Scalable On-Chain and Off-Chain Blockchain for Sharing Economy in Large-Scale Wireless Networks

Ting Cai, Wuhui Chen, Kostas E. Psannis, Sotirios K. Goudos, Yang Yu, Zibin Zheng, and Shaohua Wan

Abstract

In the future sharing economy, billions of underutilized IoT devices will be deployed to enable a powerful and large-scale sharing market that produces economic, environmental, and social benefits. Given the fact that communications in numerous IoT devices through wireless links are unreliable, blockchain technology, as a promising solution, has emerged to achieve reliable and secure sharing services in a decentralized manner. However, applying blockchain in large-scale wireless networks confronts scalability challenges. This motivates us to propose a real-time, trusted data interactive, and fine-grained transaction supportable sharing framework, the core of which is a novel two-layer scaling blockchain design. In the on-chain layer, sharing-oriented sharding is employed to enable secure and efficient processing of macro-transactions on the chain. In the off-chain layer, cross-zone off-chain channels are set up to provide real-time sharing transactions with high-frequency micro-trading scenarios. Finally, a proof-of-concept case study of electric vehicle sharing data is implemented with experimental results to demonstrate the feasibility of our framework.

Introduction

Recently, the Global Alliance of Sharing Economy (GLASE) presented the strategic vision of "double 50 percent reduction 2030 initiative," that is, the reduction of global new resource consumption and labor hours by 50 percent, respectively [1]. Such endeavors emphasize the promotion of sharing and efficient utilization of global resources. Correspondingly, it will trigger an increase of applications of the Internet of Things (IoT), such as bicycle sharing, self-service supermarkets, intelligent garbage recycling stations, and supply chains with automatic payments [2].

For most smart wearables and machine-to-machine (M2M) devices in IoT networks, data are delivered via wireless communications. Due to the lack of trust and data integrity assurance from untrusted terminals, message broadcast in the IoT wireless network is not always reliable. Therefore, how to ensure reliability and security becomes a significant challenge. Blockchain technology has been exploited as a promising solution to protect data integrity, provenance, and consistency for various IoT networks [3, 4]. Undoubtedly, promotions of sharing economy in the near future will give rise to an extremely large quantity of blockchain nodes communicating through wireless IoT links. Nevertheless, scalability is an inherent limitation of blockchain, which has been widely accepted as a major barrier to large-scale blockchain applications. Taking Bitcoin as an example, the maximum throughput of blockchain is about seven transactions per second (TPS); thus, the client requires at least 10 minutes on average for a launched transaction to be included in the blockchain. In contrast, the mainstream payment technologies, like Paypal or Visa, perform about 200 TPS or more than 5000 TPS, respectively. On the other hand, massive storage, used for duplicating distributed ledgers and storing large raw data derived from numerous IoT terminals, can potentially enhance the urgency of scaling the blockchain.

Many researchers have concentrated on the improvement of scalability that powered the performance of blockchain [5, 6]. To summarize, there are two types of advancements in their explorations. On one hand, some prefer employing on-chain scaling to provide high performance for large-scale wireless networks [5]. The core idea is partitioning the blockchain nodes into multiple independent groups, called shards, where each shard processes and maintains related sub-transactions in parallel. Such on-chain scaling promises to achieve real-time communications in large-scale wireless networks by largely reducing the overheads of communication, computation, and storage.

On the other hand, some works of the literature explore *off-chain scaling* to address scalability issues [6]. For example, payment channels have been proposed to enable blockchain as a settlement network, allowing the system to process a nearly unlimited number of payment transactions off the blockchain. Thanks to such a design, high-volume transactions could be processed without committing to the blockchain; consequently, instant M2M payments become possible for the blockchain applied in large-scale wireless networks. However, off-chain scaling cannot record comprehensive transaction details, encountering the disadvantage of imperfect support for some transaction scenarios, for example, large transactions that are highly guaranteed in terms of providing rigorous security.

