Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

Review paper

Service and Energy Management in Fog Computing: A Taxonomy Approaches, and Future Directions

***S. M. Hashemi* [1], *A. Sahafi* [2,*], *A. M. Rahmani* [3], *M. Bohlouli* [4,5,6]**

[1]*Department of Computer Engineering, Qeshm branch, Islamic Azad university, Qeshm, Iran.*

[2]*Department of Computer Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran.*

[3]*Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.*

[4]*Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences, Zanjan, Iran.*

[5]*Research and Innovation Department, Petanux GmbH, Bonn, Germany.*

[6]*Research Center for Basic Sciences and Modern Technologies (RBST), Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran.*

Article Info

Article History:
Received 11 February 2023
Reviewed 24 March 2023
Revised 22 May 2023
Accepted 27 May 2023

Keywords:
Fog computing
Internet of Things (IoT)
Systematic literature Review (SLR)
Service management
Energy management

*Corresponding Author's Email Address: *sahafi@iau.ac.ir*

Abstract

Background and Objectives: Today, the increased number of Internet-connected smart devices require powerful computer processing servers such as cloud and fog and necessitate fulfilling requests and services more than ever before. The geographical distance of IoT devices to fog and cloud servers have turned issues such as delay and energy consumption into major challenges. However, fog computing technology has emerged as a promising technology in this field.

Methods: In this paper, service/energy management approaches are generally surveyed. Then, we explain our motivation for the systematic literature review procedure (SLR) and how to select the related works.

Results: This paper introduces four domains of service management and energy management, including Architecture, Resource Management, Scheduling management, and Service Management. Scheduling management **has** been used in 38% of the papers. Therefore, they have the highest service management and energy management. Also, Resource Management is the second domain that has been able to attract about 26% of the papers in service management and energy management.

Conclusion: About 81% of the fog computing papers simulated their approaches, and the others implemented their schemes using a testbed in the real environment. Furthermore, 30% of the papers presented an architecture or framework for their research, along with their research. In this systematic literature review, papers have been extracted from five valid databases, including IEEE Xplore, Wiley, Science Direct (Elsevier), Springer Link, and Taylor & Francis, from 2013 to 2022. We obtained 1596 papers related to the discussed subject. We filtered them and achieved 47 distinct studies. In the following, we analyze and discuss these studies; then we review the parameters of service quality in the papers, and ultimately, we present the benefits, drawbacks, and innovations of each study.

Introduction

The Internet of Things (IoT) [1], [2] advancement has resulted in the urge of having fast and robust processing power for devices in important applications. In addition, data processing in the centralized cloud cannot meet requirements in such an environment. There are characteristics such as health monitoring and emergency response that should consider data delay and the data volume transferred to the cloud [3], [4]. Cisco proposed a novel paradigm known as fog computing (FC). This is an extended version of cloud computing [5], providing services toward the network edge. FC [6], [7] is a model for the management of a virtual and distributed environment to provide network services and computations between the cloud data centers [4], [8] and sensors [9].

Similar to Cloud [5] Fog provides data, services of computing, storage, and software for end-users and customers [10]. However, the Fog can be better than the cloud according to the following reasons:

- Closer topological proximity to end-users
- Geographical and large-scale distribution
- Support for mobility

These features will solve many issues to a large extent. In this regard, fog computing is generalized toward the network edge [11], [12] to reduce delay and congestion on the network and resolves several challenges such as high delay, low capacity, and network defects [4] Fig. 1 shows the cloud, Fog, and IoT services.

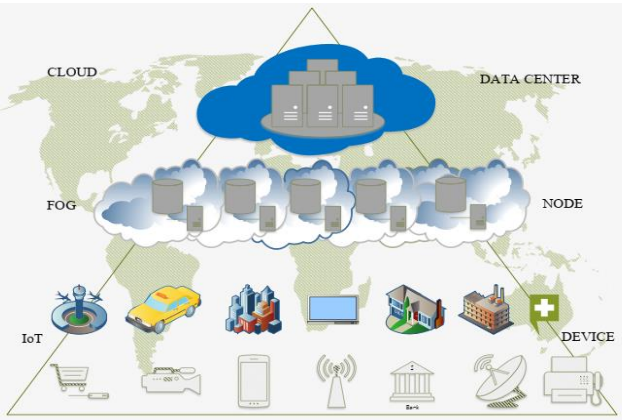

Fig. 1: Cloud, Fog and, IoT services.

There are several studies on specialized approaches/protocols. However, there is no review paper on service and energy management in Fog/edge Computing. The main purpose of this study is to discuss different service and energy management approaches in Fog/edge Computing. Some review studies usually discuss a separate topic, for example, energy approaches in [13] or service approaches/resource management in [14] and this topic has not been explored in a comprehensive study yet. In this SLR method, we investigate and discuss recent approaches/protocols in fog computing. Table 1 presents some studies in different fields of fog computing.

Table 1: Related investigations in fog computing service and energy management approach

Reference	Main Topic	Type	year of Publication	year Covered
[15]	Orchestration in FC	SLR	2022	**2015-2021**
[16]	simulation frameworks and research directions in FC	SLR	2021	**2015-2020**
[17]	Task scheduling approaches in FC	SLR	2020	**2012-2020**
[18]	Resource management approaches in FC	SLR	2019	**2014-2018**
[19]	Fundamental, network applications, and research challenges in FC	SURVEY	2018	**2014-2018**
[20]	Challenges in fog computing	SLR	2017	**2014-2017**
[21]	FC and its role in the IOT	SURVEY	2016	-

Related Studies

Researchers in [15] presented an SLR paper focusing on orchestration in fog computing. Their research period was from 2012 to 2021. They selected articles from 4 valid databases. By reviewing the literature of published articles, they limited their research to 50 articles written between 2015 and 2021. Their study consisted of 5 questions. Given the novelty of the topic of orchestration in fog computing, they presented a suitable conclusion for researchers, including the benefits, challenges, and future tasks.

Researchers in [16] used a systematic method to collect and identify articles on fog computing and simulation tools. Their study period was from 2015 to 2020. They selected 3 types of information sources, including articles from journals, conferences, and dissertations. Their main goal was to introduce fog computing simulation tools and the challenge of developing such software applications.

In [17] the authors discussed the fog computing challenges and their open problems. They consider problems such as resource constraints, heterogeneous resources, dynamic and unpredictable nature of the environment using resource management. Despite the resource management importance, there is no regular, comprehensive, and detailed review in the field of resource management methods in fog computing. They presented a systematic literature review (SLR) about approaches to resource management in fog computing based on a classical classification to identify advanced mechanisms. The classification has 6 primary areas, including programs, resource scheduling, performing tasks, resource allocation, resource supply, and load balancing. A Comparison of resource management approaches is done regarding essential factors, like performance criteria, techniques used, case studies, evaluation instruments, and strengths and drawbacks.

In [18], the authors presented an SLR-based scheme for examining the scheduling methods in fog computing. In their work, they presented scheduling algorithms and remarked on open issues. This study uses search methods to examine 100 papers published between 2012 and 2020. Then, they reached 36 final papers on scheduling approaches. In an SLR-based scheme, analysis of all available studies is not allowed. Thus, we ignored non-English, old, unreliable, and inaccessible works. The present review paper would be helpful for researchers for understanding the various dimensions of the reviewed subject properly. Nevertheless, it cannot consider all studies in this field because their number is very high and continuously increases. For future research, they want to propose a scheduling algorithm that can support dynamic environments and consider some evaluation criteria, like security and availability since most present algorithms only consider delay and cost as evaluation criteria.

In [19] the authors introduced a basic and general overview of fog computing architecture. They reviewed various resource and service allocation methods for addressing several important subjects, like delay, energy consumption, and bandwidth in fog computing. In comparison with other surveys, the present work presents an overview of advanced network programs and the main aspects of designing these networks. Moreover, the current research presents the Fog computing procedure and related challenges. This review paper discusses various architectures and determines the main research challenges. Fog computing can be used in many network applications because it can be considered the new version of predictive computations. Finally, they introduce some open research challenges and the essential design principles in their paper.

In [20] the authors considered the fog computing model as an option for IoT applications. This research examines concerns or challenges associated with fog computing for IoT. Researchers aim to address these problems associated with fog computing for IoT applications using a systematic literature review (SLR). They use the SLR scheme and apply search criteria to investigate an initial set including 439 papers from 2014 to 2017 and identify and review 17 studies related to this field.

These papers were organized into four main categories based on the challenges. This paper can help physicians and researchers understand concerns about fog computing and provide several useful views for future research directions. The scope of this paper is limited to the number of papers reviewed from the database. Based on the results of this study, they want to study precisely fog data dominance and business optimization techniques to use services of the IoT applications.

In [21] the authors presented a survey paper about the perspective, key features of fog computing, and new services and programs on the network edge. They stated that Fog must be a rich enough integrated platform to provide these new services. Also, it allows us to develop emerging services and new programs. In this paper, based on the fog computing characteristics, they introduced it as a suitable basis for supporting services and important IoT applications, smart networks, connected vehicles, smart cities, and generally wireless actuators and sensors. Researchers referred to three critical subjects in its research:

1) Fog architecture for massive computational infrastructure, storage, and communication devices.
2) Orchestration and management of fog nodes.
3) Services and programs supported by the Fog.

Fog Architecture and Its Characteristics

Fog computing can change how to provide services for customers for meeting IoT requirements. The fog infrastructure and services are extended both in the range of network and the cloud-to-Things continuum for allowing computational resources, which are located anywhere in this continuum, such as edge, cloud, or things for collecting these distributed resources and supporting programs [21] It is a high potential for task transfer from the cloud to the fog service providers, which are close to data sources or end-users. This can decrease the delay and bandwidth needed for data transmission to the cloud [22].

Fog computing is a new computational platform extending traditional cloud computing and services toward the network edge, providing communication, computation, storage, control, and service capabilities at the network edge.

The difference between the decentralized design and other common computational models is in terms of architecture. Fog computing means an integrated network concept that it stretches from the outer edges, which produce data, to where it is ultimately stored, whether in the cloud or the customer data center [23]

Accordingly, fog-computing framework development allows organizations to have more choices for data processing wherever it is more appropriate. In some applications, it is necessary to process data quickly. For example, when using a product connected device should immediately respond to the incident [7].

Fog computing can communicate with devices and analyze data at a low time.

This architecture can reduce the required bandwidth compared to when data should be sent to a data center or cloud. Therefore, it can also be applied in scenarios, in which there is no bandwidth for transferring data, so that information is processed somewhere close to its production location. As another benefit, users can add security features to a fog network. This can perform by dividing network traffic to the virtual firewall [24]. In the following, we present some of the strengths and drawbacks of fog computing.

A. Strengths of Fog Computing

- Reducing data transferred to the cloud
- Saving bandwidth
- Lowering the response time
- Increasing security through nearing data to the edge
- Supporting mobility

B. Drawbacks of Fog Computing

- Requiring hardware and more cost
- Requiring continuous access to the fog equipment

SLR Methodology

This section provides a summary of the SLR procedure to recognize, analyze and summarize the literature on a particular theme called SLR. The SLR attempts to find the original study to answer one or more questions [25], [26]. Through using a systematic literature review, the essential questions that can be asked about the research will be performed by considering the alternatives of the critical essential components. Then, by dividing the research into some main groups (domains) into the problem-solving approaches, the subsequent exploration string is defined [27], [27]:

(("service" OR "fog computing" OR "energy consumption" OR "energy saving" OR "energy-efficient" OR "energy management") AND (IoT OR Internet of Things)).

Fig. 2 shows the total papers selected from 5 valid databases, including IEEE, Wiley, Science Direct (Elsevier), Springer, and Taylor & Francis, based on the publisher and the number of papers published from 2013 to 2022. According to these figures, IEEE with 280 papers, about 18%, Springer with 258 papers, about 16%, Elsevier with 383 papers, about 24%, Wiley with 579 papers, about 36%, and Taylor & Francis with 96 papers, about 6%, were selected.

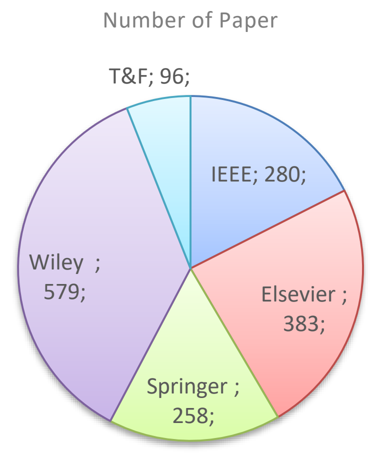

Fig. 2: A general comparison of the paper published by publishers.

