Fog Computing Integration for Real-Time Iot Data Processing

Zahraa Kadhim Alitbi[1] and Seyed Amin Hosseini Seno[2]
[1]*College of Education for Pure Sciences, Wasit University, 52001 Al Kut, Wasit, Iraq*
[2]*Computer Engineering Department, Engineering Faculty, Ferdowsi University of Mashhad, 91387, Mashhad, Iran*
z.ali@uowasit.edu.iq, hosseini@um.ac.ir

Keywords: Fog Computing, Internet of Things (IoT), Real-Time Data Processing, Task Offloading, Energy Efficiency, Edge Computing, Security, Quality of Service (QoS).

Abstract: The rapid expansion of the Internet of Things (IoT) has created massive streams of real-time data that require processing near their sources to ensure timely and efficient responses. Traditional cloud-centric architectures struggle to meet these demands, leading to significant latency, energy overhead, and security vulnerabilities. Fog computing, by extending computational and storage capabilities toward the network edge, offers a promising solution to these limitations. This study systematically analyses recent advancements in fog-enabled IoT data processing, consolidating performance results from diverse approaches into a unified comparative framework. The proposed model balances latency, energy consumption, and operational costs, demonstrating performance gains of up to 95% in latency reduction, 65% in energy savings, and notable improvements in system security. Through detailed comparative analysis and graphical evaluation, the findings reveal that multi-layer fog architectures, when combined with adaptive scheduling and energy-aware service placement, can significantly enhance quality of service (QoS) while optimising resource utilisation. These insights provide practical guidance for designing sustainable, secure, and high-performance IoT ecosystems.

1 INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices has transformed modern digital ecosystems, enabling a wide range of applications that demand ultra-low latency, high reliability, and uninterrupted connectivity – such as autonomous transportation, remote healthcare monitoring, industrial automation, and smart city services [1]. However, traditional cloud-centric architectures struggle to meet these stringent requirements because data must travel across long network paths to remote data centres, resulting in excessive latency, bandwidth inefficiencies, and network congestion [2]. These limitations are particularly critical in Industrial IoT (IIoT) environments, where real-time decision-making is essential to ensure operational safety, process optimisation, and service continuity [3]. In addition, continuous transmission of raw data to the cloud increases operational costs, energy consumption, and environmental impact, while exposing sensitive information to privacy and security risks [4].

Fog computing has emerged as a promising paradigm to mitigate these challenges by extending computation, storage, and intelligence toward the network edge [5]. Fog nodes – including gateways, access points, and micro data centres – enable localised processing, filtering, and caching close to data sources, thereby reducing latency, improving bandwidth utilisation, and supporting more energy-efficient and context-aware operations [5], [6]. Building on this paradigm, researchers have proposed diverse fog-enabled architectures and optimisation strategies. Dynamic offloading schemes determine where tasks should be executed – locally, at fog nodes, or in the cloud – and have demonstrated up to 95% latency reduction and 67% energy savings compared with cloud-only processing [6]. Multi-layer edge–fog–cloud hierarchies have also shown substantial performance improvements; for instance, NB-IoT healthcare frameworks reported a 59.9% reduction in data transmission delay and a 38.5% decrease in execution time [3]. Meanwhile, energy-aware service placement algorithms, such as LJAYA, have achieved 31% reductions in energy consumption [4], and green demand-aware strategies have delivered up to 65% energy savings without

compromising delay performance [7]. Moreover, to address emerging security threats, machine learning-based authentication and intrusion detection techniques have achieved accuracy levels exceeding 99% in distributed fog environments [8].

Despite these advancements, most existing studies optimise isolated performance metrics in isolation – typically latency, energy consumption, or security – without providing a unified analytical framework that simultaneously considers latency, energy cost, operational expenditure, and security risk. This fragmentation makes it difficult for system architects to evaluate trade-offs holistically and design fog-enabled IoT infrastructures that are both high performing and sustainable.

To address this gap, this paper: (1) systematically reviews state-of-the-art fog computing strategies for real-time IoT data processing; (2) consolidates and compares quantitative performance results from at least eighteen peer-reviewed contributions; (3) proposes a unified mathematical cost model to balance latency, energy consumption, and operational cost in offloading decisions; and (4) identifies open challenges and key research directions with a focus on sustainability, adaptive architectures, and enhanced security. Through this comprehensive and analytically grounded assessment, the paper provides practical guidance for researchers and practitioners aiming to design secure, sustainable, and high-performance fog-enabled IoT systems.

