Sharding for Blockchain based Mobile Edge Computing System: A Deep Reinforcement Learning Approach

Shijing Yuan*§, Jie Li*, Jinghao Liang†, Yuxuan Zhu†, Xiang Yu*, Jianping Chen‡, Chentao Wu*
*Department of Computer Science and Engineering, Shanghai Jiao Tong University, China
†University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, China
‡Jiangsu Key Laboratory of Intelligent Building Energy Efficiency,
Suzhou University of Science and Technology, China
§Research Center for Intelligent Network, Zhejiang Lab, China
Email: *{2019ysj,lijiecs,jh.liang, zyx-max-sjtu, yu-xiang}@sjtu.edu.cn, ‡alan@mail.usts.edu.cn, *wuct@cs.sjtu.edu.cn

GLOBECOM 2021 - 2021 IEEE Global Communications Conference | 978-1-7281-8104-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GLOBECOM46510.2021.9685883

***Abstract*—With the growth of data scale in the mobile edge computing (MEC) network, data security of the MEC network has become a burning concern. The application of blockchain technology in MEC enhances data security and privacy protection. However, throughput becomes the bottleneck of the blockchain-enabled MEC system. Hence, this paper proposes a novel hierarchical and partitioned blockchain framework to improve scalability while guaranteeing the security of partitions. Next, we model the joint optimization of throughput and security as a Markov decision process (MDP). After that, we adopt deep reinforcement learning (DRL) based algorithms to obtain the number of partitions, the size of micro blocks and the large block generation interval. Finally, we analyze the security and throughput performance of proposed schemes. Simulation results demonstrate that proposed schemes can improve throughput while ensuring the security of partitions.**

I. Introduction

The growth of intelligent devices promotes the rise of mobile services, especially in computing offloading and content caching. Mobile edge computing (MEC) can speed up the processing speed of computing tasks by offloading computing tasks to distributed edge nodes or MEC servers. It also brings the challenges of privacy disclosure and data security [1] [2] [3] [4]. As a promising technology, the integration of blockchain and mobile edge computing can provide users with secure and reliable business services [5] [6]. For example, the distributed ledger of blockchain can record content caching, spectrum allocation and computing resource allocation, providing a reliable platform for multi-party transactions [7]. In addition, the smart contract mechanism of blockchain can be used as a middle-ware to connect heterogeneous networks and provide automated mobile services for users [8] [9]. Nevertheless, in the face of the growing number of devices and transaction size, low throughput becomes the bottleneck of the integration of blockchain and mobile edge computing [10] [11] [12]. Therefore, increasing attention has been paid to the research of high-throughput blockchain system.

This work is partially sponsored by the National Key R&D Program of NSFC (No.61932014, 61972246), China (No.2018YFB0105203), Research Collaboration Grant from NII, Japan, and Zhejiang Lab's International Talent Fund for Young Professionals. Jie Li is the corresponding author.

On the one hand, some works focus on improving the scalability on and off the chain. The ways of capacity expansion on the chain include optimizing block size, block interval and partition mechanism. Xu et al., [13] propose an efficient cross-slice protocol to improve the efficiency of parallel processing. In order to extend the blockchain system linearly, Wang et al., [14] introduce an asynchronous consensus mechanism to avoid the overhead of multi-stage commit protocol. There are two ways to expand capacity under the chain: side chain technology and state channel based mechanism. The former allows Bitcoin transactions to be processed on multiple side chains, which makes the throughput of the blockchain system increase geometrically [15]. Sivaraman et al., [16] propose a state channel routing scheme based on multi-path transport protocol to achieve high throughput and traffic balance in the payment channel network (PCN).