The large-scale wireless networks inevitably bring complexity and diversity to the sharing economy, like ultra-dense small cells, massive IoT devices, diverse demands, and varied transactions. Therefore, a good scaling solution should have capacities to support all types of transactions, that is, being supportive in handling fine-grained transactions, which include *macro-* and *micro*-transactions. As illustrated in Fig. 1, *micro-transactions* refer to transactions of a small payment amount executed frequently in a real-time manner, expecting to create on-demand and profitable business models [6], for example, hotspot sharing of WiFi connection, or M2M payments in IoT. *Macro-transactions* indicate transactions of large value, requiring the recording of all transactional details on an important business deal in which security is of high concern. Concrete examples are sharing transactions of highly valuable resources, wherein secure tracking, monitoring, and intelligent control are preferred [7].

To provide fine-grained transaction support, we exploit combinations of on-chain and off-chain scaling. Nevertheless, a direct combination will bring about new challenges; for example, atomically processing transactions across shards bring high costs with slow processing speed. As such, more sophisticated approaches that are capable of performing the verification of cross-shard transactions efficiently under a joint architecture of on-chain and off-chain are required.

In this article, we employ blockchain as the backbone architecture, and propose a *two-layer* scaling sharing framework combining on-chain and off-chain for fine-grained transaction support in large-scale wireless networks. The **Layer-One** on-chain layer, designed as sharing-oriented sharding, handles macro-transactions on the chain with security guarantees. The **Layer-Two** off-chain layer, built up cross-zone channels, allows micro-transactions to be executed in real time off the chain. With combinatorial designs, we further present protocols of automated transaction workflows on efficient validations of cross-zone transactions. Finally, a proof-of-concept prototype is implemented in a case study of data-shared electric vehicles to demonstrate the effectiveness of our work.

RELATED WORK

This section briefly discusses the related work.

Large-Scale Wireless Networks: Recent applications of large-scale wireless networks have redefined many research problems toward computing offloading [8], edge caching [9], resource allocation [10], and security and privacy protection [11]. Particularly, we leverage mobile edge computing (MEC) and caching technologies to enhance computing ability and processing power for large-scale wireless networks, thereby expanding sharing economy.

Blockchain-Empowered Sharing Transactions: Traditional centralized servers have been heavily debated since they suffer from communication overhead and single points of failure. Blockchain, as a decentralized technology, has provided direct solutions to centralized sharing architecture. Guo *et al.* [4] utilized blockchain to implement a trusted access system, achieving authenticated and collaborative sharing among multiple IoT terminals. Aloqaily *et al.* [12] deployed smart contracts on a blockchain-based energy trade framework, such that transactions could be verified and handled in real time by using encoded logics. However, existing studies simply apply the blockchain directly without adjusting its structure.

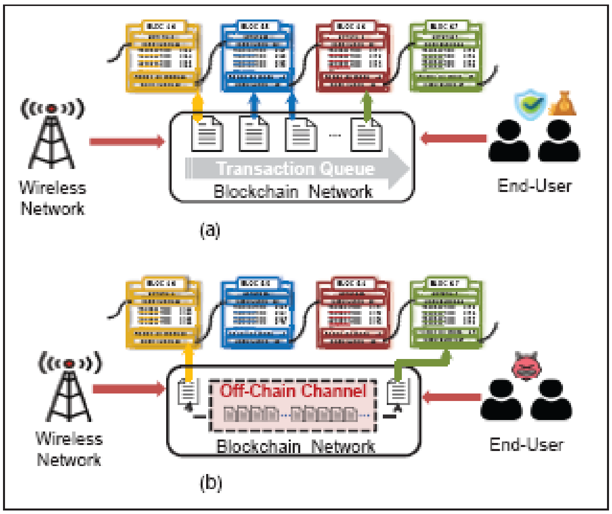

FIGURE 1. Motivation: in large-scale wireless networks. Single on-chain or off-chain scaling cannot support fine-grained sharing economy transactions. Here are illustrations of fine-grained transactions, which include macro-transactions and micro-transactions.

Scalable Designs on Blockchain: Multiple solutions to scalability have been proposed to deal with this inherent limitation of blockchain, for example, directed acyclic graph (DAG) [13], off-chain channels [6, 14], and sharding [5]. Generally, the transactions in the DAG are no longer organized as a chain structure. Sharding mainly modifies the blockchain designs to support efficient transaction processing. Off-chain channels can make agreements without all participants but with security guarantees, thus allowing users to handle small but high-frequency transactions off the chain via private communications. Nonetheless, there is a lack of combinations of on-chain and off-chain. To this end, we apply cross-shard off-chain channels to address the low-efficiency issues of cross-shard transaction verification in sharding, aiming to support all types of transactions for the sharing economy in large-scale wireless networks.

