Indonesian Journal of Electrical Engineering and Computer Science
Vol. 41, No. 1, January 2026, pp. 90∼98
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v41.i1.pp90-98

Comparative analysis of fractional-order sliding mode and pole placement control for robotic manipulator

Ahmed Bennaoui[1], Salah Benzian[2], Idrees Nasser Alsolbi[3], Aissa Ameur[1]
[1]Institute of Sciences, University Center Aflou El Cherif Bouchoucha, Aflou, Algeria
[2]Department of Data Science, College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia
[3]Faculty of Technology, University of Amar Telidji, Laghouat, Algeria

Article Info

Article history:

Received Oct 19, 2025
Revised Nov 4, 2025
Accepted Dec 13, 2025

Keywords:

Control effort
Fractional-order sliding mode control
Lagrangian mechanics
Pole placement control
Robotic manipulator
Robustness
Trajectory tracking

ABSTRACT

Fractional-order sliding mode control (FOSMC) is benchmarked against pole placement control (PPC) on a nonlinear two-link manipulator subjected to identical trajectories and 10 N·m square disturbances. Quantitative head-to-head evidence against industrial PPC is scarce, leaving engineers uncertain when fractional designs justify their added complexity. We derive the plant via Lagrange dynamics, implement both controllers in Python, and evaluate tracking and torque effort using SciPy-based simulations. Under the adopted fractional derivative approximation, FOSMC attains RMSEs of 0.458 rad (q_1) and 0.453 rad (q_2) whereas PPC limits the errors to 0.365 rad and 0.337 rad. The fractional design, however, requires lower mean torques of 69.2/29.0 N·m compared to PPC's 86.1/41.4 N·m, exposing a precision–energy trade-off that now favours PPC on accuracy and FOSMC on actuation effort. The benchmark delivers deployable evidence that fractional sliding surfaces shift torque demand even when their tracking performance lags, and it motivates hardware-in-the-loop validation to close the identified accuracy gap.

Corresponding Author:

Ahmed Bennaoui
Institute of Sciences , University Center Aflou El Cherif Bouchoucha
Aflou- 03001, Algeria
Email: a.bennaoui@lagh-univ.dz

1. INTRODUCTION

Seminal assessments of manipulator dynamics underscore the enduring importance of robust robot control [1]. Robotic manipulators support precision manufacturing, surgery, and autonomous systems, yet their coupled nonlinear dynamics complicate linear feedback design [2], [3]. Classical controllers such as PID and pole placement control (PPC) are attractive for their simplicity, but their performance degrades in the presence of parameter variations, disturbances, and joint constraints, while high-gain nonlinear alternatives risk actuator stress and chattering [4].

Computed-torque and adaptive baselines remain standard references for robot control tuning [3]. Recent surveys on sliding-mode and fractional-order design catalogue delay, estimation, and robustness challenges that underline the need for reproducible benchmarks [5].

Classical sliding-mode theory and its modern refinements remain the backbone for robust robot control design [6]-[8]. Broad robotics primers likewise codify the modeling assumptions and feedback architectures adopted here [9], [10]. On the fractional side, foundational texts and frequency-domain approximations continue to motivate non-integer controllers for manipulators [11]-[15].

Fractional-order control introduces non-integer calculus into the feedback path, enabling smoother sliding dynamics without sacrificing disturbance rejection [16]-[25]. Recent fractional sliding-mode designs report strong robustness gains, yet surveys note that matched-condition benchmarks against industrial PPC are rarely documented, leaving a practical guidance gap for engineers [5]. Without quantitative evidence that isolates accuracy, torque effort, and settling behaviour under the same plant, practitioners cannot decide when fractional designs justify their added implementation complexity.

To close this gap, we derive a two-degree-of-freedom (2-DOF) manipulator via Lagrange dynamics, implement FOSMC and PPC in a single Python workflow, and challenge both controllers with sinusoidal references plus a 10 N·m square disturbance between $t = 2$ s and $t = 3$ s. Performance is evaluated using root mean square error (RMSE), mean torque, and transient settling metrics that align with industrial specifications. The main contributions of this paper are summarized as follows:

– A fully reproducible Python workflow (supplementary file S1) that implements both controllers with matched dynamics, disturbance window, solver tolerances, and post-processing;
– A quantitative comparison between FOSMC and PPC under identical operating conditions using Python simulations, yielding the RMSE and mean-torque metrics cited throughout the manuscript;
– A clarified accuracy–torque trade-off analysis that explains when PPC's precision outweighs FOSMC's actuation savings for surgical robotics, industrial automation, and collaborative systems.

