Mapping Mobile Robot States to an Upper Body Haptic Feedback Interface

Student: Ibrahim Youssef
Advisors: Amy Wu, Vivek Ramachandran
Professor: Auke Ijspeert

June 2018

Contents

1 Introduction

The integration of haptic technology to control electronic devices provides users with an additional dimension of feedback. Research into this field extends as far back as the 1970s, where investigators identified performance benefits to force feedback for operators carrying out simulated space tasks [1]. Today, one of the most common forms of haptic feedback is the use of vibrators in electronics systems. Commercial instances of vibration haptics were introduced as early as the 1990s, when they were used to add an entirely new dimension to video game controllers [2]. Haptic interfaces have since been integrated into a much wider range of common devices, which has had the effect of increasing both the usability and quality of these products. Besides being nearly ubiquitous in modern touch-screen smart phones and video game consoles, haptic technology has been integrated into applications including surgical devices [3], audio systems [4], and UAV control interfaces [5].

This work developed an experimental design to evaluate the performance of a novel haptic feedback interface based on flexible electroadhesive clutch plates. A mobile robot and controller were designed in order to take advantage of this haptic feedback. The objective of the experiment was to test how well operators could guide the mobile robot through an obstacle course using only sensory information provided by the flexible clutch plates. By strategically mapping the mobile robot states to clutch activation patterns, end users were supplied with feedback information that allowed them to navigate out of a maze.

This project involved the implementation of three different control strategies that were intended to make use of a previously developed haptic sleeve [6]. We investigated the effectiveness of the different strategies, and finally tested the best of these three strategies on a group of volunteers. It was found that with the appropriate mapping and control paradigms, device operators were able to successfully navigate the mobile robot.

2 Methods

The objective of this work was to evaluate the performance of the interface for use as a haptic device. In order to do this, it was necessary to first identify and then implement an application on which to test the interface. Hoping to build on current investigations into multi-modal interfaces for robot control [7][8], it was decided to integrate the haptic sleeve as the sole form of operator feedback for a mobile robot system. Specifically, we investigated the effectiveness of using only this interface to guide a mobile robot through an obstacle course. This involved first constructing a mobile robot and controller, implementing a wireless communication link, and finally designing the experimental setup to evaluate performance.

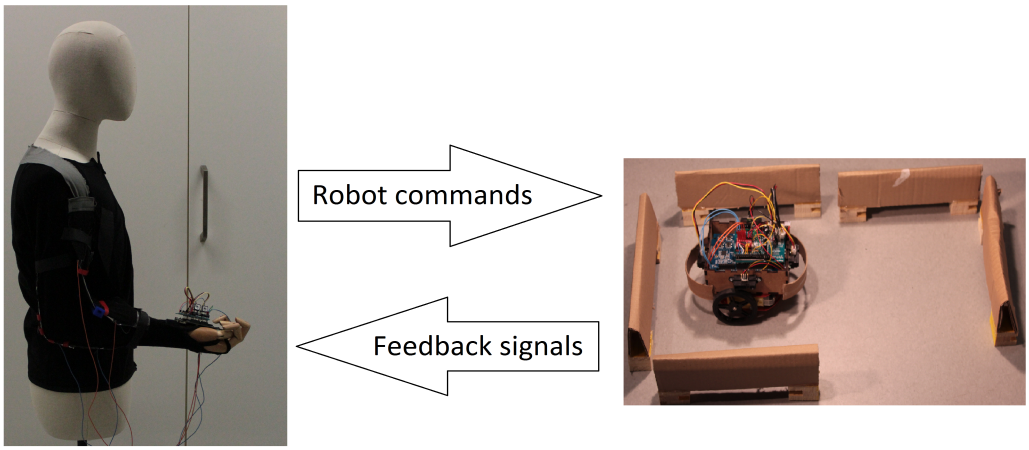

Figure 1: Overview of the evaluation system. A device operator is fitted with the haptic sleeve. Using a custom designed robot controller, the individual is tasked with guiding a mobile robot out of an obstacle course using only haptic feedback.

2.1 Haptic Interface

The crux of this work relied on a novel haptic sleeve strategy, currently being investigated by the Laboratory of Intelligent Systems (LIS) at EPFL [9]. This strategy is to create electroadhesive clutch composed almost entirely of fabric. By stimulating the clutch, motion along its tangential axis can be constrained. Since the clutch is flexible and made of fabric, it can easily be fitted to the body, and used as a tool to constrain motion along joints.

The clutch, which is essentially a flexible parallel plate capacitor, is implemented by taking advantage of the phenomenon of electrostatic adhesion. Each clutch is comprised of two flexible electrodes separated by a dielectric, two strips of inextensible fabric, and two strips of stretchable fabric. The flexible electrodes are made of aluminized plastic (PET), and are coated in a dielectric layer composed of BaTiO3 and TiO2. The clutch design is depicted in Figure 2.

