Synergy of Edge Intelligence and Blockchain: A Comprehensive Survey

Xiaofei Wang, *Senior Member, IEEE*, Xiaoxu Ren, *Student Member, IEEE*, Chao Qiu, *Member, IEEE*, Zehui Xiong, *Member, IEEE*, Haipeng Yao, *Senior Member, IEEE*, Victor C.M. Leung, *Life Fellow, IEEE*

Abstract—Edge intelligence (EI), as an emerging technology, has been attracting significant attention. It pushes the frontier of artificial intelligence (AI) from the cloud to the network edge, aiming to embrace and support the next-generation communications while unleashing AI services. However, it faces challenges in its decentralized management and security, which limit its capabilities to support services with numerous requirements. On the other side, blockchain (BC), as a promising decentralized technology, is beneficial to tackle the above issues. However, there exist some technical challenges for BC, such as transaction capacity, scalability, and fault tolerance. Motivated by significant current interest around EI and BC, this survey examines whether the synergy of EI and BC can make a powerful network with combined functionalities of both cutting-edge technologies. Accordingly, we develop EI-chain and Chain-intelligence to realize reliable computing-power management, data administration, and model optimization at the edges, while improving the functions of BC by leveraging complementary characteristics of EI and BC, further making up for their own limitations. In this survey, a wide spectrum of literature is carefully reviewed to enable the EI-chain and Chain-intelligence. Moreover, we cover the technological aspects of EI-chain and Chain-intelligence: overview, motivations, and frameworks. Finally, some challenges and future directions are explored. We believe this survey will provide developers and researchers a comprehensive view on the synergy of EI and BC, while accelerating the design of a powerful integrated network.

Index Terms—Edge intelligence (EI), blockchain (BC), synergy, EI-chain, Chain-intelligence.

I. INTRODUCTION

Recently, driven by advancements in fifth-generation (5G) networks, the internet-of-things (IoT) technology continues to mature with widespread devices changing people's lifestyle and enhancing social efficiency. Meanwhile, immense volume of data are generated by mega-scale terminal devices instead of traditional cloud datacenters. According to the prediction by Ericsson [1], IoT devices will generate 45% of the 40 zettabyte (ZB) global Internet data in 2024. Nevertheless, global devices transferring extremely vast data to

cloud datacenters will demand high bandwidth and powerful computational resources [2], thus resulting in a series of challenges in network transmission capabilities, computing power of computing infrastructures, strict delay requirements, etc. Indeed, edge computing [3] has emerged as a complementary processing architecture to eliminate the above challenges, and it deploys computation tasks in proximity to the data sources for prompt response performance and reliable services. Meanwhile, with the breakthrough of artificial intelligence (AI) and deep learning (DL) technologies, as well as the improvements of computing architectures, the marriage of edge computing and AI, i.e., edge intelligence (EI), pushes the AI frontier from the cloud to the network edge to unleash AI services [4], [5]. Specifically, EI is a burgeoning paradigm that integrates network, computing, storage and AI, while providing EI services and satisfying the critical requirements of the Internet era in agile connection, real-time business, data optimization, application intelligence, security and privacy protection, etc. Notably, the celebrated Gartner hype cycle has regarded EI as an emerging technology that will enter a stationary phase in the following 5 to 10 years [6]. Meanwhile, new research activities on EI and its applications are making impacts in a broad spectrum of fields, such as IoT [7], smart cities [8] and smart home [9].

However, EI technology is at the preliminary stage of development, while it faces multiple challenges, further limiting its capabilities to support services with numerous requirements. Moreover, EI leverages cooperations among various AI services across a multitude of end devices, edge nodes and clouds. Such cooperations may lead to poor performance and energy efficiency, heterogeneous resource management, privacy leakage and other issues. These issues are still waiting for us to investigate and address, further accelerating the large-scale implementation of EI applications.

On the other side, blockchain (BC), as an essential concept and kernel architecture in Bitcoin [10], is anticipated to play an outsize role due to the distinct technical features, including consensus protocol, cryptography algorithm, distributed ledger technology, etc. Moreover, BC enables a shift in the computing paradigm from centralized control to decentralized control [11], and it records transactions between two parties in a permanent way without the need of third-party authentication. Furthermore, BC brings many benefits in terms of reducing costs for trusted third parties, improving resource management efficiency, verifying data authenticity, protecting privacy and ensuring security. Hence, it can be regarded as a

X. Wang, X. Ren and C. Qiu are with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China. E-mails: xiaofei-wang@tju.edu.cn, xiaoxuren@tju.edu.cn, chao.qiu@tju.edu.cn.

Z. Xiong is with the Pillar of Information Systems Technology and Design, Singapore University of Technology and Design, Singapore 487372, Singapore. E-mail: zehui_xiong@sutd.edu.sg.

H. Yao is with Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100088, China. E-mail: yaohaipeng@bupt.edu.cn.

Victor C.M. Leung is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518061, China, and also with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver BC V6T 1Z4, Canada. E-mail: vleung@iee.org.

Corresponding author : Chao Qiu (chao.qiu@tju.edu.cn)

Fig. 1: EI-chain and Chain-intelligence

complementary technology to make up for the EI limitations. Nevertheless, BC is facing many technical challenges, such as transaction capacity, scalability, and fault tolerance, that prevent many BC systems from coming to fruition [12].

Therefore, due to the substantial current interest around EI and BC, the synergy of EI and BC has received significant attention. This survey aims at examining whether the synergy of EI and BC can make a powerful network with combined functionalities of both cutting-edge technologies. Accordingly, we develop EI-chain and Chain-intelligence to enable reliable computing-power management, data administration, and model optimization at the edges, while improving the functions of BC by leveraging complementary characteristics of EI and BC, further making up for their own limitations. As shown in Fig. 1, we describe EI-chain and Chain-intelligence. **On the one hand**, implementing EI on the BC, i.e. EI-chain, synthesizes decentralized data storage, computing and AI models leverages BC. Further, EI-chain can provide decentralized, personalized, secure and intelligent sharing services to advanced applications by employing some learning-based methods to address the existing problems at edges with lower complexity. **On the other hand**, the BC system assisted by EI, i.e. Chain-intelligence, endows it with plenty of computational resources and AI models distributed at edges. The major goal of Chain-intelligence is to provide significantly better performance of BC, while satisfying the diverse requirements in EI scenarios at a faster pace by integrating the learning-based algorithms into the BC system. This potentially addresses the bottlenecks in BC, further enabling scalable storage and computation on the BC.

To our best knowledge, many studies on 5G, IoT, BC, edge computing and related issues have been conducted from a wide range of technical aspects. Moreover, there have been many efforts that attempt to converge IoT, BC, AI or edge computing to compensate for each other's shortcomings. In [13], driven by the breakthroughs in 5G network and BC technology, the synergy of BC with 5G networks and beyond is discussed. Specifically, this study explores and analyzes some of the key technologies and opportunities that BC empowers diversified

5G services, including cloud computing, edge computing, network slicing, etc. In addition, the recent advances in BC also bring solutions to address the challenges of IoT. In [14], the authors propose the synthesis architecture of BC and IoT, illustrating the potentials brought by BC to 5G-beyond networks, while discussing the industrial applications and the major challenges of such synthesis architecture. In [15], the authors propose the integration system of BC and cloud of things. This system aims to establish a decentralized management architecture, improve data privacy and system security, significantly reduce the complexity of system implementation, and thus serve more application scenarios. Moreover, the authors in [16] incorporate BC and edge computing to securely support massive network servers, data storage, and validity computation in proximity to the end. This survey elaborates how the integrated system meets the needs of networking, storage and computing. Furthermore, in [17], the convergence of BC and machine learning (ML) for communications and networking systems is investigated. Benefit from the complementary characteristics of BC and ML, this convergence can improve the performance of ML in data and model sharing, decentralized intelligence, etc., while enhancing the functions of BC in scalability, security and privacy, etc.

Nonetheless, the above studies mainly focus on the convergence of the two technologies in BC, AI, edge computing or other techniques. There is no existing work to integrate the combination of AI and edge computing (i.e., EI) with BC. As such, this survey provides a comprehensive investigation on the synergy of EI and BC on many aspects, ranging from overview, motivations, integrated frameworks to research challenges and future directions.

A taxonomy graph of this paper is presented as Fig. 2. We first given an overview of EI and BC in Section II. Then, the motivations of EI-chain and Chain-intelligence are explained in Section III. Next, we describe the above two enabling technologies in detail, i.e., EI-chain (Section IV) and Chain-intelligence (Section V). Section VI discusses research challenges and future directions. Lastly, the conclusion is drawn in Section VII.

II. OVERVIEW OF EDGE INTELLIGENCE AND BLOCKCHAIN

In this section, we will give a brief overview of EI and BC.

A. Edge Intelligence

EI is the next stage of the development of edge computing and interweaves many concepts and technologies together in a complex manner. Specifically, EI pays more attention to the combination with AI applications to promote the landing and realization of related technology and industry, which has strong engineering guiding significance. Instead of uploading all data to the cloud of cloud intelligence, key processing and decision making will be completed at the edge of the local network, even in the vicinity of the terminal devices. This will not only reduce the latency, energy and bandwidth usage, but enable users to keep their private data within their

Fig. 2: The taxonomy graph of synergy of BC and EI

own infrastructure. Next, we describe EI from the following aspects: the definition, division and deployment.

1) *Definition of EI*: Although EI is surging in popularity, there is still no universal standard for its relevant concepts. In 2019, International Electrotechnical Commission defines EI as the capabilities of the data acquisition, storage, analysis and aggregation with ML algorithms at the network edge [18]. The work in [19] regards EI as the capability to enable edges to execute AI algorithms with four-element tuple $\langle \text{Accuracy}, \text{Latency}, \text{Energy}, \text{Memory footprint} \rangle$ to describe EI abilities. In [20], EI is considered as a platform to complete DL training and inference offloaded from the cloud. The work in [5] believes that EI should be a paradigm for leveraging available data and resources in the hierarchy of end devices, edge nodes, and cloud centers, rather than being restricted on the edge servers or devices, to optimize the overall performance of AI training and inference.

2) *Division of EI*: EI involves many concepts, technologies and components intertwined in an enormously sophisticated manner. According to the correlation between AI and edge environment, the EI can be classified into three categories.

- *Component-based division of EI*: Combined with the edge environment, the significant components of AI, i.e., data, algorithms and computation, will turn into edge caching, edge training, edge inference, and edge offloading [21]. As shown in the top left of Fig. 3, based on the edge-cached datasets, edge training learns the optimal model parameters to provide a more precise prediction for edge inference. In addition, edge offloading aims to offload the tasks on devices with limited resources to the edge servers, enabling the computation services to run efficiently, thereby supporting other components.
- *Relationship-based division of EI*: As shown in the bottom left of Fig. 3, there is an interactive relationship between AI and edge computing. Deng *et al.* [22] distinguish EI into: AI for edge (solving optimization problems in edge computing with AI technologies) and AI on edge (how to run AI models on edges). Wang *et al.* [20] believe EI is to push DL computation from the cloud to the edges as much as possible, while classifying Edge DL into five enabling technologies (i.e., DL on/at/in/for Edge and Edge for DL).

Fig. 3: The division of EI

- *Cooperation-based division of EI*: As shown in the right side of Fig. 3, the AI tasks, including training or inference, are not solely executed at the edges, but can run in the coordination of cloud-edge-end manner via data offloading. Based on the amount and path length of data offloading, the EI can be rated into six levels, from cloud-edge co-inference (the bottom level) to all on-device (the top level) [5].

3) *Deployment of EI*: As neural networks (NNs) become increasingly larger in size and more complex in terms of structure, the computation burdens bringing from EI deployment, including enormous demands of data processing and computing power, are also gradually increasing, leading to the difficulty of running sophisticated tasks for edge devices. Therefore, the EI deployment needs to accommodate the resource constraints on edge nodes. Considering that the research interest focuses on model, data, and computing power, we divide the deployment methods of EI into the following categories, as shown in Fig. 4.

Computing-power resources optimization: the first method is to optimize computing-power resources at the edges, including software and hardware, as shown in the bottom of Fig. 4. **On the one hand**, studies on software optimization are centred on optimizing, scheduling, and managing the resources allocation based on AI models in order to reduce the redundancy latency and improve the system performance [23], [24]. **On the other hand**, the potential enabling hardware of EI is mainly aimed at accelerating the calculation of NNs. Specifically, the different types of hardware, such as GPU-based [25], FPGA-based [26], and ASIC-based hardware [27], are the direct ways to improve the velocity of AI calculation. At present, some commodities for edge nodes, i.e. Huawei's Atlas modules [28], NVIDIA's Jetson [29], etc., and frameworks for edge computing, i.e. Kubernetes [30], OpenEdge [31], etc., have been developing rapidly to execute and deploy AI tasks in edge environments.

Data management: the second method is to manage a tsunami of data at the edges, including data caching and offloading, as shown in the middle of Fig. 4. **On the one hand**, data caching at the edges can reduce transmission

and computation latency, while enhancing users' quality of experience. This makes EI become a service model for better and faster delivery of intelligent services. Specifically, after quantifying and analyzing the users' requests, the caching deployment can intelligently determine the suitable caching places, data, AI models and methods [32], [33]. The caching update process then replaces the old data with the new in a timely manner [34], [35] to provide real-time, high-quality services to end users. **On the other hand**, data offloading can offload tasks by the AI algorithms to other edge devices, fully utilizing the resources available in the edge environment. This further provides the computing services for EI deployment. Specifically, device-to-cloud, device-to-edge, and other hybrid offloading strategy coordinate different levels of edge devices capacity, while flexibly scheduling, decoupling, and assigning of AI tasks to achieve latency minimization, energy-efficiency for the EI application [36].

Fig. 4: The Deployment of EI

Model establishment: the third method is to establish the lightweight model, including model training and inference, as shown in the top of Fig. 4. **On the one hand**, for the model training at the edge, there exists numerous enabling techniques for improving the critical performance of model training. Specifically, transfer learning (TL) [37] migrates the model suitable for big data to small data, while realizing personalized migration and reducing the training energy cost. Federated learning (FL) [38], [39] and deep neural network (DNN) splitting [40] leverage locally-trained model updates and partially-processed data to enhance the security of lightweight AI models in edge scenarios, rather than utilizing the raw data. Gossip training [41], as a rising decentralized training method, shortens the training latency by exchanging and computing information in an arbitrarily

connected network of nodes. **On the other hand**, for the model inference at the edge, model design is committed to reducing the size and latency of NNs for promoting the model effectiveness [42]–[45]. Model compression, such as pruning [46], data quantization [47] and knowledge distillation [48], can minimize the model complexity to relieve the pressure of the end devices. Model early-exit [49], [50], model selection [51], model partition [52], [53] and input filtering [54], [55] can realize inference acceleration, further speeding up the deployment on edge devices with the limited memory.

B. Blockchain

Looking back at computer technology, architecture, and related design practices over the past half century, Internet and centralized computing architectures enabled global access from various kinds of computing devices. Currently, computing resources, e.g. computing power, storage, infrastructure and protocols, have gone through a radical shift of resources distribution from concentration to dispersion. BC is the crucial innovation that makes this shift possible [11]. Some permissionless BCs attempt to give control of digital assets to end users without the need of intermediate nodes. Other permissioned BCs are designed to maintain logical centralization of some information while adopting a decentralized architecture [56]. Furthermore, BC has evolved over the last decade into one of today’s largest breakthrough technologies, with the potential to impact every industry from financial to manufacturing to educational institutions. Then we describe BC from the following aspects: the definition, architecture categories and working procedure.

1) *Definition of Blockchain*: BC is not a fresh, stand-alone technology, rather it is an innovative application mode of existing technologies, including distributed data storage, point-to-point transmission, consensus mechanism, encryption algorithm, etc. In the narrowest sense, BC, as a chained data structure, combines data blocks sequentially in a chronological manner, while it can also be regarded as an encrypted distributed ledger that cannot be tampered with. In the broad sense, BC, as a novel kind of distributed infrastructure and computing paradigm, uses the block-type data structures to validate and store data, while utilizing the distributed consensus algorithm to generate and update the data. Meanwhile, BC leverages the cryptography way to ensure the security of data transmission and access, and takes advantage of automated smart contracts to process and operate data [57].

2) *Architecture of Blockchain*: Generally, the BC architecture is decoupled into essential main layers [58], [59], including data, network, consensus, incentive, contract and application layer.

- *Data Layer*: As the lowest layer in BC architecture, the data layer encapsulates the chained structure of time-stamped data blocks with asymmetric encryption technology [60].
- *Network Layer*: Once the transaction generates between the parties, it is broadcast to all neighboring nodes by the peer-to-peer network.

Fig. 5: Capabilities comparison of public, private and consortium BCs.

- *Consensus Layer*: It is one of the key technologies and the governance mechanism of BC community to make highly decentralized nodes reach consensus. Currently, the mainstream consensus mechanisms include: proof of work (PoW) [10], proof of stake (PoS) [61], practical byzantine fault tolerance (PBFT) [62], etc.
- *Incentive Layer*: Economic incentives will be issued as rewards and allocated to corresponding nodes for driving these nodes to contribute their efforts to verify data.
- *Contract Layer*: A type of self-executing various scripts, algorithm mechanisms and smart contracts will bring programmability into BC, enabling BC unalterable and resistant to malicious attacks [63].
- *Application Layer*: The highest layer in BC is the application layer, consisting of various applications, such as IoT, smart city, smart business, smart healthcare, etc.