Recent FOSMC studies concentrate on fractional sliding-surface design, adaptive observers, and fixed-time parameter tuning, yet they seldom provide a reproducible, side-by-side benchmark against the pole-placement controllers still prevalent in industrial deployments [17]-[20]. Rather than proposing another derivative of the fractional surface, this work positions itself as the missing empirical bridge between fractional and industrial controllers by i) rigorously matching plant model, disturbance window, reference trajectories, and numerical tolerances across FOSMC and PPC; ii) quantifying the accuracy–energy trade-off with identical RMSE, torque, and settling metrics; and iii) releasing an open Python workflow (supplementary file S1) that future studies can extend toward hybrid or learning-enhanced control schemes. This framing aligns the manuscript with current trends that emphasise deployable benchmarks and transparent robustness–energy reporting, thereby clarifying the novelty relative to textbook derivations.

2. PROPOSED METHOD

This section details the modeling foundation and controller formulations that constitute the proposed benchmarking workflow. The 2-DOF manipulator's dynamics, derived using Lagrangian mechanics, are expressed as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{d}(t) \tag{1}$$

Where $\mathbf{q} = [q_1, q_2]^T$ is the joint angle vector, $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis/centripetal matrix, $\mathbf{G}(\mathbf{q})$ is the gravity vector, $\boldsymbol{\tau}$ is the control torque, and $\mathbf{d}(t)$ is the disturbance. The manipulator has links of length $L_1 = L_2 = 1$ m, mass $M_1 = M_2 = 1$ kg, and gravity $g = 9.8$ m/s^2. Desired trajectories are $q_{d1} = \sin(4.17t)$, $q_{d2} = 1.2\sin(5.11t)$, with a disturbance $\mathbf{d}(t) = [10, 10]^T$ N.m (t=2–3 s). This canonical representation follows established nonlinear manipulator modeling and Lyapunov-based stability analysis practices [21].

2.1. Dynamic model

The inertia matrix $\mathbf{M}$, Coriolis/centripetal vector $\mathbf{C}$, and gravity vector $\mathbf{G}$ are defined as:

$$\begin{aligned}
m_{11} &= (M_1 + M_2)L_1^2 + M_2L_2^2 + 2M_2L_1L_2\cos(q_2),\\
m_{12} &= m_{21} = M_2L_2^2 + M_2L_1L_2\cos(q_2),\\
m_{22} &= M_2L_2^2,\\
c_1 &= -M_2L_1L_2\sin(q_2)(2\dot{q}_1\dot{q}_2 + \dot{q}_2^2),\\
c_2 &= M_2L_1L_2\sin(q_2)\dot{q}_1^2,\\
g_1 &= -(M_1 + M_2)gL_1\sin(q_1) - M_2gL_2\sin(q_1 + q_2),\\
g_2 &= -M_2gL_2\sin(q_1 + q_2).
\end{aligned}$$

These sine-based gravity torques mirror the expressions embedded in supplementary file S1, keeping the analytical description and executable workflow synchronized.

2.2. FOSMC design

FOSMC uses a sliding surface:

$$s_i = \dot{e}_i + \alpha D^{1.5} e_i + \gamma e_i^{\beta}, \quad i = 1, 2 \tag{2}$$

where $e_i = q_i - q_{di}$, and the fractional derivative $D^{1.5} e_i \approx 5 e_i$ [17], [19]. The control law is:

$$\boldsymbol{\tau} = \mathbf{M}\left(\ddot{\mathbf{q}}_d - \mathbf{f} - \mathbf{FF} - K_s \mathbf{S} + \mathbf{DD}\right) + \mathbf{C}\dot{\mathbf{q}} + \mathbf{G} \tag{3}$$

with $\mathbf{f} = \mathbf{M}^{-1}(-\mathbf{C}\dot{\mathbf{q}} - \mathbf{G})$, $\mathbf{FF} = [\alpha\mu D^{0.5} e_1, \alpha\mu D^{0.5} e_2]^T$, $\mathbf{DD} = [-\beta\gamma\dot{e}_1 e_1^{\beta-1}, -\beta\gamma\dot{e}_2 e_2^{\beta-1}]^T$, $K_s = 10$, $\alpha = 4$, $\gamma = 9$, $\beta = 3$, $\mu = 1.5$.

