Machine Learning-Based Optimization of Distributed Computing Systems for Real-Time Data Processing

Sarah Michelle J
Department of Electrical and Computer Engineering, Stanford University, California, USA

* Corresponding Author: **Sarah Michelle J**

Article Info

ISSN (Online): 3107-6580
Volume: 01
Issue: 05
September - October 2025
Received: 03-07-2025
Accepted: 01-08-2025
Published: 01-09-2025
Page No: 01-03

Abstract
The exponential growth of data generation in modern computing environments necessitates efficient optimization strategies for distributed computing systems handling real-time data processing. This paper presents a novel machine learning-based optimization framework that dynamically adjusts resource allocation, load balancing, and task scheduling in distributed computing environments. Our proposed methodology integrates reinforcement learning algorithms with predictive analytics to achieve optimal system performance while maintaining low latency requirements. Experimental results demonstrate a 34% improvement in processing efficiency and 28% reduction in response time compared to traditional optimization approaches. The framework successfully handles varying workloads and adapts to changing system conditions in real-time scenarios.

Keywords: Machine Learning, Distributed Computing, Real-time Processing, Optimization, Resource Allocation

1. Introduction

Distributed computing systems have become the backbone of modern data processing infrastructure, handling massive volumes of information across multiple computational nodes. The challenge of optimizing these systems for real-time data processing has intensified with the proliferation of Internet of Things (IoT) devices, social media platforms, and streaming services that generate continuous data streams requiring immediate processing and analysis [1, 2].

Traditional optimization techniques often fall short in addressing the dynamic nature of distributed computing environments, where workloads fluctuate unpredictably and system resources must be allocated efficiently to maintain optimal performance. Machine learning approaches offer promising solutions by enabling systems to learn from historical data patterns and make intelligent decisions about resource management and task distribution [3, 4].

The primary objective of this research is to develop an adaptive optimization framework that leverages machine learning algorithms to enhance the performance of distributed computing systems in real-time data processing scenarios. Our approach focuses on three critical aspects: dynamic resource allocation, intelligent load balancing, and predictive task scheduling.

2. Literature Review

Recent studies have explored various approaches to optimize distributed computing systems. Chen et al. [5] proposed a genetic algorithm-based optimization technique that achieved significant improvements in task scheduling efficiency. However, their approach lacks the adaptive capabilities required for real-time environments where system conditions change rapidly.

Reinforcement learning has emerged as a powerful tool for dynamic optimization problems. Wang and Liu [6] demonstrated the effectiveness of Q-learning algorithms in cloud resource management, achieving notable improvements in resource utilization. Similarly, deep reinforcement learning approaches have shown promise in autonomous system optimization, as evidenced by the work of Thompson et al. [7].

Machine learning-based predictive models have been successfully applied to various distributed computing challenges. The research by Anderson and Brown [8] highlighted the potential of neural networks in predicting system performance and proactively adjusting resource allocation strategies. These findings provide a strong foundation for our proposed methodology.

3. Methodology

3.1 System Architecture

Our proposed framework consists of four main components: the Data Collection Module, Machine Learning Engine, Optimization Controller, and Execution Layer. The Data Collection Module continuously monitors system metrics including CPU utilization, memory consumption, network bandwidth, and task completion rates across all distributed nodes.

3.2 Machine Learning Engine

The core of our optimization framework employs a hybrid machine learning approach combining reinforcement learning with predictive analytics. We utilize a Deep Q-Network (DQN) algorithm to learn optimal resource allocation policies through interaction with the distributed computing environment. The DQN agent observes system states, takes actions related to resource distribution, and receives rewards based on performance metrics such as throughput, latency, and resource utilization efficiency.

3.3 Predictive Analytics Component

To enhance the reactive capabilities of the reinforcement learning agent, we integrated a predictive analytics component based on Long Short-Term Memory (LSTM) networks. This component analyzes historical workload patterns and predicts future resource demands, enabling proactive optimization decisions that prevent system bottlenecks before they occur.

3.4 Optimization Strategy

The optimization strategy operates on three levels: node-level optimization focuses on individual computing nodes, cluster-level optimization manages resource distribution across node clusters, and system-level optimization coordinates overall system performance. The multi-level approach ensures scalability and maintains optimization effectiveness as system size increases.

4. Experimental Setup and Results

4.1 Experimental Environment

We conducted extensive experiments using a simulated distributed computing environment consisting of 50 computing nodes with varying computational capabilities. The testbed processed synthetic data streams mimicking real-world scenarios from financial trading systems, social media analytics, and IoT sensor networks.

4.2 Performance Metrics

Our evaluation focused on four key performance indicators: average response time, system throughput, resource utilization efficiency, and adaptation speed to workload changes. We compared our machine learning-based approach against traditional optimization methods including round-robin scheduling, weighted least connections, and static resource allocation strategies.

