Design and implementation of fault-tolerant control strategies for a real underactuated manipulator robot

Claudio Urrea · John Kern · Exequiel Álvarez

Received: 17 September 2021 / Accepted: 8 April 2022 / Published online: 4 May 2022 © The Author(s) 2022

Abstract
This paper presents the design and implementation of four control strategies applied to a real underactuated manipulator robot with 3-DOF (Degrees of Freedom). Additionally, an original methodology for controlled oscillatory compensations is designed and implemented to mitigate the effect of a passive joint on the overall performance of this manipulator robot. The objective of this methodology is to create controlled oscillations that allow the faulty link and its (passive) joint to physically align with their adjacent previous link. The implemented control techniques are sinh–cosh, neural compensation, gain scheduling PID, and gain scheduling sinh–cosh. The real robot in which these four control strategies and oscillatory compensation methodology are implemented is a SCARA (Selective Compliant Assembly Robot Arm) robot. To assess controller performance—once the 3-DOF underactuated manipulator robot starts its trajectory—after $t = 4.5$ s, a fault is activated in its joint No. 2, converting it into a passive joint. The performance indicators IA (index of agreement), RMS (Root Mean Square), and RSD (Residual Standard Deviation) are used to analyze, compare, and evaluate the behavior of the four control strategies and the compensation methodology.

Keywords Underactuated robots · Optimization · Genetic algorithm · Passive joint · Non-linear control · MatLab/Simulink

Introduction
Currently, the robots most manufactured in the world are industrial manipulators, also called conventional robots. These robots are widely used in health, aerospace, and security, among other fields [1–3]. In the industrial field, these robots are still embedded in a rigidly programmed work structure. In this sector, fully actuated industrial robots are employed, which are equipped with a set of sensors that allows them to identify (internal and external) physical variables, and that acts as a control unit to properly coordinate the movement of their joints [4–6].

According to the standard ISO 10218-2, industrial robot systems must undergo a risk assessment prior to commissioning. In current industrial practice, risk assessments are conducted on the basis of experience, expert knowledge, and simple tools such as checklists [7].

The continuous advance of technology applied to industrial environments like the industry 4.0 has made hyper-connectivity, variable acquisition and Big Data possible in machines and smart systems. Manipulator robots are fundamental elements of manufacturing processes as their links follow Cartesian trajectories to perform tasks such as cutting, painting and deburring.

The logical link between controller devices and manipulator robots is based on industrial communication protocols like CAN1 bus. In general, there is high electromagnetic pollution in the industry that affects control links and causes data losses [8].

Therefore, in [9] the kinematic and dynamic model of a redundant SCARA manipulator robot is developed to assess three controllers applied to a robot subject to hostile work conditions.

1 Control Area Network.
Literature on the PID (Proportional–Integra–Derivative) control technique used for trajectory tracking in fully actuated industrial robots [10, 11] is extensive. However, the study of control techniques for underactuated industrial robots represents a new line of research, specifically their application in human–machine collaborative work environments [12].

Current studies on the dynamics of industrial robots are centered on adaptive control. The use of this type of control provides these robots with the necessary capacities to perform tasks in unknown environments, i.e., the ability to adapt their characteristic kinematics to such environments [13, 14].

Computed-torque control is a strategy that uses the robot’s input/output linearization to obtain a linearized and decoupled model of a fully actuated industrial robot. Subsequently, classic control strategies can be employed for each joint. Explicit matrices and vectors present in the modeled dynamics of the robots are used to achieve this goal [15, 16].

The development of industrial robots is increasingly relevant, as they surpass conventional robots in functions and versatility, with more DOF than the strictly necessary for the required tasks, which are often impossible for traditional robots. However, if any link of a conventional or redundant robot is damaged, the robot automatically becomes a human–machine robot. Underactuated robots—having more degrees of freedom than actuators—exhibit advantages over conventional robots because they are lighter, more compact, and consume less energy. Nevertheless, controller design for underactuated manipulator robots is much more complex than for conventional robots.

A fault in an industrial robot could occur due to the absence of control signal in its actuators, the presence of defective actuators and/or the lack of electricity supply in the components of these robots, among other causes. The effect of a faulty joint generates:

- Blocked joints (without movement).
- Free joints (with free movement).

Some of the techniques applied in fully actuated manipulator robots as well as manipulator robots underactuated due to a blocked joint are [17]:

- Control based on hyperbolic sine and cosine functions.
- Computed torque control.
- Fuzzy-logic control.
- Adaptive control.

On their own, control strategies applied to robot systems underactuated due to a blocked joint do not perform well in systems underactuated due to a free joint. Therefore, these strategies do not allow for performing a control action that limits the unwanted effects of a free joint. In [18], this problem is addressed, and control strategies are designed to compensate for the performance of a manipulator robot after a fault due to a blocked joint. Nevertheless, this article deals with a fault caused by a free joint due to the unexpected absence of electricity in the second joint of a manipulator robot. This type of fault manifests as a null control torque and is represented through the generalized forces vector expressed in Eq. (2). To mitigate the control loss of this joint—through controlled oscillations—a virtual link is generated between the links of the actuated joints No. 1 and No. 3. In this way, the initial original system—which has 3-DOF—will only have 2-DOF after the fault, which allows for locating the end effector in the vicinity of the desired target in the Cartesian plane through the third actuated joint. In these cases, when the gravitational forces vector affects perpendicularly the movement of the robot links, the force of gravity does not interfere in the dynamic performance of the robot.

Interest in underactuated systems began with the development of the controllers for 2 DOFs presented in [19–22], which employ the force of gravity. In these works, controllers are proposed to balance the links of acrobot (2-DOF underactuated planar robot with a passive actuator in the first joint) and pendubot (2-DOF underactuated planar robot with a passive actuator in the second joint) robots in their unstable equilibrium positions. Employing the gravity force, the control process is performed in two steps:

- The links are swung up to bring them to the vicinity of the desired position by oscillating the first link, achieving partial linearization by feedback.
- The robot is kept stable—in the balance position—using a linear technique for optimal feedback of the system state.

Thanks to the above, the links of the 2-DOF underactuated manipulator robots are kept within a local stability range. Other control techniques are presented in [22, 24], where both the concepts of rotational and translational kinetic energy were incorporated to the stages described above. In this way, the parameters corresponding to the linearization matrix of 2-DOF underactuated manipulator robots are identified by applying passivity.

The diverse control strategies exposed in [19, 20] and [23] have in common that they use potential energy to keep the end effector of 2-DOF underactuated manipulator robots at a desired stable position. Additionally, other control strategies based on Lyapunov functions have been proposed to control underactuated systems [25, 26].

