Applied Computer Systems
ISSN 2255-8691 (online)
ISSN 2255-8683 (print)
December 2021, vol. 26, no. 2, pp. 139–149
https://doi.org/10.2478/acss-2021-0017
https://content.sciendo.com

Fog Computing Algorithms: A Survey and Research Opportunities

Shaifali P. Malukani[1*], C. K. Bhensdadia[2]
[1,2] *Department of Computer Engineering, Dharmsinh Desai University, Nadiad, Gujarat, India*

Abstract **– The classic Internet of Things-Cloud Computing model faces challenges like high response latency, high bandwidth consumption, and high storage requirement with increasing velocity and volume of generated data. Fog computing offers better services to end users by bringing processing, storage, and networking closer to them. Recently, there has been significant research addressing architectural and algorithmic aspects of fog computing. In the existing literature, a systematic study of architectural designs is widely conducted for various applications. Algorithms are seldom examined. Algorithms play a crucial role in fog computing. This survey aims to performing a comparative study of existing algorithms. The study also presents a systematic classification of the current fog computing algorithms and highlights the key challenges and research issues associated with them.**

Keywords **– Algorithms, cloud computing, fog computing, Internet of Things (IoT), survey, taxonomy.**

I. Introduction

This Internet of Things (IoT) is a world where everyday objects are connected to the Internet and the data from the local environment is measured, collected, stored and processed. The advent of the IoT and mobile Internet has created an unparalleled flood of the enormous amount of data. Nonetheless, in most situations, resource-constrained IoT systems are not adequate to process or store the generated data directly. The IoT, therefore, requires assistance from a powerful computing paradigm like cloud computing (CC). However, the centralized service model of CC causes increased service delay and bandwidth consumption as the volume of data increases. Additionally, the data exchange through an open network increases the security threat to sensitive data [1]. To overcome the issues, a group of researchers [2] from Cisco introduced Fog Computing (FC) in 2012. FC provides computation and storage facilities at the network edge [3]. Latency-sensitive, bandwidth-efficient and secure computing is supported by getting some of the computing and storage close to the end devices, rather than doing them in the cloud.

Rapidly growing fields of IoT, VANETs and the Internet of Drones create numerous applications that need quick and real-time responses and thus can be benefited by fog computing [4]. Recently, FC has attracted the significant attention of researchers. Basic research in the area can be classified into architectural aspects and algorithmic aspects of FC [5]. Recently, several studies have been conducted considering an architectural point of view. Byers [6] surveyed practical deployment of fog computing considering system and application design, software implementation, security, computing resource management and networking from an architectural point of view. Nahaet at al. [7] identified use cases that can get benefit from fog computing. They also listed the architectural requirements in the context of the fog computing use cases. Aazam and Huh [8] reviewed several existing architectures to identify the research issues in big data-related application execution. Wanget et al. [9] studied existing architectures of cloud/fog/edge for connected vehicles and classified them into the categories of computer-aided and computational enabled architectures. Alli and Alam [10] presented a survey on state-of-the-art fog computing architectures, standards, tools and applications.

Algorithmic aspects are seldom examined in the existing literature. Authors [5] have examined the existing architectures and algorithms and compared them over various criteria. However, a dedicated comprehensive study and comparative analysis of new fog algorithms are required in the rapidly growing field of fog computing. Fog algorithms are essential for the effective utilization of the developed fog architecture and the provision of secure and timely services to various fog computing applications. The relevance of fog computing algorithms and the lack of systematic study of the algorithms are the impetus for this survey. This paper aims to review and analyse recent fog algorithms. Relevant work needs to be structured appropriately because of the diversity of the literature on fog computing algorithms. This study represents a comprehensive study and systematic classification of current literature from an algorithmic point of view. In contrast with [5], this survey compares and analyses the algorithms over category-specific parameters for each of the categories identified for the fog computing algorithms. This survey also highlights the research opportunities and key challenges associated with each category of the fog algorithms.

The paper is organised as follows. Section II introduces the concept of fog computing and associated technologies. Section III discusses the research methodology and Section IV presents the classification and review of the existing algorithms,

[*] Corresponding author's e-mail: shaifalimalukani.ce@ddu.ac.in

Section V discusses the open issues, and Section VI concludes the survey.

II. Fog Computing and Associated Technologies

The Fog is a distributed network that is closely associated with Cloud Computing and the Internet of Things.

A. Fog Computing

Fog computing, a promising extension of CC, brings computing to the edge of the network. It helps develop low latency, bandwidth efficient, geographically distributed and privacy-aware applications. A basic three-tiered structure is comprised of the stratum of the cloud, the Fog Nodes (FNs), and IoT/end users [11]. The IoT layer function includes sensing the local environment and transmitting the sensed data to the fog layer. A large number of FNs are hierarchically organised and widely distributed between cloud and end devices. A resource-rich end device like vehicle, surveillance camera, or a computing node with advanced power like switches, gateways, cellular base stations or core networking router can act as a FN [1]. FNs may facilitate computation or temporary storage services. The cloud layer facilitates extensive computation, analysis and permanent storage.

B. Cloud Computing

Cloud computing provides on-demand services of data storage and computing power. Here, computing is performed on shared resources rather than executing on local servers. It provides enhanced elasticity, a large resource pool, reduced cost and anywhere-anytime access of updated resources to its users. It has three key business models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). IaaS provides essential computing components such as servers, storage and networking as a service. PaaS provides the infrastructure and tools on top of which developers are expected to build applications. SaaS offers remote access to software and its functions.

C. Internet of Things

IoT refers to the giant network of the billions of physical objects connected to the Internet, both gathering and exchanging data. Sensed data are transmitted over the Internet for analysis and storage. IoT applications are seen in several domains, ranging from smart homes to smart cities, personal health care to smart hospitals, smart agriculture to smart industry, and a lot more. Automation of the task without human intervention is possible due to IoT. IoT provides a better quality of life because of improved comfort, convenience and management.

III. Research Methodology

The primary goal of this survey is to study the recent development in fog computing algorithms, structure the diversified research in the area of fog algorithms and identify the research issues and opportunities in the field. To figure out the recent trends, this survey is conducted only on the research published between January 2015 and May 2021. To attain the required results that comply with the goals of this study, a set of keywords has been chosen. Basic keywords for the search process are Fog computing, algorithms, data, computation, security, application, communication, etc. Only papers written in English and published in different scientific conference proceedings and journals have been included. Subjective analysis has been conducted to select the diversified studies to give a broader view of recent developments, current trends, and existing gaps in the subject area. The papers on fog architectures are excluded from our survey efforts as they are better covered in other dedicated surveys. Based on these criteria, 87 algorithms published in various research publications have been selected for this study. All the selected algorithms are published in the reputed databases like Springer (20), IEEE (42), Elsevier (15), ACM (5), and other reputed journals (5). Figure 1 shows a histogram of the total number of research publications reviewed to attain the research goal and the corresponding publication year.

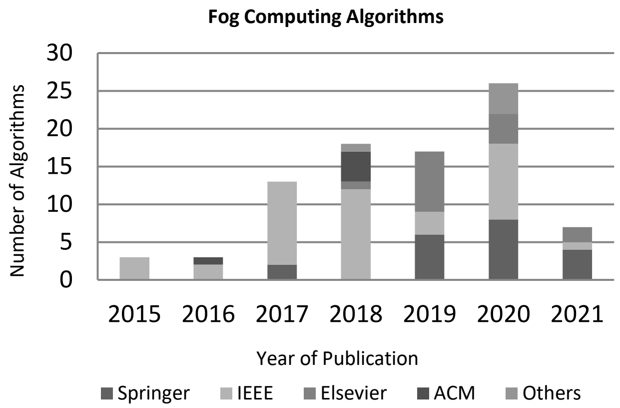


Fig. 1. Histogram of the papers considered for this study.

IV. Fog Computing Algorithms

The diversified research in the area of fog computing can be structured as shown in Fig. 2. A detailed discussion about each category and corresponding subcategories is carried out in this section.

A. Computation Management

Efficient management of computation activity carried out at fog enhances the lifetime and performance of FNs. The year-wise distribution of algorithms considered in this review for computation management is approximately 4 %, 6 %, 11 %, 27 %, 13 %, 33 %, and 6 % across the years 2015 to 2021, respectively.

