Achieving Long-Term Autonomy: A Self-Correcting Deep Reinforcement Learning Agent for Edge IoT Using Digital Twin-Based Drift Compensation

Jhon Monroy, Miguel Paco, Miguel Portella, Geral Basurco, Jeymi Valdivia, Fiorela Jara, Guido Anco
Escuela profesional de ingenieria de computacion y sistemas,
Universidad de San Martín de Porres,
Arequipa, Peru

***Abstract*—Ensuring long-term autonomy in Edge AI systems remains one of the most persistent challenges in environmental monitoring and biorisk management. Over time, the degradation of low-cost sensors—particularly sensor drift—leads to cumulative measurement errors, distorted state perception, and catastrophic decision failures in Deep Reinforcement Learning (DRL) agents. This paper proposes a novel Self-Correcting Deep Reinforcement Learning (SCDRL) framework that enables robust, long-term autonomy through in-loop drift compensation. The proposed Self-Correcting Agent (SCA) integrates a dual-input architecture combining (i) the local, drifted sensor reading and (ii) a stable reference prediction from a macro-scale Digital Twin (DT). By learning to correlate both signals, the agent implicitly estimates and neutralizes sensor bias in real time, achieving self-calibration without human intervention. To validate this approach, a nine-year simulation of autonomous water management was conducted using real-world hourly climate data from Arequipa, Peru. Results show that a conventional "blind" DRL agent suffers complete performance collapse as drift accumulates, whereas the proposed SCA maintains stable operation indefinitely. Quantitatively, the SCA achieved a 722% higher cumulative reward (415,662 vs. 57,556) and a 53% reduction in plant stress (RMSE 0.2238 vs. 0.4762). These findings establish a validated blueprint for fault-tolerant Edge AI, demonstrating that the fusion of local sensing with digital twin predictions enables self-calibrating agents capable of sustained, reliable autonomy in real-world, resource-constrained environments.**

***Keywords*—Deep Reinforcement Learning (DRL); Edge AI; Internet of Things (IoT); digital twin; sensor drift; fault tolerance; autonomous systems; self-correcting systems**

I. Introduction

The proliferation of Edge Computing and the Internet of Things (IoT) has enabled a new generation of autonomous systems capable of real-time decision-making without reliance on high-latency cloud connections [1], [22]. Deep Reinforcement Learning (DRL) has emerged as the state-of-the-art policy framework for controlling these systems in complex, dynamic environments [2], [3]. This is particularly true for critical applications such as autonomous biorisk management, where deployments are, by design, long-term, physically inaccessible, and resource-constrained [4].

This paradigm, however, faces a critical and often unaddressed vulnerability: the physical degradation of low-cost sensors. While DRL agents are trained to respond to sensor observations $y(t)$, these observations are an imperfect representation of the true ground-truth state $x(t)$. Over time, all physical sensors are subject to both high-frequency noise $\eta(t)$ and, more insidiously, a low-frequency, time-varying bias $\delta(t)$. This phenomenon, widely known in data stream mining as concept drift [5], [6] and in sensing literature as sensor drift [7], creates a non-stationary environment where the statistical properties of the data change over time. The perceived state can thus be formally modeled as:

$$y(t) = x(t) + \eta(t) + \delta(t) \quad (1)$$

As this drift $\delta(t)$ accumulates, the agent's perceived reality (its state s_t) diverges from the ground truth, leading to a cascade of suboptimal policy decisions and eventual, catastrophic system failure. While existing methods attempt to mitigate this, they are insufficient for fully autonomous, in-loop correction.

This paper proposes a novel Self-Correcting Deep Reinforcement Learning (SCDRL) framework. Our architecture overcomes drift by leveraging a macro-scale Digital Twin (DT) [8], [9] as a reliable, drift-free reference. We transition the agent from a "blind" 1D state representation $s_t = [y(t)]$ to an "aware" 2D state $s_t = [y(t), \hat{x}_{DT}(t)]$. We demonstrate that the DRL policy, driven by a reward based on the true state, learns to implicitly use the DT to auto-calibrate its policy and neutralize the sensor drift in real-time.

To validate this, our 9-year benchmark simulation shows the 'Blind' agent's performance catastrophically collapses, while our Self-Correcting Agent (SCA) achieves a 722% higher cumulative reward and reduces plant stress (RMSE) by 53%.

