# BiRTH:

# Biomimetic Robotic Teleoperated Haptics

Abrar Rahman, Kerestell Smith

# Contents

## Abstract

In robotic control applications, the robot operator needs to practice manipulating robotic appendages that are an abstract analogue of a human body part, such as a claw in place of a hand. This paper proposes a solution to remove the learning period to master the robot and to improve on fine control. This system is intuitive because it operates a robotic hand by mimicking the movements of a human hand. The fine control comes from position based force-feedback implemented using artificial tendons, similar to those of a human hand. To date we have implemented a prototype for a single finger with one-degree of freedom and have the design for scaling up to four-degrees of freedom.

# 1 Introduction

The goal of teleoperation is similar to that of remote control: to manipulate a machine from a distance [1]. One of the predecessors of teleoperation was puppeteering. Principally in teleoperation, there is a separation between the human and what they are controlling, like the gap between a puppet and the puppet-master which is only breached by string. As technology advanced, humans operated machines from control panels, separated from the object of their control by gears and wires. The true advent of teleoperation came from the development of radio and wireless communication. At this point, nothing visible was needed for a human to manipulate the machines.

Even now, there is still a connection to the puppet-masters of old. Puppeteers need to be trained in their craft; it was not as if the puppet just obeys mental commands. As such, operating puppets requires skill and practice, and not just anyone can put on an impressive performance on their first try. The same is true for modern teleoperated devices. Vehicles, heavy machinery and other tools require a learning curve to gain mastery. This is because the operation of these things is foreign to the regular operation of the body.

Even at the forefront of development, teleoperated devices today still use abstractions to perform the operation of an object. In many situations, the abstraction is made purposefully, to simplify the construction of the machine while sacrificing fine control, as with a carnival crane game. But there are some situations that require fine motor manipulation. Defusing a bomb is not something you would want done by a clumsy machine; any small mistake could spell disaster. Similarly, working in volatile environments or with hazardous substances is not loss tolerant work, but you would not want to be present for the task. Unlike the previous two situations where being present is undesirable, there are other cases when the urgency of a task, such as surgery, prevents the luxury of being on site for the operation.

One solution to this problem is to have people trained especially for these kinds of tasks using abstracted controls. A well known example of this is the bomb defusing robot. It needs a deft and practiced hand to manipulate the controls of an awkward interface, such as a pincer claw, to defuse a bomb [2]. Robotic paradigms, such as claw arms, are used because of their simplicity. They operate with one-degree of freedom on one joint, reducing both the computational and building complexity. At the same time, the freedom and agility of a real human hand is compromised.

Another factor that promotes the abstraction paradigm is the control interface. Buttons, levers and joysticks do not allow for easy translation to multidimensional control. Also, these interfaces differ slightly from application to application, changing sensitivity and speed for appropriate situations. It takes some fiddling to get used to the controls, and even then that does not guarantee competency. Therefore, at its theoretical best, the optimal system is one that moves like a human hand. Furthermore, the optimal device would need the most natural way of interfacing with the human hand; in other words, there would be a one-to-one mapping between the movement of a human hand and the movement of the device.

Even with a robotic hand that mimics a human hand, the model breaks down when the robot hand is unable to move to a position the human hand indicates because of a physical obstruction. With visual feedback, the human controller could choose to avoid that position, but the robotic hand would be useless if it could not interact with physical objects. Therefore the human hand not only needs visual feedback, it also needs to feel what is happening at the other end. This can be done with touch sensation, or more informatively, feedback information about the forces impacting the robot hand. With force being fed back to the human controller, they should be able to interact with objects as if they were using their own hand, making it an extension of their own body.

To better model the movements of a human hand, the paradigm of biomimetics is followed. The principle of biomimetics is to use biological systems or models for the design and construction of technology [3]. This principle is specifically used for the design of the robotic finger and glove part. The model for the finger and glove draws from the model of the human hand, which uses tendons for actuation.

The eventual goal of this project is an artificial hand that can be manipulated by a human hand with one to one correspondence and force feedback control. In layman's terms, the project is to construct a robot hand to mimic the movements of a gloved hand. The robotic hand is intended to copy the human hand in real time. The force feedback control is meant to let the gloved hand feel the resistance that the robotic hand encounters. The robotic hand is intended to have all the degrees of freedom that digits of a human hand has.

## 2 Controlling a Robotic Hand

There are two main goals to achieve through controlling the robotic hand and the glove. The first is to ensure that the robot hand matches the movement of the glove, and the second is to ensure that the external forces the robot hand experiences are translated back to the glove. These goals can be achieved in many ways, but this implementation requires two main components: actuators that physically apply forces to both the robot hand and the glove, and sensors that can measure the position and forces on both the robot hand and the glove.

For the actuation of the components, there are many options. For example, there is a variety of motors that could be used, including normal DC motors, single or multiphase AC motors, servos, stepper motors, etc. Linear actuators could also be used, as well as hydraulics or pneumatics [4]. The methods in which these motors could be applied to the system also vary, such as the decision between putting a motor in an individual joint, versus transmitting the force using a connecting element like a rope, a wire or a bar. Packaging of the force actuator is a very large component of the project, so factors like size, power and cost become very important considerations.