2.3. PPC design

PPC uses computed torque control:

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{u}_{\text{ppc}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{G} \tag{4}$$

where $\mathbf{u}_{\text{ppc}} = \ddot{\mathbf{q}}_d - \mathbf{K_d}\dot{\mathbf{e}} - \mathbf{K_p}\mathbf{e}$, with $\mathbf{K_p} = \text{diag}(300, 300)$, $\mathbf{K_d} = \text{diag}(20, 20)$ [22], [25].

3. METHOD

The dynamics and controllers for the 2-DOF robotic manipulator were implemented in Python 3.8, and the complete runnable script is now hosted in Supplementary Listing S1 (`supplementary_code.tex`). That document compiles the imports, `TwoDOFRobot` class, simulation driver, plotting routines, and animation pipeline that underpin the workflow summarized here. Simulations were conducted using `scipy.integrate.solve_ivp` (SciPy 1.7.3) with the RK45 solver, a maximum time step of 0.01 s, and relative and absolute tolerances of 10^{-6} [23]. The simulation spanned 5 seconds with initial conditions $[q_1, q_2, \dot{q}_1, \dot{q}_2] = [-\pi, \pi, 0, 0]$ rad, rad/s. A square wave disturbance of 10 N.m was applied to both joints from $t = 2$ s to $t = 3$ s to evaluate robustness [24]. Simulation data was interpolated onto a 500-point uniform grid using `scipy.interpolate.interp1d` for consistent analysis. Table 1 consolidates the numerical and controller parameters together with their selection rationale, while the subsequent paragraphs detail the numerical validation workflow.

Table 1. Controller and numerical parameters with associated rationale

Parameter	Value	Role	Justification
α	4	FOSMC derivative gain	Maintains steep sliding slope without amplifying sensor noise.
γ	9	FOSMC nonlinear gain	Provides 15% overshoot margin during the 10 N·m disturbance.
β	3	Sliding polynomial order	Supplies cubic stiffness that removes steady bias.
μ	1.5	Fractional feedforward scaling	Matches attenuation predicted in [17].
K_s	10	Sliding surface gain	Guarantees < 1 s convergence per [20].
glfdiff gain	5	Fractional placeholder	Approximates Grünwald–Letnikov behaviour while limiting chattering.
$\mathbf{K_p}$	diag(300,300)	PPC proportional gain	Places poles at $\omega_n \approx 17$ rad/s [25].
$\mathbf{K_d}$	diag(20,20)	PPC damping gain	Produces damping ratio $\zeta \approx 0.8$.
r_{tol}, a_{tol}	$10^{-6}, 10^{-8}$	RK45 tolerances	Keep numerical error $< 10^{-4}$ rad vs tighter runs.
$\Delta t_{\max}$	0.01 s	Max solver step	Resolves the 5 Hz reference without aliasing.
Disturbance	10 N·m (2–3 s)	Robustness stress test	Replicates the payload surge in [24].
Initial state	$[-\pi, \pi, 0, 0]$	Start condition	Excites both joints with opposing deflections.
Grid size	500 samples	Post-processing	Aligns RMSE/torque metrics on a shared time base.

Numerical validation proceeded in three stages. First, the RK45 solver was re-run with $r_{tol} = 10^{-7}$ and $a_{tol} = 10^{-9}$; deviations from the nominal trajectories remained below 4×10^{-5} rad and 6×10^{-3} N·m, confirming tolerance sufficiency. Second, disturbance-free simulations ($t < 2$ s) were compared against a symbolic linearized model to verify that the PPC gains produced the targeted $\zeta = 0.8$ and $\omega_n = 17$ rad/s poles.

Third, the 500-point interpolated grid was cross-checked against the native solver timestamps, and the resulting RMSE changed by fewer than 0.5%, validating the resampling pipeline.

The initial condition vector $[-\pi, \pi, 0, 0]$ places the links at opposing extremes with zero velocity so both controllers must overcome coupled inertia before the disturbance. A 1 s warm-up interval allows the references to synchronize prior to the 10 N·m pulse between 2–3 s, which stresses robustness and chattering mitigation. All state, torque, and reference arrays are logged before interpolation, enabling reproducible recomputation of RMSE and mean torque as defined below.

