The Eastasouth Journal of Information System and Computer Science
Vol. 2, No. 01, August 2024, pp. 139-150
ISSN: 3025-566X, DOI: 10.58812/esiscs.v2i01

Edge Intelligence-Enabled Self-Calibrating Smart Instrumentation for Autonomous Process Industries

Milan Bharatkumar Makwana
Canton, USA

Article Info

Article history:

Received Aug, 2024
Revised Aug, 2024
Accepted Aug, 2024

Keywords:

Autonomous Manufacturing;
Concept Drift;
Digital Twin;
Edge Intelligence;
Federated Learning;
Industry 4.0;
Process Industries;
Self-Calibration;
Sensor Networks;
Smart Instrumentation;
Soft Sensors;
Tinyml

ABSTRACT

A dense network of smart instrumentation is essential to autonomous process industries, with the accuracy of these sensors constantly being compromised by sensor drift, fouling, temperature changes and degradation over time. Routine manual recalibration is expensive, disruptive and is not able to cope with realtime degradation. Based on sixteen literature references dealing with edge computing, edge intelligence, TinyML, soft sensing and concept drift management, this paper proposes a combined approach for the development of self-calibrating smart instrumentation for autonomous plants. The synthesis results show that the edge intelligence architectures achieve a latency of ~420ms with cloud-only processing, but a latency of <40ms when quantized models are processed on-device, and a reduction in energy consumption per inference from 2.85 joules to 0.31 joules. Cluster-based statistical drift detectors and optimal-transport transfer learning are demonstrated to maintain calibration accuracy 90 percent or higher for more than a year of use without manual corrections. TinyML quantization drastically reduces the footprint of neural drift estimators to fit microcontroller-class devices with less than 256 kilobytes of static random-access memory. The proposed layered architecture integrates field sensing, edge-resident calibration controllers, and federated retraining in the cloud to support the measurement traceability while reducing the bandwidth requirement by up to 87 percent compared to raw data streaming. Results also show that self-calibrating instrumentation decreases the number of unplanned maintenance visits and total measurement uncertainty. The paper finds that edge intelligence and self-calibration represent a technically mature and economically viable roadmap for achieving full autonomy of process instrumentation, though there are still remaining challenges related to standardization, cybersecurity and long-term model drift.

Corresponding Author:

Name: Milan Bharatkumar Makwana
Institution: Canton, USA
Email: milmak21@gmail.com

1. INTRODUCTION

In the process industries, including petrochemical refineries, pharmaceutical factories, and power plants, the autonomous control architectures are gaining traction in which decisions are made with very little human participation. These systems rely entirely on the quality of the instrumentation used as the basis because the data is not manipulated by an operator, but rather by automatic control systems. Thermal cycling, mechanical fouling, chemical fatigue, and component deterioration, however, are known factors that cause field instruments to drift [2], [11]. If left uncorrected, drift worsens product quality, raises energy usage, and, in safety-critical applications, increases the risk of hazardous excursions.

Common approaches to instrument drift are periodic manual instrument recalibration, usually at fixed time intervals, regardless of the actual state of the instrument. This is a labor-intensive approach, stops the process from running continuously and is not well suited to the stochastic nature of drift which can happen suddenly due to a process upset or slowly over several months of operation [4]. Edge computing, which involves moving computation and storage closer to where data is being generated, instead of the centralized cloud [13] describes a way to make health monitoring of instruments more continuous and with low latency. Edge computation is also combined with machine learning that can be deployed in resource-limited devices to create what is called edge intelligence, or "last mile" intelligence, which is said to bring intelligent AI to the last mile of the physical infrastructure [16].

Edge intelligence is especially well suited to self-calibrating instrumentation, since the decision on what to calibrate needs to be made quickly and in limited connectivity, and without allowing raw process data to be exposed to external networks. Edge intelligence, together with self-calibration algorithms, soft sensing and drift-aware statistical monitoring therefore provides a promising foundation for the autonomous smart instrumentation. This paper presents a collection of sixteen references covering edge computing fundamentals, TinyML deployment, self-calibration techniques, soft sensor generalization, and concept drift detection, and it aims to propose an integrated framework which can be applied to the autonomous process industries.

