Hybrid force/position grasp control for underactuated hands in physics engine simulation

David Alexandre Castanheira Ribeiro
david.a.c.ribeiro@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

September 2021

Abstract

Manipulation of objects is a key factor to allow humans to perform their tasks and it is divided into two different phases: reaching, here the vision is the main sensor, and grasping, where the fingers apply force and use tactile feedback to constantly evaluate if the grasp is stable.

Vizzy robot has a local hand control unit that works at a high frequency as a low-level force feedback controller, allowing a hierarchical control architecture. Its fingers have three DC motors with one encoder each and a total of twelve joints. The hand is under-actuated and under-observed due to the lack of encoders. Thus, making it essential to close the loop with tactile information, mainly when a load is applied on the finger and simple position control of the joint is not effective.

The main objective of this work was to develop a grasp control system based on position and tactile force feedback in Gazebo simulated environment. This was done with a hybrid position/force controller. The DC motor encoders give feedback to control the finger while it is not touching the object. When contact is detected, the control is done with the force measured by the sensors.

The control system developed was able to perform precision and power grasps, have taken into account the physical properties of the real robot, hardware available and its control architecture. The quality was assured by introducing noise to the actuators of the hand and measuring the force applied by the fingers.

Keywords: Grasp control, Tactile feedback, Force control, Grasp in simulated environment, Grasp stability, Hybrid control

1. Introduction

Humans have great abilities to manipulate objects in a vast range of conditions [1]. However, transferring this ability to robots is not as trivial as it seems. To get robots to take our places in repetitive tasks like washing dishes, chopping vegetables, and folding clothes, or save us the effort of carrying heavy loads, implies giving them strong handling capabilities.

The process of manipulating an object can be divided into two phases [3]. The first one, when the hand is guided to the correct position following visual information, and the second one, when tactile feedback is used to control the force applied to the object and prevent slippage, performing a stable grasp.

In this work, the main objective is to develop a grasp control system for Vizzy hand (figure 2) in a simulated environment.

Vizzy is a human-like upper-body and a car-like lower-body robot. It has been designed to interact and help people with daily tasks. It can be seen in figure 1. His hands have four similar fingers and three DC motors. As can be seen in figure 2 motor 1 is connected to the thumb, motor 2 is connected to the index and motor 3 to the remaining two fingers. These connections are done through a
string that pulls the finger under a motor actuation. Each finger is equipped with three magnetic tactile sensors, one at each phalanx, that work with Hall Effect as a transducer mechanism, and relates the voltage measured with the force applied in the surface of the sensors, a string on the palmar side that goes through all the finger phalanges, from the pulley to the fingertip, and three dental rubber bands on the dorsal side of the fingers, to assure that the hand opens when the motor unwinds the string.

Figure 2: The image on the top left corner shows the motors that pull the strings that move the fingers (1 for the thumb, 2 for index and 3 for the remaining two fingers). The image on the top right shows in orange the tactile sensors. The bottom left image shows the string path at cyan. And finally, the bottom right shows at cyan the dental rubber bands [8].

Vizzy hands are also equipped with a compact four-layer PCB developed in [2]. This local module is responsible for the grasp low-level control and should work at a rate of 1000 Hz to be able to process the same bandwidth as human tactile sensing. Human skin can detect frequencies until 500 Hz [6], and according to Nyquist [7] the sampling frequency has to be at least double to assure the correct sample signal.

The objective of this work is to produce a grasp controller that is able to, from an open hand, use position control to move the fingers to the object and to use tactile feedback to track force references for each finger to perform a stable grasp. This is done in Gazebo simulator, with a hybrid force/position controller, with force references determined based on the kind of grasp, the mass of the object and friction coefficients of the contacts.

2. Background

The main concern of this work is the low-level controller of the grasping process. This controller has to run continuously, in a high-frequency loop. The measurements come from the encoders, to control the position of the fingertips, and from tactile sensors, that gives both intrinsic (used at grasp, the magnitude of force at the contact point) and extrinsic (detect slippage) data, and it is used to grasp and maintain the grip [13]. These measurements from the contact of the tactile sensors need to be updated at a really high rate for accurate and timely detection of slip [5].

