Grasp Control with Tactile Feedback

Diogo Costa Arreda
Instituto Superior Tecnico

***Abstract*—This work focuses on locally control a robotic hand using tactile feedback in order to achieve a stable grasp. The hand in question belongs to the humanoid robot Vizzy. In the current options for robotic hand interaction with the environment, typically, a centralized architecture is used, in which the primary system directly actuates the hand. Such architectures presents deficits in the speed of communications and, consequently, in the action-reaction grasp control. A local solution is then proposed. This solution consists of a hardware module in Vizzy's hand dorsal side. The primary system sends grasping commands to the local module, and this one is responsible for controlling the hand. The local module incorporates a layered architecture: high and low level. The high-level layer is responsible for validating and interpreting commands from the primary system and generating trajectories for the low-level to follow. The low-level layer makes the system follow the high-level trajectories and, for that, it applies controllers to the system variables. Independent controllers for the different parts of a complete grasp, contact and non-contact, were developed to build a more efficient low-level controller. Typically, only one variable is used when performing grasping control however, a hybrid controller, gathering force and position, is proposed to actuate the hand during contact. Speed control is the alternative for non-contact control. The complete architecture is implemented in an electromechanical system simulator, developed in *Simscape*, with the purpose to validate and tune all the development done.**

***Index Terms*—Grasp Control, High-Level Control, Layered Architecture, Low-Level Control, Standalone module, Tactile Sensing.**

I. Introduction

Humans are constantly using their sense of touch to recognize, feel and even perceive the surrounding environment [1]. This interaction is only possible due to humans advanced tactile sensing and, with the raising value that the robots are gaining nowadays makes it essential to try to replicate it, the best way possible. This skill allows the robot to perform tasks in unstructured environments, enables the robot to cope with significant uncertainties and, most importantly, it promotes the safety between robots, objects and interaction with humans [2]. Therefore, it is necessary to understand tactile sensing, and the process of designing, building and controlling of robotics hands and end-effectors, keeping in mind that the main purpose of such devices includes grasping and precise manipulation of objects [3]. The major difficulty of grasping relies on measuring with precision the state of the object and robotic hand in terms of physical variables that influence the grasp, such as position, force or friction [4]. The main motivation of this work is to locally control a humanoid robotic hand with tactile feedback in order to perform a stable grasp.

A. Topic Overview

Grasping is the act of taking, holding or seizing firmly with the hand. An example of a grasp is the handshake, where two people grasp each other's hand. It seems an effortless action however, in robotics, even a simpler action, such as autonomously grasping an object, remains a challenging problem [6].

Robotic hands belong to a vast field, stretching from firefighter cutting-tools to micrograppers. However, in this work, the focus is on dexterous robotic hands. Dexterity, by definition, means the capability of changing the position and orientation of the manipulated object from a given reference configuration to a different one [8]. In the area of dexterous robotic hands, there are also several options. Still, the focus of this project is on Vizzy's robotic hand, shown in Figure 1.

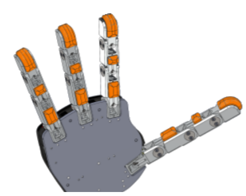

Fig. 1. Vizzy Robotic Hand [9]

There are different technologies for actuating a dexterous hand: direct-drive magnetic actuators, piezoelectric motors, and pneumatic actuators [8]. Vizzy's actuators are DC micromotors placed on the dorsal side.

The goal is to have a local solution to control Vizzy's hand and, this implies that the space to attach hardware is quite limited, not only in terms of length and width but especially in terms of height. It is also an unstable area, which means that any electronics placed here need to be extremely robust. The reason why this solution is planned to be local and individual is to achieve a sampling and response frequency nearest as possible to human grasping. The systems developed until now, typically, have a centralized architecture is used, in which the primary system directly actuates the hand. This can restrict the system in terms of speed of communications and, consequently, the action-reaction grasp control.

1) Human Processing Power: The human tactile characteristics in terms of sampling and acting frequency are divided into four different receptors: Merkel receptor, Meissner corpuscle, Ruffini cylinder and Pacinian corpuscle. The highest stimulus frequency detected by human's skin mechanoreceptors is 500 Hz [7], and according to the Nyquist theorem, the sampling rate must be at least two times the highest frequency component of the signal, in order to correctly sample the signal [5], meaning that the sensor's sampling frequency in this project should be the nearest as possible of 1000 Hz. All the electronics, communications and controllers should have this goal while being developed.

2) Vizzy: Vizzy's grasping abilities [9] are the main scope of this work. His hands are composed of four fingers: a thumb, index and two middle fingers. The thumb and the index have individual wires connected to the respective pulley which is attached to a motor shaft; The two middle fingers share the wire, and the same motor controls both. These wires are called the finger tendons and, to control the fingers, it is necessary to actuate the motors, once the motors pull the wires, the hand will slowly close. The readjustment to the reference position - open hand - is done by a group of dental rubber bands placed on the dorsal side of each finger. Each finger has three phalanxes, and each one has a magnetic tactile sensor, as shown in Figure 1. Although simple, Vizzy's hand skills include the basic manipulation actions and gesture execution, additionally can perform a precision grip (tip-to-tip) and three types of power grasp: cylindrical, spherical and hook.

B. Objectives

The main goal of this project is to create an independent, standalone hardware module capable of receiving different grasping commands and perform them in real-time control while acquiring data from the robotic hand. This implies the compilation of different small goals, such as design an efficient system architecture, model all the system components, design the most stable, robust and fast controllers possible for the contact and non-contact stages of the grasp, generate alarms for unexpected behaviours of the system, and build a local module that can perform all of these tasks as quick as possible in order to achieve the most similar grasp to human's as conceivable (which as mentioned before, implies a sampling frequency of 1 kHz).

II. State of Art

Robotics plays a vital role in everyday life, from home and health care to military use as emergency response, robots are fast becoming a fixture in society. For that, a lot of research has been done in robotics, and in particular in robot control. Robot control is a quite complex area, especially due to the need to achieve high-level complex goals, the need to interact with a dynamic environment, while ensuring the systems dynamics. The need to handle noise and uncertainty from the sensors and, the need to be reactive to unexpected changes, such as collision or dropped objects, which are quite common [10].

