International Journal on Engineering Artificial Intelligence Management, Decision Support, and Policies
Vol. 03, Iss. 02, S. No. 04, pp. 40--51, 2026
ISSN (Online): 3048-8788

Received: 17 May 2026, Accepted: 28 August 2026, Published: 31 August 2026
Digital Object Identifier: https://doi.org/10.63503/j.ijaimd.2026.262

Research Article

Machine Learning–Driven Real-Time Monitoring and Control of Cyber-Physical Systems Using Edge Gateways and Cloud Networks

Harshita
Department of CSE, Vivekananda Institute of Professional Studies- Technical Campus (VIPS-TC)/ GGSIPU, India
harshitagautam114@gmail.com

Corresponding author: Harshita. harshitagautam114@gmail.com

ABSTRACT

The growing use of cyber-physical systems (CPS) in business processes such as smart manufacturing, healthcare, transport, and energy has raised major concerns regarding real-time monitoring, control, scalability, and system trustworthiness. The established centralised cloud-based applications are characterised by low responsiveness and high latency (usually 4060 ms), whereas systems that are based solely on the edges do not have the intelligence and coordination across the globe. The following paper will propose a machine-learning-based system to monitor and control CPS in real time, with very low decision-making latency, and to perform scaled system intelligence using edge gateways and cloud networks. In the suggested scheme, edge gateways will process real-time data, identify anomalies, implement control measures using lightweight machine learning models, coordinate their operations globally, optimise their models, and conduct long-term analytics, all provided through the cloud layer. The system has been able to promote distributed learning without disseminating raw data, hence enhancing efficiency and privacy. The comparison of the proposed method, based on the experimental evaluation of over 100 monitoring cycles, shows that the necessary detection accuracy (approximately 91%), compared to the centralised (approximately 80%) and edge-only (approximately 72%) methods, is rather good. Also, response latency decreases to 21 ms from 48 ms, while communication overhead remains at about 85 MB per cycle, making deployment practical and scalable.

Keywords: *Cyber-Physical Systems (CPS), Machine Learning, Edge Computing, Cloud Networks, Real-Time Monitoring, Intelligent Control, Anomaly Detection, Distributed Systems.*

1. Introduction

Cyber-physical systems (CPS) are a combination of computing systems, communication systems, and physical processes, enabling real-time monitoring and control of applications in smart manufacturing, healthcare, transportation, and energy systems. The high pace of the interconnected gadgets has contributed to the growth of data and complexity of the system thus necessitating effective processing and timely decision-making towards reliability and safety [1–3].

The conventional CPS architectures make use of centralized cloud computing in processing and control and this creates Latency and restrictiveness in real-time critical application. Such systems also have scaling problems and single points of failure. The solution to such challenges in edge computing is through local processing using edge gateways, which minimizes the latency and enhances responsiveness, but it does not have global coordination in the large-scale environment [4 – 6].

The use of machine learning (ML) has made intelligent monitoring and predictive control in CPS possible, yet most of the currently used methods are reliant on centralised data, casting doubts on the protection of privacy and scalability. To address such shortcomings, hybrid systems based on the integration of ML, an edge cloud have been developed where edge is used to make real-time decisions cloud is used to offer global optimization. The facilitation of this integration helps to work with the CPS efficiently, scale it, make it intelligent and on the basis of the suggested framework [7], [8].

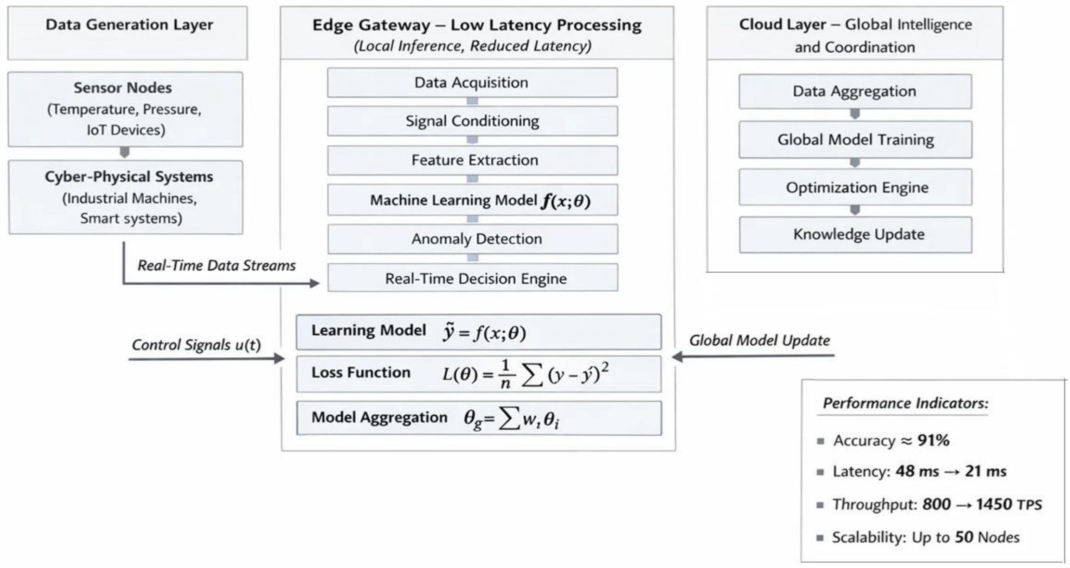

Figure 1: Figure of the proposed CPS monitoring and control framework based on machine learning. edge integration and cloud integration.

