

Research on the Position Control of a 1-DoF Set-Up Powered by Pneumatic Muscles

A. Pujana-Arrese, A. Mendizabal, J. Arenas, S. Riaño, and J. Landaluze

Department of Control Engineering and Power Electronics
IKERLAN Technological Research Center

Arrasate/Mondragon, The Basque Country (Spain)
APujana@ikerlan.es

Abstract—A one-degree-of-freedom set-up driven by pneumatic muscles was designed and built in order to research the applicability of pneumatic artificial muscles in industrial applications, especially in wearable robots such as exoskeletons. The experimental set-up is very non-linear and very difficult to control properly. As a reference, an enhanced PID controller was designed. At the same time, a robust controller H_∞ and a sliding-mode controller based on an observer were designed and implemented. After that, a new position controller based on an internal pressure loop for each pneumatic muscle was tuned up. Firstly, this paper presents the experimental set-up and the system's linear models. After that, it summarizes the enhanced PID controller, H_∞ controller and the sliding-mode controller that have been designed. Then, it focuses on the position controller based on the internal pressure loops. Finally, the controllers are compared by means of experimental results.

Keywords-pneumatic muscle; robotic arm; position control; pressure control; sliding-mode control; PID control; robust control

I. INTRODUCTION

In recent last years, the Ikerlan research centre has been working on the design of an upper limb IAD (Intelligent Assist Device) [1][2], a wearable exoskeleton for helping the user perform a routine activity in the workplace. One of the design specifications considered was to use, if possible, non-conventional actuators. Of the various alternatives, pneumatic artificial muscles (or McKibben muscles) are of particular interest. For the purpose of researching their applicability in this kind of application, an experimental one-DoF set-up powered by pneumatic muscles manufactured by Festo was designed and built. Initially, a pneumatic muscle model was developed in the Modelica modelling language [3], and on this basis the model of the whole set-up was obtained in *Dymola/Modelica* and validated experimentally [4]. Linear models were then obtained. The experimental set-up was very non-linear and very difficult to control properly. Owing to the fact that the results obtained with a classical PI controller were not good, other advanced control techniques were applied. Firstly, a PID-based controller was enhanced with linear and non-linear internal loops. However, good performance requires the use of robust or non-linear control techniques [5][6] and in this context, the application of different control techniques is found in the literature. Therefore, a robust linear control

technique H_∞ [7], and a robust non-linear technique, sliding-mode [8], were applied.

Despite the fact that the results achieved with these controllers are quite good, owing to the non-linearity of the set-up and air pneumatic muscles themselves, the performance level is not identical throughout the displacement range. In order to use such algorithms in an exoskeleton-type practical application, controllers should be tuned for different areas of the operating range and a gain-scheduling strategy then implemented.

Subsequently, based on an idea applied in work by Caldwell and Tsagarakis [9], a new position controller was developed based on an internal pressure loop for each muscle. This new position algorithm requires the use of one servo-valve for each pneumatic muscle instead of one single valve for each DoF, as used with the algorithms that were designed and implemented previously.

The main aim of this paper is to present this position control algorithm based on the internal pressure loops, comparing its characteristics with the results achieved using the robust controllers that were initially developed. After describing briefly the experimental set-up, this paper goes on to explain the control objectives, the system modelling and the nominal linear models used in the controller design processes. Then, the different position controllers designed are presented: the previously designed robust controllers are summarized and the new controller based on internal pressure loops is described in detail. Finally, the paper concludes by presenting some experimental results.

II. DESCRIPTION OF THE EXPERIMENTAL SET-UP

A real picture of the set-up that was built can be seen in Fig. 1, where two *DMSP-20-200N* pneumatic muscles manufactured by Festo perform the role of the actuator. It consists of a robotic arm with a displacement of around 60° and a maximum mass of 8 kg to be moved at the tip. The arm mass is 0.987 kg and, considering that the arm is in the horizontal position, the centre of the arm mass with regard to the centre of rotation is at a height of 17.6 mm and at a horizontal distance of 205 mm. The additional masses are placed on the end of the arm at a horizontal length of 367 mm from the centre of rotation. The set-up may be rotated so that the arm moves along a horizontal plane and the effects of gravity are therefore

The material used in this paper was partly supported by the Spanish Ministry of Education and Science and European FEDER Fund (research project DPI2006-14928-C02-01).