2.1. Hybrid force-position control

One of the most popular approaches to do the grasping low-level control is the hybrid force/position control. It allows to specify the motion during the unconstrained task directions and the force of contact during the constrained task directions [11].

The main idea of hybrid control is to have distinct closed loops for each degree of freedom of the system. These loops can do the control based on force or in position feedback, and act independently to control each one of the joints.

A degree of freedom could be constrained in motion by an environment force. In those cases, the feedback control is based on force instead of position. The hybrid control actuator drive signal has one signal for each joint, where some will be loops controlled by force and others by the position. All the controllers start with position control and have a switch (contact detection) to change for force control. In both cases, the actuator is the same. Assuming a revolut joint that works with Torque \( T_i \), the value of actuation can be described by the following expression:

\[
T_i = (1 - S_i) \cdot C_{P_i}(\Delta P_i) + S_i \cdot C_{F_i}(\Delta F_i),
\]

where \( S_i \) is a Boolean value (0 or 1) that represents the detection of contact in the \( i \)th joint, \( \Delta P_i \) and \( \Delta F_i \) the difference between the position and force desired and the one measured, respectively, and \( C_{P_i} \) and \( C_{F_i} \) the respective control functions. This approach of control has been proved stable for grasping control by Raibert and Craig [11].

2.2. Grasp stability

In order to validate the quality of the grasp is necessary to evaluate its robustness and stability. The conventional way of evaluating robustness and stability considers form–closure and force–closure. These concepts were introduced by Reuleaux [12] and describe the capacity of constraining the motion of the object that is being grasped.

Form-closure implies that a contact point does not have any kind of motion in any direction [5]. The object is not able to move, independently of the force applied in the contact points, because it is totally constrained by the set of contacts.
According to Nguyen [9] a grasp on an object is force-closure if and only if it is possible to exert arbitrary force and moment on the object, through the set of contacts. It means that the force applied offers resistance to the motion of the object, and the contact between the fingertips and the object is not going to be lost without actuation of external work.

Figure 3: Examples of force-closure grasps [9].

While in form-closure the object is constrained by the set of contacts, the force-closure depends on the relation between the force applied and the contact friction, as well as the location of the contact points.

2.3. Friction force

According to Coulomb, the frictional force between two objects that are being pressed against each other presents two properties, static friction and kinetic friction [10].

Static friction happens when the objects are static in relation to the other. Considering an object resting on a surface, to move the object, it is needed to apply a force that overcomes the force of static friction \( F_s \) that is given by

\[
F_s = \mu_s F_N,
\]

where \( F_N \) is the normal force applied by the object against the surface, and \( \mu_s \) is the coefficient of static friction.

Kinetic friction happens when the force of static friction \( F_s \) has been overcome and the object is moving, and a resisting force \( F_R \) acts in the opposite direction of movement of the object, as can be seen on the scheme of the figure 4 and it is given by

\[
F_R = \mu_k F_N,
\]

where \( \mu_k \) is the coefficient of kinetic friction.

The friction coefficients \( \mu_s \) and \( \mu_k \) show dependency on the material of contact of the surfaces, but do not depend on the area of contact either in the roughness. Both coefficients have similar values. These proportionalities between the normal force applied and the force of friction are known as Amontons’s law [10].

Figure 4: A block being acted upon by normal and tangential force on a plane; in the corresponding free body diagram, the reaction force and the frictional force can be seen [10].

Figure 5: Example of a gripper that grasps objects with friction force [1].

In figure 5 it is represented a gripper that is able to secure an object by applying two forces in opposite directions. To have the object static in the hand the force of static friction cannot be overcome. Knowing the static friction coefficient and the mass of the object, using the expression (2), the force to be applied by the gripper has to be enough to create a force of static friction higher than the weight

\[
W \leq 2C_f F \iff \frac{W}{2C_f} \leq F.
\]

So, it is necessary to assure that the force \( F \) applied by the gripper (in both sides) has to be higher than the ratio of the weight \( W \) by two times the coefficient of friction \( C_f \), to have a grasp.