A. Survey Goals and Research Questions

The paper presented in the SLR method, which is an evidence-based software engineering (EBSE), will answer the following Review Questions (RQ) to achieve the goals of this research. The objective of this chapter is to name the most important issues and difficulties in service, energy efficiency, energy-saving, and energy management in the fog computing field. This research is to direct the subsequent Research and Analytical Questions:

RQ1: Which scope and primary contexts are classified in service and Energy management in fog computing?

RQ2: What QOS parameters are utilized for evaluating the service and Energy management in fog computing?

RQ3: What assessment situations are utilized for the estimation of the service and Energy management in fog computing?

RQ4: What used tools for service and Energy management in fog computing?

RQ5: What is the significance of service and Energy management in fog computing?

RQ6: Which problems, future research directions, and challenges are identified concerning service and Energy management for future trends in fog computing?

These procedures can lead to comprehensive responses within the domain of this paper.

B. Search Query and Database Selection

The field of research is determined by selecting the most commonly used words to prepare our topic. Henceforth, seven keywords have been chosen, including "Service", "Fog", "Fog Computing", "Energy Consumption", "energy saving", "Energy Efficient" and "Energy Management". After various phases and using the results of our primary investigation as a pilot to analyze the coverage of the results, the inquiry is characterized. To be specific, the query string is developed to include more keywords because the research in our model is not recovered by the essential inquiry, for instance, "IoT" OR "Internet of Things". To expand the domain of practical research, the search keywords and strings are just applied to the titles. The search is accomplished in October 2022, with a specified time range from 2013 until 2022.

C. Selection Criteria

The inclusion/exclusion scale for the final studies is applied after providing the analytical questions. Only the indexed ISI journal articles were investigated to limit the number of published papers. These articles are peer-reviewed papers on service management methods in the IoT field. For examining and addressing the referenced questions, 47 peer-reviewed papers are selected—these journal articles are presented in detail in next Sections. The selection ideology and assessment flowchart designed for the studies is shown in Fig. 3.

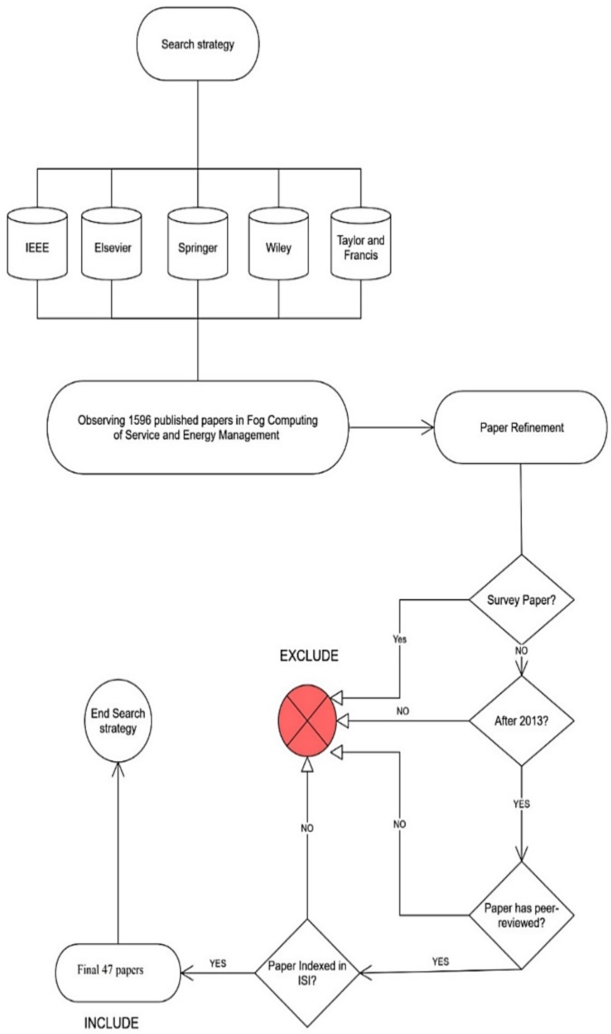

Fig. 3: Filtering approach of papers in this research.

There is some exclusion in our research, for example, short papers, low-quality studies, and non-peer-reviewed research (like predatory journals). The book chapters and white papers will be ignored because there is no research-based conversation and scientific data.

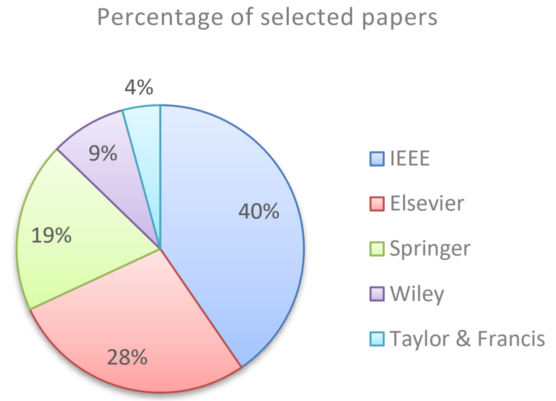

Fig. 4: The final percentage of selected papers based on the publisher in this research.

In this study, Fig. 4 displays the final papers selected from 5 valid databases, including IEEE, Wiley, Science Direct (Elsevier), Springer, and Taylor & Francis. Based on this figure, IEEE with 19 papers, about 40%, Springer with 9 papers, about 19 %, Elsevier with 13 papers approximately 28 %, Wiley with 4 papers, about 9%, and Taylor & Francis with 2 papers, about 4% were selected.

Service and Energy Management and Organization of the Study

IT management in today's environment is complex. cloud computing and fog computing introduces new path within IT and related organizations and industries [29]. The cloud computing can provide many services, in any case, how to deal with these services and ensure the quality of service has become one of the main Elements of its expansion. cloud computing services can be joined into complicated services and applications. cloud computing has been one of the challenges in improving cloud computing services for users and customers, and how to deal with these services and ensure the quality of (QoS) cloud services [30]. On the other hand, the worldview of fog computing has risen to be supplementary to cloud computing to simplify latency-critical, and applications with centralized bandwidth. In most cases, fog computing is deployed near the IoT devices/sensors and expands the cloud-based computing systems, storage accumulation, and facilities [31]. fog computing can be utilized as a viable stage for preparing power management as a service for various networking frameworks [31]. A fog service is a method for conveying quantity to clients by simplifying the results that clients need to accomplish. The services will deliver value to customers without the ownership of specific costs and risks [29].

One of the common goals of the research community, vendors, and suppliers is to design self-matchable solutions that can react to unpredictable workload oscillation and change the utility principles [32] The aspect of cloud computing services has created a noteworthy pattern of associations to make decisions about these services [33] Along these lines, the client can get crops with minimum price, high profitability, and business proficiency in a cloud environment [34] Because of the nature of fog Computing, which is distributed and has a diverse conditions, resource allocation is a significant issue. Many challenges need to be addressed to expand profitability and distribute appropriate asset tasks [35] In the real world, there is a limited resources and intensive and centralized computing applications to ensure a good experience [36]. The golden solution for this situation is task scheduling in distributed computing systems created by fog computing environments [37], [38]. The new advances in technology, cost, and scale of features have empowered us to manufacture computing devices with less power and performance than in the past [39] With different complicated digital physical vitality, the energy management frameworks should be actualized to be able to productively Monitor and deal with the task. To implement the energy management framework in the system, intelligence, interoperability, adaptability, and versatility are required [40].

Task scheduling will cause an energy consumption that is similarly performed between the poles of the system, and the organization of load network circulation happens consistently to extend the system lifetime [41] cloud suppliers need to address various key difficulties, such as finding some kind of harmony between the effectiveness of ideal vitality and the fulfillment of expanding requests and high-efficiency outlooks for clients [42].

The task scheduling in fog computing depends on whether there is an attachment between the scheduled tasks. It tends to be partitioned into two task scheduling: related and independent scheduling [43].

In this section, first, the presented 47 articles on Service and Energy Management are summarized. The next subdivision shows the various kinds of research in Service and Energy Management. Likewise, the various investigations will be thought about in various directions, for example, Main topic, Strong point of research, Deficiency of research, and new discovery. Fig. 5 shows the taxonomy of Service and Energy Management.

A. Architecture

In this section, we review two common categories in the Architecture section.

Design: In [44] the authors suggested an energy-efficient cross-layer-sensing clustering method (ECCM). Their algorithm applies the sensing-event-driven mechanism for putting fog nodes on the sensing layer and creating a robust virtual control node. In sensor networks, the cluster-based routing scheme is loaded on the fog layer, and the fog calculations use event nodes to achieve distributed clustering. Then, the optimized data aggregation routing has been made. Ultimately, the particle swarm optimization algorithm (PSO) is used to optimize the routing protocol by selecting an optimal set of nodes as cluster heads. The results of the proposed scheme show that it has the capability of optimizing the data aggregation process and improving network energy consumption and performance.

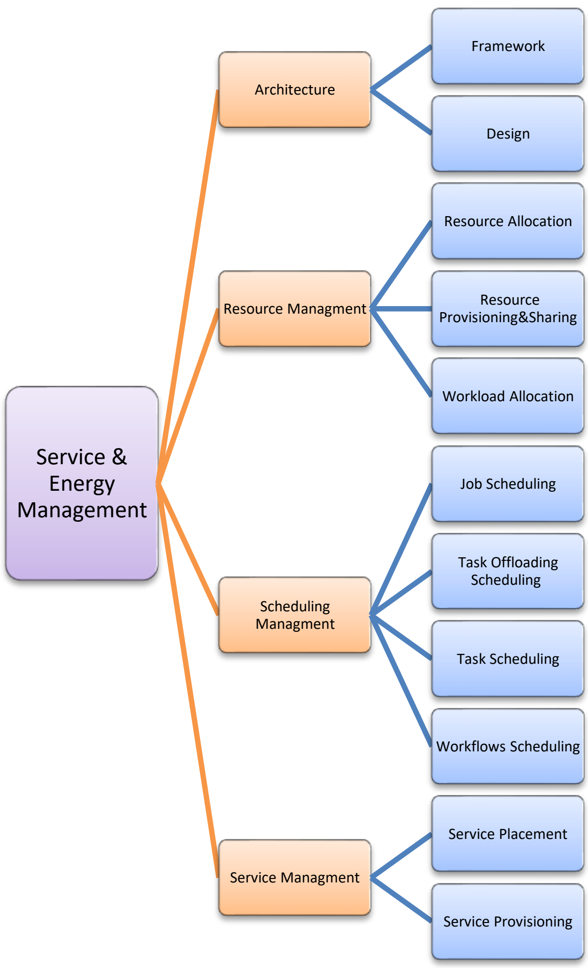

Fig. 5: The taxonomy of Service and Energy Management.

In [45] the energy efficiency of IoT resources was discussed as a critical subject in their research, and they proposed a low-consumption architecture for the IoT. It includes 3 layers: information processing, sensing and control, and presentation. In this architecture, sensors' sleep intervals are anticipated based on the residual energy of the battery and their previous use history. When the sensor nodes are in the sleep state, it is possible to use the predicted amount for strengthening the use of cloud resources by previsioning the allocated resources. This mechanism provides the optimal use of all IoT resources. Experiments show that sensor nodes can save a significant amount of energy, and they can improve the use of cloud resources. The main characteristic of their model is the energy-information exchange between two layers. The proposed PA design is compared with EGF, SoT, OGL, and ECH. It shows that PA can reduce energy consumption compared to other schemes. Researchers in [46] conducted a study on an IPv6-based standard in fog computing networks aimed at improving scalability, latency, and performance, and a three-tier architecture. They designed a scheduling algorithm with 6TiSCH technology in fog computing, ensuring high reliability and low latency for emergency data transmission. They proposed a smart local discharge system to improve energy efficiency by rank-based Q-learning algorithm and to reduce latency by the fuzzy Bayesian learning method. The simulation results of their proposed design indicate its better results than the DIVA Priority and DeTAS methods in terms of energy consumption, latency, and other parameters.

