IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 May 2022, accepted 5 July 2022, date of publication 22 July 2022, date of current version 4 August 2022.
Digital Object Identifier 10.1109/ACCESS.2022.3191861

RESEARCH ARTICLE

Edge Blockchain Construction Efficiency Maximization for COVID-19 Detection in a Body Area Network

GUOZHI LI[1], XIAOFEI LI[1], XUEKUN SONG[1], (Member, IEEE), AND YUE ZENG[2]
[1]Institute of Information and Technology, Henan University of Chinese Medicine, Zhengzhou 450008, China
[2]Department of Computer Science and Engineering, Nanjing University, Nanjing 210046, China

Corresponding author: Xuekun Song (songlei51260@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 81703946 and Grant 61702164, and in part by the Henan Province Key Scientific Research Projects of Colleges and Universities of China under Grant 22A520032.

ABSTRACT Detecting traces of patients with novel coronavirus pneumonia (COVID-19) is a prerequisite for avoiding the virus's rapid spread. However, too much patient privacy data uploaded to the cloud centre will overwhelm the network and cause user information security not to be guaranteed. In this paper, we propose a personal prediction method for COVID-19 infections by perceiving the information of worn biosensors and monitoring equipment in a body area network (BAN). Edge computing and blockchain technology are introduced to solve the problems of user privacy protection and perceptual data transmission and storage. We first construct an edge body area network (EBAN) and characterize the edge blockchain cost's maximization function by considering the bandwidth, storage space, and energy consumption constraints. Then we build a blockchain without redundant perception information and select effective transmission paths by using the edge blockchain construction efficiency maximization (EBCEM) algorithm. Finally, we use the network simulator (NS-2) to simulate the performance of the EBCEM algorithm and compare it with the excellent assignment game algorithm (AGA) in terms of the effective requester ratio (ERR), effective provider ratio (EPR), edge blockchain construction success ratio (EBCSR), and average storage usage ratio (ASUR) in the EBAN.

INDEX TERMS Body area network, edge blockchain construction efficiency maximization algorithm, edge blockchain cost, edge computing, novel coronavirus pneumonia (COVID-19).

I. INTRODUCTION

The novel coronavirus pneumonia (COVID-19) has infected more than ninety million people worldwide in an unprecedented manner [1]. Although the designed early warning system based on a cloud computing network architecture can sense effective data, such as a crowd's running trajectory, contact and fever symptoms, it is difficult to judge whether a person is infected with COVID-19 [2]. As such, weak early warning networks for viral infections have prevented governments and medical institutions from quickly finding potentially infected persons and virus carriers, thereby enabling the rapid spread of the COVID-19 virus [3]. With the rise in popularity of fitness bands and smart watch devices in the body area network (BAN), physiological signs, such as heart rate, activity, sleep, etc., can be conveniently acquired from these wearable biosensors [4]. In contrast with big data from web search engines, data from wearable devices can provide more objective information on the health status of users. Through a comprehensive analysis of human body characteristics data perceived by wearable devices and the infection path detected by early warning systems, it can be predicted whether people are infected with COVID-19, as well as the method of transmission, in order to improve the prevention and treatment efficiency of COVID-19.

However, the sensing devices in the early warning networks and the wearable biosensors in EBAN upload a large amount of COVID-19 detection data to the cloud centre,

The associate editor coordinating the review of this manuscript and approving it for publication was Christos Anagnostopoulos.

which causes link congestion, meaning real-time data cannot be transmitted to the terminal management system. Thus, the problem causes a failure in finding people with COVID-19 infections in time [5]. Edge computing technology can concentrate the source-aware COVID-19 virus data on edge for processing, thereby greatly reducing the transmission load of the network [6]. Therefore, edge computing technology is the key to solving the shortcomings of predicting and tracking the COVID-19-infected population. We call the fusion between early warning networks and the internet of medical things at the sensing end through edge computing technology the edge body area network (EBAN).

Moreover, uploaded perceptual data require confidentiality, are highly redundant, and have storage difficulties [7]. These characteristics lead to the inability of management staff to search for people who have come into contact with patients diagnosed with COVID-19, thereby preventing effective virus control measures, such as early detection and early isolation. Blockchains are verifiable, non-tamperable, and traceable [8]. Also, blockchain technology can encrypt the sensed COVID-19-infected data to prevent it from being stolen. Crowd data sensed by edge computing networks will be stored on different COVID-19 detection devices in a distributed manner after being processed by blockchain technology, which will become a new design of virus infections in the EBAN [9].

The leakage of a patient's COVID-19 privacy will cause panic in society and increase the difficulty of epidemic prevention and control [10]. The key issue of blockchain-based privacy protection is decentralization, which prevents illegal users from stealing complete COVID-19 patient information [11]. Thus, the distributed storage blockchain is the best way to protect user privacy, and it can effectively resist the attacks on COVID-19 information by illegal external users.

A. MOTIVATION

To more accurately determine whether the wearing human body is infected with the COVID-19 virus, we need to read the running trajectory of the feverish person from the constructed blockchain and save the current perception data of COVID-19 to the new blockchain, which is typically characterized by effective and non-redundant information [12]. Therefore, we first need to build a blockchain-based on COVID-19 data in the EBAN and then collect data on susceptible people through COVID-19 detection devices on the network edge side, such as images and fever monitors. Then, the data will be encrypted by blockchain technology to prevent the information of infected people from being tampered with by criminals. Finally, the encrypted data will be distributed and stored in the appropriate nodes so that the prevention managers of the COVID-19 virus can consult the valuable information of infected people in a timely manner and ensure the robustness of EBAN.

Combining blockchain with edge computing technology is gaining significant attention [13]. The integration of the blockchain into EBAN will improve the efficiency of judging the COVID-19-infected population. Therefore, a new model of blockchain-based EBAN needs to be constructed, requiring us to reconsider how the implementation methods affect the network performance during the data transmission process. The blockchain can also offer a method of ensuring the information security of infected people, as well as protecting privacy, thus allowing for greater adoption of EBAN for sensing COVID-19-infected information.

The adoption of blockchain based on the COVID-19 detection data in the EBAN will lead to the following challenges:

- Bandwidth: Blockchain construction and transmission based on the COVID-19 data of infected people requires a large amount of network bandwidth, and an effective bandwidth needs to be reasonably allocated to avoid data transmission congestion in the EBAN.
- Storage space: The blockchain registers must be distributed and stored on different monitor nodes in the EBAN. As the network runs, the remaining storage space will continue to decrease. The lack of effective storage space scheduling can cause uneven distribution problems.
- Energy consumption: The EBAN is composed of devices with different computing capabilities. All the devices need to use limited energy resources when handling computing tasks and data transmissions. Therefore, a less efficient energy allocation strategy will cause the detection node to fail prematurely and therefore cannot effectively monitor the potential population of patients with COVID-19.

B. CONTRIBUTIONS

The main goal of this article is to determine the probability of COVID-19 infections based on the encrypted perceptual information of a COVID-19-infected population in an EBAN. We fully consider the bandwidth, storage space, and energy consumption in the blockchain construction process and select the optimal transmission paths and blockchain storage detection nodes, thereby maximizing the network resources to achieve a balanced distribution and extending the life cycle of the EBAN.

In summary, the contributions of this article can be summarized as follows.

- We establish a body area network for edge computing (EBAN) that builds a blockchain of the COVID-19-infected population. EBAN is responsible for saving and transmitting the COVID-19 data detected by wearable biosensors and edge sensing devices. We then give the system model based on the bandwidth, storage space and energy consumption of the EBAN.
- We describe the maximization function of the edge blockchain cost and the constraints of the bandwidth, storage space and energy consumption according to the real-time status of the EBAN and use the edge blockchain cost optimization (EBCO) algorithm to find the optimal value. The optimal value of the edge blockchain cost is a reference scale for selecting the transmission paths and blockchain storage nodes.

- We propose the edge blockchain construction efficiency maximization (EBCEM) algorithm in order to construct the blockchain of COVID-19-infected people and store it in effective detection nodes in the EBAN, which provides managers with high-value information, high confidentiality and traceability.

The remainder of this article is organized as follows: Section III provides the network and system model. Previous related work is presented in Section II. Section IV formalizes the edge blockchain cost function. The edge blockchain cost optimization (EBCO) algorithm is presented in V. In Section VI, we propose the edge blockchain construction efficiency maximization (EBCEM) algorithm. Section VII evaluates the performance of the EBCEM algorithm. Finally, the conclusions of this article are summarized in Section VIII.

II. RELATED WORK

Recent studies have been published that highlight the applications, benefits, research issues and challenges of the use of blockchains in environmental inspection networks. The paper proposed a distributed blockchain-based protection framework to enhance the self-defensive capability of modern power systems against cyber-attacks [14]. And for building upon the idea of using blockchain as the underlying technology to enable tracing transactions for service contracts and dispute arbitration, this paper proposes a novel consensus protocol that is suitable for crowdsourcing as well as the general online service industry [15]. The authors in [16] provided a comprehensive top-down survey of the most recent proposed security and privacy solutions in IoT. They discuss particularly the benefits that new approaches such as blockchain and Software Defined Networking (SDN) can bring to the security and privacy in the Internet of Things (IoT) in terms of flexibility and scalability. However, only a few articles have introduced the use of blockchain for protecting user privacy. The authors in [17] built a double-layer blockchain architecture for the local area information transmission of a vehicular named data networking (VNDN). Then, the article proposed an assignment game algorithm (AGA) that addresses the problem of VNDN data sharing, which broadly addresses reputation management, information security, data supply and demand matching.

III. NETWORK AND SYSTEM MODEL

A. EDGE BODY AREA NETWORK

In this section, an edge body area network (EBAN) is constructed to sense the information of infected people and share these data securely. Due to the complex flow of people on roads and because only infrared cameras can be used to detect fever, coughing, and other symptoms, it is difficult to determine whether individuals in a group are infected with the COVID-19 virus [18]. According to a study on fever and cardiac rhythm, the heart rate increases by 8.5 beats per minute, on average, for every 1°C increase in body temperature; thus, an elevated resting heart rate (RHR) might be related to a fever caused by COVID-19 or an influenza-like illness [19]. The basic anomaly detection method is based on the elevated RHR. Because a shortened sleep length also causes an increase in the RHR [20], we weaken the contribution of this factor in the physiological anomaly detection method. Obviously, if a person has the above characteristics and has been in contact with persons suspected of being infected with the COVID-19 virus in a recent period, the probability of this person being infected with the COVID-19 virus will be much higher.

In addition, there will be a large amount of redundant data in the perceived crowd information due to the mobility of infected people. In the process of data transmission and blockchain construction, we divide the EBAN into different groups. Each COVID-19 detection device transmits the sensed information to a node with stronger data computing and storage capabilities in the current group. We call this type of group an edge computing system. Blockchain construction and storage node selection are completed through such nodes, which are called edge computing servers (ECSs). Moreover, some blockchain construction requirements may not be satisfied within the scope of the current group. In this case, the edge computing server of each group will send its unsatisfactory requirements to the edge computing servers of other groups.

The EBAN is shown in Fig. 1. The red virtual round indicates the safe distance $\mathcal{R}$. When the distance between the suspected infected person and the virus carrier is less than $\mathcal{R}$, we believe that the suspected infected person will be infected with COVID-19. Otherwise, the suspected infected person will be excluded from the COVID-19 infection because they have not been in contact with the virus carrier. The main core components of the EBAN include:

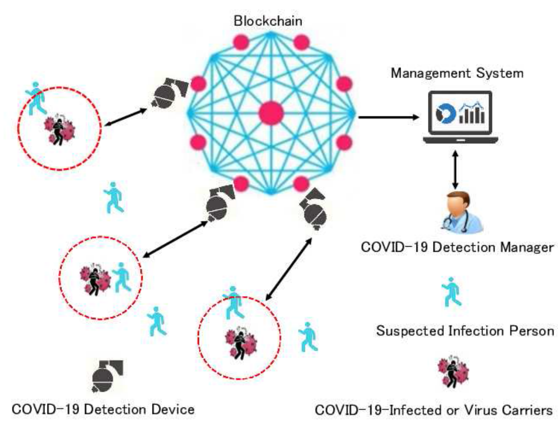


FIGURE 1. Edge body area network.

(1) COVID-19 Detection Units (CDUs): Each COVID-19 detection device and wearable biosensor is equipped with different types of CDUs for detecting the statuses of infected people. CDUs can sense personal information, body temperature, proximity, walking trajectory, etc. The edge device can then collect sensed data from different environments according to the COVID-19 detection request.

When the CDUs submit the perceived COVID-19 detection data to the edge computing server to build the blockchain, we call these CDUs *requesters*.

(2) Construction and Scheduling Units (CSUs): CSUs are stored on the edge computing servers of each group in the EBAN. We can find the encrypted data stored on the COVID-19 detection device by using the CSUs. Each sensed data point corresponds to the CSUs on the edge computing server, which improves the data retrieval efficiency.

(3) Edge Blockchain Units (EBUs): The storage space of each COVID-19 detection device is considered an EBU. The EBUs store the sensed COVID-19 data of the local region on different blockchains, and then we encrypt the stored data through a set mechanism. In addition, we attempted to remove as much redundant COVID-19 detection data on the EBUs as possible in order to save on the precious storage resources of the EBAN. When we use the storage space in the COVID-19 detection device to store the blockchain, such devices are called *providers*.

During the construction and storage of the COVID-19 data blockchain, the improper selection of the CSUs and EBUs will result in an imbalanced allocation of storage and bandwidth resources because the status cannot be correctly sensed in the EBAN. In addition, we assume that the detection data of COVID-19-infected people are transmitted to the cloud centre, where a large amount of data will cause an enormous computational and transmission burden on the cloud centre. Therefore, this is a very appropriate choice for edge computing networks built by the blockchain in this article. When a person's body temperature and heart rate meet the RHR rules, the CDUs will send a COVID-19 confirmation request to the CSUs (the CSU is also *requester*), and the CSUs will confirm the infection of the COVID-19 virus according to the running trajectory of the person detected and store the completed blockchain distributed in one or more EBUs. These responding EBUs are also *providers*. The results of all the constructed blockchains will be displayed by the COVID-19 detection system responsible for the COVID-19 detection manager, which forms a complete edge body area network.

B. SYSTEM MODEL

In the construction process of EBAN, n CDUs are divided into k groups, where each group is considered an edge computing system. The CDUs in each edge computing system can be linked to one another, but they cannot be connected to the devices of other edge computing systems due to the limited transmission distance. $R_{i\to j}$ indicates that the COVID-19 data blockchain of $requester_i$ constructed by the edge computing server is connected to $provider_j$. In the process of constructing a blockchain that senses COVID-19-infected people's information and effectively transmits it to the provider, three key factors need to be considered: the link bandwidth, the storage space of the provider, and the energy consumed during data transmission. Therefore, we outline three models for the effective bandwidth level, storage blockchain level, and effective energy consumption level in this section.

(1) Effective bandwidth level (EBL): Successful COVID-19 blockchain allocation requires a reliable idle link bandwidth to support it. Otherwise, the completed blockchain cannot be transferred to the storage space of the providers, which makes the COVID-19 blockchain construction fail. The EBL reflects the idle level of link bandwidths in the EBAN; thus, we define the EBL as follows:

$$EBL_{i\to j} = \frac{B_{i\to j}}{\sum\limits_{j\in J} B_{i\to j}} \tag{1}$$

where $B_{i\to j}$ represents a valid link bandwidth from $requester_i$ to $provider_j$, and J is a set of all available providers.

(2) Storage Block Level (SBL): in the EBAN, the storage block level (SBL) reflects the concept that the provider satisfies the storage capability of the requester's blockchain construction requirement. The larger the SBL value is, the more reliable the storage resources that can be provided to the requester, and vice versa. $S_{i\to j}$ indicates that the blockchain storage requirement of $requester_i$ needs to be satisfied by $provider_j$. Since there are 2 types of SBL allocations, i.e., success $S_{i\to j}$ (SS) and failure $S_{i\to j}$ (FS), the SBL can be classified into two categories:

$$SS_{i\to j} = \frac{SS_{i\to j}}{\sum\limits_{j\in J} SS_{i\to j}}, FS_{i\to j} = \frac{FS_{i\to j}}{\sum\limits_{j\in J} FS_{i\to j}} \tag{2}$$

where SS/SF indicates the proportion of successful/failed storage blockchain allocation. In this way, SS/SF denotes the positive/negative SBL. Therefore, we define an SBL as follows:

$$SBL_{i\to j} = SS_{i\to j} f'_{SS_{i\to j}} - FS_{i\to j} f'_{FS_{i\to j}} \tag{3}$$

where f' is the frequency of successful/failed storage blockchain allocations.

(3) Energy Consumption Level (ECL): During the construction of the COVID-19 blockchain, each completed blockchain needs to be transmitted to different providers. Therefore, it is necessary to consider the residual energy of the CDUs and the lower energy consumption of the relay nodes. The EBL is given by the following:

$$ECL_{i\to j} = \frac{E_{i\to j}}{\sum\limits_{j\in J} \varsigma_{i\to j} E_{i\to j}} \tag{4}$$

where $\varsigma_{i\to j}$ is the effective residual energy ratio and $E_{i\to j}$ indicates the residual energy of $provider_j$ for satisfying the blockchain required by $requester_i$.

We comprehensively consider three factors in the process of building a blockchain in the EBAN, i.e., the effective bandwidth b, the effective storage space of provider s, and the effective energy e required for blockchain transmission. We define the system model of the edge blockchain cost as

$$C(b, s, e) = sSBL_{i\to j} \log_2^{(1+bEBL_{i\to j})^{eECL_{i\to j}}} \tag{5}$$

where b, s, and e represent the minimum values of the bandwidth, storage space, and energy consumption, respectively, required by $requester_i$ for blockchain construction and the transmission requirements.

Note that any ECS can communicate freely with other ECSs in the EBAN, and at least one valid link can be found for each established COVID-19 blockchain. In the constructed edge blockchain model, we should first select $|J|$ nodes as storage nodes, which takes $O(|J|)$ time to complete. It then takes $O(1 + |J|)$ time to generate a COVID-19 blockchain, and we need $O(|J||J|)$ time to select the paths and transfer the completed built blockchain to the destination node, $|J| \leq n$. Therefore, the total time complexity of this model is $O(n^2 + 3n + 1)$.

IV. PROBLEM FORMULATION

To establish a reliable edge blockchain based on COVID-19 detection information, we should avoid transmitting perceptual data to unreliable providers [21]. In addition, the distributed storage of edge blockchains can save limited computing and storage resources of the CSUs in the EBAN. The establishment of an edge blockchain requires sufficient computing and energy resources, therefore ordinary CSUs cannot perform blockchain construction in the edge computing system of the EBAN. The EBAN consists of three parts, where the CDUs are responsible for sensing infected population data. The CSUs and EBUs respond to the CDUs and manage crowd requirements, and a scheduling mechanism is established by the edge computing server in the EBAN.

To ensure the establishment of a reliable edge blockchain, the edge storage requirement of the requester at any time point cannot exceed the sum of the free storage space and the remaining storage space of the provider at the current time; i.e.,

$$s_j(t) - s'_j(t-1) - r_j(t) \leq 0 \tag{6}$$

where $s_j(t)$, $s'_j(t)$, and $r_j(t)$ represent the blockchain storage requirements of $requester_i$, the remaining storage space, and the released storage space of $provider_j$ at time t, respectively.

To reasonably use the storage space of the providers, the sum of the remaining storage space and the released space at any time point minus the blockchain storage requirement of $requester_i$ cannot exceed the maximum storage limit of $provider_j$, thereby preventing the storage space of the CSUs from being exhausted.

$$s'_j(t-1) + r_j(t) - s_j(t) - \mathcal{U} \leq 0 \tag{7}$$

where $\mathcal{U}$ is the corresponding upper limit of the storage space of $provider_j$ in the EBAN.

By comprehensively considering the bandwidth, storage space, and energy consumption, we use the edge blockchain cost function to describe the problem of blockchain construction in order to select the costliest blockchain and the appropriate blockchain transmission paths in the EBAN. Therefore, the objective function for edge blockchain construction is

$$OPT-1 \quad \max \sum_{i\in I}\sum_{j\in J} \rho_{ij} C(b, s, e) \tag{8}$$

subject to $\forall i \in I, \forall j \in J$,

$$\begin{aligned} &C1: \sum_{i\in I}\sum_{j\in J}\frac{b_{ij}(t)}{\pi_{ij}} + \sum_{i\in I}\sum_{j\in J}\frac{\tilde{b}_{ij}(t)}{\pi_{ij}} \leq 1,\\ &C2: \rho_{ij} \in \{0, 1\},\\ &C3: s_j(t) - s'_j(t-1) - r_j(t) \leq 0,\\ &C4: s'_j(t-1) + r_j(t) - s_j(t) - \mathcal{U} \leq 0, \end{aligned} \tag{9}$$

where $b_{ij}(t)$ and $\tilde{b}_{ij}(t)$ indicate the bandwidth requirement of $requester_i$ and the released bandwidth in the ij-th link at time t in the EBAN, respectively, and π is the bandwidth capacity of the linked requester.

A CSU cannot send or receive data from multiple nodes at the same time in the EBAN. Therefore, constraint $C1$ describes the relay capability condition of the relay nodes. The selection scale of effective providers is expressed in constraint $C2$, where the selected provider has the highest edge blockchain cost. All the providers must have sufficient storage space to ensure the construction of an edge blockchain in the EBAN, where the constraint derived below $C5$ based on constraint $C3$ and $C4$ indicates that the remaining storage space of the provider remains stable, to a certain degree.

According to Equation (6) and Equation (7), we obtain the constraint that $provider_j$ responds to the edge blockchain storage requirement of $requester_i$ via Theorem (1) in the EBAN.

Theorem 1: At time t, the remaining storage space of $provider_j$ ($s'_j(t)$) is not less than $s'_j(0) + \sum_{t=1}^{T}(r_j(t) - s_j(t))$ in the EBAN, where $0 \leq t \leq T$.

Proof: According to Equation (6) and Equation (7), when $t = 1$ we can obtain

$$\begin{aligned} s'_j(1) &\leq \min(\mathcal{U}, \max(0, s'_j(0) - s_j(1) + r_j(1)))\\ &= \min(\mathcal{U}, s'_j(0) - s_j(1) + r_j(1))\\ &= s'_j(0) - s_j(1) + r_j(1) \end{aligned} \tag{10}$$

We generalize Equation (10) to any time point; thus, we can obtain $s'_j(t)$ as follows.

$$\begin{aligned} s'_j(t) &\leq s'_j(t-1) - s_j(t) + r_j(t)\\ &= s'_j(t-2) - (s_j(t) + s_j(t-1)) + (r_j(t) + r_j(t-1))\\ &= \cdots\cdots\\ &= s'_j(0) + \sum_{t=1}^{T}(r_j(t) - a_j(t)) \end{aligned} \tag{11}$$

which completes the proof.

Therefore, we merge constraints $C3$ and $C4$ into the following constraint via Theorem (1).

$$C5: \sum_{i=1}^{I}\sum_{j=1}^{J} |T|(s_{ij}(t) - r_{ij}(t)) \leq |J|(s'_{ij}(0) - s'_{ij}(t)), \tag{12}$$

The optimization function (OPT-1) maximizes the utilization of the edge blockchain cost, which is constrained by the bandwidth requirement sent by $requester_i$ and the effective storage space provided by $provider_j$ for establishing and storing the blockchain of the COVID-19-infected population data in the EBAN. Constraint $C2$ guarantees that the value of ρ will be either one or zero. We reformulate this constraint to the selection of proper relay nodes as follows.

$$C6 : \sum_{i\in I}\sum_{j\in J}\rho_{ij}e_{ij} \geq |J|e_{\omega} \tag{13}$$

where e_{ω} is the energy consumption for satisfying the data transmission needs of $requester_i$ in the EBAN.

We adopt the Lagrangian dual method [22] to solve the optimization function (OPT-1), where the optimization variables b^*, s^*, and e^* will be found. The maximization function value $C(b^*, s^*, e^*)$ is a metric for selecting reliable providers in the EBAN.

V. EDGE BLOCKCHAIN COST SOLUTION

In this section, we propose an edge blockchain cost optimization (EBCO) algorithm to find the optimal value of the optimization function (OPT-1). Therefore, we first use the Lagrangian dual method to iterate the optimization mechanism of the objective function (OPT-1) and the constraints. Then we obtain the iteration termination conditions for the objective function (OPT-1) and the constraint conditions by discussing the Lagrangian dual method.

A. LAGRANGIAN DUAL APPROACH

We first use the Lagrangian dual decomposition approach [23] to convert the optimization function (OPT-1) into an unconstrained optimization expression. Afterward, the Lagrangian multipliers μ^*, ν^* and δ^* are derived using the following steps.

We introduce Lagrangian multipliers μ, ν and δ to relax constraints $C1$, $C5$ and $C6$ [24]. The Lagrange function can then be expressed as

$$\begin{aligned}
L(&b, s, e, \mu, \nu, \delta)\\
&= \sum_{i\in I}\sum_{j\in J}\rho_{ij}C(b, s, e)\\
&\quad +\mu_{ij}(1-\sum_{i\in I}\sum_{j\in J}\frac{b_{ij}(t)}{\pi_{ij}} - \sum_{i\in I}\sum_{j\in J}\frac{\tilde{b}_{ij}(t)}{\pi_{ij}})\\
&\quad +\nu_{ij}(|J|(s'_{ij}(0)-s'_{ij}(t)) - \sum_{i=1}^{I}\sum_{j=1}^{J}|T|(s_{ij}(t)-r_{ij}(t)))\\
&\quad +\delta_{ij}(\sum_{i\in I}\sum_{j\in J}\rho_{ij}e_{ij} - |J|e_{\omega})
\end{aligned} \tag{14}$$

where $\mu \geq 0, \nu \geq 0, \delta \geq 0$ and μ, ν, δ are Lagrangian multipliers. The dual problem of the optimization function (OPT-1) can be expressed as

$$\min D(\mu, \nu, \delta) \tag{15}$$

Among them,

$$D(\mu, \nu, \delta) \equiv \max_{b,s,e} L(b, s, e, \mu, \nu, \delta) \tag{16}$$

Equations (14) and (16) can then be rewritten as

$$D(\mu, \nu, \delta) = \max(D_0 + D(b, s, e, \mu, \nu, \delta)) \tag{17}$$

where

$$\begin{aligned}
D_0 &= \mu_{ij}(1-\sum_{i\in I}\sum_{j\in J}\frac{\tilde{b}_{ij}(t)}{\pi_{ij}}) + |J|\nu_{ij}(s'_{ij}(0)\\
&\quad - s'_{ij}(t)) - |J|\delta_{ij}e_{\omega}
\end{aligned} \tag{18}$$

$$\begin{aligned}
D(b, s, e, \mu, \nu, \delta) &= \sum_{i\in I}\sum_{j\in J}\rho_{ij}C(b, s, e)\\
&\quad -\mu_{ij}\sum_{i\in I}\sum_{j\in J}\frac{b_{ij}(t)}{\pi_{ij}} - \mu_{ij}\sum_{i\in I}\sum_{j\in J}\frac{\tilde{b}_{ij}(t)}{\pi_{ij}}\\
&\quad -\nu_{ij}(\sum_{i=1}^{I}\sum_{j=1}^{J}|T|(s_{ij}(t)-r_{ij}(t)))\\
&\quad +\delta_{ij}\sum_{i\in I}\sum_{j\in J}\rho_{ij}e_{ij}
\end{aligned} \tag{19}$$

We find the gradient of the multivariable function (Equation (16)) as follows:

$$\nabla D(b, s, e, \mu, \nu, \delta) = 0 \tag{20}$$

The optimal available bandwidth, storage space and energy consumption are derived using Equations (19) and (20).

$$\begin{aligned}
b^* &= \frac{\ln 2eECL_{i\to j}(1+bEBL_{i\to j})^{eECL_{i\to j}-1} + EBL_{i\to j}}{\ln 2(1+bEBL_{i\to j})^{eECL_{i\to j}}}\\
&\quad -\mu_{ij}\sum_{i\in I}\sum_{j\in J}\frac{1}{\pi_{ij}}
\end{aligned} \tag{21}$$

$$s^* = SBL_{i\to j}\log_2^{(1+bEBL_{i\to j})^{eECL_{i\to j}}} - |T|\sum_{i\in J}\sum_{j\in J}\nu_{ij} \tag{22}$$

$$\begin{aligned}
e^* &= \frac{\ln(eECL_{i\to j}+2)(1+bEBL_{i\to j})^{eECL_{i\to j}-1}}{\ln 2(1+bEBL_{i\to j})^{eECL_{i\to j}}}\\
&\quad +\sum_{i\in J}\sum_{j\in J}\rho_{ij}\delta_{ij}
\end{aligned} \tag{23}$$

We update the recursive form of the optimal Lagrangian multiplier iteratively as follows:

$$\mu(t'+1) = (\mu(t') - \tilde{s}_1(1-\sum_{i\in I}\sum_{j\in J}\frac{b_{ij}(t)}{\pi_{ij}} - \sum_{i\in I}\sum_{j\in J}\frac{\tilde{b}_{ij}(t)}{\pi_{ij}}))^+ \tag{24}$$

$$\begin{aligned}
\nu(t'+1) &= (\nu(t') - \tilde{s}_2(|J|(s'_{ij}(0)-s'_{ij}(t))\\
&\quad - \sum_{i=1}^{I}\sum_{j=1}^{J}|T|(s_{ij}(t)-r_{ij}(t)))^+
\end{aligned} \tag{25}$$

$$\delta(t'+1) = (\delta(t') - \tilde{s}_3(\sum_{i\in I}\sum_{j\in J}\rho_{ij}e_{ij}) - |J|e_{\omega})^+ \tag{26}$$

where t' is the number of iterations and $\tilde{s}_1$, $\tilde{s}_2$ and $\tilde{s}_3$ are the iteration step sizes.

B. EDGE BLOCKCHAIN COST OPTIMIZATION ALGORITHM

Appropriate iteration numbers and step sizes will increase the speed and accuracy of the function convergence [25]. We therefore select the effective bandwidth b required for the construction of the blockchain with data on COVID-19-infected people for analysis in the EBAN. When $\| b^{t'+1} \|$ is less than $\| b^{t'} \|$ during the iterative process, oscillation in the random variable space $\mathcal{X}$ will occur, which reduces the speed of function convergence. Therefore, we can obtain

$$\| b^{t'+1} \| \geq \| b^{t'} \| \tag{27}$$

Equation (27) is equivalent to

$$(b^{t'+1} - b^{t'})^T (b^{t'+1} + b^{t'}) \geq 0 \tag{28}$$

To ensure that the optimization function converges quickly, $b^{t'+1}$ and $b^{t'}$ must satisfy the following conditions.

$$\begin{cases} (b^{t'+1} - b^{t'})^T b^{t'+1} \geq 0, \\ (b^{t'+1} - b^{t'})^T b^{t'} \geq 0. \end{cases} \tag{29}$$

In the t'-th iteration, the reliability decision value of the constraint function $\vartheta^{t'}$ mainly depends on the search results $b^{t'}$ of the optimization function (OPT-1). Similarly, the reliability decision value of the constraint function $\vartheta^{t'+1}$ is still terminated by the search result $b^{t'+1}$ of the optimization function in the $(t'+1)$-th iteration. In the process of obtaining $b^{t'+1}$, the constraint target $\vartheta^{t'}$ has a given initial position for the current search step and does not directly participate in calculating the $(t'+1)$-th iteration. Therefore, the determination value of t' will directly determine the speed and accuracy of the convergence of the optimization function (OPT-1).

According to the Karush-Kuhn-Tucker (KKT) condition [26], the optimal solution $C(b^*, s^*, e^*)$ should satisfy the following conditions.

$$\begin{cases} \nabla \|C(b^*, s^*, e^*)\| - \mu_{ij} \nabla \mathcal{B}(b^*) - \nu_{ij} \nabla \mathcal{S}(s^*) \\ -\delta_{ij} \nabla \mathcal{E}(e^*) = 0, \\ \|C(b^*, s^*, e^*)\| = \|C(b^{t'+1}, s^{t'+1}, e^{t'+1})\|. \end{cases} \tag{30}$$

where μ, ν, and δ are Lagrange multipliers and $\mathcal{B} = \sum_{i \in I} \sum_{j \in J} \frac{b_{ij}^*(t)}{\pi_{ij}} + \sum_{i \in I} \sum_{j \in J} \frac{\tilde{b}_{ij}(t)}{\pi_{ij}}$, $\mathcal{S} = \sum_{i=1}^{I} \sum_{j=1}^{J} |T| (s_{ij}^*(t) - r_{ij}(t))$, and $\mathcal{E} = \sum_{i \in I} \sum_{j \in J} \rho_{ij} e_{ij}^*$.

Equation (30) can be rewritten and the following formulas can be obtained:

$$\begin{cases} \frac{C(b^*, s^*, e^*)}{\|C(b^*, s^*, e^*)\|} - \mu_{ij} \nabla \mathcal{B}(b^*) - \nu_{ij} \nabla \mathcal{S}(s^*) \\ -\delta_{ij} \nabla \mathcal{E}(e^*) = 0, \\ \|C(b^*, s^*, e^*)\| = \|C(b^{t'+1}, s^{t'+1}, e^{t'+1})\|. \end{cases} \tag{31}$$

Both sides of the first formula in Equation (31) are multiplied by $\nabla \mathcal{B}(b^*)$, $\nabla \mathcal{S}(s^*)$, and $\nabla \mathcal{E}(e^*)$.

$$\begin{aligned} &\frac{(\nabla \mathcal{B}(b^*) \nabla \mathcal{S}(s^*) \nabla \mathcal{E}(e^*))^T C(b^*, s^*, e^*)}{\|C(b^*, s^*, e^*)\|} \\ &\quad -\mu_j \nabla \mathcal{S}(s^*) \nabla \mathcal{E}(e^*) \|\nabla \mathcal{B}(b^*)\|^2 \\ &\quad -\nu_j \nabla \mathcal{B}(b^*) \nabla \mathcal{E}(e^*) \|\nabla \mathcal{S}(s^*)\|^2 \\ &\quad -\delta_j \nabla \mathcal{B}(b^*) \nabla \mathcal{S}(s^*) \|\nabla \mathcal{E}(e^*)\|^2 = 0, \end{aligned} \tag{32}$$

The optimal values b^*, s^*, and e^* represent the surface of the edge blockchain cost function $C(b^{t'+1}, s^{t'+1}, e^{t'+1}) = C(b^*, s^*, e^*)$, where the spherical surface with radius $\mathcal{B}(b^{t'+1}) = \|b^*\|$, the spherical surface with radius $\mathcal{S}(s^{t'+1}) = \|s^*\|$, and the spherical surface with radius $\mathcal{E}(e^{t'+1}) = \|e^*\|$ are tangent to points b^*, s^*, and e^*, respectively. Therefore, the gradients $\nabla \mathcal{B}(b^*)$, $\nabla \mathcal{S}(s^*)$, and $\nabla \mathcal{E}(e^*)$ of the function surface $C(b^*, s^*, e^*) = L(b^*, s^*, e^*)$ at points b^*, s^*, and e^* are in opposite directions; that is, $(\nabla \mathcal{B}(b^*) \nabla \mathcal{S}(s^*) \nabla \mathcal{E}(e^*))^T C(b^*, s^*, e^*) = \|\nabla \mathcal{B}(b^*) \nabla \mathcal{S}(s^*) \nabla \mathcal{E}(e^*)\| \|L(b^*, s^*, e^*)\|$. We input this formula into Equation (32), and obtain $\mu = \frac{1}{\|\nabla \mathcal{B}(b^*)\|}$, $\nu = \frac{1}{\|\nabla \mathcal{S}(s^*)\|}$, and $\delta = \frac{1}{\|\nabla \mathcal{E}(e^*)\|}$. We substitute these formulas into Equation (31), which can obtain the expression of $C(b^*, s^*, e^*)$ under the constraint conditions as

$$\begin{aligned} C(b^*, s^*, e^*) &= \frac{\nabla \mathcal{B}(b^*)}{\|\nabla \mathcal{B}(b^*)\|} \|C(b^*, s^*, e^*)\| \\ &\quad + \frac{\nabla \mathcal{S}(s^*)}{\|\nabla \mathcal{S}(s^*)\|} \|C(b^*, s^*, e^*)\| \\ &\quad + \frac{\nabla \mathcal{E}(e^*)}{\|\nabla \mathcal{E}(e^*)\|} \|C(b^*, s^*, e^*)\|, \end{aligned} \tag{33}$$

If $\|C(b^{t'+1}, s^{t'+1}, e^{t'+1})\|$ is the current optimal solution $C(b^*, s^*, e^*)$, then

$$\begin{aligned} &C(b^{t'+1}, s^{t'+1}, e^{t'+1}) \\ &\quad = \frac{\nabla \mathcal{B}(b^{t'+1})}{\|\nabla \mathcal{B}(b^{t'+1})\|} \|C(b^{t'+1}, s^{t'+1}, e^{t+1})\| \\ &\quad + \frac{\nabla \mathcal{S}(s^{t'+1})}{\|\nabla \mathcal{S}(s^{t'+1})\|} \|C(b^{t'+1}, s^{t'+1}, e^{t'+1})\| \\ &\quad + \frac{\nabla \mathcal{E}(e^{t'+1})}{\|\nabla \mathcal{E}(e^{t'+1})\|} \|C(b^{t'+1}, s^{t'+1}, e^{t'+1})\|, \end{aligned} \tag{34}$$

Through the above analysis, Equation (34) will be the termination condition of the iterative optimal search process for the edge blockchain cost function [27]. When the iteration value is greater than $C(b^{t'+1}, s^{t'+1}, e^{t'+1})$, we set the current iteration number to $t'+1$. In this way, we can iteratively find the providers that are most suitable for building the blockchain for the COVID-19-infected population data in the EBAN. The detailed process of the EBCO algorithm is described in Algorithm 1.

VI. EDGE BLOCKCHAIN CONSTRUCTION EFFICIENCY MAXIMIZATION ALGORITHM

In this section, we first describe the process of the edge blockchain construction efficiency maximization (EBCEM)

Algorithm 1 Edge Blockchain Cost Optimization (EBCO) Algorithm

1: **Require**: Convergence accuracy ε, $t' = 0$, initialization multiplier μ_0, ν_0, δ_0.
2: Obtain the optimal values $b^*(\mu_{ij}, \nu_{ij}, \delta_{ij})$, $s^*(\mu_{ij}, \nu_{ij}, \delta_{ij})$ and $e^*(\mu_{ij}, \nu_{ij}, \delta_{ij})$ by formulas (21), (22), (23);
3: **while** $|C(\mu^{t'+1}, \nu^{t'+1}, \delta^{t'+1}) - C(\mu^{t'}, \nu^{t'}, \delta^{t'})| > \varepsilon$ **do**
4: Update $\mu^{t'}$, $\nu^{t'}$, and $\delta^{t'}$ using Equations (24), (25), and (26), respectively;
5: $t' = t' + 1$;
6: **end while**.
7: Find the optimal value of the Lagrangian multipliers μ^*, ν^*, δ^*.
8: **Output** The optimal edge blockchain cost $C(a^*, s^*, e^*)$.

algorithm, and then we analyse the time complexity of the EBCEM algorithm.

A. EBCEM ALGORITHM DESCRIPTION

To improve the efficiency of edge blockchain construction, we should comprehensively consider the appropriate data transmission bandwidth b, the available storage space s, and the energy consumption e in the EBAN. Therefore, the proposed EBCEM algorithm constructs an edge blockchain to satisfy requirement ω and select the optimal transmission paths.

In the first step, the ECSs first divide the edge blockchain construction requirement ω sent into l parts in the EBAN, where $l \leq L'$ and L' is the maximum value of the data flow split, X represents the packet set of split requirement ω, and T_k indicates the historical set of all split packets. The COVID-19 detection devices of ECSs in an edge computing system are in the same or a similar environment. As a result, the different COVID-19 detection devices of ECSs perceive a large amount of the same redundant data in the EBAN. When the blockchain construction requirement ω is split, we need to remove these redundant packets in order to reduce the energy and computing resources of the EBAN. In addition, we sort the nonredundant split data packets in an increasing order to improve the efficiency of the construction and retrieval of the edge blockchain.

The second step is to establish the edge blockchain based on the data packets split in the EBAN. The edge blockchain built is a polynomial with split COVID-19 perception detection data ω. The ECSs only need to store the serial numbers of the corresponding items with the split data packet and the polynomial, where the corresponding items containing the split data are stored in the appropriate COVID-19 detection devices in the EBAN. Each ECS stores the numbers of all the COVID-19 detection devices in the EBAN. These numbers are the proof-of-work (PoW) that COVID-19 managers need to provide for reading the data, thereby enhancing the security of the COVID-19-infected population during virus control.

The optimal transmission path with the maximum edge blockchain construction cost $C(b^*, s^*, e^*)$ is selected by setting T_j' in the third step. The larger the value of the edge blockchain cost $C(b^*, s^*, e^*)$, the more efficient and secure the generated blockchain, and the blockchain construction requirements can be efficiently transmitted to the ECSs. Appropriate parameter constraints b^*, s^* and e^* will optimize the selected storage space in the COVID-19 detection device of ECS, thereby saving storage, computing and energy resources, and extending the life cycle of the EBAN. $\mathcal{L}$ is the number of effective blockchain transmission paths. Finally, the network status in the EBAN will be updated.

We summarize the procedure of the edge blockchain construction efficiency maximization (EBCEM) algorithm in Algorithm 2.

B. TIME COMPLEXITY ANALYSIS

To prove the efficiency of the edge blockchain construction efficiency maximization (EBCEM) algorithm, we analyse the time complexity of the EBCEM algorithm as follows.

First, the EBCEM algorithm will split requirement ω for edge blockchain construction and arrange these nonredundant packets in an increasing order; the procedure consumes $O(\lfloor \frac{\omega}{l} \rfloor \lg k + (k-1)^2)$ time, where $l \leq k \leq L'$. Second, it takes $O(k-3)$ time to generate a polynomial curve. In the end, the EBCEM algorithm needs to select the optimal transmission path with the maximum edge blockchain cost function $C(b^*, s^*, e^*)$. The transmission path needs to compare $C(b^*, s^*, e^*)$ for the reliable paths in the EBAN, where the process takes $O(|J| \lg \mathcal{L})$. Then the EBCEM algorithm will send the constructed blockchain results by employing $O(|I|+|J|)$ time, which takes $O(|J| \lg \mathcal{L}+|I|+|J|)$, where $|I| \leq |J| \leq n$. Therefore, the total time complexity of the EBCEM algorithm is as follows:

$$
\begin{aligned}
&(\lfloor \frac{\omega}{l} \rfloor \lg k + (k-1)^2 + k - 3 + |J| \lg \mathcal{L} + |I| + |J|) \\
&\quad \leq O(L' + L'^2 - L' - 2 + n + n + n) \\
&\quad \leq O(L'^2 + 3n)
\end{aligned} \tag{35}
$$

Note that the complexity of the EBCEM algorithm is linearly related to the number of COVID-19 detection devices in EBAN n and the number of split blockchain requirement packets L'; the time complexity is much lower if the values of n and L' are very smaller. Therefore, the EBCEM algorithm is time-efficient due to the lower time complexity in the EBAN.

VII. PERFORMANCE EVALUATION

In this section, we evaluate the edge blockchain construction efficiency maximization (EBCEM) algorithm by considering the different influencing factors using NS-2 [28]. Specifically, we first set the simulation parameters according to the actual COVID-19 detection scenario. Then we compare the performance of the EBCEM algorithm with the lack of data transmission bandwidth, storage capacity, and energy consumption factors in the EBAN. Finally, we compare in detail the efficiency of the EBCEM algorithm and the assignment game algorithm (AGA) in the same simulation environment.

Algorithm 2 Edge Blockchain Construction Efficiency Maximization (EBCEM) Algorithm

Require: Edge blockchain construction requirement ω, energy status e_{ij}, and effective storage space a_j in the j-th COVID-19 detection device, and the real-time status of the established EBAN.
Ensure: The construction mechanism with the maximal edge blockchain cost.
/*Step 1: split the edge blockchain construction requirement ω into l parts */
Send requirement ω to the current ECS;
for $i = 1$ *to* $\lfloor \frac{\omega}{l} \rfloor$ **do**
  $L'_i = X_i^{(l)} - \bigcup_{i=i+1}^{\lfloor \frac{\omega}{l} \rfloor} X_{i+1}^{(l)}$;
  if $L'_i \notin T_k$ **then**
    Place the data packet L'_i into set T_k;
  end if
  $i = i + 1$;
end for
Sort the data packet set $\{L'_i\}$ from smallest to largest (i.e., $L'_1 \leq L'_2 \leq \cdots \leq L'_{k-1}$);
/*Step 2: Generate an edge blockchain to satisfy requirement ω */
Randomly select a number α to generate a curve $F = \alpha x + L'_1$;
Pick two points from the resulting curve F (i.e., $A_{L'_1}(1) = F_1(1)$ and $A_{L'_2}(2) = F_2(2)$);
for $i = 2$ *to* $k - 1$ **do**
  Generate a polynomial curve with the previously generated points and the data packet set $\{L'_{i-2}\}$, i.e., $F_i(x) = A_{L'_i}(i) * x^i + A_{L'_{i-1}}(i-1) * x^{i-1} + \cdots + A_{L'_1}(1) * x + L'_i$;
end for
if $i \neq k - 1$ **then**
  Select $i + 1$ points as secret shares on this curve and delete the previously generated secret share;
else
  Output $k - 1$ degree polynomial (i.e., $F_{k-1}(x) = A_{k-2}(k-1) * x^{k-1} + A_{k-2}(k-2) * x^{k-2} + \cdots + A_{k-2}(1) * x + L'_{k-2}$);
end if
Select L' points as an L' piece of fragment data in the $k - 1$ degree polynomial obtained (i.e., $\mathcal{T} = (1, F_{k-1}(1))$, $\mathcal{T} = (2, F_{k-1}(2)), \cdots, \mathcal{T} = (L', F_{k-1}(n))$);
/*Step 3: Select the optimal COVID-19 detection devices and transmission paths*/
for $j = 1$ *to* $|J|$ **do**
  Find all the COVID-19 detection devices in set J to satisfy parameters b, s, and e;
  if $\mathcal{L} \geq 2$ **then**
    Select the path with the optimal edge blockchain cost $C(b^*, s^*, e^*)$;
  else
    Deliver the path to set T'_j;
  end if
  $j = j + 1$;
end for
Sends the control commands of the edge blockchain construction to the corresponding devices.

TABLE 1. Parameter setting.

Parameter	Description
Packet Transmission Rate	512 kbit/s
Timeout	864 μs
Time-slot	0.5 ms
Perception Radius	10m
Communication Distance	15m
$\varsigma_{i \to j}$	0.75
$\mathcal{U}$	0.9
α	1.5
Simulation Time	100 min

A. SIMULATION PARAMETER SETTINGS

We consider that randomly distributed requesters store the perceived COVID-19-infected sample data in different storage spaces through the edge computing servers in the EBAN. Each edge computing server is responsible for the perceptual data analysis, blockchain construction in the current edge computing system, and contact with other edge computing servers [29]. The simulation monitoring areas are set to 1000 m by 1000 m and 4000 m by 4000 m with 1000 to 2000 COVID-19 detection devices as the ECSs. Each requester is suspected of being infected with the COVID-19 virus based on the biosensor worn by himself/herself. It is assumed that the blockchain requirement issued by each requester is satisfied by sufficient providers for the requirements of the blockchain construction. Table 1 lists the simulation parameter settings in this experiment.

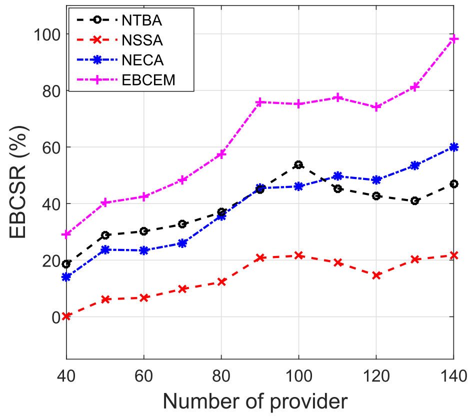
B. COMPARISON OF THE EDGE BLOCKCHAIN CONSTRUCTION SUCCESS RATIO

The edge blockchain construction success ratio (EBCSR) is the main criterion for evaluating the efficiency of the blockchain construction in the EBAN. The larger the value of *EBCSR*, the higher the success probability of edge blockchain construction, and vice versa, which effectively avoids the frequency of data retransmissions and saves valuable network resources.

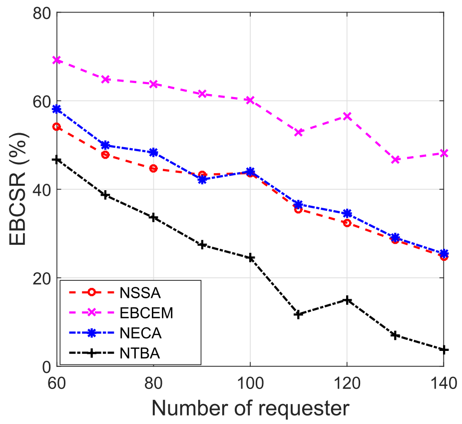
Definition 1 (Edge Blockchain Construction Success Ratio): The edge blockchain construction success ratio *EBCSR* represents the proportion between the sum of the blockchain construction requirements sent by all the COVID-19 detection devices of ECSs per unit time and the sum of all the requirements and retransmission data packet requirements in the EBAN; i.e.,

$$EBCSR = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} \omega_{ij}}{\sum_{i=1}^{I} \sum_{j=1}^{J} (\omega_{ij} + \sum_{l=1}^{L'} R_{ij}^l)} \tag{36}$$

where ω represents a valid blockchain construction request from any COVID-19 edge sensing device, and R_{ij}^l denotes l data packets that need to be retransmitted to the j-th provider



(a) The different number of providers.



(b) The different number of requesters.

FIGURE 2. Comparison of edge blockchain construction success ratio (EBCSR).

because the unsuccessful blockchain construction originates from the i-th requester.

The EBCEM algorithm comprehensively considers three factors: the bandwidth of the blockchain construction requirement b, the storage space of provider s, and the energy consumption e. We divide the EBCEM algorithm into ones that do not consider one of these factors; that is, the no transmission bandwidth algorithm (NTBA), the no storage space algorithm (NSSA), and the no energy consumption algorithm (NECA). Then we evaluate the performance of the four algorithms under the same EBAN environments, with the comparison results shown in Figure 2(a) and Figure 2(b). In addition, we use the two factors of different requesters and providers to evaluate the influence on the efficiency of blockchain construction.

Figure 2(a) shows that the efficiency of the blockchain construction continuously improves as the number of providers continues to increase in the EBAN. The construction performance of the EBCEM algorithm is better than that of the other three algorithms, which verifies that the bandwidth b, the storage space s, and the energy consumption e are the indispensable key factors of the EBCEM algorithm. The NTBA and NECA have similar blockchain construction levels, where the performance of the NSSA is relatively lower. The reason is that the limited storage space of the COVID-19 detection devices in the EBAN will make the blockchain construction fail, thereby increasing the retransmission numbers of the requested data packets.

Figure 2(b) shows that an increase in the number of requesters will reduce the blockchain construction efficiency in the EBAN. The reason is that an excessive number of requesters will increase the computational load of the edge computing server as well as the transmission load of the EBAN. This will cause the phenomenon where the same provider satisfies different requesters at the same time, which results in the retransmission of requirements caused by data conflicts. In addition, the NTBA declines faster than the other three algorithms because the multiple requesters send blockchain construction requirements at the same time, which will cause congestion during data transmission and result in a large number of blockchain requirement retransmissions in the EBAN.

C. COMPARISON OF THE NETWORK STABILITY EFFICIENCY AND THROUGHPUT

In this subsection, we first set the experimental conditions of the network stability efficiency (NSE) and throughput and then compare and analyse the performance of the EBCEM algorithm, NTBA, NECA, and NSSA on the NSE and throughput, respectively.

We set 8 buffers in each edge computing server. When multiple requesters send blockchain requirements to the edge computing server of the current edge computing system, the edge blockchain is constructed in turn according to the first-in, first-out (FIFO) principle in the EBAN. People who are suspected of being infected with the COVID-19 virus need to be verified by reading the blockchain that has been constructed. Network instability will then occur when the buffers in the EBAN are congested.

Definition 2 (Network Stability Efficiency): The network stability efficiency *NSE* indicates the bearing status of the blockchain construction requirement in the buffers of the edge computing server in the EBAN. The NSE reflects that the network builds the equilibrium level of the edge blockchain. The larger the NSE is, the more stable the network operation status is, and vice versa.

Figure 3(a) illustrates how the NSE continues to grow as the uptime increases in the EBAN. Obviously the NSE of the EBCEM algorithm quickly increased to nearly 100% in the shortest time period, which shows that the EBCEM algorithm can quickly process the blockchain construction requirement sent by the requesters so that the network operation status will stabilize as quickly as possible. In addition, the NSSA can also achieve a better stability efficiency because it can perceive the effective storage space of the providers in advance, thereby effectively allocating the limited buffers. The stability of the NTBA and NECA is obviously inferior to that of the EBCEM algorithm and NSSA. The reason for this is because the NTBA and NECA only consider the data transmission bandwidth and the energy consumption,

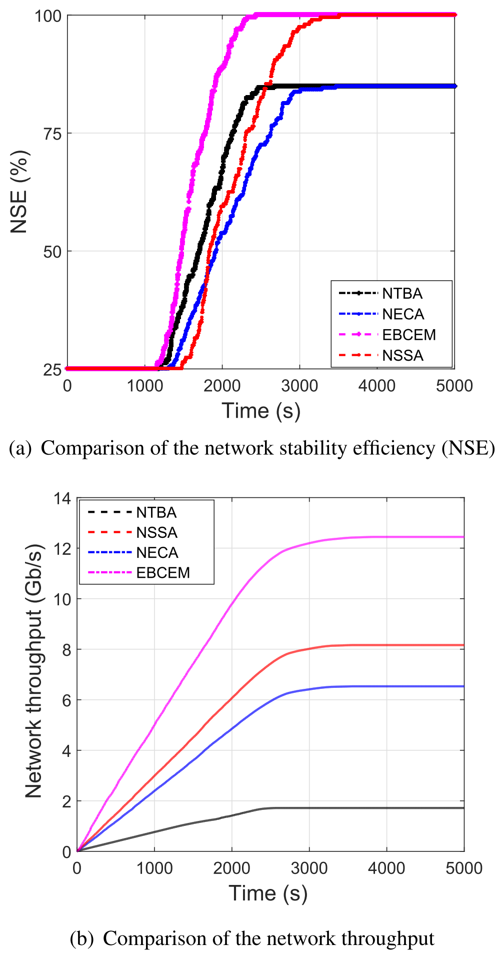


FIGURE 3. Comparison of the network stability efficiency (NSE) and throughput.

which cannot effectively schedule the buffer space; thus, the blockchain construction requirements cannot be responded to by the edge computing server for a long time in the EBAN. Therefore, the network status will be extremely unstable. In summary, the network stability efficiency of the EBCEM algorithm can be improved by considering the effective storage space factor of the EBAN so that the network resources can be balanced, and the network life cycle can be extended.

Network throughput is an important criterion of network efficiency when building a blockchain in the EBAN. The higher its value is, the faster the network can build the blockchain per unit time, and vice versa. Figure 3(b) shows that the EBCEM algorithm can quickly improve the overall network throughput after a period of time. The reason is that the EBCEM algorithm comprehensively considers key network factors (i.e., the bandwidth, storage space and energy consumption) in the EBAN so that the blockchain can be effectively constructed and allocated under the scheduling of the edge computing servers. Among them, the performance of the NTBA is relatively slower than that of the other three algorithms. The reason for this is that the network cannot effectively schedule the bandwidth allocation. Therefore, the multiple blockchain requirements are transmitted on a transmission link at the same time, which causes network congestion and reduces the overall network throughput in the EBAN.

D. COMPARISON OF THE AGA AND THE EBCEM ALGORITHM

The assignment game algorithm (AGA) is a blockchain construction strategy based on the Internet of Vehicles (IoV). Experiments show that the AGA method can effectively build a blockchain based on the perception of vehicle operation information. We compare the blockchain construction efficiency of the AGA and our EBCEM algorithm in terms of four aspects: the effective requester ratio (ERR), the effective provider ratio (EPR), EBCSR, and average storage usage ratio (ASUR).

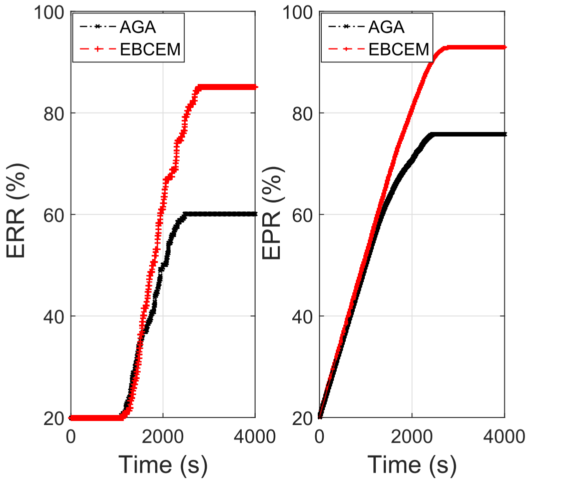
Definition 3 (Effective Requester Ratio): The effective requester ratio *ERR* represents the proportion between the number of immediate responses to the blockchain construction requirements sent by the requester and the total requirements in the EBAN.

Definition 4 (Effective Provider Ratio): The effective provider ratio *EPR* indicates that the proportion between the number of providers that can immediately satisfy the scheduling requirements of the current edge computing server for building blockchains per unit time and the total number of providers in the EBAN.

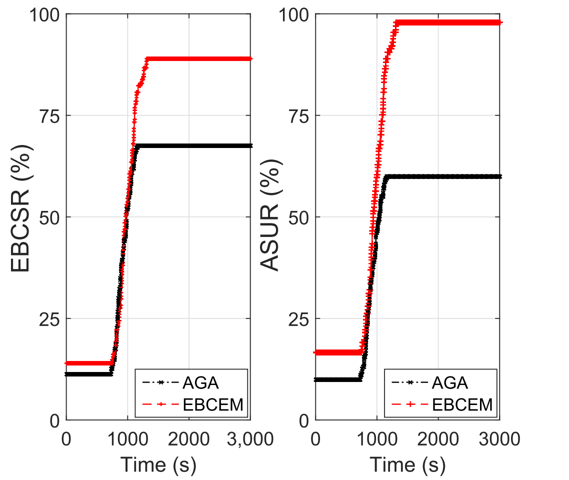
The ERR and EPR are the metrics of the operating efficiency of COVID-19 detection devices that satisfy the needs of blockchain construction in the EBAN. The larger the ERR and EPR are, the more valid the requesters and providers are, and vice versa.

Figure 4(a) demonstrates the efficiency comparison of the AGA and the EBCEM algorithm in terms of the ERR and EPR. The EBCEM algorithm has a better effective scheduling capability for the COVID-19 detection devices of ECSs than the AGA in the EBAN. In the ERR comparison, both the AGA and the EBCEM algorithm have an adjustment phase. At the beginning of the experiment, both algorithms need to adjust the number of blockchain construction requirements sent by the requesters and adjust them through the buffers of the current edge computing server in the EBAN. The execution process of the AGA and the EBCEM algorithm requires some time. In addition, the EBCEM algorithm has a higher ERR value than the AGA, which shows that the EBCEM algorithm can satisfy more blockchain construction requirements. In the performance comparison of the EPR, neither of these two algorithms has an adjustment phase, which are directly improved upon in the stable phase. The reason is that the providers only send their own resource information to the current edge computing server and then wait and save the completed blockchain sent by the edge computing server in the EBAN. Moreover, the EBCEM algorithm can schedule more effective providers than the AGA, which ensures that the EBCEM algorithm has sufficient providers to satisfy the edge blockchain construction requirements of the EBAN.

Definition 5 (Average Storage Usage Ratio): The average storage usage ratio *ASUR* represents the proportion between



(a) Comparison of edge blockchain construction success ratio (EBCSR).



(b) Comparison of average storage usage ratio (ASUR).

FIGURE 4. Comparison of different algorithms.

the number of effective storage spaces that satisfy the blockchain construction requirements sent by the requesters and the total storage space in the EBAN; i.e.,

$$ASUR = \frac{|J|\bar{S}_j}{\sum_{i=1}^{I}\sum_{j=1}^{J} s_{ij}} \tag{37}$$

where $\bar{S}$ indicates the average storage space that each edge computing system can respond to in terms of the blockchain construction requirement in a timely manner.

Figure 4(b) shows experiments and compares the performance of the AGA and the EBCEM algorithm on the EBCSR and ASUR. Obviously the EBCEM algorithm shows a relatively better operating efficiency than the AGA in the EBAN. In the comparison of the EBCSR, both the AGA and the EBCEM algorithm have a relatively good transmission efficiency of effective data packets at the initial stage. However, the overall operation efficiency of the EBCEM algorithm is relatively higher than that of the AGA. The reason for this is that the EBCEM algorithm can remove redundant and irrelevant COVID-19 detection data by calculating the edge computing server in the EBAN, which improves the construction efficiency of the edge blockchain and correspondingly transmits the more effective data packets in the EBAN. In addition, the ASUR execution efficiency of the EBCEM algorithm can be close to 100% in the ASUR experiment, which shows that the EBCEM algorithm can balance the storage space scheduling of providers. The EBCEM algorithm has a better storage space scheduling capability than the AGA, because the EBCEM algorithm can sense the remaining storage status of all providers in the EBAN in advance and can avoid the failure of the COVID-19 detection devices of ECSs due to storage space exhaustion, thereby prolonging the life cycle of the EBAN.

VIII. CONCLUSION AND FUTURE WORK

In preventing the COVID-19 virus, the greatest challenge is to effectively sense the information and protect the personal privacy of the infected people. Traditional detection systems cannot guarantee the reliable transmission of sensed data or the effective establishment of a blockchain. However, this first-established EBAN can transmit the perceptual data needed to build the blockchain to the current edge computing server in real time. Then the edge computing server removes redundant information based on the saved blockchain data in wearable biosensors and COVID-19 detection devices and saves the built blockchain to the EBUs in reliable storage nodes based on the real-time status of the EBAN. Finally, a series of experiments prove that our proposed EBCEM algorithm has a relatively higher efficiency in constructing a blockchain.

Although our proposed EBCEM algorithm shows a good performance in the transmission and storage performance of a COVID-19-based blockchain, our work did not study reading the blockchain in the EBAN. Therefore, in future work we will consider the key influencing parameters of the EBAN when legal users and illegal users read the COVID-19-based blockchain and then specify the optimal blockchain read scheduling strategy to save network resource consumption to the greatest extent.

REFERENCES

[1] L. Li, Q. Zhang, X. Wang, J. Zhang, T. Wang, T. Gao, W. Duan, K. K. Tsoi, and F. Wang, "Characterizing the propagation of situational information in social media during COVID-19 epidemic: A case study on Weibo," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 2, pp. 556–562, Apr. 2020.
[2] X. Wang, Y. Zhang, V. C. M. Leung, N. Guizani, and T. Jiang, "D2D big data: Content deliveries over wireless device-to-device sharing in large-scale mobile networks," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 32–38, Feb. 2018.
[3] J. Fischer, J.-P. Redlich, J. Zschau, C. Milkereit, M. Picozzi, K. Fleming, M. Brumbulli, B. Lichtblau, and I. Eveslage, "A wireless mesh sensing network for early warning," *J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 538–547, Mar. 2012.
[4] G. Zhu, J. Li, Z. Meng, Y. Yu, Y. Li, X. Tang, Y. Dong, G. Sun, R. Zhou, H. Wang, K. Wang, and W. Huang, "Learning from large-scale wearable device data for predicting the epidemic trend of COVID-19," *Discrete Dyn. Nature Soc.*, vol. 2020, pp. 1–8, May 2020.
[5] L. Chen, C. Shen, P. Zhou, and J. Xu, "Collaborative service placement for edge computing in dense small cell networks," *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 377–390, Feb. 2021.

[6] X. Hou, Z. Ren, J. Wang, W. Cheng, Y. Ren, K. Chen, and H. Zhang, "Reliable computation offloading for edge-computing-enabled software-defined IoV," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7097–7111, Aug. 2020.
[7] S. Guo, J. Liu, Y. Yang, B. Xiao, and Z. Li, "Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 319–333, Feb. 2019.
[8] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
[9] G. Li and Y. Xu, "Energy consumption averaging and minimization for the software defined wireless sensor networks with edge computing," *IEEE Access*, vol. 7, pp. 173086–173097, 2019.
[10] C. Chakraborty and A. Abougreen, "Intelligent Internet of Things and advanced machine learning techniques for COVID-19," *EAI Endorsed Trans. Pervasive Health Technol.*, vol. 7, no. 26, Jul. 2018, Art. no. 168505.
[11] M. Han, Y. Feng, X. Zhao, C. Sun, F. Hong, and C. Liu, "A convolutional neural network using surface data to predict subsurface temperatures in the Pacific Ocean," *IEEE Access*, vol. 7, pp. 172816–172829, 2019.
[12] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.
[13] G. Li, S. Guo, Y. Yang, and Y. Yang, "Traffic load minimization in software defined wireless sensor networks," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1370–1378, Jun. 2018.
[14] G. Liang and S. R. Weller, "Distributed blockchain-based data protection framework for modern power systems against cyber attacks," *IEEE Trans. Smart Grid*, vol. 46, no. 7, pp. 358–372, May 2018.
[15] J. Zou, B. Ye, L. Qu, W. Yan, and L. Lei, "A proof-of-trust consensus protocol for enhancing accountability in crowdsourcing services," *IEEE Trans. Services Comput.*, vol. 26, no. 8, pp. 546–559, Apr. 2018.
[16] D. E. Kouicem, A. Bouabdallah, and H. Lakhlef, "Internet of Things security: A top-down survey," *Comput. Netw.*, vol. 141, no. 9, pp. 631–643, 2018.
[17] C. Chen, C. Wang, T. Qiu, N. Lv, and Q. Pei, "A secure content sharing scheme based on blockchain in vehicular named data networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3278–3289, May 2020.
[18] L. Garg, E. Chukwu, N. Nasser, C. Chakraborty, and G. Garg, "Anonymity preserving IoT-based COVID-19 and other infectious disease contact tracing model," *IEEE Access*, vol. 8, pp. 159402–159414, 2020.
[19] J. Karjalainen and M. Viitasalo, "Fever and cardiac rhythm," *Arch. Internal Med.*, vol. 146, no. 6, pp. 1169–1171, Jun. 1986.
[20] L. Faust, K. Feldman, S. M. Mattingly, D. Hachen, and N. V. Chawla, "Deviations from normal bedtimes are associated with short-term increases in resting heart rate," *NPJ Digit. Med.*, vol. 3, no. 1, pp. 1–9, Dec. 2020.
[21] J. Qiu, D. Grace, G. Ding, J. Yao, and Q. Wu, "Blockchain-based secure spectrum trading for unmanned-aerial-vehicle-assisted cellular networks: An operator's perspective," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 451–466, Jan. 2020.
[22] J. H. Zheng, T. Y. Ji, M. S. Li, Q. H. Wu, and P. Z. Wu, "Constrained optimization applying decomposed unlimited point method based on KKT condition," in *Proc. Comput. Sci. Electron. Eng. Conf.*, 2013, vol. 6, no. 5, pp. 87–91.
[23] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.
[24] S. Verma, Y. Kawamoto, and N. Kato, "Energy-efficient group paging mechanism for QoS constrained mobile IoT devices over LTE-A pro networks under 5G," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9187–9199, Oct. 2019.
[25] Z. Kuang, L. Li, J. Gao, L. Zhao, and A. Liu, "Partial offloading scheduling and power allocation for mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6774–6785, Aug. 2019.
[26] X. Yu, J. Chu, K. Yu, T. Teng, and N. Li, "Energy-efficiency optimization for IoT-distributed antenna systems with SWIPT over composite fading channels," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 197–207, Jan. 2020.
[27] Y. Huang, M. Liu, and L. Yuan, "Energy-efficient SWIPT in IoT distributed antenna systems," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2646–2656, Aug. 2018.
[28] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*, vol. 3, no. 7. Berlin, Germany: Springer, 2012, pp. 21–40.
[29] D. Liu, Z. Cao, Y. Zhang, and M. Hou, "Achieving accurate and real-time link estimation for low power wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2096–2109, Aug. 2017.

GUOZHI LI was born in Xinyang, Henan, China, in 1978. He received the M.S. degree in system analysis and integration from Chongqing Normal University, Chongqing, China, in 2010, and the Ph.D. degree in applied mathematics from Southwest University, Chongqing, in 2018.

Since 2019, he has been a Lecturer with the Institute of Information and Technology, Henan University of Chinese Medicine. He has authored 11 articles, ten research projects, and two inventions. His current research interests include edge computing and wireless sensor networks.

XIAOFEI LI was born in Lushan, Henan, China, in 1979. He received the M.S. degree in physical chemistry from Zhengzhou University, Zhengzhou, China, in 2004.

Since 2012, he has been an Associate Professor with the Institute of pharmacy, Henan University of Chinese Medicine. He has authored eight articles and two research projects. His current research interests include edge computing and computational chemistry.

XUEKUN SONG (Member, IEEE) was born in Shangqiu, Henan, China, in 1982. He received the M.S. degree in computer science and technology from Guangxi University, Nanning, China, in 2007, and the Ph.D. degree in biomedical engineering from Harbin Medical University, Harbin, China, in 2017.

Since 2015, he has been an Assistant Professor with the Institute of Information and Technology, Henan University of Chinese Medicine. He has authored one book, 13 articles, ten research projects, and three inventions. His research interests include computational bioinformatics, evolutionary computation, and medical AI.

YUE ZENG received the M.S. degree from the Department of Electronic Information Engineering, Southwest University, Chongqing, China, in 2019. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Nanjing University, China.

He has published three articles. His research interests include network functions virtualization, software defined networking, machine learning for networking, distributed computing, and edge computing.