TWO-LAYER SCALING SHARING FRAMEWORK BASED ON LARGE-SCALE WIRELESS NETWORKS

FRAMEWORK OVERVIEW

Figure 2 illustrates the proposed general three-part framework, including sharing economy applications running on the top layer, an on-chain and off-chain blockchain system running in the middle, and IoT wireless networks running at the bottom.

IoT wireless networks introduce the MEC architecture to constitute a cloud-edge-terminal collaborative system, which is capable of providing powerful computing and processing capacities for large-scale wireless networks. The on-chain and off-chain blockchain system serves as the core part, enabling high TPS and low-latency processing on different types

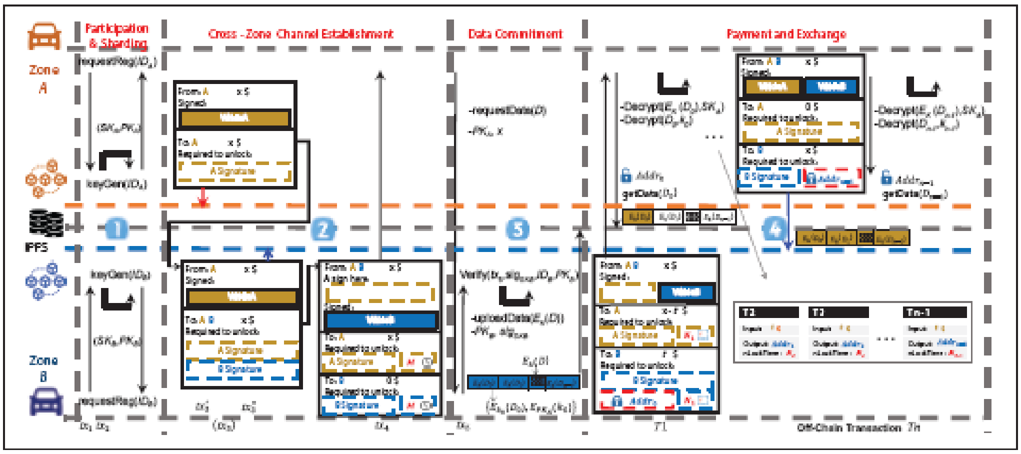

FIGURE 4. Case study implementation of ITS-data-sharing economy, wherein, a transaction processing consists of four procedures: EVs paticipation and sharding, cross-zone channel establishment, data commitment, and the payment and exchange.

station, roadside units, 5G small cells, and MEC servers; meanwhile, there are multiple dynamic participating EVs. The fixed IoT devices initially divide zones and constitute the whole blockchain network; then they act as endorsers that participate in consensus and maintain blockchains. Under these assumptions, we subsequently present a proof-of-concept implementation for our proposals, using a concrete case: a red EV *A* wants to get shared data $\mathcal{D}$ from a blue EV *B* by paying *x* dollars.

PROOF OF CONCEPT IMPLEMENTATION

To simulate fine-grained sharing transactions, we assume data $\mathcal{D}$ can be split into n chunks, that is, $\{\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_{n-1}\}$, each with valuable and independent meanings. Figure 4 illustrates the procedure details.

EV Participation and Sharding: *A* and *B* first register to join the blockchain, where an authenticator module will generate their key pairs, (SK_A, PK_A) and (SK_B, PK_B), using an efficient elliptic curve cryptography algorithm. Thereafter, they can be assigned as clients to corresponding zones; for example, *A* belongs to *Zone A* and *B* belongs to *Zone B*.

Cross-Zone Channel Establishment: In order to enable high-frequency data trading, opening up a cross-zone off-chain channel for *A* and *B* is necessary. The details are as follows:

- Step 1: *A* signs and publishes a funding transaction tx_3, sending *x* dollars via two cross-zone transactions, tx_3' and tx_3'', to a 2-of-2 multisig address of *Zone B* as deposit funds in the channel.
- Step 2: *B* creates a subsequent refund transaction tx_4 to commit that the locked *x* dollars can be unlocked by *A* after *M* nLockTime; *B* signs the half of tx_4 and then sends it to *A*.
- Step 3: If both *A* and *B* agree on the transactions and make sure tx_3'' has been recorded on the ledger of *Zone B*, tx_3' can be added to the ledger of *Zone A*. Thereafter, the off-chain channel across *Zone A* and *Zone B* is open.

