Control in effort of a robotic hand

2017 / 2018

Veronica Elena

16/04/2018

PI controller

Introduction

The objective of this report is to present the implementation of a Proportional-Integral (PI) effort control on the Alpes Instruments' robotic hand. It is a five-fingered robotic hand with 15 degrees of freedom, but only six actuators. Since the number of degrees of freedom is greater than the number of the actuators, the robot is an underactuated one, and the passive joints are driven by a system of pulleys and cables.
The hand communicates with the computer via a serial link of COM port type. In particular, the communication between the PC and the hand is based on readings and writings of registers.
The robotic hand is controlled from the computer. It can operate following two distinct modes: the position mode and the tension one. In the former case the desired position of the encoder, positioned in correspondence of each motor, is passed via the proper register to the robot and a PID position control is performed. Using the latter mode, the user acts directly on the tension applied on the motor. This is the mode that will be used to control the hand through a force feedback.
It is important to notice that the robotic hand takes as input the tension with a value between [-1150, +1150], which is equivalent to $\pm 11,50$ Volts. The bound is given by the motor specification, indeed the motor has a maximum applicable tension of 12 Volts. Moreover, motors have a current control to avoid tension values which are too high. Thus it is possible to limit the default limit current value (750) for safety reasons, writings on the corresponding register.

In order to implement a force control, the applied force of the robotic finger on a rigid and fixed object should be measured. Therefore a FlexiForce sensor is attached on the fingertip of the considered finger. This sensor has been chosen because it is very thin, flexible and easy to manage, but it is not robust to noise and it is able to measure the normal component of the force only.
The force sensor is not the only sensor of the robot. Indeed, an encoder sensor is present in correspondence of each DC motor.

In the following the PI controller is introduced and some experimental results are presented in order to find out limitation and advantages of this control strategy.

Chapter 1

Design of the control

In the previous chapter is introduced the Alpes instruments' robotic hand. From the control point of view, the robot is the system which is necessary to control. It takes as input the voltage value and it returns as output the encoder position and the applied force of each finger on a certain rigid object.

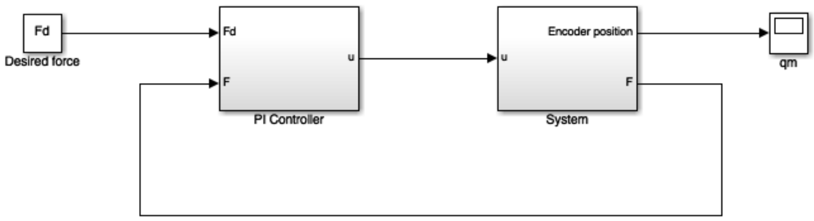

Figure 1.1: PI control scheme.

In (Figure 1.1) is shown the control scheme. The input signal of the system (u) is the output of the control and it has the form:

$$u = K_p(F_d - F(t)) + K_i \int (F_d - F(\tau))d\tau, \tag{1.1}$$

where K_p and K_i are the proportional and integral gains respectively and $e = (F_d - F(t))$ is the control error.
The feedback signal is the measured force. It comes from a FlexiForce sensor which measures the applied force of the finger on a rigid object when contact occurs. The sensor is positioned on the fingertip and it only measures the normal component of the force. For this reason a proper configuration of the finger in contact position is necessary during the control application. It means that the sensor has to perfectly contact the object surface and it should be placed in an horizontal position with respect to the surface itself.

The input voltage is applied to the system via the following Matlab function:

```
mode_tensionDoigt (Voltage, Limit_Current, Finger, Serial_Port).
```

It writes on the register 0, 1 and 2 in order to set the Tension Mode, the tension value and the current value respectively. These last two values are passed to the function with the integer number corresponding to the considered finger, the serial port associated to the Matlab object representing the robotic hand.

The integral term of the controller is identified using the Euler's method. It is recursive numerical integration based on the past contribute of the integral and the current value of the evaluated function (the control error in this case). The step value (h) is chosen equal to the sampling period of the force measurements.

$$\begin{aligned} y'(t) &= e(t, y(t)), \qquad\qquad y(t_0) = y_0 \\ y_{n+1} &= y_n + h \cdot e(t_n, y_n) \end{aligned} \tag{1.2}$$

In the following, the control gains are identified and the remaining error is evaluated using a tuning procedures. It is the well known Ziegler-Nichols method and it has been chosen because it represents a fast and easy way to tune the control gains. Clearly it is not a precise and definitive procedure, thus the results should be finely adapt to the physical behaviour of the system.

