statistical methods employed and presents the results. Also included in this section is representative recorded haptic data from one "HD^2 with attached drill" case.

5.3.1 Representative Profile Data of a Mastoidectomy Procedure

The recorded haptic data of a practice trial utilizing the HD^2 with attached drill is shown in the following figures.

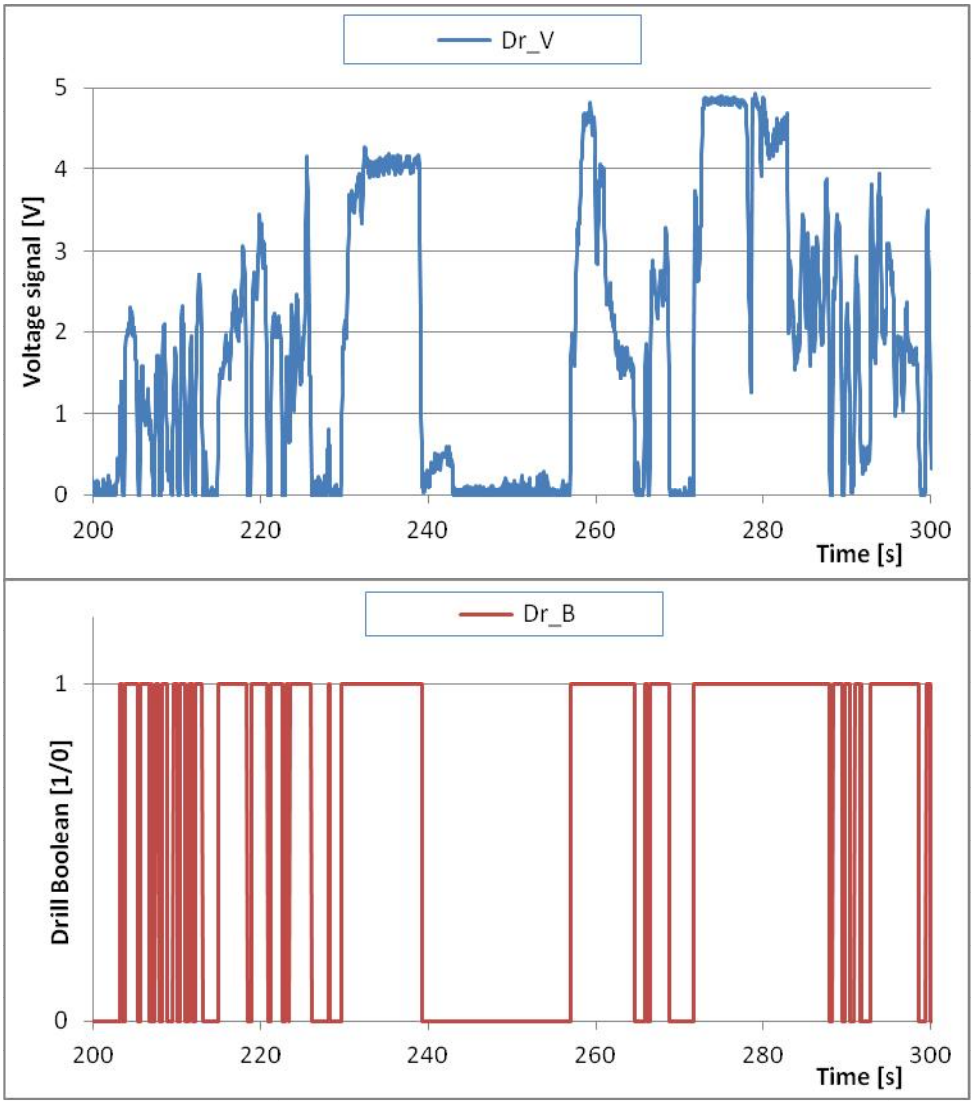

Figure 5-5: Graph showing step pedal voltage and drill Boolean

Figure 5-5 shows recorded drill state. The current software architecture does not utilize the 0-5V signal, so it is mapped to binary 0/1 Boolean.

Figure 5-6 and Figure 5-7 show X position and commanded F_X respectively. It can be seen that F_X is proportional to the difference between the actual haptic position (a_x) and its locked haptic position (l_x).

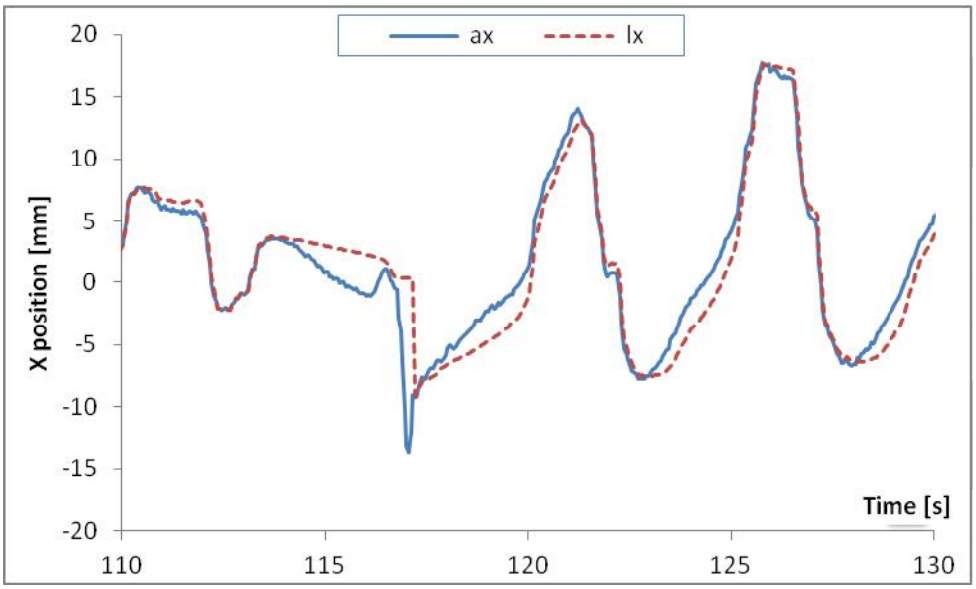

Figure 5-6: Actual haptic X (ax) position vs. locked haptic X (lx) position (simulation reference frame)

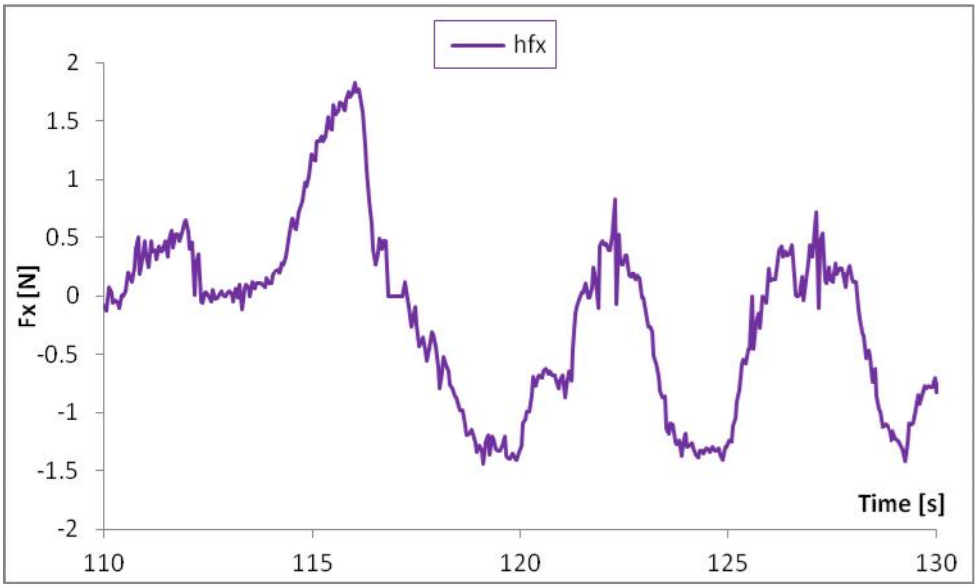

Figure 5-7: Resulting haptic force, F_x is a function of the difference between locked and actual X positions

Figure 5-8 shows the Y position and commanded Y haptic force, F_Y. It is important to note that the locked haptic position only updates when the drill tip penetrates virtual bone. This explains the profile observed between 170 – 180 s.

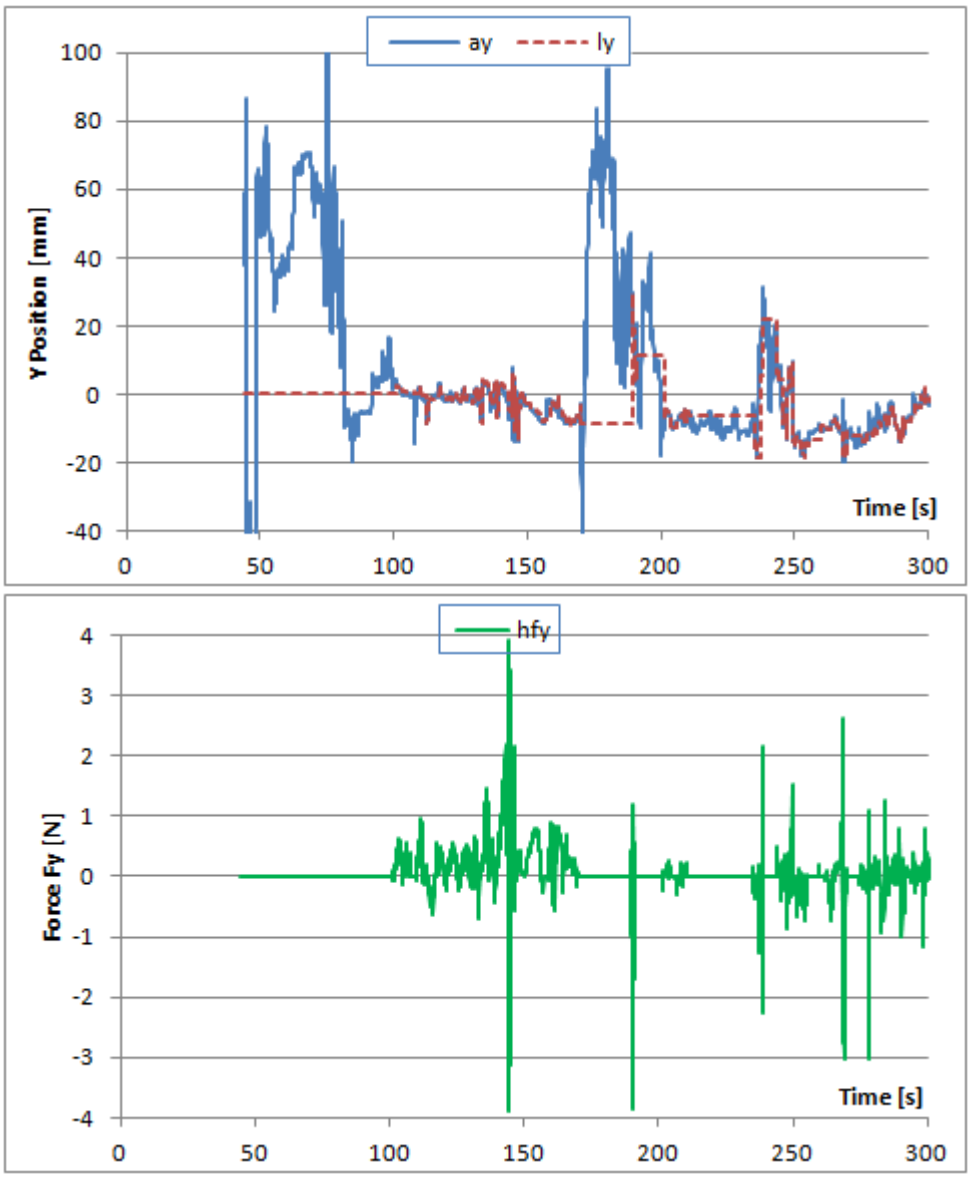

Figure 5-8: Actual (ay) and locked (ly) Y position and Y force (hfy)

Figure 5-9 shows the Z position and commanded F_Z haptic force.

For both Figure 5-8 and Figure 5-9, the smaller movements occur during drilling and when the drill tip is in contact with the virtual bone. The larger moments occur when the drill tip is not in contact with the virtual bone and therefore coincide with periods of zero haptic force.

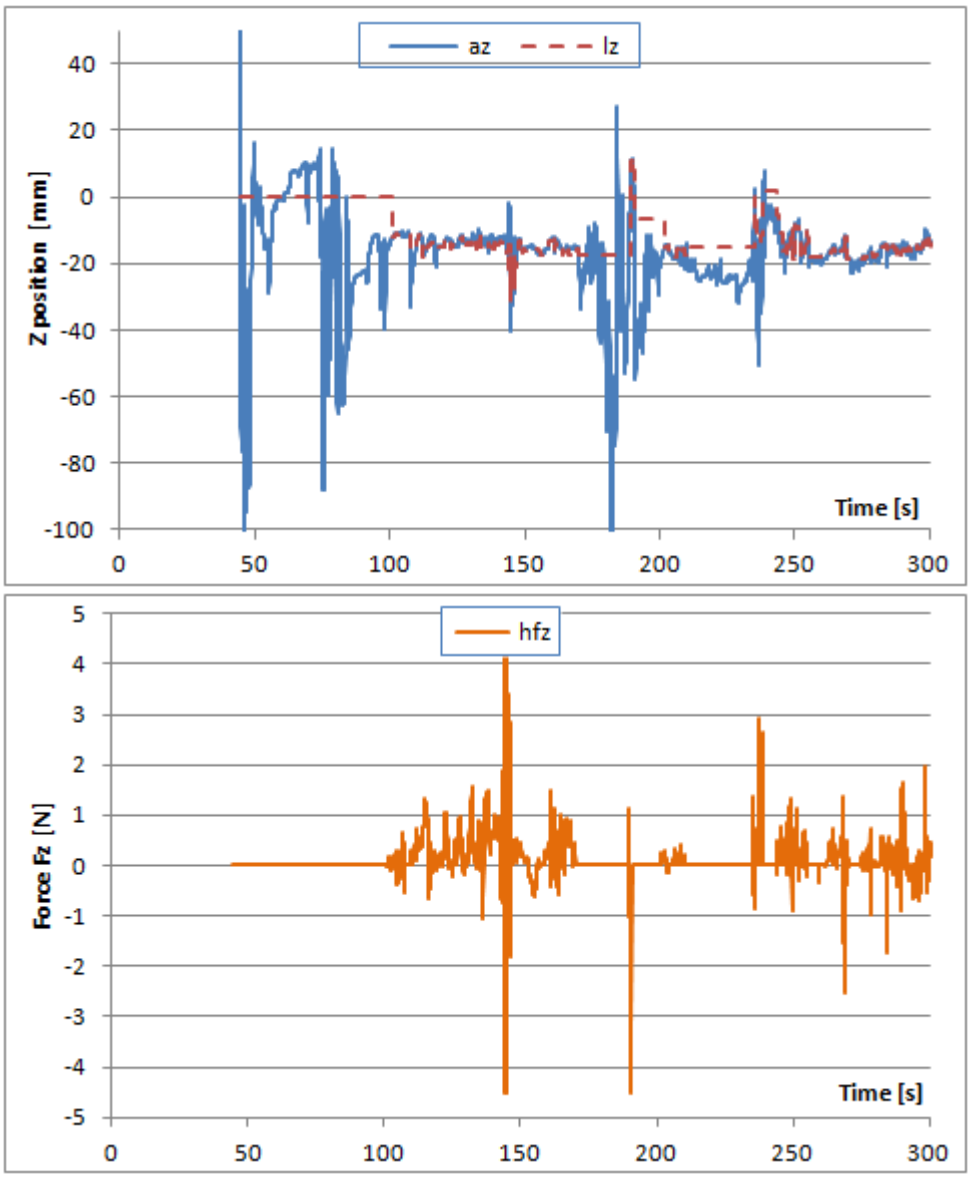

Figure 5-9: Actual (az) and locked (lz) Z position and Z force (hfz)

Figure 5-10 shows the orientation angles of the HD^2. θ_x and θ_y are the rotations about the local X and Y axis and are defined in Equations (C-10) and (C-11) (Appendix Section C.2.5).

Figure 5-11 shows the joint angles of the HD^2 linkages. In total, six angles are measured, three for the upper arm and three for the lower arm.

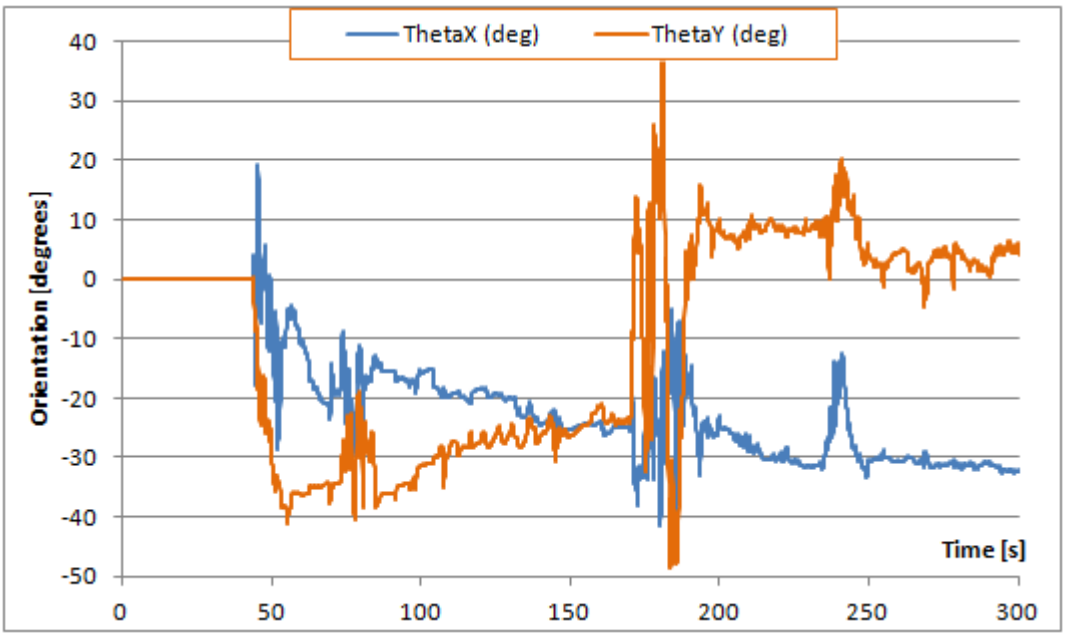

Figure 5-10: Orientation angles θ_x and θ_y (HD^2 reference frame)

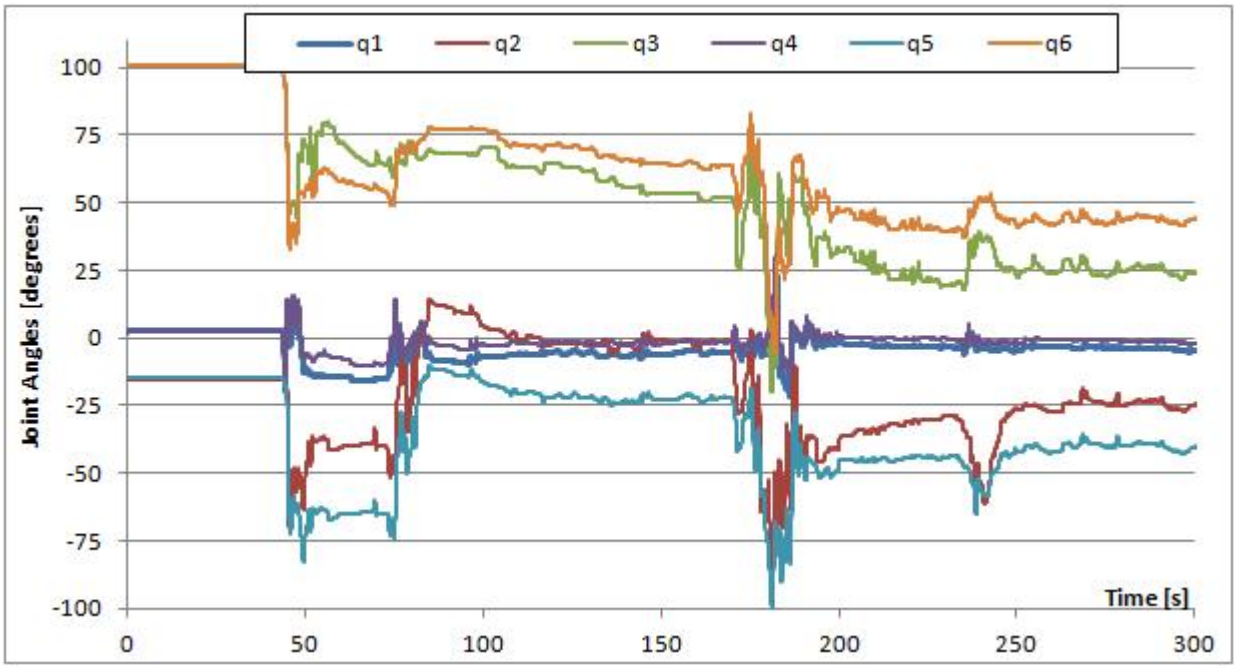

Figure 5-11: HD^2 joint angles

Figure 5-12 shows the joint torques and motor currents. Though motor currents are not directly measured, the two are related by a simple constant (Equation (C-13), Appendix Section C.2.7). It

can be seen that the device operates safely within its continuous motor limits which is ±3A (Table 12, Appendix A – HD2 Detailed Technical Specifications).

