BIO-SD: A Blockchain-empowered Intelligent Resource Management for Symbiotic Devices

Runze Cheng, *Student Member, IEEE,* Yao Sun, *Senior Member, IEEE,* Yijing Liu, *Student Member, IEEE,* Ying-Chang Liang, *Fellow, IEEE,* and Muhammad Imran, *Fellow, IEEE*

***Abstract*—Symbiotic communication (SC), exploiting the analogy of biological ecosystem to establish communication device ecosystems, can enable cooperative service/resource exchanges across heterogeneous devices thus realizing the complementarity among different communication resources. However, considering unstable wireless links, high network dynamics, and complex electromagnetic interference in such an ecosystem, it is difficult to perform service/resource exchanges without securing a trusted environment. Moreover, multi-dimensional service/resource exchange demanded by massive symbiotic devices (SDs) in the ecosystems exposes additional challenges for exchange decision-making. To deal with the above difficulties, in this paper, we propose a blockchain-empowered intelligent coevolution for symbiotic devices (BIO-SD). Specifically, to guarantee the trustworthiness of service/resource exchange and resist malicious attacks, a direct acyclic graph (DAG)-based blockchain architecture is applied to the BIO-SD scheme. Furthermore, a modified multi-agent deep deterministic policy gradient (MADDPG) approach is adopted to make service/resource exchange decisions under this trusted environment. The simulation results show that the proposed BIO-SD scheme outperforms some conventional solutions in terms of transmission rate and transmission latency under both non-attack and malicious attack scenarios.**

***Index Terms*—Symbiotic device, Blockchain, Intelligent resource management.**

I. Introduction

THE proliferation of wireless applications has led to a rapid expansion in the number and variety of communication devices, whereas the growing complexity of electromagnetic space poses challenges to accommodate massive heterogeneous devices with limited resources. Symbiotic communication (SC), an innovative concept inspired by biology, views the communication network as an ecosystem where multi-dimensional resources like spectrum, time, energy, and computing power are rationally managed via service/resource exchanges [2]. In this ecosystem, symbiotic devices (SDs) can perform *coevolution*, i.e., cooperatively optimizing individual service/resource exchange policies in evolutionary cycles to establish firm symbiotic relationships with others. Since the service/resource boundary of devices is broken by forming symbiotic relationships, different resource bottlenecks for individual SDs can complement each other, thus improving the overall resource utilization.

However, in the absence of a trusted service/resource exchange environment, SDs are unable to fully accomplish cooperation and form firm symbiotic relationships. The SDs in the ecosystem normally belong to different network operators/radio systems without effective consensus [3], which brings difficulties in building a trusted environment for service/resource exchanges. Furthermore, even in an optimistic case of the non-attack wireless network, unreliable information exchanges can be prevalent due to unstable wireless links, high network dynamics, and complicated electromagnetic interference [4]. These unreliable information exchanges further degrade the trust among SDs when performing service/resource exchanges.

Meanwhile, even though a trusted service/resource exchange environment is secured, intelligent decision-making policies are necessitated for guiding SDs to perform rational service/resource exchanges. Given the fact that a symbiotic ecosystem may involve massive SDs with diverse design objectives in terms of latency, throughput, and reliability, it is challenging to utilize service/resource exchanges to achieve the collective objective of all SDs while balancing individual SDs' requirements [5]. Additionally, a symbiotic relationship is typically established on the basis of multiple types of services and resources, which may necessitate SDs to exchange multi-dimensional services/resources [2]. As a result, the intelligent service/resource exchange decision-making model should become more complicated.

Intuitively, interplaying blockchain and deep reinforcement learning (DRL) can overcome the aforementioned challenges, in which blockchain secures a trusted exchange environment and DRL makes the intelligent exchange policy [6]–[8]. Blockchain, serving as a decentralized, tamper-proof digital transaction ledger, harnesses the power of smart contracts and encryption to ensure the secure and automated execution of service/resource exchanges that are both traceable and transparent [9]. Through transaction verification, only validated service exchanges are recorded, bolstering resilience against malicious attacks and establishing robust symbiotic relationships. Meanwhile, a DRL model can be utilized to efficiently decide the optimal service/resource exchange for SDs [10]. With powerful neural networks, DRL can process multi-dimensional and multi-variety data generated from sev-

Preliminary results of this work have been presented in part of the IEEE International Conference of Communication (ICC), 2023 [1].