Since the focus is in trying to create a system that models the human hand as accurately as possible, one of the main methods investigated for applying torques to joints has been using a system of artificial tendons [5]. This way, the power actuators are located separately from the joints they are actuating, which allows for both larger and more powerful actuators to be used, as well as more compact geometry for the hand itself. There are several benefits to this approach relating to cost, power transmission, and geometry concerns. Smaller components that have the required strength are very expensive, so this method allows us to use larger but cheaper actuators. It is also relatively easy to transfer forces through this method; artificial tendon sheaths, which can be something as simple as tubing, allow the tendons to be routed through the machine directly to where they need to go without interfering with any other parts. Finally, since the actuators do not need to be placed inside the hand, this method would allow us to package the actuators separately, which greatly simplifies the design of both the hand and the glove.

Similarly, there are many options in terms of sensors [6]. To determine the position of the robot hand and glove, potential sensors include potentiometers, optical encoders, magnetic field angle sensors, flex sensors, and fiber optic angle sensors, to name a few. To determine the forces on the hand and the glove, there are also a few options, such as using force or pressure sensors, which are relatively expensive and difficult to use to measure torque. However, there is another option that does not require a sensor to directly measure force; this is known as sensorless detection. Essentially, rather than directly measure a certain parameter, a sensorless detection method involves determining what the measure must be depending on other factors that can already be measured, possibly through other sensors [7]. In the case of measuring the forces on the hand, this can be done simply through knowing the position of the hand. If the position is known over time, then the velocity and acceleration can be easily determined as well. Based on the acceleration, the current net torque on the finger or joint can be easily calculated as follows:

$$T_{net} = \frac{d^2\theta}{dt^2} * I$$

where $\theta$ is the position, t is the time interval, and I is the inertia of the system. Since the input torque applied to the joint is known, the difference between the net torque and the input torque can be determined. This difference is the external torque:

$$T_{external} = T_{net} - T_{input}$$

One drawback to this method is that, although it requires fewer and cheaper sensors, it is only really useful for measuring torque and not force, so it is impossible to know where the external forces are being applied, and only what the magnitude of their resultant torque is.

## 2.1 PID Control

The Proportional Integral Derivative (PID) controller is one of the most widely used feedback controller systems used in industry [8]. It is used in the maintenance of environmental factors such as temperature and pressure as well as motorized systems. The first implementation of PID was needed for the purpose of automatic ship steering [8]. For the ship steering application, the PID controller adjusted the ship's course to compensate for changes in water currents. The equation governing PID is given by:

$$\mathrm{u}(t) = K_p e(t) + K_i \int_0^t e(\tau)\mathrm{d}\tau + K_d \frac{d}{dt} e(t)$$

where $K_p$, $K_i$, and $K_d$ are the gain weights to each of the P, I and D components [9].

The simplest portion of the PID controller is the proportional, or P, component [9]. In implementation, it is just the error fed back into the system with a gain. This drives the system from its current value toward its desired value. For the simplest of feedback controls, proportional feedback is enough. The problem with only proportional control is that if it overshoots its desired value, then the system will oscillate around its target instead of actually reaching it. To compensate for this problem, a integral component is needed.

The integral, or I, component of a PID controller affects the speed at which the system gets to its target value [9]. The integral component accounts the accumulation of past errors. If the system is far from its target, then the I component grows large and strongly drives the system. But as the desired value draws near, the integral component dies out. Because of the strong drive that the integral gives at the beginning, this component tends to cause the system to overshoot its target and then oscillate around the value. With an appropriate gain on the integral component, the oscillations can be dampened.

To deal with the aforementioned overshoot, the derivative, D, term is needed [9]. The derivative component is the slope of the error and affects the rate of change at which the system changes. The derivative component slows the system down, decreasing the tendency to overshoot, but since the system is slower, any oscillations that occur would be drawn out and linger longer.

Changing the gains on each of the components is a balancing act between the speed of the system and the magnitude of the overshoots. It is fairly easy to go for one characteristic over the other [9]. A high $K_i$ would cause a fast overshoot and then oscillate. This behavior is termed an under-damped system. A high $K_d$ would have a slow crawl toward the target, but would not go past it. This response comes from an over-damped system. Somewhere in between is the fastest drive to the target without oscillations. This is a critically damped system. The desired response depends on the application of the system itself. In some situations oscillations are not an issue, and in others you would want a slower response for safety reasons.

The process of choosing gains for the components is called tuning [9]. This can be done based on details of the system itself to get theoretical tuning values. They can also be found by testing the system at different values and experimentally obtaining the desired response. The tuning could also be done adaptively, so the gains would change based on the given input.

## 2.2 Microcontrollers

Microcontrollers are necessary to perform computation without being hard-wired to a computer. The simplest microcontroller setups consist of only an integrated circuit and a crystal oscillator [10]. The microcontrollers themselves are programmed through a computer. Depending on the microcontroller brand, they are programmed in a variety of languages; the most basic microcontrollers are programmed in assembly language.

For microcontrollers coded in assembly, most lines of code take one clock cycle of the crystal oscillator to execute. With these programs, it is fairly easy to predict how long they will take to run. For microcontrollers that use higher level languages, unless the underlying workings of each function is known, it is difficult to predict how long a program will take to execute. This is a problem when high speed operations, like sampling, are required. If the code runs longer than the sampling period, then data points are missed and errors accumulate. Printing values are especially time consuming, so it might be difficult to monitor parameters in real time if speed is of the essence.