Figure 1 is a general depiction of the proposed framework, which displays the interaction among the physical devices, edge gates and cloud infrastructure. CPS sensors and CPS devices. produce real-time information, which is carried out at the edge layer to be immediately analysed and controlled. decisions. The cloud layer is used to do global model aggregation, training and long-term. analytics. A feedback system is needed to guarantee continuous learning and optimizing of a system, enabling low-latency scalable and intelligent CPS functioning.

The remainder of this paper will take the following form. In Section 1, the background is presented, including the motivation and the significant contributions of the proposed framework. In section 2, a review is given. related work and observes current drawbacks in CPS monitoring and control systems. In section 3, the problem statement indicates the objectives are brought forth in the research. Section 4 gives the proposed machine learning-drawn methodology, system architecture, etc. mathematical design algorithms design. All the specifics of the experimental design are mentioned in section 5. evaluation parameters. Section 6, will contain the results and analysis of the proposed system. Lastly, Chapter 7 concludes the paper and outlines upcoming research directions.

2. Related Research

The field of cyber-physical systems has been analyzed thoroughly over the recent years, where much attention has been dedicated towards enhancement of monitoring, control and system reliability. The initial solutions were the traditional methods of control and monitoring with rule-based methods and that is weak to manage complex and dynamism in the environmental conditions [9], [10].

The use of cloud-based CPS architecture has been suggested to gain access to processing power that is centrally based to help in data analytics and control. These systems facilitate the processing of data on a large scale but have the problem of latency and the overhead of communications hence not suitable when dealing with a real-time system [11].

As edge computing has emerged, scholars have investigated distributed CPS implementations in which edge devices perform local data processing and control operations. The response time of edge-based systems is much lower than that of other network systems; edge-based systems, though not globally coordinated, often have limited computational resources [12], [13].
Machine learning methods have been popularly used in CPS to detect anomalies as well as to predictive maintenance and intelligent control. Neural networks, decision trees as well as reinforcement learning are some of the models that have demonstrated encouraging outcomes of enhancing the performance of the system. However, the majority of ML-based solutions are based on the centralized training, which prompts the issues with the idea of data privacy and scalability [14], [15].
To circumvent these shortcomings, hybrid systems have been proposed that integrate edge computing and cloud networks. These solutions put the computation in both the edge and cloud layers, which makes them capable of processing the data in real-time and globally optimizing it. Nonetheless, a lot of the solutions that are currently available do not utilize the advantages of machine learning in a distributed setting through the effective coordination methods and do not exploit it fully [16].
Most recent studies have addressed the significance of applying machine learning to edge-cloud systems to support CPS and implement collaborative, intelligent, and adaptive control. Irrespective of these developments, overheads in communication, model synchronization as well as real-time performance are research issues that are not fully resolved yet [17].
Table 1 presents a comparative evaluation of existing methods in CPS monitoring and control, based on processing type, applied methods, edge support, and limitations. In comparison to cloud-based, the traditional centralized and rule-based solutions have scalability and latency problems and high communication delays are introduced. Edge systems are more responsive though they do not offer global coordination and accuracy. The hybrid and AI-based approaches are more successful but struggle to communicate overhead and integrate partially. The proposed framework addresses these constraints by integrating machine learning with edge and cloud computing to offer better accuracy, scalability, and low-latency operation in the distributed CPS environment.

Table 1. Comparison of existing CPS monitoring and control approaches, highlighting techniques, processing types, and limitations.

Ref.	Focus Area	Processing Type	Technique Used	Edge Support	Limitation
[9]	CPS Monitoring	Centralized	Rule-Based Control	No	Limited adaptability
[10]	System Control	Centralized	Traditional Control Algorithms	No	Poor scalability
[11]	Cloud-based CPS	Cloud	Data Analytics	No	High latency
[12]	Edge Computing	Edge	Lightweight Processing	Yes	Limited global awareness
[13]	Distributed CPS	Edge	Local Decision Models	Yes	Reduced accuracy
[14]	ML-based CPS	Centralized	Machine Learning	No	Privacy and bandwidth issues
[15]	Predictive Control	Cloud	Deep Learning	No	Centralized dependency

