Computer Communications 209 (2023) 469–498

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Review

Blockchain based resource allocation in cloud and distributed edge computing: A survey

Gaurav Baranwal [a,*], Dinesh Kumar [b], Deo Prakash Vidyarthi [c]

[a] Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi, 221005, U.P, India
[b] Department of Computer Science and Engineering, National Institute of Technology Jamshedpur, Jamshedpur, 831014, India
[c] School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, 110067, India

A R T I C L E I N F O

Keywords:
Cloud computing
Edge computing
Blockchain
Decentralization
Resource allocation
Computation offloading

A B S T R A C T

Cloud computing and distributed edge computing provide computing resources to meet the surging demands for computing caused by developments in technologies such as the Internet of Things (IoT) and Mobile communication (5G). Centralized resource allocation approaches in both computing paradigms suffer from single-point failure, tampering, modification in allocation results, and biased actions. Recently, blockchain has become popular in designing decentralized systems because of its features, including transparency, decentralization, and anti-tamper. In this paper, we provide a comprehensive survey of the research works applying blockchain in resource allocation in both computing paradigms, addressing the issues in centralized resource allocation approaches. Firstly, we identify several key research questions acting as motivation. To provide background knowledge, first, we discuss the centralized resource allocation in both computing paradigms and associated challenges. Then we discuss blockchain, its structure, working, characteristics and types, followed by its benefits to resource allocation. We identify several metrics to provide a comparative study of the works. We present a depth overview of blockchain-based resource allocation works in three domains: cloud computing, distributed edge computing and integrated edge and cloud computing. In each domain, works are summarized from three aspects: works using blockchain platforms, works providing blockchain frameworks and works advocating blockchain. We discuss consensus mechanisms in the works related to blockchain-based resource allocation, as the consensus mechanism is a fundamental part of the blockchain. Further, we provide key challenges requiring our attention. Finally, we conclude the survey.

Contents

* Corresponding author.
E-mail addresses: gaurav.baranwal@bhu.ac.in (G. Baranwal), dineshkumar.cse@nitjsr.ac.in (D. Kumar), dpv@mail.jnu.ac.in (D.P. Vidyarthi).

https://doi.org/10.1016/j.comcom.2023.07.023
Received 11 April 2023; Received in revised form 27 June 2023; Accepted 20 July 2023
Available online 22 July 2023
0140-3664/

1. Introduction

Various technologies have recently emerged, such as the Internet of Things (IoT), Big Data, Machine Learning, and Mobile communication (5G). Applications of these technologies, such as smart healthcare, smart city, virtual reality, mobile games, autonomous vehicles, smart homes, and smart agriculture, generate huge amounts of data in the era of 5G. They also need to run various computational jobs such as data cleaning, processing, and training, requiring higher computing resources.

Cloud computing, introduced in 2008, is a cost-effective solution that provides on-demand computing infrastructures that involve no maintenance by the user and can be provisioned rapidly [1]. In recent years, IoT has significantly improved our quality of life as many of its applications are shaping our lives. Recent IoT-based applications need high requirements for latency, energy consumption, bandwidth, reliability, etc. [2,3]. But due to high latency, limited bandwidth, huge energy consumption and large distance between the physical infrastructure of the cloud and its users, it is hard to meet the high requirements of applications by cloud computing, specially delay-sensitive applications, context-aware applications etc. Recently various distributed edge computing paradigms, such as Fog Computing, Edge computing, and Mobile edge computing, have emerged to provide computing services near to the users. These edge computing paradigms are considered extensions of cloud computing. Taking computing services from distributed edge computing provides powerful computing, communication and storage in the proximity of the users, thus resulting in reduced latency and energy consumption and enhanced security.

The market of cloud computing has already been established. According to Cisco [4], many billions of edge computing nodes will be deployed in the near future and seeing the providing capability of a better quality of service and quality of experience to users, a huge market of edge computing like cloud computing is inevitable. In the cloud computing and edge computing market, many computing service providers may be present at a time. Their main aim will be to maximize revenue by offering various computing services to attract users. This introduces a challenge for users, i.e., selecting the best service or computing node among available ones. Users may explore each available service to select the most suitable one, but it leads to low service allocation efficiency and consumes the users' time. Therefore, various centralized framework-based works related to computing resource allocation in cloud computing and in distributed edge computing paradigms are present in the literature. Various surveys and reviews thoroughly discuss centralized resource allocation approaches in different computing paradigms: Cloud computing in [5,6], Fog computing in [7], Edge computing in [8], Mobile edge computing in [9,10].

One notable common point among major resource allocation approaches is that they involve centralized trusted entities responsible for resource matching and allocation. They are called with different

Table 1
Research questions.

S.No.	Research questions	Motivations
1	What are the current challenges in centralized resource allocation in the cloud and distributed edge computing?	A centralized trusted entity runs the resource allocation algorithm, introducing challenges such as single point of failure, lack of trust, mutable environment, security, data integrity, etc.
2	What are the major characteristics of the blockchain?	The major characteristics of blockchain are transparency, immutability, decentralization, persistency, anonymity, etc.
3	What are the benefits that blockchain can provide to centralized resource allocation?	Major benefits are a trustless environment, transparent and secure resource allocation, immutable resource allocation results, no role of centralized entity, no single point of failure, etc.
4	What is the current progress in blockchain-based resource allocation?	Many works designed resource allocation using blockchain. Few considered some existing blockchain platforms, and few gave novel blockchain frameworks for resource allocation.
5	What are the open challenges in blockchain-based resource allocation?	Incentive mechanisms, scalability, and consensus mechanisms specific to resource allocation are a few open challenges in blockchain-based resource allocation.

names in different contexts, e.g., in cloud computing, this entity is called cloud exchange, cloud broker, cloud coordinator, and cloud mediator. In distributed edge computing, it is known as edge broker, fog broker, gateway, fog gateway and node. Centralization, caused by these centralized entities, offers various benefits such as efficient resource allocation, monitoring of services, better reputation management, etc. However, it also raises security and privacy concerns, such as single-point failure, tampering requests or offers, manipulation of allocation results, collusion to increase profit, and biased actions. Therefore, decentralized approaches for resource allocation are required, keeping the benefits of centralized approaches intact and addressing the related challenges for the long-term growth of cloud computing and distributed edge computing. Blockchain has the potential to develop a decentralized ecosystem for the allocation of computing resources.

1.1. Integration of blockchain and cloud and distributed edge computing

Blockchain technology is gaining popularity because it promises transparency, immutability, non-repudiation, enhanced security, decentralized systems, trustless systems, etc. Blockchain is a distributed ledger in which information is stored in blocks, and blocks are linked using hash pointers. The distributed ledger is managed by a peer-to-peer network that removes the need for a trusted third party. Blockchain term abstracting as "a cryptographically secured chain of blocks" was first coined in [11] in 1991. However, it became popular and got wide attention after the introduction of Bitcoin in 2008 [12]. Cryptocurrencies such as Bitcoin and Ether are popular applications of blockchain technology. In cryptocurrencies, a block stores financial transaction. In the blockchain, smart contracts can also be stored to define rules and execute them automatically. In the beginning, blockchain became popular for financial services, but later many domains, such as IoT, Cloud Computing, Edge Computing, Supply Chain market, etc., benefited from the blockchain. Smart contracts providing automated control acted as a catalyst.

Blockchain and computing paradigms, such as cloud computing and distributed edge computing, can be helpful to each other. Generally, blockchain is expected to cover three aspects: Decentralization, Security, and Scalability. Decentralization makes the blockchain system permissionless. Security in blockchain requires immutability and resistance to various attacks, and scalability requires low latency and high throughput. However, one can achieve at most two of decentralization, security, and scalability [13]. This constraint is known as blockchain trilemma. Cryptocurrencies are achieving decentralization and security but not scalability. Scalability issues are making blockchain-based solutions less practical. Cloud and distributed edge computing can help to address the scalability issues of blockchain by providing computing resources for computing requirements of blockchain, such as solving the Proof of Work (PoW) puzzle during mining. It can also provide storage to store bulky public blockchain or can provide a confidential environment to a private blockchain. Blockchain can also be helpful for cloud and distributed edge computing to enhance security and privacy and provide automatic control of resources. Blockchain can address the security issues related to control, storage, computation, and network in the cloud and distributed edge computing. In summary, there are two possibilities of integration between blockchain and computing paradigms like cloud and distributed edge computing: (i) incorporation of cloud computing and distributed edge computing in blockchain to provide computing resources satisfying the need of computing requirements of blockchain, and (ii) incorporation of blockchain in the cloud and distributed edge computing to design decentralized resource allocations systems enhancing security, privacy, transparency etc. Many studies are reported in the literature that surveyed the works related to the integration of blockchain and cloud and distributed edge computing [14,15].

1.2. Motivations

Computing resource allocation is one of the most studied challenges in cloud and distributed edge computing, and we observed a huge development in blockchain-based resource allocation recently. Blockchain is becoming a candidate technology to develop a decentralized ecosystem for allocating computing resources. However, to our best knowledge, no works provide a comprehensive survey on blockchain-based resource allocation in cloud computing and distributed edge computing. The main motivation of this survey is to understand the developments in blockchain-based resource allocation. We identify the several key research question, summarized in Table 1, that work as motivation for this survey.

1.3. Survey methodology

To provide a survey on blockchain-based computing resource allocation, first, we define research questions related to resource allocation in cloud computing and distributed edge computing using blockchain. After analyzing the research questions, we discuss the centralized computing resource allocation and its challenges that the blockchain can address. Further, we briefly discuss the blockchain and its benefits to resource allocation. After defining the clear research scope, we searched for the works on IEEE Explore, Springer, ACM Digital Library, ScienceDirect, Elsevier, Wiley, and Google Scholar published during 2017–2022. Our search phrases included 'blockchain', 'edge computing', 'fog computing', 'mobile edge computing, 'cloud computing, 'mobile cloud computing, 'computation offloading', 'resource allocation', and 'resource scheduling. We also studied references in the relevant works to find other potential works. Moreover, we also explored the 'Cited by' list of relevant works on Google Scholar to discover other potential works. We have included all the works that present blockchain-based computing resource allocation in this survey. We present the extracted information that addresses the research questions in this survey.

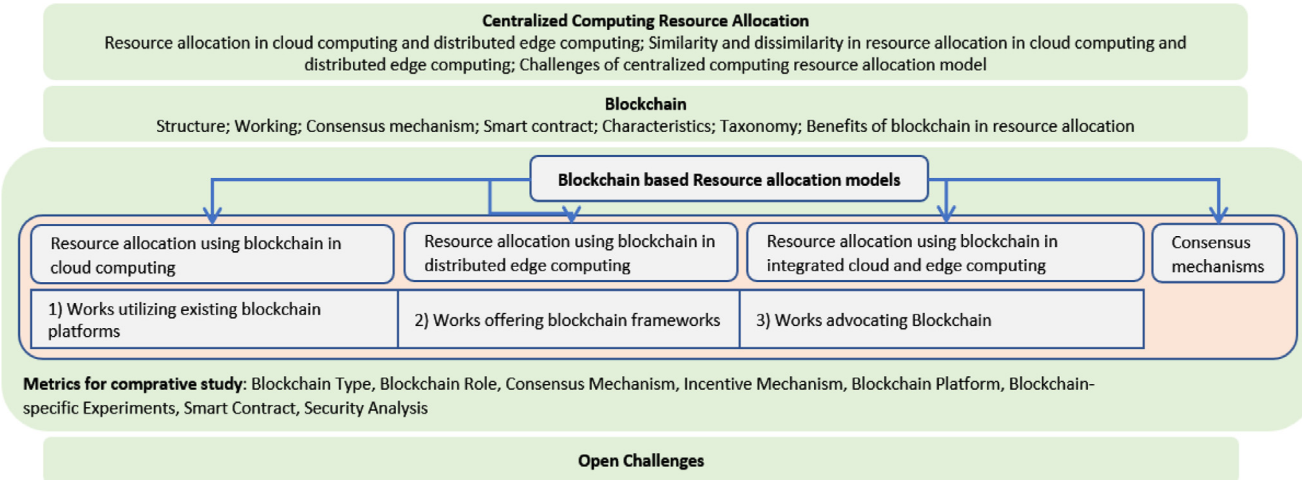

Fig. 1. Logical organization of survey.

1.4. Logical organization

A huge development can be observed in computing resource allocation in the cloud and distributed edge computing. However, major resource allocation approaches involve centralized trusted entities responsible for resource matching and allocation. As discussed, these centralized entities introduce many challenges that blockchain can address. Since the main objective of this work is to provide a survey on blockchain-based computing resource allocation in the cloud and distributed edge computing, we need to first understand centralized computing resource allocation and its associated challenges. Therefore, we have explained centralized computing resource allocation in cloud computing and distributed edge computing.

Further, we elaborate on associated challenges with centralized computing resource allocation so that one can clearly understand the need for blockchain in resource allocation. To explain how blockchain can be benefited to address the associated challenges with centralized computing resource allocation, we first explained the basics of blockchain, its consensus mechanisms, its types and characteristics, and then the benefits of blockchain in resource allocation. Many works on resource allocation consider either distributed edge computing or cloud computing from where one can get computing resources. However, many works consider the integration of cloud computing and distributed edge computing as both paradigms complement each other. Therefore, we consider three categories: works in cloud computing, works in distributed edge computing and works in the integration of cloud computing and distributed edge computing. We study each category's works related to blockchain-based resource allocation separately. While studying the works in each category, we found three types of works. The first type is where existing blockchain platforms are used to design resource allocation approaches. The second type is where blockchain frameworks are developed, especially for computing resource allocation. The last type is where authors discuss the role of blockchain in resource allocation and highlight the benefits of blockchain but do not enlighten on the procedure to integrate blockchain in resource allocation. Therefore, considering these three types, we further created three sub-categories in each category. We identified various metrics to provide a comparative study, such as Blockchain Type, Blockchain Role, Consensus Mechanism, Incentive Mechanism, Blockchain Platform, etc. Discussion on blockchain helps to understand these metrics. Since the consensus mechanism is the backbone of the blockchain, we have created a fourth category to discuss the works related to consensus mechanisms. After this, we discussed open challenges. A logical organization of this survey is given in Fig. 1.

1.5. Contributions

In comparison with existing surveys, summarized in Section 2, that provide earlier developments in the integration of blockchain and cloud and distributed edge computing, we provide an extensive survey on blockchain-based computing resource allocation covering many aspects, from recent developments and current status to open challenges and research possibilities to educate and motivate researchers. The main contributions of this survey are as follows:

- We provide a comprehensive discussion on resource allocation problems and challenges of centralized resource allocation in cloud computing and distributed edge computing.
- We provide a comprehensive discussion on blockchain, its characteristics of blockchain and the role of blockchain in addressing the challenges of centralized resource allocation in the cloud and distributed edge computing.
- To provide the current development status, we summarize the works related to blockchain-based resource allocation in the cloud and distributed edge computing.
- We identify challenges and issues from the extensive review of blockchain's role in resource allocation in the cloud and distributed edge computing.
- We provide future research directions on the role of blockchain in resource allocation problems in the cloud and distributed edge computing.

1.6. The structure of the paper

The paper is organized as follows: Section 1 gives the introduction. Section 2 discusses the related surveys present in the literature, validating the contributions of this survey. Section 3 discusses resource allocation in the cloud and distributed edge computing, including the challenges. Section 4 discusses the blockchain and its benefits to resource allocation. Section 5 provides a discussion on the existing works in blockchain-based resource allocation in cloud computing and distributed edge computing. Section 6 presents the open research challenges and future research opportunities. Finally, Section 7 concludes the survey. An overview of the structure of the paper is given in Fig. 2.

2. Related surveys

Resource allocation is one of the most studied challenges in cloud and distributed edge computing. After the popularity of blockchain, many works have been reported in the literature that design decentralized resource allocation in cloud computing and distributed edge

1 Introduction
1.1 Integration of blockchain and cloud and distributed edge computing
1.2 Motivations
1.3 Survey methodology
1.4 Logical Organization
1.5 Contributions
1.6 The structure of the paper
2 Related Surveys
3 Centralized Computing Resource Allocation
3.1 Resource allocation in cloud computing
3.2 Resource allocation in distributed edge computing
3.3 Similarity and dissimilarity in resource allocation in cloud computing and distributed edge computing
3.4 Challenges of centralized computing resource allocation model
4 Blockchain
4.1 Structure of blockchain
4.2 Working of blockchain
4.3 Consensus mechanism
4.4 Smart contract
4.5 Characteristics of blockchain
4.6 Taxonomy of blockchain
4.7 Benefits of blockchain in resource allocation
5 Overview of Blockchain based Resource allocation models in cloud computing and distributed edge computing
5.1 Resource allocation using blockchain in cloud computing
5.1.1 Resource allocation utilizing existing blockchain platforms
5.1.2 Resource allocation offering blockchain frameworks
5.1.3 Resource allocation advocating blockchain
5.2 Resource allocation using blockchain in distributed edge computing
5.2.1 Resource allocation utilizing existing blockchain platforms
5.2.2 Resource allocation offering blockchain frameworks
5.2.3 Resource allocation advocating Blockchain
5.3 Resource allocation using blockchain in integrated cloud and edge computing
5.3.1 Resource allocation utilizing existing blockchain platforms
5.3.2 Resource allocation offering blockchain frameworks
5.3.3 Resource allocation advocating Blockchain
5.4 Consensus mechanisms
5.5 Discussion
6 Challenges
6.1 Incentive mechanisms
6.2 Security and privacy
6.3 Scalable blockchain
6.4 Reputation and trust management
6.5 Global reference architecture
6.6 SLA monitoring
6.7 Consensus mechanism
6.8 Internal cryptocurrency/token
6.9 Smart contract and gas cost
6.10 Joint optimization of resource allocation and blockchain parameters
6.11 Simulator
7 Conclusion

Fig. 2. Structure of the paper.

computing. In the literature, many studies have surveyed works related to resource allocation in cloud computing and distributed edge computing extensively [5–7,9,10]. However, these studies do not cover blockchain-based resource allocation. Studies on the integration of blockchain and cloud and distributed edge computing gave little attention to blockchain-based resource allocation.

In [14], the works related to computation, storage and network of integrated blockchain and edge computing systems are discussed, stating that blockchain and edge computing can be integrated to address each other's challenges. In [15], the potential benefits of blockchain for cloud computing are discussed, and the works related to blockchain and cloud computing are categorized based on services of cloud. In [16], the works related to blockchain-based trust management in cloud computing are discussed, followed by a cloud–edge hybrid framework and a double blockchain-based transaction model for flexible trust management.

In [17], possible attacks on cloud and blockchain-based cloud solutions and possible solutions to address the attacks are discussed. In [18], the works related to blockchain and fog computing integration are discussed. In [19], the benefits of cloud computing and blockchain are given, followed by the discussion on issues of the cloud that can be addressed by the blockchain, such as interoperability, SLA, Data management, and data encryption. A model of integrated blockchain and cloud architecture is also given. In [20], decentralized consensus systems, i.e., blockchain-based systems, for edge computing-based IoT and related open issues are discussed. In [21], the challenges of edge computing that can be addressed by the blockchain, such as identity management, reliability, and addressing of devices, are discussed.

In [22], a comprehensive discussion on the integration of blockchain and the cloud of things, its applications and platforms are given. Further, challenges are discussed, and future directions are given. In [23], a survey is given on the works that use blockchain to power the cloud or works that utilize the cloud to reinforce blockchain. The role of blockchain in the cloud is also discussed to address the issues related to data provenance, access control issues, searchable encryption, data deduplication, and resource allocation. Further, the role of the cloud in blockchain to address the issues related to high computing and high storage requirements is also discussed. [24] provides a survey on blockchain-based resource allocation methods in cloud–edge computing, categorizing the methods into three classes: auction-based, optimization-based, and revenue-based methods. Major limitations of [24] are that it covers only a few papers, does not provide a comparison of works based on various blockchain-related metrics, lacks a discussion on the need of blockchain in resource allocation, provides a limited discussion on challenges and does not provide future directions. Related surveys are summarized in Table 2.

3. Centralized computing resource allocation

The section provides a basic background of resource allocation problems in cloud computing and distributed edge computing and discusses various challenges of centralized resource allocation mechanisms.

3.1. Resource allocation in cloud computing

Cloud computing provides computing services such as computing resources and storage on-demand basis. In cloud computing, computing resources are provided to the users in virtualized form (e.g., Virtual Machines (VMs)). It helps to run multiple application services on a single physical machine. These VM instances are allocated to different users for their tasks' execution. In cloud computing, multiple providers provide computation and storage services to their users. As resources are limited, and there is a large number of users, the available resources must be efficiently allocated to users [25]. Several objectives are considered during the cloud resource allocation, such as providers' profit, resource utilization, QoS, users' satisfaction and cost. An overview of resource allocation is given in Fig. 3, where cloud broker is the centralized entity that acts as an interface between users and cloud service providers and does the resource allocation.

Resource allocation primarily deals with allocating the available resources to the incoming applications/workload on time so that these applications can use these resources efficiently. This mapping of applications to resources is done based on QoS requirements (e.g., response time, CPU and memory requirements, throughput, deadline etc.) to maximize the provider's profit or minimize the user's cost.

3.2. Resource allocation in distributed edge computing

Cloud computing has computational power and storage to address the need of IoT devices. However, the quality of computing services may be affected by various factors, such as the large distance between the cloud data center and its users and the stability of the connection. Especially, latency-sensitive applications require a timely response, which cloud computing may not be able to provide. To

Table 2
Summary of related surveys.