Offloading

The resource-hungry activities of resource-limited end devices are to be outsourced to FNs. This technique is often called offloading. However, benefits of offloading can be obtained, if it is optimally decided when, where, what and how to offload [12]. Offloading algorithms are developed to decide whether the complete or partial task is to be offloaded (What to offload) and to decide the offloading destination, i.e., to cloud or best suitable FN (Where to offload). The algorithm can suggest a method to upload (How to offload) or how much to offload to fog/cloud (How many tasks/ How much to offload). Algorithms for When to offload decision determine whether or not to offload. Algorithms for who offloading decision determine who will get the limited fog resources from the competing end devices. Resource-consuming activities from a device (D), Edge Device (ED), or Fog (F) can be offloaded to Fog (F), Edge Device (ED), or cloud (C). In the complete type of offloading, the whole task is offloaded to FN, in contrast, in the partial type; a part of the task is offloaded. Offloading algorithms are developed for an interaction between a single user and a fog node or fog node shared by multiple devices. Table I presents the summary of the available offloading techniques.

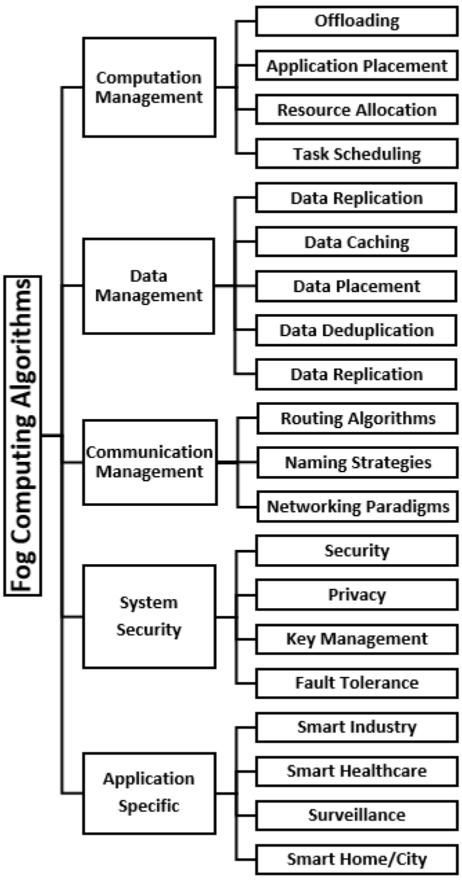


Fig. 2. Classification of Fog Algorithms.

Meng et al. [13] reduce total energy consumption considering the delay constraint to decide cloud-only, cloud-first, fog-only, or fog-first computation. However, their scheme lacks the consideration of other important QoS factors like authentication, confidentiality and availability. Rahbari and Nickray [14] also decide the best destination for offloading under the constraints of authentication, confidentiality and availability along with integrity, capacity, speed and cost in the mobile IoT environment. Their scheme does not implement fault tolerance and the incorporation of machine learning algorithms can improve the performance of the scheme. Fricker et al. [15] presented an analytical model to study the collaboration of neighbouring fog data centers during heavy load to reduce the number of blocked requests without deteriorating the performance. Authors have not analysed the system performance when requests at the big data centre are blocked and are offloaded to small data centres. In an integrated edge-fog environment, Chiti et al. [16] formulate the problem of minimization of the worst completion time and the energy consumption as a matching game with externalities and incomplete preference lists. A post matching procedure is also suggested to ensure stability. The scheme does not consider the mobile fog environment. Privacy and security related aspects are also not discussed for the proposed method. The proposed method by Liu et al. [17] minimizes energy consumption, delay, and payment costs for mobile devices. Scalarization is used to transform the problem into a single-objective optimization. Interior Point Method based algorithm is applied to reduce the accumulated error and improve the calculation accuracy. The method presented by Hassan et al. [18] considers dynamic partitioning of the application to speed up the mobile computation. They use a multilayer perceptron to predict the performance of tasks in a different environment. The schemes proposed in [17] and [18] consider the mobile environment but privacy and security related aspects are not discussed. Shah-Mansouri and Wong [19] view the competition among users for fog resources as a potential game. The aim is to improve the Quality of Experience (QoE) by reducing computation energy and delay. The time for Nash equilibrium may increase exponentially with the number of IoT users. They propose a near-optimal solution to address the issue. However, their scheme has not considered the dynamic arrival of tasks and the issue of joint allocation of computation and communication resources to mobile users. Du et al. [20] reduced the maximum system cost, comprising the delay and energy consumption, considering fairness for all user devices. They formulated the joint optimization of offloading decision, transmit power, bandwidth allocation and resource allocation as a mixed-integer nonlinear programming problem. A semi-definite relaxation and randomization are used for the solution. Their scheme lacks the discussion on user mobility, delay of the user request, multiple FNs and dynamic characteristics of the wireless medium. Jazayeri et al. [21] consider a compromise between the energy consumption and execution time while selecting the best destination for offloading. Their method uses the Monitor-Analyse-Plan-Execute loop and a hidden Markov model auto-scaling offloading to select the offloading destination. Load prediction by machine learning and modeling of renewable energy can enhance the performance of the presented scheme. Jazayeri et al. [22] consider power consumption and execution time of the modules. Their method uses the Monitor-Analyse-Plan-Execute loop and deep reinforcement learning is used to decide the best destination. Baranwal and Vidyarthi [23] propose a game-theoretic approach for computational offloading to decide which tasks to run on local servers and which to run in a fog/cloud environment. They propose a non-cooperative nonzero-sum game with n +1 players where n is the number of tasks and the local server is another player to minimize the cost and to increase the energy efficiency for the delay-sensitive and delay-tolerant tasks. In their scheme, inclusion of other QoS metrics and the use of heuristic and metaheuristic algorithms can yield finer decisions. An improved contract net protocol and beetle antennae search algorithm based task offloading scheme is proposed in fog computing networks by Li et al. [24]. In their approach, they consider cooperation from the FNs to complete a task and model the task offloading process as a bidding process. After receiving the bids, the task node optimally divides the task into subtasks to reduce the delay and energy consumption of the task node. Their work lacks the consideration of the trustworthiness of the neighbour node while offloading the task.

TABLE I
OFFLOADING ALGORITHMS

Ref.	Entities	Type	Decision	No. of devices
[13]	D-F/C	Complete	Where	Single
[14]	D-F	Complete	Where	Multiple
[15]	F-F	Complete	How	Single
[16]	ED-ED/F/C	Complete	Where	Multiple
[17]	D-F/C	Complete	How many	Multiple
[18]	D-F/C	Partition	What	Single
[19]	D-F/C	Complete	Who	Multiple
[20]	D-F/C	Complete	Where	Multiple
[21]	D-F/C	Complete	Where	Multiple
[22]	D-F/C	Complete	Where	Multiple
[23]	D-F/C	Complete	Where	Single
[24]	F-F	Partial	Where, How-many	Single

Application Placement

Research work to map the application components to FNs via a placing algorithm is carried out in the literature.

Skarlat et al. [25] use the exact optimization approach, a greedy first-fit heuristic, and a genetic algorithm to obtain the placement to reduce application response time, deadline violations and cost. Their scheme does not discuss the fault tolerance for the mobile fog landscape. Mahmud et al. [26] use fuzzy logic-based approaches that determine the Rating of Expectation, i.e., priority value of request and Capacity Class Score, i.e., status of FN. They obtain that linearly optimized mapping between the placement requests to a fog instance. However, their scheme is yet to be translated and tested in a real environment. Two backtracking algorithms and two heuristics are proposed by Xia et al. [27] to reduce the average response time and to improve the quality of the placement policy and scalability. Consideration of intrinsic volatility of FN and the migration cost is required. The latency-aware approach proposed by Mahmud et al. [28] optimizes the number of active nodes by relocating the modules. The proposed scheme lacks real-world implementation and customized settings by user and mobility. Guerrero et al. [29] propose a heuristic solution to optimize weighted hope count for the most requested services near users. Their policy experiences degraded performance for less requested applications in the higher workload and more service migrations. Binary tree-based searching is used by Taneja and Davy [30] to find a capable network node with more resources than the application requirement. The method lacks a dynamic strategy to handle network connectivity and node failure at run time. Mouradian et al. [31] use tabu search based algorithm to obtain a sub-optimal solution for the large scale. A nearly optimal solution is obtained for the small-scale system. Their approach does not consider the dynamic arrival of applications. A three-step methodology of Bench-marking, Evaluation, and Testing (BET) is proposed by Venticinque and Amato [32] to satisfy the application QoS requirements. Dynamic optimization of the application deployment is to be considered. Al-Tarawneh [33] uses the non-dominated sorting genetic algorithm to solve the placement problem considering the application criticality and security requirements. However, application trust requirements are not considered. Nashaat et al. [34] consider the QoE and FN computing capability to map the requests to the appropriate FN to maximize user satisfaction. However, their approach yet is to be evaluated for a different type of IoT application and real-time fog environment.

