An Advanced Intelligence Protocol for Context-Aware Edge Computing Devices in Distributed Systems

Vishnu Suryawanshi
School of Computing, MIT Art, Design and Technology University, Pune, India
vishnusam2007@gmail.com (corresponding author)

Nakul Sharma
Department of Computer Science Engineering (IOT CS inc. Blockchain Technology), Vishwakarma Institute of Technology, Pune, India
nakul777@gmail.com

Nandkishor P. Karlekar
School of Computing, MIT Art, Design and Technology University, Pune, India
nandkishor.karlekar@mituniversity.edu.in

Abhijeet Cholke
School of Computing, MIT Art, Design and Technology University, Pune, India
abhijeet.cholke@gmail.com

Vishal Bogam
School of Computing, MIT Art, Design and Technology University, Pune, India
vishal.bogam@gmail.com

Raju Prakash Gurav
School of Computing, MIT Art, Design and Technology University, Pune, India
gurav.m.p@gmail.com

Bharat Devhare
Department of Computer Engineering, Bharti Vidyapeeth College of Engineering, Pune, India
brdevhare@gmail.com

Received: 1 March 2026 | Revised: 25 March 2026, 19 April 2026, and 22 April 2026 | Accepted: 23 April 2026

 DOI: https://doi.org/10.48084/etasr.18467

ABSTRACT

The increasing complexity of integrated computing environments requires sustainable architectures that can adapt, evolve, and remain reliable throughout their lifetime. Current edge computing environments are frequently deficient in dynamic context management, adaptive scheduling, and continuous compliance. This work aims to mitigate these limitations and presents an open-source framework that unifies multidimensional contextual analysis, adaptive task scheduling, lifecycle-conscious security, and continuous compliance within a single closed-loop system. The proposed solution utilizes a hybrid methodology consisting of Bayesian inference for probabilistic context reasoning and Deep Reinforcement Learning (DRL) for adaptive decision-making. Experimental analysis shows that the proposed protocol framework is superior to currently available models, resulting in a 24 % reduction in latency, a 14 % decrease in power consumption, and a 96 % compliance rate. These findings confirm that the proposed approach is scalable and enables intelligent, secure, and compliance-aware edge computing systems.

Keywords-Model Context Protocol (MCP); distributed computing; edge intelligence; Reinforcement Learning (RL); Bayesian inference; IoT security; context modeling

I. INTRODUCTION

Edge devices form an essential component of data analytics at the point of data generation. The large volumes of data generated require timely processing and rapid responses to ensure high quality of service. These devices share the computational burden with cloud servers during data processing. Consequently, architectural enhancements are required to reduce latency and minimize backbone bandwidth utilization. Edge computing devices are widely used in autonomous vehicles, industrial automation, smart surveillance, and healthcare monitoring applications [1-3].

With the rapid growth of Internet of Things (IoT) systems, the number of connected devices and the volume of generated data have increased significantly. This has further emphasized the importance of edge computing in enabling edge systems to operate intelligently and respond in real time, rather than relying solely on centralized cloud processing [3-5].

Despite these advantages, edge environments are inherently distributed, heterogeneous, and highly dynamic. Edge nodes vary significantly in computational capability, power resources, network connectivity, and trust levels, and they operate under rapidly changing workloads and environmental conditions [2, 4, 6].

These characteristics introduce significant challenges in context management, adaptive resource allocation, and secure device control. Traditional centralized cloud-based control and fixed resource provisioning approaches are not well suited for such environments, often resulting in degraded performance and limited flexibility. A major limitation of existing edge computing solutions is their reliance on overly simplified or single-factor context models. In practice, most implementations consider only a limited subset of available information, such as network traffic or device status, while ignoring the complex interactions among device behavior, network dynamics, user activities, application requirements, and environmental conditions [7, 8]. As a result, this incomplete representation of context fails to capture real operational complexity, leading to poor system responses under dynamic and uncertain conditions [8]. Consequently, many edge-based solutions struggle to maintain reliable service levels under fluctuating workloads.

The Model Context Protocol (MCP) addresses these challenges by enabling systems to jointly analyze multiple relevant inputs within a unified control structure. By leveraging multiple context sources, MCP allows edge platforms to dynamically adjust tasks, manage resources more efficiently, and maintain stable operation under changing conditions [6, 7]. When integrated with intelligent control mechanisms, MCP enhances the adaptability and autonomy of edge computing systems. Figure 1 presents a conceptual comparison between the proposed Advanced Intelligence Protocol (AIP) and MCP.

However, despite these advantages, MCP-based systems still face limitations in handling security, trust, and real-time compliance in edge environments [9, 10].