Ref.	Topic	Contribution	Issues
[14]	Blockchain and edge computing	Integration of blockchain and edge computing is discussed.	Less discussion on blockchain based resource allocation in edge computing and lacks the discussion on blockchain based resource allocation in cloud computing
[15]	Blockchain and cloud computing	Discussion on works related to blockchain based cloud services	Less discussion on blockchain based resource allocation in cloud computing and lacks the discussion on blockchain based resource allocation in distributed edge computing
[16]	Blockchain and cloud computing	Discussion on blockchain-based trust approaches in cloud computing	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[17]	Cloud and blockchain	Discussion security attacks in cloud and blockchain based cloud solutions	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[18]	Blockchain and fog	Survey on integration of blockchain and fog computing and its applications such as IoT, IoV, IIoT.	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[19]	Blockchain and cloud computing	Discussion on issues of cloud addressed by the blockchain and benefits of blockchain based cloud computing	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[20]	Blockchain and edge centric IoT	Discussion on decentralized consensus systems and research issues of decentralized systems for edge centric IoT	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[21]	Blockchain and edge computing	Challenges of edge computing that can be addressed by the blockchain	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[22]	Blockchain and cloud of things	Survey on integration of blockchain and Cloud of Things, related applications and platforms.	Lacks the discussion on blockchain based resource allocation in cloud and distributed edge computing
[23]	Blockchain and cloud computing	Survey on works using blockchain to address the issues of cloud computing and works using cloud to address the issues of blockchain.	Only few works related to blockchain based resource allocation in cloud computing are discussed.
[24]	Blockchain and cloud–edge computing	Survey on blockchain based resource allocation in cloud–edge computing	Covers few papers. Lacks comparison of works and discussion on need of blockchain. Limited discussion on challenges and future research directions.

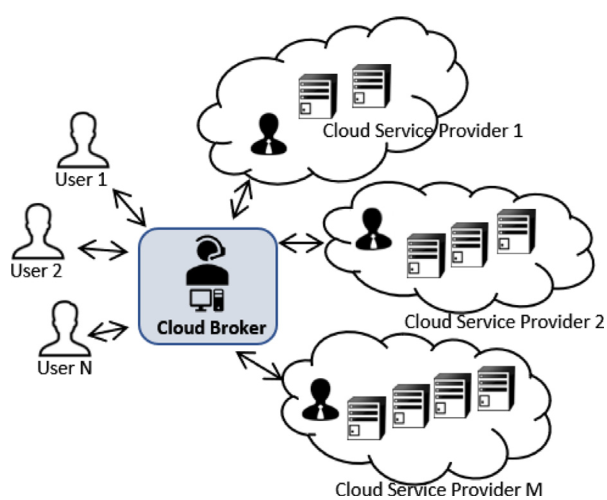

Fig. 3. Resource allocation in cloud computing.

supplement cloud computing, distributed edge computing, such as fog computing, edge computing, cloudlet, etc., is introduced. In distributed edge computing, computing services are provided by the edge servers to handle latency, mobility and location-sensitive applications. In distributed edge computing, generally, more than one owner of edge servers cooperates to maximize their resource utilization and profit. Usually, a node is deployed as a centralized authority that runs the resource allocation algorithms, matches service users and providers, decides the most suitable host among edge and cloud computing nodes, etc. Resource allocation problems in distributed edge computing can be categorized into three problems. These three problems and their basic details are as follows.

3.2.1. Computation offloading

Computation offloading is the process of transferring compute-intensive tasks of a resource-constrained device to remote computing servers with significant computing capacity. It has several objectives, such as maximizing the battery life and computing efficiency of end-user devices, minimizing the overall execution time, reducing execution cost and many more. The tasks are generally offloaded from resource-constrained devices (e.g. IoT devices, smartphones, sensors) to resource-rich servers. Depending on the task's resource requirement, the offloaded location may be a remote cloud server, a fog node, an edge server, or a MEC server. The formulation of offloading problem considers several parameters such as delay, user location, task's deadline, execution cost, network bandwidth and current battery level of the end-user device. The problem becomes complex when the number of tasks and edge/cloud servers increases. Like the service placement problem, the computation offloading problem is also proved to be an NP-Hard problem. Primarily, several decisions need to be taken during the offloading process, such as when, where and whether to offload.

3.2.2. Service placement

Service placement problem deals with the placement of application components (microservices, tasks or a single monolithic service) to fog nodes or fog resources. The placement is done to optimize some objectives, such as the number of placed services, latency, energy, resource utilization, deployment cost and QoS. In most research studies, multiple objectives are considered during the placement process in which objectives are optimized collectively using the weighted sum method or individually in the Pareto sense. Decision variables in the placement problem are binary in nature, describing whether a particular service is placed on a specific node or not. The many-to-many

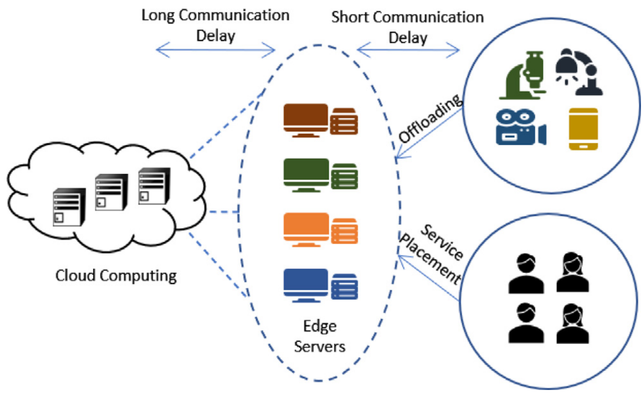

Fig. 4. Resource allocation in distributed edge computing.

mapping between the services and resources also makes the problem difficult to solve, normally in the class of NP-Hard problems. At worst, it requires the evaluation of n^m mapping possibilities where n and m are the numbers of fog/edge nodes and services, respectively. The literature studies related to this problem verify the potential of heuristics and meta-heuristics approaches, which gives the near-optimal solution in a reasonable amount of time.

3.2.3. Virtual machine allocation

Many research works proposed VM allocation model in distributed edge computing domain. Like cloud computing, the problem also involves the process of allocating the VMs of edge service providers to end-users. Research works in [26–28] utilized auctions for allocating the VMs to end users.

An overview of resource allocation in distributed edge computing environment is given in Fig. 4.

3.3. Similarity and dissimilarity in resource allocation in cloud computing and distributed edge computing

3.3.1. Similarity

The resource allocation in cloud computing and distributed edge computing are similar in many aspects. Generally, the resource allocation algorithm decides the quantity and type of resource allocated to each user. A central entity is required to run this resource allocation algorithm in both computing paradigms to obtain the allocation results. The results are then communicated to all service providers and users. This central entity may be a broker, auctioneer, or coordinator. For example, in auction-based resource allocation models, an auctioneer acts as a central entity responsible for solving the allocation problem. Similarly, a cloud provider can become a coordinator in a cloud federation scenario and run the allocation algorithm. In general settings, a resource allocation problem in the cloud and distributed edge computing is an NP-Hard problem and difficult to solve in a reasonable time when the number of service providers and users is large. Similar kinds of approaches have been used in both computing domains to solve the resource allocation problems, such as auction, negotiation, linear programming relaxation, approximation, heuristics, meta-heuristics and optimization-based solvers/software.

3.3.2. Dissimilarity

Resource allocation problem is a fundamental and well-defined problem to both the computing domains. Although there are a few similarities, several factors make the cloud resource allocation problem different from the problem of allocating edge resources to its users. These dissimilarities are as follows.

(i) In distributed edge computing, as there is a strong requirement for ultra-low latency between the edge service providers and end-user devices, parameters and constraints related to the underlying network must be considered. They should be a part of the formulated resource allocation problem, whereas the cloud resource allocation problem generally emphasizes allocating computing and storage resources to cloud users.

(ii) Distributed edge computing reduces the overall latency of the tasks, which eventually depends upon the communication time and task execution. Therefore, the utility of a task would depend upon the time it gets on the edge resource for its execution. Unlike edge computing, tasks executed in cloud computing are mainly resource-oriented, and tasks' utility depends upon the number of cloud resources (e.g. VMs) assigned for its execution.

(iii) A resource allocation process considers several parameters, such as tasks' requirements, resource constraints, location of resources/end users, network bandwidth and resource heterogeneity. In distributed edge technologies, resources are distributed over multiple levels ranging from sensors, smartphones and IoT devices to resource-constrained devices like gateways, routers, base stations and cloudlets. Unlike edge computing, resources in cloud computing are centralized. This results in novel objectives that need to be optimized during resource allocation, such as context and locality awareness.

(iv) Unlike cloud computing, the edge resources and end-user devices in edge computing may be mobile. Therefore, handling mobility in these computing environments must be considered during the allocation process. Traditional cloud computing consists of centralized data centers in which cloud service users purchase the cloud services, and cloud service providers are static in nature.

(v) The resource allocation problem in cloud computing focuses on the allocation of resource-rich and powerful cloud servers to its users, whereas in edge computing, it focuses on the allocation of distributed, heterogeneous and resource-constrained edge resources to the end-users.

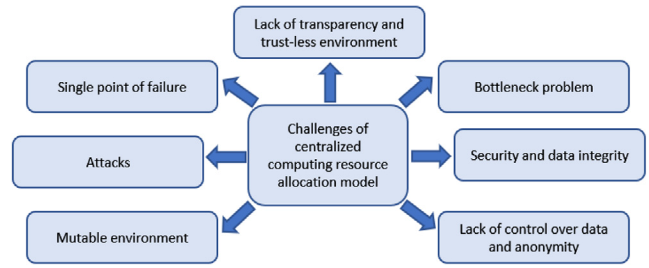

Fig. 5. Challenges of centralized computing resource allocation model.

3.4. Challenges of centralized computing resource allocation model

This section explains various challenges in centralized resource/service allocation models, summarized in Fig. 5, that must be addressed.

Single point of failure: Generally, a trusted broker runs the resource allocation algorithm and assigns the resources of service providers to the users. Failure of the broker can cause a full system failure. Because of the broker, i.e., centralized entity, various attacks such as DDoS and man-in-middle attacks are possible, leading to the system's failure.

Lack of transparency and trust-less environment: A broker is assumed to be a trusted entity, but it may be biased towards any user or service provider, creating a malicious environment. Checking the biased actions of the broker is a challenging job. Cloud or edge resources are like a black box, and users do not know the operational mechanisms. Lack of transparency and a trust-less environment becomes a challenge.

Bottleneck problem: A large number of users and providers connected to a broker can cause bottleneck problems.

Lack of control over data and anonymity: The broker can access and tamper with the data of users and providers without telling them. Both lack control over their data. Therefore, some mechanism is required to ensure that only authorized entities have access or control over their data, and maintaining anonymity is a challenge in the centralized system.

Security and data integrity: Centralized systems attract a wide range of attackers. Since the central authority manages everything, a small breach from the attackers may cause a leak of bulk information, resulting in serious harm to the system. Though access and authorization techniques exist, they do not have a full guarantee. A small change by the attacker in data may also cause serious harm to users and providers.

Mutable environment: Trust and reputation management is incorporated into resource allocation to prevent service providers and users from abusing the system. Trust and reputation of service providers are calculated based on given feedback by users. A central entity or a broker takes users' feedback to calculate trust and reputation. A broker biased towards a service provider can change the feedback of users, or a service provider may hack the system to change the feedback. Similarly, a user may change its feedback to promote or demote a service provider. A broker can change the past history of service providers and users. Therefore, the resource allocation system must be immutable.

*Attacks:*A centralized resource allocation system is prone to various attacks, such as DDoS attacks, constant attacks, man-in-middle attacks, whitewashing attacks, etc. The DDoS attack makes the central authority or broker unreachable by flooding the superfluous requests in the network. An attacker targets an entity in a constant attack by providing false information. In a man-in-middle attack, an attacker eavesdrops on the communication between entities and poses as a legitimate entity to access the information illegally. In a whitewash attack, a malicious entity reenters the system with a new identity. Resource allocation systems must be safe from these attacks.

Above discussed issues act as motivation to users and computing resource providers to switch from centralized resource allocation models to decentralized resource allocation models. After the introduction of cryptocurrencies and their acceptance, such as bitcoin and Ether, blockchain emerged as one of the promising solutions to implement decentralized operations. Blockchain has the potential to address the above-discussed issues.

4. Blockchain

Blockchain is a digital distributed database popularly known as a distributed ledger. In the blockchain, blocks storing the history of digitally signed transactions in a period are linked using hash pointers. Blockchain became popular after the introduction of Bitcoin in 2008 by Satoshi Nakamoto (a pseudo name). Bitcoin is a digital currency, popularly known as cryptocurrency, that can be used to pay one party from another party without involving any central entity such as banks. However, blockchain is not limited to implementing cryptocurrencies for online payments. It can also be used in various domains, such as edge computing, IoT, supply chain management, healthcare and many more, because of its unique characteristics, such as immutability, anonymity, transparency, and decentralization.

4.1. Structure of blockchain

In a blockchain, blocks are connected via hash pointers similar to a linked list. Each block is generally divided into two parts: Header and Body. The body contains the transactions that occurred in the network. The header contains a cryptographic hash pointer pointing to the previous block and other information such as time-stamp (publication time of the block), nonce, Merkle tree root of transactions stored in the block's body, etc., depending on the requirements. Cryptographic hashes of transactions are grouped into pairs to form a Merkle tree, as given in Fig. 6.

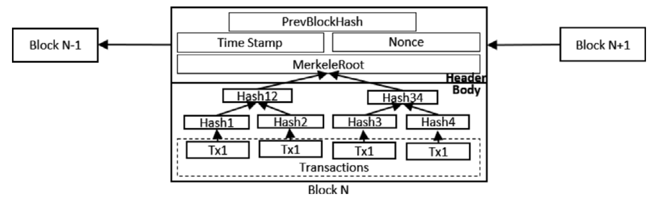

Fig. 6. Blockchain structure.

The first block is known as the genesis block. The current block keeps the previous block's cryptographic hash value, also known as the hash pointer. Since cryptographic hashing is irreversible and each block stores the cryptographic hash value of the previous block in the header, it helps in achieving the immutability property. Each peer or node requires a private-public key pair to make a transaction. The peer uses its private key to perform a digital signature on the transaction, e.g., transfer of some cryptocurrency or data to a recipient peer's address. Generally, the address of a node or peer is the cryptographic hash value of its public key.

4.2. Working of blockchain

Nodes maintain a transaction pool in which transactions published in the network are stored. In the blockchain, a miner/a leader (a winner node among participating nodes in some challenge or a node with earned position) clubs transactions collected in the transaction pool over a period, places them in a block and proposes the block in the network. A miner may earn some incentive for proposing the block. Connected nodes in the network verify the correctness of the proposed block and add the legitimate block to its existing local blockchain.

4.3. Consensus mechanism

In a centralized system, a central authority is responsible for maintaining and updating information or to take a decision. In contrast, in a decentralized system consensus of all stakeholders is required to take a decision. In a decentralized computing system, a consensus mechanism is a process used to achieve consensus among involving parties/nodes over data or the network state. In the blockchain, if multiple nodes participate who might be malicious, generating a huge number of transactions, the consensus mechanism should be secure, fair, effective and efficient enough to ensure that nodes/involved parties agree on the legitimate transactions in a legitimate block in the blockchain. A general process in consensus mechanisms is as follows: A leader is selected among the nodes based on some predefined rules. Each node maintains a transaction pool in which transactions occurring in the network are stored. This leader adds legitimate transactions from the local pool to a block and publishes the block in the network. After receiving the block, other nodes verify the leader's and the block's validity. Other nodes add the block in their local copy of the blockchain if the block is legitimate. Since few nodes might be malicious and the network may not be perfect; therefore, a block is considered a valid block when the majority of the nodes add the block. A leader may also get some reward for creating a new block. In such a scenario, nodes compete to become a leader. Proof of Work (PoW), Proof of Authority (PoA), Practical Byzantine Fault Tolerance (PBFT), Proof of Stake (PoS), and Proof of Reputation (PoR) are a few popular consensus mechanisms. A detailed comparative discussion on consensus mechanisms used in blockchain-based resource allocation in the cloud and distributed edge computing is given in Section 5.4.

4.4. Smart contract

In a smart contract, interactions among multiple parties can be coded, which can be executed automatically. These parties may be unknown to each other, and the smart contract builds trust among them. A smart contract, a recent development in blockchain, has accelerated the development of blockchain-based applications. As a smart contract is stored in the blockchain, rules written in the smart contract exist across the network. Further, since data related to the smart contract is stored in the blockchain, it is tamper-proof, non-repudiation and persistent. A smart program is self-executed when it meets the required condition. Since it is a part of the blockchain, the consensus of the majority of nodes is also needed. The nodes in the blockchain can verify a smart contract. Since it runs the rule without involving a central authority, it creates ample opportunities in various fields where creating a trustful environment was a big challenge. Ethereum is a very popular blockchain for designing decentralized applications, as it supports the execution of smart contracts. In Ethereum, a smart contract is written using the solidity programming language.

4.5. Characteristics of blockchain

Decentralization: Participating entities communicate with the centralized server or some centralized entity in a centralized system. In blockchain-based systems, all transactions among participating entities are stored in a distributed ledger, and each connected node keeps a copy of the distributed ledger. Each participating node is responsible for maintaining consistency in the system that distributes the role of a centralized server or centralized entity. Decentralization also addresses the single-point-failure problem of centralized systems. If a node fails or loses some data from its local copy of the distributed ledger, the node can quickly recover with the help of other nodes.

Immutable and tamper-resistant: Since blocks are connected using the hash pointer, to modify any block in the blockchain, one has to modify each block after the changed block in the blockchain. These modifications must happen in the local copy of the blockchain of each node in the network. That is why tampering is nearly impossible in the blockchain, which makes it immutable and tamper-resistant.

Persistency: In the blockchain, each transaction is verified before adding into the block. Further, each block is verified before its inclusion in the chain by the consensus of the majority of the participants. Therefore, any malicious action or attempt to change the information in a transaction or a block will be visible to others and will not be committed. Therefore, once the information is stored, it cannot be modified and becomes persistent.

Anonymity: Each participant or node gets a unique asymmetric pair of keys: A public key and a private key, to perform any operation/action in the blockchain. The public key works as a public address and may be known to others. The private key is a secret key, only known to its owner. Since only these keys are used to generate transactions or perform any actions, one cannot link these keys to the owner's real identity. This way, blockchain ensures the anonymity of its users.

Non-repudiation: A participant must digitally sign any information using its private key before sharing it in the network, and the private key is only known to its owner. Therefore, the participant's digital signature ensures its involvement in the transactions. After a transaction is verified and added to the blockchain, a participant cannot deny its involvement.

Transparency: In the blockchain, all members can view all transactions happening in the blockchain. Further, members contribute their computational effort to decide the leader/miner and verify and validate transactions and blocks. Therefore, blockchain maintains transparency in the system, ensuring data integrity. This characteristic of blockchain improves trust in the system.

Auditability: Since the blockchain is immutable and tamper-resistant, it ensures non-repudiation, and each node keeps a copy of the blockchain, users can look up and verify the transactions specific to any blockchain address. Users can track and verify any transaction that happened in the past.

4.6. Taxonomy of blockchain

Generally, blockchains are categorized into three categories based on the availability and accessibility of its users: Public Blockchain, Private Blockchain, and Consortium blockchain.

Public blockchain: This blockchain is permissionless. It is an open and fully decentralized ledger in which a node can enter the blockchain network and can be involved in processing verification and validation of transactions and blocks using the consensus mechanism without taking permission. A node can also only be involved in issuing the transactions without verifying and validating transactions or blocks. Some rewards are given to encourage nodes involved in verification and validation. Cryptocurrencies are generally run on the public blockchain.

Private blockchain: This is a permissioned blockchain. Unlike a public blockchain, this blockchain is controlled by a central authority that decides who will have the accessibility to information. This blockchain is limited to single organizations or a few organizations. This blockchain is more like a centralized blockchain. Roll backing is possible by the organization anytime, unlike public blockchain. A private blockchain is used as a synchronized distributed database within the organization. No reward is given for the processing of transactions.

Consortium blockchain: This blockchain is also known as a federated blockchain. This blockchain is somewhere between public and private blockchains. This blockchain ensures partial decentralization because it is controlled by a predefined number of nodes, i.e., organizations or stakeholders. Other nodes may present in the network without involving in the consensus mechanism. Organizations/service providers can form a consortium blockchain to collaborate, where service providers are involved in consensus. Service users may be a type of other nodes that can request the services but are not involved in the consensus mechanism. Here also, blockchain is a synchronized distributed database but within the members of the consortium. Some rewards may be given for the processing of transactions.

4.7. Benefits of blockchain in resource allocation

No dependency on central authority: Blockchain helps in designing decentralized resource allocation mechanisms. Smart contracts especially help in the execution of rules of resource allocation in a decentralized manner. Users and service providers can both interact system without involving the central authority; thus, dependency on central authority is addressed.

Unbiased resource allocation: Since there is no central authority, and resource allocation rules are written in the smart contract, biased resource allocation is not possible if the majority of nodes are honest.

Secured resource allocation: Since all transactions are connected in a block using the Merkley tree and blocks are connected via a hash pointer, even a minor change by attackers is nearly impossible if the majority of nodes are honest. Above that, if an attacker wants to make a minor change or a successful breach, it has to ensure the change in the local copy of the blockchain of each node to achieve consistency. Decentralization and cryptography techniques in blockchain make resource allocation secure.

No single-point failure: Since each node in the blockchain maintains a copy of the blockchain, a single-point failure is not possible. Even if a node is not available or fails, other nodes will still be working.