Faticanti et al. [35] perform throughput-based partitioning of application followed by the orchestration of application on region-based infrastructure. Partitioned application is optimally placed using a pseudo gradient approach. The mobility and security requirements of an application are not taken into account in this scheme. Djemai et al. [36] propose a mobility-aware genetic algorithm for service placement in fog architecture considering the energy efficiency of the infrastructure and the application QoS requirement. However, the authors have not studied the scalability of the approach. Baranwal et al. [37] present a modified TOPSIS based policy to map the application module with high application dependent metrics with a computing instance with high computing metrics. However, the proposed placement policy does not incorporate a pricing policy for FNs.

Resource Allocation

The algorithms are proposed to efficiently allocate the limited fog resources to end users. There has been significant research to effectively allocate fog computing (processing and storage resources) and/or fog networking resources (bandwidth of communication). Fig. 3 shows the classification of RA algorithms based on the type of resources they allocate.

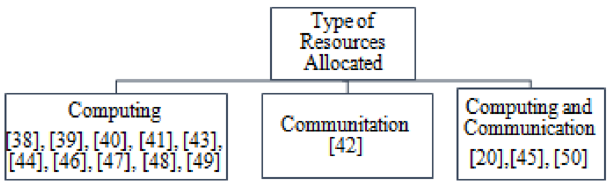


Fig. 3. Resource allocation algorithms.

Zhang et al. [38] model the Stackelberg games to set the price resource and use many-to-many matching to allocate the resources. In the offline auction-based mechanism presented by Jiao et al. [39], two bidding schemes – a constant demand scheme and a multi-demand scheme – are proposed to optimize social welfare. Multi-demand scheme gives higher bid flexibility but gets a sub-optimal solution. Gu et al. [40] formulate computation and radio spectrum allocation problem as a MINLP problem. They model the problem as a student project allocation problem. Matching is obtained to satisfy the delay and data size requirements of users. Network stability and performance are improved by a user-oriented co-operation policy. Jia et al. [41] proposed a Deferred Algorithm based double matching strategy for a double two-sided matching game to maximize cost efficiency. The performance of the scheme is to be tested considering more hierarchy levels in the architecture and utilization of UAV (Unmanned Arial Vehicle). Abedin et al. [42] use AHP to prioritize the requirements of service. The formulated college admission problem is solved by a one-to-many matching game to maximize the QoS. Later, the best fit RA strategy is applied to ensure stability in user association. Du et al. [20] propose a Bisection method for Computation Resource Allocation to optimize the resource allocation. However, their scheme does not consider the impact of user mobility, queuing delay of request, multiple FNs, dynamic nature of the wireless environment. Yin et al. [43] allocate the resources to reduce the task delay using a standard linear programming algorithm of the interior point method in the smart industry environment. Their scheme does not consider the limitations due to finite cloud resources and cloud computing time. Do et al. [44] consider the problem of joint resource allocation and minimizing carbon footprint in the content delivery network. They propose a distributed algorithm based on the proximal algorithm and alternating direction methods of multipliers. The performance of the scheme is yet to be studied in the environment of multiple data centers. Ni et al. [45] propose the priced time Petri net-based resource allocation strategy. The method predicts time and price for the completion of the task. Resources that satisfy the task requirements are allocated to users. The scheme is to be evaluated on different parameters like average completion time, fairness etc. as performance metrics. Luong et al. [46] propose a deep learning based optimal auction approach for resource allocation in blockchain application. Feed-forward neural network is used to construct the assignment and payment systems. The authors have not investigated the scenario with multiple resource providers, the dynamic arrival of devices and the dynamic availability of the resources. Peng et al. [47] consider the multi-attribute-based auctioning method for reasonable resource allocation using price and non-price attributes. Cao et al. [48] tackle the resource allocation problem using modified TwoArch2 with hierarchical clustering. Talaat et al. [49] present a resource allocation method based on reinforcement learning and genetic algorithm. Incoming requests are distributed by observing the network traffic. Naha et al. [50] propose a resource ranking algorithm that ranks resources and resource provisioning is done in a hybrid and hierarchical manner by deadline based prioritizing the processing request. The algorithm does not discuss dynamic changes in resources and resource failures.

Task Scheduling

Task scheduling techniques may schedule the task, the job, or the workflow. The task is defined as the smallest independent entity which cannot be further subdivided. A job can be divided into independent tasks. Workflow is a set of dependent tasks, where the execution of one task depends on the execution of its predecessor. Fig. 4 shows the classification of the studied task scheduling techniques.

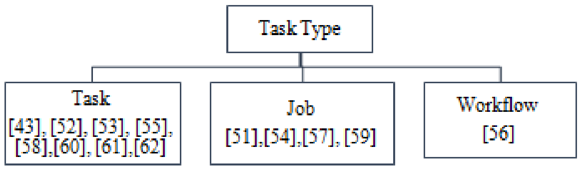


Fig. 4. Task scheduling algorithms.

Workload based destination selection for the bag of tasks is considered by Tychalas and Karatza [51] to reduce the cost. Their scheme has little increase in the response time. In the work by Aladwani [52], Virtual Machines (VMs) with low, medium, and high capability execute low, medium, and high priority tasks, respectively. However, their scheme does not consider the security aspect and location awareness in the selection of VMs. Zeng et al. [53] solve the problem of a minimization of the maximum task completion time using the linear programming relaxation method. Their work lacks the study of the memory management of the system. Yin et al. [43] consider delay and resource requirement of the task and use resource threshold based method to select the destination in the container based smart manufacturing environment. They show that the number of accepted tasks can be increased with a dynamic threshold value. The authors do not consider image placement problems in containers and limitations due to finite cloud resources and cloud computing time. In the method proposed by Bitam et al. [54], the job is decomposed into tasks and bees life algorithm is used for an optimal assignment for each FN to minimize the CPU execution time and allocated memory. Liu et al. [55] propose a scheduling algorithm for load balancing. Consideration of energy consumption and energy-latency trade-off can improvise the performance of the scheme. Pham and Huh [56] discuss a graph-based scheduling policy. Latency based task priority is calculated. A higher priority task is assigned to the processor node with maximum utility. Addition of the more constraints like the budget of the fog provider and deadline can enhance the performance of the system. In the method proposed by Choudhari et al. [57], tasks are prioritized based on tolerance to the delay under deadline constraint. If the task cannot be completed by the FN, it is divided into subtasks and completed by the collaboration of the remaining fog servers and cloud. The scheme lacks in considering the dynamic priority. Zhao et al. [58] consider a wireless fog-enabled multi-tiered content delivery network and the technique for Lyapunov optimization for centralized assignment of access node at the control level. Their scheme does not consider resource management, fading channel statistics, traffic distributions and network dynamics. For latency-critical applications, an optimized shortest job first scheduling algorithm is proposed by Jamil et al. [59]. However, jobs with larger lengths might face starvation. Wang et al. [60] propose an improved genetic firework algorithm that uses the explosion radius detection mechanism of fireworks and characteristics of resources and tasks. The proposed scheme does not consider the energy consumption of fog devices while processing the task. Hosseinioun et al. [61] use the dynamic voltage and frequency scaling technique for energy-aware scheduling. The hybrid invasive weed optimization and culture are used to construct a valid task sequence. In their approach load balancing, trust and privacy are not discussed. Ghanavati et al. [62] present a dynamic fault-tolerant learning automata task scheduling approach for optimizing response time while ensuring reliable execution of the tasks. However, the method does not consider the energy required by the task and available memory.

