Automatika
Journal for Control, Measurement, Electronics, Computing and Communications

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/taut20

A novel blockchain enabled resource allocation and task offloading strategy in cloud computing environment

G. Senthilkumar, K. N. Madhusudhan, Y. Jeyasheela & P. Ajitha

To cite this article: G. Senthilkumar, K. N. Madhusudhan, Y. Jeyasheela & P. Ajitha (2024) A novel blockchain enabled resource allocation and task offloading strategy in cloud computing environment, Automatika, 65:3, 973-982, DOI: 10.1080/00051144.2024.2314906

To link to this article: https://doi.org/10.1080/00051144.2024.2314906

Published online: 12 Mar 2024.

AUTOMATIKA
2024, VOL. 65, NO. 3, 973–982
https://doi.org/10.1080/00051144.2024.2314906

A novel blockchain enabled resource allocation and task offloading strategy in cloud computing environment

G. Senthilkumar[a], K. N. Madhusudhan[b], Y. Jeyasheela[c] and P. Ajitha[d]

[a]Dept. of Computer Science and Engineering, Panimalar Engineering College, Chennai, India; [b]Department of Electronics and Communication Engineering, B.M.S. College of Engineering, Bangalore, India; [c]Dept. of IT, NICHE, Thuckalay, India; [d]Department of IT, School of Computing, Sathyabama Institute of Science and Technology, Chennai, India

ABSTRACT
Large amounts of processing resources are required for the sensed raw big data processing during the data generation process. Furthermore, as sensed data are typically privacy sensitive, blockchain technology can be used to ensure the privacy concerns. This study examines a multiuser mobile offloading network that consists of a cloud server located remotely and an edge node. We formulate the offloading problem as the joint optimization of task offloading decision making of all users, the computation resource allocation among the edge executing applications, and the radio resource assignment among all the remote-processing applications. The goal is to minimize the maximum weighted cost of all users. When compared to other benchmark approaches, the simulation results show that the proposed algorithm achieves optimal results in terms of both energy consumption and delay as a result of collaboration. Finally the resource allocation and optimal offloading strategy with 93% efficiency is obtained.

ARTICLE HISTORY
Received 2 January 2024
Accepted 31 January 2024

KEYWORDS
Block-chain; IoT; resource allocation; cloud computing and joint task offloading

1. Introduction

An increasing number of people are using mobile devices to store data on the internet due to the rapid advancement of information technology. This alters how we live and work by realizing the seamless connection between the real world and internet. Furthermore, because of the data explosion, the conventional architecture is not able to handle large amounts of data. This critical issue can be temporarily resolved because to cloud computing's vast array of processing and storage resources. However, if mobile devices need to upload data to the distant cloud, then there will be a considerable data delay for the user.

In certain unique applications the timeout scenario poses a risk to an individual's life. Luckily, it has been discovered that edge data processing connected to mobile devices is a great method to address the aforementioned problems. The data delay can be significantly decreased when edge devices communicate directly with the closest edge node when they request data. While edge computing architecture is a potential solution, there are other obstacles to overcome. A potential technology called "device-free sensing" [1] can anticipate a person's presence, activity, motion, position, and other details without the need for wearable or Internet of Things devices.

With the use of the widely available wireless signals, DFS is able to detect nearly anyone, anywhere, and use the reflection, scattering, diffraction, shadowing, and other effects that people have on wireless transmission connections to estimate their condition [2]. In order to accomplish this, DFS can offer a wealth of human data that we can use for tasks like intelligent interaction, intelligent monitoring, emergency rescue, security monitoring, crowd allocation calculation, and analysis of human living patterns. Even though DFS has recently advanced significantly, there are still a lot of problems that need to be solved.

Even though DFS has advanced significantly, there are still a lot of issues that need to be resolved. First, the extraction of effect information from the raw big data requires a substantial investment of energy and computing power. Furthermore, the data processing typically requires very small processing delays in many circumstances [3], which increases the demands on processing capacity. Furthermore, data about human states is always sensitive to privacy, making it extremely difficult to protect individuals' private during the DFS process [4].

Smart user equipment has gained popularity as practical aids in our daily life in recent years. These gadgets are now widely used by consumers to process sensor data in a variety of circumstances. This implies that users are now even more adaptable and helpful since they can now gather and handle data on their own rather than depending exclusively on other sensors and

CONTACT G. Senthilkumar gsenthilkumar.cse23@gmail.com Dept. of Computer Science and Engineering, Panimalar Engineering College, Bangalore Trunk Road, Varadharajapuram, Poonamallee, Chennai, Tamil Nadu 600123

devices. Because of this, the potential uses for smart user equipment are always growing, and in the future, we should expect to see even more creative applications for these gadgets.

However, users typically have constrained resources, including low computational capacity and battery life, which makes it extremely difficult to run sophisticated applications on them [5]. By shifting work to edge nodes, mobile edge computing also referred to as fog computing [6] has been proposed as a useful addition to cloud computing. Cloud computing serves as a significant tool for easing the burden. Blockchain [7] is a potential distributed technology that can successfully address the issues of security and privacy for the data collected by sensors.

Nevertheless, blockchain activities typically need a lot of energy and time, and MEC and cloud computing can also resolve computation-related issues [8]. The potential advantages of combining blockchain technology, cloud computing, and mobile edge computing in the field of device free sensing are what inspired this research. In addition to increasing the efficacy and efficiency of DFS data processing, the integration of these technologies may also improve the security and privacy of DFS data. Through the use of blockchain technology in DFS, we can guarantee the private and secure transmission and storage of data.

Furthermore, MEC and cloud computing can offer the computational power required for efficient processing of sensing data and blockchain job execution. On the network's edge, MEC technology enables distributed processing power, while cloud computing offers scalable resources for more difficult jobs. Therefore, we can address DFS's security and privacy issues by combining blockchain technology with DFS. The research for this work is motivated by the need to provide sufficient processing capability for effective sensing data processing and blockchain job execution, which we can achieve by utilizing MEC and cloud computing.