On the other hand, the quantitative analysis of blockchain has been under the spotlight. Chen et al., [17] adopt a graph analysis to describe user's activities on Ethereum, including transfer of property, smart contract call and generation. Xiao et al., [18] propose an analytical model to evaluate the impact of blockchain network connectivity on consensus security. To obtain the appropriate block size and block frequency, Liu et al., [19] model the consensus process as Markov decision process (MDP), and then employ a deep Q-network (DQN) based algorithm to obtain the optimization strategy to improve the throughput. In [20], block size and resource allocation of the MEC system are formulated as a joint optimization problem to satisfy the time-varying channel environment.

Nevertheless, the above works mainly emphasize the design of the high throughput blockchain system and quantitative analysis of throughput. In the MEC system enabled by blockchain, the optimization of throughput and partition security of the blockchain is separated, which leads to a sub-optimal performance. Therefore, we propose a novel blockchain sharding framework based on partition and hierarchical consensus, which quantifies the security of partition and maximizes the transaction throughput under the con-

straint of a given security threshold. Additionally, to adapt to the time -varying and dynamic communication network, we adopt deep reinforcement learning (DRL) based algorithms (double-dueling Deep Q-network [21] [22], Asynchronous Advantage Actor-Critic [23] and Deep Deterministic Policy Gradient [24]) to obtain the optimization strategy of the partition number, block size and block generation interval. The main works of this paper are summarized as follows:

- We propose a blockchain sharding framework based on hierarchical and partition consensus to improve the throughput of blockchain system.
- We conduct a joint quantitative analysis on the throughput of blockchain and the security of partition mechanism.
- We adopt DRL-based algorithms to obtain the number of partitions, large block generation interval and the size of micro blocks.
- We compare the performance of the proposed scheme with typical schemes in terms of throughput and security. Simulation results show that our scheme outperforms typical schemes.

The rest of this paper is organized as follows. The system model is given in Section II. Section III describes the metric of performance analysis. In Section IV, the problem formulation is present. Next, experiments and analysis are shown in Section V. Finally, the conclusion is discussed in Section VI.

II. System Model

In this section, the system framework is introduced, and the consensus model is described.

A. System scenarios and system framework

The sharding framework of the MEC-oriented blockchain system is shown in Fig. 1. The small base stations (SBSs) equipped with the MEC server not only provide mobile services for users in the cell, but also act as the node of the blockchain system due to their abundant computational resources. To improve the efficiency of the verification and consensus process, we layer and partition the blockchain system. Specifically, it is divided into K preliminary consensus groups (PG) and a final consensus group (FG), which is equipped with the trusted execution environment (TEE). $PG_k, k \in \mathcal{K}$ accomplish the preliminary transaction consensus, package verified transactions into a micro block, and then send it to FG. After FG receiving micro blocks from PGs, they package micro blocks into a large block, and then complete the final consensus.

B. Consensus Model

Since practical Byzantine fault tolerant (PBFT) has good security in asynchronous network, it is adopted as the consensus algorithm in this paper. According to [25], the consensus process includes five stages: request, pre-prepare, prepare, commit and reply. Specifically, the client node issues a large block, other nodes audit and compare it with each other, and finally reply the auditted results and signatures to the client node.

III. Metrics of Performance Analysis

A. Security of Sharding

Security refers to the ability of blockchain system to resist Byzantine node tampering with the consensus mechanism, which is an important feature to evaluate the performance of the blockchain system. In this paper, consensus algorithm is PBFT, whose adversary model is $n \geq 3f + 1$, where n represents the number of all consensus nodes, f represents the number of malicious nodes. In other words, the proportion of malicious nodes in the network does not exceed $1/3$. Additionally, for a partitioned blockchain systems, security is also relevant to the number of partitions. Therefore, we can quantify the security of partition and the global security of blockchain system.