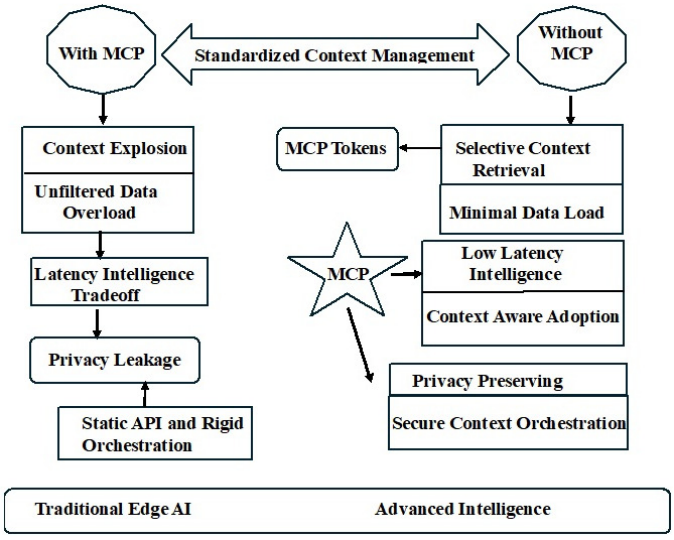

Fig. 1. System model of the MCP-based and proposed protocols.

In existing systems, MCP-based protocols are designed to preserve context across different conditions and scenarios. These systems follow a client–server architecture, where the client first requests a session identifier from the server. This identifier is linked to the current context, enabling the server to retrieve and maintain relevant information. The context is continuously updated as new requests are received under dynamic operating conditions.

However, context awareness alone does not ensure that edge computing systems are secure, reliable, or trustworthy. Although recent advances in edge intelligence and Reinforcement Learning (RL)-based scheduling have improved response time and efficiency, security is still often treated as an auxiliary component rather than an integrated system property [11-13].

In addition, legal requirements and policy enforcement are typically handled outside the runtime system through periodic inspections or offline audits, which are unsuitable for real-time and highly dynamic edge environments. As a result, despite significant progress, current edge intelligence solutions still exhibit three major limitations:

1. Contextual information is often represented in a static or one-dimensional manner.
2. Security mechanisms are decoupled from real-time decision-making processes and system execution stages.
3. Compliance verification is not continuously enforced through automated control mechanisms [9, 14].

Collectively, these limitations reduce the robustness, scalability, and adaptability of edge systems in real-world deployments. To address these challenges, this work proposes a context-aware computing framework based on the AIP. The proposed architecture integrates contextual information from multiple sources, supports adaptive task allocation, enforces

lifecycle-aware security, and enables continuous policy verification within a unified control framework. Experimental results demonstrate that the proposed approach improves performance, reduces energy consumption, enhances adaptability, and strengthens trust management compared to existing methods [11, 13, 15].

The main contributions of this work are summarized as follows:

- A lightweight framework that extends MCP by integrating multiple contextual sources to enable intelligent and secure system operation.
- An integrated Bayesian inference and Deep Reinforcement Learning (DRL)-based model for adaptive and uncertainty-aware task scheduling in edge computing systems.
- A system-wide security preservation and continuous compliance framework designed for dynamic edge computing environments.
- A comprehensive evaluation demonstrating improvements in latency, energy efficiency, adaptability, and compliance management.

Several related studies further highlight different aspects of context-aware and edge intelligence systems.

A context-aware mobile crowd-sensing system was previously proposed in which authors collected contextual information from mobile devices to evaluate sensed data quality and estimate device reliability for client selection [16]. A classification model was used to map real-time context, including hardware information and user activity, to sensed data quality.

In this direction, extensive studies on context modeling and reasoning techniques have highlighted the importance of multi-dimensional context interpretation for intelligent system behavior. However, many earlier approaches rely on static or single-context models, limiting their effectiveness in highly dynamic environments. Recent studies indicate that such simplified models cannot maintain stable performance under fluctuating workloads and varying network conditions [17].

From an application perspective, the IoT further expands these challenges by enabling large-scale Machine-to-Machine (M2M) communication with minimal human intervention. The emergence of edge intelligence represents a significant advancement through the integration of Artificial Intelligence (AI) capabilities directly into edge nodes. This paradigm enables on-site learning, prediction, and control, reducing reliance on remote cloud infrastructure. RL-based techniques have been widely used to optimize service latency and resource allocation, whereas DRL methods support real-time large-scale task offloading. Figure 2 illustrates the traditional context-aware edge computing architecture.

Context information includes factors such as location, activity, and environmental conditions, which influence system behavior and decision-making [18]. Therefore, IoT-based smart systems rely on contextual data collected through sensors. A major challenge lies in effectively modeling and analyzing large-scale heterogeneous sensor data for accurate context adaptation. These processes include context acquisition, modeling, reasoning, and dissemination, collectively forming the context management life cycle.