Chapter 2

Tuning the gains

In this section, the tuning procedure of the gains is presented. A specific configuration of the index finger is fixed as "contact configuration" as in (Figure 2.1).
The chosen method is the so called Ziegler-Nichols and it is applied considering the open loop system. A step function is applied as input of the system, the system response is acquired and the tangent line to the curve is computed. From its slope (m) and its intersection with the x-axis (T_r) the gain values are calculated.

Figure 2.1: Tuning procedure configuration.

In the following, the method is better explained and the experimental results are presented. Thanks to this method, the gain values are found out and they are applied to the controller in order to evaluate the its performance in terms of stability and static error.

2.1 Ziegler-Nichols method

Ziegler and Nichols conduced numerous experiments and proposed rules for determining the values of the PI gains. In the following the first method is applied to the robotic hand system considering the open loop response when a step function is applied as input. Once the system response is acquired, the tangent line at the inflection point is computed and its slope (m) and its intersection with the x-axis (T_r) allow to compute the gain values (Figure 2.2):

$$K_p = 0.9mT_r \quad T_i = 3.3T_r. \tag{2.1}$$

When this method is applied, the following control law is considered:

$$u = K_p e(t) + \frac{K_p}{T_i} \int e(\tau) d\tau, \tag{2.2}$$

where $e(t) = F_d - F(t)$.

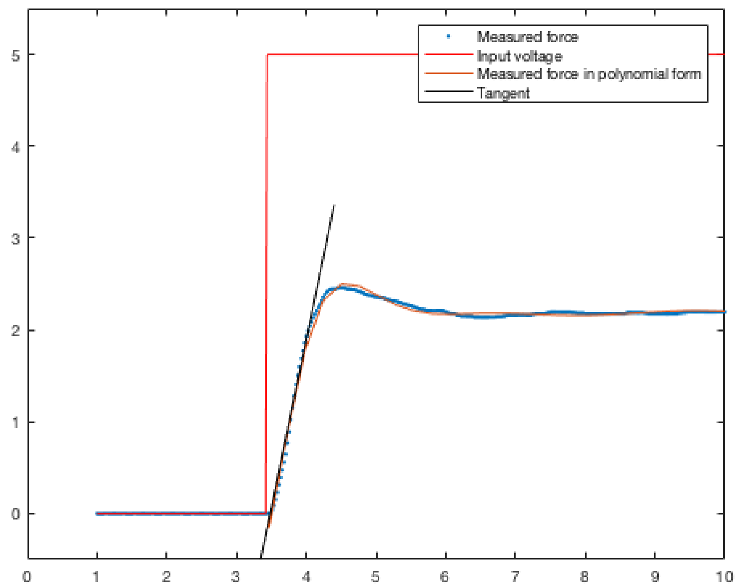

Figure 2.2: Response of the system to a step input.

2.1.1 Experimental results

In order to compute the value of K_p and K_i, the amplitude of the step function corresponding to the voltage applied to the motor is increased trial after trial and the corresponding system response is acquired. Identifying the slope and the intersection with the x-axis of the tangent line to the response curve in its inflection point and applying Equation 2.1, the proportional and integral gain values are defined for each input step. The implemented Matlab function is named "ZieglerNichols_method.m" and it takes as input the step amplitude and returns directly the computed values of the gains.

In (Figure 2.3) the values of the gains are shown and in (Figure 2.4) the numerical values are reported. It results that the mean value for K_p and K_i and 9.4072 and 0.5273 respectively. It is important to notice that these values have to be adapted to the close loop behaviour of the physical system.