Framework: In [47] the authors offered a 3-layer cloud-fog computing architecture to manage energy in adjustable micro-networks based on the dynamic thermal rating (DLR) constraints. It is possible to improve flexibility, reliability, and stability in power networks by decomposing large-scale networks into micro-networks. The problem is proposed as a mixed-integer linear optimization problem according to DLR constraints. The proposed scheme is compared with the backward-forward sweep scheme. The simulation results show that the proposed scheme has high performance to satisfy load requests and their constraints at any time, as well as the effective distribution of power in the network. In [48] the authors suggested a framework that calculates the tidal trust for the nodes, and based on prespecified values, malicious nodes are identified. The presented framework is assessed based on several IoT and FE devices. In the proposed framework, the rank and trust of IoT devices are dependent on the communication behavior of nodes. It is obtained via the security mechanism presented in this paper, which is more efficient than other methods. Identifying malicious nodes in early steps can improve the network performance in terms of efficient use of network congestion, network resources, packet loss rate (PLR), network power, and end-to-end delay.

B. Resource Management

In this section, we review three common categories regarding the studies conducted on resource management. The approaches are presented in Figure 5 as follows:

Resource allocation: This method is used to estimate resources or load resources to optimize them about the most appropriate resource.

Resource provisioning: This refers to the time of provisioning resources in the cloud and fog, which occurs by increasing and decreasing the duration as well as the time of providing/depleting real resources.

Resource sharing: Resource sharing is an approach to implementation and cooperation in the surface layers of the cloud or fog. These strategies are presented for sharing information, resources, and IoT programs.

Workload allocation: Transferring a large volume of IoT data to the cloud causes a long delay in workload processing, in which case workload allocation can be used for workload distribution according to energy and delay criteria.

Resource Allocation

In [49] the authors proposed a suitable scheduling algorithm with the 3-layer fog-to-cloud architecture for providing high quality and optimally using the fog-to-cloud sources. It has a successful performance in terms of service failure probability and delays. They enhanced the 3-layer fog-to-cloud architecture using the proposed design to decrease the delay in the data transmission process for delay-sensitive applications. Their simulation results confirm the performance and effectiveness of the scheduling algorithm proposed in this work. The proposed method outperforms existing algorithms such as the conventional cloud scheduling algorithm, the first-fit algorithm, and the random scheduling algorithm, but it does not consider energy consumption.

In [50] the authors addressed the efficient resource allocation problem in the fog environment, which is a fundamental problem in fog computing. They described how to allocate resources and how to put them in the virtual machine in the single-fog computing environment. For evaluating the presented algorithm in the fog environment, the architecture and the efficient resource allocation algorithm are executed in the CloudSim tool. According to the results, the presented method can allocate resources using an optimal method compared to the default resource allocation strategy. Their proposed scheme has a lower total processing time than other methods. The experiments show that the proposed framework improves the quality of a fog environment.

In [51] the authors discussed the joint computation and the communication resource allocation problem for the user's fog computing with the mixed task. They offered a mixed-task model for different types of computational tasks. This model supports binary loading and partial loading. Given the user satisfaction impact on the Fog computing services, their resource allocation goal was user-weighted energy efficiency (UWEE). Also, they utilized the concerned user mechanism (UCM) to draw the social characteristics of users. Then, to maximize UWEE, the resource allocation problem is designed as a mixed-integer nonlinear programming (MINLP) under the constraint of users' satisfaction. However, it cannot be correctly solved due to binary depletion and traditional relaxation algorithms. a Lagrange-based resource allocation scheme called the augmented Lagrange method (ALM) is presented for solving this joint optimization problem iteratively, where AMSGRAD is used for accelerating convergence. The results of the simulation indicate the success of the ALM-based resource allocation scheme concerning UWEE.

In [52] the authors provided a PTPN-based resource allocation strategy for fog computing. It helps the user automatically select satisfactory resources from a group of pre-assigned sources. Their strategy considers comprehensively the time and cost spent to carry out a task and the trust evaluation of fog resources and users. Based on the fog features, they create the PTPN task models in fog computing. In the proposed scheme, they suggested an algorithm predicting the task completion time. Also, they consider an evaluation scheme to calculate the validity of the fog source. The dynamic fog source allocation algorithm has a higher throughput compared to static allocation strategies in terms of concerning cost and time. However, researchers ignore energy consumption in this study.

In [53] the authors addressed the challenges of medical cyber-physical systems (MCPS) and long-delayed and unstable links between the cloud data center and medical devices. For dealing with this issue, they presented “Mobile edge cloud computing” or “Fog computing” as a solution. The integrated fog computing and MCPS introduced the FC-MCPS scheme, firstly formulating the problem as a mixed-integer non-linear linear program (MINLP) and then formulating it as a mixed-integer linear programming (MILP). To solve computational complexity, they suggested a two-phase linear programming-based heuristic algorithm (LP). In the presented scheme, the results generate an accurate and proper solution. As a result, it outperforms a greedy scheme. In addition, the proposed method is cost-efficient.

Researchers in [54] studied the connection of IoT devices generated requests and concerns about energy consumption and delays in fog networks. To solve this problem, they formulated the problem and used the mesh adaptive direct search algorithm (MADS) algorithm to find the optimal results. The proposed algorithm was more efficient than OOA and ESA algorithms. The simulation results show that the performance of MADS can be similar to the two algorithms but their proposed method has less computational complexity.

Resource Provisioning & Resource Sharing

In [55] the focus of the authors is on the resource allocation problem between fog nodes and mobile users to consider cost efficiency. Their purpose is to allocate the load for users and reduce costs to meet computational and communication constraints. They considered three conditions to allocate resources to several users in fog computing. These conditions are the single-version resource allocation, resource allocation with several duplications of the transmission cost model, and resource allocation with several duplications of various transmission costs. Two models have been discussed in the proposed method: the different transmission cost model and the transmission cost model. Users can load multiple versions with fixed transmission costs for the transmission cost model. So, a suitable greedy solution is proposed. The transmission cost is associated with the distance between pairs of fog nodes for the different transmission models. Therefore, they suggested a non-adaptive algorithm. The performance of the proposed algorithms is evaluated based on two real databases. The results confirm the efficiency and effectiveness of the proposed algorithm.

In [56] the authors presented a computational fog structure and a crowd-funding algorithm to integrate additional resources into the network. In the proposed algorithm, they proposed an encouraging mechanism to incentivize owners with more resources to share their resources with the resource set and supervise and support resources during the active tasks. The results and the simulation indicate the effectiveness of the presented encouraging mechanism in reducing the violation of SLA and accelerating the completion of tasks.

Researchers in [57] discussed the IoT layer, fog computing, and the choice of close source or sources. They examined resource discovery. They used hidden Markov chain learning in their proposed method and compared it with TOPSIS, VIKOR, and SAW methods. Compared with the existing methods, the efficiency of their proposed method led to a reduction in energy consumption.

workload Allocation

In [58] the authors addressed finding the node’s location in the Fog. In this regard, their purpose is to solve the problem of the fog node’s location for users who use mobile phones with limited batteries. Therefore, they provided a solution that supports limited energy sources and computations with low delay. They used the mixed-integer linear programming (MILP) formula in the proposed solution to solve the problem. Also, they provided innovative solutions to solve large-scale problems. The results obtained from a real mobility database indicate that the innovative solution is more accurate than the MILP formula. As a result, it can significantly save energy for end-users.

In [59] the authors investigated the delay and energy-efficient load allocation problem in the IoT-Edge-Cloud computing system. They adjusted a load allocation problem based on delay. Their scheme is the optimal load allocation among local edges, neighboring servers, and clouds to minimize energy consumption and reduce delay. In their proposed method, they applied the delay-based workload allocation (DBWA) algorithm, the drift theory, and the Lyapunov penalty to solve this problem. The simulation results show that the proposed method improves energy efficiency and reduces delay in an IoT-Edge-Cloud system.

Researchers in [60] provided a collaborative scheme on the challenge of service delivery and the long distance to the computing resources. In this solution, a three-tier architecture is used to overcome the mentioned limitation. They proposed a trust-based offloading method, the efficiency of which is evaluated in comparison with SSLBA, NFA, and RWA methods. The simulation results show that the proposed method can significantly reduce latency and improve the service response rate.

C. Scheduling Management

In this section, we review four common categories regarding the studies conducted on scheduling management. The approaches are presented in Fig. 5 as follows:

Workflow scheduling: Transferring the workflow to the cloud/fog computing environment and using different services to facilitate the implementation of the workflow.

Job scheduling: This refers to allocating users' tasks to virtual machines for execution. From the users' perspective, an appropriate scheduling algorithm should be able to perform the required tasks within the shortest time.

Task scheduling: This refers to allocating tasks to processing resources in such a way that some system performance parameters such as execution time can be optimized.

Task offloading scheduling: Mobile devices are faced with limitations such as limited battery life, low processing power, and limited storage space, which are effective in improving the quality of services provided to customers. To overcome these limitations, tasks that require processing and storage should be transferred to the cloud or other mobile devices around the user or to a combination of both.

Job Scheduling

In [61] the authors discussed fog computing and better responsiveness to all users' demands and computational requirements. They develop and simulate the fog computing system based on networks, smartphones, containers, and clouds in their paper. Their purpose is to check the possibility of using any available resources to reduce the total costs. It is performed with a Bag-OF-Tasks load model. The simulation results show that it reduces costs without increasing the average response time. In this study, energy consumption is not considered.

In [62] the authors suggested a new optimization approach called BLA8. It aims to solve the task scheduling problem in the fog computing environment. This method seeks to distribute some tasks among all fog nodes optimally. Its main goal is to make an appropriate tradeoff between the CPU runtime and the assigned memory to improve mobile users' needs for Fog computing services. The performance evaluation results indicate that their proposed scheme outperforms traditional algorithms, like PSO and GA, in terms of CPU runtime and the allocated memory. However, they ignore energy consumption in this study.

In [63], the authors presented a model to schedule the requests of the IoT service to minimize the overall service request delay. In this model, they used integer programming for solving the optimization problem. Furthermore, they introduced a customized genetic algorithm as an innovative method for scheduling requests of IoT and minimized the delay. They tested the customized genetic algorithm in a simulation environment, with consideration of the environment's dynamic nature. They evaluate the customized genetic algorithm and compare it with three schemes, including priority-strict queuing (PSQ), waited-fair queuing (WFQ), and round-robin (RR). It outperforms others in terms of delay and meeting the requests deadlines.

In [64] researchers tried to solve the problem of energy demand optimization. They proposed a scheduling method based on fog and cloud computing, which focused on reducing electricity consumption demand during peak hours. In the proposed method, the fog nodes provided consumers' priorities, which makes resource allocation more efficient by creating interaction between nodes. The use of node allocation reduces scheduling delays.

Task Offloading

In [65] the authors examined the loading problem in dynamic computing on the IoT-Fog systems. Their research assumes that channel status information can be fully obtained by the depletion agent. They presented a partially visible depletion scheme, enabling the IoT device to make an optimal depletion decision using incomplete channel status information. To minimize energy consumption and delay, the optimization issue has been formulated using the partially observable Markov decision process (POMDP). For finding the optimal loading solution, an offline algorithm based on the deep recurrent Q-network (DRQN) has been created. In the proposed POMDP solution, a DRQN-based offline algorithm is created. It combines LSTM and DQN. Compared to other depletion schemes, as shown by numerical results, their presented scheme can effectively decrease the energy consumption in the IoT devices and reduce delay during processing the computational tasks.

In [66] the authors suggested a task-loading design based on an enhanced contract NET protocol and the beetle antennae search algorithm in fog computing networks. In this problem, the distribution of the task nodes and fog nodes is done uniformly in a circular area with R radius. The responsibility of the task node is to divide the task into sub-tasks and send them to the fog nodes. The proposed scheme has been applied for reducing the task node cost. The proposed algorithm combines the beetle antennae search algorithm and GA. This method only focuses on reducing costs and does not address other parameters, like energy consumption.

In [67] the authors offered a four-layer architecture for determining the decision maker for task depletion. In this study, the issue is formulated as a population (evolutionary) game, which is solved with Replicator Maynard dynamics. In this study, optimization objectives are time and energy consumption. They emphasize using realistic parameters and values to simulate the proposed scheme. The findings show the practicality of their design in reducing main traffic.

In [68] authors considered a three-layer fog computing architecture. Mobility in user equipment is determined according to sojourn time in each Fog computing node. It can be expressed as an exponential distribution. The purpose of the researchers is to maximize the efficiency of user equipment and optimize decisions of loading and computational resource allocation to reduce the migration probability. They formulated this problem as an MINLP and divided the problem into two parts: (1) Loading tasks and (2) Resource allocation. They introduced a fog computing node selection scheme based on the Ginni coefficient, which is called GCFSA, to obtain an optimal off-loading strategy, as well as a distributed resource optimization algorithm based on the GA called ROAGA for solving the resource allocation problem. Their simulation results show that their scheme outperforms other basic algorithms and can obtain a quasi-optimal performance. In this simulation, they focus on reducing migration.