Another critical dimension in such systems is regulatory compliance, which remains a major challenge in edge and IoT environments. Existing solutions typically rely on offline audits or fixed policy verification mechanisms, which are inadequate for the dynamic and real-time nature of edge environments. Furthermore, current frameworks lack integrated mechanisms to coordinate context reasoning, adaptive task scheduling, security enforcement, and compliance monitoring within a unified operational framework [19].

Existing systems therefore exhibit limited flexibility and insufficient support for integrated regulatory compliance in edge and IoT environments. The proposed methodology addresses these limitations through adaptive context-specific intelligence and lifecycle-aware security mechanisms designed for dynamic environments. These features distinguish the proposed approach from existing solutions.

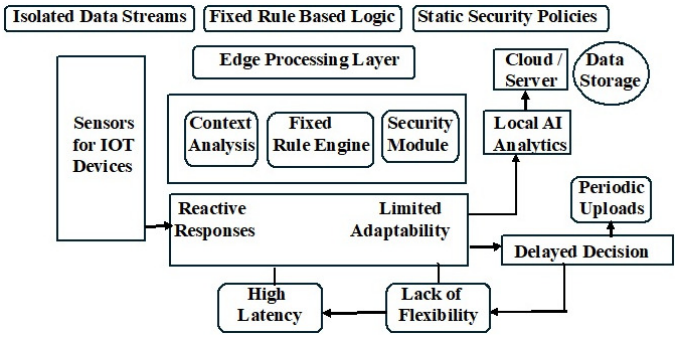

Fig. 2. Traditional context aware edge-architecture.

II. PROPOSED IMPLEMENTATION ARCHITECTURE

The proposed protocol, AIP, uses an MCP-enabled context-aware edge-computation architecture and is designed to deliver adaptive intelligence, lifecycle-centric security, and seamless compliance management within highly dynamic and mixed-edge environments. The architecture adopts a modular, layered design that enables scalability, interoperability, and resilience while supporting real-time decision-making and secure task execution across distributed edge nodes. The procedure is organized into four successive, interrelated steps, as described below. The proposed protocol integrates context-aware computing, Bayesian reasoning, and RL into a unified advanced intelligence-enabled edge computing architecture. It also introduces a lifecycle-aware security and compliance mechanism that dynamically adapts to runtime context changes. This protocol provides a multi-layer decision pipeline (IoT–Edge–Cloud) that optimizes both trust and resource efficiency simultaneously.

A. Context Acquisition Layer

The layer gathers multi-dimensional contextual data, including operational metrics, environmental parameters, user behavior, device conditions, and network conditions. IoT

sensors, edge devices, system logs, and network monitors are the primary data sources. This approach, in contrast to existing single-context models, allows the system to be dynamically adapted to changing runtime contexts and forms the basis for robust and intelligent edge operations.

The proposed AIP-enabled context-aware edge-computation architecture is designed to deliver adaptive intelligence, lifecycle-centric security, and seamless compliance management within highly dynamic and heterogeneous edge environments. The subsystem adopts a modular and layered design, enabling scalability, interoperability, and flexibility while supporting real-time decision-making and secure task execution throughout the distributed part nodes.

This design is consistent with existing context-aware IoT frameworks, which emphasize the aggregation of diverse contextual information as a key requirement for enabling intelligent and adaptive system behavior. In contrast to traditional single-context models, such frameworks enable systems to adapt to dynamic runtime conditions, supporting robust edge intelligence.

B. Multi-Context Processing Engine

The multi-context processing engine integrates contextual information with a hybrid framework combining Bayesian inference and DRL. Bayesian reasoning is used for probabilistic context modeling and uncertainty management, whereas DRL is employed for adaptive task orchestration and resource allocation. This combination enables the system to continuously optimize decision-making strategies under dynamic workloads, varying network conditions, and operational constraints, providing higher adaptability compared to traditional context-aware edge systems.

C. Lifecycle Security Manager

The Lifecycle Security Manager (LSM) is used to enforce dynamic security policies across edge service lifecycle operations. Unlike traditional static or perimeter-based security systems, it adapts to real-time conditions such as anomaly detection, node mobility, variations in trust levels, and changes in network behavior. This ensures proactive mitigation of threats, secure execution of tasks, and long-term trust in decentralized environments.

The Development Safety Executive (DSE) subsection extends this functionality by enforcing security policies throughout the complete operational lifecycle of edge facilities. In contrast to static or perimeter-based security methodologies, it addresses runtime conditions such as anomalies, node mobility, trust-level fluctuations, and evolving network behavior. The design is based on lifecycle-aware and context-driven security principles, ensuring proactive threat mitigation, secure task execution, and trust preservation in decentralized edge environments.

