Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.037214

Article

Survey of Resources Allocation Techniques with a Quality of Service (QoS) Aware in a Fog Computing Environment

Wan Norsyafizan W. Muhamad[1], Kaharudin Dimyati[2], Muhammad Awais Javed[3], Suzi Seroja Sarnin[1,*] and Divine Senanu Ametefe[1]

[1]School of Electrical Engineering, College of Engineering, University Teknologi Mara, 40450, Shah Alam, Selangor, Malaysia

[2]Department of Electrical Engineering, Faculty of Engineering, University Malaya, 50603, Kuala, Lumpur, Malaysia

[3]Department of Electrical and Computer Engineering, COMSATS University Islamabad, 45550, Pakistan

*Corresponding Author: Suzi Seroja Sarnin. Email: suzis045@uitm.edu.my

Received: 27 October 2022; Accepted: 08 February 2023; Published: 09 June 2023

Abstract: The tremendous advancement in distributed computing and Internet of Things (IoT) applications has resulted in the adoption of fog computing as today's widely used framework complementing cloud computing. Thus, suitable and effective applications could be performed to satisfy the applications' latency requirement. Resource allocation techniques are essential aspects of fog networks which prevent unbalanced load distribution. Effective resource management techniques can improve the quality of service metrics. Due to the limited and heterogeneous resources available within the fog infrastructure, the fog layer's resources need to be optimised to efficiently manage and distribute them to different applications within the IoT network. There has been limited research on resource management strategies in fog networks in recent years, and a limited systematic review has been done to compile these studies. This article focuses on current developments in resource allocation strategies for fog-IoT networks. A systematic review of resource allocation techniques with the key objective of enhancing QoS is provided. Steps involved in conducting this systematic literature review include developing research goals, accessing studies, categorizing and critically analysing the studies. The resource management approaches engaged in this article are load balancing and task offloading techniques. For the load balancing approach, a brief survey of recent work done according to their sub-categories, including stochastic, probabilistic/statistic, graph theory and hybrid techniques is provided whereas for task offloading, the survey is performed according to the destination of task offloading. Efficient load balancing and task-offloading approaches contribute significantly to resource management, and tremendous effort has been put into this critical topic. Thus, this survey presents an overview of these extents and a comparative analysis. Finally, the study discusses ongoing research issues and potential future directions for developing effective management resource allocation techniques.

Keywords: Resource management; task offloading; load balancing; QoS; latency; energy consumption

1 Introduction

Fog computing can be used for several types of networks such as cellular, IoT and vehicular. One characteristic of IoT-based fog networks is that the task sizes have stringent latency and energy requirements. The need for more computational power and storage coupled with the advent of IoT applications certainly enhances the quality of life. However, QoS is a crucial performance evaluation indicator that must be appraised to meet the diverse demands of these IoT applications [1]. The QoS is a critical factor that must be considered when dealing with real-time services such as virtual reality (VR), real-time video streaming applications, online gaming and many others [2–5]. The conventional computing approach is ineffective in dealing with the strict-delay requirements services. To address these limitations, a new fog computing model has been advancing significantly over time. The fog computing approach can manage such requests, allowing computing capacity to move closer to the network edge [6]. Nevertheless, relocating the processing, communication, and memory capacity closer to the source at the network's edge does not guarantee the complete obliteration of the problem. Since fog nodes have limited storage capacity, optimal allocation of resources is vital to improving the effectiveness of the fog computing approach [7].

Systematic literature reviews on resource management strategies in fog networks are few and far between, thus making it challenging to assess and specify research gaps, various trends, and particularly potential future directions for resource management in a fog environment. A comparative review of recent works can be determined, categorised, and synthesised using an organised literature review. Additionally, systematic review facilitates knowledge transfer throughout the research community. Therefore, this study aims to identify, taxonomically classify, and systematically compare the existing studies. The approach concentrates on organising, conducting, and validating resource management for fog computing environments. A methodological review of this studies has identified the following main contributions: the paper highlights the practical reasons for using resource management approaches for fog computing applications. Secondly, this review classifies variable resource management techniques into two main categories: load balancing and task offloading techniques. In addition to that, the load-balancing approaches are organised based on their subcategories, including stochastic, probabilistic/statistic, graph theory and hybrid techniques. Following that, performance metrics are also reviewed for each of the techniques discussed in the paper. Finally, several open research issues and potential future directions in the field of fog computing resource management strategies are discussed.

2 Background

2.1 Overview of Resource Management

The management of resources in the fog layer is a trending issue that must be handled with efficient resource management approaches. As indicated in Fig. 1, the scope of this review is to discuss load balancing and task offloading techniques and their sub-categories respectively. Recent works developed for these two categories will be further discussed in Section 3.

2.2 Overview of Load Balancing

Load balancing, as a mechanism, allocates the workload among multiple resources to prevent any overload or underload on those resources. The main objectives of load balancing methods are