Local producers have responded by introducing a range of server models. For example, Huawei has produced general entry level and general computing servers. While Alibaba has established servers based on computing, memory and general on the CPU to memory ratio. Using blockchain technology server execution data that is distributed to many edge facility providers can be stored [7,8]. However, the following two considerations are crucial to keep in mind. To begin with, the information stored on the blockchain is open and transparent. The information recorded at each moment should be minimal, according to the storage mechanism. When recording, sensitive material should be kept to a minimum and attention should be paid to metrics that assess how well a task is being executed.

Second, it's important to be aware of forking attacks because every edge facility provider in a given location has set up numerous servers to process user requests [6–10]. If a large number of edge servers involved in edge collaboration are controlled by the same edge provider. This can turn into a vicious cycle and lower the reputation value of edge servers. It is possible to alter information illegally by overwriting, deleting useful data from other servers and ignoring historical in conventional data. When evaluating whether distributed ledger technology can be used in edge collaboration, the forking attack is an important issue. Yet it has been mostly ignored by researchers.

The proposed strategy has various advantages over some sophisticated algorithms and classical methods in terms of load balancing, energy usage, and EDR delay. In order to further enhance edge node load balancing, we integrate blockchain technology to optimize the trust and incentive mechanism of edge collaboration [11,12]. Furthermore, we propose complementary approaches from the standpoints of software as well as hardware to address the potential forking attack problem that could emerge during the blockchain's implementation in edge collaboration.

2. Literature survey

In edge cloud environment based on mobile crowdsourcing the research of task assignment has received increasing interest in the past several years [13]. The goal is to create the best task offloading technique possible with minimal latency, minimal energy consumption, and maximum service quality. Numerous academics have studied this in-depth and proposed feasibility studies.

A new paradigm in computing which is edge computing has been emerged because of the growing popularity of IoT and the advancements made in cloud services [14]. Edge computing is the term for the processing of data at the edge of the network [15]. This mode protects the confidentiality and privacy of data while lowering request latency and network bandwidth [16–18]. Moving some or all of the cloud computing tasks close to mobile devices is the fundamental component of edge computing. This extremely promising method can address a few of the drawbacks associated with cloud computing [19].

In order to address the issue of excessive bandwidth consumption in conventional cloud platform Zhao et al. [11] developed a novel mobile device data transmission system that made use of edge nodes to facilitate data transmission and incorporated edge computing into the cloud platform centric architecture [20]. This technique uses the edge computing concept to find out how much bandwidth edge nodes consume.

Ren et al. researched the issue of cooperative computing resource allocation and communication technologies with the goal of determining the best way to reduce latency in cloud and edge cloud collaboration systems. And designed a distributed computing based offloading strategy that may provide excellent computing offloading capabilities and adapt accordingly to changes in user scale [21] and solved the issue of multi-user edge cloud resource offloading in an environment where multiple wireless channels interfer.

One area of active research in edge computing is the issue of offloading computing tasks [22]. Task offloading in the real crowdsourcing environment will be influenced by a number of external factors, including the device's hardware capability, the network environment in which the worker resides, and the worker's customized preference [16,23]. Because of this, it is especially crucial to develop a sensible task offloading approach that adapts to the external environment over time. The core investigation of various previous studies is how to decide which tasks to assign in an offline or online environment. The majority of research concentrates on reducing resource usage and task completion time as the optimization goal.

Developing and simulating a neural network for the aim of analysing and learning about the human brain is the primary goal of deep learning which is a new technology based on machine learning algorithms [14]. Using neural networks to further abstract low level features into high level features through hierarchical feature representation of the data is the basic method of deep learning. With DNNs composed of multilayer senses, significant progress has been made in the areas of natural language processing, robot control, and image classification and identification.

Mnih et al. [10] has proposed a deep Q-network [1], a unique surrogate model that uses deep neural networks to bridge the gap between high-level sensory input and decision-making behaviours. In the area of wireless communication, deep learning is also frequently applied to problems with data caching [19], signal detection [18], resource allocation [24], and other issues. Using deep learning models some researchers have recently addressed the job offloading issue in edge cloud environments.

The author [24] has another paper that proposes a task scheduler made specifically for IoT devices that run without batteries. The energy aware technique used by this scheduler takes into account the amount of energy that has been gathered and made available, the energy consumption of each task and the priority given to it in order to decide the best time to complete it. Prioritizing the tasks that have the highest priority for execution prior to each iteration allows the scheduler to keep things moving along smoothly.

A novel deep reinforcement learning methodology has been proposed in another paper [25]. This paper proposes to reduce the average cost of long term services by taking in mind the delay caused by power consumption and buffering. For discrete action domains, this methodology use the double deep Q networks duelling strategy for continuous action domains, it uses the deep deterministic policy gradient technique.

The author [11] also proposes another strategy that could be used for communication offloading, which involves moving communication-intensive tasks to edge or cloud servers. This technology raises important concerns about security and privacy, even if it may lessen the communication load associated with IoT devices.

The author [20] proposes a solution for hybrid method to computation and communication offloading. With this method, jobs requiring a lot of processing or communication power are moved to servers in the cloud or at the edge. This methodology can lead to a balanced equilibrium among communication overhead and energy use. But it's also critical to recognize that it can lead to more complexity in the process of choosing which tasks to offload.

Current optimization approaches fail to consider the computational dimension limitations and typical optimization algorithms are unable to effectively handle data complexity. Prior research did not take resource allocation into account when optimizing the job offloading scheme. This paper explores the relationship between resource consumption and task completion rate as well as service quality in accordance with the aforementioned challenges. To solve these kind of issues on optimization an effective and energy saving cloud computing based algorithm for resource allocation and offloading decision using blockchain is proposed in our methodology.