A. Grasping

Manipulation is an advanced capability that sets robots apart from most other computerized or automated systems. The three most essential functions of the human hand are to explore, to restrain, and to move objects precisely [3]. However, grasp control is a very complex concept because it relates to several topics, such as grasp stability, grip strength, object's slip control, and object exploration.

Grippers and fixtures have been used extensively in the industry; however, the robot grasping field only started in 1979 with the work of Asada and Hanafusa [11] and Salisbury's first three-fingered robotic hand [12]. Since then, many hand designs have been proposed, ranging from rather simple devices to highly sophisticated multi-fingered hands, such as the TUAT/Karlsruhe humanoid hand [13].

1) Control: There are three main approaches to general control [14]. Hierarchical, behavioural and hybrid. The hierarchical control is the classic sense-plan-act architecture. It starts with high-level goals that are decomposed into different sub-tasks, also known as the "Top-Down" approach. Another control approach is the behavioural, known as the "Bottom-up" approach. This control deals with independent modules that are concurrently monitoring sensor values and triggering actions. This option is hard to organize into complex systems because it gets promptly chaotic. Finally, the hybrid control approach, which deliberates the goals at a high level, but it is reactive at a low level, implying that the sensors directly determine actions. All of these control approach state that in layered control systems, the high-level control runs infrequently while the low-level controller runs continuously [15].

2) Grasping Structure: One way of dealing with the grasping complexity is through modularity within a given structure. This way, overall system complexity can be reduced by decomposing the structure into smaller components with well-defined abstraction levels and interfaces between them [10]. Therefore, in grasp control is common to have, at least, a two layers architecture, an high and a low layer. Kakoty [16] presented a two-layer architecture for a five-fingered prosthesis, where the superior hand control is actuated through electromyogram signals for recognition of the user's intended grasp. Then it interacts with local hand control for controlling the fingers' joint angles and speed. Upon Sebastian Engell [17] work, an hybrid control architecture for dexterous manipulation is presented, where a hybrid reference generator sends the parameter for a grasping force optimization that will interact with a robotic hand in order to perform a force control with the respective fingers.

3) High Level Controller (HLC): A high-level control program in a system is the set of logical formulae describing actions and their effects, those of both the robot and other agents [18]. The HLC initiate low-level routines, such as references following for force, speed or position, that are further executed. Upon termination, these routines communicate the status of the robot's sensors back to the HLC, which then decides how to proceed.

4) Low Level Controller (LLC): Real-time performance is required for a LLC. Therefore the code to issue actuators commands or process sensor readings must run very frequently. At this level, it is very usual for robotic hands to measure force distribution using tactile sensors [19], as stated before, the expected contact between the object and the hand are crucial details to perform quality control upon grasp.

There are several approaches for the best grasping low-level control strategy. A position control is suggested by Kasim [20] in order to perform different grasps in a three-fingered hand, where DC micromotors actuate the fingers. Force control is likewise very frequent in grasping [21], [22]. A hybrid force-position trajectory control is also proven to be a stable and robust option for grasping control [23].

5) Grasp Stability: Independently of which controller is used for grasping, the goal is common: achieve a stable grasp. In order to define what grasping robustness and stability are, the notions of form-closure, and force-closure are fundamental. These properties were first introduced by Reuleaux [25] and it concerns the capability of the grasp to completely or partially constrain the motions of the manipulated object. In this line of work, Salisbury [26] proposed a basic framework for testing the stability of a grasp. It states that a grasp is stable if the stiffness matrix (which characterizes the grasp) is positive definite. Furthermore, J. Montana [27] shows that by modelling each finger-object contact as a virtual spring, force closed grasps can be made stable just by adjusting the applied forces at each finger.

B. Our Contribution

In the current grasping systems, all the control is done in the central system, and then low-level commands are sent to the actuators. The most significant issue in this architecture is the acquisition-actuation process time. In a local solution, with efficient electronics, this process will have a faster response. In the proposed solution, the central system sends the type of grasp that is desired and the expected force. The local unit will have a mixture of hierarchical and hybrid architecture. This system has high-level goals that are generated from the information received. If this information is valid, the HLC gathers it and convert into a trajectory described by references for the LLC to follow, in order to obtain the desired grasp. The LLC will be reactive, meaning that actions will be determined according to the acquired data.

Opposite to the solutions proposed by [20]–[23], in which the same low-level controller covers all the grasping process this project will implement a dual phase low-level architecture, where independent controllers will actuate the system for the non-contact and contact part of the grasp. In the non-contact part, a speed controller is implemented. For the contact part, several options for the control are proposed: deformation, force and hybrid. These controllers will be compared in terms of stability, performance and robustness, to determine which one is the best for the grasping process.

Vizzy's grasping mechanical system is modulated in a *Simscape* simulator in order to interact with the proposed controllers, so that every controller designed can be tested and tuned.

The local hardware solution, designed in *Altium*, is a compact four-layer PCB, with Ethernet communication to the primary system, I2C communications to the tactile sensors, dual-drivers for the DC motor control and 12-bit analogue inputs for current consumption measurement. This solution is placed in Vizzy's hand dorsal side and is similar to the one in presented in Figure 2.

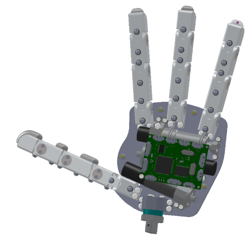

Fig. 2. Local Hand Control Goal

III. METHODOLOGY

Control theory in control systems engineering is the subfield of mathematics that deals with the control of operating systems in engineered processes. This project deals with two levels of control. First, a conceptual model of control, where does not exists a specific parameter to be controlled, either a group of factors which will be regulated aiming to obtain a specific final state - HLC. Secondly, the LLC, that actuates the real system with the purpose to follow the references given by the HLC.

The system starts when received a grasping command from the central system, Vizzy. This command includes the pretended type of grasp and the desired force. Upon HLC this information is validated and translated in individual trajectories for the fingers. Each finger will have a certain speed and desired force that should follow in order to achieve a stable grasp. Speed and force parameters are then forwarded to the LLC, which will actuate the fingers accordingly.

In the LLC, two independent blocks of control exists, one for the non-contact and other for the contact control.

The architecture of the system can be seen in Figure 3 moreover every component is explained with more detail in the following sections.