Runze Cheng, Yao Sun, and Muhammad Ali Imran are with the James Watt School of Engineering, University of Glasgow, Glasgow G12 8QQ, U.K.

Yijing Liu is with the National Key Lab on Communications, University of Electronic Science and Technology of China.

Ying-Chang Liang is with the Center for Intelligent Networking and Communications, University of Electronic Science and Technology of China, Chengdu 611731, China

Yao Sun is the corresponding author. (Email:Yao.Sun@glasgow.ac.uk)

eral SDs, and decide coevolution with diverse service/resource exchanges [11]–[13].

In this paper, we propose a blockchain-empowered intelligent coevolution for symbiotic devices (BIO-SD), which exploits direct acyclic graphs (DAG) blockchain architecture in the modified multi-agent deep deterministic policy gradient (MADDPG)-enabled SD network. Simulation results demonstrate the robustness and effectiveness of our proposed BIO-SD under both non-attack and malicious attack scenarios, and verify the performance gain of our BIO-SD scheme by comparing it with other DRL-based resource management schemes. Here, the main contributions of our study are summarized:

- Regarding the incomplete observation sharing among SDs with different network operators/radio systems, we formulate the multi-agent symbiotic resource management problem as a shared parameters partial observable Markov decision process (SP-POMDP).
- Considering the requirement of high transaction throughput with a low computing energy consumption in coevolution, we exploit a DAG-based blockchain in BIO-SD to secure a trusted environment and facilitate SDs to form symbiotic relationships.
- We develop the modified MADDPG-based decision-making method that allows SDs to cooperatively maintain individual policies and make mutually beneficial decisions for all SDs.

The remainder of this paper is organized as follows. Section II presents the system model of a symbiotic wireless network and followed by problem formulation and BIO-SD framework in Section III. Then, we illustrate the two components of BIO-SD, DAG-based blockchain and modified MADDPG decision-making policy in Sections IV and V, respectively. Section VI describes the workflow of BIO-SD scheme. Then, we evaluate the performance gain of BIO-SD by simulations in section VII. Section VIII concludes the paper.

II. System Model of Symbiotic Network

In this section, we provide a brief introduction to SC and the SD ecosystem before delving into the obligate and facultative symbiotic relationships between SDs. Then, we present the system model of a symbiotic network.

A. Preliminary: Symbiotic Device Ecosystem and Symbiotic Coevolution

Symbiotic communication is initially introduced as a concept inspired by mutualism spectrum sharing and reflection surfaces in the context of backscattering communication. Borrowing this idea, our proposed symbiotic ecosystem is expanded upon the conventional backscattering-based SC in terms of participated devices and service/resource exchanges. In our interpretation of SC, as analogous to diverse species that collectively form a communication device ecosystem, relaying devices like reflection surfaces and network access points such as base stations can all be deemed as SDs [8]. Analogous to organisms of a biological ecosystem that consume various resources, including food, light, etc., SDs in a device ecosystem consume communication resources such as spectrum, time, and energy. While species accomplish specific tasks like protection, feeding, etc., according to instinct, SDs can exchange services such as relaying and computing [2]. Similar to biological symbiosis, a typical win-win symbiotic relationship, i.e., mutualism, can be formed in such a device ecosystem, where all the participated SDs can benefit via efficiently managing resources.

Specifically, the mutualism symbiotic relationships can be classified into facultative relationships and obligate relationships [2]. In this work, the facultative relationship is that SDs cooperate to better provide service for UEs, although each SD is able to work as an independent server. Meanwhile, an obligatory relationship is a cooperation situation in which an SD cannot offer service without the assistance of other SDs. As shown in Fig. 1, there are multiple SDs from different network operators/radio systems. Either SD 1 or SD 2 can independently provide network access services, but SD 2 takes over UE 1's service request from SD 1 to provide a better service quality. This relationship is similar to the shark and remora relationship, in which both shark and remora can survive independently, but acquire cleaning service and food from cooperation, respectively. In Fig. 1, SD 3 cannot provide network access service for UE 2 without the support of SD 4, analogous to the mutualism relationship between figs and fig-wasps, in which fig-wasps help pollinate figs while getting necessary survival environment from figs.

