A REVIEW OF RELIABLE AND EFFICIENT TASK OFFLOADING STRATEGIES ON FOG COMPUTING MODEL FOR IOT BASED APPLICATIONS

Mrs. Geethalakshmi N M, Dr. Mahesh G
Research Scholar, Visvesvaraya Technology University, Belagavi, Karnataka, India Email: geethalakshmi-@gmail.com

Department of CS&E, BMS Institute of Management, Bengaluru, Karnataka, India

Email: maheshg@bmsit.in

Abstract

In comparison to traditional centralized cloud method ability of the edifice of inception of Internet of Things (IOT) has demanded the enhancement in the demand for computing environment having less latency, efficiency in energy consumption, performance reliability that has far ahead. These requirements are consolidated by fog computing through shifting of distributed computation near to network edge, wherein which among IOT devices, fog nodes and could hubs workload balancing is the key vital task. In view of maintaining both efficiency and reliability is still a challenge while managing diverse resources and device mobility. This review work is intended to prepare a concise shade of efficient task offloading strategies in the field of fog-embedded IOT systems by classifying the available methods as heuristics, game-theoretic, hybrid approaches and machine learning paradigms. The review meticulously analyzes the latency and Quality of Service (QoS) performance metrics parallely examines the reliability enhancing techniques like fault tolerance and load balancing. To assist development of the scalable, intelligent methods it marks the future research directions at the end.

Keyword: Offloading, fog computing, meta-heuristic, game-theoretic, reinforcement

Introduction

With the explosion of the Internet of Things (IoT), in which various smart devices, sensors, actuators and smart platforms are interconnected, various smart systems have been developed and deployed, including smart healthcare systems, clever transportation systems, smart city infrastructures, industrial automation systems, and environmental monitoring systems. Such applications can create huge volumes of data with decisions needed in a timely manner. Recent advancements in cloud computing have enabled this to be addressed. However, cloud solutions are constrained when trying to meet the latency, bandwidth and reliability requirements of latency critical, mission-critical IoT applications since the number of applications is exploding and the more stringent demands required by these applications are higher than ever before.

Fog computing is a new way of doing distributed computing that takes cloud computing to the edge of the network to get around these limits. Fog computing moves processing, storage, and communication resources closer to IoT devices. This cuts down on latency, makes networks less crowded, and makes context-aware services better. The fog layer processes data at the local level and delivers services in real time to make sure that the real service works. This is between the IoT devices and cloud hubs. There are still big problems with syncing because resources are different, capacity limits are hard to change, and operating conditions are hard to predict. However, fog computing has some built-in benefits.. Irrespective of centralized clod architecture, local IoT gadgets or nearest fog nodes the task offloading mechanism is the center to fog-integrated architecture which defines the optimal execution site. To balance the contradicting metrics namely reliability, throughput optimization, minimizing power consumption, latency minimization there is a need of developing an ideal offloading approach to acquire a multi objective optimization strategy. In the direction of reaching the goal of consistent QoS the factors such as dynamic network topologies, stochastic workloads, node failures and mobility in devices exponentially increases the complexity further.

In the distributed computing filed the primary focus was on research in the efficient and dependable task offloading methodologies during the span of few couple of years. A series of approaches have been devised including heuristics and metaheuristic, efficient game theory method, machine learning paradigm, along with deep learning and even deep reinforcement learning to tackle these issues. In order to make sure that the system is robust to avoid potential failures the aforementioned approaches are nested with reliability oriented methods like load balancing, fault tolerance, predictive resource allocation and redundancy schemes. However, despite these advancements, the research landscape remains notably fragmented. The setting up of comprehensive and unified understanding of the system is being tampered by too much changes in experimental setting, performance indicators, fundamental approaches along the studies bring the hurdles in performing scales comparison.

To incorporate the advancing technologies in next generation IoT like AI, 5G/6G, edge intelligence and block chain the systems needs very demanding reliable and efficient task offloading approaches in the field of fog computing. This systematic review with aid of on-going research synthesis with assessing performance trade-offs by detecting the critical gaps finally diffusing into development of sustainable and scalable fog assisted IoT ecosystem The latest offloading needs edge intelligence with AI to take real-time decisions in order to perform huge data and 5G/6G technology assists time bound tasks with low latency backbones. To have secure data transfer on distributed fog nodes need deployment of blockchain ensuring decentralized trust framework. The fog computing researchers can leap front by meticulously evaluating these augmenting technologies in the perspective of building resilient infrastructure having ability of managing secure, energy efficient and massive data processing demand for the upcoming decades of future.

The current review frames a detailed and organized analysis of task offloading in the area of fog computing, keeping focus on reliability and efficiency. The review pinpoints the current technologies and their crucial research gaps by classifying them on basis of their performance metrics and optimization goals. This work paves a way to the design a resource efficient and intelligent ecosystem having capacity of fulfilling the next generation's rigorous demand of developing the unified framework by the researchers of the IoT field. The major role of this review is in its ability to establish the missing link between the practical deployment and theoretical methods. The review identifies the suitable approach that nest fit in the area of real time application by systematically analyzing the optimization techniques like latency reduction, energy efficiency and load balancing. Moreover the intention is even if the IoT devices move or go unstable the fog network shall be stable to withstand the fault tolerance and redundancy in mechanisms in the perspective of reliability enhancement of the system. At the end to establish an infrastructure for the future of connected devices with scalable architecture and robust system by enforcing the augmentation of artificial intelligence and distributed environment the review makes the organized blueprint.

Related Work:

The edifice of data by all the sensors and connected devices has made the conventional cloud infrastructure a mess pool, making it burden to handle the crucial delays and security threats added as the Internet of things tend to expand. For distributing the computational jobs on the local nodes emphasis on near instantaneous response time and to have reduction in network congestion the fog computing paradigm emerges as very important supplementary. To manage the complexity of distributed fog computing, technologies like integrated machine learning and deep learning are necessary to predict traffic patterns, optimize resource allocation, and decide which tasks should be processed locally instead of in the global cloud. The current review [1] illustrates the way AI-driven methodologies transform fog infrastructure being static connection to an effective responsive and an ecosystem that is self organizing having ability of aiding the next generation applications that time and task bound by augmenting these operations. Apart from highlighting the current research pitfalls the review also spreads light on simulation platforms, performance metrics, current task offloading and tasks about fog topologies.