Transparency in resource allocation: In blockchain-based resource allocation, the ledger is shared among all stakeholders, i.e., service providers and service users. Any stakeholder can verify the resource allocation process and the results. If a smart contract is used to implement a resource allocation algorithm, its self-execution in a decentralized environment after meeting necessary conditions also ensures transparency in the system.

Attack resilient: Blockchain is itself resistant to many attacks, such as Sybil, DDoS, etc., and proper designing of resource allocation mechanism in blockchain environment has potential to other attacks such as self-promotion attacks, constant attacks, etc.

The benefits of blockchain in resource allocation are given in Fig. 7.

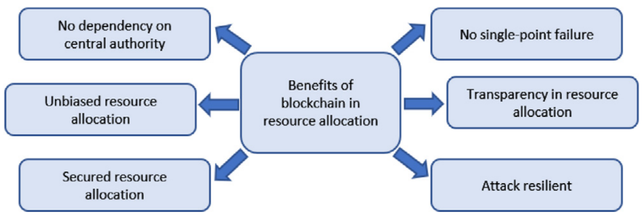

Fig. 7. Benefits of blockchain in resource allocation.

5. Overview of blockchain based resource allocation models in cloud computing and distributed edge computing

Blockchain has been applied to design decentralized resource allocation models in cloud computing, distributed edge computing and integrated edge and cloud computing. Therefore, we have considered this as the first categorization of works. Further, the works can be categorized based on blockchain implementation for resource allocation. As discussed before, in smart contracts, rules of interactions among multiple parties can be coded and executed automatically. Since a smart contract is stored in the blockchain, it provides great flexibility to its users. An existing blockchain platform supporting smart contracts can be directly used to design decentralized resource allocation. Resource allocation rules, i.e., resource allocation methods, can be written in the smart contracts, and these smart contracts act as decentralized brokers. Further, no overhead will be needed to design consensus mechanisms, transactions format, etc.

Ethereum and Hyperledger are blockchain platforms supporting smart contracts implementation. Ethereum is very popular for designing decentralized applications, also known as DAPP. Therefore, it is also popular in edge computing and cloud computing for resource allocation because one has to write the resource allocation method in the smart contracts and deploy them in Ethereum, and work will be done. Many works used existing blockchain platforms as supporting tools. However, consensus mechanisms in the existing blockchain platforms are less suitable for resource allocation. Therefore, few works designed new consensus mechanisms and validated the work in a simulated environment. Many works only recommended adopting blockchain in resource allocation, considering the benefits of blockchain without providing the details of how to do it. Therefore, we have categorized the works into three categories: Resource Allocation utilizing Existing Blockchain Platforms, Resource Allocation offering blockchain frameworks, and Resource Allocation advocating blockchain. Works in the first category used some existing blockchain platforms for resource allocation. The works in the second category proposed novel blockchain frameworks that address the issues in the existing blockchain platforms and resource allocation. Works in the third category identify the merits of blockchain and only consider blockchain for resource allocation. Unlike works in the first and second categories, these works do not give any framework, use any existing blockchain platform, or provide any guidance. We classify the works related to blockchain-based computing resource allocation models as shown in Fig. 1.

Blockchain technology is considered in resource allocation to provide security and improve trust. Considering the role of blockchain in resource allocation, we identify various comparison metrics to provide a comparative study. Each comparison metric is discussed as follows:

Domain: We identify the area of resource allocation. For example, many works designed auction-based resource allocation methods and considered blockchain to provide security.

Objective: This survey covers the works that aim to allocate resources. Therefore, we identify the objective of the considered resource allocation problems.

Role of blockchain (BR): We identify the five roles of blockchain in resource allocation. Blockchain can be used for storing information related to resource allocation in decentralized manner, identity management for involved entities, payment management, trust and reputation management, and resource management.

Type of blockchain (BT): As discussed before, there are three types of blockchain: public, private and consortium. A public blockchain, such as bitcoin and public Ethereum, is suitable for payment, but for resource allocation, it is not suitable because public blockchain supports full decentralization, i.e., anyone can join and leave the network without permission. It suffers from scalability issues, slow transactions and low efficiency. A private blockchain is more like a centralized blockchain; therefore, it does not help in achieving decentralization in resource allocation. A consortium blockchain is the most suitable blockchain for resource allocation seeing the market of cloud computing and edge computing where participating entities involved in consensus may be service providers who are competitors and known to each other. Unlike the public cloud, consortium blockchain has an authorization scheme to join the network, and it involves limited nodes; therefore, it provides better scalability, transaction speed and efficiency compared to the public blockchain. In summary, the type of blockchain should be decided based on the system's needs.

Consensus mechanism (CM): The consensus mechanism is the backbone of the blockchain. If a blockchain-based resource allocation model uses some existing consensus mechanisms, the model will inherit the limitations of the consensus mechanism, if there are any. For example, most works considered PoW as a consensus mechanism, which suffers from no consensus finality, high energy consumption, slow transaction speed, etc. A detailed discussion on consensus mechanisms used for blockchain-based resource allocation is given later in this section.

Incentive mechanism (INC): Blockchain requires computation effort from the nodes involved in the consensus mechanism and maintaining a decentralized ledger. No node will provide computation effort if it does not get some incentive as motivation in return. For example, in PoW, nodes try to solve a computational puzzle, and a node with the solution becomes the winner and gets the reward. In blockchain-based resource allocation, some incentive mechanism is also needed to motivate the nodes to participate in the consensus process.

Blockchain platform (BP): Validation and performance analysis are needed in blockchain-based resource allocation. Therefore, we observed the used blockchain platforms to validate and study the performance.

Smart Contract (SC): We observe whether smart contracts are used in resource allocation.

Blockchain-specific experiments (BSE): It is obvious that works related to resource allocation will perform the experiments related to resource allocation, but in a blockchain-based system, it is interesting to see whether the works provide the blockchain-specific experiments to study the involvement of blockchain in resource allocation.

Security analysis (SA): Section 4 discusses the benefits of the blockchain in resource allocation. The blockchain prevents the malicious actions of involved entities such as brokers, users, and providers. Therefore, a security analysis is needed to validate that the proposed solution prevents the malicious actions of involved entities in the system.

While giving the overview of the works, we discuss only those metrics discussed above, which are present in work. If some metrics are not given in the overview of any work, it means discussion on those metrics is unavailable.

5.1. Resource allocation using blockchain in cloud computing

5.1.1. Resource allocation utilizing existing blockchain platforms

Auction-based resource allocation

For resource procurement, the auction is used in which providers bid considering the users' requirements. Auction-based resource procurement is proposed in [29,30]. In [29], a user specifies the requirement for computing resources and storage services and selects a provider considering the offered price and credibility. In [30], a user specifies the cloud storage requirement and implements Vickrey–Clarke–Groves (VCG)-based auction to select a provider. Both works implement auction rules in the smart contracts using solidity programming language and deploy them on the Ethereum blockchain to develop a prototype.

In [31], blockchain is applied to achieve auction fairness and trade fairness in auction-based resource allocation in cloud computing. Auction fairness means the malicious users cannot change their bid after submission and will be punished if they quit the auction. The auction result cannot be changed. Trade fairness means the auction will be automatically executed, and a malicious user cannot get the VMs without paying. A malicious provider cannot get the payment without providing the VM. Participants must deposit the currency in a smart contract to participate in an auction that ensures their correct behavior. The smart contract is also used to implement the submission of bids and the auction. The authors provided a security analysis to show auction fairness and trade fairness in the system. The work is implemented in the Ethereum blockchain.

In [32], an auction-based model for federated cloud services is proposed in the blockchain environment, where more than one service provider can be the winner. Two smart contracts are used to run the proposed model. The main contract manages the auction process, and the second contract generated for each winner ensures SLA between the customer and winner service providers in the auction. To ensure SLA, new entities, witnesses, are introduced that will monitor the services offered by the service providers. Based on the reports given by the witnesses, service fees will be paid to the providers. In case of SLA violation, some penalty is imposed. This work is implemented in the public Ethereum blockchain and provides gas cost and security analysis.

In [33], a double auction-based cloud market is proposed where buyers are cloud service providers who require more computing power and sellers are also cloud service providers who sell their idle computing resources. Buyers and sellers both bid in the auction. Maximizing social welfare, i.e., maximizing the satisfaction of buyers and sellers, is the objective of resource allocation. The proposed resource allocation algorithm is implemented using a smart contract in the consortium blockchain. Buyers give feedback for obtained computing resources from the buyers by scoring the seller. Scores given to a seller are converted into a rating used in resource allocation. The work is implemented in Hyperledger fabric.

Cloud federation and joint cloud

In [34], a blockchain-based cloud federation model is proposed in which three smart contracts are implemented. The first smart contract is used to register cloud service providers, the second is used to create a profile of cloud providers and their transactions, and the third is used to request a service. The work helps in utilizing underutilized cloud resources by cloud providers with limited infrastructure to satisfy their customers' needs, eliminating the need for trust in federation. A dynamic differential game is designed to maximize the profit of cloud providers participating in the federation. The work is implemented in the public Ethereum blockchain, and gas cost analysis is given. In [35], a cloud provider can outsource computing resources from other cloud providers using smart contracts. Verifier agents are introduced to ensure SLA between requester and provider because smart contracts cannot monitor SLA. These verifier agents are trusted third-party agents, and they charge to monitor. A three-player dynamic Stackelberg differential game is formulated among the requester, provider and verifier for optimal resource allocation. The work is implemented in the Ethereum blockchain using smart contracts.

Collaboration among multiple cloud service providers (JointCloud) can fulfill the demand of IoT devices. Trust is a basic problem in cooperation among cloud providers and IoT devices. JointCloud may also lead to the risk of the leak of privacy of IoT devices. In [36], a blockchain-based JointCloud model is proposed, and three types of credit-based entities are considered. Cloud providers with high credit are responsible for managing other involved entities. The second entity type is cloud providers with high complaint rates or low credit value. The third entity type is IoT devices that can buy or sell services from other IoT devices or cloud providers. A credit-based bonus-penalty scheme using a smart contract is proposed that prevents the illegal behavior of participants in JointCloud. The proposed model is implemented in Hyperledger Fabric, and performance is analyzed based on transaction delay.

Multi-tenant resource allocation

Virtualization technology helps in achieving multi-tenancy in cloud computing, i.e., same physical resource can be allocated to more than one user or tenant. But a tenant can attack another tenant sharing the same physical resource, known co-resident attack. In [37], blockchain is used to record the activities of users. It helps in tracking the malicious actions and malicious users. The work is implemented in smart contract in Ethereum. In [38] also, a smart contract-based system for cloud tenant management is proposed that performs four types of services: identity management, authentication of users, authorization of cloud services, and payment. The smart contract is implemented in Quorum blockchain system, which does not require gas cost for running the smart contracts.

Above discussed works are summarized in Table 3.

5.1.2. Resource allocation offering blockchain framework

In [39], blockchain-based task scheduling is proposed in cloud computing. Task manager is an entity from where nodes in the blockchain can access the tasks and virtual machines' characteristics to design a schedule. A node that generates a schedule publishes the schedule in the blockchain. If 50% of nodes accept the schedule in the blockchain, the schedule is considered as good. This process is known as Proof of Scheduling. Among nodes in the blockchain who are willing to publish the block are known as validators. A leader among validators is elected based on the history of adding transactions. The idea can be supported by public and private blockchains. The work implements the BCSchedCloudSim simulator to evaluate the performance.

5.1.3. Resource allocation advocating blockchain

In cloud computing, for the allocation of resources, negotiation can be used where providers and users exchange their offers until an agreement or rejection is achieved. However, a trusted third party is required to exchange offers. Therefore, a participant may deny its offer or the received offer, and there is a possibility of manipulation in the offer after giving it. To overcome these issues and to maintain anonymity during negotiation, in [40], a framework of blockchain is proposed that can be used for negotiation to provide transparency in the exchange of offers, tamper-safe storage for stored exchanged offers and correct execution of negotiation conditions. The work lacks a discussion on consensus mechanism and miner rewards.

In [41], requests to run required virtual machines (VMs) are served by cloud service providers. The resource allocation objective is to minimize energy consumption costs from VM scheduling and migration. The resource allocation method is implemented in the smart contract. Blockchain is used to provide a decentralized resource management environment. For each request migration, the smart contract decides a migration location using a reinforcement learning-based method. A new consensus mechanism is introduced to save the energy that selects a data center as a miner, which has the highest load and has not mined last a constant number of blocks.

Above discussed works are summarized in Table 4.

Summary: Auction-based resource allocation methods in cloud computing can prevent the control of cloud computing in the hands of a few cloud service providers. Further, a federation of cloud service providers or collaboration among service providers, i.e., joint cloud, can help small cloud service providers to sustain themselves in the market. However, collaboration among competitive cloud service providers and centralized platforms running auctions or providing cloud tenant management can create distrust among service providers and users. Blockchain platform supporting smart contracts is very useful as one

Table 3
Comparison of papers focusing on resource allocation in cloud computing utilizing existing blockchain platforms.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Auction based resource allocation	[29]	To increase competition in the cloud market	NA	R	NA	NA	Ethereum	NA	Yes	NA
	[30]	Secure and competitive resource trading	NA	R	NA	NA	Ethereum	Gas cost	Yes	NA
	[31]	To achieve auction and trade fairness	NA	R	NA	NA	Ethereum	Mining time	Yes	Yes
	[32]	To select cost-effective service providers and to monitor SLA	Public	R	PoW	PoW	Ethereum	Gas cost analysis	Yes	Yes
	[33]	To maximize the social welfare	Consortium	R	NA	NA	Hyperledger fabric	Time cost analysis	Yes	NA
Cloud federation and joint cloud	[34]	To maximize the profit of cloud providers participating in the federation, eliminating the need of a trusted federation	Public	R	PoW	PoW	Ethereum	Gas cost analysis	Yes	NA
	[35]	To introduce a verifier to monitor SLA	Public	R	PoW	PoW	Ethereum	Gas cost analysis	Yes	NA
	[36]	To design a credit based join cloud model	Consortium	T, R	NA	NA	Hyperledger fabric	Transaction delay	Yes	NA
Multi-tenancy	[37]	To prevent malicious activities of users	NA	R	NA	NA	Ethereum	NA	Yes	NA
	[38]	To manage tenant	Consortium	I, R, P	NA	NA	Quorum	NA	Yes	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

Table 4
Comparison of papers focusing on resource allocation in cloud computing advocating blockchain.

Domain	Ref	Objective	BT	BR	CM	INC	SC	SA
Negotiation based resource allocation	[40]	To ensure transparency and integrity in negotiation between untrusted participants	NA	R	NA	NA	NA	NA
Resource allocation using deep reinforcement learning	[41]	To minimize the energy consumption cost from VM scheduling and migration	Consortium	R	New consensus	NA	Yes	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; SC: Smart Contract; SA: Security Analysis; R: Resource management.

has to write rules on the interaction among cloud service providers and cloud users in a smart contract and deploy it in the blockchain. Further, blockchain acts as a decentralized distributed database where all the activities happening via deployed smart contracts will be recorded. Therefore, blockchain providing transparency and immutability can be used to design trustless resource management systems in cloud computing. In cloud computing, a cloud service provider may deny providing service after agreement or paying the penalty in case of malicious activity. Similarly, a user may refuse to accept an offer after agreement or to pay for a service. Blockchain acts as an immutable database where all the activities can be recorded and later used for auditing purposes to decide on the real culprit. A pictorial representation of blockchain-based resource allocation in cloud computing is given in Fig. 8. Unlike Fig. 3, here there is no need of broker.

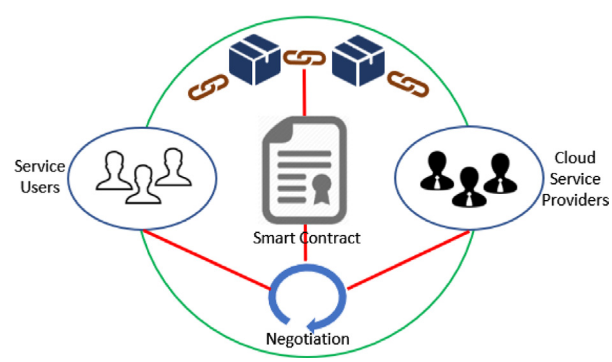

Fig. 8. Blockchain-based resource allocation in cloud computing.

5.2. Resource allocation using blockchain in distributed edge computing

5.2.1. Resource allocation utilizing existing blockchain platforms

Auction based resource allocation

In [42], a blockchain-based decentralized reverse auction is proposed to allocate fog computing resources. A smart contract is used to define the rules of the auction, in which an IoT device sets its preferences for QoS requirements. The IoT device also sets an expected reputation value for the assigned fog service provider. Both IoT devices and service providers have to deposit a fee in the smart contract. The deposited amount by an IoT device is returned to the IoT device when it gives the feedback. The deposited amount by a service provider is returned to the service provider if it does not violate its promised quality of service. After a certain time, the auction ends automatically, and the winner is selected. The proposed auction is implemented using the public Ethereum blockchain and smart contract. Similarly, in [43], auction-based edge computing resource allocation using smart contract and consortium blockchain is proposed. The work aims to maximize the satisfaction of IoT devices that depend on various quality parameters. Both works [42,43] provide Gas cost and security analysis.

In [44], a three-layer hierarchical framework is considered in which edge servers are present at the top layers, and at the second and third layers, IoT devices are present. IoT devices at the second layer purchase resources from the edge servers using ascending clock auction and resell them to IoT devices at the bottom layer using ascending clock auction. In [45], joint resource allocation by fog nodes is considered. A user requests a bundle of computing resources, and fog nodes bid to provide the resources. Descending combinatorial auction is used for resource allocation. Both works are implemented in Ethereum, and the gas cost is analyzed.

IoT devices connected to a blockchain network cannot participate in the mining process as they have limited computation power, and mining processes such as PoW require great computing efforts. The computing resources of edge computing can be used for the mining process. Therefore, in [46], a double auction-based resource allocation mechanism is proposed that improves the utility of users and edge resource providers. The auction mechanism is implemented in a smart contract in Ethereum to remove the need for a trusted platform for running the auction mechanism.

Credit based resource management and payment

In [47], a credit-based resource management system in the permissioned blockchain environment is proposed to decide how much computing resource an IoT device can get based on application type, priority and past behavior. Edge computing nodes act as mining nodes. An internal currency or coin system is used. Each IoT device is bootstrapped with the same amount of coins. Smart contracts enforce rules and regulations related to handling the behavior of IoT devices and resource allocation policies. If an IoT device behaves maliciously, its credit value is reduced, and it has to return the coin based on the change in the credit. Dynamic pricing is implemented to allocate computing resources. The price of computing resources is decided based on the requested edge resource, the total and available amount of the edge resources, and application priority. It is suggested that mining in the proposed environment is less resource-sensitive; therefore, consensus mechanisms like PBFT can be used, and there is no need to incentivize miners. The work is implemented in Ethereum Blockchain using a smart contract where PoW is the consensus mechanism. The work is evaluated based on overhead caused by blockchain, i.e., computation cost in mining, communication and storage cost of block synchronization, and computation and communication cost of the smart contract. In [48], an agent purchases computing resources from edge servers on credit and resells them to IoT devices. A two-level Stackelberg Game based computing resource trading mechanism that uses a credit-based payment approach is introduced. Trading of computing resources is implemented using a smart contract implemented in Ethereum. The gas cost analysis is given.

Trading of resources using coin

In [49], a decentralized resource allocation mechanism using blockchain and smart contracts is proposed considering the mobility of users. Since fog computing nodes are geographically distributed, a few nodes may be compromised for a short period, causing privacy and security concerns. A new cryptocurrency Fogcoin like bitcoin, issued by fog computing service providers, is introduced. The work also suggested allowing limited nodes with high computing powers to work as miners and maintain the ledger. The work is implemented in the private Ethereum blockchain.

In [50], Trust problems between mobile equipment (MEs) and edge servers is resolved using a blockchain platform for trading. In [50], authors identified two problems in blockchain-based resource allocation in mobile edge computing: Cold start and Long return. In a blockchain, coins are needed for trading, but a new ME may not have coins to purchase computing resources from edge serves. It is known as a cold start problem. Auditing and verifying transactions in the new block may be costly in terms of time, known as the long return problem. To overcome both problems, it is proposed that MEs can take a loan from the banks, and banks may set their rate of return. For MEs, two decisions are important: where to offload and from which bank loan should be taken. Therefore, a joint optimization problem is proposed to minimize energy, computing time and digital coin. Since MEs will decide on their interest, the problem is considered a non-cooperative game, and a distributed algorithm to find the solution is introduced. The work uses two smart contracts in Ethereum (one for trading computing resources and another for loaning) and performs the gas cost analysis.

Big data processing

Big data requires adequate availability of computing power. Though cloud computing can provide services, there are three fundamental challenges with cloud vendors: transparency, security and privacy. Recently, distributed computing paradigms have been used in big data processing, where data processing tasks are distributed among computing contributors. However, a centralized entity is needed to control activities in distributed computing. Therefore, computing contributors have to trust the centralized entity, generating security and privacy issue. In [51], a blockchain-based decentralized computing platform is developed in which computing contributors fully control the deep learning model and their local data. They share their learning models, not the raw data, using homomorphic encryption techniques. They can leave and join anytime without disturbing others. PoW helps in mitigating the effect of byzantine nodes, and voting by the verification contributors is used to evaluate the contributions of computing contributors. The work is implemented using the Ethereum blockchain. Computing nodes willing to participate as computing contributors have to deposit a security fee to register.