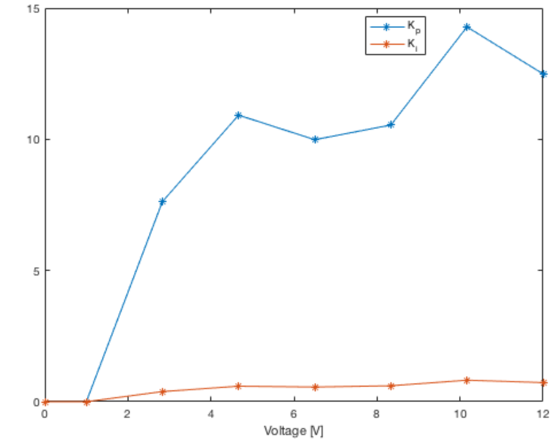

Figure 2.3: Computed gain values increasing the applied voltage.

Volts	Kp	Ki
1	0	0
2.8333	7.6286	0.38615
4.6667	10.92	0.59057
6.5	9.9849	0.55891
8.3333	10.544	0.60679
10.167	14.283	0.81962
12	12.489	0.72872

Figure 2.4: Numerical values of the gains for increasing amplitude of the applied voltage.

Once the gains are identified and used in the control law, the closed loop system is studied (”PI_controller.m”).
The error evolution is recorded for increasing constant values of the reference force. In particular the desired force is applied to the closed loop system form 0.1 to 5 N. This interval has been identified from the potential payload of one hand (about 500 g), but it is bigger than the real one.
Table 2.7 reports the numerical values of the static error, while (Figure 2.5 and 2.6) show the graphical results. The static error is calculated evaluating the time evolution of the error and computing the mean value and the standard deviation of the Gaussian distribution fitting with the error data. It can be noticed that the remaining error value increases for high reference forces. In particular the mean value of the static error is in the order of 10^{-1} N till 2 N are applied as desired force, while for higher reference force values it increases to 1 N. In order to improve the controller performance it is possible to increase the gain values. Setting $K_p = 11$ and $Ki = 0.7$ the results in Table 2.8 are obtained.
However, these results clearly show that the maximum applicable force of the robotic finger ensuring good performance is 2.5 N.

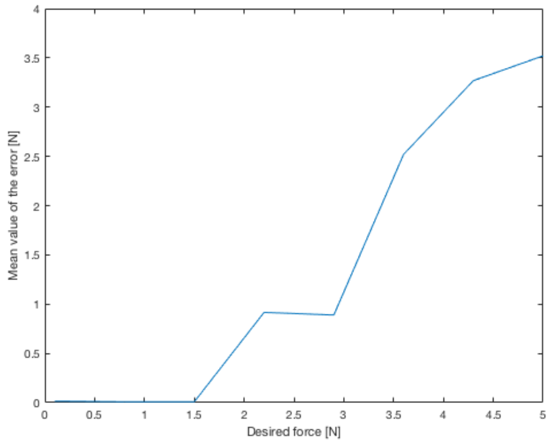

Figure 2.5: Mean value of the remaining error in function of the reference constant force.

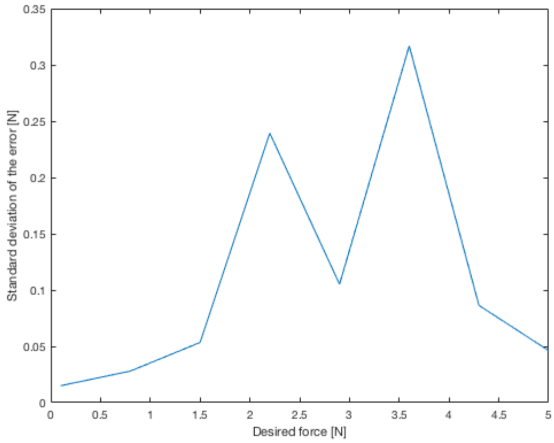

Figure 2.6: standard deviation of the remaining error in function of the reference constant force.

Force	Mean_Value	Standard_Deviation
0.1	0.015266	0.015045
0.8	0.0086598	0.028061
1.5	0.0086859	0.053497
2.2	0.91643	0.23931
2.9	0.88963	0.10519
3.6	2.5204	0.31679
4.3	3.2691	0.086387
5	3.5207	0.046221

Figure 2.7: Numerical values of the remaining error with $K_p = 9.4072, Ki = 0.5273$.