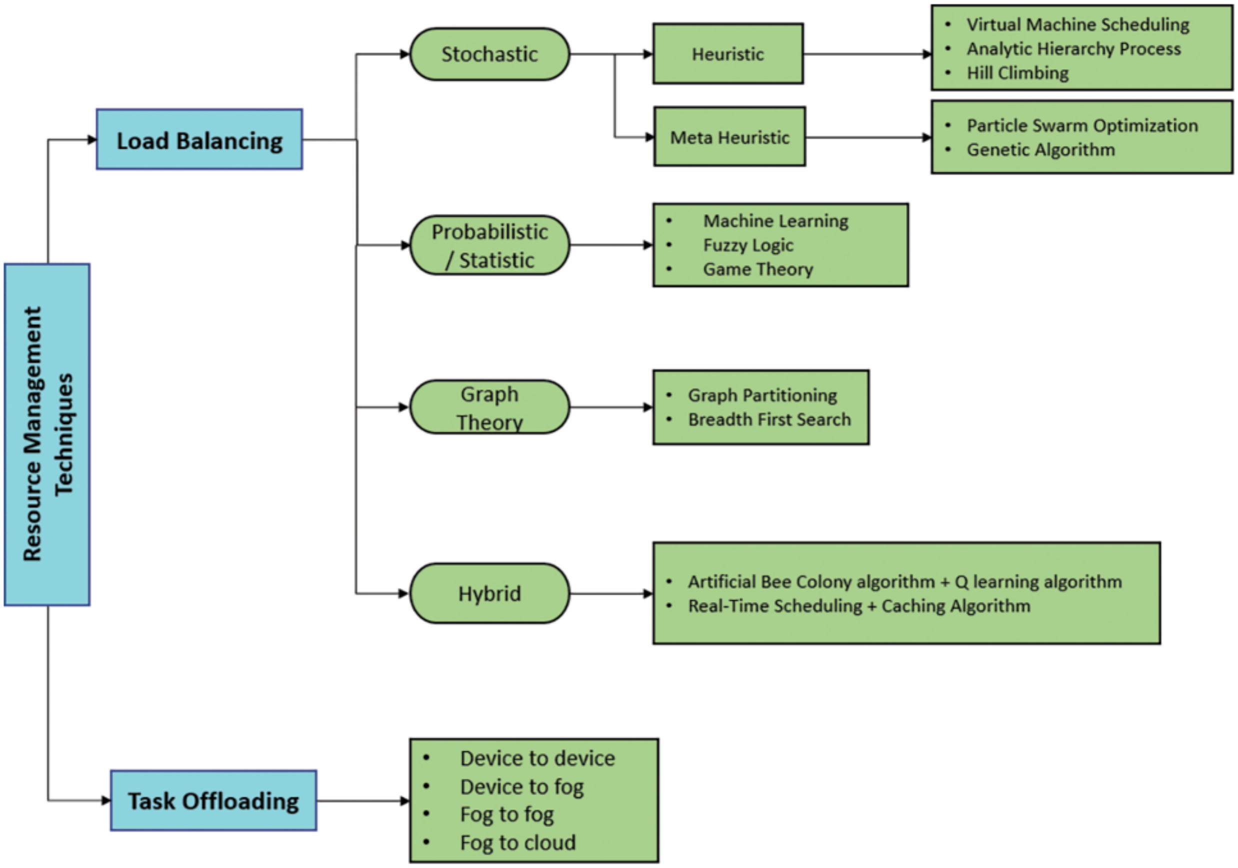

Figure 1: Resource management classification in fog computing

improving the QoS such as throughput, resource usage, feedback time, performances, optimization of traffic as well as increased scalability in dispersed environments [8]. As IoT users proliferate, the strain within the fog layer also grows, thereby increasing load imbalance in fog nodes. Therefore, once more requests are made, fog nodes should be made accessible to handle them quickly and to return responses to end users in real-time. Processing substantial amounts of data in the fog layer requires consideration of accessible resources and equal allocation of workload across nodes. The idle fog nodes use energy like the active nodes when more active fog nodes are present. Thus, energy usage can be decreased at the fog layer by evenly distributing the workload across entire accessible fog nodes, which will also shorten execution time and reduce implementation costs [9]. The devices in the fog computing layer have limited storage and processing power compared to cloud data centres. The only task that requires quick processing is handled by the fog layer. All other tasks are transferred to cloud data centres. Sometimes fog resources are unable handle a high volume of user requests, which then leads to an imbalanced load. Therefore, load balancing is needed to maintain a balanced workload across all resources at the fog layer. The fog computing load balancing framework is shown in Fig. 2. At the fog layer, the load balancer accepts user requests and evaluates the performance and capacity of the Virtual Machines (VM). If a VM is under loaded, tasks from overburdened VMs will be transferred to the under-loaded VMs [10].

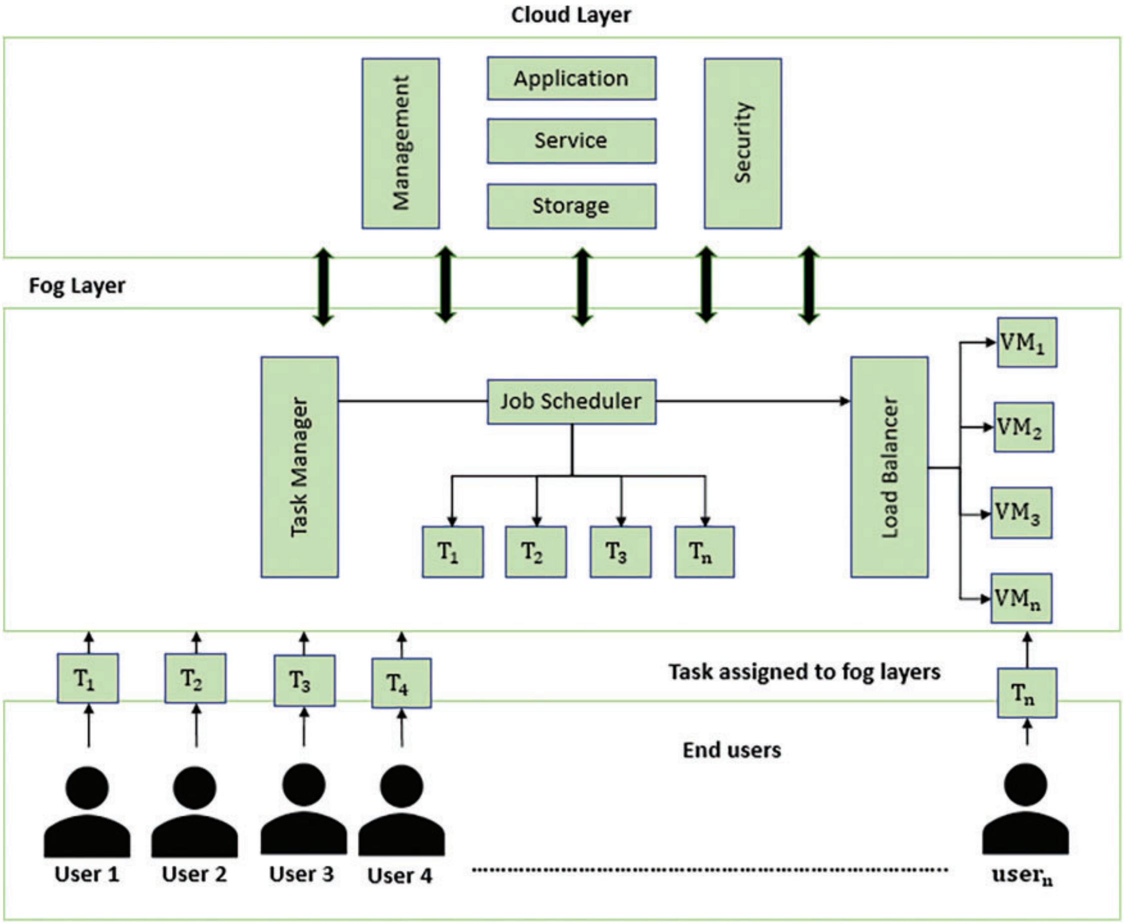

Figure 2: Framework for load balancing in fog computing

2.3 Overview of Task Offloading

Task offloading refers to transferring of resource-comprehensive computing processes to a peripheral platform with significant resources, similar to those utilized in the cloud, edge, or fog computing. Applications requiring massive resources and latency-sensitive can be accelerated by offloading all or some of the work to other processors or servers [11]. Task offloading is a complicated method and may be embargoed by some factors. In particular, task offloading entails software partitioning, offloading choice making and disbursed workload execution [12,13]. Fig. 3 shows a common infrastructure used in an offloading situation. This layer can reduce bandwidth usage, allowing real-time services, reducing energy usage and extending the lifespan of the device's battery [14]. There are a few categories of task offloading options that can be considered, including device-to-device (D2D), device-to-fog (D2F), fog-to-fog (F2F) and fog-to-cloud (F2C).

