International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN: 2147-6799 www.ijisae.org Original Research Paper

Metaheuristic Techniques for Resource Provisioning in fog Computing

Kirandeep Kaur[1] Arjan Singh[2] Anju Sharma[3]

Submitted: 15/04/2020 **Revised:** 25/05/2020 **Accepted:** 05/06/2020

Abstract: Fog computing has become a promising alternative to traditional cloud computing, especially for Internet of Things (IoT) applications that demand low latency and quick data processing. By bringing computing power, storage, and communication closer to the edge of the network, fog computing supports faster responses and improves overall system performance. Despite its benefits, managing resources in fog environments presents real challenges. These systems include a wide variety of devices, each with different capabilities, and often operate with limited resources. Allocating these resources efficiently becomes even more difficult when the workload changes frequently or unexpectedly. This paper examines how metaheuristic algorithms can improve resource allocation and provisioning in fog computing systems. It discusses several techniques—such as Particle Swarm Optimization (PSO)**,** Genetic Algorithms (GA)**,** Cat Swarm Optimization (CSO)**,** greedy algorithms**,** and distance-based strategies—that aim to balance key objectives like reducing delay, saving energy, lowering cost, and improving Quality of Service (QoS). These algorithms are known for their ability to adapt to changing conditions in real time. They help prevent problems like resource overload or underuse. However, there are still some challenges. Many proposed solutions have yet to be tested in real-world environments. Simulation tools like iFogSim and CloudSim are helpful for early research but fall short in capturing the complexity of actual fog systems. Additionally, concerns about data security and access control remain important and must be addressed as these systems grow. In summary, metaheuristic techniques offer significant potential for improving how resources are managed in fog computing, particularly for IoT use cases. They help make better use of available resources while maintaining performance. Nonetheless, more research is needed—especially in areas like real-world testing, dynamic task scheduling, and secure resource management—to fully realize the benefits of these approaches in practical applications.

Keywords: *Fog computing, Resource provisioning, Metaheuristic techniques, Cloud computing, Resource management, Internet of Things (IoT).*

1. INTRODUCTION

The rapid evolution of the Internet of Things (IoT) has been strongly influenced by advancements in cloud computing and virtualization. As billions of devices become interconnected, vast and complex networks are formed, generating large volumes of data that must be processed efficiently and securely. Managing such data within traditional cloud frameworks presents several limitations, particularly in applications requiring real-time processing and minimal latency (Zanella et al., 2014; Chiang & Zhang, 2016).

Conventional cloud architectures rely on centralized data centers that are often geographically distant from end-user devices. This physical separation introduces latency, making cloud computing less suitable for delay-sensitive applications such as remote healthcare monitoring, autonomous vehicles, and industrial automation. Additionally, a significant portion of computing and storage resources available at the network edge—such as gateways, routers, and local servers—often remains underutilized in centralized models (Chiang & Zhang, 2016).

To address these shortcomings, fog computing has emerged as a viable solution. By extending cloud services to the edge of the network, fog computing brings computational, storage, and networking capabilities closer to the data source.

[1]Punjabi University, Patiala, India, [2]Punjabi University, Patiala, India, [3]Punjab State Aeronautical College, Patiala, India
Correspondence: Kirandeep Kaur, Punjabi University, Patiala, India. Email: kirandeep_rs17@pbi.ac.in

This shift enables faster response times, reduces bandwidth consumption, and enhances system reliability. In essence, fog computing acts as a bridge between the cloud and edge devices, offering localized processing that supports the real-time demands of IoT ecosystems (Tuli et al., 2020). Furthermore, fog computing improves data privacy, mobility, and security by enabling local data processing and storage. It reduces the need to transmit sensitive information over long distances, thereby minimizing potential exposure to cyber threats. This decentralized approach gives users greater control over their data and ensures more responsive and context-aware computing. Real-time applications—such as smart traffic systems, environmental monitoring, and telemedicine—benefit significantly from the reduced latency and energy efficiency offered by fog computing. As a result, many telecommunications providers and service operators are integrating fog and edge computing into their infrastructures to better support bandwidth-intensive, time-critical applications at lower cost and with enhanced performance (Mehta et al., 2016).

To clearly understand how fog computing helps overcome the limits of traditional cloud systems and supports the needs of modern IoT applications, it is helpful to look at its structure. The next section explains the fog computing architecture and shows how computing tasks are shared between the cloud, fog, and edge layers for faster and more efficient processing.

2. FOG COMPUTING ARCHITECTURE: A LAYERED MODEL FOR EFFICIENT RESOURCE MANAGEMENT

Fog computing is organized into a structured, layered architecture that enables efficient task handling and better use of available resources. This design plays a crucial role in supporting the growing needs of Internet of Things (IoT) applications, where rapid processing and minimal delay are essential. The architecture typically includes three main layers: the Cloud Layer, the Fog Layer, and the Edge Layer as shown in figure 1 below: -

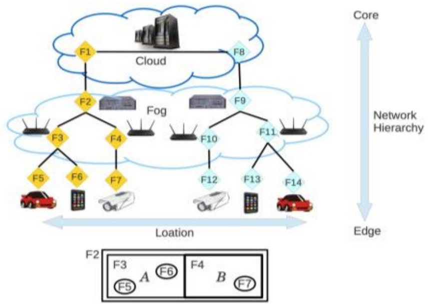

Figure 1 Fog Computing Architecture (Hong et al., 2013)

At the top of this architecture is the Cloud Layer, which represents centralized data centers like nodes F1 and F8. These cloud servers offer high processing power and extensive storage, making them suitable for large-scale data analytics, decision-making across systems, and long-term storage. However, since cloud servers are usually located far from end-user devices, sending data back and forth can cause delays. This makes cloud computing less suitable for time-sensitive tasks, such as medical monitoring, traffic control, or autonomous systems.

The middle layer, known as the Fog Layer, bridges the gap between the distant cloud and nearby devices. It includes fog nodes like F2, F4, F9, F10, and F11, which act as local processing units or gateways. These nodes have moderate computing capabilities and are often placed in network infrastructure such as base stations or routers, closer to the source of data. Their role is to process incoming data locally—applying basic filtering or quick analytics—and forward only essential information to the cloud. This approach helps reduce both delay and bandwidth usage.

The diagram further illustrates how fog nodes are organized. For instance, F2 is connected to F3 and F4, which then manage additional nodes like F5 to F7. Similarly, F9 connects to F10 and F11, which support nodes F12 to F14. This tiered setup allows computing tasks to be handled closer to where data is created, which is especially useful for applications that require immediate action. At the bottom of the hierarchy is the Edge Layer, which includes the actual IoT devices—such as vehicles, surveillance systems, and mobile phones. These devices, linked through nodes like F5, F6, F7, F12, F13, and F14, continuously generate large amounts of data. Many of these devices operate in real time, so sending data to a nearby fog node ensures quicker responses compared to relying solely on distant cloud servers.

A key strength of this architecture is its support for location-aware task allocation. As shown under node F2, edge devices are grouped into sets—Set A and Set B—managed by fog nodes F3 and F4. This allows tasks to be handled locally based on geographical proximity, reducing communication time and improving processing speed. Such task grouping enhances overall system performance, particularly in areas where rapid response is needed. Additionally, the diagram highlights the network hierarchy and communication flow. A vertical arrow on the right shows how processing moves from the centralized cloud down to the distributed fog and edge layers. Data can also travel back up the hierarchy when needed. A horizontal arrow at the base marked "Location" emphasizes how devices are spread out in real-world environments. This physical distribution supports low-latency communication and better bandwidth management (Hong et al., 2013).