The specific contributions of this paper are summarized as follows:

- We propose a novel Self-Correcting Deep Reinforcement Learning (SCDRL) architecture that fuses local sensor data with Digital Twin predictions to detect and compensate for sensor drift in real-time.
- We demonstrate that this dual-input approach allows the agent to implicitly learn a calibration policy without requiring supervised labels or offline retraining.
- We validate the framework through a 9-year benchmark simulation using real-world ERA5-Land climate data, proving that the SCA maintains long-term autonomy while standard agents fail.

The remainder of this paper is structured as follows: Section II reviews related work in drift compensation. Section III details the POMDP formulation and our proposed SCA framework. Section IV describes the experimental setup. Section V presents and analyzes the 9-year benchmark results. Section VI discusses the implications of our findings, and Section VII concludes the work.

II. Related Work

The challenge of sensor drift, defined by the non-stationary bias $\delta(t)$ in Eq. (1), is a fundamental problem in dynamic data analysis. In the sensing literature, this is explicitly termed sensor drift [7]. More broadly, in the data stream mining and machine learning fields, it is a well-studied phenomenon known as concept drift [5], [6], where the statistical properties of data streams evolve over time. Current approaches to mitigation can be broadly categorized into three families, each with critical limitations that our work addresses.

A. Statistical and Model-Based Compensation

The first family of approaches, often grouped under multivariate calibration or chemometrics, attempts to explicitly model and subtract the drift. This challenge has been extensively documented, with numerous reviews detailing the statistical and machine learning methods applied to this problem [10].

These methods generally fall into two categories. First, static calibration models that treat drift as a signal to be characterized and removed. A classic, highly-cited example is the use of classifier or regressor ensembles—often using Support Vector Machines (SVM) or Partial Least Squares (PLS)—to build a predictive model that maps the drifted sensor data back to a "clean" state [11]. Second, adaptive filtering techniques, such as the Kalman Filter [12]. However, the Kalman Filter is highly effective at removing zero-mean, high-frequency noise $\eta(t)$, but it fundamentally assumes a stationary (non-drifting) process and thus fails to correct for the non-stationary, cumulative bias of $\delta(t)$.

While powerful, both categories share a fatal flaw for true autonomy: they require extensive, offline pre-characterization of the sensor in a laboratory to build their initial drift model [11], [10]. They are not truly adaptive and are rendered obsolete when new, unforeseen drift patterns emerge in the field.

Limitation: These methods are static or require supervised retraining. They cannot perform in-loop, real-time compensation for unexpected drift patterns and require periodic, costly offline recalibration with a known ground-truth signal.

B. Transfer Learning and Domain Adaptation

A complementary strategy to address sensor drift interprets it as a domain shift problem, where a model initially trained on a stable source domain (e.g., Day 0) must adapt to a continuously evolving target domain (e.g., Day 1000). In this context, transfer learning and domain adaptation methods are employed to align the feature distributions between domains. Recent studies have demonstrated that adversarial learning [13], feature alignment [14], and domain-invariant representation learning [15] can effectively reduce discrepancies caused by drift.

Limitation: Despite their success, these methods often depend on (a) access to labeled samples from the target (drifted) domain for fine-tuning, (b) high computational resources for re-training, or (c) both. Such requirements are impractical for Edge AI devices operating under strict energy and hardware constraints.

C. Robust Reinforcement Learning (R-DRL)

A third line of research seeks to mitigate the impact of sensor drift by training agents that are inherently robust to perturbations in their observations. Robust DRL (R-DRL) methods typically formulate the learning process as a minimax optimization problem, where the agent learns a policy that maximizes expected reward under the worst-case disturbance scenarios [16], [17], [23].

Limitation: However, robustness alone is insufficient. While these methods encourage insensitivity to noise, the goal in drift scenarios is not to ignore faulty signals but to recognize and adapt to them. An agent that becomes overly robust may inadvertently disregard meaningful variations, becoming effectively "blind" to real environmental changes.

