Residual Hybrid Motor Controller with Multi-Phase Learning

Johann Wiens, Christoph Reich
Institute of Data Science, Cloud Computing and IT-Security
Furtwangen University of Applied Sciences
Furtwangen, Germany
Email: johann-wiens@outlook.com, christoph.reich@hs-furtwangen.de

***Abstract*—Direct-Current (DC) motors serve as the fundamental actuators in emerging intelligent systems, including advanced robotics, autonomous vehicles, and smart home infrastructure. Precise and energy-efficient control of these motors is paramount, as control quality directly dictates system safety, movement fluidity, and operational longevity. However, conventional Proportional-Integral-Derivative (PID) controllers in compact motor drives often struggle with repeatable nonlinearities and unmodeled dynamics. Examples are friction, cogging torque, and saturation effects. These effects can reduce stability margins and require frequent manual retuning. This paper proposes a hybrid motor controller. A tuned PID controller provides the baseline loop. A Long–Short-Term-Memory (LSTM) module adds a residual path. The baseline loop guarantees stability, transparency, and hard actuator limits. The compact three-layer LSTM adds a bounded corrective voltage to compensate repeatable nonlinearities. Learning runs in two phases. (1) Behavior Cloning: the LSTM first learns to imitate the PID controller. This fixes the control direction and creates a safe starting point for Phase 2. (2) Advantage-Weighted Regression (AWR): a task-specific advantage function rewards residual actions only when they reduce trajectory error. A conservative gate discards degrading updates. For realistic and reproducible evaluation, the geared DC drive is modeled as an averaged Pulse-Width-Modulation (PWM) Brushless-Direct- Current (BLDC) motor model. The model includes Stribeck friction, iron/core losses, cogging torque, current and voltage saturation, and a lumped thermal winding model. Tracking tasks follow quintic S-curve trajectories with output-side load disturbances. One trajectory is held out to test generalization. Across all scenarios, the residual path lowers position Mean-Squared-Error (MSE) compared to pure PID and reduces overshoot at similar rise times. Comparing the MSEs, PID+LSTM reduces the error by about 98.5% on average relative to PID only.**

Keywords-Hybrid controller; Smart System AI-Controlled; Proportional–integral–derivative controller; LSTM controller; Behavior Cloning; Advantage-Weighted Regression; DC motor; motor modeling.

I. Introduction

PID control is still the de facto standard in industrial motion systems. It is simple, interpretable, and when properly tuned robust to moderate uncertainty [1]. Modern compact drives, however, increasingly operate in regimes with strong nonidealities. These include static and Stribeck friction, iron and core losses, cogging torques, saturations, and thermal drift. In such regimes, purely linear control often faces a trade-off between steady-state accuracy, overshoot, and actuator stress. Laborious retuning for each operating point is then common [2].

A promising alternative is a residual learned policy on top of a proven baseline controller. The baseline loop guarantees reasonable behavior. A bounded data-driven term compensates repeatable, state-dependent errors [3]. This pattern is attractive for safety-relevant actuation. The learned component can be range-limited, while the stabilizing structure stays in place.

This study considers a geared DC drive and adds a compact LSTM residual to a strongly tuned PID. Learning proceeds in two stages. First, Behavior-Cloning (BC) aligns the LSTM with the PID to give a conservative starting point. Then, Advantage-Weighted Regression (AWR) uses a simple task-centered advantage. A conservative policy-improvement gate prevents regressions [4]. The BLDC model includes key multiphysics effects (Stribeck friction, iron-loss torque, cogging, thermal dynamics) and hard limits. Smooth quintic S-curves serve as references. They are a canonical setup for point-to-point motions [5].

The contributions of this work are:

- Realistic, reproducible motor model of a geared DC drive with multiphysics effects.
- Compact hybrid PID+LSTM controller with two-phase training (BC $\rightarrow$ AWR) and conservative update gating.
- Systematic evaluation on S-curve trajectories including load disturbances and a hold-out trajectory for generalization.

The remainder of the paper is as follows. Section II reviews related work. Section III explains the controller and learning scheme. Section IV describes the model and scenarios. Section V reports results and limitations. Section VI concludes.