2.1. Contributions of the proposed system

The paper presents an efficient and energy saving technique with energy harvesting for Internet of Things scenarios. The following is a summary of the principal contributions.

(i) In order to lower the total energy usage of every terminal device based on the joint consideration of resource allocation and offloading choice an optimization problem is developed.
(ii) An efficient and energy saving system for allocating resources and making offloading decisions is proposed based on cloud computing. By integrating the regularization paradigm into the network parameter updating process, this approach builds a deep neural network architecture with regularization. This leads to a rapid convergence of the training process.
(iii) The proposed approach has been thoroughly analysed and simulated and it significantly outperforms local computing and full offloading techniques in terms of energy consumption reduction.

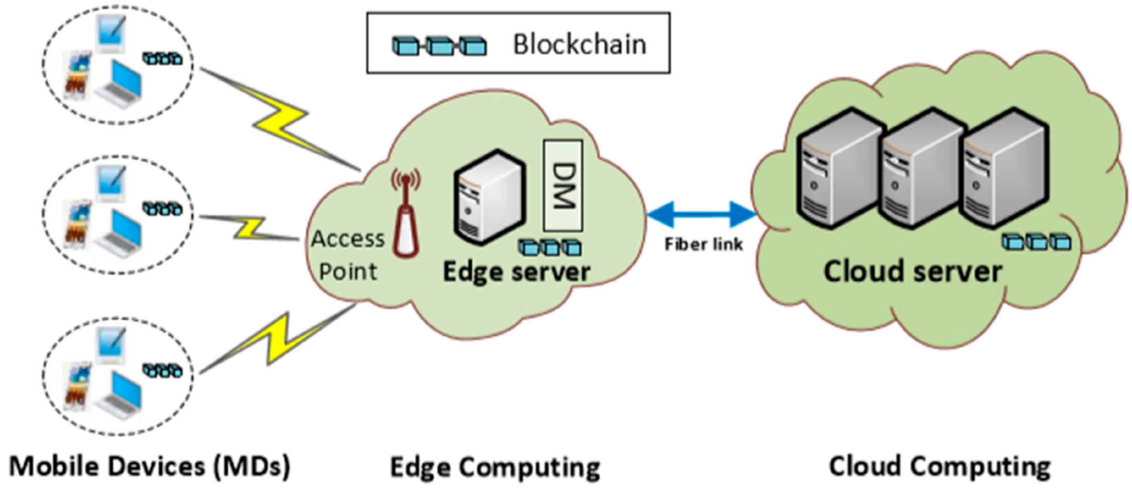

Figure 1. The proposed system model.

2.2. Organization of the paper

The remainder of the work is structured in this manner. The system model is explained in section 3. This section also provides an explanation of the optimization challenge. Section 3 provides a detailed explanation of the proposed algorithm. The numerical results are analysed in section 4 in order to validate and evaluate the proposed strategy. Everything is summarized in section 5.

3. Methodology

As seen in Figure 1, we propose an integrated edge cloud architecture for blockchain enabled IoT networks that consists of three primary layers the cloud computing layer, the edge computing layer, and the layer for mobile devices.

3.1. The proposed system model

In order to optimize the energy consumption of the system resulting from task computing and offloading, the research proposes an edge computation offloading strategy with energy harvesting for the internet of things [26–29]. The proposed network architecture is displayed in Figure 1. The user layer and the edge cloud layer are the two layers that make up the model. N terminal devices is the intended configuration for the user layer. Assume that i represents device i, with a range of $i \in \{1, 2, \ldots\ldots, N\}$ and that the devices are unrelated to one another. A task with size c_i is generated by all the user terminal i for the time period T.

Whether or not to offload the task will be determined based on the unpredictable task size and the limitations of the local computing resources of the local device. The decision for offloading of device i is represented as $x_i \in \{0, 1\}$. When $x_i = 0$, the work is executed in local computing mode. If not, the edge server receives the task across the wireless network. The device will use a lot of energy when it is performing local or offloading computation. The battery's inadequate capacity prevents it from meeting the device's needs.

This paper proposes the use of rechargeable batteries for each device at the user layer in order to improve user experience and QoS by reducing terminal energy consumption. Wireless charging takes place primarily from base stations for renewable energy that are located all around the device. Because the edge server has so many resources, an edge node with a server in the edge cloud is utilized to supply computing services for terminal devices inside its communication coverage. Devices at the user layer are subsequently given access to the computing results. In the parts that follow, the definition model of task offloading, energy harvesting, and local computing will be thoroughly discussed.

3.1.1. Mobile device layer

A network of MDs, including smartphones, tablets, and sensor devices, make up this layer. The blockchain connects these MDs to one another in the IoT network. Every MD has a blockchain account that they may use to connect to the network and access features like data collection and task offloading to cloud servers.

3.1.2. Layer of edge computing

This layer is made up of a wireless access point for wireless communication with neighbouring devices, a decision maker for task execution decisions, and a lightweight edge server for real data processing. This layer can provide low-latency computation services at the edge of the network. Complicated calculation tasks must be sent from the edge server via a wired line to the resourceful cloud server in order to avoid work overload on the edge layer. The edge server functions as a blockchain entity to establish trustworthy connections with cloud nodes and MDs on the blockchain network, further ensuring security. All transactions and offloading activities within the offloading system will be broadcast to the edge server and recorded by blockchain in order to come to an agreement on offloading management.

3.1.3. Cloud computing layer

This layer allows complex computing tasks to be solved from local IoT devices using several virtual machines with powerful computation and storage capabilities.

3.2. Block-chain based data storage system

Figure 2 demonstrates the network model of the proposed methodology. SM_i which is a smart metre is connected to multiple consumers based on the network model. To multiple smart metres a service provider SP_j is connected. By using a group of service providers P2P network service providers are established. All the smart metres SM_i which are connected must be registered in offline mode with a reliable registration authority. Similarly the service providers SP_j must be registered to registration authority. The registration authority completes the registration process in a safe and secure manner. By using an access control mechanism the smart metres SM_i and service providers SP_j communicates securely. This is established between the service provider and smart metre to exchange information.