4.3 Results Analysis

The experimental results demonstrate significant improvements across all performance metrics. Our ML-based optimization framework achieved an average response time of 145 milliseconds compared to 201 milliseconds for traditional approaches, representing a 28% improvement. System throughput increased by 34%, processing an average of 12,500 requests per second versus 9,300 requests per second for baseline methods.

Resource utilization efficiency improved substantially, with our approach achieving 87% average resource utilization compared to 68% for traditional methods. The framework demonstrated excellent adaptability, adjusting to workload variations within 30 seconds while traditional approaches required 2-3 minutes for similar adaptations.

5. Discussion

The superior performance of our machine learning-based optimization framework can be attributed to its adaptive learning capabilities and predictive analytics integration. The reinforcement learning component continuously improves its decision-making policies based on real-time feedback, while the predictive analytics component enables proactive resource management that prevents performance degradation.

The multi-level optimization strategy proves particularly effective in managing complex distributed environments where different optimization approaches may be optimal at different system levels. This hierarchical approach ensures that local optimizations do not negatively impact global system performance.

6. Conclusion and Future Work

This research presents a novel machine learning-based optimization framework for distributed computing systems handling real-time data processing. The experimental results validate the effectiveness of our approach, demonstrating significant improvements in system performance across multiple metrics. The framework's adaptive capabilities and predictive analytics integration position it as a promising solution for modern distributed computing challenges.

Future research directions include exploring federated learning approaches for multi-organization distributed systems, investigating edge computing integration, and developing specialized optimization strategies for emerging technologies such as quantum-classical hybrid computing systems. Additionally, we plan to conduct real-world deployments to validate our findings in production environments.

References

1. Zhang H, Liu W, Chen Y. Distributed computing optimization in big data environments. IEEE Trans Parallel Distrib Syst. 2023;34(5):1456-1468.
2. Anderson JK, Thompson RE, Wilson SA. Real-time data processing challenges in modern computing systems. J Comput Sci Technol. 2023;38(2):234-248.
3. Kumar R, Singh P, Sharma V. Machine learning applications in distributed system optimization. ACM Comput Surv. 2023;55(4):78-95.
4. Wang L, Brown CD, Davis MJ. Adaptive resource management using artificial intelligence. IEEE Trans Cloud Comput. 2022;10(6):2134-2147.
5. Chen X, Li Y, Zhou M. Genetic algorithm-based task scheduling for distributed systems. J Parallel Distrib Comput. 2023;175:45-58.
6. Wang S, Liu H. Reinforcement learning for cloud resource management optimization. Comput Networks. 2023;225:108-122.
7. Thompson KL, Miller AB, Johnson CR. Deep reinforcement learning in autonomous system

optimization. Neural Networks. 2023;159:67-82.
8. Anderson PW, Brown EL. Predictive models for distributed computing performance enhancement. IEEE Trans Comput. 2022;71(12):3045-3058.
9. Martinez C, Rodriguez A, Lopez F. Load balancing strategies in heterogeneous distributed environments. Distrib Parallel Databases. 2023;41(3):345-362.
10. Taylor JM, White RS, Green DK. Scalability analysis of machine learning optimization frameworks. J Supercomput. 2023;79(8):8756-8774.
11. Kim SH, Park JY, Lee MK. Performance evaluation of distributed computing optimization algorithms. Cluster Comput. 2023;26(2):1123-1138.
12. Al-Hassan M, Ahmed K, Ibrahim S. Resource allocation in distributed computing using swarm intelligence. Appl Soft Comput. 2023;136:110-125.
13. Nakamura T, Suzuki H, Watanabe K. Edge computing integration with distributed optimization systems. IEEE Internet Things J. 2023;10(11):9876-9888.
14. Petrov V, Ivanov A, Sidorov N. Fault-tolerant optimization in distributed computing environments. Reliab Eng Syst Saf. 2023;231:109-124.
15. García M, Fernández L, Ruiz P. Energy-efficient optimization strategies for green distributed computing. Sustain Comput Inform Syst. 2023;37:100-115.
16. O'Connor D, Murphy S, Kelly B. Security-aware optimization in distributed computing systems. Comput Secur. 2023;127:103-118.
17. Zhao Y, Wu X, Huang Z. Blockchain-enhanced distributed computing optimization. IEEE Trans Serv Comput. 2023;16(4):2567-2580.
18. Fischer A, Weber M, Bauer T. Quantum-inspired algorithms for distributed system optimization. Quantum Mach Intell. 2023;5(1):23-38.
19. Silva R, Santos M, Costa A. Federated learning approaches in distributed computing optimization. IEEE Trans Neural Networks Learn Syst. 2023;34(7):3456-3469.
20. Bennett C, Adams R, Clark J. Future trends in machine learning for distributed systems. Computer. 2023;56(8):45-53.