II. Related Work

Recent work couples classical PI/PID loops with small learned compensators in a parallel or residual path. The goal is to handle repeatable nonlinearities while keeping the baseline structure. On a DC servo test stand, the authors of [6] use a PID baseline and add an online Artificial-Neural-Network (ANN) / Recurrent-Neural-Network (RNN) precompensator gated by a fuzzy system. They report improved overshoot and steady-state accuracy in hardware. In contrast, the present work uses a direct LSTM residual with two-phase off-policy learning. This simplifies the architecture and training compared to the fuzzy-gated RNN in [6], For Permanent-Magnet-Synchronous-Motor (PMSM) Field-Oriented-Control (FOC), the authors of [7] add a small feedforward network that corrects PI transients. After pruning and quantization, the network runs on Microcontroller Unit (MCU) hardware and reduces overshoot. Both approaches share the same safety idea used here: a bounded learned term adds to the conventional loop, instead of replacing it.

Recurrent networks, especially LSTMs, can capture temporal dependencies in torque ripple, flux dynamics, or friction memory. In Direct-Torque-Control (DTC) of induction machines, the authors of [8] replace the switching table with a ConvLSTM selector and improve low-speed behavior in simulation. For synchronous machines, the authors of [9] use an LSTM-driven predictive current controller. Two-degree-of-freedom designs (feedforward + feedback) also profit from recurrent models. Yin et al. [10] train an LSTM inversion model as a feedforward compensator and combine it with linear feedback for nanopositioning. Broader evaluations of ML-based PMSM drive controllers also support the use of recurrent architectures [11].

In this paper, the learned component is deliberately residual and strictly bounded. The two-stage training (BC $\rightarrow$ AWR) is used to preserve stability and interpretability of the baseline.

Key differences to prior work: Compared to hybrid PI/PID+network approaches such as [7], this study differs in three main aspects:

- *Model and task:* The focus is a geared DC drive modeled as an averaged PWM BLDC. The model includes friction (with Stribeck), iron/core losses, cogging torque, saturation, and a lumped thermal model. The evaluation uses S-curve point-to-point references and output-side load disturbances. Prior work often uses simpler friction models or different machines with less detailed multiphysics.
- *Controller architecture:* The controller uses a small LSTM as a residual voltage path in parallel to a fixed, tuned PID. The PID is never turned off. The residual voltage is hard-bounded. Thus, authority and stability margins remain with the classical loop, unlike approaches that replace switching tables or MPCC [8].
- *Learning procedure:* Many earlier works rely on direct supervised training or problem-specific optimization. Here, learning uses two phases. First, Behavior Cloning imitates the PID and fixes the control direction. Second, Advantage-Weighted Regression updates the residual, with improvements measured against the PID baseline and protected by a conservative gate.

Scope of this paper: This paper:

- designs a compact three-layer LSTM residual that augments a tuned PID position loop for a realistic geared DC drive;
- trains this residual in two phases: PID-mimicking Behavior Cloning, followed by Advantage-Weighted Regression using a simple, trajectory-error-based advantage and conservative policy-improvement gating; and
- compares the hybrid controller against the pure PID on multiple S-curve trajectories, including a hold-out trajectory and load disturbances. Metrics include position MSE, overshoot, rise time, and actuator limits.

The results show that a strictly bounded residual LSTM can consistently improve tracking over a strong industrial baseline without changing the underlying control structure.

III. Hybrid Control Approach

The proposed hybrid approach combines a robust PID baseline with a small residual LSTM. The PID provides a well-understood, stabilizing foundation, while the LSTM is trained in two phases (Behavior Cloning, then AWR) to compensate repeatable, state-dependent nonlinearities that a fixed-parameter PID cannot fully address.

Phase 1: Behavior Cloning (BC)

In Phase 1 (cf. Figure 1), the LSTM is trained by supervised learning to imitate the PID output $u_{\text{PID}}(t)$ [12]. Let $u_{\text{LSTM}}(t;\theta)$ denote the LSTM output. The horizon $\mathcal{T}$ is used to minimize

$$\min_{\theta} \ \mathcal{L}_{\text{BC}}(\theta) \ = \ \sum_{t\in\mathcal{T}} \left\| u_{\text{LSTM}}(t;\theta) - u_{\text{PID}}(t) \right\|_2^2. \qquad (1)$$

Only improving updates are accepted; degrading ones are rejected. This yields a conservative starting point.

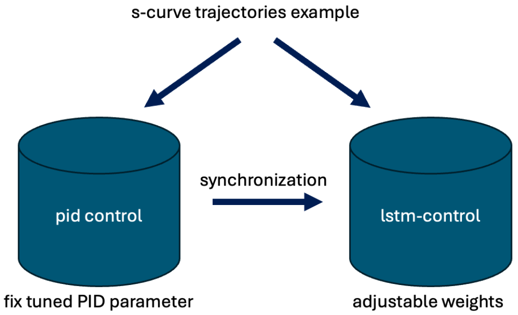

Figure 1. Phase 1: Behavior Cloning (BC).