In conclusion, this three-tier architecture enables fog computing to handle the limitations of centralized systems. By processing tasks closer to where data is created, the model enhances responsiveness, reduces network burden, and provides a scalable solution for managing IoT-based systems. This makes fog computing a strong foundation for modern applications that demand fast, efficient, and context-aware computing.

Expanding from the architectural setup, one of the key challenges in fog computing is how to wisely assign and manage resources across different layers. This leads us to the important concept of resource provisioning, which helps the system stay responsive, efficient, and able to support the wide variety of needs in IoT applications.

3. RESOURCE PROVISIONING

The term "Resource Provisioning" describes the process of choosing the most appropriate resources (such as cloud resources and resources from nearby fog nodes) from a fog network to meet the needs of IoT users. It satisfies the requests' quality of service standards while improving responsiveness.(El Kafhali & Salah, 2017). Controlling the demand for IoT services requires dynamically allocating the right amount of fog or edge resources to avoid over- or under-provisioning issues while keeping quality of service requirements intact. The ever-changing and unexpected nature of these situations makes resource management a particularly challenging subject to address. (Ghobaei-Arani et al., 2020).

The system's limited virtual and physical resources are managed by a method called resource management so that user tasks may be completed. In Distribution and assignment to meet the needs of applications or services, resource provisioning entails the supply of resources such as computing power, storage space, network bandwidth, and others. Resource scheduling, on the other hand, involves assigning the right resources to tasks or applications based on their needs and urgency. In edge and fog environments, this becomes even more important, as these systems often operate in remote or distributed settings. When resource scheduling is combined with good provisioning techniques, it helps deliver the necessary resources efficiently to where they are needed most.

To get the best performance from fog and edge systems, both scheduling and resource allocation must be handled carefully. One useful approach is dynamic resource provisioning, where resources are assigned based on how much workload is present at any given time. This helps avoid situations where too many or too few resources are assigned to a task. However, because fog and edge environments involve many different types of devices spread across various locations, scheduling and provisioning become more complicated. It takes time to find the best match between tasks and resources, especially since the problem is complex (also known as NP-hard). Scheduling aims to assign jobs to the most suitable resources but doing this efficiently remains a major challenge. It has been shown that meta-heuristic-based techniques may provide almost ideal solutions for certain problems in a reasonable period (Mishra et al., 2018). It seems to be quite challenging to find global optimal solutions for a number of complex multi-modal design challenges in industry and engineering. Due to the possibility of being stuck in local optima, conventional optimization techniques perform poorly in these circumstances. Consequently, the employment of nature-inspired meta-heuristic algorithms is recommended. Because of its quick convergence speed toward the near-optimal solution and capacity to avoid stagnation in local optima, meta-heuristic optimization algorithms have had a major impact on a number of fields in recent decades. Numerous optimization issues are addressed by these methods, particularly those pertaining to engineering. Because of its noteworthy outcomes, meta-heuristic algorithms are now the most popular optimization model for resolving a variety of optimization issues in a respectable amount of time. Meta-heuristics provide a set of solutions that may be used for a variety of situations by assuming certain things about the optimization problems. These algorithms need less processing to find practical answers (Jayasena & Thisarasinghe, 2019).

Optimal resource provisioning in fog computing via the application of metaheuristic approaches is investigated in this review paper, with the aim of meeting the real-time requirements of the Internet of Things. Unfortunately, resource allocation remains a difficult and ever-changing problem in fog computing, even though local processing enables low-latency applications (in contrast to centralized cloud computing). Because they can avoid local optima and adapt to changing circumstances, metaheuristic algorithms, including particle swarm optimization and genetic algorithms, etc., offer efficient, almost flawless solutions. In order to deliver better QoS and system efficiency, this paper emphasizes how well these algorithms enhance resource management for IoT.

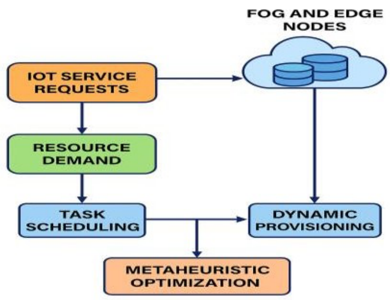

Figure 2 Conceptual Framework linking Resource Provisioning with Fog Computing using Metaheuristic Techniques

The interaction between IoT service demands and fog computing infrastructure, showing how real-time task scheduling and resource provisioning are supported through metaheuristic optimization to ensure efficient and responsive system performance is illustrated in figure 2 above. With this framework explained, the next section discusses why this study is important and points out the main problems in current fog computing systems that need better ways to manage resources.

4. MOTIVATION AND RESEARCH GAPS

When devices and clients make requests for applications and services via the Internet, resource provisioning is a method for allocating storage, networking, and computer resources. Due to a shortage of memory, storage capacity, and processing power, current sensors and gateways cause communication delays, which might be worse in densely populated regions or even farther away from sensors (Kobo et al., 2017). Due to the large bandwidth needs, it becomes difficult to upload all obtained data to the cloud, making centralized solutions unsuitable for Web of Things. Fog computing reduces data transmission to the cloud by doing processing and analysis locally. (Samie et al., 2016).

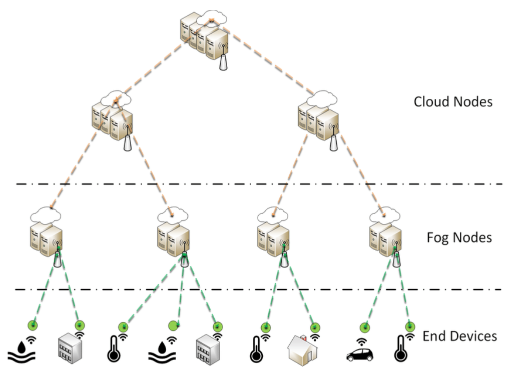

Figure 3 High-level view of fog computing (Kobo et al., 2017).

The overall design of the Fog Computing system is shown in Figure 3. The main purpose of wireless gateways connected to a fog layer via several fog nodes (FNs) is to facilitate communication between end devices, sensors, and actuators, as opposed to a central cloud system. Afterwards, CNs are used for communicating with the cloud layer. Having said that, it is still early days for actual applications of fog computing ideas, and resource allocation for fog computing systems continues to provide a number of challenge (Santos et al., 2019).

The number of resource provisioning systems especially designed for fog computing remains fairly restricted to date. Hong et al. provide an algorithm for programmatic resource allocation that incorporates a straightforward approach dependent on workload thresholds; put otherwise, in the event of a certain fog cell is used more than a predefined amount, another fog cell is purchased (Hong et al., 2013). Aazam and Huh proposed an advanced resource supply system predicated on the anticipation of resource needs (Aazam & Huh, 2015a; Aazam & Huh, 2015b). Resource allocation is executed proactively throughout the system's design phase. This methodology relies on cost optimisation, with resource allocation contingent upon the probabilistic variations in user demand, service kinds, and pricing models. The parameters of the cost function are established during the negotiation of the contract between the user and the supplier. The methodology also considers user incentives and motivational factors. However, Skarlat et al.'s study addressed dynamic architectural changes inside a fog colony by providing runtime resource allocation (Skarlat et al., 2020).