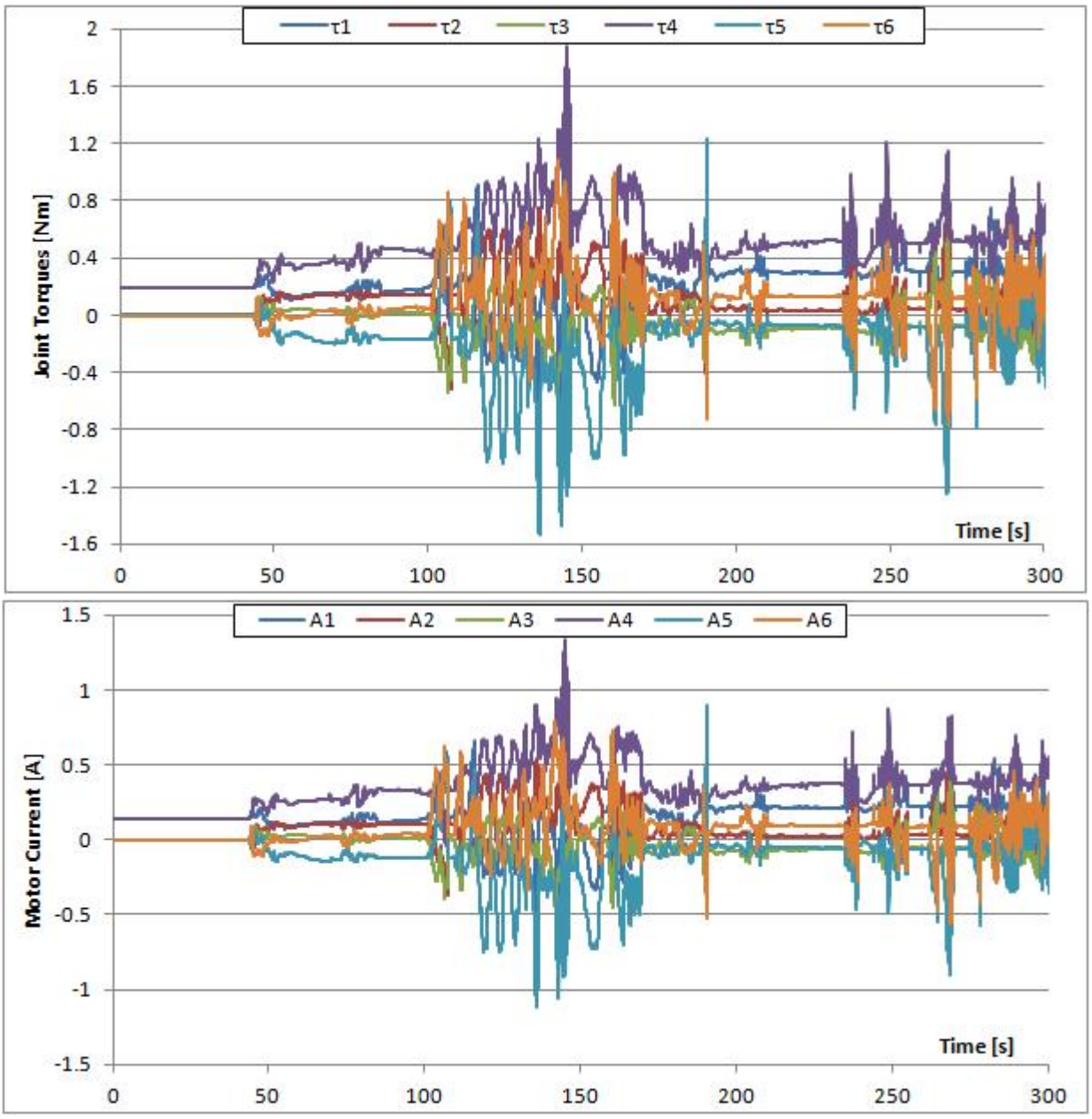

Figure 5-12: HD^2 joint torques and motor currents

5.3.2 Evaluative Categories Results

The compiled data from the evaluative surveys is shown in Table 7 and Figure 5-13. The average rating of each category and its standard deviation is also included.

As can be seen in Table 7 or Figure 5-13, the "HD^2 with otic drill" scored higher in all categories. However, to accept these results, it must first be determined that each data set or population is statistically distinct from one another.

Table 7: Compiled Participant Evaluation Data from Survey Instruments

	Post Disection Results (1-7)	S1	S2	S3	S4	S5	S6	Avg	STDV
Omni	Hardness of Cortical Bone	2	5	5	3	3	5	3.833333	1.32916
	Hardness of Trabecular Bone	2	5	2	2	2	4	2.833333	1.32916
	Vibrational Properties/ Feel	2	5	1	1	2	4	2.5	1.643168
	Acoustic Properties/ Sound	2	4	1	1	1	2	1.833333	1.169045
	Drill Skip	4	4	1	2	1	5	2.833333	1.722401
	Overall appreciation/ similarity of bone	2	3	2	1	1	4	2.166667	1.169045
	Air-cell system	2	6	3	2	2	3	3	1.549193
HD2 Only	Hardness of Cortical Bone	3	5	5	3	2	5	3.833333	1.32916
	Hardness of Trabecular Bone	3	5	6	2	2	4	3.666667	1.632993
	Vibrational Properties/ Feel	2	5	1	3	2	5	3	1.67332
	Acoustic Properties/ Sound	2	4	1	1	2	1	1.833333	1.169045
	Drill Skip	4	5	1	2	1	3	2.666667	1.632993
	Overall appreciation/ similarity of bone	3	5	5	3	2	4	3.666667	1.21106
	Air-cell system	3	5	5	3	3	5	4	1.095445
Hd2 w/drill	Hardness of Cortical Bone	4	4	6	5	2	5	4.333333	1.36626
	Hardness of Trabecular Bone	5	4	6	4	2	5	4.333333	1.36626
	Vibrational Properties/ Feel	6	3	7	3	2	5	4.333333	1.966384
	Acoustic Properties/ Sound	7	5	7	5	4	5	5.5	1.224745
	Drill Skip	4	5	1	4	2	5	3.5	1.643168
	Overall appreciation/ similarity of bone	4	3	6	5	2	5	4.166667	1.47196
	Air-cell system	4	5	6	5	2	5	4.5	1.378405

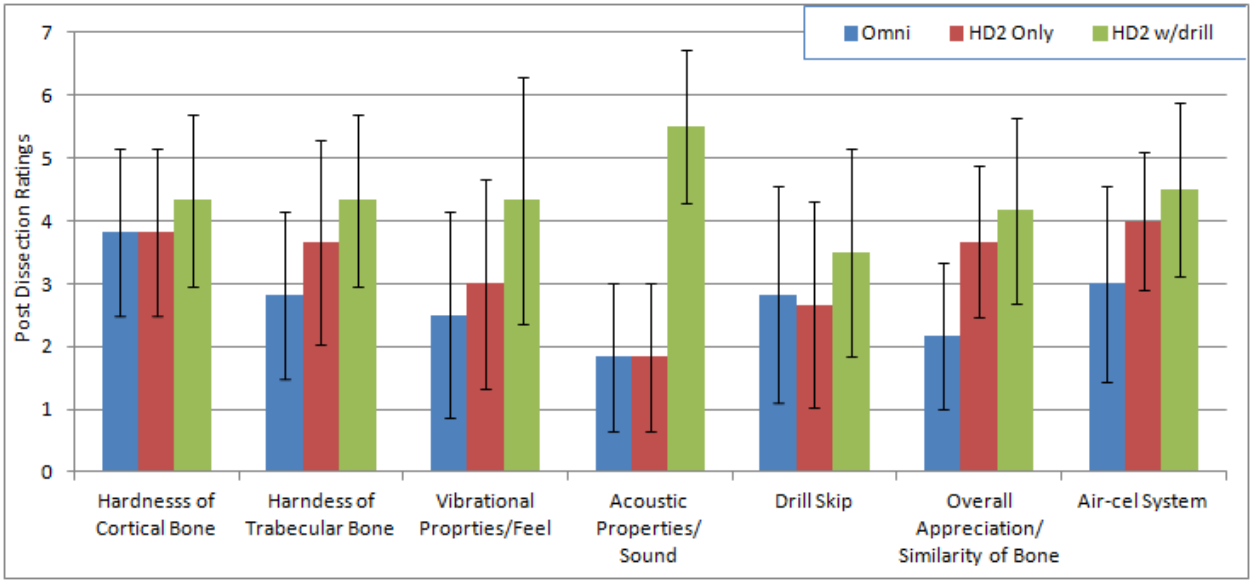

Figure 5-13: Mean participant evaluative data with standard deviation error bars, n= 6

To examine the hypotheses, these three data sets must be compared with one another. Analysis of variance (ANOVA) testing was carried out on each of these seven categories and is summarized in Table 8 below.

Table 8: Summary of ANOVA Testing of Haptic Data

	F	*P-value*	*F crit*	*H_0 Rejected?*
Hardness of the Cortical Bone	0.277778	0.761278	3.68232	No
Hardness of Trabecular Bone	1.613757	0.231862	3.68232	No
Vibrational Properties/ Feel	1.725979	0.211527	3.68232	No
Acoustic Properties/ Sound	19.05512	7.62E-05	3.68232	Yes
Drill Skip	0.42	0.664539	3.68232	No
Overall appreciation/ similarity of bone	3.9	0.043269	3.68232	Yes
Air-cell system	1.909091	0.182537	3.68232	No

In ANOVA testing, the Null Hypothesis, H_0, assumes that each data set or population is equivalent to one another; with any differences attributable to random variation. If F is greater than F_{crit}, H_0 is rejected and at least one of the data sets is statistically different from the rest.

"Statistically different" implies that the population distributions do not significantly overlap [49, 50].

Table 8 shows that only two evaluative categories, "*Acoustic Properties/ Sound*" ($P \approx 0$) and "*Overall Appreciation/ Similarity of Bone*" ($P = 0.043$) were deemed to be produce results that were statistically unique.

Two-tail, Student's *t*-Test was then carried out on these two categories to determine which haptic cases were statistically distinct from one another. Just like ANOVA testing, the Null Hypothesis in a *t*-Test assumes both data sets are equivalent and non-distinct. To reject H_0, t Stat < t Critical or t Stat > t Critical [51].

The conducted two-tailed t-tests are summarized in Table 9.

For the category, "*Acoustic Properties/ Sound*", the results show that participants graded the Phantom Omni and the HD^2 with default manipulandum identically. The HD^2 with drill evaluative results were deemed to be statistically unique from both the standard HD^2 case ($P = 0.0003$) and the Phantom Omni case ($P = 0.0003$).

Table 9: Summary of Conducted Two-Tailed *t*-Tests

t-Test: Two-Sample Assuming Unequal Variances		***t* Stat**	**P(T<=*t*) two-tail**	***t* Critical two-tail**	***H_0* Rejected?**
Accoustic Properties/ Sound	*Omni and HD2*	0	1	2.228139	No
	Omni and HD2 w drill	-5.30467	0.000345	2.228139	Yes
	HD2 and HD2 w drill	-5.30467	0.000345	2.228139	Yes
Overall Appreciation/ Similarity of Bone	*Omni and HD2*	-2.18282	0.053987	2.228139	Almost but NO
	Omni and HD2 w drill	-2.60623	0.02621	2.228139	Yes
	HD2 and HD2 w drill	-0.64253	0.534982	2.228139	No

For the category, "*Overall Appreciation/ Similarity of Bone*" the "HD^2 with drill" case displays statistically significant difference to the "Phantom Omni" case (P = 0.026), but is not statistically different to the "HD^2 with standard manipulandum" case (P = 0.535). Comparing "HD^2 with standard manipulandum" and "Phantom Omni" data sets result in a p value of 0.0540 which can be described as being on the edge of significance.

5.3.3 Forced Ranking Results

The forced ranking participant data is summarized in Table 10. It is evident that the HD^2 with drill is highly preferred over the Phantom Omni and HD^2 with standard manipulandum.

For "Overall Realism", 4/6 participants or 66.6% preferred the HD^2 with drill over the other two haptic options. Assuming that this result was the result of random probability, the statistical significance would be p = 0.0823.

For "Osseous Realism", "Drilling Realism", "Preferred for Education", "Preferred for Preoperative Planning", "Ease of Use" and "Preferred Simulation" 83.3% of the participants preferred the HD^2 with drill (P = 0.0164).

Table 10: Compiled Forced Ranking Data

Table shows number of times ranked as either 1st, 2nd or 3rd		Phantom Omni			HD2 Only			HD2 w Drill		
		1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
Overall Maniplandum Comparison	Overall Realism	2	1	3	0	4	2	4	1	1
	Osseous Realism	1	1	4	0	5	1	5	0	1
	Drilling Realism	1	1	4	0	5	1	5	0	1
	Preferred for Education	1	2	3	0	4	2	5	0	1
	Preferred for Preoperative Planning	1	2	3	0	4	2	5	0	1
	Ease of Use	1	2	3	0	4	2	5	0	1
	Preferred Simulation	1	2	3	0	4	2	5	0	1

5.4 Discussion and Analysis

There is no clear preference exhibited in choosing between the standard HD^2 haptic device and the Phantom Omni haptic device. This neutral preference is demonstrated in both the evaluative 7 point categorical scoring as well as in the results of the comparative forced ranking.

Though the HD^2 haptic device produced higher forces (5N vs. 3.3N), this was not reflected in the "Hardness of Cortical (Compact) Bone" category. The category, "Overall Appreciation/ Similarity of Bone" the HD^2 did score higher than the Phantom Omni by 1.5 points but with $p = 0.054$. The HD^2 with default manipulandum scores higher than the Phantom Omni in all categories except acoustics which is a tie. However, no data set pairs are statistically distinct.

From the current trials and data thus far, Hypothesis 1 should be rejected. There is no statistically significant difference in how the temporal bone simulation is perceived between the Phantom Omni and the HD^2.

Hypothesis 2 however should be accepted. It has been shown that using the HD^2 with an attached surgical drill is highly preferred by expert surgeons. This is reflected in both evaluative results as well as in the forced ranking. It is therefore thought to be provide a superior learning experience than when compared to the standard manipulandum. To fully prove or disprove this notion, future construct validity testing will be required.

As expected, this configuration scored very high on "Acoustic Properties" ($\mu = 5.5$, $p = 0.000345$). This added a greater sense of realism to the simulation, especially due to an "accidental feature" which was perceived and appreciated by all the senior surgeons.

If the drill was spinning whilst the tip penetrates virtual bone, power is drawn from the outlet at a greater rate to the HD^2 than to the surgical drill. This small loss of power resulted in a noticeable drop in the pitch of the drill, indicating that the drill loses speed slightly in these cases. This drill pitch modulation matched reality closely and was positively commented upon by the surgeons.

The other category which had statistically unique results was "Overall Appreciation/ Similarity of Bone". HD^2 with drill scored highest in this category ($\mu = 4.1667$, $p = 0.0262$).

Though the HD^2 mean score was higher in the other five evaluative categories, the results were not found to be statistically significant due to high variance and low participant count. Regardless, the HD^2 with drill case was highly favored in the forced ranking in all categories. This is indicative that the sound and feel of a real surgical drill is a critically important and desirable feature in simulating temporal bone surgery.

Due to unavoidable low participant numbers, the value of these results is questionable. With higher participant numbers, evaluative categories may become statistically distinct. Intern trials would potentially increase the participant count by 10 – 15.

5.5 Observations and Recommendations

1. *Constraints on orientation or manipulability of the haptic device have an impact on both comfort and surgical realism.*

This point is illustrated in Figure 5-14. One participant strongly disliked being forced to drill in a horizontal orientation with the Phantom Omni. Since the Omni is a relatively small device, it can be attached to a designed platform to adjust its orientation.

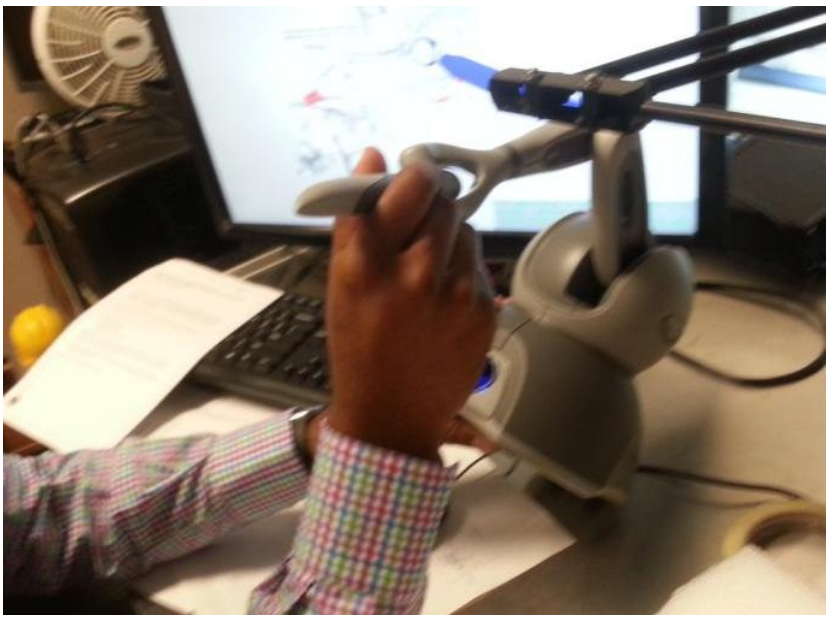

Figure 5-14: Participant forcing the Omni Phantom into a vertical orientation during evaluative testing

The HD^2 with attached drill had the opposite issue; it forced users to hold the drill almost vertically (approximately ± 30° from the vertical). The one participant that rated the HD^2 with drill last in the forced comparative ranking preferred to hold the drill horizontally. Drilling horizontally is impossible due to the mechanical limitations of the set up. This manipulability constraint can be alleviated by redesigning how the drill is held by the HD^2.

In contrast, the unmodified HD^2 with default manipulandum has a wide, natural range of manipulability with a roll range of 180° and pitch range of 180° [29].

A potential future project might entail redesigning the drill gripper attachment and/or designing an adjustable platform for the Phantom Omni so that users can drill in the vertical plane if they desire.

2. *Realistic drill acoustics are essential to an immersive haptic simulation*

Stanford's Salisbury Robotics Lab has a working temporal bone surgical haptic simulator and have identified that realistic drill acoustics are critically important. The Stanford team have identified that the drill sound/pitch must be modulated with respect to drilling force. Another factor that determines the drill pitch is the thickness of the bone [52, 53].

A modulating drill acoustic function could be added to the existing C++ simulation. It would be an interesting study to see if the presence of drill sounds can have an impact on an experienced surgeon's perception of drill force and/or realism.

3. *Existing drill gripper design which relies on static friction unsuitable at higher forces*

It is recommended to modify the existing drill gripper before the second phase of testing (ENT Residents) commences. Relying on static friction is potentially unsafe. A notch or key is recommended where the gripper meets the end of the HD^2 handle as shown in Figure 5-15.

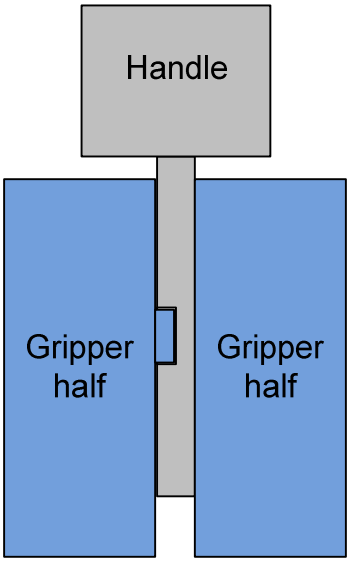

Figure 5-15: Notch or key design improvement to existing drill clamp design

Where the gripper meets the drill, one half can be brazed to the drill itself eliminating any possibility of slipping. Alternatively, a notch can be machined onto the drill itself.

4. *Utilize recorded haptic data to evaluate and validate performance metrics*

A future study can utilize the recorded data to develop or validate performance metrics. Metrics mentioned in the literature include visibility, drilling technique, appropriate bone removal, distance from nerves, drill path, drill speed and drill forces [54 – 57].

This in itself is a large project. Metrics may be ascertained post haptic surgery from raw recorded data or it may be coded into the haptic simulation itself. This should allow the haptic simulation to be a more comprehensive training tool.

5.6 Chapter Summary

Senior ENT Surgeons were recruited to evaluate three distinct haptic configurations which ran an identical temporal bone simulation. The 3 cases being evaluated were the Phantom Omni, the

standard HD^2 and the modified HD^2 with attached surgical drill. For each case, the software was optimized so that the simulation felt as close as possible to reality.

It was found that the HD^2 with an attached surgical drill configuration was the most widely favoured, and is reflected in both the evaluative results as well as in the forced ranking results. Evaluative results between the Phantom Omni and the HD^2 are near identical showing no statistically significant difference. However, in forced ranking, the HD^2 appears to be favored over the Phantom Omni.

6 Conclusions and Contributions

The primary goal of this research was to design, develop and validate a mixed reality haptic temporal bone surgical simulator. The value of achieving a mixed reality haptic simulation is tremendous as it is a novel concept not yet reported in the literature [9-13, 25]. Thus the development of such a prototype would simultaneously advance both haptic research and medical training.

In addition to the objective of achieving a mixed reality surgical simulation, this research also sought to answer three questions:

1. How would altering the mechanical configuration of the HD^2 impact its performance, stability and usability when interfaced with the virtual reality surgical haptic simulation? How would these problems (if observed) be mitigated?
2. Is there a discernible difference in end user experience if the same virtual haptic surgical simulation program is run on different haptic devices? In other words, would a surgeon have a significantly different experience if a virtual haptic mastoidectomy is performed on the standard HD^2 vs. a Phantom Omni haptic device?
3. Does the addition of a surgical drill to the haptic simulation improve the realism of the temporal bone virtual haptic simulation? Would expert surgeons prefer using the standard manipulandum of the HD^2 or the attached drill as the end-effector?

Research findings, contributions and future works are presented in this final thesis chapter.

6.1 Limitations of the HD^2 System

The HD^2 is unsuitable for precise mixed reality haptic applications. With the current HD^2 system, fiducial registration (i.e. alignment between real and virtual objects) is not possible. The device has a proven inaccuracy that is difficult to quantify and to correct. The outputted end positions of the upper and lower universal joints of the handle are approximate and are not accurately calculated. Using handle length as an error metric, it was shown that the difference

between absolute handle length and the calculated handle length was in the range of -4mm to +8mm (Section 2.5).

While the HD^2 listed specifications should make it suitable for this application, it was proven that these specifications were false. Furthermore, since the HD^2 is relatively new, it is an untried device with very few publications centered on it.

Substantial modification and testing will be required to rectify the current positional inaccuracy of the HD^2 system. This is discussed further in Section 6.3 Future Work.

6.2 Impact due to Mechanical Clamp Attachment

The current mechanical modification of the HD^2 negatively impacts efficiency, stability, available workspace and manipulability.

Due to the long offset length between the handle center and the drill tip, large yet opposing forces are required at the handle joints to create significantly smaller forces at the drill tip (Section 3.3). To simulate a force of 1N in the X-Y plane at the drill tip requires forces of 1.6N and 2.6N at the upper and lower handles respectively. This corresponds to a 76% reduction in system efficiency.

These large opposing forces have a tendency to create large jerking moments which can result in haptic chatter or instability at the virtual bone surface. Though the haptic chatter was mitigated utilizing low pass-filters, this has a detrimental impact on haptic performance. The addition of the low-pass filter introduces haptic delays to the system and reduces the bandwidth. This reduction in bandwidth can result in loss of tactile information such as surface texture or vibration (Chapter 4).

Due to intrinsic errors of the current HD2 system, the upper and lower end positions are not accurately calculated. The chosen mechanical modification exacerbates the positional uncertainty due to the long offset length between the lower joint of the handle and the drill tip (Section 2.5).

The drill gripper design significantly reduces the useable workspace. Furthermore, the drill has a significantly reduced manoeuvrability compared with the standard manipulandum. While staying

within the temporal bone haptic workspace, the drill can only tilt by about 30° from the vertical; whereas the standard manipulandum can be moved ±90° from the vertical (Section 5.5).

The current design completely relies on static friction. This design choice is potentially unsafe when haptic forces are high. At the very least, this design should be modified so that there is a key or notch groove to lock the pieces into one another (Section 5.5).

6.3 VR Haptic Simulation Achieved on Modified HD^2

The HD^2 was successfully interfaced with an existing temporal bone surgical simulation to produce a robust, stable and safe experience. To accomplish this, an algorithm to change the haptic interaction point was derived and implemented into the Simulink driver. Additionally, the weight of the dill clamp attachment was negated. Furthermore a C++ API was coded to allow communication between the haptic simulation and the HD^2 hardware.

The haptic simulation has been run on the HD^2 with both the standard manipulandum and with the attached drill countless times by numerous lab members and volunteers over a period of approximately one year. Therefore, it can be surmised that the system is robust and safe. Furthermore, the HD^2 is operating well within its safety limits. Recording and graphing the motor currents show that all six motors operate within ± 1.5A (Section 4.4, Section 5.2.1.5). This is far from the continuous maximum and peak maximum of 3A and 5A respectively [28] (Appendix A – HD2 Detailed Technical Specifications).