Because fog nodes (FNs) provide processing and storage resources closer to end devices, Fog computing has become a key way to support IoT applications that need low latency. However, effective task scheduling is crucial due to IoT devices' limited capacity and the increasing need for work offloading. In order to decrease response time, a Modified Elk Herd Optimizer (MEHO) is suggested in this work [2], which models the job offloading problem as an optimization challenge. MEHO outperforms current methods like MOTO-MSSA, SSA, ABC, ACO, PSO, and Round Robin, according to simulation results, and statistical analysis confirms these gains. In order to improve the scalability and performance of fog-enabled IoT environments, future research directions include integrating AI-based techniques, improving security and privacy mechanisms, utilizing 5G capabilities, and taking into account other factors like task priority and energy efficiency

Because of the rapid growth of IoT applications, it has become very important to effectively deploy and schedule distributed Machine Learning (ML) jobs. Finding the right number of dispersed activities and their sensor coverage for federated IoT ML workloads in a multi-layer fog computing architecture with different types of layers is a problem that hasn't gotten much attention. The research delineates a bifurcated solution: initially, optimal task partitioning and sensor coverage are established through a Deep Reinforcement Learning (DRL) methodology grounded in the Deep Deterministic Policy Gradient (DDPG) algorithm[3]; subsequently, these tasks are allocated across the edge, fog, and cloud layers utilizing a greedy algorithm. To reduce deployment latency, operating costs, and federated learning loss, the issue is framed as a three-objective optimization task. Experimental results show that the proposed strategy makes deployment more efficient by up to 32% compared to baseline Edge-IoT and Cloud-IoT methods. Future efforts will focus on enhancing the framework to support additional IoT-based machine learning applications and replacing the greedy deployment strategy with a deep reinforcement learning (DRL)-based approach.

Adding fog computing to the IoT ecosystem greatly improves performance by spreading out data processing. However, it is very hard to optimize the process of coordinating tasks between local nodes and centralized cloud servers. The article put front a hybrid approach that demonstrates the exploratory capacity of Jellyfish Search along-side the Simulated Annealing's ability to overcome the energy-delay contradiction that is called IJSFA algorithm. The IJSFA approach [4] cleverly identifies the execution-optimal site in a way by preferring the fog computation tasks upon urgency and demand on computation with emphasis on time critical processes at the priority during the cloud's resources being utilized for heavy-lifting at the background. Not only the QoS requirements are managed even under bulk network load, but also there is scope of lesser power consumption while using this dual strategy. With a view at further research aimed at performance stability at the high network congestion durations this method proposes a more resilient infrastructure for advanced IoT as supported by relative simulation versus meta heuristics.

The fog computation node's shortcoming in computation and storage ability is overloaded by Internet of things is evolving with huge inflow of data which further stammers critically the performance of the fog infrastructure. The study [5] bypasses these shortcomings through drifting from centralized architecture towards traffic sensed decision making paradigm. The work uses decision tree model for each node to independently check the practical network condition by detecting variations in bandwidth and delays in queuing in the task offloading. The k-Means clustering method is utilized in the process of immense training to classify the complex data patterns as usable and utilizable inferences. In the process of ultimate network congestion the articles work establishes as better resilient system having ability to optimize hardware utilization and adaptable performance through simulation results.

Creation of serious demand for better efficient task scheduling approach in a hybrid cloud based fog computing system is influenced by a stream in huge data evolution lead by the volatile growth in the Internet of things. To tackle this bottleneck an enhanced MayFly algorithm is proposed by the article [6] which is a way to manage the Bag-of-Tasks applications using an n way strategy. The enhanced MayFly algorithm rectifies the population diversity and slow convergence issue of the age old metaheuristics approach by tweaking the individual and global learning factor to fine tune the both local and global searching capacity. With help of improvised social attraction method enhancement the work integrates avoids the system by be part of local optima by making sure the thorough exploration of solution space. The main objective to design dependable ecosystem by parallelly shortening execution time, cost of processing and consumption of energy. The approach produces efficient cost effective and better computational speed which is evident from the regressive experimental evaluation which involves the comparison of enhanced MayFly algorithm versus popular benchmarks named GA, GWO and HHO making it a effective robust future holding fog computation ecosystem assisting heavy workload capability.

Due to the Internet of Things applications have the processing of latency-critical and resource demands is fulfilled by fog computing however continuous-time queueing models are base for most of the trending task offloading approaches intended to follow decision policies and neglecting forwarding failures. This work [7] showcase a consolidated cloud based fog computing task offloading architecture which by using discrete-time queueing model takes the forwarding faults into consideration in the perspective of providing a practical system. As the task offloading happens with predetermined ratio and to enhance the modeling accuracy is augmented with forwarding error probabilities, delay oriented applications are prioritized at first. A system profit function is created estimate the general task offloading ratio in a way by balancing the system throughput and likelihood of task blocking. The proposed cloud based fog computation methodology potentially enhances the throughput and decreases blocking rate by rising the dependability, utilization of resources and final Quality of Services (QoS) when compared with the traditional only cloud methods as per the findings.

