Virtual Fixtures for Concentric Tube Robots in Minimally Invasive Surgery

Design Proposal

BME 489

September 28, 2016

Andrew Lim	999675542
Gloria Wu	999876530
Katlin Kreamer-Tonin	999709739
Sakeena Panju	999699642

Centre for Image Guided Innovation & Therapeutic Intervention (CIGITI)

Dr. Thomas Looi, Dr. James Drake

1 Executive Summary

This document proposes a feasible and user-friendly solution for the implementation of virtual fixtures for a concentric tube robot (CTR) on the da Vinci surgical system. If implemented successfully, this project would allow for CTRs to be used effectively with the da Vinci system, a commonly used robotic surgery platform, in minimally invasive surgery. CTR-based robotic manipulators pose benefits over the existing da Vinci manipulators because they allow for smaller tools to be used in surgery, which results in less patient trauma, quicker recovery times, and potential use in pediatric cases.

The current CTR implementation with da Vinci is not intuitive for users because the workspace of the CTR is restricted due to the geometry of its structure. Operators of a da Vinci system, who have a less restrictive workspace, may sometimes move their controls and not see a response from the robotic manipulators as a result. Haptic feedback in these scenarios would allow the user to know when they are inside or outside this CTR workspace.

Solutions for different virtual fixtures and haptic feedback mechanisms were explored, and evaluated against criteria such as intuitiveness, patient safety, speed to perform tasks, and implementation complexity. The proposed solution will use forbidden region haptic feedback in order to provide an intuitive mapping of the CTR's workspace to a user. This haptic feedback will be implemented through admittance virtual fixtures.

Once implemented, the solution will undergo several testing stages, including timing how long it takes a user to perform a standard surgical task, testing the boundaries of the workspace and measuring co-ordination between the master's position and the slave's position, and user acceptance testing in the form of a survey to evaluate the usefulness of the virtual fixtures.

The proposed solution will cost $250 to implement an initial proof of concept. Considering that in 2015 the revenue from tools used in da Vinci systems was $1,198 million [1], this is a worthwhile investment in the field of minimally invasive surgery.

2 Table of Contents

3 Project Description

3.1 Background and Motivation

Surgical techniques that were once performed on an open cavity are increasingly moving towards minimally invasive approaches. This involves operating through small incisions, ultimately leading to faster patient recovery and less scarring. A common method of performing of minimally invasive surgery (MIS) is through robotic surgery, where a surgical robot performs the required tasks while being teleoperated by an operator in another location. This has incredible implications in removing the physical barrier for surgeons, allowing surgeons to be available from anywhere in the world. A commonly accepted and widely used robotic surgery platform is the da Vinci surgical system, which operates by translating what the surgeon does on a controller into smaller, more precise movements on tools within the body. Over 3 million patients worldwide have been operated on with this system in a variety of applications [1]. Since 1998, over 8500 peer-reviewed publications have appeared in various clinical journals on da Vinci Surgery, emphasizing its importance in the surgical field [1].

Conventional da Vinci system tools range from 5 to 8 millimeters in diameter. While small compared to open surgery, shrinking this diameter is desirable to create smaller surgical workspaces. However, creating small actuation joints for robotic arms remains a challenge. To address this challenge, focuses have been shifted to alternative ways to actuate robotic arm links. For example, Concentric Tube Robots (CTR) work by altering the translational and rotational positions of pre-curved, concentric nitinol tubes to navigate to the surgical site. This greatly reduces the size of the actuation joints, allowing diameters to be much smaller than conventional tools. CTRs would allow surgeries to be performed in smaller spaces such as within the brain, or in pediatric surgery. Additionally, CTR technology using da Vinci systems has huge potential from a financial perspective. Instrumentation for a da Vinci system costs $700-3200 per surgery, and in 2015 the revenue from tools used in da Vinci systems was $1,198 million [1].

A current limitation of CTR technology is that the tool has an unintuitive workspace. This makes its implementation with da Vinci surgical systems a challenge. Typically, in the da Vinci system, the surgeon controls a “master” set of tools that directly maps to the “slave” (patient) tools. Without modifying the hardware of the master side, there is a mismatch between the workspace of the master and the CTR tool. If the surgeon attempts to move the CTR to a location that it cannot reach, the tool will “hover” in space, oscillating near the target point.

To prevent this "hovering" problem, an intuitive map of the workable space of the CTR needs to be implemented with the da Vinci’s master control system, without physically modifying the CTR or da Vinci system. The cost to implement virtual fixtures on the current da Vinci system is minimal, as it requires only software modification.

If implemented successfully, this project would allow for CTRs to be used effectively with a da Vinci robot in minimally invasive surgery. Smaller tools can be used, resulting in less patient trauma, quicker recovery times, and potential use in pediatric cases.