Performance was assessed using RMSE for tracking accuracy and mean absolute torque for control effort, defined as:

$$\text{RMSE}_i = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(q_i(t_k) - q_{di}(t_k))^2}, \quad i = 1, 2 \tag{5}$$

$$\text{Mean Torque}_i = \frac{1}{N}\sum_{k=1}^{N}|\tau_i(t_k)|, \quad i = 1, 2 \tag{6}$$

where $q_i(t_k)$ and $q_{di}(t_k)$ are the actual and desired joint angles at time t_k, $\tau_i(t_k)$ is the control torque, and $N = 500$ is the number of time steps [3]. Controller parameters were tuned empirically: FOSMC parameters ($\alpha = 4$, $\gamma = 9$, $\beta = 3$, $\mu = 1.5$, $K_s = 10$) were selected to balance robustness and chattering reduction [19], while PPC gains ($\mathbf{K_p} = \text{diag}(300, 300)$, $\mathbf{K_d} = \text{diag}(20, 20)$) were chosen to ensure stable pole placement [25]. The simulation setup was validated against established robotic control benchmarks [2], [4].

Executing `python main.py` from the project root replays the workflow end-to-end: it runs both controllers with the 10 N·m disturbance injected between 2–3 s, interpolates the trajectories onto the 500-point grid, saves Figure 1 structure of a 2-DOF robot manipulator and Figures 2–5 plus the animation, and prints the RMSE and mean-torque statistics compiled in Table 2. This command therefore keeps the manuscript narrative synchronized with the artifacts preserved in Supplementary File S1.

4. RESULTS AND DISCUSSION

Simulations demonstrate that PPC now provides tighter tracking envelopes with RMSE values of 0.365 rad and 0.337 rad for q_1 and q_2, respectively, while FOSMC settles at 0.458 rad and 0.453 rad (Table 2), Figure 2). Figures 2(a)-(b) shows the PPC error bands remaining narrower throughout the disturbance window, whereas FOSMC's fractional surface, approximated via a high-gain placeholder, leaves wider residuals and slower convergence in Figure 3. Despite this accuracy penalty, FOSMC commands lower mean torques of 69.2 N·m and 29.0 N·m (Figures 4(a)-(b)) relative to PPC's 86.1 N·m and 41.4 N·m, demonstrating that the fractional sliding surface can shave actuator effort even when it does not outperform linear pole placement on tracking. Joint velocities still remain comparatively smooth under the sliding-mode action during the 10 N·m disturbance (Figures 5(a)-(b)), indicating that the fractional dynamics damp high-frequency chattering at the cost of additional steady-state bias.

These head-to-head metrics are significant for robotics researchers and integrators because they translate abstract fractional-order concepts into the actuator torques, settling times, and RMSE thresholds used in manufacturing and surgical benchmarks. In practical terms, the results state that matching PPC-level accuracy under the tested disturbance requires further tuning of the fractional derivative approximation, whereas energy-conscious deployments can leverage FOSMC to trim roughly 15-30.

Positioning the study within prior literature, most fractional-order works report improvements relative to PID or adaptive baselines under bespoke trajectories [17]-[20]. By matching the plant, disturbance, solver tolerances, and sampling grid across both controllers, this benchmark supplies the reproducible dataset that those surveys cite as missing. The accuracy–effort curves therefore complement industrial PPC deployments rather than replacing them outright, and the associated Python workflow allows researchers to inject additional nonlinear or learning-based modules under identical numerical assumptions.

Future research can build directly on these findings in three layers. First, controller designers can reuse the released workflow to hybridise FOSMC with adaptive or data-driven observers while keeping the PPC base-

line as a reference envelope, aiming to recover the lost accuracy without abandoning the torque savings. Second, system-level engineers can incorporate actuator thermal and energy storage models so that the quantified torque reductions translate into measurable life-cycle benefits. Third, roboticists focusing on collaborative or surgical manipulators can extend the fractional surface design to higher-DOF or compliance-dominated mechanisms where smoother torques are prioritised over strict tracking. Because recent trends favour benchmarkable workflows over isolated controller tweaks, the open dataset and matching protocol supplied here constitute the principal novelty: they transform well-known structures into a comparative evidence base that was previously missing.

Three key experiments are now required to translate the simulated trade-offs into deployable practice: i) hardware-in-the-loop trials with encoder noise and actuator saturation to determine whether the observed accuracy deficit persists once more faithful fractional operators are implemented [24]; ii) Monte Carlo stress tests that perturb link masses, payloads, and viscous damping so the reported RMSE distributions can be mapped to manufacturing tolerances [24]; and iii) energy auditing on a regenerative drive bench that compares life-cycle efficiency and thermal rise between FOSMC and PPC over representative duty cycles [4]. Completing this trio will reveal whether the torque advantage remains meaningful in hardware and will highlight any controller retuning needed for flexible links.