3) *Categories of Blockchain*: According to different applications and thresholds, BC is generally classified as public BC, private BC, and consortium BC. Specifically, public BC is also known to be the permissionless system, where anyone can join in the BC and reach consensus freely. Moreover, public BC normally relies on the incentive-based consensus mechanisms for securing the system, due to the fact that it would be computationally expensive to tamper with the chained contents. A private BC, also known as the permissioned system, is managed by a single organization. Hence, private BC provides a feasible way for some organizations and applications to handle sensitive data and record-keeping. Consortium BC, also known as the permissioned BC, is operated by a group of pre-selected organizations. Only a small part of nodes would be selected to participate in the BC and reach consensus. In Fig. 5, we summarize the comparison among public, private, and consortium BCs in the following aspects.

- *Decentralization*: The major difference among the three types of BC is that public BC is fully decentralized, while consortium and private BCs are partially decentralized or fully centralized, respectively.
- *Immutability*: It is almost impossible to tamper with

Fig. 6: An overview of Bitcoin workflow.

transactions in a public BC. Differently, the participants on the consortium and private BCs can perform read or write operations.

- *Nonrepudiation, transparency and traceability*: Since a full copy of all transactions ever executed is stored to all the nodes on public BC, it could totally reassure the non-repudiation, transparency, and traceability of transactions of the BC system. As for private and consortium BCs, they cannot or can only partially ensure these properties.
- *Scalability*: In the public BC, it has the weaknesses of cost of low transaction-validation rate, high latency and extra storage space consumption, limiting the scalability of public BC. As for private and consortium BCs, they have better scalability as the consensus determined by one or multiple groups can be easily reached.
- *Flexibility*: Compared with private and consortium BCs, public BC has less flexibility as configurations of private and consortium BCs are more regulable.

4) *Working Procedure of the Blockchain*: Generally speaking, BC, as a chained data structure, is a linear structure beginning with a so-called genesis block and continuing with every new block linked to the chain in a chronological manner. To facilitate a better understanding of BC technique, we introduce the working procedures of various BC types with different examples respectively as follows.

Public BC: we take Bitcoin as an example to elucidate the working scheme of public BC. Rather than entirely trusting on the third party, Bitcoin takes advantage of some sort of proof under cryptographic primitives. A classic Bitcoin maintenance mechanism works as shown in Fig. 6.

When a newly initiated transaction is submitted, it is hashed and disseminated to each node on BC by its submitter. Then, the nodes in the BC validate the transaction and the identity of the node submitting the transaction. After that, the special nodes (i.e., miners or voters) package the different subset of newly verified transactions into their candidate blocks and broadcast them to the whole network. In terms of some cryptographic puzzles defined by the consensus protocol, all or part of the BC nodes engage the block verification and fight for the bookkeeping. Next, the verified block packaging the verified transactions is attached to the BC, while each BC node updates its local replica. Arguably, all blocks are linked sequentially using the hash pointers as a linear linked list. By this means, the latest block is linked to its immediately-previous one through a hash label with the exception of the first block,

while the new block is constantly being superimposed on the old blocks. Hence, it is nearly impossible to modify or alter the content of BC.

Consortium BC: we take Hyperledger Fabric in the consortium BC as an example. Unlike the Bitcoins and Ethereum in the public BC, the nodes in Hyperledger Fabric enroll through a trusted membership service provider. Moreover, Hyperledger Fabric, as a modular BC framework, consists of several modules, including the access authentication, consensus service, the endorsement, the block submit module. Therefore, the developer can replace modules according to the specific business scene, realizing the plug-in module management. In addition, the peer nodes in Hyperledger Fabric can be divided into committer, endorser, leader and orderer. The working procedure of Hyperledger Fabric is shown as Fig. 7.

Fig. 7: An overview of Hyperledger Fabric workflow.

Specifically, the process is summarized as follows. ① *Sending a trade proposal*: The client sends a trade proposal to the peer node, which contains parameters required for the transaction. ② *Simulating execution of trade proposals*: The endorser invokes the chain code to simulate the execution of the proposal. Each execution produces the data sets that are read and written to state data, called RWsets. ③ *Returning proposal response*: The endorser endorses the RWsets, generates a proposal response and returns it to the application. ④ *Submitting transactions*: The client submits transactions containing endorsements and execution results to the orderer peer. ⑤ *Ordering the transactions*: The orderer peer orders transactions and packages them into a block. ⑥ *Broadcasting block*: The block will be broadcasted to the whole BC. ⑦ *Saving blocks to the ledger*: The leader peer will save the blocks to the ledger. ⑧ *Synchronizing block*: The peer nodes in the BC network would synchronize the block information. ⑨ *Validating the transactions and updating BC*: After the

TABLE I: Important abbreviations list in alphabetical order

Abbr.	Definition	Abbr.	Definition	Abbr.	Definition
AI	artificial intelligence	FL	federated learning	PoCI	proof of common interest
BNN	bayesian neural network	FFNs	feed-forward neural networks	PoET	proof of elapsed time
BC	blockchain	5G	fifth- generation	PoS	proof of stake
CD	computing device	HRL	hierarchical reinforcement learning	PoQ	proof of training quality
CSP	computing service provider	IC	incentive compatibility	PoW	proof of work
CDA	continuous double auction	IR	individual rationality	PoDL	proof-of-deep-learning
CNN	convolutional neural network	IIoT	industrial internet of things	PoK	proof-of-knowledge
DL	deep learning	IDC	international data corporation	PoL	proof-of-learning
DNN	deep neural network	IoV	internet of vehicles	PoT	Proof-of-trading
DQL	deep Q-learning	IoT	internet-of-things	PoU	proof-of-utility
DQN	deep Q-network	LM	learning markets	QoS	quality of service
DRL	deep reinforcement learning	LSTM	long short-term memory	SFC	service function chain
DTWN	digital twin wireless networks	ML	machine learning	SP	service provider
DAG	directed acyclic graph	MDP	markov decision process	TPS	transactions per second
DML	distributed machine learning	MEC	mobile edge computing	TL	transfer learning
DSIC	dominant-strategy incentive compatibility	MARL	multi-agent reinforcement learning	UDEC	ultra-dense edge computing
ECSP	edge computing service provider	NNs	neural networks	VANET	vehicle ad hoc network
EI	edge intelligence	P2P	peer-to-peer	WAN	wide area network
ePoW	enhanced-proof-of-work	PBFT	practical byzantine fault tolerance	ZB	zettabyte

synchronization of the block, the committer peer validates all transactions in the block. Furthermore, the node updates the RWsets to the status database and appends the block to the BC, so as to update the ledger. Then, each committer notifies the client whether the transaction has been completed successfully.

Private BC: the private BC offers a feasible way for a single entity or an enterprise to share and verify transactions internally. Hence, the members are allowed to join in the private BC only through an authentic proposal. As the blocks are published by delegated nodes within the network, the participants in the private BC could modify or tamper with some rules or even data. Therefore, the private BC can be seen as a distributed ledger or database [11], [64]. Specifically, the process is summarized as follows. Firstly, a trusted intermediary, such as a joint venture entity or a guarantee company, controls the access right to the private BC. Secondly, the trusted intermediary will build the BC software used to set up nodes, and further forming the private BC. Thirdly, the trusted intermediary will run the core collection nodes, responsible for validating the transactions prior to them being recorded on the ledger. Fourthly, participants can access the private BC by the built software, so as to send transactions for recording on the BC.

In this paper, we summarize the important definitions and related abbreviations are listed in Table I.

III. MOTIVATIONS OF EI-CHAIN AND CHAIN-INTELLIGENCE

As shown in Fig. 8, the limitations of EI and BC are painfully clear. Spontaneously, the appearance of EI-chain and Chain-intelligence would be expected to pave the way for the development of emerging intelligent services. In this section, we first discuss the motivations of EI-chain and Chain-

intelligence. Then, we elaborate the benefits EI-chain and Chain-intelligence.

A. The definition of EI-chain and Chain-intelligence

We first give the brief definitions of EI-chain and Chain-intelligence. Concretely, EI-chain refers to deploying EI on the BC, while Chain-intelligence focuses on improving the performance of BC by EI. Specifically, EI characterized by decentralization, low latency, intelligentizing, etc., enables the edge resources to gain AI insight while making the intelligence closer to the distributed edges. Furthermore, BC has the following key characteristics, including decentralization, autonomy, transparency, data integrity, traceability, reliability, etc., ensuring a unified transaction ledger and providing trusted computation among a group of distributed participants. Hence, the same decentralization mechanism of both EI and BC, as well as their complementary characteristics are destined to the appearance of EI-chain and Chain-intelligence. They are expected to facilitate the evolution of intelligent services, further achieving mutual complementarity and benefits in the interest of their common development.

That is, the technical feasibility of EI-chain and Chain-intelligence consists in the complementary characteristics of BC and EI, while the availability of EI-chain and Chain-intelligence aims at giving play to different advantages of BC and EI, and taking advantage of complementary features to make up for their own limitations.

B. The motivations of EI-chain and Chain-intelligence

There is no gainsaying that the EI and BC play the role of the driver for diversified intelligent applications, while there are still major challenges to be addressed. Then we demonstrate why we concentrate on the EI-chain and Chain-intelligence, i.e., the limitations of EI and BC.

Fig. 8: The motivations of EI-chain and Chain-intelligence.

1) *The Limitation of Edge Intelligence*: Despite the many benefits that EI offers, it remains a challenging new area of research. According to the type of EI deployment methods, we correspondingly divide the EI architecture into four layers, including edge devices, computing-power, data, and model layer. As shown in the left of Fig. 8, we emphasize the challenges facing the last three layers.

Computing-power layer: the soaring demands for ubiquitous AI services are now posing new challenges on the computing-power management in the computing-power layer. For example, i) IoT renders various devices of different capacities concatenate in a network to communicate with each other, while it is also the root of other problems such as heterogeneity [65]. How to manage the heterogeneous computing-power resources to adapt to the diversification of customers' demands has become a significant concern. ii) Typically, edge nodes, characterized by the dispersibility, employ the widely scattered devices, which suffer from the limited computing capabilities, storage resource and battery life due to their physical constraints [3], and they could not support the power-hungry or computation-intensive AI services. How to integrate the dispersed resource to provide a series of computing solutions while offering a capillary distribution of computing power to the resource-constrained computing platforms need to be addressed. iii) Moreover, the computing devices with limited resources do not support additional hardware security features, such as TPM, HSM, SGX Enclave and hardware virtualization, making them vulnerable to malicious attacks.

Data layer: the ensuing deluge of data generated at network edges places a heavy burden on data administration for the data layer. According to the prediction by international data corporation (IDC), global data sphere would reach 175 ZB by 2025, while Ericsson forecasts 45% of the 40 ZB global Internet data will be generated by IoT devices in 2024. These massive amounts of data lack effective data administration solutions, and trigger other issues. For instance, i) most organizations consider data as a valuable strategic resource. As such, they

tend to regard the locally generated data as private property, and are amotivational or even reluctant to share the local data with other organizations. ii) Moreover, the collaborative edge is utilized to connect the geographically distributed edges of multiple devices for handling the intensive distributed data collaboratively. However, this data collaboration strides across wide area network (WAN), which may compromise the acceptable time latency. iii) Furthermore, the overwhelming majority of distributed devices are subordinate to a hub-and-spoke topology, or a server-client architecture, which is likely to cause security risks for data administration [66], [67].

Model layer: EI enables mobile terminals or edge servers to learn models while performing the local calculation based upon on-device data samples in the model layer. The various devices can cooperative communication mutually through the EI technology, wherein there exist certain vulnerabilities. For example, i) the independent computations and communications of edge devices in distributed learning model are hard to get the global model [23]. ii) Currently, most of mobile devices in the existing distributed learning mainly follow with their own interests or gains. Therefore, some self-interested mobile devices will be unwilling to participate in distributed learning tasks, which makes the devices suffer from considerable overheads and affects the performance of distributed learning [68]. iii) Some learning models have the confidence deficit issue. For FL, the data owners may intentionally or unintentionally mislead a global model during the distributed learning process. Specifically, they may intentionally send malicious updates or indeliberately update low-quality models to adversely affect the global model parameters resulting in the deviation of distributed learning [39].

2) *The Limitation of Blockchain*: Although BC brings new favourable opportunities for intelligent services and applications, it is still in its infancy, and there remain several critical issues. According to the architecture of BC, we correspondingly divide the BC into five layers, including foundation (i.e. data and network), consensus, incentive, contract and system

layer. As shown in the right of Fig. 8, we emphasize the challenges faced in the last four layers.

Consensus layer: the consensus protocol in BC system is of vital importance, but has certain vulnerabilities. i) The popular consensus protocol (i.e., PoW) is committed to breaking a complex but meaningless cryptography puzzle, which places more heavy demands on the computation of hardware [69]. While with the explosion of global computing power, the renewal and iteration of mining machinery are gathering pace. Plenty of antique mining equipment puts aside to do not need, resulting in substantial resource waste and capital cost. Over the past few years, the amount of resources consumed by Bitcoin has grown dramatically, and it now requires more energy than the energy consumption of many countries such as Denmark, Ireland and Nigeria. ii) The current consensus protocols are mechanical, while they cannot be changed dynamically in response to users' needs [70]. iii) As a supporting technology, the consensus mechanism is open to malicious attacks, including 51% attacks, double spending attacks, and other issues [71].

Incentive layer: the incentive mechanism provides a certain economic reward, enabling more members to participate in the BC system. This, however, raises concerns on the flexibility of mining strategy and the security holes of incentive mechanism, for both academia and industry. i) The mechanical mining strategy may consume intensive resources, further making it difficult to run complex BC process in resource-constrained environments. Meanwhile, the flexible mining strategy enables the BC to be deployed at the edges, further promoting BC-based systems to be applied widely [72]. ii) Meanwhile, the incentive mechanism has brought forward some cybercrime risks against the crypto-graphic currency, including selfish mining [71], money laundering [73], Ponzi schemes [74], etc., resulting in the difficulty of ensuring the security of the BC system.

Contract layer: the smart contract can be considered as a self-executing procedure depending on the script codes, ensuring the unalterability of BC. But elsewhere, the existing smart contracts are not smart. i) Once smart contracts are deployed on a BC, it is difficult to modify and the implementation is incredibly inefficient. Due to the low robustness of the contract, the performance of BC degrades significantly when the code in smart contract is rewritten [75]. ii) The smart contracts are vulnerable to threats from malicious parties [76]. Specifically, the smart contracts in BC have become prime targets for hackers, due to the transparent nature of the source code and the potentially lucrative benefits from conducting a successful attack.

Application system layer: the performance of application system, referring to the scalability, throughput, latency, and cost per transaction, is one of the major concerns in the BC system. These factors have become the challenging issues with the ever-increasing number of transactions. At present, the number of transactions increasingly grow with the greater adoption of BC, and the currency networks need about 200 GB capacity to store the heavy transactions information.

Nonetheless, the slow processing speed of transactions is a serious but unavoidable problem. Typically, bitcoin BC can process on average 7 transactions per second (TPS), while Ethereum can perform an average of 20 TPS [77]. Such low throughput is really unacceptable, compared with *Visa*, which is already offering 24,000 TPS. Recently, a great variety of approaches is proposed to address the scalability issues for enabling practical applications, such as increasing the block size, sharding, pruning, and off-chain, etc [78]. However, these methods still have their respective problems. The scalability issue remains a severe bottleneck, being responsible for the poor performance of the BC application system.

Therefore, the limitations of the above-mentioned of EI and BC may restrict their wide applications. Fortunately, their complementary characteristics are conducive to the combinations of EI and BC, a.k.a., EI-chain and Chain-intelligence. They provide promising ways to work around the above limitations, while creating numerous benefits.

C. The Benefit of EI-chain and Chain-intelligence

The clear descriptions of EI-chain and Chain-intelligence are given as Section III-A.

On the one hand, EI can be deployed on BC to solve the challenges while of three layers in EI architecture, denoted as EI-chain. EI-chain places emphasis on exalting the capability of resource management, collaborative data administration, and model optimization supported by BC, for giving birth to novel applications from cooperative and reliable data sharing, ubiquitous computing to intelligent model optimization. In Fig. 8, for the computing-power layer, the foundation, consensus, incentive, and contract layer in BC will help EI manage the heterogeneous computing-power, and make this layer valuable, adaptable, and safe. Meanwhile, due to the built-in advantages in BC, the same layers as utilized above can contribute to administrating the diverse data, further making this layer profitable, credible and cooperative. Furthermore, provided by the potential advantages from the above layers in BC, the AI models in EI will be well optimized, further enabling the layer model to achieve availability, efficiency and security. We will discuss these opportunities brought by EI-chain in Section IV.

On the other hand, the inherent characteristic in EI technology could offer the potential solutions for establishing the intelligentizing BC system, and possibly address the challenges of four layers in BC architecture, denoted as Chain-intelligence. Chain-intelligence places emphasis on improving the functions of consensus protocol, incentive mechanism, smart contract and BC application system. As shown in the right of Fig. 8, for the consensus protocol layer, the edge devices, computing-power, data, and model layer in EI will help BC enhance the compatibility of consensus protocol. Meanwhile, due to the complementary characteristics of EI, the same layers in EI will help BC improve the operability of the incentive layer in BC. Furthermore, provided by the potential advantages from EI, the smart contract will gain great intellectuality. For the application system layer, EI will help BC enhance the system scalability, further speeding up a

truly global BC adoption. We will discuss these opportunities brought by Chain-intelligence in Section V.