3.2 Project Goal

The goal of this project is to implement virtual fixtures on the master manipulators of da Vinci research kit (DVRK) based on the CTR’s workspace, with the goal of making surgical manipulation using the DVRK quicker, safer, and more intuitive for the operator. The operator should be aware of the CTR’s workspace, and be able to manipulate it in order to perform tasks.

The scope of this project is to modify the communication between the master controller and tools. The solution will make use of a currently implemented Simulink system in order to control and communicate between the CTR and DVRK. Modifying the hardware of the master controller or the CTR is out of scope for this project.

3.3 Project Requirements

3.3.1 Functional requirements

1. **Mapping CTR workspace**: The system must be able to determine if the master controller has breached the limitations of the CTR workspace. (This can be tested using position test cases).
2. **Haptic Feedback:** The user must be able to detect some form of haptic feedback when they attempt to penetrate the CTR's workspace boundaries, based on a blind user survey. Feedback detection should occur successfully at least 80% of the time (generally significant success rate).
3. **Speed of Feedback**: The time between an illegal position attempt and the feedback provided to the user must be less than 200ms, based on the average human tactile reaction time [2].
4. **Validation**: The user must be able to complete standard surgical tasks (peg transfer [3]) using the CTR.

3.3.2 Objectives

1. The employed system should not compromise the patient's safety. (evaluated by surgeon)
2. The employed solution should minimize the time spent "hovering". (time, ms)
3. The employed solution should minimize the magnitude of the "hover". (distance, mm)
4. Time taken to perform a task with the virtual fixtures should be less than the time taken without the virtual fixtures. (time, ms)
5. The time between an illegal position attempt and the feedback provided to the user should be as small as possible. (time, ms)
6. The system should require minimal change to clinical workflow. (evaluated by surgeon)

3.3.3 Constraints

1. The master controller on the da Vinci system must not be physically modified. (boolean)
2. The CTR must not be physically modified. (boolean)
3. Must be implemented with current DVRK; either in C++ or Simulink. (boolean)

3.4 Validation and Acceptance Tests

Table 1. Outlines the validation and verification tests for each of the functional requirements

Validation/Verification Process	Pass/Fail Criteria
1. Mapping CTR workspace	
Input position test cases	Pass: points outside workspace must be flagged Fail: points inside flagged, points outside not flagged
2. Haptic Feedback	
Test case: User tries to leave the workspace	Pass: haptic feedback occurs when outside workspace
3. Speed of Feedback	
Measure the time it takes for virtual fixture mechanism to actuate while master is moved incrementally towards workspace limitation, slowly and quickly (may be obtained by comparing timestamps in code).	Pass: <200ms between exiting workspace and virtual fixture mechanism actuation
4. Validation	
Ability to perform PEG transfer	Pass/Fail
Measure time it takes to perform PEG transfer with and without virtual fixtures	Pass: time with < time without
Issue qualitative survey asking users to rate effectiveness of virtual fixture (1-10)	Pass: average rating >5/10

4 Technical Design

4.1 Possible Solutions and Design Alternatives

4.1.1 Design Variable 1: Type of Virtual Fixture

There are two main types of virtual fixtures used in robotic surgical systems, whose purpose is to limit the slave from entering into the forbidden region (FR). The impedance method is defined as creating a virtual wall using haptic feedback to “impede” master movement into the FR thereby limiting slave movement into the FR [3]. On the other hand, the admittance method is defined as simply not sending any commands to the slave if the master is in the FR (see Figure 1, left) [3].

A common implementation of the impedance method is to correlate the physical force felt on the surgical tool directly to the magnitude of impedance experienced by the master controller [3]. This requires a force sensor to be implemented on the tool. For our application, haptic feedback should be provided whenever the master controller attempts to breach the limitations of the CTR workspace.

Therefore, haptic feedback forces would have to be generated through a mathematical model of the CTR's FR rather than sensed through the slave. This type of haptic feedback can also be applied to the admittance method. These altered methods are shown in Figure 1, right.