Even though, the primary objective of developing a mixed reality haptic surgical simulation could not be achieved, partial success was attained. The current modified HD^2 system with an attached, integrated surgical drill can be described as a partial mixed reality haptic system. There are no temporal bone haptic simulations which utilize an actual surgical drill as its end-effector.

6.4 Phantom Omni and Standard HD^2 were Rated Equivalently

The temporal bone haptic simulation running on the standard HD^2 and the Phantom Omni were rated equivalently by expert surgeons.

Based on the results of the "Evaluation of Various Haptic Manipulandum and Devices in Temporal Bone Simulation" study, there was no clear preference exhibited in choosing between the Phantom Omni device and the HD^2. Both devices received similar rating in the evaluative categories, with any differences between data sets being statistically insignificant. This neutral preference between devices was also reflected in the forced comparative ranking. This finding may suggest that different haptic hardware running the same haptic simulation software does not produce a significantly different user experience.

6.5 HD^2 with Attached Surgical Drill was Most Preferred

The HD^2 with attached surgical drill was the most preferred hardware to run the Temporal Bone Haptic Simulation

In all the forced ranking comparisons, the HD^2 with attached otic drill was highly preferred being ranked 1st by most participants in all categories. In the evaluative survey results, the HD^2 with attached drill scored higher than the other two cases in all 7 categories. Due to high variance and/or low sample size, only two categories exhibited statistical significance: "Overall appreciation/similarity of bone" ($\mu = 4.1667$, $p = 0.0262$) and "Acoustics" ($\mu = 5.5$, $p = 0.000345$).

Based on participant feedback, the presence of a surgical drill is an important and desirable feature in temporal bone haptic simulation. Also of great importance is authentic drill acoustics, a feature which is absent in our temporal bone haptic simulation.

6.6 Contributions

The primary contributions of this thesis include the following:

- It was discovered that the positional accuracy of the HD^2 system does not match its listed specification and is thus unsuitable for precise mixed reality haptic applications.

- The HD^2 was modified to secure a surgical drill below the default manipulandum. It was found that this prior mechanical modification negatively impacts the efficiency, stability, available workspace and manipulability of the haptic device.

- Virtual reality temporal bone haptic simulation was achieved on the HD2; for both the standard configuration and the modified configuration with an attached surgical drill.

- The Phantom Omni and standard HD2 were rated equivalently by expert surgeons. Therefore, it can be inferred that different haptic hardware running the same haptic simulation software do not produce significantly different user experiences.

- The modified HD2 with attached surgical drill was the most preferred by end users. Therefore it was shown that the addition of a surgical drill is an important and desirable feature that improves surgical realism.

- An algorithm was derived to transform the haptic interaction point from the default handle center to any other point. This algorithm was also used to negate the weight of the mechanical clamp attachment. These equations are general and can be re-applied if the drill attachment pieces are ever redesigned.

- A platform has been created that can facilitate further research, testing or design. The C++ simulation as well as the Simulink driver has already been modified to output critical data such as time, force, position, amps, joint angles, torques etc.

- A study was designed to evaluate haptic devices running a temporal bone haptic simulation. The existing software and survey instruments can be used on additional volunteering expert surgeons to increase the sample size. Furthermore, the identical survey instruments can be used in any future studies with other haptic temporal bone simulations software or with other haptic hardware.

- A clamping assembly which rigidly attaches the HD^2 end-effector or linkages to a force sensor was designed and machined (Appendix B – Related Technical Work). This can be used to facilitate future studies or tests.

6.7 Future Work

There is a very large scope of potential future work. This section is divided into three sub-sections: future medical studies and evaluations, improvements specific to the haptic hardware and improvements to the temporal bone haptic simulation.

6.7.1 Expert Evaluation or Medical Education Studies

In the short term, the evaluative haptic study needs to be carried out using ENT residents. This would be valuable as it will increase the final sample size of the pilot study. The two groups, senior surgeons and residents, can be compared to one another. Will younger surgeons rate the haptic systems more highly or will they produce similar rating patterns?

Future studies can involve concurrent testing to answer the research questions, "Is temporal bone haptic simulation an effective training tool?" and "Which haptic system best teaches trainees proper drill techniques?" This could be accomplished by taking three distinct populations of novices and having them train on one of three hardware cases; after which each participant will drill a physical model which would be graded by a panel of expert ENT surgeons. A fourth population group would drill the physical models without any haptic training as a control. Analysing the results should shed light on the chosen research questions.

6.7.2 Improving the Haptic Hardware

Further testing and design can be carried out to improve the current low-pass filter that was applied to the modified HD^2 system. Ideally, the final result would be a stable haptic simulation with minimal loss in tactile bandwidth.

Correcting the inherent inaccuracy of the HD^2 was beyond the scope of this thesis. In no particular order, some recommendations that may alleviate this problem are listed below.

- Retrofit the end-effector universal joints with encoders. There are no encoders on these joints and the current system makes assumption as to its position and orientation
- The current calibration technique is inaccurate and should be improved. Utilize a laser to accurately measure the initial angles and the linkages of the device. Tracking lasers can be used to detect non-geometric parameters if they exist [58].
- Implement real time position tracking utilizing either lasers or 3D motion capture cameras. This would allow for the position to be measured independently of the HD^2 system and can be fed directly into the C++ simulation.

It is also recommended that the current drill attachment assembly be redesigned such that the drill is either between the upper and lower joints of the end-effector or parallel to it. The current design negatively impacts the overall performance of the system and reduces the overall manipulability.

Alternatively, the HD^2 can be replaced with a haptic device that has a proven position resolution of $\leq$ 0.25mm. Ideally, the surgical drill attachment mechanism should be designed before the device is purchased.

6.7.3 Improvements to the Temporal Bone Haptic Simulation

Trabecular bone did not score highly on the evaluative survey for all three hardware cases. This is indicative that the problem lies with the haptic simulation and not with the haptic hardware. The current build of the software treats all bone as a single uniform material with a single stiffness value. The literature indicates that the elastic modulus of trabecular bone is 10 – 40% less than the elastic modulus of cortical bone. Furthermore, the elastic modulus varies based on the anatomical site [59-61].

The realism of the haptic surgical simulation can be improved by having a variable bone stiffness value. Depending on the 'locked location' of the drill burr (defined in Section 5.2.1.1), the bone stiffness value is decreased or increased. This should give end users the impression that the inner

trabecular bone is less stiff than the outer cortical bone. A simpler solution might be to modify the 3D model so that the temporal bone is distinctly made up of two types: cortical bone and trabecular bone and have two different stiffness values.

Another recommended improvement would be the addition of a modulating drill acoustic function. A study can also be built around this addition, examining whether or not users' perception of force is altered by the frequency of drill pitch that is heard.

Furthermore, the analysis of drilling profiles belonging to experts, trainees and novices can be examined. From this, error metrics can be designed. End users can know how long they took and how close were they from important blood vessels or nerves. These error metrics can be ascertained either in real time via embedded logics within the program or by examining recorded data after haptic drilling is complete.

Finally, to make the haptic simulation more immersive, the use of either virtual reality or augmented reality glasses can be utilized.

Appendix A – HD² Detailed Technical Specifications

The detailed manufacturer specifications of the HD^2 are shown in the following two tables.

Table 11: Detailed Specifications of the HD^2 [25]

Device Geometry:		
Force-Feedback Workspace At Operating Position:		
Translation Along X	± 0.520	m
Translation Along Y	200 – 550	mm
Translation Along Z	- 100 – 275	mm
Rotation About X (Roll)	± 90	°
Rotation About Y (Pitch)	± 90	°
Rotation About Y (Pitch)	Continuous	°
Maximum Output Force At Operating Position:		
Maximum Continuous Output Force Along X	10.84	N
Maximum Continuous Output Force Along Y	10.84	N
Maximum Continuous Output Force Along Z	7.67	N
Maximum Continuous Output Torque About X	0.948	N.m
Maximum Continuous Output Torque About Y	0.948	N.m
Maximum Continuous Output Torque About Z	0.948	N.m
Peak Output Force At Operating Position:		
Peak Output Force Along X	19.71	N
Peak Output Force Along Y	19.71	N
Peak Output Force Along Z	13.94	N
Peak Output Torque About X	1.72	N.m
Peak Output Torque About Y	1.72	N.m
Peak Output Torque About Z	1.72	N.m

Table 12: Detailed HD^2 Specifications (continued) [25]

Description	*Value*	*Unit*
DC Motors (Each)		
Motor Power Rating	90	W
Motor Torque Constant	0.115	N.m/A
Motor Peak Current	5	A
Linear Current Amplifiers (Each Channel):		
Linear Amplifier Peak Power	140	W
Linear Amplifier Maximum Continuous Current	3	A
Linear Amplifier Peak Current	5	A
Linear Amplifier Gain	1/(2.41)	A/V
Build-In Power Supplies (Each):		
Power Supply Power	315.9	W
Power Supply Voltage	27	VDC
Optical Encoders (Each):		
Encoder Line Count	1,000	lines/rev
Encoder Resolution (In Quadrature)	4,000	counts/rev
Encoder Sensitivity (In Quadrature)	0.0015	°/count
Encoder Type	TTL	
Encoder Signals	A, B, Index	
Current Sense:		
Current Sense Calibration At ±10%	0.5	V/A
Device Geometry:		
Motor Arm Length	0.280	m
Forearm Length	0.290	m
End-Effector Arm Length	0.175	m

Appendix B – Related Technical Work

This appendix section briefly looks at two research objectives that were not included in the main body of the thesis. The first is investigating the potential to use of the Quarc Visualization interface to create 3D graphics and its applications to surgical simulation. The 2nd objective was to carry out force testing or force validation of the HD^2.

B.1 Quarc Visualization

Quarc is the name of Quanser's Simulink Interface which allows programming and communication between Simulink and Quanser hardware devices. Besides being its own program, Quarc acts as an expansion or add-on onto Simulink giving end users access to Quanser specific Simulink blocks and code.

Figure B-1 below shows the Medtronic Drill being rendered in the Quarc 3D viewer.

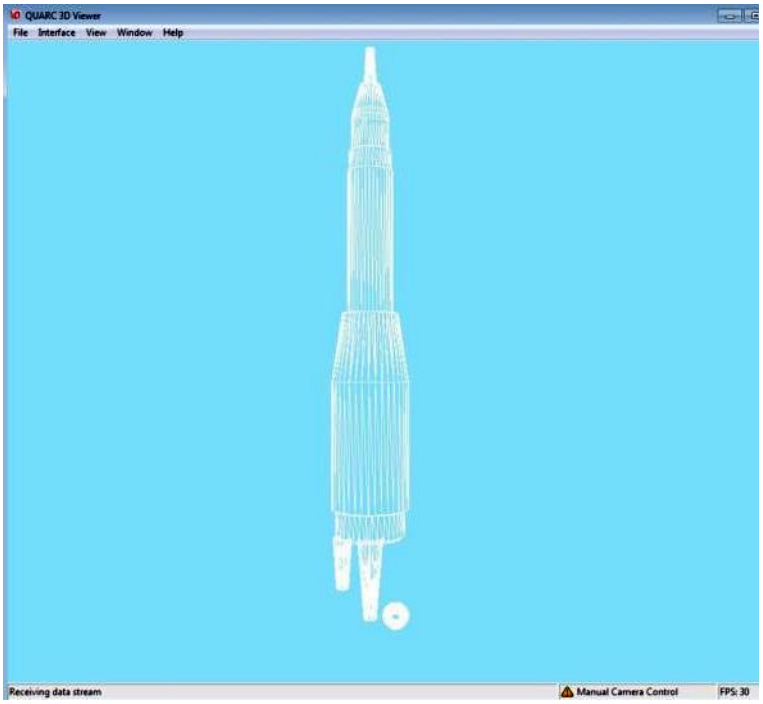

Figure B-1: Drill Model Loaded into the Quarc Visualization Interface

The pros and cons of this software are as follows:

Pros:

1. Relatively easy to use and Quanser provides an online tutorial for their 3D software
2. No complex object oriented programming is required. If a user can use Simulink, he or she can start creating 3D animations or 3D movable parts.

Cons:

1. Very limited in what a user can do. All objects, while movable, cannot be deformed, destroyed or modified in real time.

Due to the fact that objects are not destructible, the Quarc Visualization interface was deemed not suitable for temporal bone surgical simulation.

B.2 Force Validation

The benefit of force validation/testing would have been to verify or test the transformation carried out in changing the haptic interaction point. However, since the haptic device behaved as expected, force validation testing was deemed unnecessary.

To test the forces at the drill tip and at different points on or near the handle, a clamp was needed which would attach to the force sensor present in the lab. It was decided that the clamp should be able to clamp the drill tip, the handle itself, the upper and lower ends which hold the handle, and the ends of link 3 for both the upper and lower arms. See the Figure B-2 for reference below.

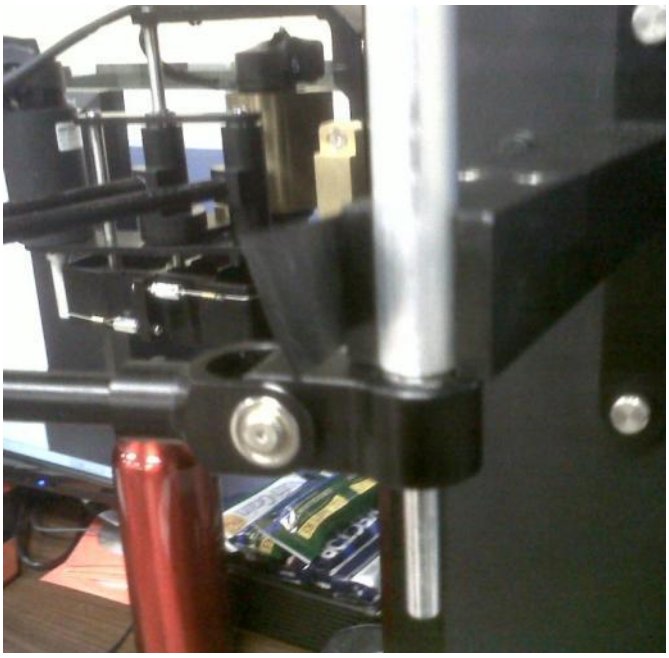

Figure B-2: HD^2 Handle Close-up

To meet those requirements, a clamp designed to hold onto both horizontal and vertical cylinders/ objects was conceptualized and designed in SolidWorks. The CAD model is shown in Figure B-3 in both the horizontal and vertical modes.

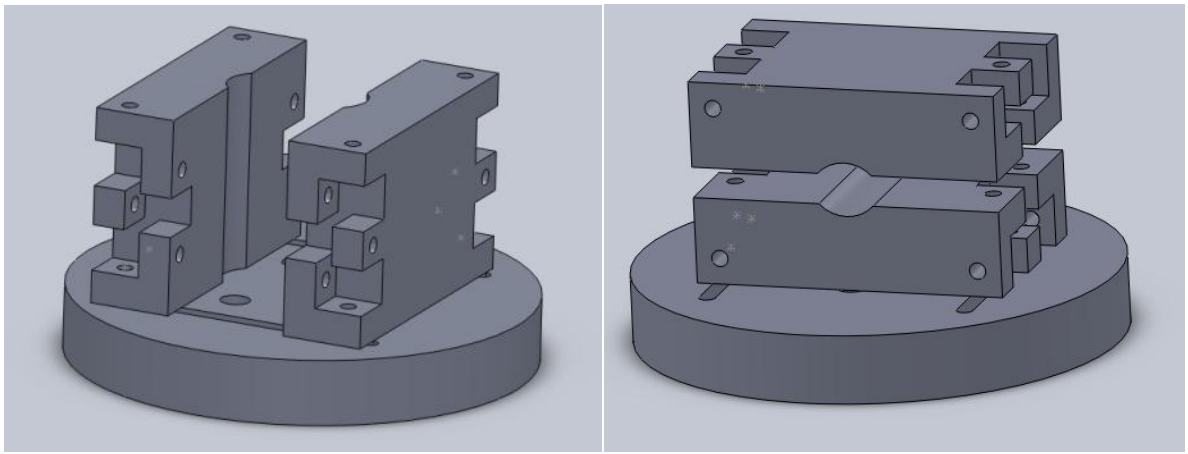

Figure B-3: Initial Clamp Design

The arc in between the two clamping pieces was carefully chosen to allow the pieces to clamp both objects as small as a drill bit, as large as the handle ends and all objects of other sizes within that range.

The two pieces are clamped into place via 10-32 screws with one sided being threaded. The blocks are allowed to slide along flat circular base along a groove. The pieces are restricted to movement along the grove lines via long bolts and nuts which passes through both the clamping pieces and the circular base.

For ease of machining and use, a simpler design was favored. The clamp was therefore redesigned. The CAD model and finished part is shown in Figure B-4 below.

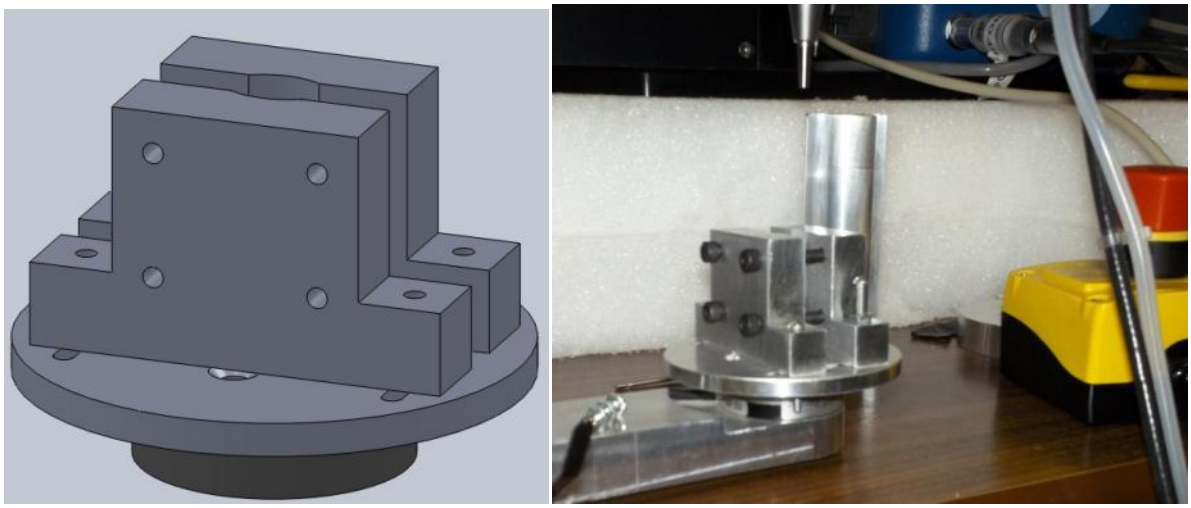

Figure B-4: Chosen Clamp Design

The Simulink driver was modified to take manual inputs from a user. However, the actual force testing/ validation never occurred and time was transferred to other tasks.

Appendix C – Driver Details

The first task was to understand and document the HD^2 Simulink driver. This appendix section gives a detailed overview on the calculations carried out by the driver.

C.1 HD^2 Driver – Upper Level Architecture

The driver logic is coded using Simulink and runs at 1000 Hz. Routines or functions are represented by blocks. These blocks are nested within one another. The highest level of the HD^2 driver is shown in Figure C-1.

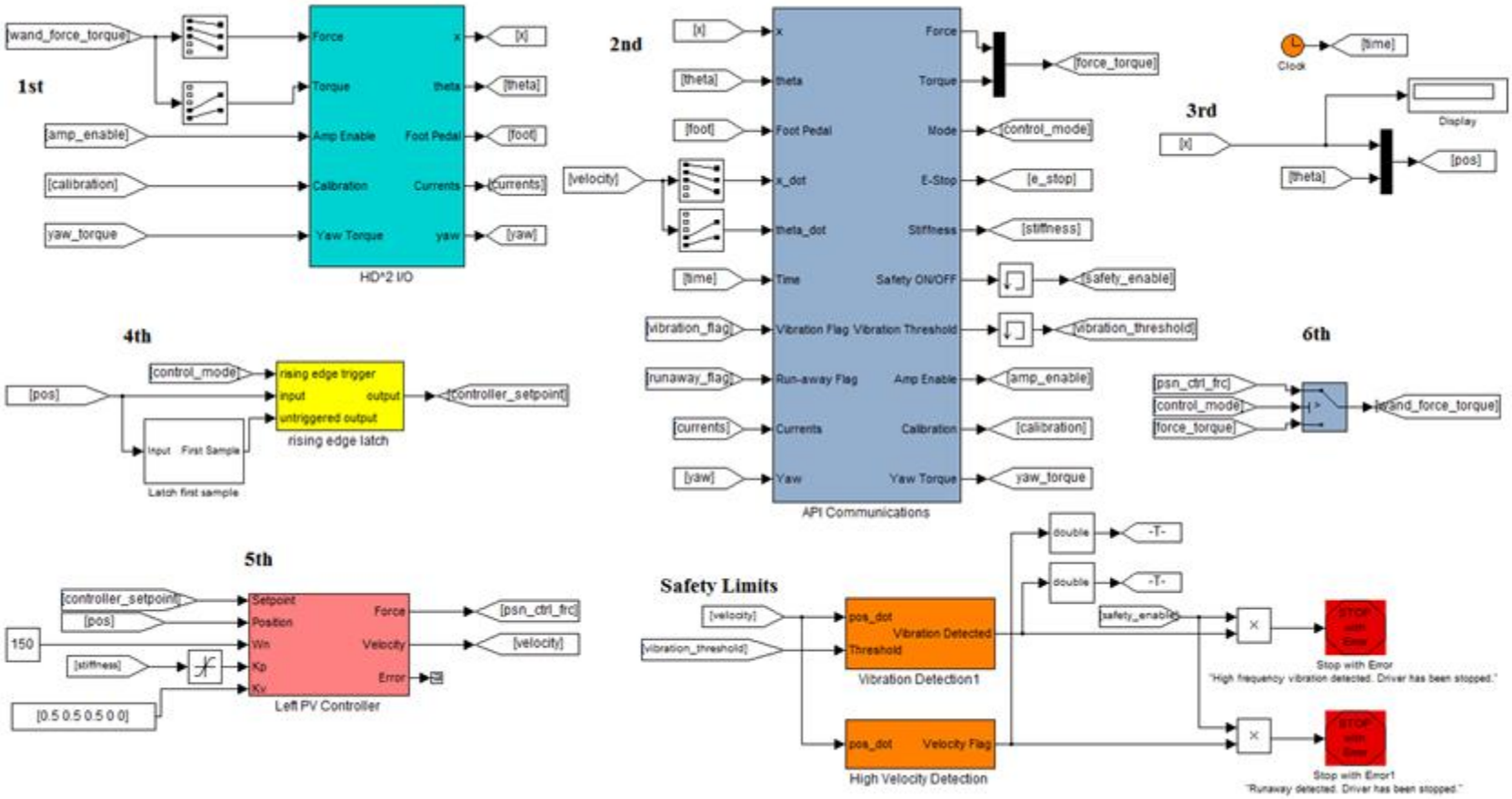

Figure C-1: Highest level in the Simulink model of the 'HD2_Driver' [62]

The order of execution of the above blocks is as follows:

1. 'HD^2 I/O (light blue)
2. API Communication (large dark blue)
3. Multiplexing [x] and [theta] into [pos]
4. Rising Edge Latch (Yellow)
5. Left PV Controller (Red)
6. A switch to output the wand_force_torque

The orange blocks are safety limits which will shut the HD^2 down if the vibration or velocity measurements exceed preset limits. The blocks execute continuously in a loop.