Collaboration of edge service providers

Edge servers can collaborate to share their resources. It will improve resource utilization and customer satisfaction. But a trusted environment is needed to implement real-time collaboration. In [52], a blockchain framework is proposed where edge servers can collaborate and maintain trust. In the given blockchain framework, an edge node requests a service from the cloud node. The Cloud node runs a monitoring tool and fetches the edge servers suitable for the request. A suitable edge server willing to provide service is assigned to the request. The framework is implemented using Hyperledger fabric, and it is found that the overhead of blockchain is negligible compared to improved latency and throughput.

In [53] edge nodes provide computing resources to each other with the help of a cloud node. They are connected to a blockchain network. An edge node needing computing resources requests the service from the cloud node. The cloud node sends the details of the most suitable edge nodes to the requesting edge node. The requesting edge node runs the offloading problem and decides the edge nodes that optimize the result. The requesting node and provider node start a smart contract to allocate resources. In offloading problem, four objectives are considered: processing time, energy consumption, transaction success rate and block size in a weighted sum function to optimize. The last two objectives are related to blockchain. The work is implemented in Hyperledger fabric.

In [54], it is mentioned that an incentive should be given to edge serve providers to encourage them to share their resources with each other. A provider who has provided good services in the past should be given preference. A weighted function of reputation and latency in the resource allocation model decides a winner. The provider gets a basic reward for completing the task and an extra reward for rapid task completion. The reputation of a service provider is calculated based on past behavior. The authors developed a new consensus mechanism, namely Proof of Edge Reputation (PoER), that selects an edge server with the highest reputation as a leader. The prototype of the proposed model is implemented in Hyperledger Sawtooth.

In [55], an overloaded edge server can offload its computing jobs to some other servers belonging to other service providers. The work provides incentives so that service providers are ready to cooperate. To provide the trust among service providers, the work considers blockchain. Blockchain also acts as a coordinator among service providers to ensure cooperation. An edge server publishes a task with an associated reward in the blockchain network, and a reputable edge server satisfying the task's requirement is selected for running the task. The reputation of a server is based on its performance on offloaded computing jobs in the past. A secret leader-based byzantine fault tolerant (SL-BFT) consensus mechanism is proposed in which, based on the reputation value, 2N/3 edge servers become eligible to participate in consensus, and a leader is elected using a verifiable random function. In [56], to provide collaborative computing where edge nodes collaborate to enhance processing, blockchain is considered as a platform that provides required trust among the edge nodes. Edge nodes act as miners, also. User equipment submit their requirement, and edge nodes bid to fulfill the requirement of the user equipment. This auction process is implemented using a smart contract. All participating edge nodes have to deposit an amount. After deciding the winner, non-winning edge nodes get their deposit back. The winner gets the deposit back when reports regarding its performance are satisfactory. Edge nodes can verify the performance of the winning edge node and submit their report. If the winner does not perform satisfactorily, a certain amount from the deposited amount is given to the user equipment as compensation, and the remaining amount is given to edge nodes who identified that the winner did not perform satisfactorily. Further, the trust of each node is revised based on their performance. The work is implemented in Ethereum.

Miscellaneous

[57] uses expectation–maximization algorithm and polynomial matrix factorization to get the hash puzzle solution in fewer iterations in PoW. Experiments show that the proposed PoW based on statistical method improves memory consumption, energy consumption and convergence time to solution. Therefore, it can be used in fog computing.

In [58], ChainFasS is a blockchain-based platform to utilize underutilized personal computers as serverless computing platforms where billing depends on the execution time. Serverless computing is suitable for tasks which are not time critical. ChainFaaS is a permissioned blockchain. A consensus mechanism that does not require high energy consumption and supports smart contracts is recommended. The prototype is implemented in Hyperledger Fabric. In [59], drones are considered for caching to improve the visibility of mobile edge computing service providers, and blockchain is used to increase the trust of edge service providers and IoT devices in the system. Since data is recorded in the blockchain, it helps in achieving transparency and verifiability. An IoT device sends its computing tasks to a drone, and the drone decides the suitable edge computing node by running a smart contract. The smart contract implements the offloading policy. Since participants are limited, a permissioned private blockchain is found suitable. The work is implemented using the smart contract in Ethereum private blockchain. The gas cost analysis is performed for various offloading policies such as nearest policy, delay-aware policy, random policy, etc.

In [60], resource matching between consumers and service providers in edge computing during resource trading is done using smart contracts. The authors deployed the smart contracts in Ethereum to do the gas cost analysis. In [61], blockchain is integrated with fog computing for resource sharing and transactions. Fog user equipment and fog access point are two types of fog nodes. The user equipment can be a consumer and service provider, and access points have huge computing power; therefore, they act as service providers only. Smart contracts are used for resource matching, and deep reinforcement learning improves the system's performance. Authors considered public Ethereum as a blockchain platform and stated that the PoS consensus mechanism is better than PoW. In [62], a blockchain-based edge video streaming system is considered and formulated problem that considers offloading, resource allocation and video adaptive compression jointly that becomes a mixed-integer programming problem (MIP). Generalized Benders decomposition is used to find the solution. The work uses the Ethereum platform to write smart contracts to incentivize those edge nodes which participate in computation.

Above discussed works are summarized in Table 5.

5.2.2. Resource allocation offering blockchain frameworks

Resource allocation using deep reinforcement learning

In [63], a scenario is considered where a mobile device may have weak connectivity with edge computing nodes; therefore, it offloads its computing tasks via a relay node to the edge nodes. However, this relay node may be malicious. Therefore, a blockchain-based environment is considered where edge nodes play the role of consensus nodes. Since relay or consensus nodes may be malicious, a method to calculate their trust value is given. A resource allocation problem is formulated that maximizes the computation rate and transaction blockchain's throughput. The problem is formulated as a Markov decision process considering the dynamism in available resources and wireless channels. An asynchronous advantage actor–critic (A3C) reinforcement learning is used to solve the problem.

Auction based resource allocation

In [64], a scenario is considered in which a mobile device may require multi-task computing services from multiple edge servers, i.e., the need of a mobile device may be matched with multiple edge servers, and an edge server may provide services to multiple mobile devices. Trust, privacy and security become challenges in such an environment. Therefore, blockchain is suggested. The resource allocation problem is formulated as a double auction where the objective maximizes the system's efficiency. The double auction-based resource allocation method is implemented using a smart contract, removing the need for a centralized entity. Delegated Proof of Stake (DPoS) is suggested as a consensus mechanism, which is more democratic than PoS and more energy-saving than PoW. The work is simulated to study the performance. From the simulation, it has been shown that there is a linear relationship between storage occupied by the blockchain and the number of auction rounds, concluding that the blockchain requires minimal storage resources. The simulation also shows that blockchain reduces the probability of invalid auctions. An auction is considered invalid if trading information has been tampered with or a majority of the validators collude to tamper with the auction results.

In [65], a blockchain-aided edge computing market is considered where a data service operator rents a group of edge nodes using a smart contract that maximizes social welfare. Further, data service operators lease the edge nodes to users using a smart contract that implements social welfare improved double auction for the leasing. The authors proposed a trustworthiness-driven PoS consensus mechanism that outperforms PoW and PoS.

Computation Offloading

In [66], the authors considered a scenario where a vehicle wants computing services from the fog servers. A vehicle first generates a query to find the fog servers in its locality and selects a fog server based on its need. To create a trusted environment, fog servers are connected within a blockchain environment. A fogcoin like bitcoin can be converted in local currency for payment to fog servers, and PoW is considered a consensus mechanism. The authors simulated the work and found better average query time in blockchain compared to a centralized environment.

In [67], a user device can offload computing jobs to the device-to-device (D2D) cooperation group and fog nodes. D2D cooperation group is formed using the user devices, and the computing job is divided into N parts where N is the number of devices in the group. If the computing job is offloaded to a fog node, there is no need to break, as fog nodes have sufficient computing power. For offloading, blockchain is considered as a platform. Since blockchain adds overhead in the resource allocation process, the work minimizes network cost, where network cost depends on offloading cost and consensus cost. Further, the authors proposed a voting-based PoS consensus mechanism where a user device has one vote for one fog node and fog nodes with the top half of votes participate in the consensus process, and a miner is selected depending on the stock value of fog nodes in the verifier set like classical PoS. Simulation experiments show the work performs better in terms of network cost.

Table 5
Comparison of papers focusing on resource allocation in edge computing utilizing existing blockchain platforms.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Auction based resource allocation	[42]	To design a decentralized reverse auction-based allocation	Public	P, T, R	PoW	PoW	Ethereum	Gas cost analysis	Yes	Yes
	[43]	To maximize the utility of IoT devices	Consortium	R	PoA	NA	Ethereum	Gas cost analysis	Yes	Yes
	[44]	To design a smart contract based hierarchical auction mechanism for edge computing resource allocation	Public	R	PoW	PoW	Ethereum	Gas cost analysis	Yes	NA
	[45]	To maximize the profit of fog nodes	NA	R	NA	NA	Ethereum	Gas cost analysis	Yes	NA
	[46]	To improve utility of users and providers	NA	R	NA	NA	Ethereum	NA	Yes	NA
Credit based resource management	[47]	Credit based resource management in edge computing using blockchain	Consortium	P, R	PBFT	NA	Ethereum	Blockchain overhead in terms of computation, communication, and storage	Yes	NA
	[48]	To address vulnerability and inefficiency in the computing resource trading	NA	R	PoW	PoW	Ethereum	Gas cost analysis	Yes	NA
Trading of resources using coin	[49]	To design a decentralized fog computing resource allocation considering mobility of users	Private	R	PoW	NA	Ethereum	Average query time is analyzed	Yes	NA
	[50]	To minimize energy, computing time and digital coin	NA	P, D	NA	NA	Ethereum	Gas cost	Yes	NA
Big data processing	[51]	To design a blockchain powered decentralized computing	NA	D, P, R	PoW	PoW	Ethereum	NA	Yes	Yes
Collaboration among edge servers	[52]	To provide a trusted environment for collaboration	Consortium	R	Solo/raft	NA	Hyperledger Fabric	Latency and throughput	Yes	NA
	[53]	To maximize processing time, energy consumption, transaction success rate and block size	Consortium	R	Solo/raft	NA	Hyperledger Fabric	Latency and throughput	Yes	NA
	[54]	Selection of a winner based on reputation and latency	Consortium	R	PoER	NA	Hyperledger Sawtooth	Consensus time	NA	Yes
	[55]	To balance computing load using cooperation	Consortium	T, R	SL-BFT	NA	Hyperledger Fabric	Consensus time	NA	Yes
	[56]	To provide a trusted environment for collaboration	Public	R, T	PoW	NA	Ethereum	Incentive	Yes	Yes
Consensus mechanism	[57]	PoW based on statistical method	Public	NA	PoW	NA	Hyperledger	Memory consumption, energy consumption, convergence time to solution	NA	NA
Personal computer as serverless computing	[58]	To increase the utilization of underutilized computers	Public/ Permissioned	R	NA	NA	Hyperledger Fabric	NA	Yes	NA
Offloading in droned-aided mobile edge computing	[59]	To increase the visibility of edge service providers	Private	R	NA	NA	Ethereum	Gas cost	Yes	NA
Resource trading	[60]	To minimize latency	Consortium	R	NA	NA	Ethereum	Gas cost	Yes	NA
Resource allocation using Deep reinforcement learning	[61]	Efficient use of computing resources	Public	R	PoS/PoW	NA	Ethereum	NA	Yes	NA
Video task processing	[62]	Maximization of incentive	NA	P	NA	NA	Ethereum	NA	Yes	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

In [68], a blockchain-based edge computing environment is considered where edge devices offload the job at edge servers when they do have sufficient computing power. The work proposed Proof of Reputation consensus mechanisms where reputed edge devices act as miners. Their reputation depends on the vote of IoT devices/sensors. The work considers offloading utility and mining utility jointly to maximize the system utility. Multi-agent-based deep reinforcement learning (MA-DRL) is used to obtain the solution. MA-DRL is trained on edge servers in a centralized manner, and the edge devices execute the learned model in a decentralized manner.

Data Transactions

In [69], the authors considered a scenario where edge devices generate data and users pay to access those data. For this, a private blockchain is considered as a platform. Since edge devices have limited storage, a storage allocation method is designed that stores data items and blocks in edge devices so that they are accessed quickly. The work cache the recent blocks so that missing blacks can be accessed with less overhead. Further, meta-data is stored in the blocks instead of data because edge devices have limited storage. For each block generation, one token is created and given to the miner. The work modified the PoS consensus mechanism. In the modified consensus, instead of wealth, a node with a high value that depends on the number of tokens, data items, and blocks has a high probability of becoming a miner.

Resource allocation

In [70], to address the scalability issue in public blockchain for fog computing, a new public blockchain, Groupchain, is proposed. It follows two chain structure. To achieve high transaction efficiency, a leader group is responsible for committing the transactions in the blockchain. A node must solve the hash puzzle in PoW to enter the leader group. The size of the leader group is fixed. Therefore, when a new node solves the computational puzzle, the oldest puzzle solver in the group has to leave the group. A node generates a group block via PoW to join the group. Another type of block is viceblock. Members of the leader group can generate viceblock. Groupblock is generated after every 10 min like bitcoin, and viceblock is generated after each 1 s. A small number of people in the leader group created via groupblock to commit the transactions in the blockchain, and frequent generation of viceblock containing transactions related to IoT and fog computing helps in achieving high transaction efficiency and scalability. The work implements an incentive mechanism to regulate the behaviors of nodes in the leader group. Though the main contribution of the work is to design a scalable and high transactions efficient public blockchain, resource allocation mechanisms can be implemented above this new public blockchain. The work is simulated by extending the Bitcoin simulator [71] to show its effectiveness.

In [72], blockchain-based edge computing is considered where user devices offload tasks or rent computing services from edge servers. User devices rate edge servers based on their performance. Similarly, the reputation of user devices is calculated based on historical rating records and the number of requests. The work considers an entity Gateway that provides anonymous ID to servers and user devices and does not participate in the blockchain. Gateway will decide a threshold of reputation to filter malicious user devices and edge servers. User devices and edger servers with high reputation value participate in the consensus process. A participant with a high reputation value and high computing power becomes the leader. When a leader proposes the block in the network, other participants vote, and after getting more than 50% votes, the block becomes valid. The work implemented in a simulated environment gives satisfactory results in terms of the reputation value of participants and the time consumption of the consensus process.

Task Scheduling In [73], a task scheduling problem is formulated where the objective is the minimization of makespan and energy consumption, and the firework algorithm is used to find the solution to the problem. The work considers blockchain for security. Applications' requests and mapping results are recorded in the blockchain as transactions. DPoS is used as a consensus mechanism. It seems that the entity running the scheduling algorithm can change the mapping results. The work lacks a discussion on the implementation of blockchain

Above discussed works are summarized in Table 6.

5.2.3. Resource allocation advocating blockchain

Workflow scheduling

In [74], a blockchain-powered workflow scheduling in edge computing is proposed. In the considered workflow, tasks are dependent, i.e., a task can be executed after its parent task. Mapping of tasks in a workflow to computing nodes is workflow scheduling. In edge computing, nodes may fail, which introduces uncertainty in edge computing. Therefore, while scheduling the workflow on edge nodes, the uncertainty of edge should be considered. To consider the uncertainty in scheduling, the real-time status of nodes and used workflow scheduling scheme should be stored at some place to access them. The work suggests using a permissioned private blockchain to store the information to make workflow scheduling secure and efficient. In [75], multimedia applications are modeled as workflows. The tasks in a workflow will be offloaded or scheduled on cloudlet for processing. Latency, energy consumption and operational cost are the QoS requirement of the workflow that need to be minimized. NSGA-III (non-dominated sorting genetic Algorithm III) is applied to the optimization problem, and ELECTRE (Elimination Et Choix Tradulsant la Realtite) method is used to select the most suitable optimal scheduling strategy. Here also, the work suggests using blockchain to record the scheduling strategy to ensure the integrity of offloading decisions and related data.

Auction based resource allocation

In [76], a state prediction model using the Markov model is proposed to predict the staged destination of vehicles needing computing resources. Further, double auction-based resource allocation is formulated to maximize the satisfaction of vehicles and service providers, which is solved using an improved genetic algorithm. The work suggests using blockchain to make the model independent of the third party.

Resource allocation using deep reinforcement learning

In [77], a scenario of Industrial IoT (IIoT) is considered where edge servers provide computing resources to IIoT devices for running the application. A scheduling node collects IIoT devices' demand and edge resources' status. It applies deep reinforcement learning to decide the resource scheduling strategies that minimize the cost of resource consumption of the network. Transactions related to edge services are recorded in base stations in a consortium blockchain environment, and edge servers act as verifiers or validators. A new credit differentiated transactions approval consensus mechanism is designed for blockchain.

In [78], a scenario is considered where a data service subscriber collects data from the IoT devices and breaks the data into segments. These segments are sent to edge computing for processing. Edge computing resources are allocated to segments using asynchronous advantage actor–critic (A3C) (a machine learning algorithm) implemented in a smart contract in a private blockchain. Ethereum is recommended as a possible platform for implementation.

Credit-based resource allocation

IoT devices connected to a blockchain may suffer from two problems: cold start and long return. Cold start means a new IoT device connected to the blockchain may not have resource coins to purchase the computing resources, and long term means the consensus process is costly in terms of time. Therefore, in [79], it is proposed that an IoT device may borrow coins from other IoT devices or edge servers. A broker is a market manager who runs the double auction mechanism to resolve the conflict between borrowers and lenders. The work suggests using blockchain and PoW to record the interactions among borrowers, lenders, and brokers.

Miscellaneous

In [80], a framework for the integration of IoT, edge computing and cloud computing is developed. In [81], the offloading problem is formulated that optimizes task offloading time, energy consumption, and load balancing, and nondominated sorting genetic algorithm

Table 6
Comparison of papers focusing on resource allocation in edge computing offering blockchain frameworks.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Resource allocation using deep reinforcement learning	[63]	To maximize computation rate and blockchain transaction throughput	NA	R	dBFT	NA	Simulated environment	Effect block size, transaction size and block interval	NA	NA
Auction based resource allocation	[64]	To increase system efficiency	NA	R	DPoS	NA	Simulated environment	Storage consumption, Reduced probability of invalid auction	Yes	NA
	[65]	Maximization of social welfare	NA	R	Trustworthiness driven PoS	NA	Simulation	Rewards to stakeholders	Yes	NA
Computation offloading	[66]	To reduce query processing time	NA	R	PoW	NA	Simulation	Query processing time	NA	NA
	[67]	To minimize network cost	NA	R	Voting base PoS	NA	Simulation	Network cost	NA	NA
	[68]	To maximize offloading and mining utility jointly	NA	R	PoR	NA	Simulation	Block verification utility and system utility	Yes	NA
Data transactions	[69]	Data and block storage allocation for quick access	Private	R, D	Modifies PoS	Yes	Simulation	Power Consumption	NA	NA
Resource allocation	[70]	To design a scalable and high transactions efficient public blockchain	Public	R	PoW	Yes	Bitcoin simulator	Propagation and consensus latency, transaction throughput	NA	Yes
	[72]	To reduce time consumption of consensus process	Consortium	T	RVC	NA	Simulation	Reputation and time consumption	NA	Yes
Task scheduling	[73]	Minimization of makespan and energy consumption	NA	D	DPoS	NA	Simulation	Execution time	NA	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

III (NSGA III) is used to generate solutions. In [80,81], blockchain is recommended for providing reliability in communication and exchanging data. In [82], a widely known resource matching algorithm, i.e., deferred acceptance algorithm [83], matches the requirement of computation-intensive mobile applications with virtual machines running on the edge servers considering execution time and energy consumption. The work recommends to use smart contract to implement resource matching algorithm.

In [84], blockchain-secured fog computing architecture is given. Fog nodes are organized in clusters; a fog node cluster is viewed as a peer-to-peer network. An access control list is shared among a cluster's fog nodes, and blockchain is used to protect this list. Since fog nodes have limited computing power and storage, blockchain in a cluster can have a fixed number of maximum blocks and all fog nodes in a cluster cooperatively solve the hash puzzle. Thus, no reward is needed as fog nodes are cooperative, not competitive. The authors also proposed a method to utilize the computing resources of mobile devices and fog nodes in a cluster to solve the hash puzzle.

In [85], it is considered that nearby computing facilities can provide computing resources to mobile devices for running the applications, and a blockchain framework is introduced to allocate computing resources. A device submits its requirement as a transaction in the network. Computing facility providers submit the request–response transaction to provide the details of their offers. The selection of computing facilities depends on their performance. A provider's performance depends on its historical credits, which decrease rapidly if the provider performs malicious action and increase slowly if the provider performs honestly. PoW is considered an underlying consensus mechanism.

In [86], a blockchain-enabled resource trading mechanism is proposed where edge devices can collaborate to run IoT applications. Edge devices can be users and providers. Users send their demands, and providers send their offers to a trusted authority that verifies the certification of users and providers and deploys the smart contract to implement the resource allocation method considering resource request, resource demand and various constraints. After receiving the computation results from the providers, users have to rate the providers' reputation. In the blockchain, the transaction is composed of resource trading records and updated reputation scores. The work recommends a proof of reputation (PoR) consensus mechanism in which a node with the highest reputation score works as a miner and creates the block in the blockchain.

Above discussed works are summarized in Table 7.

