2025 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
Funchal, Portugal – April 2-3, 2025

Performance Comparison Between Position Controllers for a Robotic Arm Manipulator

João Braun
CeDRI
Polytechnic Institute of Bragança
Bragança, Portugal
0000-0003-0276-4314

Arezki Abderrahim Chellal
CeDRI
Polytechnic Institute of Bragança
Universidade de Trás-os-Montes e Alto Douro
Bragança, Portugal
0000-0002-9190-6865

José Lima
CeDRI
Polytechnic Institute of Bragança
Bragança, Portugal
INESC TEC
0000-0001-7902-1207

Vítor H. Pinto
SYSTEC (DIGI2)
University of Porto
Porto, Portugal
0000-0002-7840-0333

Ana I. Pereira
CeDRI
Polytechnic Institute of Bragança
Bragança, Portugal
0000-0003-3803-2043

Paulo Costa
INESC TEC
University of Porto
Porto, Portugal
0000-0002-4846-271X

2025 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) | 979-8-3315-3860-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICARSC65809.2025.10970174

***Abstract*—This paper compares five PID controller architectures for robotic manipulator position control, addressing the challenge of maintaining performance under varying inertial loads while providing accessible implementations for research and education. The five PID controller architectures for a three degrees-of-freedom SCARA manipulator position control are a basic Proportional-Derivative (PD), PD with Feed-Forward (FF), Parallel PD-PI-FF, Cascade PD-PI-FF, and Cascade PD-PI-FF with dead zone (DZ) compensation. The controllers were evaluated under varying inertial loads to assess robustness, extending beyond previous work's idealized conditions. Results show advanced configurations reduced errors by up to 64% compared to the baseline PD, with Parallel-FF achieving optimal dynamic performance and Cascade-FF-DZ excelling in steady-state control. The Feed-Forward addition enhanced tracking performance, while DZ compensation effectively eliminated limit cycles. The work provides open-source implementations and simulation environments, supporting research reproducibility and educational applications in robotics control.**

***Index Terms*—position controllers, SCARA, dead zone compensation, PID, control theory, robotic arm manipulator, education 4.0**

I. INTRODUCTION

Position control of robotic joints remains a fundamental challenge in robotics. While seemingly straightforward in theory, practical implementations face various non-linearities, such as motor dead zones and control signal saturation, alongside imperfect dynamic models that vary with environmental conditions [1]. This work extends previous work [2] by thoroughly analyzing various position controller architectures for a three degrees-of-freedom Selective Compliance Assembly Robot Arm (SCARA) manipulator. The prior work established baseline comparisons under idealized conditions; this study evaluates controller performance under more challenging scenarios. Specifically, the research examines how controllers perform when the plant model must handle operations across different joint configurations—effectively testing robustness against varying moments of inertia. The study implements and compares five controller architectures: a basic Proportional-Derivative (PD) controller, PD with Feed-Forward (FF), Parallel PD + Proportional-Integral (PI) + FF, Cascade PD+PI+FF, and Cascade PD+PI+FF with dead zone (DZ) compensation. Each configuration is tested using a cubic Hermite spline reference signal without a proper plant estimation to evaluate its robustness and performance characteristics. This paper offers two main contributions. First, it provides deeper insights into controller behavior under varying inertial loads, extending the simplified scenarios presented in [2]. Second, it serves as an educational resource by providing complete access to controller designs, implementations, and simulation environments. The remainder of this paper is organized as follows: Section II describes the state of the art in this area of educational robotics with different positional controllers' architectures. Section III presents the system's architecture, controllers' designs, and their implementation details. Section IV presents and analyzes the experimental results. Finally, Section V concludes the paper and suggests future research directions.