In [69] the authors presented a distributed learning scheme to minimize the fog computations' average cost if there is no knowledge about random traffic in non-DTN application scenarios. They presented a fully distributed learning approach to minimize the average cost and time of fog computations. Stochastic gradient descent is used to separate optimal operations between time slots to create a distributed evolutionary heuristic for separating and achieving semi-optimal approximation. Online learning can reduce the drop-in optimality caused by distributed scheduling. Their simulation results show that the proposed distributed learning is better than other schemes in terms of operating power and energy efficiency.

Researchers in [70] examined a multi-objective problem to optimize task completion time and energy consumption by combining GA and PSO algorithms. They proposed a task offloading plan to decide on offloading, select appropriate fog nodes, and allocate computing resources. The proposed method had better performance than GA, PSO, local computing, random offloading, and uniform offloading algorithms in terms of energy consumption and overall offload overhead.

Task Scheduling

In [71] the authors designed a model to solve the multi-objective task scheduling problem in fog computing. Their scheme is an adaptive multi-objective optimization method to schedule tasks. In the adaptive multi-objective optimization task scheduling scheme (AMOSM), they have considered the total runtime and the cost of task resources in the fog network as the resource allocation optimization objectives. Their experiments show that the presented scheme had a better performance compared to other methods concerning total runtime, cost of resources, and loading.

In [72] the authors addressed the complex task scheduling problem. In addition, they consider the consumed energy to reduce energy consumption if a mixed deadline condition is met in the IoT applications. They adjusted a limited optimization in the cloud-fog environment for solving the task scheduling problem. This problem can be solved using the laxity and ant colony system algorithm (LBP-ACS). In their proposed scheme, a hybrid task scheduling strategy is considered. It includes the priority of a task and its deadline. To manage the delay sensitivity in a task, the Laxity-based priority algorithm seeks to build a task scheduling sequence with proper priority. Furthermore, the limited optimization algorithm applies the ant colony algorithm to minimize total energy consumption. They compared their proposed method with other algorithms, the results of which indicate the effectiveness of the presented algorithm in reducing energy consumption to process all tasks. It also can ensure the appropriate scheduling length and decrease the failure rate of the scheduling of tasks with different deadlines.

Researchers in [73] conducted a study on Elastic Optical Networks (EONs) in the underlying basic tiers. Their main focus was on solving the traffic problem of fog services and reducing excess energy consumption. They proposed an Energy-efficient Deep Reinforced Traffic Grooming (EDTG) algorithm. They extracted features that they implemented with the Advantage Actor-Critic (A2C) algorithm and an artificial neural network (ANN). The results show that the proposed algorithm can significantly reduce energy consumption compared with the two DRL and SGA algorithms.

Researchers in [74] proposed an efficient method of task scheduling in a heterogeneous virtual cloud by focusing on the energy consumption reduction problem. Their proposed method uses a logical balance method between task scheduling and energy saving. The mechanism of their proposed method is such that they will first have an initial schedule to reduce the execution time and then re-plan to find the best execution place in due time with less energy consumption. The proposed EPETS method has significantly better performance than energy-efficient scheduling methods such as RC-GA, AMTS, and E-PAGA in terms of energy consumption.

Researchers in [75] developed an alternative technique for IoT requests called AEOSSA in a cloudy environment. The AEOSSA method uses a combination of AEO and SSA algorithms to solve the task scheduling problem. The performance of the AEOSSA approach designed to solve the scheduling problem was compared with five traditional metaheuristic techniques. The simulation results showed that the proposed method had better throughput and makespan than the other 5 methods.

Researchers in [76] proposed an IEGA algorithm to solve the problem of scheduling tasks in fog computations. The mechanism of the proposed method consists of two steps: first, the mutation rate and the crossover rate are set to find the optimal combinations, and second, several solutions are mutated based on a certain probability to discover a better solution and not get stuck at local minima. The proposed method was compared with five evolutionary optimization algorithms. The proposed method was shown to be superior to other algorithms in terms of energy consumption, makespan, and several other parameters.

Workflows Scheduling

In [77] the authors presented a three-layer architecture, including Fog, cloud, and consumer layers. They presented a meta-heuristic algorithm called the improved particle swarm optimization with levy walk (IPSOLW) for load balancing. This algorithm combines PSO and LW. Users send their requests to the fog servers and then receive services. When the fog layer is damaged, the cloud is used for storing the consumers' records and providing services to them. Finally, a comparison is made between their algorithm and available algorithms like BAT, PSO, BPSO, CLW, and GA. They evaluated some parameters, including response time, cost, and processing time. Experiment results show that their proposed algorithm outperforms other algorithms.

In [78] the researchers presented a hybrid architecture for the dynamic scheduling of several tasks in the real-time IoT. In traditional approaches, the IoT task processing is conducted on the fog layer; whereas, in their approach, it is attempted to schedule computational tasks with low communication necessities in the cloud and tasks with compact communications and low computational requirements in the Fog. Their scheme considers the communication cost during the scheduling process, which is due to data transferred from devices and sensors in the IoT layer to the fog layer. The performance of their scheme is evaluated by simulating an unaware cloud strategy. The simulation results show that their proposed scheme has a lower deadline compared to the base policy.

A. service management

In this section, we review two common categories regarding the studies conducted on service.

Service Placement

In [79], the authors provided a method, which is based on the monitoring, analysis, decision-making, and execution (MADE) methodology, to order the IoT services. This scheme operates autonomously. In this scheme, the first existing resources and the status of program services are controlled at runtime. In the next step, the requested services are prioritized according to the deadline of program services. Then, the evolutionary strength Pareto II algorithm is used to decide on ordering program services as a multi-objective optimization problem. Finally, the decisions taken in previous phases are implemented in the fog environment. Their proposed scheme outperforms MOPSO and NSGAII algorithms in terms of various criteria such as service latency, fog utilization, and cost.

In [80] the authors described the various computing paradigms, like cloud computing, fog computing, and their combination, which is known as F2C computing. Their purpose was to study the benefits of fog computing and F2C computing and the delay support pattern in the Fog. They believed that F2C systems have a suitable performance because they can provide and implement distinct strategies for evaluating distributed services in F2C. The results show that the distributed implementation of sources in F2C has many benefits concerning service response time and main network load. In this study, the results of the cloud-based resource allocation scheme (CL) are compared with the three placement strategies, including FF, BF, and BQ, for four modes, including network delay, processing delay, general delay, and main network load. However, they do not address heterogeneous sources of the Fog.

In [81] authors examined the services on mobile networks equipped with fog computing. They suggested a QoS-aware scheme based on existing delivery methods to support vehicle services in real time. This paper presented three designs, including without any migration services, with migration services of HANDOVER, and the proposed design. The two first designs have their weaknesses. Thus, the third design has been introduced. The main idea of the third design is to combine two strategies to minimize migration overhead and maintain end-to-end performance at an acceptable level to meet QoS requirements. In their research, the authors have conducted a case study based on the real vehicle mobility pattern in a small European country. The proposed design is evaluated based on three criteria, including delay, reliability, as well as migration costs.

In [82] the authors investigated the SFC migration problem/reconstruction developed by user motion in cloud-fog computing environments. In the first step, they formulated the SFC migration problem as an integer linear programming. Then, they suggested two SFC migration approaches, including the minimum number of VNF migration strategies and a two-phase migration strategy. The two-phase proposed migration strategy is simulated in the cloud-fog computing environment. It can improve the configuration cost, reconstruction success rate, migration time, and failure of the proposed algorithm compared to basic algorithms.

In [83] the authors examined the resource supply problem during the use of fog computing resources. They proposed a conceptual, computational framework. Then, they formulated the service replacement problem for IoT applications in fog resources as an optimization problem. The authors consider heterogeneous resources and applications in their paper. They proposed a genetic algorithm as evolutionary solution to solve the optimization problem. The simulation results show that the implementation of this service can reduce communication delays in the fog network.

Researchers in [84] discussed the challenges of the geographic distribution of nodes for the proper management and processing of requests in fog computing. They provided a two-tier fog framework. They formulated the problem with the EGA algorithm considering the parameters to reduce the service time, costs, and energy consumption and thus ensure the QoS of the IoT system. The results of the proposed method had better performance than DEBTS, DMS, FIRST-FIT, branch and bound, and GAPSO algorithms in terms of service cost, energy consumption and service time.

Researchers in [85] provided a trust management system (TMS) regarding the security of loading and offloading requests to the cloud computing layer. In the proposed system, the service requester first checks the trust condition of the service provider and then sends the requests. In addition to QoS issues, it also provides QoS. Fuzzy logic is used in their proposed method to ensure the security of services. The proposed method has better performance than a method without TMS in terms of Delay and Throughput.

In [86] researchers proposed a solution using MAPE-K methodology in a fog environment and the Whale Optimization Algorithm (WOA) to solve the service placement problem. In this solution, operational capacity and energy consumption were considered to be the main objective of their research. This method is implemented on a three-layer architecture to show the interaction between the IoT device and fog layers. The results of their proposed solution reduce resource consumption, service delay, and energy consumption compared to other meta-heuristic methods.

In [87] researchers used a genetic algorithm to solve the service placement problem, reduce the delay of programs in the cloud-fog environment, and use the network. In the proposed method, they defined a penalty parameter to reduce the delay. In the results of their proposed method, delay, network use, energy consumption, and cost were improved to an acceptable extent.

In [88], researchers proposed a computational framework to solve the service placement problem in the cloud-fog environment and optimize the IoT services. They formulated the service placement problem as an automatic planning model considering the heterogeneity of programs and resources. They proposed the use of the PSO algorithm to solve the problem of IoT service placement to maximizing the use of fog resources. Their simulation was based on various QoS criteria, which led the PSO-based method to exhibit better performance than other advanced methods.

Service Provisioning

In [89] the authors first suggested FOGPLAN; and then they introduced the QDFSP framework. The purpose of this framework is to dynamically provide QoS-aware fog services. QDFSP is related to the dynamic deployment of application services in fog nodes or the release of application services, already deployed in fog nodes for meeting the lowest cost, delay, and QoS prerequisites of applications. FOGPLAN is a practical framework, which can operate with no hypotheses and with the lowest information concerning the IoT nodes. The authors used an integer nonlinear programming formula as an optimization issue. Also, they presented two greedy algorithms, namely Min-Cost, and efficient Min-Cost, to address QDFSP periodically. In larger settings, Min-Cost generally has less runtime, meaning that it is faster. The Speed of Min-Viol is lower than Min-Cost, but Min-Viol has fewer delay violations and an average service delay compared to Min-Cost. Finally, QDFSP cannot be used for solving the optimization problem in a periodical manner, especially for large networks. This research mainly focuses on cost and delay and does not examine large-scale heterogeneous networks.

In [90] authors investigated the multimedia fog computing support algorithm, synchronization aspects during the use of resources in large-scale systems, and their ability to guarantee the QoS requirements. They added fusion technology of privacy protection and service location to large-scale multimedia applications. contrary to computing, energy supply, network access capacity, storage, and other factors of fog nodes, the fusion technology has the ability to obtain location data of the positioning service in real-time mobile terminals. The proposed method evaluates the time cutting, hardware, bandwidth, and other network resource balancing technologies, and the Fog computing support algorithm. It can improve the security and performance of the multimedia fog computing system. They simulated the FCS-FLSPP algorithm. Then, its results are compared with the MBCS-VSD algorithm in terms of different parameters, including the high use of multimedia system resources, and their ability for load balancing in real-time. It has a better performance than the MBCS-VSD algorithm.

Discussion

This section provides some systematic reports on the planned explanatory inquiries, which are as follows:

RQ1: Which scope and primary contexts are classified in service and Energy management in fog computing?

Table 2 present a comparison of the Service and Energy Management based on the presented taxonomy in the previous Section. There are four scopes are considered, including Architecture, Resource Management, Scheduling management, and finally Service Management. Scheduling Management has the greatest portion of the area with 38% usage in the literature. Furthermore, Resource management has about 26% usage in fog computing. Fig. 6 represents the amount of service and energy management domains.