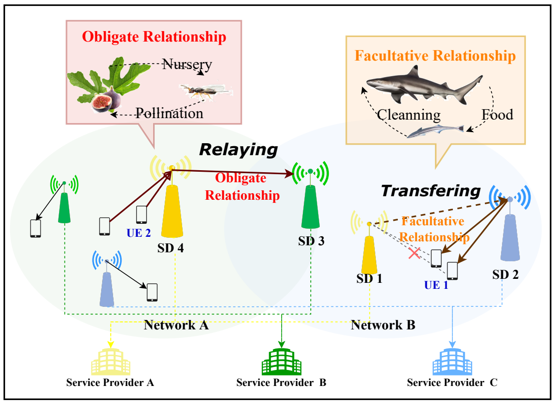


Fig. 1: System model of the symbiotic network.

Based on the benefits of symbiotic relationships, SDs, like species, are capable of stepping towards achieving coevolution. Symbiotic coevolution, in which SDs are orchestrated in a cooperative manner for service/resource exchanges, is the key to intelligent resource management in an SD ecosystem.

B. Symbiotic Network Model

We consider a symbiotic network composed of multiple SDs $\mathcal{K} = \{1, 2, ..., K\}$ and UEs $\mathcal{U} = \{1, 2, ..., U\}$, where the SDs are with different network operators/radio systems, as shown in Fig. 1. We assume UEs are associated with the nearest SD to get access service.

To achieve symbiotic coevolution, when a UE cannot get the required network access services from the current associated SD, other SDs can cooperate to provide the required services via two modes as below:

1) Transferring service: If one SD cannot provide satisfied services for UEs, it transfers the access requests to a neighbor SD (such as SD 1 and SD 2 in Fig. 1). From the perspective of a symbiotic ecosystem, SDs will establish a facultative relationship through service transfers, in which the two SDs can independently provide network access service for UEs but prefer cooperating to provide better service.
2) Relaying service: When UEs cannot directly access the required SD, another SD acts as a relay to bridge the UEs and their desired SD (like SD 3 and SD 4 in Fig. 1). For example, a drone could act as a relay for the link between a terrestrial UE and a satellite. In symbiotic coevolution, relaying services can lead to an obligate relationship where an SD can only provide service with the help of another SD.

To achieve symbiotic coevolution and cooperatively provide appropriate services for UEs, multi-hop links among SDs are considered. Meanwhile, transmission rate and latency are used as the two service performance metrics. The transmission rate of this link is represented as $W_{i,m} = B_{i,m} \cdot log_2\left(1 + \mathit{SINR}_{i,m}\right)$, where $B_{i,j}$ denotes the bandwidth allocated by SD i to receiver m, and $\mathit{SINR}_{i,m}$ is the signal-to-interference-plus-noise ratio (SINR) of the link from SD i to receiver m. [1]

For latency, basically, we assume the latency is composed of two parts, propagation delay $L^P_{i,m}$ and transmission delay $L^T_{i,m}$, i.e., $L_{i,m} = L^P_{i,m} + L^T_{i,m}$. We consider propagation delay because the symbiotic network might involve SDs within a large-scale network, e.g., the SDs from satellite networks. We denote $d_{i,m}$ as the communication distance between SD i and a receiver m. The propagation delay is denoted as $L^P_{i,m} = \frac{d_{i,m}}{\nu}$, and ν denotes the speed of light. Meanwhile, the transmission delay is $L^T_{i,m} = \frac{c_u}{W_{i,m}}$, where c_u denotes the data size requested by UE u. Since multi-hop links might exist, when SD j acts as a relay to bridge the UE u and its desired SD i, the latency of UE u accessing network service from SD i is $L_{i,u} = L_{i,j} + L_{j,u}$.