II. STATE OF THE ART

Recent developments in robotics and control systems emphasize the advancement of robust control methodologies and the importance of accessible educational platforms. In the control domain, [3] highlights the integration of traditional control theory with modern intelligent approaches, particularly for systems requiring real-time adaptation. This evolution in control strategies extends to domains like connected and automated vehicles, as demonstrated in [4], where platooning and traffic smoothing strategies are explored to optimize transportation efficiency. In the educational aspect of control systems and robotics, authors in [5] introduce a multi-level interface robot platform for research and education. In contrast, [6] demonstrates the effectiveness of using low-cost components for teaching motion control. Related to simulations for education, authors in [7] demonstrate how digital twins can enhance robotics development efficiency and learning. This approach aligns with [8], emphasizing the importance of accessible frameworks for experimentation and control system development. Simulation tools with structured educational methodologies are integrated in [9], providing an approach to robotics education. These works have advanced control systems and robotics education, but robust control assessment studies that follow open-source approaches for education are lacking. This research analyzes PID controller topologies under different inertial loads and provides implementations and simulation settings.

III. SYSTEM ARCHITECTURE

The implementation utilizes the SimTwo simulator [10], which uses the Open Dynamics Engine for high-performance rigid body dynamics simulation. SimTwo provides accurate dynamic behavior and realistic sensory errors [2], making it suitable for controller evaluation. Figure 1 shows the simulator's integrated development environment, featuring the 3D visualization window, data logging sheets, real-time plotting tools, and XML configuration editor for system parameters configuration, including a description of the simulation scene. These integrated tools provide robot visualization, data analysis, and system configuration. Thus enabling thorough testing and validation of control strategies.

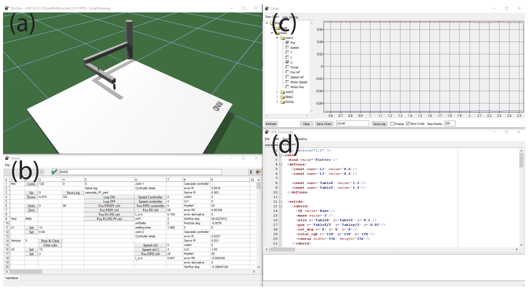

Fig. 1. SimTwo environment showing: (a) 3D visualization, (b) data sheets, (c) plotting tool, and (d) XML configuration editor.

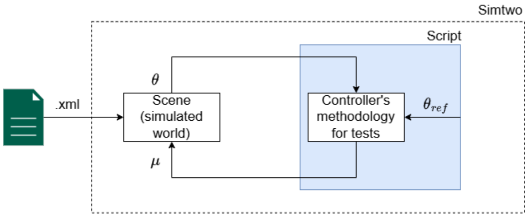

Fig. 2. Control architecture overview showing the interaction between simulator components.

The control architecture is shown in Figure 2, where μ represents the controllers' inputs and θ represents the system output. The control architecture operates with a 10 ms sampling period. It implements the motor model in Section III-A, the position reference signal in Section III-B, and the control architectures described in Section III-C.

A. Motor Model

The joints use 12V DC motor models[1] with parameters extracted using steady-state and transient response tests [7]. The simplified, neglecting the armature's inductance, the transfer function of the motor's angular speed $\omega(s)$ in relation to the input voltage $U(s)$ is:

$$H(s) = \frac{\omega(s)}{U(s)} = \frac{\frac{k}{BR_i+k^2}}{\frac{JR_i}{BR_i+k^2}s+1} = \frac{K_p}{\tau_s+1} \tag{1}$$

where k is the back e.m.f constant, B viscous friction, R_i armature resistance, and J is the moment of inertia of the motor plus load. The relation of two equations derived from the dynamic and electrical models of the DC motor was analyzed:

$$\frac{U}{\omega} = R_a\frac{i_a}{\omega} + k, \tag{2}$$

$$k\ i_a = B\ \omega + T_q, \tag{3}$$

where i_a is armature current and T_q the static friction. The motor parameters were obtained through steady-state tests using linear least squares optimization, as shown in Figures 3 and 4.