1) The security of partition: We divide PG into K groups, each group contains n nodes. We assume that the malicious probability of each node is p_0. We denote the discrete variable X as the number of malicious nodes in partition PG_k. According to [26], the security probability of PG_k can be quantified as $Ps(K)$

$$\begin{aligned} \mathrm{Ps(K)} &= 1 - \Pr\left[X \geq n/3\right] = 1 - \sum_{m=n/3}^{n} \Pr[X = m] \\ &= 1 - \sum_{m=n/3}^{n} \binom{n}{m} p_0^{n-m} (1 - p_0)^m, \end{aligned} \tag{1}$$

where $n = \lfloor N/K \rfloor$. By observing (1), we find that partition security is a monotonic decreasing function of K.

2) The global security: After that, we consider the partition as a large node of blockchain system. Hence, the global security probability of the proposed blockchain system $Pg(K)$ can be quantified according to partition security probability $Ps(K)$

$$\mathrm{Pg(K)} = 1 - \sum_{m=\lfloor K/3 \rfloor}^{K} \binom{K}{m} Ps(K)^{K-m} (1 - Ps(K))^m. \tag{2}$$

Next, we guarantee the security of the proposed blockchain system by imposing security constraints on the blockchain system $Pg(K) \geq 1 - 2^{-\lambda}$, where λ is the safety parameter.

B. Confirmation Time of Transactions

According to the system model in II-A, transactions are first verified by $PG_k, k \in \mathcal{K}$ to complete the preliminary consensus, then packaged into micro blocks and sent to FG, and finally packaged into large blocks by FG to complete the final consensus.

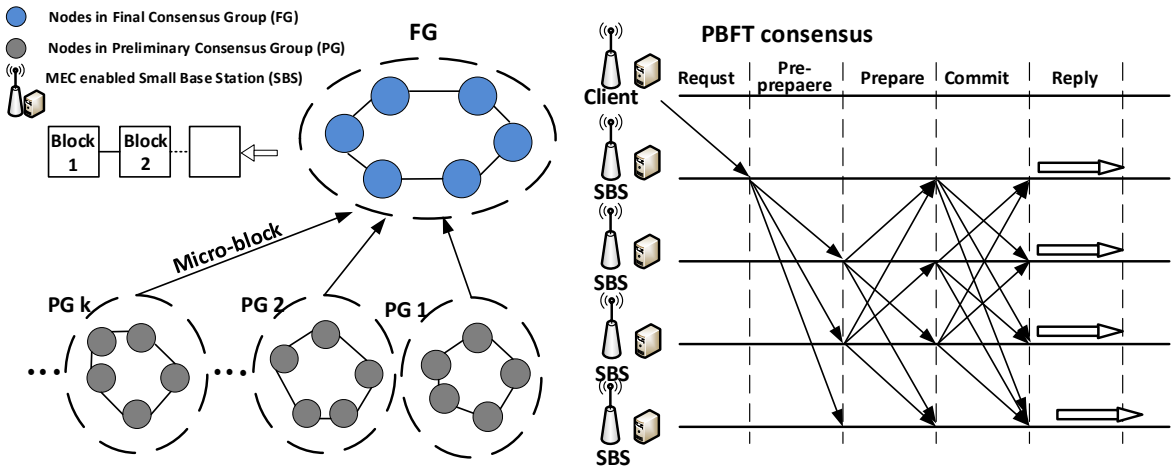

Fig. 1. The framework of the hierarchical and partitioned blockchain system.

1) The processing time of PG: The consensus time includes packing time, consensus time, and delivery time.

- Packing time: The packing time of PG_k can be calculated as

$$t_{pak_k}^{(1)} = \frac{S^M \nu}{c_k}, \tag{3}$$

where S^M denotes the size of micro block, ν is the computational resource required to process micro block per unit size, c_k represents the computational resource allocated to PG_k.

- Consensus time: The consensus time of PG_k can be calculated as [19]

$$t_{PBFT_k}^{(1)} = \begin{pmatrix} \min\{\frac{S^M}{R_{c,p}}\} + \min\{\max_{i\neq c,p} \frac{S^M}{R_{p,i}}\}+ \\ \min\{\max_{i\neq j;i,j\neq c} \frac{S^M}{R_{i,j}}\}+ \\ \min\{\max_{i\neq j} \frac{S^M}{R_{i,j}}\} + \min\{\max_{i\neq c} \frac{S^M}{R_{i,c}}\} \end{pmatrix}, \tag{4}$$

where $R_{i,j}$ denotes the transmission rate of consensus nodes in PG_k.