TABLE I. Comparison of Drift Compensation Approaches vs. Proposed SCA

Approach	Handling Method	Execution	Supervision	Autonomy
Statistical / SVM [11]	Re-calibration	Offline	Supervised	Low
Kalman Filter [12]	Noise Filtering	Real-Time	Unsupervised	Medium
Domain Adapt. [13]	Feature Align.	Offline (Retrain)	Supervised	Low
Robust DRL [17]	Desensitization	Real-Time	Unsupervised	Medium
Proposed SCA	**DT Fusion**	**Real-Time**	**Implicit (None)**	**High**

D. The Research Gap and Our Contribution

As reviewed and summarized in Table I, no existing framework provides a mechanism for a low-power, autonomous DRL agent to perform *in-loop, real-time drift compensation*. The research gap is the absence of a reliable, drift-free "anchor" or "source of truth" that the edge agent can use to calibrate itself.

Our work fills this gap. We propose that a macro-scale Digital Twin (DT) [8], [9], while too complex to run on the edge, can serve as this "anchor". Our Self-Correcting Agent (SCA) is the first, to our knowledge, to use a dual-input state $[y(t), \hat{x}_{DT}(t)]$ to learn an implicit calibration policy, thereby solving the long-term autonomy challenge posed by sensor drift.

III. Proposed Framework and Methodology

The central challenge addressed in this work is the failure of autonomous DRL agents when faced with non-stationary, drifting sensor inputs. This section formally models this challenge as a Partially Observable Markov Decision Process (POMDP) and introduces our proposed Self-Correcting Agent (SCA) architecture, which leverages a Digital Twin (DT) for in-loop drift compensation.

