International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 4, July-August 2025||

DOI:10.15662/IJARCST.2025.0804001

Performance Analysis of Edge Computing Architectures for Low-Latency Networking

Ramvriksh Benipuri

Presidency University, Bangalore, Karnataka, India

ABSTRACT: Edge computing has emerged as a cornerstone of low-latency networking, promising to minimize response times by processing data closer to the source. This paper critically analyzes edge computing architectures, evaluating performance in scenarios demanding real-time responsiveness. It integrates findings from architecture proposals, analytic studies, experimental evaluations, and comparative benchmarks, highlighting the contexts in which edge outperforms centralized cloud solutions—and when it doesn't.

We propose a taxonomy of evaluation metrics—latency, tail latency, queuing delays, throughput, resource utilization, and scalability—and outline criteria for architectural comparison. Core contributions include in-depth review of: (1) distributed offloading frameworks optimizing stateless task execution at edge nodes, demonstrating similar or better delays than centralized systems with lower network usage arXiv; (2) analytic and experimental studies revealing "performance inversion"—where edge systems under high utilization yield worse end-to-end latency than cloud alternatives arXivSC21; (3) performance analysis of serverless platforms deployed on resource-constrained edge hardware (e.g., Raspberry Pi clusters), where OpenFaaS delivered lower response times than cloud offerings, albeit with reliability trade-offs arXiv; (4) benchmarking across edge, fog, and cloud layers—in object detection workloads, fog surpassed both, with edge significantly lagging due to limited compute resources MDPI.

The paper presents an evaluation methodology combining realistic workloads, diverse hardware profiles, and layered comparisons. Key findings underscore that while edge brings latency advantages in light-load scenarios, its constrained compute may limit performance under stress. Energy-efficiency and hybrid edge-cloud models offer balanced solutions for industrial applications MDPI. Future work should address dynamic resource provisioning, quantifying queuing thresholds for performance inversion, and developing adaptive offloading strategies.

KEYWORDS: Edge computing; low-latency networking; performance analysis; distributed offloading; serverless edge; fog computing; performance inversion; edge vs cloud; response time; edge benchmarking.

I. INTRODUCTION

The rise of latency-sensitive applications—such as augmented reality, IoT analytics, autonomous systems, and real-time control—has driven the shift from centralized cloud processing to **edge computing**, where data is processed near the source to minimize response time. Reducing latency improves user experience and enables fast decision-making in critical domains.

Edge architectures vary: from micro datacenters near cellular base stations to device-level compute nodes. Understanding their performance trade-offs is vital. While edge query responses may face lower network latency, limited processing power and queuing delays can offset benefits, sometimes resulting in **performance inversion**—where cloud systems outperform under certain workloads arXivSC21.

This paper explores the performance behaviors of various edge computing architectures, including **distributed offloading** frameworks, **serverless server deployments on constrained hardware**, and layered comparisons between **edge, fog, and cloud** sophistication levels. We draw upon prior experimental and analytic studies to systematically evaluate where edge excels—and where it might fall behind.

By defining a performance evaluation framework and synthesizing key findings, this work aids architects in selecting the appropriate infrastructure for different latency-critical use cases. The introduction transitions into the performance-focused literature review, emphasizing metrics, methodologies, and architectural insights for latency-optimized edge deployments.

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 4, July-August 2025||

DOI:10.15662/IJARCST.2025.0804001

II. LITERATURE REVIEW

The literature reveals multiple dimensions of edge performance analysis:

1. **Distributed Offloading Architectures**
2. Cicconetti et al. propose a hybrid architecture combining edge and serverless cloud, using real-time executor selection to offload stateless tasks. Their framework shows comparable or improved delay performance compared to centralized solutions, while using less network resources arXiv.
3. **Performance Inversion under Load**
4. Analytic and empirical studies by Ali-Eldin et al. demonstrate that under medium to high utilization, queuing delays at edge nodes can outweigh lower network latency, resulting in worse end-to-end latency compared to cloud processing—termed **edge performance inversion** arXivSC21.
5. **Serverless Platforms on Edge Hardware**
6. Javed et al. evaluated platforms like OpenFaaS, AWS Greengrass, and OpenWhisk on Raspberry Pi clusters. Results showed that OpenFaaS offered the lowest response time compared to cloud serverless (AWS Lambda, Azure Functions), although cloud had higher reliability arXiv.
7. **Benchmarking across Edge, Fog, Cloud**
8. In object detection workloads using YOLO models, fog consistently outperformed both cloud and edge—cloud due to higher latency and edge due to insufficient resources—highlighting that fog may strike the performance ideal for such compute-heavy tasks MDPI.
9. **Hybrid Architectures and Industrial Use Cases**
10. For industrial condition monitoring, hybrid edge-cloud architectures demonstrated benefits: edge enabled real-time preprocessing and anomaly detection; cloud handled heavy analytics and storage, reducing both latency and bandwidth needs MDPI.
11. **Performance Metric Taxonomy**
12. A comprehensive review defines the metrics essential for evaluating edge/fog/cloud systems: latency (including tail), throughput, resource utilization, energy consumption, reliability, and cost—offering a framework for balanced performance assessment ScienceDirect.

The literature thus reveals contexts where edge provides clear latency gains, and scenarios where its limitations suggest alternative layering or hybrid outcomes.

III. RESEARCH METHODOLOGY

This study synthesizes analytic insights and empirical results to propose a robust methodology for evaluating edge computing performance:

1. **Define Evaluation Metrics**
2. Using the taxonomy from IoT performance reviews, we focus on: **response latency (mean and tail)**, **throughput**, **queuing delay**, **resource utilization**, **energy footprint**, **reliability**, and **network usage** ScienceDirect.
3. **Architecture Classification**
4. We categorize deployments into:
   - **Pure Edge Offloading** with dynamic executor selection arXiv,
   - **Serverless Edge** on low-power hardware arXiv,
   - **Fog Middle Layer** representing intermediate-tier compute MDPI,
   - **Hybrid Edge-Cloud** platforms used in industrial IoT MDPI.
5. **Workload Modeling**
6. Simulations employ representative workloads: stateless tasks, image processing pipelines, and industrial monitoring events, reflecting those used in literature studies.
7. **Analytic vs Experimental Evaluation**
8. Analyze performance inversion thresholds through queuing theory (as per Ali-Eldin et al.) arXiv and cross-validate via observed response time metrics from experimental platforms arXiv+1MDPI.
9. **Comparative Layer Benchmarking**
10. Compare edge, fog, and cloud layers under identical workloads and hardware configurations where possible, using published findings.
11. **Application Context Sensitivity Analysis**
12. Evaluate performance under varying utilization levels and task complexity to identify regimes where edge is beneficial or disadvantageous.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 4, July-August 2025||

DOI:10.15662/IJARCST.2025.0804001

13. **Recommendations Framework**
14. Derive actionable guidance: when to choose edge, fog, or hybrid, based on latency, scale, and hardware constraints.

This methodology balances analytical modeling with empirical insight to provide nuanced performance guidance.

IV. KEY FINDINGS

Our synthesis highlights key performance patterns:

- **Latency Advantage at Light Load**: Edge offloading architectures (e.g., distributed stateless executor systems) can match or improve delays versus centralized cloud, with reduced network usage arXiv.
- **Performance Inversion Under Utilization**: At moderate to high utilization, queuing delays at constrained edge nodes can degrade performance below that of cloud platforms arXivSC21.
- **Serverless Edge Promise With Caveats**: Platforms like OpenFaaS on Raspberry Pi achieve lower response times than cloud-based serverless solutions, though cloud remains more reliable arXiv.
- **Fog Layer Often Outperforms Both Edge and Cloud**: In compute-intensive tasks (e.g., YOLO object detection), fog offers the best balance—faster than cloud, yet more capable than edge MDPI.
- **Hybrid Architecture Benefits**: For industrial IoT workflows, combining edge real-time processing with cloud analytics reduces bandwidth, lowers latency, and improves energy efficiency MDPI.
- **Metric Diversity Matters**: Evaluating only latency risks misleading conclusions. Factors like tail latency, energy use, reliability, and cost must inform architectural decisions ScienceDirect.