D. Adaptive Task Scheduler

The Adaptive Task Scheduler (ATS) applies the information of the engine in assigning computational workloads to edge nodes and cloud resources. The scheduling decisions are optimized with respect to multiple objectives, including latency minimization, energy consumption reduction, workload balancing, and compliance requirements. Scheduling can be implemented using RL to ensure improved performance over static and heuristic approaches, maintaining system responsiveness and efficient resource utilization under dynamic conditions.

E. Compliance Monitor

The compliance monitor ensures compliance with regulatory and policy standards such as the General Data Protection Regulation (GDPR) and ISO/IEC 27001. Unlike traditional compliance mechanisms that rely on offline audits, this module operates in real time and dynamically adapts its configurations to ensure compliance under changing operational conditions. The compliance monitoring process is integrated into the control loop to ensure that performance optimization aligns with legal, ethical, and organizational requirements. The interaction between IoT devices, edge nodes, and the engine is illustrated in Figure 3.

F. Architectural Integration and System Operation

The architecture integrates compliance enforcement into the control loop to ensure that performance optimization adheres to technical, legal, ethical, and organizational constraints. The combination of multi-context reasoning, lifecycle security enforcement, adaptive scheduling, and continuous compliance monitoring forms a unified decision-making framework that operates as a single intelligent control system. This comprehensive design addresses fragmentation in existing systems and contributes to secure, resilient, and regulation-aware operation in large-scale distributed environments. Although the proposed AIP-based architecture builds upon established principles, the integration of these components effectively addresses key limitations of current edge computing systems.

III. METHODOLOGY

The proposed approach is based on Bayesian modeling and adaptive decision-making through DRL, enabling adaptive, secure, and compliance-aware decision-making in autonomous edge environments. By tightly integrating context interpretation, intelligent task scheduling, and lifecycle-aware security mechanisms, the framework provides a unified workflow capable of responding to rapidly evolving operational conditions while respecting system constraints.

Context data are captured by IoT sensors and edge devices, and probabilistic reasoning is performed where contextual states are estimated using Bayesian inference. These inferred states are then provided to the DRL agent, which evaluates potential actions based on a reward function that considers trust, latency, and compliance. The selected scheduling decisions are executed in the environment, and feedback is used to update the agent's policy through continual learning. This closed-loop workflow ensures adaptive, secure, and regulation-aware coordination across distributed edge systems.

Raw data collected from IoT sensors, edge devices, and network logs are transformed into multi-dimensional context vectors capturing temporal, spatial, behavioral, and operational characteristics. This representation enables effective modeling

of heterogeneous and complex environmental conditions. Multi-context modeling has been widely recognized as essential for intelligent and adaptive edge systems. The proposed approach, by moving beyond single-context or fixed representations, enhances resilience to varying workloads and dynamic environmental conditions.