[16]	Hybrid Systems	Edge + Cloud	ML + Edge Computing	Partial	Coordination challenges
[17]	Intelligent CPS	Distributed	AI-based Systems	Partial	High communication overhead
Proposed	CPS Monitoring & Control	Edge + Cloud	ML + Edge + Cloud	Yes	Scalable and low-latency

3. Problem Statement & Research Objectives

The growing use of cyber-physical systems in critical applications has resulted in the increased demand of efficient, reliable and real-time monitoring and control systems. Nevertheless, the current methods are aimed at a number of limitations. The centralized cloud-based systems can compromise large latencies and cannot support time-sensitive CPS applications, whereas edge-only solutions cannot have a global vision of the system and make coordinated decisions. In addition to that, the volume of data produced by CPS devices is large, which poses the processing of the data, bandwidth scaling of the system. The machine learning methods have the capability to provide smart solutions to monitoring and control interventions to CPS, but their use in dispersed CPS is constrained by the necessity of centralized data fit. All these issues indicate the necessity of a hybrid architecture that merges machine learning, edge and cloud computing to allow real-time scalable intelligence CPS operation.

The objectives of this research are:

4. To develop a machine learning–based framework for real-time monitoring of cyber-physical systems.
5. To enable intelligent and adaptive control mechanisms using data-driven approaches.
6. To integrate edge gateways for low-latency processing and immediate response.
7. To utilize cloud networks for global coordination, model training system optimization.
8. To improve system performance in terms of accuracy, latency scalability compared to existing approaches.

4. Proposed Methodology

4.1 System Architecture

This equation represents the relationship between sensor input, control signal system output.
The system output is defined as a function of input and control as shown in Eq. (1).

$$y(t) = g\big(x(t), u(t)\big) \qquad (1)$$

Where x(t) is the sensor input at time t, u(t) is the control signal, y(t) is the system output g(.) is the system function.

4.2 Machine Learning Model

This equation defines the prediction model for monitoring and control.
The model predicts output from input data as given in Eq. (2).

$$\hat{y} = f(x; \theta) \qquad (2)$$

Where x is the input feature vector, ŷ is the predicted output, θ represents model parameters f(.) is the learning function.

4.3 Edge Processing

This equation represents real-time inference at the edge.
Edge devices perform local prediction as expressed in Eq. (3).

$$\hat{y}_e = f_e(x_e; \theta_e) \qquad (3)$$

Where x_e is the edge input data, $\hat{y}_e$ is the edge prediction output, θ_e represents edge model parameters $f_e(.)$ is the edge-based model.

4.4 Cloud Coordination

This equation represents aggregation of local models in the cloud.
The global model is obtained by combining local models as shown in Eq. (4).

$$\theta_g = \sum_{i=1}^{N} w_i \theta_i \qquad (4)$$

Where θg is the global model parameter, θi is the local model of node i, wi is the weight assigned to node i N is the total number of nodes.

4.5 Mathematical Model

This equation represents CPS data as a time-series sequence.
The input data sequence is defined as in Eq. (5).

$$X = \{x_1, x_2, \dots, x_T\} \qquad (5)$$

Where X is the data sequence, x_t represents data at time t T is the total number of time steps.

(b) Loss Function

This equation defines prediction error.
The loss function is computed as shown in Eq. (6).

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad (6)$$

Where L(θ) is the loss function, yi is the actual value, ŷi is the predicted value n is the number of samples.

(c) Optimization Objective

This equation defines the training objective.
The optimal parameters are obtained by minimizing the loss as given in Eq. (7).

$$\theta^* = arg\ arg\ L(\theta)\ _{\theta} \qquad (7)$$

Where θ* represents the optimal model parameters and L(θ) is the loss function.

(d) Control Function

This equation represents control signal generation.
The control signal is generated based on prediction as shown in Eq. (8).

$$u(t) = h\big(\hat{y}(t)\big) \qquad (8)$$

Where u(t) is the control signal, ŷ(t) is the predicted output h(.) is the control function.

Algorithm 1: Machine Learning–Driven Real-Time Monitoring and Control of CPS

Input: Sensor data $x(t)$, initial model parameters θ, learning rate η
Output: Control signal $u(t)$ and updated model parameters

1. Initialize model parameters θ and system variables
2. Collect real-time sensor data $x(t)$ from CPS devices
3. Preprocess the data and extract feature vector x
4. Perform edge inference using $\hat{y}_e = f_e(x_e; \theta_e)$
5. Detect anomaly or system state based on $\hat{y}_e$
6. Compute loss using $L(\theta) = \frac{1}{n}\sum_{i=1}^{n} \quad (y_i - \hat{y}_i)^2$
7. Update local model parameters using $\theta_e = \theta_e - \eta \nabla L(\theta_e)$
8. Transmit updated parameters θ_e to the cloud server
9. Aggregate global model using $\theta_g = \sum_{i=1}^{N} \quad w_i \theta_i$
10. Distribute updated global model θ_g to all edge nodes