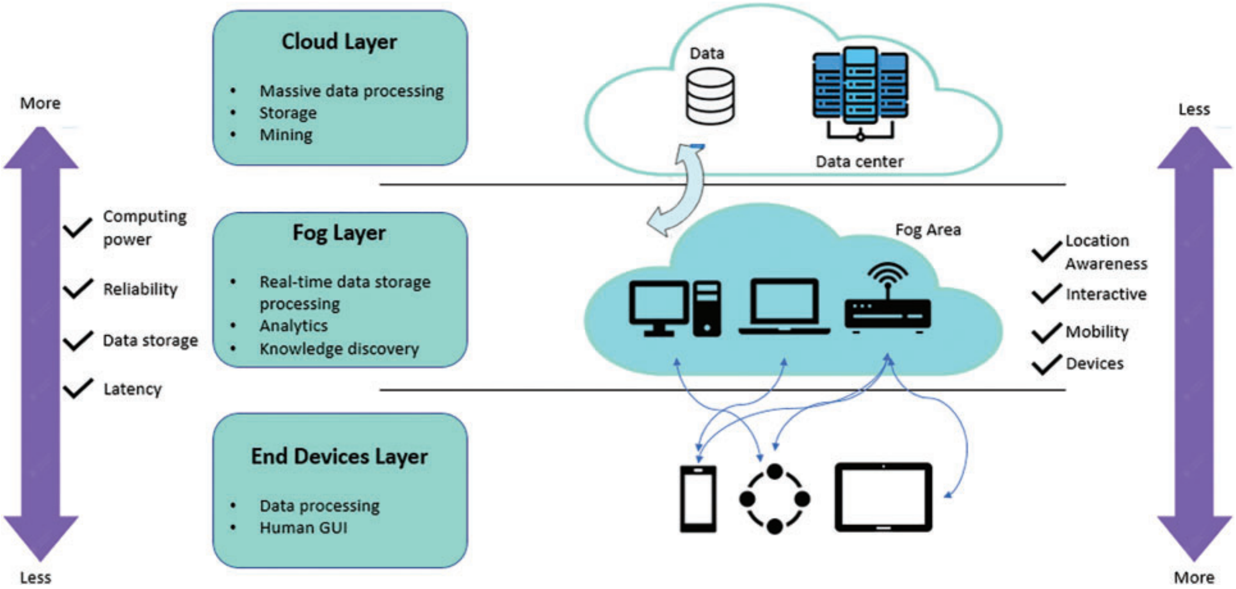

Figure 3: Infrastructure components during task offloading

Among the many distinct offloading strategies, task offloading is a crucial one. This workload offloading method is especially appealing for IoT services and fog-cloud computing [15–18], [19–21]. Offloading deals with several issues that affect optimization, including data management, computational an application's processing needs, latency management, energy management, and more [22,23]. In general, the offloading of a task can be implemented as local execution, full offloading or partial offloading. In the case of local execution, the entire process is carried out locally. This scenario is usually intended for a task which requires low computational power. On the other hand, full offloading is a complete uninstallation process where the entire workload is offloaded and moved to the fog server. Conversely, partial offloading allows data partitioning where flexible components of the tasks are offloaded. A portion of the tasks is managed locally, while the remaining portion is transferred to the fog node. In fractional offloading, a portion of the original task is typically given as; a_j, $0 < a_j \leq 1$. The most important offloading goals, reducing energy usage and latency, depend on choosing the fractions of offloading.

2.4 *Motivation of Our Work*

A plethora of survey articles on resource management strategies are discussed in the literature [24–27]. A number of the papers reviewed recent developments in energy-efficient resource management, energy-aware scheduling approaches, job scheduling, load balancing, etc. In contrast to prior surveys, this paper focuses on resource allocation methods in a fog computing environment with a QoS focus. The strategies for load balancing and work offloading are the main topic of this survey article. One of the most important aspects of our survey paper is how it discusses current trends in load balancing and job offloading approaches, primarily by evaluating a number of papers from 2020–2022. The various resource management strategies are contrasted with one another in light of key elements such as the particular method used, performance measurements, outcomes/contributions, and discussion of their benefits.

3 Recent Works in Load Balancing

The main areas of focus are the general technique, specific method used, evaluation criteria, and outcome of the suggested load balancing systems. The five fundamental classifications of the load balancing approaches are heuristic, meta-heuristic, probabilistic, graph-theoretic, and hybrid. The following section presents numerous techniques proposed during the last few years which could assist potential researchers in bridging the research gap. An overview of the work done in load balancing is shown in Table 1.

Table 1: Summary of load balancing techniques

Authors	Main idea	Method	Performance metrics	Advantage/outcome
Xu et al.	Applied virtual machine scheduling method	Heuristic- VM scheduling method	Resource utilization Load balance variance	Minimize the workload balance variance Increase resource utilization
Fatemah et al.	Employ multiple gateways architecture with Multi-criteria selection-making.	Heuristic-mixed integer linear programming (MILP) Analytic hierarchy process (AHP)	Execution time Energy consumption Fairness	Providing prompt feedback while using less energy Achieve A global load fairness
Zahid et al.	Propose a hill-climbing method	Hill climbing load balancing algorithm	Processing time Network delay	Reduced processing time Minimize network delay
Butt et al.	Proposed genetic algorithm (GA) and binary particle swarm optimization (BPSO)	Meta-heuristic algorithm	Feedback time, processing period, computational budget	Shorten processing and reaction times. Minimize computational cost
Jabour et al.	The Particle swarm optimization (PSO) method utilized Electroencephalography (EEG) signal actuators	Meta-heuristic PSO algorithm	Energy consumption Latency	Reduce energy Consumption and latency
Singh et al.	Implemented fuzzy load balancer for conducting link analysis.	Probabilistic- fuzzy logic-based load balancer	Delay sensitivity Energy consumption Link saturation	Increase efficiency of the overall network
Sardel Fakhrul et al.	Developed a load-balancing approach empowered by Narrowband internet of things (NB-IoT)	Probabilistic game theory	Response time and execution time Energy consumption	Low feedback time Reduce energy consumed
Puthal et al.	Combines both the authentication and load-balancing techniques for Edge datacenter (EDC).	Graph theory—breadth-first search	Success rate Response time	Improves efficiency Enhance security