Taken together, edge is optimal for light-load, latency-critical tasks close to the data source. For sustained heavy workloads, fog or hybrid models offer better performance reliability.

V. WORKFLOW

A structured workflow for evaluating and designing low-latency edge-enabled systems:

1. **Define Application Requirements**
2. Characterize latency bounds, workload types (e.g., stateless tasks, image processing), scale, and reliability needs.
3. **Select Architecture**
4. Choose from:
   - Pure Edge Offloading
   - Serverless Edge on micro-hardware
   - Fog Layer
   - Hybrid Edge-Cloud
5. **Benchmark Infrastructure**
6. Measure key metrics from existing studies: mean/99th percentile latency, throughput, queuing delays, and energy consumption for representative workloads.
7. **Model Load Impact**
8. Apply analytic models to assess queuing and performance inversion thresholds, informed by Ali-Eldin's framework arXiv.
9. **Choose Deployment Strategy**
   - Light-load, latency-critical → Edge
   - Compute-intensive moderate load → Fog
   - Mixed or scaling workloads → Hybrid
   - High utilization edge stress → Cloud fallback
10. **Monitor and Adapt**
11. Continuously measure system utilization and latency; dynamically shift tasks between edge and fog/cloud based on load and performance thresholds.
12. **Optimize Resources**
13. Use resource provisioning or metaheuristic techniques (e.g., VM placement, greedy heuristics) to minimize latency, energy use, and cost MDPI.
14. **Evaluate Holistically**
15. Periodically assess against the comprehensive metric set: latency, energy efficiency, reliability, resource usage, and operational cost.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 4, July-August 2025||

DOI:10.15662/IJARCST.2025.0804001

This adaptive workflow enables informed architecture selection and runtime optimization for low-latency edge computing.

VI. ADVANTAGES & DISADVANTAGES

Advantages

- **Low Latency under Light Load**: Edge hosts can serve requests with minimal network delay.
- **Bandwidth Savings**: Processing data locally reduces upstream traffic.
- **Energy Efficiency and Responsiveness**: Local computation enables real-time response with low power overhead.
- **Fog-Hybrid Flexibility**: Allows balancing compute power and latency needs.

Disadvantages

- **Performance Inversion Risk**: Under load, queuing delays can outweigh latency benefits.
- **Resource Constraints**: Edge hardware may struggle with heavy compute tasks.
- **Reliability Trade-Offs**: Serverless edge may be less stable than cloud alternatives.
- **Complex Architecture Management**: Hybrid models demand dynamic scheduling and monitoring.

VII. RESULTS AND DISCUSSION

Evidence supports that edge computing delivers tangible latency improvements for light and medium workloads, especially for stateless or simple tasks arXiv. In these scenarios, task offloading to nearby nodes often beats centralized processing, with reduced network traffic.

Yet, edge performance isn't uniformly better. Under moderate to heavy utilization, queuing delays and processor limitations can invert performance benefits—making cloud alternatives faster overall arXivSC21. This emphasizes the need for load-aware deployment decisions.

Serverless edge platforms exhibit promise; for example, OpenFaaS on Raspberry Pis achieved lower response times than AWS Lambda—but cloud offerings delivered superior reliability arXiv.

In tasks like real-time image detection, fog computing has clearly outpaced both edge and cloud, offering optimal latency-performance trade-offs MDPI. This suggests fog as the most balanced tier for compute-heavy, latency-sensitive workloads.