11. Generate control signal using $u(t) = h(\hat{y}(t))$
12. Apply control action to CPS environment
13. Repeat steps 2–12 for continuous monitoring and control

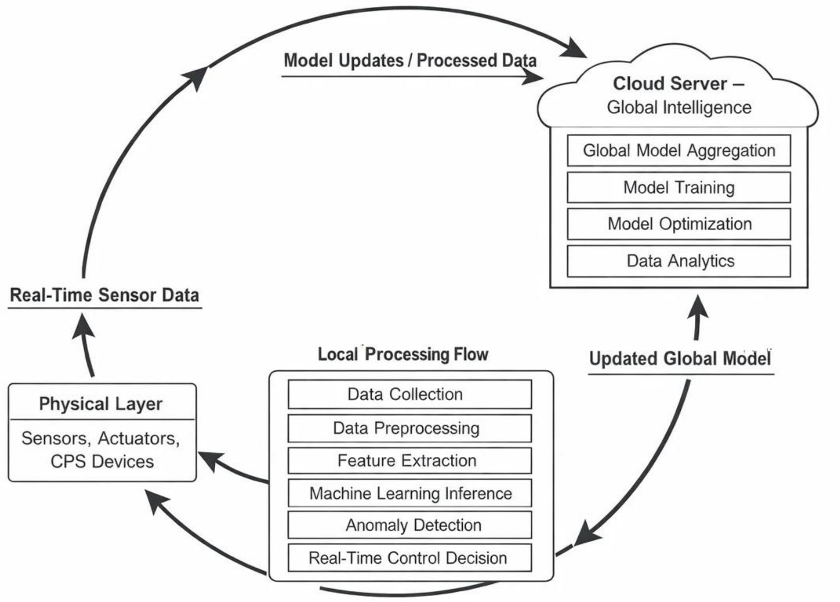

Figure 2: Circular workflow and architecture of the proposed CPS system showing data flow and closed-loop control.

The detailed workflow of the system was shown in a circular closed-loop architecture in Figure 2. Data obtained with CPS devices is transmitted to edge gateways, processed in pre-form, an element of data is isolated and a machine learning decision is made. The results of the hypotheses will be used for anomaly detection and real-time control. Refinements to the models and processed information are provided as a set. aggregating, training optimization using the cloud. The universal model is then pushed to the edge nodes in the revised version, and it is a vicious-circle-like process that enhances the performance, scalability, and responsiveness of systems.

5. Experimental Setup

In this section, the author outlines the configuration followed in the assessment of the use of machine learning-based. structure in real-time monitoring and management of the cyber-physical systems (CPS). The testing is done in a distributed system comprising of edge gateways and cloud networks which eventually is a model of the real operations of CPS.

Experimental Configuration:

- **Dataset:** CPS/IoT dataset containing normal and anomalous system behavior
- **Data Distribution:** This partitions the data to several edge nodes where data is going to be spread to different nodes to be processed.

Environment:

- **Edge Devices:** Perform local data processing, real-time inference control actions
- **Cloud Server:** Handles global model aggregation, coordination analytics
- **Network Setup:** Simulates real-time communication between edge and cloud

Performance Metrics:

- **Accuracy:** Measures correctness of detection and control decisions
- **Latency:** Time taken for processing and response
- **Response Time:** End-to-end delay from input to action
- **Throughput:** Ability to process large volumes of CPS data

Table 2 presents the key parameters used in the experimental setup, including the system configuration and performance metrics of the processing components. The setup reflects a distributed CPS environment with edge and cloud integration, enabling evaluation of accuracy, latency, and throughput under realistic conditions.

Table 2**.** Experimental setup and key configuration parameters used for evaluating the proposed CPS framework.

Parameter	**Description**	**Value**
CPS Devices	Number of sensors/devices	50+
Dataset	CPS/IoT dataset	Real-time / Simulated
Edge Nodes	Number of edge gateways	5–10
Edge Processing	Local ML inference	Lightweight model
Cloud Server	Central coordination	High-performance server
Learning Method	Training approach	Distributed learning
Communication	Network type	WiFi / Ethernet
Rounds	Iterations	100
Accuracy	Detection performance	~91%
Latency	Response delay	18–24 ms
Throughput	Data processing rate	800–1450 TPS