The design and application of controllers has been fundamental to achieve stability in underactuated systems. In [27], a control strategy for first order sliding modes is presented, which is applied to 1- and 2-DOF systems. The
1-DOF system corresponds to a pendubot that uses the force of gravity to maintain an unstable equilibrium point, while the 2-DOF system—SCARA-type—has an actuated rotational joint. However, for industrial robots with joints that do not have direct drive joints, the sliding mode control is complex to implement, as it is based on activations and deactivations of the actuator’s power source, which causes a negative effect on them, mainly due to the use of high commutation frequencies. To counter this undesired effect, in [28] the design of a sliding surface with non-singular finite time is proposed, which allows for finite time stability that translates into enhanced trajectory tracking for the robot’s links. Additionally, to avoid developing and implementing a structurally complex control rule, time-delayed signals are used in the soft linearities. In another vein, in [29] an integral sliding mode algorithm is employed, whose commutation structure changes the architecture of the movement controllers of industrial robots. Thanks to the latter, the control efforts and saturation of actuators are reduced.

In synthesis, a first line of research on control techniques for underactuated robots—formed by links—has maintained the end effector in the vicinity of a desired position as its main objective, which is called position control. Nevertheless, this type of objective does not comprise maintaining the control of link trajectory. Therefore, considering the initial conditions of the robot, the best possible approximation of the end effector to a desired position is prioritized. However, a second line of research on control techniques for underactuated robots prioritizes the control of their link trajectory.

Currently, due to the level of automation, especially of industrial processes, the analysis and control of underactuated robots formed by links is growing day by day. Since most commercialized industrial robots are fully actuated, they work in mixed environments that require human operators, i.e., human–machine interaction, as these robots have not been manufactured to be fault tolerant. The main risks for a human being working with industrial robots under normal operating conditions or when these present some faults are:

(a) Collision or rear-end crashes.
(b) Projection of tools or objects manipulated by the end effector.
(c) Entrapment between robot links and an obstacle.
(d) Electrical failure internal or external to the robot.
(e) Fault of the robot actuators.
(f) Fault of the robot sensors.

The failure of a completely actuated industrial robot can leave some joints without direct drive joints, i.e., underactuated. If the possibility of underactuation is not considered in control systems, as it happens nowadays, accidents as the ones below may occur:

- In [30], accidents involving fully actuated robots in the mining industry—with human–machine interaction—are described, as shown in Table 1.
- It must be noted that, according to some studies conducted by Japan’s National Institute of Occupational Safety and Health (JNIOSH), 90% of the accidents in robot lines occur during maintenance, adjustment, and programming operations, whereas only 10% happen during normal line operation [31].
- According to the reports of the United States Occupational Safety and Health Administration (OSHA), over the last 30 years, industrial robots have caused at least 33 deaths and multiple injuries at work centers. In connection to this, Markoff and Claire [32] presents an account of accidents directly associated with the use of industrial robots in several countries, which are shown in Table 2.

In particular, Finland is a country that stands out due to its contributions to industrialization, as it uses manipulator robots in collaborative work. Additionally, Finland develops reports on accidents in which manipulator robots have been involved and creates databases of serious accidents that include deaths in the workplace associated with manipulator robots. These reports are written within the framework established by the Federation of Accident Insurance Institutions in Finland (FAII) and, according to the records of this institution, accidents are classified as severe and mortal, with the latter being the most serious [36]. Some major accidents reported by FAII are the following:

- 2000: While feeding cows in a farm, the head of an operator was crushed between a conveyor belt and the end effector of a manipulator robot.
- 2005: While packing bricks, the body of an operator was crushed between a transport vehicle and the end effector of a manipulator robot.
- 2006: The body of an operator was crushed between a transport vehicle and the end effector of a manipulator robot when the robot was placing trays on the conveyor belt of a dairy industry.

Based on the relevance of the information presented and considering the possibility of underactuation in the joints of industrial robots, the topic addressed in this article contributes to further advances in the design and implementation of observation and control smart systems for underactuated manipulator robots, such as multivariable analysis and optimization and synthesis and generalization of control systems. Moreover, an original methodology for controlled oscillatory compensations is designed and implemented to mitigate the effect of a passive joint on the overall performance of a real underactuated manipulator robot. Furthermore, four control strategies applied to a real underactuated manipulator robot
with 3-DOF are designed and implemented. To analyze, compare and assess the performance of these control strategies in the presence and in the absence of a controlled oscillatory compensation methodology, performance indexes are employed.

This article is organized as follows: The next section introduces the mathematical description of an underactuated robot and its dynamic model, as well as the desired Cartesian trajectory to be executed by a real underactuated manipulator robot. The subsequent section deals with the design of four control strategies and a controlled oscillatory compensation methodology for mitigating the effect of a passive joint on the global performance of a real underactuated manipulator robot. In the following section, the effect of a fault (e.g., passive joint) on a real underactuated manipulator robot is described when these four control strategies and the oscillatory compensation methodology are implemented, and the results of these implementations are analyzed. The penultimate section presents the performance indicators on which are based the analysis, comparison, and evaluation of the four control strategies, and oscillatory compensation methodology. Finally, conclusions are drawn.

**Description of a 3-DOF underactuated manipulator robot**

The manipulator robot under study is of the RRR (rotational–rotational–rotational) type. The joints of this robot are characterized by performing angular trajectories directly, i.e., without the need to perform inverse kinematics calculations, as reported in [25, 37]. Angular trajectory tracking is widely used for the analysis of control strategies applied in underactuated systems. However, the tracking of Cartesian trajectories performed by an RRR-type robot—such as the one proposed in this work—entails a high complexity in terms of angular trajectory as a Cartesian trajectory composed of segments has saturation points that must be analyzed through the calculation of inverse kinematics for a correct coding, for example, in MatLab/Simulink. Robotics applied to the manufacturing industry addresses tasks of great importance, such as MIG/MAG or arc welding; especially in the shipbuilding process, welding work-load accounts for 40% of the total workload [38]. Some examples of welding robots are Kuka ArcTech, Panasonic TM1400, Fanuc, and Arc Mate, which should describe trajectories in the Cartesian plane previously configured and even in three-dimensional segments. This shows the relevance of the analysis and implementation of Cartesian trajectory tracking by robots such as the one used in this work.

A diagram of the real manipulator robot tested is observed in Fig. 1. A redundant SCARA-type manipulator robot was selected as test bed, since by having a number of joints larger than the minimum required for performing a specific task this robot is able to form links that locate the end effector in the vicinity of the desired Cartesian positioning objective through the generation of a virtual link between active and passive links.