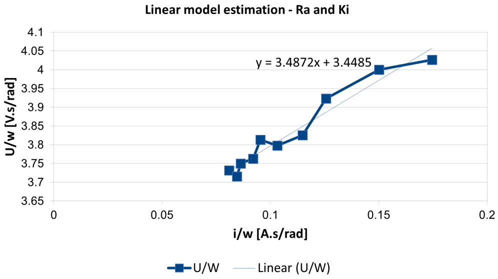

Fig. 3. Linear model estimation of armature resistance and back e.m.f constants.

Figure 3 displays the linear model estimation for armature resistance and back e.m.f constants. In contrast, Figure 4 estimates viscous and static friction constants through voltage and current measurements at various speed operating points, ignoring the motor's dead zone.

J was obtained by isolating it from the system's time constant, which was measured from the transfer function by performing an open loop step response. Therefore, the key motor parameters for joint one, estimated with link two at 90 degrees relative to link one, as shown in Figure 1, are shown in Table I.

The position of link two influences the J of the motor of joint one, and in this way, tests will be performed with link two in different positions, stressing the assessed controllers.

---

[1] Available at: https://www.robotshop.com/products/e-s-motor-dc-planetary-geared-motor-w-encoder-diameter-36mm-12v-18rpm

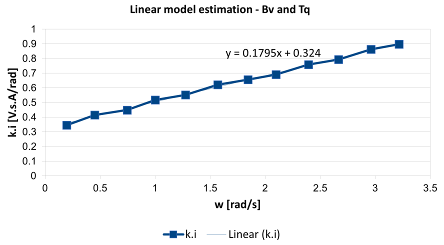

Fig. 4. Linear model estimation of viscous and static friction constants.

TABLE I
MOTOR PARAMETERS

Parameter	Value	Units
r_i	3.487	Ω
L_i	0	mH
k	3.448	$N.m/A$
V_{MAX}	12	V
I_{MAX}	5	A
$rotor\ j$	0.771	$kg.m^2$
$gear\ ratio$	1	-
B	0.179	$N.s/rad$
T_q	0.924	N
τ	0.215	s

B. Reference Signal

A cubic Hermite spline has been implemented to generate smooth reference position trajectories and is described in the following equation:

$$\begin{aligned}\theta(t) =&(2t^3 - 3t^2 + 1)p_0 + (t^3 - 2t^2 + t)m_0 + \\ &(-2t^3 + 3t^2)p_1 + (t^3 - t^2)m_1\end{aligned} \tag{4}$$

where p_0 and p_1 are the start and end points of the spline, and m_0 and m_1 are their respective derivatives (set to 0). A spline was designed to run approximately between $[0, 0.873]$ rad in 600 ms for the tests. This trajectory was used to stress-test the controllers while remaining within their design capabilities.

The reference trajectories are illustrated in Figure 5, which shows the position command, corresponding speed, and acceleration profiles. The speed and acceleration references are defined only during the transition interval, maintaining zero values outside this period. It is important to note that the position and speed references were plotted in the left y axis range while the acceleration signal was drawn following the right y axis range.

C. Controllers architectures

1) PD Controller: Due to its widespread industrial adoption, the PD controller represents the baseline architecture for comparison. Figure 6 illustrates its implementation, where $G(s)$ represents the PD controller and $H(s)$ the motor-load plant.

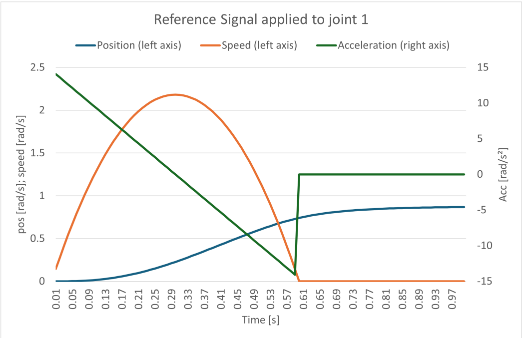

Fig. 5. Generated reference signals in the interval from 0 to 1s. The speed and acceleration profiles are null outside this interval. The left y axis range represents the position and speed functions, while the right y axis describes the acceleration profile.