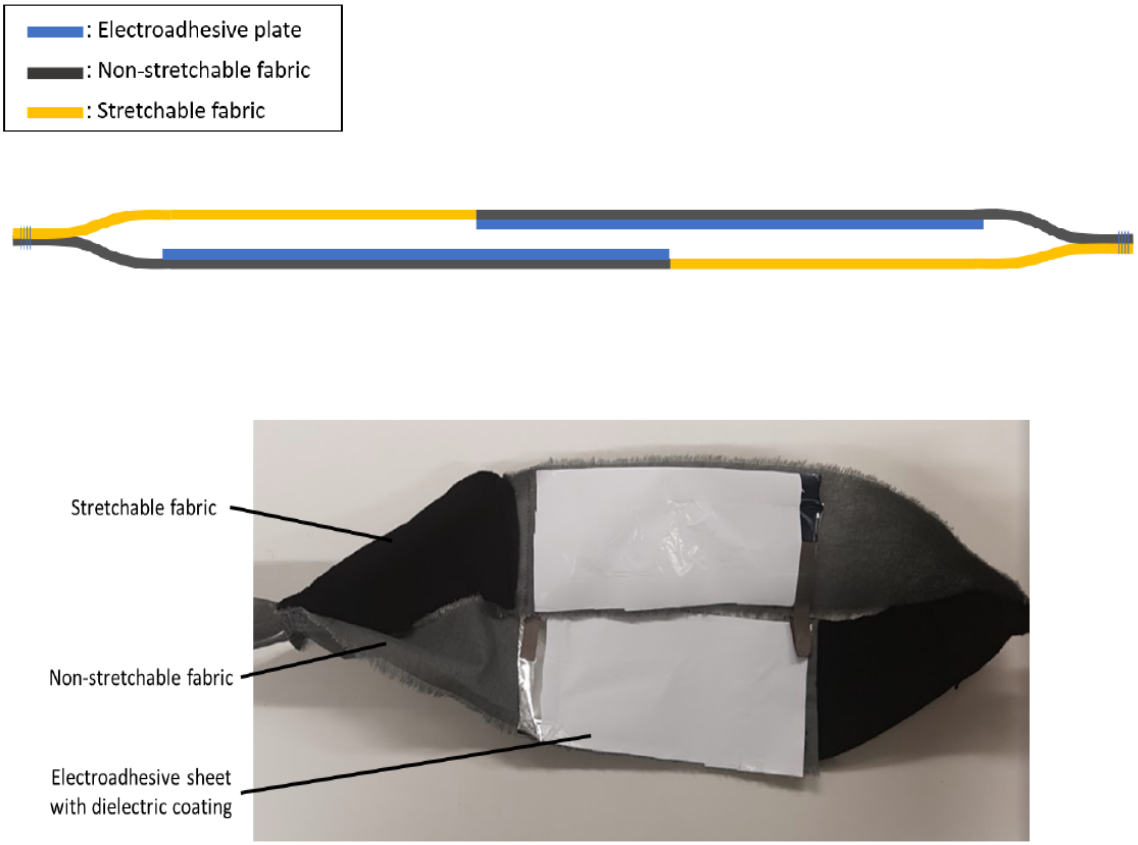

Figure 2: Top: A schematic of the electroadhesive clutch plates used for the haptic feedback. The materials used in the design are indicated. Bottom: A picture of one of the two clutch plates used to create the haptic sleeve interface.

The extensible fabric strips, which are each fixed to a strip of inelastic fabric, allow the device to stretch out when it is unstimulated. However, when a voltage, Φ, is applied across the terminals of the clutch plate, an attractive force is developed between the two electrodes [10].

$$F_e = \frac{\epsilon_r \epsilon_0 A \Phi^2}{2d^2}$$

where F_e is the electrostatic force, d is the distance separating the two electrodes of the clutch, A is the surface area of the clutch plates, Φ is the applied voltage, ϵ_r is the relative permitivity of the material, and ϵ_0 is the permitivity of vacuum.

This electrostatic force is normal to the direction of motion of the clutch plates. The friction force, f, resisting motion along the tangential axis of the clutch plates therefore increases as the electrostatic force, F_e, increases.

$$f = \mu F_e$$

where μ is the coefficient of friction between the two plates of the haptic interface.

Figure 3 illustrates the two modes of operation of the electroadhesive clutch plates. In the case that the two electrodes of the clutch plate are *not* stimulated, the two faces of the sleeve are able to easily slip past one another by applying a small force f_o. However, once a voltage is applied across the interface, an electrostatic force is generated causing the two electrodes to attract. In this scenario, the force tangential to the strap needed to cause slippage becomes significantly higher.

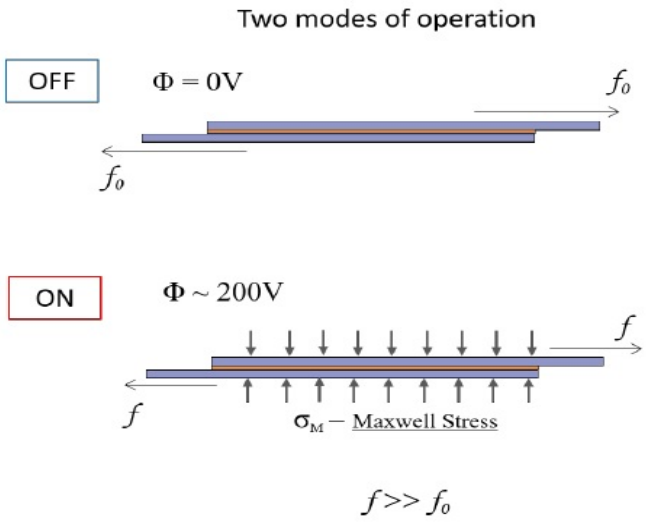

Figure 3: Different operating modes of the electroadhesive clutch plates. Top: No voltage is applied across the two electrodes, and the clutch is inactive. Bottom: Applying a voltage across the two terminals of the clutch activates the device, increasing its resistance to stretching.

As a part of our evaluation into the haptic interface itself, we investigated the times needed to charge and discharge the clutch plates. It was important to characterize these delays experimentally because analytically computing a consistent value of capacitance for the clutch plates was not feasible. The clutch plates were designed to be fitted to an operator's elbow joint, which is constantly in motion. The separation distance between the clutch plate is therefore neither uniform throughout the straps, nor uniform across time. Additionally, based on the operator's arm position at any given instant, the area of overlap of the two plates varied dramatically. Such variations effectively changed the capacitance associated with the clutch plate. Therefore, to verify that this interface could charge and discharge fast enough to be integrated into a dynamic system, we recorded the voltage across the electrodes during charging and discharging.

2.2 Wireless Communication

The Bluetooth Low Energy (BLE) protocol was used in order to realize wireless communication between the mobile robot and the sleeve module. BLE is an increasingly

popular wireless protocol that operates at 2.4 GHz. A BLE network consists of at least one master and one slave, though up to 7 slave devices can exist in any one network. Within this network, the master device is responsible for coordinating data transfers. Data can be written to a slave by the master, or the master can request data from its slave devices. This protocol is often used in relatively short range applications (less than 100 meters), making it an appropriate option for this proof-of-concept system. All information on the BLE protocol was extracted from *Getting Started with Bluetooth Low Energy* [11], and *Bluetooth Low Energy: The Developer's Handbook* [12]. For readers unfamiliar with the BLE protocol, a brief explanation of the relevant terms used in this document is provided in the appendix of this report.

The Nordic nRF52 system-on-chip (SoC) supports a wide range of wireless protocols, including BLE. Its ability to support BLE, its comprehensive SDK, extensive online documentation, and cost affordability motivated the decision to select a pair of nRF51 SoCs as the microprocessors for the system. The two devices were used to log sensor data, compute commands, and drive the outputs on both the robot and the sleeve module.

Figure 4: The Nordic nRF51 development kit used in the system implementation.

Using this wireless link, each device was responsible for communicating high level commands to the other. The robot module was responsible for sending signals to the sleeve module notifying it of any collisions. This was done using the two most significant bits (MSBs) of each data packet. Each bit was set or cleared to indicate that the robot was free, or was facing a barrier. If a bit was set to '1' it meant that the sleeve module should stimulate the appropriate clutch to prevent continued motion in that direction. If a bit was cleared to '0' it meant that the path was clear in that direction, and the operator should remain free to move.

At the same time, the sleeve module was responsible for sending motion commands to the mobile robot using this link. The 3 MSBs of the data packet sent from the sleeve module to the robot were used to encode the motion commands. For this proof of concept system, one of five simple commands were transmitted to the robot each connection interval. The sleeve could command the robot to remain still, to move forward at a constant speed, to move in reverse at a constant speed, to turn left by 90°, or to turn right by 90°. For future work, this could be easily modified to send more complex motion commands such as velocities and turning angles.

The structures of each packet are depicted in figures 5 and 6. Upon each connection interval, a data packet was sent from the slave to the master, followed immediately by a packet sent from the master to the slave.

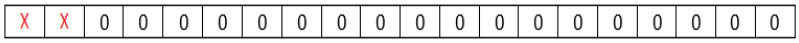

Figure 5: The structure of the 20 byte BLE packet sent to the haptic module from the mobile robot. Each bit represents the state of the front or rear of the robot. If a bit is set to '1' it signals to the sleeve module that an obstacle has been detected and the appropriate clutch plate should be activated.