C.1.1 Overview of H2^2 I/O

A close up of the "HD^2 I/O" block is shown in Figure C-2 below.

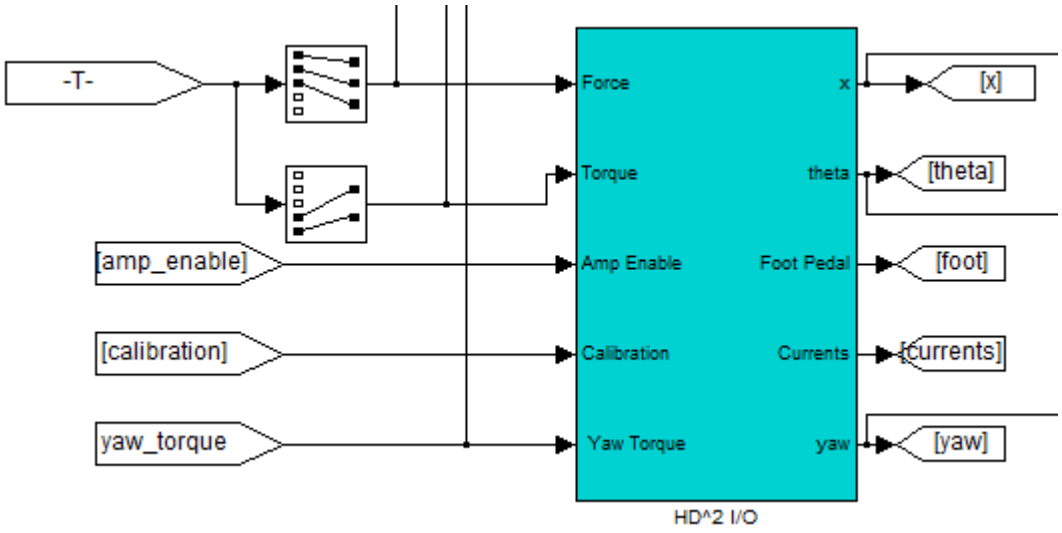

Figure C-2: 'HD^2 I/O' block [62]

The first block takes the present values for Force and Torque as input. Yaw Torque (about the local z-axis) is trivial as it is never used. Calibration and Amp enable are two flags that are used

to determine if the loop repeats again. If Amp Enable is disabled (zero), no voltage is sent to the motors of the HD^2. If Calibration flag is zero, the encoder counts are not updated.

The outputs from this block are wand position [x], orientation angles [theta], binary foot pedal states [foot], motor currents [currents] and the local yaw angle of the HD^2 end-effector wand.

This block talks to the hardware via read and write operations. It reads encoders as well as writes values out which dictates the voltages supplied to the HD2 motors. Additionally, all the kinematic and force equations are within this block.

For reference, the workspace co-ordinate system is shown in Figure C-3 below. Position, Force vectors and Torque vectors are defined with respect to this co-ordinate system. Yaw Torque (as seen in Figure C-2 refers to the local Z Torque about the handle.

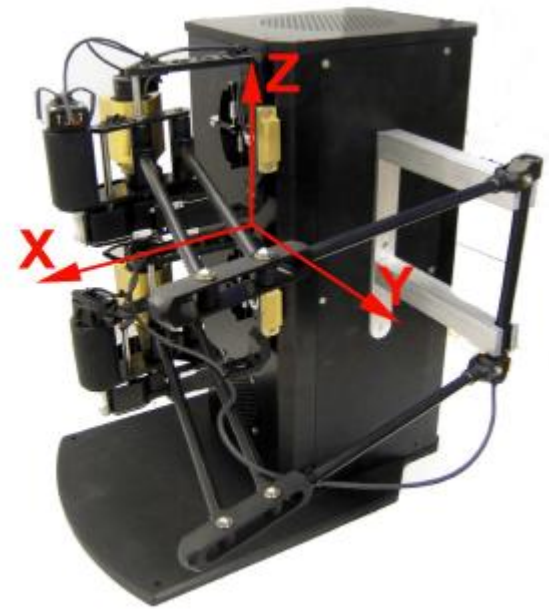

Figure C-3: Base co-ordinate reference frame of the HD^2 [25]

If one imagines a parallel co-ordinate reference frame sitting in the center of the haptic wand, the orientation angles would be defined as the angles created between the wand and the local X-Z and Y-Z planes. Another way of stating this is that θ_x and θ_y is the handle's rotation about the local X and Y directions.

This block is described in greater detail in Section C.2.

C.1.2 Overview of API Communications Block

The "API (Application Programming Interface) Communications" block is executed after the "HD^2 I/O" block. This block allows the HD^2 Driver to communicate with other software and is shown in Figure C-4.

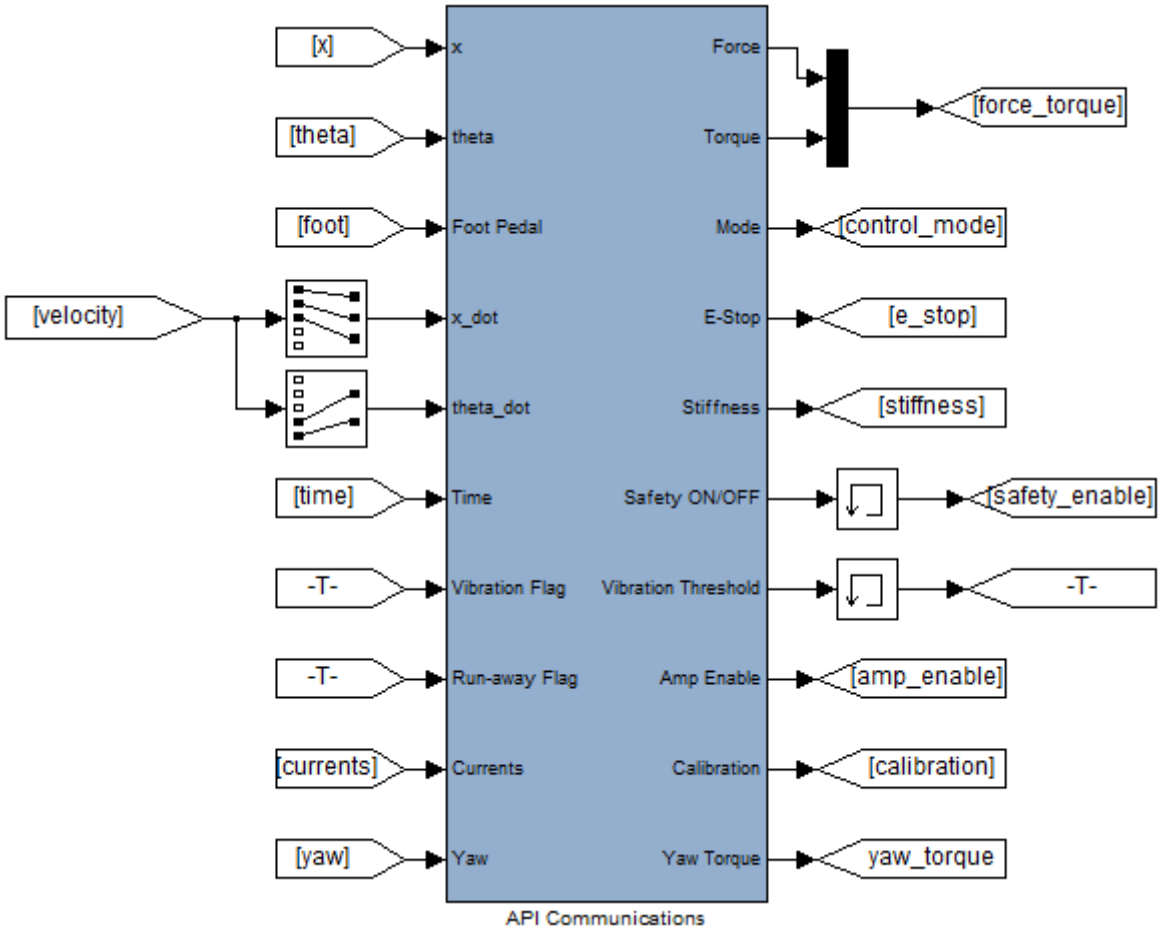

Figure C-4: API Communications block [62]

This block does no calculations nor does it read or write data from/to the HD2 hardware. The inside of this block is shown in Figure C-5.

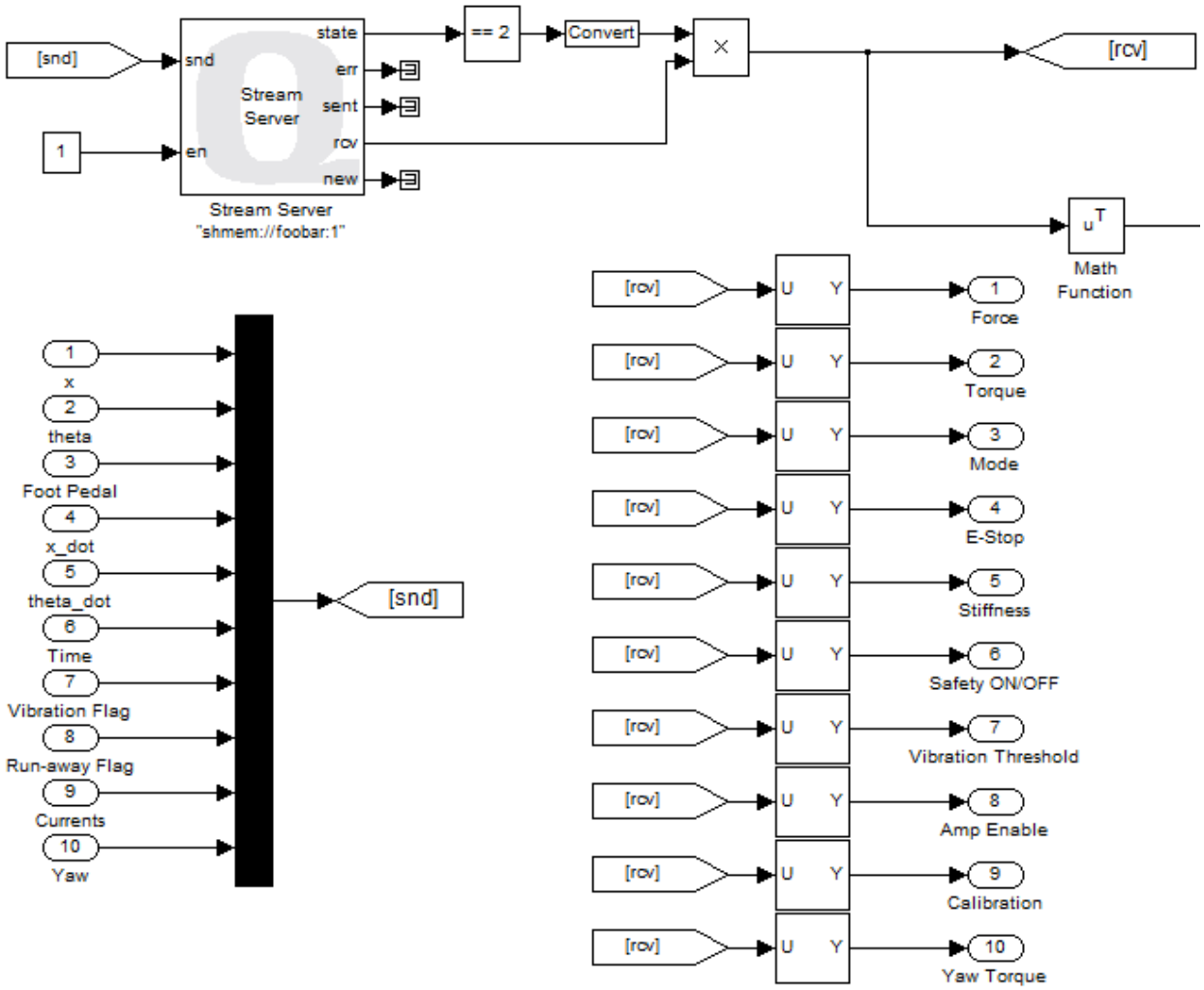

Figure C-5: Inside API Communication block [62]

When the 'Stream Server' QUARC block is in connection with a host, (i.e. a user C++ or MATLAB program), this block acts an API or Application Programming Interface. A user program is able to read the 22 values from the API block [9]. A user program may also write 18 values to the API block.

The labels seen entering the 'API Communication' from the left can be read by a user program in the form of a 22 element vector of doubles. The structure is shown in Figure C-6:

- [0:2] :end-effector x, y and z task space position in SI units.
- [3:4] :end-effector roll, pitch task space orientations in SI units.
- [5:6] :left and right pedal status of the supplied foot-pedal (1: pressed)
- [7:11] :end-effector x, y, z, roll, and pitch velocities in SI units.
- [12] :time elapsed since the driver was started in seconds.
- [13] :safety flag showing one of freedom degrees is vibrating with frequency higher than the allowed threshold.
- [14] :safety flag showing that one of the freedom degrees has acquired a velocity higher than the allowed threshold.
- [15:20] :current measured at each motor in Amps.
- [21] :end-effector rotation about its own axis in radians (yaw)

Figure C-6: Output variables from HD2 driver [27]

User programs can feed the following data into the driver. The format is an 18 element vector of doubles as shown in Figure C-7:

- [0:2] :force to be applied to the end-effector in x, y and z DOF's in SI units.
- [3:4] :torque to be applied to the end-effector in roll, pitch DOF's in SI units.
- [5] :control mode (0: force control, 1: position control)
- [6] :emergency stop flag. Raising this signal from 0 to 1 stops the driver.
- [7:11] :position control stiffness for the 5 freedom degrees in N/m for x, y and z and N.m/rad for roll and pitch.
- [12] :safety enable flag with default value of 1. When this value is 0 if a safety flag is raised, the driver will continue running. When the value is set to 1 if a safety flag raises the driver will stop automatically.
- [13] :vibration safety threshold. This sets the number of vibrations the device is allowed to have in each of its freedom degrees in 1.9 seconds before the vibration safety flag is raised. The default is 40.
- [14:15] :amp enable signal. [1 0] enables the amps while any other configuration keeps them disabled.
- [16] :calibration flag. Raising this flag will reset the encoder counts on the Q8 DAQ.
- [17] : torque to be applied to the end-effector in Yaw DOF in SI units.

Figure C-7: Input variables to the HD2 driver [27]

These user inputs become outputs from the API block. These outputs are used by other blocks within the driver.

C.1.3 HD^2 Position

A new signal is created called [pos] and is shown in Figure C-8 below. This is simply the multiplexing of the [x] and [theta] signals. These two signals are the output from the light blue HD^2 I/O block.

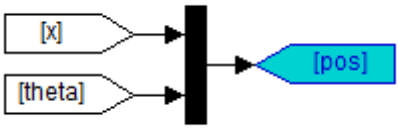

Figure C-8: HD^2 position and orientation [62]

C.1.4 Rising Edge Trigger

The fourth executed block is shown in Figure C-9.

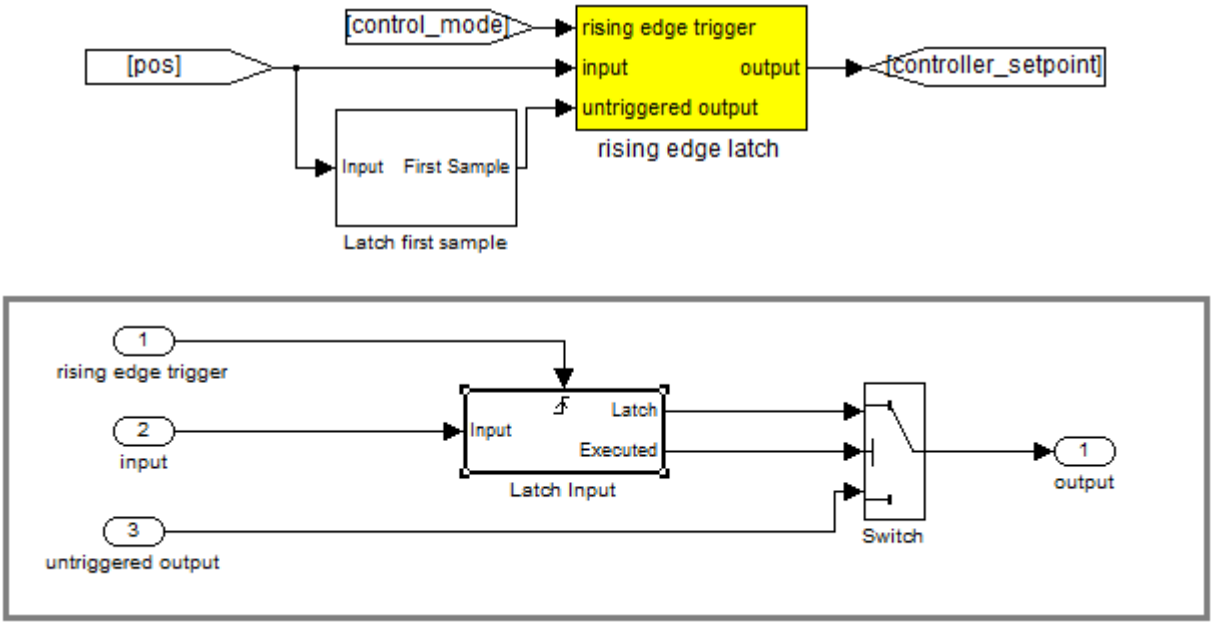

Figure C-9: Rising edge trigger and its inner block diagram [62]

The signal [pos] has the x, y and z positions of the middle of the wand as well as the angles of pitch and roll.

The block outputs [pos] under a new label, [controller_setpoint] whenever the switch is triggered. The updated output value is held until it is replaced by a newer [pos] value when the rising edge is re-triggered. [controller_setpoint] is used as an input to the position-velocity controller (Section C.1.5).

The signal which triggers the switch is [control_mode] which is an output from the API Communications block. This happens when [control_mode] has a value of 1. This occurs when the control mode is operating on position control. If the control mode is on force control, this latch is never activated and the set point is never updated.

C.1.5 Locked Position Controller

This block is essentially a position-velocity controller and is shown in Figure C-10.

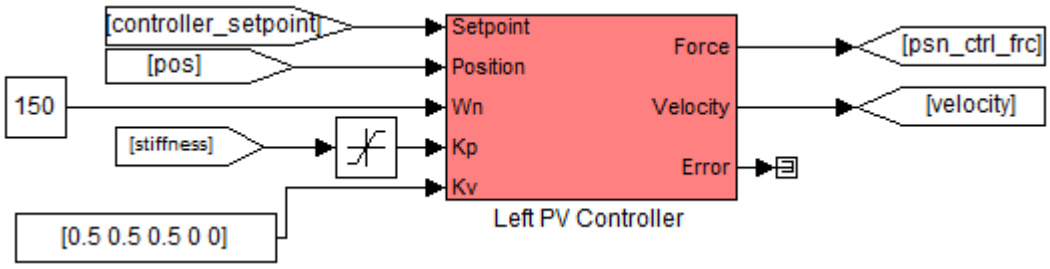

Figure C-10: Position controller [62]

[controller_setpoint] is the output to the yellow 'rising edge latch' block. [pos] is the current measured position in terms of x, y, z, roll and pitch. [stiffness] is a user defined parameter and is sourced from the API Communications block. These stiffness values are limited between [2500 2500 2500 4 4] and -1*[2500 2500 2500 4 4] and become the Kp controller values for x, y, z, roll and pitch. The Kv values are set at 0.5 and 0 for position and angle respectively.

Figure C-11 shows the inside of the PV controller.

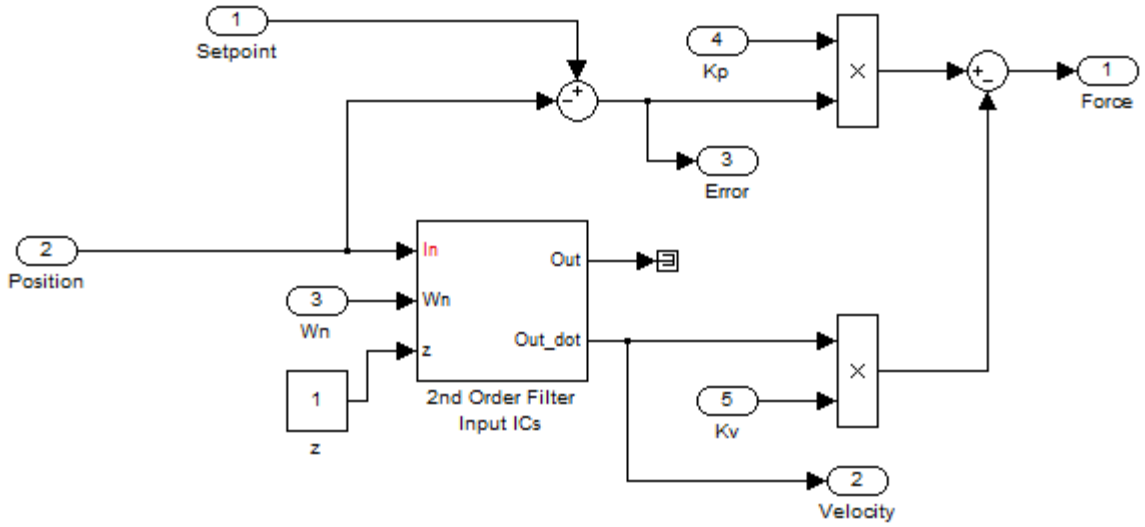

Figure C-11: Inside the position-velocity controller [62]

This block outputs the filtered velocity and a parameter called [psn_ctrl_frc]. The velocity, v_f, is obtained using a 2nd order filter of the raw position.

[psn_ctrl_frc] is equal to:

$$psn_{ctrl_{frc}} = k_p(position\ error) - k_v * v_f \quad \text{(C-1)}$$

Where,

position error = set point – measured position

v_f is the filtered velocity

It is important to note that this block has no effect when the control mode is in Force Control Mode. For the purposes of this haptic simulation project, only Force Control Mode is used.

It is also important to note that this position controller does not perform trajectory position control. It simply locks the HD2 wand in its current position. Inverse kinematics and trajectory planning is outside the immediate short term goal of achieving Haptic Surgical Simulation with the HD2. It is however, a useful future project to pursue once more immediate goals are achieved.

C.1.6 Control Mode Switch

The switch responsible for setting the HD^2 to either force control or position control is shown in Figure C-12.

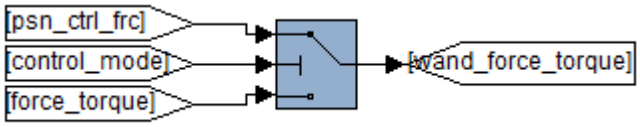

Figure C-12: Wand_Force_Torque output from control switch [62]

The default value for [control_mode] is zero which results in the haptic being in Force Control Mode, having [force_torque] as the output. [force_torque] comes directly from the API Communications block.

If the control mode has a value of 1, the HD^2 is being used in Position Control and the [wand_force_torque] takes the value of [psn_ctrl_frc] which comes from the PV Controller block. [wand_force_torque] (re-labelled as [-T-] in Figure C-13) is the major input to the HD^2 I/O block.

C.2 HD^2 I/O Block – Original

This block reads encoder values from the hardware, writes voltage values to the motors, and calculates inverse and forward kinematics. The inputs and outputs to the block is re-shown in Figure C-13 below. It is important to note that the Force is defined in terms of x, y and z components and the Torque is defined in roll (rotation about x) and pitch (rotation about y) components.