- Delivery time: The delivery time of PG_k can be given as

$$t_{del_k}^{(1)} = \frac{S^M}{R_{PG_k,FG}}, \tag{5}$$

where $R_{PG_k,FG}$ denotes the transmission rate between PG_k and FG. Hence, the total transaction processing time of PG can be represented as

$$T^{(1)} = max\{t_{pak_k}^{(1)} + t_{PBFT_k}^{(1)} + t_{del_k}^{(1)}\}. \tag{6}$$

2) The processing time of FG: The total transaction processing time of FG includes the block generation time and the final consensus time.

- Block generation time : FG packeges micro blocks received from PG into large blocks. The generation time of the large block is denoted as T^G.
- Final consensus time : The final consensus time $t_{PBFT}^{(2)}$ can be approximated as [19]

$$t_{PBFT}^{(2)} = \begin{pmatrix} \min\{\frac{KS^M}{R_{c,p}}\} + \min\{\max_{i\neq c,p} \frac{KS^M}{R_{p,i}}\}+ \\ \min\{\max_{i\neq j;i,j\neq c} \frac{KS^M}{R_{i,j}}\}+ \\ \min\{\max_{i\neq j} \frac{KS^M}{R_{i,j}}\} + \min\{\max_{i\neq c} \frac{KS^M}{R_{i,c}}\} \end{pmatrix}, \tag{7}$$

where $R_{i,j}$ represents the transmission rate of consensus nodes in FG. Hence, the total transaction processing time of FG can be given as

$$T^{(2)} = T^G + t_{PBFT}^{(2)}. \tag{8}$$

C. Throughput Analysis

Throughput refers to the number of transactions processed by the blockchain system per second. It can be calculated as

$$\Phi(K, S^M, T^G) = \frac{KS^M}{T^G}, \tag{9}$$

where K represents the number of partitions, T^G is the large block generation time and S^M denotes the micro block size.

IV. Problem Formulation

In this Section, partition security, block generation interval and micro block size are formulated as a joint optimization problem to simultaneously improve the throughput and security of the blockchain system. Next, three deep reinforcement learning (DRL) based algorithms are adopted to solve the non-convex optimization problem.

A. State Space

We define the state space of the blockchain system as $\mathcal{S} = \{s(t), t \in \mathcal{T}\}$, where $s(t)$ is the state of blockchain system at time period t, which contains the transmission rate

Algorithm 1 DDPG based algorithm

Initialize actor network μ and critic network Q with random parameters θ^{μ} and θ^{Q}.
Initialize target networks Q' and μ' with $\theta^{Q'} \leftarrow \theta^{Q}$, $\theta^{\mu'} \leftarrow \theta^{\mu}$.
Initialize replay buffer R.
for each $episode \in [1, episode\_max]$ **do**
  Obtain a random noise N for action exploration.
  Receive initial observation state s_1.
  for each epoch $t \in [1, epoch\_max]$ **do**
    Select action by $a_t = \mu(s_t|\theta^{\mu}) + N_t$ and act.
    Observe reward r_t and new state s_{t+1}.
    Store transition (s_t, a_t, r_t, s_{t+1}) into R.
    Sample a random minibatch R' from R.
    for each $transition \in R'$ **do**
      $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
      Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^{Q}))^2$
      Update policy with sampled policy gradient: $\Delta_{\theta^{\mu}} J = \frac{1}{N}\sum_i \Delta_a Q(s_i, a_i|\theta^{Q}) \Delta_{\theta^{\mu}} \mu(s_i|\theta^{\mu})$
      Update the target networks: $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$, $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
    end for
  end for
end for