## 2.3 Related Work

There are a three main areas of background research:

1. It is necessary to explore other types of interactive, force feedback mechanisms. This includes apparatuses that are not necessarily attempting to be employ a one to one correspondence between the machine and human, including mechanisms where the control is abstracted. An examples of this is a teleoperated surgery robot. Applicable principles that can be used include how the parts communicate, how the force is transferred, and how the robot senses forces and motion.

2. It is also necessary to investigate mechanisms that attempt to replicate the human hand, to see how they solve specific problems such as generating torque, measuring positions, solving geometries, and so on.

3. Furthermore, it is necessary to research control techniques, such as PID, so that we can control our system rapidly and intelligently.

Related to the first point, one of the most popular robotic surgery device is the da Vinci surgical system [11]. This is a robot that is capable of doing complex surgery through tiny incisions in the body. Usually, the operator of the machine is in the same room as the robot, so this is not a long-range teleoperated device. However, it does allow surgeons to perform surgery using less damaging methods to the body, which allows people to recover faster.

To control the robot, there are two small mechanisms that you grab, squeeze, twist and turn, and the movement of these grips translates to the robot's movement. The system also includes some force feedback, although it is somewhat unclear as to the magnitude at which this force feedback will actually push back.

Another system investigated was one created by professors across a few universities who were attempting to create a "4 DOF portable haptic interface... for the index finger" [12]. The device that they ultimately created fit over an index finger and was capable of exerting force on different parts of the finger to keep it from moving when it moved into the path of a virtual object. They used a brake, similar to that of a bike, on the hand when it hits the virtual object. There is no smooth application of force, so the hand essentially hits a "wall" and is immediately stopped whenever it comes into contact with something; this hand is limited to interacting with hard surfaces. Another issue with the hand is the joints; they are not sufficiently constrained, so there is extra give and movement of the hand, and its position cannot be evaluated precisely.

Another relevant system is the CyberForce CyberGlove system [13]. This system uses what they call a CyberGlove that fits over the whole hand. Embedded in the glove are sensors that detect the position of the hand, which is then read by a computer. There is an additional apparatus that is placed over the glove which is able to translate force to the hand. This provides the capability of interacting with virtual objects, and there are a few videos that show the system in use. A couple of the problems with the system are that there is only one-degree of freedom for each finger, so only the fingertips have forces exerted on them, and there are relatively low forces being applied.

The second avenue of inquiry is necessary because the project involves not only systems of transferring forces to an actual hand, but also fabricating a robotic hand. Two hands were researched related to this: the Shadow Hand, created by the Shadow Robot Company [14], and a robust robotic hand created by the DLR Institute of Robotics and Mechatronics in Germany [15].

The Shadow Hand is actuated by pneumatics or motors, depending on the version. It has sensors that detect both position and either pressure in the pneumatic hand or tendon force in the motor version of the hand. It uses a system of tendons to connect the power source to the fingers of the hand, so all the pneumatics and motors can be placed outside of the hand, which helps greatly with size constraints. There are pressure sensors on the fingers which allow for sensing textures and finer details of objects, as well as forces on the fingers themselves. There are some videos of the hand grabbing an egg or a light bulb without damaging either, so there is also very fine control of the forces being exerted by the hand on its environment.

The DLR hand is similar in that it uses tendons and motors, and has two motors per degree of freedom. This device uses two motors to act in opposition on each degree of freedom. This helps make the hand more robust by allowing it to change stiffness depending on the need. It also uses a special elastic material for the tendons that can store energy which helps the hand stand up to strong impulses or impacts. To check what kind of forces the hand is experiencing, there are sensors that measure the elongation of the tendons, which correspond to what forces are acting on those tendons.

In terms of controlling the glove and hand assembly, [16] discusses the use of Kalman Filters to control power based on a human actor. While [16] is not directly applicable to this project, it does discuss in detail different methods of controls and filters which are necessary to effectively control the system. It describes how to model a system, how to use sensors to determine how the model is acting, and how to use both a feed-forward and feedback system to control the system.

# 3 Project Description

## 3.1 One-Degree of Freedom Model

The material ultimately chosen for this project was ABS plastic [18]. There were a few different factors that led to this decision. The most important one was that it allowed rapid prototyping of different setups, since it was realized early that the anticipated parts of the prototype would need to be redesigned due to unforeseen issues or mistakes made in designing the parts. ABS allows the use of a laser cutter, which can fabricate many different parts out of ABS extremely quickly. The constraint that this placed on the project, however, is that it is only possible to make two dimensional objects out of the ABS itself. This has led to a few problems in fastening the parts together. As a result, most of the ABS parts are connected to each other orthogonally using tabs and screws with nuts to fasten them together. These considerations cause the design to be quite difficult in some areas due to the size of the ABS components that needs to be fastened together as well as the relative size of the fasteners being used. Consequently, large periods of the total time spent on the project have been used to design a prototype that has everything fit together correctly.