(Continued)

Table 1: Continued

Authors	Main idea	Method	Performance metrics	Advantage/outcome
Zubair et al.	Presented a hybrid approach combining the bin-packing and the genetic approaches.	Hybrid	Overall cost, Process and feedback time.	Small delay Minimal cost High security
Talaat et al.	The technique based on probabilistic neural networks	Hybrid-combination of neural network and fuzzy logic	Latency Cost Energy	Suited for real-time applications

3.1 Stochastic Approach

3.1.1 Heuristic

For specific optimization issues, heuristic algorithms are developed via experience to discover the best solution swiftly and efficiently through trial and error. Selected heuristic-based algorithms, such as the VM scheduling approach, the AHP, and hill-climbing, are covered in this section. Xu et al. [28] suggested a scheme for dynamically scheduling virtual machines by using the live migration of existing VM. As a result, performance analyses are done to prove the efficiency of the proposed strategy. A queuing problem must first be solved to assess the network task execution latency. Hence, a load-balancing system based on the AHP was developed by [29] to distribute the network entities' global load fairly. Their work presented multiple gateway-based queuing models to assess the effectiveness of traffic load balancing. After modelling, a load-balancing approach based on Multi carrier decision-making (MCDM) is utilized to distribute traffic between the gateways to achieve global load fairness and lower processing latency in the IoT system. The proposed approach provides fast responses to inquiries and achieves global load fairness. Zahid et al. [30] presented a heuristic technique based on the hill climbing load balancing (HCLB) algorithm and the policy of optimizing the best service. The HCLB algorithm efficiently manages the request among VM. This algorithm uses a loop and is executed until the nearest available VM is discovered. Following that, The VM is given the task of processing the requests. The proposed technique minimizes processing time (PT) and response time (RT) within fogs to users. Another work presented by Karypiadis et al. [31] suggests a Smart Auto Scaling Agent (SCAL-E) that balances huge data loads. This work deals with optimal load balancing in virtual machines based on Kubernetes scenarios. The proposed system utilizes MongoDB's scaling, replicating, and sharding features. SCAL-E ensures the efficient distribution of resources and increases the effectiveness of the big data storing and forwarding tasks.

3.1.2 Meta-Heuristic Approach

Jabour et al. [32] implemented the metaheuristic based on the PSO algorithm to manage network resources. The proposed PSO method is represented by utilizing EEG signal actuators. The "iFogSim" simulator is employed to set up an experiment and construct a case study network in the fog layer based on a virtual reality EEG game. The PSO improves the overall energy consumption and the latency of the VR game application compared to previous algorithms. Butt et al. [33] implemented the meta-heuristic technique which proposed a GA and optimization, carried out via a BPSO technique. This technique proposed a multi-layered structure comprising cloud, fog and consumer layers. The main objective of the proposed scheme is to efficiently handle consumer requests at the fog layer and to reduce energy consumption. The techniques are compared with an existing PSO algorithm as well as the Bat Algorithm (BA), in terms of the computational cost, RT and PT. Due to the load balance on fog and implemented BPSO, the proposed scheme reduces the RT, PT and computational costs.

3.2 Probabilistic/Statistic Approach

The framework of the Fuzzy load balancer based on a probabilistic approach was proposed by Singh et al. [34]. According to the investigation, the 3-level design for load balancing in fog zones uses less energy since it uses fewer intervals in the fuzzy design, has less provisioning overhead, and is more responsive. The outcome of this work proves that the suggested solution is effective for various physical link types since it can recognize a wide range of traffic flows in the specified network. Another work that focuses on a probabilistic approach via game theory is proposed by [35]. The author established a fog load balancing approach to reduce the workload-balancing expenditure of a fog surrounding (NB-IoT). First, a bankruptcy game represented the NB-IoT time resource scheduling problem. The author imposes the Shapley value-based comprehensive strategy for the NB-IoT devices while solving the transmission costs within the game framework. In addition, the work also suggests greedy iterative time scheduling (GITS), a less complicated alternative to Shapley value-based scheduling. The simulation findings demonstrate that the scheme greatly lowers the average workload balancing cost compared to the standard methods.

3.3 Graph Theory

A unique graph partitioning strategy that facilitates load balancing in a distributed environment was presented by the author [36]. The proposed approach considers memory usage and throughput as partitioning criteria to select the load on each node. The method divides the graph using hot data. Finally, to evenly distribute the load in a distributed environment, vertex-cut-based dynamic graph partitioning was performed by using a vertex replication index, vertex-replicated indexes that load hot data on each node. To verify the superiority of the proposed partitioning scheme, the author compares it with the existing partitioning schemes through a variety of performance evaluations. To probe the efficacy of the proposed scheme, results were compared with existing current partitioning methods and they demonstrate the viability and superiority of the proposed scheme.

3.4 Hybrid

A hybrid technique multi-objective task scheduling optimization based on the Artificial Bee Colony Algorithm (ABC) with a Q-learning algorithm [37] is proposed as an independent task scheduling approach for cloud computing. The suggested approach seeks to overcome the constraints of concurrent concerns by optimizing scheduling and resource utilization, maximizing VM throughput, and creating a load balancing amongst VMs based on makespan, cost, and resource utilization. Performance analysis is carried out via three datasets: Random, Google Cloud Jobs (GoCJ), and Synthetic workload and was compared with existing load balancing and scheduling algorithms. The experimental findings demonstrated that the suggested approach outperforms existing techniques in terms of resource utilization, cost and makespan. Another hybrid technique was developed by Talaat et al. [38], known as an influential load balancing scheme (ELBS) intended for healthcare applications. The proposed approach achieves load balancing in fog-cloud areas through a combination of two schemes which are a real-time scheduling and caching algorithm. A complete methodology is established by introducing five modules to achieve consistent interconnections within fog nodes. The ELBS was developed using iFogSim and was implemented. Therefore, to probe the efficacy of the proposed approach, a comparison with other works of literature in load balancing has proven that the ELBS results have the lowest failure rate and average turn-around time. Therefore, the ELBS is suitable for strict-delay services such as healthcare due to its guaranteed reliable execution.

4 Recent Works in Task Offloading

IoT resources are grouped from most to least in a tiered order: cloud, fog nodes, and edge devices. Table 2 summarizes several works proposed by researchers in task offloading. The following section discusses recent works for each task offloading option.