IV. EI-CHAIN: IMPLEMENTATION OF EDGE INTELLIGENCE ON BLOCKCHAIN

EI-chain focuses on implementing EI on the BC, which can address the challenges of EI as described in Section III-B. In this section, we present the benefits that can be realized with the assistance of EI-chain, including computing-power management, data administration and model optimization.

A. Computing-Power Management

Along with the rise of ubiquitous EI applications, the computing demands have been showing continuous and rapid growth. Moreover, as the computing architecture sinks from the cloud center to the edges, the resource-constrained edge devices are not able to support the power-hungry or computation-intensive EI applications. As one of the most important pillars of EI development, ample computing power fuels the booming of ubiquitous EI applications. However, the tension between computing-hungry EI and computing-constrained edges mainly leads to the following challenges of computing framework and management.

- **Weak incentive:** the incentives of computing-power sharing is weak. Accordingly, some EI devices cannot support the increasingly complex computing tasks due to the limited computing resources, while other nodes with abundant computing power may be idle due to a lack of value incentive.
- **Complex environments:** the computing-power allocation should take full account of diversified complex factors in EI-chain environments, further perplexing the design of resource allocation strategy.
- **Inefficient performance:** in view of complicated scenario due to the introduction of EI, the service performance that the computation offloading faces is generally inefficient. It is necessary to deeply consider how to reduce the task offloading latency, promote the quality of service (QoS) of users, while ensuring security, privacy, and fairness in the process of computation offloading. Accompanied by the EI deployment on the BC, EI-chain offers a viable solution to tackle the above issues, and manage the heterogeneous while distributed computing-power resources.

1) *Value-Driven Computing-Power Sharing:* Computing cooperations among fogs, edges and end devices are achieving great success in empowering end-users with rich experience by utilizing resource virtualization and sharing [79]. However, due to the lack of value-driven computing-power sharing, the computing resources among the distributed computing devices are underutilized, causing the contradiction of computation-power supply. BC is utilized to incentivize computing nodes to share their computing resources while avoiding trading privacy leakage. In the following, we investigate the related works of EI-chain in computing-power sharing. As shown in Fig. 9, prior works mainly focus on designing the incentive

mechanisms for the computing device (CD) to rent computing power from the computing service provider (CSP) to run the mobile BC, under the novel computing paradigms and communication protocols. Then the strategies of CSP and CD will be optimized by the *game-based methods* or *auction-based methods* for realizing the value-driven computing-power sharing.

Fig. 9: The value-driven computing-power sharing based BC

Game-based methods: there already exist several studies on the game theory and pricing models for computing-power sharing [80]–[82]. Specifically, in industrial internet of things (IIoT), the work in [80] investigates the resource management for the IIoT-based BC network, while modeling the interaction between the computing-power provider and miners as a Stackelberg game. Then, a multi-agent reinforcement learning (MARL), a model-free algorithm, is invoked to realize the near-optimal policy for maximizing the value profits of members in the EI-chain system with incomplete information. Assisted by the token-incentive of BC, the devices in EI-chain prefer to contribute their computing power to maintain the distributed shared ledger that records the details of the computing sharing. Meanwhile, as more devices move from the role of free riders to consensus nodes, computing-power sharing will become more robust.

In the mobile edge computing (MEC) network, the interactions between the service provider (SP) and miners are described by a stochastic Stackelberg game under private information about the miner's actions [81]. Due to the fact that every miner's hash rate decision is unknown and unobservable for other miners, the miners' decision-making is modeled by a partially-observable markov decision process (POMDP). Then, a hierarchical reinforcement learning (HRL) framework is designed for the SPs and miners, so that the miners can select the best response strategies under the optimal price assigned by the SP. Further, to reduce the complexity of the POMDP, the authors in [82] model the uncertainties in the EI-chain system by a bayesian neural network (BNN), which can reduce the size of the unobservable state space and, thus, the complexity of the learning algorithm.

Auction-based methods: the auction is an alternative solution for value-driven computing-power sharing while holding essential properties, such as individual rationality (IR), incentive compatibility (IC), etc. As illustrated in [83], by the continuous double auction mechanism (CDA), a BC-enabled system for computing-power trading enables providers and customers share their computing resources in a safe and fair manner. Significantly, the transaction data can be recorded in the block, including pseudonyms of providers and customers, trading price, trading quantities, etc. Then they are verified by the consensus layer, while the smart contract deployed on the EI-chain acts as a broker in the auction and does not need to reap profits from the spread, so as to guarantee credibility and fairness of the computing-power sharing. Further, a deep reinforcement learning (DRL) algorithm is leveraged to optimize bidding strategies while maximizing the payoffs of computing-power providers and customers.

The work in [84] constructs a hybrid BC-based computing-power trading system while designing a rational reverse auction mechanism assisted by the smart contract in edge computing. The hybrid BC framework is composed of consortium BC and public BC. On the one hand, the consortium BC is employed as the decentralized platform for auctions among requesters and edge nodes, while recording information. The public BC and payment channel technique, on the other hand, are adopted for promoting the efficiency of trading and ensure the security for all parties in the trading framework. Moreover, the smart contract-based incentive mechanism is designed to ensure the automation, autonomy, and security of auctions. Furthermore, due to the advantages of the consortium BC in efficiency, scalability, and privacy, the deployment of smart contracts on EI-chain will become low-cost, high-frequency and low-latency.

In addition, an optimal auction based on DL for edge resource sharing between the edge computing service provider (ECSP) and mobile users is introduced in the EI-chain network [85]. To achieve the optimal bidding strategies and maximize the revenue of the ECSP, the NNs architecture is constructed to precisely fit the optimal auction. In the NNs, the monotone transform functions are leveraged to determine the allocation and payment rules of the NNs architecture, ensuring the IR and dominant-strategy incentive compatibility (DSIC). Different from [85], [86] leverages feed-forward neural networks (FFNs) to derive the optimal auction for resource allocation in the EI-chain network.

In general, the above studies provide the possibility of promoting the value-incentive of devices to sharing their computing abilities by the introduction of BC, while the performance of value-incentive mechanism aiming to maximize the profits of the members in the EI-chain is improved by utilizing the learning algorithms.

2) *Performance-Driven Computing-Power Allocation:* Currently, there exist many excellent works focusing on resource allocation in the EI environment. However, there is a problem that the EI is a dynamic scenario with complex structure and many factors due to user mobility, network

condition varying, malicious attack, etc. As such, how to comprehensively cogitate multiple factors in the EI to improve the system performance becomes a critical challenge in the future computing-power allocation. A high-performance resource allocation should consider the following key issues: i) *adaptive computing allocation*, and ii) *privacy-preserving computing scheduling*. Due to the performance requirements in EI environments and the superior features of BC technology, numerous novel resource allocation schemes, integrating BC into EI, are designed. They aim to improve the adaptive performance of EI, i.e. the capability of complex tasks caching, computation and collaboration, as well as enhancing the security and reliability of the edge-centric computing architecture.

Adaptive computing allocation: recently, several excellent works have been conducted to improve the adaptivity of the BC-based resource allocation [87]–[89]. In [87], S. Yu *et al.* investigate the integrated framework by jointly optimizes the application partitioning, resource allocation and service caching placement for ultra-dense edge computing (UDEC). During the resource allocation, BC can be leveraged to guarantee efficient resource cooperation and reliable caching of UDEC networks. To minimize the task execution time and network resource usage, a novel two-timescale deep reinforcement learning (2Ts-DRL) approach is designed. Moreover, the 2Ts-DRL is trained by FL in a distributed manner, aiming to ensure data security. In [88], X. Fu *et al.* take latency of services and operational cost into consideration, while formulating the adaptive resource allocation problem as a multi-objective optimization problem. Especially, a consensus protocol is presented to ensure that the network-wide views can be synchronized and collected in a simplified manner across various EI systems. Specifically, the consensus layer in BC is used to verify the validity of resource information stored in the block while updating the state of the corresponding information about resources. Additionally, a novel DRL takes the highly dimensional complex data as input, and yields an optimal action, therefore reducing the complexity of the optimization problem.

Although the introduction of BC into the EI can enhance the adaptivity of resource allocation, the deployment of EI on BC consumes a large amount of energy as well. The work in [89] defines the performance of adaptive resource allocation as the latency of the users, throughput of BC, time to finality, decentralization and security, while formulating the spectrum allocation, block size, number of consecutive blocks as an optimization problem. Here the self-organized smart contracts in the BC can deliver the transaction data and execute tasks computation, and the transaction data represents the required information to complete the computation tasks. Then an offline DRL-based is first utilized to learn the optimal policy in an offline manner. After the model is trained, it can be leveraged to allocate the resources to optimize the performance of the EI-chain system in an online way, so as to address the problem of high dimensional raw data input.

Privacy-preserving computing scheduling: in addition, many secure and creditable BC-based computing resource

TABLE II: Details of EI-chain in computing-power management

Benifits	Ref.	Application	Model	Algorithm	Consensus mechanism	Strategy	Objective function	Property	Security Analysis	Smart Contract
Value-Driven Computing-Power Sharing	[80]	IIoT	Stackelberg game	MARL	PoW	Determination of service demand and price	Payment minus cost	Nash equilibrium	×	×
	[81]	MEC	Stackelberg game	HRL	PoW	Determination of service price	Discounted payment	Nash equilibrium	×	×
	[82]	IoT	Stackelberg game	HRL	-	Determination of service	Payment minus cost	Bayesian equilibrium, fairness	×	×
	[83]	IoT	Auction	DRL	PoW/PoS	Determination of bid for computing power	Payment minus valuation	Fairness	✓	✓
	[84]	Edge computing	Auction	FL	PoW	Determination of service	Valuation	IR, IC, fairness	✓	✓
	[85]	IoT	Auction	DL	PoW	Determination of bid for computing power	Payment	DSIC, IR	×	×
	[86]	Fog computing	Auction	DL	PoW	Determination of renting price and service price	Payment	IR, IC, fairness	×	×
Performance-Driven Computing-Power Allocation	[87]	UDEC	Joint optimization	2Ts-DRL/FL	-	Determination of resource allocation	System cost	Efficient utilization	✓	×
	[88]	MEC	Multi-objective optimization	DRL	Robust BFT	Determination of view changes, access selection, computing, caching and bandwidth allocation	System rewards	High-dynamic and large-dimensional	✓	×
	[89]	MEC	Multi-objective optimization	Double-dueling DQN	PBFT/DPoS	Determination of spectrum allocation and block attributes	Long term reward	High-dynamic and large-dimensional	✓	✓
	[90]	MEC	MDP	RL/DRL	PoW/PoS	Determination of resource allocation	Payment minus cost	Credibility	✓	×
	[91]	IoT	Optimization	DRL	PBFT	Determination of resource allocation	System rewards	High-dynamic and large-dimensional	✓	×
	[92]	IoT	Combinatorial Optimization	DRL	PBFT	Determination of resource allocation	System cost	Credibility	✓	×
	[93]	IoT	Continuous-time MDP	DRL	PoW/PoS	Determination of resource allocation	Profit minus cost	Credibility	✓	✓
Service-Driven Computing Offloading	[94]	MEC	Joint optimization	DRL	Delegated BFT	Determination of offloading decision, power allocation, and block attributes	System reward	Cooperativity	✓	×
	[95]	IoV	Joint optimization	DRL	Redundant BFT	Determination of offloading decision	System cost	High-dimensional	✓	✓
	[96]	VFC	Lyapunov optimization	ML	PoW	Determination of offloading decision	Average task offloading delay	Trustfulness, fairness	✓	✓
	[97]	MEC	Joint optimization	MADRL	-	Determination of offloading decision	System cost	Cooperativity	×	×
	[98]	IIoT	Joint optimization	DRL	PBFT	Determination of energy allocation	System cost	High-dynamic and large-dimensional	✓	×
	[99]	MEC	Joint optimization	DRGO	PoW/PoS	Determination of offloading decision	System cost	High-dimensional	×	×

allocation schemes are proposed. The BC-based trust mechanism can not only optimize the edge CPU resource allocation policy, but help MEC address the problem of the selfish edge attacks and the faked service record attacks [90]. Accordingly, in this scheme, the computational performance of edge devices would be evaluated, further broadcasted by BC. The optimal amount of computational resources allocated to devices would be formulated as a markov decision process (MDP) and solved by a RL-based algorithm, while a DRL version is presented to promote the computational performance further. By jointly considering the tasks caching, computing, as well as BC system usage, the work in [91] presents a novel resource allocation framework to reduce the unnecessary latency, while promoting the caching efficiency and system security in machine-to-machine communications. After data computing and processing, they can be uploaded into the BC, while being authorized by the consensus layer. In order to deal with the complicated and high-dimension features of in the decision process, a dueling deep Q-network (DQN) algorithm is used to formulate the actions and states offline, while obtaining the network updating online.

Additionally, the types of BC would affect the security performance of resource allocation as well. In [92], S. Guo *et al.* introduce consortium BC and DRL to establish the trusted and auto-adjust service function chain (SFC) orchestration architecture. Based on the consortium BC, the consensus nodes and light block nodes would be selected for resource registration, authentication, and transaction registration with smart contracts, to guarantee reliable and autonomous computing resources allocation. Meanwhile, the DRL algorithm executes multiple agents synchronously in parallel, which can accelerate the orchestration optimization process and more adapt to the high-mobility EI network. Y. He *et al.* [93] design a private BC while taking advantage of the smart contract to automatically allocate the computing resources for a new arrived data segment, which eliminates the problem of uneven distribution of resources. Instead of outputting the indirect value function as the suggestion, the asynchronous advantage actor-critic algorithm directly provides the explicit policy, effectively accelerating the allocation efficiency of resources.

3) *Service-Driven Computing Offloading*: In the edge scenarios, the emerging AI applications would generate a large volume of delay-sensitive and computation-intensive services. For the resource-limited devices, offloading the complex services to edge computing nodes rather than executing directly on the device, can extend the limited capabilities of the terminal devices while elevating the quality of service. Generally, the edge servers, deployed in the heterogeneous network, aim to provide computation offloading services for mobile terminals. For the above EI services, the computing offloading issues have been widely discussed in recent studies to enhance offloading efficiency. The main challenges in computing offloading could be summarized as follows [94]: security and privacy protection, dynamic computing optimization, etc. Fortunately, benefit from the advantages of BC, the BC-enabled EI systems could manage the above challenges, while

satisfying the services demands of the system.

Secure offloading strategy: BC, known as its security and immutability, can provide a promising solution to build truth among the EI system. In [95], an access control mechanism is implemented on the BC to validate and authorize access to mobile devices for the vehicle ad hoc network (VANET). The BC-based access control leverages smart contracts to effectively verify offloading tasks by triggering transactions, further preventing the illegal offloading action of VANET devices. Then the extended DRL can learn the offloading strategy with no need of state transition probability, and the trained DNN can describe the setting properly in the training process, which is more suitable for the high-dimensional VANET scenarios. The work in [96] presents a BC-based task offloading framework. It leverages BC to publish the malicious attacks and records task offloading failure transactions, so as to ensure the fairness and security. Meanwhile, it takes advantage of Merkle hash tree, constructed by the offloading data fragments, to further accelerate the verification of task offloading. Additionally, a queuing-delay aware, handover-cost aware, and trustfulness aware upper confidence bound (QUOTA-UCB) algorithm is developed to reach a well-balanced tradeoff among the above three dimensions of awareness, further increasing the possibility of achieving computing offloading success.

Dynamic computing optimization: due to the dynamics in the BC-enabled EI system, computing offloading should consider long-term and comprehensive offloading performance. Recently, many valuable works focus on the above points [97]–[99]. In these studies, the main role of the BC is to build trust among multiple parties of the EI environment, while guaranteeing the data authenticity in the process of computing offloading by the consensus process. The difference between these efforts lies in the formulated optimization problems and the type of solving algorithms. X. Li *et al.* in [97] focus on minimizing the long-term cost of cooperative computing offloading, while league learning is introduced to promote offloading performance by enabling the hierarchical agents to explore the environment collaboratively. In [98], the BC-enabled EI framework can reduce the computation overhead and the energy consumption of systems by jointly taking into account the selection of computing-power nodes, offloading decision and block size. In accordance with DNN, the DRL approach can handle the high-dynamic and large-dimensional offloading optimization problem by approximating the action-state value of the agent. However, the DRL-based approaches fail to converge quickly and accurately. The work in [99] introduces an adaptive genetic algorithm into the exploration of DRL, namely DRL combined with genetic algorithm (DRGO). It generates the candidate solution set of the action to avoid largely unexplored action space, further accelerating the learning process and the convergence performance of the BC-enabled computing offloading.

Summary: with the help of BC, the integrated system (i.e., Ei-chain) enables the valuable, adaptable and safe computing-power management of sharing, allocation and offloading, and addresses the problems of heterogeneity, dispersibility and

security attack in the computing-power layer of EI. All in all, we summarize the above-mentioned works of EI-chain in computing-power management in Table II.