The actuators chosen for the project are DC motors [4]. When compared to other types of actuators, like hydraulics and pneumatics, electric motors are small, cheap and simple, meaning that they should be able to be incorporated into a more compact design. They can be relatively weak, but the powers needed are not excessive, so this is not a problem. DC motors have been chosen as opposed to AC motors since they are generally smaller and easier to use. One of the most important reasons is that the torque they output is very simple to control. The method used is to supply them with a pulse-width modulated (PWM) voltage. At every speed of the motor, there is an associated torque rating, which is controlled by the duty cycle of the PWM. As long as the revolutions per minute are accurately measured, the torque that the motor is outputting is known. Depending on the desired torque, a PWM signal is output to the h-bridge to drive the motor. Depending on the duty cycle, the motor is powered for that fraction of the time. The behavior of the motor is such that this PWM signal effectively modulates the torque of the motor; that is, at a duty cycle of 50%, the torque the motor is outputting is half of what it is rated. This behavior is what makes it simple to control and measure what kind of torques different parts of our system are outputting.

The position of the finger is a function of the rotation of the motors, so tracking the rotation of the motors records the movement of the finger. The position of the motors are tracked by optical incremental encoders. The motors turn a code wheel that have tick marks along the outside. That outside edge passes through a sensor module which is composed of a ultra-violet emitter and two light detectors. The detectors are 90° out of phase with each other, so they have the same value half the time as the code wheel passes in between the emitter and detectors. Whichever detector changes its state first is indicative of which direction the motor is turning. It is important that the mircocontroller's sampling of the detector outputs is fast enough to catch all the of the transitions, or the position values will not be accurate.

The prototype model is split up into three sections: the electronics, the motors, and the glove and hand sections. Figure 1 shows the general layout of the system.

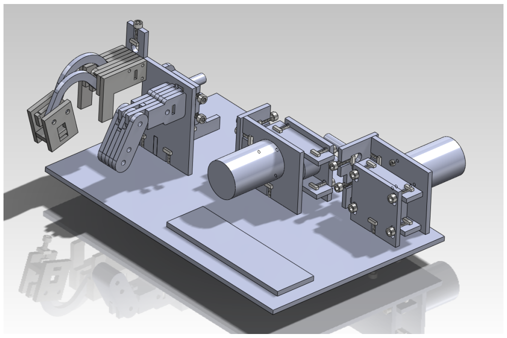


Figure 1: General Layout of System

The electronics are mounted mostly on a breadboard and an Arduino microcontroller, which are both attached to the ABS board using adhesives. The sensors and motors are connected to the breadboard, which houses an h-bridge that is used to drive the motors. Everything is connected to the Arduino which controls the system.

The motor section is where the motors are mounted to the ABS board. Attached to the motors are the pulleys which pull on the tendons that are connected to the glove and the hand. Also attached are the motor encoders which determine the position of the motor, and as a result, the position of the hand and the glove. The tendons which are attached to the pulleys are then routed through plastic tubing, which brings them to the hand and the glove. Figure 2 shows a detailed view of the assembly.

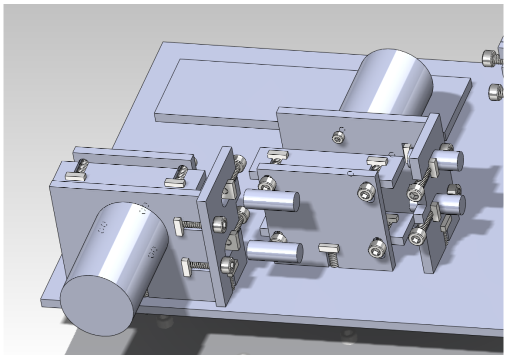


Figure 2: Motor Section of System

The robot hand and glove assembly is where the actual robotic finger with one-degree of freedom, as well as the single degree of freedom glove, is located. The finger is mounted horizontally so that it can match the movement of the glove, which is mounted in a way that is easy to manipulate. The setup also allows for something to be mounted beneath the robotic finger, so we can simulate a few different situations, such as a spring or a rigid object. A detailed view of this assembly is shown in Figure 3.

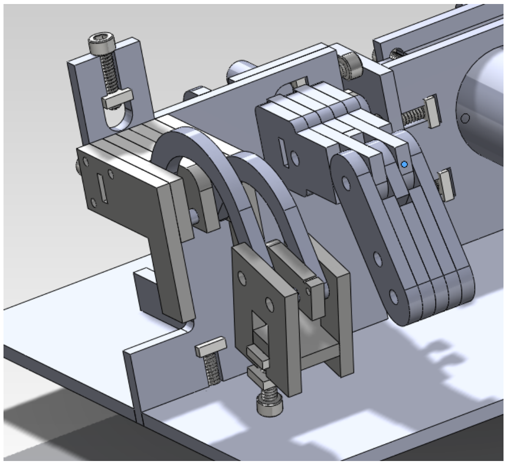


Figure 3: Glove Finger and Robotic Finger of the System

Two of the most design intensive parts of the project are the geometry of the glove assembly and the motor assembly. For the glove, this is a result of the requirement that the user should be able to put their finger in it and be able to move their finger freely without restrictions. Also, the glove itself should not interfere with other fingers or the glove assembly. Thus, the glove is located mostly above the finger. This also necessitates the design of a joint that does not have some kind of pin that the assembly can rotate around, but rather has a center of rotation that is located in the person's joint, without any material actually being in that location. This makes the design much more difficult, although there are a few methods that can be used to get around this limitation. One method is to create a circular device that has a center of rotation at the center of the finger, as shown in Figure 4.