Phase 2: Reinforcement Learning with AWR

In Phase 2 (cf. Figure 2), the full closed-loop drive is used. Advantage-Weighted Regression (AWR) [13] acts as the policy-improvement method. PID and LSTM both receive the control error $e(t)$; PID parameters remain fixed. The LSTM is updated by weighted regression. Only updates that improve performance are accepted (conservative policy improvement [4]).

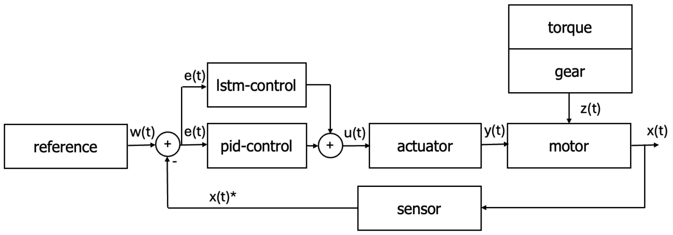

Figure 2. Phase 2: Reinforcement Learning (AWR).

Implementation Notes

- **Safety bounds:** The PID path is always active. The LSTM output is bounded via saturation or a blend gain.
- **Curriculum:** BC starts with simple references (step, ramp, sine) and then uses richer trajectories before AWR.
- **Validation:** Trajectory error, control effort, and robustness (parameter variations) are evaluated systematically.

Copyright (c) IARIA, 2026.   ISBN: 978-1-68558-351-4

IV. Simulation Environment and Modeling

The goal is a realistic and reproducible assessment of the hybrid PID+LSTM controller on S-curves.

Simulation instead of hardware: Evaluation starts in simulation to make disturbances and drifts reproducible and to avoid risk and wear during early policy iterations. The setup follows the idea of a digital twin: a high-fidelity dynamic model used as a virtual test bench before deployment. The model captures dominant nonidealities: Stribeck friction, iron losses, cogging, saturations, and thermal effects. Hardware validation is planned as a next step.

References: smooth S-curves: Quintic S-curves with zero boundary conditions (position, velocity, acceleration) are used as references. They have low jerk and are common in industry. They also isolate controller performance from high-frequency artifacts [5].

A. Overview and Time Discretization

All dynamics are integrated with explicit Euler at step $dt{=}10^{-4}\,\mathrm{s}$. PWM is modeled as an *averaged* actuation voltage with small ripple. The core structure is residual:

$$V_{\text{tot}}(k) \;=\; V_{\text{PID}}(k) \;+\; V_{\text{LSTM}}(k). \tag{2}$$

B. Reference Trajectories and Scenarios

The numbers are not chosen arbitrarily, but are the mathematically compelling result when searching for a curve that starts at 0 and ends at 1 and has no velocity or acceleration at either end. For $0{\le}t{\le}T$,

$$s(t) = 10\big(\tfrac{t}{T}\big)^3 - 15\big(\tfrac{t}{T}\big)^4 + 6\big(\tfrac{t}{T}\big)^5 . \tag{3}$$

Ten scenarios with different reach times T and end positions are evaluated; see Table I. An external load $T_{\text{Load}}{=}0.8\,\mathrm{N\,m}$ acts by default from $40\,\%$ to $80\,\%$ of the scenario duration.

C. Motor Model: BLDC with Gearbox, Averaged PWM

The BG 42x15 with planetary gearbox PLG 42S is represented as a DC drive with states current i, angular velocity ω, angle θ, and winding temperature T_{cu}. The fundamental equations are:

$$v = R(T)\,i + L\,\dot{i} + K_e^*\,\omega, \tag{4}$$

$$J\,\dot{\omega} = K_t^*\,i - T_{\text{fric}}(\omega,T) - T_{\text{iron}}(\omega) - T_{\text{cog}}(\theta) - T_{\text{load}}. \tag{5}$$

Cf. [14]. Parameters K_e^* and K_t^* vary slightly with current and temperature.