The mathematical fundamentals and the development of the dynamic model of the underactuated manipulator robot under study are introduced below.

In general, the dynamic model of manipulator robots with \( n \) joints can be expressed as follows [39]:

\[
\tau_i = \sum_{j=1}^{n} M_{ij}(q) \ddot{q}_j + \sum_{j=1}^{n} \sum_{k=1}^{n} C_{ijk}(q) \dot{q}_j \dot{q}_k + G_i(q) + F_i(q),
\]

where \( \tau \) represents the generalized forces vector, \( M(q) \) represents the inertia matrix, \( C(q, \dot{q}) \) represents the centrifugal
and Coriolis forces vector, \( \mathbf{G}(\mathbf{q}) \) corresponds to the gravitational forces vector, and \( \mathbf{F}(\dot{\mathbf{q}}) \), to the friction forces vector.

As shown in Fig. 1 the robot under study has a 3-DOF SCARA-type morphology, but redundant and with three rotational joints (RRR) in the \( x-y \) plane. The Denavit–Hartenberg parameters for this robot are shown in Table 3.

The physical parameters of the robot under study are presented in Table 4.

where \( I_{zz} \) represents the inertial momentum of the link \( i \) with respect to the first \( z \) axis of its joint [kg\cdot m\(^2\)], \( F_{vi} \) represents the viscous friction of the link \( i \) (clockwise) [N\cdot m\cdot s / rad], and \( F_{ecbi} \) represents the Coulomb friction of the link \( i \) (counterclockwise) [N\cdot m\cdot s / rad], \( i = 1, 2, 3 \).

Considering the second joint of the underactuated manipulator robot under study passive, i.e., \( \tau_2 = 0 \), from Eq. (1), it is obtained that

\[
\begin{bmatrix}
\tau_1 \\
0 \\
\tau_3
\end{bmatrix} =
\begin{bmatrix}
M_{11} & M_{12} & M_{13} \\
M_{21} & M_{22} & M_{23} \\
M_{31} & M_{32} & M_{33}
\end{bmatrix}
\begin{bmatrix}
\ddot{q}_1 \\
\ddot{q}_2 \\
\ddot{q}_3
\end{bmatrix}
+ \begin{bmatrix}
c_1 \\
c_2 \\
c_3
\end{bmatrix}
+ \begin{bmatrix}
f_1 \\
f_2 \\
f_3
\end{bmatrix}.
\] (2)

Therefore, the generalized forces are described in Eqs. (3)–(5).

\[
\begin{align*}
\tau_1 &= M_{11} \cdot \ddot{q}_1 + M_{12} \cdot \ddot{q}_2 + M_{13} \cdot \ddot{q}_3 + c_1 + f_1 \\
0 &= M_{21} \cdot \ddot{q}_1 + M_{22} \cdot \ddot{q}_2 + M_{23} \cdot \ddot{q}_3 + c_2 + f_2 \\
\tau_3 &= M_{31} \cdot \ddot{q}_1 + M_{32} \cdot \ddot{q}_2 + M_{33} \cdot \ddot{q}_3 + c_3 + f_3.
\end{align*}
\] (3, 4, 5)

From Eq. (5) the following angular acceleration is obtained for the underactuated joint \( \ddot{q}_2 \):

\[
\ddot{q}_2 = -M_{22}^{-1} \cdot (M_{21} \cdot \ddot{q}_1 + M_{23} \cdot \ddot{q}_3 + c_2 + f_2).
\] (6)

Replacing Eq. (6) in (3) and (4), it is obtained that

\[
\begin{align*}
\tau_1 &= M_{11} \cdot \ddot{q}_1 + M_{12} \\
&\quad \cdot \left( -M_{22}^{-1} \cdot (M_{21} \cdot \ddot{q}_1 + M_{23} \cdot \ddot{q}_3 + c_2 + f_2) \right) \\
&\quad + M_{13} \cdot \ddot{q}_3 + c_1 + f_1 \\
\tau_3 &= M_{31} \cdot \ddot{q}_1 + M_{32} \\
&\quad \cdot \left( -M_{22}^{-1} \cdot (M_{21} \cdot \ddot{q}_1 + M_{23} \cdot \ddot{q}_3 + c_2 + f_2) \right) \\
&\quad + M_{33} \cdot \ddot{q}_3 + c_3 + f_3.
\end{align*}
\] (7, 8)

To reduce the resulting expressions in Eqs. (7) and (8), based on common terms, Eqs. (9)–(12) are obtained.

\[
\begin{align*}
m_{11} &= M_{11} - M_{12} \cdot M_{22}^{-1} \cdot M_{21}, \\
m_{12} &= M_{13} - M_{12} \cdot M_{22}^{-1} \cdot M_{23}, \\
m_{21} &= M_{31} - M_{32} \cdot M_{22}^{-1} \cdot M_{21}, \\
m_{22} &= M_{33} - M_{32} \cdot M_{22}^{-1} \cdot M_{23}.
\end{align*}
\] (9, 10, 11, 12)

Therefore, based on the actuated joints of the robot and from Eqs. (9)–(12), the vector of generalized forces can be
obtained as presented in Eq. (13):

\[
\begin{bmatrix}
\tau_1 \\
\tau_3 \\
\end{bmatrix} = \begin{bmatrix}
m_{11} & m_{12} \\
m_{21} & m_{22} \\
\end{bmatrix} \cdot \begin{bmatrix}
\ddot{q}_1 \\
\ddot{q}_2 \\
\end{bmatrix} + \begin{bmatrix}
c_1 - M_{12} \cdot M_{22}^{-1} \cdot c_2 \\
c_3 - M_{32} \cdot M_{22}^{-1} \cdot c_2 \\
\end{bmatrix} + \begin{bmatrix}
f_1 - M_{12} \cdot M_{22}^{-1} \cdot f_2 \\
f_3 - M_{32} \cdot M_{22}^{-1} \cdot f_2 \\
\end{bmatrix},
\]

(13)

\[C_1 = c_1 - M_{12} \cdot M_{22}^{-1} \cdot c_2.\]

(14)

\[C_3 = c_3 - M_{32} \cdot M_{22}^{-1} \cdot c_2.\]

(15)

\[F_1 = f_1 - M_{12} \cdot M_{22}^{-1} \cdot f_2.\]

(16)

\[F_3 = f_3 - M_{32} \cdot M_{22}^{-1} \cdot f_2.\]

(17)