Data Commitment: To purchase data from *B*, *A* needs to commit a request message, indicating wanted data $\mathcal{D}$, public key PK_A, and offered price *x*. Then *B* prepares the data by encrypting data chunks $\{\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_{n-1}\}$ with digital envelope [15]. Herein, the ciphertext pair $\{E_{k_i}(\mathcal{D}_i), E_{PK_A}(k_i)\}$, denoted by $E_k(\mathcal{D}_i)$, stands for the well-prepared data $\mathcal{D}_i$. The split data chunks, $\{\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_{n-1}\}$ are individually encrypted by using randomly generated symmetric keys like $\{k_0, k_1, \ldots, k_{n-1}\}$. Afterward, the well-encrypted datasets $\{E_k(\mathcal{D}_0), E_k(\mathcal{D}_1), \ldots, E_k(\mathcal{D}_{n-1})\}$ are prepared to be uploaded to the off-chain repositories, for example, Interplanetary File System (IPFS).

For upload, *B* needs to create a transaction tx_5, making a data commitment signed by sig_{SK_B} and waiting for the blockchain's verification from *Zone B*. Once tx_5 is confirmed, $E_k(\mathcal{D})$ can be uploaded to the IPFS.

Payment and Exchange: The next implementation offers the exchange of data storage addresses and paid dollars between *A* and *B*. Taking the payment and exchange of chunk $\mathcal{D}_0$ as an example:

- Step 1: First of all, *B* creates an update transaction *T*1, where the price is $t(t = x/n)$ and the nLockTime sets $N_1(N_1 < M)$. Then *B* adds on the data address $Addr_0$, signs it with SK_B, and sends T_1 to *A*.
- Step 2: Once received, *A* confirms *T*1 and sends it back to *B* after signing it.
- Step 3: *A* obtains $\mathcal{D}_0$'s storage address $Addr_0$, thereby downloading the encrypted data $E_k(\mathcal{D}_0)$ from the IPFS.
- Step 4: To decrypt $E_k(\mathcal{D}_0)$, *A* uses the private key SK_A to decrypt and get k_0; after that, it utilizes k_0 to decrypt $E_{k0}(\mathcal{D}_0)$ and finally obtain $\mathcal{D}_0$ in plaintext.

Thus, leveraging the update transactions, like *T*1, can effectively enable the exchange of transferring *t* dollars to *B*, thereby allowing *A* to obtain its wanted data addresses. To perform the update transactions more quickly, the nLockTime in subsequent transactions should be set smaller (i.e., $N_n < N_{n-1}$). Finally, *A* specifies storage hash values to download corresponding data from the IPFS, then retrieves the data via two decrypts: *Decrypt*($E_k(\mathcal{D})$, SK_A) and *Decrypt*($\mathcal{D}$, *K*).

The cross-zone off-chain transactions between *A* and *B* can last until the exchange of the last

Settings		Procedure 1 (ms)		Procedure 2 (ms)	Procedure 3 (ms)	Procedure 4 (ms)	
		keyGen(A)	keyGen(B)	Create channel	Encrypt and upload	Update channel	Download and decrypt
Bandwidth, delay (20, 100)	CPU: 0.5	0.14	0.16	1207.67	1722.50 ± 40.88	2527.65 ± 40.98	1732.85 ± 10.75
	CPU: 0.3	0.22	0.16	1209.17	1764.39 ± 43.06	2569.57 ± 43.15	1618.98 ± 2.92
	CPU: 0.8	0.14	0.16	1209.27	1697.38 ± 122.15	2502.69 ± 122.31	1611.22 ± 54.28
Delay, CPU (100, 0.5)	Bandwidth: 10	0.14	0.16	1208.28	1697.08 ± 121.43	2502.37 ± 121.33	1610.49 ± 54.04
	Bandwidth: 30	0.26	0.49	1214.75	1727.17 ± 145.86	2535.06 ± 145.87	1622.26 ± 54.34
CPU, bandwidth (0.5, 20)	Delay: 50	0.26	0.47	614.76	914.64 ± 73.22	1321.59 ± 73.52	824.52 ± 33.42
	Delay: 200	0.26	0.30	2414.72	3273.44 ± 236.65	4880.54 ± 236.37	3210.29 ± 109.38