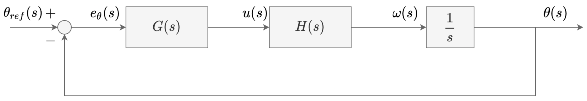

Fig. 6. PD position controller block diagram.

The integral term of the PID controller is not used as the control system inherently includes an integrator in the position calculation from encoder measurements. The closed-loop transfer function is:

$$\frac{\theta(s)}{\theta_{ref}(s)} = \frac{\frac{K_c K_p}{\tau}(1 + T_d s)}{s^2 + \frac{(1+K_c K_p T_d)}{\tau}s + \frac{K_c K_p}{\tau}} \tag{5}$$

where K_p represents the plant's open-loop gain, T_d the derivative time, K_c the controller gain, and τ the plant's open-loop time constant. The controller parameters were tuned using normalized Bessel polynomial roots [11] for a 400 ms settling time, yielding closed-loop poles at $p_{1,2} = -4.0530 \pm j2.3400$. This time was chosen so that the controller could follow the reference without lagging and be able to react quickly enough to changes in the reference.

2) PD with Feed-Forward: An FF block augments the PD controller by anticipating system behavior through inverse dynamics, as shown in Figure 7.

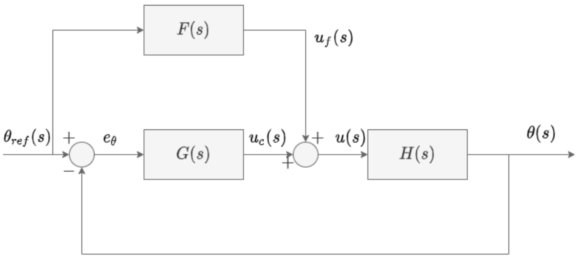

Fig. 7. Block diagram of PD-FF Controller.

The FF function, derived from the inverse plant model, is obtained by superposition:

$$\frac{\theta(s)}{\theta_{ref}(s)} = \frac{F(s)H(s) + G(s)H(s)}{1 + G(s)H(s)} \tag{6}$$

Given $H(s) = \frac{K_p}{(\tau s+1)s}$ and $F(s)H(s) = 1$, the FF control law in time domain becomes:

$$u_f(t) = \frac{1}{K_p}(\tau\ddot{\theta}_{ref}(t) + \dot{\theta}_{ref}(t)) \tag{7}$$

where $\ddot{\theta}_{ref}(t)$ and $\dot{\theta}_{ref}(t)$ represent the acceleration and speed of the reference signal, respectively.

3) Parallel PD-PI with Feed-Forward: The Parallel PD-PI with FF configuration combines the previous PD-FF design with a PI speed controller, as shown in Figure 8.

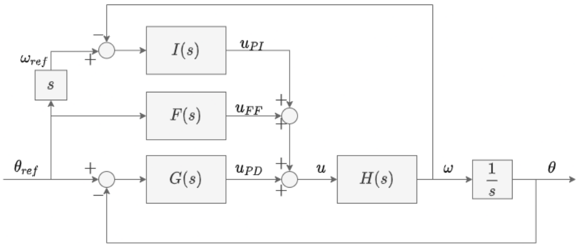

Fig. 8. Block diagram of the Parallel PD-PI controller with Feed-Forward.