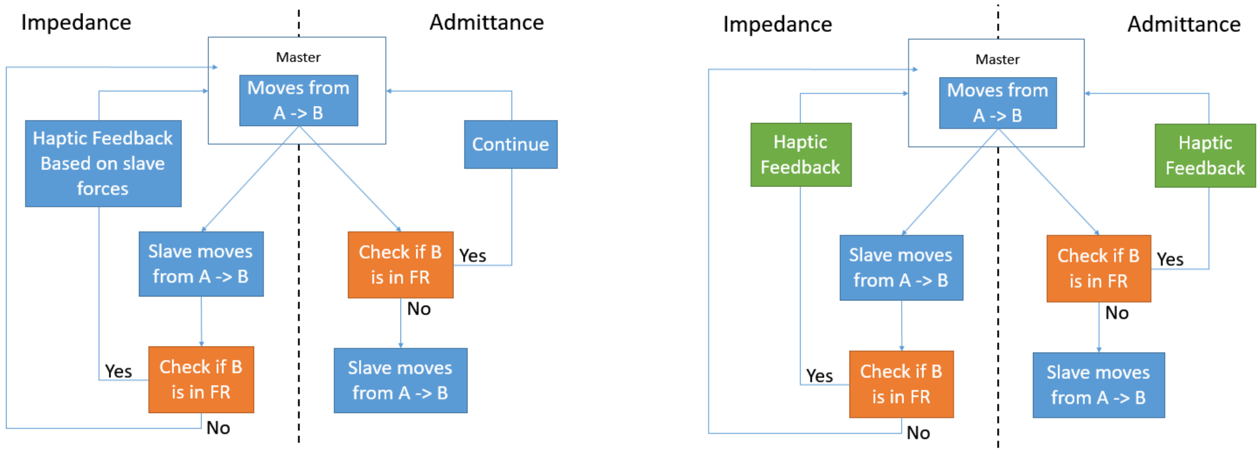


Figure 1: Left - comparison between basic definitions of the impedance and admittance methods. Right - comparison between impedance and admittance methods after altering the methods

4.1.2 Design Variable 2: Type of Haptic Feedback

4.1.2.1 Tactile (Vibrational) Feedback

A commonly used method for providing physical feedback to users in electronic devices is vibration [3]. A wide range of household items employ this type of feedback to alert the user in various ways, such as video game controllers and cells phones. With robot assisted MIS, vibrations can be used to provide alerts to the user when they are attempting to break the boundary of the CTR workspace. A possible implementation method is to cause a vibration proportional in magnitude to the distance travelled outside the workspace. Some disadvantages to vibrational feedback are that the user may experience unsteadiness with their tool or even psychological shock, simply due to the sudden vibration.

4.1.2.2 Spring Form (Guidance) Feedback

In this method of haptic feedback, the user would feel forces that are opposite in direction and proportional in magnitude to distance travelled past the boundary. Typically, this type of force is meant to guide the user back into the desired workspace, or along a path that is within the workspace [3]. An advantage to this type of feedback is that if the virtual fixture is incorrectly calibrated, the user can override the guidance and still perform the task [3]. It is also potentially task dependent, and studies show that it is optimal for tasks requiring you to follow a path.

4.1.2.3 Hard Stop (Forbidden Region) Feedback

Unlike the previous two methods, this type of force feedback restricts the user from breaching the CTR's defined boundaries [3]. This notion works well for this application since there are hard boundaries where the CTR simply cannot reach, potentially creating a more intuitive framework. One potential downside to this method is user frustration, caused by constantly restraining the operator from moving the tool in certain directions. Additionally, if there was any error or mismatch in the workspaces between the master and slave side of the DVRK, the user would not be able to override the virtual fixtures and may be unable to complete tasks without full calibration.

4.2 Assessment of Proposed Design

4.2.1 Design Variable 1: Type of Virtual Fixture

The following comparison (Table 2) of the two designs is based on the altered methods of impedance and admittance shown on the right side of Figure 1. The admittance method meets the objectives better in all cases, thus it was chosen as the proposed design.

Table 2: Comparison of impedance and admittance strategies for system implementation. Blue shows the prioritized criteria.

Objectives	Impedance	Admittance
1. Patient safety	May cause hovering of CTR which is not a controlled event and could have adverse effects	No adverse effect
2&3. Will the CTR hover?	When the CTR attempts to follow the master into FR it will hover	No since CTR will not ever attempt to enter into the FR
4. Does it improve the speed of tasks?	The speed between the two methods is equivalent.	
5. Is feedback immediate	Feedback is delayed until after the slave has moved	Feedback occurs slightly after the Master has moved
6. Workflow change?	Slightly improved, since user is alerted when the CTR has attempted to enter into FR	Improved, since feedback is administered before CTR attempts to enter into workspace which notifies user and limits hovering

Other - Implementation complexity	Must implement in DVRK/or create a new link to SIMULINK. May also require the more complex implementation of haptic feedback to limit CTR movement to FR (for example haptically defining a boundary region in order to warn a user when they are approaching the FR rather than limiting feedback to moments when the user is already in the FR).	Can be directly added to the existing link to MATLAB. MATLAB and SIMULINK are built to handle matrices and our team is familiar with the tool. Can use the Inverse Kinematics module to develop an error function to work as the check.

4.2.2 Design Variable 2: Type of Haptic Feedback

Table 3 summarizes the evaluation between the three haptic feedback designs, where the tactile and hard stop are the more relevant options since many of the guidance upsides are not applicable. By focusing on the prioritized criteria (blue) the hard stop method was the clear winner.