TABLE 1. Performance of the proposed framework in running 100 data sharing transactions by varying the bandwidth (10 Mb/s, 20 Mb/s, 30 Mb/s), the delay (50 ms, 100 ms, 200 ms), and the CPU (0.3, 0.5, 0.8).

data chunk $E_k(\mathcal{D}_{n-1})$. The settlement transaction *Tn* comes into effect as soon as it is signed by both *A* and *B*, which can be recorded in the ledger of *Zone B*. Thereafter, the channel will be closed.

Performance Evaluation

To evaluate the efficiency and effectiveness of the proposed framework, we implement a proof-of-concept prototype in Go 1.13. We employ Mininet 2.3.0d6 to build a corresponding physical network to simulate the topology, the bandwidth and delay of links, and the CPU usage of hosts. We run the Mininet on Ubuntu 16.06 on a Dell PowerEdge T630 server to emulate different EVs-data-trading in sharing economic scenarios. Table 1 displays the results of multiple experiments running 100 data sharing transactions under all possible combinations of the bandwidth (10 Mb/s, 20 Mb/s, 30 Mb/s), delay (50 ms, 100 ms, 200 ms), and the CPU (0.3, 0.5, 0.8).

In Table 1, Procedure 4 is the largest time-consuming process mainly due to the step of updating the channel, where each round costs an average of 2.65 s on an off-chain update transaction. It seems a little high because such a time overhead contains the consumption of uploading a 100 kB data chunk to the IPFS involved in Procedure 3. Moreover, Procedure 1 uses Curve 25519 to implement signatures, which only needs approximately 0.22 ms to generate new public-private key pairs for each registered EV. In Procedure 2, opening up a cross-zone off-chain channel between two hosts shows a time approximately up to 1.28 s; however, once the channel has been established, it can be used multiple times until the channel is closed. In the experiment, the delay displays a huge amount of influence over overhead costs, whereas the bandwidth under real-world settings of 10 Mb/s, 20 Mb/s, and 30 Mb/s may influence them a little. Overall, the communication overhead in this experimental study on ITS-data-sharing economy can demonstrate the feasibility of our framework on the millisecond timescale.

We further conduct a series of experiments to make a performance comparison for three-type transaction processing approaches, as illustrated in Fig. 5. Specifically, we use the Mininet to simulate a three-tier network topology, where network switches and links are fixed while different hosts (i.e., *A*, *B*, and IPFS nodes) are randomly connected via the switches. As expected, the results in Fig. 5a indicate that the time cost of non-blockchain in total shows the best; however, it is considered to be unsafe without the guarantee of blockchains. The average time overhead for on-blockchain performs well, since it has a confirmation delay of 100 transactions in the experimental test. In contrast, our approach with combinations of on-chain sharding and off-chain channel shows superior performance in that it is as efficient as non-blockchain, and meanwhile, it also inherits security properties from the blockchain. Thanks to opening up cross-zone channels, our approach can efficiently perform the verification of massive cross-zone transactions off the blockchain; Fig. 5b shows such advantages in terms of latency. Moreover, our combinatorial approach delivers linear growth and scales well as the number of nodes increases, which is suitable for implementing shared economy services in large-scale wireless networks.

Conclusion and Open Issues

In this article, we have presented a sophisticated combinatorial blockchain architecture to enable scalable sharing economy systems that can provide real-time, trusted data interactive, and fine-grained sharing transactions for large-scale wireless networks, allowing high throughput, low latency, and massive storage. Nonetheless, there remain some open issues.

Contract-Based Complete Business Logic: In reality, a contract-based programmable framework performing complete business logic actions is needed. Then zone-based smart contracts can be developed to realize efficient sharding settings, for example, automatically determining the zone assignment of registered IoT devices. Finally, state channels should be applied to allow the mutually distrustful wireless terminals to execute off-chain sharing-defined programs among themselves.

Multi-Blockchain-Driven Platform for Heterogeneous Wireless Sharing Applications: There are limitations on empowering a platform that supports heterogeneous functions for diverse wireless sharing applications, such as cross-platform identity authentication and security assumptions. Extending sharding to multi-blockchain is an effective solution. However, since parameters vary greatly in different blockchains, making communication across multiple blockchains could be difficult as well as causing some security issues, which requires in-depth study.