3. Implementation

To develop a proper control system for the hand it is necessary to have a realistic simulation of the hand, a way to read the contacts in the fingers and an actuation over the fingers. In Vizzy’s hand, there are four fingers and three phalanges per finger where each one has a tactile sensor. It also has three DC motors, that will actuate over the fingers (little and middle fingers share the same motor). The hand is described by a URDF file and controlled in Gazebo by a model plugin. To get the contacts between the hand and the objects a world plugin
is used. Both the plugins will be exchanging messages with the program that will be responsible for the control of the hand.

3.1. Vizzy hand in the simulator

The Vizzy’s hand is composed of a group of links and joints in a URDF file. Most of the links are described by mesh files, with some parameters defined in the URDF file, such as mass, colour, inertia, etc. To allow movement on the hand there are two different types of joints. The virtual joints are responsible to move (prismatic joint) and rotate (revolute joint) the hand, allow it to float freely in the space without being attached to an arm. In the real joints, there are two types used: the fixed joints that implement a rigid attachment between links, and revolute joints to move the fingers and phalanges. The joints are characterised by the child and parent links references, the axis of rotation or movement, and have position, velocity and effort limits set in the URDF file, as well as physic properties such as friction and damping factor. To have the most similar behaviour as possible compared to the real hand, the values of velocity and effort limits and damping factor are set based on the values obtained in [2] by Arreda with the Vizzy robot. From there it is possible to get that for each joint of the fingers that the maximum velocity is 3.68 rad/s, maximum torque is 1.2 N.m and the damping factor is 0.326 N.m/(rad/s).

Vizzy just has three DC motors to control all the twelve joints of the fingers. The index and thumb have a motor for each and the little and middle share the same motor. In figure 6 is possible to see how the actuation over the joints of the fingers works. The red dots are the joints that are directly actuated. The joints with the green dots share the actuator with the red ones which they are connected to, and are called mimic joints.

<table>
<thead>
<tr>
<th>Nº of the joint</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thumb (th)</td>
<td>0.5 * $T_{th}$</td>
<td>0.5 * $T_{th}$</td>
<td></td>
</tr>
<tr>
<td>Index (ind)</td>
<td>0.5 * $T_{ind}$</td>
<td>0.5 * $T_{ind}$</td>
<td></td>
</tr>
<tr>
<td>Middle (mid)</td>
<td>0.5 * $T_{mid}$</td>
<td>0.5 * $T_{mid}$</td>
<td></td>
</tr>
<tr>
<td>Little (lit)</td>
<td></td>
<td>0.5 * $T_{mid}$</td>
<td>0.5 * $T_{mid}$</td>
</tr>
</tbody>
</table>

Table 1: Multiplier factor of the joints.

In the table assuming that torque is being applied, and the joints are numbered 1 to 3, where 1 is the joint next to the hand and 3 the last one in the finger, the torque applied in the green dots is proportional to the torque applied by the red ones with a factor of 0.5, except the first joint of the little finger that has a factor of 1.

The plugin to control the hand, HandPlugin, loads the information from the URDF file, sets the PID gains for the joints and the initial pose for the hand and subscribes the topic with the messages that contain the information about the type and value of actuation for the joints.

3.2. Contact detection

The contact detection is made based on a world type plugin. This plugin loads the contact manager of Gazebo and subscribes to a request topic. It will check and return for all contacts at a high rate (around 3 kHz). There are multiple contacts in Gazebo environment, for instance, between the ground and the objects, between objects, between the palm of the hand and objects, etc. For all those contacts just the ones with the sensors on the phalanges of the Vizzy’s hand are important, so all the other ones are discarded. The sensor on the phalanges could have more than one contact point with the object, so all the contacts are summed to have the total force applied in the phalange.