Despite ongoing research on fog computing resource provisioning, there is still a large gap in the creation of systems that make efficient use of metaheuristic techniques. The limited number of resource provisioning systems developed specifically for fog computing highlights the need for innovative ideas capable of satisfying the particular challenges of this technology.

Previous studies, such as those by Hong et al. and Aazam and Huh, have introduced improved techniques for resource allocation in fog computing (Hong et al., 2013; Aazam & Huh, 2015a; Aazam & Huh, 2015b). However, many of these approaches fall short when it comes to exploring metaheuristic methods in detail. Often, these studies focus on fixed system setups or only address limited aspects of resource management. As a result, there has been little attention given to dynamic and diverse resource allocation, which is a key challenge in fog environments (Skarlat et al., 2020).

This gap in the literature highlights the need for a more focused analysis of how metaheuristic techniques can be applied to improve resource provisioning in fog computing. While earlier reviews have discussed the use of metaheuristics in general optimization or edge computing, they rarely address the specific needs of fog systems—such as managing different types of resources close to the network edge or meeting the needs of applications that are sensitive to delays.

The growing role of fog computing in supporting IoT applications makes it essential to explore new ways of managing resources more effectively. Metaheuristic methods offer a promising direction, and this review aims to examine their potential in this context. By focusing specifically on fog computing, this study fills a key gap in past research and provides insights into how these techniques can enhance performance, reduce delays, and better support IoT systems. Figure 4 below shows the flow from user and device demands on the internet to the efficient handling of those demands through fog computing. It highlights how earlier approaches to resource allocation often fell short, particularly in exploring advanced techniques. By incorporating metaheuristic methods, the framework addresses key challenges—improving performance, reducing delays, and offering stronger support for time- sensitive IoT applications.

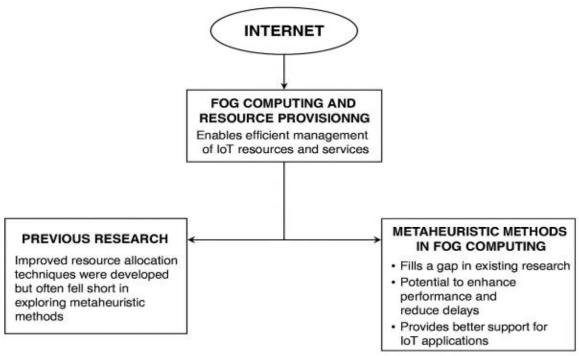

Figure 4 Conceptual Framework linking Fog computing with Metaheuristic Resource Provisioning

Continuing from this background, the next section presents the key contributions of this review. It explains how this study adds to the understanding of resource provisioning in fog computing by exploring the role of metaheuristic techniques in improving efficiency, reducing delays, and supporting IoT systems more effectively.

a. Key Contributions of this Review:

1. **Focused Analysis on Fog Computing:** Offers a comprehensive examination of contemporary metaheuristic methods used exclusively for resource provisioning in fog computing, differentiating it from broader optimization research.
2. **Categorization of Metaheuristic Algorithms:** identifies and classifies a variety of metaheuristic methods (including evolutionary algorithms and particle swarm optimization) and evaluates how well they meet fog environment Quality of Service (QoS) requirements.
3. **Evaluation Metrics:** Highlights important performance criteria like latency, energy economy, and scalability utilized in assessing various

approaches, therefore offering a clear standard for further research.

4. **Identification of Challenges and Research Gaps:** Summarizes difficulties and open research topics in resource provisioning for fog computing, with particular focus to the dynamic, diverse character of fog environments.
5. **Future Research Directions:** Based on gaps found in present investigations, recommendations for future research paths help to shape more flexible and strong metaheuristic solutions for fog computing.

The contributions discussed in this section are clearly illustrated in figure 5 below, which provides a visual summary of the key focus areas of the review. The diagram captures the structured effort made to understand the role of metaheuristic techniques in fog computing. It highlights important aspects such as grouping different algorithms, assessing their performance, recognizing current challenges, and suggesting future directions to strengthen research in this area.

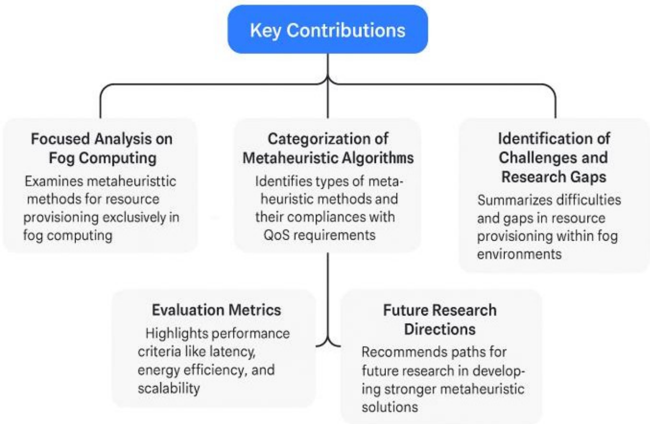

Figure 5 Key Contributions of the review

To build a strong foundation for these insights, it is important to understand how the selected research was gathered and analyzed. The next section outlines the methodology followed in this review, detailing the systematic process used to identify, evaluate, and interpret existing literature on metaheuristic approaches in fog computing

5. REVIEW METHODOLOGY:

The process used to select and assess research articles on metaheuristic this section explains in detail how fog computing handles resource provisioning. It outlines the selection criteria, article search methodology, and research topics that underpin this review.

a. Article Search Strategy:

Several academic databases, such as IEEE Xplore, ScienceDirect, SpringerLink, and Google Scholar, were analyzed in this study. Keywords and phrases include "metaheuristic algorithms," "resource provisioning," "fog computing," "edge computing," and "IoT optimization" helped relevant research be found. Boolean operators helped to reduce the search; only last decade peer-reviewed papers were selected to ensure current insights. The process of review methodology is shown in figure 6 below: -

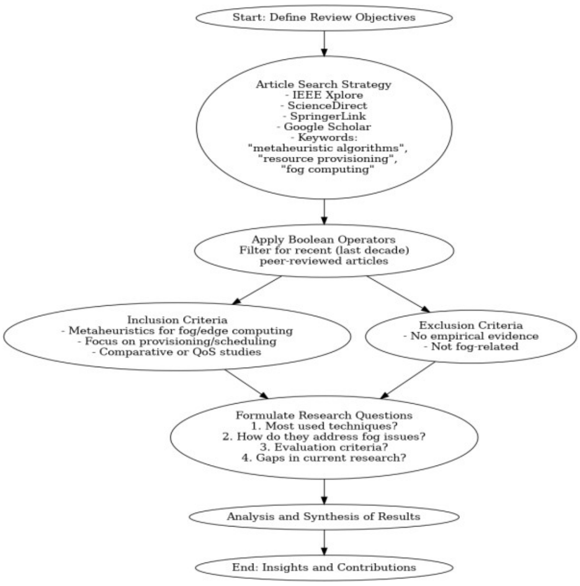

Figure 6 Process of review

b. Inclusion and Exclusion Criteria:
Articles providing metaheuristic approaches to be used in fog or edge computing settings for resource allocation, provisioning, or scheduling were included. Furthermore, taken under consideration were comparative study of many metaheuristic strategies or evaluation of their efficacy in improving quality of service. Articles without empirical evidence or unrelated to fog computing were not included.

c. Research Questions:
To direct the review, the following research questions were created:

1. Which metaheuristic techniques are most often used in fog computing to resource provisioning?
2. In what ways do these methods tackle the unique difficulties of managing resources in fog environments?
3. Which are the key performance criteria used in fog computing to evaluate different solutions?
4. Which areas of existing knowledge need further research?