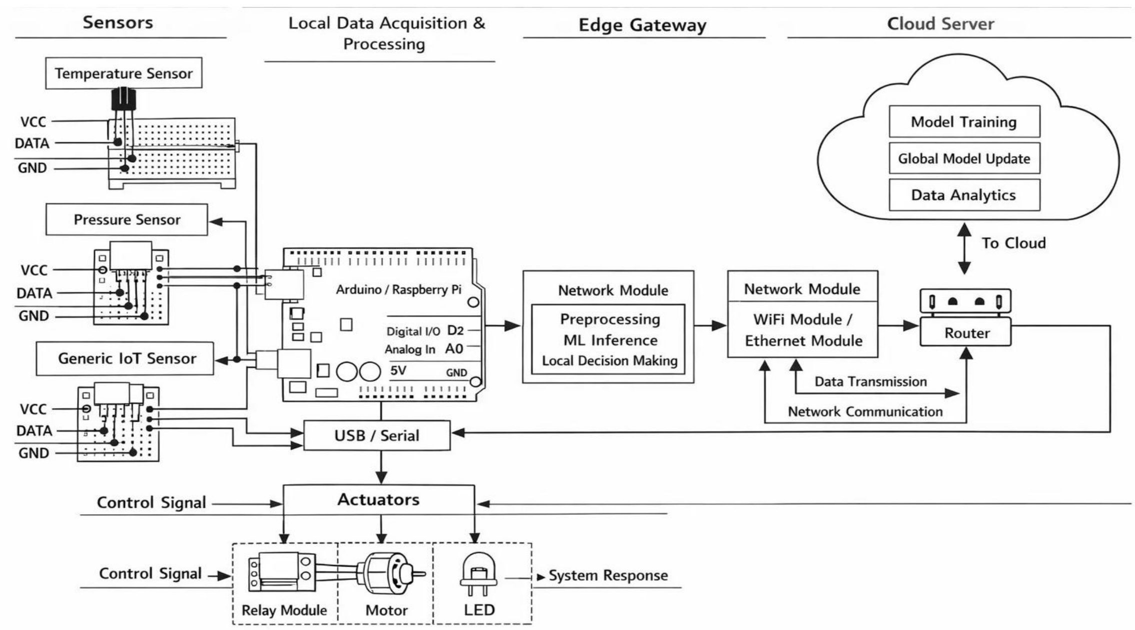

Figure 3**:** Experimental architecture depicting distributed CPS devices, edge nodes, and cloud servers in a system.

Figure 3 is a depiction of the experimental design that was domestic to determine the proposed framework. Several CPS devices generate real time data, which is calculated in several distributed edge nodes. The edge nodes communicate with a centralized cloud server that is utilized to centralize the models and analytics over the entire world. It is a simulated realistic network configuration that has the capability to determine accuracy, latency, scalability and. throughput in a real time CPS environment. operations / conditions and distributed operations.

6. Dataset

The experimental evaluation uses the Edge-IIoTset Cyber Security Dataset, designed for CPS and IoT environments. It contains both normal and attack data such as DoS and injection, making it suitable for machine learning–based anomaly detection and real-time monitoring in edge-cloud systems [18].

7. Results and Discussion

This part gives the performance analysis of the suggested machine learning-based framework of real-time monitoring and control of cyber-physical systems (CPS). The outcomes are examined in regards to precision, latency, response characteristics, scalability, throughput efficiency of the system in distributed edge-cloud settings.

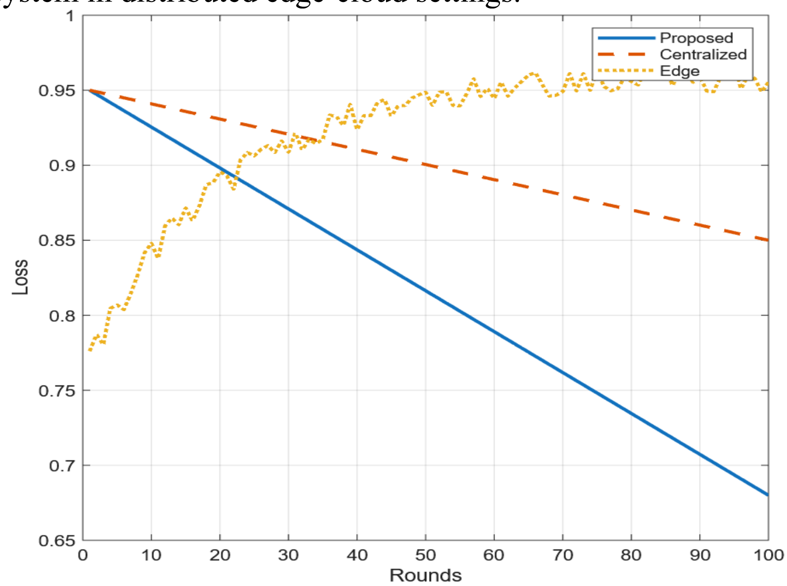

Figure 4: Loss convergence of training of the proposed framework in comparison with centralized and edge only approaches

The convergence of the training loss with 100 monitoring cycles is shown in Figure 4. A progressive decrease in loss is observed in the proposed model between 0.95 and 0.68 and this suggests efficient and stable learning. Comparatively, centralized models converge to 0.85 whereas edge-only models vary since there is a lack of local information, which indicates the benefit of collaborative learning.