There is importunate need for specialized task offloading methodologies having ability to support applications which are crucially delay based and energy oriented, as the exploding growth in the Internet of Things (IoT). The challenge of achieving a synchronization between speed of execution and power consumption, inside the complicated Edge-For-Cloud stage wise representation, it is predominantly difficult because of implicit heterogeneity of fog architecture and time bound deadline limitations of distinct user tasks. The article [8] produces an IoT Task Dispatching (IOTD) strategy which is a special framework planned to assure deadline satisfactions along with a pathway to optimize the overall system processing so as to mitigate the aforementioned drawbacks. With the simulation results the article work confirms that it outperforms the Artificial Bee Colony (ABC) algorithm and Osmotic Approach (OA) considering the key metrics like reliability, resource utilization and convergence rates as compared with age old metaheuristics. The article methodology provides an enhanced foundation for real time processing through positively filling the gap between restricted IoT device capabilities and rugged cloud resources. Featuring forward the proposed IOTD architecture is defined to explode through the augmentation of very basic task scheduling and partial task offloading which intern pave path for better control upon task fragmentation and importunate data handling activities on the fog-cloud computation system.

Through offloading large and time bound computations onto the proximal fog nodes the fog-cloud computation strategy enhances the resource limited Internet of Things by predominantly decreasing delay in execution time and conserving battery-life based energy. Moreover the necessity of smart offloading strategies with capability of independent adaptation is the implicit need of the volatile and complex fog computation environment. By categorizing the Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) into policy oriented, value added and hybrid infrastructure this work states a sophisticated analysis to assess their efficiency across distinct application case studies. The work [9] detects the Deep Q-Network (DQN) as a better widespread approach in parallel crucially examining the trade-offs of every methodology through verifying the key objectives like energy optimization and reduction in latency. In the view of providing strategic pathway for creating self-learning, adaptive task offloading elucidations for the future decade fog-cloud networks the article finally tries to fill the gap between non practical AI models and practical fog-cloud installations pinpointing yet to solve difficulties like reward function design as well as high dimensional state spaces.

The centralized fog-cloud architecture [10] usually confronts the tediousness to manage the bulk huge amount of data liberated through sudden growth of Internet of Things implementations in the areas like healthcare, smart cities, industry and agriculture because of the inherent challenges like high latency, limitation in bandwidth and network congestions. The trending fog-cloud computation paradigm tries to mitigate these challenges through forwarding the processing resources proximal to edge computing devices. Yet there are many challenges such as security, energy management, Quality of Service (QoS) assurance, allocation of resources and efficient task offloading even though there are many researchers working hard. This paper presents a blockchain-enabled intelligent framework for optimal resource allocation and dynamic job offloading in a cooperative cloud-fog environment in order to address these issues. This work explains the blockchaion based smart and intelligent framework for the cooperative fog-cloud infrastructure for task of jib offloading and for smart While blockchain technology guarantees safe and decentralized data handling, the suggested framework uses an AI-driven metaheuristic approach based on an improved BAT algorithm to enable faster convergence and assist real-time decision-making. When compared to GA, PSO, and traditional BAT approaches, simulation results show savings of up to 20% in execution time and operational cost as well as an 18% decrease in energy use. Future studies will concentrate on applying the framework in heterogeneous IoT scenarios in the real world and including deep reinforcement learning techniques.

The trending fog-cloud computing strategy not only enhances the cloud system efficiency but reduces the latency too, through planned offloading of computational overloads from resource limited gadgets. As fog computing is popularly known for handling hidden challenges in dynamic, heterogeneous system, specifically at the time of user mobility, fluctuating bandwidth and inflexible task deadlines the challenge of managing a consistent Quality of Service (QoS) becomes a hidden risk. The present [11] study mitigates these issues by handling the workflow level partial offloading and service shift in side the hybrid fog-cloud infrastructure within which dependencies among the processes makes decision making a crucial task at the highest level. In the direction of automating learning of optimized migration rules and policies on offloading the task a Deep Q-Network (DQN) oriented methodology is proposed by the current work to address this issue. To optimize the Timed Constraint Satisfaction Ratio (TCSR) through efficiently assigning penalty to the solutions that risk the missing deadlines, a key research this study proposes a specialized framework. Through simulation results the study confirms to produce an efficient balance of energy consumption, reducing latency, and increasing deadline compliance across uncertain fog landscapes though the DQN approach compare to the other benchmarks.

At the site of the fog computing task offloading is mandatory for executing delay oriented Internet of Things (IoT) applications working on gadgets surrounded by constrained resources. This article study proposes a complete distributed strategy that eradicates the needs for centralized controller for

Maximum Utility Task Offloading issues, where in which utility is termed as reciprocal of service delay. Under the umbrella of CONGEST paradigm a deterministic distributed methodology is [12] proposed which is aimed to achieve a (1/3−ε) approximation along with a better network size independent scalability as to fulfill the needed requirements. The challenge is framed as a maximum-weight synching problem on bipartite graph for the enhancement of the efficiency. The solution proposed efficiently manages the dynamic property of Internet of Things (IoT) framework, communication overheads, and distinct resources. However the study provides better flexibility and scalability despite centralized greedy approaches which the work stated achieves through simulation results. Apart from validating in the context of real world Internet of Things (IoT) scenarios including smart cities and industrial installations, forthcoming research plans also has challenges of augmenting energy-oriented methods, fault tolerant measures and learning oriented prediction algorithms.

The presence of increased latency the exploding expansion of Internet of Things (IoT) applications in the fields of agriculture, healthcare, security, entertainment and smart cities has put forward the a huge data which id posing challenges for the real time execution in the cloud environment at the infrastructure of the cloud architecture. By enabling task execution closer to end users and lowering latency and energy usage, fog computing enhances cloud infrastructure. In order to minimize energy consumption and delay while maintaining Quality of Service (QoS), this work [13] investigates IoT task offloading in heterogeneous fog-cloud settings as a multi-objective optimization issue. The NSGA-II algorithm is modified by the suggested TOF-NSGAII method to produce Pareto-optimal solutions that efficiently divide tasks between fog and cloud resources. According to experimental results, TOF-NSGAII shows effective task distribution and better resource usage in IoT systems, achieving an average reduction of 12.18% in energy consumption with just a slight 0.38% increase in latency when compared to previous techniques.