Finally, from Eqs. (13) to (17), the following dynamic model of the underactuated manipulator robot under study is obtained:

\[
\begin{bmatrix}
\ddot{q}_1 \\
\ddot{q}_2 \\
\end{bmatrix} = \begin{bmatrix}
m_{11} & m_{12} \\
m_{21} & m_{22} \\
\end{bmatrix}^{-1} \cdot \begin{bmatrix}
\tau_1 - C_1 - F_1 \\
\tau_3 - C_3 - F_3 \\
\end{bmatrix}
\]

(18)

Table 3 Denavit–Hartenberg parameters for the robot

| Joint | \(q_i\) | \(d_i\) | \(a_i\) | \(\alpha_i\) |
|-------|--------|--------|--------|--------|
| 1     | \(q_1^a\) | 0      | 0      | 0      |
| 2     | \(q_2^b\) | 0      | 0      | 0      |
| 3     | \(q_3^a\) | 0      | 0      | 0      |

\(^a\)Variables
\(^b\)Underactuated variable

Table 4 Physical parameters of the robot

| Link 1 | Link 2 | Link 3 | Units |
|--------|--------|--------|-------|
| \(l_1\) = 0.2 | \(l_2\) = 0.2 | \(l_3\) = 0.2 | [m] |
| \(l_{c1}\) = 0.0229 | \(l_{c2}\) = 0.0229 | \(l_{c3}\) = 0.0983 | [m] |
| \(m_1\) = 2.048 | \(m_2\) = 2.048 | \(m_3\) = 2.048 | [kg] |
| \(I_{zz}\) = 0.0116 | \(I_{zz}\) = 0.0116 | \(I_{3zz}\) = 01213 | [kg·m²] |
| \(F_{v1}\) = 0.025 | \(F_{v2}\) = 0.025 | \(F_{v3}\) = 0.025 | [N·m·s/rad] |
| \(F_{eca1}\) = 0.05 | \(F_{eca2}\) = 0.05 | \(F_{eca3}\) = 0.05 | [N·m] |
| \(F_{ecb1}\) = -0.05 | \(F_{ecb2}\) = -0.05 | \(F_{ecb3}\) = -0.05 | [N·m] |

Desired trajectory

When the generation of trajectories for manipulator robots is planned in the joint space, their control systems convert specifications—established in the workspace—into a set of desired values for the joint variables using inverse kinematics. To this end, the position, speed, and acceleration for each robot joint are specified. However, the tasks of manipulator robots are usually planned by users that, due to their familiarity with the environment, specify tasks in the Cartesian plane.

Figure 2 shows the desired Cartesian trajectory to be executed by the underactuated manipulator robot, whose starting and arrival points are (0.6; 0), and (0.2; 0.5), respectively. Additionally, this robot is redundant, which makes it possible to reconfigure the robot in the event of possible failures in one of its redundant links, expanding its workspace and enabling the elimination of some singular configurations inherent to manipulator robots in general.

Description of the fault in joint no. 2

After \(t = 4.5\) s, from the start of the 3-DOF underactuated manipulator robot trajectory, a fault is activated in joint No. 2 by interrupting the electric supply of its actuator, which corresponds to a zero-excitation signal in the actuator of that joint. The fault produced in joint No. 2 consists of disrupting the electricity supply to the servo actuator, making the mechanical torque of the actuator zero. Therefore, when joint No. 2 is in a free state (i.e., passive joint), link No. 2 receives the effects of the forces and torques imposed by the other links (No. 1 and No. 3), because their respective joints are active.
Design of control strategies

In general, the effect caused by a fault in a manipulator robot—due to the absence of power in a joint or the damage suffered in one of its actuators—transforms the robot into an underactuated robot, as mathematically modeled in Eqs. (2)–(18) for the 3-DOF underactuated manipulator robot under study.

The problem with an underactuated system such as the one considered in this work is that, using a traditional control sequence for a fully actuated system, the end effector will not meet the desired Cartesian positioning objective. Therefore, a controlled oscillatory compensation methodology is proposed that allows—by means of the actuated joint No.1—to drive a virtual link—formed by links No. 1 and No. 2—through an oscillation torque whose mathematical equation is described below:

\[
\tau_{osc} = A \cdot \text{sgn}(q_2) \sin(\omega t). \tag{19}
\]

To calculate the values of \(A\), \(\omega\), and \(k\), and considering the physical parameters of the manipulator robot, a compact expression of oscillation torque is employed, which corresponds to the following target function:

\[
\tau_{osc} = |A \cdot \sin(\omega) - k|. \tag{20}
\]

To conduct the analysis of complex system, metaheuristic algorithms are employed, which are a powerful tool to optimize fitness multivariate functions [38]. Then, the optimization technique based on Genetic Algorithms (GA) described below is used:

**Genetic Algorithm**

\[t:=0\]

the initial population is computed \(B_0 = (b_1, t, b_2, t, b_3, t, ..., b_m, t)\);

WHILE (the condition that is not met is stopped) DO.

BEGIN

FOR \(i:=1\) TO \(m\) DO

an individual is selected \(b_i, t+1\) from \(B_t\);

FOR \(i:=1\) TO \(m - 1\) DO STEP 2 DO

IF Random \([0, 1]\] \(\leq p_c\) THEN

Crosses \(b_i, t+1\) with \(b_{i+1}, t+1\);

FOR \(i:=1\) TO \(m\) DO

eventually mutate

offspring is created by crossing individuals;

they eventually mutate \(b_i, t+1\);

\[t:=t+1\]

END

where \(B_0\) corresponds to the initial population, \(b_i, t+1\) corresponds to individuals generated in a time \(t\) de from a list that ranges from “i” to “m”, \(p_c\) corresponds to the crossover
probability, and \( m \) to the size of the population. The structure of the genetic algorithm is based on four basic elements, namely selection, crossover, mutation and substitution.