of the wireless link $\mathcal{R} = r_{n,m}, n \neq m, n, m \in \mathcal{N}$, the computational capability of blockchain nodes $\mathcal{C} = c_n, n \in \mathcal{N}$. Therefore, $s(t)$ can be described as

$$s(t) = [\mathcal{R}, \mathcal{C}]^{(t)}. \tag{10}$$

B. Action Space

To maximize the throughput and partition security of the blockchain system, the number of partitions, block generation interval and micro-block size need to be adjusted at every time period t. Therefore, action space is defined as $\mathcal{A} = \{a(t), t \in \mathcal{T}\}$, where $a(t)$ can be represented as

$$a(t) = [K, S^M, T^G]^{(t)}. \tag{11}$$

C. Reward Function

In this paper, our goal is to maximize both partition security and throughput. Therefore, the reward function is defined as follows

$$\begin{aligned} &\mathcal{P}1: \max_{K, S^M, T^G} R(K, S^M, T^G) \\ s.t. \quad &C1: \quad 0 \leq KS^M \leq \dot{S}, \\ &C2: \quad T^{(1)} + T^{(2)} \leq \mu\, T^G, \\ &C3: \quad Ps(K) \geq 1 - 2^{-\lambda}, \end{aligned} \tag{12}$$

where C1 guarantees that the size of micro block does not exceed the upper limit of size, C2 ensures the confirmation time is bounded within the generation time of a certain number of blocks, and C3 ensures the security of each blockchain partition. Additionally, the reward function $R(K, S^M, T^G)$ represents the long term reward, which can be calculated as

$$R(S^M, K, T^G) = \sum_{t=t'}^{T} \eta^{t'-t} r(t), \tag{13}$$

where η represents the discount rate, and $r(t)$ denotes the immediate return, which is defined as the weight of global security and throughput of the blockchain system

$$r(t) = \begin{cases} \theta\Phi(K, S^M, T^G) + (1-\theta)Pg(K), & C1 - C3\ hold, \\ 0, & otherwise, \end{cases}$$

where θ is the weight coefficient, $\theta \in [0, 1]$.

Next, we adopt algorithms based on double-dueling Deep Q-Network (DQN), Asynchronous Advantage Actor-Critic (A3C) and Deep Deterministic Policy Gradient (DDPG) [22] [23] [24] to solve (10). In the application of DDPG, the action space and state space are continuous. We give the algorithm of DDPG in Algorithm 1.

D. Complexity analysis

To figure the complexity of proposed schemes, we first analyze the size of state space. There are M and L different discrete states for $\mathcal{R}$ and $\mathcal{C}$ of the state space $\mathcal{S} = [\mathcal{R}, \mathcal{C}]$. Hence, the state space size is $M^P \times L^P = (ML)^P$, where P is the number of total nodes.

Next, we consider the cases of different algorithms. In double-dueling DQN, it only uses one network. The action can be expressed as a function of the states. Therefore, the complexity is $O(\mathcal{S}) = O((ML)^P)$.

TABLE I
SIMULATION PARAMETERS

Symbol	Definition
the malicious probability of nodes in PG	p_0
the total number of nodes in PGs	N
the number of PG	K
the set of PG	$\mathcal{K}$
the size of the micro block	S^M
the large block generation time	T^G
the security probability of PG	$Ps(K)$
the global security probability	$Pg(K)$

V. EXPERIMENTS AND ANALYSIS

In this section, the proposed partitioned blockchain scaling framework is evaluated. In the simulation, the GPU version is GK210GL Tesla K80. The software environment is Pytorch v1.8.1 with Python v3.7/ TensorFlow v1.14.0 with Python v2.6 on Ubuntu 16.04.6. Other important parameters are listed in Table II.

Fig. 2 shows the convergence performance of the proposed scheme. Specifically, double-dueling DQN, A3C and DDPG converge after 300 episodes, 400 episodes and 500 episodes, respectively, which proves that the three schemes proposed have good convergence performance.