Based on the defined methodology and identified research questions, the following section systematically reviews existing literature, focusing on the evolution and application of heuristic and metaheuristic techniques to overcome the persistent challenges in resource provisioning within fog computing environments.

6. LITERATURE REVIEW:

The rapid growth of Internet of Things (IoT) applications has made fog computing an important solution for managing real-time data. Unlike cloud computing, fog computing processes data closer to where it's created, which helps reduce delay and supports time-sensitive operations. Still, it's not easy to manage resources in such systems. This is because the amount and type of work can change quickly, edge devices often have limited power, and fog nodes vary widely in their capabilities.
Previous methods of assigning resources often fail to keep up with these challenges. To solve this, researchers have explored smarter techniques that improve task scheduling, save energy, and ensure smooth service performance.
As shown in figure 7 below, the rising demand for faster data processing has caused a shift from centralized cloud systems to be decentralized Fog and Edge computing models. These new systems better support real-time tasks, reduce energy use, and lower delays. To manage these requirements, many resource management approaches have been introduced.

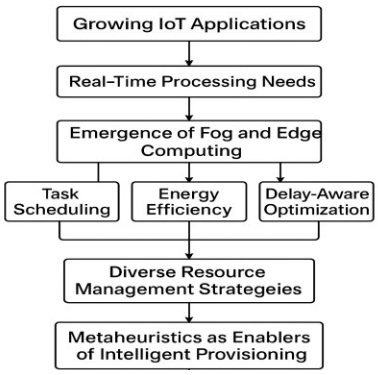

Figure 7 Conceptual framework for Literature Review

Among them, metaheuristic techniques have shown strong results. They help make resource management more intelligent and flexible in fog computing. The sections that follow will explore these areas in detail—starting with Fog and its architecture, resource scheduling, resource provisioning, real time applications, energy-efficient methods, and delay-sensitive approaches, and leading into how metaheuristic techniques play a key role in improving resource allocation.

6.1 Fog Architectures and Scheduling Models

Effective task scheduling in fog computing plays a crucial role in ensuring system responsiveness and efficient use of resources. One solution involves using genetic algorithms to optimize communication between physical devices and their virtual counterparts. This method helps reduce delays by streamlining how tasks are distributed and executed across the fog layer, making it well-suited for latency-sensitive applications (Aakizadeh & Aashi, 2017).
In related efforts, task scheduling and virtual machine (VM) placement have been enhanced through techniques that pair tasks with the most suitable resources. By integrating methods like the Hungarian algorithm with genetic optimization, these models achieve better load distribution and minimize network congestion, especially in dense fog environments (Akintoye & Bagula, 2019).
Another important advancement is the development of task placement strategies that consider both system resource availability and task urgency. These approaches ensure that high-priority tasks are executed promptly without overburdening any single fog node, thus maintaining balance in dynamic environments (Eyckerman et al., 2020).
Similarly, some frameworks apply multi-objective evolutionary algorithms to distribute workloads efficiently while optimizing energy consumption, which is essential for sustaining performance in resource-constrained fog networks (Jijin et al., 2020).
Architecture-level improvements have also been explored. Introducing intermediate fog data centers between end-users and the cloud helps offload traffic from central servers. This layered model improves data access speeds and reduces latency, which is particularly beneficial in mobile edge computing scenarios (Mehta et al., 2016).
For IoT devices with limited computational capacity, offloading intensive tasks to nearby fog nodes significantly enhances performance and battery life. Such task offloading strategies are vital for ensuring smooth operations without overwhelming low-power devices (Samie et al., 2016).

In adaptive environments, some models enable fog nodes to form dynamically, activating only when needed based on real-time demand. This on-demand fog formation approach minimizes idle resource usage and supports elasticity in service delivery (Sami & Mourad, 2020). To further streamline resource provisioning, predictive mathematical models like FogQN have been introduced. These models allow systems to forecast future resource needs and make autonomous, informed decisions about task assignment and load balancing (Tadakamalla & Menasce, 2018, 2019). Table 1 below presents a consolidated view of various research efforts aimed at improving fog computing architectures and scheduling strategies. These studies explore optimization models, resource-aware algorithms, and dynamic deployment methods that enhance the efficiency, responsiveness, and adaptability of fog environments, particularly in real-time and resource- constrained scenarios.

Table 1 Summary Table of Fog Computing Architecture and scheduling strategies

Study	Method/Model Used	Focus Area	Key Contribution
Aakizadeh & Aashi (2017)	Genetic Algorithm	Physical-virtual fog communication optimization	Reduced communication delay between fog nodes and improved task distribution
Akintoye & Bagula (2019)	Hungarian Algorithm + GA	VM placement and scheduling	Enhanced load balancing and reduced network traffic
Eyckerman et al. (2020)	Priority-aware task placement	Fog resource utilization and urgency handling	Assigned tasks based on urgency and available resources for improved responsiveness
Jijin et al. (2020)	MOEA/D (Multi-Objective Evolutionary Algorithm)	Task assignment and energy optimization	Balanced workload and energy usage across fog nodes
Mehta et al. (2016)	Intermediate data center layer	Mobile edge computing	Offloaded cloud traffic to reduce latency and improve data access speeds
Samie et al. (2016)	Task offloading strategy	Low-power IoT device support	Enabled efficient processing and extended device battery life
Sami & Mourad (2020)	On-demand fog formation	Adaptive fog deployment	Allowed elastic resource provisioning based on real-time demand
Tadakamalla & Menasce (2018, 2019)	FogQN (Analytic predictive model)	Automated fog/cloud resource management	Forecasted workload for better scheduling and self-managed systems

6.2 Resource Management Frameworks and Taxonomies

Understanding how resources are managed in fog computing is crucial for developing efficient systems. Several studies have grouped existing methods into categories or taxonomies to help identify which strategies work best under different conditions. These reviews have also outlined current challenges, such as handling mobility, limited processing power, and service delays, which need to be addressed to improve future systems. (Ghobaei-Arani et al., 2020; Yousefpour et al., 2019)

Some research introduced programming models tailored for fog computing, which make it easier to deploy applications that can scale and respond to changing needs. These models help developers manage resources better across multiple fog nodes, improving responsiveness. (Hong et al., 2013)

Other works have focused on mapping existing resource allocation strategies and pointing out areas that lack development. These studies act as a guide for new research by highlighting where improvements are needed. (Lahmar & Boukadi, 2020)

Fog computing also plays a key role in smart cities. Studies in this area describe how it can support applications like traffic management, healthcare, and public safety by bringing processing power closer to the user. (Zanella et al., 2014)

In addition, software-defined networking (SDN) is being used in fog systems to allow for more flexible control of resources. SDN helps reconfigure the system in real time, which is especially useful when conditions change rapidly. (Kobo et al., 2017). Table 2 below highlights important studies that focus on organizing and improving how resources are managed in fog computing. These works introduce frameworks, taxonomies, and practical models that help make sense of different strategies used in this field. By identifying what has been done and where gaps remain, they offer useful direction for future research. Many of these contributions also present real-world approaches that can support flexible, scalable, and efficient deployments—especially in fast-changing and urban environments.