B. Data Administration

Nowadays, zillions of bytes of data are constantly being generated and accumulated at network edges. The drastically growing volume and the types of data incur an urgent need to push the frontiers of AI to the edges, i.e., EI. However, the inefficient data administration among distinct multi-parties causes serious setbacks to the development of EI technique, for the following reasons:

- **Self-interest:** the EI devices are often selfish, and they may be reluctant to share data with others. Data sharing should be reasonably monetized to incentivize the paid data sharing by selfish EI devices.
- **High latency:** data sharing in collaborative end-edge-cloud network will generate the unacceptable latency, limiting the effectiveness of some applications, especially for the real-time service with high demands.
- **Security risk:** along with the value the data brings, serious issues about data collaboration in EI comes, such as privacy leakage, network communication failure, and malicious attackers. However, the present mechanism of data collaboration cannot ensure data security and reliability, especially at the edges of distributed parties.

BC has drawn much attention recently, as a promising approach to address the above issues. As shown in Fig. 10, implementing EI on the BC, i.e., EI-chain, can improve the efficiency of data administration, enable intelligence closer to the user. Meanwhile, the breakthroughs for AI algorithms will further pave the way for an EI-chain with greater profitability, credibility and cooperativity. Specifically, effective data administration of EI-chain is embodied in incentive data trading, data caching strategy, and reliable data collaboration.

Fig. 10: The data administration of EI-chain.

1) *Incentive Data Trading:* Generally, there are vast amounts of data waiting to be digested, transmitted, and shared in the distributed EI environment. However, the temporal-spatial diversity and heterogeneity of edge data, as well as the selfish edge devices, play an obtrusive effect on the data

sharing. Some current researches only focus on how to share edge data and cooperate with each other via AI algorithms, while ignoring how to incentivize more devices to contribute their data [100], [101]. BC can establish an Internet of data value, while unlocking the enormous potential of a data market economy. Some exceptional properties that BC brings can be of great benefit in realizing the exchange and sharing of data value without traditional intermediaries. Furthermore, BC enables the data sharing monetized in a distributed peer-to-peer (P2P) manner, further ensuring the efficiency of the EI-chain system. Especially, BC can solve the challenges of data trading by the following aspects.

The foundation layer, characterized by distributed data ledger and P2P network of BC, can remove trusted third parties from data trading, addressing the unrealistic and inefficient issues of centralized control and data retention in the data trading center. Moreover, the BC-assisted P2P data trading mode can be more adaptable to EI environments with the large-scale geographical distribution of massive EI devices, thereby maximizing the potential of the data market economy. In addition, the specific consensus process in the *consensus layer* can decrease the transaction verification latency and resource consumption, promoting the trading efficiency and accelerating the funds turnover in the trading process. In the frequent data-trading scenarios, such as internet of vehicles (IoV), because of the transaction verification delays and the high computing consumption, the devices requesting data generally do not have enough tokens to execute the next transaction immediately. Proof-of-trading (PoT) protocol [102], jointly considering the market interests of high-power nodes' and resource consumption, eliminates the above vulnerabilities. Moreover, the monetary aggregates in knowledge trading can be regarded as the stake, and the difficulty of the hash puzzle can be dynamically adjusted on the basis of the stake. As such, the PoT protocol can accelerate successful knowledge trading instead of solving a meaningless hash puzzle. In [103], a proof-of-knowledge (PoK) consensus mechanism replaces the complicated cryptography puzzle with the knowledge learning process. It allows the computation consumption in the knowledge trading process to be used for consensus verification in the BC, enabling the trading process to be more lightweight.

In the incentive layer, the incentive mechanism can release tokens while distributing the corresponding rewards to the miners, motivating more EI devices to contribute data as well as ensuring fairness. Different from the token incentive, a Shapley value solution [104], as a scheme for sharing revenues generated by a coalition, is a novel attempt to incentivize EI devices to perform actively in medical data sharing. Beyond that, the smart contracts in the *contracts layer* enable the automatic execution of multiple phases in the data trading process, improving the efficiency of the EI-chain. Based on smart contract and ML, the work in [105] proposes a secure data trading mode solution and framework in IoT. Benefit from the programmability and non-repudiation of smart contracts, the token trading between transaction participants can be automatically executed as well, incentivizing the devices to

act honestly. Meanwhile, the ML algorithm is leveraged to solve disputes about the availability of data trading.

2) *Data Caching Strategy*: The long distance among the edge devices, as well as the limited capacity of backhaul links, make it difficult to meet the actual needs of the increasing demand for content delivery and low latency in edge environments. Data caching provides all-in-one solutions to mitigate the mobile traffic on backhaul links effectively. Nevertheless, the high complexity of the EI scenarios and the strong mobility of the EI devices give rise to dynamic edge network topology and time-varying wireless channel conditions, which in turn hinders the design of optimal data caching strategy. In addition, EI devices are generally reluctant to store their data with a caching provider, due to the fact that the caching provider may be less trustworthy and the caching process may leak the private data. Fortunately, BC has the capacities to establish an autonomous, transparent and invigorative data storage market.

Data caching strategy: in the following, we investigate some related works that BC can benefit EI in data caching strategy. With the help of BC, the completed data transactions can be recorded into the block after the transaction verification, addressing the untrusted issues in data caching. In the healthcare field, the work in [106] leverages the BC and the off-chain pseudo anonymous data storage, so as to help users quickly collect and query data. Furthermore, the off-chain data storage strategy can quickly generate anonymous medical data sets, which can be used for ML and data mining, accelerating healthcare data administration's efficiency. Moreover, in [107], the authors propose the neural-BC-based drone-caching framework for the optimal caching and data sharing. In this framework, the distributed ledgers in BC offer reliable communication for the drones, while the NNs provide the support for intelligent data transport during drone caching. Besides, the new block verifier selection methods, i.e., consensus mechanisms, rendering data caching more fast and efficient. In [108], the authors integrate the data caching with the consensus protocol, and propose the proof-of-utility (PoU) consensus for vehicular edge computing. In the block production process, a certain number of data caching transactions would be packaged and be hashed. In the verification process, the correctness of data caching would be verified. As such, PoU provides deep integration of the BC and data caching. Combining with an advanced DRL approach, the optimal data caching strategy can be derived by learning edge network topology and channel conditions.

Data caching incentive: additionally, the token-incentive motivates EI devices to cache data while broadcasting them [109]. In this regard, the EI devices can download the cached contents by receiving broadcast information transmitted by the BC. Meanwhile, the data can be recommended and broadcasted to the relevant EI devices combined with the ML or DL methods, further improving the cache hit rate and robustness of data caching. Furthermore, the work in [110] adopts the cryptocurrency-based token to encourage the caching nodes to share the cached data, where the caching provider is rewarded

for data delivery and the smart contract supports the data delivery transactions. Moreover, the authors formulate the caching node selection and caching placement as a MDP, which can be solved via a DRL approach to obtain the optimal caching strategy by maximizing the accumulated reward.

3) *Reliable Data Collaboration*: The fast growth in the volume of data in network edges, opens up new possibilities for enhancing the quality of service for the emerging applications through data collaboration and sharing, but also poses serious concerns due to the heterogeneity and decentralization of EI devices, such as data tampering, data leakage, etc. These factors may be the bottleneck of EI [111]. BC employs cryptographic functions to create an append-only, tamper-evident log, so that the records in the BC cannot be tampered with and the activities of malicious customers or manufacturers are traceable. In addition, The consensus mechanism and smart contract technologies in BC enable the secure data collaboration among EI devices without central authorities. Therefore, it can establish trust among distributed devices and enhance data security, thereby building a safe and credible EI-chain system. For the secure data collaboration in EI-chain system, some existing studies have made attempts from different perspectives.

Due to the *distributed ledger characteristic and encryption algorithms* of BC, it is desired to solve the data consistency issue during the data collaboration [112]. BC can synchronize and broadcast the root hash value of the transaction among EI nodes, which in turn supports data registration while guaranteeing data consistency and ownership. In [113], data sharing can be encapsulated as a form of transaction, while a private BC is designed to store and share data. The introduction of the DRL algorithm can help establish an efficient data collection and sharing scheme in which the fully distributed DRL scheme is leveraged for data collection, and then Ethereum is used to share the collected data encrypted by the mobile devices to pertain to their security levels. At the same time, assisted by the ML, the EI datasets can be processed on-chain [114], and exclude the issues in data-related such as repetition, loss of data value, errors, and disruption, providing a lightweight yet effective solution for secure and private EI-chain system.

Based on the advantages of *consensus mechanism* in trusted transaction verification, all peers can audit and record the interactions between EI nodes during data sharing, thus supporting multi-party incentives and privacy protection. An enhanced-proof-of-work (ePoW) protocol can authenticate data records and prevent data poisoning attacks caused by tampering with the original data [115]. It can be combined with the variational autoencoder (VAE) and long short-term memory (LSTM) algorithm to protect data from leakage while performing anomaly detection. In [116], proof of training quality (PoQ) protocol can enhance the capability of data sharing while increasing the utilization of computing resources. Meanwhile, the data sharing can be formulated into a ML problem, so that data privacy can be preserved by sharing data models in security via BC, rather than exposing actual data. In addition, the EI-chain system is based on the *smart contract* that can support the

TABLE III: Details of EI-chain in data administration

	Ref.	Application	Algorithm	BC Type	Consensus Algorithm	Contribution	The main role of BC	Smart Contract	Security Analysis
Data Trading Realization	[102]	IoT	ML	Consortium/Private	PoT	Build a BC-enabled data trading market	Manage and trade knowledge in edge-AI enabled IoT	✓	✓
	[103]	IoV	FL	Hierarchical	PoK	Design a hierarchical BC framework and a FL algorithm for knowledge sharing	Improve the security during the data trading process	×	✓
	[104]	Healthcare	ML	Consortium/Private	PoW/PoS/PBFT	Construct a business model for medical data sharing	Share medical data	✓	×
	[105]	IoT	ML	Public	PoW	Establish a distributed data trading mode	Eliminate the needs of the trusted third parties in the data trading	✓	✓
Data Caching Optimization	[109]	CIoV	ML/DL	-	-	Propose a BC-based content caching scheme	Motivate EI devices to broadcast cached contents	×	×
	[106]	Healthcare	ML	-	-	Introduce a BC and an off-chain centralised data storage system	Record, collect and query medical data	×	×
	[107]	UAVs	NNs	Public	PoW	Present a neural-BC based drone-caching mechanism for intelligent transport	Guarantee ultra-reliability and provide flatten services	×	×
	[108]	D2D	DRL	Consortium/Private	PoU	Design an optimal content caching scheme	Ensure an intelligent and secure content caching among vehicles	✓	✓
	[110]	V2V	DRL	Consortium/Private	PoU	Design an optimal content caching scheme	Ensure an intelligent and secure content caching among vehicles	✓	✓
Data Collaboration Security	[112]	IoT	FL	Private	-	Realize the secure collaboration of multiparty data computation	Guarantee data consistency and ownership	✓	×
	[113]	IIoT	DRL	Public	Paxos/PBFT	Establish an efficient data collection and sharing scheme	Encapsulate data sharing as a form of transaction	✓	✓
	[114]	Smart home	ML	Public	PoW	Introduce a lightweight solution for intrusion identification and prediction	Process data on-chain	×	×
	[115]	Smart power networks	DL	Public	ePoW	Design a privacy-preserving framework	Check data integrity for authenticating data collections	×	✓
	[116]	IIoT	FL	Consortium/Private	PoQ	Build a BC-enabled data sharing architecture for distributed multiple parties	Share data while reduce the risk of data leakage	×	✓
	[117]	Healthcare	ML	Public	PoW	Present a personal health data sharing system	Share personal health data	✓	✓

more flexible implementation of data collaboration strategy, while realizing the trustworthy value exchange between EI nodes [117]. Generally speaking, among the above works, the *BC type* is permissioned BC, consisting of trusted organizations, which can establish more secure connections among credible EI devices through its encrypted records, further preventing network communication failure, malicious devices attacks, etc.

Summary: for the sake of the BC based storage and computation infrastructures to coexist with the EI infrastructure, the data trading, caching and collaboration need to be integrated with profitability, cooperativity and credibility. EI-chain leverages BC to collect, record, encrypt and broadcast the data transactions, while employing some learning-based methods to address the problems of data administration with lower complexity. To sum up, we summarize the above-mentioned works of EI-chain in data administration in Table III.

C. Model Optimization

Specifically, to accommodate the resource-constrained edge scenarios, some problems, such as the size of AI models, computation requirements of modeling, etc., should be optimized accordingly. However, the current model optimization in edge scenes faces many challenges, including i) *weak incentives*, ii) *inefficiency performance*, and iii) *security attack*. Thanks to the breakthrough of BC technology, many researchers are now offering viable solutions through the BC to tackle the above issues, such as availability, efficiency, and security.

1) *Availability:* Specifically, the availability of learning-based algorithms refers to how to provide some incentives for edge servers to improve learning accuracy [118]. These incentive mechanisms for the AI algorithms in edge environments can motivate edge devices to offer high-quality data and participate in model learning. Generally, continuously updated datasets are required to help improve the performance of the EI model. **On the one hand**, the block reward in BC can motivate the distributed edge devices to exchange and verify the local model updates. In [119], based on the nature of

BC, the BC-based FL (BlockFL) algorithm could promote the collaboration among edge devices with a larger number of training samples by providing rewards proportional to the training sample sizes. **On the other hand**, in [120], the BC can ensure the reliable reputation calculation to support the contract theory based incentive mechanism. This incentive-mechanism could host an efficiently updated learning model, stimulate the high-reputation workers that have high-accuracy and reliable local training data to join in the learning process, and improve the accuracy of learning models. Despite the financial incentive, the work in [121] rewards the data contributors of the AI model with points and badges. These non-financial incentives can be recorded on-chain in a smart contract, and identified by the contributor's wallet address for training a good AI model.

2) *High efficiency*: Besides, BC goes well with learning-based models and assists in getting more accurate models with high efficiency. Specifically, each edge device will have a copy of shared information, which supports the training of learning-based models. That is, the BC-enabled system, such as the digital twin wireless networks (DTWN), will increase the efficiency and accuracy of EI models, leading to better learning performance [122]. Moreover, R. Doku *et al.* in [123] bring EI to end-nodes by incorporating FL and BC, making AI more ubiquitous, named iFLBC. Specifically, the BC stores the shared learning model generated by the aggregation of the edge nodes learning results. Then the aggregated model will be provided to the EI to client nodes, and further achieving the high-quality AI services. Meanwhile, the proof of common interest (PoCI) protocol concentrates on sorting out relevant data from unrelated data, solving the scarcity issue of the relevant data. Different from centralized learning methods, decentralized learning approaches can overcome the inefficient and untrusted model aggregation. FL-Block, as a decentralized FL approach, can coordinate and verify the local learning updates by the consensus mechanism of the BC, solving the subsequent inefficiency of EI. Differently, Chain FL utilizes the private BC and smart contracts for model storage and model update aggregation [124]. Learning markets (LM), as a novel decentralized AI collaboration framework, leverages BC to construct a credible environment for collaboration and transaction, while utilizing smart contracts to encapsulate and process collaboration relationships [125]. These methods prevent poor quality data, handle large models efficiently with other decentralized storage options, and are expected to jointly complete the complex learning tasks.

3) *Security*: EI model, as the collaborative AI paradigm, brings great benefits for edge networks, while it is still vulnerable to various security attacks of the distributed clients. Currently, there are many works that have studied the security of the EI model, and introduced some corresponding solutions.

BC-aided FL models: specifically, to address the single point of the potential failure issue while providing a self-motivated and reliable learning environment, the BC-aided FL models are established [126], [127]. As shown in Fig. 11, the BC-aided FL models replace the central server with BC to

aggregate the local models uploaded from the end devices for obtaining the global model, extending the federation range among the untrustworthy edge devices. In this case, the poisoning attack could be resisted in a high-confidence way. Nevertheless, BC-assisted FL for intelligent edge computing may give rise to unintended property leakage and increase the risk to edge devices' sensitive data.

Fig. 11: The BC-aided FL models.

Focusing on the above problems, J. Kang *et al.* in [128] introduce reputation, managed by the consortium BC deployed at edge nodes, to defend against unreliable model updates for reliable FL. The consortium BC adopts proof of elapsed time (PoET) consensus and consists of the reliable workers. It can record and update the reputation opinions with the workers' digital signatures. Then, for a certain worker candidate, any FL task publishers can obtain the latest reputation opinions by downloading them from BC. By this means, the publishers are in a position to select the high reputation workers to perform the FL tasks securely. Moreover, a BC-based crowdsourcing FL system leverages differential privacy to prevent adversaries while ensuring that all model updates are held accountable [129]. Despite storing the models on-chain, the locally trained models are stored off-chain by using IPFS. After hash operations, they are sent to BC as the transactions. Then, the consensus nodes verify the authenticity of these transactions, and obtain the global model by aggregating the model parameters received from the distributed devices. Note that, only the consensus nodes of successful mining can upload the global model to BC. In [130], the authors integrate BC and FL into IoV, and develop a hybrid BC—PermiDAG, consisting of the permissioned BC and the local directed acyclic graph (DAG). In order to improve the reliability and security of the proposed scheme, the asynchronous FL architecture is proposed, consisting of three phases: node selection, local training, and global aggregation. Specifically, the DRL algorithm is leveraged to select participating nodes. The selected nodes then perform the asynchronous local training utilizing BC and execute the synchronous global aggregation based on the DAG. Moreover, the learned parameters could be integrated into the BC and be verified the qualities through two-stage verification for the higher reliability of learning models.