Table 2: Categorization of the service and energy management approaches in the fog computing field

Category	Research	Strong point of research	Deficiency of research	New discovery
Architecture	[44]	Low Energy consumption	Not multiple objective	**Algorithm**
	[45]	Low energy Consumption Low response time Low resource utilization	data generated and considered in this study by each sensor was few. just a 2-hour experiment was considered	**Architecture**
	[46]	Low energy consumption Low delay Low response time Low transmission efficiency High Throughput	Not considering assessment of computation complexity	**Architecture** **Algorithm**
	[47]	Low latency High data Transmission Speed Time	very mathematical formulation Not compared to conventional networks	**Framework**
	[48]	Low Response Time Low resource utilization Low Cost Low Execution time	Only Simulation with 3 VM Machine	**Framework**
Resource Management	[49]	Low response time	Not consider energy consumption	**Architecture**
	[50]	Enhances the Efficiency Resource Management	Only 1 parameter were simulated	**Algorithm**
	[51]	less computation complexity	very mathematical formulation	**Algorithm**
	[52]	Low Completion Time Low price cost heterogeneous network	Not consider energy consumption very mathematical formulation	**Algorithm** **Strategy**
	[53]	Low cost	Only cost parameter were simulated	**Algorithm**
	[54]	Low Energy consumption low computational power very little complexity	Experiment with fewer IoT nodes and FOG devices	**Algorithm**
	[55]	low Replication cost low Transmission cost	Only 1 parameter (cost) were simulated very mathematical formulation	**Scheme**
	[56]	Low completion time	Not consider resource utilization Not consider energy consumption	**Algorithm**
	[57]	Low Energy consumption Resource Efficiency	Not consider delay	**Algorithm**
	[58]	Low Energy consumption	Very mathematical formulation not consider heterogeneous end-user devices	**Algorithm**
	[59]	Low Energy consumption Low delay	Not compared to popular algorithms	**Algorithm**
	[60]	Reduced Service Latency	support only static IoT devices	**Algorithm**

Category	Research	Strong point of research	Deficiency of research	New discovery
		Service Response Rate		
Scheduling Management	[61]	Low response time Low Utilization of Resources Low cost	Not consider energy consumption	**Model**
	[62]	Low execution time High Allocated memory Low Cost	Not consider energy consumption	**Algorithm**
	[63]	Low Latency Low Runtime Improved meeting of Requests Deadlines	Not multiple objective	**Algorithm**
	[64]	Low delay Low energy consumption Low cost	Not considering other qos metric	**Algorithm**
	[65]	Low delay Low Energy Consumption	Simulator engine was not mentioned	**Algorithm**
	[66]	Low Average Cost	Only cost parameter were simulated	**Algorithm**
	[67]	Low delay Low Energy Consumption Improved Convergence Time	Only offloading parameter were simulated	**Architecture**
	[68]	Low Migration Times Improved the Revenue of UEs Low Energy Comparison	Not considering other qos metric	**Algorithm**
	[69]	Low Energy Comparison Low Delays Low Running Times Low Time Cost	very mathematical formulation	**Algorithm**
	[70]	Task Completion time and Energy Consumption.	Not consider other qos parametrs	**Algorithm**
	[71]	Low Delay Low Execution Time Low cost	very mathematical formulation not implemented in physical environment	**Algorithm**
	[72]	Low Energy Consumption Low Scheduling Length of the Task	Not consider delay	**Algorithm**
	[73]	Reduce Energy consumption	simulated only with 14 Nodes and 21-link Not consider other Qos parameters	**Algorithm**
	[74]	Low Energy Consumption Improve Performance	execution time	**Algorithm**
	[75]	Makespan time Throughput	Not consider other Qos parameters	**Algorithm**
	[76]	Makespan, Flow Time Fitness Function Carbon Dioxide Emission Rate Energy Consumption	Not considering scaling large.	**Algorithm**
	[77]	Low response time Low Average processing time. Low Cost.	Not consider energy consumption	**Architecture. Algorithm**
	[78]	High Tasks Executed Low Cost Low Deadline Miss Ratio	Not consider communication cost by the transfer of data in fog layer	**Algorithm**
Service management	[79]	Low Utilization of resources Low Latency Low Cost	Not considering energy consumption Not considering their priority policies	**Algorithm Framework**
	[80]	Low Delay Low Response Time	heterogeneous fog resources Only delay parameter was simulated	**Algorithm**
	[81]	Low Latency Low Migration Cost High Reliability	Not considering other qos metric	**Scheme**
	[82]	Low cost Low Running Time Low Migration Time Low Down Time	Not considering scaling large.	**Algorithm**
	[83]	Low Response Times Low Service Execution Delays Low Resources Utilization	heterogeneous	**Method**

Category	Research	Strong point of research	Deficiency of research	New discovery
	[84]	Low Service cost Low Energy Consumption Low Service Time Low Resources Utilization	Docker settings are not described	**Algorithm**
	[85]	High Throughput Low Delay	Low range low number of clients	**Method**
	[86]	Low Resources Utilization Low Energy Consumption Low Delay	Only Simulation with 5 and 10 service type in fog nodes	**Algorithm**
	[87]	Low Resources Utilization Low Energy Consumption Low Latency Low Execution Time Cost	Only Simulation with maximum 4 fog node and 8 End Devices	**Algorithm**
	[88]	Low Execution Time Average Waiting Time Low Failed Services	Only Simulation with 5 service type and 5 fog nodes. Not considering energy consumption.	**Algorithm** **Framework**
	[89]	Low cost Low Delay	Not considering scaling large.	**Framework** **Algorithm**
	[90]	Low Resources Utilization Low average waiting time of Service	real-time performance is slightly	**Algorithm**

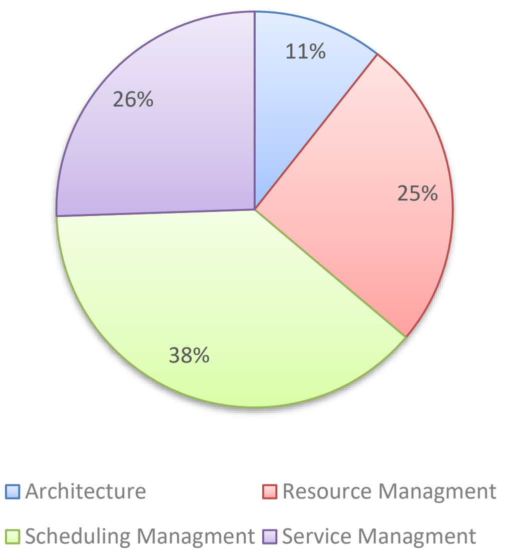

Fig. 6: Percentage of Service and Energy Management domains.

RQ2: What QOS parameters are utilized for evaluating the service and Energy management in fog computing?

Table 3 lists the QoS parameters used in fog computing.

This table introduces six parameters: response time, latency/delay, energy consumption, cost, resource utilization, throughput, and execution time. As shown in Fig. 7, the cost parameter has the highest percentage (i.e., 51%).

After the cost parameter, the two parameters, namely energy consumption & execution time, have a higher percentage than others (i.e., 42%). In this field, the percentage of resource utilization and latency/delay is equal to 33% and 30%.

Response time has the least percentage (i.e., 23%). Hence, it seems that it has been less investigated in the research literature.

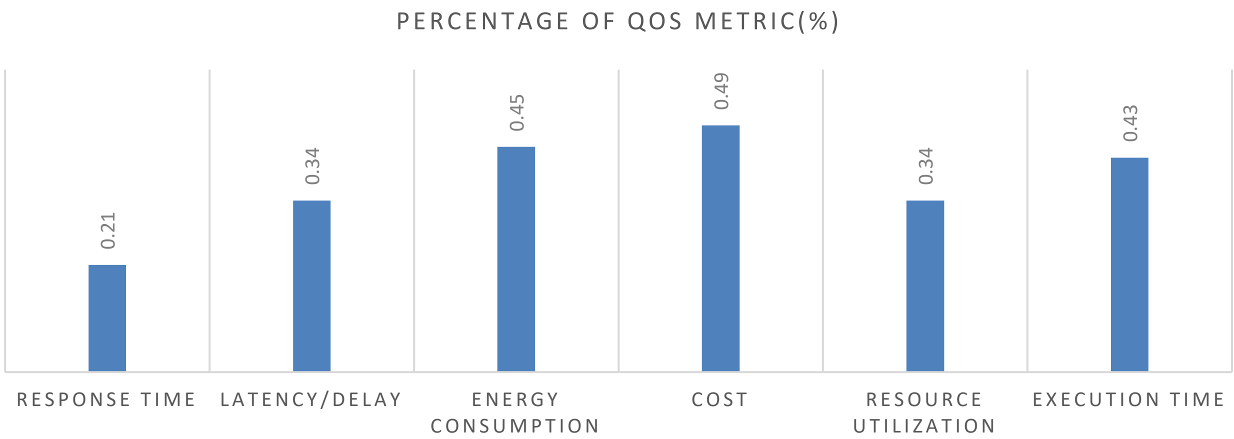

Fig. 7: The QoS parameters evaluated in this research.

Table 3: The QoS parameters reviewed in this research

Reference	Year	Response Time	Latency/delay	Energy consumption	Cost	resource utilization	Execution time
[44]	2019			✓			
[45]	2015	✓		✓		✓	
[46]	2021	✓	✓	✓			
[47]	2020	✓		✓	✓	✓	
[48]	2020	✓			✓	✓	✓
[49]	2020	✓					
[50]	2020				✓		
[51]	2020			✓			
[52]	2017				✓		✓
[53]	2015				✓		
[54]	2021			✓			
[55]	2020				✓		
[56]	2017						✓
[57]	2021	✓		✓		✓	✓
[58]	2020			✓		✓	
[59]	2019		✓	✓			
[60]	2021		✓				✓
[61]	2020	✓			✓	✓	
[62]	2018				✓	✓	✓
[63]	2020		✓				✓
[64]	2022		✓	✓	✓		
[65]	2020		✓	✓			
[66]	2020				✓	✓	
[67]	2020					✓	
[68]	2019			✓	✓		
[69]	2018		✓	✓	✓		✓
[70]	2021			✓			✓
[71]	2020		✓		✓		✓
[72]	2019			✓			
[73]	2021			✓			
[74]	2021			✓		✓	✓
[75]	2021						✓
[76]	2021			✓			✓
[77]	2019	✓			✓		✓
[78]	2019				✓		✓
[79]	2020		✓		✓	✓	
[80]	2018	✓	✓				
[81]	2019		✓		✓		
[82]	2019				✓		✓
[83]	2017	✓	✓		✓	✓	✓

Reference	Year	Response Time	Latency/delay	Energy consumption	Cost	resource utilization	Execution time
[84]	2021			✔	✔	✔	✔
[85]	2021		✔		✔		
[86]	2022		✔	✔		✔	
[87]	2022		✔	✔		✔	✔
[88]	2022						✔
[89]	2019		✔		✔		
[90]	2019				✔	✔	
Count		10	16	21	23	16	20

RQ3: What assessment situations are utilized for the estimation of the service and Energy management in fog computing?

Table 4 investigates different parameters, such as simulator, simulation environments, algorithm, and architecture used in papers. In service and energy management, 37 papers have only simulated their scenario; 7 papers have implemented their ideas for service management in the real environment. In addition, 3 papers have implemented a part of their work; they have done simulations to analyze and compare its results with others.

Fig. 8 represents the simulation of papers on energy consumption and service management approaches. Fig. 8 indicates the statistical percentage of the schemes. It is observed that 81% of the papers have been simulated using different tools. Furthermore, 15% of the papers have been implemented in the real environment, and 4% of them are simulated with other data; they belong to the third category.

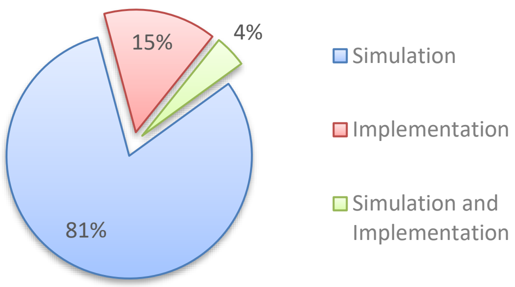

Fig. 8: Simulation and implementation used in the research literature.

The energy consumption and service management approaches consider an important subject, including architecture and framework, along with algorithms. This is depicted in Fig. 9. This figure displays the statistical percentage of the schemes. It can be seen that researchers present a framework or architecture in 30% of the papers. Also, 70% of the papers have no architecture or framework.

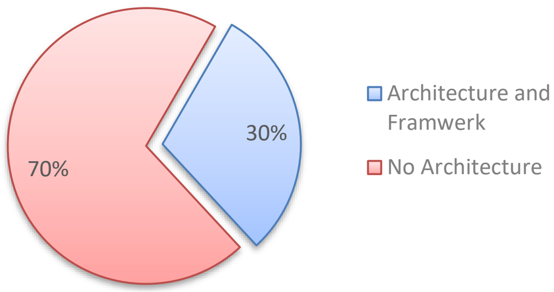

Fig. 9: Percentage of representation of architecture in the research literature.