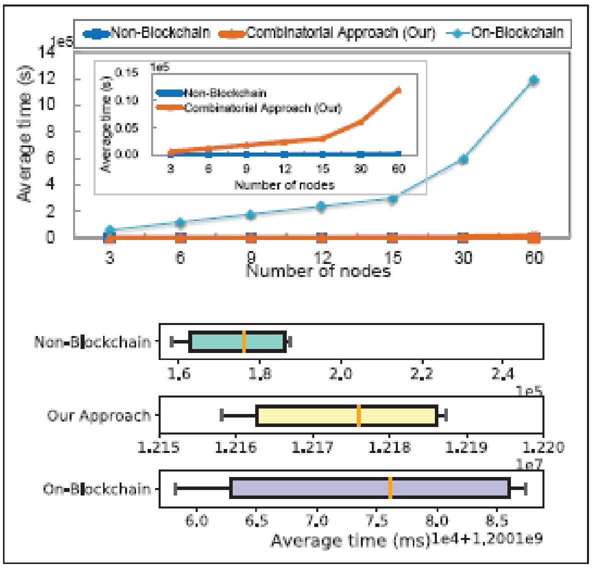

FIGURE 5. Comparison of three-type transaction supportable approaches, which are on-blockchain: on-chain processing with security guarantees; non-blockchain: off-chain processing without security guarantees; our approach: on-chain and off-chain combinations: a) average time vs. scales of networks; b) the latency of three-type networks when the number of nodes is 60.

ACKNOWLEDGMENTS

The work was supported by the National Key Research and Development Plan (2021YFB2700302), the National Natural Science Foundation of China (62172453, 62032025, 62172438), the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (2017ZT07X355), the Pearl River Talent Recruitment Program (2019QN01X130), and the Science and Technology Research Program of Chongqing Municipal Education Commission (KJZD-K201802401).

REFERENCES

[1] N. M. Barbosa *et al.*, "Designing for Trust: A Behavioral Framework for Sharing Economy Platforms," *Proc. Web Conf. 2020*, 2020, pp. 2133–43.
[2] S. Papavassiliou *et al.*, "A Paradigm Shift Toward Satisfaction, Realism and Efficiency in Wireless Networks Resource Sharing," *IEEE Network*, vol. 35, no. 1, Jan./Feb. 2020, pp. 348–55.
[3] S. Hu *et al.*, "Blockchain and Artificial Intelligence for Dynamic Resource Sharing in 6G and Beyond," *IEEE Wireless Commun.*, 2021.
[4] S. Guo *et al.*, "Blockchain Meets Edge Computing: A Distributed and Trusted Authentication System," *IEEE Trans. Industrial Informatics*, vol. 16, no. 3, 2019, pp. 1972–83.
[5] G. Wang *et al.*, "Sok: Sharding on Blockchain," *Proc. 1st ACM Conf. Advances in Financial Technologies*, 2019, pp. 41–61.
[6] L. Gudgeon *et al.*, "Sok: Off the Chain Transactions," *IACR Crypt. ePrint Arch.*, vol. 2019, 2019, p. 360.
[7] P. Li, T. Miyazaki, and W. Zhou, "Secure Balance Planning of Off-Blockchain Payment Channel Networks," *Proc. IEEE INFOCOM 2020*, 2020, pp. 1728–37.
[8] H. Guo *et al.*, "Intelligent Task Offloading in Vehicular Edge Computing Networks," *IEEE Wireless Commun.*, vol. 27, no. 4, Aug. 2020, pp. 126–32.
[9] H. Huang *et al.*, "Deep Learning for Physical-Layer 5G Wireless Techniques: Opportunities, Challenges, and Solutions," *IEEE Wireless Commun.*, vol. 27, no. 1, Feb. 2019, pp. 214–22.
[10] K. Lin *et al.*, "Data-Driven Joint Resource Allocation in Large-Scale Heterogeneous Wireless Networks," *IEEE Network*, vol. 34, no. 3, May/June 2020, pp. 163–69.
[11] A. Elgabli *et al.*, "Harnessing Wireless Channels for Scalable and Privacy-Preserving Federated Learning," *IEEE Trans. Commun.*, vol. 69, no. 8, 2021, pp. 5194–5208.
[12] M. Aloqaily *et al.*, "An Energy Trade Framework Using Smart Contracts: Overview and Challenges," *IEEE Network*, vol. 34, no. 4, July/Aug. 2020, pp. 119–25.
[13] S. Popov, "The Tangle," White paper, vol. 1, no. 3, 2018.
[14] J. Poon and T. Dryja, "The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments," 2016.
[15] R. Ganesan *et al.*, "A Novel Digital Envelope Approach for a Secure E-Commerce Channel," *Int. J. Net. Secur.*, vol. 11, no. 3, 2010, pp. 121–27.