Taken together, the comparative evidence, contextual framing, and prescribed experiments provide a clear take-away for readers: FOSMC presently exchanges accuracy for lower torque demand under the adopted approximation, while PPC remains the preferred option whenever stringent precision specifications dominate the design brief. Figure 1 summarizes the two-link manipulator geometry used throughout the study, highlighting joint locations, link lengths, and reference frames needed for interpreting the subsequent tracking results.

Table 2. Performance comparison of FOSMC and PPC

Controller	RMSE q_1 (rad)	RMSE q_2 (rad)	Mean Torque τ_1 (N.m)	Mean Torque τ_2 (N.m)
FOSMC	0.458	0.453	69.2	29.0
PPC	0.365	0.337	86.1	41.4

Figure 2 compares the tracking error trajectories of both controllers; panel Figure 2(a) focuses on joint 1 while panel Figure 2(b) presents joint 2 so that transient and steady-state deviations can be contrasted side-by-side before examining the absolute angle responses. Figure 3 then overlays the commanded and actual joint angles, with Figure 3(a) covering q_1 and Figure 3(b) covering q_2, providing context for the error magnitudes reported in Figure 2.

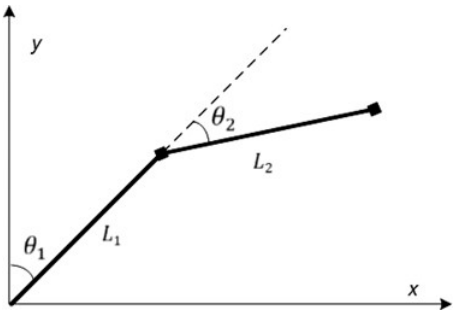
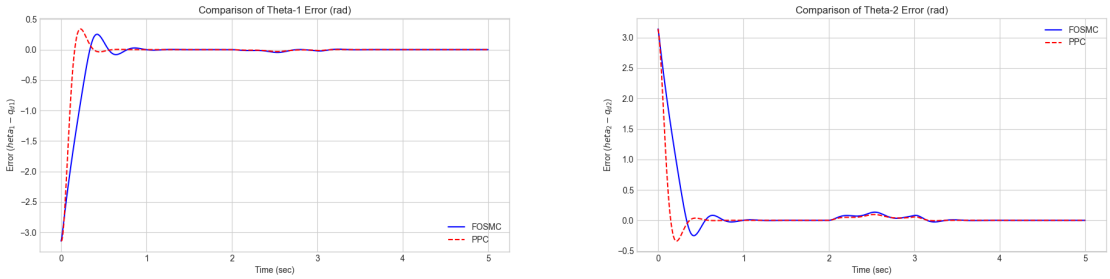


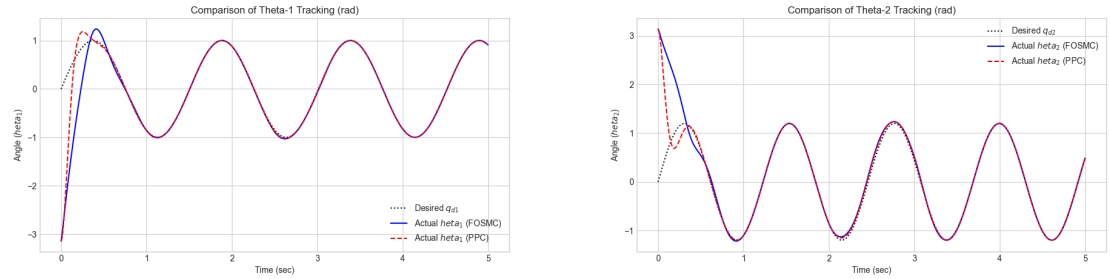
Figure 1. Structure of a 2-DOF robotic manipulator

Figure 4 details the control efforts supplied to each joint; Figure 4(a) reports τ_1 while Figure 4(b) reports τ_2, enabling a direct comparison of the energy cost associated with the improved tracking. Finally, Figure 5 gathers the joint velocity responses, where Figure 5(a) depicts $\dot{q}_1$ and Figure 5(b) depicts $\dot{q}_2$, highlighting the smoother transients achieved by FOSMC during the disturbance interval.