Table 2 Comparison of Resource Management Frameworks and Taxonomies

Study	Focus Area	Key Contribution
Ghobaei-Arani et al. (2020)	Classification of fog resource management methods	Proposed a detailed taxonomy and outlined current challenges in the fog environment
Yousefpour et al. (2019)	Comprehensive review of fog and edge paradigms	Summarized orchestration, scheduling, and QoS techniques across fog and edge systems

Hong et al. (2013)	Scalable programming for fog-based IoT applications	Introduced the MobileFog model to support efficient deployment across fog nodes
Lahmar & Boukadi (2020)	Mapping of resource allocation techniques	Identified well-researched areas and gaps to guide future fog allocation strategies
Zanella et al. (2014)	Fog computing in smart city applications	Explored how fog supports services like traffic control, healthcare, and public safety
Kobo et al. (2017)	Use of SDN in fog and wireless sensor networks	Highlighted SDN's role in making fog networks more flexible and responsive

6.3 Dynamic Resource Provisioning & Scaling

These studies emphasize dynamic provisioning, elasticity, and adaptation in fog and edge contexts.

Efficient resource provisioning in fog computing is essential to ensure real-time performance and adaptability across varying workloads. One approach focused on using fog-based micro-datacenters to estimate resource demands dynamically and adjust pricing accordingly. This model, developed to support the growing scale of IoT applications, helped in managing resource usage more effectively by adjusting to demand in real time (Aazam & Huh, 2015a, 2015b).

In industrial IoT environments, configuring fog nodes based on workload changes is critical. A solution was proposed to adapt fog node settings by predicting upcoming workloads and adjusting resource configurations. This helped maintain smooth operations and avoid performance issues in fast-changing industrial setups (Chen et al., 2018).

A lightweight autoscaling technique was introduced to handle sudden changes in industrial workloads. The model allowed fog systems to automatically scale resources up or down with minimal overhead. It proved especially useful in environments where resources are limited and fast adjustments are necessary to meet service demands (Tseng et al., 2018).

Scalability in edge-cloud systems was also explored through a method that scales resources efficiently based on load variations. This model helped reduce delays and improved the responsiveness of services during traffic peaks, ensuring better user experiences without overloading the system (Li et al., 2020).

In the context of smart cities, a dynamic provisioning framework allowed real-time deployment of fog resources across urban environments. The system supported a wide range of city applications, adapting quickly to shifting demands and offering a reliable platform for scalable services (Santos et al., 2018, 2019).

Machine learning was also applied to automate the scaling of virtual network functions (VNFs) in fog environments. By predicting traffic trends, the system adjusted resources to maintain service quality while minimizing operational costs. This approach balanced performance with cost-efficiency (Rahman et al., 2018).

To improve healthcare delivery, a fog-based framework named HealthFog was developed. It used deep learning to detect heart conditions and supported real-time decision-making in healthcare applications. This system combined intelligent diagnostics with scalable fog infrastructure to ensure timely and efficient medical services (Tuli et al., 2020).

Lastly, elastic resource provisioning was demonstrated in practical deployments. This method allowed IoT services to quickly adapt to workload changes, ensuring continuous service with minimal delays. The system highlighted the importance of flexible fog-based provisioning in real-world IoT scenarios (Zanni et al., 2018). Table 3 below offers a structured comparison of key research efforts dedicated to dynamic resource provisioning and scaling in fog and edge computing environments. These studies tackle essential challenges such as workload prediction, automatic scaling, service elasticity, and intelligent resource management. Their contributions span across critical domains including industrial systems, healthcare services, and smart city infrastructures, highlighting the diverse strategies employed to enhance performance, adaptability, and service continuity in real-time applications.

Table 3 Comparison of Dynamic Resource Provisioning & Scaling Approaches

Study	Approach Used	Application Area	Key Outcome
Aazam & Huh (2015a, 2015b)	Dynamic pricing and demand estimation in micro-datacenters	IoT scalability	Helped manage resources more effectively by adjusting in real time
Chen et al. (2018)	Adaptive configuration based on workload prediction	Industrial IoT	Improved fog node performance under changing workload conditions
Tseng et al. (2018)	Lightweight autoscaling	Industrial fog environments	Enabled quick scaling with minimal overhead for real-time response
Li et al. (2020)	Load-based resource scaling	Edge-cloud systems	Reduced delays and enhanced responsiveness during load fluctuations
Santos et al. (2018, 2019)	Dynamic fog provisioning framework	Smart cities	Supported real-time service deployment in large-scale urban applications
Rahman et al. (2018)	ML-based VNF auto-scaling	Fog service management	Maintained quality of service while cutting down resource cost
Tuli et al. (2020)	Deep learning-based HealthFog framework	Smart healthcare	Enabled quick health diagnostics using scalable fog infrastructure
Zanni et al. (2018)	Elastic provisioning mechanism	IoT service deployment	Ensured reliable and adaptive service with minimal delay during demand shifts

6.4 Real-Time and Healthcare-Focused Implementations

Fog computing has been especially helpful in healthcare applications. Systems have been built to monitor patient health in real time, allowing faster response and reducing the need for remote cloud servers. (Ben Hassen et al., 2019)

Workload balancing in fog networks is important to keep IoT services stable and responsive. Some studies used optimization methods to ensure tasks are spread evenly across fog nodes. (Skarlat et al., 2020)

In smart cities, multi-level fog systems have been proposed to improve the delivery of content like video streaming or emergency messages. These systems adjust based on user needs and network conditions. (Santos et al., 2020)

Lastly, some frameworks are designed specifically for applications that run entirely at the network edge. These models support high-speed, low-latency services that are easy to scale, making them ideal for real-time fog environments. (Wang et al., 2019)

The following table 4 highlights key research focused on using fog computing in real-time and healthcare applications. These efforts demonstrate how placing data processing closer to users improves service responsiveness, especially in critical domains like health monitoring and smart cities.

Table 4 Summary Table: Real-Time and Healthcare-Focused Implementations

Study	Application Area	Approach Used	Key Contribution
Ben Hassen et al. (2019)	Healthcare (Elderly Monitoring)	Fog-based IoT integration	Enabled continuous health tracking with quicker response, reducing cloud dependency.
Skarlat et al. (2020)	General IoT Services	Task scheduling and load balancing	Improved service stability by evenly distributing tasks across fog devices.
Santos et al. (2020)	Smart City Content Delivery	Multi-tier fog orchestration	Delivered adaptive services like video streaming, adjusting to user needs in real time.
Wang et al. (2019)	Edge Applications	Edge-native application framework	Offered fast and scalable solutions for real-time needs with low-latency processing.