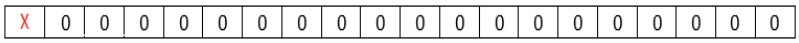

Figure 6: The structure of the 20 byte BLE packet sent to the robot module by the sleeve module. The 3 most significant bits (MSBs) are used to encode one of five possible commands each connection interval. The robot can be command to stop, move forward, move in reverse, or turn left/right.

The BLE network for this application was configured to send data packets of 20 bytes between the two devices. Using a connection interval of 200 ms, we were able to realize a throughput of 800bits/s in any one direction. For the purposes of this application, the maximum packet size of 20 bytes was not needed. However, the BLE connection was still configured to send 20 bytes per connection interval. This was done to verify that there were no problems sending data at the maximum packet size, meaning that constructing a larger and more complex system is easily achievable using the BLE protocol. Table 1 summarizes the BLE settings and characteristics used in this implementation.

Connection Interval	0.2 seconds
Connection Supervision Timeout	40 seconds
Slave Latency	0 connection events
Throughput	800 bits/s

Table 1: BLE Connection Parameters

2.3 Robot Design

The robot designed to test the haptic feedback was a simple wheeled mobile robot. The robot frame consisted of two small metal plates, supporting two DC motors in between them. One free-wheel at the head of the robot allowed it to remain balanced, and allowed for turning in place.

A 7.4V battery was used to power the prototype, allowing it to move anywhere within the 100m BLE range. As the nRF51 development board was rated to handle a maximum input voltage of 5V, a voltage regulator was used to reduce the supply voltage.

Two H-bridges were used to allow for clockwise and anti-clockwise motor rotation. This allowed the robot to move in reverse, as well as to turn in place. These H-bridges were obtained as part of an integrated motor driver. Additionally, the motor driver limited the current being drawn from the board output, protecting the system from any damages. The speed of the motors, and consequently the mobile robot, was set using a pulse-width modulated (PWM) digital output.

Due to their cost-effectiveness and common use in robotics systems seeking to implement object avoidance, infrared (IR) sensors were used to detect barriers to the front and rear of the mobile robot [13]. As a measure to reduce high frequency signal noise, each IR sensor output was filtered by an analog low-pass filter. This was done using simple first order RC filters with a cutoff frequencies of 2.25Hz.

Figures 7 and 8 illustrate the final constructed robot and the circuitry used to implement the design.

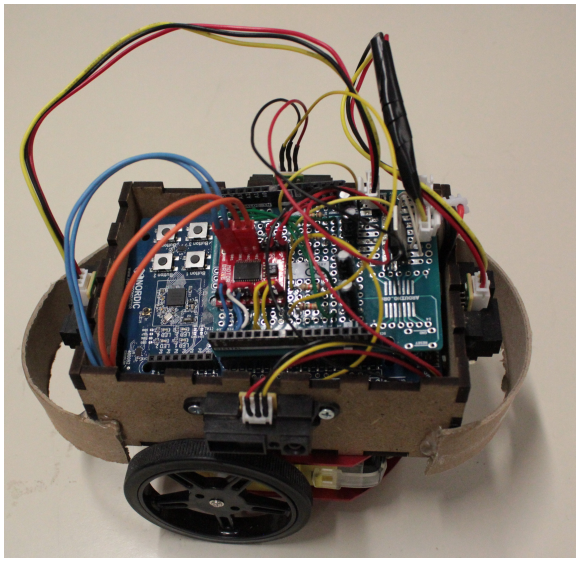

Figure 7: The final mobile robot prototype used for the haptic feedback evaluation.

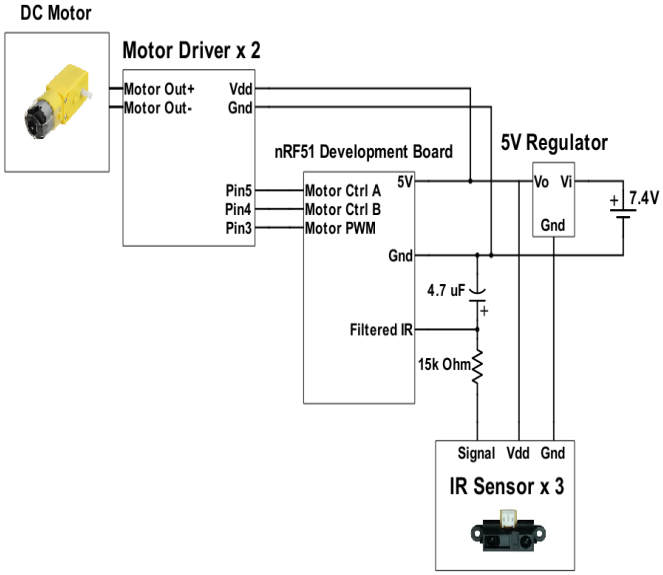

Figure 8: The circuit schematic for implementing the robot module.

2.4 Sleeve Module Design

In order to receive clutch trigger commands, stimulate the haptic interface, and control the mobile robot, we implemented an embedded sleeve module. As was the case for the robot module, the sleeve module was also implemented on the Nordic nRF51 development kit.

An inertial measurement unit (IMU) allowed us to measure the operators forearm angle, and generate robot commands. For communicating sensor data between the IMU and the microprocessor, we relied on the wired inter-integrated circuit (I2C) protocol [14].

In order to stimulate the clutch plates, a high voltage (in the range of 200 - 300 volts) was required. However, the nRF51 is only able to set its GPIO pins to a maximum voltage of 2.8V. In order to overcome this limitation, a set of DC-DC converters were used to step up the voltage by a factor of 100 [15]. By switching one of the low voltage GPIO pins to high, it was therefore possible to activate the haptic clutch with a voltage of 280V[1].

Since the haptic interface is essentially a parallel plate capacitor, we also had to construct a discharging circuit in order to bring the potential across the electrodes back down to 0V. This was done using a transistor to switch between the charging, and then the discharging circuit. To make sure that the full capacitor voltage of 280V did not appear across the GPIO pins during discharging, we placed a $27k\Omega$ resistor in series with the capacitor during the discharging. This was a protective measure to ensure that the microprocessor was not damaged during discharging. The final sleeve module and general circuit schematic are depicted in Figures 9 and 10.

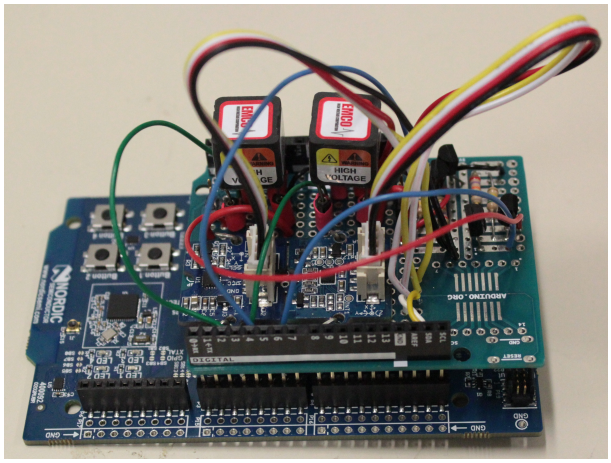

Figure 9: The final sleeve module used to simulate the haptic interface.