Nonidealities:

- **Friction** (viscous, Coulomb, Stribeck) cf. [15]:

$$\begin{aligned} T_{\text{fric}}(\omega) \;=\; & B\,\omega + T_c\,\mathrm{sgn}(\omega) + \\ & (T_s{-}T_c)\,\exp\!\Big(-\Big(\frac{\omega}{\omega_s}\Big)^2\Big)\,\mathrm{sgn}(\omega). \end{aligned} \tag{6}$$

- **Iron losses** cf. [16]:

$$P_{\text{iron}}(\omega) = k_h\,|\omega| + k_e\,\omega^2, \tag{7}$$

$$T_{\text{iron}}(\omega) = \frac{P_{\text{iron}}(\omega)}{\max(|\omega|,\varepsilon)}. \tag{8}$$

- **Cogging torque:**

$$T_{\text{cog}}(\theta) \;=\; T_{\text{cog, amp}}\sin(n_{\text{el}}\,\theta). \tag{9}$$

The harmonic is included as in [17].

- **PWM artifacts:** averaged actuation voltage with small $v_{\text{ripple}}(t)$ at $f_{\text{PWM}}{=}20\,\mathrm{kHz}$.
- **Saturations:** current and voltage limits

$$|i|{\le}i_{\text{peak}}, \qquad |v|{\le}V_{\text{bus}}. \tag{10}$$

- **Thermal model:** A lumped RC model is used cf. [18]:

$$C_{\text{th}}\,\dot{T}_{\text{cu}} \;=\; i^2R(T_{\text{cu}}) + |T_{\text{iron}}\omega| - h\,(T_{\text{cu}} - T_{\text{amb}}). \tag{11}$$

The output torque is reflected through gear ratio r and efficiency η:

$$T_{\text{load,m}} \;=\; \frac{T_{\text{load,out}}}{r\,\eta}. \tag{12}$$

Linear position follows from $\theta_{\text{out}}{=}\theta/r$ and shaft radius r_s:

$$p \;=\; \frac{\theta}{r}\,r_s, \tag{13}$$

cf. the arc-length relation [19].

D. Baseline Controller (PID)

With position error $e(k) = p_{\text{ref}}(k) - p(k)$:

$$V_{\text{PID}}(k) \;=\; K_p\,e(k) + K_i\sum_{j\le k} e(j)\,dt + K_d\,\frac{e(k)-e(k-1)}{dt}. \tag{14}$$

A moderately aggressive tuning (K_p=180, K_i=200, K_d=10) is employed; V_{PID} is limited to $|V| \le V_{\text{bus}}$ [1].

E. Residual Controller (LSTM)

The LSTM (3 layers, hidden size 12) receives sequences of the last $T_{\text{seq}}{=}10$ samples with features

$$\left[\frac{t}{T_{\text{end}}},\;\frac{i}{i_{\text{peak}}},\;\frac{e}{e_{\text{scale}}}\right] \in [-1,1]^3, \qquad e_{\text{scale}}{=}0.30\,\mathrm{m}. \tag{15}$$

The output is mapped to a physical voltage:

$$V_{\text{LSTM}}(k) \;=\; \tfrac{1}{2}V_{\text{bus}}\cdot\tanh\!\big(\mathrm{Head}(\mathrm{LSTM}(\mathrm{Seq}))\big), \tag{16}$$

ensuring $|V_{\text{LSTM}}| \le \frac{1}{2}V_{\text{bus}}$ (authority remains primarily with the PID) [20]; the combination follows Equation 2.

F. Two-Stage Learning Scheme

Phase 1: Behavior Cloning (BC): Datasets from PID rollouts (S-curves, load window) provide supervision. The target is the normalized PID voltage; optimization uses MSE with soft clipping through $\tanh(\cdot)$ [21].

Phase 2: Advantage-Weighted Regression (AWR): Hybrid rollouts (PID+LSTM) are then generated. Advantage relative to PID is defined as

$$A_k \;=\; \frac{|e_{\text{PID},k}| - |e_{\text{hyb},k}|}{e_{\text{scale}}}, \qquad w_k \;=\; \exp(\beta A_k),\;\beta{=}4. \tag{17}$$

Regression on the residual actions is performed, weighted by w_k; updates are conservatively accepted using BC validation [4].

Copyright (c) IARIA, 2026.   ISBN: 978-1-68558-351-4

V. Results and Discussion

Among the ten S-curve scenarios in Table I, scenarios 1–9 are used both for training and evaluation, whereas scenario 10 is held out and used only for testing. We, therefore, treat scenario 10 as the primary generalization case: the controller must track a reference it has never seen during training.

TABLE I. Reference trajectory scenarios.