Hybrid edge-cloud architectures further enable systems to leverage edge responsiveness and cloud power—ideal for industrial IoT scenarios involving real-time monitoring and analytics MDPI.

Crucially, performance evaluation must include a broad spectrum of metrics beyond average latency—tail latency, energy use, reliability, and cost are equally important ScienceDirect.

In sum, choosing between edge, fog, and cloud involves trade-offs. Adaptive, workload-aware models that pivot among tiers based on utilization and latency budgets yield the most robust solutions.

VIII. CONCLUSION

This paper consolidated insights into the performance dynamics of edge computing architectures for low-latency networking. Key conclusions:

- **Edge excels** for low-utilization, latency-critical tasks, offering minimal network delay.
- **Performance inversion** occurs when edge nodes are overloaded, leading to higher overall latency than cloud alternatives.
- **Serverless edge models** can reduce response time, though reliability remains a concern.
- **Fog layers deliver optimal performance** for compute-intensive, latency-sensitive workloads.
- **Hybrid edge-cloud architectures** provide a balanced approach, harnessing fast edge responsiveness and powerful cloud analytics.

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 4, July-August 2025||

DOI:10.15662/IJARCST.2025.0804001

- **Comprehensive evaluation** across latency, energy, throughput, and reliability is essential for informed decision-making.

Architects must carefully analyze workloads and infrastructure capabilities to design systems that meet latency requirements without overloading edge infrastructure.

IX. FUTURE WORK

Future research should focus on:

1. **Dynamic Load-Aware Offloading Algorithms**
2. Algorithms that monitor edge utilization in real time and dynamically offload tasks to fog or cloud when thresholds are exceeded.
3. **Quantifying Performance Inversion Thresholds**
4. Modeling and benchmarking the exact conditions (load levels, queue lengths) under which edge performance degrades.
5. **Resource Provisioning Optimization**
6. Apply heuristic/metaheuristic methods (e.g., greedy, particle swarm) to place edge VMs and balance latency vs cost MDPI.
7. **Energy-Aware Deployment**
8. Incorporate energy consumption metrics into scheduling—to optimize edge usage under power constraints.
9. **Scalable Hybrid Orchestration Frameworks**
10. Develop lightweight orchestration layers to coordinate edge, fog, and cloud tiers for latency-sensitive tasks.
11. **Benchmark Standardization for Edge Performance**
12. Establish benchmarks and standards akin to those proposed for IoT performance metrics to ensure comparability ScienceDirect.
13. **Edge Hardware Evolution Tracking**
14. Study how advances in edge hardware (e.g., AI accelerators) impact latency capabilities.

By addressing these areas, future edge systems can deliver consistently low latency while adapting to variable load and architectural constraints.

REFERENCES

1. Claudio Cicconetti, Marco Conti, Andrea Passarella. “Architecture and Performance Evaluation of Distributed Computation Offloading in Edge Computing.” *arXiv*, 2021. arXiv
2. Ahmed Ali-Eldin, Bin Wang, Prashant Shenoy. “The Hidden Cost of the Edge: A Performance Comparison of Edge and Cloud Latencies.” *arXiv*, 2021. arXivSC21
3. Hamza Javed, Adel N. Toosi, Mohammad S. Aslanpour. “Serverless Platforms on the Edge: A Performance Analysis.” *arXiv*, 2021. arXiv
4. Applied study comparing edge, fog, and cloud using YOLO models for object detection. *MDPI*, detail. MDPI
5. Hybrid edge-cloud architecture for industrial condition monitoring. *MDPI*, detail. MDPI
6. Performance evaluation metrics taxonomy for cloud, fog, and edge. *IoT Journal*, 2020. ScienceDirect
7. Edge computing benefits over cloud: latency, scalability, energy usage. *Quantum Zeitgeist article referencing studies*.