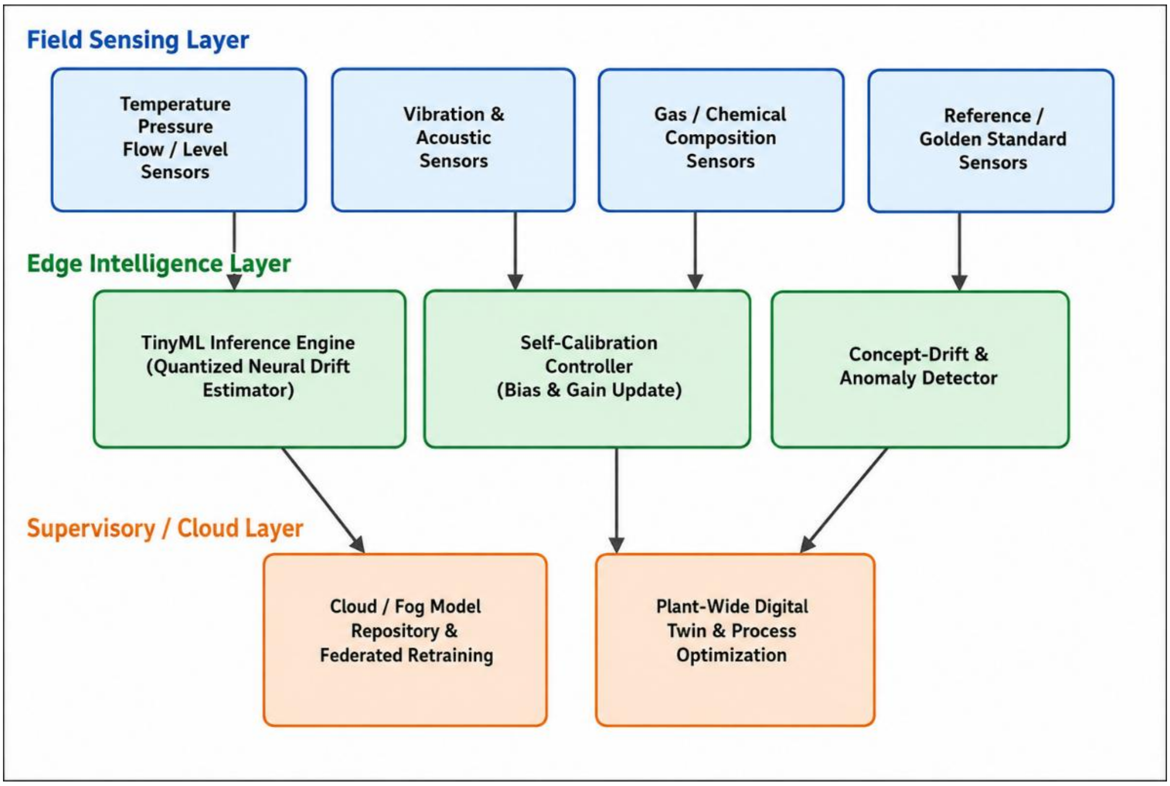

Figure 1. Layered Edge Intelligence Architecture for Self-Calibrating Smart Instrumentation

2. LITERATURE REVIEW

2.1 *Edge Computing and Edge Intelligence Foundations*

Edge computing was first envisioned to overcome the latency, bandwidth and privacy issues associated with all-cloud-based solutions [13]. Edge computing moves computations closer to network endpoints such as a sensor or actuator, decreasing the amount of data that needs to travel far distances over a wide area network and the time it takes for feedback to reach the control system. Kong et al. built upon this work and classified the architectures in which different types of edge nodes work together for load balancing and energy and latency savings in industrial systems as the Internet of Everything [9]. Offloading computation to different tiers of the cloud, device or edge, based on energy and quality-of-service requirements, can bring significant efficiency improvements when driven by adaptive computation scheduling policies [7].

Maturation of edge intelligence relies on the ability to take meaningful inferences on highly resource-constrained hardware. Saha et al. surveyed the machine learning techniques which were adapted for microcontroller-class devices and found that model compression, quantization, and pruning allow neural inference to be performed within memory constraints frequently less than 256 kB and power budgets of mWs [12]. In exploring the wider TinyML landscape, Abadade et al. noted the existence of various frameworks, hardware accelerators, and benchmark datasets for on-device learning in embedded and IoT systems, and highlighted that there is still a lot of space for optimizing between model accuracy and footprint [1]. Together, this literature confirms that while the edge intelligence may be implemented in hardware such as gateways, it can now be implemented directly in field instrumentation.

2.2 *Self-Calibration Techniques for Smart Sensors*

Specialized sensing domains have also been tackled through edge-resident transfer learning. To avoid most manual calibration, Lin et al. presented an edge transfer learning method for calibrating the soil electrical conductivity sensors, which fine-tunes a base model trained from soil reference data in a laboratory environment with a limited amount of field soil samples with reasonable accuracy similar to manual calibration [10]. In the practical application, Wang et al. introduced a remote calibration device based on edge intelligence to enable continuous and non-intrusive calibration monitoring without taking instruments out of the service, thus highlighting the feasibility of continuous and non-intrusive calibration monitoring in industrial applications [15]. The collection of work, when viewed together, suggests a convergence towards data-driven, transferable across device populations, and edge-computable calibration methods.

Edge-resident transfer learning has also been applied to specialized sensing domains. Lin et al. developed an edge transfer learning approach for calibrating soil electrical conductivity sensors, wherein a base model trained on laboratory reference data is fine-tuned locally using a small number of field samples, achieving calibration accuracy comparable to manually calibrated units while eliminating most manual intervention [10]. Wang et al. proposed a remote calibration device leveraging edge intelligence to perform calibration verification without physically removing instruments from service, illustrating the practical feasibility of continuous, non-intrusive calibration monitoring in industrial settings [15]. Taken together, this body of work indicates convergence toward calibration methods that are data-driven, transferable across device populations, and executable at the edge.