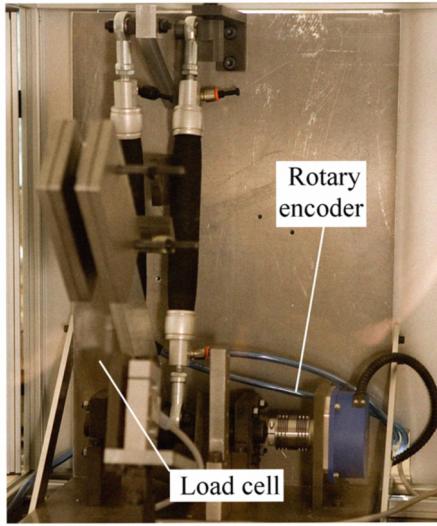


Figure 1. Picture of the experimental set-up

cancelled out. The prototype includes a *FAGOR S-D90* encoder, which supplies 180,000 pulses per turn, and a load cell on the lower stop of the model. A more detailed description can be found in [3].

The schematic diagram of the set-up, which includes the control hardware, sensors and pneumatic circuit, is shown in Fig. 2. As the figure shows, a *Festo MPYE-5-1/8HF* pneumatic servo-valve is used for actuation as standard. It is capable of operating the two pneumatic muscles. However, as mentioned earlier, with the new position control algorithm based on internal pressure loops, two servo-valves are used, each linked to one pneumatic muscle and controlled independently by the controller. The controller hardware is *PIP8*, an industrial PC made by the company MPL, which is very similar to The MathWorks' *xPCTargetBox*. A PC104 card (*Sensoray model 526*) was incorporated into the *PIP8* in order to read and write all the system signals. Control algorithms were implemented in *Simulink* and code was generated and downloaded in the aforementioned hardware by means of two of The MathWorks' tools: *RTW* and *xPCTarget*.

III. CONTROL OBJECTIVES AND SYSTEM MODELS

One of the main objectives of the prototype is to emulate the movement of the forearm orthosis, analysing the adaptation of the pneumatic muscles in such applications. Assuming that the hand with the orthosis should be able to transport a determined weight, for the design of the controllers the nominal conditions were considered as when the robotic arm has a load of 3 kg at the tip. Furthermore, the load may be reduced or increased by up to 6 kg. The specifications therefore presume that the system must be robust for this load interval.

In a human arm orthosis (an end application considered for pneumatic muscles), the set-point position is generated by the user's intention of movement; it is not a path pre-set by a controller. On tuning the controller, small jumps in position of 10° have therefore been used. The performance specifications are that the response to these jumps must be as quick as possible with very little overshoot and no vibration.

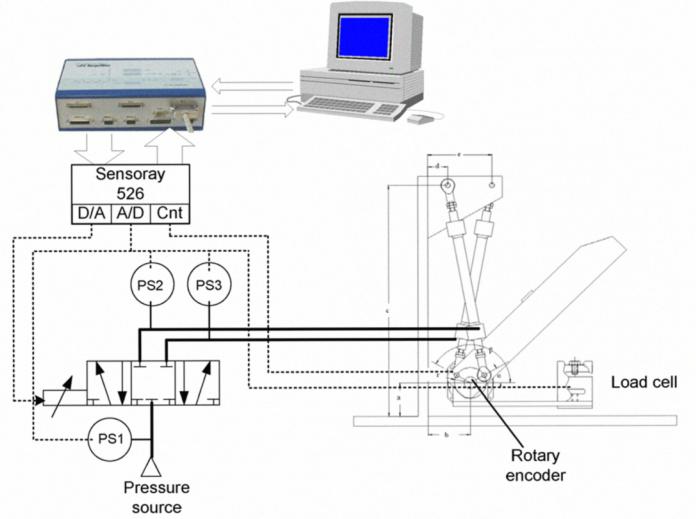


Figure 2. Schematic diagram of the set-up and pneumatic circuit

A full non-linear model of the experimentally validated prototype had previously been developed in *Dymola/Modelica* [3]. On the basis of this model, reduced linear models for use in designing controllers were deduced. The nominal model is taken as having a weight of 3 kg at the tip of the arm. In the standard case, using only one servo-valve (Fig. 2), the system is considered as a SISO system, where the input is the input voltage to the servo-valve and the output is the angle position of the robotic arm. The resulting transfer function of the nominal model is as follows:

$$G_{3N}(s) = \frac{0.2134s^2 + 30.938s + 175.051}{0.074489s^3 + 0.4259s^2 + 9.36255s + 0.003264}$$