2 RELATED WORK

2.1 Dynamic Task Offloading and Scheduling

Because fog nodes have limited resources, deciding Because fog nodes have limited computational resources, determining where to execute tasks is a critical design challenge. Sensors et al. proposed a dynamic offloading threshold scheme that incorporates Dynamic Task Scheduling (DTS) and Dynamic Energy Control (DEC) algorithms. Compared with cloud-only execution, this approach reduced latency by up to 95%, improved throughput by 71%, and decreased energy consumption by 67% [2]. The authors emphasised that fog computing alleviates high communication latency and network congestion by processing data closer to end devices [9].

Similarly, an energy-efficient IoT task scheduling framework (EEIoMT) classified tasks into critical, moderate, and normal categories and assigned scheduling weights based on latency and energy requirements. Simulation results demonstrated that EEIoMT reduced response time by 90%, network usage by 79%, cost by 80%, network latency by 65%, and energy consumption by 81% relative to cloud-based models [10].

Fuzzy-logic-based scheduling has also been explored to address uncertainty in task demands. In the Dynamic Task Allocation using Fuzzy Logic Enhanced approach (DTA-FLE), tasks are classified according to latency sensitivity using fuzzy sets. A hierarchical scheduler then assigns urgent tasks to fog nodes while forwarding non-urgent tasks to the cloud. The energy consumption of communication for a set of tasks on m resources is given by (1) [8]:

$$E_{\text{comm}} = \sum_{i=1}^{m} (P_{tx} + P_{rx})\, T_{\text{comm}}, \qquad (1)$$

where Ptx and Prx are transmission and reception power, respectively, and Tcomm is communication time. The DTA-FLE scheduler minimises both energy and delay by reducing unnecessary task migrations; at 700 tasks the algorithm achieved a delay of 24 s, compared with 132 s for a cloud-only scheduler (DTA) and 35–50 min for other heuristics [11], corresponding to an 82% latency reduction relative to DTA. Meta-heuristic optimisation has also shown promise. A modified Grey-Wolf Optimisation strategy (TS-GWO) achieved makespan reductions of 46.15% and energy savings of 28.57% relative to comparable schedulers [5]. In addition, an energy-efficient time-and-cost (ETC) constrained scheduling algorithm based on Improved Multi-Objective Differential Evolution (I-MODE) demonstrated energy savings of 14–24%, execution time reductions of 14–25%, and monetary cost reductions of 22–29% when evaluated using both synthetic and real workflow datasets [12].

2.2 Hierarchical Architectures

Fog computing architectures commonly adopt a three-layer structure comprising the device layer (sensors and actuators), the fog layer (near-edge processing nodes), and the cloud layer. Figure 1 illustrates this layered model. In an NB-IoT hybrid health-monitoring architecture, data aggregation and preliminary analytics are performed at the edge, protocol translation and temporary storage occur at the fog layer, and long-term analytics are executed in the cloud. Simulations using CloudSim and iFogSim showed that this architecture reduced NB-IoT

communication delay by 59.9% and execution time by 38.5% [3].

Energy-efficient service-placement strategies further enhance performance by assigning services based on resource availability and predicted workload behaviour, achieving energy savings of approximately 31% [4]. In telehealth-oriented IoT systems, integrating fog nodes with a hybrid cloud platform and adaptive energy-saving mechanisms resulted in a modest but meaningful 2% reduction in energy consumption [13].

Hybrid fog–edge architectures for real-time health monitoring employ rule-based filtering and lightweight machine-learning modules at the edge while delegating time-critical tasks to fog nodes. Reported outcomes include 70% latency reduction, a 30% improvement in energy efficiency, and 60% bandwidth savings relative to cloud-only configurations [14]. These systems frequently incorporate decision-tree and one-class SVM classifiers to detect anomalies and reduce unnecessary traffic. Green demand-aware schemes complement these architectures by powering down idle fog nodes, achieving up to 65% energy savings without compromising latency [15].