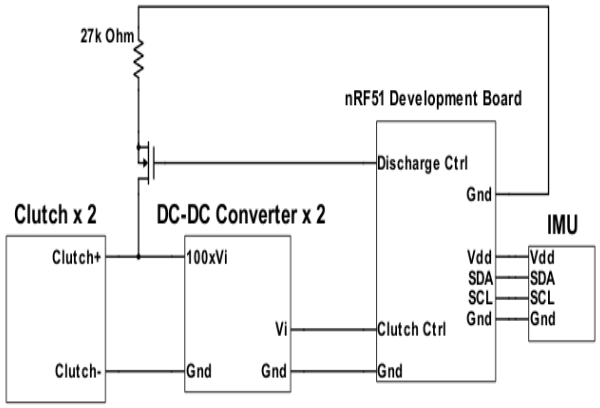

Figure 10: The circuit schematic for implementing the sleeve module.

2.5 Course Design

An obstacle course was designed in order to assess the effectiveness of the haptic sleeve. Four of the six possible obstacle course configurations are displayed in Figure 11. The concept behind this course design was simple. Five 20cm slits were constructed and arranged to form a rectangle. The mobile robot was initialized from the same position

[1]It should be noted that though the voltage applied was very high, the current drawn into the converter was in the order of microamperes. The power needed to operate the haptic clutches was therefore only in the order of milliwatts, which meant the system was not hazardous.

for all trials, and one slit was removed, creating an exit. The robot operator was then tasked with directing the robot out of this obstacle course using the sleeve. For each trial, the operator was given one of six possible exits. The individual had no way of knowing which configuration they were navigating. Throughout the trials, the operators were asked to turn away from the obstacle course in order eliminate all visual feedback. They were also provided with a set of headphones which were playing white noise. This was done to deprive participants of any auditory feedback, since operators could deduce whether they had collided with a barrier based on the sounds emitted by the mobile robot.

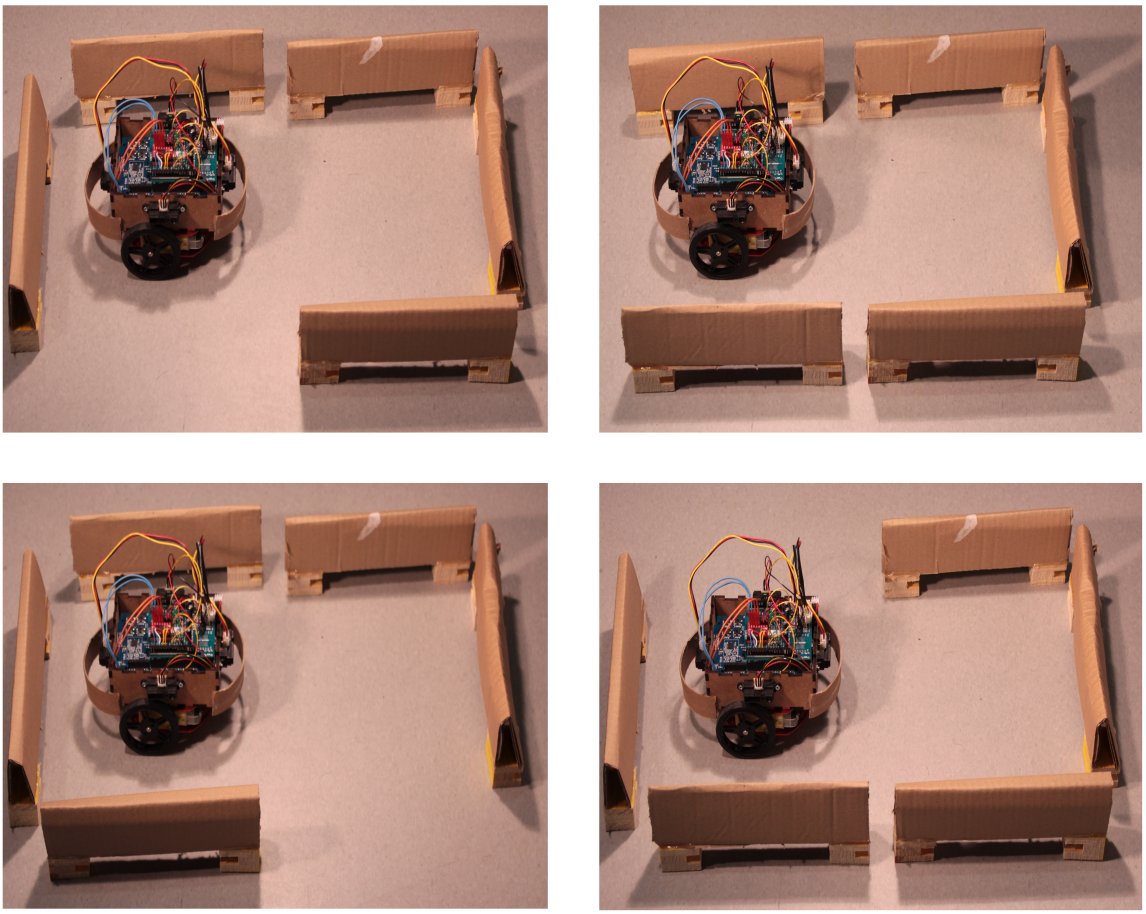

Figure 11: Different configurations of the obstacle course used to evaluate the haptic feedback strategy.

2.6 Control Strategies & Feedback Mapping

Three different control algorithms were implemented and tested. The three strategies attempted to complement the nature of the haptic sleeve. Since the sleeve is able to constrain the motion of the elbow when stimulated, the control methods were

designed to interpret elbow movement as motion commands for the mobile robot.

Elbow movements were approximated using an IMU, which allowed us to compute the angle of the forearm relative to the transverse plane. For estimating the state of the elbow, the elevation angle was used rather than the joint angle. Defined as the variations of the limb with respect to gravity vector, elevation angles are much less variable than joint angles due to the use of gravity as an absolute reference [16].

The feedback mapping used for the control strategies was based on whether the robot encountered an obstacle to its front or rear. If faced with a barrier to its front, the posterior strap was stimulated, countering any further movement in that direction. Similarly, an obstacle to the rear of the mobile robot activated the anterior strap. Figure 12 depicts the positioning of the anterior and posterior straps on the arm.

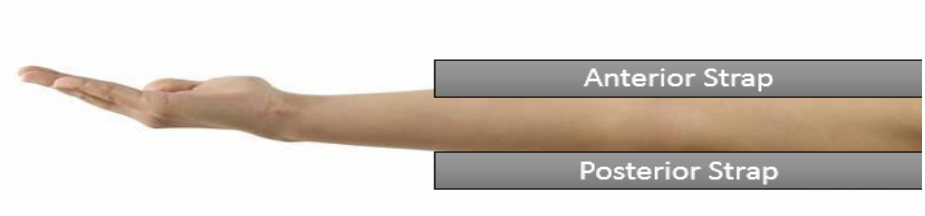

Figure 12: The anterior and posterior straps used to implement the haptic feedback. When faced with an obstacle to the front of the mobile robot, the posterior strap was programmed to clutch, constraining further movement in that direction. Similarly, the posterior strap was stimulated to prevent rear motion when faced with a barrier to the back.