Through this optimization technique, the value of the target function constants can be found when \( \tau_{\text{osc}} \to 0 \). To this end, inequalities need to be established that can be expressed in the following form:

\[
\sigma_i \cdot x_i \leq b_i, \quad (21)
\]

where \( \sigma_i \in \mathbb{R}^{n \times m} \) is a restriction matrix and \( b_i \in \mathbb{R}^{1 \times m} \) is a limit vector from which it is obtained that

\[
\begin{align*}
A & \leq 10 \\
\omega & \leq 9 \\
k & \leq 6 \\
-k & \leq -4 \\
A + \omega & \leq 15 \\
A - \omega & \leq 1 \\
-A + \omega & \leq 0.
\end{align*} \quad (22)
\]

Therefore, the extended form of expression (22) is

\[
\begin{bmatrix}
1 & 1 & 0 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
1 & -1 & 0 \\
-1 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & -1
\end{bmatrix}
\begin{bmatrix}
A \\
\omega \\
k
\end{bmatrix} =
\begin{bmatrix}
15 \\
10 \\
9 \\
1 \\
0 \\
6 \\
-4
\end{bmatrix} \quad (23)
\]

The values of \( x_i = [A\omega k]^T \), obtained after 68 iterations, are the following:

\[
\begin{align*}
A &= 7.906[n \cdot m] \\
\omega &= 6.981[\text{rad/seg}] \\
k &= 5.080[n \cdot m].
\end{align*} \quad (24)
\]

By replacing the values of (24) in (19), it is obtained that:

\[
\tau_{\text{osc}} = -7.906 \cdot \text{sgn}(q_2) \cdot \sin(6.981 \cdot t) + 5.080. \quad (25)
\]

Finally, from (24) and (25), the value of \( \omega \) is approximated to

\[
\omega \approx \frac{9 \cdot \pi \text{ rad}}{4 \text{ seg}}. \quad (26)
\]

Based on the direct interaction between link No. 1 (with active joint) and link No. 2 (with passive joint), a virtual link is created for the real 3-DOF underactuated manipulator robot (shown in Fig. 3).

The real underactuated manipulator robot in which the control strategies and controlled oscillatory compensations are implemented requires of the set of devices and electronic components specified in Fig. 4. The relay module and the current sensor create and identify a fault in a joint of this robot, as shown in Fig. 5. In turn, using information from the robot sensors, and using our methodology for controlled oscillatory compensations, the real manipulator robot is able to restructuring itself to be tolerant to the fault. The potence system, which allows for the interaction between servo actuators and across the control stages of the robot, is implemented using H-bridge integrated circuits and using an ATmega2560 data acquisition card.
Fig. 5 Logical detection of the electrical fault in the servo actuator No. 2

Fig. 6 a 3-DOF fully actuated manipulator robot, b 2-DOF fully actuated manipulator robot (3-DOF underactuated manipulator robot with a virtual joint between $q_1$ and $q_3$)
In this study, two test scenarios are considered to analyze the behavior of the real robot. The first one uses a 3-DOF fully actuated manipulator whose end effector should follow the Cartesian trajectory described in Fig. 2 at all moments. In the second scenario, 4.5 s after the trajectory tracking of the robot is initiated, an electrical fault is introduced to joint No. 2 through the deenergization of its servo actuator, as shown in Fig. 5; therefore, the 3-DOF fully actuated manipulator robot becomes 3-DOF underactuated manipulator robot with a virtual non-actuated joint between its two fully actuated links \((q_1 \text{ and } q_3)\). In this second scenario, the objective is to reduce the effect of the fault by reorganizing the entire robot control system in order to control the 2-DOF fully actuated manipulator robot, but this time, with a virtual non-actuated joint whose movement is restricted to a \(|q_2| \leq 10^\circ\) value, as observed in Fig. 6, which implies a new inverse kinematics.

Unfortunately, control analysis and synthesis for underactuated robots are now much more complex than in conventional robots. Consequently, it is necessary to design control systems that include passive joints but in the context of a 2-DOF manipulator robot, as presented in the control scheme of Fig. 7. Although the second link does not have an active joint, the robot under study—as a whole—still has to comply with the trajectory tracking requirements specified by the user, although it mostly has to maintain the end effector in the vicinity of a desired position [37].

Figure 8 shows the main blocks developed for the implementation of the control strategies and the methodology for controlled oscillatory compensations. In particular, “Fault activation” is the block where the activation of the fault that occurs in joint No. 2 is designed and programed, while in the block “Complementary filter”, the Inertial Measurement Units (IMUs) are encapsulated. The latter also receives an analog signal from the potentiometer of the servo actuator No. 2.

The design of a controlled oscillatory compensation methodology is proposed below to mitigate the effect produced by a passive joint on the overall performance of a real underactuated manipulator robot. The goal of this controlled oscillatory compensation is to generate controlled oscillations that enable the physical alignment of the link with the faulty joint with its previous adjacent link, which has no faults. To achieve this goal, robot links are imposed sustained oscillations—through active joints No. 1 and No. 3—to track the desired trajectory despite faulty joints (without actuation, i.e., passive joint), for example, joint No. 2. The condition to activate this oscillation torque is: \(|q_2| \geq 10^\circ\); the virtual link is then actuated, and the initial dynamics of the robot are reduced from 3-DOF to 2-DOF.

The methodology for controlled oscillatory compensations is described below:

Step 1: Trajectory tracking with 3-DOF actuated.
Step 2: In \(t = 4.5\) s, after the robot starts operation, the fault in joint No. 2 is activated (i.e., loose joint).
Step 3: Controlled oscillations are initiated in links No. 1 and No. 3.
Step 4: Virtual joint No. 2 \((q_2)\) is forced to stay at \(\pm 10^\circ\) with respect to joint No. 1 \((q_1)\), as presented in Sect. 3.1.
Step 5: An adaptation process 3-DOF \(\rightarrow\) 2-DOF is conducted, Fig. 9.

At the implementation level, switching allows for the activation of mitigation through controlled oscillation based on the information from the two possible logical values—0 or 1—,
that the variables “Osc” and “Failure Indicator” (FI) can take. This process operates in the following way: if there is a fault, then FI = 1 and if |q_2| ≥ 10°, then the controlled oscillation Osc = 1 is activated. In this way, the controlled oscillation will remain activated until the condition q_2 ≤ 10° is met. This switching activation is conducted through the “Selectors” blocks shown in Fig. 10, after which PWM\(^2\) signals are generated in the H-bridges that feed the servo actuators of joints q_1 and q_3.

The design and implementation of four control techniques for the robot under study are presented below. The main objective of these control techniques is to keep the end effector in the vicinity of a desired position. In addition, the parameters of each controller and the oscillation torque are specified.

### Sinh–cosh control strategy

A sinh–cosh controller is formed by a proportional part—based both on a hyperbolic sine and on a hyperbolic cosine—and a derivative part—based on a hyperbolic sine—as indicated in Eq. (27) [40] and as shown in the Fig. 11.

\[
\tau = K_p \sinh(q_e) \cosh(q_e) - K_v \sinh(q'),
\]

where K\(_P\) is the \(n \times n\) dimension proportional gain, K\(_V\) is the derivative gain of \(n\times n\) dimension, \(q_e\) is the error vector of positions of \(n\times 1\), \(q' = dq = \dot{q}\) is the first derivative of the angular position with respect to time, \(q_{i, pv}\) is the process variable, and \(q_{i, sp}\) is the set point variable of the \(i\)-th. Using the “tic-toc” command from MatLab/Simulink, the execution time of the sinh–cosh control strategy is 9.6 s.