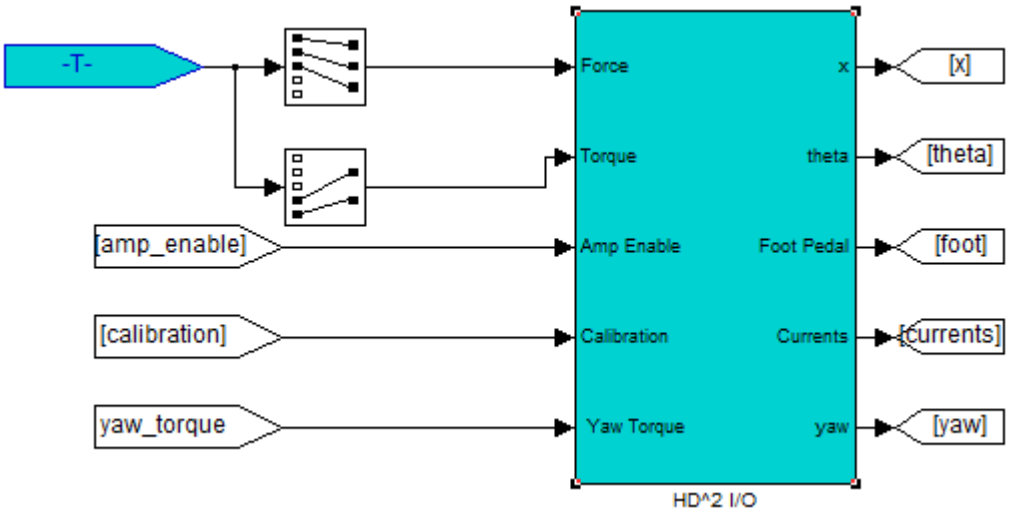

Figure C-13: HD^2 I/O block [62]

This block has numerous operations taking place within it. This section will accurately present the flow of calculation as clearly as possible.

C.2.1 Hardware Inputs from the HD^2

Figure C-14 shows blocks responsible for reading the hardware inputs and Forward Kinematics.

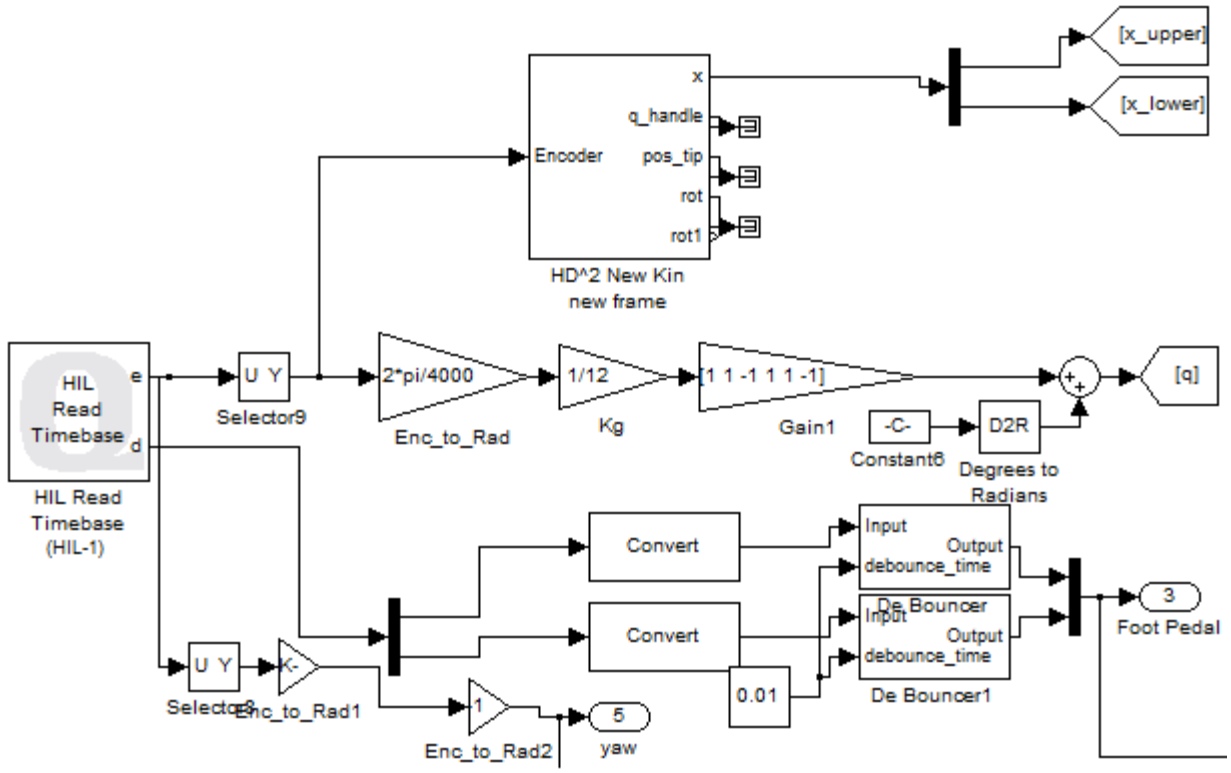

Figure C-14: Reading the encoders and calculating forward kinematics [62]

The HIL Read Timebase block reads the encoder values from the HD2 hardware as well as the digital on/off values for the HD^2 foot pedal. The encoder assignment is shown in Figure C-15. It is important to note that the encoders are located on the motors and not on the actual joints [25]. Furthermore, there is a 7th encoder value being read into the HD^2 driver. This is the yaw of the wand.

Q8 Encoder Input Channel	*Motor/Joint ID Used in Modeling*
EI #0	Shoulder: Motor #1
EI #1	Shoulder: Motor #2
EI #2	Base: Motor #3
EI #3	Shoulder: Motor #4
EI #4	Shoulder: Motor #5
EI #5	Base: Motor #6

Figure C-15: Encoder assignment [25]

From Figure C-14, the output [q] gives the motor (joint) angles in radians. The encoder values are converted to radians by the constant 2π/48000. A column vector of constants [-C-] is added to the encoder angle. These are the default angles (in degrees) when the program starts: [-104 18 -2 -104 18 2]. The angle values are reset to this when the calibration program is run. This should only be done when the wand is in the home position.

The HIL Read Timebase Quarc block also reads digital inputs for two pedals. The debounce block simply uses a counter to ensure that the pedal has been suppressed for a particular number of counts before it is registered as a switch between a signal of 0 or 1.

C.2.2 Forward Kinematics of the Original Driver

The block "HD^2 New Kin new frame" conducts a forward kinematic calculation to determine the positions [x_upper] and [x_lower] using joint angles, [q]. [x_upper] and [x_lower] are the co-ordinate positions for the top and bottom of the wand respectively.

The inside of the sub-block is shown in Figure C-16.

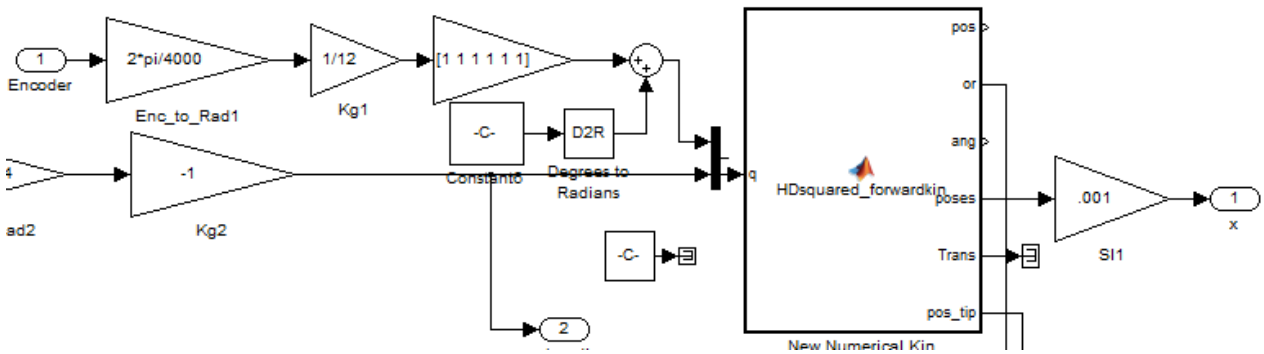

Figure C-16: Inside 'HD^2 New Kin new frame' block [62]

The block New Numerical Kin carries out the forward kinematic calculation. It uses the Newton's Method Iteration to find alpha_u and theta_u (for the upper arm) and alpha_l and theta_l (for the lower arm). This was necessary since there were no encoders on the end effectors of the upper and lower arms.

Using these calculated orientation angles; the upper and lower arm positions can be calculated. The full derivation of this iterative formulation can be found in Quanser's Maple Worksheet, "HD2 Kinematics and Dynamics" [28].

C.2.3 Converting Torque into Upper and Lower Force Components

Figure C-17 shows the blocks which convert the desired roll and pitch torques into upper and lower force components. These roll and pitch torques refer to the torques felt by the user about the end-effector haptic handle.

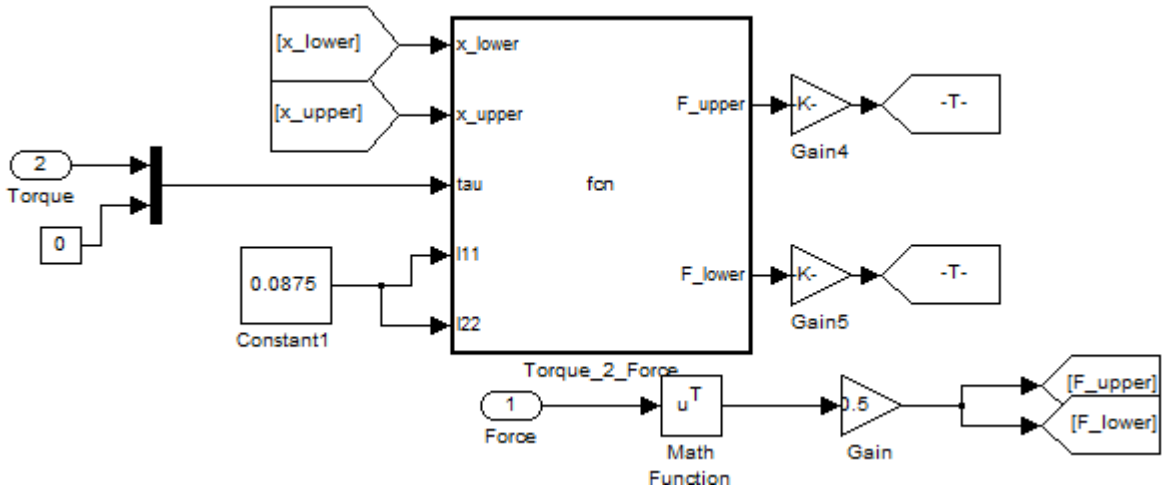

Figure C-17: Blocks responsible for calculating required upper and lower arm forces [62]

The input Force (which is defined in x, y and z components) is divided evenly between the F_{upper} and F_{lower}. (This is the force that is generated at the top and bottom of the wand respectively.)

$$F_{upper} = F_{lower} = \frac{1}{2} Force \tag{C-2}$$

The roll and pitch torques, together with the co-ordinates of the top and bottom of the wand is fed to a code block called, 'Torque_2_Force'. The inputs to this block can be seen in Figure C-22. The sequence of calculation is shown below:

$$Length, L = \sqrt{(x_{u1} - x_{l1})^2 + (x_{u2} - x_{l2})^2 + (x_{u3} - x_{l3})^2} \tag{C-3}$$

$$unit\ vector, u = \begin{bmatrix} x_{u1} - x_{l1} \\ x_{u2} - x_{l2} \\ x_{u3} - x_{l3} \end{bmatrix} \frac{1}{L} \tag{C-4}$$

$$F_{upper\_tau} = \left(\begin{bmatrix} T_{\theta x} \\ T_{\theta y} \\ 0 \end{bmatrix} \times [u] \right) \frac{1}{2\, l_1} \tag{C-5}$$

$$F_{lower\_tau} = -\left(\begin{bmatrix} T_{\theta x} \\ T_{\theta y} \\ 0 \end{bmatrix} \times [unit\ vector) \right) \frac{1}{2\, l_2} \tag{C-6}$$

Note: $l_{11} = l_{22} = l_1 = l_2 = 0.0875$

C.2.3.1 Issues with Quanser's Torque Calculations

Following the above math, it is clear that the Forces and Torques all use vectors with respect to the global reference frame. Therefore it is a serious error that the Yaw Torque is forced to zero. Therefore, the couple forces computed will be incorrect.

Furthermore, any torque that is not completely perpendicular to the haptic wand will not be correctly simulated. (Since this block does not take into account the local Yaw Torque.)

According to Quanser, l_{11} and l_{22} should be manipulated to create a torque about any point other than the wand center. This claim however was incorrect since manipulating L11 and L22 will result in an impure torque (i.e. a torque plus a force).

Section 3.3 shows a correctly re-derived Torque to Couple Force formulation.

Due to this flaw discovered in the driver, a newer I/O Block was provided by Quanser and is detailed in Appendix Section C.3.

C.2.4 Output of Position of Wand Center

The center of the wand is easily found by finding the average between the upper and lower positions. This is shown in Figure C-18 below.

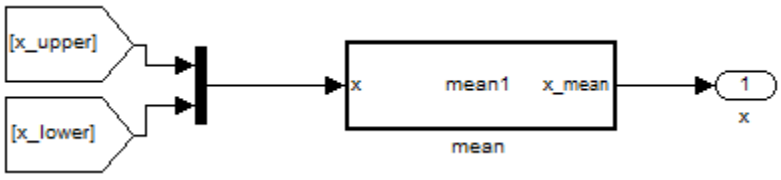

Figure C-18: Center of the haptic wand [62]

The co-ordinate frame is reproduced once more in Figure C-19 for convenience above. It is important to note that the base co-ordinate frame is located exactly between the upper and lower arms.

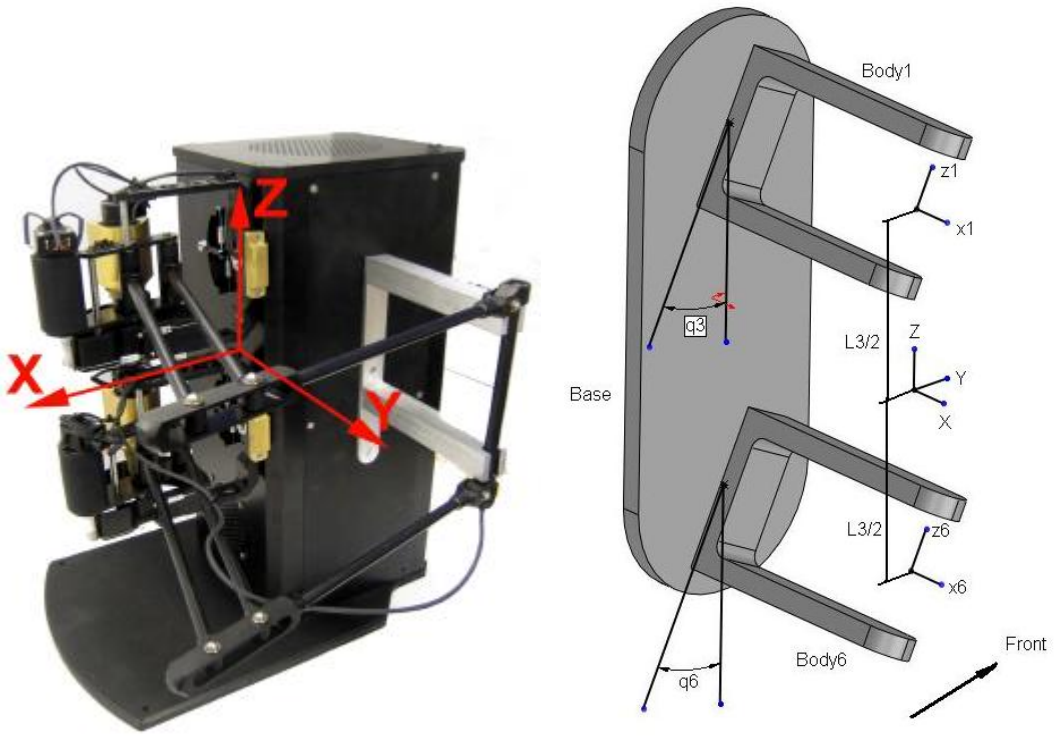

Figure C-19: The base co-ordinate system of the HD^2 [25, 28]

C.2.5 Calculating Wand Orientation Angles

The wand orientation is defined by two angles, θ_x and θ_y respectively. The block which calculates these angles is shown in Figure C-20 below.

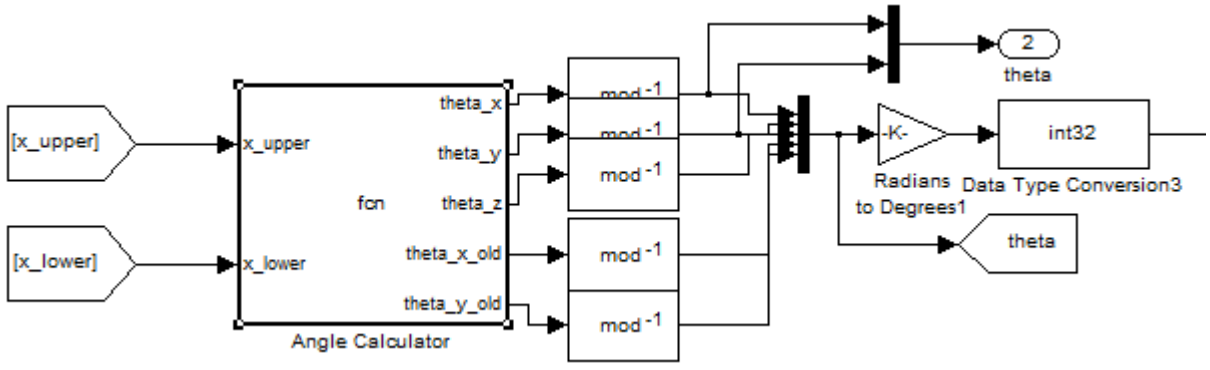

Figure C-20: 'Angle Calculator' block [62]

The important equations relevant to the orientation angles are shown below:

$$d = x_{upper} - x_{lower} \tag{C-7}$$

$$dx = d(1), \qquad dy = d(2), \qquad dz = d(3) \tag{C-8}$$

$$L = \sqrt{d(1)^2 + d(2)^2 + d(3)^2} \tag{C-9}$$

$$\theta y = \sin^{-1}\left(\frac{dx}{L}\right) \tag{C-10}$$

$$\theta x = -\sin^{-1}\left(\frac{dy}{L\cos(\theta y)}\right) \tag{C-11}$$

To understand these angles, assume that there is a reference co-ordinate system passing through the center of the wand which is parallel to the home reference system. The local home position is with the wand along this local Z axis. Angle, θ_y is the handle's rotation about the Y axis relative to the home position. Angle, θ_x is rotation of the handle relative to the X axis. These two angles are shown below in Figure C-21.

Based on the angle definitions, the rotation matrix, $Rot_x(\theta_x)\ Rot_y(\theta_y)$ defines a 3x3 orientation matrix for the HD^2 handle. This matrix finds application in Sections 3.3 and 3.5.

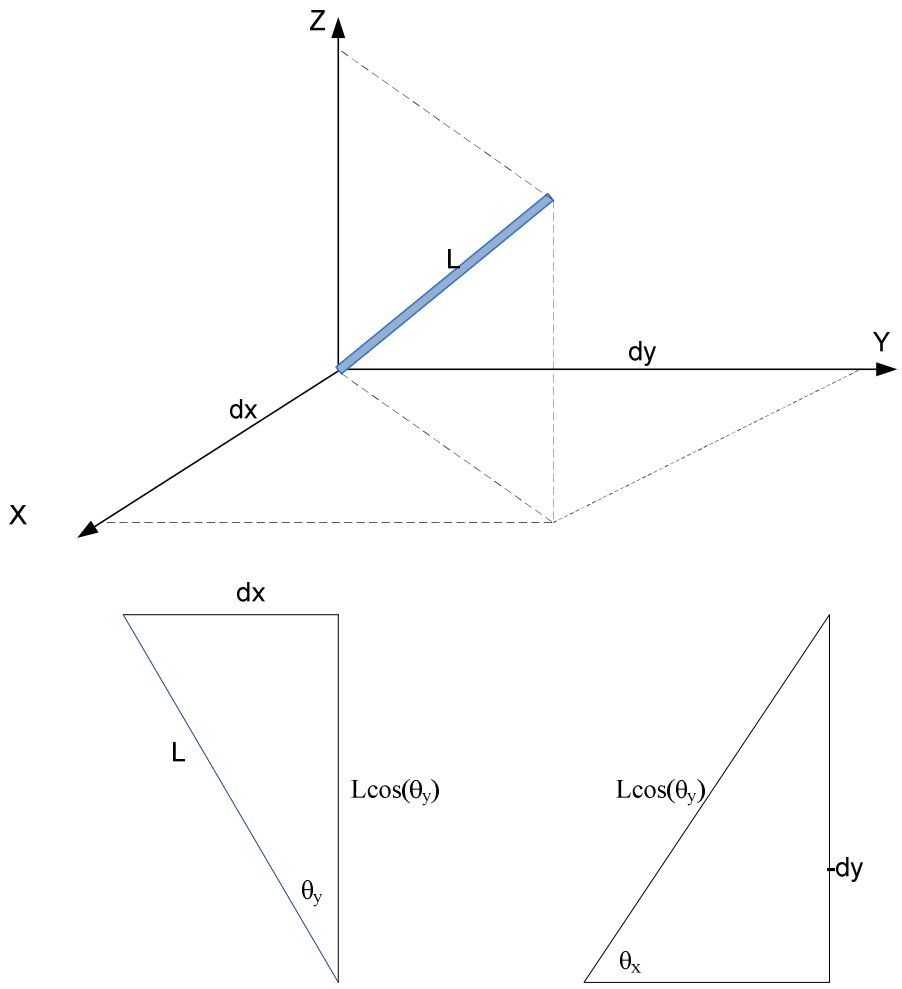

Figure C-21: Haptic wand and orientation angles [62]

C.2.6 Calculating Joint Torques

Figure C-22 shows the Simulink blocks responsible for calculating the haptic joint torques.

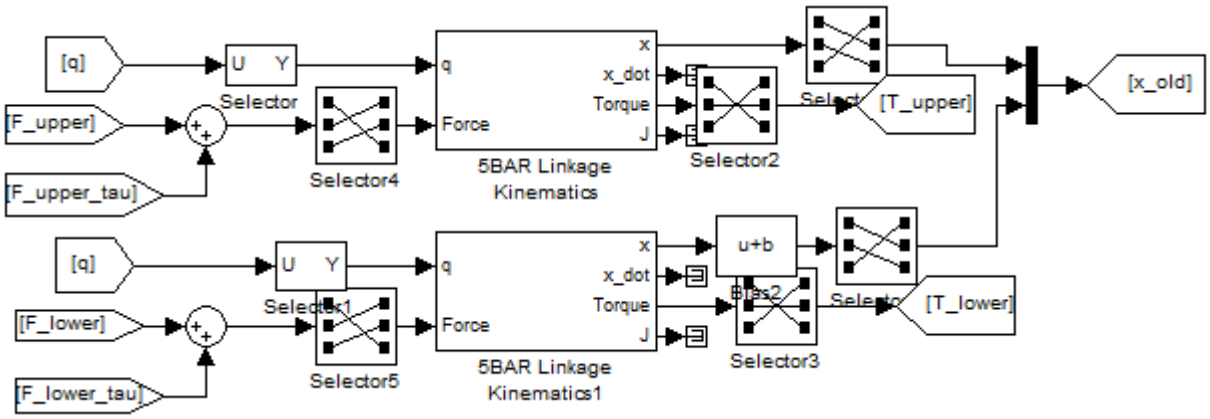

Figure C-22: Calculating joint torques [62]

F_upper and F_upper_tau is summed together to give the total force required at the top of the wand. The same is done at the bottom of the wand. The summed force is a column vector with 3 components. These components are re-organized as [2, 3, 1] before entering the block '5BAR Linkage Kinematics'.