6.5 Energy-Efficient and Delay-Aware Optimization

Reducing energy use without compromising performance is one of the biggest goals in fog computing. Some research has developed models that manage resources in a way that balances energy consumption with the system's workload, helping to avoid unnecessary power use. (Bahreini et al., 2019)

In industrial applications, ensuring that systems are both reliable and timely is essential. Some studies have built resource allocation methods that take into account possible delays or system failures, helping these systems perform consistently even in demanding environments. (Dehnavi et al., 2019)

Others have focused on managing service chains across fog and cloud networks. Their goal is to reduce how long it takes for data to travel between nodes, which is especially important for applications that rely on real-time processing. (Siasi et al., 2020)

Mobile IoT networks also benefit from smarter resource provisioning. Strategies that include energy-saving techniques help extend device battery life while keeping performance steady. (Yao & Ansari, 2019)

In fast-moving networks, like those in vehicles, scalable deployment methods are important for keeping services running smoothly. Some solutions focus on making sure these networks stay connected and responsive. (Tonini et al., 2019)

There are also designs that use virtualized fog resources to schedule tasks more efficiently, helping save energy while still supporting real-time needs. (Vinueza Naranjo et al., 2018). Table 5 below presents a summary of studies that explore how to make fog computing more energy-efficient and responsive. These works focus on balancing power usage with system demands, ensuring reliable performance, and improving data flow—especially in settings where quick responses are crucial, such as industrial systems, mobile networks, and real-time IoT applications.

Table 5 Comparative analysis of Energy-Efficient and Delay-Aware Optimization techniques

Study	Approach	Application Area	Main Findings
Bahreini et al. (2019)	Energy-aware resource management	General fog computing	Helped lower energy use by aligning power consumption with the system's workload
Dehnavi et al. (2019)	Reliability-based provisioning model	Industrial applications	Improved consistency by handling delays and system failures effectively
Siasi et al. (2020)	Delay-aware service function chaining	Fog–cloud communication	Reduced the time it takes for data to move between services
Yao & Ansari (2019)	Energy-saving strategy for mobile devices	IoT networks	Helped extend battery life while keeping service quality steady
Tonini et al. (2019)	Scalable deployment model	Vehicular networks	Ensured reliable performance even in fast-moving, changing environments
Vinueza Naranjo et al. (2018)	Energy-efficient task scheduling using virtualization	Real-time IoT services	Improved scheduling and energy use while meeting real-time application needs

6.6 Metaheuristic Techniques for Fog Resource Allocation

These works explore the application and evolution of metaheuristic algorithms for optimizing fog resource provisioning.

A whale optimization algorithm (WOA) was proposed to improve task scheduling in fog computing environments, particularly in smart healthcare systems. This approach successfully reduced energy consumption and task execution time compared to traditional methods like Particle Swarm Optimization (PSO) and Shortest Job First (SJF). The technique proved highly effective in dynamic and time-sensitive scenarios, ensuring better quality of service for critical healthcare tasks (Jayasena & Thisarasinghe, 2019).

A multi-objective resource allocation strategy combining binary PSO, the Bat algorithm, and hybrid metaheuristics was designed for industrial fog computing. The model effectively balanced energy consumption and delay, demonstrating that combining algorithms can lead to more efficient service allocation in real-time industrial IoT systems (Mishra et al., 2018).

To manage resources in cloud–fog–edge environments supporting cyber–physical–social systems, a hybrid approach based on fuzzy clustering and the Flower Pollination Algorithm (FCM-FPA) was introduced. This method adapted well to varying system loads, improving both response time and overall resource utilization, especially in dynamic real-time applications (Porkodi et al., 2020).

A comparative review of metaheuristic techniques for minimizing service-level agreement (SLA) violations in cloud and fog systems revealed that hybrid algorithms generally perform better than single strategies. The analysis emphasized the importance of adaptive, scalable methods in handling fluctuating workloads and maintaining quality of service (Kumar & Bawa, 2020).

A foundational review categorized key metaheuristic scheduling techniques such as Genetic Algorithms (GA), Ant Colony Optimization (ACO), PSO, and Artificial Bee Colony (ABC), analyzing their strengths and weaknesses in cloud environments. The insights from this work have served as a basis for adapting these techniques to fog computing scenarios where latency and scalability are critical (Kalra, & Singh, 2015).

A hybrid heuristic–stochastic method was presented to enhance population initialization in multi-objective evolutionary algorithms for edge computing. The proposed model aimed to reduce SLA violations and improve task processing efficiency by creating diverse and high-quality initial populations, especially in resource-constrained environment (Adyson et al., 2020).

A real-time healthcare monitoring system for diabetic patients was developed using a hybrid decision-making framework based on the VIKOR technique. By integrating fog computing with IoT and soft computing methods, this system significantly improved medical response time and service reliability, showcasing the potential of fog in critical healthcare applications (Abdel-Basset et al., 2019).

Table 6 below provides an overview of key studies that explore heuristic and metaheuristic techniques for enhancing resource provisioning in fog and edge computing environments. These efforts aim to improve task scheduling, reduce energy consumption, and meet service-level agreements more effectively. Each study applies a different optimization approach—such as Whale Optimization, fuzzy logic, or hybrid models—tailored to real-world domains like healthcare, industrial IoT, and cloud–fog systems. Together, they offer practical insights and strategies that can guide future developments in managing complex and dynamic fog-based infrastructures.

Table 6 Comparison of Metaheuristic Techniques Applied in Fog Computing

Study	Technique Used	Focus Area	Key Outcome
Jayasena & Thisarasinghe (2019)	Whale Optimization Algorithm (WOA)	Smart healthcare task scheduling	Reduced energy use and task time; performed better than PSO and SJF methods
Mishra et al. (2018)	Binary PSO, Bat algorithm, and hybrid combinations	Industrial fog environments	Balanced energy efficiency and delay using multiple optimization approaches
Porkodi et al. (2020)	Fuzzy clustering with Flower Pollination Algorithm (FCM-FPA)	Cloud–fog–edge resource allocation	Improved response time and handled changing system loads efficiently
Kumar & Bawa (2020)	Review of hybrid metaheuristic techniques	SLA management in cloud and fog systems	Highlighted that combining algorithms is more effective in managing workloads
Kalra & Singh (2015)	Overview of GA, ACO, PSO, ABC scheduling methods	Cloud computing (relevant for fog systems)	Provided useful insights for adapting algorithms in fog environments
Adyson et al. (2020)	Heuristic and stochastic-based initialization	Task processing in edge computing	Improved starting solution quality, leading to better resource management
Abdel-Basset et al. (2019)	Multi-criteria decision model (VIKOR-based)	Health monitoring using fog and IoT	Boosted response speed and reliability in monitoring diabetic patients

Following the detailed literature survey, it is appropriate to explore the advantages of metaheuristic techniques in resource provisioning for fog computing. The next section highlights their key benefits and practical applications in this domain.

7. ADVANTAGES OF METAHURISTICS IN RESOURCE PROVISIONING:

Resource provisioning is the process of effectively assigning resources to satisfy needs in a variety of fields, including disaster management, cloud computing, and telecommunications. In this regard, metaheuristic algorithms are useful optimization tools that help businesses become more productive and efficient. This section outlines the many benefits of using metaheuristics in resource provisioning, such as their capacity to provide adequate solutions, their adaptability in solving various issues, their efficiency in managing challenging situations, and their potential for practical implementation.