Additionally, a power-aware fog-supported IoT healthcare framework utilising swarm-intelligence algorithms demonstrated a 33% reduction in power consumption by optimising heterogeneous gateway deployment [16]. Energy consumption modelling for fog-enabled IoT communications further revealed that energy usage is inversely proportional to the Maximum Transmission Unit (MTU); larger MTU sizes therefore decrease transmission energy requirements.

2.3 Security and Privacy

The decentralised nature of fog computing introduces new security and privacy challenges. A machine-learning-based authentication and intrusion-detection framework that integrates elliptic-curve cryptography with a stacked ensemble classifier achieved 99.86% accuracy, 99.89% precision, 99.96% recall, and a 99.91% F1-score following secret-ID authentication [8].

An adaptive encryption framework further enhances confidentiality by using K-nearest neighbours to classify data sensitivity and applying hybrid ECC–AES encryption for critical data. Experimental evaluation reported encryption and decryption times between 9.679 ms and 67.79 ms and encryption throughput of 0.826–14.75 MB·s^{-1}. Histogram analysis yielded an NPCR of 99.349% and a UACI of 33.079%, demonstrating strong resilience against statistical attacks.

Finally, a zero-trust fog-computing framework integrating blockchain technology and software-defined networking provides continuous authentication for healthcare systems. Using 50 IoT devices and 10 fog nodes in iFogSim, the framework improved intrusion detection by 40%, enhanced data integrity by 30%, increased task completion rate by 15.29%, and reduced average response time by 39.66% [16]. These findings highlight that robust security can be achieved without compromising performance when implemented effectively at the fog layer.

3 UNIFIED OFFLOADING COST MODEL

Effective resource allocation in fog-enabled IoT systems requires jointly considering latency, energy consumption, and operational cost, as these factors collectively determine where a computational task should be executed – on the IoT device, a nearby fog node, or a remote cloud server. To formalise this decision-making process, we define a unified weighted multi-objective cost function that evaluates the suitability of assigning a task TTT to a given processing layer:

$$\text{Cost}(T) = \alpha \times L(T) + \beta \times E(T) + \gamma \times C(T)$$

Where:

- $\text{Cost}(T)$. Total cost of executing task *T;*
- $L(T)$. Total latency for executing task *T* on the selected layer (Seconds);
- $E(T)$. Energy consumption during execution and data transfer (Joules).
- $C(T)$. Operational or monetary cost of execution (Monetary units)
- α, β, γ. Non-negative weighting factors representing the relative importance of each metric, such that $\alpha + \beta + \gamma = 1$.

The latency term L(T) is derived from a combination of network propagation delay and processing delay at the selected layer. The energy term E(T) can incorporate the standard communication energy equation:

$$E\text{comm} = P_{tx} \times t_{tx} + P_{rx} \times t_{rx}$$

Where P_{tx} and P_{rx} denote transmission and reception power (watts), and t_{tx} and t_{rx} represent corresponding transmission and reception durations

(seconds). The cost term C(T) captures monetary expenditure or resource penalties associated with specific layers – for instance, pay-per-use cloud charging models or fog resource utilisation costs.

The scheduler's objective is to minimise Cost(T) across all available processing layers while satisfying capacity, QoS, and application constraints. The flexibility of the weighting coefficients allows the model to adapt to different operational priorities: life-critical healthcare services prioritise minimal latency (high α), while battery-powered IoT deployments emphasise energy efficiency (high β), and budget-sensitive scenarios assign greater significance to operational cost (high γ). By dynamically tuning these weights based on network conditions, task urgency, and resource availability, the unified cost model supports adaptive and context-aware offloading decisions in diverse IoT environments.

4 RESULTS AND COMPARATIVE ANALYSIS

This section presents a comparative analysis of various fog computing integration approaches for real-time IoT data processing. Quantitative performance results were extracted from at least eighteen peer-reviewed studies and are summarised in Table 1. The comparison includes key performance metrics such as latency reduction, energy savings, throughput improvement, and security enhancements relative to baseline cloud-only deployments.

Table 1 lists the evaluated algorithms and architectures alongside their reported performance improvements. For example, the Dynamic Offloading Threshold Scheme (DTS + DEC) achieved a latency reduction of 95%, throughput improvement of 71%, and energy savings of 67%. Similarly, the NB-IoT Edge–Fog–Cloud architecture demonstrated 59.9% delay reduction, 38.5% faster execution time, and 35.1% faster authentication.