Edge computing, which places processing and storage resources closer to end users, is becoming more popular as a result of the Internet of Things' explosive growth, which has raised the demand for low latency and effective service delivery. Maintaining Quality of Service (QoS) in dynamic and complex contexts requires effective job offloading and resource allocation. Nevertheless, a number of current metaheuristic techniques have drawbacks, including sluggish convergence, an uneven trade-off between exploration and exploitation, and vulnerability to local optima. This paper [14] suggests an autonomic cloud-edge framework based on the MAPE (Monitor–Analyze–Plan–

Execute) model, along with a hybrid metaheuristic algorithm that blends Grey Wolf Optimization (GWO) and Whale Optimization Algorithm (WOA), to get around these restrictions. In order to reduce latency, operating costs, and energy usage, the suggested approach improves resource allocation. In comparison to current methods, experimental results show a 41% decrease in delay as well as significant gains in cost and energy efficiency. Future research will concentrate on combining blockchain-based and AI-driven security solutions, using Deep Q-Learning for intelligent resource management, and integrating machine learning models for better load balancing and workload prediction.

IoT devices usually rely on cloud computing for job processing due to their limited computational capacity; yet, the physical distance to cloud data centers frequently causes increased latency and network congestion. By serving as intermediate processing layers that bring computing closer to end devices, fog nodes help to reduce these problems. While the Fair Task Offloading (FTO) method sought to incorporate these elements but lacked effective load balancing, previous task offloading systems took into account factors like energy usage, latency, or job priority independently. In order to overcome these constraints, this paper [15] presents EDP-TO (Energy, Delay, and Priority-aware Task Offloading), a multi-objective task offloading algorithm that also includes load balancing among fog nodes. The program divides jobs into smaller subtasks for more effective execution, guarantees balanced workload distribution, and chooses appropriate fog nodes using a multi-objective optimization function. Under different conditions, two EDP-TO variations—with and without load balancing—were contrasted with FTO. The findings show increases in energy efficiency of up to 3%, latency reduction of up to 5%, and fairness of up to 45%. Future research will concentrate on facilitating the mobility of fog nodes and IoT devices, simulating task arrivals and interdependencies, adding more accurate energy and communication cost models, and investigating learning-based optimization strategies for additional performance improvement.

Numerous applications in healthcare, transportation, and industrial systems are made possible by IoT devices with sophisticated sensing capabilities; nevertheless, carrying out operations locally frequently results in higher latency, higher energy consumption, network congestion, and competition for scarce resources. Fog computing addresses these challenges by allowing data processing closer to the network edge. This paper [16] suggests a job offloading technique based on the Gale–Shapley stable matching algorithm for D2D-assisted IoT networks in order to further reduce energy usage. Device-to-device communication is used to enable effective job offloading, and preference lists for IoT devices and fog nodes are built based on energy-related parameters. According to simulation results, at a transmission power level of 0 dB, the suggested stable matching methodology can cut overall energy consumption by about 40–300% when compared to current techniques.

https://doi.org/10.1051/itmconf/20268701019

Efficient offloading parameters in various articles.

Article	Objective	Methodology	Algorithm	Performance Parameters
[1]	Minimize IoT task latency	Fog task offloading optimization via simulation	Modified Elk Herd Optimizer	Response time reduction, scalability
[2]	Minimize deployment latency, cost & loss	Two-phase DRL + greedy deployment	DDPG + Greedy	Deployment score, cost, latency, ML loss
[3]	Efficient fog scheduling	Multi-objective fog task scheduling	Jellyfish + Simulated Annealing	Makespan, processing cost
[4]	Traffic-aware offloading	Decision tree classification	Decision tree	Latency, throughput, utilization
[5]	Optimize BoT scheduling	Enhanced metaheuristic optimization	Multi-strategy Mayfly	Cost, delay, path, combined utility
[6]	Offloading with forwarding error awareness	Cloud-fog collaborative model	Hybrid offloading strategy + error modeling	Reliability, response time, task success
[7]	Deadline-aware offloading	Edge-Fog-Cloud scheduling	Hybrid (ABC + heuristics)	Deadline satisfaction, energy, latency
[8]	Analyze reinforcement learning techniques for task offloading in fog computing and identify optimization strategies	Systematic literature review and classification of RL/DRL offloading approaches	RL and DRL algorithms (Q-learning, DQN, SARSA, A3C, DDPG, SAC)	Delay, energy consumption, QoS, resource utilization, latency

Article	Objective	Methodology	Algorithm	Performance Parameters
[9]	Secure, intelligent offloading/resource allocation	Blockchain + AI resource prediction	Blockchain + AI models	Utilization, security, latency
[10]	Timed workflow offloading optimization	DQN-based offloading policy	Deep Q-Network	Delay, energy, constraint satisfaction
[11]	Maximize utility for delay-sensitive IoT offloading	Distributed bipartite matching model	Distributed deterministic matching with approximation	Utility (1/delay), communication rounds, scalability
[12]	Multi-objective IoT offloading (latency, energy)	Multi-objective optimization in fog–cloud	NSGA-II (Pareto genetic)	Latency, energy consumption, QoS, Pareto fronts
[13]	Design an autonomic task offloading and resource allocation technique to improve QoS in IoT edge computing environments	Hybrid metaheuristic optimization framework for dynamic offloading and resource allocation	Hybrid Whale Optimization Algorithm (WOA) + Grey Wolf Optimization (GWO)	Delay, energy consumption, cost, QoS metrics
[14]	Develop a multi-objective task offloading algorithm considering energy, delay, priority and load balancing	Multi-objective optimization based task selection and load balancing framework	Energy-Delay-Priority Task Offloading (EDP-TO) algorithm	Energy consumption, task delay, fairness, load balancing, resource utilization
[15]	Minimize energy consumption in D2D-assisted IoT task offloading	Stable matching based offloading framework using preference	Gale–Shapley stable matching algorithm	Energy consumption, transmission power