Other distributed learning-based models: moreover, the security issues of other distributed learning-based models

TABLE IV: Details of EI-chain in model optimization

	Ref.	Name	Algorithm	BC Type	Consensus Algorithm	Contribution	The main role of BC	Smart Contract	Security Analysis
Availability	[119]	BlockFL	FL	Public	PoW/PoS/BFT	Enable on-device ML with no need for centralized training data	Promote the collaboration among edge devices	×	×
	[120]	-	FL	Consortium	-	Stimulate high-reputation devices to participate in model learning	Ensure the reliable reputation calculation	×	×
	[121]	-	ML	Public	-	Train a model by the dataset built collaboratively on the BC	Record the incentives	✓	✓
High efficiency	[122]	DTWN	FL	Consortium/Private	-	Develop a BC-enabled FL framework for collaborative computing	Share the model information	×	×
	[123]	iFLBC	FL	Public	PoCI	Bring EI to end-nodes by incorporating FL and BC	Store the shared learning model	×	✓
	[124]	Chain FL	FL	Private	Proof-of-authority	Present a BC-based FL to delegate the responsibility of the model storage	Store the learning model and aggregate the model update	✓	✓
	[125]	LM	FL	Public	PoCI	Provide a learning market framework for AI collaboration	Encapsulate and process collaboration relationships	✓	×
Security	[126]	Flchain	FL	Consortium/Public	PBFT/PoW	Realize the secure collaboration of multiparty data computation	Store local model parameters for each global iteration	×	×
	[127]	FL-Block	FL	Public	PoW	Develop a novel BC-enabled FL to solve the subsequent inefficiency of EI	Verify the global learning model	×	✓
	[128]	-	FL	Consortium	PoET	Introduce reputation managed by BC for reliable FL	Record and update the reputation opinions	×	✓
	[129]	-	FL	Consortium	Algorand	Realize a BC-based crowdsourcing FL system to prevent adversaries	Aggregate the model parameters from the distributed devices	✓	✓
	[130]	PermiDAG	FL	Consortium/Private	Simplified PoW	Propose a hybrid BC architecture to improve the security of model parameters	Perform the asynchronous local training	×	×
	[131]	DPPDL	DL	Private	-	Propose a decentralized DL framework to ensure both fairness and privacy	Publish the differentially private data	×	✓
	[132]	-	ML	Consortium/Private	PBFT/PoET	Develop a DML model to prevent any attacks in the model aggregation process	Validated the model parameters validated	×	✓
	[133]	DeepBlock IoTNet	DL	Private	-	Facilitate secure collection and aggregation of the local DL model	Assist to upload, download and process the model parameters	✓	✓

without data centralization have been studied in many works. In [131], a decentralized privacy-preserving deep learning (DPPDL) framework is developed, where the differentially private data is published instead of the raw data or model parameters by a private BC. Meanwhile, it devises a three-layer onion-style encryption scheme to ensure both accuracy and privacy in a collaborative DL algorithm. The private BC contains two kinds of blocks for minimizing the effects of low-quality devices on the training model. An init block initializes the benchmark of the effectiveness of training data for every device, while an operation block consists of a series of transactions that define the upload or download operations. As such, the DPPDL further ensures fairness during the data download and upload processes of the model training. Differently, H. Kim *et al.* in [132] take a new error-based aggregation rule for preventing attacks by adversarial nodes in the aggregation process. This rule can aggregate the nearest learning results utilizing low errors saved in the immutable ledger as a log. Additionally, the committed block is composed

of the global model parameters validated by the authority node and aggregated by the error-based rule, enhancing the model security. Moreover, in [133], to mitigate the security challenges in IoT and EI, DeepBlockIoTNet integrates DL and BC to facilitate secure collection and aggregation of the local DL model from multiple edge servers. Beyond that, there are two key smart contracts in the DeepBlockIoTNet. The learning contract assists the learning devices to upload the model parameters and download the model updates. In addition to this, the processing contract helps the learning devices download the local model updates from the BC server for further processing to obtain the global model update, and then upload it to BC.

Summary: EI-chain can stimulate the secure collection and aggregation of model information from distributed devices, enhancing the availability, efficiency, and security of model optimization. Overall, we summarize the above-mentioned works of EI-chain in model optimization in Table IV.

V. CHAIN-INTELLIGENCE: INTELLIGENTIALIZING BLOCKCHAIN BEYOND EDGE

Chain-intelligence focuses on improving the performance of BC by EI, aiming to address the challenges of BC as described in Section III-B. In this section, we show the benefits that can be realized assisted by Chain-Intelligence, including flexible consensus protocol, effective incentive mechanism, intellectuality smart contract and scalable BC system.

A. Flexible Consensus Protocol

It is well known that the consensus protocols in BC networks face some critical challenges. **Redundancy:** the existing consensus mechanisms are redundant. For example, the PoW protocol requires brute-force to solve the mathematical puzzle, so as to determine the bookkeeping right of BC. As such, this redundant block mining gives rise to energy waste, and further dilutes the value of the BC. **Incompatibility:** due to the static configuration, the current BC lacks compatibility, only enabling one consensus protocol used. When implementing BC in edge scenarios, employing one consensus mechanism is almost impossible to adapt to the changing needs of edges. **Security:** there are widespread threats against consensus mechanisms, such as majority-attack, double spending attack, etc. The above challenges are yet to be addressed.

EI, as the emerging technology, promises a series of solutions to the mechanized BC system. In this paper, implementing EI in the BC means designing a lightweight and intelligent BC system while deploying it in an edge network. Assisted by the breakthrough of EI, consensus protocol would be more flexible, while the collaboration ability of members in BC networks would be tremendously enhanced. Specifically, flexible consensus protocol supported by EI is embodied in lightweight design, pluggable consensus mechanism, and attack defense.

1) *Multi-Functional Design:* Popular cryptocurrencies generally adopt the work-based consensus protocols to validate the transactions in a distributed ledger. However, these protocols need to consume considerable computing resources to solve the hash puzzle, which is expensive, redundant and not applied for anything other than validating transactions. Hence, many researchers are absorbed in designing a lightweight BC system by introducing the multi-functional consensus protocols [134]. EI, as a promising technology, provides a solution to address the redundant issue of consensus protocols. By designing a multi-functional consensus assisted by EI, BC recycles the computing resources, helping it become a more lightweight system that is easily adaptable to the edge scenarios.

We compare and summarize the different learning-based protocols, as shown in Fig. 12. In general, these learning-based protocols replace the laborious and pointless hashed calculation with DL training, enabling BC to share more advanced intelligence among edges.

PoDL: specifically, proof-of-deep-learning (PoDL) mechanism is an energy-recycling protocol [135], as shown in the top half part of Fig. 12. In PoDL-based BC, there exist a model requester, miners and full nodes. The model requester

Fig. 12: The comparison of different learning-based protocols

outsources DL model training to miners, then the miners engage in the DL training tasks, rather than the meaningless hash calculation, and the full nodes validate the training models on test datasets. This protocol only incrementally adds components to block headers, which can be generalized to any PoW-based BC. **PoL:** F. Bravo-Marquez *et al.* in [136] propose the proof-of-learning (PoL) consensus mechanism, as shown in the bottom half of Fig. 12. Different from PoDL, PoL includes three types of actors: suppliers, trainers and validators, which is inspired by ML competitions. Specifically, the suppliers host ML competitions and publish a task, the trainers next focus on training and submitting models for the released task, and finally the validators verify the performance of the model while deciding the winner to reach consensus. By this means, this protocol can not only support ML training, but create an open ML model and data set repository.

Proof-of-useful-work: Coin.AI [137], as a novel BC-based cryptocurrency, develops a proof-of-useful-work scheme to support its normal running, as shown in the left side of Fig. 12. Similar to the PoDL and PoL, the mining process in this BC is to train AI models. The distinction between these consensus protocols is that the digest containing the list of transactions, nonce and the hash of the previous block will be hashed, while the obtained hash value will then determine the hyperparameters for the trained DL model. Afterwards, the miner would win once the miner's DL performance exceeds a given threshold. Moreover, a proof-of-storage mechanism is introduced to provide the storage of DL models in a distributed manner, which is combined with the proof-of-useful work scheme to support the proliferation of Coin.AI.

PoQ: this protocol [116], integrating FL into the consensus process of permissioned BC, promotes the utilization of computing resources while increasing the performance of the FL algorithm, as shown in the right side of Fig. 12. Specifically, the PoQ consensus process is as follows. ① The committee node with the highest accuracy would be elected as a committee leader. ② The leader gathers all the received transactions, and broadcasts the block to all committee nodes for approval. ③ The model transaction track of a block is verified by the verifying node with the prediction accuracy.

TABLE V: Details of Chain-intelligence in flexible consensus protocol

Benifits	Ref.	Algorithm	Consensus Algorithm	Contributions	The main role of EI	Smart Contract	Security Analysis	Storage Analysis
Multi-Functional Design	[135]	DL	PoDL	Enable miners to perform DL training and utilize trained models as proofs	Provide a distributed and collaborative training environment	×	×	✓
	[136]	ML	PoL	Create a public distributed and verifiable database of ML models and experiments	Training and submitting models for the released task	×	×	×
	[137]	DL	Proof-of-Useful-Work	Regard DL models as the mining scheme for supporting a cryptocurrency running over a BC	Train DL models	×	×	✓
	[116]	FL	PoQ	Promote the utilization of computing work for consensus to FL	Build learning models used in the consensus protocol	×	✓	×
Compatibility Enhancement	[70]	DRL	PBFT, Zyzzyva, Quorum	Design a service-oriented permissioned BC and launch three consensus protocols based on the users' QoS requirements	Learn the transition regularities of system environment	×	×	×
	[138]	DRL	PBFT, VR, Paxos, Spinning	Launch different consensus protocols to meet the requirements of different users	Allow the selection strategies of the protocols much faster	✓	×	×
Attack Defense	[139]	ML	PBFT	Propose an AI-enabled blockchain with a two-step consensus protocol for enhancing the security and robust of BC	Reach consensus over new data in milliseconds, and enhancing the overall fault tolerance of the BC	✓	×	×

④ An approval will be distributed to the leader once the accuracy is within limits. ⑤ Further, the transactions will be recorded in the BC and sent to all nodes after signed with the leader's signature, if every committee node approves the block containing all transactions.

2) *Compatibility Enhancement*: It is well-known that none of the consensus mechanisms are perfect, and each has its own merits and demerits [140]. According to the actual application scenarios, the BC should be compatible with multiple consensus mechanisms and has the flexibility to select the most appropriate one to improve the compatibility performance of the BC [141].

Recently, EI helps the BC realize the pluggable consensus mechanism, which switches the different protocols flexibly to meet the diversified QoS requirements of edge scenarios, while making the BC system more efficient and robust [70], [138]. Since the learning-based algorithms can learn the transition regularities of the system environment, the QoS requirements of the users, and the situation of enabling resources, it would provide a feasible way for BC to select more suitable protocols to perform. The work in [70] designs a service-oriented permissioned BC and launches three consensus protocols (PBFT, Zyzzyva, and Quorum) based on the users' QoS requirements. Then, the consensus protocol selection, block producer selection, and network bandwidth allocation can be formulated as a joint optimization problem. Furthermore, it is addressed by a dueling DRL algorithm to achieve a service-oriented BC and enhance the performance of the BC system. Similarly, in [138], the authors focus on the adaptivity for BC, and quantify four consensus protocols, instead of using only one of them. Each protocol's qualitative performance, including throughput and latency, is analyzed for selecting suitable consensus protocols to meet multiple users' QoS requirements. Moreover, a DRL algorithm that learns the process and the architecture of the dynamic BC network, allows the selection strategies of the

protocols much faster for the various users' QoS requirements. As such, implementing learning can assist in designing the pluggable consensus mechanisms for the BC-based system, while improving the compatibility performance of system operation and making BC more suitable for the edge scenarios.

3) *Attack Defense*: The consensus protocols can verify the transactions, enabling the data stored in the BC immutable. However, these internal mechanisms in BC have many identified bugs and vulnerabilities, which raises serious concerns about BC security. Specifically, most of the BC nodes in edge networks are resource-constrained and very vulnerable to different types of cyber-attacks, such as majority-attack, double spending attack, etc. Recently, several AI algorithms have been utilized for a secure and robust consensus protocol in BC-based edge networks. M. Salimitari *et al.* in [139] propose a two-step consensus protocol for AI-enabled BC. The first step leverages an outlier detection algorithm for the second step consensus (i.e., PBFT) in edge networks. The outlier detection is committed to verifying the compatibility of new data, while discarding the suspicious ones. Moreover, applying ML to reach consensus in a BC-enabled edge network can not only reach consensus over new data in milliseconds, but also enhance the overall fault tolerance of the BC.

Summary: on the one hand, Chain-intelligence helps BC develop the multi-functional consensus protocols to recycle resources, while designing the pluggable consensus protocols to enhance the compatibility performance of BC. On the other hand, some security threats could be alleviated within the Chain-intelligence system. Overall, we summarize the above-mentioned works of Chain-intelligence in flexible consensus protocol in Table V.

B. Effective Incentive Mechanism

Currently, BC is picking up footing and becomes an innovative tool for changing people's lifestyles due to its incentive

TABLE VI: Details of Chain-intelligence in effective incentive mechanism

Benifits	Ref.	Algorithm	BC Type	Contributions	The main role of EI	Security Analysis	Storage Analysis
Mining Strategy Optimization	[142]	PL	Public	Formulate a multi-leader multi-follower Stackelberg game to solve the computing resource management problem in mobile BC networks.	Solve the mining task scheduling problem and obtain the optimal mining strategy	×	×
	[143]	DRL	Public	Address the task scheduling problem by a DRL-based algorithm to achieve an efficient intelligent strategy in mobile BC for IoT applications	Optimize the long-term system performance and obtain the optimal mining strategy	×	×
	[144]	DRL	Public	Formulate the mining task scheduling problem as a MDP to encourage the participation of the edge devices in the mobile BC mining	Address the mining task offloading problem and obtain the optimal mining strategy	✓	✓
Risk Prevention against Cryptocurrency	[145]	ML	-	Propose a model-based ML detection method to identify the fraudulent transactions in BC	Detecting the fraudulent behavior of the end-users	✓	×
	[146]	ML	Public	Reduce the anonymity of the Bitcoin BC to ensure complete data reliability	Predict the type of entity that has not been identified	×	×
	[147]–[149]	ML	Public	Detect the illicit patterns in Bitcoin transaction	Address the clustering problem formulated by the de-anonymization of cryptocurrency	×	×
	[150]	DL	Public	Detect the illicit activities in Bitcoin transaction	Obtain a few labels for achieving performance close to the optimal supervised baseline	×	×

mechanisms. However, the existing incentive mechanism faces various problems. i) The prevalent mining process, such as PoW, is mainly oriented to resource-rich nodes. General devices suffer from the limited computing power, and it is hard for them to participate in the mining process. Accordingly, the BC is difficult to be effectively integrated into EI systems, reducing the motivation for the distributed edge devices to join in BC. ii) While benefiting people's lives, the financial incentives of BC-based cryptographic currency also bring about cybercrime activities, such as money laundering, and ponzi schemes.

In recent years, several works have been conducted to address the above problems leveraging EI, i.e., Chain-intelligence. It further enhances the effectiveness of incentive mechanisms, including mining strategy optimization and risk prevention against cryptocurrency.

1) *Mining Strategy Optimization*: Generally, the public BC technology has been widely deployed in mobile environments. Nevertheless, the computation-intensive mining process impedes the practical use of public BC in mobile scenarios, due to the fact that the consumed computing power to perform mining tasks is prohibitively high for edge devices. As EI is prevalent, it has been validated to be able to effectively overcome the limitation mentioned earlier.

For example, the authors in [142] consider an ECSP and a CSP, who provide the computing power for executing the mining task. As shown in Fig. 13, utilizing the RL algorithm, the optimal mining strategy in resource management will be obtained by addressing the Stackelberg game between the SP and miners. To incentivize the participation of the edge devices in the mobile BC mining, the mining task scheduling problem is formulated as a MDP in the work of [143]. In order to achieve an efficient and intelligent strategy, a DRL-

based algorithm is then developed to optimize the long-term system performance. Unlike [143], D. C. Nguyen *et al.* in [144] consider the impacts of system performance and privacy levels of all BC users. They formulate the task offloading, user privacy preservation and mining profit as a joint optimization problem. Due to the fact that there is no prior knowledge of the system dynamics, a DRL-based algorithm is designed to efficiently solve the above problem and promote the offloading performances of mining tasks for larger-scale BC scenarios.

Fig. 13: The mining strategy optimization

2) *Risk Prevention against Cryptocurrency*: Recently, a number of cybercriminal activities attacking cryptocurrencies have proliferated, such as ransomware, money laundering, Ponzi schemes, and so on [74]. Since the cryptocurrencies have a high level of anonymity, they have been considered as the go-to currencies for illegal activities.

For example, a model-based ML detection method is proposed to identify the fraudulent transactions, while detecting the fraudulent behavior of the end-users with the additional investigation overhead though [145]. However, the traditional supervised algorithms are not applicable for the case that labels are scarce [146]. In the BC network, only a small minority of end-users have been reported as fraudsters, enabling the supervised technique infeasible [151]. As such,

a list of unsupervised approaches appear, they regard de-anonymization of cryptocurrency addresses as a clustering problem [147]–[149]. Through clustering objects and detecting roles, the unsupervised approaches locate anomalous behaviors in the BC. Moreover, to improve the performance of the unsupervised approach, active learning can be leveraged to obtain a few labels for achieving performance close to the optimal supervised baseline [150].