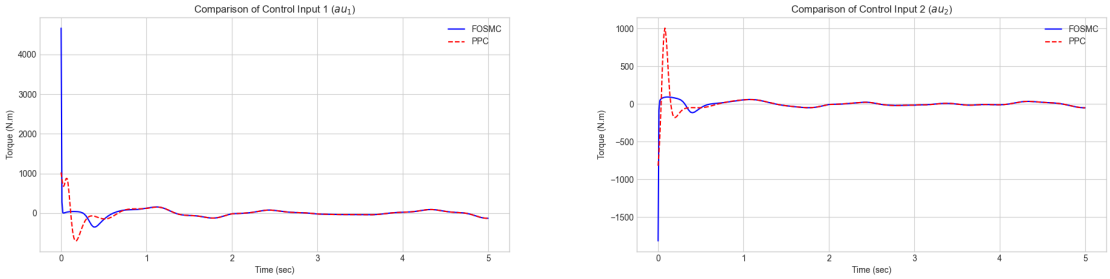
(a) (b)

Figure 2. Tracking error comparison for FOSMC and PPC (a) tracking error for joint 1 and (b) tracking error for joint 2



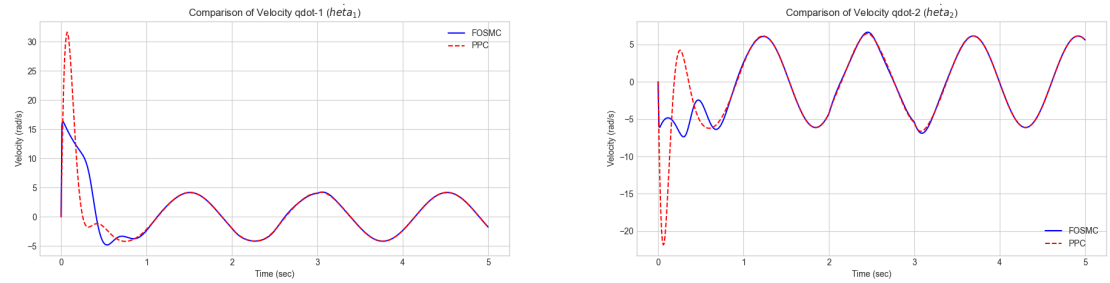
(a) (b)

Figure 3. Joint angle tracking for FOSMC and PPC (a) joint angle q_1 tracking performance and (b) joint angle q_2 tracking performance



(a) (b)

Figure 4. Control input comparison for FOSMC and PPC (a) control input τ_1 for joint 1 and (b) control input τ_2 for joint 2



(a) (b)

Figure 5. Joint velocity comparison for FOSMC and PPC (a) joint velocity $\dot{q}_1$ and (b) joint velocity $\dot{q}_2$

5. CONCLUSION

This study presented a detailed comparative analysis between Fractional-Order Sliding Mode Control (FOSMC) and Pole Placement Control (PPC) for a two-link robotic manipulator. The FOSMC was designed using a fractionalorder sliding surface to provide greater control flexibility and improve the system's dynamic response, whereas PPC served as a linear benchmark. Both controllers were implemented with identical Lagrange models, trajectories, and disturbance profiles to ensure a fair evaluation.

Quantitatively, the current fractional implementation yields RMSE values of 0.458 rad for q1 and 0.453 rad for q2, trailing PPC's 0.365 rad and 0.337 rad. This accuracy gap is offset by lower mean torques of 69.2/29.0 N·m versus PPC's 86.1/41.4 N·m, clarifying that the benchmarked FOSMC profile is attractive when torque limits dominate and PPC remains preferable when tight tracking is mandatory.

Despite these benefits, the evaluation remains limited to ideal rigid-body dynamics, high-gain approximations of the fractional derivative, and simulated sensor data. The absence of joint friction, payload variation, and network induced delays may overestimate achievable robustness margins, and hardware implementation could reveal actuator bandwidth constraints. To close these gaps, we prioritize three experiments: i) hardware-in-the-loop tests with realistic actuator and encoder noise, ii) Monte Carlo stress testing that perturbs inertial and damping parameters, and iii) energy auditing with regenerative drives to quantify life-cycle efficiency relative to PPC.

Overall, the findings confirm that PPC retains the accuracy lead for the tested nonlinear manipulator while FOSMC delivers measurable torque savings whenever actuator effort is the binding constraint. The open simulation workflow offers a reproducible baseline for researchers targeting experimental validation, real-time deployment, and higher-degree-of-freedom extensions, and it anchors future studies to shared quantitative figures-of-merit.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ahmed Bennaoui	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Salah Benzian	✓		✓	✓		✓			✓		✓		✓	
Idrees Nasser Alsolbi		✓				✓		✓	✓	✓	✓	✓		
Aissa Ameur		✓				✓		✓	✓	✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

SUPPLEMENTARY MATERIALS

Supplementary File S1 (`supplementary_code.tex`) provides the complete, reproducible Python workflow: robot model, FOSMC and PPC controller implementations, simulation driver, plotting routines generating Figures 2–5, and the animation script. All solver settings, gain selections, and post-processing steps referenced in the manuscript can be re-run directly. The file is self-contained; executing it with a standard scientific Python stack (NumPy, SciPy, Matplotlib) regenerates all reported performance metrics.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article.