As extreme cases in the considered working range, a load of 6 kg is considered at the tip of the arm, with the following reduced linear model:

$$G_6(s) = \frac{20.5694s + 63.54}{0.00842s^4 + 0.0889s^3 + 1.015s^2 + 5.263s - 0.00497}$$

and the case of no weight at the tip, where the linear model is:

$$G_0(s) = \frac{0.000845s^3 + 0.3302s^2 + 61.74311s + 142.5554}{0.0000312s^5 + 0.001527s^4 + 0.058s^3 + 1.9637s^2 + 5.9797s - 0.00023}$$

Fig. 3 shows these plants in the frequency domain. As can be seen, the models show significant resonance. When the load is increased, the resonance frequency is reduced, together with the gain. The differences between the linear models are considered as nominal model uncertainties. To obtain these linear models, the effect of gravity has not been taken into account, as the arm is supposed to move along the horizontal plane. Furthermore, these linear models do not reflect the significant influence of friction, which is very different throughout the robotic arm's range of displacement. These

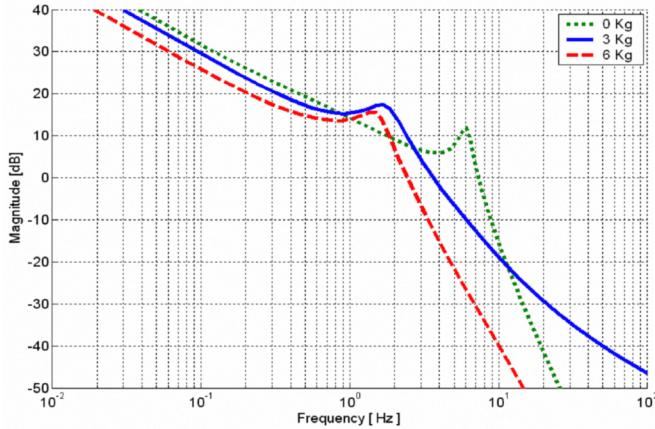


Figure 3. Linear transfer functions for 0, 3 and 6 kg

linear models were used to design the robust position controllers for the standard configuration with a servo-valve. For the new control algorithm design, which requires two servo-valves, analytical models were not used.

IV. POSITION CONTROLLERS

A. Enhanced PID controller

Initially, as a reference, a classic PI controller was tuned up. After that, the structure was completed with other internal linear and non-linear loops. Fig. 4 shows the structure of the most elaborate controller that was built. The position PID was complemented with a speed loop and an acceleration loop. A speed feedforward was also added to improve the dynamic response. A more internal non-linear loop partially compensated the effects of gravity. For the basic PI controller, the gains were adjusted to the values $K_p=0.062$ and $K_i=0.029$. For the case of the enhanced PID, the tuned gains were $K_p=8.5$, $K_i=0.009$, $K_d=0.8$, $K_{vp}=0.02$, $K_{vi}=0.005$, $K_{ff}=0.0025$ and $K_a=0.0002$.

B. H_∞ controller

The H_∞ problem can be defined as that of finding a controller K so that the value of the norm ∞ of a characteristic vector of the system being controlled is minimised below the unit:

$$\begin{bmatrix} W_1 \cdot S \\ W_2 \cdot R \\ W_3 \cdot T \end{bmatrix} \leq 1 \quad (1)$$

where the weight functions W_1 , W_2 and W_3 set the limits for the different specifications required in the frequency domain from the controlled system. The functions S , R and T , designated *sensitivity functions*, show the real performance of the system and together form the *mixed sensitivity* vector.