Summary: Chain-intelligence can leverage the learning-based algorithm to offer the flexible mining strategy for deploying the BC system in edges, promoting the mobile BC applications. Moreover, employing AI algorithms in the BC system can detect the malicious behaviors of devices, provide a secure end-edge environment, and further prevent cryptocurrency risk. To sum up, we summarize the above works of Chain-intelligence in effective incentive mechanism in Table VI.

C. Intellectuality Smart Contract

In spite of the fact that BC brings more reliable and convenient services, there still exist a few impediments in smart contracts and numerous issues remain to be addressed, mainly including the following several points. i) For the turing complete BC system, the current smart contracts are not smart. Actually, smart contract is a tiny computer program, which runs automatically when some specific conditions are triggered. Nevertheless, the legal viability, flexibility management and difficult implementation deteriorate the performance of smart contract, preventing smart contract becoming the basis of real-world agreements [17]. ii) The smart contracts are lack of robustness while they are readily threatened by some malicious parties.

Fig. 14: The intellectuality smart contract supported by EI

Meanwhile, with the rising of EI, the extension of EI methods to security management for BC has been a promising solution. Specifically, as shown in Fig. 14, EI gets the utmost out of the intelligent approaches, such as distributed machine learning (DML) and convolutional neural network (CNN), to improve the performance and enhance the robustness of the smart contract in BC, further benefiting the edges.

1) *Performance Improvement for Smart Contract:* Smart contract, as one prominent new application, provides a user-defined logical concept to BC. Actually, smart contract is an automated computer program that can be executed when certain predefined conditions are triggered. Accordingly, it supports automatic peer-to-peer transactions. While in the

meantime, there are potential efficiency problems in the current smart contracts, raising serious concerns about the lightweight BC systems. In this subsection, we aim to investigate the benefits that EI can bring smart contracts.

Robustness: smart contracts may be subject to errors due to human fallibility. Moreover, they are generally difficult to modify once they are deployed on the BC. Meanwhile, when the code in a smart contract is rewritten, its performance degrades dramatically. Therefore, many researchers are absorbed in the robustness improvement of the smart contracts.

Eth2Vec [152], as a ML-based static analysis tool, owns the robustness against code rewriting of the smart contracts. This tool can automatically learn features and separate feature extraction from the technical difficulty of the smart contracts analysis by ML. As such, the Eth2Vec performs quite well even after the codes are rewritten. Although smart contracts can be rewritten to handle BC’s scalable business behaviors, they require users to write programs in a specified programming language. Designed to be the novel generation of BC, MATRIX [153] develops an automatically generated smart contract by incorporating the BC and AI technologies. The code generator based on a DNN in MATRIX can automatically transform the script, including the core elements of the users, into an equivalent program. In this way, MATRIX can boost the transaction speed and make the smart contract more effective, further supporting the flexible operations of BC in the edge network. Furthermore, smart contracts are not very smart which are limited by their capabilities at the design stage. Cortex [154], as a new public chain, provides the support for combining AI algorithms with smart contracts. Specifically, it can run both existing contracts and inferred contracts with AI. The AI models trained by AI developers are appended to the BC, while smart contracts access the AI models by paying a certain amount of token for creating different types of smart contracts. This integration makes the smart contracts more diversified, generalized and efficient, providing credible assurance for numerous intelligent services.

High efficiency: even though smart contract is being exploited in various fields due to its reliability, there are still problems with their efficiency, including correctness, validity, flexibility, selection, etc.

By incorporating smart contracts, DML and IoT-based condition monitoring, a smart logistics solution is proposed in the supply chain management [155]. In this functional architecture of the solution, a cloud-based system can facilitate the implementation of smart contract, while distributed smart contract can monitor infrastructure in the IoT cloud. Meanwhile, the dynamic ML-based method efficiently selects the smart contract for guaranteeing the trust, traceability and accountability of the supply chain. Other than [155], the authors in [156] leverage ML to analyze and decrease the validity of criminal smart contracts. Specifically, by formalizing data feed parameters using random distribution, the criminal smart contracts are formulated as the state-based games, and their validity will be evaluated through state arrival probabilities. Moreover, a Q-learning algorithm is introduced

TABLE VII: Details of Chain-intelligence in intellectuality smart contract

Benifits	Ref.	Algorithm	Contributions	The main role of EI	Security Analysis	Storage Analysis
Intellectuality Improvement for Smart Contract	[152]	ML	Propose a static analysis tool to automatically learn features	Separating feature extraction from the technical difficulty of the smart contracts analysis	✓	×
	[153]	ML	Develop an automatically generated smart contracts by incorporating the BC and AI technologies	Automatically transformed the script into an equivalent program	✓	×
	[154]	ML	Develop a new public chain, Cortex, to provide the support for combining AI algorithms to smart contracts	Provide an access to the smart contract to enable it more diversified, generalized and efficient	✓	✓
	[155]	ML	Propose a smart logistics solution in the supply chain management area by incorporating smart contracts, DML and IoT-based condition monitoring	Efficiently select the smart contracts for guaranteeing the trust, traceability and accountability of the supply chain	×	×
	[156]	RL	Leverage RL algorithm to decrease the validity of criminal smart contracts for preventing criminals' illegal behaviors	Train data feed parameters for the low validity of criminal smart contracts	×	×
	[157]	ML	Design an AI-enabled smart contracts for enhancing cyber resilience, while optimizing complex exchanges of distributed energy resources	Enable the system more control and flexibility at the edges	✓	×
Threat detection against smart contract	[158]	ML	Strengthen the vulnerability detection capabilities of smart contracts	Enable BC analyze and detect defects without requiring predefined or expert knowledge	✓	×
	[159]	ML	Develop a ML classification based on the smart contracts opcodes feature	Assess and detect smart contracts vulnerability	✓	×
	[160]	DL	Present a DL-based method for clone detection, bug detection and contracts validation on smart contracts	Learn the structural code embeddings for smart contracts	✓	×
	[161]	DL	Precisely detect reentrancy bug for smart contracts	Capture essential semantic information, while controlling flow dependencies	✓	×
	[162]	ML	Present a color representation method to detect the compiler vulnerabilities of smart contracts	Learn and extract the feature of the encoded image automatically	✓	×

to train data feed parameters for the low validity of criminal smart contracts, which can greatly reduce contracts' validity and further prevent criminals' illegal behaviors. The work in [157] proposes the AI-enabled smart contract to enhance cyber resilience while optimizing complex exchanges of distributed energy resources in the electricity market. The ML-based method provides a better understanding of the data patterns between the various devices. Meanwhile, it helps the AI-enabled smart contract improve the efficiency and automate the auction among buyers and sellers in energy markets more efficiently and dynamically, enabling the system more control and flexibility at the edges.

2) *Threat detection against smart contract*: The smart contract can execute any task automatically, making the transactions stored in the BC credible. However, since smart contracts are difficult to patch while they lack the evaluation criteria, it is effortless for hackers to attack. Unlike traditional software program, smart contract code cannot be changed after its deployment, giving hackers opportunities to exploit potential problems to attack smart contracts, bringing challenges to BC maintenance. Therefore, there is a great need to promote the security of the self mechanism in the BC system. EI is a powerful tool to achieve secure contracts. Specifically, AI algorithms can strengthen the vulnerability detection capabilities of smart contract, enabling BC to analyze and detect defects without requiring predefined or expert knowledge, which will be successfully applied to the edge network [158]. ML can be combined with fuzz testing for smart contract vulnerability assessment, and this incorporation can rapidly

adapt to new unknown weaknesses [159]. Moreover, learning vector representation (structural code embeddings) for smart contracts with the DL-based approach is useful in response to bugs and exploits created by attackers [160]. This learning-based approach is able to detect a known set of vulnerabilities of the smart contract, further assisting to enhance the users' confidence in the reliability of the smart contract.

A representation of smart contract can not only semantically interrelate with each other but also critically capture essential information [161]. With the DL-based approach, the reentrancy bugs can be precisely detected after vectorizing the representation and inputting it into the models. Different from many efforts that have applied static and dynamic analyses to find smart contract vulnerabilities, the work in [162] presents a color representation method. It focuses on translating the byte-code of the contract develop language into RGB color code, then transforming them into an encoded image. Next, the encoded image is inputted to the CNN to learn and extract the feature automatically, detecting the compiler vulnerabilities of the smart contract.

Summary: in general, Chain-intelligence could leverage some learning algorithms to improve the robustness and safety of self mechanisms in the BC system, and this makes it easier for the BC to deploy on edge networks. The summaries of benefits for intellectuality smart contract of Chain-intelligence are listed in Table VII.

D. Scalable Blockchain System

Although BC has been a potential technology for improving EI performance, it is still facing significant challenges when

TABLE VIII: Details of Chain-intelligence in scalable blockchain system

Benifits	Ref.	Algorithm	Application	Performance Factors	Contributions	The main role of EI	Security Analysis
Scalable Blockchain System	[165]	DQL	NFV-MANO	Throughput, tasks processing delay, operational costs	Develop a BC-based framework for synchronizing and updating the message in NFV-MANO	Make the solution more quickly and accurately for reaching consensus among edge devices	×
	[166]	DRL	IIoT	Scalability, decentralization, security, latency	Develop a DRL-based performance optimization framework to satisfy the high throughput demands of the IIoT system	Dynamically select block producers, consensus protocol, block size and block interval to improve performance	✓

applied to edge scenarios. The main performance bottleneck of the BC system is caused by the broadcast delay of block data, which is essentially limited by the bandwidth and communication latency of the Internet, further directly degrading the compatibility and restricting the scalability of the BC system. Specifically, scalability is the critical barrier that prevents BC from being used as a generic platform. Moreover, the existing solutions addressing scalability issue only focus on improving scalability at the expense of other important features, such as decentralization, security, and latency. None of the existing enabling technologies has all the good characteristics of BC. Generally, the scalability can be measured by the following metrics [163]. i) **Throughput**: the number of transactions processed per time unit; ii) **networking**: the time for transactions to be verified; iii) **storage**: the size of BC that can be handled by the miners with limited storage capabilities.

Currently, some attempts based on EI provide assistance in improving the scalability performance of BC systems. As shown in Fig. 15, the scalable BC system supported by EI focuses on integrating learning-based algorithms to BC in edge systems, the demands that devices request could be satisfied at a faster rate.

Fig. 15: The scalable BC system supported by EI

Meanwhile, the consensus protocol of BC is conducive to synchronize messages of distributed edge systems. Taking the PBFT protocol for example, the whole consensus process in the EI-based BC system includes six phases: request,

admission, pre-prepare, prepare, commit, and reply. Then, the transactions validated by the PBFT can be recorded by BC for supporting the learning algorithms to analyze and optimize the performance of the BC system. Specifically, in the software-defined IIoT based on BC [164], the trust features of BC nodes and controllers, as well as the computing capability of the system can be formulated as a joint optimization problem. Leveraging the learning algorithms can accelerate the convergence process of the optimization problem while enhancing the searching efficiency of the resource allocation strategy and further improving the throughput of the BC. Unlike most of the existing works that only focus on throughput, the authors in [165] develop a BC-based framework for synchronizing and updating the message in distributed network function virtualization management and orchestration (NFV-MANO). Specifically, they jointly consider the BC throughput, the processing delay of computation tasks of BC and operational costs as a multi-objective optimization problem. Due to the high-dimensional actions or states of the above problem, a dueling deep Q-learning (DQL) is utilized to make the solution more fastly and accurately by approximating the value function using NNs. In addition, the authors in [166] consider the four-way tradeoff, i.e., scalability, decentralization, security, and latency, while quantifying the performance of BC systems from the above four aspects. Moreover, a DRL-based algorithm is designed to dynamically select block producers, consensus protocol, block size and block interval to improve performance, so as to handle the dynamic and large-dimensional features of IIoT systems.

Summary: Chain-intelligence integrates the learning-based algorithms into the BC systems in edge scenario to meet the diversified requirements that devices request at a faster pace. By this way, the fear of scalability bottleneck would be further eliminated. The summaries of benefits for scalable blockchain system of Chain-intelligence are listed in Table VIII.

VI. RESEARCH CHALLENGES AND FUTURE DIRECTIONS

Although the synergy of BC and EI has promising prospects, there still exist ongoing challenges that need to be considered in future research. In this section, we separately discuss open issues related to two enabling technologies, i.e., comprehensive architecture for EI-chain and improvement for Chain-intelligence.

A. Comprehensive Architecture for EI-chain

1) *Computing-Power Network Framework for Resource Allocation:* Currently, the emerging technologies, such as AI,

5G and edge computing, are accelerating the computing power proliferation from the cloud server to the distributed network devices. These new opportunities have given rise to the emerging computing frameworks that require ubiquitous connectivity and computing power while considering the evolving requirements of networking and computing [167]. As such, the computing-power network rises in response to the proper time and conditions for ubiquitous AI. It aims to connect the resources between clouds, edges and ends through the network, so as to provide more flexible high-quality AI services, while breaking down the island and monopoly of resources [168], [169]. However, the existing studies focus on resource utilization in an uncompensated manner. BC, as a trusted platform, can support reliable resource management and credible paid service for autonomous computing-power members. Nevertheless, the complexity of the computing-power network, coupled with the introduction of BC, makes resource allocation a daunting task. Therefore, how to incorporate BC into the computing-power network to provide the paid services is still a challenging issue.

2) *Integrated Networking, Storage, and Security Data Management*: The development of BC technology, and its general growth in popularity, can support data management while playing a prominent part in enabling a great variety of use cases in EI research. Concretely, there are many factors involved in data management in EI, including the end-edge-cloud collaborative network architecture, the communications and storage capabilities, privacy security, to name a few. Nevertheless, many researches merely focus on one or two of the above factors. In [170], a general architecture integrating BC and IoT systems is proposed for self-organized data management, considering the availability of edge/cloud services, the dynamics in edge/cloud environments, as well as the security of BC. However, storage issues remain in the above work when BC is integrated into edge systems as a general-purpose distributed ledger for data management. How to incorporate a range of influence factors into the data management schemes of EI, is still an open issue.

3) *Learning Markets for Distributed Model Collaboration*: As the number of edge nodes increases, the collaboration between learning agents in the EI research will become a huge challenge in the integrated system of BC and EI. Traditional learning methods are no longer suitable for large-scale distributed AI collaboration among devices in open and dynamic edge environments. To facilitate the collaboration of the learning machines, a variety of algorithms and methods have been designed [171], [172]. For example, a configurable framework for training the AI model collaboratively on the BC is proposed to make model updates more efficient [121]. However, most research efforts have paid less attention to large-scale distributed collaboration mechanisms. In [125], a scalable market mechanism, i.e., LM, is introduced for collaboration and transaction. LM leverages the smart contract to encapsulate the scalable collaboration relationships, and construct a trusted database for AI models in the whole collaboration cycle. Therefore, the collaborative learning mechanisms based on BC are desirable to develop distributed LM.

B. Improvement for Chain-intelligence

1) *Multi-role Applications for Consensus Protocol*: The benefits of Chain-intelligence stem from the fact that learning-based consensus protocols can support lightweight deployment of BC. Recently, many studies are conducted with the aim of designing the intelligentizing protocols while extending them to diverse applications. For instance, AI-Chain proposes PoL to share the local intelligence via the improved BC system [173]. This protocol regards the training process as a working puzzle. Based on this protocol, every edge node can play multiple roles, including the AI servicer and the miner. That is, the edge node can not only train AI models but also participate in the mining process [174], which improves the learning abilities of edges, making AI-Chain lightweight and further enabling AI closer to ubiquitous edge devices. Therefore, the multi-role applications for consensus protocol will be helpful for edge nodes to unleash the true potential of AI in stochastic edge environments.

2) *Balancing Interests for Incentive Mechanism*: Due to the prominent advantages of BC, cryptocurrency prices hit one after another record in recent years. To this end, a growing number of people keep joining the mining queue to scramble for interests, which makes the computing power of the whole network progressively increase and it is no longer promising for a single device to mine tokens. As such, it becomes a solution to gather the computing power of multiple miners to establish a mining pool for block rewards. The higher the probability of grabbing the bookkeeping rights and the higher the probability of getting rewards. Nonetheless, self-interested behaviors exist in the cooperative mining process resulting in uneven benefits to miners. The miner may gain more benefits by taking illegal actions. How to design the balanced-interests incentive mechanism for Chain-intelligence needs to be further studied in the future.

3) *Security Enhancements for Smart Contract*: Although the smart contract has advantages in terms of security enhancements for the Chain-intelligence system, the security vulnerabilities that are inherent in smart contracts also contribute to the difficulty of deploying Chain-intelligence in real scenarios. Existing works mainly concern about security solutions of Chain-intelligence, but has yet formed a complete comprehensive protection system. To this end, the design of smart contracts requires careful considerations:

- *Protection*: unified coding standards definition and intelligent vulnerability detection for the released codes.
- *Verification*: fine-grained detection of implementation vulnerabilities, and dynamic verification of contracts.
- *Remediation*: credible review of the execution results in smart contracts enables stakeholders to make timely representations and rulings.

Given the above, the direction in enhancing the security of smart contracts through AI techniques seems promising, but clearly calls for considerable research time and effort.

4) *Tradeoffs Between Multiple Scalability Factors*: Scalability is a major performance evaluation indicator of a Chain-

intelligence system, which focuses on the universal platform capacity for different services and applications. Currently, existing works provide many methodologies to measure the performance of scalable Chain-intelligence systems. For example, the work in [163] analyzes the scalability issue from the perspectives of throughput, storage and networking. In [166], the authors consider a four-way tradeoff, including scalability, decentralization, security, latency. Nonetheless, due to the well-known trilemma in BC, most of the attempts described above can only achieve part of features but sacrificing others in the real synergy deployment of the Chain-intelligence system. As such, how to address and balance the tradeoff between multiple scalability factors is a meaningful direction for future research.