Table 3: Comparison of haptic feedback mechanisms. Blue shows the prioritized criteria.

Objective	Tactile (vibration)	Spring (guidance)	Hard Stop (forbidden region)
1. Patient safety	Possible shock/unsteadiness creates hazard	No new safety risks.	No new safety risks.
2&3. Will the CTR ever hover?	For all cases hovering is limited by the admittance algorithm, the CTR will never attempt to enter the FR.		
4. Does it improve speed of doing tasks?	Clear when boundary is reached, will minimize time spent in FR, and thus increase speed of tasks.	Unclear when boundary is reached, any user attempts to override feedback and enter past the boundary may increase resistance and slow user	Cannot enter into boundary, will eliminate all time spent in FR and thus significantly increase speed of tasks.
5. Is feedback immediate?	Feedback is immediate for all 3 alternatives		
6. Workflow changes?	Possible shock from vibrations	No change	Some areas inhibited, may cause frustration.
Failure mode: mismatched workspaces	Master does not lock up.	User can override the haptic feedback and work through the forces.	Surgeon cannot override the feedback. If workspaces misaligned, user cannot do anything about it.
Other - Implementation complexity	Da Vinci's ability to provide vibration is unknown. May have to hardcode a vibration function.	Requires the definition of a vector to generate forces, whereas the current system only includes position values	Simply make use of built-in DVRK control system to command master to return to boundary if the user ever exits.

4.3 System-level Overview

Figure 2 shows sample iteration of the movement of the da Vinci system starting from the user moving the master controller, sending that information to SIMULINK, calculating the required actuator values, and ending at the resulting movement on the CTR side. The proposed implementation of the admittance virtual fixtures is represented by the orange pathway and includes the addition of both the check function and the haptic feedback modules to inform the user when they have reached the workspace limitation of the CTR (i.e. Forbidden Region). Note that a more detailed description of the new modules is included in the module-level descriptions.