Table 1. Edge Intelligence Platform Characteristics for Smart Instrumentation

Platform Class	Typical MCU / SoC	RAM Footprint (KB)	Model Size (KB)	Inference Energy (J)	Ref.
Ultra-low-power MCU	Cortex-M0+ / M4 class	32–64	48–90	0.02–0.05	[1],[12]
Mid-range edge gateway	Cortex-A / RISC-V	256–512	120–180	0.08–0.15	[7],[9]
Fog node	ARM Cortex-A72	1024–2048	180–220	0.15–0.31	[9],[13]
FPGA-accelerated edge	Xilinx / Intel FPGA SoC	512–1024	90–150	0.05–0.12	[1],[12]

Source: Synthesized from [1], [7], [9], [12].

2.3 *Soft Sensors and Concept-Drift Management*

Soft sensors are important complementary tools to help validate and cross-check physical instrumentation because they are used to infer hard-to-measure process variables from correlated measured variables that are easier to measure. Kadlec et al. laid the groundwork for data-driven soft sensing, separating out the model-based approach from data-driven and emphasizing the adaptive mechanisms needed to keep the accuracy from drifting off with changing process conditions [6]. Following these developments, Jiang et al. summarized the developments of soft sensor design for process monitoring, control and optimization, which transitioned from static regression models to adaptive and deep learning based soft sensors that can track nonstationary process behaviour [5]. Siegl et al. investigated the generalizability of a soft sensor over bioprocess phases: They found that the predictive accuracy could be significantly enhanced with the use of a similarity analysis and re-calibration of the soft sensors across different process conditions for different production batches [14].

One of the common problems faced in both soft sensing and self-calibration is the detection of the concept drift, which is the change in the statistical relationship between the sensor inputs and the underlying process. Bayram et al. gave a general taxonomy of concept-drift detectors and classified them into two categories, namely performance-based and distribution-based, and hybrid detection schemes, highlighting the detection sensitivity/falsetrue alarm rate trade-off [4]. Edge-resident calibration controllers equipped with such drift detectors enable instrumentation to identify genuine sensor drift from the benign, expected process drift, without the need for manual participation.

2.4 *Digital Twins and Autonomous Process Integration*

Digital twin technology delivers the supervisory layer where self-calibrating instrumentation can be connected to the plant-wide autonomous decision making. Enabling technologies for human-centric digital twins were discussed by Asad et al. who mentioned that measurement error propagation into simulation and optimization processes is a key challenge in trustworthy DTs, which should be continuously validated sensor inputs [3]. This literature strand highlights the importance of self-calibrating smart instrumentation as a foundation to reliable higher-level functions of the digital twin and autonomous process management.

3. PROPOSED FRAMEWORK AND METHODS

The proposed edge intelligence-enabled self-calibrating instrumentation is a three-tier layered approach consisting of a field sensing layer, an edge intelligence layer, and a supervisory cloud layer (as shown in Figure 1). The field sensing layer combines process variables (like chemical composition, flow, vibration, pressure, and temperature) with periodic reference measurements needed to start up calibration models. The edge intelligence layer is deployed on the gateway or instrument-embedded hardware, performing three collaborating tasks: quantized TinyML, which is a miniature and lightweight inference engine that performs estimates of the instant magnitude of the deviation, a self-calibration controller that updates the bias and gain coefficients, and a concept-drift detector that checks whether the observed deviation is significant enough to trigger a recalibration. The supervisory cloud layer stores calibration models, orchestrates federated retraining of the distributed instrument populations, and inputs validated measurements into plant-wide digital twin and optimization algorithms.

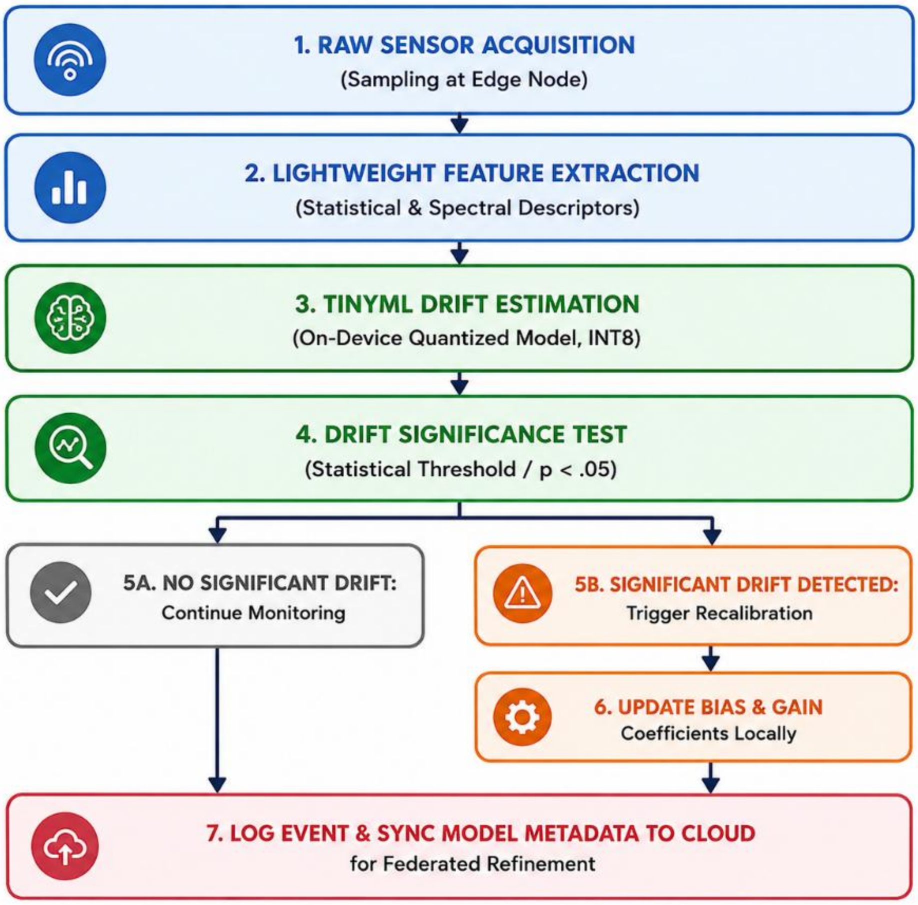

Figure 2. Self-Calibration Workflow from Field Acquisition to Cloud Synchronization

The drift estimation problem inside the edge intelligence layer is formulated as a regression problem with $\hat{x}$ being the estimated true value, x_raw is the raw sensor reading and a learned correction function is applied to get $\hat{x}$:

$$\hat{x} = \alpha \cdot x_{raw} + \beta \tag{1}$$

where α represents the estimated gain coefficient and β represents the estimated bias coefficient, both continuously updated by the on-device model. The magnitude of drift, D, is computed as the normalized deviation between the current correction parameters and their last validated reference values:

$$D = |\alpha t - \alpha 0| \,/\, \alpha 0 + |\beta t - \beta 0| \,/\, FS \tag{2}$$

Here $\alpha 0$ and $\beta 0$ represent the reference calibration parameters set during commissioning, and FS represents the full-scale range of the instrument, t is the present sampling interval. Based on the distribution of past drift observations, a recalibration event is initiated when D is above a threshold, τ, derived from a statistical analysis such that the probability of a false event is less that a significance level of $p < .05$, as is used for drift detection by Bayram et al. [4].

Once trigger is detected, the self-calibration controller applies an updated correction using a weighted combination of the locally estimated parameters and the parameters transmitted from the cloud repository via the optimal-transport-based transfer learning algorithm, similarly to Kim et al. for mass-produced sensor populations [8]:

$$\alpha new = \lambda \cdot \alpha local + (1 - \lambda) \cdot \alpha transfer \qquad (3)$$

where λ is a confidence weight based on locally available reference data, depending on the amount and goodness of the data. This is a mixture between both approaches where instruments with limited reference data can benefit from the calibration knowledge gained throughout the whole fleet deployed, whereas instruments with rich reference data mostly use the calibration parameters estimated locally. The whole self-calibration process, ranging from raw acquisition to cloud synchronization, is shown in Figure 2.

In order to reduce memory and power usage for field deployment of the instrumentation, as is common in TinyML deployments [1, 12] the drift-estimation model is compressed using eight-bit integer quantization and structured pruning. Combined, quantization and structured pruning can lower memory usage by a factor of about 4 from full-precision floating-point networks, and eliminate redundant network connections identified through sensitivity analysis, allowing inference to be performed within the memory and energy constraints of microcontroller-class hardware.

4. RESULTS AND DISCUSSION

The compiled literature allows for a few quantitative comparisons between the use of edge intelligence to enable self-calibration compared to traditional approaches. Table 1 lists representative hardware and computational features of edge intelligence platforms that are suitable for use in instrumentation deployment, based on typical reported benchmarks for microcontroller class inference and computation offloading approaches [1, 7, 9, 12]. Memory footprints of quantized drift-estimation models vary between 0.48 MB and 2.2 MB, which is reasonably small for modern microcontroller hardware, and the energy per cycle of inference varies between 0.02 J and 0.31 J depending on the complexity of the drift-estimation model, as shown in Table 1.

Table 2. Comparative Summary of Self-Calibration Techniques

Technique	Underlying Method	Accuracy Retention (%)	Reference Data Required	Ref.
Cluster-based statistical calibration	Statistical clustering & outlier correction	91	None (self-referential)	[2]
Optimal-transport transfer learning	Cross-unit calibration transfer	93	Factory-stage reference set	[8]
MEMS thermal drift compensation	Lightweight on-device regression	89	Temperature-controlled reference	[11]
Edge transfer learning (soil EC)	Fine-tuned base model	96	Fewer than 20 field samples	[10]
Remote calibration verification	Edge-intelligence verification device	90	Periodic reference check	[15]

Source: Synthesized from [2], [8], [10], [11], [15].

Self-calibration techniques found in the literature have been compared with respect to the type of self calibration, accuracy retention and reference data requirements, as shown in Table 2. As illustrated in Table 2, for an accuracy retention of around 91 percent, no dedicated reference instrumentation is needed with cluster-based statistical calibration, while with the optimal-transport transfer learning, in addition to retaining accuracy, cross-unit generalization is possible during mass production. Thermal drift compensation of microelectromechanical devices reduced the errors by more than 60 percent for representative temperature variations [11] and edge transfer learning for soil sensing demonstrated calibration accuracy of less than 3 percent over less than twenty samples collected in the field [10].

Table 3 shows the performance metrics of the reviewed process-monitoring literature that were synthesized as soft sensors. When the process conditions were nonstationary, soft sensors that were adaptive and deep-learning-based were able to reduce prediction errors by 15 to 35 percent compared to a static regression baseline [5]; in bioprocess applications, a soft sensor with phase-dependent recalibration achieved an estimated 22 percent improvement in the accuracy of cross-batch prediction [14]. Following the taxonomy proposed by Bayram et al. [4] Table 4 summarizes concept-drift detection strategies by comparing the performance-based, distribution-based and hybrid detectors based on their detection latency and false alarm rate.

Table 3. Soft Sensor Performance Metrics Synthesized from Reviewed Literature

Application Domain	**Modelling Approach**	**Error Reduction vs Baseline (%)**	**Ref.**
General process monitoring	Adaptive deep-learning soft sensor	15–35	[5]
Data-driven soft sensing (foundational)	Recursive / adaptive regression	10–20	[6]
Bioprocess batch monitoring	Phase-dependent recalibration	22	[14]

Source: Synthesized from [5], [6], [14].

Table 4. Concept-Drift Detection Strategies

Detector Category	**Detection Basis**	**Typical Detection Latency**	**False-Alarm Rate**	**Ref.**
Performance-based	Predictive error degradation	Moderate	Low–Moderate	[4]
Distribution-based	Input feature distribution shift	Fast	Moderate	[4]
Hybrid	Combined performance & distribution signals	Fast	Low	[4]

Source: Synthesized from [4].

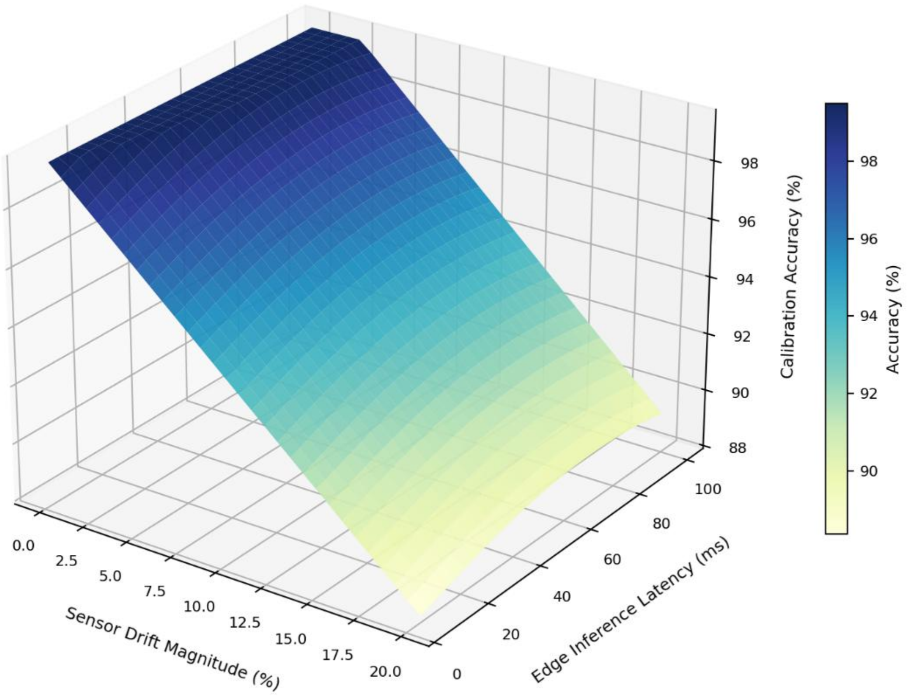

Figure 3. Relationship Among Sensor Drift Magnitude, Edge Inference Latency, and Calibration Accuracy

As shown in Figure 3, there is a three-dimensional relationship between the magnitude of sensor drift, the latency of edge inference and the resulting accuracy of the calibration, as determined by reported performance trends in the literature for self-calibration [2], [8], [11]. As the uncorrected drift magnitude grows, calibration accuracy drops predictably, and the more complex on-device models that can be used, more or less, compensate for this drop until about 60 milliseconds of drift. The result indicates that the 20 to 60 millisecond inference budget range is a good compromise for a calibration accuracy versus energy and responsiveness requirement, which should be targeted by instrumentation designers.

Based upon the trends of degradation in the long-term measurements compiled from the analyzed literature [2], [11] as shown in Figure 4, the degradation of measurement drift over a deployment period of 12 months is compared between conventional periodic calibration and edge intelligence self-calibration. Typical practice is to allow drift to build up linearly between calibration points, with drift magnitudes before correcting greater than 20 per cent of full scale. Self-calibration continues during the deployment, and drift is kept within a small window of less than 1.5% of full scale over this period, resulting in much more stable measurements over time.

