10.24425/acs.2022.141718

*Archives of Control Sciences*
*Volume* 32(LXVIII), 2022
*No.* 2, *pages* 409–427

# Performance analysis between a hybrid force/position and conventional controllers for a wrist exoskeleton

Valeria AVILÉS, Oscar F. AVILÉS, Jorge APONTE, Oscar I. CALDAS
and Mauricio F. MAULEDOUX

This study analyses the performances of various path controlling strategies for a 3-degrees of freedom wrist exoskeleton, by comparing key indicators, such as rise time, steady-state error, and implementation difficulty. A model was built to describe both system's kinematics and dynamics, as well as 3 different controllers (PID, PD+, and a hybrid force/position controller) that were designed to allow each joint to perform smooth motions within anatomic ranges. The corresponding simulation was run and assessed via Matlab (version 2020a). In addition to the performance comparison, remarkable characteristics could be identified among controllers. PD+ showed higher response speed than the other controllers (about 4 times), and PID was reinforced as the technique with the easiest implementation due to the smallest matrices. The study also allowed to greater potential of the hybrid controller to interact with its environment, i.e., the robotic device.

**Key words:** exoskeleton, hybrid control, force, performance, position

## 1. Introduction

Many of the causes for motor impairment in upper limbs are related with communication abnormalities between neural motor generators and muscle motor units, such as after spinal cord injuries, stroke, and congenital diseases that compromises the motor function and negatively impact the development of daily living activities [21]. Consequently, recovering mobility becomes the primary need for these kinds of individuals when expecting to restore quality of life.



All authors are with Davinci Research Group, Mechatronics Engineering, Militar Nueva Granada University, Cr 11 No 101-80, Bogotá, Colombia e-mails: {u1802892, oscar.aviles, jorge.aponte, oscar.caldas, mauricio.mauledoux}@unimilitar.edu.co

This work was supported by the Research Vice-Rectory of the Universidad Militar Nueva Granada – Colombia, through the project ING-IMP-3124.

Received 17.06.2021.

Physical rehabilitation by conventional means is commonly agreed as repetitive and highly dependent on the manual dexterity of the healthcare provider, which favors overextended efforts and hinders quality of single sessions. Therefore, many devices have being developed in the last decades to overcome with these limitations, by combining mechanical, electronic and control systems with intuitive interfaces and robotic principles, i.e., execution of planned routines, imitation of human joints motions and integration with advanced control techniques [20].

In fact, many of the recent developments in the field are now evident in exoskeletons [11], specially by better and more flexible mechanical and electronic designs and controller-based configurations, all focused on attending the users need, such as support, movement amelioration and strengthening. Control systems, particularly, represent the basis of security and well-functioning as leading human-based decision making via diverse and complex control algorithms [19]. There are different kind of approaches to control exoskeletons, from de simplest ON/OFF technique until the more robust controllers that require kinematic and dynamic modeling and algorithms based on artificial intelligence, among other requirements [10, 18, 27, 31].

In fact, rehabilitation mechanisms for upper limbs relates to the control systems according to the type of actuation, e.g. Shape-Memory Alloy-based actuator (SMA) and Series-Elastic actuators (SEA) are intended to be reduce the influence of non-linearities, which highlights the benefits of impedance-based, PID, and PD controllers [1, 13, 14, 28, 30]. Pneumatic muscles, on the other hand, typically require regulation of air pressure to control position rather than force, which leads to less robust but smooth controllers that minimize transmitting torques to human joints, which could be highly harmful [2, 4, 12]. Electrical actuators such as DC and servomotors, are used in application that offer the chance to control several variables, which also diversify the feedback control technique [3, 7, 22].

This paper follows the above-mentioned rationale by describing the development of three different strategies that appropriate for exoskeletons as controlling smooth motions safely within the human joint's ranges. We authors hypothesize that the hybrid force/position controller could be more appropriate because its capability to better interact with its environment while controlling multiple variables. To evaluate such suitability, a comparative study was performed between PID, PD+ and the hybrid controller, to identify the real benefits of the last one in applications with need of high precision and repeatability, such as the wrist exoskeleton. The next chapters describe the procedure carried out to assure each controller to reach the desired position and velocity per joint in the wrist exoskeletons, by using design, modeling, and simulation theoretical methods, as well as computational tools included in Matlab (e.g., Simscape).

## 2. Background

Most of the goals in control of robotic devices are related with following desired paths or tracking position trajectories [26]. In the first case, it is important to satisfy precision demands during the end effector following a defined continuous path inside a workspace and in time domain, whereas in the second case precision is only needed at a partial destination that belongs to a sequence of positions to be followed by the robot [16, 17]. The objectives at each control strategy are:

In path-following a vector function $\tau$ is designed to force a set of joint positions $q$ to move towards a desired set of positions $q_d$, while joint velocities $\dot{q}$ follow the desired velocities $\dot{q}_d$. This is described by Eq. (1).

$$\lim_{\tau \to \infty} \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} q_d \\ \dot{q}_d \end{bmatrix}. \tag{1}$$

Position trajectory-planning is only intended for finding a vector function $\tau$ that command joint positions $q$ to tend towards desired positions $q_d$, as in Eq. (2).

$$\lim_{\tau \to \infty} \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} q_d \\ 0 \end{bmatrix}. \tag{2}$$

In both cases, the kay function must produce an error equal to zero (See Eq. (3)).

$$\lim_{\tau \to \infty} \frac{\delta}{\delta t} \begin{bmatrix} \widetilde{q}(t) \\ \dot{\widetilde{q}}(t) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{3}$$

where $\widetilde{q}$ and $\dot{\widetilde{q}}$, respectively represent position and velocity errors due to Eq. (4) and Eq. (5).

$$\widetilde{q} = q_d(t) - q(t), \tag{4}$$

$$\dot{\widetilde{q}} = \dot{q}_d(t) - \dot{q}(t). \tag{5}$$

Therefore, the closed loop control law for both trajectory-planning and path-following in terms of the joints are shown in Eq. (6) [16, 17], being $D(q)$, $C(q, \dot{q})$, and $G(q)$ the matrices that describe the system dynamics.

$$\frac{\delta}{\delta t} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ D(q)^{-1} \left[ \tau(t) - C(q, \dot{q})\, \dot{q} - G(q) \right] \end{bmatrix}. \tag{6}$$

These controllers are considered as advanced control strategies, since they use both the kinematics and dynamics of the robot [26]. The most common approaches are PD, PID, and PD+ with gravity compensation [8].