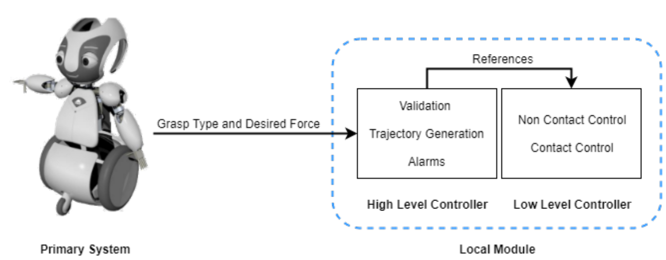

Fig. 3. Local Control Block Diagram

A. High Level Control

As mentioned before, the HLC will be the connection between the user's command and the LLC, which controls the actuators of the system. Although the main task of this layer is to convert the received commands and convert them into low-level parameters, it will also evaluate them, and report any unexpected event during the grasping.

		Actuation [boolean]	Velocity [m/s]
Cylindrical Power Grip	Thumb	True	0.06
	Index Finger	True	0.06
	Middle Fingers	True	0.06
Spherical Power Grip	Thumb	True	0.06
	Index Finger	True	0.04
	Middle Fingers	True	0.07
Hook Power Grip	Thumb	False	-
	Index Finger	True	0.04
	Middle Fingers	False	-
Pinch Precision Grip	Thumb	True	0.04
	Index Finger	True	0.04
	Middle Finger	False	-

TABLE I
FINGERS ACTUATION BY THE HIGH LEVEL CONTROLLER

1) Grasping: The robotic hand will be performing different kinds of grips. The grips can be categorized into two classes: power and precision. Upon this project, it will be considered four different types of grasping: Cylindrical, Spherical, Hook, and Pinch Precision. Each one of these grips will have different configurations upon Vizzy's index, thumb and middle fingers.

Within the Cylindrical Power Grip, all fingers will be actuated with equal speed, at the same time and equal force. In the case of Spherical Power Grip, all fingers will also be in action. However, the middle finger's speed will be higher than the thumb and index fingers. This way, the two middle fingers arrive first to the object. Regarding the Hook Power Grip, only the index finger will be actuated. It will move at a slow speed while seeking for contact. Finally, for the Pinch Precision Grip, only the thumb and index finger will be engaged. They will move with equal speed and must reach a point where the same tactile force is detected in contact with the respective object. These parameters and configurations are represented in Table I. Additionally, it is important to emphasize that this is an iterative table, meaning that at the present moment, the used values are the result of theoretical studies, they should be iterated according to the obtained results upon real tests, aiming to improve the high-level controller quality and precision. Furthermore, it is also important to refer that the finger's theoretical maximum speed is around 0.12 m/s and that the maximum detected strength by the sensors is 3N, however the working area for both these parameters will be $v_f = [0.0, 0.07]m/s$ and $f_f = [0, 2]N$. If the imposed force by the user is not within any of the valid classes, then it will not be validated.

The HLC main goal is to assure that the commands sent to the LLC will achieve a stable grasp. One of the most used grasp evaluation criteria is the Force-Closure criteria [28]. This method makes a binary decision depending on whether fingers have the ability to exert forces and torques in any direction. Meaning that, given a wrench applied to an object, the goal is to apply contact forces which generate an opposing wrench. In this way, the fingers can resist arbitrary external forces, and it is possible to assume that a stable grasp is achieved. On the presented work, the classification performed in order to evaluate the qualification of the grasp is more straightforward. It is assumed that if the perpendicular forces in each one of the tactile sensors are proximal to each other, for a certain period of time, and following the desired references from the HLC, then it will be classified as a stable grasp.

B. Low Level Control

In this type of controllers, the goal is to develop a model for controlling systems using a control action in an optimum manner, without delaying or overshooting, and ensuring several goals, such as stability and disturbance rejection; hence, a controller with the requirement for corrective behaviour. This controller monitors a controlled process variable (e.g θ_m), and compares it with the reference or set point (e.g θ_{ref}). The difference between the actual and desired value of the process variable called the error signal (e.g. e_m). The control action goal is to achieve the error signal as near as possible to zero, called the steady state error.

The feedback is given through the sensors available in the system: the tactile sensors on the fingertips, the encoders on the motors and the respective current consumption. The real state of the system can be described by estimating the variables with the data acquired from these sensors. The controller chosen was a proportional-integral-derivative controller (PID). This choice lies in the fact that it has a very robust performance and it has been widely used for various purposes, including robotic hands actuation [29], [30].

When designing the controllers the goals to achieve are: having less than 2% of steady state error with settling time less than 1s, peak time less than 0.5s, and overshoot less than 10%.

IV. IMPLEMENTATION

The local module will be quite complex due to the diverse tasks that it needs to perform. It is necessary to control three DC micromotors, which work at 24V. Each one has a 400 pulses encoder, and it is also necessary to acquire the respective current consumption of each one of them. Logically, it is also necessary to acquire the tactile sensors data, which communicates through I2C. In the end, all of this data needs to be streamed to the central system, and due to the high sampling frequency proposed, it is also necessary to have a robust communication protocol between the device and the primary system. The chosen one for this work was Ethernet, due to its properties in terms of speed, data transmission and reliability.

An electromechanical system simulator was developed for testing, validating and tuning the development done to perform the grasp control. Therefore, every mechanical component of the grasping system needs to be modulated, validated and tested under various conditions.

Regarding the hardware that controls such system, there are some essential features which need to be considered and, knowing that this is a local solution, the space to attach the

PCB is quite limited. In order to meet the space requirement, the majority of the components will be SMD, and the PCB will have four layers.

A. Sensors and Actuators

1) DC micromotors: The mathematical equations which the DC motor is governed can be divided into two sections: the electrical and the mechanical.