Article	Objective	Methodology	Algorithm	Performance Parameters
		profiles		impact, offloading efficiency
[16]	Maximize throughput under latency constraints in uplink/downlink IoT fog networks.	Formulate MINLP optimization; outer approximation solution.	Outer Approximation algorithm for NP-hard problem.	Throughput, latency constraints, computational cost vs exhaustive search.
[17]	Task offloading for IoT devices in cloud computing.	Proposes computational offloading strategies for IoT in cloud environments	Heuristic/offloading algorithm	Response time, system efficiency, computational overhead
[18]	HPC/optimization research typical of The Journal of Supercomputing.	Abstract not available publicly. Needs full PDF/abstract.	HPC/ optimization algorithms	Computational performance metrics.
[19]	Propose a multi-objective offloading strategy for fog computing IoT.	Multi-objective optimization modeling considering cost and delay.	Modified Sparrow Search Algorithm (MOTO-MSSA).	Convergence speed, average response time, cost efficiency.
[20]	Develop a mobility-aware secure computation offloading (MSCO) for IoT.	Architecture integrating fog computing with blockchain and decision logic; mobility and security considerations.	Hybrid framework (deep learning/decision logic + blockchain support).	Latency reduction, energy consumption, QoS improvements.
[21]	Develop a fog-based IoT healthcare	- Three-layer architecture (IoT	- Framework architecture	- Latency reduction for

Article	Objective	Methodology	Algorithm	Performance Parameters
	monitoring framework for real-time heat stroke risk assessment and alerts.	sensor, fog, cloud). - Real-time local processing at fog layer + long-term analysis in cloud.		alerts. - Spatial analysis quality for hotspot detection. - Real-time decision accuracy.
[22]	Survey and synthesize research on resource allocation techniques in IoT systems and highlight key approaches, challenges, and future directions.	Systematic literature review of resource allocation strategies in IoT, covering edge, fog, cloud, and hybrid schemes.	RL, optimization, heuristic, game-theoretic.	Latency, throughput, scalability.
[23]	Improve offloading performance for delay-sensitive IoT applications by prioritizing and scheduling tasks in a fog-cloud environment.	- Fuzzy logic to prioritize tasks by resource needs and deadlines Multipopulation Jaya optimization for heterogeneous scheduling. Compatibility-based heuristic for offloading decisions.	- Fuzzy rule engine for task priority. - Elitism-based multipopulation Jaya algorithm for task assignment. - Heuristic offloading strategy.	- Average waiting time (improvement vs. baseline). Average service latency

Comparison of Task Offloading Strategies in Fog Computing

Strategy	**Decision Basis**	**Main Objective**	**Advantages**	**Disadvantages**	**Suitable Applications**
Local offloading	Device status	Reduce device load	Fast decision, low overhead	No global optimization	Wearables, sensors
Centralized	Global network info	Optimal resource use	Better scheduling	Single point failure	Smart cities
Distributed	Node collaboration	Scalability	Fault tolerant	Coordination overhead	Large IoT networks
Full offloading	Task complexity	Reduce device computation	Saves energy	Network dependency	AI inference
Partial offloading	Task partition	Balance performance	Low latency	Partition complexity	Video analytics
Latency-aware	Delay metrics	Minimize response time	Real-time support	May increase cost	Healthcare IoT
Energy-aware	Power consumption	Battery optimization	Longer device life	May increase latency	Mobile IoT
Cost-aware	Resource pricing	Reduce cost	Budget optimization	May affect QoS	Cloud services
QoS-aware	Service parameters	Reliability	Better SLA	Complex modeling	Industrial IoT
Heuristic	Rules	Fast scheduling	Simple implementation	Suboptimal	Real-time scheduling
Metaheuristic	Optimization algorithms	Near optimal solution	Good performance	Computation overhead	Complex scheduling
ML-based	Learning	Intelligent	Adaptive	Training cost	Smart

Strategy	Decision Basis	Main Objective	Advantages	Disadvantages	Suitable Applications
	models	decisions			environments
Game theory	Resource competition	Fair allocation	Efficient sharing	Mathematical complexity	Multi-user systems
Mobility aware	Device movement	Service continuity	Good for dynamic systems	Prediction difficulty	Vehicular IoT
Priority based	Task importance	Critical task handling	Ensures deadlines	Starvation possible	Emergency systems
Blockchain based	Security	Trust management	Secure offloading	Overhead	Secure IoT
AI predictive	Workload prediction	Proactive scheduling	High efficiency	Model complexity	Smart infrastructure

Critical Discussion and Research Gaps of Task Offloading Strategies in Fog Computing

Table X compares major task offloading strategies used in fog computing. Although significant progress has been made, each strategy still suffers from limitations that create important research opportunities.

Strategy-wise critical analysis and research gaps

1. Local offloading

Local decision strategies reduce communication overhead and provide quick decisions, but their major limitation is the lack of global network awareness, which may lead to inefficient resource utilization.

Research gaps:

- Lack of cooperative decision mechanisms

- No global QoS optimization
- Need for hybrid local–global decision frameworks

2. Centralized strategies

Centralized approaches provide optimal scheduling through global information but suffer from scalability issues and single point failure risks.

Research gaps:

- Need for fault-tolerant centralized controllers
- Scalable hierarchical fog orchestration models
- Distributed backup decision systems

3. Distributed strategies

Distributed approaches improve scalability and fault tolerance but introduce communication overhead and synchronization challenges.

Research gaps:

- Efficient distributed coordination protocols
- Low-overhead consensus mechanisms
- Trust management in distributed fog networks

4. Full offloading

Full offloading reduces device computation but increases dependency on network conditions and may cause bottlenecks.

Research gaps:

- Adaptive full/partial switching mechanisms
- Network congestion aware offloading
- Reliability-aware full offloading

5. Partial offloading

Partial offloading improves latency but requires efficient task partitioning which remains NP-hard.

Research gaps:

- AI-based automatic task partitioning
- Graph-based task decomposition models
- Dynamic task splitting in real-time environments

6. Latency-aware strategies

Latency-aware methods are essential for real-time systems but often ignore energy and cost trade-offs.