## 3. Methods

According to the rationale exposed above this work focused on path following, since this approach is agreed to promote smooth motions [25,29]. Hence, the path planning is firstly done by creating position, velocity and acceleration functions that accomplish the required time for each joint to reach the user-defined desired position.

### 3.1. Path following

The function is a third-grade polynomial able to describe a uniform motion by applying two constrains to position $q(t)$ and another two constrains to velocity $\dot{q}(t)$. Constrains for $q(t)$ are current position equal to initial value and desired position equal to final value, whereas $\dot{q}(t)$ is constrained to a continuous velocity function, namely, initial and final velocities mut be equal to zero [29]. A cubic polynomial satisfies all four conditions since it has four coefficients that can be defined to solve the problem: Eq. (7), Eq. (8), and Eq. (9) correspond to position, velocity, and acceleration as consecutive derivatives.

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3, \tag{7}$$

$$\dot{q}(t) = a_1 + 2a_2 t + 3a_3 t^2, \tag{8}$$

$$\ddot{q}(t) = 2a_2 + 6a_3 t. \tag{9}$$

Replacing all four constrains in Eq. (7), Eq. (8), and Eq. (9) leads to an equivalent system of four equations and four variables, as shown in Eq. (10), Eq. (11), Eq. (12) and Eq. (13).

$$a_0 = q_0\,, \tag{10}$$

$$a_1 = 0, \tag{11}$$

$$a_2 = \frac{3}{t_f^2}\left(q_f - q_0\right), \tag{12}$$

$$a_3 = \frac{-2}{t_f^3}\left(q_f - q_0\right). \tag{13}$$

Finally, the third-grade polynomial were built to relate any initial joint angle with the final position. In this case, the initial angles are equal to zero, and the final angles are defined by the range of motion of the wrist exoskeleton of Aponte et al., described in Fig. 1 and Table 1 [6].

Motion profiles for position, velocity and acceleration are shown in Fig. 2.

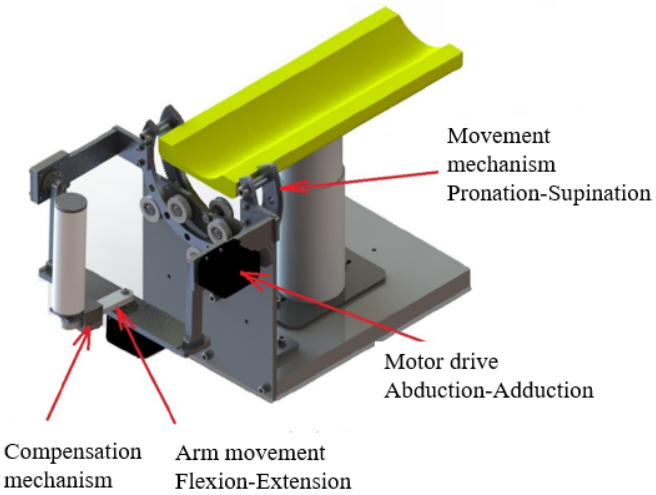



Figure 1: Adopted wrist exoskeleton. Modified with permission from (Aponte, 2018) [6]

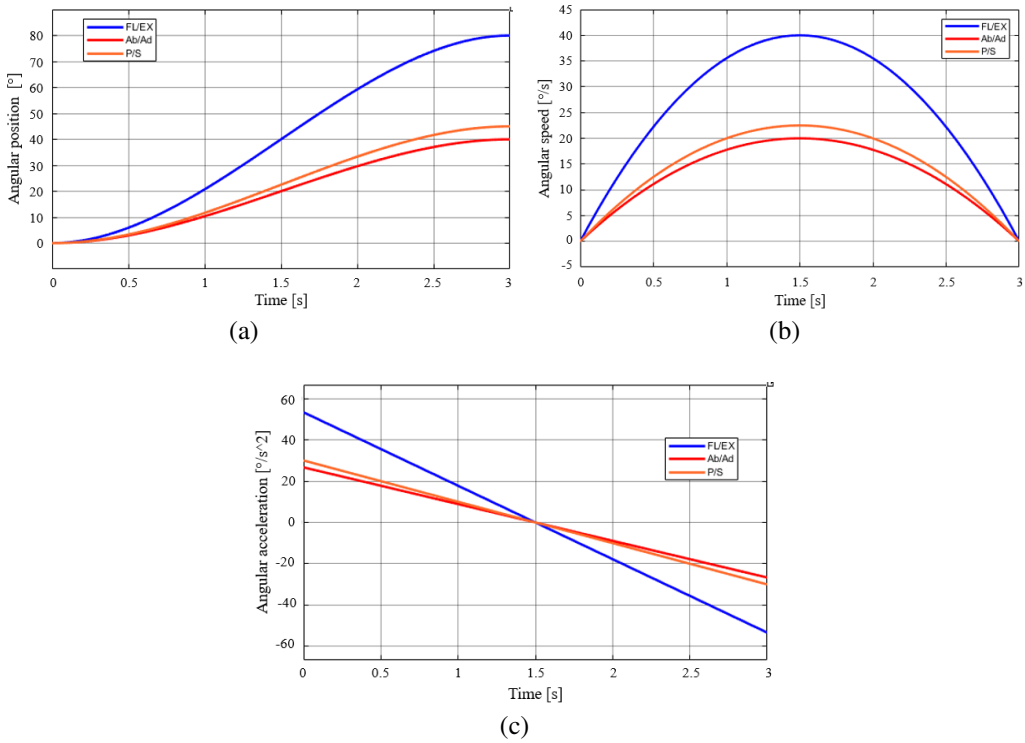



Figure 2: Motion profiles in wrist joints at desired final angle $\theta_d$: (a) position, (b) velocity, and (c) acceleration. FL/EX is flexion-extension at $\theta_d = 80°$, P/S is pronation-supination ($\theta_d = 45°$), and Ab/Ad is abduction-adduction ($\theta_d = 80°$)