Scenario number	Position reach time [s]	Position move [m]	Simulation time [s]	**MSE PID only**	**MSE PID+LSTM**	Mode
1	0.600	0–0.30	0–0.600	**2.23e-05**	**3.79e-07**	Train+Test
2	0.725	0–0.10	0–0.725	**2.14e-05**	**4.18e-07**	Train+Test
3	0.550	0–0.15	0–0.550	**2.27e-05**	**2.54e-07**	Train+Test
4	0.300	0–0.15	0–0.300	**2.13e-05**	**2.72e-07**	Train+Test
5	0.350	0–0.05	0–0.350	**2.14e-05**	**3.60e-07**	Train+Test
6	0.650	0–0.30	0–0.600	**2.28e-05**	**4.13e-07**	Train+Test
7	0.450	0–0.10	0–0.725	**1.96e-05**	**3.69e-07**	Train+Test
8	0.350	0–0.15	0–0.550	**2.05e-05**	**2.41e-07**	Train+Test
9	0.250	0–0.15	0–0.300	**1.88e-05**	**2.56e-07**	Train+Test
10	**0.475**	**0–0.15**	**0–0.475**	2.26e-05	2.35e-07	**Test only**

The hybrid controller (PID+LSTM) improves trajectory tracking on all tasks compared to the standalone tuned PID. It shows lower error energy and faster settling at similar rise times, while respecting the same actuator limits.

A. Training Dynamics and Phase Contributions

Validation MSE: Figure 3 shows a "big step + fine-tuning" pattern: Behavior Cloning (BC) reduces validation MSE from 1.049×10^{-2} to 4.22×10^{-4} (epoch 7). Advantage-Weighted Regression (AWR) further refines to 4.34×10^{-4} (epoch 4). A conservative gate rejects later degrading epochs (epochs 5–6). *Interpretation:* BC brings the LSTM close to the PID policy and avoids risky exploration. AWR then uses a task-centered advantage relative to the PID baseline. It shifts the residual action where it reduces trajectory error. Conservative policy improvement prevents large regressions. This explains why most of the gain appears in BC, while AWR adds robustness and small refinements, for example near friction transitions.

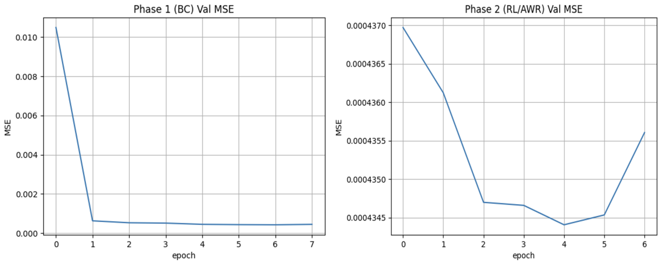

Figure 3. Validation MSE: Phase 1 (BC, left) and Phase 2 (AWR, right).

B. Tracking on S-Curves and Error Statistics

Position and velocity profiles: In Figure 4 and Figure 5, the hybrid controller tracks the reference more tightly and with less overshoot. Velocity is smoother, with less ripple. Peak velocities stay on the PID level ($\approx 80\,\mathrm{rad/s}$). Thus, the hybrid keeps timing but reduces error energy. *Error energy and overshoot:* Error energy and overshoot.: Across scenarios, the residual path consistently reduces position MSE relative to pure PID. In the test scenario, the reduction is roughly one order of magnitude, see Figure 6.

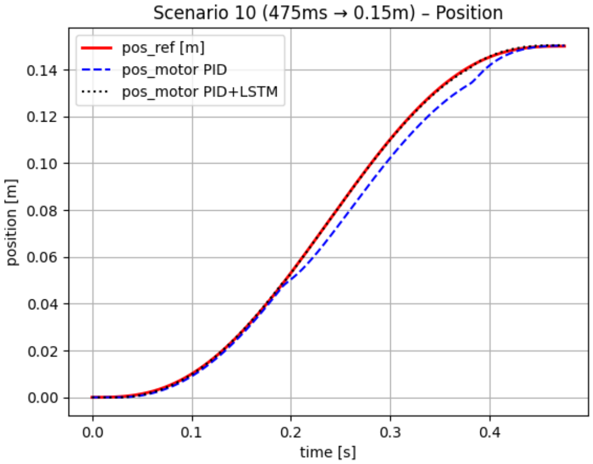

Figure 4. Test scenario 10 position: hybrid vs. PID control.