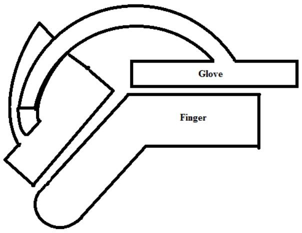


Figure 4: Circular Joint

The design in Figure 4 will ensure that the glove assembly rotates correctly. However, constricting the movement of such an assembly is difficult without introducing large frictional areas or jamming forces. Furthermore it would require either a relatively large glove or one with very small, highly engineered components, such as the one used by Mark J. Lelieveld [12] shown in Figure 5. Experiments with this glove have shown that it is not completely constrained, and there is some extra "wiggle" as a result, which is undesirable.

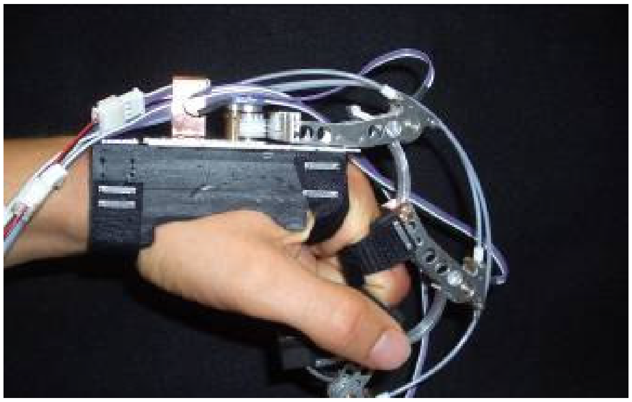


Figure 5: Lelieveld's Four Degree of Freedom Hand

An alternative is to use a four bar linkage system. This approach is described in [17], which investigates a knee joint. Essentially, this system allows for both rotational and lateral movement around a joint, which in our case is supposed to surround a joint rather than simulate one. Depending on the sizes of the four bars, the motion that the joint undergoes changes. The path of the joint that the bars describe is very sensitive to small changes in the lengths of the different bars, as well as the distance from the bars to the joint. Figure 6 shows some of the variables that go into the system. As a result, there was a lot of trial and error that went into the design of this glove assembly.

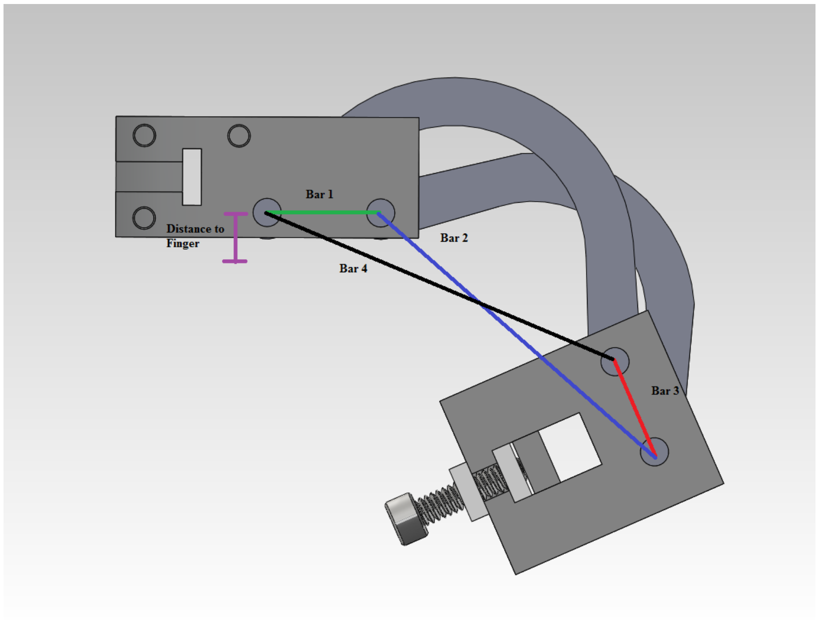


Figure 6: Variables involved in determining Four Bar Linkage Geometry

The geometry of the motor assembly is also very particular. One of the goals of the project is to keep the costs as low as possible, and one way this has been accomplished is through the design of the pulleys that connect the motors to the joints. A cursory examination of the system reveals that it makes sense to limit the number of motors and sensors to one per degree of freedom for both the hand and the glove. However, this is not a simple feat to accomplish. Because of the use of artificial tendons in the project, rather than directly having a motor in each joint, the lengths of the tendons that connect to the top and bottom part of the joint are different. Since it is desirable to use one motor per joint, the geometry of the two pulleys need to be such that there is no slack in either pulley, since this would introduce backlash and would greatly increase the response time and the inaccuracies of the system. However, the geometries involved are very complex, and a solution to them has not yet been discovered. So, for the first prototype, the system only works in one direction, the "curling" of the finger, since that movement of the finger is what is used to interact with most objects. The assembled one-degree of freedom model can be seen in Figure 7.