The design process of the H_∞ controller for the standard configuration of the experimental set-up can be found in [7]. When the mixed sensitivity vector (1) was minimised, the following controller $K(s)$ was obtained:

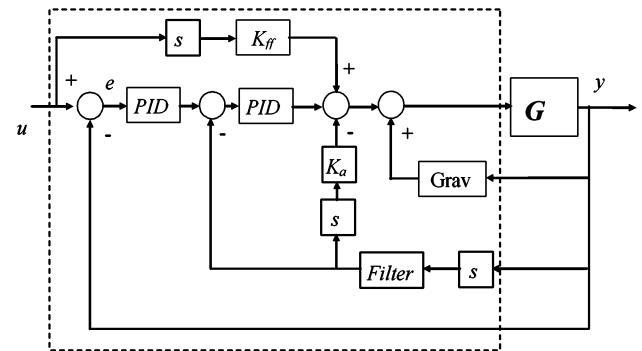


Figure 4. Structure of the enhanced PID based control

$$K(s) = \frac{15.51 s^7 + 782.7 s^6 + 3.001e4 s^5 + 1.021e6 s^4 +}{s^8 + 124.3 s^7 + 8263 s^6 + 3.363e5 s^5 + 6.246e6 s^4 +} \\ \frac{4.511e6 s^3 + 6.01e6 s^2 + 4.425e6 s + 170.1}{2.041e7 s^3 + 2.43e7 s^2 + 1.574e7 s + 0.006295}$$

C. Sliding-mode controller

To develop the sliding-mode controller [8], work by Edwards and Spurgeon [10] was taken into account. A tracking requirement was incorporated using a comprehensive action approach. From the initially considered system,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \quad (2)$$

which is assumed to be square, where A and B are the matrices representing the nominal system and $x(t)$ is the state vector, an integral action is introduced by means of an additional state:

$$\dot{x}_r(t) = r(t) - y(t) \quad (3)$$

where $r(t)$ represents the set-point value and $y(t)$ is the system output.

The sliding-mode control is based on taking the system to a surface where the closed loop dynamics will be governed by the equations that are established, free from any unmodelled disturbances. In this way, bearing in mind the incorporation of a tracking requirement, the sliding surface is defined by:

$$S = \left\{ \tilde{x} \in \mathbb{R}^{n+p} : S\tilde{x} = S_r r \right\}; \quad S = [S_1 \ S_2] \quad (4)$$

where S and S_r are design parameters governing the movement dynamics. S_1 will be of dimension n , and S_2 of dimension p .

To establish the sliding surface S , the system's uncertainties must be taken into account, estimated through the nominal models that are presented. Once S has been set, the control signal for designing a sliding-mode controller including an integral action approach and tracking requirement is:

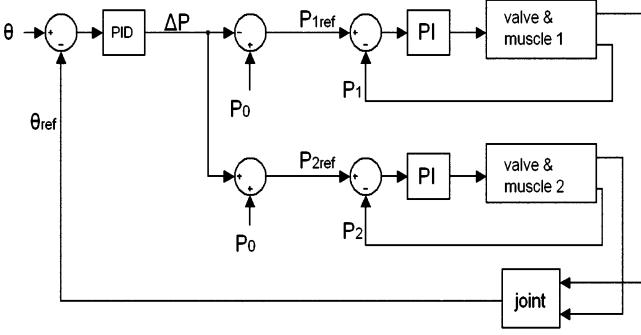


Figure 5. Control diagram of the internal pressure loops based algorithm

$$u = u_L(\tilde{x}, r) + u_N \quad (5)$$

where the linear component is:

$$u_L(\tilde{x}, r) = L\tilde{x} + L_r r + L_{\dot{r}}\dot{r} \quad (6)$$

$$\begin{aligned} L &= -[SB]^{-1}(S\tilde{A} - \phi S); \quad L_r = -[SB]^{-1}(\phi S_r + S_2 M) \\ L_{\dot{r}} &= [SB]^{-1}S_r \end{aligned} \quad (7)$$

and where S_r and ϕ are design matrices.