The input, [q] is the angles of Motors 1 through 6. Motors 1, 2 and 3 determine the co-ordinate position for the top of the wand, while motors 4 – 6 determine the co-ordinate position of the bottom of the wand.

The exact same blocks (and therefore calculations) are done to determine the Jacobians and Torques for both the Upper and Lower arms.

Figure C-23 shows the internal blocks which are contained within the block, '5Bar Linkage Kinematics'.

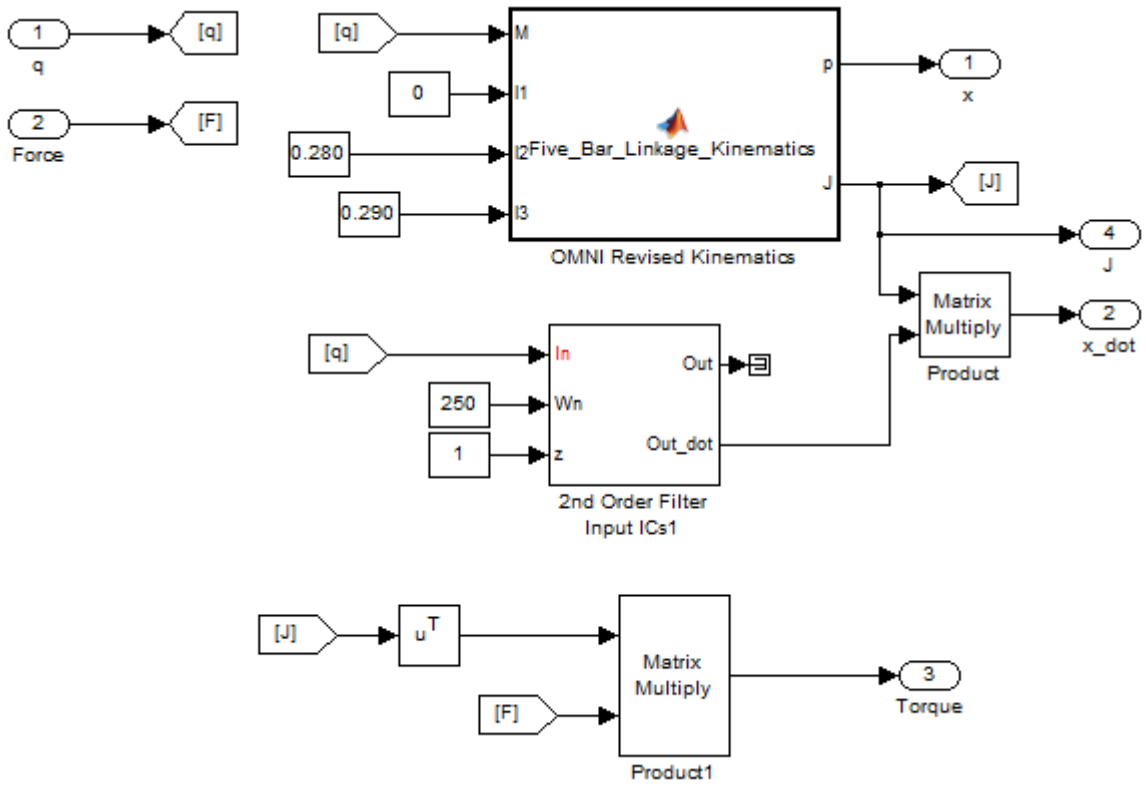

Figure C-23: '5Bar Linkage Kinematics' block [62]

Inside this block is a MATLAB code block called Five_Bar_Linkage_Kinematics. This code block calculates the Jacobian Matrix of the 3-link arm and is shown below:

```
function [p, J] = Five_Bar_Linkage_Kinematics (M, l1, l2, l3)
R = l2*cos(M(2)) + l3*cos(M(3));
x = cos(M(1))*R;
y = sin(M(1))*R;
z = l2*sin(M(2)) + l3*sin(M(3)) + l1;
p = [x y z]';
J1 = [-sin(M(1))*R      -cos(M(1))*l2*sin(M(2))     -cos(M(1))*l3*sin(M(3))];
J2 = [cos(M(1))*R       -sin(M(1))*l2*sin(M(2))     -sin(M(1))*l3*sin(M(3))];
J3 = [0                 l2*cos(M(2))                l3*cos(M(3))];
J = [J1;J2;J3];
end
```

The 2nd order filter is set at a frequency of 250 rad/s and simply filters the raw values for the encoder angles, [q]. However, this output is not utilized. What is extracted from the 2nd order filter, is the rate of change of [q], i.e. the angular velocity,$[\dot{q}]$.

Multiplying the Jacobian by the angular velocity gives x_dot or the linear velocity. This is not used outside the 5Bar Linkage Kinematic Block.

Multiplying the Jacobian by the Force, gives the Joint Torques, τ.

$$\tau = J^T F \tag{C-12}$$

This is a useful result which is used later in determining the necessary currents to feed the motors.

The x, y and z co-ordinates are calculated simply by using the relationship between geometry and the angles of the three motors. The position is saved as [x_old] and is not used anywhere else in the driver program.

It is important to note that [x_old] gives the positions of the ends of the upper and lower arm before the universal joints of the linking haptic wand. Therefore it is not the true end positions.

C.2.7 Converting Joint Torques to Motor Voltage

These blocks carry out a proportional conversion.

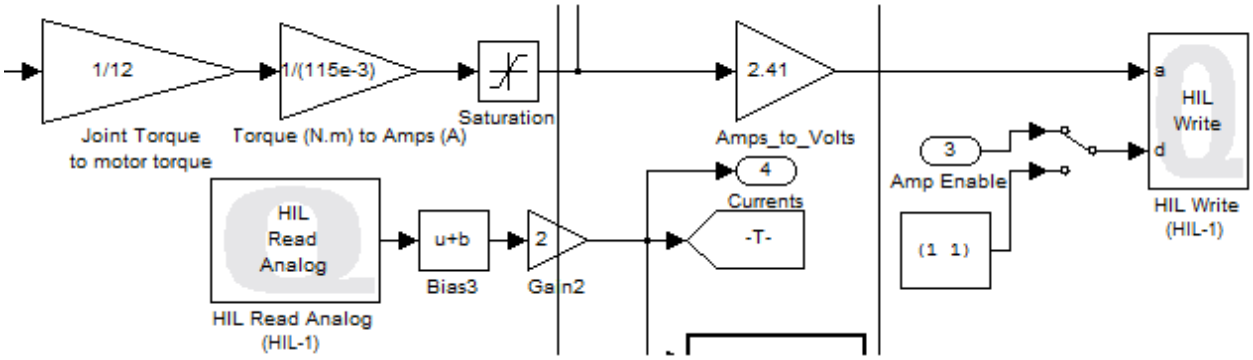

Figure C-24: Converting joint torque to motor voltage [62]

The overall conversion is shown below:

$$Motor\ Voltage = \frac{1}{12}\left(\frac{1}{0.115}\right)(2.41)\tau_q \tag{C-13}$$

The HIL Write block allows the voltage value to be written (outputted) to the hardware. The HIL Read Analog block reads the motor currents from the hardware. 'Currents' is one of the outputs of the HD^2 I/O block.

C.3 Updated I/O Block

Due to discovered errors in the original HD^2 IO block, a replacement was provided by Quanser. See Figure C-25 below. A noted difference is that the Torque Vector input now includes the Z component as well. The Yaw torque refers to an additional local Z torque along the handle.

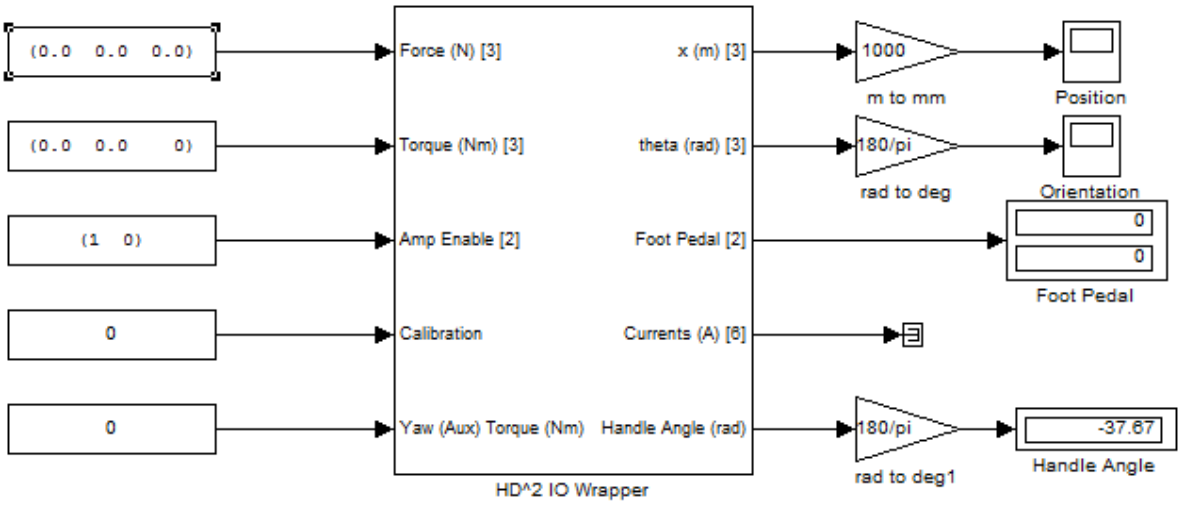

Figure C-25: Replacement 'HD^2 IO' Simulink block [62]

The inside of the block is shown in Figure C-26. The green (1st) section highlights the inputs and outputs to this inner block. This section also has the calibration block. This sets the calibration ticks to zero when the driver is first started.

The pink (2nd) and red (3rd) sections highlight the blocks responsible for reading values from the HD^2 hardware. The block, "HD2 Sensor Output" converts encoder ticks to joint angles in radians. One important difference in this block is that the encoder array is read from the hardware in a different order. Previously, Encoder 1 was associated with Motor 1 and so on. The values are rearranged such that the encoder's position in the array is in the order of the joints going from the base towards the end effector.

The blue (4^{th}) section highlights the major block which carries out calculations for the Forward Kinematics, Torque to Force Couple Calculation, Jacobian and Joint Torques.

The yellow (5^{th}) section converts the Joint Torques into Motor Voltage commands and outputs them to the HD^2 hardware. For both sensor input and motor voltage output, the previously defined conversion constants from the Original HD^2 I/O block are used.

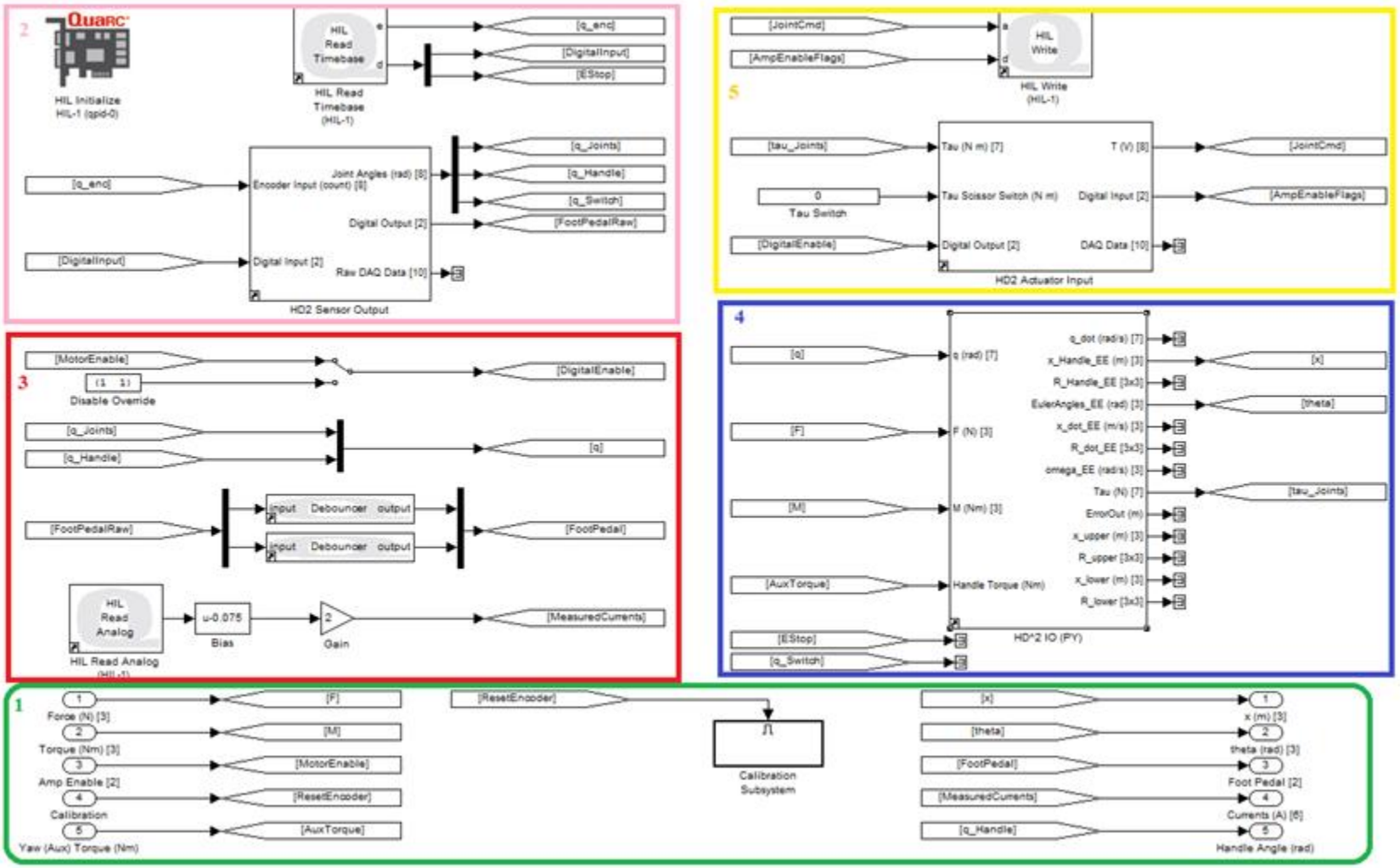

Figure C-26: Inside new IO block [62]

C.3.1 HD² Positive Y Configuration Block

The majority of the mathematical changes lie within the "HD^2 IO (PY)" block. This driver is built around the assumption that the 'home position' occurs when linkages 2 and 3 point in the Y axis direction. See Figure C-27 below.

In this orientation the joint angles are all zero. It is important to note that this Y-axis oriented position is theoretical as the linkages 2 and 3 can never be parallel due to mechanical constraints.

It is also important to note that θ_2 and θ_3 is defined with respect to global Y-axis direction and not to the previous link.

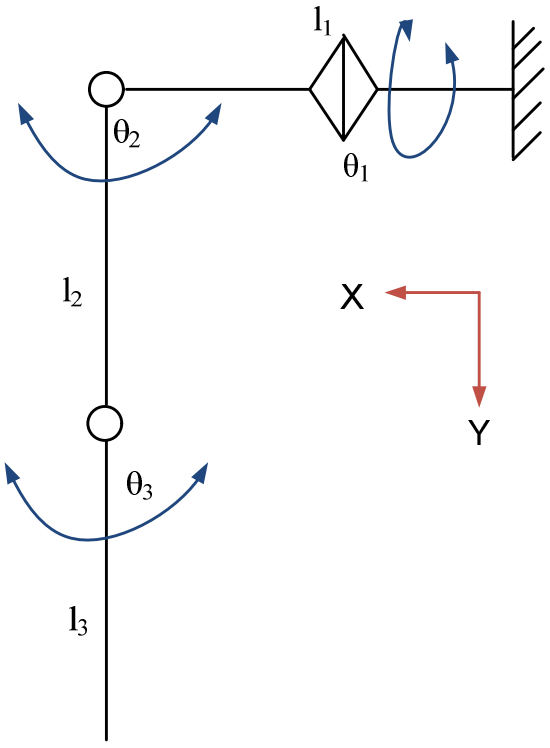

Figure C-27: Theoretical home position of upper and lower arms

Just as before, the desired Force at the center of the handle is split in half and distributed to both the top and bottom kinematics chains. This is shown in Figure C-28 below.

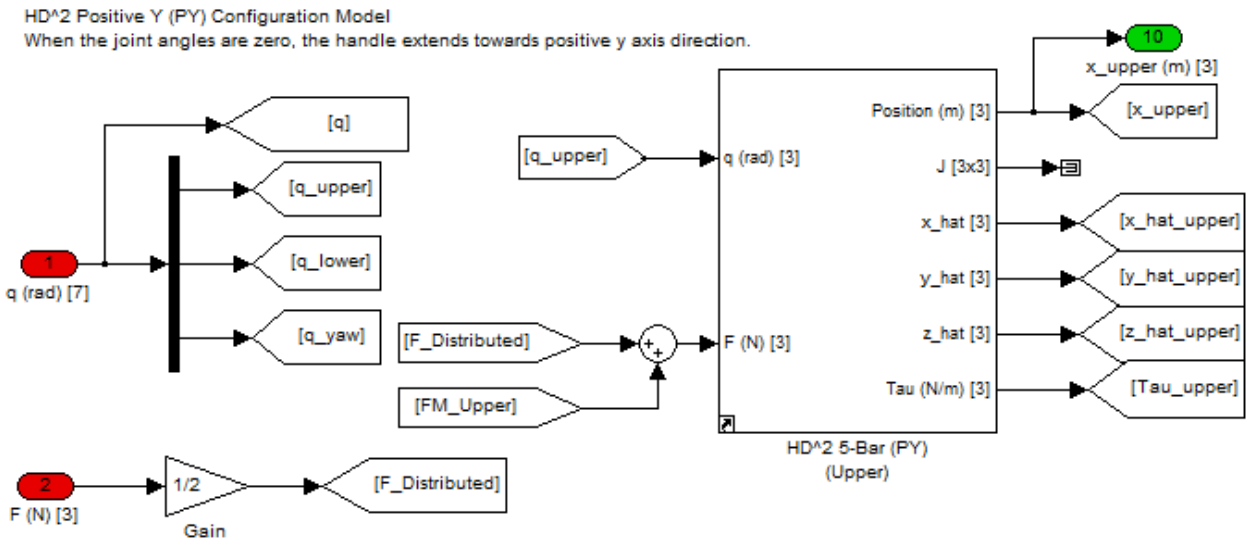

Figure C-28: Implementation of forward kinematics and Jacobian calculation [62]

Figure C-28 also shows "HD^2 5-Bar (PY)", the sub-block responsible for the Forward Kinematics and Joint Torques calculation. There are two copies of this block; one used for the Upper Kinematic chain and the second used for the Lower. This block takes as its input a Force vector and three joint angles. The Force vector is a summation of the desired Force plus a Force couple. The force couple is the result of decomposing a desired Torque into upper and lower couple forces and is an output from the "HD^2 Moment Projection" block.

Figure C-29 shows Links 2 and 3, which combined, make up a 5-bar mechanism. (The base bar not shown is Link 1.) This is important to note the name "5-Bar end effector frame" does NOT refer to the Handle's frame but instead it refers to the frame seen at the end of Link 3.

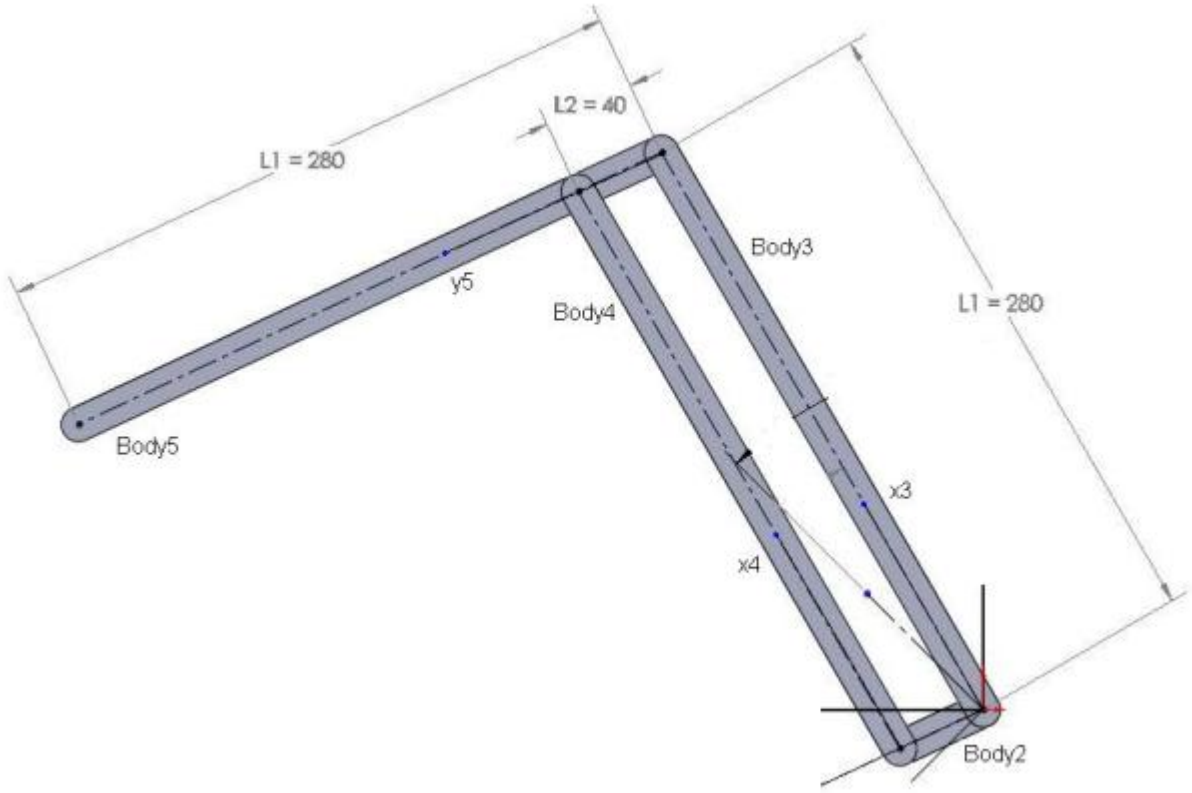

Figure C-29: 5-bar mechanism which makes up links 2 and 3 [28]

C.3.2 Forward Kinematics and Joint Torques

In this 'Positive Y Configuration', the joint angle is the angle between the link and the global Y direction. Therefore, the DH convention was not used to derive the Forward Kinematics. Instead the Forward Kinematics is derived as a function of geometry.