Force	Mean_Value	Standard_Deviation
0.1	0.00090929	0.014055
0.8	0.0046058	0.013259
1.5	0.084768	0.11805
2.2	0.012281	0.071783
2.9	1.0849	0.24631
3.6	1.5182	0.069087
4.3	2.3409	0.064704
5	1.9519	0.27572

Figure 2.8: Numerical values of the remaining error with $K_p = 11, Ki = 0.7$.

A second test is performed applying a variable reference force. The evolutions of the error and of the measured force over time are shown in (Figure 2.9 and 2.10) when a variable reference force is considered: $F_d = 1 + \sin(\alpha)$. The sinusoidal wave has been chosen to illustrate the repeatability of the experiments, as well as the similar behaviour during increase and decrease of the reference command.
It can be noticed that the measured force is able to follow quite well the reference curve in red and the static error value is (0.06804 ± 0.02642). Moreover, the static error is bigger when the reference force is close to 0 and 2 N. Also this result leads to conclude that the finger is able to

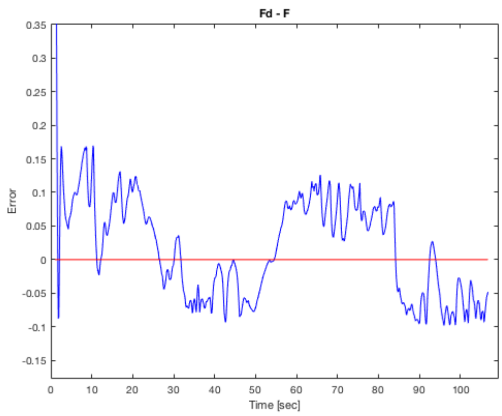

Figure 2.9: Error evolution over time applying $F_d = 1 + \sin(\alpha)$.

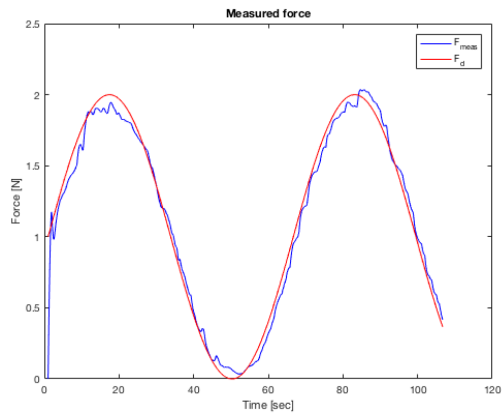

Figure 2.10: Measured force evolution over time applying $F_d = 1 + \sin(\alpha)$.

apply up to 2 N reaching good performance in terms of static error. However, the force upper bound can vary depending on the finger configuration with respect to the object surface. In particular if the fingertip, where the sensor is attached, is not perfectly parallel to the rigid surface, only the normal component of the force is measured and it is returned as feedback to the control which will read a smaller force than the real one and it will increase it. This is the main origin of the error.

Conclusions

In this report, the PI control on the Alpes Instruments' robotic hand is presented. The system takes as input the voltage applied directly to the DC motor and it returns as output the applied force of one considered finger on a rigid object. The force is measured via a FlexiForce sensor attached to the fingertip. The applied force is the feedback control.
The gain values if the controller are tuned using the Niiegler-Nichols method. It computes the gains studying the step response of the open loop system.

In general, the static error is in the order of 10^{-2}. This value is reasonable since the force sensor is not robust to the noise and it is not the the most suitable sensor in terms of precision. However, for high reference force values the error increases and it is no more acceptable. This result brings to conclude that the finger is able to apply till 2.5 N ensuring good performance of the PI controller.
For the experimental results come out that the performances of the control highly depend on the chosen contact configuration, the set-up and the relative sensor position with respect to the object surface.
Since the control has a force feedback, it is necessary that the robotic finger is in contact with a rigid object. For this reason a contact configuration of the finger is chosen and it has to ensure that the force sensor is perfectly in contact with the object surface and, in the ideal case, in horizontal position to the ground such that the applied force has only the normal component. Moreover, the used set-up has not to move when some force is applied by the finger.

As said before the static error is reasonable in the reference force interval of [0, 2.5] N. However the some oscillations are still present and they could lead to poor grasping procedure. Thus, another control strategy which keeps into account the dynamic model of the robotic hand will be implemented.