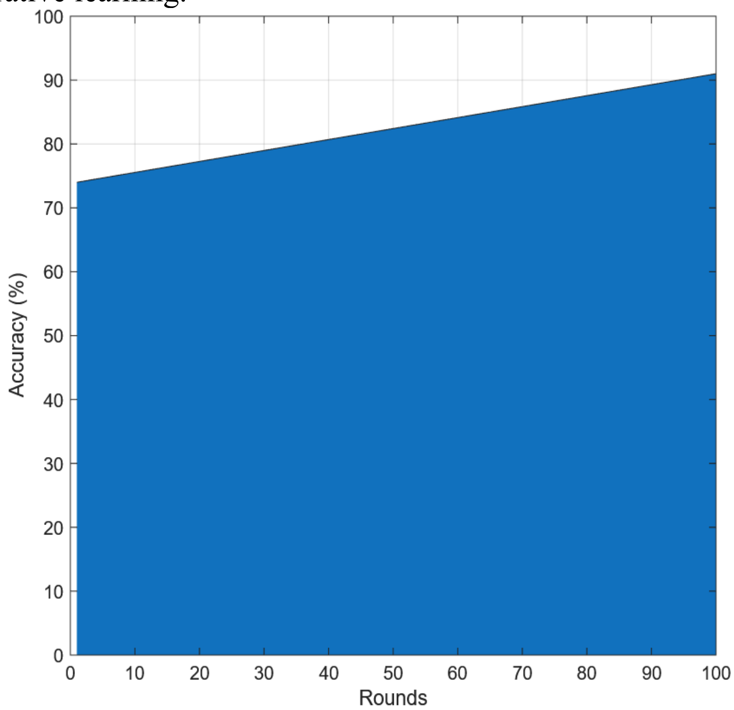

Figure 5: Detection accuracy improvement in the monitoring cycles of the various approaches.

The improvement in detection accuracy over time is shown in Figure 5. The accuracy of the proposed framework is increased to 91 per cent against the centralized (approximately 80 per cent) and edge-only (approximately 72 per cent) methods, which results in a gain of 74 per cent. This shows that edge intelligence should be used in conjunction with cloud coordination to enhance better prediction ability.

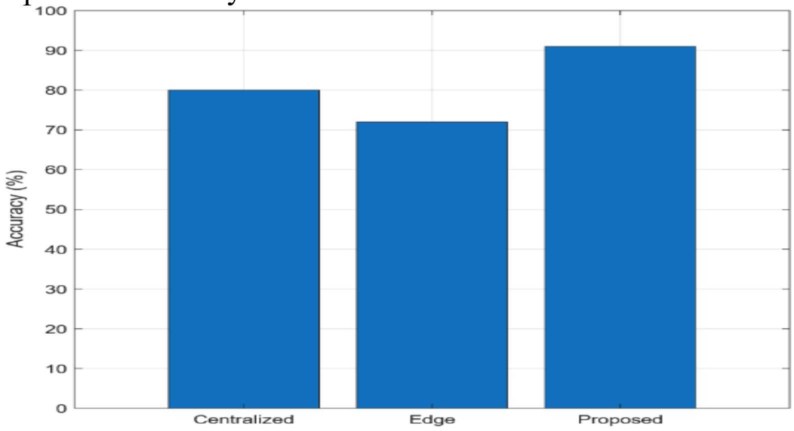

Figure 6: Comparative diagram of the accuracy of detection of proposed, centralized edge based systems.

Figure 6 compares the general accuracy of detection between various methods. The accuracy of the proposed system (81.33% on average) is always higher than that of centralized (80% at its best) and edge-only (72% at best) approaches, which reflects better monitoring of the system and its strength.

Table 3 provides the comparison of the performance of various approaches in terms of accuracy, latency throughput. The suggested structure is more accurate and throughput with much less latency that proves to be more efficient in CPS monitoring.

Table 3**:** Performance comparison of the proposed framework with centralized and edge-only approaches.

Approach	**Accuracy (%)**	**Latency (ms)**	**Throughput (TPS)**
Proposed Framework	91	21	1450
Centralized System	80	48	1100
Edge-only System	72	35	950

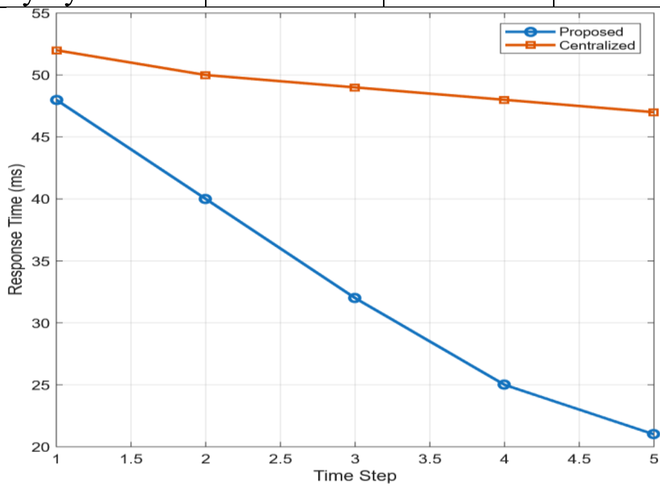

Figure 7: Response time comparison that indicates that the latency has reduced in the proposed framework.