III. Problem Formulation and BIO-SD Framework

In this section, we first elaborate on the problem formulation of the service exchange in the symbiotic network. Then, we outline the framework of the proposed BIO-SD scheme.

A. Problem Formulation

Due to the limited resources available for data sharing (such as state, and action) among SDs in the network, and considering that SDs are primarily influenced by other SDs within their coverage, assuming a cooperative network environment with incomplete data sharing among partial SDs is reasonable. Therefore, we formulate the multi-agent service exchange problem as an SP-POMDP problem. In this problem, we orchestrate multiple SDs in a cooperative manner to rationally manage resources, thus providing UEs with satisfied services in accordance with diversified requirements. The action, state, reward, and objective function of this problem are defined as follows.

Action: In this multi-agent symbiotic network, each SD acts as an agent. In this work, we optimize the spectrum resource management among SDs via service exchanges (i.e., SD coevolution), while the resource allocation to UEs is decided by a heuristic policy (prioritize resources for UEs with low demand). Therefore, we denote $\mathbf{a}^t_i = \left(a^t_{i,1}, a^t_{i,2}, \cdots, a^t_{i,K}\right)$ as the action of SD i at time t, where $a^t_{i,j} \in [0,1]$ denotes the percentage of consumed bandwidth when SD i offers relaying or transferring service for SD j. The decision to offer either a relaying service or a transferring service to UEs is predetermined based on the functional characteristics of different SDs.

State: In our SP-POMDP problem, SDs can share observations with other SDs to achieve coevolution. The state is denoted as $\mathbf{s}^t_i = \left(\mathbf{B}^t_i, \mathbf{H}^t_i, \bar{\mathbf{W}}^t_i, \bar{\mathbf{L}}^t_i\right)$. Specifically, $\mathbf{B}^t_i = \left(b^t_{i,1}, b^t_{i,2}, \cdots, b^t_{i,K}\right)$ represents the set of average bandwidth provided by SD i for other SDs in the last certain time slots. Meanwhile, $\mathbf{H}^t_i = \left(h^t_{i,1}, h^t_{i,2}, \cdots, h^t_{i,K}\right)$ is denoted as the average probability of that SD i exchanges service with other SDs in the nearest certain time slots. Moreover, $\bar{\mathbf{W}}^t_i = (\bar{w}^t_1, \bar{w}^t_2, \cdots, \bar{w}^t_K)$ and $\bar{\mathbf{L}}^t_i = \left(\bar{l}^t_1, \bar{l}^t_2, \cdots, \bar{l}^t_K\right)$ are the average transmission rate requirements and the average network latency requirements received by SDs $\mathcal{K}$, respectively.

Reward: Since one common total reward cannot clearly reflect the performance of every SD in this SP-POMDP problem, we allow each SD to separately calculate the individual reward using the same reward function [14]. To evaluate the service quality in terms of transmission rate and latency, we introduce the utility functions R^W_u and R^L_u. These functions consider the transmission rate requirement $[\check{W}u, \hat{W}u]$ and the latency requirement $[\check{L}u, \hat{L}u]$, respectively.

$$R^W_u = \begin{cases} 1, & W_{i,u} > \hat{W}_u, \\ \frac{W_{i,u} - \check{W}_u}{\hat{W}_u - \check{W}_u}, & \check{W}_u < W_{i,u} < \hat{W}_u, \\ 0, & otherwise. \end{cases} \quad (1)$$

$$R^L_u = \begin{cases} 1, & L_{i,u} < \check{L}_u, \\ \frac{\hat{L}_u - L_{i,u}}{\hat{L}_u - \check{L}_u}, & \check{L}_u < L_{i,u} < \hat{L}_u, \\ 0, & otherwise. \end{cases} \quad (2)$$

In this work, to achieve coevolution, we make SDs cooperate to provide diverse services to UEs with different requirements, thus the reward is set as

$$R_i = \frac{1}{U_i}\sum_{u=1}^{U_i} p_{i,u}\lambda^W_u R^W_u + \frac{1}{U_i}\sum_{u=1}^{U_i} p_{i,u}\lambda^L_u R^L_u, \quad (3)$$

where $p_{i,u}$ is the probability that SD i provides network access service to UE u via either transferring service, relaying service, or direct access. Furthermore, λ^W_u and λ^L_u represent two weights related to UEs' requirements on the transmission rate and latency, and $\lambda^W_u + \lambda^L_u = 1$.