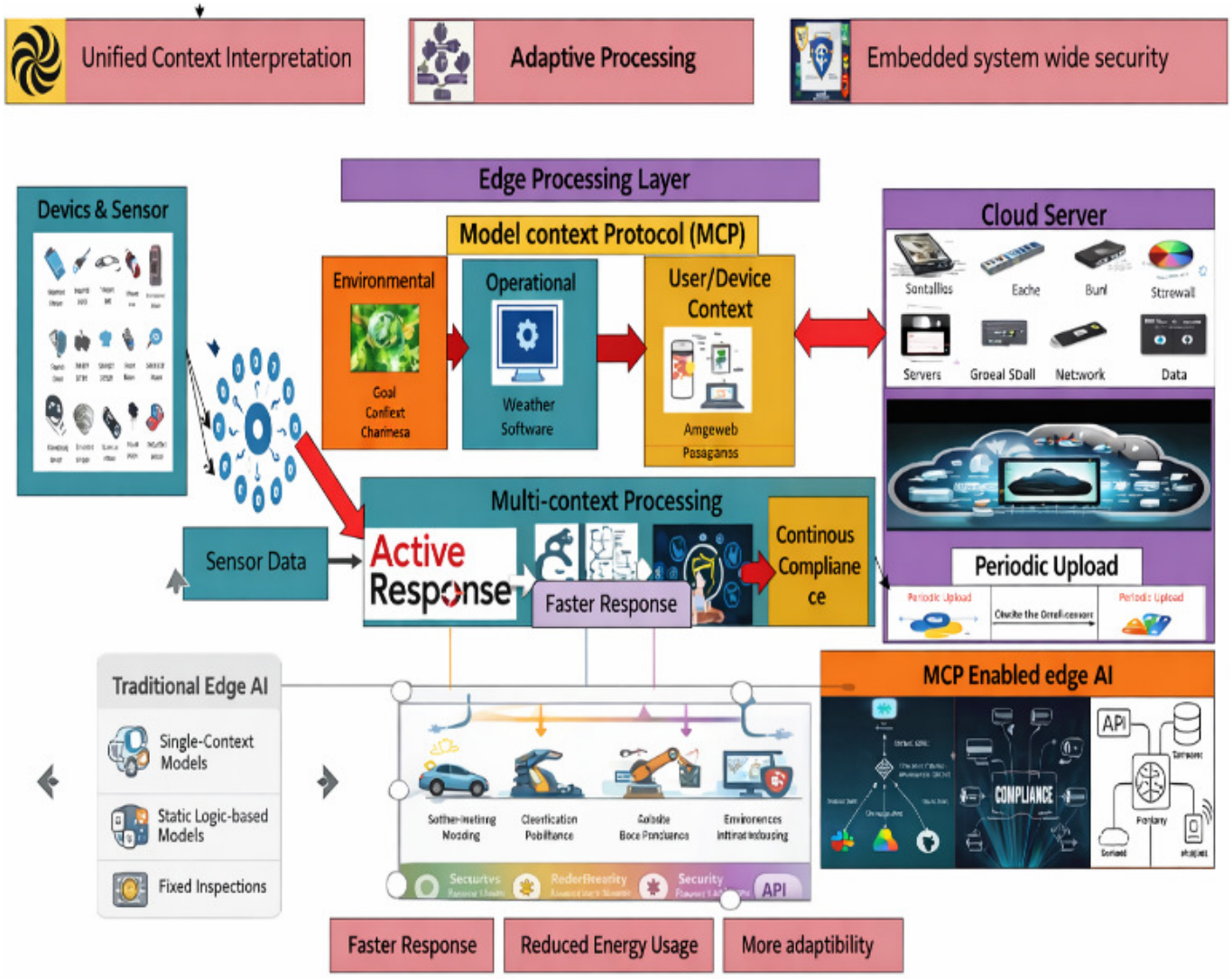

Fig. 3. AIP-enabled context-aware edge computing architecture.

A. *Probabilistic Context Reasoning*

Bayesian inference is applied to approximate the probability distribution of various contextual states. Bayesian reasoning enables the system to operate effectively in the presence of uncertainty and incomplete information, which are common characteristics of distributed edge environments. This probabilistic reasoning model supports reliable context interpretation and reduces decision instability caused by noise or missing data. The derived contextual states serve as high-level inputs for subsequent adaptive decision-making processes.

B. *Adaptive Task Scheduling Using Deep Reinforcement Learning*

The DRL-based adaptive task scheduling module is responsible for dynamic task allocation, where a reinforcement learning agent interacts with the edge environment to learn optimal policies for task assignment and offloading. Using contextual information derived from Bayesian reasoning, the DRL agent maximizes cumulative rewards based on multiple objectives, including latency minimization, energy efficiency, workload balancing, and compliance awareness.

Reinforcement learning-based scheduling has been shown to achieve better overall performance than static and heuristic scheduling methods in dynamic environments. The integration of probabilistic context reasoning and DRL enables the scheduler to continuously adapt to changing device states and operational constraints.

C. *Security and Compliance Adaptation*

Lifecycle-aware security and compliance mechanisms are enforced based on runtime triggers such as anomaly detection,

node mobility, trust variation, and network changes. In contrast to conventional static security models, the proposed approach regulates security policies in real time to address emerging threats throughout the system lifecycle. Simultaneously, the compliance mechanisms continuously monitor regulatory standards, including GDPR and ISO/IEC 27001, and dynamically reconfigure system settings to ensure regulatory adherence under autonomous operational conditions. This tight integration of intelligence, security, and compliance enables reliable and regulation-aware edge system operation.

D. Implementation Environment

The proposed methodology is implemented in Python for model development and DRL training, using TensorFlow and PyTorch frameworks. NumPy and Pandas are used for data preprocessing and probabilistic computations. The computational modules are containerized using Docker to enable scalable and reproducible deployment, whereas Kubernetes is employed for orchestration and resource management across distributed edge nodes. The overall implementation follows modern edge-deployment practices and ensures interoperability and scalability in heterogeneous environments.

IV. RESULTS AND COMPARATIVE ANALYSIS

The proposed AIP-enabled context-aware edge computing framework is evaluated in a simulator-based IoT environment consisting of 150 heterogeneous edge nodes, representing realistic deployment scenarios typically considered in edge computing research under simulated conditions. The proposed method is compared with three standard baseline models in edge computing research: Context-Aware Edge (CAE), Secure Edge Intelligence (SEI), and Federated Context-Aware Edge (FCAE). The evaluation focuses on five key performance metrics widely adopted in edge intelligence studies: latency, energy efficiency, compliance accuracy, decision accuracy, and adaptability.