The PI controller was designed using the Internal Model Control (IMC) method [12]. For the speed loop with $G(s)$ and $F(s) = 0$:

$$\frac{\omega(s)}{\omega_{ref}(s)} = \frac{I(s)H(s)}{1 + I(s)H(s)} \tag{8}$$

With $G(s)$ representing the PD controller, $I(s) = K_c(1 + \frac{1}{sT_i})$ and $H(s) = \frac{K_p}{\tau s+1}$, the IMC tuning rules ($K_c = \frac{\tau}{K_p\tau_{cl}}$ and $T_i = \tau$) yield:

$$\frac{\omega(s)}{\omega_{ref}(s)} = \frac{1}{\tau_{cl}s + 1} \tag{9}$$

The closed-loop time constant τ_{cl} was set to 250 ms, considering the motor's open-loop time constant from Table I. This configuration eliminates zeros from the closed-loop transfer function while maintaining unity gain.

4) Cascade PD-PI with Feed-Forward: The Cascade configuration modifies the previous parallel design as shown in Figure 9.

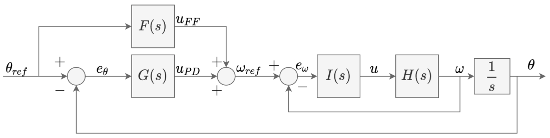

Fig. 9. Block diagram of the Cascade controller with Feed-Forward.

The inner PI loop maintains the same IMC design from Section III-C3 with $\tau_{cl} = 250$ ms. However, the outer PD loop's transfer function becomes:

$$\frac{\theta(s)}{\theta_{ref}(s)} = \frac{\frac{K_c}{\tau}(1 + T_d s)}{s^2 + \frac{(1+K_cT_d)}{\tau_{cl}}s + \frac{K_c}{\tau_{cl}}} \tag{10}$$

The PD controller uses normalized Bessel polynomial roots to maintain the 400 ms settling time design. Additionally, the FF block now accounts for the PI controller's closed-loop dynamics:

$$u_f(t) = \tau_{cl}\ddot{\theta}_{ref}(t) + \dot{\theta}_{ref}(t) \tag{11}$$

where $\ddot{\theta}_{ref}(t)$ and $\dot{\theta}_{ref}(t)$ represent the acceleration and speed of the reference signal, respectively.

D. Cascade PD-PI with DZ Compensation

This configuration addresses the limit cycle phenomenon caused by motor dead zones, which occurs when minimal control signals fail to generate motor movement, leading to error accumulation in integral control. Figure 10 shows the modified cascade architecture.

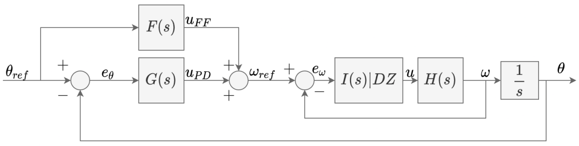

Fig. 10. Block diagram of the Cascade-FF controller with DZ compensation.

The DZ compensation is implemented as follows:

```
if (abs(Per) < MIN_ER && abs(speed_ref) < MIN_VEL)
{
  speed_ref = 0;
  Ierror = Ti*(offset_dz)/Kc;
  Volts_to_motor = 0;
}
```

where `Per` is position error, `MIN_ER` and `MIN_VEL` define minimum position and speed thresholds, and `offset_dz` represents the minimal effective motor voltage. The controller resets integral errors to maintain responsiveness while preventing oscillation.

E. IIR Filter Implementation

To compensate for the zero introduced by PD control, a first-order IIR filter processes the reference signal. The filter pole is computed as $e^{-(\frac{T_s}{T_d})}$, where T_s is the sampling period and T_d the derivative time constant.

IV. RESULTS

Tests evaluated the controllers' performance under varying inertial loads by positioning link two at 0°, 90°, and 180° while controlling joint one under the same reference signal described in Section III-B. Figures 11, 12, and 13 show these configurations.

These configurations represent different inertial loads on joint one, with 0° providing the maximum load, 90° matching the model estimation condition, and 180° providing the minimum load.

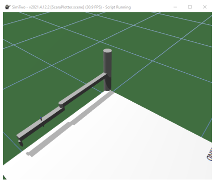

Fig. 11. Link two at 0°.