Table 5 presents the gain values of the controller designed and implemented in the 3-DOF underactuated manipulator robot. The values of these gains were obtained through practical iterations.

---

\(^2\) Pulse Width Modulation.
Neurocompensated control strategy

Control strategies based on artificial neural networks play a key role in the controllability of non-linear dynamic systems [41–43]. The control strategy of a neural compensator starts from the learning process of an artificial neural network—through the recognition of parameters corresponding to the system’s dynamic—and a control strategy governing the manipulator robot. Figure 12 shows a scheme of the neurocompensated control strategy designed for the real 3-DOF underactuated manipulator robot. The strategy is composed of an artificial feed-forward neural network trained with backpropagation and contains 20 neurons in the first layer. This training process is conducted offline as the artificial neural network is previously trained based on a known controller. To conduct training, the sinh–cosh control strategy is employed. This neurocompensated control strategy is denominated compensated because it compensates for the action of the primary control strategy, in this case, the sinh–cosh control strategy.

Particularly, the control strategy designed and implemented for training the neural compensator of the real 3-DOF underactuated manipulator robot is sinh–cosh. Using the “tic-toc” command from MatLab/Simulink, the execution time obtained for the neurocompensated control strategy is 9.5 s.

The gain values of this controller’s tuning are presented in Table 6.

| Gain | $K_{p1}$ | $K_{p2}$ | $K_{p3}$ | $K_{v1}$ | $K_{v2}$ | $K_{v3}$ |
|------|----------|----------|----------|----------|----------|----------|
| Value | 9        | 7        | 10       | 0.003    | 0        | 0.00058  |
Gain scheduling control strategy

Gain scheduling is an adaptive control strategy that is widely used in nonlinear control [44]. The design and implementation of two gain scheduling control strategies—for the real 3-DOF underactuated manipulator robot—are presented below, which are based on dividing the action of a controller into segments. According to the considered error segment from trajectory tracking, independent values are assigned to controller gains.

Gain scheduling PID control strategy

The control law of this control strategy is presented in Eq. (28).

\[
\tau = K_p(\rho_n)e + K_I(\rho_n)\int e(\tau)d\tau + K_d(\rho_n)\frac{d}{dt}e(t).
\]

Gain values of the sinh–cosh controller used for training the artificial neural network

| Gain | \(K_p^1\) | \(K_p^2\) | \(K_p^3\) | \(K_v^1\) | \(K_v^2\) | \(K_v^3\) |
|------|--------|--------|--------|--------|--------|--------|
| Value | 5.5    | 6.5    | 6      | 0      | 0.00121| 0      |

In particular, the gain scheduling control technique is applied to each actuated link of the real 3-DOF underactuated manipulator robot. This strategy consists of assigning constant values within intervals—in this case, error intervals—for \(K_{p_i}(\rho_n)\), \(K_{I_i}(\rho_n)\) and \(K_{d_i}(\rho_n)\), \(i = 1, 2, 3\). Figure 13 presents a scheme of the gain scheduling PID control strategy designed for the 3-DOF underactuated manipulator robot. Using the “tic-toc” command MatLab/Simulink, the execution time obtained for the gain scheduling PID control strategy is 9.4 s.

Tables 7, 8, and 9 present the gain values for each joint and error range.
Table 7 Gain values for the control of joint No.1

| Gain | $e < -0.08$ | $-0.08 \leq e < 0.02$ | $e \geq 0.02$ |
|------|-------------|------------------|--------------|
| $K_{p1}(\rho_n)$ | 10          | 30               | 14           |
| $K_{i1}(\rho_n)$ | 0.002       | 0.007            | 0.003        |
| $K_{d1}(\rho_n)$ | 0.05        | 0                | 0            |

Table 8 Gain values for the control of joint No. 2

| Gain | $e < 0$ | $0 \leq e < 0.1$ | $e \geq 0.1$ |
|------|---------|------------------|-------------|
| $K_{p2}(\rho_n)$ | 15       | 20              | 6           |
| $K_{i2}(\rho_n)$ | 0.004    | 0               | 0.005       |
| $K_{d2}(\rho_n)$ | 0        | 0.07            | 0           |

Table 9 Gain values for the control of joint No.3

| Gain | $e < -0.06$ | $-0.06 \leq e < -0.02$ | $e \geq -0.02$ |
|------|-------------|------------------|--------------|
| $K_{p3}(\rho_n)$ | 3          | 10               | 2            |
| $K_{i3}(\rho_n)$ | 0.004      | 0.005            | 0            |
| $K_{d3}(\rho_n)$ | 0.007      | 0                | 0            |

Gain scheduling sinh–cosh control strategy

Figure 14 presents a scheme of the gain scheduling sinh–cosh control strategy designed for the real 3-DOF underactuated manipulator robot.

Using the “tic-toc” command from MatLab/Simulink, the execution time obtained for the gain scheduling sinh–cosh control strategy is 9.2 s.

The gain values for each joint and error range are presented in Tables 10, 11 and 12.

Results of the implementation of the real 3-DOF underactuated manipulator robot

Unlike in a fully actuated system, in general, the dynamic effects of a passive joint on an underactuated system are first that the Cartesian positioning objective is not fully met, and second that the desired values of minimum positioning errors are not achieved. Therefore, the best dynamic performance of an underactuated system is lower than the performance of a fully actuated system.

Figure 15 shows the scenario of the tests conducted on the real 3-DOF underactuated manipulator robot.

The results of the methodology for controlled oscillatory compensations that is applied in the real 3-DOF underactuated manipulator robot are presented below. Additionally, the results of the implementation of each controller designed and tuned are presented with the main purpose of keeping the end effector in the vicinity of a desired position.

Figures 16, 17, 18 and 19 show the behavior of the real 3-DOF underactuated manipulator robot during the execution of the specified Cartesian trajectories.

Figures 20, 21, 22 and 23 show the trajectory tracking error, i.e., the difference between the desired trajectory and the actual trajectory performed by the real 3-DOF underactuated manipulator robot.

Discussion

The implementation of robotized systems is a very complex process, as it requires knowledge of the dynamic behavior of the physical variables in such systems, the configuration of its components, and the design of both electronic and control devices that need tuning, among other data.