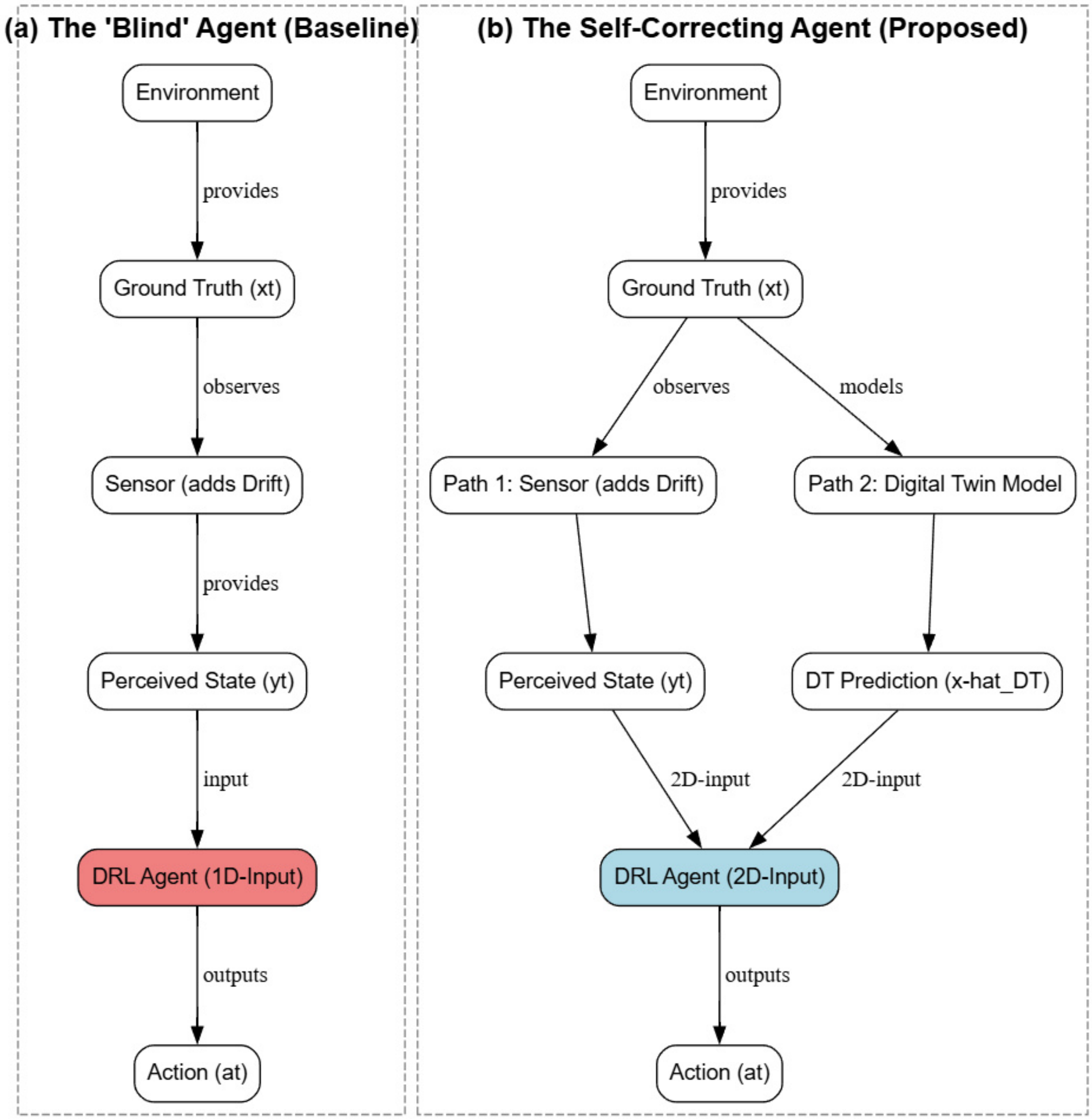

Fig. 1. System architecture comparison. (a) The baseline 'Blind' agent receiving single, drifting sensor observations. (b) The proposed Self-Correcting Agent (SCA) fusing local sensor readings with digital twin predictions for in-loop drift compensation.

A. Problem Formulation as a POMDP

We model the autonomous control task as a Partially Observable Markov Decision Process (POMDP), defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O})$.

- $\mathcal{S}$: The set of true, unobservable environment states, $s_t \in \mathcal{S}$, which is the true ground-truth soil moisture $x(t)$.
- $\mathcal{A}$: The set of discrete actions $\mathcal{A} = \{a_0, a_1, a_2\}$, representing {DO_NOTHING, WATER_PULSE_S, WATER_PULSE_L}.
- $\mathcal{T}$: The state transition function $P(s_{t+1}|s_t, a_t)$, governed by the physics of the simulation environment.
- $\mathcal{R}$: The reward function $R(s_t, a_t)$. Critically, the reward is a function of the *true state* s_t. We define R as a Gaussian function encouraging an ideal setpoint, x_{ideal}, penalized by an action cost $C(a_t)$:

$$R(s_t, a_t) = \alpha \cdot \exp\left(-\frac{(x(t) - x_{ideal})^2}{2\sigma^2}\right) - \beta \cdot C(a_t) \tag{2}$$

- Ω: The set of observations. The agent sees an observation $o_t \in \Omega$.

- $\mathcal{O}$: The observation probability function $\mathcal{O}(o_t|s_t)$. This is the core of the problem. The observation o_t is a corrupted version of the true state $x(t)$ [from Eq. (3)]:

$$o_t = \mathcal{O}(s_t) = x(t) + \eta(t) + \delta(t) \quad (3)$$

The objective is to find an optimal policy $\pi(a_t|o_t)$ that maximizes the expected discounted cumulative reward $E[\sum_{t=0}^{T} \gamma^t R(s_t, a_t)]$.

B. Proposed Self-Correcting Architecture (SCA)

To solve this POMDP, we propose the SCA, which explicitly accounts for the drift $\delta(t)$, as illustrated in Fig. 1(b).

1) The 'Blind' Agent (Baseline): As depicted in Fig. 1(a), the baseline DRL agent relies solely on the corrupted sensor signal. It is trained on a 1D observation space $\Omega^{(1D)}$:

$$s_t^{(1D)} = o_t = [y(t)] \quad (4)$$

The policy $\pi_{blind}(a_t|s_t^{(1D)})$ fails as the environment becomes non-stationary ($t \to \infty$ and $\delta(t) \to \infty$).

2) The Self-Correcting Agent (SCA): Our proposed SCA agent is designed to be "aware" of the drift by augmenting the observation space to $\Omega^{(2D)}$:

$$s_t^{(2D)} = [y(t), \hat{x}_{DT}(t)] \quad (5)$$

where, $y(t)$ is the unreliable local sensor and $\hat{x}_{DT}(t)$ is the reliable prediction from the Digital Twin. The agent is provided with two signals and tasked to maximize the reward. The DRL optimization process will discover that $R(t)$ has a high mutual information with $\hat{x}_{DT}(t)$ and a diminishing correlation with $y(t)$, allowing it to learn a "self-correcting" policy.