Table 2: Summary of task offloading techniques

Paper	Key idea	Destination				Offloading fraction		Performance metrics
		D2D	D2F	F2F	F2C	Full	Partial	
Wang et al.	Using the Knapsack problem-based preallocation (PA) algorithm	/					/	Energy delay
Lin et al.	Using the convex optimization method for an optimal decision of task offloading	/				/		Energy
Jiang et al.	Real-time fog dispatcher determines the offloading target		/				/	Energy delay
Li et al.	Optimal offloading is determined by comparing the completion between the IoT device and edge server.		/				/	Energy delay task completion ratio
Hazra et al.	Energy aware offloading policy is used to deploy the scheduled tasks on suitable computing devices with Hall's theorem.		/				/	Queuing delay Energy Co$_2$ emission
Alam et al.	Machine learning-based user's location prediction and fog selection mechanism for critical IoT applications.			/		/		Delay Resource allocation
Roshan et al.	Secure task offloading framework using blockchain technology for cooperative fog computing network.			/		/		Delay security/ authentication

(Continued)

Table 2: Continued

Paper	Key idea	Destination				Offloading fraction		Performance metrics
		D2D	D2F	F2F	F2C	Full	Partial	
Zhang et al.	Fair and energy-minimized task offloading (FEMTO) algorithm Based on a fairness scheduling metric				/	/		Energy consumption Fairness level
Zhu et al.	Proposed offloading policy considers various factors, namely various offloading policy (VOP) were introduced.		/					Energy consumption Execution time
Liu et al.	Formulated a multi-objective optimization problem by finding optimal offloading probability and transmit power.		/					Energy consumption delay Payment cost
Chang et al.	Utilize queuing theory to formulate optimization problems via alternating direction method of multipliers (ADMM)-based distributed algorithm.		/					Energy consumption Execution delay
Assila et al.	Exploited the gale shapley algorithm in matching strategy coupled to caching capabilities on distributed fog computing		/			/		Energy consumption Backhaul traffic load
Chiti et al.	Pursuit the stability analysis of the outcome matching, and provide a post-matching procedure to reach a stable final matching configuration.		/					Energy consumption Task completion time

(Continued)

Table 2: Continued

Paper	Key idea	Destination				Offloading fraction		Performance metrics
		D2D	D2F	F2F	F2C	Full	Partial	
Huang et al.	Introduced joint task offloading and QoS-aware resource allocation, to optimize the offloading decisions and minimize network overhead.		/			/		Network overhead Resource block (RB) utilization
Chittaranjan et al.	Matching-theory-based efficient task offloading in IoT-fog interconnection networks target to reduce overall latency and energy consumption.		/			/		Energy consumption Completion time Execution time

4.1 Device to Device Offloading

The first device-to-device offloading was presented by Wang et al. [39] using the Knapsack problem-based PA algorithm to derive the offloading decisions. To acquire the best transmission power, the variable substitution technique is also used to recast the created nonconvex issue as convex. The author then suggests a novel alternative optimization algorithm to optimise task offloading and transmission power. Simulations demonstrate that the suggested algorithm minimises the latency and energy usage as compared to the conventional method. Another work applies the convex optimization method [40] to produce an optimal, well-structured solution for two users who can dynamically switch computing workloads using D2D offloading, hence lowering overall energy usage. To accomplish this, they jointly optimise their local computation and task exchange decisions while taking into account the recently added task causality and completion requirements. Simulation results prove that the suggested D2D cooperative computing design significantly lowers the system's energy usage when compared to previous benchmark techniques.

4.2 Device to Fog Offloading

The work done by [41–43] presented a device fog offloading approach to save device energy resources. Numerous fog devices with different applications were taken into consideration by Jiang et al. [41], where unloading one function would affect the performance of other applications. As a panacea to this problem, an energy-aware cloud-fog offloading is proposed to observe the available bandwidth and schedule queues, approximate power usage, and decide on an offloading arrangement. Following that, all offloading processes are programmed with a two-stage deadline to flexibly react to changes in the run-time network bandwidth and scheduling delays triggered by numerous devices with various duties. The decision to offload to a remote system is only made if it reduces power consumption and meets end-to-end deadlines with the available network bandwidth and scheduling queues.

Li et al. [42] suggested a framework of workload offloading for Mobile Edge Computing (MEC). The comparison of the IoT device and edge server completion yields the best offloading technique. There are 5 phases involved. The first phase calculates the completion period within the IoT device and edge server. Second phase focus on calculating energy consumption. The following phase proposes an offloading strategy. Phase four deals with subtask sorting and the final phase allocate the subtask to a specific destination. The proposed technique improves efficiency in energy consumption, delay, and task completion ratio. The study by [43] investigated the effects of task scheduling with emergency considerations and data offloading in industrial sensor networks. The author proposed an Energy Efficient data offloading (EaDO) technique. First, incoming tasks are given priorities based on emergency information using an energy-conscious scheduling methodology. Next, a multilevel feedback queue technique is applied to identify a possible scheduling sequence. Lastly, an energy-conscious offloading policy is executed to assign the planned activities to suitable computing hardware using Hall's theorem. Average queue times, energy use, and carbon dioxide (CO_2) emissions are all performance parameters that are taken into account. The suggested method reduces the energy consumption rate by 23%–30% compared to the current algorithms.

Dang et al. [44] introduced an adaptive task-offloading method called Fog Resource Aware Adaptive Task Offloading (FRATO). Fundamentally, FRATO selects an optimal offloading policy that includes a collaborative task offloading solution based on the data fragment concept. Simulation analysis shows that the FRATO substantially minimises the average latency. In order to execute inter-dependent tasks in a Sensor Mobile Edge Computing (SMEC) environment, Chakraborty et al. [45] suggest a genetic algorithm-based sustainable task offloading decision (GAME). In comparison to previous offloading schemes, simulation findings reveal a reduction in energy consumption and a delay. In order to offload the task for IoT-sensor applications, a meta-heuristic scheduler called Smart Ant Colony Optimization (SACO) is developed in this study [46]. According to numerical results, the suggested technique significantly reduces latency when compared to Round Robin (RR), throttled scheduler algorithm, modified particle swarm optimization (MPSO), and the Bee life algorithm.

4.3 Fog to Fog Offloading

Alam et al. [47] presented a task offloading scheme focusing on fog selection for mobile IoT networks. The selection of fog nodes is executed via a learning-based location awareness which applied a method for predicting users' positions based on linear regression. The suggested method takes into account the available computing and storage resources as well as the quality of service (QoS) criterion while choosing a fog node for smooth processing. An actual data set is used to calculate and measure the mobile device's location. Performance analysis of the proposed scheme shows significant improvement in latency and resource allocation as compared to baseline algorithms. In addition, loads of the fog nodes are efficiently balanced by allocating the jobs to suitable fog nodes which directly improves the overall network performance. Roshan et al. [48] developed a robust task-offloading system for shared fog-computing environments leveraging blockchain solutions, simpler to integrate different IoT applications into the fog network, even without prior trusted relation between fog nodes. With the described method, a fog node can use smart contracts to safely offload duties to a peer fog node. To stop unauthorized access to the fog, the public key structure is utilized to validate the validity of nodes with lesser reputations. In addition, the security analysis is presented to analyze the performance of the proposed framework under various attack scenarios. The proposed framework's security analysis demonstrates how non-validated fog nodes are prohibited from accepting offloading jobs.