All turning was carried out by pressing one of two push buttons. When pressed, these allowed the mobile robot to rotate in place by a fixed amount of 90° clockwise or counter clockwise.

Zone Control

The first and most naive method implemented is the Zone Control method. This method simply looks at the elevation angle, θ, of the arm at each time instant, and generates a motion command, $M(t)$, based on the following criteria.

$$M(t) = \begin{cases} reverse & -90^\circ \leq \theta(t) \leq -20^\circ \\ stop & -20^\circ \leq \theta(t) \leq 20^\circ \\ forward & 20^\circ \leq \theta(t) \leq 90^\circ \end{cases}$$

Nudge Control

The second control scheme is referred to as Nudge Control. This method takes into account the fact that the clutch plates do not produce any correcting force, but simply generate an increased *resistance to motion* when stimulated. In order to generate a

new command, the operator must move the arm from the rest position into the correct zone. Before a new command can be issued, the controller must first return to the rest position. Essentially, the operator “nudges" the robot forward by discrete amounts using this strategy.

$$M(t) = \begin{cases} reverse & -90^\circ \leq \theta(t) \leq -20^\circ \wedge M(t-1) = stop \\ stop & -20^\circ \leq \theta(t) \leq 20^\circ \\ forward & 20^\circ \leq \theta(t) \leq 90^\circ \wedge M(t-1) = stop \end{cases}$$

Pumping Control

The final control scheme tested during this project was dubbed Pumping Control. The concept behind this strategy is to keep the arm in a constant state of motion in order to encode forward or backward commands. This was implemented by computing the angular velocity as the discrete time derivative of the elevation angles. If the arm was pumped at an angular velocity with a magnitude greater than the threshold value $\omega_{threshold}$, the robot was commanded to move.

$$\omega(t) = \frac{\theta(t) - \theta(t-1)}{\Delta t}$$

The sampling rate, Δt was synchronized to the BLE connection interval time of 200ms. In order to filter out noise and smoothen the angular velocity signal, a non-linear median filter was applied using a sliding window of the 6 most recent $\omega(t)$ values. The result of this operation, $\tilde{\omega}(t)$, was used to determine the current command $M(t)$. The direction of motion was determined by computing the mean of the 6 most recent $\theta(t)$ values and checking if the resulting $\bar{\theta}(t)$ was greater or less than 0°.

Using this control paradigm, we were able to produce continues motion commands, while still allowing operators to take advantage of the clutch feedback.

$$M(t) = \begin{cases} reverse & |\tilde{\omega}(t)| \geq \omega_{threshold} \wedge \bar{\theta}(t) < 0 \\ stop & |\tilde{\omega}(t)| < \omega_{threshold} \\ forward & |\tilde{\omega}(t)| \geq \omega_{threshold} \wedge \bar{\theta}(t) > 0 \end{cases}$$

2.7 Subject Testing Protocol

Finally, volunteers were invited to test the system. This was a crucial step in our evaluation, as it allowed us to gauge how effective the haptic sleeve actually was in a real-life scenario. Volunteers were briefed on the objectives of the experiments, given time to learn how to use the robot controller, and taught how to use the haptic feedback. They were then given a set of headphones playing white noise to mask any sounds from the robot, and they were asked to turn away from the obstacle course to eliminate visual feedback.

The volunteers carried out a total of six attempts. The participants were given one try for each obstacle course configuration. After each attempt, the maze was randomly re-configured. The subjects were never made aware of the course configuration.

1. **3 minutes:** Volunteers were given time to familiarize themselves with the robot controller. This involved moving forward, backward, and rotating the mobile robot to the left or right.

2. **5 minutes:** Time was allocated to adjust the positions of the anterior and posterior straps. This was a process that was carried out for each new volunteer. It was critical to identify the strap configuration for each user that allowed them to experience the most resistance.

3. **5 - 10 minutes:** Volunteers were given time to experience feeling the feedback while being able to see the robot. This allowed users to learn what the feedback felt like, and therefore be able to interpret when the interface was clutching during the blind tests.

4. **20 - 25 minutes:** Finally, the volunteers carried out the experiment. They were deprived of visual and aural feedback, and attempted to navigate the mobile robot out of the obstacle course.

Each test took between approximately 33 and 43 minutes to complete. The experiment was performed with three different volunteers. Figure 13 shows a trial of the experiment being carried out by a volunteer subject.

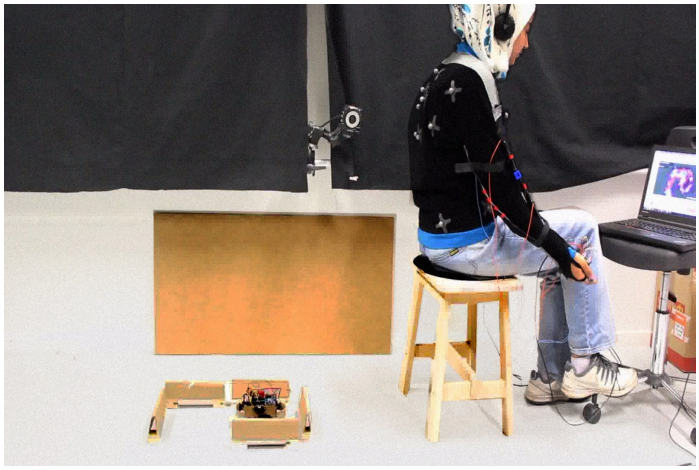

Figure 13: A volunteer attempting to guide the mobile robot out of the obstacle course using only the haptic feedback. The participant was unable to see the configuration of the maze, and was wearing a headset playing white noise to eliminate any auditory cues from the robot.

3 Results

In this section, we present the results of our experiments. We begin with our subjective evaluation of the different control algorithms implemented during the project. We then present quantitative analysis of our implementation. Finally, we present the results of our subject tests.

3.1 Control Algorithm Selection

Of the three methods implemented to test with the haptic feedback system, the control algorithm best suited to our application was the *nudge control* method.

- **Zone Control:** The haptic feedback could not be felt at all by the operator when activated.
- **Nudge Control:** The haptic feedback could be felt by the device operator. It was possible to correctly identify whether or not objects were placed in the robot's path using this strategy. The device operator had sufficient time between commands to first decide whether feedback was being applied, and then issue a new command.
- **Zone Control:** The haptic feedback could felt by the device operator. It was possible to correctly identify whether or not objects were placed in the robot's path using this strategy. However, with this method the device operator was unable to react quickly after experiencing the feedback. For this reason, it was not selected as the control method to test during the subject trials.

3.2 Clutch Plate Delays

The clutch plate delays were measured using an oscilloscope. The voltage across the two electrodes of the clutch plate was recorded over a period of 10 seconds. At the start of this time, the sleeve module was made to stimulate the clutch plate. We measured the time needed for the clutch voltage to increase from 0V to 280V to be 180ms.

$$t_{charge} = 180ms$$

We also recorded the time needed for the sleeve to discharge after being fully charged to be 4.2s.

$$t_{discharge} = 4.2s$$

The results of this test are displayed in Figure 14. The charging period is marked in red. The discharging period is indicated in green.