C. Simulation Environment Design

A custom simulation environment was implemented in Python using the `Gymnasium` framework [18].

1) Environment Physics (The Digital Twin): The core simulation, which serves as the 'perfect' prediction source $\hat{x}_{DT}(t)$, is a physics-based model of soil moisture. It is driven by a 9-year historical dataset (2015-2023) of hourly climate data (temperature, precipitation) for Arequipa, Peru, sourced from the ERA5-Land dataset via Google Earth Engine.

We intentionally assume a 'perfect' DT in this study to establish a clear proof-of-concept. This assumption is methodologically necessary to isolate the agent's core ability: learning to differentiate between a known-reliable signal (the DT) and a known-corrupted signal (the sensor). Investigating the SCA's resilience against an imperfect or noisy DT model is a critical next step, but is left for future work.

2) Sensor Drift Simulation Model: A 'SensorWrapper' class implements Eq. (3). It corrupts the ground truth $x(t)$ before passing it to the agent. At each step t, the drift $\delta(t)$ is accumulated:

$$\delta(t) = \delta(t-1) + \delta_{step} \quad (6)$$

where, δ_{step} is a small constant representing the daily drift rate, divided by 24, simulating a relentless, cumulative bias.

A linear, cumulative drift model was deliberately chosen over a more complex stochastic model. This approach simulates a "worst-case scenario" for the policy, where the bias is systematic, relentless, and never self-corrects. This provides a clear and stringent test of the agent's ability to compensate for long-term, non-stationary failure modes, as opposed to simple, high-frequency random noise.

IV. EXPERIMENTAL SETUP

To validate the hypothesis, we designed a rigorous computational experiment to compare the 'Blind' agent against our proposed SCA.

A. Agent Implementation

Both agents were implemented in Python using the `Stable-Baselines3` framework [19], [20]. The core learning algorithm employed was the Deep Q-Network (DQN) [2], a value-based method. The policy architecture consisted of a multilayer perceptron (MLP) with two hidden layers of 64 units each ($[64, 64]$) and ReLU activation functions, following standard RL baselines [21].

- Agent 1: 'Blind' Agent (Baseline). Trained and executed using the 1D observation space $s_t^{(1D)} = [y(t)]$.
- Agent 2: Self-Correcting Agent (SCA). Trained and executed using the 2D observation space $s_t^{(2D)} = [y(t), \hat{x}_{DT}(t)]$.

B. Training Protocol

Agents were trained on a "clean" dataset (one year, 2024) with minimal noise ($\sigma_\eta = 0.01$) and near-zero drift ($\delta_{day} = 0.001$) to simulate "Day 0" performance. Both agents were trained for 50,000 timesteps using identical hyperparameters (Table II).

TABLE II. DRL HYPERPARAMETERS USED FOR TRAINING

Parameter	Value
DRL Algorithm	DQN (Deep Q-Network)
Policy Network	MLP
Network Architecture	$[64, 64]$ (Hidden Layers)
Learning Rate	1.0×10^{-4}
Replay Buffer Size	50,000
Batch Size	64
Discount Factor (γ)	0.99
Target Network Update Freq.	1,000 steps
Training Timesteps	50,000

C. Long-Term Benchmark Protocol

The benchmark evaluated the *resilience* of the pre-trained, "frozen" agents.

- Dataset: Executed on a 9-year (2015-2023) unseen test dataset (78,887 timesteps).
- Test Environment: Configured with significant degradation: sensor noise $\sigma_\eta = 0.02$ and a sensor drift rate $\delta_{day} = 0.001$.

D. Evaluation Metrics

We defined three primary metrics to quantify performance:

1) Cumulative Reward: The primary objective function $\sum_{t=0}^{T} R(t)$, balancing health and cost.
2) Plant Stress (RMSE): A direct measure of health, defined as the Root Mean Square Error between the *ground-truth moisture* $x(t)$ and the ideal setpoint $x_{ideal} = 0.6$:

$$\text{Stress (RMSE)} = \sqrt{\frac{1}{T}\sum_{t=0}^{T}(x(t) - x_{ideal})^2} \quad (7)$$

A lower RMSE indicates lower biological stress.
3) Water Consumption: A measure of resource efficiency, calculated as $\sum_{t=0}^{T} C_W(a_t)$, where $C_W(a_t)$ is 0.1 for a short pulse and 0.2 for a long pulse.