The non-linear component of the control law (5) will be a function of the surface S multiplied by a gain K established in the design process.

$$u_N = -K \cdot \text{sgn}(S\tilde{x}) \quad (8)$$

To design the controller described above the state vector should be measured, which is not possible in the experimental set-up under study. Therefore, an observer was designed for its estimation. In a linear system such as (2), the observer can be expressed by the formula:

$$\dot{\hat{x}}(t) = Ax(t) + Bu(t) - G(C\hat{x}(t) - y(t)) + FBV \quad (9)$$

where \hat{x} represents the state vector estimated. In this way, the observer takes the form of a system model, which is impelled by the misalignment between the plant output and the observer output. The linear gain G is calculated as follows:

$$G = T_0^{-1} \begin{bmatrix} A_{12} \\ A_{22} - A_{22}^s \end{bmatrix} \quad (10)$$

where T_0 is the matrix representing the change of coordinates between the given system and its canonical form and A_{22}^s is a

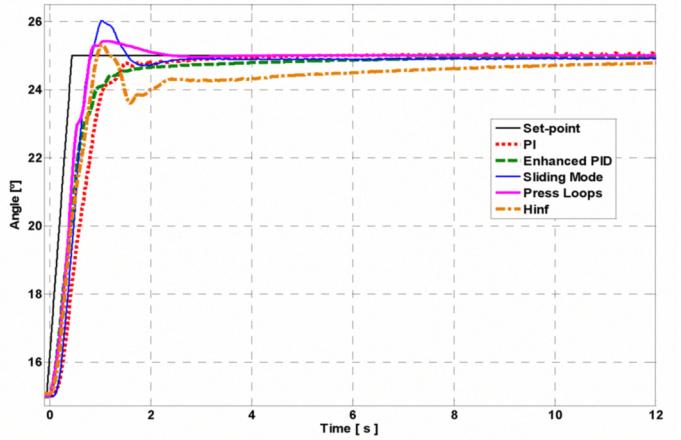


Figure 6. Experimental results in the upper displacement zone

diagonal matrix established in the design. v is the sign function of $C\hat{x}(t) - y(t)$ multiplied by a design constant K . The matrix F is calculated by means of $F = B^T P$, where P is the solution to the Lyapunov equation:

$$A_{22}^s P + P A_{22}^{s\ T} + I = 0 \quad (11)$$

As a summary, the controller design and implementation process was as follows: on the basis of the nominal model G_{3N} , the first step was to convert the system given by a transfer function into its expression by means of the state space, obtaining the A , B and C matrices required for the design (see Appendix). Once the matrices that defined the system were obtained, the next step was to design the observer to obtain the matrices G and F . To do this, the value A_{22}^s had to be established. This was done by trial and error, setting the desired output estimation error pole. After designing the observer, the sliding-mode controller was designed in accordance with the equations described. The desired λ poles had to be set for the closed loop system, which provides the system robustness. The rest of the design matrices (see Appendix) were set experimentally to obtain good response performance. A more detailed description of the design process can be seen in [8]. The design matrices used in the controller appear in the Appendix.

D. Internal pressure loops based controller

One different approach when controlling an actuated joint using a pair of opposing muscles consists of independently controlling the pressure of each muscle [9]. Of course, this requires the incorporation of one valve for each muscle on the device, assuming the cost this involves.

Despite the fact that this new solution initially doubles the variables that have to be controlled for each degree of freedom, it can be considered as a single-variable approach for each joint. Based on the symmetrical co-contraction of the opposing muscles, an asymmetrical variation is set in the pressure of each muscle. Thus, based on an initial pressure (P_0) the setting

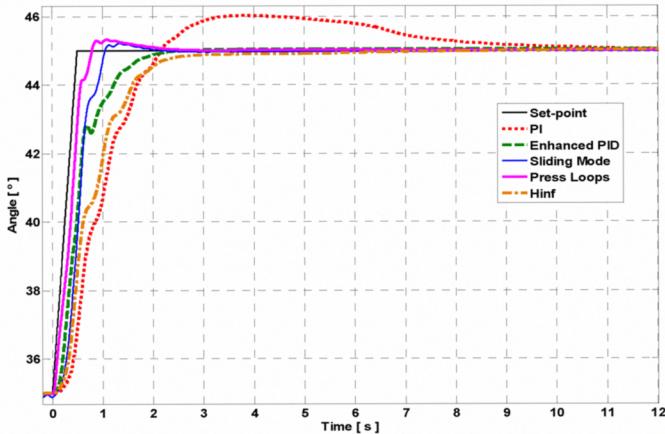


Figure 7. Experimental results in the middle displacement zone

is increased in one of the muscles and reduced by the same amount (ΔP) in the other.