Summary: In edge computing, it is expected that edge service providers with relatively low capacity compared to cloud service providers will emerge because edge computing provides computing services near to the users. They may collaborate if they cannot individually satisfy the need for computing resources. Further, vendors providing edge computing services to users may cheat or act maliciously. Many works advocated using blockchain for resource management in edge computing, but they require validation. Blockchain platforms supporting smart contracts can be used to design trustless resource management systems in edge computing. A simple architecture of blockchain-based resource allocation in distributed edge computing is given in Fig. 9, where blockchain is the platform for resource allocation.

Like bitcoin, new coins can be explicitly designed to edge computing, unlike public blockchain. It will improve payment confirmation time. Further, coins can be used for credit-based resource management systems implemented using the smart contract. Similarly, resource allocation methods, such as auctions, can be implemented in a smart contract to ensure transparency. Many works used blockchain platforms supporting smart contracts to design resource allocation systems in edge computing. However, existing blockchain platforms have their

Table 7
Comparison of papers focusing on resource allocation in edge computing advocating blockchain.

Domain	Ref	Objective	BT	BR	CM	INC	SC	SA
Workflow scheduling	[74]	To handle uncertainty in edge computing using blockchain	Permissioned private	D	NA	NA	NA	NA
	[75]	To design a blockchain based multimedia workflow scheduling in cloudlet computing	NA	D	PoW	NA	NA	NA
Auction based resource allocation	[76]	To design a decentralized edge computing resource allocation by predicating the location of vehicle	NA	R	PoW and PoS	NA	Yes	NA
Resource allocation using deep reinforcement learning	[77]	To minimize the cost of resource consumption	Consortium	T	New consensus	Yes	NA	NA
	[78]	To design a blockchain based resource allocation in edge computing using machine learning method	Private	R	NA	NA	Yes	NA
Credit based resource allocation	[79]	To solve cold start long return problem	NA	P, R	PoW	NA	NA	NA
Integration of IoT, edge computing and cloud computing	[80]	To develop a scalable, cost-effective cross platform software systems for integration of IoT, edge computing and cloud computing that performs secure operations on data.	NA	D, I	PoW	NA	NA	NA
Offloading	[81]	To optimize task offloading time, energy consumption, and load balancing	NA	D	PoW	NA	NA	NA
VM allocation	[82]	To optimize execution time and energy consumption	NA	R	NA	NA	Yes	NA
Security improvement	[84]	To design blockchain secured fog computing and minimize computing time of hash puzzle	NA	R	PoW	NA	NA	NA
Nearby computing facility allocation	[85]	A reliable resource allocation mechanism	NA	R	PoW	NA	Yes	NA
Collaboration among edge devices	[86]	To run IoT applications on edge devices	Consortium	R	PoR	NA	Yes	Yes

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

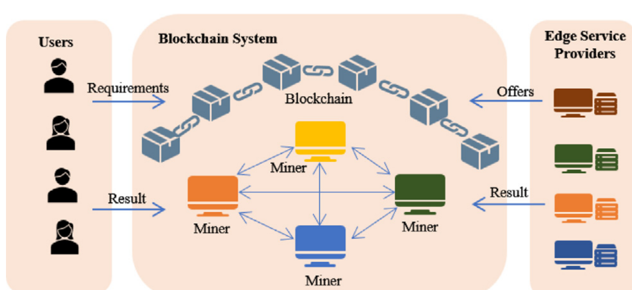

Fig. 9. Blockchain-based resource allocation in distributed edge computing.

default consensus mechanisms. Limitations of existing consensus mechanisms may have some limitations making them less suitable for resource allocation in edge computing. Therefore, few works have given new frameworks of blockchain specific to resource allocation in edge computing, addressing the issues of existing blockchain platforms and consensus mechanisms. One notable observation is that implemented resource management in a smart contract in Ethereum has polynomial time complexity. One possible reason is that running a smart contract in Ethereum involves gas costs. However, the gas cost can be controlled in the private Ethereum blockchain.

5.3. Resource allocation using blockchain in integrated cloud and edge computing

5.3.1. Resource allocation utilizing existing blockchain platforms

Resource allocation using deep reinforcement learning

In [87], all the activities related to offloading are recorded as transactions in the blockchain. A smart contract is used to implement an access control mechanism. Mobile devices submit their offloading requests to the smart contract, and requests authorized from the smart contract are recorded in the block in the blockchain. A decision-maker running in edge computing runs the offloading decision algorithm to allocate edge computing and cloud computing resources to offloading requests. Computation latency, energy consumption and gas cost of the smart contract in Ethereum are objectives in the offloading problem. The deep reinforcement learning algorithm is used to solve the offloading problem. The work is implemented in a private Ethereum blockchain, and gas cost analysis is given. The decision-maker running on edge computing makes the work prone to a single-point failure.

E-market of computing resources

In [88], a decentralized computing market using blockchain is proposed, which also opens the market for small providers. There are three types of participants: Publishers, Farmers and Auditors. A publisher publishes its task with code and expects someone will run the code and return the result. A farmer is an entity that runs the code by using its computing infrastructure. The auditor resolves the dispute between a publisher and a farmer. A publisher publishes the computational task and transfers the reward to a smart contract. The smart contract informs about the published task to farmers after validating it. A farmer ready to perform the task informs the smart contract. Smart contract reserves the task for the farmer. After finishing the task, the farmer informs the smart contract, and the smart contract informs the publisher. The farmer can withdraw the reward from the smart contract if the publisher accepts the result. If a publisher suspects that result is not right, it appeals to the auditor. The auditor re-executes the task and matches the obtained result with the submitted one by the farmer. The farmer receives the reward if the results match.

Table 8
Comparison of papers focusing on resource allocation in integrated cloud and edge computing utilizing existing blockchain platforms.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Resource allocation using deep reinforcement learning	[87]	To minimize computation latency, energy consumption, and smart contract cost	Private	I	PoW	NA	Ethereum	Gas cost	Yes	NA
E-market of computing resources	[88]	To run docker	Public	R	NA	NA	Ethereum	Gas cost	Yes	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

The farmer may vanish and enter with a new identity in the system, i.e., denial of service attack. To prevent this attack, the farmer has to deposit a fee in the smart contract to participate. The publisher also has to pay some fee to the auditor, which prevents unnecessary auditing calls from the publisher. Though the it is stated that the work could be implemented in any blockchain platform supporting the smart contract, the work is implemented in the Ethereum blockchain. The computational task is the name of the docker image and SHA-256 digest. The farmer and auditor can verify the digest of the docker image. The obtained result is compressed, and its computed hash value is returned in the smart contract that can be verified anytime. The gas cost is evaluated for running the protocol.

Above discussed works are summarized in Table 8.

5.3.2. Resource allocation offering blockchain frameworks

Broker based resource allocation

In [89], a fog computing and blockchain-enabled task scheduling model in cloud computing is proposed. A fog node acts as a broker node. Users submit their tasks, and the cloud submits capacities of virtual resources to the fog node. If the fog node cannot process the resource allocation, it assigns unique ids to tasks to maintain anonymity. The fog node generates smart contracts that include an optimization algorithm based on ant colony optimization, where the objective is the maximum utilization of machines. Miners run the smart contracts and send the result to the fog node. The fog node selects the best result and assigns virtual cloud resources to tasks accordingly. Performance is evaluated in a simulated environment. The work states that the work can be integrated with public and permissionless blockchains, such as Ethereum because users interact with the fog node only, not the blockchain network. The fog node acts as a blockchain user and submits the anonymized tasks. The fog node helps in maintaining data, identity and location privacy. One major challenge with this work is that the fog node and cloud providers are assumed as trusted entities, and it does not analyze the malicious behavior of the fog node that makes this work centralized.

Resource allocation using deep reinforcement learning

In [90], requests of vehicles for offloading are converted into blockchain transactions and sent to a cloud server, where they are authorized by the smart contract and kept in a storage pool. After authorization, vehicles can offload their tasks. A main controller running at the edge server collects the demand from the storage pool and determines where tasks will be offloaded. Low latency computing is performed at edge computing, and complicated computing is performed at cloud servers. The formulated offloading problem aims to minimize computation delay and energy consumption and maximize the system throughput (network and blockchain throughput). All transactions and data are stored in decentralized storage on the blockchain. A new consensus mechanism, redundant Byzantine fault tolerance (RBFT), is given that adds a transaction authentication process. A new deep reinforcement learning (DRL) method is used to solve the offloading problem. The major role of blockchain in this work is in authentication and decentralized storage.

Provider selection and SLA monitoring

Cloud and edge service providers provide containers or microservices to manage distributed applications. They have their unique pricing schemes, management services, and QoS levels. There is a possibility of SLA violation by the providers because of inefficient scheduling policies, inaccurate resource provisioning, unavailability of failure handling mechanism, etc. Users may also claim false SLA violations to get compensation. Therefore, identifying the culprit is a challenging job. In [91], a blockchain-based microservice provider selection protocol is proposed. The work helps in monitoring SLA in an unreliable environment. A QoS monitoring tool runs at the allocated computing resource of the provider to a user for a microservice. Suppose a user reports an SLA breakdown by a provider. Then other entities involved with the provider vote based on their collected data from the QoS monitoring tool to decide whether the user is telling the truth. The work does not discuss the blockchain type and consensus mechanism. It is mentioned that Byzantine fault tolerance could be used, seeing that number of providers and users will be a few hundred to a few thousand. The work discusses the importance of reward to those who add the block in the blockchain but does not provide any reward mechanism. Network traffic, number of messages, and stored data volume with different numbers of users and providers are observed during performance analysis.

Task Offloading

In [92], Cloud–Edge-Device (CED) computing paradigm is considered capable of meeting the users' QoS requirements. The authors considered blockchain as a platform for task offloading and proposed a new consensus mechanism, Proof of Optimum (POO). Task offloading problem is formulated as a partially observable Markov decision process that focuses on energy consumption, delay and transmission condition. In POO, this task offloading problem is given to workers as a challenge. Workers are the entities with high stake value and solve the offloading problem by applying reinforcement learning-based methods. A worker with the best solution becomes the leader. A block reward is given with each block. The leader gets half of the reward, and the remaining reward is shared with other workers. The work is in a blockchain platform BROP. POO is compared with DPoS and PBFT in terms of CPU occupancy, memory occupancy, block-building latency and bandwidth consumption. The work outperforms

Above discussed works are summarized in Table 9.

5.3.3. Resource allocation advocating blockchain

Auction based resource allocation

In [93], a double auction-based decentralized market is proposed to maximize social welfare, where both requesters and providers bid. To ensure seal bidding, participants submit the encrypted bid. Participants reveal their temporary keys in the network if a block has their encrypted bids. The miner decrypts the bid, runs the matching algorithm, and sends the block having allocation results and temporary keys. Temporary keys sent by the miner help to find whether any bid is excluded, and allocation results help in finding the accuracy. Agreement between a requester and a provider is done via a smart contract. A continuous double auction-based computing resource allocation to maximize the payoff of users and resource providers is given in [94] work considers smart contracts and blockchain to provide secure allocation without involving a centralized authority but does not elaborate process of incorporation of blockchain.

Task offloading

In [95], a resource allocation problem is formulated to optimize the energy consumption of IoT devices and the response time of computing

Table 9
Comparison of papers focusing on resource allocation in integrated cloud and edge computing offering blockchain frameworks.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Broker based task scheduling	[89]	To maximize utilization of machines	Public	R	NA	NA	NA	Evaluation of privacy, execution time and network load	Yes	NA
Resource allocation using deep reinforcement learning	[90]	To minimize computation delay and energy consumption and to maximize the system throughput	NA	D, I	RBFT	NA	Simulated environment	Block size and number of consensus nodes	Yes	No
Provider selection and SLA monitoring	[91]	To select service provider and audit contracts for SLA monitoring	NA	R, T	NA	NA	NA	Analysis of network traffic, number of messages, and stored data volume	Yes	Yes
Taks offloading	[92]	To optimize delay, energy consumption and transmission condition	Public/Private	R	Proof of Optimum	Yes	BROP	CPU occupancy, memory occupancy, block building latency and bandwidth consumption	Yes	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis; R: Resource management; D: Distributed Storage; T: Trust and reputation management; P: Payment management; I: Identity management.

Table 10
Comparison of papers focusing on resource allocation in integrated cloud and edge computing advocating blockchain.

Domain	Ref	Objective	BT	BR	CM	INC	BP	BSE	SC	SA
Auction based resource allocation	[93]	To maximize the social welfare in decentralized double auction based resource allocation	Public	R	PoW	NA	NA	NA	NA	NA
Auction based resource allocation	[94]	To maximize payoff of users and resource providers	NA	R	NA	NA	NA	NA	Yes	NA
Task offloading	[95]	To optimize energy consumption and response time	NA	D, I	NA	NA	NA	NA	No	NA

BT: Blockchain Type; BR: Blockchain Role; CM: Consensus Mechanism; INC: Incentive Mechanism; BP: Blockchain Platform; BSE: Blockchain-specific Experiments; SC: Smart Contract; SA: Security Analysis.

tasks. Cloud and edge computing resources are considered where tasks can be offloaded. The work recommends using blockchain to maintain data integrity but does not elaborate on how the blockchain will be integrated.

Above discussed works are summarized in Table 10.

Summary: Cloud computing provides computing and storage services for storing, analysis and interpretation. Since cloud service providers manage data, disclosing sensitive data is possible. Cloud computing also may cause a delay in data delivery between users and cloud servers, as cloud servers are located very far from the users. Edge computing has the potential to overcome the issues of cloud computing. In edge computing, computing services are taken from the edge nodes close to users. But nodes in edge computing have limited computing capacity; therefore, cloud computing is preferred for complicated computing. The orchestration of cloud and edge computing is necessary for the current ecosystem of devices connected via the Internet. Orchestration of cloud and edge computing creates a trustless and heterogeneous computing environment; therefore, security and privacy become challenging issues.

Blockchain can address integrated cloud and edge computing's security, privacy and distrust issues. Smart contracts can act as a broker to provide services of integrated cloud and edge computing to users and provide a trustless environment for integrating cloud service providers and edge computing service providers. Blockchain and smart contracts can provide a platform for both service providers to provide computing services to users. Using this platform, resource allocation policies can be implemented. Written allocation rules in smart contracts will ensure transparency in the allocation results. Smart contracts may also be used to implement an access control mechanism. Further, if activities related to resource allocation are recorded in the blockchain, malicious activities of participating entities in resource allocation can be identified. Further, QoS monitoring tools can be installed at allocated

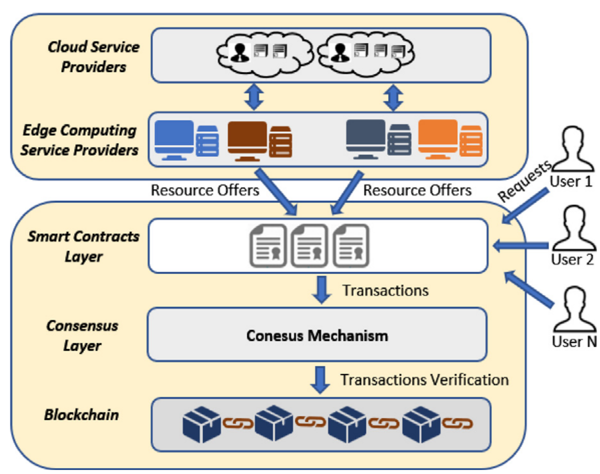

Fig. 10. Blockchain-based resource allocation in integrated cloud and distributed edge computing.

computing nodes, and collected data can be recorded in the blockchain to identify SLA violations can be identified. A pictorial representation of blockchain-based resource allocation in the integrated cloud and distributed edge computing is given in Fig. 10, where blockchain is the underlying platform for resource management, unlike Fig. 4.

5.4. Consensus mechanisms

Blockchain is composed of many components, as discussed in Section 4. Among them consensus mechanism ensures decentralization in the blockchain. Most importantly, consensus mechanisms ensure that participants agree on the current state of the ledger without involving a central authority. This section provides a discussion on consensus mechanisms used in blockchain-based resource allocation.

Many works in blockchain-based cloud computing and distributed edge computing, such as [32,34,35,42,44,48,49,70,75], used PoW as a consensus mechanism during resource allocation. In [64], DPoS is used, which is an improved version of PoS. Work in [76] used PoW and PoS alternatively to improve security. PoW and PoS require network resources to decide the leader: computation power in PoW and amount of stakes, i.e., wealth in PoS. Though PoW is a successful consensus mechanism, it is a probabilistic consensus mechanism. It provides eventual consistency and does not guarantee final consensus. Major limitations of the PoW are weak consistency, low transaction throughput, and high energy consumption [96–98]. Few works considered the PoS consensus mechanism. PoS is also a probabilistic consensus mechanism, therefore, does not guarantee final consensus. PoS can address high energy consumption and low throughput but may fail to provide full decentralization [96]. PoW provides full decentralization, but the solution to the puzzle is not useful to anyone. Further, the additional computational effort is needed to resolve the inconsistency if forking happens in the blockchain using a probabilistic consensus mechanism. Probabilistic consensus mechanisms are preferred in permissionless public blockchains and are less suitable for cloud and edge computing.

PoS is beneficial to wealthy miners and may increase wealth inequality. In [65], a modified PoS consensus is given. In modified PoS, a weighted sum of wealth, i.e., currency on stake and trustworthiness, decides the leader, where only those nodes participate in consensus who have trust values greater than a defined threshold. Similarly, in [69], instead of wealth, the number of tokens a node has and data items and blocks stored at that node is considered the value on stake, where each block creates one token. In [67], voting-based PoS is given where each user has one vote for one fog node, and fog nodes with the top half of votes act as verifiers, and the chance to become a leader depends on the stock value of the fog nodes in the verifier set like classical PoS. This modified consensus reduces the impact of the value of fog nodes at stake.

Considering the less number of participating nodes and restriction on the access of the information in cloud computing and distributed edge computing compared to a public blockchain like bitcoin and Ethereum, private and consortium blockchain with permissioned access seems more suitable to provide a trusted environment for resource allocation. Byzantine Fault Tolerance (BFT) inspired consensus mechanisms are preferred for private and consortium blockchains where the number of nodes involved in consensus is limited. They do not require network resources like PoW and PoS to decide the leader. PBFT and PoA are BFT-inspired consensus mechanisms. In PBFT, a node is randomly selected as the leader, and in PoA, leaders are generally selected in a round-robin fashion [99]. In PoA, a node may know in advance when it becomes a leader and may act maliciously. BFT-inspired consensus mechanisms also lack rewards as an encouragement to nodes involved in the consensus process. PBFT is used in resource allocation in [47]. In [63], Delegated Byzantine Fault Tolerance (dBFT) consensus mechanism is used. Unlike PBFT, in dBFT, a delegation of nodes is created that plays a role in the consensus process, and any node meeting certain requirements can become a delegate. In [90], Redundant Byzantine Fault Tolerance (RBFT) is introduced that selects participating nodes in consensus using an authentication message. Private and consortium Ethereum blockchain uses proof of authority (PoA) consensus mechanisms. Therefore, resource allocation methods developed using private or consortium Ethereum use PoA by default. In [55], a Secret leader-based byzantine fault tolerant (SL-BFT) consensus mechanism is proposed. Based on the reputation value, 2N/3 edge servers become eligible to participate in consensus, and a leader is elected using a verifiable random function. In [92], Proof of Optimum (POO) is proposed to give task offloading as a challenge to workers interested in becoming leaders and involved in the consensus process. Experiments conducted in [92] show that POO outperforms DPoS and PBFT.

Recently, a few general reputation-based and trust-based consensus mechanisms have been proposed in which the reputation or trust score of nodes involved in the consensus process act as parameters to decide the leader. For example, in [100], a set of old and well-performing nodes is selected, and a leader is elected from this subset in a round-robin fashion. A specific group of nodes is created for each node in the set to monitor the node, which calculates the reputation of the nodes. A node with a reputation value below a fixed threshold is expelled from the set. Providers form a consortium blockchain in [101]. Providers who do not submit the transactions evaluate the reputation of the providers who submit transactions. A provider with the highest reputation value becomes the leader. In resource allocation also, few works used a reputation-based consensus mechanism. In [86], proof-of-reputation (PoR) is used in which a device with the highest reputation becomes the leader and reputation is revised based on current computation and history. Similarly, in [54], an edge server with the highest reputation becomes the leader. In [41], a new consensus mechanism is introduced to save energy that selects a data center as a miner, which has the highest load and has not mined last a constant number of blocks. In [68], a Proof of Reputation consensus mechanism is proposed where the reputation of edge devices is decided based on the votes of IoT devices. Reputed edge devices participate in the consensus process like DPoS.

In [72], a Reputation and Voting based Consensus mechanism (RVC) is proposed. That calculates the reputation of participating nodes and filter malicious nodes. Among honest nodes, the node with the highest reputation and highest computing power becomes the leader. If a block proposed by a leader gets 50% votes from the nodes as a valid block, the block is added to the ledger. However, if the same node always becomes the leader, the network may move towards centralization.

In [77], considering that edge nodes may not have enough computational power required for PoW, they need some incentive as motivation to participate in the consensus process. They may want to hide their identity to maintain privacy. A new credit differentiated transactions approval consensus mechanism is proposed, in which a verifier node with the highest credit starts the transaction verification process. Other verifier nodes pay some credits for their approval. Interactions happen using anonymous message forwarding to maintain privacy. If the sum of the credit received in the approval process is above a threshold, the block is added to the blockchain and verifiers nodes involved in the approval process get the credits as a reward proportional to and higher than the credits that they spent in the approval process. Here credits decide the consensus, not the majority of verifiers.