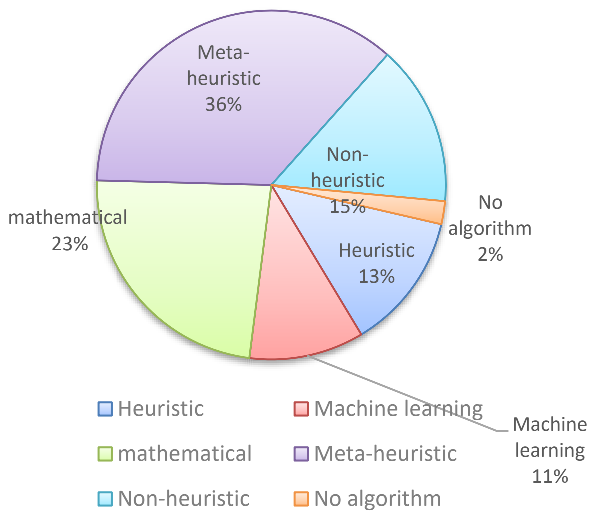

Fig. 10: Different percentages of various algorithms in the research literature.

Fig. 10 depicts algorithms and tools used in energy consumption and service management approaches. The statistical percentage of the methods is shown in Fig. 10. It can be seen that 36% of research literature includes papers, which use Meta heuristic methods. After that, mathematical schemes are included in 23% of the research. Finally, 2% of the papers do not use any algorithm. They have often presented a framework or architecture.

Table 4: Investigating the parameters of simulators, algorithms, and architecture in this research

Number	Data base	Year	Reference	Simulator	Evaluation	Algorithm	Algorithm type	Architecture
1	IEEE	2019	[44]	MATLAB	S	✓	Meta Heuristic	
2	IEEE	2015	[45]	HADOOP	I		Mathematical	✓
3	IEEE	2021	[46]	NS3	S	✓	Machine learning	✓
4	IEEE	2020	[47]	NM	I		Mathematical	✓
5	SPRINGER	2020	[48]	NS2	S		Non-Heuristic	✓
6	SPRINGER	2020	[49]	CLOUDSIM	S	✓	Non-Heuristic	✓
7	WILEY	2020	[50]	CLOUDSIM	S, I	✓	Non-Heuristic	✓
8	IEEE	2020	[51]	NM	S		Non-Heuristic	✓
9	IEEE	2017	[52]	NM	S	✓	Mathematical	
10	IEEE	2015	[53]	NM	S	✓	Heuristic	
11	WILEY	2021	[54]	NM	S	✓	Mathematical	
12	ELSEVIER	2020	[55]	NM	S	✓	Heuristic	
13	ELSEVIER	2017	[56]	HADOOP	I	✓	Machine Learning	
14	SPRINGER	2021	[57]	CLOUDSIM	S	✓	Mathematical	
15	IEEE	2020	[58]	PYTHON	I	✓	Heuristic	
16	IEEE	2019	[59]	MATLAB	S	✓	Mathematical	
17	ELSEVIER	2021	[60]	NM	S	✓	Mathematical	
18	ELSEVIER	2020	[61]	JPFF	I	✓	Non-Heuristic	
19	T & F	2018	[62]	C++	S	✓	Meta Heuristic	
20	ELSEVIER	2020	[63]	NM	S	✓	Meta Heuristic	
21	IEEE	2022	[64]	NM	S	✓	Mathematical	
22	IEEE	2020	[65]	NM	S		Machine learning	✓
23	SPRINGER	2020	[66]	NM	S	✓	Meta Heuristic	
24	SPRINGER	2020	[67]	PHYTON, MATLAB	S, I		Meta Heuristic	✓
25	IEEE	2019	[68]	NM	S	✓	Meta Heuristic	
26	IEEE	2018	[69]	NM	S	✓	Machine Learning	✓
27	ELSEVIER	2021	[70]	NM	S	✓	Meta Heuristic	
28	IEEE	2020	[71]	CLOUDSIM	I	✓	Meta Heuristic	
29	IEEE	2019	[72]	CLOUDSIM	S	✓	Meta Heuristic	
30	IEEE	2021	[73]	NSFNET	S	✓	Machine learning	
31	ELSEVIER	2021	[74]	Python	S	✓	Non-Heuristic	✓
32	ELSEVIER	2021	[75]	NM	S	✓	Meta Heuristic	
33	WILEY	2021	[76]	Java	S	✓	Meta Heuristic	
34	IEEE	2019	[77]	CLOUD ANALYST	S	✓	Meta Heuristic	✓
35	SPRINGER	2019	[78]	C++	S	✓	Heuristic	
36	WILEY	2020	[79]	IFOGSIM	S	✓	Meta Heuristic	
37	ELSEVIER	2018	[80]	NETEM	S		Non-Heuristic	
38	IEEE	2019	[81]	PYTHON	S		No Algorithm	
39	ELSEVIER	2019	[82]	NM	S	✓	Heuristic	
40	SPRINGER	2017	[83]	IFOGSIM	S	✓	Heuristic	✓
41	ELSEVIER	2021	[84]	Docker	I	✓	Meta Heuristic	
42	ELSEVIER	2021	[85]	NS3	S	✓	Mathematical	
43	ELSEVIER	2022	[86]	IFOGSIM	S	✓	Meta Heuristic	
44	SPRINGER	2022	[87]	IFOGSIM	S	✓	Meta Heuristic	
45	T&F	2022	[88]	MATLAB	S	✓	Meta Heuristic	✓
46	IEEE	2019	[89]	JAVA	S	✓	Mathematical	
47	SPRINGER	2019	[90]	MATLAB	S	✓	Mathematical	

RQ4: What used tools for service and Energy management in fog computing?

Table 4 lists the simulation tools in the papers. In the following, we introduce the most common simulators based on their use. Two simulation tools, Cloud Sim and MATLAB have been applied in 11% of the research papers. Hence, they have the most popularity in the literature. After that, Python and ifogsim have also been used to simulate 9% of the research papers. A drawback of some papers is that they have not introduced their simulator.

According to our knowledge in this research, 34% of the papers do not mention their simulators.

Fig. 11 shows the percentage of use of different simulators in this research.

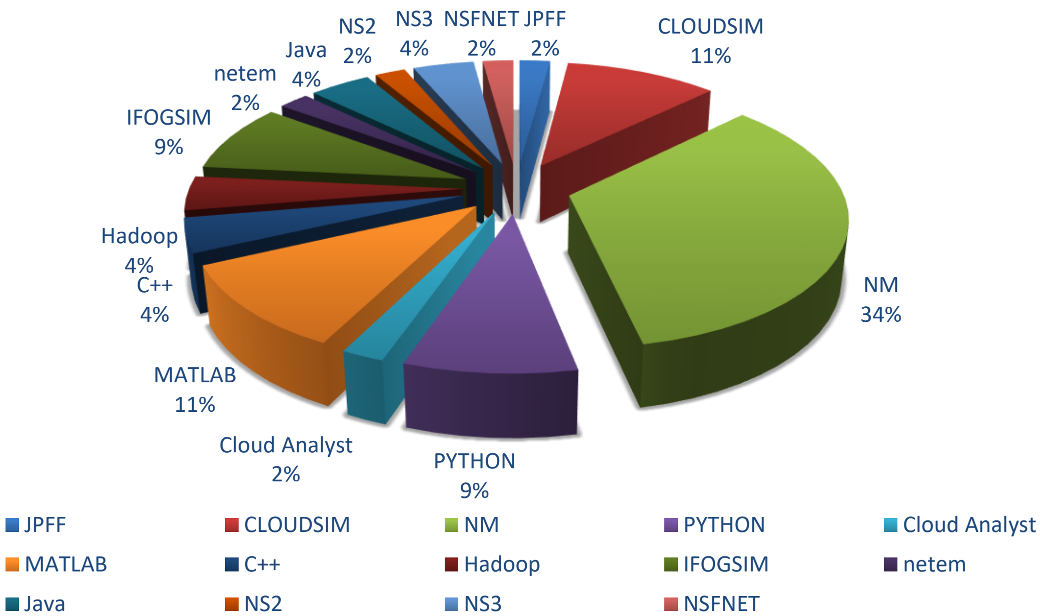

Fig. 11: The percentage of use of different simulators in the research literature.

RQ5: What is the significance of service and Energy management in fog computing?

The Fog architectures should allow storage, computing, and networking tasks to be dynamically migrated between the Tings, the Cloud, and the Fog. Therefore, the interfaces for Fog to interact with the Cloud, other Fogs, and the Things and users must:

1) Migration of the computing, control functions, and storage and facilitate flexibility through Other users
2) Permit appropriate access to the user for Fog services
3) Allow management of lifecycle in system and services effectively.

Task Scheduling will cause Energy consumption to be distributed among nodes in the network equally. In the real world, we are faced with limited resources. Resource allocation can increase efficiency and assign resources suitably to tasks.

RQ6: Which problems, future research directions, and challenges are identified concerning service and Energy management for future trends in fog computing?

In this paper, significant parts of fog computing have been investigated. In this section, future research directions for fog computing, existing gaps in studies of researchers, and open problems have been identified.

Security in Fog: Compared to the cloud, Fog has confronted the new security challenges. Fog is closer than the cloud, and its scattering is more than the cloud. Therefore, it is more vulnerable than the cloud and the centralized frameworks. The cloud works in a more secure structure, which is made by cloud operators, but the Fog works in a wide area. Security in fog computing is an open problem. Often, fog frameworks have more complexity than clouds. Therefore, they may not be able to manage their resources like clouds. In addition, the fog frameworks may not be able to recognize the dangers.

Not considering multi-objective: In some papers, authors only consider the efficiency of their method and do not examine other parameters. In fog computing, researchers must review not only cost and processing speed but also the parameters such as energy efficiency, delay, and throughput.

Not comparing with valid schemes: In various papers, an important challenge is that these schemes are not compared with valid methods. Often, they compare its method with basic methods, which is not appropriate in such a situation and shows a large difference. This means that the proposed methodology examines its efficiency compared to the classical schemes.

Simulation and implementation: The validity of a scheme is to implement it in the real environment. The simulation tool and its results are subject to real implementation. In general, some methods need the real implementation to evaluate their effectiveness. When a scheme implements in a real environment, it may encounter some problems; but, the simulation tool does not consider these constraints.

Scalability: In most simulations, the scalability of the network and the environment is limited. This causes some problems, like mobility and delay. These problems may cause data loss in emergency issues or not sending data.

Heterogeneous resources and programs: Heterogeneous resources and programs are not considered important principles in many papers. Most schemes consider resources and virtual machines in a similar form. With this view, there are some problems, such as data distribution. These problems may produce different results for users. Some papers, which considered the heterogeneity of resources, have suggested strategies such as artificial intelligence and online learning. This problem can be expressed as an open issue.

Not considering the dynamics of the environment and real-time systems: Simulations are carried out as a fixed and predefined scenario. Geographic environments are usually defined in limited environments (even less than 1 km). Therefore, it is dangerous and unreliable that the proposed scheme is implemented in important applications such as healthcare, smart applications, etc. Real-time data processing in fog computing is not considered. Chaos theory and Lyapunov are provided to solve scheduling issues and online distribution of services/data, which can be used to solve future problems.

Being math of methods: The researchers formulate the fog computing problem as mixed-integer nonlinear programming and transform it into linear programming to increase its efficiency. It has some disadvantages and advantages. These schemes are not simple, and their details are not mentioned. They are also not implemented in a real environment. Therefore, these schemes are not confirmed. The main strength of these schemes is that they are optimized using linear and nonlinear functions.

Reliability: Fog computing can be considered due to safety mechanisms, compatibility of fog nodes, fault-tolerant, high-performance service availability, and other QoS parameters. Despite these parameters, an energy-efficient scheme, which reduces energy consumption, cannot provide network reliability. Energy harvest is inherently unreliable.

Architecture presentation: The contribution of architecture in research is extremely limited, and most architect are a particular model for their research. It is suggested to present a combined proposed architecture or a more efficient method for important medical issues requiring low communication and latency.

Using container-based methods: Container technology has emerged in cloud and fog in various studies and is even known as a container as a service, which helps in better allocating and placing resources in cloudy and fog environments. Very little research has been conducted or developed on container-based methods, and it seems that this method can have a change in energy reduction and latency.