Equations (C-12) to (C-16) give the Forward Kinematic equations found in the "HD^2 5-Bar (PY)" block. Equation (C-16) takes into account the Z offset for the upper and lower arm cases.

$$x = L1 - L2\sin(\theta_2) - L3\sin(\theta_3) \tag{C-14}$$

$$y = L2\cos(\theta_1)\cos(\theta_2) + L3\cos(\theta_1)\cos(\theta_3) \tag{C-15}$$

$$z = L2\sin(\theta_1)\cos(\theta_2) + L3\sin(\theta_1)\cos(\theta_3) \pm h/2 \tag{C-16}$$

$$J = \begin{bmatrix} \partial x/\partial\theta_1 & \partial x/\partial\theta_2 & \partial x/\partial\theta_3 \\ \partial y/\partial\theta_1 & \partial y/\partial\theta_2 & \partial y/\partial\theta_3 \\ \partial z/\partial\theta_1 & \partial z/\partial\theta_2 & \partial z/\partial\theta_3 \end{bmatrix} \tag{C-17}$$

$$J = \begin{bmatrix} 0 & -L2\ c_{\theta 2} & -L3\ c_{\theta 3} \\ -L2\ s_{\theta 1}c_{\theta 2} - L3\ s_{\theta 1}c_{\theta 3} & -L2\ c_{\theta 1}s_{\theta 2} & -L3\ c_{\theta 1}s_{\theta 3} \\ L2\ c_{\theta 1}c_{\theta 2} + L3\ c_{\theta 1}c_{\theta 3} & -L2\ s_{\theta 1}s_{\theta 2} & -L3\ s_{\theta 1}s_{\theta 3} \end{bmatrix} \tag{C-18}$$

$$\tau = J^T\ {}^0_eF \tag{C-19}$$

Where,

J is the Jacobian matrix

F is the desired force at the end of the Link 3 with respect to global co-ordinates

τ is a 3x1 matrix of Joint Torques for Links 1, 2 and 3

Equations (C-17) to (C-19) give the formulation of the Jacobian matrix and the Joint Torques.

C.3.3 Decomposing Torque into Couple Forces

The blocks responsible for creating the couple forces, FM_Upper and FM_Lower are shown in Figure C-30.

The block, "HD2_FindMomentArm" takes as its input, Handle Length, HandleBarEEOffset (equal to half of Handle Length) and the 3x3 rotation orientation matrix. The outputs are r_upper and r_lower which are vectors defined as starting from the center of the wand and ending at the upper and lower positions respectfully.

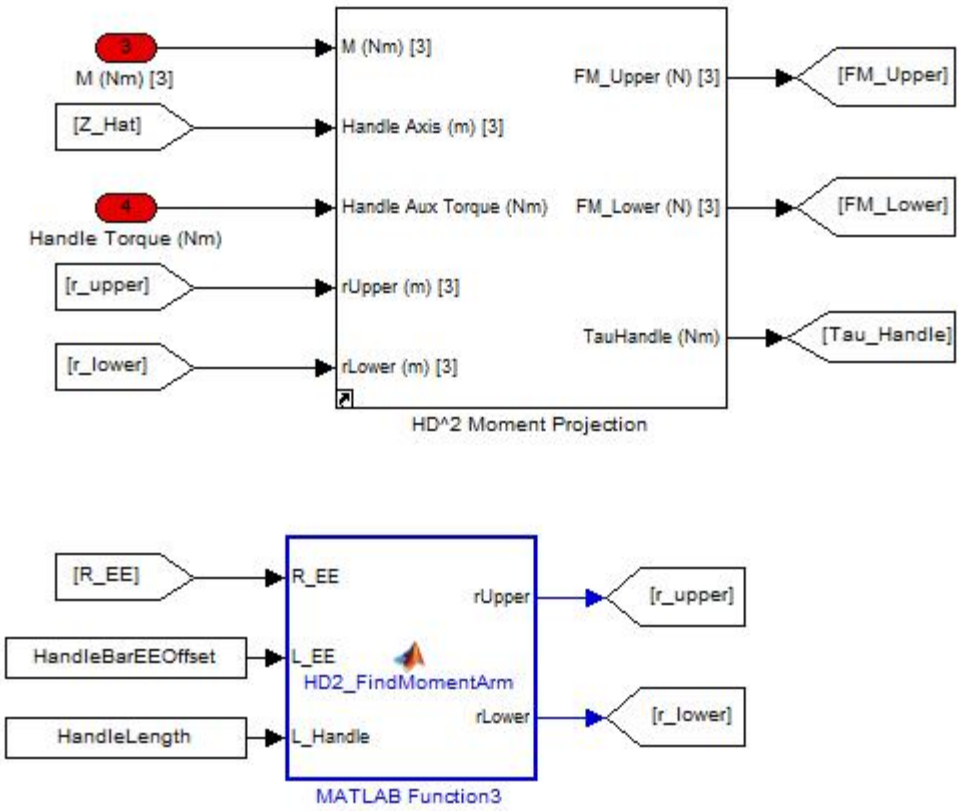

Figure C-30: Simulink Blocks 'HD^2 Moment Projection' and 'HD2_FindMomentArm' [62]

Figure C-31 shows the inside of the block, "HD^2 Moment Projection". The block "Vector Decomposition" simply takes the Torque vector and splits it into two components: a component parallel to the haptic wand and a component perpendicular to the haptic wand. The parallel torque component is added to "Handle Aux Torque" to create "TauHandle".

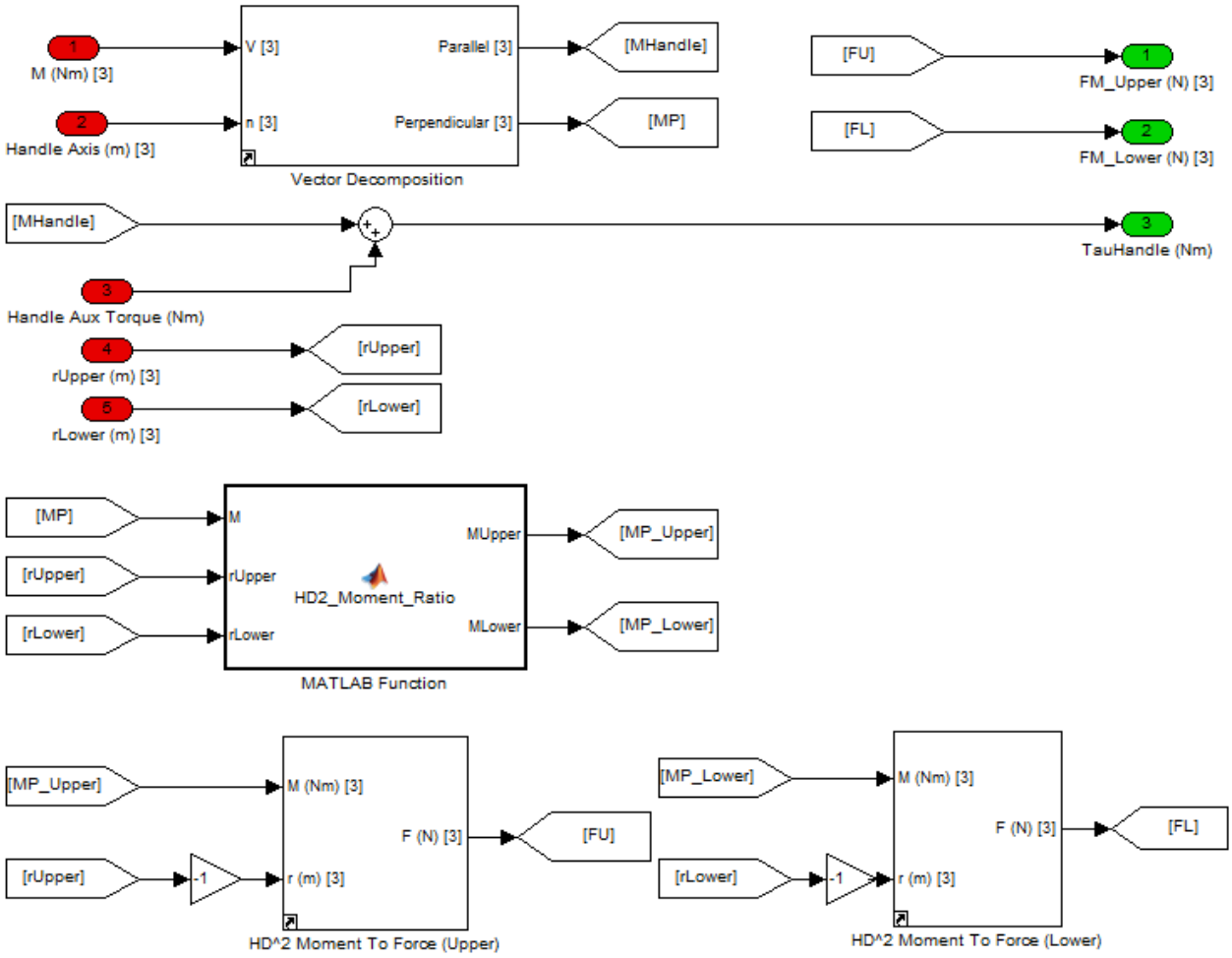

Figure C-31: Inside the "HD^2 Moment Projection" Block [62]

The blocks, "HD^2 Moment to Force" are identical for both the upper and lower cases. It calculates Couple Force, F_c via:

$$F_c = \frac{M \times r}{|r|^2} \tag{C-20}$$

Where,

M = MP_Upper or MP_Lower

r = rUpper or rLower

Naturally, the upper and lower force couples end up equal in magnitude and opposite in direction. While the torque is now correctly decomposed into Force Couples, the method has twice the necessary computation and is thus inefficient.

C.3.4 Euler Angles and Rotation Matrix

The block which outputs the Euler Angles of the Handle is shown below. The Euler Angles are not used anywhere else by this driver block so it is an immediate output. It is important to note that these angles are defined differently from the previously defined θ_x and θ_y (Section C.2.5 Calculating Wand Orientation Angles).

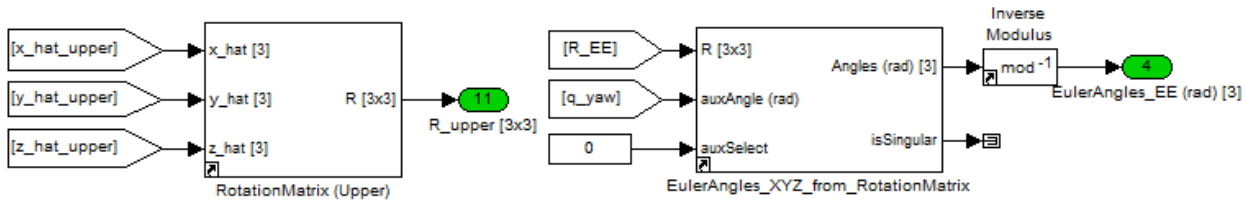

Figure C-32: Blocks Responsible for Determining Rotation Matrix and Euler Angles [62]

The rotation matrix, R_upper gives the rotation matrix from the base frame (0) to the end effector frame of Link 3. The unit vectors which comprises the rotation matrix were outputs from the 'HD^2 5-Bar (PY)' block (Figure C-28). An identical block is used to calculate the Rotation Matrix for the lower arm as well.

$$ {}^{0}_{e}R = \begin{bmatrix} s_{\theta 3} & c_{\theta 3} & 0 \\ -c_{\theta 1}c_{\theta 3} & c_{\theta 1}s_{\theta 3} & -s_{\theta 1} \\ -s_{\theta 1}c_{\theta 3} & s_{\theta 1}s_{\theta 3} & c_{\theta 1} \end{bmatrix} \quad \text{(C-21)} $$

Where

${}^{0}_{e}R$ is the rotation matrix from the base frame to the end effector frame (i.e. the end of link 3)

θ_1 is the rotation of link 1

θ_3 is the rotation of link 3 (defined with respect to the global Y direction)

C.3.5 Handle Length Error Checking

'x_upper' and 'x_lower' are the results of their respective 'HD^2 5-Bar (PY)' blocks. Using these two values, the block, 'HD2_Handle_Direction_Vector' creates two important outputs which are shown in Equations (C-22) and (C-24) below.

$$z = x_{upper} - x_{lower} \tag{C-22}$$

$$n = |z| \tag{C-23}$$

$$\hat{z} = {}^{z}/_{n} \tag{C-24}$$

Where

n is the absolute value of the vector z

$\hat{z}$ is the unit vector of vector z

As can be seen in Figure C-33 below, the unit vector $\hat{z}$ is feed into another block, while the absolute value, n is renamed as, 'x_HandleLength'.

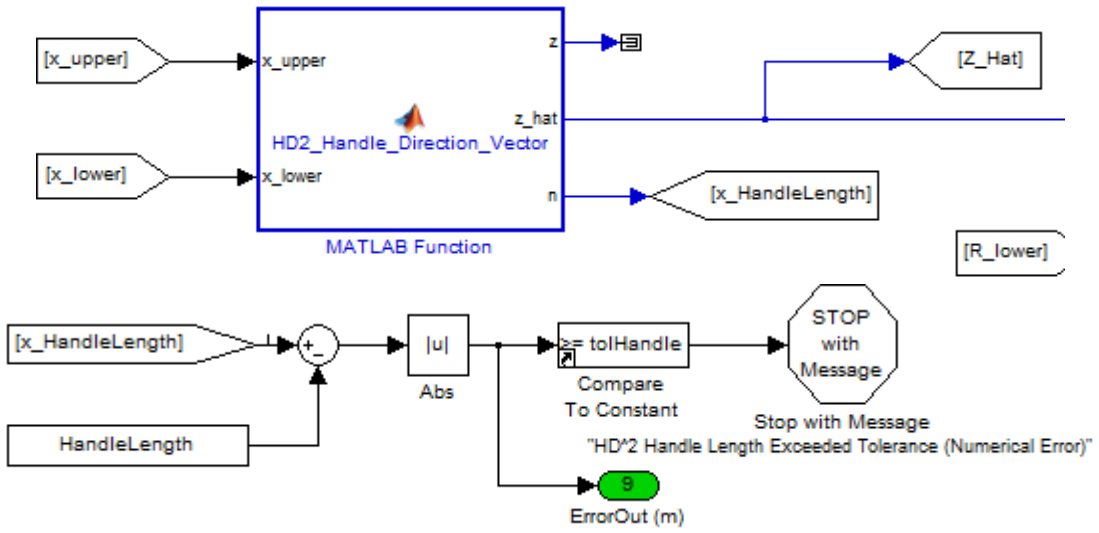

Figure C-33: Comparing the Distance between 'x_upper' and 'x_lower' to the Handle Length [62]

The error checking blocks find the difference between a constant, 'HandleLength' (with a value of 0.175m) and the calculated, 'x_HandleLength'. If the absolute difference is greater than a pre-determined tolerance value, the driver is shut down.

The default tolerance value is 0.25x0.175 = 0.04375m = 43.75mm.

C.3.6 Finding the Handle Center

Using the unit vector, $\hat{z}$ and the rotation matrix, 'R_lower', the block, 'HD_Handle_Base_POSE' calculates the rotation angles of the universal joint located at the end of the lower 5-bar mechanism.

This block is shown in Figure C-34. The code that carries out this calculation is copied below:

```
% Calculate the POSE of the handle base connector
% based on the direction unit vector between the two 5-bar EE.
% Assumption: the direction unit vector is in parallel with the handle axis


tol = eps;

% First, need to rotate the handle axis from HD^2 inertial frame to the
% 5-bar EE frame.
Z = R_lower' * unitZ;
%Z = unitZ;

if abs(Z(1)-1) < tol
    % Singular point, assign q1 to 0 arbitrarily
    q1 = 0;
    q2 = sign(Z(1))*pi/2;
else
    q2 = asin(Z(1));
    if abs(Z(3)) < tol
        % Singular point, since q2 == pi/2 is already covered before
        % q1 must be either pi/2 or -pi/2
        q1 = -asin( Z(2)/cos(q2) );
    else
        q1 = -atan2( Z(2), Z(3) );
    end
end
qh = [q1 q2]';
```

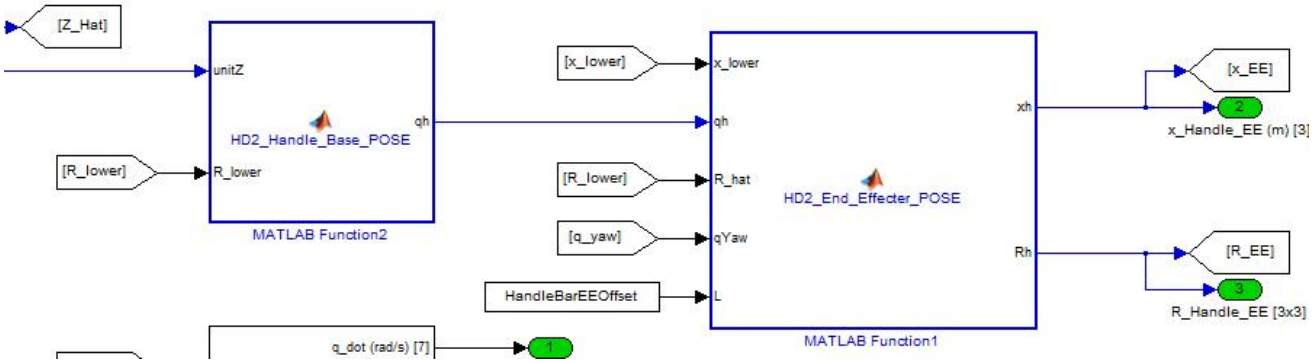

Figure C-34: "HD2_End_Effector_Pose" block [62]

q1 refers to the rotation about the local X axis, while q2 is the rotation about the local Y axis. These angles are fed to a secondary block as shown in Figure C-34. The end effector position and rotation matrix is calculated as follows:

The rotation matrix from 5-bar End Effector to Handle End Effector is:

$$RhEE = \begin{bmatrix} c_{q2} & 0 & s_{q2} \\ s_{q1}s_{q2} & c_{q1} & -s_{q1}c_{q2} \\ -c_{q1}s_{q2} & s_{q1} & c_{q1}c_{q2} \end{bmatrix} \tag{C-25}$$

The rotation matrix for Local Handle Yaw Rotation is:

$$RhEE_{Yaw} = \begin{bmatrix} c_Y & -s_Y & 0 \\ s_Y & c_Y & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{C-26}$$

Recall: R_{Lower} is the rotation matrix from the base to the end of Link 3 as defined by Equation (C-21).

Therefore, the overall Rotation Matrix from Base Frame to Handle End Effector is:

$$Rh = R_{Lower}\ \ RhEE\ \ RhEE_{Yaw} \tag{C-27}$$

The position vector with respect to 5-bar End Effector frame is:

$$x' = L\, s_{q2} - 0.022\, c_{q2} \tag{C-28}$$

$$y' = -L\, s_{q1}\, c_{q2} - 0.022\, s_{q1}\, s_{q2} \tag{C-29}$$

$$z' = L\, c_{q1}\, c_{q2} + 0.022\, c_{q1}\, s_{q2} \tag{C-30}$$

Where L = 0.175/2

$$x_h = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \tag{C-31}$$

Handle Center Position with respect to HD[2] base frame is:

$$^{0}_{e}x = x_{lower} + Rh * x_h \tag{C-32}$$

For reference, Figure C-35 shows the end effector handle of the HD[2]. The angles q1 and q2 are shown as the X and Y rotations respectfully. Also shown are the handle length of 175mm and the offset length of 22mm on the universal joint link.

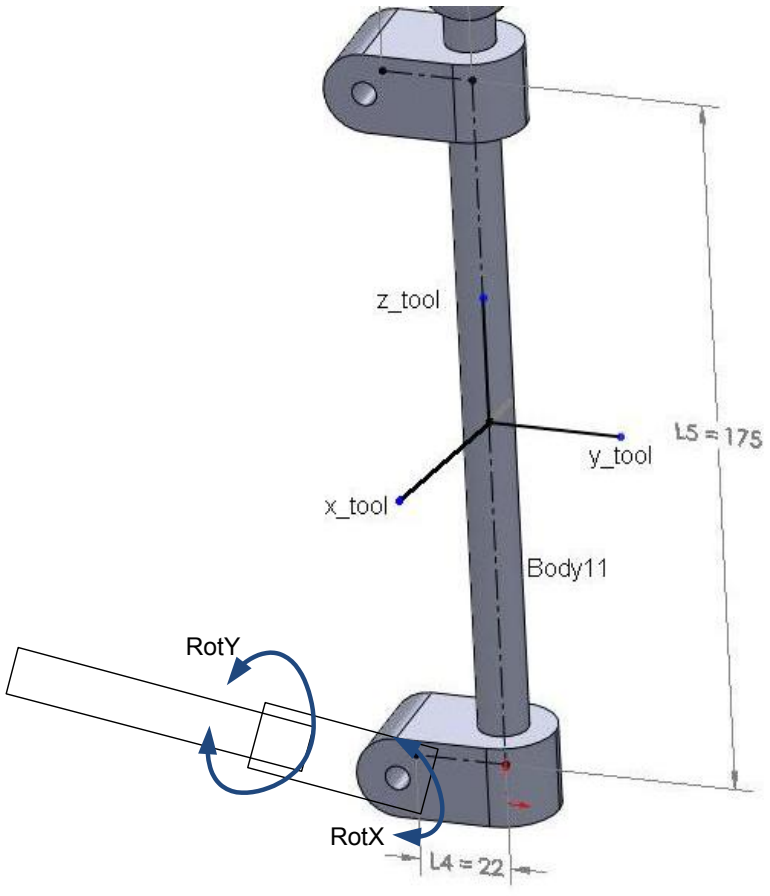

Figure C-35: Handle and angles q1 and q2. Lengths are in mm [28]