TABLE II
SIMULATION PARAMETERS

parameters	value
Learning rate(DDDQN)	0.01
Learning rate(A3C)	10^{-4}
Learning rate(DDPG)	$10^{-4}, 10^{-3}$
Replay buffer size(DDDQN)	500
Replay buffer size(DDPG)	10^5
Minibatch size(DDDQN)	32
Minibatch size(DDPG)	64
Hidden layer sizes(DDDQN)	100
Hidden layer sizes(A3C)	128, 128
Hidden layer sizes(DDPG)	128, 64
Discount factor(DDDQN, A3C)	0.9
Discount factor(DDPG)	0.99
Update iteration(A3C)	0.001
Soft target update rate(DDPG)	0.001
Maximum Episode	1000

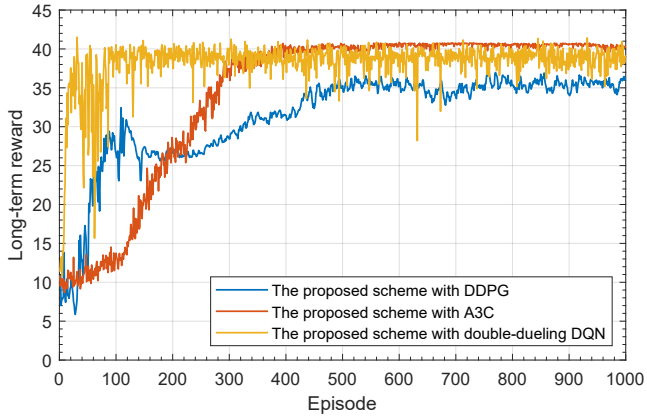

Fig. 2. Convergence of the proposed scheme.

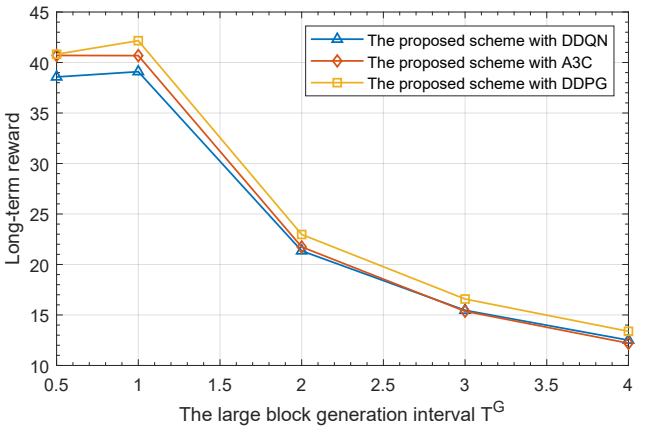

Fig. 3. The large block generation interval vs long term-reward.

Fig. 3 analyzes the impact of the generation interval T^G on the long-term reward. As can be seen from Fig. 3, the long-term reward decreases with the increase of T^G. The reason is that a larger T^G means a longer block generation and a smaller throughput of the blockchain system. In addition, compared with A3C and double-dueling DQN, the DDPG-based scheme has the best performance for the reason that it can select actions in action space accurately.

To evaluate the throughput performance of the proposed scheme, we compare the throughput with that of our scheme

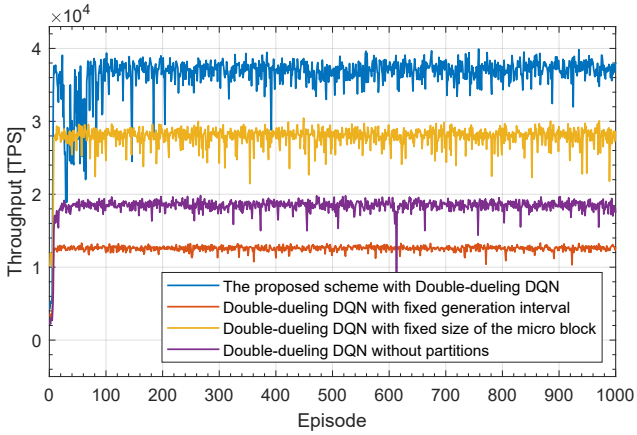

Fig. 4. Performance of throughput under different schemes.