Conclusion

In this study, SLR-based research was presented in the service/energy management approaches in the fog computing field. Service/energy management is a robust solution for improving energy efficiency. Based on existing research, this field has four domains in service management. In addition, the strengths and drawbacks of research should be considered. Challenges of each paper have been given to develop more efficient solutions in future research for service management. In this paper, we have presented an SLR method. In this paper, we have reviewed 1596 studies published from 2013 to 2022. Finally, we selected 47 studies focused on service/energy management approaches in the fog computing field. According to the RQ1, we deduced that Scheduling management has the highest use (i.e., 38%) in the field of service/energy management approaches in Fog. Based on RQ2, the statistical percentage of evaluation factors indicates that different QoS parameters in papers focus extremely on the cost (i.e., 49%). According to RQ3, it can be deduced that 81% of the research papers use the simulation environment to evaluate their proposed scheme in the fog computing platform. In addition, 15% of the papers utilize a real environment to implement their schemes in the fog computing environment. In the following, for answering the question, we deduced that some researchers use meta-heuristic and statistical methods in their schemes (about 36%). 30% of the papers have provided an architecture or framework along with their schemes. Based on RQ4, which is about simulation and modeling tools, we observed that Cloud Sim and Matlab are widely used as simulation tools in studies (about 11%). According to the SLR-based scheme, we cannot evaluate all existing studies. Therefore, some constraints are considered, including removing non-English papers, removing papers, which are less than 6 pages, conference papers, book chapters, and survey papers.

We believe that this research reviews the conceptual characteristics of service/energy management approaches in fog computing. In general, service management in a computational environment still needs further studies. It should be able to deal with its heterogeneity to improve energy efficiency by reducing requests. This research helps researchers and specialists to obtain a general understanding of this field and perform future research.

Author Contributions

This paper is the result of S.M. Hashemi Research project which is supervised and advised by A. Sahafi ,A.M. Rahmani, and M. Bohlouli respectively. S.M. Hashemi wrote the manuscript. A. Sahafi and A.M. Rahmani sketched the research framework and the roadmap. Also, he analyzed the results and tabulated the outcome derived from excerpted literatures. In this line, M. Bohlouli searched in authentic journals to gather all relevant papers. they cooperatively summed up the work.

Acknowledgment

The authors would like to thank the editor and anonymous reviewers.

Conflict of Interest

The authors declare that there is no conflict of interests regarding the publication of this manuscript.

Abbreviations

IOT	Internet of Things
FC	Fog computing
SLR	Systematic literature review
S	Simulation
I	Implementation
NM	Not Mentioned
ECCM	Energy-Efficient Cross-Layer-Sensing Clustering Method
DLR	Dynamic Thermal Rating
PLR	Packet Loss Rate
UWEE	User-Weighted Energy Efficiency
MINLP	Mixed-Integer Nonlinear Programming
ALM	Augmented LAGRANGE Method
MCPS	Medical Cyber-Physical Systems
POMDP	Partially Observable Markov Decision Process
MADS	Mesh Adaptive Direct Search
DBWA	Delay-Based Workload Allocation
DRQN	Deep Recurrent Q-Network
EONs	Elastic Optical Networks
EDTG	Energy-Efficient Deep Reinforced Traffic Grooming
IPSOLW	Improved Particle Swarm Optimization with Levy Walk

Reference

[1] M. H. Kashani, M. Madanipour, M. Nikravan, P. Asghari, E. Mahdipour, "A systematic review of IoT in healthcare: Applications, techniques, and trends," J. Network Comput. Appl., 192: 103164, 2021.

[2] T. Samizadeh Nikoui, A. M. Rahmani, A. Balador, H. Seyyed Javadi, "Internet of things architecture challenges: A systematic review," Int. J. Commun. Syst., 34: e4678, 2021.

[3] H. Kashani, A. M. Rahmani, N. Jafari Navimipour, "Quality of service-aware approaches in fog computing," Int. J. Commun. Syst., 33: e4340, 2020.

[4] A. Sadri, A. M. Rahmani, M. Saberikamarposhti, M. Hosseinzadeh, "Fog data management: A vision, challenges, and future directions," J. Network Comput. Appl., 174: 102882, 2021.

[5] M. Rahimi, N. Jafari Navimipour, M. Hosseinzadeh, M. H. Moattar, A. Darwesh, "Toward the efficient service selection approaches in cloud computing," Kybernetes, 51: 1388-1412, 2022.

[6] M. Sheikh Sofla, H. Kashani, E. Mahdipour, R. Faghih Mirzaee, "Towards effective offloading mechanisms in fog computing," Multimedia Tools Appl., 81: 1997-2042, 2022.

[7] F. Davami, S. Adabi, A. Rezaee, A. M. Rahmani, "Fog-based architecture for scheduling multiple workflows with high availability requirement," Computing, 104: 169-208, 2022.

[8] N. S. Soulegan, B. Barekatain, B. S. Neysiani, "MTC: minimizing time and cost of cloud task scheduling based on customers and providers needs using genetic algorithm," Int. J. Intell. Syst. Appl. (IJISA), 13: 38-51, 2021.

[9] A. Najafizadeh, A. Salajegheh, A. M. Rahmani, A. Sahafi, "Task scheduling in fog computing: a survey," J. Adv. Comput. Res., 11: 33-56, 2020.

[10] A. Najafizadeh, A. Salajegheh, A. M. Rahmani, A. Sahafi, "Multi-objective task scheduling in cloud-fog computing using goal programming approach," Cluster Comput., 25: 141-165, 2022.

[11] M. Abbasi, E. Mohammadi-Pasand, M. R. Khosravi, "Intelligent workload allocation in IoT–Fog–cloud architecture towards mobile edge computing," Comput. Commun., 169: 71-80, 2021.

[12] M. Laroui, B. Nour, H. Moungla, M. A. Cherif, H. Afifi, M. Guizani, "Edge and fog computing for IoT: A survey on current research activities & future directions," Comput. Commun., 180: 210-231, 2021.

[13] D. Wang, D. Zhong, A. Souri, "Energy management solutions in the internet of things applications: Technical analysis and new research directions," Cognit. Syst. Res., 67: 33-49, 2021.

[14] A. Mijuskovic, A. Chiumento, R. Bemthuis, A. Aldea, P. Havinga, "Resource management techniques for cloud/fog and edge computing: An evaluation framework and classification," Sensors, 21: 1832, 2021.

[15] B. Costa, J. Bachiega, L. Rebouças, A. P. F. Araujo, "Orchestration in fog computing: A comprehensive survey," ACM Computing Surveys (CSUR), 55(2): 1-34, 2022.

[16] M. Gill, D. Singh, "A comprehensive study of simulation frameworks and research directions in fog computing," Comput. Sci. Rev., 40: 100391, 2021.

[17] M. R. Alizadeh, V. Khajehvand, A. M. Rahmani, E. Akbari, "Task scheduling approaches in fog computing: A systematic review," Int. J. Commun. Syst., 33: e4583, 2020.

[18] M. Ghobaei-Arani, A. Souri, A. A. Rahmanian, "Resource management approaches in fog computing: a comprehensive review," J. Grid Comput., 18: 1-42, 2020.

[19] M. Mukherjee, L. Shu, D. Wang, "Survey of fog computing: Fundamental, network applications, and research challenges," IEEE Commun. Surv. Tutorials, 20: 1826-1857, 2018.

[20] A. Dasgupta, A. Gill, "Fog computing challenges: a systematic review," in Proc. ACIS 2017: 79, 2017.

[21] V. Pande, C. Marlecha, S. Kayte, "A review-fog computing and its role in the internet of things," Int. J. Eng. Res. Appl., 6: 2248-96227, 2016.

[22] P. Hu, S. Dhelim, H. Ning, T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," J. Network Comput. Appl., 98: 27-42, 2017.

[23] G. Caiza, M. Saeteros, W. Oñate, M. V. Garcia, "Fog computing at industrial level, architecture, latency, energy, and security: A review," Heliyon, 6: e03706, 2020.

[24] R. Zolfaghari, A. Sahafi, A. M. Rahmani, R. Rezaei, "Application of virtual machine consolidation in cloud computing systems," Sustainable Comput. Inf. Syst., 30: 100524, 2021.

[25] M. Effatparvar, M. Dehghan, A. M. Rahmani, "A comprehensive survey of energy-aware routing protocols in wireless body area sensor networks," J. Med. Syst., 40: 1-27, 2016.

[26] J. Ghomi, A. M. Rahmani, N. N. Qader, "Load-balancing algorithms in cloud computing: A survey," J. Network Comput. Appl., 88: 50-71, 2017.

[27] B. Kitchenham et al., "Systematic literature reviews in software engineering–a tertiary study," Information and software technology, vol. 52, pp. 792-805, 2010.

[28] A. Vakili, N. J. Navimipour, "Comprehensive and systematic review of the service composition mechanisms in the cloud environments," J. Network Comput. Appl., 81: 24-36, 2017.

[29] F. Tao, L. Zhang, Y. Liu, Y. Cheng, L. Wang, X. Xu, "Manufacturing service management in cloud manufacturing: Overview and future research directions," J. Manuf. Sci. Eng., 137(4): 040912, 2015.

[30] Y. Sun, Z. Xiao, D. Bao, J. Zhao, "An architecture model of management and monitoring on cloud services resources," in Proc. 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), 2010.

[31] R. Mahmud, R. Kotagiri, R. Buyya, "Fog computing: A taxonomy, survey and future directions," Internet of Everything: Algorithms, Methodologies, Technologies and Perspectives, Springer, 2018.

[32] E. Casalicchio, L. Silvestri, "Architectures for autonomic service management in cloud-based systems," presented at the 2011 IEEE Symposium on Computers and Communications (ISCC), Kerkyra, Greece, 2011.

[33] B. Shojaiemehr, A. M. Rahmani, N. N. Qader, "Cloud computing service negotiation: A systematic review," Comput. Stand. Interfaces, 55: 196-206, 2018.

[34] E. Jafarnejad Ghomi, M. Rahmani, N. Qader, "Service load balancing, task scheduling and transportation optimisation in cloud manufacturing by applying queuing system," Enterp. Inf. Syst., 13: 865-894, 2019.

[35] L. Liu, D. Qi, N. Zhou, Y. Wu, "A task scheduling algorithm based on classification mining in fog computing environment," Wireless Commun. Mobile Comput., 2102348: 1-12, 2018.

[36] R. M. B. Abadi, A. M. Rahmani, S. H. Alizadeh, "Challenges of server consolidation in virtualized data centers and open research issues: a systematic literature review," J. Supercomput., 76: 2876-2927, 2020.

[37] S. Gu, Q. Zhuge, J. Yi, J. Hu, E. H. M. Sha, "Optimizing task and data assignment on multi-core systems with multi-port SPMs," IEEE Trans. Parallel Distrib. Syst., 26: 2549-2560, 2014.

[38] W. Lin, S. Xu, L. He, J. Li, "Multi-resource scheduling and power simulation for cloud computing," Inf. Sci., 397: 168-186, 2017.

[39] M. A. A. Faruque, F. Ahourai, "A model-based design of cyber-physical energy systems," presented at the 2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC), Singapore, 2014.

[40] F. Faruque, K. Vatanparvar, "Energy management-as-a-service over fog computing platform," IEEE Internet Things J., 3: 161-169, 2015.

[41] H. Barati, A. Movaghar, A. M. Rahmani, "EACHP: Energy aware clustering hierarchy protocol for large scale wireless sensor networks," Wireless Pers. Commun., 85: 765-789, 2015.

[42] S. E. Dashti, A. M. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," J. Exp. Theor. Artif. Intell., 28: 97-112, 2016.

[43] K. Chronaki et al., "Task scheduling techniques for asymmetric multi-core systems," IEEE Trans. Parallel Distrib. Syst., 28: 2074-2087, 2016.

[44] Z. Sun et al., "An energy-efficient cross-layer-sensing clustering method based on intelligent fog computing in WSNs," IEEE Access, 7: 144165-144177, 2019.

[45] N. Kaur, S. K. Sood, "An energy-efficient architecture for the Internet of Things (IoT)," IEEE Syst. J., 11: 796-805, 2015.

[46] A. Rafiq, W. Ping, W. Min, M. Saleh, "Fog assisted 6TiSCH tri-layer network architecture for adaptive scheduling and energy-efficient offloading using rank-based Q-learning in smart industries," IEEE Sensors J., 21: 25489-25507, 2021.

[47] M. Dabbaghjamanesh, A. Kavousi-Fard, Z. Y. Dong, "A novel distributed cloud-fog based framework for energy management of networked microgrids," IEEE Trans. Power Syst., 35: 2847-2862, 2020.