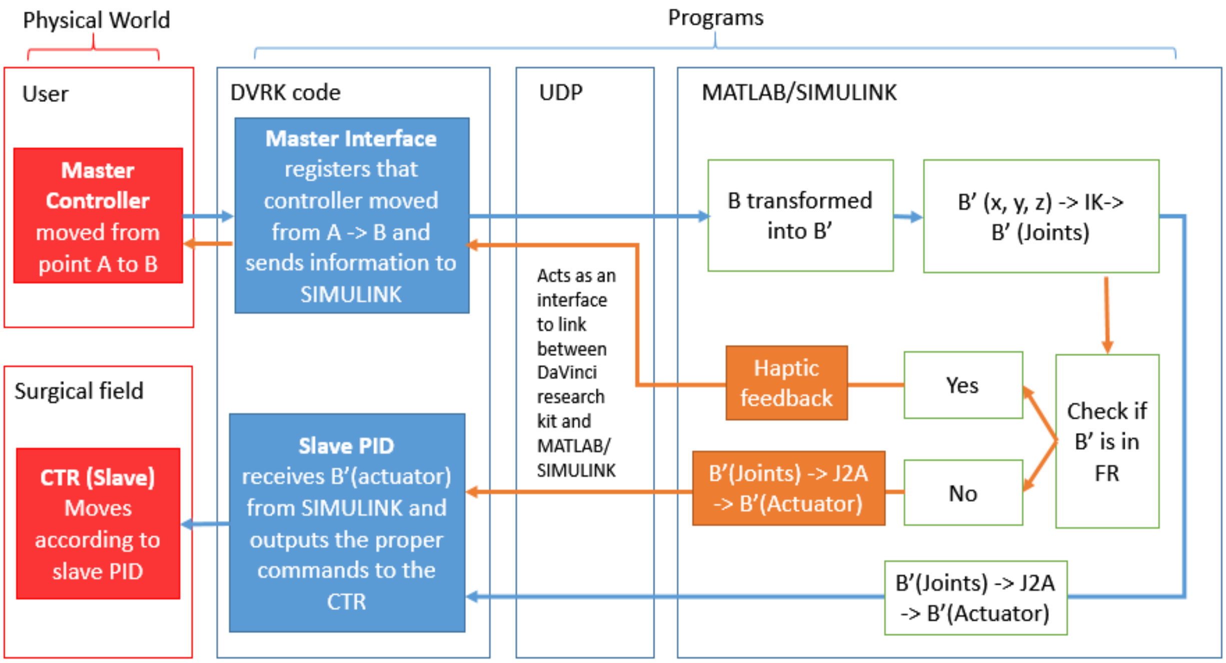
4.4 Module-level Descriptions

4.4.1 Check function: Intervention in the SIMULINK Model

The purpose of the check function module is to determine whether or not point B' (the desired location of the CTR) is in the FR. To do this, we can take advantage of the Inverse Kinematics (IK) (see Figure 2). This function takes B' as an input and outputs B* which is the "best possible" set of joint parameters that would provide Point B' through an approximation. If the point B' is within the FR then there will be a discrepancy between points B* and B', we will call this difference the error value. Then the check function would simply be to compare this error to a threshold (which must be defined empirically). The IK function, which uses the damped least squares method of approximation, takes this error quantity into account during its optimization to find the best set of joint parameters [6]. If the error is above the threshold, then the switch that sends the joint parameters to the actuators is set to "OFF" and the pathway that sends haptic feedback to the master is set to "ON" (these pathways are shown in red in Figure 3.) Note that the haptic feedback pathway is nominally "OFF".

Figure 3 shows a simplified version of the Simulink model, which performs the inverse kinematics to integrate control of the CTR with the rest of the DVRK system. The relevant components are numbered in green circles and are described below.

Da Vinci -CTR System Implementation Block Diagram

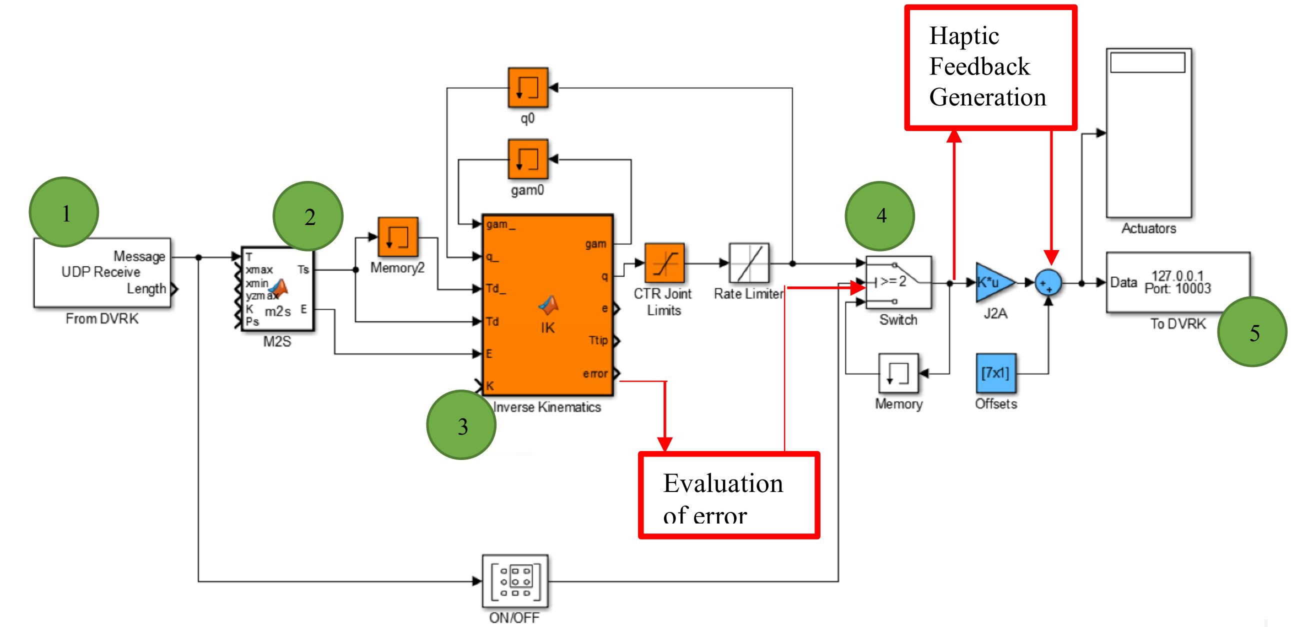


Legend

(blue line)	-	Original program workflow
(orange line)	-	Proposed implementation of haptic feedback
DVRK	-	DaVinci Research Kit, an open source version of the clinical da Vinci System with the addition of the sawIntuitiveResearchKit (open source software that monitors the manipulators and commands the desired positions)
A	-	Original position of the Master Controller in master reference frame
B	-	New position of the Master Controller in master reference frame
B'	-	Desired Position of CTR, obtained by transforming the position of the master into the slave coordinate frame
IK	-	Inverse Kinematics used to transform the desired position from (x,y,z) into the actual translation and rotation of each concentric tube such that the CTR will be at the desired position
J2A	-	Joint to actuator, transform joints into the actual disk angles required to create the desired translation and rotation of the concentric tubes
FR	-	Forbidden region (limitations of CTR workspace)

Figure 2: System level overview of the proposed changes to the communication architecture of the Da Vinci - CTR system.