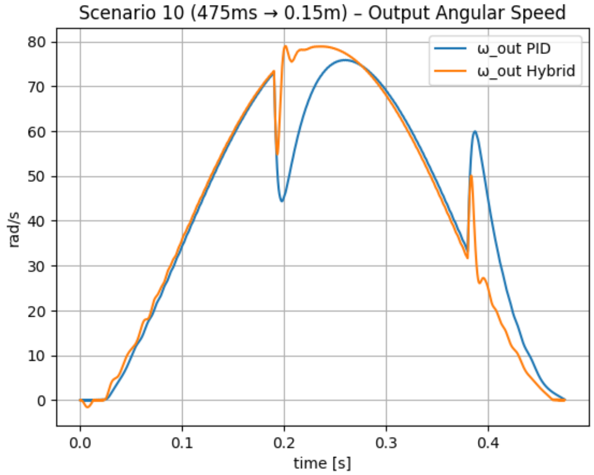

Figure 5. Test scenario 10 speed: hybrid vs. PID control.

C. Actuation Effort and Saturations

Voltages and currents: The smoother trajectories appear in the torque and voltage profiles as well. The LSTM output is strictly limited to $|V_{\mathrm{LSTM}}| \leq \frac{1}{2}V_{\mathrm{bus}}$ Thus, the PID retains the main authority and the residual cannot cause extreme actions. Transient stresses are reduced overall. *Interaction with saturations:* The total voltage $V_{\mathrm{tot}} = V_{\mathrm{PID}} + V_{\mathrm{LSTM}}$ is still clipped by the bus limit. No new saturation regimes appear. The

Copyright (c) IARIA, 2026. ISBN: 978-1-68558-351-4

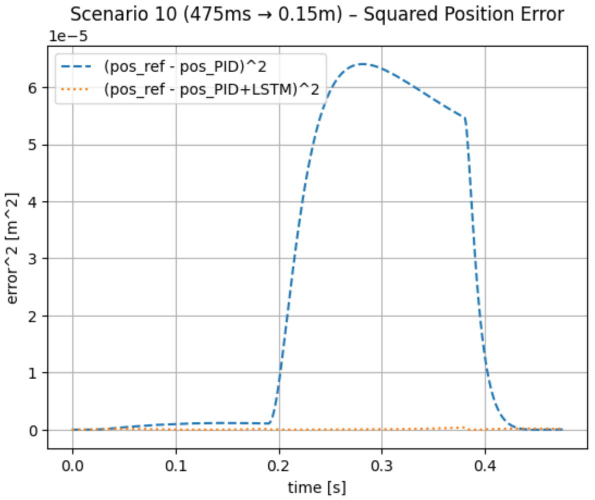

Figure 6. Test scenario 10 mean-squared error: hybrid vs. PID control.

residual mainly compensates repeatable nonlinearities (Stribeck friction, iron-loss torque, cogging) instead of creating extra peaks.

D. Robustness to Disturbances and Drift

Load window: The reference tasks include an output-side load window ($T_{\text{Load}} = 0.8\,\text{Nm}$, 40–80 % of thee scenario duration). Figure 7 shows the torque in test scenario 10. The hybrid shows smaller error peaks in this interval. It leverages recurring, state-dependent patterns in the nonidealities, while the PID ensures stability. *Parametric uncertainties:* The motor model includes temperature-dependent winding parameters, iron losses, cogging, and saturations. Parameters drift during the trajectory, for example due to heating. Because the residual is bounded and the PID remains active, stability margins are preserved. Robustness comes from this separation: the PID stabilizes, the LSTM compensates.

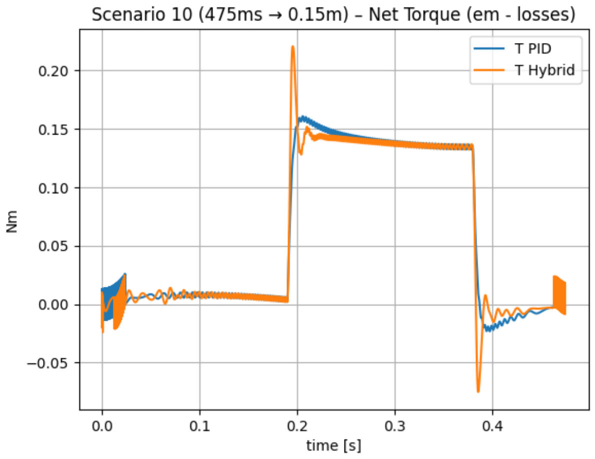

Figure 7. Test scenario 10 torque: hybrid vs. PID control.