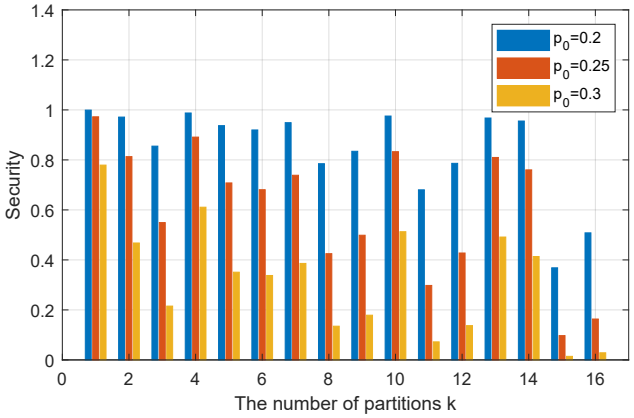

Fig. 5. The number of partitions vs Global security probability under different scales of SBSs.

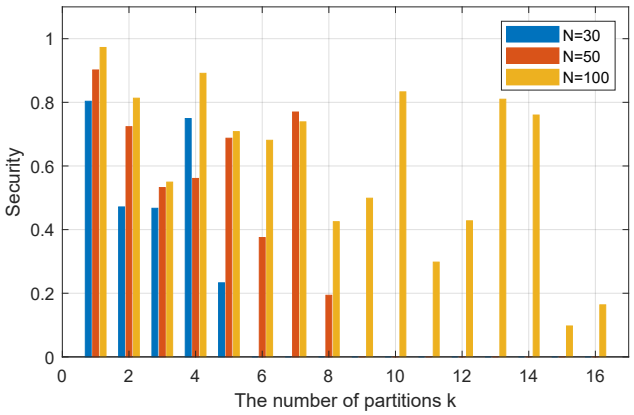

Fig. 6. The number of partitions vs Global security probability under different malicious probability.

and typical schemes. It can be seen from Fig. 4 that the proposed scheme is superior to the other three typical schemes due to the proposed scheme dynamically selects the number of partitions, the size of micro blocks and the generation interval of large blocks, and achieve the optimal throughput under the model constraint.

Fig. 5 investigates the number of partitions on the global security probability of blockchain system under different ma-

licious probabilities. We can observe that when the malicious probability is 0.2 and the number of partitions is less than 7, the proposed partition scheme has high security. Besides, with the increase of malicious probability, global security probability is reduced, as the number of nodes needed to achieve correct consensus is limited.

Fig. 6 studies the impact of the number of partitions on the global security of blockchain system under different number of SBSs. From Fig. 6, we can see that global security displays a fluctuation with a period of 3. Besides, global security doesn't decrease with the increase of the number of partitions for there are down rounding and the terms of $\lfloor n/3 \rfloor$ in (1) and (2). Therefore, the relationship between the number of partitions and the global security probability of blockchain system is not a monotonic function.

VI. Conclusion

In this work, we propose a novel partitioned blockchain framework. To improve the throughput of blockchain system, we design the consensus mechanism of layering and partitioning. Next, we jointly optimize the security and throughput of blockchain system to obtain the optimal number of partitions, micro block size and block generation interval. The proposed scheme is based on deep reinforcement learning (DRL) algorithm, which can perform adaptive partitioning for different malicious probabilities p_0 to ensure global security. Finally, we adopt three algorithms based on deep reinforcement learning to cope with the optimization problem. Simulation results show that: 1) the proposed scheme has a good convergence performance; 2) the proposed scheme can improve the throughput of the blockchain system and ensure the global security of the blockchain system.