Though PoW is a popular consensus mechanism and scales well with network size. PoW is good for permissionless blockchains. However, PoW suffers from high energy consumption, low transaction rate, and no consensus finality. PoS and proof of elapsed time (PoET) [102] are a few consensus mechanisms given by researchers that do not require intensive mining like PoW, therefore addressing the issue of high energy consumption in PoW. Further, researchers are designing consensus mechanisms that use prior wisdom. BFT-inspired consensus mechanisms come under this. For example, in PoR, reputation is a deciding factor in becoming a leader. BFT-inspired consensus mechanisms achieve high transaction capacity and consensus finality. However, they have message complexity compared to PoW. They require identity management; therefore, they are suitable for permissioned networks. Permissioned blockchain seems more suitable for resource allocation in cloud computing and distributed edge computing because recourse allocation requires consensus finality and high transaction capacity. Further, computing service providers cannot provide computing services to any users without knowing their identity. Therefore, BFT-inspired consensus mechanisms are suitable for resource allocation. However, these consensus mechanisms need to improve on high message complexity, i.e., when the number of participants is high, there will be huge communication overhead during the consensus process. To address this issue, some pre-consensus mechanisms can be designed to filter a few participants and make them eligible to participate in the consensus process. More discussion on consensus mechanisms, in general, can be found in [103–105].

5.5. Discussion

We have given an overview of several works related to blockchain-based resource allocation in the cloud and distributed edge computing. One commonality of almost all the works is that they replace the central authority for resource allocation and store meta-data/information related to resource allocation in blockchain to ensure transparency, immutability, integrity and auditability. Many works implemented or recommended to implement resource allocation methods using smart contracts. In other words, we can say smart contracts replace the central authority for resource allocation. We have also discussed the works that recommended using blockchain in resource allocation. Few works provided blockchain frameworks and consensus mechanisms specific to resource allocation addressing the challenges in existing blockchain platforms and consensus mechanisms. All discussed papers gave some novel ideas to design decentralized resource allocation using blockchain and proved that blockchain is a potential candidate to address the issues in centralized resource allocation. Important notable points observed in this survey are as follows:

Virtualization technology enables sharing economy in the field of computing. One can use virtualization technology to share computational resources in cloud computing and distributed edge computing. In both cloud computing and edge computing, brokers help in resource matching. As discussed earlier, centralization caused by brokers creates various issues, such as single-point failure and biased broker behavior. Blockchain has the potential to address the identified issues in centralized resource allocation. The Cloud computing market has already been established and is dominated by a few large players such as Google, Microsoft and Amazon. Further, cloud computing follows a centralized architecture having minimal transparency and trust settings. Therefore, a complete replacement of the centralized setting is not possible. However, small cloud service providers can create a cloud federation or joint cloud using blockchain technology, so there will be no trust issues. Brokers can create blockchain-based resource matching platforms, which will increase the trust of the users and providers in their platforms. Further, cloud service providers ensure data privacy. However, blockchain advocates complete transparency, which creates a new challenge. Centralized cloud services support scalability, while in blockchain, achieving scalability while ensuring decentralization is not easy.

The edge computing market is still growing, and blockchain applications have proved blockchain's relevance in daily life. Designing decentralization resource allocation using blockchain in edge computing will increase the acceptance of edge computing. Compared to cloud computing, distributed edge computing emerged to fulfill the strong requirement of ultra-low latency between service providers and users. The utility of tasks in edge computing depends on the time when it gets the resource. Therefore, the network parameters of the blockchain should also be considered in resource allocation. In edge computing, resources are distributed over multiple levels ranging from sensors, smartphones and IoT devices to resource-constrained devices like gateways, routers, base stations and cloudlets. Therefore, the formulation of resource allocation problems is more complex, and the process of obtaining the solution is more complex compared to cloud computing. Hence, careful consideration of resource allocation methods is needed. Further, the mobility of users and providers is also considered in edge computing. Therefore, blockchain-based resource allocation should support the joining and leaving of the nodes at any time.

Few works considered public blockchain and probabilistic consensus mechanisms for resource allocation. However, it is not expected that blockchain networks in resource allocation in cloud computing and distributed edge computing will reach the number of nodes like public blockchains such as Bitcoin and Ethereum. Further, unlike public blockchain, the network of cloud computing and distributed edge computing have authorized access to information stored in the blockchain. A limited number of participating nodes in resource allocation and limited access in cloud computing and distributed edge computing indicate that private and consortium blockchains can be considered and BFT-inspired consensus mechanisms could be used. Further, nodes participating in consensus mechanisms provide computational effort, e.g., generating new blocks. Therefore, some incentive should be given to nodes to motivate them. It will keep the network live and safe. Though, only a few works consider giving incentives.

Resource allocation problems in most of the works in cloud computing and distributed edge computing [6,8] are NP-hard. Heuristic algorithms are popular for solving NP-hard problems in cloud computing and distributed edge computing. Simple heuristics, such as greedy algorithms, are frequently utilized to give near-optimal solutions in polynomial time. Meta-heuristics such as genetic algorithm, particle swarm optimization algorithm, ant colony algorithm, and simulated annealing are also used in many works to get the solution of single objective or multi-objective problems. A survey on heuristics used in resource allocation in cloud computing and distributed computing can be found in [106,107], respectively. In works related to blockchain-based resource allocation, we can observe that many works used Ethereum or Hyperledger as platforms to implement resource allocation algorithms in smart contracts. Simple heuristics have been implemented in smart contracts for resource allocation in a few works discussed in this survey. However, no works implement meta-heuristics in smart contracts. One possible reason is that blockchain is an overhead, and running smart contracts will increase the latency. Further, in Ethereum, running smart contracts requires gas costs. Since meta-heuristics require high computational effort compared to simple heuristics, they will require higher gas costs compared to simple heuristics. However, the gas cost can be controlled in private and consortium blockchains. Blockchain technologies are still growing. Since meta-heuristics are popular in scientific community, we can expect some blockchain platforms to support the implementation of meta-heuristics in smart contracts with minimum gas cost or no cost.

Incentive compatibility is a property of an auction that ensures truthfulness, i.e., bidders reveal their true bid value. To design an incentive-compatible auction, one must first achieve individual rationality in the auction [25]. If an auction is individually rational, all bidders get non-negative utility. Suppose a service user acts as a bidder. If the user wins, the user has to pay not more than its bid value; if it loses, it does not have to pay. Similarly, if a service provider acts as a bidder in the auction, the service provider gets at least its bid value as payment for its service if it wins and in case it loses the auction, it does not lose any money. In auction-based resource allocation methods designed using smart contracts in Ethereum, if the bid submission function is implemented in the smart contract, bidders have to pay the gas cost required to process the bid submission function. Therefore, bidders who lose in the auction must suffer the gas cost required during bid submission and get negative utility. In summary, individual rationality cannot be ensured in an auction if bidders have to pay a gas cost for bid submission. Therefore, incentive computability also cannot be ensured. One should consider the above-discussed aspects of individual rationality while designing auction-based resource allocation methods.

CAP theorem states a constraint that a distributed system can deliver any two characteristics out of three desired characteristics: Consistency (C), Availability (A), and Partition Tolerance (P) [108]. Consistency means all nodes in the system have the same data. Availability means a request will always get a response. Partition Tolerance means systems work even if some nodes fail. Since only two properties can

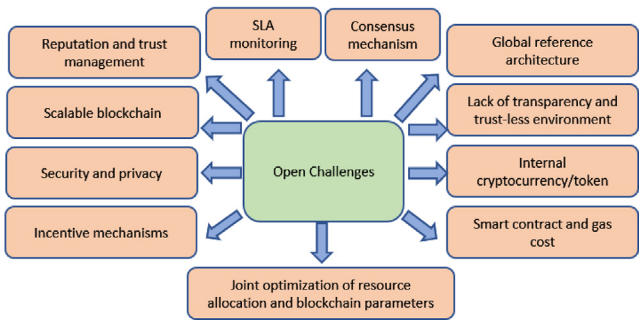

Fig. 11. Open challenges.

be satisfied, CAP theorems help in making a trade-off among the three characteristics depending on the application's requirement while designing the system. CAP theorem validates how much strong a system is based on the three properties. Blockchain is also a kind of distributed system that needs to be characterized based on the CAP theorem. In a blockchain, failure of nodes or network is inevitable; therefore, partition tolerance becomes an essential requirement. According to the CAP theorem, one has to choose between availability and consistency. In the case of bitcoin, consistency is sacrificed because availability is more important than the consistency. Though bitcoin ensures eventual consistency using its protocol, i.e., bitcoin achieves consistency over time. In summary, blockchain and its applications are complex systems. It is essential to understand the complexity and to design such systems considering the practical requirements of its users while making trade-offs. However, blockchain-based resource allocation lacks this aspect. Designing blockchain-based computing resource allocation considering the CAP theorem can unlock the full potential of blockchain.

Generally, three entities are involved in resource allocation: service providers, service users, and brokers. Service providers provide computing services, and they may have different quality of service and pricing schemes to attract users. Service users try to get the desired computing service at a lower or actual price. For users, it may be tedious to explore all service providers to select the best one. Therefore, brokers come into the picture and help service users and service providers in resource matching. Here, all three entities can act in their self-interest. Even they may act maliciously. Blockchain helps in mitigating the effects of malicious behavior of involved entities in the resource allocation process. However, only a few works provide security analysis to explain how their proposed idea mitigates the malicious actions of involved entities in resource allocation.

6. Challenges

Although a significant number of contributions have been observed in the recent past in blockchain-based computing resource allocation in cloud computing and distributed edge computing, many key challenges require attention to develop efficient decentralized resource allocation models using blockchain. The section discusses various open research challenges to provide future research directions. Fig. 11 outlines the open challenges.

6.1. Incentive mechanisms

The security of blockchain-based applications depends on the active participation of nodes competing to become a leader/a miner. Active participation of all participating nodes will prevent the domination of malicious nodes. In cloud computing and distributed edge computing, majorly computing providers will play the role of miners. Therefore, they will perform the actions strategically to increase their profit. Like bitcoin and Ethereum, the blockchain-based resource allocation model must also incorporate some incentive mechanisms to give motivation for participation. Generally, two types of incentives are given in the blockchain: block reward and transaction fee. A leader gets the block reward when it adds a new block and a transaction fee from the included transactions in the block. The role of transaction fees comes into the picture when the number of transactions is high compared to the capacity of the block. Transaction publishers compete to add their transactions to the blockchain in such situations. Though the leader gets both rewards, in resource allocation active participation of other nodes also needs some motivation.

Nodes participating in resource allocation to become leaders/miners may be given monetary and non-monetary incentives. Monetary incentive compensates the computational effort of the nodes and motivates new nodes to join, which prevents the system from attacks and malicious behavior. Non-monetary incentive in terms of reputation score or credit score helps in the cooperation of nodes. A node with a high reputation or credit score may get a high chance of becoming a leader. Non-monetary incentive ensures cooperation among untrusted entities. Though incentive mechanism is very important for security, from Tables 3–10 it can be observed that only a few works in blockchain-based computing resource allocation proposed incentive mechanisms.

Blockchain-based computing resource allocation requires specifically designed incentive mechanisms like [109], keeping in mind that leaders and non-leaders both should get some incentive based on their contribution. In [110], authors discussed incentive mechanisms in blockchain, which can be helpful in designing the incentive mechanisms in cloud computing and distributed edge computing. In computing resource allocations, users pay for the services and service providers earn. If the price of computing resources is uniform, a fixed amount may be decided as a block reward. But if the price is not uniform, e.g. price depends on users' willingness to pay in auction-based allocation, the reward for a block should depend on the total profit earned by the service providers satisfying the users' request in that block. New incentive mechanisms can be designed in future considering the number of requests included in a block and the profit service providers earn.

6.2. Security and privacy

Cloud computing follows centralized management, i.e., centrally deployed and managed. Large companies like Amazon, Microsoft, Google, etc. are cloud vendors; therefore, they have sophisticated and comprehensive security and highly reliable systems. However, in distributed edge computing, servers or computing nodes may be deployed by different owners, and they cooperate to maximize their resource utilization and profit maximization in an open ecosystem. Therefore, they have light security systems compared to the cloud. This makes the edge computing servers or nodes more vulnerable to various attacks. In blockchain-based resource allocation, decentralization, i.e., replication of ledger on various nodes, and hashing ensure consistency, accuracy and immutability. But transparency in the system, i.e., stored data visible to everyone, creates serious security and privacy concerns. There are three possible privacy leakages in resource allocation: identity, location and data [111].

As discussed earlier, in a centralized resource allocation mechanism, generally, a centralized authority is required to run the resource allocation algorithms. This centralized authority may reveal the private information of the service users and service providers. Blockchain addresses this challenge by designing decentralized resource allocation models. To achieve full decentralization, blockchain should be permissionless, causing reduced privacy of participating entities. Though privacy of identity and access control can be taken care of by the public–private key concept in permissionless blockchain and by a trusted authority in permissioned blockchain, usage of resources, kind of resource requirements, stored data and location privacy might be exposed. To preserve privacy, cryptographic techniques, zero-knowledge protocols, and multi-party computation are popular approaches that can also be explored in blockchain-based computing resource allocation.

6.3. Scalable blockchain

Cloud computing and edge computing both support many application types and are still growing. These applications may require a highly scalable environment. However, existing and widely accepted blockchain platforms still have many challenges in supporting scalable applications [112]. They do not support the real-time processing of transactions. They provide low transactions write throughputs. They are not scalable. Recently many works related to resource allocation in cloud computing and edge computing have been reported in the literature to use machine learning and deep learning techniques. However, existing blockchain platforms do not support data analytics and machine learning algorithms. Therefore, new highly scalable blockchain platforms supporting real-time processing, such as [112], are required.

6.4. Reputation and trust management

The market of cloud computing has already been established, and the market of distributed edge computing is growing. Security is a major challenge in both. Reputation and trust management are one way to secure the system. In both computing paradigms, many works related to reputational and trust management can be observed [113,114]. But most of the works calculate reputation and trust scores centrally and store them on a centralized server, which creates the possibility of error, fraud, malicious modifications, and forgery. Therefore, decentralized reputation and trust management schemes can be designed using blockchain. One major challenge with reputation and trust management schemes based on blockchain is storing reputation and trust scores and related information on blockchain that increases the storage required for blockchain. Decentralized storage systems (DSSs) based on the blockchain can be used to address this challenge. Storj [115] and InterPlanetary File System (IPFS) [116] are a few examples of DSSs.

A DSS based on blockchain is distributed peer-to-peer distributed file system. In IPFS, when a file is stored, it returns a unique hash string. One can access the file using the has string as address. This hash string can be stored in the trust and reputation management blockchain. DDSs help in reducing the data stored in the blockchain of trust and reputation management. A comprehensive survey on decentralized trust management in [117], a survey on privacy-preserving reputation systems based on blockchain in [118] and many works related to decentralized reputation and trust management in other domains such as e-commerce [119] are present in the literature. These works can be helpful in designing the decentralized reputation and trust management scheme based on blockchain in cloud computing and distributed edge computing for allocating computing resources.

A few future research directions are as follows: To evaluate the trust or reputation of computing nodes, smart contracts incorporating proper trust evaluation mechanisms can be deployed along with smart contracts for resource allocation. However, it may reduce the efficiency of the blockchain. Another possibility is to design a separate blockchain for trust management to provide the revised reputation value or trust value in the main blockchain for resource allocation. While ensuring transparency, blockchain may reveal user feedback that helps calculate trust. Privacy-preserving methods can be integrated to preserve the privacy of the users' feedback. Proper reward and punishment mechanisms that revise a computing node's trust or reputation value based on its actual behavior can be designed

6.5. Global reference architecture

In fog computing and edge computing, computing nodes cooperate to optimize their resource utilization. Various cooperation-based architectures are given in the literature [120,121]. The market of edge computing is in its nascent stage. E.g., a few edge computing service providers are offering computing resources such as vXchnge and EdgeConneX. At the same time, few help to create edge computing infrastructure and to provide underutilized computing resources on the lease, such as Mutable and MobiledgeX. Blockchain has the potential to develop computing resource allocation models where edge service providers who do not trust each other may collaborate. Resource allocation algorithms implemented in the smart contract ensure transparency in the system. Therefore, a reference architecture is needed for a successful deployment of the resource allocation model to enable collaboration across edge computing and cloud service providers, which is presently lacking. Further, in IoT, computing-based architecture involves the things layer, edge layer and cloud layer. Each layer involves heterogeneous entities, which makes resource management more complex. This complexity will be severe in blockchain-based systems if these heterogeneous entities are actively involved. The possible future direction to develop a global reference architecture is to form a consortium of industries and academic institutions interested in blockchain-based resource allocation. They can help identify the requirement for developing global reference architecture and its dissemination.

6.6. SLA monitoring

Though blockchain solves many problems of centralized resource allocation models, the resource allocation scenario is still competitive and heterogeneous. Therefore, blockchain-based resource allocation inherits the challenges caused by heterogeneity and the competitive environment. Multiple providers compete by offering various types of services using various pricing schemes. Providers may perform malicious actions to increase profit, such as overbooking and poor quality of service than promised. Similarly, users may work in their self-interest, such as disturbing the market to reduce the price and making false claims of providing poor QoS. Therefore, reliable contracts are needed that not only help in the selection of the service providers but also offer to audit them. Though various works related to blockchain-based resource allocation models used smart contracts for resource allocation, they lack contracts related to defining and auditing the service level agreement. This lacking may cause monitory loss depending on the applications and intensities.

SLAs define the agreement between the user and the provider based on identified service level objective. Generally, users and providers negotiate to define SLAs. Once an SLA is defined, providers may have to compensate if they violate the SLA. To automate the SLA management, SLA monitoring parties or tools are involved in validating if a provider violates the SLA. Generally, these tools or parties are centralized entities. Therefore, they inherit the challenges of a centralized system. Based on the above discussion, the following future direction is given. The negotiation process to formalize the SLAs can be off-chain. Once SLAs are finalized, they can be implemented via smart contracts. Providers or customers cannot deny the agreement. Further, more than one monitoring entity can be included in the blockchain, and actions can be taken based on the consensus. It opens a new direction where decentralized SLA management can be designed using a separate blockchain, which reduces the overhead on the main blockchain for resource allocation.

6.7. Consensus mechanism

6.7.1. Energy consumption

Many works related to blockchain-based cloud and distributed edge computing used the popular consensus mechanism PoW or hybrid PoW. They consume high energy and computing resources and may take a long time. PoW is also a probabilistic consensus mechanism, i.e., no guarantee of consensus finality, as discussed before. PoW is good for public blockchain or permissionless blockchain. But in the case of computing resource allocation, consortium blockchain is more suitable compared to the public blockchain. Therefore, new consensus mechanisms requiring less energy consumption and computing power and taking less execution time are needed. Further, PoW requires computing resources. Since edge service providers will play the role of miners and have limited computing power, a quantitative analysis of the required computing power for PoW in resource allocation is also needed.

6.7.2. Latency

Existing blockchain systems are suffering from low throughput and high latency. In blockchain-based cloud and distributed edge computing, the allocation of computing resources may take some time. But after allocation, the user has to take services directly from the assigned provider. Service placement and virtual machine allocation are feasible in the blockchain-based cloud and distributed edge computing. However, offloading of real-time or delay-sensitive tasks may suffer. Therefore, the consensus mechanism must be lightweight, and the decision on the size of the block becomes an important issue. Proof of Elapsed Time (PoET) is a time-based lightweight consensus mechanism in which each node runs a code that awakens the node after a random time interval. A node that first becomes awake becomes the leader. PoET does not require computing effort like PoW. PoET is more reactive and has very less complexity. PoET is suitable for applications where nodes have limited computing power or are prone to failure [102]. Like PoET, other lightweight consensus mechanisms need to be studied in the context of computing resource allocation.

6.7.3. No useful work in consensus mechanism

PoW is a popular and good consensus mechanism. The popularity and success of bitcoin prove this. In PoW, nodes compete to solve a computational hash puzzle and the node which gives the solution becomes the miner. Finding the solution may require huge computing power and energy consumption, which makes it not suitable for every application of blockchain. The solution to the puzzle only proves that someone has solved the puzzle. The solution is not useful for anything. In [122], authors proposed a Proof of Useful Work (uPoW) consensus mechanism to use hard computational problems, e.g., 3SUM problem and all-pairs shortest path problem, relevant to the scientific community in place of computation puzzle. Solutions to these problems have practical interest and can also be verified quickly. In [109], authors proposed a new consensus mechanism that considers the service placement problem as the problem that nodes compete to solve. A node with the best solution in a fixed time becomes the miner. But a full framework is not discussed. In summary, existing new consensus mechanisms should be explored, and new consensus mechanisms specific to computing resource allocation should be designed to achieve better latency and throughput, requiring less computation, storage and bandwidth.

Based on the above discussion on energy, latency and no useful work issues in the consensus mechanism, we can identify a few future directions. To reduce the energy consumption and latency problem in PoW, new consensus mechanisms can be designed by integrating PoW with some other mechanism that reduces the complexity of the computational puzzle. For example, In [123], the puzzle's complexity in PoW depends on a node's reputation score. In the computing market, the number of computing service providers is not very high, and users are less likely to participate in the mining process. Further, a service provider cannot blindly provide computing resources to anyone. Therefore, we believe that the blockchain-based computing resource allocation system will be a permissioned one. After resolving the issues in BFT-inspired consensus mechanisms, such as incorporating proper incentive mechanisms, they can be used in permissioned blockchain-based computing resource allocation.