$$P_{1\text{ref}} = P_0 - \Delta P; \quad P_{2\text{ref}} = P_0 + \Delta P \quad (12)$$

Accordingly, from the control point of view, the system is still SISO with the angular position of the joint (θ) as the output and the pressure variation (ΔP) as the input.

Special mention must be made of the relevance of the parameter P_0 in the rigidity or impedance of the joint. Indeed, it could be designed as a variable parameter whose value changes depending on certain environment conditions.

Fig. 5 shows the full control schematic based on the internal loops that control the pressure in each muscle, implemented by means of PI algorithms. As it has been already mentioned, the pressure set-point for each controller is set on the basis of an initial value (P_0), adding and subtracting the same quantity (ΔP). The value of this increase/reduction is the output of the most external loop of the controller (the position loop). This loop has also been implemented by means of a PID algorithm. The gains of both pressure loops were adjusted to the values $K_p=4$, $K_i=4$, and the gains of the position loop to $K_p=0.21$, $K_i=1.2$, $K_d=0.04$, being $P_0=3$ bar.

V. EXPERIMENTAL RESULTS

Initially, all the controllers were tuned and tested in simulation, using the non-linear model developed in Modelica and the resulting linear models [3][4]. The controllers, discretized with a sampling time of 2 ms, were then included in the actual set-up. In order to compare the performance levels of the various control algorithms Fig. 6, 7 and 8 show the experimental responses to a ramp input of 10° and a slope of $20^\circ/\text{s}$ applied in three different areas of the displacement range, being the mass at the tip the nominal of 3 kg.

Analysing the system response in the first displacement area (shown in Fig. 6), with the PI and enhanced PID controllers, there is no overshoot; however, the settling time is quite high. Of the rest, the sliding-mode is the algorithm with the biggest overshoot and the controller H_∞ takes long time

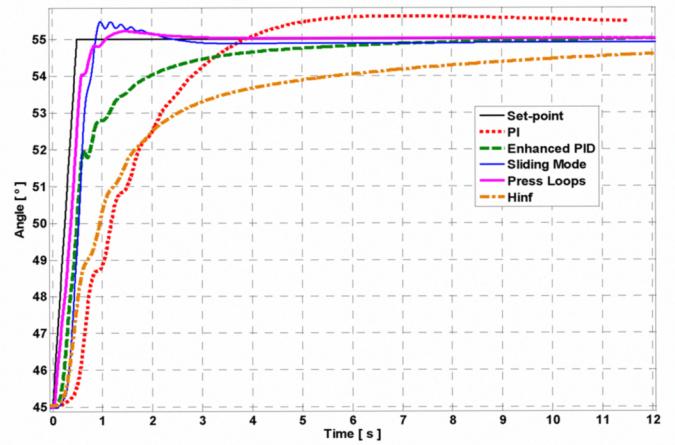


Figure 8. Experimental results in the lower displacement zone

eliminate the steady-state error. The algorithm based on the pressure loops takes the arm to 25° before any other.

Fig. 7 corresponds to a jump in the intermediate zone and shows that the PI controller involves an overshoot and an excessive settling time, whereas the response of the enhanced PID follows the set-point until a small vibration appears. The controller H_∞ has an initial behaviour that is similar to the PI, even though it has no overshoot and the steady-state error is compensated. The response of the sliding-mode controller increases quickly and, despite a slight overshoot, it is capable of eliminating the steady-state error. It can be concluded that the response of the controller based on the pressure loops is clearly the best since, despite an overpass equal to the case of the sliding-mode, its settling time is lower.