Da Vinci – CTR Simulink Block Diagram



Legend

- Indicates existing relationship in Simulink model

- Indicates new relationship in Simulink model, that is to be implemented

1 - Block 1 - receipt of information about the position that the master has just moved to (Point B)

2 - Block 2 - transformation of the master’s position to the slave’s coordinate system (conversion of Point B to Point B’)

3 - Block 3 - Inverse Kinematics function transforms the desired slave position (Point B’) into joint parameters that can be sent to actuators to produce the appropriate movement achieve the desired slave position (Point B’)

4 - Block 4 - switch that regulates activity of the actuators and information that gets sent back to DVRK

5 - Block 5 - position information sent back to DVRK

Figure 3: Simplified model of Simulink code describing the major steps involved in integration of CTR control into the DVRK framework. Blocks in red are areas of intervention.

4.4.2 Haptic Feedback Implementation

The type of haptic feedback selected for this solution is a forbidden region model. This functions as a virtual wall on the master side, where the user cannot cross this wall. The virtual wall will be implemented by commanding the master to move back to its original position (point A) if point B is in the forbidden region (see Figure 4). This will effectively be a hard stop – the master side will move back to the desired location, preventing the surgeon from moving to areas that the CTR cannot reach. This check/move loop will happen quickly compared to what a user perceives, making it appear like a wall that cannot be crossed. Note that position A must be saved so that, if required, it can be sent back to the DVRK (see Figure 2) in order to command the master controller to return to A. By commanding the master side back to a specific location (point A), we will be making use of the actuation mechanisms and control that is already built into the DVRK.

5 Work Plan

5.1 Work Breakdown Structure

Table 4 represents a proposed breakdown of the work on this project. Refer to Figure 5 for a Gantt Chart style representation.

Table 4: Proposed work breakdown for this project, R – responsible, A - Assisting

TASK #	TASK	WORK BREAKDOWN
	Haptic Feedback Implementation	
6	Exploring how to create haptic feedback on the master without system implementation	AWL (R), KKT (A)
7	Implementing haptic feedback into system	KKT (R), AWL (A)
	Boundary Check Implementation	
8	Understanding inverse kinematics model for CTR	GW (R), SP (A)
9	Utilize inverse kinematics model with simulink	GW (R), SP (A)
10	Writing error-based check function	SP (R), GW (A)
11	Implementing error-based check function with rest of the system	SP (R), GW (A)
	Validation Testing	
12	Validation testing: user based survey tests	AWL (R), SP (A)
13	Validation testing: failure mode testing	KKT (R), GW (A)