Objectives for Computation Management

During this study, common objectives of fog algorithms for computation management are identified. The major common objectives are minimizing energy consumption, minimizing latency, minimizing cost, considering deadline and its violations, load balancing, minimizing carbon footprint, considering security aspects, maximizing resource utilization, and maximizing the throughput. Table II and Fig. 5 show the objectives of the studied approaches. However, approaches may have objectives other than those listed that are already discussed in detail in previous sections.

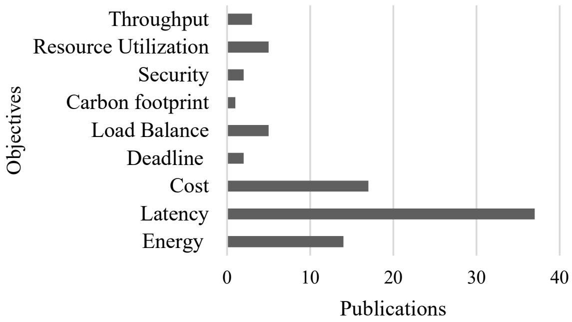


Fig. 5. Resource allocation algorithms.

TABLE II
OBJECTIVES OF COMPUTATION MANAGEMENT

	Publication
Energy consumption	[13], [16], [17], [19]–[24], [48], [51], [59], [61], [62]
Latency	[13],[16]-[27],[29],[31],[34]–[37], [40]–[43], [45], [46], [48],[50]–[60]
Cost	[14], [17], [23], [25], [31], [35], [36], [38]–[41], [45], [47], [50], [51], [57], [58]
Deadline	[25], [28]
Load Balancing	[14], [47], [48], [52], [60]
Carbon footprint	[44]
Security	[14], [33]
Resource Utilization	[28], [30], [32], [34], [45]
Throughput	[35], [42], [58]

The most widely used objective is minimization of latency. Latency minimization is a very broad category that includes minimization of average response latency, transmission time, execution time, queuing time, completion time, or any combination of them. Energy minimization and cost minimization are other popular objectives. While efficient resource utilization, carbon footprint minimization, maximization of throughput and load balancing are important but less popular objectives. Such objectives demand the attention of the researchers.

B. Data Management

Fog receives a tremendous amount of data from associated IoT sensors. Data management at fog becomes important for the effective utilization of stored data in a timely manner. The year-wise distribution of the algorithms considered in this review for the data management is approximately 7 %, 21 %, 14 % 29 %, and 29 % across the years 2015, 2017, 2018, 2019 and 2020, respectively.

Naas et al. [63] solve the data placement problem by obtaining the exact solution via a single integer program. The combinatorial explosion was tested in the large fog infrastructure. Therefore, geo-partitioning based heuristic solution was proposed to reduce solving time. However, the method does not consider workload balance and data flow dependency in geo-partitioned subparts. Naas et al. [64] suggest a divide and conquer based approach to reduce the latency of data transfer from the storage node to the consumer node using a k-way graph partitioning. Huang et al. [65] propose the multi replica data placement model. Their algorithm uses the pruning method to choose a solution with minimum latency. Their work involves trade between performance in reducing overall latency and replica count. Multiple data placement with a budget problem is discussed by Wang and Wu [66]. The authors try to minimize the overall data access latency for a given total budget. Their scheme lacks in considering the different requests with different sizes. The work in [18] uses FC for storage expansion and finds optimal locations for data placement based on frictional linear programming.

Duplicate reports have redundant data and they consume bandwidth and storage [67]. Ni et al. [68] propose a data deduplication scheme based on the BLS and homomorphic signature. In their scheme, all users who generated the data got the reward. The chameleon hash function is used to prevent users from claiming double rewards. Yan et al. [69] present a decision tree and an interval table based scheme to determine the specific server to traverse for duplicate data. Shynu et al. [70] present convergent and Modified Elliptic Curve Cryptography (MECC) algorithm based data deduplication scheme to detect the redundancy at the block level.

Vales et al. [71] propose an energy aware and adaptive distance-based data replication decision policy. The decision is taken based on data popularity, distance to the consumer node from the data node and battery of the node. Berkennou et al. [72] design data popularity based on a migration and replication strategy over their proposed hierarchical multilayer model to improve the response time and minimize the energy consumption. However, their strategies are not mobility aware.

Data caching schemes facilitate caching of popular data at the network edge. In the scheme proposed by Al Ridhawi et al. [73], frequently accessed cloud data are decomposed into blocks and placed into fog storage. The authors propose a user's historical data-based file location prediction technique. Bai et al. [74] jointly optimize the fog storage hit rate and energy consumption using NSGA-II multi-objective optimization algorithm.

Data reduction techniques reduce the amount of data exchanged between the fog and the cloud to reduce bandwidth consumption. Existing data reduction schemes can be majorly classified into compression-based and prediction-based techniques. Fu et al. [75] present compression-based data reduction. They generate a cryptographic hash to represent the chunks of the file. The duplicate chunks are then replaced with fingerprints. All the unique chunks from the file are bundled and compressed together. File-level deduplication is employed for compressed formats. Prediction based data reduction technique proposed by Yu et al. [76] implements Dual Kalman Filter at fog and cloud with identical parameters. Hence, the same predictions at both platforms are obtained when simultaneously triggered. Only measured data out of the prediction range are uploaded to achieve data reduction.

C. Communication Management

New research challenges, including the need for new policies and protocols to facilitate interaction between heterogeneous devices, have been addressed in the literature. The year-wise distribution of the algorithms considered in this review for communication management is approximately 33 %, 17 %, 33 % and 17 % across the years 2017, 2019, 2020 and 2021, respectively.

G´omez-C´ardenas et al. [77] proposed a hash-based naming strategy appropriate to the Fog-cloud environment. Guibert et al. [78] incorporate Content-Centric Network for the efficiency of communication and local storage. Kadhim and Seno [79] present an Energy Efficient Multicast routing protocol based on Software Defined Networks and Fog computing for Vehicular networks called EEMSFV with deadline and bandwidth constraint. Abidoye and Kabaso [80] propose an energy-efficient hierarchical routing algorithm. They minimize the total number of broadcast packets using an ant colony optimization. However, data transmission in the network is susceptible to different attacks and needs some adequate security solutions. Noorani and Seno [81] propose SDN and Fog computing-based Switchable Routing to switch the transmission between VANET infrastructure and the Internet to select the best path for the inter-vehicle transmission. Saito et al. [82] present a Topic-based Data Transmission protocol to deliver messages to target nodes. Their approach reduces the number of messages while the delivery ratio is smaller.

D. Algorithms for Security

Fog nodes might contain sensitive information and the personal information of the users. Therefore, algorithms to maintain the confidentiality, integrity and availability of these data have been developed. The year-wise distribution of the algorithms considered in this review for the system security is approximately 22 %, 11 %, 34 %, 11 % and 22 % across the years 2017 to 2021, respectively.

Hu et al. [83] prevent unauthorized access by storing the legal user's identity. Diffie-Helmen key agreement algorithm is used to generate session keys between fog node and user. An advanced encryption standard is used to encrypt the data. To perform integrity checking, the Secure Hash Algorithm is implemented. Wazid et al. [84] implement a lightweight one-way cryptographic function and bitwise XOR operation for resource constraint IoT devices to achieve secure key management and user authentication. Elliptic curve point multiplication and biometrics fuzzy extractor method is implemented for resource-rich devices and fog servers. Their scheme suffers from clogging attacks and it is improved by Ali et al. [85]. Their scheme obtains the same communication cost as the scheme in [84] but a 24 % increase in computation cost. Lu et al. [86] employ the homomorphic paillier encryption, one-way hash technique, and the Chinese remainder theorem for aggregating IoT data and early filtering at the network edge. Zuo et al. [87] suggest Attribute-Based Encryption with outsourced decryption as a promising solution to the chosen cipher-text attack security attack. Guan et al. [88] ensure the anonymity and authenticity of the device using a pseudonym and pseudonym certificate. Paillier's algorithm is used to ensure data privacy. A lattice-based multi-block and mixed signature scheme is constructed by Wang et al. [89] to obtain incremental authentication of the updated fog data. The proposed incremental authentication scheme resists forgery attacks while keeping public key size and signature length in a reasonable interval. They improve the computing speed of the algorithm by finishing time-consuming computation using parallel computation or pre-computing. Their scheme does not prevent the spread of the old signature while the new signature is created. The scheme proposed by Noura et al. [90] secures data in fog computing by combining the AES-GMAC operation mode with information dispersal over GF(2w). Their cryptographic solution ensures data integrity, availability confidentiality and source authentication. Al-Khafajiy et al. [91] consider the security and privacy concerns among fog nodes during their collaboration. They identify and isolate the malicious fog node by a trust and recommendation based proposed method. Their scheme lacks the consideration of the energy consumption of the fog node during collaboration.