4.4 Device to Cloud Offloading

Zhang et al. [49] presented a workload offloading according to the fairness scheduling metric for each fog node (FN). A unique FEMTO method that selects the offload FN, transmit power and offload subtask size fairly and with energy efficiency was presented based on the measure. Therefore, a compromise was reached with less energy usage and equal work offloading across many FNs. The effectiveness of the suggested approach for deciding the viability as well as minimal energy usage for the workload offloading was evaluated through numerical simulations. Simulation results also prove that the FEMTO algorithm may provide better and more reliable fairness for the FNs' energy usage compared to the Greedy Task Offloading (GTO) algorithm.

Huang et al. [50] introduced Joint task offloading and QoS-aware resource allocation, which takes into account the association between fog nodes (FNs) and Internet of Things devices (IDs), transmission, and computing resource allocation to optimize the offloading decisions. First, a framework for evaluating QoS based on AHP was developed to examine various ID categories with varying QoS criteria. Second, the author presents an algorithm for allocating RBs to IDs based on the IDs' priority, degree of satisfaction, and quality of RBs. To enhance the association among FNs and IDs, a bilateral matching game with QoS considerations is also implemented. The results of the simulation show that the suggested strategy could effectively maintain the network's loading balance, increase RB utilisation, and lower network overhead.

Swain et al. [51] presented the Matching theory-based efficient task offloading (METO) task offloading approach based on matching theory and targets to decrease overall latency and energy consumption in an IoT-fog interconnection network. The preferences of both stakeholders are generated using a hybrid criteria importance though inter-criteria correlation (CRITIC) and technique for order of preference by similarity to ideal solution (TOPSIS) based- technique since METO takes into account many criteria while making decisions. The total offloading problem is defined as a one-to-many matching game based on this rating, and the deferred acceptance algorithm (DAA) is used to provide a stable assignment. Numerous simulations show that the suggested algorithm works better than current techniques in terms of energy usage, completion time, and execution time.

Next, Low-Energy consumption in the task-offloading method is also discussed by Assila et al. [52] and their method is to reuse the cache in the local devices to lessen the burden of local computing. They offer a green solution for reducing the energy consumed within fog computing and Internet of Everything (IoE) devices by leveraging a combination of fog and caching capabilities. To handle distributed networking, processing, and storage resources, fog computing is used. IoT devices rely on new caching approaches to provide good services that necessitate a considerable amount of computational resources and a high throughput. To address the issue of the increasing number of IoT devices and the constrained resource allocation in fog computing, the author presented a many-to-one pairing game between devices and fog. The Gale Shapley Algorithm is also exploited when matching the cache with the fog nodes and local devices by a many-to-one matching game. Simulation findings show that the suggested matching approach and distributed fog computing's caching capabilities greatly outperform the conventional caching solutions. Also, the propensity of the proposed method to enhance cache attained a ratio of up to 62%, reduced average energy usage by 54% and decrease back-haul traffic load by 65%.

Chiti et al. [53] suggested a workload distribution approach to reduce power usage and completion time. Since each end device can only access a portion of the integrated computing system's fog nodes, the optimization issue is characterised as a pairing game with externalities and unfinished preference lists within tasks and computation areas. Additionally, to arrive at a stable final matching configuration, their work pursues a stable analysis of the results matching and offers a post-matching technique. Numerous computer simulations are used to generate an analytical outcome, which is then used to compare the proposed method's performance to various approaches. Finally, the suggested scheme attains a diverse type of matching depending on the nature of the computation areas.

5 Future Challenges

Various issues must be considered while planning, modelling, and implementing QoS-aware resource management fog-based IoT networks. This section describes some research opportunities and challenges that need to be addressed to enable QoS-aware solutions for the IoT-fog environment.

5.1 Energy Consumption

Along with QoS responses, energy consumption is a significant element that potential researchers should consider. The energy demands for fog servers will rise dramatically when the dynamic demand from IoT-based services keeps pace with user needs. Therefore, a significant challenge is to develop energy-and QoS-conscious approach to resource management.

5.2 Scalability

Another significant element that needs to be addressed is scalability. A number of fog computing methods need to be able to operate at a wide-ranging scale. There is no guarantee that all nodes, devices, and related processes will function well in a large-scale environment even if these methodologies have been validated on small sizes. Also, with the development of IoT-based applications, it is clear that there is a need to develop resource management approaches to manage scalable fog networks. Therefore, it is a viable question for future research.

5.3 Security and Privacy

With the implementation of resource management techniques, security and privacy are also important areas that need to be addressed. The concept of fog computing was adapted from the regular cloud processing model, and the mechanisms employed within the fog layer are security-vulnerable. Hence, to access the service, the user must be authenticated by a reliable security system. However, designing new authentication protocols and access control policies requires updates, which is challenging to implement in a fog environment.

5.4 Multi-Target Optimization

Harnessing multi-objective resource management approaches is a viable means of establishing a method that may improve some factors that facilitate the smooth operation of a fog network. However, there is no specific mechanism for defining most QoS factors to determine resource management in a fog network. For instance, some algorithms overlook factors like scalability, dependability, and security in favour of concentrating on a small number of QoS criteria like response time, resource utilization, bandwidth and energy. Also, determining how to trade off different characteristics may be a significant unresolved issue. As a result, multi-target optimization in load-balancing must be extended to consider different QoS metrics.

5.5 Interoperability

Heterogeneity in fog computing environments due to the variety of platforms used, different architectures applied and varied infrastructure handling contributes to interoperability issues. This makes interoperability a major obstacle to the successful deployment of resource management in fog computing.

6 Conclusions

This paper reviews resource management techniques and categorize them into load balancing, task offloading, resource scheduling, resource provisioning, and node placement. It provides an overview of recent works that enable efficient resource management, with QoS improvement as a primary goal. The paper focuses on two main techniques: load balancing and task offloading. In this systematic review, important areas are highlighted, including the methods applied, performance metrics and outcomes of each technique, which also answered the formulated research questions. Finally, several unresolved research issues and potential future directions in resource management strategies are also highlighted. Future work will involve integrating an artificial intelligence-based model into the fog coordination node that can categorise various Internet of Things services so that actual resource requirements can be determined.