REFERENCES

[1] P. C. Müller, "Book review: Robot dynamics and control: Mark W. Spong and M. Vidyasagar," *Automatica*, vol. 28, no. 3, pp. 655–656, May 1992, doi: 10.1016/0005-1098(92)90197-N.

[2] J. J. Craig, "Introduction to Robotics: Mechanics and Control," *4th ed. Upper Saddle River*, NJ, USA: Pearson, 2021.

[3] J.-J. E. Slotine and W. Li, "Applied Nonlinear Control," *Illustrate. Upper Saddle River*, NJ, USA: Prentice-Hall, 1991.

[4] M. Rahmani, H. Komijani, and M. H. Rahman, "New Sliding Mode Control of 2-DOF Robot Manipulator Based on Extended Grey Wolf Optimizer," *International Journal of Control, Automation and Systems*, vol. 18, no. 6, pp. 1572–1580, 2020, doi: 10.1007/s12555-019-0154-x.

[5] V. Tinoco, *et al.*, "A review of advanced controller methodologies for robotic manipulators," *International Journal of Dynamics and Control*, vol. 13, p. 36, 2025, doi: 10.1007/s40435-024-01533-1.

[6] V. I. Utkin, *"Sliding Modes in Control and Optimization,"* Berlin, Heidelberg: Springer, 1992. [Online]. Available: https://link.springer.com/book/10.1007/978-3-642-84379-2

[7] C. Edwards and S. K. Spurgeon, *"Sliding Mode Control: Theory and Applications."* London: Taylor and Francis, 1998. [Online]. Available: https://www.taylorfrancis.com/books/mono/10.1201/9781498701822/sliding-mode-control-christopher-edwards-sarah-spurgeon

[8] A. Levant, "Sliding order and sliding accuracy in sliding mode control," *International Journal of Control*, vol. 58, no. 6, pp. 1247–1263, 1993. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00207179308923053

[9] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *"Robot Modeling and Control"*, Hoboken, NJ: John Wiley and Sons, 2006. [Online]. Available: https://www.wiley.com/en-us/Robot+Modeling+and+Control-p-9780471649908

[10] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control"*. London: Springer, 2009. [Online]. Available: https://link.springer.com/book/10.1007/978-1-84628-642-3

[11] K. B. Oldham and J. Spanier, *"The Fractional Calculus: Theory and Applications of Differentiation and Integration to Arbitrary Order"*. New York: Academic Press, 1974. [Online]. Available: https://books.google.com/books/about/The Fractional Calculus.html?id=dg4dQgAACAAJ

[12] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu, *"Fractional-Order Systems and Controls: Fundamentals and Applications"*. London: Springer, 2010. [Online]. Available: https://link.springer.com/book/10.1007/978-1-84996-335-0.

[13] I. Petras, *"Fractional-Order Nonlinear Systems: Modeling, Analysis and Simulation"*. Berlin, Heidelberg: Springer, 2011. [Online]. Available: https://link.springer.com/book/10.1007/978-3-642-18101-6

[14] A. Oustaloup, "Frequency-band complex noninteger differentiator: characterization and synthesis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications,*

[15] D. Valerio and J. S. da Costa, *"An Introduction to Fractional Control"*. London: Institution of Engineering and Technology, 2012. [Online]. Available: https://digital-library.theiet.org/content/books/ce/pbce091e

[16] I. Podlubny, "Fractional Differential Equations," *San Diego: Academic Press*, 1999.

[17] T.-T. Nguyen, "Fractional-order sliding mode controller for the two-link robot arm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 6, pp. 5579–5585, Dec. 2020, doi: 10.11591/ijece.v10i6.pp5579-5585.

[18] K. Bin Gaufan, M. B. Aremu, N. M. Alyazidi, and A. Nasir, "Robust fractional-order sliding mode control for robotic manipulator system with time-varying disturbances," *Franklin Open*, vol. 12, p. 100287, Sep. 2025, doi: 10.1016/j.fraope.2025.100287.