VII. CONCLUSION

In this article, we have conducted a comprehensive survey on the synergy of EI and BC, with a focus on leveraging their complementary characteristics to make up for the limitations. To this end, we have developed the EI-chain and Chain-intelligence to explore the combined functionalities of both cutting-edge technologies. Specifically, we have given a detailed overview of the synergy of EI and BC, along with their definitions, motivations and benefits. In order to investigate the frameworks of the synergy system, we have presented the EI-chain and Chain-intelligence to realize reliable computing-power management, data administration, and model optimization at the edges, while improving the functions of BC. Further, some of the research challenges and future directions that pertain to the implementation and improvement of EI-chain and Chain-intelligence have been presented.

Overall, research on the synergy of EI and BC is in its infancy, and there are many challenges ahead. While this paper has briefly explored the technologies associated with the integrated system at a very introductory level and discussed future research that may benefit from pursuing this vision, we anticipate that this survey will increase the discussion on the synergy of EI and BC, and provide helpful current information to application developers and researchers.

REFERENCES

- [1] Ericsson, "IoT connections outlook: NB-IoT and CAT-M technologies will account for close to 45 percent of cellular IoT connections in 2024." [Online]. Available: <https://www.ericsson.com/en/mobility-report/reports/june-2019/iot-connections-outlook>
- [2] B. Heintz, A. Chandra, and R. K. Sitaraman, "Optimizing grouped aggregation in geo-distributed streaming analytics," in *Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (HPDC)*, 2015, pp. 133–144.
- [3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things J.*, vol. 3, no. 5, pp. 637–646, 2016.
- [4] E. Li, Z. Zhou, and X. Chen, "Edge intelligence: On-demand deep learning model co-inference with device-edge synergy," in *Proceedings of the 2018 Workshop on Mobile Edge Communications (MECOMM@SIGCOMM)*, 2018, pp. 31–36.
- [5] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.
- [6] "5 trends emerge in the Gartner hype cycle for emerging technologies, 2018." [Online]. Available: <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/>
- [7] Z. Zhao, K. M. Barijough, and A. Gerstlauer, "DeepThings: Distributed adaptive deep learning inference on resource-constrained IoT edge clusters," *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 37, no. 11, pp. 2348–2359, 2018.
- [8] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE Netw.*, vol. 33, no. 2, pp. 111–117, 2019.
- [9] S. Zhang, W. Li, Y. Wu, P. Watson, and A. Zomaya, "Enabling edge intelligence for activity recognition in smart homes," in *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, 2018, pp. 228–236.
- [10] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system." [Online]. Available: <https://bitcoin.org/en/bitcoin-paper>
- [11] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani, "Applications of blockchains in the internet of things: A comprehensive survey," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 2, pp. 1676–1717, 2019.
- [12] M. Vukolic, "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication," in *Open Problems in Network Security - IFIP WG 11.4 International Workshop (iNetSec)*, 2015, pp. 112–125.
- [13] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Blockchain for 5G and beyond networks: A state of the art survey," *J. Neww. Comput. Appl.*, vol. 166, p. 102693, 2020.
- [14] M. B. Mollah, J. Zhao, D. Niyato, Y. L. Guan, C. Yuen, S. Sun, K. Lam, and L. H. Koh, "Blockchain for the internet of vehicles towards intelligent transportation systems: A survey," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4157–4185, 2021.
- [15] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Integration of blockchain and cloud of things: Architecture, applications and challenges," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 4, pp. 2521–2549, 2020.
- [16] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues and challenges," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 2, pp. 1508–1532, 2019.
- [17] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. M. Leung, "Blockchain and machine learning for communications and networking systems," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 2, pp. 1392–1431, 2020.
- [18] IEC, "Edge intelligence (white paper)," 2019. [Online]. Available: <https://basecamp.iec.ch/download/iec-white-paper-edge-intelligence-en/>
- [19] X. Zhang, Y. Wang, S. Lu, L. Liu, L. Xu, and W. Shi, "OpenEI: An open framework for edge intelligence," in *39th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2019, pp. 1840–1851.
- [20] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 2, pp. 869–904, 2020.
- [21] D. Xu, T. Li, Y. Li, X. Su, S. Tarkoma, and P. Hui, "A survey on edge intelligence," *arXiv preprint arXiv:2003.12172*, 2020.
- [22] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge Intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7457–7469, 2020.
- [23] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *2018 IEEE Conference on Computer Communications (INFOCOM)*, 2018, pp. 63–71.
- [24] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *2019 IEEE International Conference on Communications ICC*, 2019, pp. 1–7.
- [25] "NVIDIA turing GPU architecture." [Online]. Available: <https://www.nvidia.com/en-us/geforce/turing/>
- [26] S. Jiang, D. He, C. Yang, C. Xu, G. Luo, Y. Chen, Y. Liu, and J. Jiang, "Accelerating mobile applications at the network edge with software-programmable fpgas," in *2018 IEEE Conference on Computer Communications (INFOCOM)*, 2018, pp. 55–62.
- [27] N. P. Jouppi, C. Young, N. Patil, D. A. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, and N. Boden, "In-datacenter performance analysis

- of a tensor processing unit,” in *Proceedings of the 44th Annual International Symposium on Computer Architecture (ISCA)*, 2017, pp. 1–12.
- [28] “Huawei Atlas: AI computing solution.” [Online]. Available: <https://e.huawei.com/en/products/servers/ascend>
- [29] “Latest jetson products.” [Online]. Available: <https://developer.nvidia.com/buy-jetson>
- [30] D. Bernstein, “Containers and cloud: From LXC to docker to kubernetes,” *IEEE Cloud Comput.*, vol. 1, no. 3, pp. 81–84, 2014.
- [31] “OpenEdge.” [Online]. Available: <https://www.progress.com/openedge>
- [32] M. S. H. Abad, E. Ozfatura, D. Gündüz, and Ö. Erçetin, “Hierarchical federated learning across heterogeneous cellular networks,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8866–8870.
- [33] Z. Chang, L. Lei, Z. Zhou, S. Mao, and T. Ristaniemi, “Learn to Cache: Machine learning for network edge caching in the big data era,” *IEEE Wirel. Commun.*, vol. 25, no. 3, pp. 28–35, 2018.
- [34] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, “In-edge AI: intelligentizing mobile edge computing, caching and communication by federated learning,” *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, 2019.
- [35] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, “Optimal and scalable caching for 5G using reinforcement learning of space-time popularities,” *IEEE J. Sel. Top. Signal Process.*, vol. 12, no. 1, pp. 180–190, 2018.
- [36] L. Huang, S. Bi, and Y. A. Zhang, “Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks,” *IEEE Trans. Mob. Comput.*, vol. 19, no. 11, pp. 2581–2593, 2020.
- [37] L. Shao, F. Zhu, and X. Li, “Transfer learning for visual categorization: A survey,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 26, no. 5, pp. 1019–1034, 2015.
- [38] N. H. Tran, W. Bao, A. Y. Zomaya, M. N. H. Nguyen, and C. S. Hong, “Federated learning over wireless networks: Optimization model design and analysis,” in *2019 IEEE Conference on Computer Communications (INFOCOM)*, 2019, pp. 1387–1395.
- [39] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y. Liang, Q. Yang, D. Niyato, and C. Miao, “Federated learning in mobile edge networks: A comprehensive survey,” *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.
- [40] D. Narayanan, A. Harlap, A. Phanishayee, V. Seshadri, N. R. Devanur, G. R. Ganger, P. B. Gibbons, and M. Zaharia, “PipeDream: generalized pipeline parallelism for DNN training,” in *Proceedings of the 27th ACM Symposium on Operating Systems Principles (SOSP)*, 2019, pp. 1–15.
- [41] M. Blot, D. Picard, M. Cord, and N. Thome, “Gossip training for deep learning,” *arXiv preprint arXiv:1611.09726*, 2016.
- [42] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.
- [43] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size,” *arXiv preprint arXiv:1602.07360*, 2016.
- [44] X. Zhang, X. Zhou, M. Lin, and J. Sun, “ShuffleNet: An extremely efficient convolutional neural network for mobile devices,” in *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6848–6856.
- [45] A. Kusupati, M. Singh, K. Bhatia, A. Kumar, P. Jain, and M. Varma, “FastGRNN: A fast, accurate, stable and tiny kilobyte sized gated recurrent neural network,” *arXiv preprint arXiv:1901.02358*, 2019.
- [46] S. Yao, Y. Zhao, A. Zhang, L. Su, and T. F. Abdelzaher, “DeepIoT: Compressing deep neural network structures for sensing systems with a compressor-critic framework,” in *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems (SenSys)*, 2017, pp. 1–14.
- [47] S. Han, H. Mao, and W. J. Dally, “Deep Compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding,” *Fiber*, vol. 56, no. 4, pp. 3–7, 2015.
- [48] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, “A survey of model compression and acceleration for deep neural networks,” *arXiv preprint arXiv:1710.09282*, 2017.
- [49] S. Teerapittayanon, B. McDanel, and H. T. Kung, “BranchyNet: Fast inference via early exiting from deep neural networks,” *arXiv preprint arXiv:1709.01686*, 2017.
- [50] C. Lo, Y. Su, C. Lee, and S. Chang, “A dynamic deep neural network design for efficient workload allocation in edge computing,” in *2017 IEEE International Conference on Computer Design (ICCD)*, 2017, pp. 273–280.
- [51] D. Stamoulis, T. R. Chin, A. K. Prakash, H. Fang, S. Sajja, M. Bogdan, and D. Marculescu, “Designing adaptive neural networks for energy-constrained image classification,” in *Proceedings of the International Conference on Computer-Aided Design (ICCAD)*, 2018, p. 23.
- [52] J. Mao, Z. Yang, W. Wen, C. Wu, L. Song, K. W. Nixon, X. Chen, H. Li, and Y. Chen, “MeDNN: A distributed mobile system with enhanced partition and deployment for large-scale DNNs,” in *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2017, pp. 751–756.
- [53] C. Hu, W. Bao, D. Wang, and F. Liu, “Dynamic adaptive DNN surgery for inference acceleration on the edge,” in *2019 IEEE Conference on Computer Communications (INFOCOM)*, 2019, pp. 1423–1431.
- [54] S. Jain, J. Jiang, Y. Shu, G. Ananthanarayanan, and J. Gonzalez, “ReX-Cam: Resource-efficient, cross-camera video analytics at enterprise scale,” *arXiv preprint arXiv:1811.01268*, 2018.
- [55] J. Wang, Z. Feng, Z. Chen, S. A. George, M. Bala, P. Pillai, S. Yang, and M. Satyanarayanan, “Bandwidth-efficient live video analytics for drones via edge computing,” in *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, 2018, pp. 159–173.
- [56] M. Belotti, N. Božić, G. Pujolle, and S. Secci, “A vademecum on blockchain technologies: When, which, and how,” *IEEE Commun. Surv. Tutorials*, vol. 21, no. 4, pp. 3796–3838, 2019.
- [57] Y. Lu, “The blockchain: State-of-the-art and research challenges,” *J. Ind. Inf. Integration*, vol. 15, pp. 80–90, 2019.
- [58] Y. Yuan and F. Wang, “Towards blockchain-based intelligent transportation systems,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 2663–2668.
- [59] G. Cui, K. Shi, Y. Qin, L. Liu, B. Qi, and B. Li, “Application of block chain in multi-level demand response reliable mechanism,” in *2017 3rd International Conference on Information Management (ICIM)*, 2017, pp. 337–341.
- [60] X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba, “A taxonomy of blockchain-based systems for architecture design,” in *2017 IEEE International Conference on Software Architecture (ICSA)*, April 2017, pp. 243–252.
- [61] V. Buterin, “A next generation smart contract and decentralized application platform,” 2014. [Online]. Available: <https://github.com/ethereum/wiki/wiki/White-Paper>
- [62] M. Castro and B. Liskov, “Practical byzantine fault tolerance and proactive recovery,” *ACM Trans. Comput. Syst.*, vol. 20, no. 4, pp. 398–461, 2002.
- [63] A. E. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, “Hawk: The blockchain model of cryptography and privacy-preserving smart contracts,” in *IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 839–858.
- [64] M. Wu, K. Wang, X. Cai, S. Guo, M. Guo, and C. Rong, “A comprehensive survey of blockchain: From theory to IoT applications and beyond,” *IEEE Internet of Things J.*, vol. 6, no. 5, pp. 8114–8154, 2019.
- [65] H. Dai, Z. Zheng, and Y. Zhang, “Blockchain for internet of things: A survey,” *IEEE Internet of Things J.*, vol. 6, no. 5, pp. 8076–8094, 2019.
- [66] M. Liu, H. Jiang, J. Chen, A. Badokhon, X. Wei, and M. Huang, “A collaborative privacy-preserving deep learning system in distributed mobile environment,” in *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2016, pp. 192–197.
- [67] J. Huang, L. Kong, G. Chen, M. Wu, X. Liu, and P. Zeng, “Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 6, pp. 3680–3689, 2019.
- [68] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y. Liang, and D. I. Kim, “Incentive design for efficient federated learning in mobile networks: A contract theory approach,” in *IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, 2019, pp. 1–5.
- [69] 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.
- [70] C. Qiu, H. Yao, F. R. Yu, C. Jiang, and S. Guo, “A service-oriented permissioned blockchain for the internet of things,” *IEEE Trans. Serv. Comput.*, vol. 13, no. 2, pp. 203–215, 2020.
 - [71] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An overview of blockchain technology: Architecture, consensus, and future trends,” in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, pp. 557–564.
 - [72] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, “When mobile blockchain meets edge computing,” *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, 2018.
 - [73] M. Möser, R. Böhme, and D. Breuker, “An inquiry into money laundering tools in the bitcoin ecosystem,” in *2013 APWG eCrime Researchers Summit*, 2013, pp. 1–14.
 - [74] M. Bartoletti, B. Pes, and S. Serusi, “Data mining for detecting bitcoin ponzi schemes,” in *Crypto Valley Conference on Blockchain Technology (CVCBT)*, 2018, pp. 75–84.
 - [75] B. Jiang, Y. Liu, and W. K. Chan, “ContractFuzzer: fuzzing smart contracts for vulnerability detection,” in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering (ASE)*, 2018, pp. 259–269.
 - [76] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F. Wang, “Blockchain-enabled smart contracts: Architecture, applications, and future trends,” *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 49, no. 11, pp. 2266–2277, 2019.
 - [77] M. Conti, S. K. E. C. Lal, and S. Ruj, “A survey on security and privacy issues of Bitcoin,” *IEEE Commun. Surv. Tutorials*, vol. 20, no. 4, pp. 3416–3452, 2018.
 - [78] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, “Solutions to scalability of blockchain: A survey,” *IEEE Access*, vol. 8, pp. 16 440–16 455, 2020.
 - [79] X. Chen, L. Pu, L. Gao, W. Wu, and D. Wu, “Exploiting massive D2D collaboration for energy-efficient mobile edge computing,” *IEEE Wirel. Commun.*, vol. 24, no. 4, pp. 64–71, 2017.
 - [80] H. Yao, T. Mai, J. Wang, Z. Ji, C. Jiang, and Y. Qian, “Resource trading in blockchain-based industrial internet of things,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 6, pp. 3602–3609, 2019.
 - [81] A. Asheralieva and D. Niyato, “Learning-based mobile edge computing resource management to support public blockchain networks,” *IEEE Trans. Mob. Comput.*, vol. 20, no. 3, pp. 1092–1109, 2021.
 - [82] A. Asheralieva and D. Niyato, “Distributed dynamic resource management and pricing in the iot systems with blockchain-as-a-service and uav-enabled mobile edge computing,” *IEEE Internet of Things J.*, vol. 7, no. 3, pp. 1974–1993, 2020.
 - [83] Z. Xie, R. Wu, M. Hu, and H. Tian, “Blockchain-enabled computing resource trading: A deep reinforcement learning approach,” in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020, pp. 1–8.
 - [84] S. Fan, H. Zhang, Y. Zeng, and W. Cai, “Hybrid blockchain-based resource trading system for federated learning in edge computing,” *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2252–2264, 2021.
 - [85] N. C. Luong, Z. Xiong, P. Wang, and D. Niyato, “Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach,” in *2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–6.
 - [86] N. C. Luong, Y. Jiao, P. Wang, D. Niyato, D. I. Kim, and Z. Han, “A machine-learning-based auction for resource trading in fog computing,” *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 82–88, 2020.
 - [87] S. Yu, X. Chen, Z. Zhou, X. Gong, and D. Wu, “When deep reinforcement learning meets federated learning: Intelligent multitimescale resource management for multiaccess edge computing in 5g ultradense network,” *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2238–2251, 2021.
 - [88] X. Fu, F. R. Yu, J. Wang, Q. Qi, and J. Liao, “Resource allocation for blockchain-enabled distributed network function virtualization (NFV) with mobile edge cloud (MEC),” in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 1–6.
 - [89] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, “Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing,” *IEEE Trans. Wirel. Commun.*, vol. 19, no. 3, pp. 1689–1703, 2020.
 - [90] L. Xiao, Y. Ding, D. Jiang, J. Huang, D. Wang, J. Li, and H. Vincent Poor, “A reinforcement learning and blockchain-based trust mechanism for edge networks,” *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5460–5470, 2020.
 - [91] M. Li, F. R. Yu, P. Si, W. Wu, and Y. Zhang, “Resource optimization for delay-tolerant data in blockchain-enabled IoT with edge computing: A deep reinforcement learning approach,” *IEEE Internet of Things J.*, vol. 7, no. 10, pp. 9399–9412, 2020.
 - [92] S. Guo, Y. Dai, S. Xu, X. Qiu, and F. Qi, “Trusted cloud-edge network resource management: Drl-driven service function chain orchestration for iot,” *IEEE Internet of Things J.*, vol. 7, no. 7, pp. 6010–6022, 2020.
 - [93] Y. He, Y. Wang, C. Qiu, Q. Lin, J. Li, and Z. Ming, “Blockchain-based edge computing resource allocation in IoT: A deep reinforcement learning approach,” *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2226–2237, 2021.
 - [94] J. Feng, F. Richard Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, “Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: A deep reinforcement learning approach,” *IEEE Internet of Things J.*, vol. 7, no. 7, pp. 6214–6228, 2020.
 - [95] X. Zheng, M. Li, Y. Chen, J. Guo, M. Alam, and W. Hu, “Blockchain-based secure computation offloading in vehicular networks,” *IEEE Trans. Intell. Transp. Syst. (Early Access)*, 2020.
 - [96] H. Liao, Y. Mu, Z. Zhou, M. Sun, Z. Wang, and C. Pan, “Blockchain and learning-based secure and intelligent task offloading for vehicular fog computing,” *IEEE Trans. Intell. Transp. Syst. (Early Access)*, 2020.
 - [97] Z. Li, M. Xu, J. Nie, J. Kang, W. Chen, and S. Xie, “NOMA-enabled cooperative computation offloading for blockchain-empowered internet of things: A learning approach,” *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2364–2378, 2021.
 - [98] L. Yang, M. Li, P. Si, R. Yang, E. Sun, and Y. Zhang, “Energy-efficient resource allocation for blockchain-enabled industrial internet of things with deep reinforcement learning,” *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2318–2329, 2021.
 - [99] X. Qiu, L. Liu, W. Chen, Z. Hong, and Z. Zheng, “Online deep reinforcement learning for computation offloading in blockchain-empowered mobile edge computing,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8050–8062, 2019.
 - [100] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, “Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching,” *IEEE Internet of Things J.*, vol. 7, no. 10, pp. 9441–9455, 2020.
 - [101] Z. Ning, Y. Feng, M. Collotta, X. Kong, X. Wang, L. Guo, X. Hu, and B. Hu, “Deep learning in edge of vehicles: Exploring trirelationship for data transmission,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 10, pp. 5737–5746, 2019.
 - [102] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, “Making knowledge tradable in edge-ai enabled iot: A consortium blockchain-based efficient and incentive approach,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 12, pp. 6367–6378, 2019.
 - [103] H. Chai, S. Leng, Y. Chen, and K. Zhang, “A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles,” *IEEE Trans. Intell. Transp. Syst. (Early Access)*, 2020.
 - [104] L. Zhu, H. Dong, M. Shen, and K. Gai, “An incentive mechanism using shapley value for blockchain-based medical data sharing,” in *2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, 2019, pp. 113–118.
 - [105] W. Xiong and L. Xiong, “Smart contract based data trading mode using blockchain and machine learning,” *IEEE Access*, vol. 7, pp. 102 331–102 344, 2019.
 - [106] M. Hanley and H. Tewari, “Managing lifetime healthcare data on the blockchain,” in *2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2018, pp. 246–251.
 - [107] V. Sharma, I. You, D. N. K. Jayakody, D. G. Reina, and K. R. Choo, “Neural-blockchain-based ultrareliable caching for edge-enabled uav networks,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 10, pp. 5723–5736, 2019.
 - [108] Y. Dai, D. Xu, K. Zhang, S. Maharjan, and Y. Zhang, “Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4312–4324, 2020.