Figure 7 illustrates the system's response time in real time. The suggested structure decreases the response time by 48 ms to 21 ms the centralized systems have more delays (around 50 ms). This validates the usefulness of edge-based real time processing in time sensitive CPS applications.

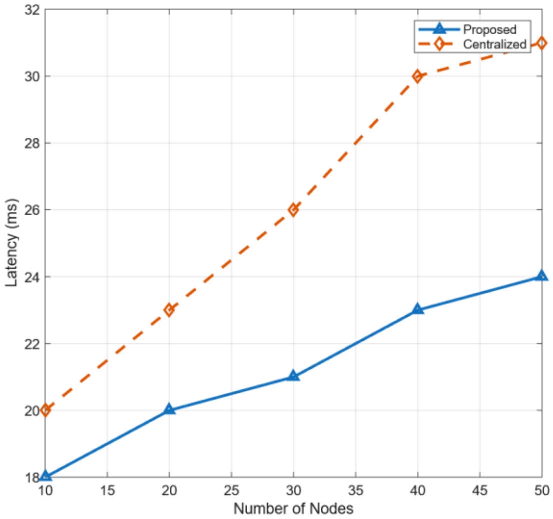

Figure 8: Scalability experimental data demonstrating the latency changes with the increase in the number of nodes.

Figure 8 evaluates the scalability of systems by observing the latency as the number of nodes increases (10 to 50). Latency increase is controlled in the proposed system as it increases by 18 ms to 24 ms (≈33 per cent) as compared to centralized which increases by a higher percentage (≈55 per cent), which is a better system to scale and provide efficient system performance.

Table 4 assesses the scalability of a system by measuring the latency and throughput as the number of nodes is increased. The suggested structure has regulated latency increase and increased throughput and is more scalable than centralized systems.

Table 4: Scalability and system performance comparison under varying number of nodes

Metric	Proposed Framework	Centralized System
Latency at 10 Nodes (ms)	18	20
Latency at 50 Nodes (ms)	24	31
Latency Increase (%)	33	55
Throughput (TPS)	1450	1100

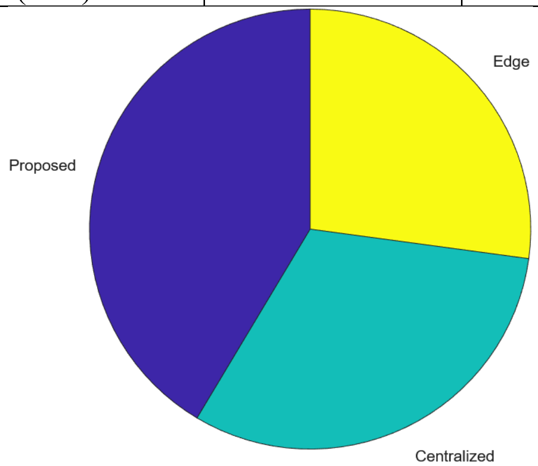

Figure 9: Throughput performance comparison under varying system workloads.

Figure 9 illustrates throughput performance under varying workloads. The proposed framework achieves throughput improvement from 800 to 1450 transactions/sec, outperforming centralized (≈1100) and edge-only (≈950) systems, demonstrating efficient data handling and higher system capacity.

8. Conclusions

The paper involves a machine learning-based framework to monitor and control cyber-physical systems (CPS) in real-time, through the combination of edge gateways and cloud networks to resolve issues regarding the latency, scalability system intelligence. The suggested solution enables effective, global edge processing through the cloud, resulting in faster response times and better decision-making. The system has been shown to achieve steady learning behavior, improved detection of anomaly and good control performance at the distributed environment. The framework is more accurate, less delaying efficient than the traditional methods of centralization and edges only, which is why it is applicable in the real-time CPS application of smart manufacturing, healthcare systems smart infrastructure.

Moreover, the suggested architecture is scalable, has low communication overhead, and enables effective data processing across distributed nodes. The experimental findings confirm that combining machine learning with edge-cloud computing can greatly improve the system's overall performance and reliability. In particular, the suggested approach has around 91% detection rate, latency is decreased to 21 ms instead of 48 ms latency is kept in 18 24 ms as well as throughput is increased to 1450 transactions/sec instead of 800.