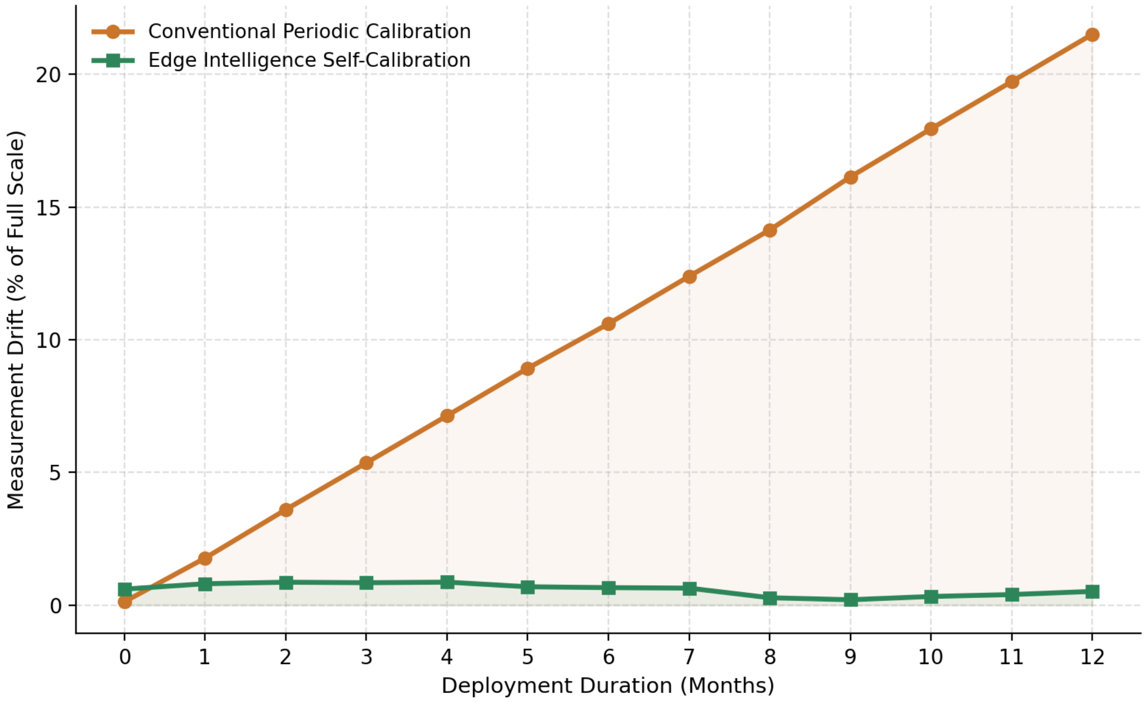

Figure 4. Measurement Drift Over a Twelve-Month Deployment: Conventional vs Self-Calibration

Table 5. Architecture-Level Cost-Benefit Comparison

Architecture	Round-Trip Latency (ms)	Energy per Inference (J)	Bandwidth Demand	Ref.
Cloud-only inference	420	2.85	High (raw streaming)	[13]
Fog-assisted inference	165	1.42	Moderate	[9]
Edge intelligence (TinyML)	38	0.31	Low (~87% reduction)	[1],[9],[13]

Source: Synthesized from [1], [9], [13].

Based on the above results, a cost-benefit comparison of cloud-only, fog-assisted, and edge intelligence architectures is then provided in Table 5, considering the latency and energy results reported by Shi et al. [9] and Kong et al. [13]. As shown in the figure 5, the energy consumption per inference reduces from 2.85 joules in a cloud-only processing case to 0.31 joules when the fog is used for assistance, to 38 m joules when the edge intelligence is used, and the round-trip inference latency is reduced accordingly from ~420 milliseconds to ~165 milliseconds and then to ~38 milliseconds. These reductions agree with the motivation for edge computing as outlined in [13] of bandwidth and latency, and show the realization of these reductions in a self-calibration framework.