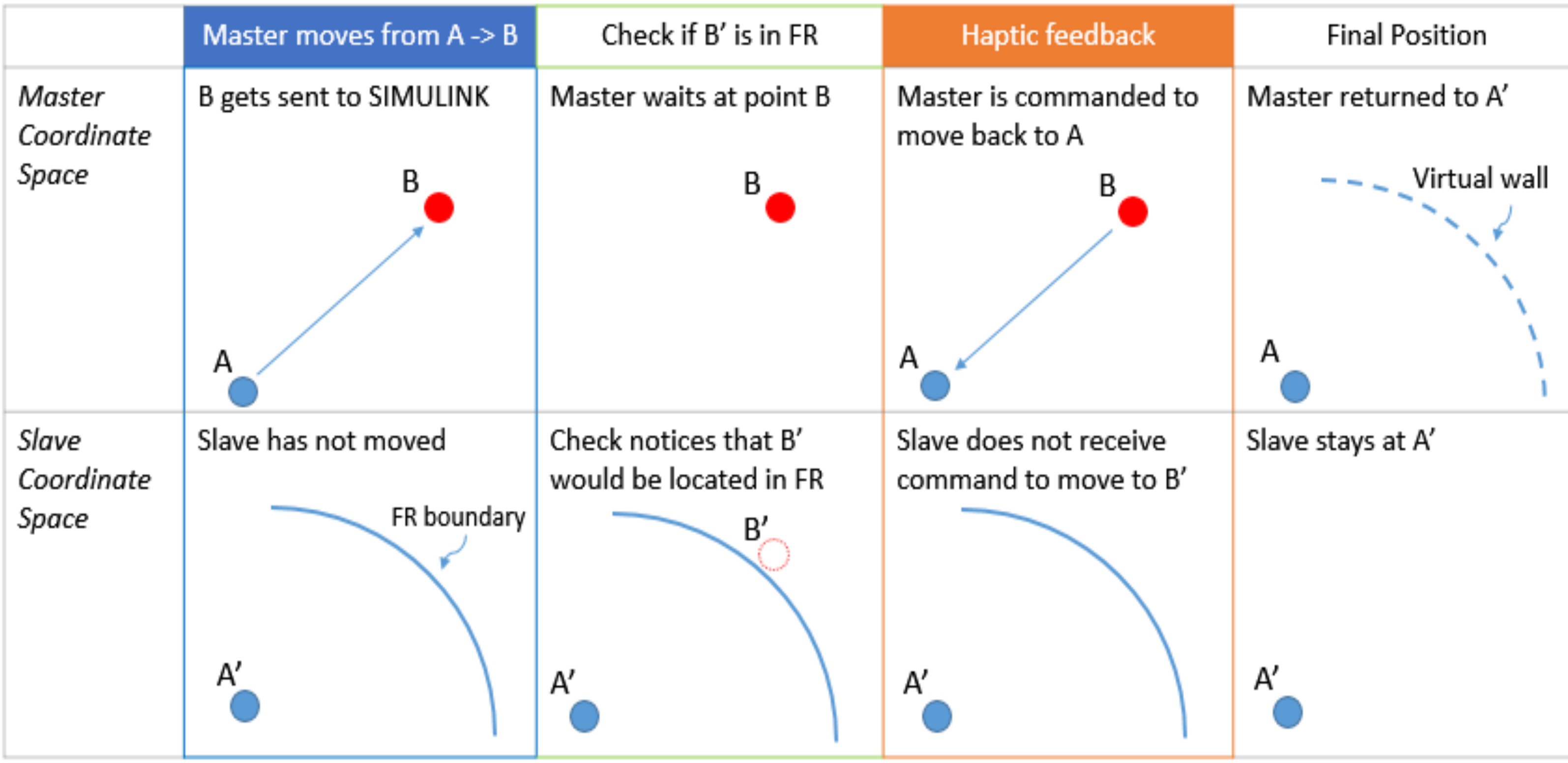


Figure 4: Detailed explanation of the Hard Stop type haptic feedback which compares the difference between the slave and master.

5.2 Gantt Chart

Figure 5 shows a proposed timeline for the completion of our work. Task Descriptions used here are equivalent to the descriptions used in Table 4 to make cross referencing easy.

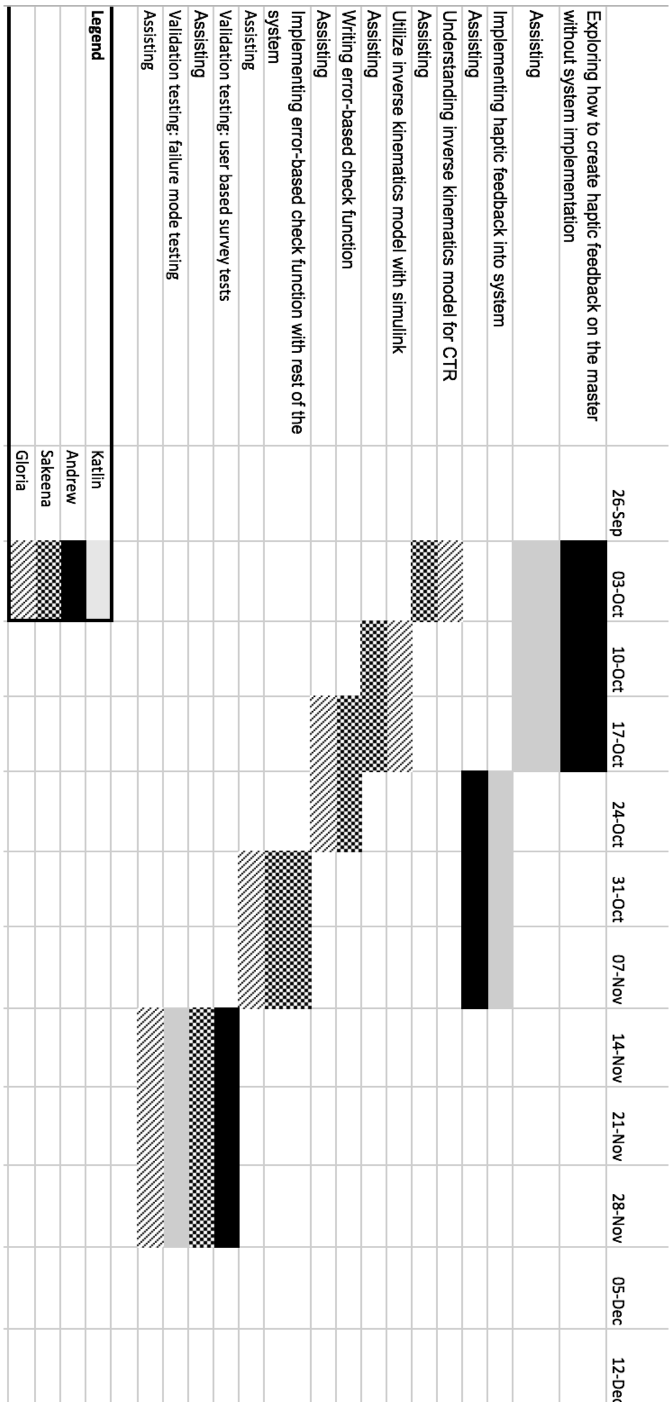


Figure 5: Gantt Chart for proposed work.