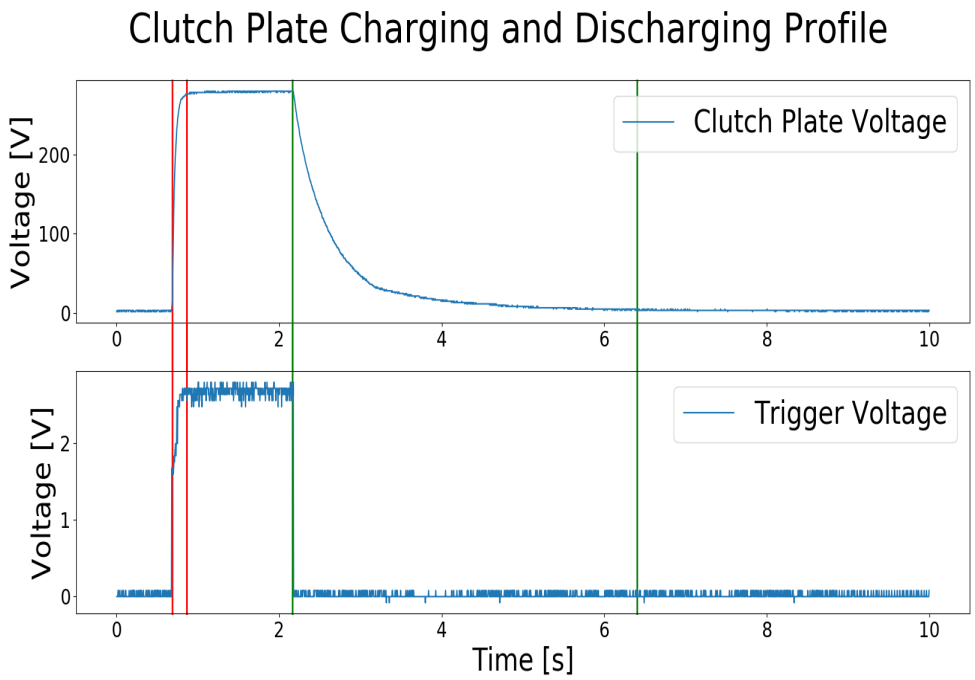

Figure 14: Top: The charging and discharging times of the flexible clutch plate. The charging time is represented in red. Discharging time is represented in green. Bottom: The state of the microprocessor's GPIO pin. When the GPIO was set to high, the clutch plate began to charge. Clearing the GPIO pin allowed the plate to discharge.

3.3 Stimulation Patterns

We recorded the stimulation commands generated by the sleeve module as a function of the IR sensor readings. The IR threshold was set so that objects with a proximity less than 15cm caused a clutch command to be generated. The results of these experiments showed that there was a minimal delay between identifying a barrier and stimulating the clutch plates. This can be seen in Figures 15 and 16. Almost immediately after identifying an object nearer than 15cm, the anterior and posterior straps are commanded to clutch. The delay associated with this is 200ms, which corresponds to the length of the BLE connection interval.

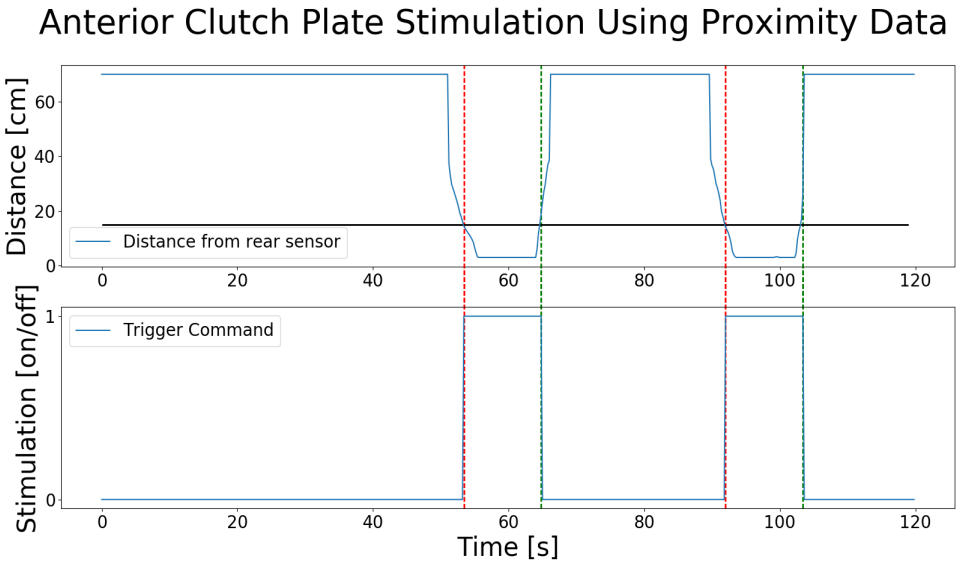

Figure 15: Top: The IR data recorded over two minutes for the rear side of the mobile robot. The threshold for proximity is 15cm. Bottom: The stimulation patterns generated on the sleeve module. It was observed that when an object was placed less than 15cm from the rear of the robot, a command was immediately generated to clutch the anterior plate. The delay between crossing the threshold and generating the command was the duration of a single connection interval (200ms).

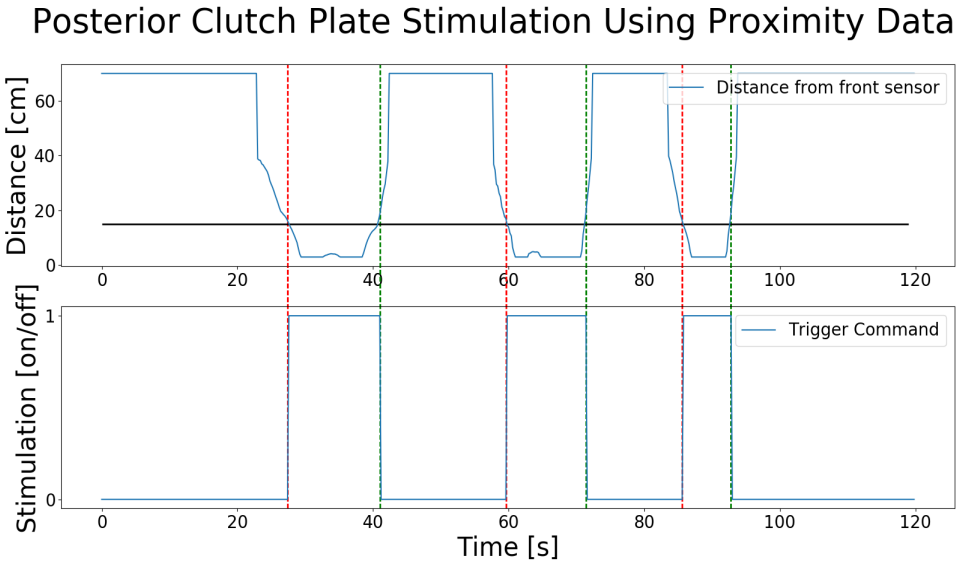

Figure 16: Top: The IR data recorded over two minutes for the front side of the mobile robot. The threshold for proximity is 15cm. Bottom: The stimulation patterns generated on the sleeve module. It was observed that when an object was placed less than 15cm from the front of the robot, a command was immediately generated to clutch the posterior plate. The delay between crossing the threshold and generating the command was the duration of a single connection interval (200ms).