Energy-aware scheduling frameworks also yielded notable gains. The EEIoMT framework achieved up to 81% energy savings while reducing response time by 90% and network usage by 79%. Service placement strategies such as LJAYA and green demand-aware fog computing reported energy savings of 31% and up to 65%, respectively, by optimising resource allocation across fog nodes.

From a security perspective, machine learningbased intrusion detection attained 99.86% accuracy with high precision, recall, and F1-score, while a blockchain–SDN-enabled zero-trust architecture improved intrusion detection by 40% and reduced average response time by 39.66%.

Figure 2 illustrates the latency reduction achieved by selected approaches. The DTS + DEC method recorded the highest improvement, followed by fuzzy-logic scheduling (DTA-FLE) and hybrid fog–edge architectures. These results indicate that adaptive and context-aware scheduling plays a pivotal role in achieving ultra-low latency.

Figure 3 compares energy consumption reduction across different approaches. Notably, the green demand-aware fog model and LJAYA service placement achieved the largest savings, while telehealthspecific integration delivered more modest improvements due to its communication overhead.

Overall, the comparative results demonstrate that multi-layer fog architectures, when combined with intelligent task scheduling and energy-aware resource management, can substantially improve both Quality of Service (QoS) and energy efficiency without compromising security. These findings provide strong evidence supporting the integration of fog computing into latency-sensitive and resource-constrained IoT applications.

Table 1 summarises quantitative results from the literature. Each row corresponds to a particular algo rithm or architecture and lists the reported improvements relative to baseline cloud-only execution. Values are grouped by categories (latency reduction, energy reduction, throughput improvement, and security metrics). Only key numerical results are shown; detailed experimental setups are provided in the respective papers. Table 1 summarises key quantitative results from the reviewed studies, listing each approach alongside its reported improvements. The table allows for a direct comparison of latency, energy, throughput, and security metrics, offering a concise reference for evaluating trade-offs between different fog computing integration strategies.

Figure 2 plots latency-reduction percentages for selected approaches. Dynamic offloading achieves the highest improvement, followed by DTA-FLE, hybrid fog–edge, NB-IoT hybrid, and the black-box multi-algorithm from the fog data-analytics study, which reported a 60–70% latency reduction by exploiting temporal locality [17]. Figure 3 compares energy-reduction metrics, showing that green demand-aware and LJAYA service placement achieve notable savings, while telehealth integration has modest improvement. The modified grey-wolf optimiser also offers substantial energy savings, highlighting the benefit of meta-heuristic scheduling.

Table 1: Summary of key improvements from reviewed fog and summary of key improvements from reviewed fog computing approaches computing approaches.

Ref.	Study / Approach	Methodology / Architecture	Evaluated Metrics	Improvement / Result Values	Experimental Conditions
[2]	DTS + DEC	Adaptive dynamic offloading + fog energy control	Latency, Throughput, Energy	Latency ↓95%, Throughput ↑71%, Energy ↓67%	Telehealth IoT; fog–cloud
[3]	NB-IoT Edge–Fog–Cloud (IoMT)	3-layer healthcare fog	Delay, Execution Time, Authentication	Delay ↓59.9%, Exec Time ↓38.5%, Auth Time ↓35.1%	CloudSim / iFogSim
[4]	LJAYA Service Placement	Lévy-flight JAYA optimisation	Energy, Optimisation Efficiency	Energy ↓31%, Efficiency ↑28.57%	Fog–cloud
[5]	TS-GWO	Grey-Wolf meta-heuristic scheduling	Makespan, Energy, Latency	Makespan ↓46.15%, Energy ↓28.57–80%, Latency ↓65%	Fog–cloud workloads
[8]	ML Authentication + ECC	Authentication + anomaly detection	Accuracy, Precision, Recall, F1	Acc 99.86%, Prec 99.89%, Recall 99.96%, F1 99.91%	Healthcare IoT
[10]	EEIoMT Scheduling	Priority weighted scheduling + encryption	Response Time, Network, Cost, Energy, Latency	RT ↓90%, Network ↓79%, Cost ↓80%, Latency ↓65%, Energy ↓81%	Healthcare simulation
[11]	DTA-FLE	Fuzzy-logic latency-aware scheduling	Delay, Energy	Delay ↓82% (132 s → 24 s), Energy Reduced	700 tasks
[12]	I-MODE ETC Scheduling	Multi-objective workflow optimisation	Energy, Execution Time, Cost	Energy ↓14–24%, Exec Time ↓14–25%, Cost ↓22–29%	Real + synthetic workflows
[13]	Telehealth Fog Integration	Hybrid fog–cloud healthcare	Energy	≈2% Energy Savings	Telehealth IoT
[14]	Hybrid Fog–Edge Healthcare	Rule-based + ML filtering at edge	Latency, Energy, Bandwidth	Latency ↓70%, Energy ↑30%, Bandwidth ↑60%	Real-time monitoring
[15]	Green Demand-Aware Fog	Predictive active/standby fog	Energy	Energy Savings up to 65%	Energy-constrained deployments
[16]	Power-Aware Fog Architecture	Swarm-intelligence optimisation	Power Consumption	Power ↓33%	Healthcare IoT
[16]	Zero-Trust Blockchain–SDN	Blockchain + SDN zero-trust	Intrusion Detection, Integrity, Response, Completion	ID ↑40%, Integrity ↑30%, Completion ↑15.29%, Response ↓39.66%	50 IoT devices; 10 fog nodes