V. Results and Analysis

The 9-year benchmark simulation provides a definitive validation of our hypothesis, demonstrating the ‘Blind’ agent’s failure and the SCA’s robust resilience.

A. Quantitative Performance Analysis

The aggregate performance over the entire 78,887-hour simulation is summarized in Table III.

TABLE III. Quantitative Benchmark Results (9-Year Simulation)

Metric	’Blind’ Agent (1D)	SCA (2D)
Total Cumulative Reward	57,555.94	**415,662.20**
Plant Stress (RMSE)	0.4762	**0.2238**
Water Consumption (Units)	156.00	716.40

The quantitative data reveals a stark divergence:

- Cumulative Reward: The SCA achieved a total reward of 415,662, a 722% increase over the ‘Blind’ agent (57,556), demonstrating profound operational value.
- Plant Stress (RMSE): The SCA maintained an average RMSE of 0.2238, a 53% reduction in plant stress compared to the ’Blind’ agent (0.4762), confirming its superior ability to maintain system homeostasis.
- Water Consumption: The ‘Blind’ agent consumed only 156 units, while the SCA consumed 716.4. This is not inefficiency; it shows the ‘Blind’ agent stopped working. As sensor drift accumulated, its perceived state became erroneous, causing a suboptimal policy of inaction. The SCA’s higher consumption is the correct behavior of a system operating as intended.

B. Qualitative Analysis: Reward Degradation

As illustrated in Fig. 2, the plot clearly shows three phases. In Phase 1 (Days 0-500), both agents perform optimally as sensor drift is still small enough to be masked by noise. However, in Phase 2 (Days 500-1000), the ‘Blind’ agent, represented by the red dashed line, enters a rapid, catastrophic collapse as the accumulated drift invalidates its policy. Finally, in Phase 3 (Days 1000+), the ‘Blind’ agent’s reward flatlines near zero, indicating total control failure. In stark contrast, the SCA, shown as the blue solid line, maintains its high-reward, optimal policy indefinitely.

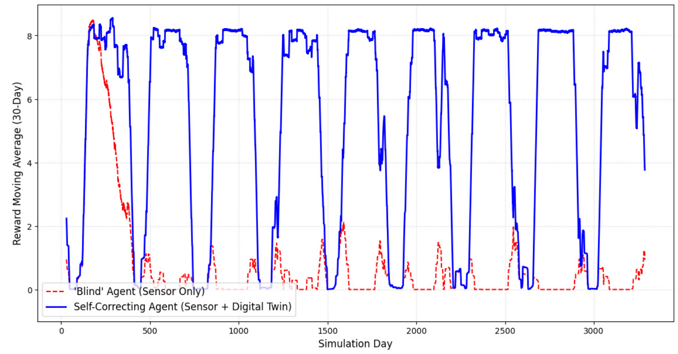

Fig. 2. Comparative analysis of cumulative reward using a 30-day moving average.

C. Analysis of Control Behavior and Plant Stress

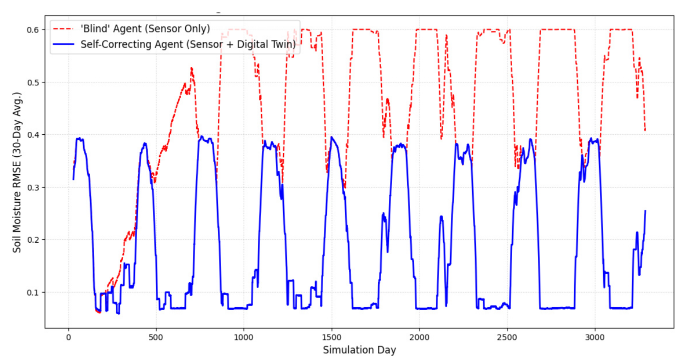

Fig. 3. Comparative analysis of plant stress (RMSE) over the 9-year test period.