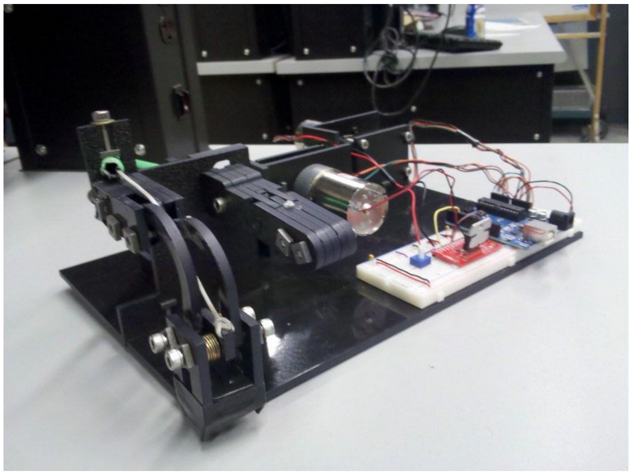
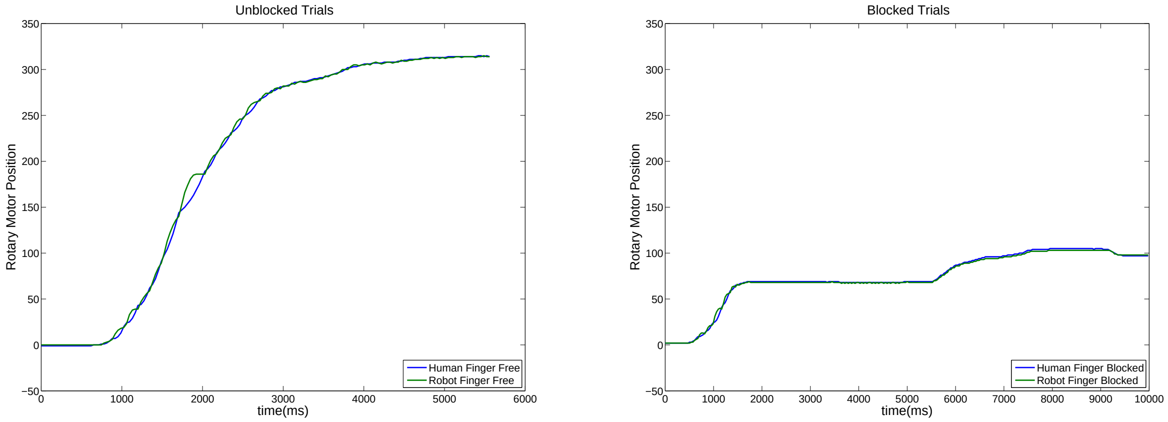


Figure 7: One Degree of Freedom Model

## 3.2 Results

To test how well the robotic finger matches the movement of the human finger, the motor position was monitored for both fingers. The finger position is a function of the motor rotation, so if the motor positions track each other, so do the finger positions. The free movement trials show the behavior of the finger positions when the human finger is flexed while inside the glove. The impeded movement trials included a solid object placed in the path of robotic finger. The human controller flexed their finger until they felt resistance, then attempted to overcome that resistance.



(a) An unobstructed sweep of the range of motion

(b) Attempting to overcome an obstacle

Figure 8: System Response with and without an Obstacle

The motor positions for these trials are shown in Figure 8. As can be seen in the graphs, the robotic finger closely follows the movement of the human finger. In unobstructed trials some oscillatory behavior is evident in the system, but the oscillations are small and can be reduced with better tuning of the PID controller. In the impeded movement trials, the point at which the robot finger encounters the obstacle is characterized by the position plateauing. Then when human finger attempts to push past the obstacle, the system does move forward a bit, and the human finger does move slightly past the robotic finger. But when pressure is released, the system moves back as the system settles. It is evident that the position value drifted when trying to overcome the obstacle, as seen by the fact that the settling position is a significant distance away from the position at which resistance is first encountered.

## 3.3 Continuing Work: Four-Degree of Freedom Model

Based on the previous model, there are a few adjustments that needed to be made before continuing on to a higher degree of freedom model. First of all, the DC motors used did not respond to a PWM voltage below approximately a 35% duty cycle. In addition, as the complexity of the system increases, there is also concern that the microcontroller is able catch all the transitions of the optical encoder. Furthermore, the geometry for a pulley that would allow one motor to control bidirectional movement was not resolved.

The first problem regarding the motors is solved by using different motors. The issue with the previous motors is that the inertia of each motor is too great for it to be driven by a lower duty cycle PWM voltage. The solution that has been implemented previously for the one-degree of freedom model implemented was to have the minimum duty cycle of the PWM signal saturate at 35%. Although this solution was sufficient for the previous model, it still represents a loss of a large scope of available speeds that is necessary for fine control, especially at the low end of the range. For the four-degree of freedom model, the DC motors have been exchanged for weaker motors. Weaker motors do not have the added inertia from the gears, making them more responsive, though weaker. Weaker motors are not an issue because the model is intended as a proof of concept and not meant for industrious use.