3.3. The proposed resource allocation and offloading decision making mechanism based on cloud computing

A mixed-integer programming problem was used to formulate the optimization problem in the preceding section. There are several limitations with traditional methods for addressing NP-hard issues. In order to solve this we have proposed a deep learning based algorithm for optimization problem P1. Using deep neural networks this technique optimizes offloading decisions while taking resource allocation into consideration. To optimize resource allocation, it is also used with momentum gradient descent. It also drastically lowers the energy usage of the system.

Considering the best offloading decision x_i^*, which is

$$P2: \begin{matrix} min \\ \alpha_i f_i^{loc} \end{matrix} \begin{matrix} \sum e_i(x_i^*) \\ i \in \{1, 2, \ldots N\} \end{matrix} \tag{1}$$

$$0 < f_i^{loc} \leq f_i^{max} \tag{2}$$

$$0 \leq \frac{c_i}{r_i^{up}} \leq (1 - \alpha_i)T \tag{3}$$

$$0 \leq e_i^{loc} \leq e_i^{har} + b_i \tag{4}$$

$$0 \leq e_i^{off} \leq e_i^{har} + b_i \tag{5}$$

$$0 \leq \alpha_i \leq 1 \tag{6}$$

As a result, the optimization problem P1 can be divided into the resource allocation P2 problems and offloading decision sub-problems.

3.3.1. Generation of offloading decision

We have proposed an energy saving and effective deep learning based optimization technique to address the optimization challenge. As seen in Figure 3, this research builds a DNN to generate binary offloading decisions in order to make a faster convergence and to stabilize it. First, we use the greedy algorithm to get the labelled data, which we then input to the first DNN to train it. As a result the optimal offloading decision is made by utilizing a well-trained DNN. To attain the optimal fitting result, regularization is applied to the output of every DNN layer. When the stochastic gradient descent approach minimizes the loss function, we use it. By using a single sample to update the gradient function, this solution technique quickens convergence. The stochastic gradient descent approach is more flexible since it may instantly modify the model in response to new information.

At first the N number of tasks with size c_i are given as input in random order where $i \in \{1, 2, \ldots N\}$. 2^N actions can be generated according to the greedy strategy. The number of possible actions increases exponentially as the number of tasks increases. Because randomly generated tasks are complicated and varied, it is imperative to avoid the decrease of system performance caused by data accumulation. The goal of our proposed methodology is to provide an optimal offloading strategy function Π based on the concept of minimizing overall energy consumption in order to acquire the optimal offloading action $x_i^* \in \{0, 1\}$. The function of the offloading approach is defined as

$$\Pi : c_i \mapsto x_i^* \tag{7}$$

Given that mapping has a one-to-many feature and a high processing time complexity, this research uses a parameterized function $F_{w,b}$ based on DNN to approximate Π. We select the action based on the idea of minimizing the overall energy consumption of user devices an optimal offloading decision can be found. This is shown in Equation (34)

$$x^* = \arg min \sum_{i \in \{1, 2, \ldots\ldots N\}} e_i c_i x_i^* \tag{8}$$

The DNN proposed in this work has three hidden layers, one output layer, one input layer and is completely connected throughout. The ELU function serves as the hidden layer's activation function because its linear component can mitigate the gradient's disappearance and its nonlinear component can increase the ELU's resistance to input changes. Simultaneously, the ELU's output mean value is almost zero, indicating a quicker rate of convergence. The loss function of all m samples can be written as follows. Assuming that the sample $q \in \{1, 2, \ldots\ldots N\}$ has the mean squared error