BIOGRAPHIES

TING CAI (cait9@mail2.sysu.edu.cn) is currently pursuing a Ph.D. degree with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. From 2012 to 2018, she was a lecturer with the College of Mobile Telecom, Chongqing University of Posts and Telecom, China. Her current research interests include blockchain, deep reinforcement learning, edge/cloud computing, and the Internet of Things.

WUHUI CHEN [M] (chenwuh@mail.sysu.edu.cn) is an associate professor at Sun Yat-sen University. He received his Bachelor's degree from Northeast University, China, in 2008. He received his Master's and Ph.D. degrees from the University of Aizu, Japan, in 2011 and 2014, respectively. From 2014 to 2016, he was a JSPS research fellow in Japan. From 2016 to 2017, he was a researcher at the University of Aizu. His research interests include edge/cloud computing, cloud robotics, and blockchain.

KONSTANTINOS E. PSANNIS (kpsannis@uom.edu.gr) is an associate professor in communications systems and networking in the Department of Applied Informatics, University of Macedonia, Greece, director of the Mobility2net Research & Development & Consulting JP-EU Lab, and a member of the EU-JAPAN Centre for Industrial Cooperation. He is serving as an Associate Editor for *IEEE Access* and *IEEE Communications Letters*. He was TPC Co-Chair of ICCCI 2021 in Japan and Conference Chair of WSCE 2021 in Greece. He has also been included in the list of the Top 2 percent of influential researchers globally (prepared by scientists from Stanford University), October 2020 and October 2021.

SOTIRIOS K. GOUDOS [SM] (sgoudo@physics.auth.gr) received his B.S. degree in physics, M.S. degree in electronics, and Ph.D. degree in physics from Aristotle University of Thessaloniki, Greece, in 1991, 1994, and 2001, respectively, a Master's degree in information systems from the University of Macedonia in 2005, and a Diploma degree in electrical and computer engineering from Aristotle University of Thessaloniki in 2011. In 2013, he joined the Department of Physics, Aristotle University of Thessaloniki, where he is currently an associate professor. His research interests include antenna and microwave structure design, evolutionary algorithms, wireless communications, and semantic web technologies.

YANG YU [M] (yuy@mail.sysu.edu.cn) received a Ph.D. degree in computer science from Sun Yat-sen University in 2007. He is currently a full professor in the School of Data and Computer Science at Sun Yat-sen University. He has published more than 60 papers and has been a reviewer for several prestigious international conferences and journals. He is a Senior Member of CCF and a member of ACM. His research interests include workflow, service computing, cloud computing, and software engineering.

ZIBIN ZHENG [SM] (zhzibin@mail.sysu.edu.cn) received a Ph.D. degree from the Chinese University of Hong Kong in 2011. He is currently a professor in the School of Data and Computer Science at Sun Yat-sen University. He has published over 120 international journal and conference papers, including three ESI highly cited papers. According to Google Scholar, his papers have more than 9100 citations, with an H-index of 46. His research interests include blockchain, services computing, software engineering, and financial big data. He serves as a PC member for IEEE CLOUD, ICWS, SCC, ICSOC, and SOSE, and is a Senior Member of ACM.

SHAOHUA WAN [SM] (shaohua.wan@ieee.org) received his Ph.D. degree in computer software and theory from the School of Computers, Wuhan University, China, in 2010. From 2016 to 2017, he was a visiting professor with the Department of Electrical and Computer Engineering, Technical University of Munich, Germany. He is currently a professor with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen. His research interests include deep learning for the Internet of Things and edge computing.