[19] S. Ahmed and A. T. Azar, "Adaptive fractional tracking control of robotic manipulator using fixed-time method," *Complex & Intelligent Systems*, vol. 10, pp. 369–382, 2024, doi: 10.1007/s40747-023-01164-7.

[20] M. Yavuz, M. Öztürk, and B. Ya, "Comparison of fractional order sliding mode controllers on robot manipulator," *International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, vol. 15, no. 2, pp. 281–293, 2025, doi: 10.36922/ijocta.1678.

[21] H. K. Khalil, "Nonlinear Systems," *3rd ed. Pearson Education*: Prentice Hall, 2002.

[22] C. C. Nguyen and F. J. Pooran, "Cartesian path control of a two-degree-of-freedom robot manipulator," Jan. 1988. [Online]. Available: https://ntrs.nasa.gov/citations/19880004526.

[23] W. Boukadida, R. Bkekri, A. Benamor, and H. Messaoud, "Trajectory tracking of robotic manipulators using optimal sliding mode control," in *2017 International Conference on Control, Automation and Diagnosis (ICCAD)*, 2017, pp. 545–550, doi: 10.1109/CADIAG.2017.8075717.

[24] P. Virtanen, *et al.*, "SciPy 1.0: fundamental algorithms for scientific computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020, doi: 10.1038/s41592-019-0686-2.

[25] H. N. Rahimi, I. Howard, and L. Cui, "Neural adaptive tracking control for an uncertain robot manipulator with time-varying joint space constraints," *Mechanical Systems and Signal Processing*, vol. 112, pp. 44–60, Nov. 2018, doi: 10.1016/j.ymssp.2018.03.042.

BIOGRAPHIES OF AUTHORS

Ahmed Bennaoui is Received his BSc degree in Electronics Engineering from the University of Mohamed Boudiaf - M'sila (2011), Algeria. He obtained MSc degrees in Electrical Engineering with a specialization in Advanced Automation from the University of ZIANE ACHOUR (DJELFA), Algeria (2012 and 2016), and a PhD degree in Electrical Engineering with a specialization in Intelligent Control and Automation (2024) from the University of Amar Telidji (Laghouat), Algeria. His interests include Nonlinear Dynamics, Fuzzy Logic Controller, Power System Control, and Optimization Techniques. He can be contacted via email at a.bennaoui@lagh-univ.dz.

Salah Benzian Received his BSc degree in Electrical and Electronics Engineering from the University of M'hamed Bougara - Boumerdes (2011), Algeria. He obtained MSc degrees in Electrical Engineering with a specialization in Advanced Automation from the University of ZIANE ACHOUR (DJELFA), Algeria (2012 and 2016), and a PhD degree in Electrical Engineering with a specialization in Intelligent Control and Automation (2021) from the University of Amar Telidji (Laghouat), Algeria. He currently serves as a lecturer at the Polytechnic School of El-Harrach, Algeria. His interests include Nonlinear Dynamics, Fuzzy Logic Controllers, Power System Control, and Optimization Techniques. He can be contacted via email at s.benzian@cu-aflou.edu.dz.

Idrees Nasser Alsolbi Received the Ph.D. degree in Big Data from the University of Technology Sydney (UTS), Australia, in 2023. He earned the M.Sc. degree in Information Technology from Monash University, Australia, in 2018, and the B.Sc. degree in Programming and Computer Science from Umm Al-Qura University, Makkah, Saudi Arabia, in 2010 (1431 H). He is currently an Assistant Professor in the Data Science Department, College of Computing, Umm Al-Qura University. His research interests include big data analytics, edge computing, machine learning, wireless sensor networks, educational data mining, and AI-based decision support systems. His work has appeared in leading journals such as Nature Scientific Reports, Artificial Intelligence Review (Springer), and Information (MDPI). He can be contacted via email at insolbi@uqu.edu.sa.

Aissa Ameur Received his Magister and Ph.D. degrees in Electrical Engineering in 2005 and 2012, from Batna University, Algeria. In 2005, he joined the Electrical Engineering Department of Laghouat University, Algeria as Assistant Lecturer. Since May 2012, Dr. Ameur is an Assistant Professor in the same department. He is a researcher in LeDMaScD laboratory, Laghouat University, Algeria. His main research interests include Modelling of Electrical Machines, Electrical Drives Control, Fault Diagnosis, Artificial Intelligence, and Renewable Energy Systems Control. He can be contacted via email at a.ameur@lagh-univ.dz.