![Control loop of gain scheduling sinh–cosh, implementation scheme](image-url)
Particularly in the tests run in the real robot, sampling time was 4 ms. Table 13 shows the technical parameters and computational resources associated with the implementation of the four control strategies and the methodology for controlled oscillatory compensations.

The efficiency of each controller has been calculated considering the closest Cartesian point to the desired Cartesian position, whose mathematical representation is shown in Eq. (29).

\[
\text{Efficiency} = \frac{\min((x_{\text{real}}, y_{\text{real}}), (x_{\text{desired}}, y_{\text{desired}}))}{\max((x_{\text{real}}, y_{\text{real}}), (x_{\text{desired}}, y_{\text{desired}}))} \times 100 \in \mathbb{R}^2
\]

The experimental results presented in Figs. 16a, b, 17a, b, 18a, b, and 19a, b show a comparison of the positive effect of controlled oscillations, through which the end effector is placed in the desired Cartesian vicinity. In turn, the evolution of the angular error of actuated and underactuated joints can be found in Figs. 20a, b, 21a, b, 22a, b, and 23a, b, which show how error exhibits a trend towards zero \( \lim_{t \to \infty} e(t) \to 0 \) thanks to the application of the controlled oscillations methodology.

The quantification of the performance of each controller is presented below through the following statistical measures [42, 43]:

a. Index of Agreement (IA): An indicator of the trend of two series to be compared. Its values range between 0 and 1, where 1 indicates total adequacy and 0, total inadequacy.

\[
IA = 1 - \frac{\sum_{i=1}^{n} (o_i - p_i)^2}{\sum_{i=1}^{n} (|\hat{o}_i| - |\hat{p}_i|)^2}. \quad (30)
\]

b. Root mean square (RMS): Indicator of a series dispersion, which is normalized between 0 and 1.

\[
\text{RMS} = \sqrt{\frac{\sum_{i=1}^{n} (o_i - p_i)^2}{n}}. \quad (31)
\]

c. Residual Standard Deviation (RSD): Indicator of a series deviation, which is normalized between 0 and 1.

\[
\text{RSD} = \sqrt{\frac{\sum_{i=1}^{n} (o_i - p_i)^2}{\sum_{i=1}^{n} (o_i)^2}}. \quad (32)
\]

where: \( |\hat{o}_i| = o_i - o_m \) and \( |\hat{p}_i| = p_i - o_m \),

\( n \) : number of data observed, \( o_i \) : observed value \( i \) th, \( p_i \) : predicted value \( i \) th, \( o_m \) : Mean value of values observed.

Figures 24, 25 and 26 show the performance of each controller, without the application and with the application of the controlled oscillations methodology. Neurocompensated control strategy—applying the controlled oscillations methodology—exhibits the best control performance over the underactuated system. In fact, compared with all the other control strategies, it yields the smallest joint angular error on both axes.

**Conclusions**

This work presents the results of the design and implementation of an original methodology for controlled oscillatory compensations aimed at mitigating the effect produced by a passive joint on the performance of a real underactuated manipulator robot. In addition, four control strategies were designed and implemented in a real 3-DOF underactuated manipulator robot. By employing this original methodology, the joints of the real robot were controlled, and the four control strategies, i.e., sinh–cosh, neural compensation, gain scheduling PID, and gain scheduling sinh–cosh, were satisfactorily applied.

The graphic results of Cartesian trajectories—employing each control strategy—enabled for verifying how the original methodology for controlled oscillatory compensations mitigated the effect of a passive joint on the overall performance of the real robot.

Furthermore, the graphic results of trajectory tracking errors from the real 3-DOF underactuated manipulator robot also demonstrated a better overall performance thanks to the controlled oscillatory compensation methodology.
To analyze, compare and assess the performance of these control strategies in the presence and in the absence of a controlled oscillatory compensation methodology, performance indexes were employed. These indexes indicated a best overall performance in the real robot owing to the use of the controlled oscillatory compensation methodology.

The controlled oscillations methodology proposed and implemented allow for the generation of a virtual link between the links of the actuated joints No.1 and No.3. In this way, the initial original system with 3-DOF only had 2-DOF after the fault, facilitating the positioning of the end effector in the vicinity of the desired Cartesian positioning objective by means of the third actuated joint.

Several tests were conducted to tune the gains of both the controllers and the parameters employed in the mitigation methodology, thanks to which excellent control results were obtained. According to the efficiency percentages and
Fig. 17  
(a) Cartesian trajectories employing neural compensation control strategies and without application of controlled oscillations methodology, 
(b) Cartesian trajectories employing neural compensation control strategies and with application of controlled oscillations methodology.

Fig. 18  
(a) Gain scheduling PID control strategy without the application of controlled oscillations methodology, 
(b) Cartesian trajectories employing the gain scheduling PID control strategy with the application of the controlled oscillations methodology.
Fig. 19  
(a) Cartesian trajectories employing the gain scheduling sinh–cosh control strategy without the application of the controlled oscillations methodology,  
(b) Cartesian trajectories employing the gain scheduling sinh–cosh control strategy with the application of the controlled oscillations methodology.

Fig. 20  
(a) Trajectory tracking error when using the sinh–cosh control strategy without the application of the controlled oscillations methodology,  
(b) Trajectory tracking error when using the sinh–cosh control strategy with the application of the controlled oscillations methodology.
Fig. 21  a Trajectory tracking error when using the neural compensation control strategy without the application of the controlled oscillations methodology. b Trajectory tracking error when using the neural compensation control strategy with the application of the controlled oscillations methodology

Fig. 22  a Trajectory tracking error when using the gain scheduling PID compensation control strategy without the application of the controlled oscillations methodology. b Trajectory tracking error when using the gain scheduling PID compensation control strategy with the application of the controlled oscillations methodology
Fig. 23  a Trajectory tracking error when using the gain scheduling sinh–cosh compensation control strategy without the application of the controlled oscillations methodology, b trajectory tracking error when using the gain scheduling sinh–cosh compensation control strategy with the application of the controlled oscillation methodology

Table 13  Technical parameters and computational resources of the implementations

| Control strategy          | Efficiency (%) |
|---------------------------|----------------|
|                            | X axis         | Y axis         |
| Sinh–cosh                  | 80.3           | 93.8           |
| Neurocompensated           | 95.2           | 98.0           |
| Gain scheduling PID        | 79.0           | 98.8           |
| Gain scheduling sinh–cosh  | 89.5           | 92.6           |

The performance indexes calculated, the controllers that presented the best response to a fault—when applying the controlled oscillations methodology—are (in descending order): neurocompensated, sinh–cosh, gain scheduling PID, and gain scheduling sinh–cosh.