A. Workload, Network, and System Model

The system is evaluated using real-world-inspired case studies in industrial automation. The workload consists of heterogeneous multimodal data streams with varying arrival rates and computational intensities. The network model follows a hierarchical architecture comprising IoT devices, edge nodes, and a cloud layer. Empirical distributions are used to model network latency and bandwidth variations, reflecting realistic edge conditions.

B. Simulation Environment and Implementation Setup

The simulation environment was implemented using the following software:

1. IntelliJ Integrated Development Environment (IDE).
2. Windows 11 operating system.
3. EdgeCloudSim simulation framework available at https://github.com/CagataySonmez/EdgeCloudSim [20].

The EdgeCloudSim framework is modified to implement the proposed AIP protocol. The updated directory structure is shown in Figure 4.

A dedicated MCP module is introduced with the following components: Compliance.java, ContextCollector, ContextData, Lifecycle, MCPEngine, and MCPOrchestrator.

The simulation is conducted using the following configuration:

1. Number of nodes: 10.
2. Simulation duration: 1000 s.
3. Task arrival rate: 0.5.
4. Task size range: 0.5–5.0.
5. Task complexity range: 0.1–1.0.

The simulation is implemented in Python.

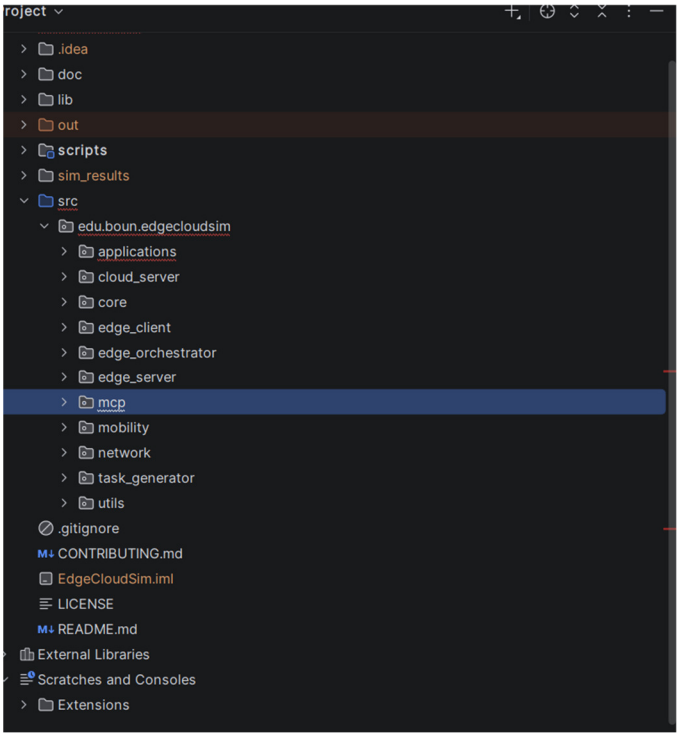

Fig. 4. Directory structure of the proposed AIP protocol.

C. Baseline Models

The proposed method is compared against the following baselines:

- CAE: Rule-based context-aware scheduling without adaptive learning [8].
- SEI: Trust-aware task allocation focusing on security mechanisms [9].
- FCAE: Federated learning-based optimization without lifecycle security or compliance integration [14].

These baselines are selected based on established edge computing frameworks in the literature. CAE provides early context-aware scheduling foundations [8], SEI focuses on security-aware IoT edge systems [9], and FCAE represents federated learning-based edge optimization approaches [14].

D. Performance Metrics

The performance metrics used are defined as follows:

1. Latency (ms): Average task completion time
2. Energy (mW): Energy consumed per task
3. Accuracy (%): Correctness of model decisions
4. Adaptability (%): Ability to respond to dynamic changes
5. Compliance (%): Degree of adherence to regulatory constraints

E. Results and Discussion

Table I presents the performance comparison of all evaluated models. The proposed AIP framework achieves the best performance across all metrics.

TABLE I. PERFORMANCE COMPARISON OF EDGE COMPUTING MODELS

Model	Latency (ms)	Energy (mW)	Accuracy (%)	Adaptability (%)	Compliance (%)
CAE [8]	45	0.82	89	78	87
SEI [9]	38	0.76	92	81	91
FCAE [14]	33	0.71	93	84	93
Proposed AIP	25	0.61	96	91	96

The proposed AIP framework achieves a 24.2% reduction in latency compared to FCAE, demonstrating the effectiveness of multi-context fusion and DRL-based adaptive scheduling. This improvement aligns with prior studies indicating that reinforcement learning enhances responsiveness in dynamic edge environments.