The electrical part equations are a consequence of applying basic electronic laws to the DC motor model, where the rotor is assumed to be a single coil characterized by inductance - L -, resistance - R -, and the back electromotive force (EMF) - e. The equation associated with such electric circuit is a direct output of using the Kirchoffs Voltage Law: the sum of the voltages of a closed (series) circuit is equal to the voltage source.

$$\frac{di}{dt} = \frac{1}{L}(V - K_e \phi \omega_m - iR) \tag{1}$$

In the mechanical part, the relation between the load torque, motoring torque, the inertia moment of motor's shaft, angular velocity and angular acceleration is expressed in an unique equation (2), knowing that the motoring torque has a direct relation with the current in the armature.

$$\frac{d\omega_m}{dt} = \frac{1}{J}(K_M \phi i - B\omega_m - T_L) \tag{2}$$

Therefore, it is possible to describe and completely modulate a DC motor with the equations (1) and (2). The DC motor actuation control can be achieved by varying the applied armature voltage and, this usually implies the use of a power driver between the motor and the microcontroller unit (MCU) that controls it. The specific case of this project, the motor control is done through two logic inputs, that represent four different modes of the motor's operation: static, rotate right, rotate left or brake. This control only works if the driver enable pin is activated. Moreover, it is also through the enable pin that the DC micromotor speed is controlled. In terms of hardware, this control is achieved using a Pulse-Width Modulation (PWM) signal in the enable driver's pin.

The DC Motor model is a crucial part of the electromechanical simulator, and there are different ways to model it. However, for this case, the model is generated using the mathematical equations that govern the DC motor, (1) and (2). The modelling of the DC micromotor is done with a specific *Simulink* tool - *Simscape*, where the model is generated with the DC motor Electrical equivalent scheme, Figure 4.

This model will be a part of a block that will interact with other blocks during the simulations. For this reason, some extra elements were added, such as a controlled voltage source, a current sensor, an ideal rotational motion sensor, and electrical and mechanical references.

The current setup on Vizzy's DC motors includes a gearbox, a Series 14/1 Planetary Gearhead, from Faulhaber. This gearhead has a ratio of 415:1, and it is necessary to include it in the motor model to achieve the realist model. For that, a Gearbox Simulink Block was used. This block represents an

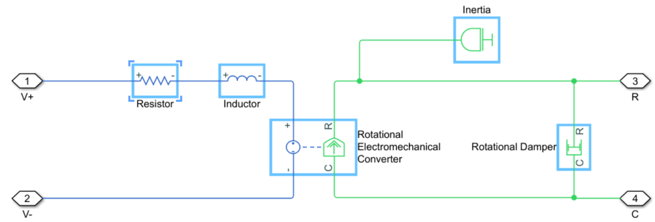

Fig. 4. Simscape DC Motor Model

ideal and fixed gear ratio box and it leads to a decrease of the motor speed by a factor of 415 at the output of the gearbox, and increase, by the same factor, the DC motor torque.

2) Encoders: In this project, the encoders used - IE2-400, from *Faulhaber* - are optical rotary without the index channel. This kind of encoders, in particular, uses a translucent disk containing equally spaced opaque sections, which determines movement. A LED is used to pass through the glass disk, which afterwards is detected by a photodetector, causing the encoder to generate a train of equally spaced pulses as it rotates.

The rotary encoder is also referred to as a quadrature encoder. The principle behind this type of encoder is that the encoder generates an output signal each time the shaft rotates a certain amount - the number of signals per turn defines the resolution of the device. Each time the encoder is powered on, it begins counting from zero, regardless of where the shaft is. Initial homing to a reference point is therefore inevitable in all positioning tasks, both upon startup of the control system and whenever power to the encoder has been interrupted. A single-channel output is commonly implemented in applications in which direction of movement is not significant. Otherwise, it is required to have a two-channel, quadrature, output. The two channels, A and B, are commonly 90°out of phase, and the electronic components determine the direction based on the phase relationship between them. This way, given that the output is in pulses per revolution, it is possible to measure the motor displacement and speed.

Knowing the lines per revolution, in this specific encoder, is 400, the respective resolution is then given by (3).

$$Resolution = \frac{360^\circ}{400} = 0.9^\circ = 0.0157rad \tag{3}$$

This resolution will induce an error, that can be categorized as a uniform distribution. In order to correctly model this error, a quantizer *Simulink* block was used, with the respective resolution value.

3) Tactile Sensors: The tactile sensors that are equipping Vizzy's fingers have three magnetic field dimensions with a permanent magnet inside of an elastomer material [31]. It allows the sensors to achieve a firm contact with the environment while detecting normal and shear forces. In order to perform this detection, the sensors use the Hall Effect as a transducer mechanism. The relation between the perpendicular forces applied, and the deformation of the material can be described by the equation (4), where F represents the perpendicular force applied (in Newton), and X is the normal displacement (in meters).

$$X = 3.75 * 10^{-4} F + 4.4 * 10^{-4} \tag{4}$$

These tactile sensors are characterized as a mass-spring-damper system in order to model them. It is necessary to add other components to correct model these sensors in a Simscape environment and prepare it to interact with the other systems, Figure 5.

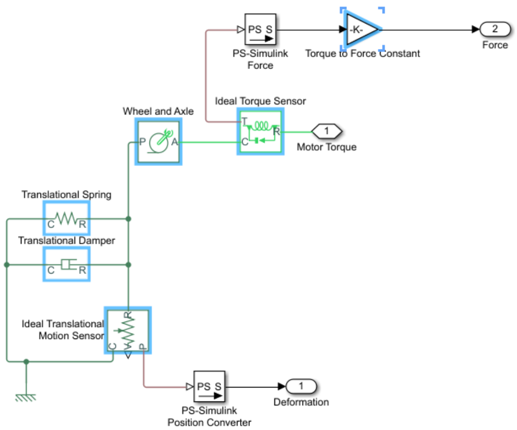

Fig. 5. Tactile Sensor Model

B. Design

The hardware development was made with the software program Altium Designer, due to personal experience and to the fact that this is one of the most popular software in PCBs design.

This hardware project is divided into three different units: the main sector, the communication sector and the sensors and actuators sector. A simplified schematic is provided in Figures 6, 7 and 8, for each one of the units, respectively.

The main unit aggregates the data from the other units, use it to decide how to control the system, actuates the motors and afterwards streams all the data to the main system.

The communications unit is the one responsible for managing the data in the local system. It has to have the necessary components to deal with the data from the sensors and actuators and send it to the main unit. It also deals with the components that allow the data stream from the local unit to the main system, Vizzy.

The sensors and actuators unit is composed of the components that allow the microcontroller to control the DC micromotors, and sampling the sensors.

The PCB is a 4-layer board, with a 0.8 mm thickness, manufactured by Eurocircuits, with 54 per 39 mm, Figure 9.