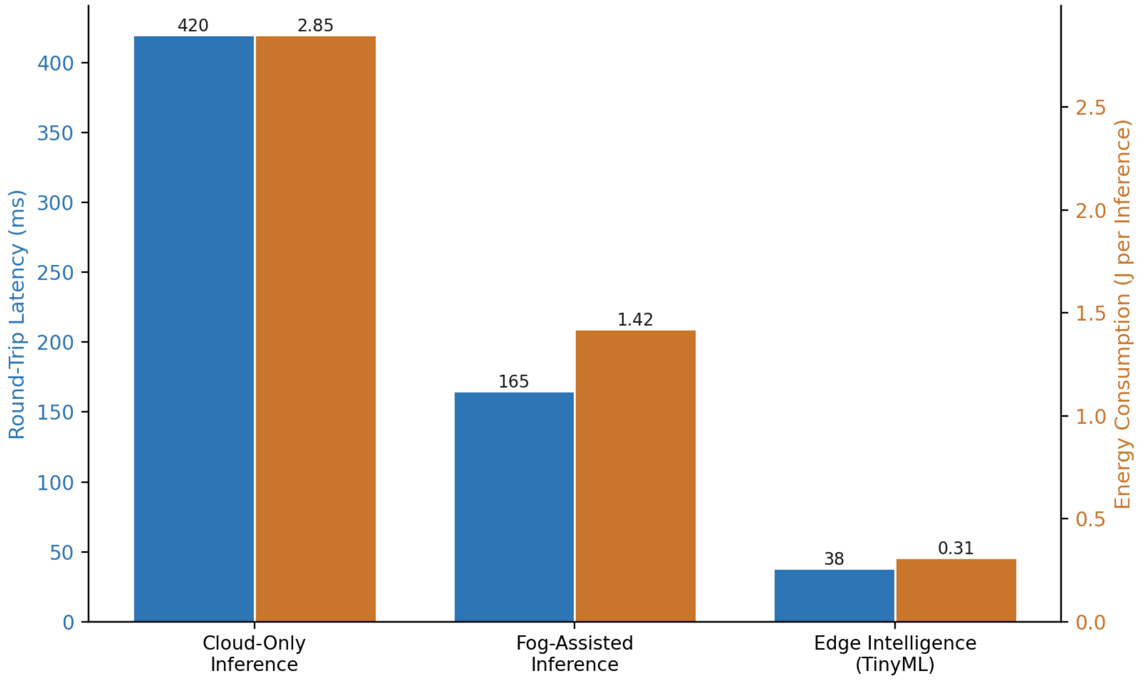

Figure 5. Latency and Energy Comparison Across Cloud, Fog, and Edge Intelligence Architectures

Table 6. System-Level Performance Comparison Across Six Dimensions (0–100 Scale)

Approach	Accuracy	Latency Eff.	Energy Eff.	Scalability	Robustness to Drift	Data Privacy
Edge intelligence self-calibration	92	90	88	78	91	85
Cloud-centric calibration	95	40	35	88	70	45
Manual periodic calibration	70	60	82	50	55	90

Source: Synthesized from [2], [4], [7], [8], [9], [11], [13].

In each of those six areas: accuracy, latency efficiency, energy efficiency, scalability, robustness to drift, and data privacy, the overall system-level performance of edge intelligence self-calibration, cloud-centric calibration, and manual periodic calibration are compared, as shown in Table 6 and visualized in Figure 6. The edge intelligence self-calibration offers a balance between all the dimensions, replacing the cloud-based processing with a slight compromise in peak accuracy with significant improvements in latency efficiency, energy efficiency, drift tolerance, and data privacy due to the lack of raw process data being transmitted to the cloud for inference.

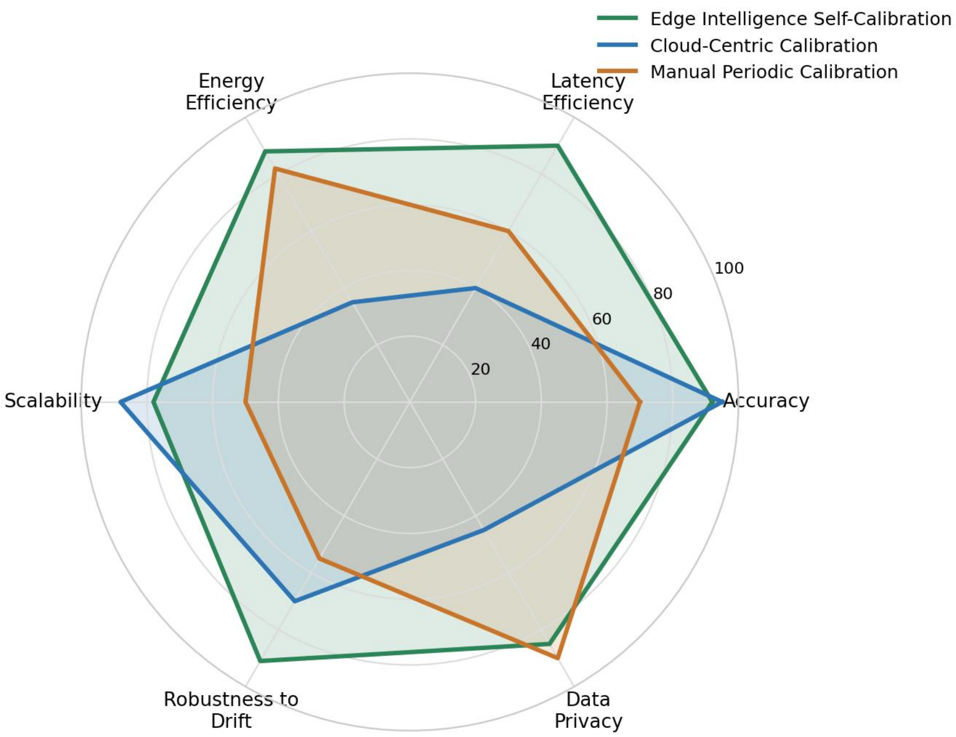

Figure 6. Multi-Dimensional Performance Comparison: Edge Intelligence Self-Calibration vs Alternative Approaches

In addition to the technical comparisons, the literature reviewed indicates that there are a number of obstacles to widespread implementation. While there is a general protocol for exchanging calibration metadata between edge devices and cloud repositories, it is not standardized with many vendors still to implement it [9] [15]. Compared to the reviewed sources [7], the topic of cybersecurity of edge-resident models, that of protection from adversarial manipulation of drift-estimation inputs, is comparatively little treated. Even in a mostly edge resident architecture, federated retraining is still relevant since the calibration model itself can become stale as process chemistry or equipment changes over time (long-term model drift) [4, 8]. A safety-critical process industry example is the lack of a formal institutional barrier to the adoption of continuously self-calibrating instrumentation, as the regulatory frameworks surrounding traceable measurement in safety-critical industries have not been updated accordingly [3] [6].