3.4 Subject Trials

The outcomes of the subject trials are reported in Table 2. Out of 18 total trials, the subjects were able to successfully navigate the robot out of the maze 13 times. 5 trials ended in the robot becoming trapped in a corner of the obstacle course. Trials where the robot had become visibly trapped after 1 minute were considered failures.

	Volunteer 1	**Volunteer 2**	**Volunteer 3**
Front Exit	Success	Success	Success
Top Right Corner Exit	Success	Success	Failure
Top Left Corner Exit	Success	Failure	Success
Bottom Right Corner Exit	Failure	Success	Success
Bottom Left Corner Exit	Failure	Success	Failure
Rear Exit	Success	Success	Success

Table 2: Subject Trials Results

4 Discussion

4.1 Control Strategies

Of the three control strategies investigated during this research, two methods (nudge control and pumping control) were compatible with the haptic interface.

Zone control was not an appropriate control strategy, since it did not necessitate enough motion along the axis of the sleeve. Since the operator commands a forward or backward motion by simply holding their arm steady in the desired zone, it is difficult to tell when the clutch is being activated.

On the other hand, nudge control worked well with this system. We opted to use this method for the subject tests because it offered operators time to process what was being felt after each new motion. The limitation of nudge control is that it only allows the robot to move by discrete amounts. Each new command drives the robot forward or backward by a fixed amount. Another limitation to this control strategy is that there is no intuitive mapping for obstacles that approach the *sides* of the robot. However, for our application this was an effective method for robot control. All subjects were able to feel the clutching, and were able to guide the robot out of the obstacle course in 13 of the 18 trials.

Finally, pumping control was also an appropriate method for experiencing feedback. This strategy was not ultimately adopted for the subject tests because users require time to consciously interpret changes in the clutch state. What was observed during preliminary testing with this strategy was that obstacles were regularly detected, but not immediately. This method is interesting as it implements continuous control, as opposed to the discrete movements allowed when using nudge control. By

increasing the rate of arm motions, it is even possible to implement speed control. This strategy may be worth investigating in order to implement more complex tasks with the haptic sleeve.

4.2 System Performance

The clutch delay test depicted in Figure 14 verified that the haptic interface could be used in a real time control system. The reason for the difference between the charging time and discharging time has to do with the circuitry involved in each process. The extremely short time of 180ms needed to charge the clutch is due to the fact that there are no resistances placed in series with this element. The only resistances are parasitic resistances of the wires and microprocessor, resulting in a very small time constant.

$$\tau = \frac{1}{RC_{clutch}}$$

On the other hand, a $27k\Omega$ resistor was connected in series with the plates during discharging. We expected this to result in a significantly larger time constant τ, and this was indeed what our results showed. This series resistance is something that can be tuned to achieve whatever discharge time is desired. Since we selected the relatively slow-paced nudge control method as our control strategy, we preferred a slower discharge time. This was simply a precaution to ensure that the microprocessor was not damaged by applying too large a voltage during discharging.

The results in Figures 15 and 16 allowed us to verify that the sleeve module logic was correctly implemented, and that signals were being communicated from the robot to the sleeve at a fast enough rate. We observed that within one connection interval of an object being introduced in the robot path, the appropriate clutch could be stimulated. Both the anterior and posterior straps could be activated 200ms after an obstacle was detected.

4.3 Haptic Interface Challenges

One of the main issues that was faced during the experiments was how to correctly place the haptic interface on the subjects' arms. The main comment raised by volunteers was that during some trials, they were unable to feel the feedback being applied. This resulted in difficulties navigating out of the obstacle course.

One potential causes of this issue was identified in previous literature on soft exosuits. Interface misalignment is thought to be a major inhibitor of soft wearable device efficacy [17]. A standardized strategy for fitting users with the haptic sleeve would help ensure that the clutch plate was well aligned with the axis of motion. Such a technique could also help reduce performance variations between subjects.

Another factor affecting the perceived clutch strength may stem from the fact that the joints in the human body are complex structures. Unlike rigid joints that simply rotate along one axis, the human elbow is actually composed of three different

components. The body's joints have more degrees of freedom than this simple clutch alone can constrain. Even if the elbow itself is completely constrained, the arm can still be moved using other joints.

Additionally, all three participants in this study were of different heights, and had different arm lengths. This could have had an effect on how clearly the volunteers were able to distinguish the feedback. The volunteer whose arm length most closely match the length of the sleeve had a better score, and performed more confidently than those with longer arm dimensions. It must be emphasized that this is only a speculation, since the sample size used in this experiment was extremely small. We cannot generalize our observations to reach any conclusions without conducting further trials.

In order to address these limitations and optimize the performance of the haptic interface, additional work is needed to identify the best strategies for anchoring and fitting the sleeve to the device operators.

4.4 Future Work & Improvements

One of the main issues faced during the evaluation of this system was the placement of the haptic sleeve. This was mainly due to the fact that the soft tissues enveloping the human body are far from rigid. It was observed during the experiments that strap placement was a key factor in making sure that volunteers could feel the haptic feedback. When properly fitted to the arm, device operators were more likely to respond that they were able to identify when an obstacle had been placed in the robot's path. In addition, there was a high variability between subjects, increasing the time needed to set up any given experiment. This was due to variations in arm lengths between individuals. In order to maximize the effectiveness of this sleeve, it would be valuable to have a parameterized method for fitting device operators with the sleeve such that the perceived resistance forces are maximized. This would improve the effectiveness of the haptic interface, as well as reduce the time needed to don the sleeve.

It would be interesting to investigate this technology's ability to provide feedback for more complex tasks. It may be possible to utilize this concept to help control the speed of a robot, instead of simply using it to communicate whether a path is free or blocked. Using the pumping control method developed in this project, it may be possible to produce stimulation patterns that help the operator regulate their arm movement to achieve a desired speed.

5 Conclusions

The aim of this project was to construct a proof-of-concept system that could make use of a novel haptic feedback strategy. A soft, flexible, haptic interface based on the principle of electroadhesion was used to construct a wearable set of clutch plates. The specific goal of this work was to find a way to guide a mobile robot through an

obstacle course, using *only* this haptic feedback. Three different control paradigms were proposed to try and control the robot; these strategies were known as *zone control*, *nudge control*, and *pumping control*. We investigated the three proposed methods, and then selected the one that was deemed most effective in allowing a device operator to interpret haptic feedback generated by the clutch plates.