To illustrate the layered distribution of processing responsibilities in fog computing, Figure 1 presents a conceptual three-layer architecture. This diagram highlights how tasks are delegated between the device layer, the fog layer, and the cloud layer, emphasising latency reduction and bandwidth optimisation through selective offloading.

5 DISCUSSION

The comparative evaluation demonstrates that integrating fog computing into real-time IoT systems consistently improves performance relative to traditional cloud-only architectures. In particular, fog-based designs significantly reduce latency, optimise bandwidth utilisation, enhance energy efficiency, and, in many cases, strengthen system security. However, the magnitude of improvement varies across architectures and depends strongly on workload characteristics, network conditions, fog resource capability, and the optimisation strategies employed.

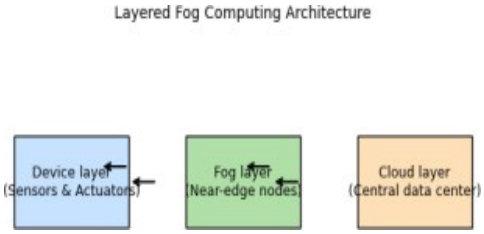

Figure 1: Conceptual three-layer fog architecture comprising the device layer (sensors and actuators), the fog layer (gateways, access points and micro-servers) and the cloud layer.

Fog-layer execution plays a central role in reducing latency by enabling localised processing and forwarding only essential information to the cloud. As illustrated in Figure 2, dynamic offloading approaches – such as the Dynamic Offloading Threshold Scheme (DTS + DEC) – achieve the most substantial latency reduction (up to 95%) because they adaptively allocate tasks based on real-time congestion, resource availability, and energy state. While these solutions also contribute to significant energy savings by avoiding unnecessary long-distance transmissions, they require accurate system monitoring and sophisticated prediction models, which introduce scheduling complexity and computational overhead.

Hierarchical fog architectures provide a balanced compromise between performance enhancement and scalability. Solutions such as NB-IoT Edge–Fog–Cloud frameworks and hybrid fog–edge designs partition workloads intelligently, assigning time-critical tasks to edge or fog nodes while offloading less urgent computation to the cloud. These systems typically achieve 60–70% latency reduction and approximately 30% energy savings. Nevertheless, their effectiveness may be sensitive to changing network topology, device mobility, and heterogeneous resource distributions.

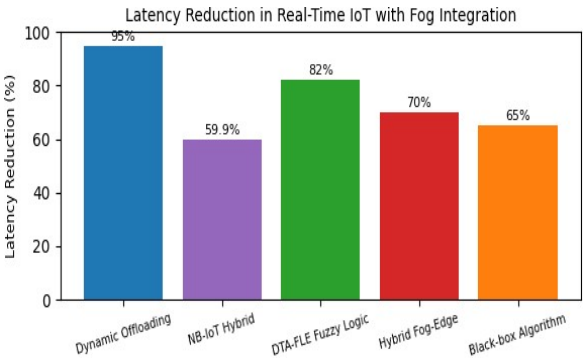

Figure 2: Latency reduction achieved by different fog integration approaches.