In the lower zone (Fig. 8), besides increasing the overshoot and the slowness of the previous zones, the position controlled by the PI begins to oscillate. The performance levels of the enhanced PID also fall because its settling time is increased. The response of the algorithm H_∞ also worsens and becomes much slower. The sliding-mode controller shows an overshoot somewhat greater and there is a certain amount of oscillation. Once again, the controller based on the pressure loops offers the best performance levels, obtaining a faster system, with smaller overshoot and without steady-state error.

Fig. 9 and 10 show the angular position of the arm with different weights on the tip in response to a ramp input of 40° amplitude and 2 s duration for the two controllers with the best dynamic performance levels: the sliding-mode controller and the controller based on internal pressure loops. Despite the fact that the transient response of the system controlled by the sliding-mode algorithm (Fig. 9) is similar for all the weights, there are small differences at the overshoot level. It is sufficient to look at Fig. 10 to see that the controller based on the pressure loops is so robust with regard to the load variation that the figure shows hardly any differences between the results of the different weights.

VI. CONCLUSIONS

An analysis has been made of the position control of a 1-DoF mechatronic device actuated by an opposing couple of pneumatic muscles. The non-linear nature of these actuators and of the set-up built itself means that the experimental set-up

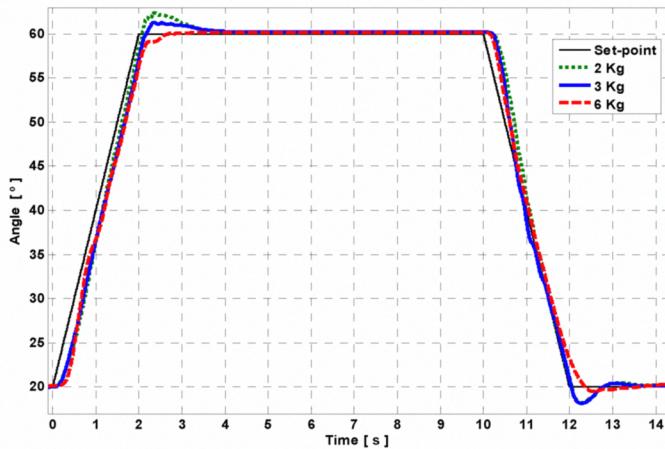


Figure 9. Experimental results with different tip masses for sliding-mode controller

is very difficult to control, since the system response varies across the movement range. Furthermore, the possibility of positioning additional weight on the tip of the arm forces to consider the controller robustness for a wide range of weights.

The paper presents various control solutions and compares the results obtained. Four of the controllers implemented (PI, enhanced PID, H_∞ and sliding-mode) work with one single servo-valve. The results obtained with the classic PI and enhanced PID controllers vary greatly from one zone to another in the displacement range, as well as with the variation of the mass at the tip. The H_∞ controller was designed to achieve a certain level of robustness with regard to the weight of the load. However, its response is not the same across the movement range. In order to achieve a level of control capable of working correctly across the range the sliding-mode technique was implemented. The results presented illustrate this decision. Furthermore, the change of the load does not seem to affect the response to any great extent. This does not mean that control algorithms such as the enhanced PID or the H_∞ are not valid, but they would have to be designed or tuned in different operating zones and a gain-scheduling strategy would have to be applied.

Finally, despite the change in the pneumatic circuit, a control algorithm was designed and implemented based on the independent control of the pressure of each muscle. The results obtained are the best of the comparison with regard to performance levels and for compensating the nonlinearity of the prototype. In addition, the study of the robustness in comparison with the load also offers very good results.