E. Application Specific Algorithms

Lightweight algorithms for fog devices for application specific tasks are developed. The year-wise distribution of the application-specific algorithms considered in this review is approximately 17 %, 33 %, 33 % and 17 % across the years 2018 to 2021, respectively.

Xu et al. [92] use fog computing to identify tumors using a modified and semi-supervised Fuzzy C Means algorithm. Work done by Wan et al. [93] uses fog computing in the smart industry environment to schedule the job on the equipment by employing improvised particle swarm optimization. Vijayakumar et al. [94] used the fog for early detection of mosquito based diseases. The similarity coefficient is used to differentiate disease and fuzzy k nearest neighbour implemented to categorize the users into infected and uninfected classes. Siddharth and Aghila [95] propose a random projection and structural similarity index based light background subtraction algorithm for motion detection. Xu et al. [96] propose a scheme to quickly deploy emergency networking and communication using available devices like routers and mobile devices as fog nodes in case of natural disaster. However, the proposed scheme is yet to be tested for diversity in the equipment and real-world environment. Ali et al. [97] propose a dynamic deep hybrid Spatio-temporal neural network (DHSTNet) to predict highly accurate traffic in every region of a city.

V. Research Issues and Opportunities

Energy efficiency. Power-conscious algorithms in all the fog computing management aspects are required to enhance the battery life of the devices. *Dynamic workload*. For the heterogeneous and mobile fog environment, the dynamic arrival and departure of the requests are not an unpredictable scenario. Policies to manage run time arrival of requests of offloading, service placement, resource allocation, scheduling, or data management can improve user satisfaction. *Mobility*. Wireless sensors, vehicles and UAVs are an integral part of fog computing architecture and they observe a high amount of mobility in an unpredictable pattern. Thus, frequent migration of fog services and user requests is unavoidable in a fog environment. Efficient handover, provision of security and resource management after migration without service interruption are important research issues. *Data dependency*. Placement of the applications considering the data dependency [98] and the amount of data flow exchange among them can enhance the system performance. *SLA violation*. Less service level violations for the allocated resources and schemes for compensation in case of violations are desired. *Revenue and customer satisfaction*. Optimum revenue for service providers considering customer satisfaction for the trade between fog and user is required to make a successful commercial model for fog computing. The flexibility of the automated negotiations for the required resources and offered prices can make the fog system more realistic. *Data clusters*. Performance of data positioning, replicating and caching policies can be improved by grouping the data that are often accessed together and even distribution of grouped data on fog nodes. *Routing*. The routing algorithms are to be developed for mobile FN with limited storage. Fault tolerance and secure routing in the highly dynamic topology of the fog environment are considerable issues in fog architecture. *Fog for more applications*. The fog computing platform can benefit a broad range of applications in the area of augmented reality, content delivery, mobile big data analytics, smart city, smart home, Healthcare, etc. [99]. Lightweight application specific algorithms for all such applications are required.

VI. Conclusion

The importance of fog computing has been realised with the need for real-time bandwidth-efficient, latency-sensitive and secure services for resource-constrained end IoT devices. Several important aspects of fog computing have drawn the attention of the researchers and significant research has been carried out in the field. The study of the existing literature is often carried out from an architectural point of view. Research on algorithmic aspects is often overlooked. This survey presents the comprehensive study of the research that has been performed in the field so far from an algorithmic point of view. First, a taxonomy for existing fog computing algorithms is derived. Later, a comprehensive study and comparative analysis of the fog algorithms are carried out. Moreover, mainstream goals for the computation management algorithms are identified and less popular but important objectives are highlighted in order to draw researchers' attention to them. Finally, significant research issues for each fog algorithm category are identified and a discussion on future directions is carried out in this survey.

References

[1] C. Puliafito, E. Mingozzi, F. Longo, A. Puliafito, and O. Rana, "Fogcomputing for the internet of things: A survey," *ACM Transactions on Internet Technology (TOIT),* vol. 19, no. 2, pp. 1–41, Apr. 2019. https://doi.org/10.1145/3301443

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, Aug. 2012, pp. 13–16. https://doi.org/10.1145/2342509.2342513

[3] R. Mahmud, R. Kotagiri, R.Buyya . "Fog computing: A taxonomy, survey and future directions". in *Internet of everything, Springer*; 2018. p. 103–130.

[4] M. Aazam, M. St-Hilaire, CH. Lung, I. Lambadaris, EN Huh." IoT resource estimation challenges and modeling in fog", in *Fog Computing in the Internet of Things. Springer*; 2018. p. 17–31.

[5] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A comprehensive survey on fog computing: Stateof-the-art and research challenges," *IEEE communications surveys & tutorials*, vol. 20, no. 1, pp. 416–464, 2017. https://doi.org/10.1109/COMST.2017.2771153

[6] C. C. Byers, "Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques for fog-enabled IoT networks," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 14–20, Aug. 2017. https://doi.org/10.1109/MCOM.2017.1600885

[7] R. K. Naha, S. Garg, and A. Chan, "Fog computing architecture: Survey and challenges," *arXiv,* no.1811.09047, 2018.

[8] M. Aazam and E.-N. Huh, "Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT," in *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*. Gwangju, Korea (South), Mar. 2015, pp. 687–694. https://doi.org/10.1109/AINA.2015.254

[9] H. Wang, T. Liu, B. Kim, C.-W. Lin, S. Shiraishi, J. Xie, and Z. Han, "Architectural design alternatives based on cloud/edge/fog computing for connected vehicles," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2349–2377, Sep. 2020. https://doi.org/10.1109/COMST.2020.3020854

[10] A. A. Alli and M. M. Alam, "The fog cloud of things: A survey on concepts, architecture, standards, tools, and applications," *Internet of Things*, vol. 9, Art no. 100177, Mar. 2020. https://doi.org/10.1016/j.iot.2020.100177

[11] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," *Big Data and Cognitive Computing*, vol. 2, no. 2, p. 10, 2018. https://doi.org/10.3390/bdcc2020010

[12] H. Wu, "Multi-objective decision-making for mobile cloud offloading: A survey," *IEEE Access*, vol. 6, pp. 3962–3976, Jan. 2018. https://doi.org/10.1109/ACCESS.2018.2791504

[13] X. Meng, W. Wang, and Z. Zhang, "Delay-constrained hybrid computation offloading with cloud and fog computing," *IEEE Access*, vol. 5, pp. 21355–21367, Sep. 2017. https://doi.org/10.1109/ACCESS.2017.2748140