5.3 Financial Plan

Table 5 shows a proposed financial breakdown of the anticipated costs for this project.

Table 5: Proposed financial expenditure for this project.

Item	Details	Cost
Prototyping materials	May be used for final presentation, to demonstrate concepts or design. May be used to create testing fixtures to validate solution.	$200
Printing	Used to print final report, poster	$50
Software	E.g. Matlab, Simulink, C++, DVRK packages	$0
	Total	$250

5.4 Feasibility Assessment and Risks

Table 6 shows the tools, knowledge, skills, and resources we'll need to complete the proposed project.

Table 6: Breakdown of knowledge, equipment and software we'll need for the completion of this project.

Resource	Details	Do you have it?	Where can we get it?
Matlab and Simulink	Software licenses	No	EngSci Common Room; remote access to ECF computers from personal computers
DVRK system	Needed to test our implementation	Yes	Sickkids lab (CITIGI)
CTR tools	Tool being driven for our application	Yes	Sickkids lab (CITIGI)
Knowledge of Simulink model	Inverse kinematics is implemented here	No	CIGITI Lab members (Vivek, Peter), online resources
Knowledge of C++ & Surgical Assistant Workstation (SAW)	Needed to modify DVRK	Yes	Resources/tutorials/ documentation online
Users for usability testing of DVRK with our solution	Needed to test functionality of our solution	No	Sickkids lab (CIGITI)
Knowledge of Inverse Kinematics function	Needed to create check function for our design	No	Online resources, Sickkids lab members (Vivek, Peter)

One risk for this project is that our envisioned haptic feedback mechanism may require a master motor force too great for the intrinsic motors on the master controller to handle. To mitigate this, we can fall back onto one of our other haptic feedback solutions which require less force.

An additional risk is that the inverse kinematics error function may not work correctly to identify the forbidden regions. To mitigate this, a solution involving the implementation of forward kinematics can be used in lieu of the proposed.

6 References

[1] da Vinci Surgery, "Clinical Evidence," March 2015. [Online]. Available: http://www.davincisurgery.com/clinical-evidence.php. [Accessed 25 September 2016].

[2] A. B. R. K. A. S. K. Jain, "A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students," *Int J Appl Basic Med Res.,* vol. 5, no. 2, pp. 124-127, 2015.

[3] J. M. P. O. A. Abbott, "Haptic Virtual Fixtures for Robot-Assisted Manipulation," *Robotics Research,* vol. 28, pp. 49-64, 2007.

[4] A. Okamura, "Haptic Feedback in Robot-Assisted Minimally Invasive Surgery," *Curr Opin Urol.,* vol. 19, no. 1, pp. 102-107, 2009.

[5] A. Okamura, "Methods for haptic feedback in teleoperated robot-assisted surgery," *Ind Rob.,* pp. 499-508, 2004.

[6] S. Buss, "Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods.," 7 October 2009. [Online]. Available: https://www.math.ucsd.edu/~sbuss/ResearchWeb/ikmethods/iksurvey.pdf. [Accessed September 2016].

[7] M. S. D. Ritter, "Design of a Proficiency-Based Skills Training Curriculum for the Fundamentals of Laparoscopic Surgery," *Surg Innov.,* vol. 14, no. 2, pp. 107-112, 2007.

7 Appendix A: Report Attribution

KKT	Executive Summary
Project Description	
KKT, AWL	Background and Motivation
KKT, SP	Project Goal
AWL	Project Requirements
SP	Validation and Acceptance Tests
Technical Design	
SP, GW	Possible Solutions and Design Alternatives and Assessment (Virtual Fixtures)
AWL, KKT	Possible Solutions and Design Alternatives and Assessment (Haptic Feedback)
GW	System-Level Overview
SP, GW	Module-Level Description (Check Function)
KKT, GW	Module-Level Description (Haptic Feedback)
Work Plan	
AWL	Work breakdown structure (WBS)
AWL	Gantt Chart
AWL, KKT	Financial Plan
AWL	Feasibility Assessment (resources, risks)

By signing this page, I acknowledge that the above report attribution table for this proposal is correct.

Member: Katlin Kreamer-Tonin	Signature: K. Kreamer-Tonin	Date: Sept. 28, 2016
Member: Andrew Lim	Signature: [illegible]	Date: Sept 28/2016
Member: Sakeena Panju	Signature: [illegible]	Date: Sept. 28, 2016
Member: Gloria Wu	Signature: Gloria Wu	Date: Sept 28, 2016