Acknowledgement: This study was funded by the Fundamental Research Grant Scheme's budget: (FRGS), 600-IRMI/FRGS 5/3(327/2019) under Kementerian Pendidikan Malaysia and Universiti Teknologi MARA.

Funding Statement: The project was funded under Grant of the Fundamental Research Grant Scheme Malaysia Higher Education: FRGS/1/2019/ICT03/UITM/03/1.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. K. Apat, B. Sahoo, P. Maiti and P. Patel, "Review on QoS aware resource management in fog computing environment," in *2020 IEEE Int. Symp. on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*, Gunupur, India, pp. 1–6, 2020.

[2] A. Shakarami, H. Shakarami, M. Ghobaei-Arani, E. Nikougoftar and M. Faraji-Mehmandar, "Resource provisioning in edge/fog computing: A comprehensive and systematic review," *Journal of Systems Architecture*, vol. 122, no. 102362, pp. 1–23, 2022.

[3] J. Seth and P. Nand, "Fog assisted-IoT based health monitoring system," in *2021 Fourth Int. Conf. on Computational Intelligence and Communication Technologies (CCICT)*, Sonepat, India, pp. 318–323, 2021.

[4] H. D. Karatza, "Keynote speech 1: Leveraging cloud and fog computing for real-time applications: Resource allocation and scheduling issues," *2021 Second International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*, Tartu, Estonia, pp. 1, 2021.

[5] D. Griffin, T. K. Phan, E. Maini, M. Rio and P. Simoens, "On the feasibility of using current data centre infrastructure for latency-sensitive applications," *IEEE Transactions on Cloud Computing*, vol. 8, no. 3, pp. 875–888, 2020.

[6] H. Sabireen and V. Neelanarayanan, "A review on fog computing: Architecture, fog with IoT, algorithms and research challenges," *ICT Express*, vol. 7, no. 2, pp. 162–176, 2021.

[7] A. Alzeyadi and N. Farzaneh, "A novel energy-aware scheduling and load-balancing technique based on fog computing," in *2019 9th Int. Conf. on Computer and Knowledge Engineering, ICCKE 2019*, Mashhad, Iran, pp. 104–109, 2019.

[8] M. Kaur and R. Aron, "A systematic study of load balancing approaches in the fog computing environment," *The Journal of Supercomputing*, vol. 77, no. 8, pp. 9202–9247, 2021.

[9] A. Chandak and N. K. Ray, "A review of load balancing in fog computing," in *2019 Int. Conf. on Information Technology (ICIT)*, Saratov, Russia, pp. 460–465, 2019.

[10] S. Batra, D. Anand and A. Singh, "A brief overview of load balancing techniques in fog computing environment," in *2022 6th Int. Conf. on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, pp. 886–891, 2022.

[11] N. Kumari, A. Yadav and P. K. Jana, "Task offloading in fog computing: A survey of algorithms and optimization techniques," *Computer Networks*, vol. 214, no. 10, pp. 1–24, 2022.

[12] L. Lin, X. Liao, H. Jin and P. Li, "Computation offloading toward edge computing," in *Proc. of the IEEE*, Torino, Italy, vol. 107, no. 8, pp. 1584–1607, 2019.

[13] J. Wang, J. Pan, F. Esposito, P. Calyam, Z. Yang *et al.,* "Edge cloud offloading algorithms: Issues, methods, and perspectives," *ACM Computing Surveys*, vol. 52, no. 1, pp. 1–23, 2020.

[14] K. Asmi, S. Dilek, S. Tosun and S. Ozdemir, "A survey on computation offloading and service placement in fog computing-based IoT," *Journal of Supercomputing*, vol. 78, no. 2, pp. 1983–2014, 2022.

[15] X. An, R. Fan, H. Hu, N. Zhang, S. Atapattu *et al.,* "Joint task offloading and resource allocation for IoT edge computing with sequential task dependency," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 16546–16561, 2022.

[16] J. Ren, J. Li, H. Liu and T. Qin, "Task offloading strategy with emergency handling and blockchain security in SDN-empowered and fog-assisted healthcare IoT," *Tsinghua Science and Technology*, vol. 27, no. 4, pp. 760–776, 2022.

[17] M. Qin, Z. Jing, T. Yang, W. Xu, Q. Yang *et al.,* "Service-oriented energy-latency tradeoff for IoT task partial offloading in MEC-enhanced multiple radio access technologies (multi-RAT) networks," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1896–1907, 2021.

[18] M. Aazam, S. U. Islam, S. T. Lone and A. Abbas, "Cloud of things (CoT): Cloud-fog-IoT task offloading for sustainable internet of things," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 1, pp. 87–98, 2022.

[19] Y. Wang, X. Qi, X. Lin and X. Wang, "Computing offloading-based task scheduling for space-based cloud-fog networks," in *2021 2nd Int. Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*, Shanghai, China, pp. 266–270, 2021.

[20] T. Alfakih, M. M. Hassan, A. Gumaei, C. Savaglio and G. Fortino, "Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on SARSA," *IEEE Access*, vol. 8, pp. 54074–54084, 2020.

[21] H. A. Alharbi, B. A. Yosuf, M. Aldossary, J. Almutairi and J. M. H. Elmirghani, "Energy efficient UAV-based service offloading over cloud-fog architectures," *IEEE Access*, vol. 10, pp. 89598–89613, 2022.

[22] T. Yang, H. Feng, S. Gao, Z. Jiang, M. Qin *et al.,* "Two-stage offloading optimization for energy–latency tradeoff with mobile edge computing in maritime internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5954–5963, 2020.

[23] R. Yadav, W. Zhang, O. Kaiwartya, H. Song and S. Yu, "Energy-latency tradeoff for dynamic computation offloading in vehicular fog computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14198–14211, 2020.

[24] H. K. Apat, K. Bhaisare, B. Sahoo and P. Maiti, "Energy efficient resource management in fog computing supported medical cyber-physical system," in *2020 Int. Conf. on Computer Science, Engineering and Applications (ICCSEA)*, Gunupur, India, pp. 1–6, 2020.

[25] J. Singh, J. Warraich and P. Singh, "A survey on load balancing techniques in fog computing," in *2021 Int. Conf. on Computing Sciences (ICCS)*, Phagwara, India, pp. 47–52, 2021.

[26] I. Martinez, A. S. Hafid and A. Jarray, "Design, resource management, and evaluation of fog computing systems: A survey," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2494–2516, 2021.