Table 1: Ranges of motion from the three joints in the wrist

| Joint | Motion | ROM [°] | Motion | ROM [°] |
|---|---|---|---|---|
| 1 | Flexion | 80 | Extension | 70 |
| 2 | Abduction | 30 | Adduction | 40 |
| 3 | Pronation | 45 | Supination | 45 |

ROM – range of motion

### 3.2. PID controller

PID stands for Proportional-Derivative-Integral controller and this last component is used to eliminate the steady-state error and includes a gravity compensation, i.e., since the motion is described in a horizontal plane a vector needs to be constantly applied against gravity (vertically) to assure error and velocity to be equal to zero when reaching the desired position [24]. Eq. (14) describes this three-term controller.

$$\tau_{\text{PID}} = K_p\widetilde{q} - K_v\dot{q} + K_i\int_0^t \widetilde{q}(\mu)\,\mathrm{d}\mu + G(q), \tag{14}$$

where $K_p$, $K_v$ and $K_i$ are diagonal $3 \times 3$ matrices containing the proportional, derivative, and integral gains, respectively; $G(q)$ is the gravity vector, $\widetilde{q}$ is position error, and $\dot{q}$ resulting velocity. The controller general scheme can be seen in Fig. 3.

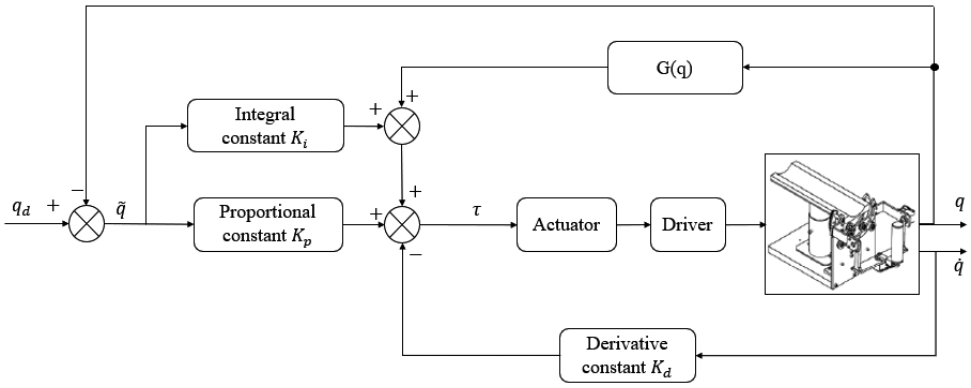



Figure 3: PID controller's block diagram

Simulation was made in Matlab Simulink (version 2020a) to evaluate path following and the error for each joint. The Matlab PID block and its auto-tuning tool was used to define the controller's constants, including the derivate filter.

A significant oscillations reduction is achieved, which implies an advantage to be considered for implementation. Eq. (15) describes the final PID architecture.

$$\tau_{\text{PID}} = P + I\frac{1}{s} + D\frac{N}{1 + N\frac{1}{s}} + G(q). \tag{15}$$

### 3.3. PD+ controller

The PD+, which also contains a gravity compensation, is made by three components: the proportional controller for position error, the proportional controller for velocity, and the exoskeleton dynamics, thus it also requires the dynamic model besides the desired paths, velocities and accelerations [24]. This controller, as described in Eq. (16), reacts fast to frequency changes to correctly follow the path at desired velocity.

$$\tau_{\text{PD+}} = K_p\widetilde{q} + K_v\dot{\widetilde{q}} + D(q)\ddot{q}_d + C\left(q, \dot{q}\right)\dot{q}_d + G(q), \tag{16}$$

where $K_p$ and $K_v$ are diagonal $3\times3$ matrices containing the proportional gains for position and velocity; $D(q)$ is the inertia matrix; $C(q,\ \dot{q})$ is the Coriolis matrix; $G(q)$ is el gravity vector, $\widetilde{q}$ is position error, and $\dot{\widetilde{q}}$ is velocity error. See Fig. 4 for general scheme.

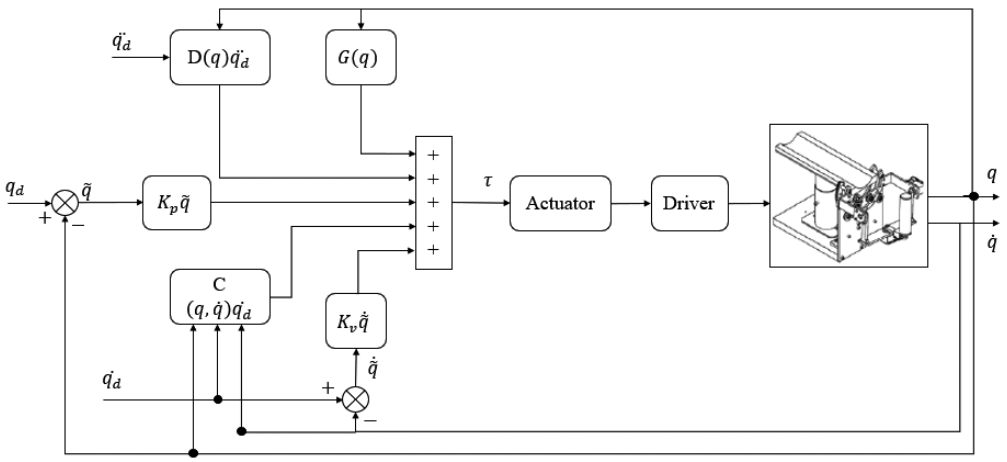



Figure 4: PD+ controller's block diagram

It is important to highlight that this controller has a greater computational cost in spite of its high precision, i.e., processing time is significantly higher to other controllers [9, 29], besides, it is able to regulate all three position, velocity and acceleration, which also implies a more complex implementation process.