Appendix D – HD² C++ API Code

The API code which allows the C++ Temporal Bone Haptic Simulation to communicate with the HD^2 is shown below.

```
#include "stdafx.h"
#include "HD2_Interface.h"
#include "Matrix.h"
#include <Math.h>

// extern static float g_pedalvoltage;

HD2Interface *p_HD2pointer;

HD2Interface::HD2Interface()      //Constructor
{
       m_HD2ControlThreadInfo.pParent = this;
       p_HD2pointer = this;
       m_HD2ControlThreadInfo.done = false;

       }

HD2Interface::~HD2Interface(void)        //Deconstructor
{
/*     HANDLE hthread;
       if (m_HD2ControlThreadInfo.SimThreadPtr != NULL) {
              hthread = m_HD2ControlThreadInfo.SimThreadPtr->m_hThread;
              m_HD2ControlThreadInfo.done = true;
              ::WaitForSingleObject(hthread,100);
       }
*/
       m_HD2SendBuffer[14] = 0;    //[14, 15] Amp disabled

       int result = stream_send_doubles(p_HD2pointer->m_ClientStreamID, p_HD2pointer-
>m_HD2SendBuffer, SND_STREAM_SIZE);

       m_HD2ControlThreadInfo.done = true;
}


//=========================================================================
// void Init(); // init the interface
//
//=========================================================================
void HD2Interface::InitRenderer()  //Initialization thread
{
       m_HD2ControlThreadInfo.done = false;
       m_HD2ControlThreadInfo.pParent = this;


       const t_boolean nonblocking = false;
       const t_int   send_buffer_size = 8000;
       const t_int receive_buffer_size = 8000;         // */
```

```
	const char uri[] = "shmem://foobar:1";
	result = stream_connect(uri, nonblocking, send_buffer_size, receive_buffer_size,
&m_ClientStreamID); //

	// from Position_Control.cpp, quansar api position example

	for (int i = 0; i < RCV_STREAM_SIZE; i++)
		m_HD2ReceiveBuffer[i] = 0;
	for (int i = 0; i < SND_STREAM_SIZE; i++)
		m_HD2SendBuffer[i] = 0;

	/*[0:2] :force to be applied to the end-effector in x, y and z DOF's in SI
	[17] : torque to be applied to the end-effector in Yaw DOF in SI units.*/

	m_HD2SendBuffer[12] = 0;		// safety mode is on if 1
	m_HD2SendBuffer[13] = 0.0775;	//Originally Vibration Threshold NOW USED AS
Drill bit tip offset in Metres [m]
	m_HD2SendBuffer[14] = 1;
	m_HD2SendBuffer[15] = 0;	//[14, 15] Amp enable signal [1 0] is on
//	m_HD2SendBuffer[16] = 0;	// Calibration flag. Calibrates when 1
//	m_HD2SendBuffer[18] = 77.5; //Additional input for drill bit length [mm] (initial
set) DELETED

	m_HD2ControlThreadInfo.SimThreadPtr = NULL;
	m_HD2ControlThreadInfo.SimThreadPtr =
			AfxBeginThread(ThreadHD2Control,(LPVOID) &m_HD2ControlThreadInfo,
THREAD_PRIORITY_NORMAL);
	::WaitForSingleObject(m_HD2ControlThreadInfo.SimThreadPtr->m_hThread,1000);
	m_bEnabled = m_HD2ControlThreadInfo.success_init;

}

//=========================================================================
// UINT Thread that gets and sends values to HD2(LPVOID lpInfo)
//
//=========================================================================
UINT HD2Interface::ThreadHD2Control(LPVOID lpInfo) {

	HD2Interface::THREADINFO *threadinfo = (HD2Interface::THREADINFO *)lpInfo;
	//lpInfo is a pointer to this threadinfo struct
	HD2Interface *parent = p_HD2pointer;

	double rad_x, rad_y, x, y, z;		//

/*	//Calibration
	// */

	float F_limit = 10;

	while (parent->m_HD2ControlThreadInfo.done == false) {

		//Receive data from HD2 Driver
		parent->result = stream_receive_doubles(parent->m_ClientStreamID, parent-
>m_HD2ReceiveBuffer, RCV_STREAM_SIZE);	//Copied from HD2Test

		rad_x =  parent->m_HD2ReceiveBuffer[3] ; //Roll (theta x) in radians
	HD2 driver gives angle output in radians
```

```
            rad_y =  parent->m_HD2ReceiveBuffer[4] ; //Roll (theta y) in radians

            //Position of Wand Center
            x = 1000*(parent->m_HD2ReceiveBuffer[0] - 0.07 );            //constants
are to center the home position nicely relative to the bone model
            y = 1000*(parent->m_HD2ReceiveBuffer[1] - 0.27 );
            z = 1000*(parent->m_HD2ReceiveBuffer[2] + 0.368);      //  For case with no
drill 0.18  For case With drill +0.368

            Matrix trans;              //Will pre multiply matrix later on
(trans*matrix)
            MatrixTranslation(trans,0,0,-parent->GetTipOffset()); // the tip offset

            Matrix matrix;                    //Orientation Matrix

            matrix.m[0][0] = -cos(rad_y);
            matrix.m[0][1] = -sin(rad_x)*sin(rad_y);
            matrix.m[0][2] = -cos(rad_x)*sin(rad_y);  //z
            matrix.m[0][3] = 0;
            matrix.m[1][0] = 0;                               //Manual population of the
transposed orientation matrix (First two colums made NEGATIVE x&y )
            matrix.m[1][1] = -cos(rad_x);     // Matrix is transpose of Rotx*Roty
            matrix.m[1][2] = sin(rad_x);      //z
            matrix.m[1][3] = 0;
            matrix.m[2][0] = -sin(rad_y);     // Transpose of transformation matrix
used because this is the convention used by Jay/Omni Phantom
            matrix.m[2][1] = sin(rad_x)*cos(rad_y);
            matrix.m[2][2] = cos(rad_x)*cos(rad_y);   //z
            matrix.m[2][3] = 0;
            matrix.m[3][0] = -x;
            matrix.m[3][1] = -y;
            matrix.m[3][2] = z;         //-40
            matrix.m[3][3] = 1;                               // */


            parent->PushMedPedalState(parent->m_HD2ReceiveBuffer[5], parent-
>m_HD2ReceiveBuffer[12]);   //Hd2 time also sent
            //Above line is to allow the medtronic drill state (either 0 or 1 or
voltage float 0-5.5V) to be sent to HapticControl. (Final destination is HapticSpace)

            g_pedalvoltage = parent->m_HD2ReceiveBuffer[5];

            matrix = trans*matrix;
            parent->SetCurrentMatrix(matrix);

            Point3 pos = Point3((float)-x, (float)-y, (float)-z ); //This is necessary
for the HD2 to match the Graphic's LeftHanded co-ord system
            p_HD2pointer->SetCurrentPosition(pos);


            // OUTPUTS - Default Output is everything is zero
            parent->m_HD2SendBuffer[5] = 0;          //Control Mode - force
            parent->m_HD2SendBuffer[0] = 0;          //Fx   Forces are generated at the
drill tip
            parent->m_HD2SendBuffer[1] = 0;          //Fy
            parent->m_HD2SendBuffer[2] = 0;          //Fz
            parent->m_HD2SendBuffer[3] = 0;          //Tx = 0 (global) ALWAYS EQUAL TO
ZERO
```

```
		parent->m_HD2SendBuffer[4] = 0;		//Ty = 0 (global) ALWAYS EQUAL TO
ZERO

//		parent->GenerateNextForce(parent->m_HD2ReceiveBuffer[5]);		//
HD2Interface inherits members from the class IHapticRender
		parent->GenerateNextForce();
		if (parent->IsForceOutput())
		{
			Point3 forceoutput = parent->GetForceOutput();  //Fetches force from
Jay PID force calculation

			if( forceoutput.x > F_limit )
				parent->m_HD2SendBuffer[0] = -F_limit;	//Force limiter.
Also x = -x
			else if( forceoutput.x < -F_limit )
				parent->m_HD2SendBuffer[0] = F_limit;	//Force limiter.
Also x = -x
			else
				parent->m_HD2SendBuffer[0] = -forceoutput.x;	//Set force
as is. (Falls within limits)

			if( forceoutput.y > F_limit )
				parent->m_HD2SendBuffer[1] = -F_limit;	//Force limiter.
Also y = -y
			else if( forceoutput.y < -F_limit )
				parent->m_HD2SendBuffer[1] = F_limit;	//Force limiter.
Also y = -y
			else
				parent->m_HD2SendBuffer[1] = -forceoutput.y;

			if(forceoutput.z > F_limit)
				parent->m_HD2SendBuffer[2] =  F_limit;
			else if(forceoutput.z < -F_limit)
				parent->m_HD2SendBuffer[2] =  -F_limit;
			else
				parent->m_HD2SendBuffer[2] = forceoutput.z;
	//Force limit on Z
		}

//		if(parent->m_HD2ReceiveBuffer[5] > 0.5 )  //If either pedal depressed,
assume rotation
//			parent->m_HD2SendBuffer[3] = 0.5;		// Test

		//Send the force & torque values to HD2 Driver
		parent->result = stream_send_doubles(parent->m_ClientStreamID, parent-
>m_HD2SendBuffer, SND_STREAM_SIZE);

		parent->result = stream_flush(parent->m_ClientStreamID);		// in
the api might as well use it
		threadinfo->success_init = true;

		//Sleep(1);   //Allows other threads to run
	}
	return 0;
}
```

Appendix E – Survey Instrument

The survey instrument used in the "Evaluation of Various Haptic Manipulandums and Devices in Temporal Bone Simulation" study is copied here.

Evaluation of Various Haptic Manipulandums and Device in Temporal Bone Simulation

Study Coordinator: Vivek Rampersad

Instructions

You will drill a virtual temporal bone using three different systems. The order of use will be randomized. The three systems are an Omni (Geomagic Touch) haptic device, an HD^2 haptic device with a standard manipulandum and an HD^2 with a otic drill replacing the standard manipulandum. We are attempting to determine how device and manipulandum affect users' perception of realism when compared to their prior experience with cadaveric training.

The haptic software and graphic visualization are identical for all three systems. All systems use a foot pedal controller for drilling. Please grasp the manipulandum in a way which feels natural to you.

You will be asked to perform several virtual procedures.

Order of Scheduled Dissection:

- Cortical Mastoidectomy
- Facial Recess Dissection
- Labyrinthectomy
- Translabyrinthine Approach to the IAC

After completing these procedures on a given device, and before moving on the next device, please fill out the corresponding questionnaire on that device. Please enter your participant identifier at the top right corner of each page.

After completing the procedures and questionnaires on all three devices please complete the overall manipulandum survey.

Thank you for your participation in this study.

Omni/HD^2 Only/ HD^2 with Drill - Post-Dissection Survey

Physical representation of simulated bone: Rate the similarity of this simulation to actual cadaveric simulated surgery with respect to the following osseous properties: 1 – Very Dissimilar 4 – Undecided 7 – Very Similar							
	1	2	3	4	5	6	7
Entire Construct							
Hardness of Cortical Bone							
Hardness of Trabecular Bone							
Vibrational properties/Feel							
Acoustic properties/Sound							
Drill Skip							
Overall appreciation/similarity of bone							
Air-cell System							

Overall Manipulandum Comparison Survey

For each of the following please indicate which form of haptic simulation was superior: Rank from 1 to 3 where 1 is most preferred			
	Omni	HD2 Only	HD2 with Drill
Overall Realism			
Osseous Realism			
Drilling Realism			
Preferred for Education			
Preferred for Preoperative Planning			
Ease of Use			
Preferred Simulation			

References

[1] M. Sanna, T. Khrais M. Falcioni, A Russo and A Taibah, *The Temporal Bone: A Manual for Dissection and Surgical Approaches*, Thieme, New York, 2006.

[2] J. Lane and R. Witte, "2 Anatomy" in *The Temporal Bone An Imaging Atlas*, Springer, New York, 2010, ch. 2, pp 7-28.

[3] R. A. Nelson, *Temporal Bone Surgical Dissection Manual*, Second ed., House Ear Institute, Los Angeles, 1991.

[4] Henry Gray, "II. Osteology – 5c. The Exterior of the Skull" in *Anatomy of the Human Body*, 20th ed. Philadelphia: Lea & Febiger, ch. II, sec. 5.c, 1918.

[5] Henry Gray, "II. Osteology – 5a.4. The Temporal Bone" in *Anatomy of the Human Body*, 20th ed. Philadelphia: Lea & Febiger, ch. II, sec. 5.a.4, 1918.

[6] Henry Gray, "II. Osteology – 2. Bone" in *Anatomy of the Human Body*, 20th ed. Philadelphia: Lea & Febiger, ch. II, sec. 2, 1918.

[7] J. Hochman, J. Kraut, K. Kazmerik and B. Unger, 'Generation of a 3D Printed Temporal Bone Model with Internal Fidelity and Validation of the Mechanical Construct', Otolaryngology -- Head and Neck Surgery, vol. 150, no. 3, pp. 448-454, 2013.

[8] J. Kraut, J. Hochman and B. Unger, 'Temporal bone surgical simulation employing a multicore architecture', *26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, Regina, SK, 2013.

[9] L. Javia and E. Deutsch, 'A Systematic Review of Simulators in Otolaryngology', *Otolaryngology - Head and Neck Surgery*, vol. 147, no. 6, pp. 999-1011, 2012.

[10] J. Flávio, N. Júnior and D. Cruz, 'Real Models and Virtual Simulators in Otolaryngology: Review of Literature', Brazilian Journal of Otorhinolaryngology, Vol. 76, pp. 129-135, 2010.

[11] K. Abou-Elhamd, A. Al-Sultan, U. Rashad, "Simulation in ENT Medical Education", *The Journal of Larngology & Otology*, vol. 24, pp. 237 – 241, 2009.

[12] A. P. George, R. De, "Review of temporal bone dissection teaching: how it was, is and will be," *The Journal of Laryngology & Otology,* vol. 124, no. 2, pp. 119–125, 2010.

[13] N. Clifton, C. Klingmann, H. Khalil, "Teaching Otolarngology Skills through Simulation", *Oto-Rhino-Larngology*, vol. 268, p 949-953, 2011.

[14] B. Jun, S. Song, J. Cho, C. Park, D. Lee, K. Chang and S. Yeo, "Three-dimensional reconstruction based on images from spiral high-resolution computed tomography of the temporal bone: anatomy and clinical application", *Journal of Laryngology & Otology,* vol. 119, p 693 – 698, 2005.

[15] M. Sørensen, A. Dobrzeniecki, P. Larsen, T. Frisch, J. Sporring, T. Darvann, "The visible ear: a digital image library of the temporal bone", *ORL J Otorhinolaryngol Relat Spec*, vol. 64, pp. 378–381, 2002.

[16] L. Chan, S. Manolidis, K. Kaber, L. Hayman, "Surgical Anatomy of the Temporal Bone: An Atlas", *Neuroradiol*, vol. 43, p 797 – 808, 2001.

[17] S. Nayak, "Segmental Anatomy of the Temporal Bone", *Seminars in Ultrasound, CT and MRI*, vol. 22, p. 184-218, 2001.

[18] M. Sorensen, J. Mosegaard and P. Trier, "The Visible Ear Simulator: A Public PC Application for GPU-Accelerated Haptic 3D Simulation of Ear Surgery Based on the Visible Ear Data", *Otology & Neurology*, vol. 30, p. 484-487, 2009.

[19] A. Arora, L. Lau, Z. Awad, A. Darzi, A. Singh and N. Tolley, 'Virtual Reality Simulation Training in Otoaryngology", *International Journal of Surgery*, vol 12, Issue 2, p. 87 – 94, 2013.

[20] M. Zirkle, D. Roberson, R. Leuwer, A. Dubrowski, "Using a virtual reality temporal bone simulator to assess otolaryngology trainees," *Laryngoscope,* vol. 117, no. 2, p. 258–263, 2007.

[21] E. Deutsch, "Simulation in Otolaryngology: Smart Dummies and More", *Otolarngology – Head and Neck Surgery*, vol. 145, pp. 899-903, 2011.

[22] PHACON, *PHACON Temporal Bone System* [Online], Available: http://www.phacon.de/index.php/en/ent/temporal-bone-surgery, [cited Oct 2014].

[23] M. Garcia, J. Manuel, A. Costa et al, "Temporal Bone Dissection Practice Using a Chicken Egg", *Otology & Neurotology*, vol. 35, pp. 941-943, 2014.

[24] [21] P. Mick, C. Arnoldner, J. Mainprize, S. Symons, J. Chen, "Face Validity Study of an Artificial Temporal Bone for Simulation Surgery", *Otology and Neurotology*, vol. 34, pp. 1305-1310, 2013.

[25] J. Hochman, B. Unger, N. Sepehri, M Khazaraee, V. Rampersad, J. Kraut, J. Pisa, "Mixed Reality Temporal Bone Surgical Dissector: Mechanical Design", *Journal of Otolaryngology*, vol. 43, pp. 23-28, August 2014.

[26] M. Srinivasan, "What is haptics?", [Online], Available: http://www.geomagic.com/files/7713/4857/8044/what_is_haptics.pdf, [cited Oct 2014].

[27] M. Srinivasan, MIT Touch Lab – Touch Lab Research, [Online], Available: http://touchlab.mit.edu/oldresearch/index.html, [cited Oct 2014].

[28] Quanser, "Quanser's High Definition Haptic Device – User Manual", Markham ON, 2009.

[29] Quanser, "System Specifications – High Definition Haptic Device", [Online], Available: http://www.quanser.com/Products/Docs/2195/HD%5E2_High_Def_Haptic_Device_System_Specifications_v1.2.pdf, [cited July 2014].

[30] Quanser, "HD2 API Quick Start Guide", Markham, ON, 2009.

[31] Quanser, "Maple Worksheet – HD2 Kinematics and Dynamics", Markham, ON, 2009.

[32] Gilbert Lai, Systems and Controls Engineer, Quanser, Markham, ON, Email Correspondence, 2013 – 2014.

[33] L.F. Lee, M. Narayanan, F. Mendel, V Krovi and P. Karam, "Kinematic Analysis of In-Parallel 5-DoF Haptic Device", Dept. Of Mech. & Aerospace Engineering, University at

Buffalo and Quanser Inc, *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Montreal, Canada, July 6-9, 2010.

[34] L. Sciavicco and B. Siciliano, "2. Kinematics," in *Modelling and Control of Robot Manipulators*, Springer Science & Business Media, 2000, pp 20-35.

[35] M. Chan, G. Cohen & J. Deasy, "Qualitative Evaluation of Fiducial Markers for Radiotherapy Imaging", *Technology in Cancer Research & Treatment*, vol. 1(7), 2014.

[36] C. Shih, C. Ruo, "Auto-Calibration of an SMT Machine by Machine Vision", *Intl Journal of Advanced Manufacturing Technology*, Aug. 2005, vol. 26, Issue 3, pp 243-250.

[37] H. Kjer and J. Wilm, Bachelors Thesis: "Evaluation of surface registration algorithms for PET motion correction", BSc Thesis, Informatics and Mathematical Modelling, Technical University of Denmark, June 2010, [Online], Available: http://www2.imm.dtu.dk/~jakw/publications/bscthesis.pdf.

[38] B.P. Lathi, "7.4 Filter Design by Placement of Poles and Zeros of H(s)" in *Signal Processing & Linear Systems*, Carmichael, CA, Berkeley-Cambridge Press, 1998, ch.7.4, pp 495-504.

[39] A. Sedra and K. Smith, "16.4.1 First-Order Filters" in *Microelectronic Circuits*, 6th ed., New York, Oxford University Press, 2010, chp 16.4.1, pp. 1271-1272.

[40] Brooks, T.L., "Telerobotic response requirements", *1990 IEEE International Conference on Systems, Man, and Cybernetics Conference Proceedings*, pp.113-120, 1990.

[41] L, Grebler, (2010, Jan, 31), "High Definition Haptics in Manitoba", [Online], Quanser, Markham, ON, Available: http://quanser.blogspot.ca/2010/01/high-definition-haptics-in-manitoba.html, [cited Dec 2014].

[42] Quanser Inc, (2014, Feb, 27), "Research Work Using Quanser 6 DOF Telepresence System Published in a Prestigious Journal", [Online], Available: http://quanser.blogspot.ca/2014/02/research-work-using-quanser-6-dof.html, [cited Dec 2014].

[43] H. Marcus, K. Zareinia, L. Gan, F. Yang, S. Lama, G. Yang and G. Sutherland, "Forces exerted during microneurosurgery: a cadaver study", *International Journal of Medical Robots and Computer Assisted Surgery*, vol. 10, Issue 2, pp 251-256, Jan 2010.

[44] J. Kraut, "Haptic Simulation Write Up" Laboratory for Surgical Modeling, Simulation and Robotics, University of Manitoba, Feb 2013.

[45] Geomagic 3D Systems. (2015, Aug.). "*Geomagic Haptic Brochure*", [Online], Available: http://www.geomagic.com/files/6114/3940/9416/Haptic_Device_brochure-8-2015-final.pdf [Sept 2015].

[46] F. Baumgart, "Stiffness – an unknown world of mechanical science?", *Injury, Int. J. Care Injured*, vol. 31. 2000.

[47] J. Guignard, A. Arnold, C Weisstanner, M. Caversacco and C. Stieger, "A Bone-Thickness Map as a Guide for Bone-Anchored Port Implantation Surgery in the Temporal Bone", *Materials*, vol. 6(11), 2013, pp. 5291-5301.

[48] D. Wong, J. Hochman, B. Unger and J. Kraut, "Comparison of Cadaveric and Isomorphic Virtual Haptic Simulation in Temporal Bone Education", Journal of Otolaryngology - Head & Neck Surgery, Oct 2014.

[49] Montgomery, Runger and Hubele, "5-8 What If We Have More Than Two Samples?" in *Engineering Statistics*, 4th ed. Arizona State University, Wiley, 2007, ch. 5, pp 256-270.

[50] Andrew Gelman, "Analysis of Variance – Why it is More Important than Ever", Columbia University, *The Annals of Statistics*, Vol. 33, No.1, 1-53, 2005.

[51] Montgomery, Runger and Hubele, "5-3 Inferences on the Means of Two Populations, Variances Unknown" in *Engineering Statistics*, 4th ed. Arizona State University, Wiley, 2007, ch. 5, pp 224-234.

[52] D. Morris, C. Sewell, N. Blevins, F. Barbagli and K. Salisbury, "*A Collaborative Virtual Environment for the Simulation of Temporal Bone Surgery*", Proceedings of MICCAI, Sept 2004, Springer-Verlag Lecture Notes in Computer Science, Volumes 3216 and 3217.

[53] D. Morris, C. Sewell, F. Barbagli, N. Blevins, S. Girod and K. Salisbury, "*Visuohaptic Simulation of Bone Surgery for Training and Evaluation*", IEEE Transactions on Computer Graphics and Applications, Nov 2006, pp. 48-57.

[54] C. Sewell, D. Morris, N. Blevins, S. Dutta, S. Agrawal, F. Barbagli, and K. Salisbury, "Providing Metrics and Performance Feedback in a Surgical Simulator", *Computer Aided Surgery*, vol. 13, Issue 2, March 2008, pp. 63-81.

[55] C. Sewell, D. Morris, N. Blevins, F. Barbagli, K. Salisbury, "Quantifying Risky Behavior in Surgical Simulation", *Proceedings of MMVR*, *Stud Health Technol Inform*, vol. 111, 2005, pp. 451-457.

[56] C. Sewell, D. Morris, N. Blevins, F. Barbagli, K. Salisbury, "Evaluating Drilling and Suctioning Technique in a Mastoidectomy Simulator", *Proceedings of MMVR*, *Stud. Health Technol Inform*, vol. 125, 2007, pp. 427-432.

[57] C. Sewell, D. Morris, N. Blevins, S. Dutta, F. Barbagli, K. Salisbury, "Validating Metrics for a Mastoidectomy Simulator", *Proceedings of MMVR*, *Stud. Health Technol Inform*, vol. 125, 2007, pp. 421-426.

[58] S. Aoyagi, A. Kohama, Y. Nakata, Y. Hayano and M. Suzuki, "Improvement of Robot Accuracy by Calibrating Kinematic Model Using a Laser Tracking System – Compensation of Non-Geometric Errors Using Neural Networks and Selection of Optimal Measuring Points Using Genetic Algorithm", *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, Oct 18-22, 2010.

[59] E. Morgan, H. Bayraktar and T. Keaveny, "Trabecular bone modulus-density relationships dependent on anatomic site" in *Journal of Biomechanics*, vol. 36, 2003.

[60] H. Bayraktar, E. Morgan, G. Niebur, G. Morris, E. Wong and T. Keaveny, "Comparison of the elastic and yield properties of human femoral trabecular and cortical bone tissue" in *Journal of Biomechanics*, vol. 37, 2004.

[61] J. Rho, T. Tsui and G. Pharr, "Elastic properties of human cortical and trabecular lamellar bone measured by nanoindentation" in *Biomaterials*, vol. 18, 1997.

[62] Quanser Inc. *HD^2 Driver* [HD2_Driver.mdl]. Markham, ON Canada, 2013.