[27] R. Archana and P. M. Kumar, "Utilization of fog computing in task scheduling and offloading: Modern growth and future challenges," in *2022 Int. Conf. on Electronic Systems and Intelligent Computing (ICESIC)*, Chennai, India, pp. 23–28, 2022.

[28] X. Xu, Q. Liu, L. Qi, Y. Yuan, W. Dou *et al.,* "A heuristic virtual machine scheduling method for load balancing in fog-cloud computing," in *2018 IEEE 4th Int. Conf. on Big Data Security on Cloud (BigDataSecurity)*, Omaha, Nebraska, USA, pp. 83–88, 2018.

[29] F. Banaie, M. Hossein Yaghmaee, S. A. Hosseini and F. Tashtarian, "Load-balancing algorithm for multiple gateways in fog-based internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7043–7053, 2020.

[30] M. Zahid, N. Javaid, K. Ansar, K. Hassan, K. Khan *et al.,* "Hill climbing load balancing algorithm on fog computing," in *Proc. of the 13th Int. Conf. on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2018)*, Taichung, Taiwan, pp. 238–251, 2018.

[31] E. Karypiadis, A. Nikolakopoulos, A. Marinakis, V. Moulos and T. Varvarigou, "SCAL-E: An auto scaling agent for optimum big data load balancing in kubernetes environments," in *2022 Int. Conf. on Computer, Information and Telecommunication Systems (CITS)*, Athens, Greece, pp. 1–5, 2022.

[32] I. M. Jabour and H. Al-Libawy, "An optimized approach for efficient-power and low-latency fog environment based on the PSO algorithm," in *2021 2nd Information Technology to Enhance e-Learning and Other Application (IT-ELA)*, Baghdad, Iraq, pp. 52–57, 2021.

[33] A. A. Butt, S. Khan, T. Ashfaq, S. Javaid, N. A. Sattar *et al.,* "A cloud and fog based architecture for energy management of smart city by using meta-heuristic techniques," in *2019 15th Int. Wireless Communications & Mobile Computing Conf. (IWCMC)*, Tangier, Morocco, pp. 1588–1593, 2019.

[34] S. P. Singh, A. Sharma and R. Kumar, "Design and exploration of load balancers for fog computing using fuzzy logic," *Simulation Modelling Practice and Theory*, vol. 101, no. 10, pp. 1–29, 2020.

[35] S. F. Abedin, A. K. Bairagi, M. S. Munir, N. H. Tran and C. S. Hong, "Fog load balancing for massive machine type communications: A game and transport theoretic approach," *IEEE Access*, vol. 7, pp. 4204–4218, 2019.

[36] D. Choi, J. Han, J. Lim, J. Han, K. Bok *et al.,* "Dynamic graph partitioning scheme for supporting load balancing in distributed graph environments," *IEEE Access*, vol. 9, pp. 65254–65265, 2021.

[37] B. Kruekaew and W. Kimpan, "Multi-objective task scheduling optimization for load balancing in cloud computing environment using hybrid artificial bee colony algorithm with reinforcement learning," *IEEE Access*, vol. 10, pp. 17803–17818, 2022.

[38] F. M. Talaat, S. H. Ali, A. I. Saleh and H. A. Ali, "Effective load balancing strategy (ELBS) for real-time fog computing environment using fuzzy and probabilistic neural networks," *Journal of Network and Systems Management*, vol. 27, no. 4, pp. 883–929, 2019.

[39] H. Wang, Z. Lin and T. Lv, "Energy and delay minimization of partial computing offloading for D2D-Assisted MEC Systems," in *2021 IEEE Wireless Communications and Networking Conf.*, Nanjing, China, pp. 1–6, 2021.

[40] Q. Lin, F. Wang and J. Xu, "Optimal task offloading scheduling for energy efficient D2D cooperative computing," *IEEE Communications Letters*, vol. 23, no. 10, pp. 1816–1820, 2019.

[41] Y. L. Jiang, Y. S. Chen, S. W. Yang and C. H. Wu, "Energy-efficient task offloading for time-sensitive applications in fog computing," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2930–2941, 2019.

[42] J. Li, M. Dai and Z. Su, "Energy-aware task offloading in the internet of things," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 112–117, 2020.

[43] A. Hazra, M. Adhikari, T. Amgoth and S. N. Srirama, "Fog computing for energy-efficient data offloading of IoT applications in industrial sensor networks," *IEEE Sensors Journal*, vol. 22, no. 9, pp. 8663–8671, 2022.

[44] H. T. Dang and D. S. Kim, "FRATO: Fog resource based adaptive task offloading for delay-minimizing IoT service provisioning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 10, pp. 2491–2508, 2021.

[45] S. Chakraborty and K. Mazumdar, "Sustainable task offloading decision using genetic algorithm in sensor mobile edge computing," *Journal of King Saud University—Computer and Information Sciences*, vol. 34, no. 4, pp. 1552–1568, 2022.

[46] A. Kishor and C. Chakarbarty, "Task offloading in fog computing for using smart ant colony optimization," *Wireless Personal Communications*, vol. 127, no. 2, pp. 1683–1704, 2022.

[47] M. Alam, N. Ahmed, R. Matam and F. A. Barbhuiya, "L3Fog: Fog node selection and task offloading framework for mobile IoT," in *IEEE INFOCOM 2022—IEEE Conf. on Computer Communications Workshops*, New York, NY, USA, pp. 1–6, 2022.

[48] R. Roshan, R. Matam, M. Mukherjee, J. Lloret and S. Tripathy, "A secure task-offloading framework for cooperative fog computing environment," in *GLOBECOM 2020–2020 IEEE Global Communications Conf.*, Taipei, Taiwan, pp. 1–6, 2020.

[49] G. Zhang, F. Shen, Z. Liu, Y. Yang, K. Wang *et al.,* "FEMTO: Fair and energy-minimized task offloading for fog-enabled IoT networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4388–4400, 2019.

[50] X. Huang, Y. Cui, Q. Chen and J. Zhang, "Joint task offloading and QoS-aware resource allocation in fog-enabled IoT networks," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7194–7206, 2020.

[51] C. Swain, M. N. Sahoo, A. Satpathy, K. Muhammad, S. Bakshi *et al.,* "METO: Matching-theory-based efficient task offloading in IoT-fog interconnection networks," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12705–12715, 2021.

[52] B. Assila, A. Kobbane, A. Walid and M. El Koutbi, "Achieving low-energy consumption in fog computing environment: A matching game approach," in *2018 19th IEEE Mediterranean Electrotechnical Conf.*, Marrakesh, Cyprus, pp. 213–218, 2018.

[53] F. Chiti, R. Fantacci and B. Picano, "A matching game for tasks offloading in integrated edge-fog computing systems," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, pp. 1–14, 2020.