6.8. Internal cryptocurrency/token

Cryptocurrency is a popular application of blockchain. Even blockchain became popular after bitcoin. There are various benefits of cryptocurrency because of cryptography and blockchain. Decentralization in cryptocurrency makes payment secure. It can be used as a medium of exchange without involving any third party. It has no national border; one can send it to others in a different country without difficulty. Cryptocurrency transfer is faster compared to wire transfers. Transparency in cryptocurrency reduces fraudulent transactions. The legitimacy of a transaction can be easily proved. In computing resource allocation, payment can be secured if some existing cryptocurrency is used. But payment will be more secure if an internal cryptocurrency or token system is designed. It will inherit the security features of the existing cryptocurrency, and participating nodes have control over it, i.e., transactions will be stored in the blockchain, which is a secure and non-modifiable database, and auditing will be possible [47]. The internal currency will enable the smooth and automated execution of smart contracts forming the resource allocation rules. Therefore, implementing internal cryptocurrency or token and its distribution is an open challenge.

Considering the need for new cryptocurrency in blockchain-based resource allocation, there are three possible ways to create a new cryptocurrency. The first is creating a new blockchain platform and its native cryptocurrency. Most of the blockchain platforms are open access. The second possible way to create a new cryptocurrency is to modify the code of the existing blockchain platform. Many blockchain platforms, such as Ethereum and Ripple, allow users to create a new cryptocurrency on their existing platforms, which is the third way. Generally, the cryptocurrency created on an existing platform is known as a token. Among these three possible ways, the last is the easiest. However, the new cryptocurrency will directly depend on the underline blockchain platform. The first one is the most complicated because one has to start from scratch and design everything, such as blockchain architecture and consensus mechanism

6.9. Smart contract and gas cost

Most works implement resource allocation algorithms in smart contracts to enforce necessary resource allocation rules and eliminate the need for a central authority. The reason behind using the smart contract is its decentralized execution, because of which malicious actions of service users and service providers performed intentionally, such as making ingenuine requests or offers, erratic behavior, denial of service, etc., are not possible. A smart contract is executed based on the initially programmed policies. In Ethereum, if service users and service providers behave maliciously, they have to pay in terms of gas cost, as in Ethereum, the execution of a smart contract or any computing effort needs payment from the participant. If rules in the smart contract are properly formed, malicious entities can be identified. They can be blocked for further actions since all operations in the smart contract are recorded as transactions in the blockchain. Hyperledger and Ethereum blockchains are mostly used to implement computing resource allocation models. Hyperledger does not require execution costs for operations in smart contracts.

Execution cost, popularly known as gas cost, for computational effort in Ethereum, makes it a powerful tool for developing blockchain-based decentralized applications, also known as Dapp. However, most of the work lacks a depth analysis of the role of gas cost in computing resource allocation. In public Ethereum, if one wants to run any operation in the smart contract, one has to pay the gas cost that will be converted into Ether. Therefore, if a resource allocation algorithm is implemented in the smart contract in public Ethereum, each participating entity has to pay a gas cost in terms of Ether. For example, to request the computing service user has to run a service request function in the smart contract, which will require some gas cost, i.e., payment in terms of Ether.

One has to understand that if payment of gas cost by the participants is not feasible, deployment of smart contract in public Ethereum will be meaningless. One such analysis is given in [32]. Further, the price of Ether is also volatile; therefore, the analysis may be irrelevant in future. Further, Ethereum also supports consortium blockchain where the value of gas cost can be controlled, and even fee-free consortium blockchain can be deployed. Computing resource allocation problems are generally NP-hard problems that may require the implementation of heuristics and meta-heuristics in smart contracts requiring a huge gas cost. An analysis of the gas cost of these algorithms is lacking. Further, an efficient resource allocation algorithm that does not only allocate resources efficiently but also requires less gas cost is an open problem.

Greedy resource allocation methods are generally preferred when a quick decision is needed because their computational complexity is lower. Gas cost depends on the computation effort required to run the written program in the smart contract. In blockchain-based systems, computation complexity includes time complexity and space complexity. Greedy resource allocation methods can also be preferred in blockchain-based systems. They will not only reduce the gas cost but may also play a role in reducing the overhead caused by the decentralization in the blockchain. Assume that two resource allocation methods are giving similar results. Here the method which requires less gas cost will be preferred. In non-blockchain-based systems, resource allocation methods do not consider space complexity. However, new greedy resource allocation methods can be designed in blockchain-based systems considering time and space complexity.

6.10. Joint optimization of resource allocation and blockchain parameters

During computing resource allocation, various parameters are considered, such as latency, resource utilization, profit of service providers, the satisfaction of service users, energy, etc. When blockchain-based resource allocation systems are designed, various blockchain-based parameters, such as throughput, block size, block time, block reward, etc., come into the picture. Throughput represents the number of committed transactions per unit of time. Block size refers to the size of the block, i.e., the number of transactions included in the block. Block time refers to the average time required to publish a block in the network, and block reward refers to the reward earned by the leader to publish a new block. If PoW is used as a consensus mechanism, service providers may have to involve their computing resources in the consensus mechanism to solve the puzzle. In the case of other consensus mechanisms, some computing effort is still needed, such as for transaction validation, leader election, block validation, smart contract execution etc. These computing efforts are given by the service providers from their computing resources. Blockchain-based parameters are important for the success of any blockchain-based application.

Distributed edge computing reduces the latency of delay-sensitive applications. The use of blockchain for computing resource allocation ensures decentralization, transparency, immutability, etc.. However, these characteristics require various overheads such as publication of transactions and blocks in the network, processing of consensus mechanism, etc., which may increase the delay in executing applications. Therefore, blockchain-related parameters should also be considered in blockchain-based resource allocation. E.g., suppose the number of requests and offers coming per unit of time is very high compared to the block size. In that case, most of the requests and offers will not be allocated within a limited time, increasing dissatisfaction among service providers and users towards the blockchain-based system. Further, if the block size is very large, it will take time to propagate the block in the network. In the case of latency-sensitive applications, a large size block may not give the results in a limited time to the users. Therefore, some optimization techniques are needed to decide the values of blockchain-based parameters considering computing resource allocation. Joint optimization of resource allocation and blockchain parameters is an open challenge in blockchain-based computing resource allocation.

To address this challenge, a possible future direction is to formalize the resource allocation problem considering blockchain and resource allocation parameters together. It may increase the complexity of the resource allocation problem. We have to design solutions or optimization techniques specific to these joint resource allocation problems to decide the values of blockchain-based and resource allocation parameters.

6.11. Simulator

Blockchain is a complex system, and its applications are becoming popular daily. Many new blockchain-based applications or blockchains specific to some applications are being developed. Such efforts require their validation. However, it would be difficult to validate the blockchain in a repeatable manner under different configurations in real environment, as building real infrastructure for the blockchain is expensive, tedious and time-consuming. Simulation tools are alternatives to create reliable, controllable, and flexible simulation environments with low cost, mimicking real-life scenarios. Reproduction of results will also be easy. To simulate cloud computing and distributed edge computing, various simulation tools such as CloudSim [124], IoTSim-Edge [125], ECSim++ [126], iFogSim [127], YAFS [128] are present, but they do not simulate the blockchain-based environment. To simulate blockchain, various simulators like BlockSim [129], Simblock [130], Vibes [131] are popular, but they do not support the simulation of computing resource allocation. Seeing the recent developments in blockchain-based computing resource allocation, a simulator is needed to validate blockchain-based computing resource allocation models before deploying them in a real environment.

Based on the above discussion, we believe that there are three possible directions for future work. One can design a new simulator from scratch or extend an existing simulator in blockchain or computing after studying the simulators in the other field. The third possible direction is integrating simulators in computing and blockchain, which are based on a common platform. For example, YAFS and Blocksim are Python-based simulators and can be explored for possible integration.

7. Conclusion

This paper presents a comprehensive survey on blockchain-based resource allocation in cloud computing and distributed edge computing. First, we identify several key research questions that motivate this survey. Then, we discuss centralized resource allocation in cloud computing and distributed edge computing and associated challenges. Further, we discuss blockchain, its structure, working, characteristics and types, followed by its benefits to resource allocation. Discussion on resource allocation and blockchain provides background for this survey. Then we present a depth overview of blockchain-based resource allocation. We summarize the works in three domains: cloud computing, distributed edge computing and integrated edge and cloud computing. In each domain, further works are summarized from three aspects: works using blockchain platforms, works providing blockchain frameworks and works advocating blockchain. We identify several comparison metrics to provide a comparative study of the works. Since consensus mechanism is the backbone of blockchain, we discuss consensus mechanisms used in blockchain-based resource allocation. Further, we shed light on the key challenges in blockchain-based resource allocation that require our attention and research investment.

A few important points concluded from this survey are as follows:

- Using blockchain, decentralization, immutability, and non-repudiation can be achieved in resource allocation in distributed edge computing and cloud computing.
- Blockchain ensures trustfulness, data integrity and data availability.
- Smart contracts can be used to implement resource allocation methods and for autonomous control.

- Blockchain makes resource allocation models more robust by addressing the single-point failure challenge in conventional systems.
- Use of blockchain in resource allocation makes the whole system transparent; therefore, privacy becomes challenging.
- Though blockchain ensures decentralization, consortium blockchain seems the most suitable model for resource allocation in distributed edge computing and cloud computing markets.
- Since only verified participants can participate in consortium blockchain, some central authority is needed for verification.
- Decentralized systems cannot fully replace centralized systems.
- Cryptography technologies and hashing used in blockchain make the resource allocation model more secure.
- Though few works used PoW as consensus mechanisms for resource allocation, seeing the suitability of consortium blockchain for resource allocation, other lightweight consensus mechanisms, such as PoA, PBFT, PoET, are more suitable than PoW.
- New consensus mechanisms specific to resource allocation can be designed.
- Blockchain parameters should also be considered in blockchain-based resource allocation models.
- Blockchain-based resource allocation models involving transaction processing fees, like gas cost in Ethereum or transaction fee in bitcoin and Ethereum, will be acceptable only when the transaction processing fee will not be significantly high compared to conventional resource allocation models.
- Blockchain resource allocation model adds overhead compared to a centralized system to ensure decentralization, security, immutability, etc. Therefore, it must be cost-effective. An analysis of cost-effectiveness is needed.
- Some incentive should be given to nodes participating in the consensus process. Incentives can be monetary or reputation based.

CRediT authorship contribution statement

Gaurav Baranwal: Conceptualization, Data curation, Formal analysis, Methodology, Investigation, Resources, Validation, Visualization, Writing – original draft, Writing – review & editing. **Dinesh Kumar:** Formal analysis, Methodology, Investigation, Resources, Validation, Visualization, Writing – review & editing. **Deo Prakash Vidyarthi:** Methodology, Investigation, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, Ivona Brandic, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Gener. Comput. Syst. 25 (6) (2009) 599–616.

[2] Helin Yang, Arokiaswami Alphones, Wen-De Zhong, Chen Chen, Xianzhong Xie, Learning-based energy-efficient resource management by heterogeneous RF/VLC for ultra-reliable low-latency industrial IoT networks, IEEE Trans. Ind. Inform. 16 (8) (2019) 5565–5576.

[3] Zheng Ma, Ming Xiao, Yue Xiao, Zhibo Pang, H Vincent Poor, Branka Vucetic, High-reliability and low-latency wireless communication for internet of things: Challenges, fundamentals, and enabling technologies, IEEE Internet Things J. 6 (5) (2019) 7946–7970.

[4] Cisco, Cisco and SAS edge-to-enterprise IoT analytics platform, 2023, https://www.cisco.com/.

[5] Xingjia Li, Li Pan, Shijun Liu, A survey of resource provisioning problem in cloud brokers, J. Netw. Comput. Appl. (2022) 103384.

[6] Nguyen Cong Luong, Ping Wang, Dusit Niyato, Yonggang Wen, Zhu Han, Resource management in cloud networking using economic analysis and pricing models: A survey, IEEE Commun. Surv. Tutor. 19 (2) (2017) 954–1001.

[7] Ismael Martinez, Abdelhakim Senhaji Hafid, Abdallah Jarray, Design, resource management, and evaluation of fog computing systems: a survey, IEEE Internet Things J. 8 (4) (2020) 2494–2516.

[8] Quyuan Luo, Shihong Hu, Changle Li, Guanghui Li, Weisong Shi, Resource scheduling in edge computing: A survey, IEEE Commun. Surv. Tutor. 23 (4) (2021) 2131–2165.

[9] Chuan Feng, Pengchao Han, Xu Zhang, Bowen Yang, Yejun Liu, Lei Guo, Computation offloading in mobile edge computing networks: A survey, J. Netw. Comput. Appl. (2022) 103366.

[10] Pavel Mach, Zdenek Becvar, Mobile edge computing: A survey on architecture and computation offloading, IEEE Commun. Surv. Tutor. 19 (3) (2017) 1628–1656.

[11] Stuart Haber, W. Scott Stornetta, How to Time-Stamp a Digital Document, Springer, 1991.

[12] Satoshi Nakamoto, Bitcoin: A peer-to-peer electronic cash system, Decentralized Bus. Rev. (2008) 21260.

[13] Qiheng Zhou, Huawei Huang, Zibin Zheng, Jing Bian, Solutions to scalability of blockchain: A survey, Ieee Access 8 (2020) 16440–16455.

[14] Ruizhe Yang, F Richard Yu, Pengbo Si, Zhaoxin Yang, Yanhua Zhang, Integrated blockchain and edge computing systems: A survey, some research issues and challenges, IEEE Commun. Surv. Tutor. 21 (2) (2019) 1508–1532.

[15] Mallikarjun Reddy Dorsala, V.N. Sastry, Sudhakar Chapram, Blockchain-based solutions for cloud computing: A survey, J. Netw. Comput. Appl. 196 (2021) 103246.

[16] Wenjuan Li, Jiyi Wu, Jian Cao, Nan Chen, Qifei Zhang, Rajkumar Buyya, Blockchain-based trust management in cloud computing systems: a taxonomy, review and future directions, J. Cloud Comput. 10 (1) (2021) 1–34.

[17] Osama Alkadi, Nour Moustafa, Benjamin Turnbull, A review of intrusion detection and blockchain applications in the cloud: approaches, challenges and solutions, IEEE Access 8 (2020) 104893–104917.

[18] Hamza Baniata, Attila Kertesz, A survey on blockchain-fog integration approaches, IEEE Access 8 (2020) 102657–102668.

[19] Ch VNU Bharathi Murthy, M Lawanya Shri, Seifedine Kadry, Sangsoon Lim, Blockchain based cloud computing: Architecture and research challenges, IEEE Access 8 (2020) 205190–205205.

[20] Kimchai Yeow, Abdullah Gani, Raja Wasim Ahmad, Joel JPC Rodrigues, Kwangman Ko, Decentralized consensus for edge-centric internet of things: A review, taxonomy, and research issues, IEEE Access 6 (2017) 1513–1524.

[21] Showkat Ahmad Bhat, Ishfaq Bashir Sofi, Chong-Yung Chi, Edge computing and its convergence with blockchain in 5G and beyond: security, challenges, and opportunities, IEEE Access 8 (2020) 205340–205373.

[22] Dinh C Nguyen, Pubudu N Pathirana, Ming Ding, Aruna Seneviratne, Integration of blockchain and cloud of things: Architecture, applications and challenges, IEEE Commun. Surv. Tutor. 22 (4) (2020) 2521–2549.

[23] Keke Gai, Jinnan Guo, Liehuang Zhu, Shui Yu, Blockchain meets cloud computing: A survey, IEEE Commun. Surv. Tutor. 22 (3) (2020) 2009–2030.

[24] Xing Liu, Towards blockchain-based resource allocation models for cloud-edge computing in IoT applications, Wirel. Pers. Commun. (2021) 1–19.

[25] Gaurav Baranwal, Dinesh Kumar, Zahid Raza, Deo Prakash Vidyarthi, Auction based Resource Provisioning in Cloud Computing, Springer, 2018.

[26] Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, Nigel Davies, The case for vm-based cloudlets in mobile computing, IEEE Pervasive Comput. 8 (4) (2009) 14–23.

[27] Guoju Gao, Mingjun Xiao, Jie Wu, He Huang, Shengqi Wang, Guoliang Chen, Auction-based VM allocation for deadline-sensitive tasks in distributed edge cloud, IEEE Trans. Serv. Comput. 14 (6) (2019) 1702–1716.

[28] Tayebeh Bahreini, Hossein Badri, Daniel Grosu, An envy-free auction mechanism for resource allocation in edge computing systems, in: 2018 IEEE/ACM Symposium on Edge Computing, SEC, IEEE, 2018, pp. 313–322.

[29] Katerina Doka, Tasos Bakogiannis, Ioannis Mytilinis, Georgios Goumas, Cloudagora: Democratizing the cloud, in: Blockchain–ICBC 2019: Second International Conference, Held As Part of the Services Conference Federation, SCF 2019, San Diego, CA, USA, June 25–30, 2019, Proceedings 2, Springer, 2019, pp. 142–156.

[30] Yonggen Gu, Dingding Hou, Xiaohong Wu, A cloud storage resource transaction mechanism based on smart contract, in: Proceedings of the 8th International Conference on Communication and Network Security, 2018, pp. 134–138.

[31] Zhili Chen, Wei Ding, Yan Xu, Miaomiao Tian, Hong Zhong, Fair auctioning and trading framework for cloud virtual machines based on blockchain, Comput. Commun. 171 (2021) 89–98.

[32] Zeshun Shi, Huan Zhou, Cees de Laat, Zhiming Zhao, A bayesian game-enhanced auction model for federated cloud services using blockchain, Future Gener. Comput. Syst. 136 (2022) 49–66.

[33] Xuyang Ma, Du Xu, Katinka Wolter, Blockchain-enabled feedback-based combinatorial double auction for cloud markets, Future Gener. Comput. Syst. 127 (2022) 225–239.

[34] Mona Taghavi, Jamal Bentahar, Hadi Otrok, Kaveh Bakhtiyari, Cloudchain: A blockchain-based coopetition differential game model for cloud computing, in: Service-Oriented Computing: 16th International Conference, ICSOC 2018, Hangzhou, China, November 12-15, 2018, Proceedings 16, Springer, 2018, pp. 146–161.

[35] Mona Taghavi, Jamal Bentahar, Hadi Otrok, Kaveh Bakhtiyari, A blockchain-based model for cloud service quality monitoring, IEEE Trans. Serv. Comput. 13 (2) (2019) 276–288.

[36] Hui Yang, Jiaqi Yuan, Haipeng Yao, Qiuyan Yao, Ao Yu, Jie Zhang, Blockchain-based hierarchical trust networking for JointCloud, IEEE Internet Things J. 7 (3) (2019) 1667–1677.

[37] Vorameth Reantongcome, Vasaka Visoottiviseth, Wudhichart Sawangphol, Assadarat Khurat, Shigeru Kashihara, Doudou Fall, Securing and trustworthy blockchain-based multi-tenant cloud computing, in: 2020 IEEE 10th Symposium on Computer Applications & Industrial Electronics, ISCAIE, IEEE, 2020, pp. 256–261.

[38] Sambit Nayak, Nanjangud C Narendra, Anshu Shukla, James Kempf, Saranyu: Using smart contracts and blockchain for cloud tenant management, in: 2018 IEEE 11th International Conference on Cloud Computing, CLOUD, IEEE, 2018, pp. 857–861.

[39] Andrzej Wilczyński, Joanna Kołodziej, Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology, Simul. Model. Pract. Theory 99 (2020) 102038.

[40] Benedikt Pittl, Werner Mach, Erich Schikuta, Bazaar-blockchain: A blockchain for bazaar-based cloud markets, in: 2018 IEEE International Conference on Services Computing, SCC, IEEE, 2018, pp. 89–96.

[41] Chenhan Xu, Kun Wang, Mingyi Guo, Intelligent resource management in blockchain-based cloud datacenters, IEEE Cloud Comput. 4 (6) (2017) 50–59.

[42] Mazin Debe, Khaled Salah, Muhammad Habib Ur Rehman, Davor Svetinovic, Blockchain-based decentralized reverse bidding in fog computing, IEEE Access 8 (2020) 81686–81697.

[43] Gaurav Baranwal, Dinesh Kumar, Deo Prakash Vidyarthi, BARA: A blockchain-aided auction-based resource allocation in edge computing enabled industrial internet of things, Future Gener. Comput. Syst. 135 (2022) 333–347.

[44] Hui Lin, Zetao Yang, Zicong Hong, Shenghui Li, Wuhui Chen, Smart contract-based hierarchical auction mechanism for edge computing in blockchain-empowered IoT, in: 2020 IEEE 21st International Symposium on" a World of Wireless, Mobile and Multimedia Networks"(WoWMoM), IEEE, 2020, pp. 147–156.

[45] Vibha Jain, Bijendra Kumar, Combinatorial auction based multi-task resource allocation in fog environment using blockchain and smart contracts, Peer-to-Peer Netw. Appl. 14 (2021) 3124–3142.

[46] Tonglai Liu, Jigang Wu, Long Chen, Yalan Wu, Yinan Li, Smart contract-based long-term auction for mobile blockchain computation offloading, IEEE Access 8 (2020) 36029–36042.

[47] Jianli Pan, Jianyu Wang, Austin Hester, Ismail Alqerm, Yuanni Liu, Ying Zhao, EdgeChain: An edge-IoT framework and prototype based on blockchain and smart contracts, IEEE Internet Things J. 6 (3) (2018) 4719–4732.

[48] Zetao Yang, Kang Liu, Yufei Chen, Wuhui Chen, Mingdong Tang, Two-level stackelberg game for IoT computational resource trading mechanism: A smart contract approach, IEEE Trans. Serv. Comput. 15 (4) (2020) 1883–1895.