Though more work needs to be done in order to optimize the interface, the outcome of this project was encouraging. All three volunteers were able to properly interpret the feedback and guide the mobile robot out of the enclosure. The total number of successful navigations was recorded to be 13 out of 18 total trials. We conclude that this interface *can* be implemented as a source of feedback for real time systems. In particular, it would be interesting to study how this feedback method performs when coupled with other forms of feedback. There is potential to integrate such a device into multi-modal strategies for human-robot interaction.

6 References

References

[1] D. A. Kugath, "Experiments Evaluating Compliance and Force Feedback Effect on Manipulator Performance", NASA Center for AeroSpace Information (CASI), 1972.

[2] D. Chang, "Haptics: gaming's new sensation," in Computer, vol. 35, no. 8, pp. 84-86, Aug 2002. doi: 10.1109/MC.2002.1023794

[3] J. K. Koehn, K. J. Kuchenbecker, "Surgeons and non-surgeons prefer haptic feedback of instrument vibrations during robotic surgery," in Surgical Endoscopy, vol. 29, no. 10, pp 2970–2983, October 2015. doi: https://doi.org/10.1007/s00464-014-4030-8

[4] J. Armitage K. Ng "Feeling sound: exploring a haptic-audio relationship," in Music, Mind, and Embodiment. CMMR 2015. Lecture Notes in Computer Science, vol, 9617. Springer, Cham. doi: https://doi.org/10.1007/978-3-319-46282-0_9

[5] S. Stramigioli, R. Mahony, and P. Corke, "A novel approach to haptic tele-operation of aerial robot vehicles," in IEEE International Conference on Robotics and Automation, Anchorage, AK, July 2010, pp. 5302-5308. doi: 10.1109/ROBOT.2010.5509591

[6] P. Banzet, "Design of a soft, wearable sleeve for haptic feedback," EPFL semester project, January 2018.

[7] B. Gromov, L. M. Gambardella, and G. A. D. Caro, "Wearable multi-modal interface for human multi-robot interaction," IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2016.

[8] T. Kollar, A. Vedantham, C. Sobel, C. Chang, V. Perera, and M. Veloso, "A multi-modal approach for natural human-robot interaction," Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[9] V. Ramachandran, J. Shintake, D. Floreano, "All Fabric Electroadhesive Clutch," (unpublished).

[10] J. D. Jackson, "Classical electrodynamics." New York: Wiley, 1999.

[11] Townsend, K., Cufi, C., Davidson, A., & Davidson, R. (2014). Getting started with bluetooth low energy. Sebastopol, CA: OReilly & Associates.

[12] Heydon, R. (2014). Bluetooth low energy: the developers handbook. Upper Saddle River, NJ: Prentice Hall.

[13] J. F. Roberts, T. S. Stirling, J. C. Zufferey, and D. Floreano, "Quadrotor Using Minimal Sensing For Autonomous Indoor Flight," 3^{rd} US-European Competition and Workshop on Micro Air Vehicle Systems & European Micro Air Vehicle Conference and Flight Competition, September 2007.

[14] NXP Semiconductors Technical Staff, "I2C-bus specification and user manual," NXP Semiconductors, April 2014

[15] XP EMCO, "Isolated, Proportional DC to HV DC Converters," Q01 datasheet.

[16] A. Kamiar, A. Ijspeert, G. Courtine. BIOENG-404. Class Lecture, Topic: "Neuroscience of Locomotion I." Section of Life Science and Technology, EPFL, 2017.

[17] A. T. Asbeck, R. J. Dyer, A. F. Larusson, and C. J.Walsh, "Biologically-inspired soft exosuit," in Rehabilitation robotics (ICORR), 2013 IEEE international conference on, pp. 1–8, IEEE, 2013.

[18] Nordic Semiconductors, "Nordic nRF5 Software Development Kit," download.recurser.com. [Online]. Available: https://developer.nordicsemi.com/nRF5_SDK/nRF5_SDK_v11.x.x/. [Accessed: 04-Jun-2018].

7 Appendix: Bluetooth Low Energy

Packet Types

There are two different types of BLE packets that can be used to transfer data from a server to a client; **advertising** packets and **data** packets. These are different ways to communicate data between devices, each with benefits and disadvantages. By understanding some of the differences between the two alternatives it becomes clear why data packets were the best option for our application.

Advertising Packets

Advertising packets are useful for applications that don't need rapid data transfer, and where losing a few data packets is not a critical issue. For devices to share data via advertising, *no* connection is established. Advertising makes use of at most three different frequency channels to broadcast data between devices. There must be one advertiser device, and one or more scanner devices. The scanner (the device listening for data) will only receive data when it *happens* to scan the correct channel while a packet is being advertised on that channel. Since the advertiser and scanner are not coordinated, there is no way to guarantee that a given packet will be successfully transmitted.

The advantage of using advertising is that the overhead associated with setting up a connection between devices is eliminated. An advertiser simply broadcasts data at a fixed time interval (between 20 ms and 10.24 s) to any device that is listening.

This method is not suited to the haptic feedback application for several reasons. Due to the fact that the advertiser and scanner do not coordinate their frequencies, packet loss is inevitable. This introduces a risk that an incorrect stimulation command may be sent from the robot module to the sleeve. This is problematic since it is important that the robot be able to timely and reliably communicate commands to the sleeve module. Additionally, it is impossible to secure data via encryption using this type of packet since no connection is established. For these reasons, we opted to make use of data packets.

Data Packets

Data packets are sent using an exclusive connection between BLE devices. Instead of advertiser/scanner roles, data transfer occurs between a master device and one or more slaves. The master is responsible for initiating a connection with a slave device, and for setting the connection parameters. A BLE connection simply refers to a series of pre-scheduled packet transfers between master and slave devices. Unlike with advertising packets, the sequence of frequency channels is known by *both* the master and slave, eliminating the chance of missing data. On each connection event,

the devices “hop" to a new channel in a pattern that is known by both devices[2].

This type of packet is more appropriate for our application than the advertising packets. Using data packets it is possible to send a payload as fast as once every 7.5 ms, which is much faster than what advertising packets can achieve. Packet loss is *not* a concern using this method since a 24-bit cyclic redundancy check (CRC) is used to identify transmission failures. If a transmission failure is observed, a retransmission request is sent until the data is finally received. Finally, using data packets gives the option of encrypting the link between devices, in contrast to when advertising packets are used.

Connection Parameters

Connection Interval

A connection interval is defined as the time between two consecutive BLE data transfer events. The connection interval is set by the master device (the client in this application), and can range anywhere from 7.5 ms to 4 seconds with increments of 1.25 ms.

Connection Supervision Timeout

This parameter specifies how much time may elapse following the most recent data exchanges before a link is considered lost. This parameter is used so that the two devices do not remain behaving as if they were still connected if the link was in reality lost.

Slave Latency

The slave latency dictates the number of connection events that the server can choose to ignore without disconnecting. This can have benefits when power needs to be saved as it allows a server to stay sleeping if data is not yet available, but then still transmit whenever needed. This parameter was not manipulated in this study since data was transmitted on *every* connection interval, and read requests were not used. We selected this parameter to be 0.

[2]There are 40 channels, which are frequencies between 2.4 GHz and 2.4835 GHz. Both devices know which channel to “hop" to based on a parameter defined by the master device upon connection.