E. Generalization and Hold-out

One trajectory is used as a *hold-out* and appears only in testing; cf. Table I). On this unseen reference, the hybrid controller again achieves tighter tracking, smoother velocity, and lower error energy. This suggests that the residual mainly exploits repeatable, state-dependent patterns. It does not simply memorize specific trajectories.

F. Residual Contribution and Interpretability

Small recurrent architecture: The LSTM is intentionally compact (3 layers, hidden size 12) and receives a short history ($T_{\text{seq}} = 10$) of normalized features. A final *tanh* enforces a hard bound on V_{LSTM}. This keeps the PID "in charge" and improves interpretability and safety. The division of labor is clear: PID handles linear and broadband effects; the LSTM corrects structured nonlinearities.

Conservative updates: During AWR, an update is kept only if it improves validation over the BC baseline. This avoids sudden policy jumps. Table II) shows that epochs 5 and 6 are rejected, even though their MSE change is small.

TABLE II. TRAINING RESULTS ACROSS PHASES AND EPOCHS.

Phase	Epoch	Val-MSE	Status
PHASE 1: BEHAVIOR CLONING			
Phase 1/BC	01	0.010490	improved
Phase 1/BC	02	0.000625	improved
Phase 1/BC	03	0.000525	improved
Phase 1/BC	04	0.000507	improved
Phase 1/BC	05	0.000445	improved
Phase 1/BC	06	0.000430	improved
Phase 1/BC	07	0.000422	improved
Phase 1/BC	08	0.000444	kept
PHASE 2: RL FINE-TUNING (AWR)			
Phase 2/RL	00	0.000437	baseline before updates
Phase 2/RL	01	0.000436	improved
Phase 2/RL	02	0.000435	improved
Phase 2/RL	03	0.000435	improved
Phase 2/RL	04	0.000434	improved
Phase 2/RL	05	0.000435	rejected
Phase 2/RL	06	0.000436	rejected

G. Computational Load and Deployment Readiness

The small LSTM and bounded residual authority are chosen with embedded hardware in mind. The PID loop remains unchanged. The LSTM adds one saturated summation path. Real-time feasibility will depend on the target platform but is aided by the compact model size.

H. Limitations and Implications

Simulation domain: All results are obtained in simulation. Simplifications, such as averaged PWM and a single cogging harmonic, may smooth some hardware effects. Metrics focus on position MSE; energy and thermal objectives are not optimized. Architecture and hyperparameters were tuned manually. Systematic ablations (e.g., over residual gain, hidden size, or sequence length) are left for future work.

Practical significance: Despite these limits, the study suggests that residual learning is a practical way to gain

Copyright (c) IARIA, 2026. ISBN: 978-1-68558-351-4

performance without discarding established control structures. This is important in safety-critical and certified systems. The consistent error reduction under strict limits and a gentle learning pipeline supports future hardware or Hardware-in-the-Loop (HiL) tests.

VI. Conclusion and Future Work

This paper presented a hybrid controller that augments a tuned PID with a compact, strictly bounded LSTM in a residual path. In a realistic simulation of an averaged PWM BLDC, including friction, iron-loss, cogging, saturation, and thermal effects, the hybrid consistently outperforms pure PID. It reduces trajectory errors (about an order-of-magnitude reduction of error energy in test scenario 10), smooths velocity and torque, and maintains similar rise times without extra peaks. The two-stage learning scheme is simple to use: BC gives most of the gain, and AWR refines it conservatively.

Limitations: All experiments are in simulation. Simplified models of iron losses, cogging harmonics, and PWM may hide some hardware details. Metrics focus on position MSE; energy and temperature are treated as constraints, not optimization targets. Architecture and hyperparameters are not yet systematically explored. Generalization is tested only on S-curves with one hold-out trajectory.

Outlook: Future work will address: (i) hardware or HiL validation with the same residual bounds and safety limits, (ii) other motion profiles (trapezoidal velocity, jerk-limited), speed or torque control, and contact/backlash effects, (iii) multi-objective cost functions that include error, energy, and temperature, and (iv) more formal guarantees for safe policy updates.

Acknowledgment

This work was funded by the state of Baden-Württemberg within the MoDe ProBio project (FEIH_PAN_2685068).