[49] Wanchun Dou, Wenda Tang, Bowen Liu, Xiaolong Xu, Qiang Ni, Blockchain-based mobility-aware offloading mechanism for fog computing services, Comput. Commun. 164 (2020) 261–273.

[50] Zhen Zhang, Zicong Hong, Wuhui Chen, Zibin Zheng, Xu Chen, Joint computation offloading and coin loaning for blockchain-empowered mobile-edge computing, IEEE Internet Things J. 6 (6) (2019) 9934–9950.

[51] Gihan J Mendis, Yifu Wu, Jin Wei, Moein Sabounchi, Rigoberto Roche, A blockchain-powered decentralized and secure computing paradigm, IEEE Trans. Emerg. Top. Comput. 9 (4) (2020) 2201–2222.

[52] Angelo Vera Rivera, Ahmed Refaey, Ekram Hossain, A blockchain framework for secure task sharing in multi-access edge computing, IEEE Netw. 35 (3) (2020) 176–183.

[53] Angelo Vera-Rivera, Ahmed Refaey, Ekram Hossain, Blockchain-based collaborative task offloading in MEC: A hyperledger fabric framework, in: 2021 IEEE International Conference on Communications Workshops (ICC Workshops), IEEE, 2021, pp. 1–6.

[54] Liang Yuan, Qiang He, Siyu Tan, Bo Li, Jiangshan Yu, Feifei Chen, Hai Jin, Yun Yang, Coopedge: A decentralized blockchain-based platform for cooperative edge computing, in: Proceedings of the Web Conference 2021, 2021, pp. 2245–2257.

[55] Liang Yuan, Qiang He, Siyu Tan, Bo Li, Jiangshan Yu, Feifei Chen, Yun Yang, CoopEdge+: Enabling decentralized, secure and cooperative multi-access edge computing based on blockchain, IEEE Trans. Parallel Distrib. Syst. (2022).

[56] Bo Wu, Ke Xu, Qi Li, Shoushou Ren, Zhuotao Liu, Zhichao Zhang, Toward blockchain-powered trusted collaborative services for edge-centric networks, IEEE Netw. 34 (2) (2020) 30–36.

[57] Gulshan Kumar, Rahul Saha, Mritunjay Kumar Rai, Reji Thomas, Tai-Hoon Kim, Proof-of-work consensus approach in blockchain technology for cloud and fog computing using maximization-factorization statistics, IEEE Internet Things J. 6 (4) (2019) 6835–6842.

[58] Sara Ghaemi, Hamzeh Khazaei, Petr Musilek, Chainfaas: An open blockchain-based serverless platform, IEEE Access 8 (2020) 131760–131778.

[59] Shuyun Luo, Hang Li, Zhenyu Wen, Bin Qian, Graham Morgan, Antonella Longo, Omer Rana, Rajiv Ranjan, Blockchain-based task offloading in drone-aided mobile edge computing, IEEE Netw. 35 (1) (2021) 124–129.

[60] Jinyue Song, Tianbo Gu, Yunjie Ge, Prasant Mohapatra, Smart contract-based computing resources trading in edge computing, in: 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, IEEE, 2020, pp. 1–7.

[61] Yang Gao, Wenjun Wu, Pengbo Si, Zhaoxin Yang, Fei Richard Yu, B-ReST: Blockchain-enabled resource sharing and transactions in fog computing, IEEE Wirel. Commun. 28 (2) (2021) 172–180.

[62] Shijing Yuan, Jie Li, Chentao Wu, JORA: Blockchain-based efficient joint computing offloading and resource allocation for edge video streaming systems, J. Syst. Archit. 133 (2022) 102740.

[63] Jie Feng, F Richard Yu, Qingqi Pei, Xiaoli Chu, Jianbo Du, Li Zhu, Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: A deep reinforcement learning approach, IEEE Internet Things J. 7 (7) (2019) 6214–6228.

[64] Wen Sun, Jiajia Liu, Yanlin Yue, Peng Wang, Joint resource allocation and incentive design for blockchain-based mobile edge computing, IEEE Trans. Wirel. Commun. 19 (9) (2020) 6050–6064.

[65] Yu Du, Zhe Wang, Jun Li, Long Shi, Dushantha Nalin K Jayakody, Quan Chen, Wen Chen, Zhu Han, Blockchain-aided edge computing market: Smart contract and consensus mechanisms, IEEE Trans. Mob. Comput. (2022).

[66] Wenda Tang, Xuan Zhao, Wajid Rafique, Wanchun Dou, A blockchain-based offloading approach in fog computing environment, in: 2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom), IEEE, 2018, pp. 308–315.

[67] Xiaoge Huang, Xin Liu, Qianbin Chen, Jie Zhang, Resource allocation and task offloading in blockchain-enabled fog computing networks, in: 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), IEEE, 2021, pp. 01–05.

[68] Dinh Nguyen, Ming Ding, Pubudu Pathirana, Aruna Seneviratne, Jun Li, Vincent Poor, Cooperative task offloading and block mining in blockchain-based edge computing with multi-agent deep reinforcement learning, IEEE Trans. Mob. Comput. (2021).

[69] Yaodong Huang, Jiarui Zhang, Jun Duan, Bin Xiao, Fan Ye, Yuanyuan Yang, Resource allocation and consensus of blockchains in pervasive edge computing environments, IEEE Trans. Mob. Comput. 21 (9) (2021) 3298–3311.

[70] Kai Lei, Maoyu Du, Jiyue Huang, Tong Jin, Groupchain: Towards a scalable public blockchain in fog computing of IoT services computing, IEEE Trans. Serv. Comput. 13 (2) (2020) 252–262.

[71] Arthur Gervais, Ghassan O Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, Srdjan Capkun, On the security and performance of proof of work blockchains, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016, pp. 3–16.

[72] Zhuofan Liao, Siwei Cheng, RVC: A reputation and voting based blockchain consensus mechanism for edge computing-enabled IoT systems, J. Netw. Comput. Appl. 209 (2023) 103510.

[73] Ashish Mohan Yadav, S.C. Sharma, Cooperative task scheduling secured with blockchain in sustainable mobile edge computing, Sustain. Comput.: Inform. Syst. (2023) 100843.

[74] Zhanyang Xu, Qingfan Geng, Hao Cao, Chuanjian Wang, Xihua Liu, Uncertainty-aware workflow migration among edge nodes based on blockchain, EURASIP J. Wireless Commun. Networking 2019 (2019) 1–12.

[75] Xiaolong Xu, Yi Chen, Yuan Yuan, Tao Huang, Xuyun Zhang, Lianyong Qi, Blockchain-based cloudlet management for multimedia workflow in mobile cloud computing, Multimedia Tools Appl. 79 (2020) 9819–9844.

[76] Kaile Xiao, Weisong Shi, Zhipeng Gao, Congcong Yao, Xuesong Qiu, DAER: A resource preallocation algorithm of edge computing server by using blockchain in intelligent driving, IEEE Internet Things J. 7 (10) (2020) 9291–9302.

[77] Ke Zhang, Yongxu Zhu, Sabita Maharjan, Yan Zhang, Edge intelligence and blockchain empowered 5G beyond for the industrial internet of things, IEEE Netw. 33 (5) (2019) 12–19.

[78] Ying He, Yuhang Wang, Chao Qiu, Qiuzhen Lin, Jianqiang Li, Zhong Ming, Blockchain-based edge computing resource allocation in IoT: A deep reinforcement learning approach, IEEE Internet Things J. 8 (4) (2020) 2226–2237.

[79] Zhenni Li, Zuyuan Yang, Shengli Xie, Wuhui Chen, Kang Liu, Credit-based payments for fast computing resource trading in edge-assisted Internet of Things, IEEE Internet Things J. 6 (4) (2019) 6606–6617.
[80] Shreshth Tuli, Redowan Mahmud, Shikhar Tuli, Rajkumar Buyya, Fogbus: A blockchain-based lightweight framework for edge and fog computing, J. Syst. Softw. 154 (2019) 22–36.
[81] Xiaolong Xu, Xuyun Zhang, Honghao Gao, Yuan Xue, Lianyong Qi, Wanchun Dou, BeCome: Blockchain-enabled computation offloading for IoT in mobile edge computing, IEEE Trans. Ind. Inform. 16 (6) (2019) 4187–4195.
[82] Shuming Seng, Changqing Luo, Xi Li, Heli Zhang, Hong Ji, User matching on blockchain for computation offloading in ultra-dense wireless networks, IEEE Trans. Netw. Sci. Eng. 8 (2) (2020) 1167–1177.
[83] David Gale, Lloyd S. Shapley, College admissions and the stability of marriage, Amer. Math. Monthly 120 (5) (2013) 386–391.
[84] Di Wu, Nirwan Ansari, A cooperative computing strategy for blockchain-secured fog computing, IEEE Internet Things J. 7 (7) (2020) 6603–6609.
[85] Ao Zhou, Qibo Sun, Jinglin Li, BCEdge: Blockchain-based resource management in D2D-assisted mobile edge computing, Softw. - Pract. Exp. 51 (10) (2021) 2085–2102.
[86] Guanhua Qiao, Supeng Leng, Haoye Chai, Arash Asadi, Yan Zhang, Blockchain empowered resource trading in mobile edge computing and networks, in: ICC 2019-2019 IEEE International Conference on Communications, ICC, IEEE, 2019, pp. 1–6.
[87] Dinh C Nguyen, Pubudu N Pathirana, Ming Ding, Aruna Seneviratne, Secure computation offloading in blockchain based IoT networks with deep reinforcement learning, IEEE Trans. Netw. Sci. Eng. 8 (4) (2021) 3192–3208.
[88] Matteo Nardini, Sven Helmer, Nabil El Ioini, Claus Pahl, A blockchain-based decentralized electronic marketplace for computing resources, SN Comput. Sci. 1 (2020) 1–24.
[89] Hamza Baniata, Ahmad Anaqreh, Attila Kertesz, PF-BTS: A privacy-aware fog-enhanced blockchain-assisted task scheduling, Inf. Process. Manage. 58 (1) (2021) 102393.
[90] Xiao Zheng, Mingchu Li, Yuanfang Chen, Jun Guo, Muhammad Alam, Weitong Hu, Blockchain-based secure computation offloading in vehicular networks, IEEE Trans. Intell. Transp. Syst. 22 (7) (2020) 4073–4087.
[91] Wilton Jaciel Loch, Guilherme Piêgas Koslovski, Maurício Aronne Pillon, Charles Christian Miers, Marcelo Pasin, A novel blockchain protocol for selecting microservices providers and auditing contracts, J. Syst. Softw. 180 (2021) 111030.
[92] Su Yao, Mu Wang, Qiang Qu, Ziyi Zhang, Yi-Feng Zhang, Ke Xu, Mingwei Xu, Blockchain-empowered collaborative task offloading for cloud-edge-device computing, IEEE J. Sel. Areas Commun. 40 (12) (2022) 3485–3500.
[93] Aleksandr Zavodovski, Suzan Bayhan, Nitinder Mohan, Pengyuan Zhou, Walter Wong, Jussi Kangasharju, DeCloud: Truthful decentralized double auction for edge clouds, in: 2019 IEEE 39th International Conference on Distributed Computing Systems, ICDCS, IEEE, 2019, pp. 2157–2167.
[94] Zixuan Xie, Run Wu, Miao Hu, Haibo Tian, Blockchain-enabled computing resource trading: A deep reinforcement learning approach, in: 2020 IEEE Wireless Communications and Networking Conference, WCNC, IEEE, 2020, pp. 1–8.
[95] Huaming Wu, Katinka Wolter, Pengfei Jiao, Yingjun Deng, Yubin Zhao, Minxian Xu, EEDTO: an energy-efficient dynamic task offloading algorithm for blockchain-enabled IoT-edge-cloud orchestrated computing, IEEE Internet Things J. 8 (4) (2020) 2163–2176.
[96] Seyed Mojtaba Hosseini Bamakan, Amirhossein Motavali, Alireza Babaei Bondarti, A survey of blockchain consensus algorithms performance evaluation criteria, Expert Syst. Appl. 154 (2020) 113385.
[97] Shijie Zhang, Jong-Hyouk Lee, Analysis of the main consensus protocols of blockchain, ICT Express 6 (2) (2020) 93–97.
[98] Christian Decker, Jochen Seidel, Roger Wattenhofer, Bitcoin meets strong consistency, in: Proceedings of the 17th International Conference on Distributed Computing and Networking, 2016, pp. 1–10.
[99] Stefano De Angelis, Leonardo Aniello, Roberto Baldoni, Federico Lombardi, Andrea Margheri, Vladimiro Sassone, PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain, in: CEUR Workshop Proceedings, Vol. 2058, 2018, pp. 1–11.
[100] Marcela T de Oliveira, Lúcio HA Reis, Dianne SV Medeiros, Ricardo C Carrano, Sílvia D Olabarriaga, Diogo MF Mattos, Blockchain reputation-based consensus: A scalable and resilient mechanism for distributed mistrusting applications, Comput. Netw. 179 (2020) 107367.
[101] Fangyu Gai, Baosheng Wang, Wenping Deng, Wei Peng, Proof of reputation: A reputation-based consensus protocol for peer-to-peer network, in: Database Systems for Advanced Applications: 23rd International Conference, DASFAA 2018, Gold Coast, QLD, Australia, May 21-24, 2018, Proceedings, Part II 23, Springer, 2018, pp. 666–681.
[102] Ittay Eyal, Blockchain technology: Transforming libertarian cryptocurrency dreams to finance and banking realities, Computer 50 (9) (2017) 38–49.
[103] Yang Xiao, Ning Zhang, Wenjing Lou, Y. Thomas Hou, A survey of distributed consensus protocols for blockchain networks, IEEE Commun. Surv. Tutor. 22 (2) (2020) 1432–1465.
[104] Marko Vukolić, The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication, in: Open Problems in Network Security: IFIP WG 11.4 International Workshop, INetSec 2015, Zurich, Switzerland, October 29, 2015, Revised Selected Papers, Springer, 2016, pp. 112–125.
[105] Ashok Kumar Yadav, Karan Singh, Ali H Amin, Laila Almutairi, Theyab R Alsenani, Ali Ahmadian, A comparative study on consensus mechanism with security threats and future scopes: Blockchain, Comput. Commun. 201 (2023) 102–115.
[106] Rajni Aron, Ajith Abraham, Resource scheduling methods for cloud computing environment: The role of meta-heuristics and artificial intelligence, Eng. Appl. Artif. Intell. 116 (2022) 105345.
[107] Dinesh Kumar, Gaurav Baranwal, Yamini Shankar, Deo Prakash Vidyarthi, A survey on nature-inspired techniques for computation offloading and service placement in emerging edge technologies, World Wide Web 25 (5) (2022) 2049–2107.
[108] Eric Brewer, CAP twelve years later: How the" rules" have changed, Computer 45 (2) (2012) 23–29.
[109] Gaurav Baranwal, Dinesh Kumar, PoSP: A novel proof of service placement in blockchain-based edge computing, in: 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops), IEEE, 2022, pp. 18–21.
[110] Rong Han, Zheng Yan, Xueqin Liang, Laurence T. Yang, How can incentive mechanisms and blockchain benefit with each other? a survey, ACM Comput. Surv. 55 (7) (2022) 1–38.
[111] Xiaodong Lin, Jianbing Ni, Xuemin (Sherman) Shen, Privacy-Enhancing Fog Computing and its Applications, Springer, 2018.
[112] Eranga Bandara, Xueping Liang, Peter Foytik, Sachin Shetty, Nalin Ranasinghe, Kasun De Zoysa, Rahasak—Scalable blockchain architecture for enterprise applications, J. Syst. Archit. 116 (2021) 102061.
[113] Mohammad Nikravan, Mostafa Haghi Kashani, A review on trust management in fog/edge computing: Techniques, trends, and challenges, J. Netw. Comput. Appl. (2022) 103402.
[114] Matin Chiregi, Nima Jafari Navimipour, Cloud computing and trust evaluation: A systematic literature review of the state-of-the-art mechanisms, J. Electr. Syst. Inf. Technol. 5 (3) (2018) 608–622.
[115] Tome Boshevski, Josh Brandoff, Vitalik Buterin, Storj a Peer-to-Peer Cloud Storage Network, Princeton, NJ, USA, 2014.
[116] Juan Benet, Ipfs-content addressed, versioned, p2p file system, 2014, arXiv preprint arXiv:1407.3561.
[117] Xinxin Fan, Ling Liu, Rui Zhang, Quanliang Jing, Jingping Bi, Decentralized trust management: Risk analysis and trust aggregation, ACM Comput. Surv. 53 (1) (2020) 1–33.
[118] Omar Hasan, Lionel Brunie, Elisa Bertino, Privacy-preserving reputation systems based on blockchain and other cryptographic building blocks: A survey, ACM Comput. Surv. 55 (2) (2022) 1–37.
[119] Zhili Zhou, Meimin Wang, Ching-Nung Yang, Zhangjie Fu, Xingming Sun, QM Jonathan Wu, Blockchain-based decentralized reputation system in E-commerce environment, Future Gener. Comput. Syst. 124 (2021) 155–167.
[120] Andreas Kapsalis, Panagiotis Kasnesis, Iakovos S Venieris, Dimitra I Kaklamani, Charalampos Z Patrikakis, A cooperative fog approach for effective workload balancing, IEEE Cloud Comput. 4 (2) (2017) 36–45.
[121] Jinlai Xu, Balaji Palanisamy, Heiko Ludwig, Qingyang Wang, Zenith: Utility-aware resource allocation for edge computing, in: 2017 IEEE International Conference on Edge Computing, EDGE, IEEE, 2017, pp. 47–54.
[122] Marshall Ball, Alon Rosen, Manuel Sabin, Prashant Nalini Vasudevan, Proofs of useful work, Cryptol. ePrint Arch. (2017).
[123] Eric Ke Wang, Zuodong Liang, Chien-Ming Chen, Saru Kumari, Muhammad Khurram Khan, PoRX: A reputation incentive scheme for blockchain consensus of IIoT, Future Gener. Comput. Syst. 102 (2020) 140–151.
[124] Rajkumar Buyya, Rajiv Ranjan, Rodrigo N. Calheiros, Modeling and simulation of scalable cloud computing environments and the CloudSim toolkit: Challenges and opportunities, in: 2009 International Conference on High Performance Computing & Simulation, IEEE, 2009, pp. 1–11.
[125] Devki Nandan Jha, Khaled Alwasel, Areeb Alshoshan, Xianghua Huang, Ranesh Kumar Naha, Sudheer Kumar Battula, Saurabh Garg, Deepak Puthal, Philip James, Albert Zomaya, et al., IoTSim-edge: a simulation framework for modeling the behavior of Internet of Things and edge computing environments, Softw. - Pract. Exp. 50 (6) (2020) 844–867.

[126] Tien-Dung Nguyen, Eui-Nam Huh, ECSim++: An INET-based simulation tool for modeling and control in edge cloud computing, in: 2018 IEEE International Conference on Edge Computing, EDGE, IEEE, 2018, pp. 80–86.
[127] Redowan Mahmud, Rajkumar Buyya, Modelling and simulation of fog and edge computing environments using iFogSim toolkit, Fog Edge Computing: Princ. Paradig. (2019) 1–35.
[128] Isaac Lera, Carlos Guerrero, Carlos Juiz, YAFS: A simulator for IoT scenarios in fog computing, IEEE Access 7 (2019) 91745–91758.
[129] Carlos Faria, Miguel Correia, BlockSim: blockchain simulator, in: 2019 IEEE International Conference on Blockchain (Blockchain), IEEE, 2019, pp. 439–446.
[130] Yusuke Aoki, Kai Otsuki, Takeshi Kaneko, Ryohei Banno, Kazuyuki Shudo, Simblock: A blockchain network simulator, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2019, pp. 325–329.
[131] Lyubomir Stoykov, Kaiwen Zhang, Hans-Arno Jacobsen, Vibes: fast blockchain simulations for large-scale peer-to-peer networks, in: Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference: Posters and Demos, 2017, pp. 19–20.

Gaurav Baranwal is faculty in the Department of Computer Science, Banaras Hindu University, Varanasi, UP, India. He did his M.Tech and PhD in Computer Science from Jawaharlal Nehru University, India. He has published research papers in various peer-reviewed international journals (including IEEE, Elsevier, Springer, Wiley etc.) and in proceedings of various peer-reviewed conferences. His research interests include resource provisioning and service coordination in cloud computing and edge Computing and Blockchain.

Dinesh Kumar is faculty in the Department of Computer Science and Engineering, NIT Jamshedpur, Jamshedpur, Jharkhand, India. He did his M.Tech and PhD in Computer Science from Jawaharlal Nehru University, India. He has published research papers in various peer-reviewed international journals (including Elsevier and Springer) and proceedings of various peer-reviewed conferences. His research interests include resource provisioning in cloud computing, Internet of Things and Edge computing.

Deo Prakash Vidyarthi is Professor in the School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India. Dr. Vidyarthi has published around 105 research papers in various peer reviewed International Journals and Transactions (IEEE, Elsevier, Springer, Wiley, World Scientific etc.) and around 50 research papers in proceedings of various peer-reviewed conferences in India and abroad. Dr. Vidyarthi has three books (research monograph) to his credit. He also has contributed chapters in many edited books. He is in the editorial board and in the reviewer's panel of many International Journals.He is the member of IEEE, ACM, International Association of Computer Science and Information Technology (IACSIT), Singapore, International Society of Research in Science and Technology (ISRST), USA and International Association of Engineers (IAENG). Research interest includes Parallel and Distributed System, Grid and Cloud Computing, Mobile Computing and Evolutionary Computing.