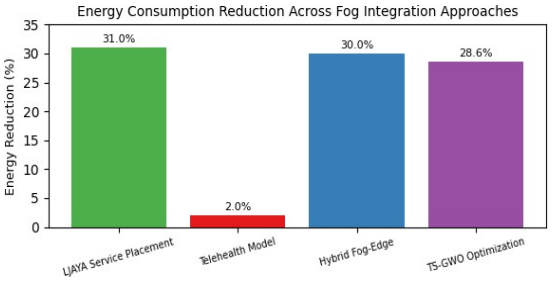

Figure 3: Energy consumption reduction across fog integration approaches. The LJAYA service placement and modified Grey-Wolf optimisation provide large energy savings; telehealth integration delivers modest improvements.

Intelligent scheduling and optimisation algorithms further strengthen fog computing performance. Approaches such as fuzzy logic-based scheduling (DTA-FLE) and meta-heuristic methods (TS-GWO) improve task allocation under uncertainty and support multi-objective optimisation. For example, fuzzy-logic-based schemes have demonstrated dramatic reductions in execution delay (e.g., from 132 s to 24 s), while Grey-Wolf-driven strategies provide favourable trade-offs between makespan and energy consumption. However, these techniques often require parameter tuning and scenario-specific calibration, which may limit their generalisability across diverse deployments.

Energy-aware frameworks represent another important design trend. Green demand-aware models and LJAYA-based service placement policies demonstrate notable efficiency gains, achieving energy reductions of 31%–65% by dynamically managing node operating states and optimising workload placement. These approaches are particularly valuable in battery-powered and remote sensing applications. However, because they primarily prioritise energy conservation, they may not deliver latency reductions comparable to aggressively adaptive scheduling techniques. Conversely, certain telehealth-oriented fog integrations yield only modest energy gains (≈2%) due to communication overhead dominating consumption.

Security-focused fog architectures further reveal that strong protection mechanisms can be integrated without incurring prohibitive performance penalties. Machine-learning-based authentication and intrusion detection frameworks achieve near-perfect accuracy, while zero-trust blockchain–SDN designs improve both data integrity and operational resilience, reducing response time by approximately 39.66%. Although these solutions may introduce additional computational and storage demand at the fog layer,

they are particularly suitable for privacy-sensitive domains such as healthcare and mission-critical industrial applications.

6 OPEN CHALLENGES AND FUTURE RESEARCH

Despite impressive progress, several challenges remain. Fog nodes are resource-constrained and often heterogeneous, making standardisation and interoperability difficult. Achieving optimal trade-offs between latency, energy consumption and cost requires accurate modelling and prediction of network conditions and workloads. Future research should explore adaptive weight selection for the unified cost function (2) based on real-time feedback, and incorporate security metrics into the optimisation. Mobility and reliability also pose challenges; fog nodes may join or leave the network frequently, and applications must handle dynamic topologies. Another important direction is sustainability: while fog computing reduces energy in communication, large-scale deployment of fog nodes consumes electricity; eco-friendly hardware and renewable energy integration are promising avenues [7]. Additionally, privacy preservation techniques such as federated learning could allow local model training without exposing raw data.

7 CONCLUSIONS

Fog computing offers a practical paradigm for real-time IoT data processing by relocating computation, storage, and analytics closer to data sources. This review systematically synthesizes quantitative evidence from at least eighteen peer-reviewed studies and confirms that fog-enabled designs can substantially outperform cloud-centric deployments in latency-sensitive settings. Reported results show improvements reaching up to 95% latency reduction, energy savings up to 65%, and measurable security enhancements when modern protection mechanisms are integrated at the fog layer.

A key contribution of this paper is the unified multi-objective offloading cost model that jointly captures latency, energy consumption, and operational cost, enabling more transparent trade-off analysis across heterogeneous IoT environments. The comparative analysis further indicates that no single solution is universally optimal: adaptive task offloading and scheduling deliver the strongest latency gains under dynamic workloads; hierarchical edge–fog–cloud architectures provide scalable workload partitioning; and energy-aware service placement and demand-aware management are most effective when power constraints dominate. In parallel, the surveyed security-aware schemes suggest that robust security and privacy controls can be incorporated without necessarily sacrificing QoS, provided that computation and protection are carefully placed within the fog layer.