Objective Function: Based on the aim of rationally exchanging services to achieve intelligent resource management during symbiotic coevolution, we design our objective function of this SP-POMADP problem as

$$\underset{u \in \mathcal{U}}{\text{maximize}}\, \lambda^W_u R^W_u + \lambda^L_u R^L_u. \quad (4)$$

[1] Either another SD or a UE can be the receiver.

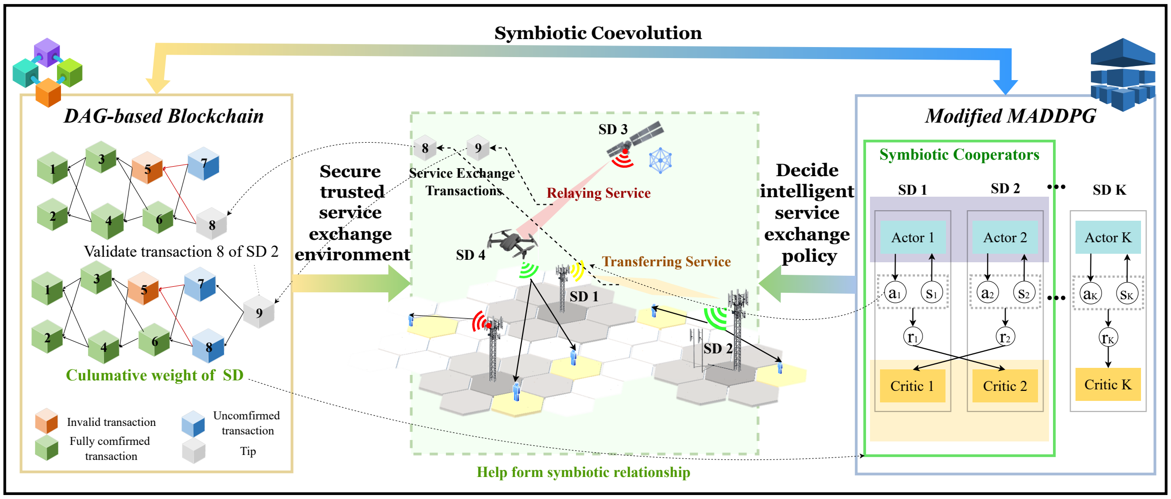


Fig. 2: Blockchain-empowered intelligent coevolution for SD.

Therefore, SDs need to decide the optimal symbiotic service exchange actions that can cooperatively fulfill all UEs' service requirements.

B. Framework of BIO-SD

To solve the aforementioned SP-POMDP, we propose a BIO-SD scheme as shown in Fig 2. Through the integration of blockchain into this framework, the states and service exchange policies of SDs can be shared in an immutable, transparent, and secure manner, facilitated by smart contracts, encryption, and transaction verification. Therefore, our BIO-SD architecture is capable of preventing malicious attacks, identifying malicious devices, and finally establishing a trusted service exchange environment. Meanwhile, with the prerequisite of a trusted environment, DRL in Fig. 2 is applied to guide SDs in cooperative learning policy and effectively make decisions according to local environments. With the intelligent service exchange decisions of BIO-SD, SDs can establish firm symbiotic relationships with appropriate neighbors. In this way, the resource bottlenecks of different SDs can complement each other, thus improving the quality of service provided to UEs. The details of BIO-SD will be elaborated in Sections IV and V for the functionality of blockchain and DRL respectively.