[14] D. Rahbari and M. Nickray, "Task offloading in mobile fog computing by classification and regression tree," *Peer-to-Peer Networking and Applications*, vol. 13, pp. 1–19, Feb. 2019. https://doi.org/10.1007/s12083-019-00721-7
[15] C. Fricker, F. Guillemin, P. Robert, and G. Thompson, "Analysis of an offloading scheme for data centers in the framework of fog computing," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS),* vol. 1, no. 4, Art no. 16, pp. 1–18, Sep. 2016. https://doi.org/10.1145/2950047
[16] F. Chiti, R. Fantacci, and B. Picano, "A matching game for tasks offloading in integrated edge-fog computing systems," *Transactions on Emerging Telecommunications Technologie*s, vol. 31, no. 2, p. Art no. e3718, Aug. 2020. https://doi.org/10.1002/ett.3718
[17] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 283–294, Dec. 2017. https://doi.org/10.1109/JIOT.2017.2780236
[18] M. A. Hassan, M. Xiao, Q. Wei, and S. Chen, "Help your mobile applications with fog computing," in *2015 12th Annual IEEE International Conference on Sensing, Communication, and Networking – Workshop (SECON Workshops)*. Seattle, WA, USA, June 2015, pp. 1–6. https://doi.org/10.1109/SECONW.2015.7328146
[19] H. Shah-Mansouri and V. W. Wong, "Hierarchical fog-cloud computing for IoT systems: A computation offloading game," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3246–3257, Aug. 2018. https://doi.org/10.1109/JIOT.2018.2838022
[20] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee," *IEEE Transactions on Communications*, vol. 66, no. 4, pp. 1594–1608, 2018. https://doi.org/10.1109/TCOMM.2017.2787700
[21] F. Jazayeri, A. Shahidinejad, and M. Ghobaei-Arani, "A latency-aware and energy-efficient computation offloading in mobile fog computing: A hidden Markov model-based approach," *The Journal of Supercomputing,* vol. 77, no. 5, pp. 4887–4916, 2021. https://doi.org/10.1007/s11227-020-03476-8
[22] F. Jazayeri, A. Shahidinejad, and M. Ghobaei-Arani, "Autonomous computation offloading and auto-scaling the in the mobile fog computing: a deep reinforcement learning-based approach," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 8265–8284, 2021. https://doi.org/10.1007/s12652-020-02561-3
[23] G. Baranwal and D. P. Vidyarthi, "Computation offloading model for smart factory," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 8305–8318, 2021. https://doi.org/10.1007/s12652-020-02564-0
[24] X. Li, Z. Zang, F. Shen, and Y. Sun, "Task offloading scheme based on improved contract net protocol and beetle antennae search algorithm in fog computing networks," *Mobile Networks and Applications*, vol. 25, no. 6, pp. 2517–2526, 2020. https://doi.org/10.1007/s11036-020-01593-5
[25] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner, "Optimized IoT service placement in the fog," *Service Oriented Computing and Applications*, vol. 11, no. 4, pp. 427–443, Oct. 2017. https://doi.org/10.1007/s11761-017-0219-8
[26] R. Mahmud, S. N. Srirama, K. Ramamohanarao, and R. Buyya, "Quality of experience (QoE)-aware placement of applications in fog computing environments," *Journal of Parallel and Distributed Computing*, vol. 132, pp. 190–203, Oct. 2019. https://doi.org/10.1016/j.jpdc.2018.03.004
[27] Y. Xia, X. Etchevers, L. Letondeur, T. Coupaye, and F. Desprez, "Combining hardware nodes and software components ordering-based heuristics for optimizing the placement of distributed IoT applications in the fog," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, Apr. 2018, pp. 751–760. https://doi.org/10.1145/3167132.3167215
[28] R. Mahmud, K. Ramamohanarao, and R. Buyya, "Latency-aware application module management for fog computing environments," *ACM Transactions on Internet Technology (TOIT),* vol. 19, no. 1, Art no. 9, pp. 1–21, Mar. 2018. https://doi.org/10.1145/3186592
[29] C. Guerrero, I. Lera, and C. Juiz, "A lightweight decentralized service placement policy for performance optimization in fog computing," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 6, pp. 2435–2452, 2019. https://doi.org/10.1007/s12652-018-0914-0
[30] M. Taneja and A. Davy, "Resource aware placement of IoT application modules in fog-cloud computing paradigm," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, Lisbon, Portugal, Vfy 2017, pp. 1222–1228. https://doi.org/10.23919/INM.2017.7987464
[31] C. Mouradian, S. Kianpisheh, M. Abu-Lebdeh, F. Ebrahimnezhad, N. T. Jahromi, and R. H. Glitho, "Application component placement in NFV-based hybrid cloud/fog systems with mobile fog nodes," *IEEE Journal on Selected Areas in Communications,* vol. 37, no. 5, pp. 1130–1143, May 2019. https://doi.org/10.1109/JSAC.2019.2906790
[32] S. Venticinque and A. Amato, "A methodology for deployment of IoT application in fog," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 5, pp. 1955–1976, 2019. https://doi.org/10.1007/s12652-018-0785-4
[33] M. A. Al-Tarawneh, "Bi-objective optimization of application placement in fog computing environments," *Journal of Ambient Intelligence and Humanized Computing,* pp. 1–24, Feb. 2021. https://doi.org/10.1007/s12652-021-02910-w
[34] H. Nashaat, E. Ahmed, and R. Rizk, "IoT application placement algorithm based on multi-dimensional QoE prioritization model in fog computing environment," *IEEE Access*, vol. 8, pp. 111 253–111 264, June 2020. https://doi.org/10.1109/ACCESS.2020.3003249
[35] F. Faticanti, F. De Pellegrini, D. Siracusa, D. Santoro, and S. Cretti, "Throughput-aware partitioning and placement of applications in fog computing," *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2436–2450, Dec. 2020. https://doi.org/10.1109/TNSM.2020.3023011
[36] T. Djemai, P. Stolf, T. Monteil, and J.-M. Pierson, "Mobility support for energy and QoS aware IoT services placement in the fog," in *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM),* Split, Croatia, Sep. 2020, pp. 1–7. https://doi.org/10.23919/SoftCOM50211.2020.9238236
[37] G. Baranwal, R. Yadav, and D. P. Vidyarthi, "QoE aware IoT application placement in fog computing using modified-TOPSIS," *Mobile Networks and Applications*, vol. 25, no. 5, pp. 1816–1832, Oct. 2020. https://doi.org/10.1007/s11036-020-01563-x
[38] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining stackelberg game and matching," *IEEE Internet of Things Journal,* vol. 4, no. 5, pp. 1204–1215, Oct. 2017. https://doi.org/10.1109/JIOT.2017.2688925
[39] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Transactions on Parallel and Distributed Systems,* vol. 30, no. 9, pp. 1975–1989, Sep. 2019. https://doi.org/10.1109/TPDS.2019.2900238
[40] Y. Gu, Z. Chang, M. Pan, L. Song, and Z. Han, "Joint radio and computational resource allocation in IoT fog computing," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 8, pp. 7475–7484, Aug. 2018. https://doi.org/10.1109/TVT.2018.2820838
[41] B. Jia, H. Hu, Y. Zeng, T. Xu, and Y. Yang, "Double-matching resource allocation strategy in fog computing networks based on cost efficiency," *Journal of Communications and Networks,* vol. 20, no. 3, pp. 237–246, June 2018. https://doi.org/10.1109/JCN.2018.000036
[42] S. F. Abedin, M. G. R. Alam, S. A. Kazmi, N. H. Tran, D. Niyato, and C. S. Hong, "Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network," *IEEE Transactions on Communications,* vol. 67, no. 1, pp. 489–502, Jan. 2018. https://doi.org/10.1109/TCOMM.2018.2870888
[43] L. Yin, J. Luo, and H. Luo, "Tasks scheduling and resource allocation in fog computing based on containers for smart manufacturing," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 10, pp. 4712–4721, Oct. 2018. https://doi.org/10.1109/TII.2018.2851241
[44] C. T. Do, N. H. Tran, C. Pham, M. G. R. Alam, J. H. Son, and C. S. Hong, "A proximal algorithm for joint resource allocation and minimizing carbon footprint in geo-distributed fog computing," in *2015International Conference on Information Networking (ICOIN),* Cambodia, Mar. 2015, pp. 324–329. https://doi.org/10.1109/ICOIN.2015.7057905
[45] L. Ni, J. Zhang, C. Jiang, C. Yan, and K. Yu, "Resource allocation strategy in fog computing based on priced timed petri nets," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1216–1228, Oct. 2017. https://doi.org/10.1109/JIOT.2017.2709814