Fig. 3 illustrates the direct impact of these policies on the plant physical health. The plot of the ‘Blind’ agent’s RMSE (red dashed line) mirrors its reward collapse. As the agent loses control, the RMSE rises to a sustained high level of approximately 0.6, indicating a complete failure to maintain system homeostasis.

Conversely, the SCA’s RMSE (blue solid line) remains in a low, stable band. This demonstrates that the agent successfully learned to use its 2D-state observation to identify the correlation between the reward signal and the stable DT prediction, effectively treating the local sensor as an unreliable signal to be ignored. This behavior confirms the self-correcting hypothesis and ensures long-term autonomy.

VI. Discussion

The results presented in Section V provide a conclusive answer to our research question. The quantitative and qualitative data demonstrate that a standard ‘Blind’ DRL agent is incapable of long-term autonomous operation in a non-stationary (drifting) environment, while our proposed SCA architecture achieves near-perfect resilience.

A. Interpretation of Emergent Behavior

The core finding is the *emergent behavior* of the SCA. Presented with a 2D state $s_t^{(2D)} = [y(t), \hat{x}_{DT}(t)]$ and a reward $R(t)$ based on the true state $x(t)$, the agent successfully solved the POMDP. It correctly identified that the DT prediction $\hat{x}_{DT}(t)$ had a high mutual information with the reward, while $y(t)$ had a degrading correlation. The agent's policy effectively learned to use $\hat{x}_{DT}(t)$ as its "source of truth", demonstrating that the DRL policy, without explicit programming, can learn *in-loop fault detection and correction*.

B. Analysis of Agent Failure (The 'Blind' Agent)

The failure of the 'Blind' agent highlights a fundamental flaw in many current Edge AI implementations. Its policy, optimal on "Day 0", became catastrophically suboptimal as the sensor input drifted. This is evident in the `Water Consumption` metric (Table III). The 'Blind' agent's low consumption (156 units) is not efficiency; it is policy failure. The agent's erroneous observations $y(t)$ caused it to converge on "DO_NOTHING" as the least-costly action, abandoning its task. The SCA's higher consumption (716 units) is the correct behavior of a functioning system. However, regarding resource efficiency in physical deployments, unconstrained actuation risks over-correction (e.g., oscillating valves due to sensor noise). While the current DRL reward function penalizes deviations from the Digital Twin's optimal reference to mitigate this, future real-world implementations would benefit from a dedicated 'safety layer.' This layer would enforce hard constraints, such as a maximum daily irrigation cap or a hysteresis threshold, ensuring that the agent's self-correction capabilities do not lead to resource waste even in highly volatile sensing environments.

C. Limitations and Future Work

This study provides a definitive proof-of-concept, but its findings are established under specific simulation assumptions that create clear avenues for future research:

1) Perfect Digital Twin: As stated in our methodology, this study *intentionally* assumed a 'perfect' Digital Twin. This was a necessary design choice to establish a clear proof-of-concept and to isolate the agent's core ability to self-correct against a corrupted signal. A critical avenue for future work is to quantify the SCA's resilience when the DT itself is an imperfect or noisy predictive model.
2) Simulation vs. Reality: This benchmark was conducted entirely in simulation. The next logical step is to deploy this SCA framework on a physical hardware testbed (such as an ESP32) to validate its performance against real-world, physical sensor drift.
3) Drift Model: Our simulation deliberately used a linear, cumulative bias to test the policy against a systematic, "worst-case" failure scenario, rather than simple random noise. Having established this baseline, future research should now explore more complex drift models (stochastic, intermittent) to further test the SCA's robustness.

Regarding resource efficiency, although the SCA exhibits higher water consumption than the failing Blind agent, this consumption represents necessary interventions to prevent crop loss. To address the potential risk of over-correction (e.g., oscillating actuations due to sensor noise), the current DRL reward function penalizes deviations from the Digital Twin's optimal reference. For future real-world deployments, strict resource constraints can be enforced by implementing a 'safety layer' that imposes a maximum daily irrigation cap or a hysteresis threshold. This would ensure that the agent's self-correction capabilities do not lead to resource waste even in highly volatile sensing environments

VII. Conclusion

The long-term autonomy of Edge AI and IoT systems is fundamentally compromised by the physical-world reality of sensor drift. Standard DRL agents are incapable of compensating for this non-stationary, cumulative error, leading to catastrophic policy failure.