IV. DAG-based Blockchain for Symbiotic Network

In our BIO-SD scheme, considering the features of symbiotic networks including low computing capability, limited energy, and unaffordability of high latency, we come up with a dedicated DAG-based blockchain to secure the trusted coevolution environment with a fast consensus process. Since DAG-based blockchain does not involve complicated hash calculation, SDs can effectively execute consensus and verify transactions with less computational and energy resources [15]. Additionally, the consensus execution duration can be reduced, because the DAG-based blockchain allows multiple transactions to be broadcasted, verified, and recorded simultaneously in an asynchronous manner. Furthermore, as a partially centralized consensus that does not require the election of leader nodes and has good scalability, the DAG-based blockchain is ideal for highly dynamic symbiotic networks. In this section, to utilize the DAG-based blockchain, four key processes of the proposed blockchain are presented in more detail.

A. Encryption and Cryptographic Technology

In order to resist potential attacks, we apply cryptographic technology to our proposed DAG-based blockchain. A public key and a private key should be exploited for encryption, where each SD holds the private key (e.g., a randomly generated sequence) to compute the corresponding public key. The public key is transparent among SDs and can be disclosed to all nodes without any risk, while the private key is merely owned by an individual SD. When an SD wants to apply for symbiotic service exchange with another SD, it selects the public key of the cooperator SD [16].

B. Smart Contract

In our scheme, SDs can exchange services according to their decision-making results via a smart contract. The smart contract, as a public agreement, directly sends the bandwidth usage of service exchanges to SDs. Then, the balances of SDs' accounts are autonomously updated as per symbiotic service exchanges without third-party involvement, which ensures all the service exchanges are traceable, transparent, and irreversible.

C. Transaction Verification

The consensus process of the proposed DAG-based blockchain is shown in Fig. 3. In the proposed DAG-based blockchain, there are four types of transactions: *tips* (newly published transactions), *unconfirmed transactions* (newly validated transactions without adequate trustworthiness weights), *fully confirmed transactions* (transactions with adequate trustworthiness weights), and *invalid transactions* (old transactions with low trustworthiness weights) [15]. For transaction verification, the first several transactions, i.e., genesis transactions,

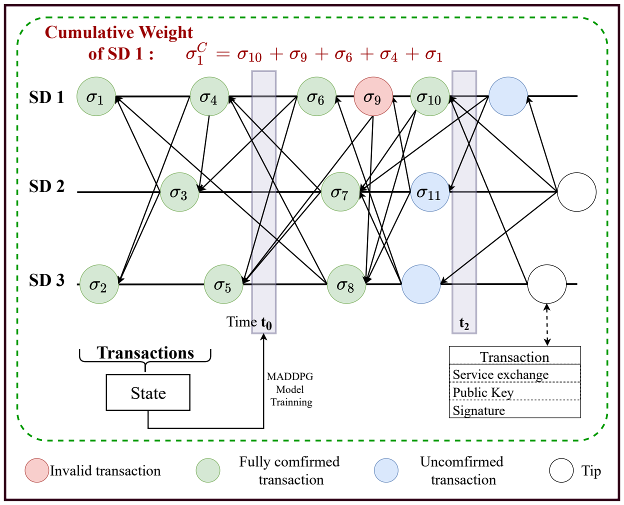


Fig. 3: DAG-based blockchain consensus.

are broadcasted as tips by using the gossip algorithm [17]. Upon verifier SDs receiving the tip τ, these verifiers check the validity of the signature and validate the service exchange within tip τ. When the verifier SDs issue other tips, the tip τ will be referred to if this tip is correct. Otherwise, the transaction will be rejected. Since complicated hash calculations are not involved, SDs can effectively verify service exchange records with fewer computational and energy resources. To avoid fake service exchange records, SDs are not permitted to validate their own service exchange records. Moreover, before an SD broadcasts its own tips (except Genesis tips), it ought to verify at least a certain number of transactions. We define a trustworthiness weight to determine the validity of transactions, i.e.,

$$\sigma_\tau = \sum_{\kappa=1}^{E_\tau} \psi_{\kappa,\tau}\sigma_{\kappa,\tau}(1 - e^{-\varphi_{\kappa,i}}), \tag{5}$$