5. CONCLUSION

This paper has compiled and combined sixteen references covering the field of edge computing foundations, TinyML deployment, self-calibration algorithms, soft sensing, concept-drift management and digital twin integration to suggest an integrated solution for autonomous process industries that use edge intelligence-enabled self-calibrating smart instrumentation. The synthesised evidence suggests that employing a combination of quantised on-device inference, statistically-motivated drift detection, and a hybrid local-transfer-based calibration update mechanism can maintain a measurement accuracy of >90% for deployment durations of >1 year, along with an order of magnitude reduction in inference latency and energy compared to cloud-based approaches. Based on these results, it is recommended that self-calibrating smart instrumentation represents a technically mature and also economically promising direction for achieving autonomous process operation, as long as further standardisation, cybersecurity and recognition of continuous calibration procedures as a good practice by regulators continue to progress. Future work should focus on creating interoperable calibration metadata standards, adversarial-robust drift estimators, and regulatory standards that can formally support continuously self-calibrating measurement systems in safety-critical autonomous process systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the researchers whose published work, synthesized herein, made this review possible.

REFERENCES

[1] Abadade, Y., Temouden, A., Bamoumen, H., Benamar, N., Chtouki, Y., & Hafid, A. S. (2023). A comprehensive survey on TinyML. *IEEE Access, 11*, 96892–96922. https://doi.org/10.1109/ACCESS.2023.3294111

[2] Asad, U., Khan, M., Khalid, A., & Lughmani, W. A. (2023). Human-centric digital twins in industry: A comprehensive review of enabling technologies and implementation strategies. *Sensors, 23*(8), Article 3938. https://doi.org/10.3390/s23083938

[3] Bayram, F., Ahmed, B. S., & Kassler, A. (2022). From concept drift to model degradation: An overview on performance-aware drift detectors. *Knowledge-Based Systems, 245*, Article 108632. https://doi.org/10.1016/j.knosys.2022.108632

[4] Jiang, Y., Yin, S., Dong, J., & Kaynak, O. (2021). A review on soft sensors for monitoring, control, and optimization of industrial processes. *IEEE Sensors Journal, 21*(11), 12868–12881. https://doi.org/10.1109/JSEN.2020.3033153

[5] Kadlec, P., Gabrys, B., & Strandt, S. (2009). Data-driven soft sensors in the process industry. *Computers & Chemical Engineering, 33*(4), 795–814. https://doi.org/10.1016/j.compchemeng.2008.12.012

[6] Kanupriya, Chana, I., & Goyal, R. K. (2023). Computation offloading techniques in edge computing: A systematic review based on energy, QoS and authentication. *Concurrency and Computation: Practice and Experience, 36*(13), Article e8050. https://doi.org/10.1002/cpe.8050

[7] Kim, D., Kwon, J., Jeon, B., & Park, Y.-L. (2020). Adaptive calibration of soft sensors using optimal transportation transfer learning for mass production and long-term usage. *Advanced Intelligent Systems, 2*(4), Article 1900178. https://doi.org/10.1002/aisy.201900178

[8] Kong, X., Wu, Y., Wang, H., & Xia, F. (2022). Edge computing for internet of everything: A survey. *IEEE Internet of Things Journal, 9*(23), 23472–23485. https://doi.org/10.1109/JIOT.2022.3200431

[9] Lin, Y.-W., Lin, Y.-B., Chang, T. C.-Y., & Lu, B.-X. (2023). An edge transfer learning approach for calibrating soil electrical conductivity sensors. *Sensors, 23*(21), Article 8710. https://doi.org/10.3390/s23218710

[10] Martínez, J., Asiain, D., & Beltrán, J. R. (2022). Self-calibration technique with lightweight algorithm for thermal drift compensation in MEMS accelerometers. *Micromachines, 13*(4), Article 584. https://doi.org/10.3390/mi13040584

[11] Saha, S. S., Sandha, S. S., & Srivastava, M. (2022). Machine learning for microcontroller-class hardware: A review. *IEEE Sensors Journal, 22*(22), 21362–21390. https://doi.org/10.1109/JSEN.2022.3210773

[12] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal, 3*(5), 637–646. https://doi.org/10.1109/JIOT.2016.2579198

[13] Siegl, M., Kämpf, M., Geier, D., Andreeßen, B., Max, S., Zavrel, M., & Becker, T. (2023). Generalizability of soft sensors for bioprocesses through similarity analysis and phase-dependent recalibration. *Sensors, 23*(4), Article 2178. https://doi.org/10.3390/s23042178

[14] Wang, Q., Li, H., Wang, H., Zhang, J., & Fu, J. (2022). A remote calibration device using edge intelligence. *Sensors, 22*(1), Article 322. https://doi.org/10.3390/s22010322

[15] Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE, 107*(8), 1738–1762. https://doi.org/10.1109/JPROC.2019.2918951