[46] N. C. Luong, Y. Jiao, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "A machine-learning-based auction for resource trading in fog computing," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 82–88, Mar. 2020. https://doi.org/10.1109/MCOM.001.1900136
[47] X. Peng, K. Ota, and M. Dong, "Multiattribute-based double auction toward resource allocation in vehicular fog computing," *IEEE Internet of Things Journal,* vol. 7, no. 4, pp. 3094–3103, Apr. 2020. https://doi.org/10.1109/JIOT.2020.2965009
[48] B. Cao, Z. Sun, J. Zhang, and Y. Gu, "Resource allocation in 5G IoV architecture based on SDN and fog-cloud computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3832-3840, June 2021. https://doi.org/10.1109/TITS.2020.3048844
[49] F. M. Talaat, M. S. Saraya, A. I. Saleh, H. A. Ali, and S. H. Ali, "A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment," *Journal of Ambient Intelligence and Humanized Computing,* vol. 11, pp. 4951–4966, Nov. 2020. https://doi.org/10.1007/s12652-020-01768-8
[50] R. K. Naha, S. Garg, A. Chan, and S. K. Battula, "Deadline-based dynamic resource allocation and provisioning algorithms in Fog-Cloud environment," *Future Generation Computer Systems*, vol. 104, pp. 131–141, Mar. 2020. https://doi.org/10.1016/j.future.2019.10.018
[51] D. Tychalas and H. Karatza, "A scheduling algorithm for a fog computing system with bag-of-tasks jobs: Simulation and performance evaluation," *Simulation Modelling Practice and Theory*, vol. 98, Art no. 101982, Jan. 2020. https://doi.org/10.1016/j.simpat.2019.101982
[52] T. Aladwani, "Scheduling IoT healthcare tasks in fog computing based on their importance," *Procedia Computer Science*, vol. 163, pp. 560–569, 2019. https://doi.org/10.1016/j.procs.2019.12.138
[53] D. Zeng, L. Gu, S. Guo, Z. Cheng, and S. Yu, "Joint optimization of task scheduling and image placement in fog computing supported software defined embedded system," *IEEE Transactions on Computers,* vol. 65, no. 12, pp. 3702–3712, Dec. 2016. https://doi.org/10.1109/TC.2016.2536019
[54] S. Bitam, S. Zeadally, and A. Mellouk, "Fog computing job scheduling optimization based on bees swarm," *Enterprise Information Systems*, vol. 12, no. 4, pp. 373–397, 2018. https://doi.org/10.1080/17517575.2017.1304579
[55] Z. Liu, X. Yang, Y. Yang, K. Wang, and G. Mao, "DATS: Dispersive stable task scheduling in heterogeneous fog networks," *IEEE Internet of Things Journal,* vol. 6, no. 2, pp. 3423–3436, Apr. 2018. https://doi.org/10.1109/JIOT.2018.2884720
[56] X.-Q. Pham and E.-N. Huh, "Towards task scheduling in a cloud-fog computing system," in *2016 18th Asia-Pacific network operations and management symposium (APNOMS)*, Kanazawa, Japan, Nov. 2016, pp. 1–4. https://doi.org/10.1109/APNOMS.2016.7737240
[57] T. Choudhari, M. Moh, and T.-S. Moh, "Prioritized task scheduling in fog computing," in *Proceedings of the ACMSE'18 conference*, Art no. 22, Mar. 2018, pp. 1–8. https://doi.org/10.1145/3190645.3190699
[58] S. Zhao, Y. Yang, Z. Shao, X. Yang, H. Qian, and C.-X. Wang, "FEMOS: Fog-enabled multitier operations scheduling in dynamic wireless networks," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1169–1183, Apr. 2018. https://doi.org/10.1109/JIOT.2018.2808280
[59] B. Jamil, M. Shojafar, I. Ahmed, A. Ullah, K. Munir, and H. Ijaz, "A job scheduling algorithm for delay and performance optimization in fog computing," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 7, Art no. e5581, Apr. 2020. https://doi.org/10.1002/cpe.5581
[60] S. Wang, T. Zhao, and S. Pang, "Task scheduling algorithm based on improved firework algorithm in fog computing," *IEEE Access*, vol. 8, pp. 32 385–32 394, Feb. 2020. https://doi.org/10.1109/ACCESS.2020.2973758
[61] P. Hosseinioun, M. Kheirabadi, S. R. K. Tabbakh, and R. Ghaemi, "A new energy-aware tasks scheduling approach in fog computing using hybrid meta-heuristic algorithm," *Journal of Parallel and Distributed Computing*, vol. 143, pp. 88–96, Sep. 2020. https://doi.org/10.1016/j.jpdc.2020.04.008
[62] S. Ghanavati, J. Abawajy, and D. Izadi, "Automata-based dynamic fault tolerant task scheduling approach in fog computing," *IEEE Transactions on Emerging Topics in Computing*, Oct. 2020. https://doi.org/10.1109/TETC.2020.3033672
[63] M. I. Naas, P. R. Parvedy, J. Boukhobza, and L. Lemarchand, "iFogStor: an IoT data placement strategy for fog infrastructure," in *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC),* Madrid, Spain, Aug. 2017, pp. 97–104. https://doi.org/10.1109/ICFEC.2017.15
[64] M. I. Naas, L. Lemarchand, J. Boukhobza, and P. Raipin, "A graph partitioning-based heuristic for runtime IoT data placement strategies in a fog infrastructure," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing,* Apr. 2018, pp. 767–774. https://doi.org/10.1145/3167132.3167217
[65] T. Huang, W. Lin, Y. Li, L. He, and S. Peng, "A latency-aware multiple data replicas placement strategy for fog computing," *Journal of Signal Processing Systems,* vol. 91, no. 10, pp. 1191–1204, Feb. 2019. https://doi.org/10.1007/s11265-019-1444-5
[66] N. Wang and J. Wu, "Latency minimization through optimal data placement in fog networks," *Fog Computing: Theory and Practice*, pp. 269–291, Apr. 2020. https://doi.org/10.1002/9781119551713.ch10
[67] J. Wang, "When data cleaning meets crowdsourcing," *AMPlab*, UC, Berkeley, 2015.
[68] J. Ni, K. Zhang, Y. Yu, X. Lin, and X. S. Shen, "Providing task allocation and secure deduplication for mobile crowdsensing via fog computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 3, pp. 581–594, 2018. https://doi.org/10.1109/TDSC.2018.2791432
[69] J. Yan, X. Wang, Q. Gan, S. Li, and D. Huang, "Secure and efficient big data deduplication in fog computing," *Soft Computing*, vol. 24, pp. 5671–5682, Jul. 2019. https://doi.org/10.1007/s00500-019-04215-9
[70] P. Shynu, R. Nadesh, V. G. Menon, P. Venu, M. Abbasi, and M. R. Khosravi, "A secure data deduplication system for integrated cloud-edge networks," *Journal of Cloud Computing*, vol. 9, Art no. 61, pp. 1–12, Nov. 2020. https://doi.org/10.1186/s13677-020-00214-6
[71] R. Vales, J. Moura, and R. Marinheiro, "Energy-aware and adaptive fog storage mechanism with data replication ruled by spatio-temporal content popularity," *Journal of Network and Computer Applications,* vol. 135, pp. 84–96, June 2019. https://doi.org/10.1016/j.jnca.2019.03.001
[72] A. Berkennou, G. Belalem, and S. Limam, "A replication and migration strategy on the hierarchical architecture in the fog computing environment," *Multiagent and Grid Systems*, vol. 16, no. 3, pp. 291–307, Oct. 2020. https://doi.org/10.3233/MGS-200333
[73] I. Al Ridhawi, N. Mostafa, Y. Kotb, M. Aloqaily, and I. Abualhaol, "Data caching and selection in 5G networks using F2F communication," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC),* Montreal, QC, Canada, Oct. 2017, pp. 1–6. https://doi.org/10.1109/PIMRC.2017.8292681
[74] W. Bai, H. Feng, Y. Wang, and X. Han, "Research on data cache algorithm of fog computing node," in *2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China, Nov. 2020, pp. 197–200. https://doi.org/10.1109/ICSESS49938.2020.9237670
[75] Y. Fu, X. Qiu, and J. Wang, "F2MC: Enhancing data storage services with fog-toMultiCloud hybrid computing," in *2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC)*, London, UK, Oct. 2019, pp. 1–6. https://doi.org/10.1109/IPCCC47392.2019.8958748
[76] T. Yu, X. Wang, and A. Shami, "A novel fog computing enabled temporal data reduction scheme in IoT systems," in *GLOBECOM 2017- 2017 IEEE Global Communications Conference*, Singapore, Dec. 2017, pp. 1–5. https://doi.org/10.1109/GLOCOM.2017.8253941
[77] A. Gómez-Cárdenas, X. Masip-Bruin, E. Marín-Tordera, S. Kahvazadeh, and J. Garcia, "A hash-based naming strategy for the fog-to-cloud computing paradigm," in *European Conference on Parallel Processing Workshops. Lecture Notes in Computer Science.* vol 10659, Springer, Cham, pp. 316–324, 2017. https://doi.org/10.1007/978-3-319-75178-8_26
[78] D. Guibert, J. Wu, S. He, M. Wang, and J. Li, "CC-fog: Toward content-centric fog networks for E-health," in *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*, Dalian, China, Oct. 2017, pp. 1–5. https://doi.org/10.1109/HealthCom.2017.8210830
[79] A. J. Kadhim and S. A. H. Seno, "Energy-efficient multicast routing protocol based on SDN and fog computing for vehicular networks," *Ad Hoc Networks*, vol. 84, pp. 68–81, Mar. 2019. https://doi.org/10.1016/j.adhoc.2018.09.018
[80] A. P. Abidoye and B. Kabaso, "Energy-efficient hierarchical routing in wireless sensor networks based on fog computing," *EURASIP Journal on Wireless Communications and Networking,* Art no. 8(2021), pp. 1–26, Jan. 2021. https://doi.org/10.1186/s13638-020-01835-w
[81] N. Noorani and S. A. H. Seno, "SDN and fog computing-based switchable routing using path stability estimation for vehicular ad hoc networks," *Peer-to-Peer Networking and Applications,* vol. 13, pp. 948–964, 2020. https://doi.org/10.1007/s12083-019-00859-4