where $\{1, 2, ..., E_\tau\}$ represents a set of tips approving transaction τ directly, and $\sigma_{\kappa,\tau}$ denotes the issued trustworthiness weight for transaction τ when broadcasting transaction κ. The more computing power consumed by tip κ's publisher SD in verifying transaction τ, the greater the transaction weight $\sigma_{\kappa,\tau}$ is. Moreover, $\psi_{\kappa,\tau}$ is an indicator, if transaction τ is verified as correct, $\psi_{\kappa,\tau} = 1$; otherwise, $\psi_{\kappa,\tau} = -1$. We denote $\varphi_{\kappa,i}$ as a discount factor, which is the credibility of transaction κ's publisher SD i. The discount factor is proportional to the cumulative weight σ_i^C of SD i. (The cumulative weight is detailed further in the final part of this section.) An unconfirmed transaction can be approved as a fully confirmed transaction only after the trustworthiness weight σ_τ is greater than a threshold $\hat{\sigma}$, as shown in Fig. 3. According to the heaviest chain rule [18], the chain with the heaviest sum weight of transactions will become the final valid chain.

In addition, to prevent lazy and malicious SDs from only validating old confirmed transactions with fake contributions, we use a transaction score to motivate SDs to verify new tips or unconfirmed transactions [18]. The transaction τ's score σ_τ^S is the total trustworthiness weight of all transactions that are directly or indirectly approved by transaction τ

$$\sigma_\tau^S = \sum_{\kappa=1}^{F_\tau} \sigma_\kappa, \tag{6}$$

where $\{1, 2, ..., F_\tau\}$ represents the set of transactions approved by transaction τ, and σ_κ denotes the transaction κ's trustworthiness weight. Therefore, new transactions, i.e., tips and unconfirmed transactions, tend to have a higher score, and SDs can verify them as a priority.

In this way, the proposed DAG-based blockchain can offer enough protection against fault service exchange records generated by malicious SDs. We give the following proposition for quantifying malicious SD tolerance.

Proposition 1. *Assuming that malicious SDs are deliberate in providing fault validation results or not responding to transaction validation, then the maximum number of tips issued by malicious SDs that can be tolerated for the proposed DAG-based blockchain should satisfy*

$$E^F < \frac{E^R}{2} - 1, \tag{7}$$

where E^R and E^F denote the number of tips issued by trusted SDs and malicious SDs, respectively.

Proof: Under the worst circumstance, assuming all E^F tips issued by malicious SDs refer to transaction τ with the incorrect opposite weight. Meanwhile, the E^R tips issued by trusted SDs are divided into two parts: the E^F trusted tips do not refer to the transaction τ and the remaining $E^R - E^F$ trusted tips refer to transaction τ with correct results. Since the $\hat{\sigma}$ is the threshold for validating the trustworthiness of transactions, according to (5), we have

$$\sum_{\kappa=1}^{E^R-E^F} \sigma_{\kappa,\tau}(1 - e^{-\varphi_{\kappa,i}}) - \sum_{\kappa=1}^{E^F} \sigma_{\kappa,\tau}(1 - e^{-\varphi_{\kappa,i}}) > \hat{\sigma}. \tag{8}$$

Moreover, if malicious SDs cannot be detected, the credibility of malicious SDs tends to be the same as trusted SDs. Therefore, the average weight of transactions τ issued by the malicious SDs is on par with that of the trusted SDs, so we let $\bar{\sigma}$ represent the average weight issued by either trusted SDs or malicious SDs for transaction τ. Thus, we have $(E^R - E^F)\bar{\sigma}_\kappa - E^F\bar{\sigma}_\kappa > \hat{\sigma}$, i.e., $E^F < \frac{1}{2}\left(E^R - \frac{\hat{\sigma}}{\bar{\sigma}_\kappa}\right)$. To ensure the trustworthiness of DAG-based blockchain, we assume a transaction needs to be validated by at least other 2 transactions, i.e., $\frac{\hat{\sigma}}{\bar{\sigma}_\kappa} > 2$, thus $E^F < \frac{E^R}{2} - 1$. ■