References

[1] K. J. Åström and T. Hägglund, *Advanced PID Control*. Research Triangle Park, NC: ISA, 2006.

[2] B. Armstrong-Hélouvry, P. Dupont, and C. C. de Wit, “A survey of models, analysis tools and compensation methods for the control of machines with friction”, *Automatica*, vol. 30, no. 7, pp. 1083–1138, 1994.

[3] T. Johannink et al., “Residual reinforcement learning for robot control”, *arXiv:1812.03201*, 2019.

[4] S. Kakade and J. Langford, “Approximately optimal approximate reinforcement learning”, in *Proceedings of ICML*, 2002.

[5] T. Flash and N. Hogan, “The coordination of arm movements: An experimentally confirmed mathematical model”, *Journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[6] Z. Huang et al., “Fuzzy inference system enabled neural network feedforward compensation for position leap control of dc servo motor”, *Scientific Reports*, vol. 14, no. 20814, 2024. DOI: 10.1038/s41598-024-71647-1.

[7] M. J. M. Elele, S. Pau Danilo und Zhuang, and T. Facchinetti, “Compensating pi controller’s transients with tiny neural network for vector control of permanent magnet synchronous motors”, *World Electric Vehicle Journal*, vol. 16, no. 4, p. 236, 2025. DOI: 10.3390/wevj16040236.

[8] S. Potturi et al., “Direct torque control of induction motor using convlstm based on gaussian pillbox surface”, *Mathematical Problems in Engineering*, 2022. DOI: 10.1155/2022/4408271.

[9] I. Hammoud, S. Hentzelt, T. Oehlschlaegel, and R. Kennel, “Learning-based model predictive current control for synchronous machines: An lstm approach”, *European Journal of Control*, 2022, Online first.

[10] R. Yin, Y. Chen, Z. Gong, and J. Ren, “Lstm-inversion-based feedforward–feedback nanopositioning control”, *Machines*, vol. 12, no. 11, p. 747, 2024. DOI: 10.3390/machines12110747.

[11] A. M. Tom and J. L. Febin Daya, “Design of machine learning-based controllers for speed control of pmsm drive”, *Scientific Reports*, vol. 15, no. 17826, 2025. DOI: 10.1038/s41598-025-02396-y.

[12] *Imitation: Documentation (behavior cloning overview)*, https://imitation.readthedocs.io/, Accessed for background on supervised behavior cloning, 2025.

[13] X. B. Peng, A. Kumar, G. Zhang, and S. Levine, “Advantage-weighted regression: Simple and scalable off-policy reinforcement learning”, *arXiv preprint arXiv:1910.00177*, 2019.

[14] D. Tilbury, W. Messner, and R. Hill, “Dc motor speed: System modeling”, Accessed: Oct. 19, 2025, Control Tutorials for MATLAB and Simulink (CTMS), 2025, Accessed: Oct. 19, 2025. [Online]. Available: https://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=SystemModeling.

[15] C. Canudas de Wit, H. Olsson, K. J. Åström, and P. Lischinsky, “A new model for friction”, *IEEE Transactions on Automatic Control*, vol. 40, no. 3, pp. 419–425, 1995.

[16] C. Oliver, “A new core loss model”, Ridley Engineering App Note, 2011, Accessed: Oct. 19, 2025. [Online]. Available: https://ridleyengineering.com/images/phocadownload/new%20core%20loss%20model.pdf.

[17] N. S. Samala, “Cogging torque and torque ripple analysis on permanent magnet synchronous machines”, 2019, Accessed: Oct. 19, 2025. [Online]. Available: https://www.politesi.polimi.it/handle/10589/148988.

[18] P. H. Mellor, D. Roberts, and D. R. Turner, “Lumped parameter thermal model for electrical machines with tefc design”, *IEEE Proceedings B (Electric Power Applications)*, vol. 138, no. 5, pp. 205–218, 1991.

[19] “Arc length and area of a sector”, Relation $s = r\theta$, Accessed: Oct. 19, 2025, LibreTexts, 2022, Accessed: Oct. 19, 2025. [Online]. Available: https://courses.lumenlearning.com/ccbcmd-math-1/chapter/arc-length-and-area-of-sector/.

[20] T. P. Lillicrap et al., “Continuous control with deep reinforcement learning”, *arXiv:1509.02971*, 2015.

[21] D. A. Pomerleau, “Alvinn: An autonomous land vehicle in a neural network”, in *Advances in Neural Information Processing Systems (NIPS 1)*, 1988.