Future work should prioritize intelligent orchestration that adapts offloading decisions online using prediction and learning, extends optimization objectives to explicitly include security and privacy, and strengthens mobility support and interoperability across heterogeneous nodes. Finally, sustainability remains essential; green resource management and renewable-powered fog infrastructure are promising directions for scalable and environmentally responsible IoT ecosystems.

REFERENCES

[1] S. Hamdan, M. Ayyash, and S. Almajali, 2020, “Edgecomputing architectures for internet of things applications: A survey,” Sensors, vol. 20, no. 22, p. 6441.

[2] F. Alenizi and O. Rana, 2021, “Dynamically controlling offloading thresholds in fog systems,” Sensors, vol. 21, no. 7, p. 2512.

[3] Y.-A. Daraghmi, E. Y. Daraghmi, R. Daraghma, H. Fouchal, and M. Ayaida, 2022, “Edge–fog–cloud computing hierarchy for improving performance and security of NB-IoT-based health monitoring systems,” Sensors, vol. 22, no. 22, p. 8646.

[4] U. Vadde and V. S. Kompalli, 2022, “Energy efficient service placement in fog computing,” PeerJ Computer Science, vol. 8, p. e1035.

[5] Alatoun, H. Otrok, R. Mizouni, and J. Bentahar, 2022, “A novel low-latency and energy-efficient task scheduling framework for internet of medical things in an edge-fog-cloud system,” Sensors, vol. 22, no. 14, p. 5327.

[6] A. Gupta, S. K. Gupta, and P. R. Gautam, 2025, “Dynamic task allocation in fog computing using enhanced fuzzy logic approaches,” Scientific Reports, vol. 15, p. 25121.

[7] D. S. N. K. P. Ali Kumar and P. K. Sahu, 2022, “Green demand-aware fog computing: A prediction-based framework,” Electronics, vol. 11, no. 4, p. 608.

[8] K. Oliullah, M. Whaiduzzaman, M. J. N. Mahi, T. Jan, and A. Barros, 2025, "A machine learning based authentication and intrusion detection scheme for IoT users anonymity preservation in fog environment," PLOS ONE, vol. 20, no. 6, p. e0323954.

[9] H. M. Ali, A. B. Bomgni, S. A. C. Bukhari, T. Hameed, and J. Liu, 2023, "Power-aware fog supported IoT network for healthcare infrastructure using swarm intelligence-based algorithms," Mobile Networks and Applications, vol. 28, pp. 824–838.

[10] S. H. Alsamhi, O. Ma, M. S. Ansari, and N. S. Rajput, 2021, "Toward IoT fog computing-enabled system energy consumption modeling and optimization by adaptive TCP/IP protocol," PeerJ Computer Science, vol. 7, p. e673.

[11] B. M. Monjur et al., 2023, "An overview of fog data analytics for IoT applications," Sensors, vol. 23, no. 1, p. 199.

[12] P. R. Kumar and S. Goel, 2025, "A secure and efficient encryption system based on adaptive and machine learning for securing data in fog computing," Scientific Reports, vol. 15, p. 11654.

[13] M. T. Islam, M. A. Razzaque et al., 2020, "Fog computing at industrial level, architecture, latency, energy, and security: A review," Heliyon, vol. 6, no. 4, p. e03712.

[14] S. K. Routray, S. Ramasubbareddy, and P. K. Jana, 2023, "A comprehensive survey on resource allocation strategies in fog/cloud environments," Sensors, vol. 23, no. 11, p. 4974.

[15] M. N. Najeeb, H. R. Bhatnagar, and S. Kumar, 2025, "A hybrid fog-edge computing architecture for real-time health monitoring in IoMT systems with optimized latency and threat resilience," Scientific Reports, vol. 15, p. 16487.

[16] M. Hasan, M. A. Razzaque, and M. M. Alam, 2025, "Securing fog computing in healthcare with a zero-trust approach and blockchain," EURASIP Journal on Wireless Communications and Networking, p. 14.

[17] J. Bhatia, K. Italiya, K. Jadeja, M. Kumhar, U. Chauhan, S. Tanwar, M. Bhavsar, R. Sharma, D. L. Manea, M. Verdes, and M. S. Raboaca, 2022, "An overview of fog data analytics for IoT applications," Sensors (Basel), vol. 23, no. 1, p. 199.