The solution to the second issue involves changing the type of sensors used to read the position. The current model monitors the transitions of the signal from the incremental encoders to detect whether to count up or down for the position. The algorithm that the microcontroller implements alternately reads the states of the encoders and updates the PID controller. If the states change faster than it takes for the PID controller to execute, then the position measurement used in the algorithm will be erroneous. The current setup does not scale to higher degrees of freedom for this reason; as the number of degrees of freedom increases, so does the computational complexity of the algorithm. To remove the limitation placed by the encoders, an absolute position sensor can be used. With an absolute position sensor, there is no minimum sampling rate required to keep track of the position. The actual sensors on the model are Hall-effect magnetic sensors. Unlike the previous model, where the sensors were housed at the motors. The Hall-effect sensors and magnets are embedded at the joints of the robot hand and glove, as seen in Figure 9 and output an angular position based on the relative positions of the sensor and magnet.

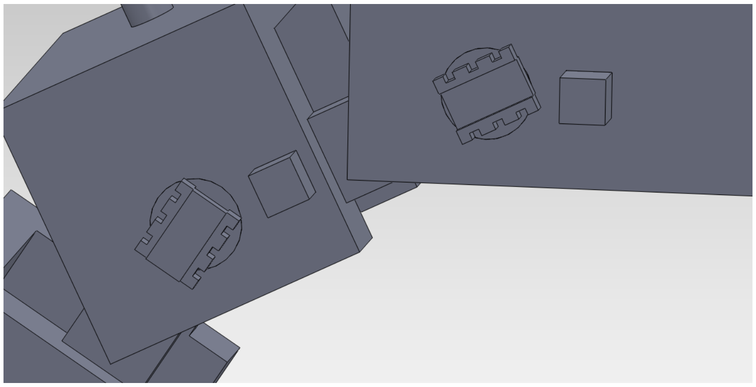
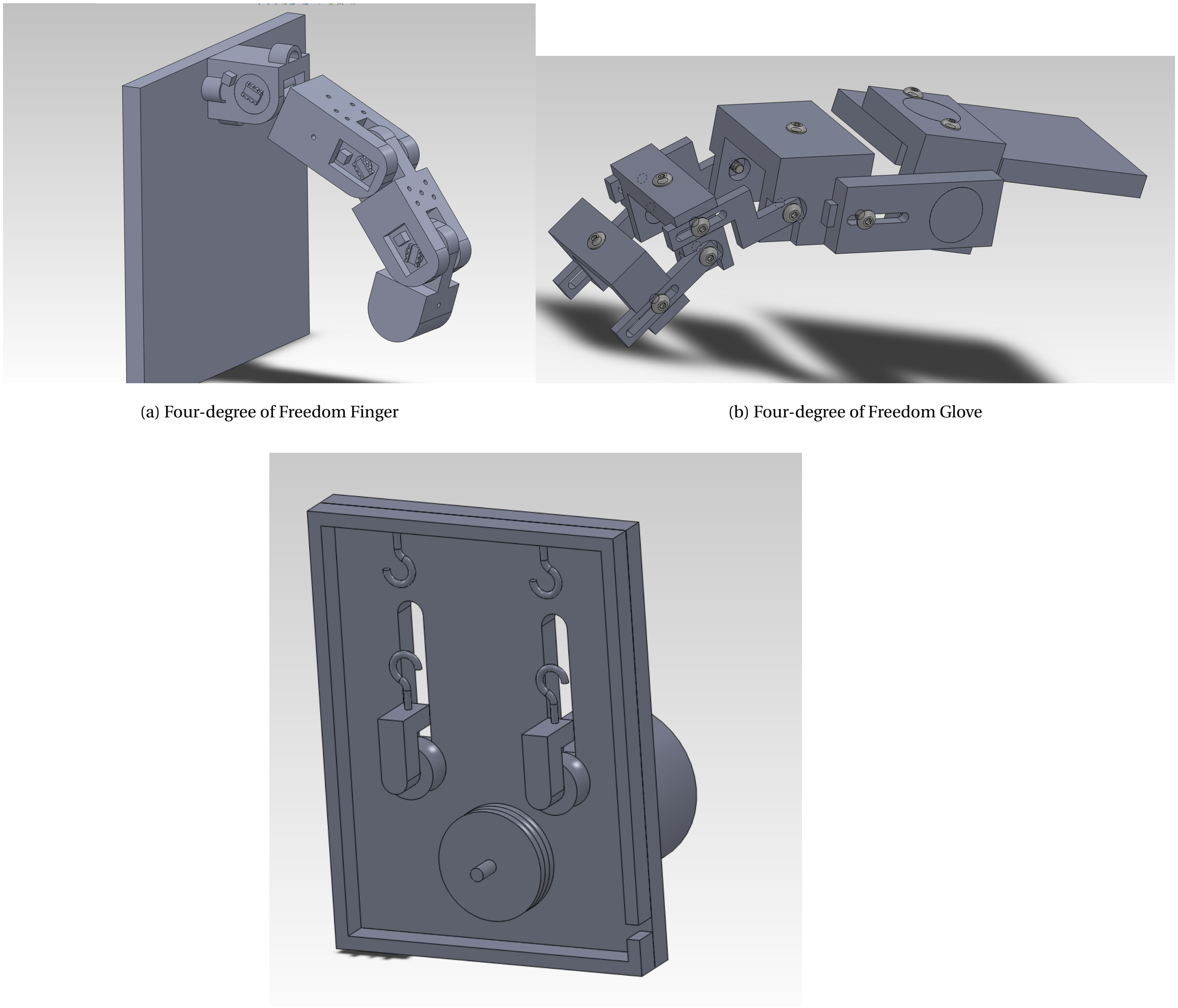


Figure 9: Embedded Sensors and Magnets at Joint

Because

The reason bidirectional movement was not implemented for the one-degree of freedom model is due to the geometry required by the pulley. This is where changing position sensor types provides an added advantage. The smaller absolute position sensors can be placed at the joints rather than the motors, allowing tensioners to be used. When the sensors are placed at the motors, the lengths of the tendons need to be manipulated so that they are a function of motor rotation. With the sensors at the joints, the motors can be used to adjust the tension on the tendons with the use of tensioners.