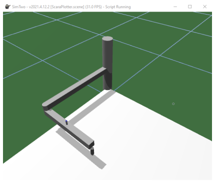

Fig. 12. Link two at 90°.

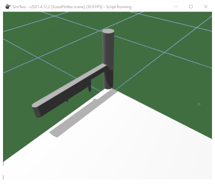

Fig. 13. Link two at 180°.

A. Test with link two at 0º

Figure 14 shows the controller responses under maximum inertial load. The Feed-Forward addition notably improved response compared to the basic PD, evidenced by reduced control signal saturation. The Parallel-FF, Cascade-FF, and Cascade-FF-DZ configurations demonstrated improved performance, more precise tracking, and reduced overshooting.

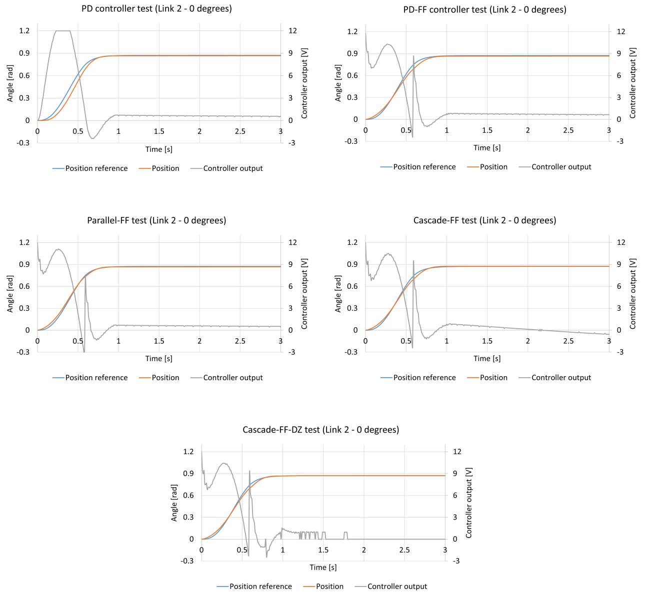

Fig. 14. Tests with the proposed controllers with link two at 0 degrees.

Figures 15 and 16 quantify these improvements, separating them in transient and steady states of the control, with advanced configurations showing significant error reduction during the transient phase and the Cascade-FF-DZ nearly eliminating steady-state error.

B. Test with link two at 90º

With link two positioned where the model of joint one was estimated, Figures 17 and 18 show the transient and steady states error for all controllers. Still, the advanced configurations maintained significant advantages, particularly in steady-state response where Cascade-FF-DZ demonstrated near-zero error.

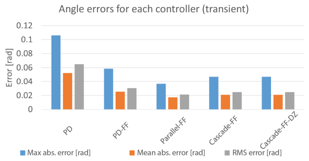

Fig. 15. Performance analysis of five controllers during the transient phase of the link two, 0-degree reference position test.

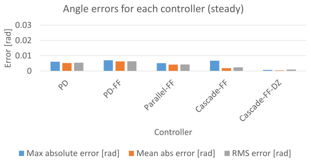

Fig. 16. Performance analysis of five controllers during the steady state phase of the link two, 0-degree reference position test.

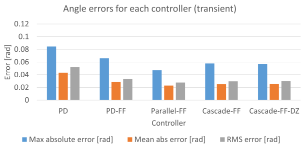

Fig. 17. Performance analysis of five controllers during the transient phase of the link two, 90-degree reference position test.

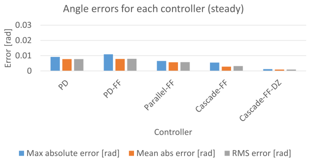

Fig. 18. Performance analysis of five controllers during the steady phase of the link two, 90-degree reference position test.