The majority of the components, including all connectors, are in the top layer, due to mechanical restrictions. The components in the bottom layer are the shunt resistors and

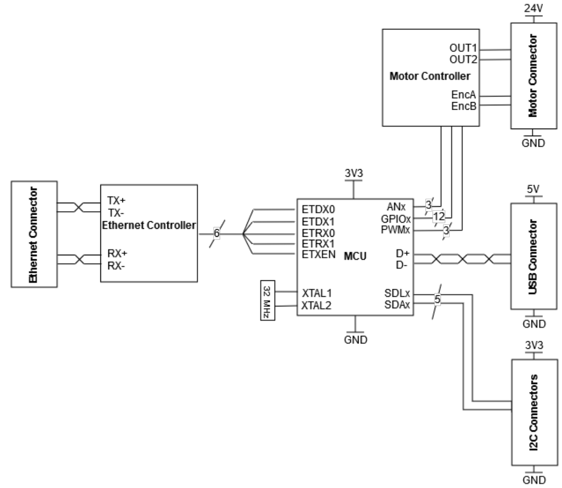

Fig. 6. General Schematic for the Local Hand Control Board

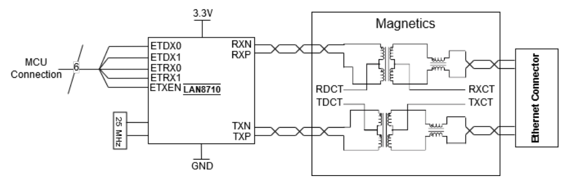

Fig. 7. Schematic for Communications Unit

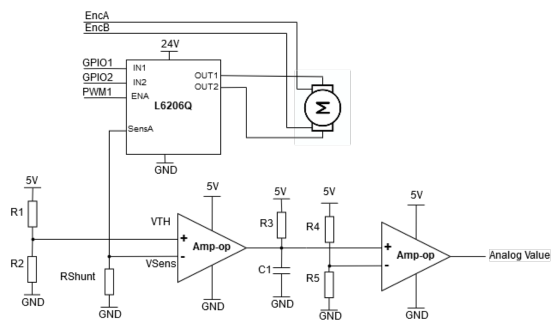

Fig. 8. Schematic for DC Motor control

the motor's drivers that will be in touch with the aluminium support for cooling the module.

1) Communication: In this module, Single-Ended Signaling (I2C) and Differential Signalling (USB and Ethernet) are used, which requires special cares when designing the hardware.

In single-ended signalling the electrical signal is transmitted by a voltage, which is referenced to a fixed potential; usually 0V - referred as "ground". The current associated with the signal goes from the transmitter to the receiver and returns to the power supply through the ground connection. In the Differential signalling, two complementary voltages signals are employed in order to transmit one information signal. This means that one information signal requires a pair of conductors: one to carrier signal and the other to carries the inverted signal.

It is quite complex to integrate differential pairs, but there

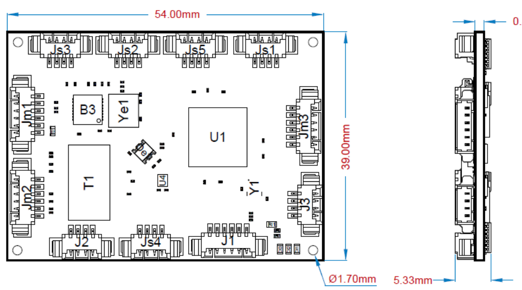

Fig. 9. PCB Dimensions

are many benefits to do it. Especially the resistance to incoming electromagnetic interference (EMI) or crosstalk, generated by nearby signals. Another benefit is the reduction of outgoing EMI and crosstalk, that generally comes from rapid transitions, like rising and falling edges of digital signals, and generates significant amounts of EMI. This happens for both single-ended and differential pairs. However, the two signals in a differential pair will create electromagnetic fields that are equal in magnitude but opposite in polarity, this ensures that the emissions from the two conductors will largely cancel each other out.

C. Controllers

1) Kinematics: A straightforward method was used to determinate the relationship between the DC motor actuation and the respective finger actuation. Considering a test that consists of moving Vizzy's index finger from its reference position to a 90°angle and returning to the initial position. It is possible to estimate the path that the finger does in this test, it can be approximated to a quarter the perimeter of the circumference with a radius equal to the length of the finger, Figure 10. Assuming that the absolute finger speed is constant is possible to estimate the finger position according to the encoder position.

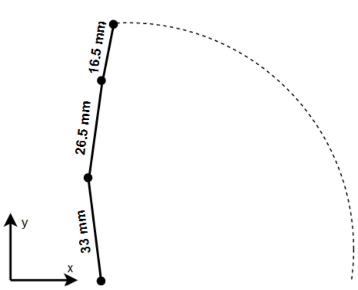

Fig. 10. Estimation Index finger path during practical test

Since the total length of the finger is $Total_{length} = 0.076m$, it is possible to approximate that during the test the finger total course would: $Total_{\Delta X} = \frac{2\pi * 0.076}{2} \approx 0.238m$. The equation (5), extracted from the relationship between the encoder values during the test and the estimated values for the finger displacement, reflects the relation between both variables. The X is the total displacement of the finger in meters, and the θ is the angular displacement in the encoder, in rad.

$$X = 0.0332\theta - 0.0041 \tag{5}$$

This relation has a y-intercept factor due to imprecision on the encoder and approximations made, however, the relation should be an equation that passes in origin because there should only exist movement on the finger when there is actuation on the DC micromotor, and both should start from the reference position. Therefore an approximation was done, and the y-intercept factor was not considered. Another important parameter that can be extracted from (5) is the relation between the motor angular velocity and finger linear velocity because it is known that speed is the derivative, in time, of the position. The relation between velocities is the constant $K_v = 0.0332$.

2) Free Trail Control: The HLC will transmit the desired speed for each finger, considering the type of grasp chosen. After that, the Free Trail control will start, and its goal is to achieve the desired speed as fast as possible and keep it stable. For that, a speed controller will actuate the system, the block diagram of Figure 11 represents this controller.

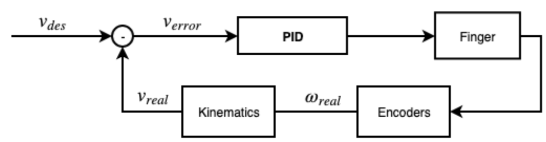

Fig. 11. Speed Control Block Diagram

The PID controller deals with the error between the real speed at the finger and the desired one, as mentioned in Chapter III-B. The real speed is not a direct output of the encoders, but it can be achieved by using the displacement of the encoder and the timestamp. Then, it is necessary to apply the finger kinematics to the encoder value to have the finger linear velocity real error.

Until an object is detected, the speed controller maintains the speed output, if no object is detected, then the finger comes back to the reference position and generates an event for the high-level controller.

3) Contact Control: When the tactile sensors detect a contact, the free trail control is no longer useful, something more complex is needed. Now the system will have two interesting variables that could control: the force detected by the tactile sensors and the position of the fingers, determined by the micromotors encoders. The position can estimate the object and tactile sensor deformation.

The first option to be considered is the deformation control. The first step in this controller is converting the desired force in the desired deformation, and this is possible by knowing the relation between the two variables in the tactile sensors used in this work, (4). An essential parameter is the absolute position where the contact happens, $P_{contact}$. This is essential because the position parameter is not only the absolute finger position but the sensor deformation as well, and the contact deformation will be estimated by the displacement of the finger absolute position. Therefore it is needed to know at which point the contact was detected, to separate the position value in displacement in the free trail section and contact section. In summary, this is a position controller for very small displacements, and the respective conceptual diagram can be seen in Figure 12.

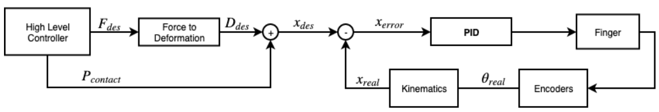

Fig. 12. Conceptual organization of deformation controller

Other option to be considered is a pure force controller. Here the variable under analysis is the error between the desired force commanded by the HLC and the real forces detected by the tactile sensors. The conceptual organization of this control is similar to the one presented for speed control in the previous section and can be seen in Figure 13.

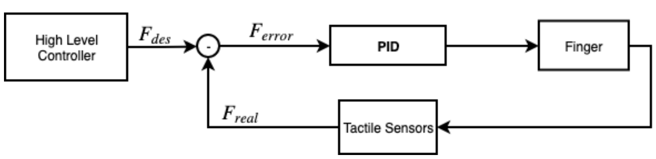

Fig. 13. Conceptual organization of force controller

The final option for the contact controller is a parallel hybrid controller similar to the one proposed by Khalil and Dombre [24], where each finger is controlled by two complementary feedback loops, one for the deformation and the other for the force. The control laws of both loops are merge together and then sent to the system, but when added together they will have different weights. A moderator block does this management of weights. Moreover, the block diagram is shown in Figure 14.

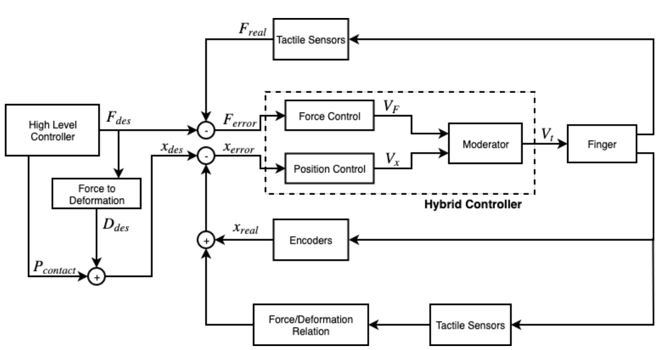

Fig. 14. Conceptual organization of hybrid controller

V. RESULTS

The gains for every controller were tuned using the auto-tune tool from *Simulink*. For the non-contact part control, the controller chosen to actuate the system was a speed controller. However, for the contact part, the decision is pending on the simulations results. The options are a deformation, a pure force or a hybrid controller.

A. Controllers

1) Free Trail Control: The controller that showed the best results for the speed control was a Proportional-Integral (PI) controller.

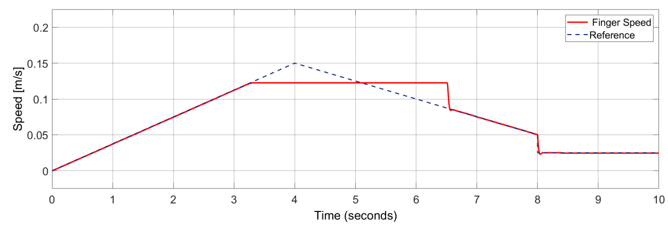

Fig. 15. Speed Control Signal Response

The Figure 15, shows the controller following a complex speed reference. The reference tracking is very satisfactory, and the requirements are met: the overshoot is around 9%, the settling time is 0.03s, and the steady state is achieved in less than 0.2s. In the period of time between 3 and 5 seconds it is not possible for the controller to follow the reference because the reference overcomes the maximum finger speed, therefore, the controller establishes the finger speed to its maximum during that period. As soon as the reference goes to acceptable values, the recovery begins. It takes some time to "catch up" the reference because during the previous period, not only it had an error, but this error was accumulating due to integrative part of the controller. Nevertheless, the recovery happens with softness, and the overall results are quite satisfactory.

2) Contact Control: Deformation: The deformation controller is a position control for the movement caused by the deformation of the tactile sensors during contact. As mentioned before, this controller receives a force reference, that is transformed in a deformation reference. In the following tests, the contact position is zero because the goal is to analyze the system during contact.

The most suitable controller for the deformation control is a proportional-derivative (PD) controller, according to the tests performed. In Figure 16 is presented the deformation controller step response to a force of $F = 1N$, meaning a deformation of $d = 8.15 * 10^{-}4m$. It shows uncertainty when following the reference, and this happens due to the encoder resolution. As explained in Chapter IV-A2, the resolution is $r_\omega = 0.0157rad(0.9)$. This means that the error on the absolute finger position is $r_P = 7 * 10^{-4}m$, and that is exactly the interval of values that the signal is oscillating around the deformation reference. Therefore, the deformation controller is not a good option for the contact control with the current mechanical setup, because the resolution is not precise enough.

3) Contact Control: Force: The force controller receives the reference from the HLC, as mentioned before, and the

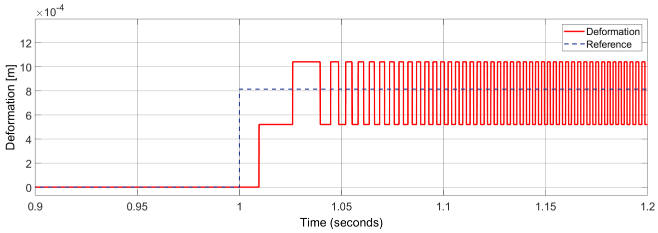

Fig. 16. Deformation Control Signal Response

best setup for the controller is a PID. As it is possible to check in Figure 17, the results are quite good. It achieves all the propose goals regarding the overshoot, 3.6%, the peak time is around 0.06s, and the time that it takes to achieve a steady state is less than 1s.

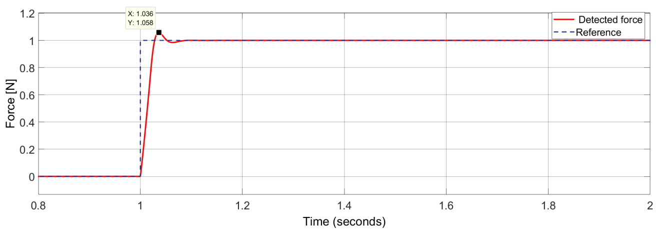

Fig. 17. Force Control Step Response

4) Contact Control: Hybrid: The final option for the contact control phase is the hybrid controller, where the two previous controllers, force and deformation, are merged. This merge happens to assign different, but complementary, weights for each one of the controllers outputs, as mentioned before.

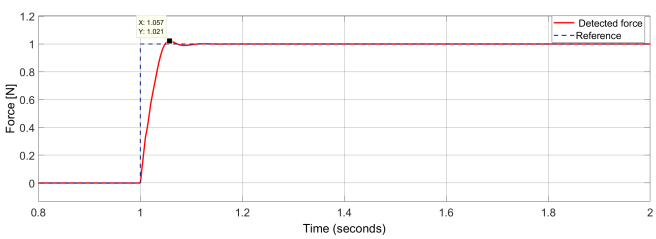

Fig. 18. Hybrid Control Force Step Response

After several simulations, the best suitable values for the moderator is 30% for the deformation control part and 70% for the force controller. This decision relays on the fact that the simulations demonstrated an inevitable imprecision on the deformation control, and this causes errors on the reference following.

In Figure 18, the step response to a $F = 1N$ reference is shown. The results are good, the tracking is very fast, and the reference following is very satisfactory, the overshoot, 2.1%, is even lower than in the pure force controller, but the peak time, 0.057s, which met the requirement but is slower than the pure force controller.

B. Total System

In the final simulation, the total architecture is tested. The index finger is subjected to a firm hook grasp, in which the respective references are: $v_{desired} = 0.04m/s$ and $F_{desired} = 1N$. The controller chosen to the contact part of the grasp was the hybrid controller because it reveals the best force tracking. The simulation consisted of starting the grasp with no contact, and at the fourth second of simulation, contact is detected.

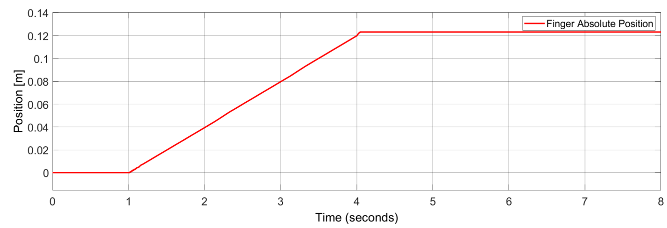

Fig. 19. Index finger position during grasp

Figure 19 shows the absolute position of the finger during the simulation. It is possible to confirm that the actuation started at 1s, and that the finger encountered contact within after 4 seconds of the simulation start. After that moment, the displacement of the finger is due to the deformation of the tactile sensors, that is why it slowly decreases and, eventually, stabilizes after achieving the desired force.

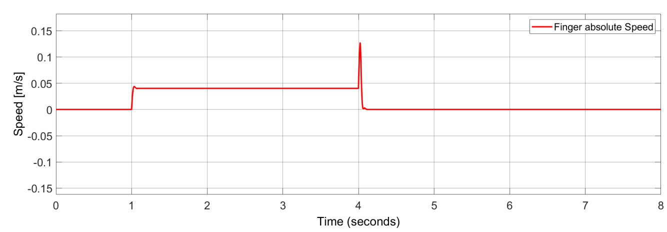

Fig. 20. Index finger speed during grasp

The Figure 20 also confirms the detection at that time. Until contact, the controller follows the given speed reference. When the contact is detected there is a peak of speed, this happens because the DC micromotor is trying to overcome that obstacle exercising more torque (and consequently more force at the finger), however, at that same moment the hybrid controller will start actuating the system trying to follow the force reference. This transition behaviour can be seen in the current consumption of the DC motor, Figure 21, where is evident that the peaks of consumption are referred to the two main events on the grasp: the beginning and the contact detection. In the contact detection, the motor tries to increase the pressure on the object detected until the desired force is achieved, to increase this pressure, the motor needs to apply more torque to the shaft, and that means more current from the DC motor.

In the force analysis the result equals the expected one, where the controller performs a very satisfactory force reference tracking.

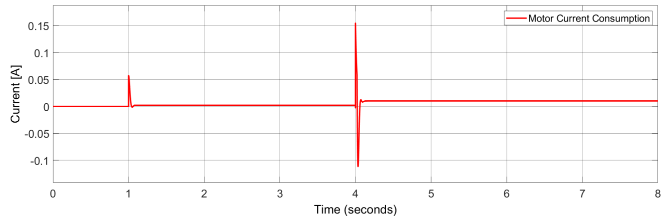

Fig. 21. Index finger DC Motor Current Consumption during grasp

VI. Conclusion

A. Achievements

This work goal was to create a solution that could provide an independent, standalone hardware module capable of receiving different grasping commands and perform them in real-time. This would imply several sub-goals, as mentioned in Section I-B. In this line of work, several goals were achieved. A layered system architecture was proposed and validated to be the correct choice for grasping control. Every important component in Vizzy's hand for the grasping process was correctly modulated and tested. Although only for the index finger, robust controllers were designed and proved to be efficient for both contact and non-contact parts of the hand actuation.