The ability of metaheuristic algorithms to identify adequate solutions in a fair amount of time is one of their main benefits when it comes to resource provisioning. Metaheuristics like Genetic Algorithms (GA) and Simulated Annealing (SA) can effectively produce solutions suitable for real-world applications without the need for exhaustive searches, in contrast to conventional optimization techniques that might have trouble with NP-hard problems (Kalra & Singh, 2015). For example, metaheuristics can rapidly reach satisfactory configurations that meet the requirements of network optimization, where the objective is to maximize throughput while minimizing delay.

Because metaheuristic algorithms are not problem-specific, they can be readily applied to a wide range of situations and domains. In resource provisioning, where many applications may impose unique requirements and limits, this flexibility is essential. As shown in Table 2, a variety of metaheuristics can be used to solve a wide range of optimization issues, helping industries from healthcare This flexibility is especially helpful in situations where resource needs change, like during resource management to transportation logistics. Efficiency is eventually increased by the acceptance and integration of a single algorithm into pre-existing systems, which is made easier by its versatility.

Managing complex and dynamic ecosystems with unpredictable needs and fluctuating resource availability is a common task for resource supply(Yousefpour, Fung, et al., 2019). By using exploration and exploitation techniques that strike a balance between broad search capabilities and solution fine-tuning, metaheuristics are excellent at negotiating these difficulties. To converge toward a near-optimal solution while maintaining some degree of randomness to prevent local optima, Particle Swarm Optimization (PSO), for instance, modifies its methodology based on the aggregate actions of numerous agents emergencies or system failures.

The strengths of metaheuristic techniques—such as their flexibility, speed, and ability to handle complex situations—are clearly shown in figure 8 below**,** which presents the main advantages these methods bring to resource provisioning.

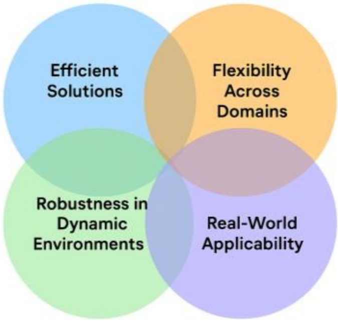

Figure 8 Advantages of Metaheuristic Techniques

8. CONCLUSION

This study underlines the growing importance of metaheuristic techniques in solving the challenges of resource management in fog computing. As applications that demand fast responses and high processing power—like those in the Internet of Things (IoT)—continue to expand, there is an urgent need for efficient and flexible resource allocation. The reviewed approaches show that heuristic and metaheuristic methods can successfully optimize key performance factors such as service quality, cost, delay, and energy use. Metaheuristics, in particular, stand out for their ability to adapt to the changing and limited-resource environment of fog systems. While the results so far are encouraging, there is still plenty of room to improve these techniques. Future efforts should aim to make them more effective for real-time, secure, and large-scale systems. Building and testing these methods in practical, real-world environments will also be important. Continued research and innovation in this area will help create more reliable, scalable, and energy-efficient fog–IoT systems that are ready to meet the evolving needs of modern technology.

REFERENCES

[1] Aakizadeh, A. A., & Aashi, A. A. (2017). Distribution of virtual devices on the fog for delay and araffic aeduction. 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), 492–496.

[2] Aazam, M., & Huh, E.-N. (2015a). Dynamic resource provisioning through fog micro datacenter. 2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), 105–110.

[3] Aazam, M., & Huh, E.-N. (2015b). Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT. 2015 IEEE 29th International Conference on Advanced Information Networking and Applications, 687–694.

[4] Abdel-Basset, M., Manogaran, G., Gamal, A., & Chang, V. (2019). A novel intelligent medical decision support model based on soft computing and

IoT. IEEE Internet of Things Journal, 7(5), 4160–4170.
[5] Akintoye, S. B., & Bagula, A. (2019). Improving quality-of-service in cloud/fog computing through efficient resource allocation. Sensors, 19(6), 1267.
[6] Baghban, H., Huang, C. Y., & Hsu, C. H. (2020). Resource provisioning towards OPEX optimization in horizontal edge federation. Computer Communications, 158(March), 39–50. https://doi.org/10.1016/j.comcom.2020.04.009
[7] Bahreini, T., Badri, H., & Grosu, D. (2019). Energy-aware capacity provisioning and resource allocation in edge computing systems. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11520 LNCS(June), 31–45. https://doi.org/10.1007/978-3-030-23374-7_3
[8] Ben Hassen, H., Dghais, W., & Hamdi, B. (2019). An E-health system for monitoring elderly health based on Internet of Things and Fog computing. Health Information Science and Systems, 7(1), 24.
[9] Chen, L., Zhou, P., Gao, L., & Xu, J. (2018). Adaptive Fog Configuration for the Industrial Internet of Things. IEEE Transactions on Industrial Informatics, 14(10), 4656–4664. https://doi.org/10.1109/TII.2018.2846549
[10] Chiang, M., & Zhang, T. (2016). Fog and IoT: An overview of research opportunities. IEEE Internet of Things Journal, 3(6), 854–864.
[11] Dehnavi, S., Faragardi, H. R., Kargahi, M., & Fahringer, T. (2019). A reliability-aware resource provisioning scheme for real-time industrial applications in a Fog-integrated smart factory. Microprocessors and Microsystems, 70(May), 1–14. https://doi.org/10.1016/j.micpro.2019.05.011
[12] El Kafhali, S., & Salah, K. (2017). Efficient and dynamic scaling of fog nodes for IoT devices. The Journal of Supercomputing, 73(12), 5261–5284. https://doi.org/10.1007/s11227-017-2083-x
[13] El Kafhali, S., & Salah, K. (2019).] S. El Kafhali, K. Salah, Performance modelling and analysis of Internet of Things enabled healthcare monitoring systems, IET Netw. 8 (1) (2018) 48–58. IET Networks, 8(1), 48–58. https://doi.org/10.1049/iet-net.2018.5067
[14] Eyckerman, R., Mercelis, S., Marquez-Barja, J., & Hellinckx, P. (2020). Distributed task placement in the fog: a positioning paper. Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2019) 14, 671–683.
[15] Ghobaei-Arani, M., Souri, A., & Rahmanian, A. A. (2020). Resource management approaches in fog computing: a comprehensive review. Journal of Grid Computing, 18(1), 1–42.
[16] Hong, K., Lillethun, D. J., Ramachandran, U., Ottenwälder, B., & Koldehofe, B. (2013). Mobile Fog: A Programming Model for Large–Scale Applications on the Internet of Things. MCC'13 : Proceedings of the 2nd, 2013 ACM SIGCOMM Workshop on Mobile Cloud Computing : August 12, 2013, Hong Kong, China, 15–20. https://doi.org/10.1145/2491266.2491270
[17] Jayasena, K. P. N., & Thisarasinghe, B. S. (2019). Optimized task scheduling on fog computing environment using meta heuristic algorithms. 2019 IEEE International Conference on Smart Cloud (SmartCloud), 53–58.
[18] Jijin, J., Seet, B.-C., & Chong, P. H. J. (2020). Multi-objective optimization of task-to-node assignment in opportunistic fog RAN. Electronics, 9(3), 474.
[19] Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. Egyptian Informatics Journal, 16(3), 275–295. https://doi.org/https://doi.org/10.1016/j.eij.2015.07.001
[20] Kobo, H. I., Abu-Mahfouz, A. M., & Hancke, G. P. (2017). A survey on software-defined wireless sensor networks: Challenges and design requirements. IEEE Access, 5, 1872–1899.
[21] Kumar, A., & Bawa, S. (2020). A comparative review of meta-heuristic approaches to optimize the SLA violation costs for dynamic execution of cloud services. Soft Computing, 24(6), 3909–3922.
[22] Lahmar, I. Ben, & Boukadi, K. (2020). Resource allocation in fog computing: A systematic mapping study. 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC), 86–93.
[23] Li, C., Bai, J., & Luo, Y. (2020). Efficient resource scaling based on load fluctuation in edge-cloud computing environment. The Journal of Supercomputing, 76(9), 6994–7025. https://doi.org/10.1007/s11227-019-03134-8
[24] Mehta, A., Tärneberg, W., Klein, C., Tordsson, J., Kihl, M., & Elmroth, E. (2016). How beneficial are intermediate layer data centers in mobile edge networks? 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS* W), 222–229.
[25] Mishra, S. K., Puthal, D., Rodrigues, J. J. P. C., Sahoo, B., & Dutkiewicz, E. (2018). Sustainable service allocation using a metaheuristic technique in a fog server for industrial applications. IEEE Transactions on Industrial Informatics, 14(10), 4497–4506.
[26] Peng, L., Dhaini, A. R., & Ho, P. H. (2018). Toward integrated Cloud–Fog networks for efficient IoT provisioning: Key challenges and solutions. Future Generation Computer Systems, 88, 606–613. https://doi.org/10.1016/j.future.2018.05.015
[27] Porkodi, V., Singh, A. R., Sait, A. R. W., Shankar, K., Yang, E., Seo, C., & Joshi, G. P. (2020). Resource provisioning for cyber–physical–social system in cloud-fog-edge computing using optimal flower pollination algorithm. IEEE Access, 8, 105311–105319.
[28] Rahman, S., Ahmed, T., Huynh, M., Tornatore, M., & Mukherjee, B. (2018). Auto-Scaling VNFs Using Machine Learning to Improve QoS and Reduce Cost. IEEE International Conference on Communications, 2018-May(October 2020), 1–6. https://doi.org/10.1109/ICC.2018.8422788
[29] Rahman, T., Yao, X., Tao, G., Ning, H., & Zhou, Z. (2019). Efficient Edge Nodes Reconfiguration and Selection for the Internet of Things. IEEE Sensors Journal, 19(12), 4672–4679. https://doi.org/10.1109/JSEN.2019.2895119
[30] Sami, H., & Mourad, A. (2020). Dynamic on-demand fog formation offering on-the-fly IoT service deployment. IEEE Transactions on Network and Service Management, 17(2), 1026–1039.