Finally, the processing time of each controller allows for comparing computational resources against the results of the Cartesian positioning of the end effector.

The topic addressed in this article will contribute to further advances in the design and implementation of observation and control smart systems for underactuated systems, such as multivariable analysis and optimization and synthesis and generalization of control systems.

Currently, the authors of this article are developing a mathematical model and a control strategy for generalizing and
Fig. 24  IA of trajectory tracking using the control strategies

Fig. 25  RMS of trajectory tracking using the control strategies
quantifying the performance of an $n$-DOF underactuated system and its control after a fault derived from blocked joints.

**Acknowledgements** This work was supported by the Vicerrectoría de Investigación, Desarrollo e Innovación of the University of Santiago of Chile.

**Declarations**

**Conflict of interest** On behalf of all the authors, the corresponding author states that there is no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

**References**

1. Ho HF, Wong Y-K, Rad AB (2007) Robust fuzzy tracking control for robotic manipulators. Simul Model Pract Theory 15:801–816
2. Safeea M, Neto P, Bearee R (2019) On-line collision avoidance for collaborative robot manipulators by adjusting off-line generated paths: an industrial use case. Rob Auton Syst 119:278–288
3. Urrea C, Kern J, López R (2020) Fault-tolerant communication system based on convolutional code for the control of manipulator robots. Control Eng Pract 101:104508. https://doi.org/10.1016/j.conengprac.2020.104508
4. Yahya S, Moghavvemi M, Mohamed HAF (2011) Geometrical approach of planar hyper-redundant manipulators: Inverse kinematics, path planning and workspace. Simul Model Pract Theory 19:406–422
5. Urrea C, Kern J (2016) Development of an electronic controller applied to a robotized manipulator. Comput Electr Eng 56:648–658. https://doi.org/10.1016/j.compeleceng.2016.01.020
6. Urrea C, Saa D (2020) Design and Implementation of a Graphic Simulator for Calculating the Inverse Kinematics of a Redundant Planar Manipulator Robot. Appl Sci 10:6770. https://www.mdpi.com/2076-3417/10/19/6770. Accessed 1 Sep 2021
7. Huck TP, Münch N, Hornung L et al (2021) Risk assessment tools for industrial human-robot collaboration: Novel approaches and practical needs. Saf Sci 141:105288. https://doi.org/10.1016/j.ssci.2021.105288
8. Yang F, Gu S (2021) Industry 4.0, a revolution that requires technology and national strategies. Complex Intell Syst 73(7):1311–1325. https://doi.org/10.1007/S40747-020-00267-9
9. Urrea C, Kern J (2012) Modeling, simulation and control of a redundant SCARA-type manipulator robot. Int J Adv Robot Syst. https://doi.org/10.5772/51701
10. Normey-Rico JE, Alcalá I, Gómez-Ortega J, Camacho EF (2001) Mobile robot path tracking using a robust PID controller. Control Eng Pract 9:1209–1214
11. Guha D, Roy PK, Banerjee S (2018) Optimal tuning of 3 degree-of-freedom Proportional–Integra–Derivative controller for hybrid distributed power system using dragonfly algorithm. Comput Electr Eng 72:137–153
12. Urrea C, Matteoda R (2020) Development of a virtual reality simulator for a strategy for coordinating cooperative manipulator robots using cloud computing. Rob Auton Syst 126:103447. https://doi.org/10.1016/j.robot.2020.103447
13. Wang H (2016) Adaptive control of robot manipulators with uncertain kinematics and dynamics. IEEE Trans Automat Contr 62:948–954
14. Nascimento JP, Brito NSD, Souza BA (2020) An adaptive over-current protection system applied to distribution systems. Comput Electr Eng 81:106545

15. Zubizarreta A, Marcos M, Cabanes I, Pinto C (2011) A procedure to evaluate extended computed torque control configurations in the Stewart-Gough platform. Rob Auton Syst 59:770–781

16. Viola J, Angel L (2018) Tracking control for robotic manipulators using fractional order controllers with computed torque control. IEEE Lat Am Trans 16:1884–1891

17. Urrea C, Kern J (2013) Fault-tolerant controllers in robotic manipulators. Performance evaluations. IEEE Lat Am Trans 11:1318–1324. https://doi.org/10.1109/TLA.2013.6710378

18. Urrea C, Kern J (2012) Modeling, simulation and control of 3-DOF redundant fault tolerant robots by means of adaptive inertia. In: Multi-Robot Systems, Trends and Development. http://www.intechopen.com/books/multi-robot-systems-trends-and-development. Accessed 12 Sep 2021

19. Fantoni I, Lozano R, Spong MW (2000) Energy based control of the pendubot. IEEE Trans Autom Contr 45:725–729

20. Shiriaev AS, Kolesnichenko O (2000) On passivity based control for partial stabilization of underactuated systems. In: Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187), vol 6. pp 2174–2179

21. Neetha P (2012) Effect of a novel control input on swing up of a pendubot. In: 2012 17th International Conference on methods & models in automation & robotics (MMAR), vol 6, pp 297–302

22. Xia D, Chai T, Wang L (2013) Fuzzy neural-network friction compensation-based singularity avoidance energy swing-up to nonequilibrium unstable position control of pendubot. IEEE Trans Control Syst Technol 22:690–705

23. Jun-Qing C, Xu-Zhi L, Min W (2015) Position control method for the pendubot. In: 2015 34th Chinese Control Conference (CCC), vol 7. pp 923–927

24. Buitrago D, Londoño G, technica AM-S et, 2006 undefined Técnicas híbridas de control aplicadas al pendubot. dialnet.unirioja.es

25. Wu J, Yang D, He X, Li X (2020) Finite-time stability for a class of underactuated systems subject to time-varying disturbance. Complex 2020:1–7

26. Oyarzún J, de Pinaga AB (2012) Desarrollo de robotica industrial en Codelco Chile. https://www.codelco.com/flipbook/innovacion/codelcodigital6/PDF/EXPOSICION/2/25p.pdf. Accessed 15 Sep 2021

27. Ovalle LR, Rios H, Llama MA (2019) Continuous sliding-mode control of position commanded robot manipulators. Control Eng Pract 81:183–198

28. Ferrara A, Incremona GP, Sangiovanni B (2019) Tracking control for robotic manipulators using fractional order controllers with computed torque control. IEEE Lat Am Trans 16:1884–1891

29. Markoff J, Claire M Robots en el trabajo. ¿un peligro para los humanos? https://www.elfinanciero.com.mx/new-york-times-syndicate/robots-un-peligro-para-los-humanos. Accessed 15 Sep 2021

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.