In terms of energy efficiency, the proposed framework achieves 0.61 mW, corresponding to a 14.1% improvement over FCAE. This improvement is attributed to context-aware workload distribution and intelligent resource sharing, which reduce redundant computation and inefficient offloading decisions.

Compliance reaches 96%, outperforming all baseline models. This demonstrates the effectiveness of integrating continuous compliance monitoring and lifecycle-aware security enforcement directly into the decision-making loop. Unlike traditional edge systems that treat compliance as a post-processing step, the proposed approach enforces policies dynamically during runtime.

The proposed framework achieves 96% accuracy and 91% adaptability, confirming the effectiveness of combining Bayesian context reasoning with DRL-based control for adaptive decision-making in edge environments. Unlike traditional approaches that optimize isolated objectives, the proposed architecture achieves balanced improvements across all key performance dimensions.

Figure 5 illustrates the comparison of latency, energy consumption, compliance, and accuracy for all models.

Compared to existing context-aware edge computing frameworks, the proposed AIP-enabled architecture achieves superior performance by integrating probabilistic multi-context reasoning, lifecycle-aware security, and compliance-aware scheduling. This integrated control mechanism improves adaptability and reduces latency in distributed environments.

Overall, the experimental results demonstrate that the proposed AIP framework consistently outperforms all baseline models across key performance metrics, including latency, energy efficiency, adaptability, and compliance accuracy. These findings confirm the effectiveness of multi-context fusion and adaptive control in intelligent distributed edge systems.

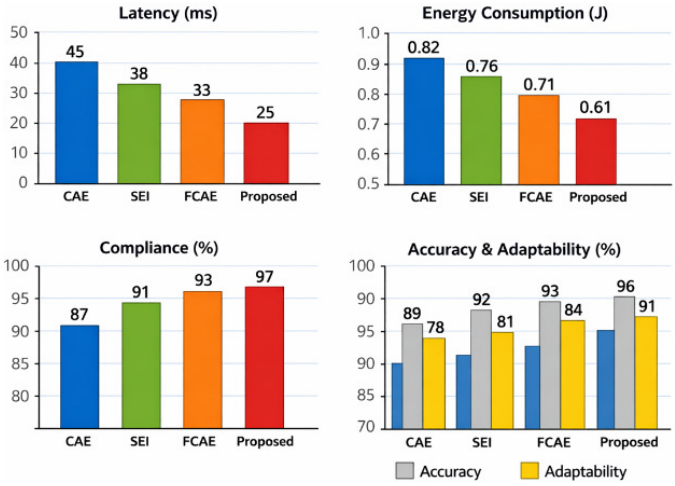

Fig. 5. Graphical representation of latency, energy consumption, compliance, and accuracy and adaptability.

V. CONCLUSION

This proposed system, an Advanced Intelligence Protocol (AIP)-enabled context-aware edge computing framework, enhances efficiency, adaptability, and security in distributed edge environments. The proposed hybrid approach integrates Bayesian probabilistic reasoning, Deep Reinforcement Learning (DRL)-based adaptive task scheduling, lifecycle-centric security enforcement, and continuous compliance monitoring. The proposed architecture addresses key limitations of existing edge computing approaches, which often treat these components in isolation. The unified design enables real-time coordination among context interpretation, security policies, and resource allocation, resulting in more resilient and efficient system behavior under high-speed operational environments.

Comprehensive experimental analysis shows that the proposed AIP framework significantly reduces latency while improving energy efficiency, accuracy, and adaptability compared to existing edge computing approaches. These improvements are achieved through effective multi-context fusion and adaptive decision-making tailored to dynamic workloads, heterogeneous devices, and evolving regulatory constraints. Notably, the results confirm that continuous compliance enforcement can be achieved without degrading overall system performance, which is a critical requirement for

intelligent edge and Internet of Things (IoT) deployment scenarios.

This study highlights the importance of lifecycle-aware intelligence in next-generation edge systems. By enabling security and compliance mechanisms to dynamically adapt to contextual changes, including device mobility, network conditions, and operational anomalies, the proposed framework ensures continuous reliability and regulatory adherence throughout the system lifecycle. This capability is particularly important for mission-critical applications such as smart healthcare, autonomous transportation, and industrial automation, where real-time responsiveness and strict compliance are essential.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no competing interests.

ACKNOWLEDGMENT

This research received no external funding.

DATA AVAILABILITY

No data are made available due to privacy issues; however, partial data may be shared upon reasonable request.

AI USE AND DECLARATION OF GENERATIVE AI USE

No generative AI tools were used.

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, Oct. 2016, https://doi.org/10.1109/JIOT.2016.2579198.

[2] M. Satyanarayanan, "The Emergence of Edge Computing," *Computer*, vol. 50, no. 1, pp. 30–39, Jan. 2017, https://doi.org/10.1109/MC.2017.9.