The tactile sensors detect force, that comes split into x, y and z components according to the world reference frame. So it is necessary to get the link frame and determine the rotation matrix to transform those components from the world reference frame to the link frame, like the one represented in figure 7. This operation allows getting the forces in the link frame, as in the real hand, where the measurements of the sensors depend on the pose of the finger.

The force of contact obtained is published split into the x, y and z components according to the link frame. The message also has the name of the link of the contact to identify the phalange. This information will be used by the controller as force measured on the phalange to perform a closed-loop control.
3.3. Tactile Force Controller

To develop the controller, a test is performed, where the hand works between two heavy boxes so it is not able to move. The test is composed of two distinct phases. In the first one, the finger is moving to the object and in the second one, it is applying force over it. Both stages have a specific control strategy that works with different types of data for their specific purpose. The switch between the first stage to the second one is done when a contact is detected. The flowchart from figure 8 shows how this process is done.

The finger needs to reach the object to apply force over it. During that stage, there is no contact and the force measured in the tactile sensor is zero. The best way to control the finger with feedback is by position control. The DC motors of Vizzy hand are equipped with an encoder that is able to measure the angle done by the finger in the first joint. With the information that comes from the encoder is possible to apply a PID controller to set the position of the finger with the following gains.

$$g(t) = K \ast u(t - \tau),$$  \hspace{1cm} (5)

where $g(t)$ is the response of the system to an input torque step signal $u(t)$, $K$ is a proportional gain that depends on the angle of contact of the finger with the object, and $\tau$ is the time delay of the system to respond to an input signal. For this system, the objective is to move the finger to the object. So, to achieve that, the reference of position increases by a small value at each 10 ms until the moment when contact with the object is detected. This increment in the reference position leads to a movement almost continuous, with a velocity of almost 0.5 rad/s. During this process, the contact sensors are turned on to detect when the finger touches the object. At that point, the position controller stops and the force controller starts.

A PID controller is used to control the force applied by the index finger over the object. The actuation is done using torque, and it is possible to get feedback from the system with the force measurements in the contacts. Comparing this force measurement with the reference force will result in an error that serves as input for the PID. The controller is going to produce a new value of torque to be applied. All this process is represented in the block diagram of figure 9.

The gains of the PID controller were determined by manual tuning and can be seen in the following table.

<table>
<thead>
<tr>
<th>Gains</th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>50.0</td>
<td>5.0</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2: PID controller gains for position control.

<table>
<thead>
<tr>
<th>Gains</th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.3</td>
<td>40</td>
<td>0.002</td>
</tr>
</tbody>
</table>

Table 3: Gains of the PID controller.
the response time $\tau$ is of 5 ms. It is the time past since the message with the value of torque is sent until the change in the measurements of the force of contact can be detected.

![Figure 10: Measurements done by the tactile sensor at green and the result of the filtering with mean at blue.](image)

The measurements done by the sensors presents some noise, as can be seen in figure 10. Applying three different static values of torque as input, during one second each (0, 0.6 and 1.2 N.m) is possible to confirm that the measurements of force at the fingertip during that period are not constant and presents noise. To reduce the effects caused by the noise, filters with the mean and the mean of the last 20 samples were tested to improve the quality of the measurements taken. Since the results are similar for both filters, only the ones taken with the mean are presented in this abstract.

The PID controller has an anti-windup module to avoid the problem of the integral windup, where this term accumulates an excessive value of error. The actuation is limited between 0 and 1.2 N.m, and every time that the value of the sum of the terms goes out of these boundaries, the integral term is adapted. The torque applied is given by $T = p + i + d$, where

$$i = \begin{cases} i, & \text{if } 0 \leq p + i + d \leq 1.2 \\ p + d, & \text{if } p + i + d \leq 0 \\ 1.2 - p - d, & \text{if } p + i + d \geq 1.2 \end{cases}, \quad (6)$$

$p$, $i$ and $d$ are the proportional, integral and derivative terms computed in the controller.

With the gains of table 3 it was possible to test the controller, and get the graph of the figure 11. The left vertical axis represents the input torque applied by the finger, and the right one the force measured in the contact between the index finger last phalange and the object.

![Figure 11: Closed-loop control of the force applied by one finger.](image)

The graph of figure 11 shows no overshoot in the response of the system, a response time of 13 milliseconds and a 2% steady-state error settling time of 158 milliseconds. The max deviation in the force measured (filtered) to the reference, due to the noise in the measurements, is 6%. Figure 12 shows the response of the controller for a variable reference.

![Figure 12: Closed-loop control of the force for a variable reference value.](image)

It is possible to see that the controller is able to follow the reference and with a fast response time, and it presents some undershoot and overshoot when the reference force changes.

4. Results

The controller developed is tested for two kinds of grasp, power cylindrical grip and precision tip-to-tip grip. Then the system is tested with the introduction of noise in the pose of the hand and then changing the force references until having the grasp in the edge of the threshold force needed to hold the object. The PID gains of the hybrid force/position
controller are the ones from table 3 and from table 2 for finger’s position.

In figure 13 is represented a precision grasp. To prevent slippage, two parallel forces ($F_{Thumb}$ and $F_{Index}$) with opposite directions are applied, one by the index finger and the other by the thumb.

The graphs of figure 14 show the response of thumb and index finger controllers during a precision grasp. In the first seconds of the procedure, the object is moving, so the fingers are not able to apply the reference force without a considerable error. When the object stops moving, both controllers can follow the reference and apply the force desired.

![Figure 13: Example of a precision grasp.](image)

![Figure 14: Closed-loop control of the force applied by the fingers during a precision grasp (reference of 1.0 N in dashed, force measured (with a mean filter) in blue and torque applied in orange).](image)

(a) Force measured and (b) Force measured and torque applied in the thumb. torque applied in the index.

The graphs represent the force measured at the last phalange of the finger in blue, and the torque applied in orange. The beginning of the lines at blue and orange indicates the moment when all the fingers are touching the object and the sensors start to measure force and the DC motors to apply torque. From these graphs is possible to get the information present in table 4.

In the figure 15 is an example of a power grasp. In this kind of grasp, all the fingers are used and it is possible to carry heavier objects due to the possibility to apply more force. All the fingers apply force in similar directions and the object is pushed against the palm that exerts a normal force.

![Figure 15: Example of a power grasp.](image)

![Figure 16: Example of a power grasp.](image)

The graphs of figures 16 show the system responses. All fingers are able to follow the reference, achieving the steady-state faster than in a precision grasp, as expected, because of the constraints in the object movement created by the addition of contact points. All the times and overshoot values are in table 5.

To test the capacity of the controller to lead with the noise, a pose disturbance is added to the system. The hand grasps an object (power grasp) and lifts it. Then a noise signal is generated to have the hand shaking in all directions. It is done by applying external position and orientation disturbance signals to the floating hand at a rate of 1000 Hz. The pose disturbances are randomly generated, with a uniform distribution between -0.025 and 0.025 (meters for position and radians for orientation).

The graphs of figures 17 show the system’s response to these disturbances.
Figure 16: Closed-loop control of the force applied during a power grasp (reference of 1 N in dashed, force measured (with a mean filter) in blue and torque applied in orange).

<table>
<thead>
<tr>
<th>Finger</th>
<th>Thumb</th>
<th>Index</th>
<th>Middle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response time (s)</td>
<td>0.141</td>
<td>0.108</td>
<td>0.045</td>
</tr>
<tr>
<td>Overshoot (%)</td>
<td>22.1</td>
<td>41</td>
<td>8</td>
</tr>
<tr>
<td>Peak time (s)</td>
<td>0.404</td>
<td>0.168</td>
<td>0.162</td>
</tr>
<tr>
<td>10% Steady-state error settling time (s)</td>
<td>0.462</td>
<td>0.21</td>
<td>0.097</td>
</tr>
</tbody>
</table>

Table 5: Results obtained for each controller in a power grasp.

response to the test. In the graph, the onset of the noise signal is represented by the green vertical dashed line. It is possible to see that, after the line, the response torque starts to vary more, but the force applied by each finger still follows the reference force.

Table 6 shows the errors between the force measured and the reference force at each finger after the noise have been applied. In all the fingers, the controller is able to maintain the average error in the 10% gap. The maximum value of the error passes that gap, being the higher deviation of almost 25%.

Unless there is a part of the hand under the object, the grasp is maintained by friction forces. A power grasp is performed with a reference of 1.0 N per finger, and it is decreased by 0.1 until the object slips from the hand. It allows to see the response of the force controller to changes in the reference and use the information of all the sensors to reduce the reference forces and still having a stable grasp.

In figure 18 is represented the force measured in four of the twelve phalanges of the hand during a power grasp. The graphs of the left side have the force measured by the third phalange of the index and middle fingers, that are used in the closed-loop control. The graphs on the right side are from phalanges that do not have their measurements taken into account for the control, but are actuated by the same DC motor. The reference force is decreased by steps, and the force controller is able to follow the reference, but, looking at the graphs of the right side, is possible to verify that the total force applied by each DC motor is higher (sum of all the forces applied at each phalange).

The fingers apply force perpendicularly to the direction of the gravity, so the following condition
needs to be met
\[ F_g \leq \sum_{n=1}^{12} F_i \mu_i. \]  \( (7) \)

If the friction coefficient is the same in all contacts it is possible to get
\[ F_g \leq \mu \sum_{n=1}^{12} F_i \Leftrightarrow \frac{F_g}{\mu} \leq \sum_{n=1}^{12} F_i, \]  \( (8) \)

where the sum of the forces measured by the fingers (3 sensors per each one of the 4 fingers) determines the ability to grasp an object.

In figure 19 is represented the sum of the forces measured during the power grasp of the figure 18. The sum of the forces reduces its value with the decreasing of the individual references. The green line represents the time when the object starts sliding from the hand. At that moment the difference between the force measured and the theoretical threshold force (horizontal orange dashed line) is 0.098 N. At that point, the reference force is 0.3 N per finger.

This reduction in the reference force leads to a reduction in the applied torque. In table 7 is shown the comparison between the torque when the reference changes from 1.0 N to 0.8 N. Both assure a stable grasp, but with a lower reference, it is possible to reduce the effort of the DC motor and reduce the energy consumption, without losing quality on the grasp.

5. Conclusions
5.1. Achievements

The major achievement of the present work is the development of a tactile feedback closed-loop controller to the Vizzy robot. A simulation of the hand was designed to be as close as possible to the real robot. Throughout a modelling and identification process, we determined the hand finger parameters, that were used to develop a force control-loop per finger. Those individual controllers were able to follow the references set during the tests for tuning of the parameters, as well as during the grasps.

The reaching to the object was done via user interface commands or spawning the hand in the desired pose. The grasp was performed with a hybrid control system. In the first part, the fingers move based on position control with the information...
from the encoders. After a contact being detected, the force that is being applied to the object by the finger is controlled based on tactile force feedback. The actuation is made with torque in both phases of the control. This process was done for a hand with underactuated fingers and poor position information, just with the feedback from the encoders in the DC motors and the tactile sensors.

Grasp stability was tested decreasing the reference force applied over the object and by adding noise to the reference pose to simulate hand shake. With the information of all sensors available in the hand, it was possible to evaluate the gap between the force applied in the grasp and the theoretical force that needs to be applied to maintain the grasp for that specific object.

6. System Limitations and Future Work

Despite we achieved good results in a simulated environment, this work does not present any result in the real robot. Due to the lack of these tests, it is not possible to assure that the results obtained in the simulator will work in the real world.

Another limitation is the presence of noise of significant magnitude in the tactile sensors. Due to this problem, a low-pass band filter has to be used, which reduces the bandwidth of the system.

In future works the system developed could be complemented with a high-level controller that chooses the references and the contact points. It also can be used in the planning stages. Similar to the work done by Vahrenkamp in [15] the simulator can be used to try a set of grasps and pick the one that has the best grasp stability.

References