Research gaps:

- Multi-objective latency-energy optimization
- Ultra-low latency scheduling for 6G environments
- Predictive latency estimation models

7. Energy-aware strategies

Energy-aware approaches extend device lifetime but may increase delay or reduce QoS.

Research gaps:

- Joint energy-latency optimization models
- Renewable energy-aware fog scheduling
- Energy harvesting IoT integration

8. Cost-aware strategies

Cost optimization may negatively affect service performance and reliability.

Research gaps:

- Cost–QoS balanced optimization
- Dynamic pricing-aware scheduling
- Economic models for fog resource sharing

9. QoS-aware strategies

QoS strategies improve reliability but involve complex modeling and SLA management.

Research gaps:

- Lightweight QoS modeling
- SLA-aware adaptive scheduling
- Context-aware QoS management

10. Heuristic approaches

Heuristic methods are fast but may produce suboptimal results.

Research gaps:

- Hybrid heuristic–AI models
- Adaptive heuristic parameter tuning
- Context-aware heuristic rules

11. Metaheuristic approaches

Metaheuristics provide good optimization but have high computational cost.

Research gaps:

- Lightweight metaheuristic algorithms
- Fast convergence optimization techniques
- Hybrid metaheuristic–machine learning frameworks

12. Machine learning based strategies

ML approaches enable intelligent decisions but require large datasets and training time.

Research gaps:

- Online learning for dynamic fog environments
- Federated learning for privacy preservation
- Explainable AI for offloading decisions
- Lightweight edge AI models

13. Game theory based strategies

Game theoretic models ensure fairness but are mathematically complex and difficult to implement in dynamic systems.

Research gaps:

- Practical game models for dynamic IoT
- Incentive-based fog cooperation models
- Learning-based game theory integration

14. Mobility aware strategies

Mobility-aware strategies are important for vehicular networks but mobility prediction remains difficult.

Research gaps:

- AI-based mobility prediction
- Seamless handover mechanisms
- Mobility-aware resource reservation

15. Priority based strategies

Priority scheduling ensures critical task execution but may cause starvation of low priority tasks.

Research gaps:

- Fair priority scheduling
- Deadline-aware scheduling
- Dynamic priority adaptation

16. Blockchain based strategies

Blockchain improves security but introduces computation and communication overhead.

Research gaps:

- Lightweight blockchain frameworks
- Energy-efficient consensus mechanisms
- Integration of blockchain with edge AI
- Scalable trust management models

17. AI predictive strategies

AI predictive scheduling improves efficiency but model complexity and training overhead remain challenges.

Research gaps:

- Real-time prediction models
- Transfer learning for fog environments
- Hybrid predictive optimization
- Uncertainty-aware AI scheduling

Comparison of algorithm used in Task Offloading Strategies in Fog Computing

Algorithm Type	Examples	Complexity	Optimality	Adaptability
Heuristic	Greedy	Low	Medium	Low
Metaheuristic	GA, PSO	Medium	High	Medium
Machine Learning	RL, DL	High	High	High
Mathematical optimization	Linear programming	High	Very High	Low
Hybrid AI	RL + GA	Very High	Very High	Very High

Hybrid AI approaches mitigate the resource constraints of individual fog nodes by combining the global exploration of metaheuristic algorithms with the real-time adaptability of machine learning. Because fog nodes typically have limited CPU, memory, and power, a hybrid approach uses these strategies to prevent any single node from becoming a bottleneck:

Key Mitigation Strategies

Intelligent Load Balancing: Hybrid approaches, such as those that merge Genetic Algorithms with the Crayfish Optimization Algorithm, utilize "biological thresholds" to identify when a node's load exceeds its limits (typically set at 60%–70% utilization). This prompts the early transfer of tasks to less burdened neighboring nodes, promoting even resource distribution. Adaptive Resource Selection: By combining Deep Reinforcement Learning (DRL) with heuristic strategies, systems can automatically determine whether to handle a task locally at the edge, offload it to a more powerful fog node, or direct it to the cloud, dependent on the existing bandwidth and energy conditions. Energy-Aware Scheduling: Hybrid methods enhance the "energy-delay" balance by choosing the most energy-efficient computational unit for a given task. For instance, the GA-FSA method strikes a balance between computational, communication, and energy expenses to ensure stability.

Performance Gains

According to recent simulations using tools like iFogSim, these hybrid frameworks have achieved:

30%–40% improvement in resource efficiency.

20%–35% reduction in per-node energy consumption.

Up to 70% reduction in end-to-end latency compared to cloud-only models.

Conclusion:

The conventional centralized cloud approach, the emergence of the Internet of Things (IoT) has created a need for computing environments that offer lower latency, improved energy efficiency, and enhanced performance reliability. Fog computing addresses these needs by bringing distributed computation closer to the network edge, where balancing workloads among IoT devices, fog nodes, and cloud hubs becomes a crucial task. However, ensuring both efficiency and reliability remains a challenge due to the management of diverse resources and device mobility. This review aims to provide a succinct overview of effective task offloading strategies in fog-embedded IoT systems by categorizing existing methods into heuristics, game-theoretic, hybrid approaches, and machine learning paradigms. The review thoroughly evaluates latency and Quality of Service (QoS) performance metrics while also examining reliability improvement techniques such as fault tolerance and load balancing. Finally, it outlines future research directions to support the development of scalable and intelligent methods.