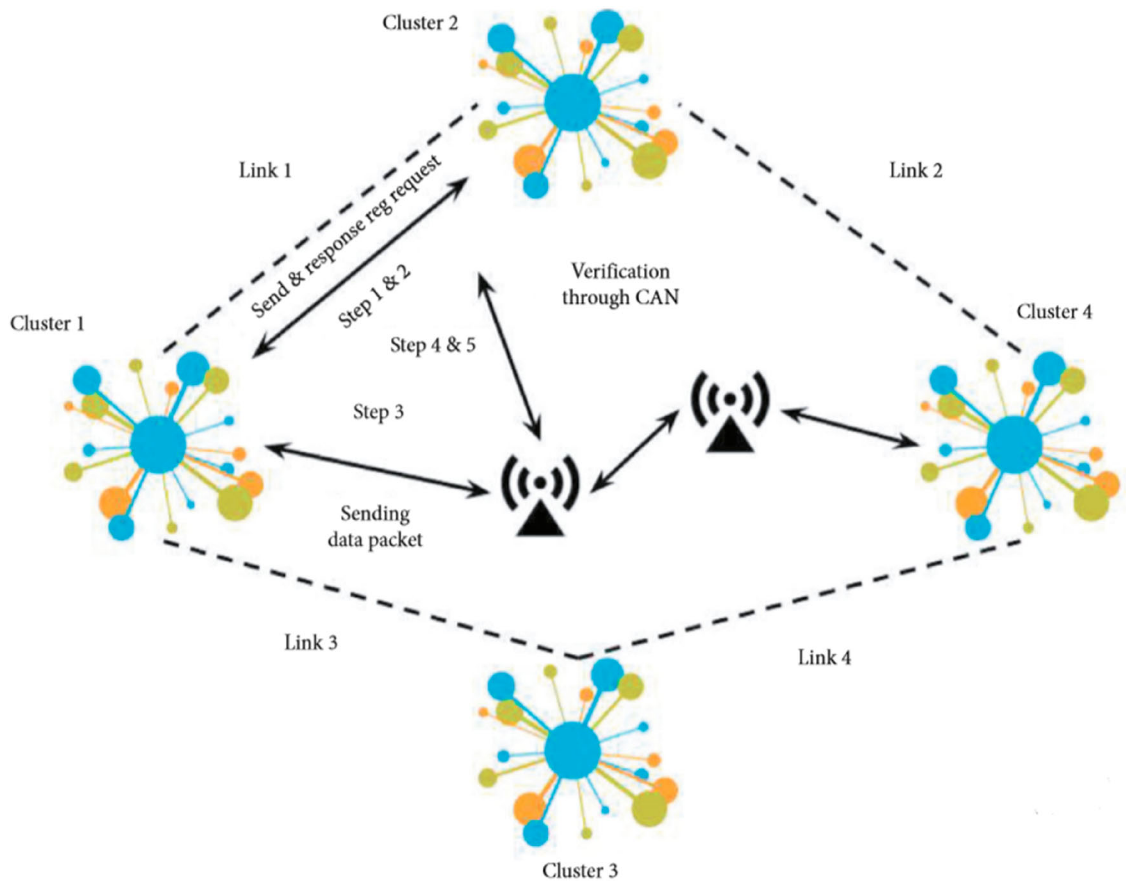

Figure 2. The block-chain enabled IoT based smart sensor network architecture.

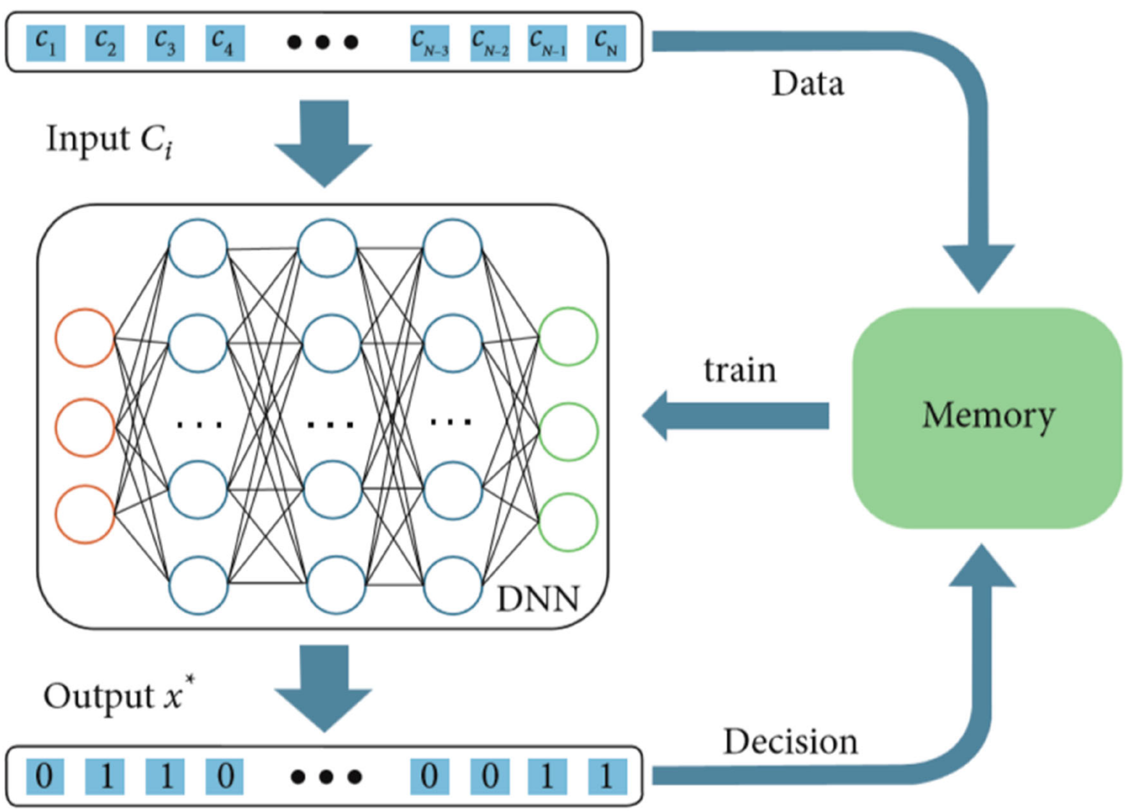

Figure 3. The process of generating offloading decisions.

loss function.

$$Loss(\omega, b) = \frac{1}{m}\sum_{q=1}^{m} y_p - F_{\omega,b}(C_q))^2 \tag{9}$$

$$Loss(\omega, b) = \frac{1}{m}\sum_{q=1}^{m} y_p - F_{\omega,b}(C_q))^2 \frac{\lambda}{m}\sum_{l=2}^{L}\omega^2 \tag{10}$$

Updated definition is given in Equation (37).

$$\omega \leftarrow \omega - \beta\nabla_{\omega}Loss(\omega, b) \tag{11}$$

$$b \leftarrow b - \beta\nabla_{b}Loss(\omega, b) \tag{12}$$

The neural network's weight coefficients and bias values attain their ideal values after training. By inputting c_i into the parameterized function $F_{\omega,b}$, the optimal offloading decision for device i will be found.

$$x^* = F_{\omega,b}(C_i) \tag{13}$$

3.3.2. Optimal resource allocation

After the optimal offloading decision has been determined, problem P1 can be changed into problem P2. It is evident from Equation (9), that standard mathematical optimization techniques can be used to solve issue P2. In order to solve problem P2, we thus introduce the momentum gradient descent. The momentum gradient descent algorithm speeds up convergence when compared to the traditional gradient descent approach.

$$e_i = (1 - x_i^*)\delta_i f_i^{loc2} C_i R + \frac{x_i^* C_i P_i}{(1-\alpha_i) r_i^{up}} - \gamma p h_i \alpha_i T \tag{14}$$

Gradient descent is used to calculate the gradient functions of α_i and f_i^{loc}, which are specified as

$$\frac{\partial e_i}{\partial \alpha_i} = \frac{x_i C_i P_i}{(1-\alpha_i)^2 r_i^{up}} - \gamma p h_i T \tag{15}$$

$$\frac{\partial e_i}{\partial f_i^{loc}} = 2(1 - x_i)\delta_i f_i^{loc} C_i R \tag{16}$$

Then, using the momentum gradient descent, the variables α_i and f_i^{loc} are computed, and the update equation is written as

$$\begin{cases} \alpha_i(j+1) = \alpha_i(j) - u_i(j+1) \\ f_i^{loc}(j+1) = f_i^{loc}(j) - z_i(j+1) \end{cases} \tag{17}$$

where,

$$\begin{cases} u_i(j+1) = \gamma.u_i(j) + S.d_i(j) \\ z_i(j+1) = \gamma.z_i(j) + S.d_i(j) \end{cases} \tag{18}$$

Here c is the attenuation coefficient. The iteration index is taken as j. The iteration step size is shown as s. The cumulative momentum of α_i and f_i^{loc} during the iteration process is represented by the variables u_i and z_i. The gradient function of e_i is represented by the variable d_i, which has the following expression.

$$d_i = \begin{cases} \dfrac{\partial e_i}{\partial \alpha_i} \\ \dfrac{\partial e_i}{\partial f_i^{loc*}} \end{cases} \tag{19}$$

When the maximum number of iterations is achieved, the user device's computing capabilities f_i^{loc*} and optimal time allocation ratio α_i^* can be determined using the solutions provided above. Consequently, it is possible to determine the total minimal energy consumption of N devices, that is,

$$e^* = \sum_{i \in \{1,2,\ldots N\}} e_i(x_i^*, \alpha_i^*, f_i^{loc*}) \tag{20}$$

The joint solution process of the aforementioned problems is condensed and given in the proposed algorithm to aid in comprehension.

The proposed algorithm.

Input: task size $C_i, i \in \{1, 2, \ldots N\}$
Output: the optimal value $x_i^*, \alpha_i^*, f_i^{loc*}$ and minimum energy consumption e^*
1. Begin
2. Initialize variables f_i^{loc*}, α_i;
3. Randomly input N tasks with size c_i;
4. Obtain 2N possible offloading actions;
5. Calculate the optimal offloading action x^* by (34);
6. Input (c_i, x_i^*) to train DNN network, and update network parameters according to (37) and (38);
7. Obtain the optimal parameterized function $F_{\omega,b}$;
8. Input random new task c_i into the well trained DNN network (i.e. the parameterized function $F_{\omega,b}$, and achieve the optimal offloading action x_i^*);
9. While constraints (28) to (29) are all satisfied and j is within the maximum value; do
10. Update the f_i^{loc} and α_i by (42) based on accumulated momentum (43) and the gradient descent function (44);
11. $j = j + 1$;
12. end while
13. Obtain, $f_i^{loc*}, \alpha_i, x_i^*$ and the minimum total energy consumption e^* according to (45).
14. end

3.3.3. Benefits of blockchain technology for edge cloud computing's secure computation offloading

Compared to traditional security solutions, blockchain is able to give greater security degrees for computation offloading in edge cloud computing because of its decentralized, immutable, and traceable qualities. In fact, security solutions like blockchain-based access control not only provide accurate authentication on all offloading behaviours to maintain cloud resources in a decentralized fashion, but also manage loading properly. The following explains how blockchain's distinct benefits over traditional access control systems make it possible to use it for security purposes, such as access control, in the computation offloading Figure 4.

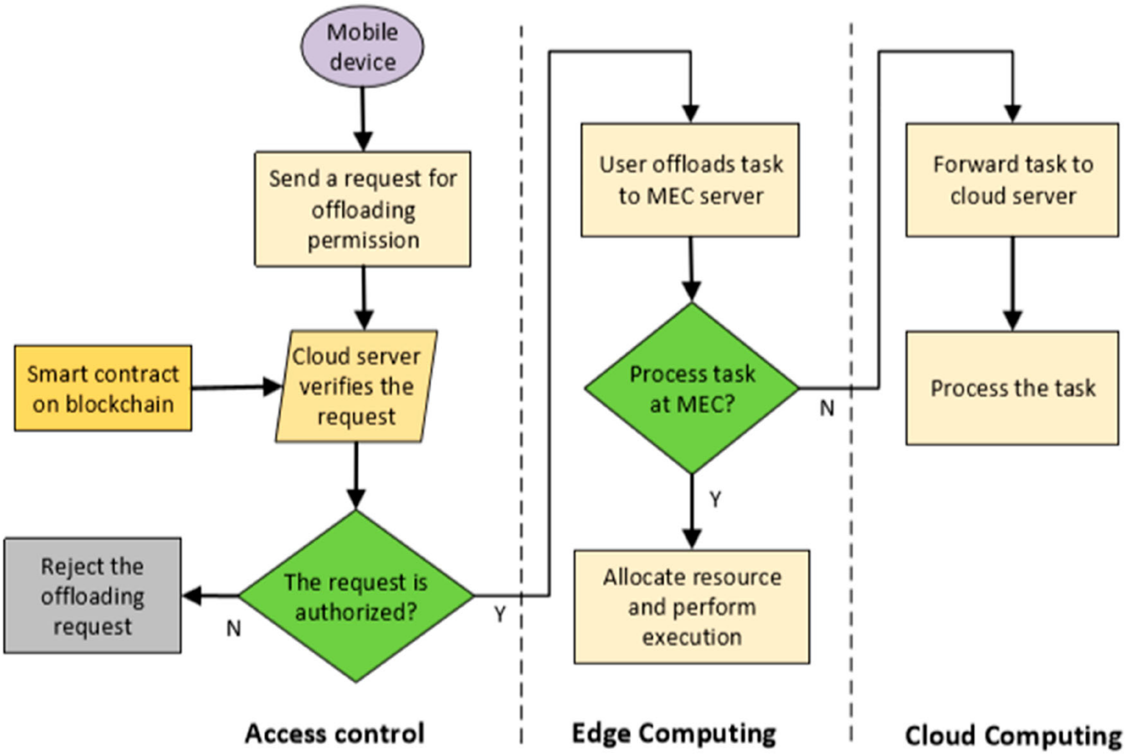

Figure 4. Offloading of computing tasks on the edge cloud with access control.

Blockchain technology can offer the offloading system a decentralized management option. Without depending on a central authority, IoT data from MDs can be kept in peer-to-peer storage on the blockchain network, ensuring quick data access and greatly enhancing mobile users' data security.

By incorporating blockchain technology into the edge-cloud computing network, the offloading system can get a reliable access control mechanism that guards against malicious offloading behaviours and possible risks to cloud compute resources. Smart contracts, which allow devices to be automatically approved and to distinguish users from adversaries, are used to achieve this. As a result, the system's offloading validity and data integrity can be greatly enhanced.

Blockchain's decentralized and unchangeable characteristics enable it to function effectively in untrusted situations, such as the IoT scenario we are considering, where trust between cloud servers, edge servers, and IoT devices is not necessary. Specifically, the blockchain's peer-to-peer network design may accomplish strong access control, high data integrity, and system security for mobile offloading.

4. Result analysis and discussion

In this section, we examine the proposed approach through numerical simulations to assess the secure computation offloading scheme's efficiency and experiments to assess the access control performance. The comparison of performance advantages of the proposed method with three other approaches are covered in this part and uses numerical simulations to assess the efficiency of the proposed method.

4.1. Experiment setup

The transmission power p_i of every device i in the proposed scenario is fixed at 0.8W. The task r_i^{up} has a transmission rate of 12 Mb/s configuration. The terminal device's computational capabilities f_i^{loc} is produced randomly between 0 and 8 MCycles/s. The terminal device needs 6 MCycles of CPU cycles in order to compute 1kbit task R. For the terminal device i, the effective capacitance coefficient δ_i is 0.00001. In the energy harvesting environment, we determine the energy conversion efficiency γ to be 0.78. The wireless channel gain h_i from the energy base station to device i is set to 1, and the starting value of the transmission power p is defined as 11W. We set the time period T to one second. Additionally, we assume that each of the ten terminal devices at the user layer has a randomly generated task size ranging from 100 to 200 kbits Figure 5.

We examine the impact on system energy usage when the local device's maximum computational capacity is set to 8, 12, and 16 MCycles/s. This bar chart illustrates that the greater the task data, the higher the overall energy consumed, regardless of the value of the highest local computing capability f^{max}. More energy is used for tasks of the same size when f^{max} is bigger. In particular, because local computing resources are relatively plentiful, more tasks opt to be executed locally when the number of tasks increases, resulting in significant energy usage.

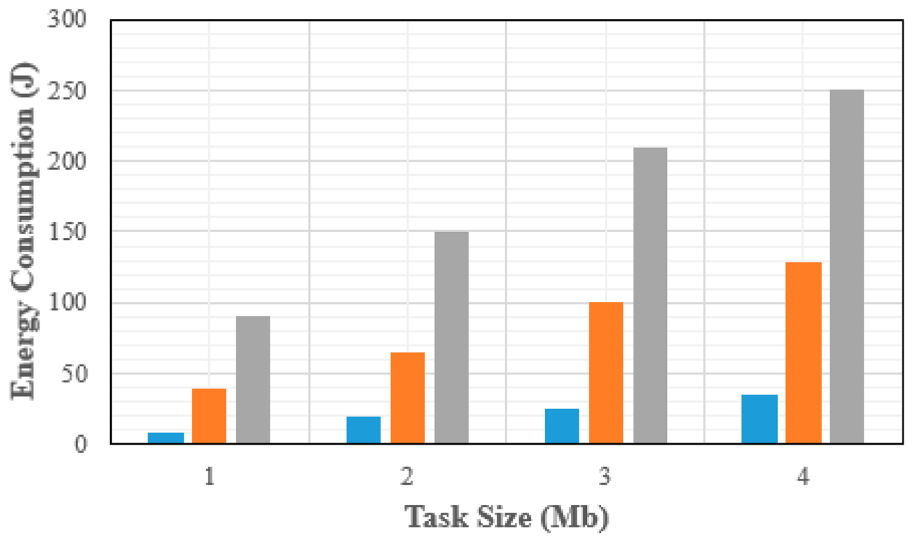

Figure 5. Total energy consumption of the proposed method under different maximum computing capabilities of local device.

When $f^{max} = 8$MCycles/s is reached, the task size increases, but the rate of rise of energy consumption is very moderate. This is illustrated by evaluating how much energy different work sizes use. In conclusion, by dynamically varying the maximum processing capacity of the local device, energy consumption can be reduced in real applications to some degree.

5. Conclusion

In this work, we present an edge computing system with trust computing and resource allocation made possible by blockchain technology. The offloading option, the time allocation ratio for energy harvesting, and the local computer resources are optimized in order to solve the prior blockchain optimization challenge. We then provide an energy efficient offloading decision and resource allocation mechanism based on cloud computing. We use simulation analysis to verify the efficiency of our proposed mechanism. Finally, we use simulation analysis to verify the efficiency of our proposed mechanism. In terms of energy savings, the proposed approach performs better than existing standard approaches and can produce the optimal value.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

[1] Jehangiri AI, Maqsood T, Ahmad Z, et al. Mobility-aware computational offloading in mobile edge networks: a survey. Cluster Comput. 2021;24(4):2735–2756. doi:10.1007/s10586-021-03268-6

[2] Hong Z, Chen W, Huang H, et al. Multihop cooperative computation offloading for industrial IoTedge-cloud computing environments. IEEE Trans Parall Distribut Syst. 2023;30(4):2759–2774.

[3] Pan J, McElhannon J. Future edge cloud and edge computing for Internet of things applications. IEEE Internet Things J. 2018;5(1):439–449. doi:10.1109/JIOT.2017.2767608

[4] Mao Y, You C, Zhang J, et al. A survey on mobile edge computing: the communication perspective. IEEE Commun Surv Tutorials. 2017;19(4):2322–2358. doi:10.1109/COMST.2017.2745201

[5] Barbarossa S, Sardellitti S, Lorenzo PD. Communicating while computing: distributed mobile cloud computing over 5G heterogeneous networks. IEEE Signal Process Mag. 2014;31(6):45–55. doi:10.1109/MSP.2014.2334709

[6] Marei M, Zaatari SE, Li W. Transfer learning enabled convolutional neural networks for estimating health state of cutting tools. Robot Comput Integr Manuf. 2021;71:1–20. doi:10.1016/j.rcim.2021.102145

[7] Arya G, Bagwari A, Chauhan DS. Performance analysis of deep learning-based routing protocol for an efficient data transmission in 5G WSN communication. IEEE Access. 2022;10:9340–9356. doi:10.1109/ACCESS.2022.3142082

[8] Lenka RK, Kolhar M, Mohapatra H, et al. Cluster-based routing protocol with static hub (CRPSH) for WSN-assisted IoT networks. Sustainability. 2022;14(12):1–17. doi:10.3390/su14127304

[9] Zhao Q, Gao Z, Sun C. Q-learning solution for optimal consensus control of discrete-time multi agent systems using reinforcement learning. J Franklin Inst. 2023;356(13):6946–6967.

[10] Sengan S, Subramaniyaswamy V, Indragandhi V, et al. Detection of false data cyberattacks for the assessment of security in smart grid using deep learning. Comput Electr Eng. 2021;93:1–13. doi:10.1016/j.compeleceng.2021.107211

[11] Ding Y, Ma K, Pu T, et al. A Deep learning-based classification scheme for false data injection attack detection in power system. Electronics (Basel). 2021;10(12):1–17. doi:10.3390/electronics10121459

[12] Chen Y, Gu W, Li K. Dynamic task offloading for internet of things in mobile edge computing via deep reinforcement learning. Int J Commun Syst. 2022;5154:1–16.

[13] Al Homssi B, Al-Hourani A, Chandrasekharan S, et al. On the bound of energy consumption in cellular IoT networks. IEEE trans green commun netw. 2019;4(2):355–364.

[14] G P, T S, S R. Deep learning approaches for ensuring secure task scheduling in IoT systems. Int J Comput Eng Res Trend. 2023;8(5):102–110.
[15] Jinaporn N, Saengudomlert P. Impact of gateway placement and energy consumption for data processing on lifetime of IoT networks, in 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2021.
[16] Wagih M, Weddell AS, Beeby S. Rectennas for radio-frequency energy harvesting and wireless power transfer: a review of antenna design [Antenna Applications Corner]. IEEE Antennas Propag Mag. 2020;62(5):95–107. doi: 10.1109/MAP.2020.3012872.
[17] Farahani B, Firouzi F, Luecking M. The convergence of IoT and distributed ledger technologies (DLT): opportunities, challenges, and solutions. J Netw Comput Appl. 2021;177:1–13. doi:10.1016/j.jnca.2020.102936
[18] Endale Mitiku A. Blockchain in healthcare and IoT: a systematic literature review. Array. 2022;14:1–11. doi:10.1016/j.array.2022.100139
[19] Song C, Sun Y, Han G, et al. Intrusion detection based on hybrid classifiers for smart grid. Comput Electr Eng. 2021;93:1–15. doi:10.1016/j.compeleceng.2021.107212
[20] Xiang C, Yang P, Wu X, et al. A step-aware sampling approach for diffusion profiling in mobile sensor networks. IEEE Trans Veh Technol. 2016;65:8616–8628. doi:10.1109/TVT.2015.2502321
[21] Kong L, Wang L, Gong W, et al. LSH-aware multitype health data prediction with privacy preservation in edge environment. World Wide Web. 2022;25:1793–1808. doi:10.1007/s11280-021-00941-z
[22] Qi L, Lin W, Zhang X, et al. A correlation graph based approach for personalized and compatible web apis recommendation in mobile app development. IEEE Trans Knowl Data Eng. 2022;11:5444–5457. doi:10.1109/TKDE
[23] Qi L, Yang Y, Zhou X, et al. Fast anomaly identification based on multiaspect data streams for intelligent intrusion detection toward secure industry 4.0. IEEE Trans Ind Inform. 2021;18(9):6503–6511.
[24] Teng T, Ma L. Deep learning-based risk management of financial market in smart grid. Comput Electr Eng. 2022;99:1–16. doi:10.1016/j.compeleceng.2022.107844
[25] Majidi SH, Hadayeghparast S, Karimipour H. FDI attack detection using extra trees algorithm and deep learning algorithm-autoencoder in smart grid. Int J Crit Infrastruct Prot. 2022;37:1–22. doi:10.1016/j.ijcip.2022.100508
[26] Tian Y, Wang Q, Guo Z, et al. A hybrid deep learning and ensemble learning mechanism for damaged power line detection in smart grids. Soft comput. 2022;26(20):10553–10561. doi:10.1007/s00500-021-06482-x
[27] Saxena S, Bhushan B, Ahad MA. Blockchain based solutions to secure IoT: background, integration trends and a way forward. J Netw Comput Appl. 2021;181:1–18. doi:10.1016/j.jnca.2021.103050
[28] Pal S, Dorri A, Jurdak R. Blockchain for IoT access control: recent trends and future research directions. J Netw Comput Appl. 2022;203:1–19. doi:10.1016/j.jnca.2022.103371
[29] Zappone A, Di Renzo M, Fotock RK. Surface-based techniques for IoT networks: opportunities and challenges. IEEE Inter Thing Mag 2022;5(4):72–77. doi:10.1109/IOTM.001.2200170