Its implementation required the use of Eq. (7), Eq. (8), and Eq. (9) and the auto-tuning tool embedded in Matlab to obtain the constants values, similar to the PID controller, which resulted in the architecture shown in Eq. (17).

$$\tau_{\mathrm{PD+}} = P + D\frac{N}{1 + N\frac{1}{s}} + D(q)\ddot{q}_d + C\left(q\dot{q}\right)\dot{q}_d + G(q). \tag{17}$$

Similarly, the matrices represent the system dynamics, inertia, Coriolis, and gravity.

### 3.4. Hybrid force/position controller

This controlling strategy depends on desired forces in the direction of the constrained workspace and a desired path in the remaining directions [25, 29].

The position controller looks for a correct path following via information collected by sensors and position feedback, whereas the force controller demands for environmental modelling, force sensors and both force and position feedback, thus, workspace is divided into two subspaces: force and position [15]. Fig. 5 shows the representation of the exoskeleton with the environment during controller application.

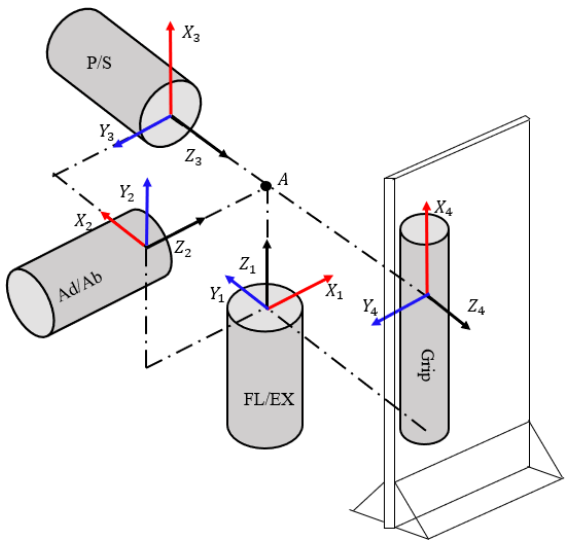



Figure 5: Scheme for wrist exoskeleton with environment in hybrid controlm

The block diagram in Fig. 6 shows two parallel closed loops for position and force, but also the contribution of each joint as vector in the final summation that is transformed into torques for the exoskeleton's actuators [26].

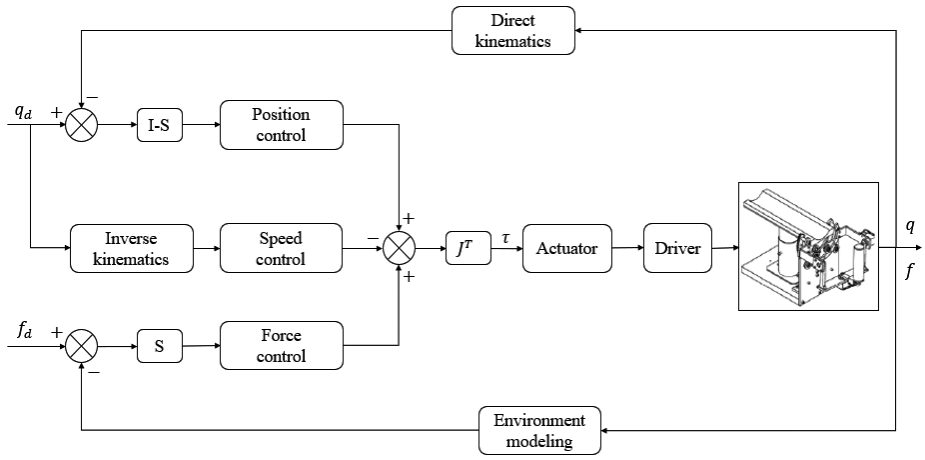



Figure 6: Hybrid force/position controller's block diagram

Blocks $S$ and $(I - S)$ are used to select the type of controller by means of an identity matrix, in order to avoid any possible conflict of simultaneous controllers' action over the same actuator [25, 26]. The mathematical approach for this type of controllers is described in Eq. (18).

$$\tau_{CH} = J^T(q)\left[K_f S\left(f_d - f_e\right) + K_p(I - S)\left(q_d - q\right) - K_v\dot{q}\right], \tag{18}$$

where ones in matrix $S$ set the directions to be controlled by force, $I \in R^{m\times m}$ is the identity matrix, $K_v \in R^{m\times m}$ is the derivative gain related with end-effector velocity, and the proportional constants for position and force controller are $K_f \in R^{m\times m}$ and $K_p \in R^{m\times m}$.

Implementation of the hybrid controller requires knowledge about the environment to be interacted with, since the exoskeleton must adapt every action and motion based on the physical characteristics of the goals within the workspace. In order to perform this action, an elastic environment was modeled to facilitate the interaction with the end-effect, due to its level of strain and flexibility [25, 26]. In fact, the mathematical model shown in Eq. (19) corresponds to an elastic compliant non-coupled environment.

$$f_e = K_e\left(q - q_e\right), \tag{19}$$

where $K_e \in R^{m\times m}$ is the surface's stiffness constant $K_e = \mathrm{diag}(1500)$ Nm.

Once the environment was established, the surface location was set to $Xe = 0.061$ m from the origin, over which the force could be applied along the $x$-axis. Then, all the controller's blocks were developed, as follows.