References

[1] X. Li, S. Liu, F. Wu, S. Kumari, and J. J. Rodrigues, "Privacy preserving data aggregation scheme for mobile edge computing assisted iot applications," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4755–4763, Jun. 2019.

[2] M. Du, K. Wang, Y. Chen, X. Wang, and Y. Sun, "Big data privacy preserving in multi-access edge computing for heterogeneous internet of things," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 62–67, 2018.

[3] X. He, R. Jin, and H. Dai, "Deep pds-learning for privacy-aware offloading in mec-enabled iot," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4547–4555, 2019.

[4] P. Ranaweera, A. D. Jurcut, and M. Liyanage, "Survey on multi-access edge computing security and privacy," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1078–1124, 2021.

[5] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Bedgehealth: A decentralized architecture for edge-based iomt networks using blockchain," *IEEE Internet of Things Journal*, 2021.

[6] Y. Wu, H.-N. Dai, and H. Wang, "Convergence of blockchain and edge computing for secure and scalable iiot critical infrastructures in industry 4.0," *IEEE Internet of Things Journal*, 2020.

[7] Y.-C. Liang, "Blockchain for dynamic spectrum management," in *Dynamic Spectrum Management*. Springer, 2020, pp. 121–146.

[8] A. Refaey, K. Hammad, S. Magierowski, and E. Hossain, "A blockchain policy and charging control framework for roaming in cellular networks," *IEEE Network*, vol. 34, no. 3, pp. 170–177, 2019.

[9] F. Shi, Z. Qin, D. Wu, and J. McCann, "Mpcstoken: Smart contract enabled fault-tolerant incentivisation for mobile p2p crowd services," in *2018 IEEE 38Th international conference on distributed computing systems (ICDCS)*. IEEE, 2018, pp. 961–971.

[10] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, and Y. Liu, "A survey on the scalability of blockchain systems," *IEEE Network*, vol. 33, no. 5, pp. 166–173, 2019.

[11] M. Zamani, M. Movahedi, and M. Raykova, "Rapidchain: Scaling blockchain via full sharding," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 931–948.

[12] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proceedings of the 2019 international conference on management of data*, 2019, pp. 123–140.

[13] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "Omniledger: A secure, scale-out, decentralized ledger via sharding," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 583–598.

[14] J. Wang and H. Wang, "Monoxide: Scale out blockchains with asynchronous consensus zones," in *16th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 19)*, 2019, pp. 95–112.

[15] A. Singh, K. Click, R. M. Parizi, Q. Zhang, A. Dehghantanha, and K.-K. R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, p. 102471, 2020.

[16] V. Sivaraman, S. B. Venkatakrishnan, K. Ruan, P. Negi, L. Yang, R. Mittal, G. Fanti, and M. Alizadeh, "High throughput cryptocurrency routing in payment channel networks," in *17th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 20)*, 2020, pp. 777–796.

[17] T. Chen, Z. Li, Y. Zhu, J. Chen, X. Luo, J. C.-S. Lui, X. Lin, and X. Zhang, "Understanding ethereum via graph analysis," *ACM Transactions on Internet Technology (TOIT)*, vol. 20, no. 2, pp. 1–32, 2020.

[18] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "Modeling the impact of network connectivity on consensus security of proof-of-work blockchain," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1648–1657.

[19] M. Liu, Y. Teng, F. R. Yu, V. C. Leung, and M. Song, "Deep reinforcement learning based performance optimization in blockchain-enabled internet of vehicle," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[20] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1689–1703, 2019.

[21] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[22] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1995–2003.

[23] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1928–1937.

[24] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[25] H. Sukhwani, J. M. Martínez, X. Chang, K. S. Trivedi, and A. Rindos, "Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric)," in *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*. IEEE, 2017, pp. 253–255.

[26] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 17–30.