electronics

MDPI

Article

Federated Learning-Based Resource Management with Blockchain Trust Assurance in Smart IoT

Xiuhua Fu [1,*], Rongqun Peng [1,2], Wenhao Yuan [1,2,*], Tian Ding [1], Zhe Zhang [1], Peng Yu [2] and Michel Kadoch [3]

[1] School of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China
[2] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[3] Department of Electrical Engineering, ETS, University of Quebec, Montreal, QC H3C 3J7, Canada
* Correspondence: xhfu@sdut.edu.cn (X.F.); yuanwenhao@sdut.edu.cn (W.Y.)

Abstract: Resource management is a key issue that needs to be addressed in the future smart Internet of Things (IoT). This paper focuses on a Federated Learning (FL)-based resource management mechanism in IoT. It incorporates blockchain technology to guarantee the security of the FL model parameters exchange. We propose an IoT resource management framework incorporating blockchain and federated learning technologies; then, a specific FL-based resource management with a blockchain trust assurance algorithm is given. We use a Support Vector Machine (SVM) classifier to detect malicious nodes in order to avoid the impact on the performance of the FL-based algorithm. Finally, we perform simulation to verify the SVM classification effect and the proposed algorithm performance. The results show that the SVM-based malicious node identification accuracy can be acceptable. Moreover, the proposed algorithm obtains better performance when malicious nodes are excluded from the FL selected participant.

Keywords: Internet of Things (IoT); federated learning (FL); blockchain; resource management; malicious nodes

Citation: Fu, X.; Peng, R.; Yuan, W.; Ding, T.; Zhang, Z.; Yu, P.; Kadoch, M. Federated Learning-Based Resource Management with Blockchain Trust Assurance in Smart IoT. *Electronics* **2023**, *12*, 1034. https://doi.org/10.3390/electronics12041034

Academic Editor: Hung-Yu Chien

Received: 17 January 2023
Revised: 16 February 2023
Accepted: 17 February 2023
Published: 19 February 2023

1. Introduction

To provide seamless global three-dimensional (3D) coverage as well as to ensure the continuity of various services, the 6th generation mobile communication (6G) will integrate various terrestrial and non-terrestrial networks. There is a broad consensus that building a space-air-ground integrated network (SAGIN) will be a fundamental task of 6G [1]. SAGIN will strongly contribute to the rapid development of Internet of Things (IoT). Future IoT applications will have larger data volumes and more traffic types, which require more computing power, more storage capacity, and larger transmission bandwidth [2]. As a result, these innovative IoT applications have stringent requirements for storage (e.g., distributed storage), communication (e.g., higher transmission rates), and computation (e.g., real-time data analysis). In addition, with the maturity of artificial intelligence (AI) technology, intelligence is bound to be the evolutionary direction of IoT [1–3].

The extreme pursuit for 6G performance and the stringent requirements of diverse applications are the two main drivers for efficient resource management in IoT. As more advanced computer technologies such as cloud computing, edge computing, and AI increasingly converge with communication networks, substantial computing power and storage capabilities are being introduced into communication networks. 6G will be the first truly converged Internet and Communications Technology (ICT) system. This means that communication resources and computer resources are equally important in 6G, and need to be managed in a unified manner. In addition, other resources, such as energy, should also be included in the unified resource management.

Federated Learning (FL), as an emerging distributed machine learning (ML) technology, has been introduced into the study of network resource management [4–6]. Unlike

the traditional centralized mechanism, in FL, the local model computation is performed at the node that generates the data and only the updated model parameters are uploaded. These local model parameters are aggregated in the FL server for generating the global model [7,8]. However, all IoT nodes involved in FL training should be trustworthy. Otherwise, it will seriously affect the reliability and validity of the aggregated global model. IoT nodes may be deployed in harsh or remote areas. Coupled with the fact that the IoT topology often changes dynamically, the network often has difficulty with identifying the inclusion of malicious nodes [9]. According to the "2020 IoT Threat Report" published by PaloAltoNetworks' security team Unit 42, 57% of IoT devices are vulnerable to medium- or high-severity attacks [10]. Therefore, it is crucial to identify the malicious nodes to avoid the bad effects on FL.

Blockchain is a distributed database technology for storing transaction records. It uses some kind of consensus algorithm to reach a consensus among different nodes. It also enables non-repudiation among blocks so that none of the nodes can deny their actions [11]. IoT nodes can be deployed in blockchains. A safe and reliable service guarantee for the model parameters exchange can then be provided. In addition, IoT companies are exploring the applications of "IoT + blockchains". In 2020, Cat.1 blockchain module-L610 was first launched in China, which is the world's leading developed based on the IoT chips platform and can be loaded into IoT devices. Future IoT devices that integrate the next-generation IoT modules and blockchain modules will have more powerful capabilities [12].

In recent years, using FL for wireless network performance optimization and using blockchain to improve privacy protection in IoT have become two research hotspots. Refs. [4,13,14] conducted a comprehensive survey on the application of FL techniques in wireless networks and analyzed various issues, challenges, and research directions. However, they did not investigate specific FL algorithms. In [15], FL was used to solve the joint power and radio resource allocation problem for ultra-reliable and low-latency communication in vehicular networks. However, the reliability of FL training and transmission was not considered in [15]. The authors in [16] studied specific scheduling strategies and analyzed the impact of transmission errors on FL performance, but they did not perform FL participant selection to optimize the performance of the resource allocation algorithm. In most FL studies, the cloud servers in the core network are used as FL servers, while the FL local nodes are at the edge of the network. With the advent of mobile edge computing and the enhancement of terminal capabilities, FL migrates to the network edge and terminals [6,13,17]. A client-edge-cloud hierarchical FL framework was proposed in [17]. A collaborative FL (CFL) mechanism was proposed in [6], which enables the edge devices to implement FL when they cannot communicate with the FL server for some reasons. Although FL is distributed, its central controller can pose security issues. Blockchain was applied to improve the security and non-repudiation of the FL mechanism in [18–23]. IEEE published the guide "an Architectural Framework for Blockchain-based Federated Machine Learning" in late 2021, which defined the types of blockchain-based federated machine learning, as well as application scenarios and other aspects from a high-level perspective [18]. In [19], malicious nodes were identified before participating in FL training. In [20], a Blockchain Federated Learning architecture (BlockFL) was proposed, and the end-to-end latency model of BlockFL was analyzed. In [21], a fine-grained FL mechanism with a blockchain-based reputation guarantee was proposed to achieve trustworthy FL training in edge networks. In [22], a blockchain-assisted decentralized FL framework was proposed to solve the single point failure problem of FL servers. Although privacy and resource allocation issues were further investigated in [22], no specific resource allocation algorithm was given. In the blockchain-based FL framework proposed in [23], the FL server incentivized the participating nodes based on the accuracy of their uploaded model parameters to improve the accuracy of the classification problem. As the further development of joint FL and blockchain, a new concept, Swarm Learning (SL), was proposed in [24]. Unlike FL, SL is a new distributed ML technique with no central controller. In [24], SL was used to detect patients with severe diseases quickly and reliably. However, SL nodes are expected

to be independent and homogeneous, i.e., they should preferably have similar capabilities. Therefore, SL may not be particularly suitable for heterogeneous IoT end devices.

We first analyze the key challenges of resource management in the future smart IoT, which motivates the research of this paper. Considering the heterogeneity of IoT end-device capabilities, we use FL for distributed intelligent management of IoT resources. In this approach, the central FL server aggregates the local model parameters from the end devices with different capabilities; then, a reliable global model can be quickly obtained and be shared among all FL nodes. To ensure the reliability of FL model parameters exchange, the FL nodes are deployed in the blockchain. In addition, we optimize FL training participant selection by performing detection on malicious nodes. The main contribution of this paper is an in-depth study of the FL-based resource management mechanism with blockchain trust assurance, and a specific resource management algorithm is given.

The rest is organized as follows: Section 2 investigates the challenges of resource management in future smart IoT. Section 3 analyzes the federated learning-based resource management mechanism with blockchain trust assurance. Section 4 proposes a specific resource management algorithm. Section 5 performs the simulation analysis. Section 6 concludes this paper.

2. Challenges for Resource Management in the Future IoT Based on 6G SAGIN

In order to achieve efficient resource management in the future smart IoT based on 6G SAGIN, a clear analysis of the major challenges is the critical first step. The major challenges are discussed below from a high-level perspective.

2.1. Integrated Design Need for Communication and Computer Resources

ICT brings the need and possibility of joint management of communication and computer resources [25]. Currently, industry is shifting from the division of network and computation to their synergy, and the concept of computation network is proposed [26]. The future smart IoT needs to achieve collaborative scheduling and efficient sharing of communication and computer resources. It will improve the efficiency of the resources utilization and meet the performance requirements of different IoT applications.

The management of communication resources and computer resources should synergize with each other. On the one hand, with the abundant supply of computer resources and the support of powerful AI/ML algorithms, the 6G network will achieve rapid development. On the other hand, with the advancement of 6G, a large amount of computer resources will be fully and effectively utilized through the networked connection.

2.2. Representation of Multi-Dimensional and Multi-Domain Heterogeneous Resources

In future networks, the objects of communication resources management will have new characteristics, such as higher frequency bands, wider bandwidths, and more 3D beams. In the ICT network, various resources in computers will also be unified with communication resources, including CPUs, GPUs, and various storage resources (such as memory, external memory, etc.).

In addition, energy will be an important resource that needs to be managed in the future IoT. Although individual IoT devices consume less energy, the total energy consumption of massive IoT devices is staggering. The power supply and energy storage volume of heterogeneous IoT devices are also very different. For example, there are various power supply methods such as batteries, lithium batteries, and solar panels.

The management resources in different domains may also differ, and the system metrics they are more concerned with may also be different. The various resources in different domains need to be managed and scheduled in a unified manner, which firstly requires solving the problem of representing these heterogeneous resources. Under this premise, it can be considered in a balanced compromise.

2.3. Localized Learning and Distributed Interaction

Future IoT end devices, including various sensors, cameras, and user devices, have increasing computing power, storage capacity, and network connectivity. These devices will generate or store more data, which will be used to support more powerful and intelligent IoT applications. Locally processing these data is important to protect data privacy and accelerate application processing time. In addition, local processing is useful to greatly ease the transmission pressure. In addition, some applications may require localized learning, such as automated intelligent driving of vehicles. The intelligent control system needs to learn more about the environment around the moving vehicle, which is not very relevant to the environment elsewhere.

However, in general, localized learning is only an auxiliary mechanism for future IoT resource management. The future smart IoT should have large-scale distributed training and localized learning capabilities. Real-time distributed interactions are needed between local devices and controllers to exchange resource management information. Integrated management and unified scheduling mechanism can achieve optimal resource allocation.

2.4. Dynamic Change of Network Environment

In the future 6G SAGIN-based IoT, the network environment will be highly dynamic and change. This dynamic change is reflected in several aspects. First, different IoT devices have different dynamic characteristics. Some end-devices may be in fast motion, such as cars, logistics end-users, etc. Second, ground base stations are more densely deployed and more diverse, so users may encounter more frequent network handoff in the communication process. Third, the movement of communication satellites is very fast; a satellite typically serves users for only a few minutes, which can lead to more frequent beam switching. Fourth, low-flying Unmanned Aerial Vehicles (UAVs), whether as airborne base stations or as airborne IoT users, will bring dynamic changes in the network environment. In addition, the expansion of traditional terrestrial wireless channels into the air also enhances the dynamics of wireless communication links, while the time-varying characteristics of the channels shift to time–space variations.

2.5. Trustworthiness Issues of IoT Nodes

The lack of effective data security protection has been a major issue for IoT technologies. For example, existing transportation and shipping management systems are suffering from serious security and privacy issues [27]. In traditional IoT, the centralized security authentication mechanism does not handle well the security and privacy of IoT data that are distributed everywhere. Each IoT node specifying its precise location information is an important prerequisite for a trustworthy IoT. However, it is difficult to find the precise location of these IoT nodes because they are often densely deployed deep in industrial buildings, or in harsh and remote areas, or due to energy constraints. Intruders may also attack IoT nodes and bring about trust issues among IoT nodes by broadcasting wrong information. For trusted IoT, these nodes then become malicious nodes, and they can seriously affect the network performance.

The trustworthiness issue of IoT also includes non-repudiation between service providers and IoT nodes. If a node provides malicious or useless (non-compliant) services to other nodes, it should be able to be identified and not be able to repudiate its actions [9].

3. Federated Learning-Based Resource Management with Blockchain Trust Assurance Mechanism

3.1. Federated Learning for Resource Management

Traditionally, computing tasks for various applications are performed in cloud servers in the core network. This centralized approach is simple to manage resources, but the requirement pressures on computer and communication resources are very high. In the current Mobile Edge Computing (MEC)-based 5G network, a portion of the computing

tasks are offloaded to the network edge. The benefit is that it greatly reduces the pressure on core cloud servers and also greatly relieves the transmission pressure on the communication network, while reducing the processing latency of applications [13,16]. This is an edge cloud operation approach. With the greatly enhanced computing and storage capacity of terminal devices, in the future, a portion of computing tasks will most likely be localized and processed at the devices. This localized operation method can achieve lower application processing latency and reduce the usage demand for communication resources while protecting local data privacy.

FL is a distributed ML algorithm. The generated data are directly trained locally to obtain the local models, and all local models parameters are uploaded to the FL server for global model updating, as shown in Figure 1. The updated global model is sent to the devices involved in the model training for subsequent local model updating. The interaction continues until the global model is trained with the required accuracy and reliability. Since the local data on the devices do not need to be uploaded to the network, the privacy of the local data can be ensured.

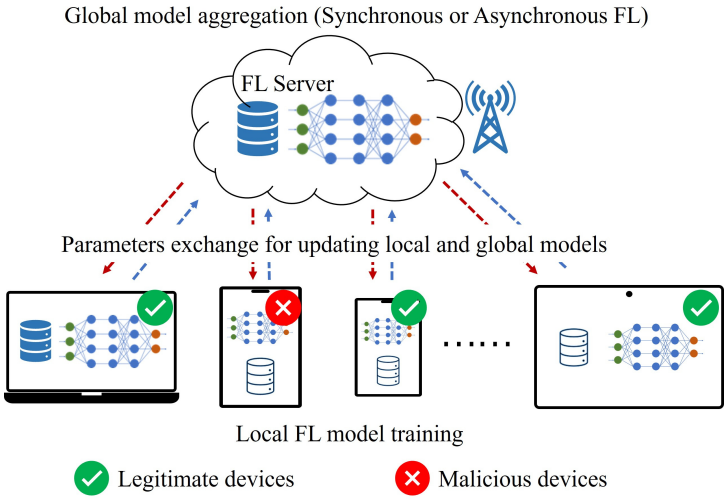

Figure 1. Federated Learning in IoT.

In Figure 1, we consider the presence of malicious nodes. If the local node participating in FL is malicious, it would obtain a wrong aggregated global model by uploading wrong local model parameters to an FL server, and this would further affect all the local nodes participating in FL training. Therefore, timely and accurate detection and removal of malicious nodes before performing FL is very important for FL-based resource management mechanisms in IoT.

3.2. Blockchain of Things

With the addition of a massive amount of IoT devices and more IoT data generation, IoT has become a big target for cyber-attacks nowadays. Blockchain, a distributed shared ledger technology, offers new technical promise for IoT security [28]. Blockchain has a strong potential to create a more secure system as it is used for recording transactions, tracking transaction updates, and establishing trust mechanisms between different nodes. In addition to its application in digital currency, blockchain technology is changing the transactions paradigm in various trustless environments. The combination of blockchain and IoT (called Blockchain of Things, BoT) will make IoT applications more powerful.

Blockchain can be applied to IoT with natural advantages. Firstly, blockchain is an open system; any IoT device can join or leave the blockchain at any time, which does not have any impact on the transactions of the whole blockchain recorded information. This is in line with the characteristics of IoT devices switching on or off at any time. Second, blockchain is a decentralized and distributed system. The nodes in the chain ensure that

their stored transaction records are consistent using consensus algorithms, which eliminates the weakness of single point of attack in traditional IoT. Third, the transactions in blockchain are secure and trustworthy. The database storing transaction records in each block can only be extended and cannot be deleted or changed, which ensures the non-repudiation of transaction records.

By using blockchain in IoT application scenarios, it is possible to identify security vulnerabilities in IoT and realize information sharing and interaction in IoT applications in a secure and effective way [19]. In addition, some IoT devices have limited resources, so it is difficult to store a complete blockchain ledger. Therefore, in BoT, IoT entities, including end devices, IoT gateways, IoT servers, etc., should all be added to the blockchain in a "decentralized" mode. Moreover, one or more blockchain nodes and "decentralized" applications (dApps) can be deployed on an IoT entity.

The exploration of industry applications of blockchain of Things started around 2015, and the typical application areas include smart city, industrial Internet, and IoT payment. At present, the integration application of blockchain and IoT has received widespread attention, and has also been rapidly developed and practiced in the industry.

3.3. Blockchain Security Assurance for Federated Learning in IoT

Security attacks and security threats are issues that must be considered in the practical deployment of the IoT FL-based resource management framework [14]. FL-based resource management faces not only a large number of heterogeneous communication and computer resources, but is also likely to face hijacked IoT nodes. In the process of FL local models and global models parameters update, IoT devices will likely face poisoning attacks or information leakage attacks, etc. The combination of blockchain technology and federated learning can provide effective security assurance for FL-based resource management in IoT [20–22,29].

Figure 2 shows the proposed IoT resource management framework incorporating blockchain and federated learning technologies in this paper. IoT end devices, IoT gateways, and edge network servers (e.g., MECs) are the physical entities for IoT resource management. In these physical entities, not only are FL training models deployed, but also blockchain nodes and "decentralized" applications. The blockchain not only ensures that the nodes involved in FL model training are secure and trustworthy, but also ensures the security of exchanging model parameters in FL. FL nodes upload their own model parameters to the blockchain and obtain the updated model parameters of other FL nodes from the blockchain. By using the security mechanism of the blockchain, the privacy and non-repudiation of model parameters are also able to be guaranteed. Blockchain consensus algorithms such as proof of work (PoW) or proof of stake (PoS) can be used to ensure the consistency of the FL models transaction records [30]. The PoW algorithm is a consensus algorithm used on the Bitcoin network. PoS is an improvement on PoW.

Figure 3 gives the data structure details of the blockchain in Figure 2. Each "block" contains a list of FL model parameters transactions. These transactions are linked together through a chain of "blocks" to form the complete transaction records during the FL model training process. In blockchain technology, transaction records are also known as transactions, which are evidence of a specific transaction action between nodes that occurred at a specific time. In the Bitcoin network, transactions record the transfer of Bitcoins. In the proposed IoT resource management framework, transactions record model parameter updates for FL local model and global model interactions. As shown in Figure 3, a block consists of a block header and a block body containing a series of transactions. The block header records the version and timestamp information of the block and the link to the previous block. The block body stores information about the FL model parameter updates. The global aggregation of the FL local models is also performed in the blockchain platform. Adding a new block to the blockchain requires confirmation by the consensus algorithm. A new node participating in the FL model training is added to the blockchain ledger only when it passes the verification of most blockchain nodes.

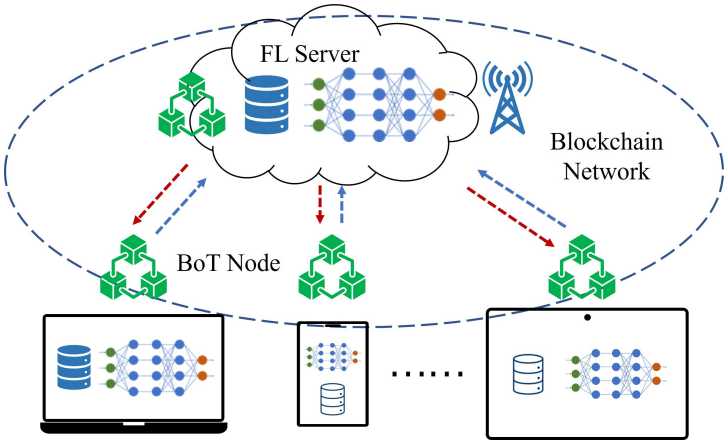

Figure 2. The proposed IoT resource management framework incorporating blockchain and federated learning technologies.

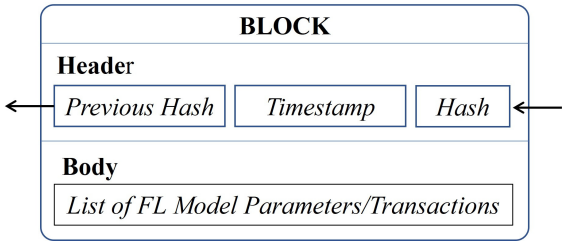

Figure 3. The data structure details of the blockchain.

4. Proposed FL-Based Resource Management with Blockchain Trust Assurance Algorithm

Currently, most IoT applications are deployed in cloud servers in core networks. As MECs are deployed on a large scale and more and more small- and medium-volume IoT applications are launched, more IoT applications will be deployed in MEC servers at network edges. Of course, this does not preclude some large IoT applications from still being deployed in the core cloud servers. In this paper, we focus on the case of IoT applications deployed in MEC servers. FL servers and FL local devices share the same training models, and FL servers are deployed in MEC servers. Different IoT applications achieve distributed collaborative resource management by invoking FL service mechanisms.

Suppose that there are N local devices involved in the FL training and there are K classes of different resources to be managed. $\text{res}_{i,j}$ denotes the size of the jth class resource of local device i, $1 \le i \le N; 1 \le j \le K$.

The normalized representation of the resources of $\text{res}_{i,j}$ is

$$r(i,j) = \frac{\text{res}_{i,j}}{\sum_{i=1}^{N} \text{res}_{i,j}}, 0 \le r(i,j) \le 1 \tag{1}$$

The total resources of participant local device i are denoted as

$$R(i,j) = \sum_{j=1}^{K} w_j \cdot r(i,j) \tag{2}$$

$R(i,j)$ is a weighted logical resource representation. w_j denotes the weight of the jth class resource in the participant selection, which satisfies

$$\sum_{j=1}^{K} w_j = 1, 0 \le w_j \le 1 \tag{3}$$

To prevent the impact of malicious nodes on the FL training, this paper identifies the malicious nodes and removes them from the network before federated training. In this

paper, malicious nodes are detected using support vector machines (SVM), which is a supervised learning technique that enables binary classification of data [9,23]. SVM finds a hyperplane (i.e., decision boundary) in N-dimensional space, and each N-dimensional data point is divided on both sides of the hyperplane, thus classifying these data points into two classes.

In this paper, the goal of SVM is to compute the binary classification of different IoT nodes (including end devices, IoT gateways, IoT servers, etc.), as shown in Equation (4):

$$C = \{(X^1, y^1), \cdots, (X^N, y^N)\}, X \in \mathbb{R}^K, y \in \{-1, +1\} \tag{4}$$

where X denotes the input data vector consisting of K classes of different resources, and y is the binary classification value:

$$X^i = [\text{res}_{i,1}, \cdots, \text{res}_{i,j}, \cdots, \text{res}_{i,K}] \tag{5}$$

X^i is the local data input of the local model FL training algorithm performed by local device i. Its size depends on the specific FL training task.

The decision boundary of the local nodes is given by Equation (6):

$$W^T X + b = 0 \tag{6}$$

where $W = (w_1, \cdots, w_K)$ is the weight vector, and $b \in \mathbb{R}$ is the threshold value, which is geometrically expressed as the intercept of this hyperplane. Hence, y in Equation (4) can be determined from Equation (7):

$$\begin{cases} W^T X^i + b \geq 0 & y^i = +1 \\ W^T X^i + b < 0 & y^i = -1 \end{cases} \tag{7}$$

The local node trustworthiness trust is defined as the probability of successful interaction of the local node with the FL server and is calculated by Equation (8):

$$\text{trust} = \frac{I_s}{I_t} \tag{8}$$

where I_t is the total number of interactions between a local node and the FL server over a period of time, and I_s is the number of successful interactions. If $y^i = -1$ or $\text{trust}^i <$ threshold, the local node i is considered as a malicious or untrustworthy node.

Since a nonlinear dynamic process requires a series of complex algorithms to be implemented [31], the linear regression FL algorithm is used in this paper in order to clearly analyze the performance of the proposed algorithm [32]. Define g_0 as the global model, g_i as the local model, and p_i as the local model parameters.

Assume that the output of the FL algorithm is z_i in local device i. The input and output of the FL algorithm follow Equation (9):

$$z = -2X + 1 + n \times 0.4 \tag{9}$$

where n follows Gaussian distribution $\mathcal{N}(0, 1)$.

In this paper, the problem of local FL training is to find an optimal model parameter p_i^* that minimizes its loss function. This problem is given by Equation (10):

$$\begin{aligned} \min \quad & \sum_{j=1}^{K} f(w_j, X^i, p_i, z_i) \\ \text{s.t.} \quad & g_1 = g_2 = \cdots = g_N = g_0 \end{aligned} \tag{10}$$

The constraint in Equation (10) states that all local FL models and the global FL model (g_0) share the same model parameters at the end of the FL training algorithm.

For different FL training tasks, different loss functions can be defined. For the FL linear regression FL algorithm in this paper, the loss function in Equation (10) can be simply defined as

$$f(w_j, X^i, p_i, z_i) = \frac{1}{2}(w_j \cdot \mathrm{res}_{i,j} \cdot p_i - z_i)^2 \tag{11}$$

In Equation (11), $w_j \cdot \mathrm{res}_{i,j} \cdot p_i$ is the predicted output. Thus, the loss function is defined as the square of the prediction error. This indicates that the value of the loss function will increase sharply as the prediction error increases. Therefore, using the loss function as the performance indicator of the FL algorithm can clearly reflect the accuracy of the FL algorithm.

The flowchart of the proposed FL-based resource management with the blockchain trust assurance algorithm is given in Figure 4. The detailed steps are as follows.

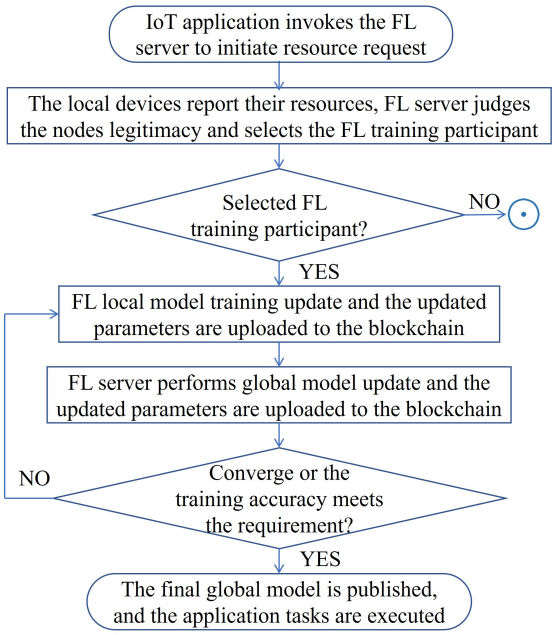

Figure 4. The flowchart of the proposed algorithm.

Step 1. The application initiates the resource request.

The IoT application deployed in the MEC server invokes the FL server to initiate a resource request to the connected local IoT devices through the edge network. These local devices will eventually take on the role of task execution.

Step 2. Resource reporting and participant selection for FL training.

The local devices that have received resource requests and agree to participate in FL training report their communication bandwidth, computing power, available storage size, data volume, and their energy status to the FL server.

The FL server normalizes the various resources for each device and then aggregates them in a weighted manner to obtain the total resource R(i) of the device. The decision boundary is calculated using the SVM mechanism, and the classification is obtained by Equation (7). The trustworthiness of the local node is calculated by Equation (8). If the FL server judges that the node is a malicious node or an untrustworthy node, the node will not be selected to participate in the FL training. This selection mechanism is simple and fast, and can reflect the degree of influence of different types of resources on the selection of FL participants by adjusting the size of the weights. Moreover, by detecting and eliminating untrustworthy nodes from participating in FL model training, the long FL model training time can be avoided and the accuracy of the global model can be guaranteed.

Step 3. FL local model training update and the updated parameters are uploaded to the blockchain.

The local devices that have received confirmation of the FL training participant selection start the FL local training update. The first-round local model update is based on the local dataset, and subsequent updates will also be based on the global model updated parameters sent by the FL server. The updated model parameters are uploaded to the blockchain after the local model training is completed, and then the FL server obtains the updated local model parameters from the blockchain.

Step 4. FL server performs a global model update, and the updated parameters are uploaded to the blockchain.

After the FL server obtains the local model updated parameters of all participant devices, it performs a global model update using the received trained local model parameters as well as application-related data. This is a synchronous update method. The global model updated parameters are uploaded to the blockchain, and then the local nodes obtain the global model updated parameters from the blockchain.

Step 5. Repeat step 3 and step 4 until the training accuracy and reliability of the global model in the FL server meet the requirements.

Step 6. The final global model is published, and the application tasks are executed.

The FL server uploads the trained final global model parameters to the blockchain and downloads it to all local devices that can communicate directly with it, not just the devices selected for participating in local model training. The local devices that cannot communicate directly with the FL server can then obtain the global model parameters from other local devices via device-to-device (D2D) method.

Note: Usually, there are two approaches for FL global model update: synchronous and asynchronous. For the case with a large number of heterogeneous IoT devices, the asynchronous global model update has high overhead and slow convergence, and the trained model may also not be accurate enough. For the synchronous approach, the FL server updates the global model only after all of the local parameters are obtained. Although the update period is longer, it is possible to converge to a stable and accurate model with fewer update iterations. In addition, it is crucial to ensure the FL training progress, i.e., all the devices involved in the FL local models training should have roughly equal time to train the local models and upload the model parameters to avoid the straggler effect as much as possible. The FL server in step 3 can ensure both that all local model update parameters are received within a guaranteed time and that there are enough local training participants to ensure the accuracy of the trained global model.

5. Simulation Results and Analysis

In this section, the proposed FL-based resource management mechanism is validated by simulation. We focus on the malicious node identification method and the FL algorithm performance for simulation analysis. In the simulation, we assume that the IoT nodes are randomly distributed in a circular area with a radius of 1000 m. There is one edge server in which the FL global model aggregation is performed. Out of the total 100 IoT nodes, there are 10 malicious and untrustworthy nodes. The main simulation parameters are listed in Table 1.

Table 1. Simulation parameters

Parameters	Values
IoT Area	1000×1000 m^2
IoT Topology	random distribution
MEC Servers	1
Total IoT Nodes	100
Legitimate Nodes	90
Malicious Nodes	10
Network Bandwidth	1000 Mbps

For the proposed FL algorithm in this paper, the input data can be normalized in advance. The FL training data in this simulation are randomly generated in $[0, 1]$. We distinguish between malicious and legitimate nodes based on the state of the managed resources. Here, it is assumed that the resources correspond to the nodes one by one. For the cases with two and three managed resources, the SVM decision bounds for a random node distribution are given in Figures 5 and 6, respectively. As can be seen from the figures, the legitimacy of the majority of IoT nodes can be correctly identified according to our identification criteria. However, since the nodes are randomly distributed, the decision boundaries shown in Figures 5 and 6 do not perfectly distinguish the two classes of nodes completely. In fact, a very small number of nodes are very close to the decision boundary, which leads to the possibility of misidentification, and this will affect the accuracy of the proposed FL algorithm.

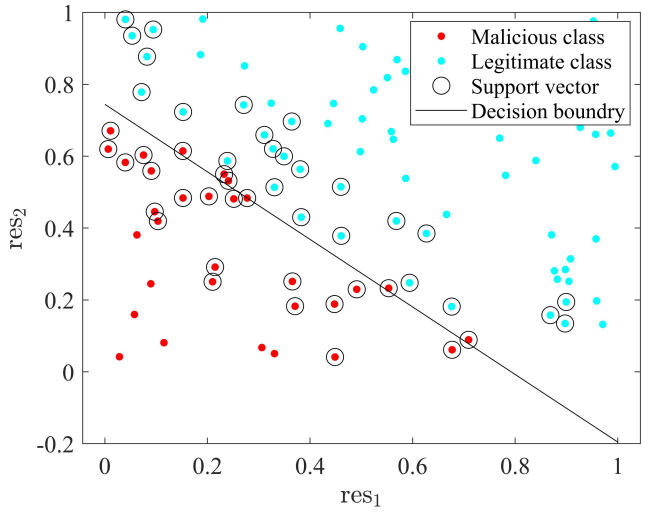

Figure 5. Decision boundary in 2D.

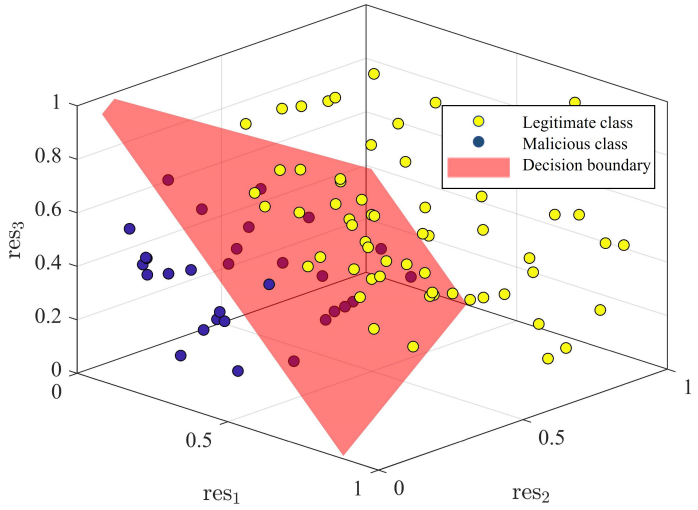

Figure 6. Decision boundary in 3D.

Figure 7 analyzes the accuracy of the proposed FL training algorithm in two cases. One is the case where only legitimate nodes participate in FL training after malicious nodes are identified and eliminated. The other is the case where all nodes (including malicious nodes) are involved in FL training. As expected, the algorithm accuracy in the first case is much higher than that in the second case. However, since not all malicious nodes can be perfectly eliminated, the algorithm accuracy is close to 90% even in the first case. We assume that 10% of the IoT nodes are malicious. It can also be seen from Figure 7 that,

when the number of IoT nodes is small, the FL algorithm accuracy is not high because fewer nodes are involved in FL training. However, as the number of IoT nodes increases, there are more legitimate nodes involved in FL training, so a more desirable FL training accuracy is obtained.

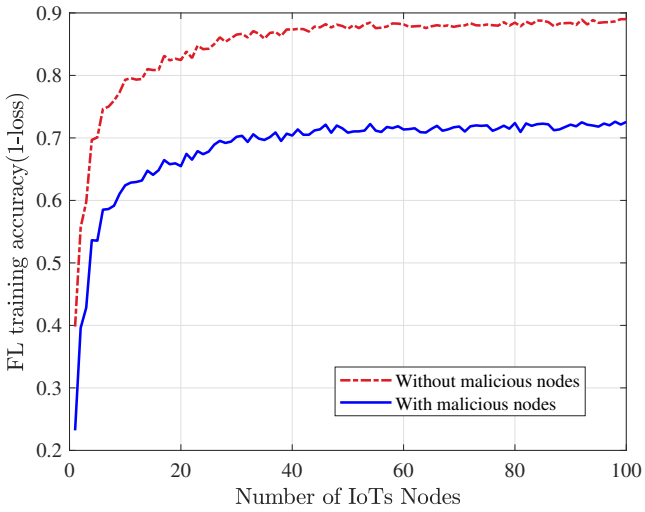

Figure 7. FL training accuracy.

6. Conclusions

This paper first investigates the key issues in resource management in the future smart IoT and starts the focus research of the later part of the paper from these issues. The focus research of this paper is the IoT resource management mechanism and the algorithm combining blockchain and federated learning techniques. To reduce the impact of adverse factors such as malicious nodes on the proposed algorithm performance, we use an SVM classifier to detect malicious nodes and exclude these malicious nodes from the FL training selected participant. Finally, we perform a preliminary simulation to validate the proposed mechanism and algorithm. In the next step, we will continue to investigate in depth the joint application of blockchain and federated learning techniques in the future smart IoT and perform a comprehensive simulation analysis of the studied algorithms on more metrics.

Author Contributions: Conceptualization and methodology, X.F. and T.D.; software, Z.Z.; validation, X.F. and W.Y.; writing—original draft preparation, X.F.; writing—review and editing, P.Y., R.P. and M.K.; funding acquisition, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Open Foundation of State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications (SKLNST-2022-1-09).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The authors have retained the analysis and simulation datasets, but the datasets are not public.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hong, T.; Lv, M.; Zheng, S.; Hong, H. Key Technologies in 6G SAGS IoT: Shape-Adaptive Antenna and Radar-Communication Integration. *IEEE Netw.* **2021**, *35*, 150–157. [CrossRef]
2. Liu, X.; Yu, J.; Wang, J.; Gao, Y. Resource Allocation with Edge Computing in IoT Networks via Machine Learning. *IEEE Internet Things J.* **2020**, *7*, 3415–3426. [CrossRef]

3. Bzai, J.; Alam, F.; Dhafer, A.; Bojović, M.; Altowaijri, S.M.; Niazi, I.K.; Mehmood, R. Machine Learning-Enabled Internet of Things (IoT): Data, Applications, and Industry Perspective. *Electronics* **2022**, *11*, 2676. [CrossRef]
4. Niknam, S.; Dhillon, H.S.; Reed, J.H. Federated Learning for Wireless Communications: Motivation, Opportunities, and Challenges. *IEEE Commun. Mag.* **2020**, *58*, 46–51. [CrossRef]
5. Wang, S.; Tuor, T.; Salonidis, T.; Leung, K.K.; Makaya, C.; He, T.; Chan, K. Adaptive Federated Learning in Resource Constrained Edge Computing Systems. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 1205–1221. [CrossRef]
6. Chen, M.; Poor, H.V.; Saad, W.; Cui, S. Wireless Communications for Collaborative Federated Learning. *IEEE Commun. Mag.* **2020**, *58*, 48–54. [CrossRef]
7. Tran, N.H.; Bao, W.; Zomaya, A.; Nguyen, M.N.H.; Hong, C.S. Federated Learning over Wireless Networks: Optimization Model Design and Analysis. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 1387–1395. ISSN 2641-9874. [CrossRef]
8. Zhao, Z.; Feng, C.; Yang, H.H.; Luo, X. Federated-Learning-Enabled Intelligent Fog Radio Access Networks: Fundamental Theory, Key Techniques, and Future Trends. *IEEE Wirel. Commun.* **2020**, *27*, 22–28. [CrossRef]
9. Abubaker, Z.; Javaid, N.; Almogren, A.; Akbar, M.; Zuair, M.; Ben-Othman, J. Blockchained service provisioning and malicious node detection via federated learning in scalable Internet of Sensor Things networks. *Comput. Netw.* **2022**, *204*, 108691. [CrossRef]
10. 2020 Unit 42 IoT Threat Report. Available online: https://unit42.paloaltonetworks.com/iot-threat-report-2020 (accessed on 10 September 2022).
11. Xu, Y.; Ren, J.; Wang, G.; Zhang, C.; Yang, J.; Zhang, Y. A Blockchain-Based Nonrepudiation Network Computing Service Scheme for Industrial IoT. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3632–3641. [CrossRef]
12. Majeed, U.; Khan, L.U.; Yaqoob, I.; Kazmi, S.M.A.; Salah, K.; Hong, C.S. Blockchain for IoT-based smart cities: Recent advances, requirements, and future challenges. *J. Netw. Comput. Appl.* **2021**, *181*, 103007. [CrossRef]
13. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated Learning in Mobile Edge Networks: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063. [CrossRef]
14. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [CrossRef]
15. Samarakoon, S.; Bennis, M.; Saad, W.; Debbah, M. Distributed Federated Learning for Ultra-Reliable Low-Latency Vehicular Communications. *IEEE Trans. Commun.* **2020**, *68*, 1146–1159. [CrossRef]
16. Yang, H.H.; Liu, Z.; Quek, T.Q.S.; Poor, H.V. Scheduling Policies for Federated Learning in Wireless Networks. *IEEE Trans. Commun.* **2020**, *68*, 317–333. [CrossRef]
17. Liu, L.; Zhang, J.; Song, S.; Letaief, K.B. Client-Edge-Cloud Hierarchical Federated Learning. In Proceedings of the ICC 2020–2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6. ISSN 1938-1883. [CrossRef]
18. IEEE P3127, Guide for an Architectural Framework for Blockchain-Based Federated Machine Learning. Available online: https://standards.ieee.org (accessed on 8 October 2022).
19. Ramasamy, L.K.; Khan, K.P.F.; Imoize, A.L.; Ogbebor, J.O.; Kadry, S.; Rho, S. Blockchain-Based Wireless Sensor Networks for Malicious Node Detection: A Survey. *IEEE Access* **2021**, *9*, 128765–128785. [CrossRef]
20. Kim, H.; Park, J.; Bennis, M.; Kim, S.L. Blockchained On-Device Federated Learning. *IEEE Commun. Lett.* **2020**, *24*, 1279–1283. [CrossRef]
21. ur Rehman, M.H.; Salah, K.; Damiani, E.; Svetinovic, D. Towards Blockchain-Based Reputation-Aware Federated Learning. In Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 6–9 July 2020; pp. 183–188. [CrossRef]
22. Ma, C.; Li, J.; Shi, L.; Ding, M.; Wang, T.; Han, Z.; Poor, H.V. When Federated Learning Meets Blockchain: A New Distributed Learning Paradigm. *IEEE Comput. Intell. Mag.* **2022**, *17*, 26–33. [CrossRef]
23. Mahmood, Z.; Jusas, V. Implementation Framework for a Blockchain-Based Federated Learning Model for Classification Problems. *Symmetry* **2021**, *13*, 1116. [CrossRef]
24. Warnat-Herresthal, S.; Schultze, H.; Shastry, K.L.; Manamohan, S.; Mukherjee, S.; Garg, V.; Sarveswara, R.; Händler, K.; Pickkers, P.; Aziz, N.A.; et al. Swarm Learning for decentralized and confidential clinical machine learning. *Nature* **2021**, *594*, 265–270. [CrossRef]
25. Talwar, S.; Himayat, N.; Nikopour, H.; Xue, F.; Wu, G.; Ilderem, V. 6G: Connectivity in the Era of Distributed Intelligence. *IEEE Commun. Mag.* **2021**, *59*, 45–50. [CrossRef]
26. Li, R.; Hirayama, T.; Xue, K.; Ruan, P.; Asaeda, H. CON: A Computation-Oriented Network for Efficient Edge Intelligence. *IEEE Netw.* **2022**, *36*, 160–166. [CrossRef]
27. Ahmad, R.W.; Hasan, H.; Jayaraman, R.; Salah, K.; Omar, M. Blockchain applications and architectures for port operations and logistics management. *Res. Transp. Bus. Manag.* **2021**, *41*, 100620. [CrossRef]
28. Si, H.; Sun, C.; Li, Y.; Qiao, H.; Shi, L. IoT information sharing security mechanism based on blockchain technology. *Future Gener. Comput. Syst.* **2019**, *101*, 1028–1040. [CrossRef]
29. Mahmood, Z.; Jusas, V. Blockchain-Enabled: Multi-Layered Security Federated Learning Platform for Preserving Data Privacy. *Electronics* **2022**, *11*, 1624. [CrossRef]
30. Auhl, Z.; Chilamkurti, N.; Alhadad, R.; Heyne, W. A Comparative Study of Consensus Mechanisms in Blockchain for IoT Networks. *Electronics* **2022**, *11*, 2694. [CrossRef]

31. Li, M.W.; Xu, D.Y.; Geng, J.; Hong, W.C. A hybrid approach for forecasting ship motion using CNN–GRU–AM and GCWOA. *Appl. Soft Comput.* **2022**, *114*, 108084. [CrossRef]
32. Chen, M.; Yang, Z.; Saad, W.; Yin, C.; Poor, H.V.; Cui, S. A Joint Learning and Communications Framework for Federated Learning Over Wireless Networks. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 269–283. [CrossRef]