Another difference in the four-degree of freedom model is that instead of consolidating the whole system on one ABS platform, it is split between three platforms machined from aluminum. The three platforms house the motors, the hand glove and the robot hand which can be seen in Figure 10. This setup better elaborates the tele-operation nature of the system.



(a) Four-degree of Freedom Finger

(b) Four-degree of Freedom Glove

(c) Motor Mount

Figure 10: Finger, Glove and Motor Platforms

# 4 Conclusion

This project demonstrates the legitimacy of accomplishing one-to-one force feedback control with current technology. The one-degree of freedom model shows that the system closely tracks and mimics the movement of a human finger and is able to provide force feedback to the operator. The positional drift that occurs when an obstacle is present is most likely do to the shifting of the motors being interpreted as forward movement by the incremental encoders. This further highlights the necessity to use absolute position sensors. With the simpler system validated, it can now be expanded for an application of higher complexity, namely full four-degree of freedom movement of the human finger. With the plans for the four-degree freedom model done, all that needs to be done is to build and test it to demonstrate the system's scalability.

# 5 References

1. N. Tesla, "Method of and Apparatus for Controlling Mechanism of Moving Vessels or Vehicles." U.S. Patent 613 809, Jul. 1, 1898

2. C. Fielding, "Can This Bomb-Defusing Robot Take Care of Grandma, Too?" http://gigaom.com/2011/05/25/can-this-bomb-defusing-robot-take-care-of-grandma-too/, May 2011.

3. J. Vincent, O. Bogatyreva, N. Bogatyrev, A. Bowyer, A. Pahl, "Biomimetics: its practice and theory." Journal of The Royal Society, Interface, Vol. 3, No. 9, pp.471âĂŞ482, Aug. 22 2006.

4. H. Jack, "Implementation," Engineer on a Disk
http://engineeronadisk.com/V2/book_implementation/engineeronadisk.html, Feb 2010

5. S. Jung, S. Kang, M. Lee, I. Moon, "Design of robotic hand with tendon-driven three fingers", Control, Automation and Systems, 2007. ICCAS '07. International Conference on, pp.83-86, 17-20 Oct. 2007.

6. "Displacement and Position Sensing" Measurement Computing
http://www.mccdaq.com/PDFs/specs/Displacement-and-Position.pdf

7. J.C. Gamazo-Real, E. Vazquez-Sanchez, J. Gomez-Gil, "Position and Speed Control of Brushless DC Motors Using Sensorless Techniques and Application Trends," Sensors, Vol. 10, No. 7, pp. 6901-6947, Jul. 2010.

8. "Our fascinating look at building automation history"
http://www.building-automation-consultants.com/building-automation-history.html, 2008.

9. Y. Li, K.H. Ang, G.C.Y. Chong, "PID control system analysis and design," IEEE Control Systems Magazine, Vol. 26, no. 1. pp. 32-41, 2006.

10. S. Heath, "Embedded systems design," Newnes, 2 ed, pp. 11-12, 2003.

11. R. Ashton, C. Connery, D. Swistel, "Robot-assisted Lobectomy," The Journal of Thoracic and Cardiovascular Surgery, Vol. 1, No. 126, pp. 292-293, Jul 2003.

12. M. Lelieveld, T. Maeno, T. Tomiyama, "Design and Development of Two Concepts for a 4 DOF Portable Haptic Interface with Active and Passive Multi-point Force Feedback for the Index Finger," ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Sept. 2006.

13. J. Zhou, F Malric, S Shirmohammadi, "A New Hand-Measurement Method to Simplify Calibration in CyberGlove-Based Virtual Rehabilitation," IEEE Transactions on Instrumentation and Measurement, Vol. 59, No. 10, pp. 2496-2504, Oct. 2010.

14. "Design of a Dextrous Hand for advanced CLAWAR applications," CLAWAR,
http://www.shadowrobot.com/downloads/dextrous_hand_final.pdf, 2003.

15. E. Guizzo, "Building a Super Robust Robot Hand,"
http://spectrum.ieee.org/automaton/robotics/humanoids/dlr-super-robust-robot-hand, Jan. 2011.

16. C. Mitsantisuk, S. Katsura, K. Ohishi, "Kalman-Filter-Based Sensor Integration of Variable Power Assist Control Based on Human Stiffness Estimation," IEEE Transactions on Industrial Electronics, Vol. 56, No. 10, Oct 2009.

17. A. Hamon, Y. Aoustin, "Cross four-bar linkage for the knees of a planar bipedal robot," 2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids), pp.379-384, 2010.

18. "Plastic Properties of Acrylonitrile Butadiene Styrene (ABS)"
http://www.dynalabcorp.com/technical_ info_ abs.asp