[31] Samie, F., Tsoutsouras, V., Bauer, L., Xydis, S., Soudris, D., & Henkel, J. (2016). Computation offloading and resource allocation for low-power IoT edge devices. 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), 7–12.

[32] Santos, H., Alencar, D., Meneguette, R., Rosário, D., Nobre, J., Both, C., Cerqueira, E., & Braun, T. (2020). A multi-tier fog content orchestrator mechanism with quality of experience support. Computer Networks, 177(May), 107288. https://doi.org/10.1016/j.comnet.2020.107288

[33] Santos, J., Wauters, T., Volckaert, B., & Turck, F. De. (2018). Towards Dynamic Fog Resource Provisioning for Smart City Applications. 14th International Conference on Network and Service Management, CNSM 2018 and Workshops, 1st International Workshop on High-Precision Networks Operations and Control, HiPNet 2018 and 1st Workshop on Segment Routing and Service Function Chaining, SR+SFC 2, 290–294.

[34] Santos, J., Wauters, T., Volckaert, B., & Turck, F. De. (2019). Resource provisioning in fog computing: From theory to practice†. Sensors (Switzerland), 19(10), 1–25. https://doi.org/10.3390/s19102238

[35] Siasi, N., Jasim, M., Aldalbahi, A., & Ghani, N. (2020). Delay-aware SFC provisioning in hybrid fog-cloud computing architectures. IEEE Access, 8, 167383–167396. https://doi.org/10.1109/ACCESS.2020.3021354

[36] Skarlat, O., Schulte, S., Borkowski, M., & Leitner, P. (2020). Resource Provisioning for IoT Services in the Fog. Computer Communications, 161, 109–131. https://doi.org/10.1016/j.comcom.2020.07.028

[37] Tadakamalla, U., & Menasc, D. A. (2018). FogQN : An Analytic Model for Fog / Cloud Computing.

[38] Tadakamalla, U., & Menasce, D. A. (2019). Autonomic resource management using analytic models for fog/cloud computing. Proceedings - 2019 IEEE International Conference on Fog Computing, ICFC 2019, 2, 69–79. https://doi.org/10.1109/ICFC.2019.00018

[39] Tonini, F., Khorsandi, B. M., Amato, E., & Raffaelli, C. (2019). Scalable edge computing deployment for reliable service provisioning in vehicular networks. Journal of Sensor and Actuator Networks, 8(4). https://doi.org/10.3390/jsan8040051

[40] Tseng, F. H., Tsai, M. S., Tseng, C. W., Yang, Y. T., Liu, C. C., & Chou, L. Der. (2018). A Lightweight Autoscaling Mechanism for Fog Computing in Industrial Applications. IEEE Transactions on Industrial Informatics, 14(10), 4529–4537. https://doi.org/10.1109/TII.2018.2799230

[41] Tuli, S., Basumatary, N., Gill, S. S., Kahani, M., Arya, R. C., Wander, G. S., & Buyya, R. (2020). HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments. Future Generation Computer Systems, 104, 187–200.

[42] Vinueza Naranjo, P. G., Baccarelli, E., & Scarpiniti, M. (2018). Design and energy-efficient resource management of virtualized networked Fog architectures for the real-time support of IoT applications. Journal of Supercomputing, 74(6), 2470–2507. https://doi.org/10.1007/s11227-018-2274-0

[43] Wang, J., Feng, Z., George, S., Iyengar, R., Pillai, P., & Satyanarayanan, M. (2019). Towards scalable edge-native applications. Proceedings of the 4th ACM/IEEE Symposium on Edge Computing, 152–165.

[44] Yao, J., & Ansari, N. (2019). QoS-Aware Fog Resource Provisioning and Mobile Device Power Control in IoT Networks. IEEE Transactions on Network and Service Management, 16(1), 167–175. https://doi.org/10.1109/TNSM.2018.2888481

[45] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., & Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. Journal of Systems Architecture, 98, 289–330.

[46] Yousefpour, A., Patil, A., Ishigaki, G., Kim, I., Wang, X., Cankaya, H. C., Zhang, Q., Xie, W., & Jue, J. P. (2019). FOGPLAN: A lightweight QoS-aware dynamic fog service provisioning framework. IEEE Internet of Things Journal, 6(3), 5080–5096.

[47] Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). Internet of things for smart cities. IEEE Internet of Things Journal, 1(1), 22–32.

[48] Zanni, A., Forsstrom, S., Jennehag, U., & Bellavista, P. (2018). Elastic Provisioning of Internet of Things Services Using Fog Computing: An Experience Report. Proceedings - 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, MobileCloud 2018, 2018-Janua, 17–22. https://doi.org/10.1109/MobileCloud.2018.00011