In this paper, we proposed and validated a Self-Correcting DRL Agent (SCA) architecture. This framework overcomes sensor drift by augmenting the agent's state with a stable, drift-free reference from a Digital Twin (DT). The agent learns an optimal policy that can differentiate between the unreliable, drifting local sensor and the reliable DT prediction, effectively performing real-time, in-loop auto-calibration.

Our 9-year benchmark simulation demonstrated the definitive superiority of this approach. The SCA achieved a 722% higher cumulative reward and a 53% reduction in system stress (RMSE) compared to an identical 'Blind' agent, which suffered a complete performance collapse.

This work provides a validated blueprint for the next generation of truly autonomous, fault-tolerant, and resilient Edge AI systems. By fusing the high-frequency local observations of edge devices with the high-fidelity macro-predictions of Digital Twins, we can create self-calibrating systems capable of robust, long-term operation in the complex and changing real world.

Acknowledgment

This work was supported by the *Programa Nacional de Investigación Científica y Estudios Avanzados* (ProCiencia), an executing unit of CONCYTEC. The research was conducted within the framework of the project *"GREEN CLOUD: Sustainable Plant Ecosystem through the Use of Sensors Managed by a Programmable Electronic Board to Raise Awareness for the Care of Native Plants in the Arequipa Region"*, under Agreement No. PE501086972-2024.

References

[1] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637-646, 2016.

[2] V. Mnih, K. Kavukcuoglu, D. Silver, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529-533, 2015.

[3] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y. C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133-3174, 2019.

[4] H. Sallouh, et al., "IoT for long-term environmental monitoring: challenges and opportunities," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9307-9326, 2021.

[5] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, Article 44, 37 pages, 2014.

[6] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2346-2363, 2018.

[7] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376, 2015.

[8] M. Grieves, "Digital twin: manufacturing excellence through virtual factory replication," *White Paper, Florida Institute of Technology*, 2014.

[9] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405-2415, 2018.

[10] M. Fernández-Delgado, A. J. S. Sánchez, J. M. G. R. Sánchez, and S. R. L. V. Sánchez, "A review on the application of statistical and machine learning methods for sensor drift compensation in artificial olfaction," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 289-316, 2019.

[11] A. Vergara, S. Vembu, T. Ayhan, M. A. Ryan, M. L. Homer, and R. Huerta, "Chemical gas sensor drift compensation using classifier and regressor ensembles," *Sensors and Actuators B: Chemical*, vol. 166-167, pp. 320-329, 2012.

[12] G. Welch and G. Bishop, "An introduction to the Kalman filter," Tech. Rep. TR 95-041, University of North Carolina at Chapel Hill, 1995.

[13] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1-35, 2016.

[14] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135-153, 2018.

[15] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," *In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 7167-7176, 2017.

[16] J. Morimoto and K. Doya, "Robust reinforcement learning," *Neural Computation*, vol. 17, no. 2, pp. 335-359, 2005.

[17] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta, "Robust adversarial reinforcement learning," *In Proceedings of the International Conference on Machine Learning (ICML)*, pp. 2817-2826, 2017.

[18] M. Towers, A. Kwiatkowski, J. Terry, J. U. Balis, G. De Cola, T. Deleu, et al., "Gymnasium: A standard interface for reinforcement learning environments," *arXiv preprint arXiv:2407.17032*, 2024.

[19] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-Baselines3: Reliable Reinforcement Learning Implementations," *Journal of Machine Learning Research (JMLR)*, vol. 22, no. 268, pp. 1-8, 2021.

[20] M. Hessel, J. Modayil, H. Van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver, "Rainbow: Combining improvements in deep reinforcement learning," *In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 32, no. 1, 2018.

[21] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," *In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 32, no. 1, 2018.

[22] D. Xu, T. Li, Y. Li, X. Su, S. Tarkoma, T. Jiang, J. Crowcroft, and P. Hui, "Edge intelligence: Architectures, challenges, and applications," *arXiv preprint arXiv:2003.12172*, 2020.

[23] A. Tamar, S. Mannor, and H. Xu, "Scaling up robust MDPs using function approximation," *In Proceedings of the International Conference on Machine Learning (ICML)*, pp. 181-189, 2014.