C. Test with link two at 180º

Under minimum inertial load, Figures 19 and 20 show the error comparison of all controllers. During the transient stage, all controllers performed similarly. However, this cannot be concluded when analyzing the steady state where the advanced architectures had the best results — specifically, the Cascade-FF-DZ significantly reduced the steady-state error.

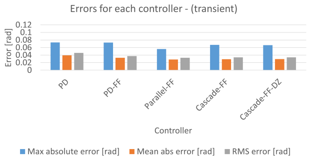

Fig. 19. Performance analysis of five controllers during the transient phase of the link two, 180-degree reference position test.

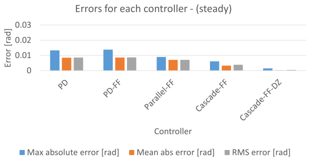

Fig. 20. Performance analysis of five controllers during the steady state of the link two, 180-degree reference position test.

D. Results Summary

Table II demonstrates the controllers' performance of the total error, i.e., including transient and steady states, across varying inertial loads, with error reductions ranging approximately from 15% to 64% compared to the baseline PD controller. The performance advantage of advanced configurations was most pronounced at maximum inertial load (0°) and converged as the load decreased (180°), highlighting their enhanced robustness to model variations.

The Feed-Forward addition significantly improved dynamic response, reducing control signal saturation while enhancing tracking performance by up to 48%. Advanced configurations are further built upon these improvements, with the Parallel-FF achieving the best overall dynamic performance through superior tracking and reduced overshooting. While showing

TABLE II
TOTAL ERROR (TRANSIENT + STEADY) PERFORMANCE COMPARISON ACROSS ALL CONTROLLER CONFIGURATIONS AND LINK TWO POSITIONS

		Link two at 0°			Link two at 90°			Link two at 180°		
Controller	Metric	Max	Mean	RMS	Max	Mean	RMS	Max	Mean	RMS
PD	Error [rad]	0.112	0.058	0.070	0.094	0.051	0.060	0.087	0.048	0.054
PD-FF	Percentage reduction [%]	-41.50	-45.00	-47.83	-18.15	-28.60	-31.25	0.12	-13.41	-15.29
Parallel-FF	Percentage reduction [%]	-62.73	-62.51	-63.71	-42.76	-43.70	-44.13	-25.14	-26.32	-26.18
Cascade-FF	Percentage reduction [%]	-52.23	-60.44	-61.40	-32.46	-45.26	-45.14	-15.93	-32.13	-30.25
Cascade-FF-DZ	Percentage reduction [%]	-57.60	-63.16	-63.46	-37.71	-48.47	-48.65	-22.16	-37.71	-36.25

similar transient characteristics, the Cascade-FF-DZ configuration demonstrated exceptional steady-state performance by effectively eliminating limit cycles and minimizing steady-state error. These results indicate that while all advanced configurations substantially outperform the baseline PD controller, their relative advantages become most evident under challenging inertial conditions. The Parallel-FF configuration emerges as the optimal choice for applications prioritizing dynamic performance. At the same time, the Cascade-FF-DZ offers superior steady-state characteristics, which are particularly valuable in precision positioning applications.

V. CONCLUSION

This work presented a comparative analysis of PID controller architectures for SCARA manipulator position control under varying inertial loads. Building upon prior research [2], the study evaluated five configurations: PD, PD-FF, Parallel PD-PI-FF, Cascade PD-PI-FF, and Cascade PD-PI-FF with DZ compensation. The results demonstrated significant improvements over the baseline PD controller, with error reductions of up to 64% for advanced configurations. The Feed-Forward addition provided substantial benefits in dynamic response, while Parallel and Cascade architectures offered enhanced robustness to model variations. The DZ compensation effectively eliminated steady-state oscillations, which is particularly valuable for precision positioning applications. The key contributions include the thorough performance analysis under varying inertial loads and the validation of advanced control architectures' robustness. Additionally, the open-source implementation supports educational use and reproducibility, providing a valuable resource for the robotics control community. Future work could explore extension to multi-joint simultaneous control, implementation on hardware platforms, integration with adaptive control techniques, energy analysis, and application to other robotic configurations. The complete simulation environment, including controller implementations and test scenarios, is available at this GitHub repository[2], facilitating reproducibility, research extension, and educational applications in robotics control.