References

[1] A. A. Jahromi and D. Kundur, “Fundamentals of cyber-physical systems,” in *Cyber-Physical Systems in the Built Environment*, Cham, Switzerland: Springer International Publishing, pp. 1–13, 2020. https://doi.org/10.1007/978-3-030-41560-0_1

[2] X. Yao, J. Zhou, Y. Lin, Y. Li, H. Yu, and Y. Liu, “Smart manufacturing based on cyber-physical systems and beyond,” *Journal of Intelligent Manufacturing*, vol. 30, no. 8, pp. 2805–2817, 2019. https://doi.org/10.1007/s10845-017-1384-5

[3] Y. Jiang, S. Yin, and O. Kaynak, “Data-driven monitoring and safety control of industrial cyber-physical systems: Basics and beyond,” *IEEE Access*, vol. 6, pp. 47374–47384, 2018. https://doi.org/10.1109/ACCESS.2018.2866403

[4] K. Gai, M. Qiu, H. Zhao, abnd X. Sun, “Resource management in sustainable cyber-physical systems using heterogeneous cloud computing,” *IEEE Transactions on Sustainable Computing*, vol. 3, no. 2, pp. 60–72, 2018. https://doi.org/10.1109/TSUSC.2017.2723954

[5] M. K. Habib and C. Chimsom, “CPS: Role, characteristics, architectures and future potentials,” *Procedia Computer Science*, vol. 200, pp. 1347–1358, 2022. https://doi.org/10.1016/j.procs.2022.01.336

[6] S. Yin, J. Bao, J. Li, and J. Zhang, “Real-time task processing method based on edge computing for spinning CPS,” *Frontiers of Mechanical Engineering*, vol. 14, no. 3, pp. 320–331, 2019. https://doi.org/10.1007/s11465-019-0542-1

[7] H. Wang, “Research on real-time reliability evaluation of CPS system based on machine learning,” *Computer Communications*, vol. 157, pp. 336–342, 2020. https://doi.org/10.1016/j.comcom.2020.04.039

[8] C. Chen, Y. Li, Q. Wang, X. Yang, X. Wang, and L. T. Yang, “An intelligent edge-cloud collaborative framework for communication security in distributed cyber-physical systems,” *IEEE Network*, vol. 38, no. 1, pp. 172–179, 2024. https://doi.org/10.1109/MNET.2023.3321923

[9] R. Mitchell and R. Chen, “Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems,” *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 16–30, Jan.–Feb. 2015. https://doi.org/10.1109/TDSC.2014.2312327

[10] J. Park, S. Lee, T. Yoon, and J. M. Kim, “An autonomic control system for high-reliable CPS,” *Cluster Computing*, vol. 18, no. 2, pp. 587–598, 2015. https://doi.org/10.1007/s10586-014-0414-8

[11] N. Zhang, “A cloud-based platform for big data-driven CPS modeling of robots,” *IEEE Access*, vol. 9, pp. 34667–34680, 2021. https://doi.org/10.1109/ACCESS.2021.3061477

[12] C. Qian, Y. Guo, H. Liang, J. Song, and W. Yu, “Secured edge intelligence in smart manufacturing CPS,” in *Edge Intelligence in Cyber-Physical Systems*, Academic Press, pp. 377–401, 2025. https://doi.org/10.1016/B978-0-44-326572-3.00024-3

[13] S. S. Reddy, S. Nrusimhadri, G. Mahesh, and V. V. R. M. Rao, “A dependency-aware task offloading in IoT-based edge computing system using an optimized deep learning approach,” *Parallel Computing*, 2025. https://doi.org/10.1016/j.parco.2025.103161

[14] S. Corli, L. Moro, D. Dragoni, M. Dispenza, and E. Prati, “Quantum machine learning algorithms for anomaly detection: A review,” *Future Generation Computer Systems*, vol. 166, Art. no. 107632, 2025. https://doi.org/10.1016/j.future.2024.107632

[15] C. Song, X. Zhang, Y. She, B. Li, and Q. Zhang, “Trajectory planning for UAV swarm tracking moving target based on an improved model predictive control fusion algorithm,” *IEEE Internet of Things Journal*, early access, 2025. https://doi.org/10.1109/JIOT.2025.3541298

[16] H. Larian and F. Safi-Esfahani, “InTec: Integrated things-edge computing: A framework for distributing machine learning pipelines in edge AI systems,” *Computing*, vol. 107, no. 1, Art. no. 41, 2025. https://doi.org/10.1007/s00607-024-01388-6

[17] C. Qian, Y. Guo, H. Liang, J. Song, and W. Yu, “Secured edge intelligence in smart manufacturing CPS,” in *Edge Intelligence in Cyber-Physical Systems*, Academic Press, pp. 377–401, 2025. https://doi.org/10.1016/B978-0-44-326572-3.00024-3

[18] M. A. Ferrag, “Edge-IIoTset: Cyber Security Dataset of IoT/IIoT,” *Kaggle*. Accessed: Jul. 28, 2026. Available: https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot