Identity Management with Hybrid Blockchain Approach: A Deliberate Extension with Federated-Inverse-Reinforcement Learning

Soumya Banerjee, Samia Bouzefrane, Amar Abane

Identity Management with Hybrid Blockchain Approach: A Deliberate Extension with *Federated-Inverse-Reinforcement Learning*

Soumya Banerjee
R&D Trasna Solutions
Trasna
Arklow, Ireland
soumya.banerjee@trasna.io

Samia Bouzefrane
CEDRIC Lab
Cnam
Paris, France
samia.bouzefrane@cnam.fr

Amar Abane
EVA team
Inria
Paris, France
amar.abane@inria.fr

***Abstract*—The widespread decentralized applications and Blockchain components significantly boost the security frameworks in many vertical applications and use-cases including different secured payment methods and smart contracts. The integral part of any smart contract is the validation of the stake-holder identity, in general, while ideally being achieved without the third-party involvement. Recent industrial research works introduce the sovereign-identity system, where Blockchain becomes a decentralized component to establish a self-certified identity and to avoid a centralized trust third party. Hence, the classification of distributed transactions with respect to identity validation across several users becomes more challenging, especially because of the massive and sensitive identities that are issued through many users and IoT devices and that are used to validate transactions. In this context, it is important to identify and classify the malicious and non-malicious types of transactions. Our proposed method achieves the target of identity classifications from variety of transaction data. Since different users may have different device usage patterns, the data samples and labels located on any individual device may follow a different distribution, which cannot represent the global data distribution. Therefore, the solution could be bi-focal to compensate the gap. This paper coins the approach of hybridizing the consensus where as to initiate a machine learning mechanism to collect the local data globally through a permission driven and a federated approach. We introduce here a Federated Reinforcement learning to be improvised for distributed independent data as a policy of consortium while binding the proof of consensus more centrally authenticated.**

I. Introduction

The deployment of Blockchain technology and the subsequent influence of various machine learning components allowed the enhancement of the security with a comprehensive level. The essence of Blockchain in the foundation of network security is the creation of blocks while bitcoin is one of the prime artifacts of this ecosystem. In fact, the Blockchain paradigm involves several key technical components, including a chained data structure, a peer-to-peer network protocol, and a distributed consensus algorithm [3]–[5]. The Blockchain concept has become a prime-technical brick in FinTech [6], Internet of Things (IoT) [7] [8] and supply chains [9]. The key feature of Blockchain is that no central authority is required to certify the authenticity of the transactions since that they are assembled by peers in the network independently and in a distributed manner. In order to guarantee the consistency of the block chains that are maintained by different peers, the peers must agree on a single universal trust about the transactions through a consensus/approval building process. However, in this context, it is worthy to mention that any security application, across network models, starts from identity verification process as a standalone component. Without verifying the credentials and the identity, no system can proceed for real-time operations. The existing standard identification systems and the credential verification mechanisms are dependent on the third party verification when transactions are done remotely through the Internet. Therefore, in many practical use-cases of Blockchain, the identity verification needs trust and authentication that are preferably free from bias through the third parties. Since it is difficult to maintain trust on the third party that is generally called Identity Provider in traditional identity management systems, the decentralization of the trust based on Blockchain technology has been investigated by introducing the self-sovereign identity concept that establishes self-certified identities where each user identity is the union of her different partial identities in each of different decentralized domains.

In the backdrop, it is realized that, in many smart applications and platforms, multiple sensor networks or IoT driven cyber-physical systems (CPS) could experience the presence of multiple types of devices for which we need to verify the identity and the trustworthiness. These devices could be part of data acquisition or data modeling process, subsequently participating in network processes. Definitely, there must be a certain hybrid mechanism to establish such automated identity and trust identification by relying on the establishment of a consortium that favors Blockchain decentralized theme so that to avoid any specific bias and influence.

One such branch that is originated from blockchain

978-1-6654-4005-9/21/$31.00 ©2021 IEEE

research is the consortium-based DLT [4]. Compared to public blockchains, the most distinctive characteristic of consortium-based DLT is that only authorized nodes participate in the network [5]. This restriction enables the network to utilize selection or polling based consensus protocols such as *Practical Byzantine Fault Tolerance* (PBFT) increasing then the scalability of the network. As a result, consortium blockchains achieve faster transaction finality and low-delay verification times of transactions. The IoT ecosystem such as Internet of Vehicle (IoV) that has benefited of DLT [6], is distributed by nature and issues a great number of identities that are generated either by users or devices. To have a full control over the identity data, the concept of self-sovereign identity has emerged to self-certify different identities within different decentralized domains while the decentralized trust authority is built on a blockchain. Hence, the classification of transactions across user/device identities in distributed systems such as in cyber-physical systems becomes more challenging. In fact, the decentralized mechanism through Blockchain solicits more adaptive and intelligent algorithm to classify malicious and non-malicious types of transactions.

This paper has put a bi-focal approach: firstly, to develop a model where Blockchain theme must sustain the concept of decentralization, and secondly a federated machine learning to train the decision engine with all the pieces of device identities. The algorithm will help to approve the most sorted device to participate in Blockchain network so that the decentralized theme will be maintained and the trust of the devices across CPS will be established. Choosing the class of Inverse Reinforcement Learning allows smooth management and initial data acquisition from different devices.

The remaining part of the paper will be organized as follows: Section 2 summarizes the relevant similar works principally by describing the scope of Blockchain in identity management scenarios and their corresponding highlights. Section 3 has ideally two parts: Section 3.1 will present the background of Blockchain initiated with Reinforcement Learning including related mathematical parameters and it is followed by the proposed federated Inverse Reinforcement Learning (f-IRL) in tune of consensus to provide the theme of decentralized control. Section 4 briefly describes the results and discussion for the proposed model with certain public data set behavior. Finally, section 5 concludes the contributed model while mentioning the immediate future scope of research in this direction.

II. Related Work

Conventionally, federated learning relies on a single server which is prone to be attacked. Therefore, it is focused on relinquishing single-point failure in federated learning using blockchain [13]. Zhao et al. [14] designed a system based on federated learning and blockchain to let clients empower full nodes and compete to serve as a central server by turns for aggregation. Kim et al. [13] proposed a block chained-federated learning architecture in which participant clients are able to store local model updates on blocks and all clients as miners can access and aggregate the updates through smart contracts, without a single central server.

Existing research works expose blockchain in federated learning for auditability and data privacy [15]. Lu et al. [17] incorporated federated learning and blockchain into a data sharing scheme to preserve privacy. Majeed et al. [7] validated local model parameters in each iteration into blocks. The incentive mechanism is another niche research direction, which is combined to federated learning to motivate participants and standardize the behavior of participants [16], [18]. Weng et al. [16] deployed the incentive mechanism and blockchain transactions to protect the privacy of local gradients and enable auditability of the training process. Kang et al. [18] proposed an incentive mechanism based on blockchain combing the reputation. This is introduced to measure the trustworthiness and the reliability of clients, to stimulate honest clients with high-quality data to participate in federated learning. Moreover, there are some recently proposed platforms to increase more unique participant roles, such as buyers, who can represent the clients in order to complete their training and modeling tasks [19].

Bao et al. [19] provided a platform for buying federated learning models and designed an architecture to reduce the time cost of buyers querying and verifying models on a blockchain. However, the above studies do not present the architecture design details of blockchain-based federated learning systems with consideration of the data heterogeneity issue in Industrial IoT failure detection. Additionally, the assurance of the integrity towards the client data is not discussed.

In this paper, we propose an innovative hybrid model of federated and reinforcement learning for multiple IoT device systems using blockchain so that to classify malicious and non-malicious types of transactions provided that sufficient data and computing resources are available for model training.

III. Proposed Model at Primary Level

The identity management system relies on the identity provider to store and control the claims and identities of users. It follows three key points:

- How claims are created (Generate algorithm)?
- How claims are stored (Transform algorithm)?
- How verification is done (Verify algorithm)?

Thanks to the configuration of the identity management system algorithms (G, T, and V), we provide the following table that correlates the choices of the configurations with the type of identity management system (See Table 1).

An identity management system at the core consists of a triplet (3) of efficient randomized polynomial time algorithms : G (Generate), T (Transform) and V (Verify) such that: G : $(m, 1n) \rightarrow (c[\ ], x)$ is the claim generation algorithm that generates an array $c[\ ] \in C$ consisting of an arbitrary (m) number of claims, where $c\ [\ ] = \{c_1, c_2, c_3, ..., c_m\}$, and $x \in \{0, 1\}n$ is some secret information known only to the subject and is used to prove that the subject owns the claim. This could be a password used to protect claims like email, date of

	Generate	Transform	Verify
Centralized	Claims are generated by Identity Provider	Claims are stored by Identity Provider	No transparency: claims are controlled by Identity Provider
Federated	Claims are generated by multiple Identity Providers	Claims are stored by the respective Identity Provider	No transparency: claims are controlled by multiple centralized authorities
User-Centric	Claims may be generated by the user but practically easier to let identity providers generate them	Claims are stored by the Identity Provider	verification results in full disclosure of claims to service providers (Authorization agents)
Decentralized	Claims may be generated by the user on Identity Providers	Claims are usually stored by subjects (users)	Verification may leak information about subject, thus, full disclosure or weakly anonymized disclosure
Self-Sovereign	Claims may be generated by the user on Identity Providers	Claims are strictly stored/controlled by the user	Verification is strictly either *Zero Knowledge Disclosure* or very strongly anonymized disclosure.

TABLE I
IDENTITY MANAGEMENT WITH KEY AS SELF-SOVEREIGN AND DECENTRALIZED

birth, insurance number, etc., when registering on a website. T : T(c[ ],y) $\implies$ (c´[ ], k).

The Transform algorithm performs a cryptographic transformation on a generated set of claims c[ ] using $y \in \{0, 1\}n$, where y is some secret information known only to the identity provider. c'[] is the transformed output and $k \in \{0, 1\}n$ is the necessary information for verifying claims. For example, in the case of a website like Facebook, the Transform algorithm takes all personal data (claims) belonging to a person, c [ ], and securely stores these claims in their database. For secure storage, the Transform algorithm uses an encryption transformation with a secret key y. The result is c'[] securely stored and unintelligible to all external parties apart from the Identity Provider (Facebook) and the Subject (Facebook user). The subject may then use k, which is say the Facebook username, and the Verify function to access his/her claims.

$V : V(c[], x, k) \rightarrow$\{0,1\}. The Verify algorithm allows external parties (other than the Identity Provider) to verify the claims. With the exception of a mandatory field k, c'[ ] and x are optional inputs to the Verify algorithm. The output 1 corresponds to a successful verification while 0 corresponds to an unsuccessful verification. In the same Facebook scenario, the Verify algorithm is invoked with k = username, and x = password. If, and only if, the Verify algorithm returns 1, the user is granted access to their account (verified claims).

In the Proof of Work (PoW) algorithm, the greater is the number of nodes participating in the consensus, the higher is the chance of honest nodes mining the block earlier. So the system gets safer but the delay as well as the computational energy consumption get longer and highly increased.

The PoW algorithm depicts the operation confirmation that relies on the confirmation of subsequent nodes while there exists a risk of rollback. Therefore, the PoW algorithm can be improved by integrating advantages of the PBFT (Proof of Byzantine Fault Tolerance) algorithm. Specifically, the nodes directly participating in the Byzantine agreement (consensus) are controlled in a small fixed range.

So we can choose an optimal set of nodes (which are probably more honest) for participation in consensus per block addition. The objective is to achieve the reduced forking ability of transaction log. It can only be possible if and only if the consensus of the algorithm can follow a minimum divergence of the transaction class after dividing the log into an equivalent class. Moreover, we cannot also ignore the power consumption and latency for each transaction nodes. Therefore, finally we need a majority minimization consensus algorithm to set this objective w.r.t all the constraints.

The background and preliminaries of introducing the model of Inverse Reinforcement Learning (RL) is required. However, the relevant concept of Federated Learning (FL) is also appreciated. FL trains a shared global model by iteratively aggregating model updates from multiple client devices, which may have slow and unstable network connections. Initially, eligible client devices first check-in with a remote server. The remote server then proceeds federated learning synchronously in rounds. In each round, the server randomly selects a subset of available client devices to participate in the training. The selected devices first download the latest global model from the server, train the model on their local datasets, and report their respective model updates to the server for aggregation.

In this paper, we consider the following scenario: each device (including the IoT and edge device) periodically creates a Merkle tree in which each leaf node represents a data record collected by sensors. The information of a client in each selection round (including the Merkle tree root, the status of training, the size and the centroid distance of client data for model training) are all stored in the pre-deployed smart contracts on the blockchain through client data anchor. This smart contract is hybrid consensus in nature. If a dispute (e.g. about failure cause) occurs, the operators of both the Master device and its client associated devices can use dispute resolver to verify the integrity of client historical data in a certain "stewardship" period, by comparing the data with the respective Merkle tree root stored on-chain. To reward the client organizations for their contribution to model training, the RL algorithm yields the distance of classes for policy data.

The consensus optimization impact for such algorithm and under Blockchain networks is described in the next sub-section.

A. Approach Towards Improving PoW Consensus Algorithm: Hybrid approach

Primarily, we include the notion behind PBFT consensus algorithm into PoW consensus algorithm. In turn, the objective is to combine PoW with the selection or polling based consensus protocols such as Practical Byzantine Fault Tolerance (PBFT). This combination will help to configure the consortium or approval driven platforms.

We then implement Reinforcement Learning along with an iterative policy based optimization.

1) Step by Step Approach:

- After the hash is generated for a particular Merkle tree constituting a particular block, it gets reflected on every node gradually. The miners start mining on the block to find the sequence of block for attaching the block on chain.
- Every participating miner solves the hash and produces a set of transaction logs in sequence. As every miner works individually, multiple valid log sets can be constructed based on the received transaction log set, but gradually it gets converged to a similar sequence.
- As the logs produced by different miners are different and unique, so we can transform them to the same sized abstract vector representation using LSTM module.
- The same sized vector produced by the LSTM module, for corresponding miners is used in optimization algorithm, whose objective is to select a subset of vectors, for which the probability of forking is the least.
- We choose then the miners, whose corresponding log vectors were selected by the optimizer. A state vector for our reinforcement learning algorithm (S), is introduced, which is basically a binary representation of the nodes selected and not selected, for example: [0, 1, 1, ...0, 1] for N nodes (participating miners).
- Transition function/action of RL can be defined as the modification of selected nodes/miners by updating the optimizer parameter, which in turn selects some different sets of abstract log vector, so that a different set of miners gets selected.
- Thus, a modified reward function under PoW and PBFT modification for RL can be defined as:

$$hash(T)\text{=}hashvalue, verify(hashvalue \geq targetvalue). \tag{1}$$

A brief explanation about the function is given here: when the chosen set of miners is found mostly malicious, i.e. our optimizer is working inappropriately, in that case the incentive is mainly received by malicious nodes. So the reward function becomes $-ve$. The RL model gets penalized for such wrong miner subset selection. Consequently, the RL takes an action and updates the optimizer parameter accordingly.

The goal of the RL model is to select the optimal/honest set of miners for mining a particular block. We can also define it as $Pf \leftarrow Pmin$, i.e., the probability of forking is less than a certain probability under which the final decision/consensus doesn't get affected, OR the probability of forking up to which the blockchain is robust/tolerant.

Algorithm 1 ***f-IRL Primary parameters*** **version 1**

f-*IRL D*ecision Process (S,A,T,β , D,R)
S : Finite set of states
A : Set of actions
T = {P_{sa}} : state transition probabilities ϵ [0,1) :
$\beta \in [0.1]$ discount factor
D : initial state distribution
R(s)=$\sigma^T \phi$(s) : reward function :
S $[0,1]^k$: k-dimensional feature vector
Policy π : S $\Rightarrow$A
2nd level :
Assume the feature distribution of the master CPS as value V is given.
If we can find a policy πsuch that $\| \mu(\pi) - V\mu \|_2$ then
we have for any underlying reward R*(s) =σ*T(s) ($\|w\|_1 \leq 1$)
here $\mu(\pi)$ feature distribution of the given policy and v is the importance or value of the policy.

A global parametric view of the primary model is shown in Algorithm 1.

The significant point of the context of distributed and federated system is also worthy to mention under Blockchain network. In a Blockchain network, a full node maintains a complete list of every single transaction that had occurred on the Blockchain. In the proposed architecture, each participating organization client, including master organization and client organization, hosts one Blockchain full node. Hence, each organization has a full replica of the data stored on the Blockchain which can be used for auditing and ensuring the availability of the whole system. The conventional federated learning proposed by Google may not be effective due to the data diversity in different CPS as well as with Blockchain networks.

Therefore, a grade or weighted class is introduced, which works as the distance between the positive class and the negative class of each client dataset.

The Blockchain-consensus update mechanism is achieved by Algorithm 2.

All PBFT and PoW require a complex crypto puzzle to resolve. In practical scenarios, PBFT possesses the fault tolerance with at least the participation of $3f + 1$ nodes [20]. Therefore the Merle root and the data structure allow towards a random verification of hash values where this participation frequency reduces in case of malicious nodes.

IV. Deployed Data, Target Simulations & Analysis

We modeled a real distributed cyber physical system on the basis of edges and nodes with their licit and illicit transactions. It should be noted that the proposed approach has not been simulated with real-time data. Only the primary efficacy is verified on the collected transactions under similar environment where plenty of IoT and edge devices are

Algorithm 2 Updating and Policy selection under Blockchain

/*Root Blockchain network of Devices*/

Initialize σ /* grade or weighted class*/

for each round t = 1, 2, ...do

for each finite set of states S = 1, 2... do

Select K clients for each client k $\in$ K clients do

Randomly pick a policy 0, set i=1

Compute t_i= $\max_{t,w}$ t

$\sigma_t^k \leftarrow$UpdateDevicePolicey( )

$d^k - t \leftarrow$ the distance between two classes in training dataset

end for

$(d_t^k) \rightarrow 1\frac{1}{d_t^k}$

$\$\sigma_t \Rightarrow \frac{\Sigma_{k=1}^{K} = 1nk \times f(d_t^k) \times w_{t-1}}{\Sigma_k^K = 1nk \times (d_t^k)})$

end for

/*Client update*/

UpdateClientDevicePolicey( )

{

Initialize local minibatch size B, local epochs E, learning rate η

for each epoch $i \in E$ do

Randomly sample State Si based on size B

Randomly pick a policy 0, set $i = 1$; Compute t_i = max(t,σ_t)

$w_i \rightarrow w_i - 1$ - $\eta \nabla g(wi-1 ; S_i)$

end for

return σ_t

}

interacting and performing transactions with respect to identity validation.

Indeed, we have experimented on an available dataset[1] from kaggle. The dataset contains a large set of transactions, each of which is either labelled as licit (exchanges, wallet providers, miners, licit services, etc.), illicit (scams, malware, terrorist organizations, ransomware, Ponzi schemes, etc.) or unknown data instances.

A. Description of Nodes and Edges

In Figure 2, the accuracy of the conventional and ideal model of IRL is addressed. The plot in ideal condition and hypothesis (without being federated) considers the number of sample data flow paths (shown as trajectory) versus the performance of the expert policy to be fed into the learning model. If we follow the color dots, green navigates only the value of IRL as output including non-zero data features. However, this condition in distributed systems is rare. Therefore, we consider a more realistic scenario of ideal IRL where it includes both zero and non-zero data features (shown in red color).

Here, few interesting analytical points need to be mentioned. Firstly, we have to use sampling estimates for the feature distribution of the expert. It means that the master models of machine learning need appropriate feature distribution with sampling estimates. Subsequently, if the data is organized as a grid of $m \times n$, assuming 4 even actions on IRL output, 70% success (otherwise random among other neighboring squares). Also, non-overlapping regions can demonstrate their features. A small number has non-zero (positive) rewards. This IRL will be more feasible when we have a network of connected cars and we illustrate how different driving styles can be learned (if it is the form of videos or in the form of image-frames and more over they are not distributed in nature).

In this case, we consider a public available data set (as mentioned). The description is follows:

- Number of nodes : 203,769
- Number of edges : 234,355
- Number of illicit nodes : 4,545
- Number of licit nodes : 42,019

The paper considers these features:

- Total number of features : 166
- Transaction Features (local features) : 94 (first)
- Nodes Features (Aggregated features) : 72 (remaining).

We have created virtual workers for each of the nodes present in the dataset. These virtual workers act as sub nodes in the federated model. In the setting, we have implemented Inverse Reinforcement Learning using different supervised classification algorithms, to help the IRL model learn an appropriate policy, which can accurately prevent occurrence of malicious/invalid transactions. We have trained the IRL model in 49 episodes.

We assume here that the given dataset can be well-described by a Gaussian (normal) distribution. Such a distribution is defined by its mean vector and its covariance matrix. Therefore, if we are able to estimate these two parameters from the training set and the data following the Gaussian assumption (or without following a probability density function that is massively different from absolute normal), then we can easily classify the transactions based on their node and transaction features. In this context, the simulation is presented in Figure 2. As depicted in this figure, it is quite evident that the IRL master model of normal maximum likelihood outperforms, while selecting precisely the level of trust and honesty of devices as opposed to the IRL with SVM as a master model. It should be noted that both the simulations have been done with the same specification and parameters of data features.

V. Conclusion

The paper proposes a unique model to support either IoT multiple-device oriented networks or the associated trust worthiness of data through federated machine learning and Blockchain-based approaches. In order to reduce the heterogeneous property of the distributed data, we implemented a Federated inverse Reinforcement Learning on the basis of distance and weights. We also combined the concepts of *Proof of Work (PoW)* and *Practical Byzantine Fault Tolerance (PBFT)* to mitigate the contradictory decentralized concept of Blockchain. However, the work

[1] "https://www.kaggle.com/ellipticco/elliptic-data-set"

Fig. 1. **Ideal IRL Experiments using Frame/Grid Like data instances**

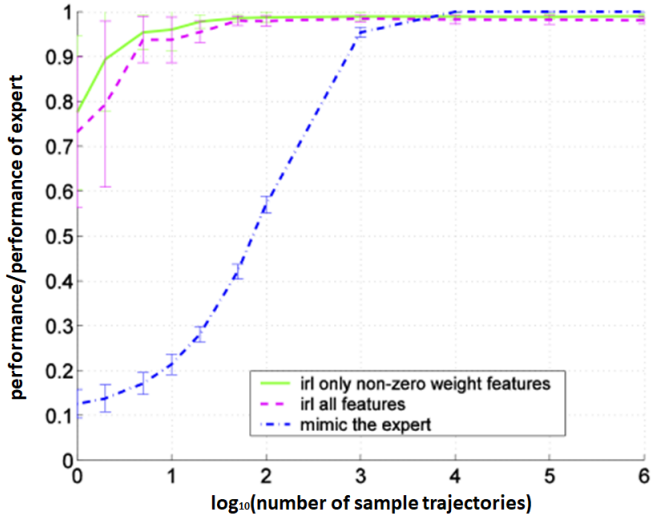
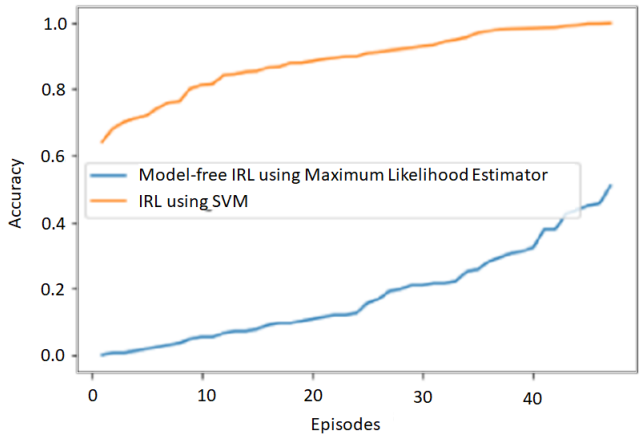


Fig. 2. **Multi-Class Simulations using f-IRL**



expects more challenges of real-time use cases of CPS to analyze the complexity of such high degree of hybrid algorithms. Moreover, the proofs of hybrid Blockchain cannot be presented here, which will require more real-identity and sovereign concepts towards further implementation.

REFERENCES

[1] F. Tschorsch and B. Scheuermann, Bitcoin and beyond: A technical survey on decentralized digital currencies, IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2084–2123, 2016.

[2] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, A survey on consensus mechanisms and mining strategy management in blockchain networks, IEEE Access, vol. 7, pp. 22 328 – 22 370, 2019.

[3] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and G. Das, Everything you wanted to know about the blockchain: Its promise, components, processes, and problems, IEEE Consumer Electronics Magazine, vol. 7, no. 4, pp. 6–14, 2018.

[4] K. Fanning and D. P. Centers, Blockchain and its coming impact on financial services, Journal of Corporate Accounting & Finance, vol. 27, no. 5, pp. 53–57, 2016.

[5] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, Blockchain technologies for the internet of things: Research issues and challenges,IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2188–2204, 2018.

[6] H.-N.Dai, Z.Zheng,andY.Zhang, Blockchain forinternet of things:A survey, IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8076–8094, 2019.

[7] S. A. Abeyratne and R. P. Monfared, Blockchain ready manufacturing supply chain using distributed ledger, 2016.

[8] K. Qi and C. Yang, Popularity prediction with federated learning for proactive caching at wireless edge, in IEEE Wireless Communications and Networking Conference, May 2020, pp. 1–6.

[9] Robert E. Hiromoto, Michael Haney, Aleksandar Vakanski, A Secure Architecture for IoT with Supply Chain Risk Management, The 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications 21-23 September, 2017, Bucharest, Romania.

[10] F. Masood and A. R. Faridi, Distributed ledger technology for closed environment, in 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE, 2019, pp. 1151–1156.

[11] K. Li, H. Li, H. Hou, K. Li, and Y. Chen, Proof of vote: A high-performance consensus protocol based on vote mechanism & consortium blockchain, in 2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE, 2017, pp. 466–473.

[12] S. Y. Lim, P. T. Fotsing, A. Almasri, O. Musa, M. L. M. Kiah, T. F. Ang, and R. Ismail,Blockchain technology the identity management and authentication service disruptor: a survey, International Journal on Advanced Science, Engineering and Information Technology, vol. 8, no. 4-2, p. 1735, 2018.

[13] H. Kim, J. Park, M. Bennis, and S. Kim, Blockchained on-device federated learning, IEEE Communications Letters, pp. 1–1, 2019.

[14] Y. Zhao, J. Zhao, L. Jiang, R. Tan, and D. Niyato, Mobile edge computing, blockchain and reputation-based crowdsourcing iot federated learning: A secure, decentralized and privacy-preserving system," 2019.

[15] U.Majeed and C.S.Hong, Fl-chain, Federated learning via ecnabled blockchain network, in 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS), Sep. 2019, pp. 1–4.

[16] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive, IEEE Transactions on Dependable and Secure Computing, pp. 1–1, 2019.

[17] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, Blockchain and federated learning for privacy-preserved data sharing in industrial iot, IEEE Transactions on Industrial Informatics, pp. 1–1, 2019.

[18] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, Incentive mechanise for reliable federated learning: A joint optimization approach to combining reputation and contract theory, IEEE Internet of Things Journal, vol. 6, no. 6, pp. 10 700–10 714, 2019.

[19] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu, Flchain: A blockchain for auditable federated learning with trust and incentive, in 2019 5th International Conference on Big Data Computing and Communications (BIGCOM), Aug 2019, pp. 151–159

[20] Yaqin Wu, Pengxin Song, Fuxin Wang, "Hybrid Consensus Algorithm Optimization: A Mathematical Method Based on POS and PBFT and Its Application in Blockchain", Mathematical Problems in Engineering, vol. 2020, Article ID 7270624, 13 pages, 2020.