ACKNOWLEDGMENT

This work was supported by national funds through FCT/MCTES (PIDDAC): CeDRI, UIDB/05757/2020 (DOI: 10.54499/UIDB/05757/2020) and UIDP/05757/2020 (DOI: 10.54499/UIDB/05757/2020); SusTEC, LA/P/0007/2020 (DOI: 10.54499/LA/P/0007/2020). Arezki A. Chellal is grateful to the FCT Foundation for its support through the FCT PhD scholarship with reference UI/BD/154484/2022.

REFERENCES

[1] M. Spong, S. Hutchinson, and M. Vidyasagar, "Robot modeling and control," *IEEE Control Systems*, vol. 26, pp. 113–115, dec 2006.

[2] A. P. Moreira, J. Lima, and P. Costa, "Improving a position controller for a robotic joint," in *2021 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pp. 97–103, IEEE, 2021.

[3] I. Zaitceva and B. Andrievsky, "Methods of intelligent control in mechatronics and robotic engineering: A survey," *Electronics*, vol. 11, no. 15, p. 2443, 2022.

[4] B. Chalaki, L. E. Beaver, A. I. Mahbub, H. Bang, and A. A. Malikopoulos, "A research and educational robotic testbed for real-time control of emerging mobility systems: From theory to scaled experiments [applications of control]," *IEEE Control Systems Magazine*, vol. 42, no. 6, pp. 20–34, 2022.

[5] S. Haddadin, S. Parusel, L. Johannsmeier, S. Golz, S. Gabl, F. Walch, M. Sabaghian, C. Jähne, L. Hausperger, and S. Haddadin, "The franka emika robot: A reference platform for robotics research and education," *IEEE Robotics & Automation Magazine*, vol. 29, no. 2, pp. 46–64, 2022.

[6] F. Sanfilippo, M. Økter, T. Eie, and M. Ottestad, "Teaching motion control in mechatronics education using an open framework based on the elevator model," *Machines*, vol. 10, no. 10, p. 945, 2022.

[7] J. Braun, J. Lima, P. Costa, and A. Moreira, "Robot@factory lite competition: A digital twin approach for the agv," in *Proceedings of the 11th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH*, pp. 311–318, INSTICC, SciTePress, 2021.

[8] A. Carron, S. Bodmer, L. Vogel, R. Zurbrügg, D. Helm, R. Rickenbach, S. Muntwiler, J. Sieber, and M. N. Zeilinger, "Chronos and crs: Design of a miniature car-like robot and a software framework for single and multi-agent robotics and control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1371–1378, 2023.

[9] R. E. Patiño-Escarcina, D. Barrios-Aranibar, L. S. Bernedo-Flores, P. J. Alsina, and L. M. Gonçalves, "A methodological approach to the learning of robotics with edurosc-kids," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 2, p. 34, 2021.

[10] J. Lima, R. B. Kalbermatter, J. Braun, T. Brito, G. Berger, and P. Costa, "A realistic simulation environment as a teaching aid in educational robotics," in *2022 Latin American Robotics Symposium (LARS), 2022 Brazilian Symposium on Robotics (SBR), and 2022 Workshop on Robotics in Education (WRE)*, pp. 430–435, 2022.

[11] R. Vaccaro, *Digital Control: A State-space Approach.* McGraw-Hill series in electrical and computer engineering: Control theory, McGraw-Hill, 1995.

[12] I.-L. Chien, "Imc-pid controller design-an extension," *IFAC Proceedings Volumes*, vol. 21, no. 7, pp. 147–152, 1988.

---

[2] https://github.com/jbneto1/pid-architectures-robotic-manipulator