[48] G. Rathee, R. Sandhu, H. Saini, M. Sivaram, V. Dhasarathan, "A trust computed framework for IoT devices and fog computing environment," Wireless Networks, 26: 2339-2351, 2020.

[49] Z. Ren, T. Lu, X. Wang, W. Guo, G. Liu, S. Chang, "Resource scheduling for delay-sensitive application in three-layer fog-to-cloud architecture," Peer-to-Peer Networking Appl., 13: 1474-1485, 2020.

[50] S. K. Mani, I. Meenakshisundaram, "Improving quality-of-service in fog computing through efficient resource allocation," Comput. Intell., 36: 1527-1547, 2020.

[51] X. Chen, Y. Zhou, L. Yang, L. Lv, "User satisfaction oriented resource allocation for fog computing: A mixed-task paradigm," IEEE Trans. Commun., 68: 6470-6482, 2020.

[52] L. Ni, J. Zhang, C. Jiang, C. Yan, K. Yu, "Resource allocation strategy in fog computing based on priced timed petri nets," IEEE Internet Things J., 4: 1216-1228, 2017.

[53] L. Gu, D. Zeng, S. Guo, A. Barnawi, Y. Xiang, "Cost efficient resource management in fog computing supported medical cyber-physical system," IEEE Trans. Emerging Top. Comput., 5: 108-119, 2015.

[54] R. Basir, S. Qaisar, M. Ali, M. Naeem, A. Anpalagan, "Energy efficient resource allocation in cache-enabled fog networks," Trans. Emerging Telecommun. Technol., 32: e4343, 2021.

[55] S. Lu, J. Wu, Y. Duan, N. Wang, J. Fang, "Towards cost-efficient resource provisioning with multiple mobile users in fog computing," J. Parallel Distrib. Computing, 146: 96-106, 2020.

[56] Y. Sun, N. Zhang, "A resource-sharing model based on a repeated game in fog computing," Saudi J. Biol. Sci., 24: 687-694, 2017.

[57] S. Kalantary, J. Akbari Torkestani, A. Shahidinejad, "Resource discovery in the internet of things integrated with fog computing using Markov learning model," J. Supercomput., 77: 13806-13827, 2021.

[58] S. Rodrigo, N. Ls, "Location of fog nodes for reduction of energy consumption of end-user devices," IEEE Trans. Green Commun. Networking, 4: 593-605, 2020.

[59] M. Guo, L. Li, Q. Guan, "Energy-efficient and delay-guaranteed workload allocation in IoT-edge-cloud computing systems," IEEE Access, 7: 78685-78697, 2019.

[60] N. Mazumdar, A. Nag, J. P. Singh, "Trust-based load-offloading protocol to reduce service delays in fog-computing-empowered IoT," Comput. Electr. Eng., 93: 107223, 2021.

[61] D. Tychalas, H. Karatza, "A scheduling algorithm for a fog computing system with bag-of-tasks jobs: Simulation and performance evaluation," Simul. Modell. Pract. Theory, 98: 101982, 2020.

[62] S. Bitam, S. Zeadally, A. Mellouk, "Fog computing job scheduling optimization based on bees swarm," Enterp. Inf. Syst., 12: 373-397, 2018.

[63] R. O. Aburukba, M. AliKarrar, T. Landolsi, K. El-Fakih, "Scheduling internet of things requests to minimize latency in hybrid Fog–Cloud computing," Future Gener. Comput. Syst., 111: 539-551, 2020.

[64] S. Chouikhi, M. Esseghir, L. Merghem-Boulahia, "Energy consumption scheduling as a fog computing service in smart grid," IEEE Trans. Serv. Comput., 16: 1144-1157, 2022.

[65] R. Xie, Q. Tang, C. Liang, F. R. Yu, T. Huang, "Dynamic computation offloading in IoT fog systems with imperfect channel-state information: A POMDP approach," IEEE Internet Things J., 8: 345-356, 2020.

[66] X. Li, Z. Zang, F. Shen, Y. Sun, "Task offloading scheme based on improved contract net protocol and beetle antennae search algorithm in fog computing networks," Mobile Networks Appl., 25: 2517-2526, 2020.

[67] H. Mahini, A. M. Rahmani, S. M. Mousavirad, "An evolutionary game approach to IoT task offloading in fog-cloud computing," J. Supercomput., 77: 5398-5425, 2021.

[68] D. Wang, Z. Liu, X. Wang, Y. Lan, "Mobility-aware task offloading and migration schemes in fog computing networks," IEEE Access, 7: 43356-43368, 2019.

[69] C. Ren, X. Lyu, W. Ni, H. Tian, and R. P. Liu, "Distributed online learning of fog computing under nonuniform device cardinality," IEEE Internet Things J., 6: 1147-1159, 2018.

[70] O. K. Shahryari, H. Pedram, V. Khajehvand, M. D. TakhtFooladi, "Energy and task completion time trade-off for task offloading in fog-enabled IoT networks," Pervasive Mob. Comput., 74: 101395, 2021.

[71] M. Yang, H. Ma, S. Wei, Y. Zeng, Y. Chen, Y. Hu, "A multi-objective task scheduling method for fog computing in cyber-physical-social services," IEEE Access, 8: 65085-65095, 2020.

[72] J. Xu, Z. Hao, R. Zhang, X. Sun, "A method based on the combination of laxity and ant colony system for cloud-fog task scheduling," IEEE Access, 7: 116218-116226, 2019.

[73] R. Zhu, S. Li, P. Wang, M. Xu, S. Yu, "Energy-efficient deep reinforced traffic grooming in elastic optical networks for cloud–fog computing," IEEE Internet Things J., 8: 12410-12421, 2021.

[74] M. Hussain, L. F. Wei, A. Lakhan, S. Wali, S. Ali, A. Hussain, "Energy and performance-efficient task scheduling in heterogeneous virtualized cloud computing," Sustainable Comput. Inf. Syst., vol. 30: 100517, 2021.

[75] M. Abd Elaziz, L. Abualigah, I. Attiya, "Advanced optimization technique for scheduling IoT tasks in cloud-fog computing environments," Future Gener. Comput. Syst., 124: 142-154, 2021.

[76] M. Abdel-Basset, R. Mohamed, R. K. Chakrabortty, M. J. Ryan, "IEGA: an improved elitism-based genetic algorithm for task scheduling problem in fog computing," Int. J. Intell. Syst., 36: 4592-4631, 2021.

[77] Z. A. Khan et al., "Energy management in smart sectors using fog-based environment and meta-heuristic algorithms," IEEE Access, 7: 157254-157267, 2019.

[78] L. Stavrinides, H. D. Karatza, "A hybrid approach to scheduling real-time IoT workflows in fog and cloud environments," Multimedia Tools Appl., 78: 24639-24655, 2019.

[79] M. Ayoubi, M. Ramezanpour, R. Khorsand, "An autonomous IoT service placement methodology in fog computing," Softw: Pract. Exper., 51: 1097-1120, 2021.

[80] V. B. Souza et al., "Towards a proper service placement in combined Fog-to-Cloud (F2C) architectures," Future Gener. Comput. Syst., 87: 1-15, 2018.

[81] J. Li et al., "Service migration in fog computing enabled cellular networks to support real-time vehicular communications," IEEE Access, 7: 13704-13714, 2019.

[82] D. Zhao, G. Sun, D. Liao, S. Xu, V. Chang, "Mobile-aware service function chain migration in cloud–fog computing," Future Gener. Comput. Syst., 96: 591-604, 2019.

[83] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, P. Leitner, "Optimized IoT service placement in the fog," Serv. Oriented Comput. Appl., 11: 427-443, 2017.

[84] B. V. Natesha, G. Ram, "Adopting elitism-based Genetic Algorithm for minimizing multi-objective problems of IoT service placement in fog computing environment," J. Network Comput. Appl., 178: 102972, 2021.

[85] S. O. Ogundoyin, I. A. Kamil, "A trust management system for fog computing services," Internet of Things, 14: 100382, 2021.

[86] M. Ghobaei-Arani, A. Shahidinejad, "A cost-efficient IoT service placement approach using whale optimization algorithm in fog computing environment," Expert Syst. Appl., 200: 117012, 2022.

[87] N. Sarrafzade, R. Entezari-Maleki, L. Sousa, "A genetic-based approach for service placement in fog computing," J. Supercomput., 78: 10854-10875, 2022.

[88] M. Salimian, M. Ghobaei-Arani, A. Shahidinejad, "An evolutionary multi-objective optimization technique to deploy the IoT Services in fog-enabled Networks: an autonomous approach," Appl. Artif. Intell., 36: 2008149, 2022.

[89] A. Yousefpour et al., "Fogplan: A lightweight qos-aware dynamic fog service provisioning framework," IEEE Internet Things J., 6: 5080-5096, 2019.

[90] Y. Chen, Y. Wang, D. Gong, "Fog computing support scheme based on fusion of location service and privacy preservation for QoS enhancement," Peer-to-Peer Networking Appl., 12: 1480-1488, 2019.

Biographies

Sayed Mohsen Hashemi received the bachelor’s degree in information technology from Payame Noor University, Iran, in 2011, and the master’s degree in software engineering from Islamic Azad University, Meybod Branch, Iran, in 2013. He is currently pursuing the Ph.D. degree in computer engineering with Islamic Azad University, Qeshm, Iran. His research interests include cloud computing, fog computing, the Internet of Things, routing algorithm in computer networks, data mining, machine learning, and meta heuristic algorithms.

- Email: mohsenpnu2009@gmail.com
- ORCID: 0000-0002-6506-9987
- Web of Science Researcher ID: HGU-0399-2022
- Scopus Author ID: NA
- Homepage: NA

Amir Sahafi received the B.Sc. degree in computer engineering from Shahed University, Tehran, Iran, in 2005, and the M.Sc. and Ph.D. degrees in computer engineering from the Science and Research Branch, Islamic Azad University, Tehran, in 2007 and 2012, respectively. He is currently an Assistant Professor with the Department of Computer

Engineering, South Tehran Branch, Islamic Azad University, Tehran. His current research interest includes distributed and cloud computing.

- Email: sahafi@iau.ac.ir
- ORCID: 0000-0002-6555-670X
- Web of Science Researcher ID: NA
- Scopus Author ID: 24528878600
- Homepage: https://stb.iau.ir/faculty/a-sahafi/fa

Amir Masoud Rahmani received the B.S. degree in computer engineering from Amir Kabir University, Tehran, in 1996, the M.S. degree in computer engineering from the Sharif University of Technology, Tehran, in 1998, and the Ph.D. degree in computer engineering from IAU University, Tehran, in 2005. He is currently a Professor of computer engineering. His research interests include distributed systems, the Internet of Things, and evolutionary computing.

- Email: rahmani@srbiau.ac.ir
- ORCID: 0000-0001-8641-6119
- Web of Science Researcher ID: K-2702-2013
- Scopus Author ID: 57204588830
- Homepage: https://srb.iau.ir/faculty/a-rahmani/fa

Mahdi Bohlouli received the Ph.D. degree from the University of Siegen, with the main focus on statistical regeneration and scalable clustering of big data using Map Reduce in the Hadoop ecosystem. He was the Group Leader of web search and data mining at the Institute for Web Science and Technologies (WeST), University of Koblenz, Germany, and a Senior Research Associate and a Project Manager at the Institute of Knowledge-Based Systems (KBS), University of Siegen, Germany. He is currently an Assistant Professor of data science and machine learning at the Institute for Advanced Studies in Basic Sciences (IASBS) with the main focus on the large-scale data analysis and next generation AI, in particular reinforcement, generative and self-attentive learning algorithms. He leads the Intelligent Systems Group (ISG), IASBS, as well as various workshops and conferences, being co-located with ICML, WWW, IEEE Big Data, and Semantics conferences and many more. He has promising scholar records in various AI fields. He was also involved in leading up various industrial software projects in the IT sector.

- Email: me@bohlouli.com
- ORCID: 0000-0002-6659-5524
- Web of Science Researcher ID:
- Scopus Author ID: 55809185800
- Homepage: www.bohlouli.com

How to cite this paper:
S. M. Hashemi, A. Sahafi, A. M. Rahmani, M. Bohlouli, “Service and energy management in fog computing: A taxonomy approaches, and future directions,” J. Electr. Comput. Eng. Innovations, 12(1): 15-38, 2024.

DOI: 10.22061/jecei.2023.9482.624

URL: https://jecei.sru.ac.ir/article_1886.html