- [109] Y. Qian, Y. Jiang, L. Hu, M. S. Hossain, M. Alrashoud, and M. Al-Hammadi, "Blockchain-based privacy-aware content caching in cognitive internet of vehicles," *IEEE Netw.*, vol. 34, no. 2, pp. 46–51, 2020.
- [110] R. Zhang, F. R. Yu, J. Liu, R. Xie, and T. Huang, "Blockchain-incentivized D2D and mobile edge caching: A deep reinforcement learning approach," *IEEE Netw.*, vol. 34, no. 4, pp. 150–157, 2020.
- [111] J. Zhang, F. Zhang, X. Huang, and X. Liu, "Leakage-resilient authenticated key exchange for edge artificial intelligence," *IEEE Trans. Dependable Secur. Comput. (Early Access)*, 2020.
- [112] B. Yin, H. Yin, Y. Wu, and Z. Jiang, "FDC: A secure federated deep learning mechanism for data collaborations in the internet of things," *IEEE Internet of Things J.*, vol. 7, no. 7, pp. 6348–6359, 2020.
- [113] C. H. Liu, Q. Lin, and S. Wen, "Blockchain-enabled data collection and sharing for industrial IoT with deep reinforcement learning," *IEEE Trans. Ind. Informatics*, vol. 15, no. 6, pp. 3516–3526, 2019.
- [114] M. A. Khan, S. Abbas, A. Rehman, Y. Saeed, A. Zeb, M. I. Uddin, N. Nasser, and A. Ali, "A machine learning approach for blockchain-based smart home networks security," *IEEE Netw. (Early Access)*, 2020.
- [115] M. Keshk, B. Turnbull, N. Moustafa, D. Vatsalan, and K. R. Choo, "A privacy-preserving-framework-based blockchain and deep learning for protecting smart power networks," *IEEE Trans. Ind. Informatics*, vol. 16, no. 8, pp. 5110–5118, 2020.
- [116] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informatics*, vol. 16, no. 6, pp. 4177–4186, 2020.
- [117] X. Zheng, R. R. Mukkamala, R. Vatrappu, and J. Ordieres-Mere, "Blockchain-based personal health data sharing system using cloud storage," in *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2018, pp. 1–6.
- [118] J. Kim and S. Moon, "Blockchain-based edge computing for deep neural network applications," in *Proceedings of the Workshop on INTElligent Embedded Systems Architectures and Applications (INTESA@ESWEEK)*, 2018, pp. 53–55.
- [119] H. Kim, J. Park, M. Bennis, and S. Kim, "Blockchain on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, 2020.
- [120] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things J.*, vol. 6, no. 6, pp. 10700–10714, 2019.
- [121] J. D. Harris and B. Waggoner, "Decentralized and collaborative ai on blockchain," in *2019 IEEE International Conference on Blockchain (Blockchain)*, 2019, pp. 368–375.
- [122] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Low-latency federated learning and blockchain for edge association in digital twin empowered 6G networks," *IEEE Trans. Ind. Informatics*, vol. 17, no. 7, pp. 5098–5107, 2021.
- [123] R. Doku and D. B. Rawat, "IFLBC: on the edge intelligence using federated learning blockchain network," in *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, 2020, pp. 221–226.
- [124] C. Korkmaz, H. E. Kocas, A. Uysal, A. Masry, Ö. Özkasap, and B. Akgün, "Chain FL: decentralized federated machine learning via blockchain," in *2020 Second International Conference on Blockchain Computing and Applications (BCCA)*, 2020, pp. 140–146.
- [125] L. Ouyang, Y. Yuan, and F. Y. Wang, "Learning Markets: An AI collaboration framework based on blockchain and smart contracts," *IEEE Internet of Things J. (Early Access)*, 2020.
- [126] U. Majeed and C. S. Hong, "FLchain: Federated learning via MEC-enabled blockchain network," in *20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2019, pp. 1–4.
- [127] Y. Qu, L. Gao, T. H. Luan, Y. Xiang, S. Yu, B. Li, and G. Zheng, "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet of Things J.*, vol. 7, no. 6, pp. 5171–5183, 2020.
- [128] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wirel. Commun.*, vol. 27, no. 2, pp. 72–80, 2020.
- [129] Y. Zhao, J. Zhao, L. Jiang, R. Tan, D. Niyato, Z. Li, L. Lyu, and Y. Liu, "Privacy-preserving blockchain-based federated learning for IoT devices," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1817–1829, 2021.
- [130] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, 2020.
- [131] L. Lyu, J. Yu, K. Nandakumar, Y. Li, X. Ma, J. Jin, H. Yu, and K. S. Ng, "Towards fair and privacy-preserving federated deep models," *IEEE Trans. Parallel Distributed Syst.*, vol. 31, no. 11, pp. 2524–2541, 2020.
- [132] H. Kim, S. Kim, J. Y. Hwang, and C. Seo, "Efficient privacy-preserving machine learning for blockchain network," *IEEE Access*, vol. 7, pp. 136481–136495, 2019.
- [133] S. Rathore and J. H. Park, "A blockchain-based deep learning approach for cyber security in next generation industrial cyber-physical systems," *IEEE Trans. Ind. Informatics*, vol. 17, no. 8, pp. 5522–5532, 2021.
- [134] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial internet of things," *IEEE Trans. Ind. Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.
- [135] C. Chenli, B. Li, Y. Shi, and T. Jung, "Energy-recycling blockchain with proof-of-deep-learning," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2019, pp. 19–23.
- [136] F. Bravo-Marquez, S. Reeves, and M. Ugarte, "Proof-of-learning: A blockchain consensus mechanism based on machine learning competitions," in *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*, 2019, pp. 119–124.
- [137] A. Baldominos and Y. Saez, "Coin.AI: A proof-of-useful-work scheme for blockchain-based distributed deep learning," *Entropy*, vol. 21, no. 8, p. 723, 2019.
- [138] C. Qiu, X. Ren, Y. Cao, and T. Mai, "Deep reinforcement learning empowered adaptivity for future blockchain networks," *IEEE Open J. Comput. Soc.*, vol. 2, pp. 99–105, 2021.
- [139] M. Salimitari, M. Joneidi, and M. Chatterjee, "AI-enabled blockchain: An outlier-aware consensus protocol for blockchain-based IoT networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.
- [140] J. Luo, F. R. Yu, Q. Chen, and L. Tang, "Blockchain-enabled software-defined industrial internet of things with deep recurrent Q-network," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.
- [141] F. R. Yu, J. Liu, Y. He, P. Si, and Y. Zhang, "Virtualization for distributed ledger technology (vDLT)," *IEEE Access*, vol. 6, pp. 25019–25028, 2018.
- [142] S. Jiang, X. Li, and J. Wu, "Multi-leader multi-follower stackelberg game in mobile blockchain mining," *IEEE Trans. Mob. Comput. (Early Access)*, 2020.
- [143] Y. Gao, W. Wu, H. Nan, Y. Sun, and P. Si, "Deep reinforcement learning based task scheduling in mobile blockchain for IoT applications," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.
- [144] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 4, pp. 2536–2549, 2020.
- [145] M. Zhdanova, J. Repp, R. Rieke, C. Gaber, and B. Hemery, "No smurfs: Revealing fraud chains in mobile money transfers," in *Ninth International Conference on Availability, Reliability and Security (ARES)*, 2014, pp. 11–20.
- [146] M. A. Harlev, H. S. Yin, K. C. Langenheldt, R. R. Mukkamala, and R. Vatrappu, "Breaking Bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning," in *51st Hawaii International Conference on System Sciences (HICSS)*, 2018, pp. 1–10.
- [147] P. Monamo, V. N. Marivate, and B. Twala, "Unsupervised learning for robust bitcoin fraud detection," in *2016 Information Security for South Africa (ISSA)*, 2016, pp. 129–134.
- [148] P. M. Monamo, V. N. Marivate, and B. Twala, "A multifaceted approach to Bitcoin fraud detection: Global and local outliers," in *15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2016, pp. 188–194.
- [149] T. Pham and S. Lee, "Anomaly detection in Bitcoin network using unsupervised learning methods," *arXiv preprint arXiv: 1611.03941*, 2016.
- [150] J. Lorenz, M. I. Silva, D. Aparício, J. T. Ascensão, and P. Bizarro, "Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity," *arXiv preprint arXiv:2005.14635*, 2020.

- [151] D. Ermilov, M. Panov, and Y. Yanovich, "Automatic bitcoin address clustering," in *16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2017, pp. 461–466.
- [152] N. Ashizawa, N. Yanai, J. P. Cruz, and S. Okamura, "Eth2Vec: Learning contract-wide code representations for vulnerability detection on ethereum smart contracts," *arXiv preprint arXiv:2101.02377*, 2021.
- [153] "MATRIX technical whitepaper." [Online]. Available: <https://www.matrix.io/html/MATRIXTechnicalWhitePaper.pdf>
- [154] X. Y. Ziqi Chen, Weiyang Wang and J. Tian, "Cortex - AI on blockchain." 2017. [Online]. Available: https://cryptorating.eu/whitepapers/Cortex/Cortex_AI_on_Blockchain_EN.pdf
- [155] S. S. Arumugam, V. Umashankar, N. C. Narendra, R. Badrinath, A. P. Mujumdar, J. Holler, and A. Hernandez, "IoT enabled smart logistics using smart contracts," in *2018 8th International Conference on Logistics, Informatics and Service Sciences (LISS)*, 2018, pp. 1–6.
- [156] L. Zhang, Y. Wang, F. Li, Y. Hu, and M. H. Au, "A game-theoretic method based on Q-learning to invalidate criminal smart contracts," *Inf. Sci.*, vol. 498, pp. 144–153, 2019.
- [157] M. Mylrea, "AI enabled blockchain smart contracts: Cyber resilient energy infrastructure and IoT," in *2018 AAAI Spring Symposia (AAAI)*, 2018.
- [158] W. J. Tann, X. J. Han, S. S. Gupta, and Y. Ong, "Towards safer smart contracts: A sequence learning approach to detecting vulnerabilities," *arXiv preprint arXiv:1811.06632*, vol. abs/1811.06632, 2018.
- [159] J. Liao, T. Tsai, C. He, and C. Tien, "SoliAudit: Smart contract vulnerability assessment based on machine learning and fuzz testing," in *Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, 2019, pp. 458–465.
- [160] P. Qian, Z. Liu, Q. He, R. Zimmermann, and X. Wang, "Towards automated reentrancy detection for smart contracts based on sequential models," *IEEE Access*, vol. 8, pp. 19 685–19 695, 2020.
- [161] L. Luu, D. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2016, pp. 254–269.
- [162] T. H. Huang, "Hunting the ethereum smart contract: Color-inspired inspection of potential attacks," *arXiv preprint arXiv:1807.01868*, 2018.
- [163] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, and Y. Liu, "A survey on the scalability of blockchain systems," *IEEE Netw.*, vol. 33, no. 5, pp. 166–173, 2019.
- [164] C. Qiu, F. R. Yu, H. Yao, C. Jiang, F. Xu, and C. Zhao, "Blockchain-based software-defined industrial internet of things: A dueling deep Q-learning approach," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4627–4639, 2019.
- [165] X. Fu, F. R. Yu, J. Wang, Q. Qi, and J. Liao, "Performance optimization for blockchain-enabled distributed network function virtualization management and orchestration," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6670–6679, 2020.
- [166] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Performance optimization for blockchain-enabled industrial internet of things (IIoT) systems: A deep reinforcement learning approach," *IEEE Trans. Ind. Informatics*, vol. 15, no. 6, pp. 3559–3570, 2019.
- [167] M. Król, S. Mastorakis, D. Oran, and D. Kutscher, "Compute first networking: Distributed computing meets ICN," in *Proceedings of the 6th ACM Conference on Information-Centric Networking (ICN)*, 2019, pp. 67–77.
- [168] D. Grewe, N. Nayak, U. Ambalavanan, and S. Schildt, "Towards in-network computing infrastructures for connected vehicles," in *2 KuVS Fachgespräch "Network Softwarization"*, 2020.
- [169] S. Huang, R. Chen, Y. Li, M. Zhang, K. Lei, T. Xu, and X. Yu, "Intelligent Eco Networking (IEN) III: A shared In-network computing infrastructure towards future internet," in *2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*, 2020, pp. 47–52.
- [170] Z. Xiong, Y. Zhang, N. C. Luong, D. Niyato, P. Wang, and N. Guizani, "The best of both worlds: A general architecture for data management in blockchain-enabled internet-of-things," *IEEE Netw.*, vol. 34, no. 1, pp. 166–173, 2020.
- [171] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Communication-efficient federated learning and permissioned blockchain for digital twin edge networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2276–2288, 2021.
- [172] L. Lyu, J. Yu, K. Nandakumar, Y. Li, X. Ma, and J. Jin, "Towards fair and decentralized privacy-preserving deep learning with blockchain," *arXiv preprint arXiv:1906.01167*, 2019.
- [173] C. Qiu, H. Yao, X. Wang, N. Zhang, F. R. Yu, and D. Niyato, "AI-Chain: Blockchain energized edge intelligence for beyond 5G networks," *IEEE Netw.*, vol. 34, no. 6, pp. 62–69, 2020.
- [174] X. Wang, X. Ren, C. Qiu, Y. Cao, T. Taleb, and V. C. M. Leung, "Net-in-AI: A computing-power networking framework with adaptability, flexibility, and profitability for ubiquitous AI," *IEEE Netw.*, vol. 35, no. 1, pp. 280–288, 2021.