[3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile Edge Computing: A Survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, Feb. 2018, https://doi.org/10.1109/JIOT.2017.2750180.

[4] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, pp. 680–698, Jan. 2018, https://doi.org/10.1016/j.future.2016.11.009.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017, https://doi.org/10.1109/COMST.2017.2745201.

[6] X. Liu, J. Jin, and F. Dong, "Edge-Computing-Based Intelligent IoT: Architectures, Algorithms and Applications," *Sensors*, vol. 22, no. 12, June 2022, Art. no. 4464, https://doi.org/10.3390/s22124464.

[7] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 414–454, 2014, https://doi.org/10.1109/SURV.2013.042313.00197.

[8] C. Bettini *et al.*, "A survey of context modelling and reasoning techniques," *Pervasive and Mobile Computing*, vol. 6, no. 2, pp. 161–180, Apr. 2010, https://doi.org/10.1016/j.pmcj.2009.06.002.

[9] E. Villar-Rodriguez, M. A. Pérez, A. I. Torre-Bastida, C. R. Senderos, and J. López-de-Armentia, "Edge intelligence secure frameworks: Current state and future challenges," *Computers & Security*, vol. 130, July 2023, Art. no. 103278, https://doi.org/10.1016/j.cose.2023.103278.

[10] M. V. Ngo, T. Luo, and T. Q. S. Quek, "Adaptive Anomaly Detection for Internet of Things in Hierarchical Edge Computing: A Contextual-Bandit Approach," *ACM Transactions on Internet of Things*, vol. 3, no. 1, Oct. 2021, Art. no. 4, https://doi.org/10.1145/3480172.

[11] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019, https://doi.org/10.1109/JPROC.2019.2918951.

[12] Z. Wang, M. Goudarzi, M. Gong, and R. Buyya, "Deep Reinforcement Learning-based scheduling for optimizing system load and response time in edge and fog computing environments," *Future Generation Computer Systems*, vol. 152, pp. 55–69, Mar. 2024, https://doi.org/10.1016/j.future.2023.10.012.

[13] G. Nieto, I. de la Iglesia, U. Lopez-Novoa, and C. Perfecto, "Deep Reinforcement Learning techniques for dynamic task offloading in the 5G edge-cloud continuum," *Journal of Cloud Computing*, vol. 13, no. 1, May 2024, Art. no. 94, https://doi.org/10.1186/s13677-024-00658-0.

[14] A. Brecko, E. Kajati, J. Koziorek, and I. Zolotova, "Federated Learning for Edge Computing: A Survey," *Applied Sciences*, vol. 12, no. 18, Sept. 2022, Art. no. 9124, https://doi.org/10.3390/app12189124.

[15] H. Mashal and M. H. Rezvani, "Multiobjective Offloading Optimization in Fog Computing Using Deep Reinforcement Learning," *Journal of Computer Networks and Communications*, vol. 2024, no. 1, Sept. 2024, Art. no. 6255511, https://doi.org/10.1155/2024/6255511.

[16] S. Liu, Z. Zheng, F. Wu, S. Tang, and G. Chen, "Context-aware data quality estimation in mobile crowdsensing," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, Atlanta, GA, USA, 2017, pp. 1–9, https://doi.org/10.1109/INFOCOM.2017.8057033.

[17] T. Al-Omari, "Context-Aware Anomaly Detection in Microservices Using GCN-Encoded Trace Graphs and LSTM-AE Metrics with Local and Global Embeddings," *Engineering, Technology & Applied Science Research*, vol. 15, no. 6, pp. 29277–29283, Dec. 2025, https://doi.org/10.48084/etasr.13590.

[18] F. Palermo *et al.*, "Advancements in Context Recognition for Edge Devices and Smart Eyewear: Sensors and Applications," *IEEE Access*, vol. 13, pp. 57062–57100, 2025, https://doi.org/10.1109/ACCESS.2025.3555426.

[19] J. A. S. Aranda, R. dos Santos Costa, V. W. de Vargas, P. R. da Silva Pereira, J. L. V. Barbosa, and M. P. Vianna, "Context-aware Edge Computing and Internet of Things in Smart Grids: A systematic mapping study," *Computers and Electrical Engineering*, vol. 99, Apr. 2022, Art. no. 107826, https://doi.org/10.1016/j.compeleceng.2022.107826.

[20] A. R. Nandhakumar, A. Baranwal, P. Choudhary, M. Golec, and S. S. Gill, "EdgeAISim: A toolkit for simulation and modelling of AI models in edge computing environments," *Measurement: Sensors*, vol. 31, Feb. 2024, Art. no. 100939, https://doi.org/10.1016/j.measen.2023.100939.