***Kinematics blocks*** containing the direct and inverse kinematics of the exoskeleton. The first block describe the position of an object within the 3-dimensional space in respect to the fixed referential system [29]. The second

seeks for the values that the exoskeleton joint's coordinates must adopt in order to the end-effector to correctly achieve the required position and orientation [5].

***Selection blocks S and IS.*** The only difference between them is the use of the identity matrix in the second one, which assures opposition. *S* controls force whereas *IS* controls position in the remaining axes. The blocks output is a $3 \times 1$ vector.

***Control blocks.*** Constants for hybrid control were extracted from the matrices in Eq. (20), Eq. (21) and Eq. (22), for position, force, and velocity, respectively.

$$\text{Matrix } K_p = \begin{bmatrix} K_{p1} & 0 & 0 \\ 0 & K_{p2} & 0 \\ 0 & 0 & K_{p3} \end{bmatrix}, \tag{20}$$

$$\text{Matrix } K_f = \begin{bmatrix} K_{f1} & 0 & 0 \\ 0 & K_{f2} & 0 \\ 0 & 0 & K_{f3} \end{bmatrix}, \tag{21}$$

$$\text{Matrix } K_v = \begin{bmatrix} K_{v1} & 0 & 0 \\ 0 & K_{v2} & 0 \\ 0 & 0 & K_{v3} \end{bmatrix}. \tag{22}$$

The first two block take the output from the selection block and set a gain to the axis that is interacting in each case. The velocity block takes the exoskeleton's current and desired position and set a gain of about 20% of the proportional gain [26]. The blocks output is also a $3 \times 1$ vector.

***Environment modeling block.*** This block takes the exoskeleton's current position and the interacting environment to evaluate the interaction force when they get in contact or otherwise it is set to zero. The block's output is also a $3 \times 1$ vector.

***Transposed Jacobian block.*** It calculates the torque from the exoskeleton's force and position values, along with the transposed Jacobian. During simulation significant oscillation were detected despite of the precise reference following, and therefore, the approach suggested by Prada et al. was used to partially solved the problem, which is shown in Eq. (23) [25].

$$\tau_{\text{CH}} = J^T \left[ K_s S \left( f_d - f_e \right) - K_v \dot{x} + K_p \left( P + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}} \right) (I - S) \left( x_d - x \right) \right]. \tag{23}$$

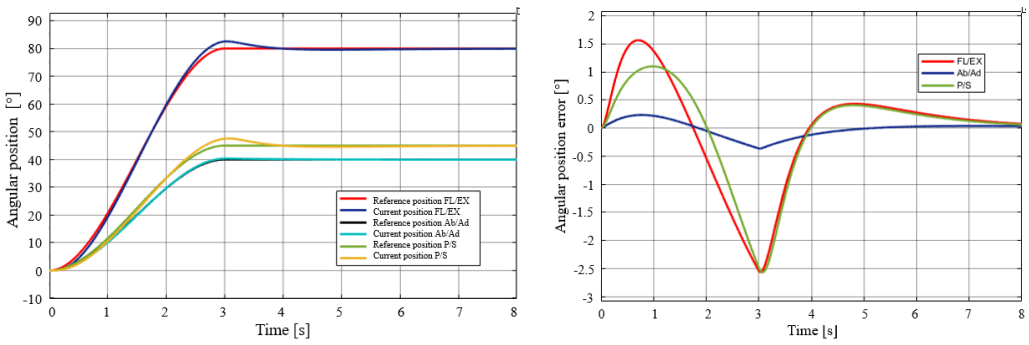
## 4. Results

### 4.1. PID controller

Table 2 collects the constants defined for the PID controller, which allowed the behavior shown in Fig. 7, given in terms of correct path following and error for all three joints. As can be seen, error tends to zero in time, which indicates that the exoskeleton performs the desired motions within the assigned ranges.

Table 2: PID controller's gain

| Matrices | Gains | | |
|---|---|---|---|
| | $K_1$ | $K_2$ | $K_3$ |
| **P** | 251794.3 | 643523.9 | 49332.2 |
| **I** | 117876.9 | 149460.3 | 23094.7 |
| **D** | 132070.6 | 615636.8 | 25875.6 |
| **N** | 54.53 | 4549.6 | 54.53 |




(a) (b)

Figure 7: Results for PID controller: (a) path following and (b) position error

Furthermore, all three joints followed the path on time and showed a decreasing error that tend to zero, with the presence of a mild overshoot in two of them caused by the values of the derivative and integral gains, which also reduces the rise time and accomplish the fast response that is expected to be appropriate for the problem [23].

### 4.2. PD+ controller

Fig. 8 shows the path, velocity, and acceleration following, considering the above-mentioned position and velocity constrains, by also ensuring fast response

after reference changes and a low steady-state error in all three joints (less than 5%).

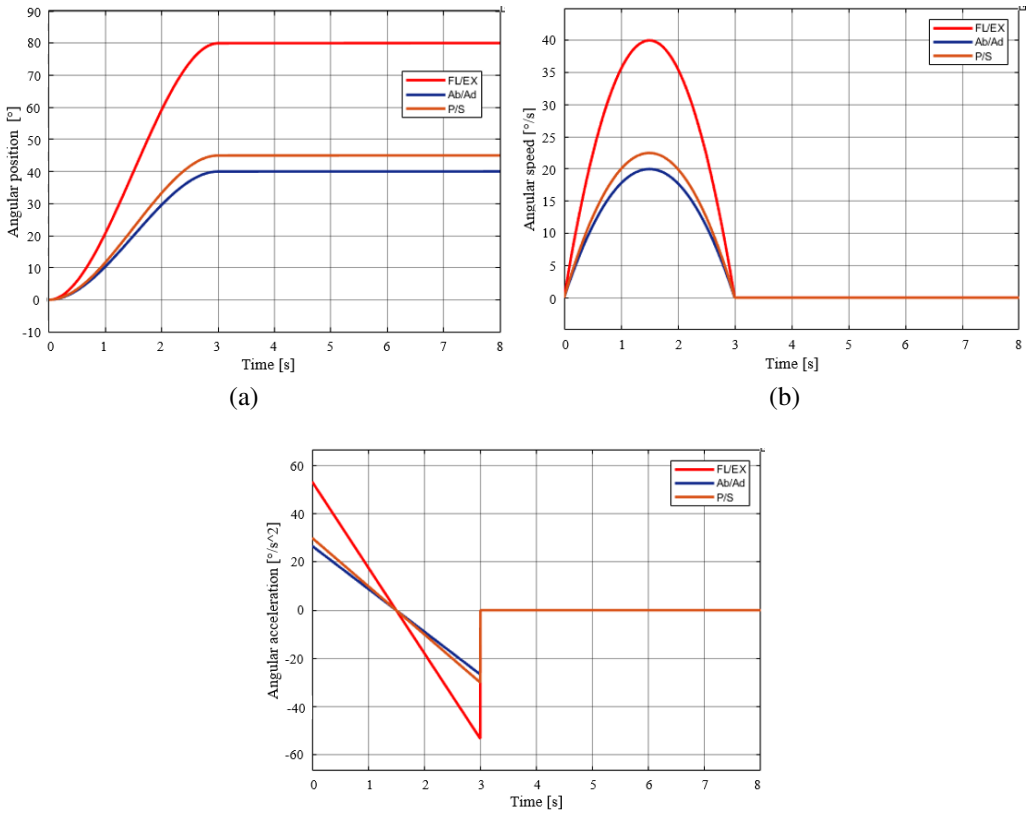



Figure 8: Results of PD+ controller: (a) paths, (b) velocities, and (c) accelerations following

Table 3 collects the constants defined for the PD+ controller.

Table 3: PD+ controller gains

| Matrices | Gains | | |
|---|---|---|---|
| | $K_1$ | $K_2$ | $K_3$ |
| **P** | 26.57 | 6.27 | 9.825 |
| **D** | 1585.7 | 374.41 | 586.35 |
| **N** | 11.42 | 11.42 | 11.42 |

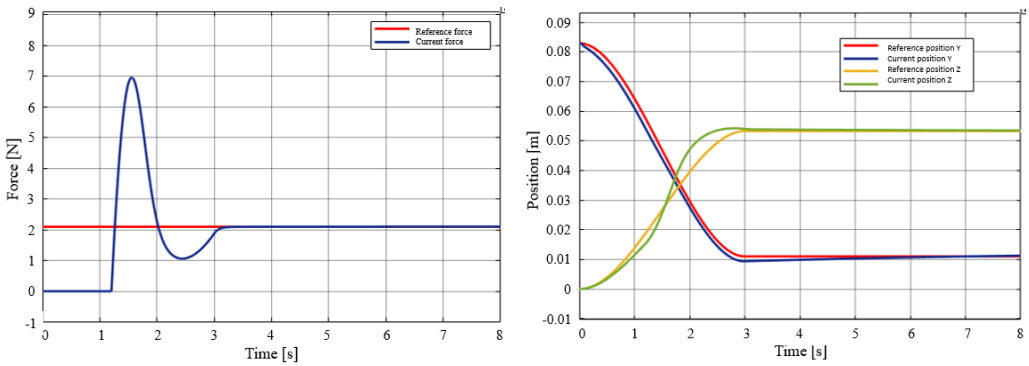
### 4.3. Hybrid force/position controller

Final gains in hybrid controller for the wrist exoskeleton are organized in Table 4.

Table 4: Hybrid controller gains

| Matrices | Gains | | |
|---|---|---|---|
| | $K_1$ | $K_2$ | $K_3$ |
| $\mathbf{K_p}$ | 4500 | 10000000 | –650000 |
| **P** | 40.611 | 190 | –0.99 |
| **I** | 0.19 | 27 | –15 |
| **D** | 2130.13 | 950 | –820 |
| **N** | 0.545 | 800 | 600 |
| $\mathbf{K_v}$ | 900 | 2000000 | –130000 |
| $\mathbf{K_f}$ | 5900000 | 100 | 100 |

Results can be seen in Figs. 9 and 10, where the both force and position errors tend to zero, due to the action of the constants obtained by Matlab's auto-tuning tool. Therefore, the exoskeleton is able to correctly follow the desired path in $x$ and $y$ axes, as well to apply the desired level of force in $x$-axis during the entire environmental interaction.

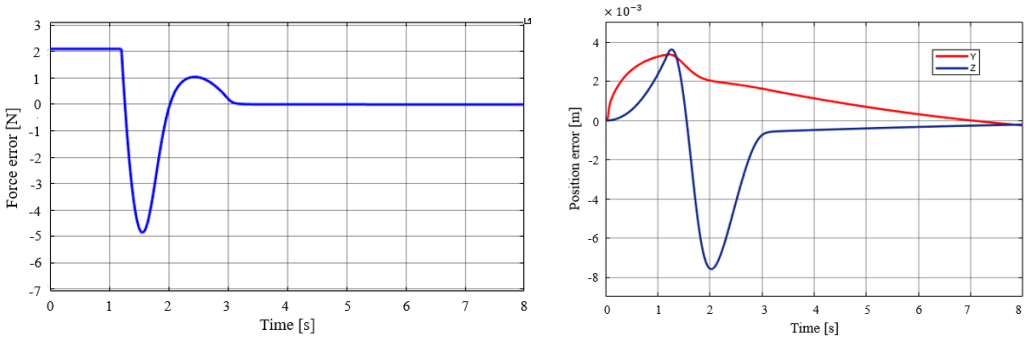



(a) (b)

Figure 9: Hybrid controller results: (a) force control in $x$ and (b) path following in $y$ and $z$

Table 5 summarizes the findings about the three developed controllers.

Accordingly, some differences can be extracted from the charts and variables from each of the controllers when trying to follow the desired paths for the wrist exoskeleton. The PD+ controller showed better performance because of using the




(a) (b)

Figure 10: Hybrid controller errors: (a) force in $x$ and (b) path in $y$ and $z$

Table 5: Main findings about controllers' performance

| PID controller | PD+ controller | Hybrid controller |
|---|---|---|
| It follows the path correctly despite not showing error equal to zero, instead, it reduces to produce a better system behavior. | Relatively high complexity Because of controlling all variables simultaneously (position, velocity, and acceleration). | It follows the path reference but describes oscillations that need to be compensated by a Matlab PID block. |
| It contains a mild position overshoot for two of the joints, due to the derivative and integral constants. | It has a significantly higher computational cost, and the processing time is high. | Each joint contributes to the controller execution by using to closed loops. |
| | High precision and fast rise time. | Balances the position and force variables while reducing the errors to zero. |

model dynamics and the gravity compensation, which inserts further stability to the system. Charts in Fig. 8 confirmed the hypothesis by showing how all three variables are controlled and respond almost immediately to the changes in the path, whereas the other two controllers take between three and four times longer to reach steady-state and had higher error values (see Fig. 7b and 10). Nevertheless, implementing this controller is rather complicated due to the high computational costs.

PID controller, on the other hand, described a similar behavior in path following except for a mild overshoot in pronation/supination and flexion/extension joints (see Fig. 7a), which has serious implication in settling time and error magnitude. However, it is important to highlight the ease of implementation since it does not require all the matrices that are mandatory for the other two controllers.

Finally, the hybrid force/position controller with autotuned PID block displayed higher error than the other controllers, which might be due to the double closed-loop environmental interaction. Results for path following for $y$ and $z$ axes are not as good as in the other controllers, mainly because of each position depending on a single angle $q_2$, while error signals tend to zero as in Fig. 10b. Regarding the force control, Fig. 9a displays the existence of oscillations before settling at 3 seconds, which is caused by the path following control law, since both loops require the contributes of all joint to work.

## 5. Conclusions

This paper focused on solving the problem of path following control for 3-degrees-of-freedom a wrist exoskeleton, through three different approaches: the first two were conventional control techniques applied to exoskeletons: PID, PD+. The first one was easier to implement but showed higher steady-state error and some overshoot when reaching the settling value. Contrarily, PD+ had a faster response at controlling the three variables (position, velocity, and acceleration) but was harder to implement.

The third technique was a hybrid force/position controller complemented with a PID autotuned block, which demonstrated to be capable of consider the dynamics of interaction between the mechanics and the environment. This feature is well priced in rehabilitation because of the consideration of the forces exerted by the user. However, some drawbacks were also identified: it not as flexible as expected and requires new tuning at every modification of environment characteristics or tasks in any axis, which was rather difficult because of the torque values oscillations, the need of path following from only one joint variable, and the fact that every joint contributed to two both controllers. Consequently, the autotuning block from Matlab needed to be used.

Then, according to what was previously exposed and performing a comparison of the important characteristics of a controller such as response time, error in steady-state, and implementation, it is possible to see how traditional controllers are simpler to implement, have a much smaller error than the hybrid controller as well. as a response time is quicker but they are as mentioned it is due to the fact that they only have a control loop that differs from the hybrid controller they have. While traditional controllers have a similar answer that is very acceptable, this differentiating value between each is found in the ease of implementation, so, depending on what the device requires the best option, it will be to develop a PID type controller (for position) or a hybrid controller (for strength/position).

Finally, the overcame simulations let to detect the ability of controller to change the exoskeleton's end-effector orientation continuously via torques. In the

case of the hybrid controller, this feature came along with the power of holding continuous force over the environment during contact.

## References

[1] P. Agarwal and A.D. Deshpande: Impedance and force-field control of the index finger module of a hand exoskeleton for rehabilitation. *IEEE International Conference on Rehabilitation Robotics (ICORR)*, (2015). DOI: 10.1109/ICORR.2015.7281180.

[2] H. Al-Fahaam, S. Davis, and S. Nefti-Meziani: Wrist rehabilitation exoskeleton robot based on pneumatic soft actuators. *International Conference for Students on Applied Engineering, ICSAE*, (2017). DOI: 10.1109/ICSAE.2016.7810241.

[3] F. Amirabdollahian, S. Ates, A. Basteris, A. Cesario, J. Buurke, H. Hermens, D. Hofs, E. Johansson, G. Mountain, N. Nasr, S. Nijenhuis, G. Prange, N. Rahman, P. Sale, F. Schätzlein, B. Van Schooten, and A. Stienen: Design, development and deployment of a hand/wrist exoskeleton for home-based rehabilitation after stroke – SCRIPT project. *Robotica*, **34**(8), (2014), 1331–1346. DOI: 10.1017/S0263574714002288.

[4] G. Andrikopoulos, G. Nikolakopoulos, and S. Manesis: Motion control of a novel robotic wrist exoskeleton via pneumatic muscle actuators. *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, (2015). DOI: 10.1109/ETFA.2015.7301464.

[5] J. Angeles: *Fundamentals of Robotic Mechanical Systems*. In *Dispensa*, (Fourth), Springer, 2007.

[6] J. Aponte: *Integracion simultánea de aspectos cinemáticos y dinamicos para el diseño optimo de un dispositivo para rehabilitacion de muñeca*. Instituto Politécnico Nacional. 2021, (in Spanish).

[7] S. Ates, J. Lobo-Prat, P. Lammertse, H. Van Der Kooij, and A.H.A. Stienen: SCRIPT Passive Orthosis: Design and technical evaluation of the wrist and hand orthosis for rehabilitation training at home. *IEEE International Conference on Rehabilitation Robotics*, (2013). DOI: 10.1109/ICORR.2013.6650401.

[8] M. Ballesteros and J. Martinéz: *Diseño y construcción de un exoesqueleto de miembros inferiores que emula la marcha humana*. Univesidad Militar Nueva Granada. 2015. http://hdl.handle.net/10654/13731. (in Spanish).

[9] A. Bautista and C. Aranguren: *Diseño, implementación y puesta en funcionamiento de un sistema de control de marcha humana para un exoesqueleto de miembro inferior*. Universidad Militar Nueva Granada. (2015). http://hdl.handle.net/10654/13795. (in Spanish).

[10] J.H. Beekhuis, A.J. Westerveld, H. Van Der Kooij and A.H.A. Stienen: Design of a self-aligning 3-DOF actuated exoskeleton for diagnosis and training of wrist and forearm after stroke. *IEEE International Conference on Rehabilitation Robotics*. (2013). DOI: 10.1109/ICORR.2013.6650357.

[11] P. Freni, E. Marina Botta, L. Randazzo and P. Ariano: Innovative hand exoskeleton design for extravehicular activities in space. In B. Pernici, S. Della Torre, B. Colosimo, T. Faravelli, R. Paolucci and S. Piardi (Eds.), *Springer Briefs in Applied Sciences and Technology*. Springer. (2014). DOI: 10.1007/978-3-319-03958-9.

[12] M. Haghshenas-Jaryani, R.M. Patterson, N. Bugnariu and M.B.J. Wijesundara: A pilot study on the design and validation of a hybrid exoskeleton robotic device for hand rehabilitation. *Journal of Hand Therapy*, **33**(2), (2020), 198–208. DOI: 10.1016/j.jht.2020.03.024.

[13] J. Hope and A. McDaid: Development of wearable wrist and forearm exoskeleton with shape memory alloy actuators. *Journal of Intelligent and Robotic Systems: Theory and Applications*. **86** (2017), 397–417. DOI: 10.1007/s10846-016-0456-7.

[14] I. Jo and J. Bae: Design and control of a wearable and force-controllable hand exoskeleton system. *Mechatronics*, **41** (2017), 90–101. DOI: 10.1016/j.mechatronics.2016.12.001.

[15] A. Karamali Ravandi, E. Khanmirza and K. Daneshjou: Hybrid force/position control of robotic arms manipulating in uncertain environments based on adaptive fuzzy sliding mode control. *Applied Soft Computing Journal*, **70** (2018), 864–874. DOI: 10.1016/j.asoc.2018.05.048.

[16] R. Kelly and V. Santibáñez: Control de movimiento de robots manipuladores. In I. Capella and M. Caicoya (Eds.), *MMW Fortschritte der Medizin*. Pearson. (2003).

[17] R. Kelly, V. Santibáñez and A. Loría: Control of robot manipulators in joint space. In M. Grimble and M. Johnson (Eds.), *Review Literature And Arts Of The Americas*. Springer, (2015).

[18] J. Lee, B.W. Song and W. Yang: Design of exoskeleton-type wrist human machine interface based on over-actuated coaxial spherical parallel mechanism. *Advances in Mechanical Engineering*. (2018). DOI: 10.1177/1687814017753896.

[19] E. Lerma, V. Baixauli, F. Selma and F. García: El papel de la rehabilitación tras las reparaciones de las inestabilidades de muñeca. *Revista Iberoamericana de Cirugía de La Mano*. (2016). DOI: 10.1016/j.ricma.2016.09.001. (in Spanish).

[20] H.S. Lo and S.Q. Xie: Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects. *Medical Engineering and Physics*, **34**(3), (2012), 261–268. DOI: 10.1016/j.medengphy.2011.10.004.

[21] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy and S. Leonhardt: A survey on robotic devices for upper limb rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, **11** (2014). DOI: 10.1186/1743-0003-11-3.

[22] F. Marini, V. Squeri, L. Cappello, P. Morasso, A. Riva, L. Doglio and L. Masia: Adaptive wrist robot training in pediatric rehabilitation. *IEEE International Conference on Rehabilitation Robotics*, (2015). DOI: 10.1109/ICORR.2015.7281195.

[23] K. Ogata: *Ingenieria De Control Moderna*. In M. Martín and E. Martín (Eds.), Prentice-Hall Hispanoamericana, S.A. Pearson. 2010. (in Spanish).

[24] J.L. Pons: *Wearable Robots: Biomechatronic Exoskeletons* (Issue 1). John Wiley & Sons. (2008). DOI: 10.16309/j.cnki.issn.1007-1776.2003.03.004.

[25] V. Prada, P. Niño and O. Avilés: *Control Híbrido control Fuerza-Posición para manipulador de 2 GDL* (Vol. 1). Editorial academica española. 2012. (in Spanish).

[26] F. Reyes: *Robótica – control de Robots manipuladores*, F. Rodríguez and M. Grillo (Eds.), Alpha Omeg. 2016. (in Spanish).

[27] A.F. Ruiz-Olaya: Towards a robotic exoskeleton for remote evaluation of elbow and wrist joints. *International Conference on Virtual Rehabilitation, ICVR*. (2015). DOI: 10.1109/ICVR.2015.7358621.

[28] D. Serrano, D.S. Copaci, L. Moreno and D. Blanco: SMA based wrist exoskeleton for rehabilitation therapy. *IEEE International Conference on Intelligent Robots and Systems*. (2018). DOI: 10.1109/IROS.2018.8593987.

[29] B. Siciliano, L. Sciavicco, L. Villani and G. Oriolo: *Robotics Modelling Planning and Control*. In M. Grimble and M. Jonhson (Eds.). Springer. (2009). DOI: 10.1007/978-1-84628-642-1.

[30] L. Sutton, H. Moein, A. Rafiee, J.D.W. Madden and C. Menon: Design of an assistive wrist orthosis using conductive nylon actuators. *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. (2016). DOI: 10.1109/BIOROB.2016.7523774.

[31] Z.G. Xiao, A.M. Elnady and C. Menon: Control an exoskeleton for forearm rotation using FMG. *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. (2014). DOI: 10.1109/biorob.2014.6913842.