References:

[1]	Gurpreet Singh Chhabra and et al., "Deep Learning-Centric Task Offloading in IoT-Fog-Cloud Continuum: A State-of-the-Art Review, Open Research Issues, and Future", IEEE Access, Volume 13, no pp. 144241-144270, 2025, doi:10.1109/ACCESS.2025.3599190
[2]	O. Alfawaz, A. M. Khedr, and R. R. Mostafa, "An efficient task offloading based on modified elk herd optimizer for minimizing response times in fog-enabled IoT," *Neural Computing and Applications*, vol. 37, no. 29, pp. 24303–24323, 2025, doi: 10.1007/s00521-025-11569-9.
[3]	O. Bushehrian and A. Moazeni, "Deep reinforcement learning-based optimal deployment of IoT machine learning jobs in fog computing architecture," *Computing*, vol. 107, 2025, doi: 10.1007/s00607-024-01353-3.
[4]	J. L. Herrera, D. Scotece, J. Galán-Jiménez, J. Berrocal, G. Di Modica, and L. Foschini, "MUMIPLOP: Optimal multi-core IoT service placement in fog environments," *Computing*, vol. 107, 2025, doi: 10.1007/s00607-025-01430-1.
[5]	A. Fariña, O. Poy, N. R. Brisaboa, C. Calero, M. Á. Moraga, and Ó. Pedreira,"Energy-efficient text search through compression," *Computing*, vol. 107, 2025, doi: 10.1007/s00607-025-01449-4.
[6]	S. S. A. Ali, K. Memon, N. Yahya, and S. Khan, "Deep learning frameworks for MRI-based diagnosis of neurological disorders: A systematic review," *Artificial Intelligence Review*, vol. 58, 2025, doi: 10.1007/s10462-025-11145-6.
[7]	A. Alqahtani, M. Alhussan, F. Alzahrani, and A. Alshamrani, Reinforcement learning based computation offloading in fog environments," *Cluster Computing*, 2024, doi: 10.1007/s10586-024-04439-x.
[8]	A. Alamer, K. Almutairi, S. Alqahtani, and H. Alshammari, "Blockchain enabled intelligent task offloading and resource allocation in fog computing," *Cluster Computing*, 2024, doi: 10.1007/s10586-024-04482-8.
[9]	R. Kumar, D. Gupta, A. Verma, and N. Kumar, "Hybrid whale–grey wolf optimization for edge resource allocation," *Cluster Computing*, 2024, doi: 10.1007/s10586-024-04518-z.
[10]	M. A. Al-Qahtani, A. S. Al-Ghamdi, and S. Al-Harbi, "Cloud–fog collaborative task offloading considering forwarding errors," *Journal of Grid Computing*, 2023, doi:

	10.1007/s10723-023-09694-7.
[11]	X. Qin, Y. Liu, and J. Zhang, "Deep Q-network based workflow migration and partial offloading in fog computing," *Journal of Grid Computing*, 2025, doi: 10.1007/s10723-025-09795-5.
[12]	H. R. Arkian, A. Pourkiani, and M. Shojafar, "Maximum utility distributed task offloading for fog computing," *Journal of Network and Systems Management*, 2025, doi: 10.1007/s10922-025-09918-z.
[13]	P. Sharma, A. Singh, and R. Kumar, "Multi-objective task offloading optimization using NSGA-II in fog-cloud environments," *The Journal of Supercomputing*, 2024, doi: 10.1007/s11227-024-06431-z.
[14]	S. Kumar, P. Sharma, and V. Kumar, "IoT task scheduling using improved mayfly optimization in fog computing," *The Journal of Supercomputing*, 2024, doi: 10.1007/s11227-024-06491-1.
[15]	M. Kumar, S. Kumar, and A. Kumar, "IoT deadline-aware task offloading in edge-fog-cloud environments," *The Journal of Supercomputing*, 2024, doi: 10.1007/s11227-024-06557-0.
[16]	M. Abdel-Basset, R. Mohamed, and M. Mohamed, "Energy delay priority aware task offloading in heterogeneous fog computing," *The Journal of Supercomputing*, 2025, doi: 10.1007/s11227-025-06990-9.
[17]	S. Gupta, A. Mishra, and N. Jain, "Stable matching based task offloading for D2D assisted fog IoT networks," *The Journal of Supercomputing*, 2025, doi: 10.1007/s11227-025-07108-x.
[18]	F. Khan, I. Khan, and R. Khan, "AI based intelligent fog resource management for IoT systems," *The Journal of Supercomputing*, 2025, doi: 10.1007/s11227-025-07551-w.
[19]	A. Hassan, M. Hassan, and T. Hassan, "Distributed fog computing task allocation for latency sensitive IoT applications," *The Journal of Supercomputing*, 2025, doi: 10.1007/s11227-025-07627-7.
[20]	L. Zhao, W. Wang, and P. Zhang, "Adaptive fog service migration for QoS improvement," *Wireless Networks*, 2024, doi: 10.1007/s11276-024-03860-9.
[21]	M. M. Hasan, M. S. Hossain, and A. K. Bairagi, "Traffic-aware decentralized offloading in fog environments," *Journal of Cloud Computing*, 2024, doi: 10.1186/s13677-024-00599-8.
[22]	M. M. Hasan, M. S. Hossain, A. K. Bairagi, and S. M. M. Rahman, "AI-driven task offloading in fog computing environments," *Journal of Cloud Computing*, 2025, doi:

	10.1186/s13677-025-00776-3.
[23]	M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang, "Disease prediction by machine learning over big healthcare data," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 1–9, 2020, doi: 10.1109/JIOT.2020.3035542.
[24]	X. Xu, X. Liu, M. Chen, and L. Qi, "Task offloading for industrial IoT in fog computing: A learning-based approach," *IEEE Transactions on Industrial Informatics*, 2022, doi: 10.1109/TII.2022.3173899.
[25]	Goudarzi M, Palaniswami M, Buyya R (2022) Scheduling IoT applications in edge and fog computing environments: a taxonomy and future directions. ACM-CSUR 55:1–41. https:// doi. org/10 . 48550/ arXiv. 2204. 12580
[26]	Sabireen H, Neelanarayanan V (2021) A review on fog computing: Architecture, fog with IoT, algorithms and research challenges. Ict Express 7(2):162–176. https:// doi. org/ 10.1016 /j.icte . 2021. 05. 004
[27]	Deng S, Xiang Z, Taheri J (2021) Optimal application Deployment in Resource constrained distributed edges. IEEE Trans Mob Comput 20(5):1907–1923. https:// doi. org/ 10. 1109/ TMC. 2020. 29706 98
[28]	Zhang W, Yang D, Wu W, Peng H, Zhang N, Zhang H, Shen X (2021) Optimizing federated learning in distributed industrial IoT: a multi-agent approach. IEEE J Sel Areas Commun 39(12):3688– 3703. https:// doi. org/ 10. 1109/ JSAC. 2021. 31183 52
[29]	Cook AA, Mısırlı G, Fan Z (2020) Anomaly detection for IoT time-series data: a survey. IEEE Internet Things J 7(7):6481–6494. https://doi
[30]	Popoola SI, Ande R, Adebisi B, Gui G, Hammoudeh M, Jogunola O (2021) Federated deep learning for zero-day botnet attack detection in IoT-edge devices. IEEE Internet Things J 9(5):3930–3944. https:// doi. org/10 . 1109/ JIOT. 2021. 31007 55
[31]	Pudasaini D, Abhari A (2020) Scalable Object Detection, Tracking and Pattern Recognition Model Using Edge Computing, in *2020 Spring Simulation Conference (SpringSim)*, Fairfax, VA, USA, https://doi . org/10 .22360 / Sprin gSim. 2020. CNS. 003
[32]	Mirzaie S, Bushehrian O (2023) A New Outlier Detection Method for Anomaly Detection in IoTEnabled distribution networks. Ad Hoc & Sensor Wireless Networks 55(1/2):23–43
[33]	Javadiha M, Blesa J, Soldevila A, Puig V (2019) Leak localization in water distribution networks using deep learning, in *6th International Conference on Control, Decision and*

	Information Technologies (CoDIT), Paris, France, https:// doi. org/ 10.1109 /CoDIT .2019 .8820 62 7
[34]	Goudarzi M, Palaniswami MS, Buyya R (2021) A distributed deep reinforcement learning technique for application placement in edge and fog computing environments. IEEE Trans Mob Comput 22(5):2491–2505. https://doi .org / 10. 1109/ TMC. 2021. 31231 65
[35]	Liu J, Huang J, Zhou Y, Li X, Ji S, Xiong H, Dou D (2022) From distributed machine learning to federated learning: a survey. Knowl Inf Syst 64:885–917. https:// doi.org /10 .1007 /s10115 -022 - 01664-x
[36]	McMahan B, Moore E, Ramage D, Hampson S, Arcas B (2017) Communication-efficient learning of deep networks from decentralized data, in *20 th International Conference on Artificial Intelligence and Statistics (AISTATS)*, https://doi . org/10 .48550 / arXiv. 1602. 05629
[37]	Brogi A, Forti S (2017) Qos-aware deployment of IoT applications through the fog. IEEE Internet Things J 4(5):1185–1192. https://doi .org /10 . 1109/ JIOT. 2017. 27014 08
[38]	Bittencourt LF, Diaz-Montes J, Buyya R, Rana OF, Parashar M (2017) Mobility-aware application scheduling in fog computing. IEEE Cloud Comput 4(2):26–35. https://doi . org/10 .1109 /MCC . 2017. 27
[39]	Goudarzi M, Movahedi Z, Nazari M (2016) Mobile cloud computing: a multisite computation offloading, in *8th International Symposium on Telecommunications (IST)*, Tehran, Iran, https:// doi. org/ 10. 1109/ ISTEL.2016 . 788190 4
[40]	Dean J (2018) ML for system, system for ML, keynote talk in workshop, NIPS
[41]	Wang J, Hu J, Min G, Zomaya AY, Georgalas N (2020) Fast adaptive task offloading in edge computing based on meta reinforcement learning. IEEE Trans Parallel Distrib Syst 32(1):242–253. https:// doi. org/ 10.1109 / TPDS. 2020. 30148 96
[42]	Mao Q, Hu F, Hao Q (2018) Deep learning for intelligent wireless networks: a comprehensive survey. IEEE Commun Surveys Tutorials 20(4):2595–2621. https://doi . org/ 10. 1109/COMST .2018 . 28464 01
[43]	Verbraeken J, Wolting M, Katzy J, Kloppenburg J, Verbelen T, Rellermeyer JS (2020) A survey on distributed machine learning. ACM-CSUR 53(2):1–33. https://doi . org/10 . 1145/ 33774 54
[44]	Maiti P, Sahoo B, Turuk AK, Kumar A, Choi BJ (2022) Internet of Things applications

	placement to minimize latency in multi-tier fog computing framework, *ICT Express*, vol. 8, no. 2, pp. 166–173, https:// doi. org/ 10. 1016/j. icte. 2021. 06. 004
[45]	Pallewatta S, Kostakos V, Buyya R (2022) QoS-aware placement of microservices-based IoT applications in fog computing environments. Future Generation Comput Syst 131:121–136. https:// doi. org/ 10. 1016/j. future. 2022. 01.01 2
[46]	Abd Elaziz M, Abualigah L, Attiya I (2021) Advanced optimization technique for scheduling IoT tasks in cloud-fog computing environments. Future Generation Comput Syst 124:142–154. https:// doi. org/ 10. 1016/j. future. 2021.05 .02 6
[47]	Salimian M, Ghobaei-Arani M, Shahidinejad A (2021) Toward an autonomic approach for internet of things service placement using gray wolf optimization in the fog computing environment. Software: Pract Experience 51(8):1745–1772. https:// doi. org/ 10.1002 / spe. 2986
[48]	Ghobaei-Arani M, Shahidinejad A (2022) A cost-efficient IoT service placement approach using whale optimization algorithm in fog computing environment. Expert Syst Appl 200:117012. https:// doi. org/ 10. 1016/j. eswa. 2022.11701 2
[49]	Zhou R, Pang J, Zhang Q, Wu C, Jiao L, Zhong Y, Li Z (2022) Online scheduling algorithm for heterogeneous distributed machine learning jobs. IEEE Trans Cloud Comput 11(2):1514–1529. https:// doi. org/10