After analyzing all of the simulations results and knowing how well every component of the system is modulated, it is possible to state that a stable grasp can be achieved by using the material developed in this thesis.

B. System Limitation and Future Work

The most significant limitation of this thesis is the lack of real tests. The controllers showed good results and the system was proved to be well modulated; however, it is still required to test the controller in the real system and perform fine tuning of the controllers.

Every simulation and controller design was done solely for Vizzy's index finger. One of the future work topics would be to perform the control design and tuning for every other finger in order to have the Vizzy's complete hand modulated, using a similar process than the one used in Vizzy's index.

Another limitation of this system is the fact that the HLC does not send new references during the grasping process, besides the initial references. Some research is done on this topic, states that a more robust grasp can be achieved if the HLC iterates the references during the grasp according to the feedback from the LLC. This way, the system will work in an adaptive way while performing a more suitable grasp for the specific object.

References

[1] E. Bruce Goldstein, *Sensation and Perception*, Cengage Learning, 2009.

[2] Ahmed M. Almassri, W. Z. Wan Hasan, S. A. Ahmad, A. J. Ishak, A. M. Ghazali, *Pressure Sensor: State of the Art, Design, and Application for Robotic Hand AhmedM*, Journal of Sensors, 2015.

[3] Radhen Patel, Jacob Segil and Nikolaus Correl, *Robotic Grasping and Manipulation, Chapter 1*, Springer International Publishing 2012.

[4] Filipe Veiga, *Robotic Grasp Optimization from Contact Force Analysis*, Instituto Superior Tecnico, 2012.

[5] H. J. Landau, *Sampling, data transmission, and the Nyquist rate*, Proceedings of the IEEE, Vol. 55, pp: 1701-1706, 1967.

[6] Saxena, Ashutosh Driemeyer, Justin Kearns, Justin Ng, Andrew Y, *Robotic Grasping of Novel Objects*, Advances in Neural Information Processing Systems 19, 2018.

[7] R. S. Dahiya and G. Metta and M. Valle, *Tactile SensingFrom Humans to Humanoids*, IEEE Transactions on Robotics, 2010.

[8] A. Bicchi, *Hands for dexterous manipulation and robust grasping: a difficult road toward simplicity*, IEEE Transactions on Robotics and Automation, vol. 16, no. 6, pp. 652-662, Dec. 2000.

[9] R. Nunes, R. Beira, P. Moreno, L. Vargas, J. Santos-Victor, A. Bernardino, *Vizzy: A humanoid on wheels for assistive robotics*, 2nd Iberian Robotics Conference, Lisbon, Portugal, 2015.

[10] Coste-Maniere, Eve and Simmons, *Architecture, the backbone of robotic systems*, Proceedings - IEEE International Conference on Robotics and Automation, 2000.

[11] Asada H. *Studies on Prehension and Handling by Robot Hands with Elastic Fingers*, PhD Thesis, Kyoto University, 1979.

[12] Mason M T, Salisbury J K *Robot Hands and the Mechanics of Manipulation*, MIT Press Cambridge, 1985.

[13] N. Fukaya and T. Asfour *Development of a five-finger dexterous hand without feedback control: The TUAT/Karlsruhe humanoid hand*, RSJ International Conference on Intelligent Robots and Systems, 2013.

[14] David S. Touretzky, Ethan Tira-Thompson, *Architectures for Robot Control*, http://www.cs.cmu.edu/afs/cs/academic/class/15494-s07/lectures/architectures.pdf 2008.

[15] R. Brooks, *A robust layered control system for a mobile robot*, IEEE Journal on Robotics and Automation, 1968.

[16] Kakoty Nayan M., Hazarika M., *A Two Layered Control Architecture for Prosthetic Grasping*, Journal of Behavioral Robotics, 2013.

[17] Sebastian Engell, Goran Frehse, Eckehard Schnieder, *Modelling, Analysis, and Design of Hybrid Systems*, 2002, vol = 279.

[18] Shanahan, Murray and Witkowski, Mark, *High-Level Robot Control Through Logic*, 2000.

[19] Ramezanifard, M. Dargahi, J. Najarian, *Design, Modeling and Fabrication of a Tactile Sensor and Display System for Application in Laparoscopic Surgery*, Engineering and Technology, 2009.

[20] K. M. Kasim, M and Ahmad Shauri, *PID position control of three-fingered hand for different grasping styles*, pages: 7–10, 2015.

[21] Stoll, Jeffrey and Dupont, Pierre *Force Control for Grasping Soft Tissue*, Proceedings of the IEEE International Conference on Robotics and Automation, 2006.

[22] D. Gunji et al. *Grasping force control of multi-fingered robot hand based on slip detection using tactile sensor*, 2008 IEEE International Conference on Robotics and Automation, pp. 2605–2610.

[23] M. H. Raibert and J. J. Craig *Hybrid Position/Force Control of Manipulators*, Journal of Dynamic Systems, Measurement, and Control, pp. 126–133, 1981.

[24] W. Khalil and E. Dombre, *Modeling Identification and Control of Robots*, Butterworth-Heinemann, Chapter: 15 2004.

[25] Franz Reuleaux, *Theoretische Kinematic. Translated 1963 as Kinematicsof Machinery*, 1875.

[26] Salisbury, J. Kenneth., *Kinematic and force analysis of articulated hands*, Stanford University. Department of Mechanical Engineering 1982.

[27] J. Montana, David, *Contact Stability for Two-Fingered Grasps*,Robotics and Automation, IEEE Transactions on 1992.

[28] Richard M. Murray, Zexiang Li, *A Mathematical Introduction to Robotic Manipulation*, Shankar Sastry University, CRC Press 1994.

[29] J. Montana, David, *Position and force control of a robotic finger with twisted strings actuation*, IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) 2011.

[30] N. Tsujiuchi, T. Koizumi, H. Kan, H. Takeda, *Modeling and control of a joint driven by pneumatic actuator*, Proceedings of 35th Annual Conference of IEEE Industrial Electronics 2009.

[31] Tiago Jos Pinheiro Paulino, *3-Dimensional Soft Magnetic Tactile Sensors for the Human-Friendly Robot Vizzy*, Instituto Superior Tecnico, Master Thesis, 2016.