REFERENCES

- [1] F. Martinez, I. Retolaza, E. Lecue, J. Basurko, and J. Landaluze, "Preliminary design of an upper limb IAD (Intelligent Assist Device)", *9th European Conference for the Advancement of Assistive Technology, AAATE 2007, October 3rd-5th, Donostia*.
- [2] F. Martinez, I. Retolaza, A. Pujana-Arrese, A. Cenitagoza, J. Basurko, and J. Landaluze, "Design of a Five Actuated DoF Upper Limb Exoskeleton Oriented to Workplace Help", *IEEE International Conference on Biomedical Robotics and Biomechatronics, IEEE BioRob 2008, 19-22 October 2008, Scottsdale, Arizona, U.S.A.*
- [3] A. Pujana-Arrese, J. Arenas, I. Retolaza, A. Martinez-Esnaola, and J. Landaluze, "Modelling in Modelica of a pneumatic muscle: Application to model an experimental set-up", *21st European Conference on Modelling and Simulation ECMS2007, 4-6 June 2007, Prague*.
- [4] A. Pujana-Arrese, J. Arenas, L. Maruri, A. Martinez-Esnaola, and J. Landaluze, "Modelling and Validation of a 1-DOF Arm Powered by Pneumatic Muscles", *Industrial Simulation Conference ISC'2007, 11-13 June 2007, Delft, Holland*.
- [5] T.D.C. Thanh, and K.K. Ahn, "Nonlinear PID control to improve the control performance of 2 axes pneumatic artificial muscle manipulator using neural network", *Mechatronics*, no. 16, pp. 577-587, 2006.
- [6] K. Balasubramanian, and K.S. Rattan, "Trajectory tracking control of a pneumatic muscle system using fuzzy logic", *NAFIP'2005, Annual Meeting of the North American Fuzzy Information Processing Society*.
- [7] A. Pujana-Arrese, S. Riaño, J. Arenas, A. Martinez-Esnaola, and J. Landaluze, "H_∞ position Control of a 1-DoF Arm Powered by Pneumatic Muscles", *8th Portuguese Conference on Automatic Control CONTROLO'2008, 21-23 July 2008, Vila Real, Portugal*.
- [8] J. Arenas, A. Pujana-Arrese, S. Riaño, J. Arenas, A. Martinez-Esnaola, and J. Landaluze, "Sliding-mode Position Control of a 1-DoF Set-up based on Pneumatic Muscles", *UKACC Control Conference CONTROL2008, 2-4 Setember 2008, Manchester (UK)*.
- [9] N.G. Tsagarakis, and D.G. Caldwell, "A compliant exoskeleton for multi-planar upper limb physiotherapy and training", *ICAR'07, The International Conference on Advanced Robotics, 21-24 August 2007, Jeju Island, Korea*.
- [10] C. Edward, and S.K. Spurgeon, *Sliding Mode Control. Theory and Applications*. Taylor & Francis Ltd. ISBN 0-7484-0601-8, 1998.

APPENDIX

The design values used in equations (2), (4), (6), (7) and (9) to design the sliding-mode controller are as follows:

$$A = \begin{bmatrix} -5.7176 & -125.6904 & -0.0044 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}; B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C = [2.9 \quad 415.3 \quad 2350]$$

$$S = [0 \quad -1 \quad -9.28 \quad -33.64]$$

$$L = [0 \quad -23.5624 \quad 93.5496 \quad -672.7956]$$

$$L_r = 0.2863; L_{rdot} = 0.0143; S_r = -0.0143$$

$$P = 0.025; \phi = -0.2; K = 0.02$$

$$G^T = [-1.9609 \quad 0.3491 \quad 0]; F = 14.3243; K(\text{obs}) = -2$$

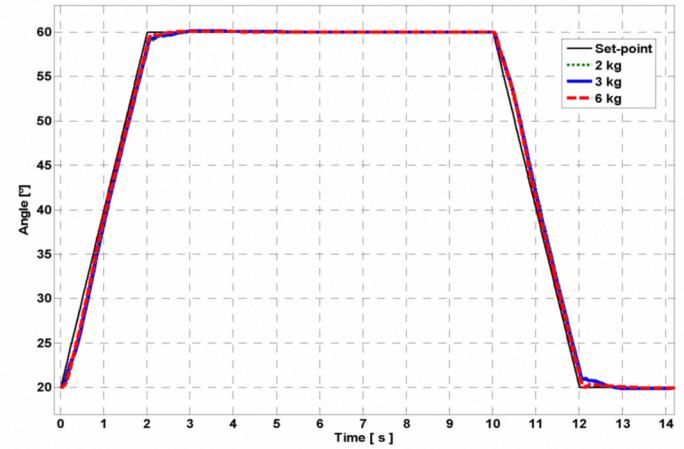


Figure 10. Experimental results with different tip masses for internal pressure loops based controller