[82] T. Saito, S. Nakamura, T. Enokido, and M. Takizawa, "Epidemic and topic-based data transmission protocol in a mobile fog computing model," in *International Conference on Broadband and Wireless Computing, Communication and Applications. Springer,* Oct. 2020, pp. 34–43. https://doi.org/10.1007/978-3-030-61108-8_4

[83] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, and X. Yao, "Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things," *IEEE Internet of Things Journal,* vol. 4, no. 5, pp. 1143–1155, Oct. 2017. https://doi.org/10.1109/JIOT.2017.2659783

[84] M. Wazid, A. K. Das, N. Kumar, and A. V. Vasilakos, "Design of secure key management and user authentication scheme for fog computing services," *Future Generation Computer Systems*, vol. 91, pp. 475–492, Feb. 2019. https://doi.org/10.1016/j.future.2018.09.017

[85] Z. Ali, S. A. Chaudhry, K. Mahmood, S. Garg, Z. Lv, and Y. B. Zikria, "A clogging resistant secure authentication scheme for fog computing services," *Computer Networks*, vol. 185, Art no. 107731, Feb. 2021. https://doi.org/10.1016/j.comnet.2020.107731

[86] R. Lu, K. Heung, A. Habibi Lashkari, and A. Ghorbani, "A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT," *IEEE Access*, vol. 5, pp. 3302–3312, Mar. 2017. https://doi.org/10.1109/ACCESS.2017.2677520

[87] C. Zuo, J. Shao, G. Wei, M. Xie, and M. Ji, "CCA-secure ABE with outsourced decryption for fog computing," *Future Generation Computer Systems,* vol. 78, no. 2, pp. 730–738, Jan. 2018. https://doi.org/10.1016/j.future.2016.10.028

[88] Z. Guan, Y. Zhang, L. Wu, J. Wu, J. Li, Y. Ma, and J. Hu, "APPA: An anonymous and privacy preserving data aggregation scheme for fogenhanced IoT," *Journal of Network and Computer Applications*, vol. 125, pp. 82–92, Jan. 2019. https://doi.org/10.1016/j.jnca.2018.09.019

[89] F. Wang, J. Wang, and W. Yang, "Efficient incremental authentication for the updated data in fog computing," *Future Generation Computer Systems*, vol. 114, pp. 130–137, Jan. 2021. https://doi.org/10.1016/j.future.2020.07.039

[90] H. Noura, O. Salman, A. Chehab, and R. Couturier, "Preserving data security in distributed fog computing," *Ad Hoc Networks*, vol. 94, Art no. 101937, Nov. 2019. https://doi.org/10.1016/j.adhoc.2019.101937

[91] M. Al-Khafajiy, T. Baker, M. Asim, Z. Guo, R. Ranjan, A. Longo, D. Puthal, and M. Taylor, "COMITMENT: A fog computing trust management approach," *Journal of Parallel and Distributed Computing*, vol. 137, pp. 1–16, Mar. 2020. https://doi.org/10.1016/j.jpdc.2019.10.006

[92] J. Xu, H. Liu, W. Shao, and K. Deng, "Quantitative 3-D shape features based tumor identification in the fog computing architecture," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 2987–2997, Feb. 2019. https://doi.org/10.1007/s12652-018-0695-5

[93] J. Wan, B. Chen, S. Wang, M. Xia, D. Li, and C. Liu, "Fog computing for energy-aware load balancing and scheduling in smart factory," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 10, pp. 4548–4556, Oct. 2018. https://doi.org/10.1109/TII.2018.2818932

[94] V. Vijayakumar, D. Malathi, V. Subramaniyaswamy, P. Saravanan, and R. Logesh, "Fog computing-based intelligent healthcare system for the detection and prevention of mosquito-borne diseases," *Computers in Human Behavior*, vol. 100, pp. 275–285, Nov. 2019. https://doi.org/10.1016/j.chb.2018.12.009

[95] R. Siddharth and G. Aghila, "A light weight background subtraction algorithm for motion detection in fog computing," *IEEE Letters of the Computer Society*, vol. 3, no. 1, pp. 17–20, 2020. https://doi.org/10.1109/LOCS.2020.2974703

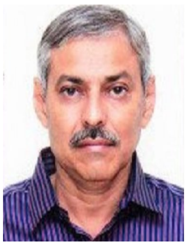
[96] J. Xu, K. Ota, and M. Dong, "Fast deployment of emergency fog service for disaster response," *IEEE Network*, vol. 34, no. 6, pp. 100–105, 2020. https://doi.org/10.1109/MNET.001.1900671

[97] A. Ali, Y. Zhu, and M. Zakarya, "A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing," *Multimedia Tools and Applications*, vol. 80, pp. 31401–31433, Jan. 2021. https://doi.org/10.1007/s11042-020-10486-4

[98] F. A. Salaht, F. Desprez, and A. Lebre, "An overview of service placement problem in fog and edge computing," *ACM Computing Surveys (CSUR)*, vol. 53, no. 3, Art no. 65, pp. 1–35, June 2020. https://doi.org/10.1145/3391196

[99] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in *2015 Third IEEE workshop on hot topics in web systems and technologies (HotWeb)*, Washington, DC, USA, Nov. 2015, pp. 73–78. https://doi.org/10.1109/HotWeb.2015.22

Shaifali P. Malukani has received her B. E. degree from Sardar Patel University, V. V. Nagar, Gujarat, India in 2011, and the M. Tech. degree from Dharmsinh Desai University, Nadiad, Gujarat, India in 2015. She is currently pursuing PhD in Computer Engineering from Dharmsinh Desai University, Nadiad, Gujarat, India. Her areas of interest are fog computing, IoT and image processing.
She has been working as an Assistant Professor at the Department of Computer Engineering at Dharmsinh Desai University, Nadiad, Gujarat, India. She has around 10 years of teaching and research experience.
E-mail: shaifalimalukani.ce@ddu.ac.in



C. K. Bhensdadia has received his B. E. degree from Dharmsinh Desai University, Nadiad, Gujarat, India in 1990, the M. Tech. degree from IIT Mumbai, India in 1996 and the PhD degree from KSV, Gandhinagar, Gujarat, India in 2013. His areas of interest are fog computing, artificial intelligence and data mining.
He is a Professor and Head of the Department of Computer Engineering at Dharmsinh Desai University, Nadiad, Gujarat, India. He has around 31 years of teaching and research experience.
He has published articles in reputed peer-reviewed national and international journals and conference procedings.
E-mail: ckbhensdadia@ddu.ac.in