D. Creditable Symbiotic Cooperator Selection

A cumulative weight σ_i^C is adopted as the sum weight of all fully confirmed transactions $\mathcal{T}_i$ of SD i,

$$\sigma_i^C = \sum_{\tau=1}^{T_i} \sigma_\tau. \tag{9}$$

A higher cumulative weight indicates that an SD participated in a larger number of trusted service exchanges. Therefore, the cumulative weight of SD can be used to evaluate the trustworthiness and activeness of different SDs, as well as guide SDs to form stable and efficient symbiotic relationships with creditable neighbors. Moreover, malicious SDs with low cumulative weight can be excluded from coevolution in the symbiotic network.

Algorithm 1 Training of Modified Multi-agent Deep Deterministic Policy Gradient for SDs $\mathcal{K}_i$

```
Initialize the parameter of SD policies $\mu_i, i \in \mathcal{K}_i$, the discount rate $\gamma$, the minibatch size $n_b$, the replay buffer $\mathcal{D}_i$ of SD $i$.
for episode=1 to $M$ do
    Initialize a random process $N$ for action exploration.
    Receive initial state $\tilde{\mathbf{s}}_i = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_{K_i}\}$ for every SD.
    for time t=1 to max-episode-length do
        Execute SDs' actions $\mathbf{a}_1^t, \mathbf{a}_2^t, \cdots, \mathbf{a}_{K_i}^t$, start symbiotic service exchange.
        for SD $i$=1 to $K_i$ do
            Calculate reward $r_i^t$, observe new individual state $\mathbf{s}_i^t$.
            Share observed information such as reward and state.
        end for
        for SD $i$=1 to $K_i$ do
            Store $\left(\tilde{\mathbf{s}}_i^t, \mathbf{a}_1^t, \cdots, \mathbf{a}_{K_i}^t, r_1^t, \cdots, r_{K_i}^t, \tilde{\mathbf{s}}_i^{t+1}\right)$ in replay buffer $\mathcal{D}_i$.
            Simple a random minibatch of $n_b$ samples $\left(\tilde{\mathbf{s}}_i^m, \mathbf{a}_1^m, \cdots, \mathbf{a}_{K_i}^m, r_1^m, \cdots, r_{K_i}^m, \tilde{\mathbf{s}}_i^{m+1}\right)$ from $\mathcal{D}_i$.
            Set $y^m = r_i^m + \gamma \bar{Q}_i^{\bar{\mu}}(\tilde{\mathbf{s}}_i^{m+1}, \mathbf{a}_1^{m+1}, \cdots, \mathbf{a}_{K_i}^{m+1})|_{\mathbf{a}_j^{m+1}=\bar{\mu}_j(\mathbf{s}_j^m)}$.
            Update critic-network by minimizing the loss $Loss(\theta_i^Q) = \frac{1}{n_b}\sum_{m=1}^{n_b}\left[Q_i^{\mu}\left(\tilde{\mathbf{s}}_i^m, \mathbf{a}_1^m, \cdots, \mathbf{a}_{K_i}^m\right) - y^m\right]^2$.
            Update actor-network using the sampled policy gradient $\nabla_{\theta_i^{\mu}} J \approx \frac{1}{n_b}\sum_{m=1}^{n_b}\left[\nabla_{\theta_i^{\mu}}\mu_i(\mathbf{s}_i^m|\mathbf{a}_i^m)\nabla_{a_i}Q_i^{\mu}(\tilde{\mathbf{s}}_i^m, \mathbf{a}_1^m, \cdots, \mathbf{a}_{K_i}^m)\right]$.
        end for
        Update Target network of each SD $i$: $\bar{\theta}_i^{Q} \leftarrow \alpha\theta_i^Q + (1-\alpha)\bar{\theta}_i^{Q}$, $\bar{\theta}_i^{\mu} \leftarrow \alpha\theta_i^{\mu} + (1-\alpha)\bar{\theta}_i^{\mu}$.
    end for
end for
```

V. MADDPG-based Symbiotic Service Exchange Policy

Considering the formulated SP-POMDP which has the features of large state space, continuous action space, and unobservable partial cooperator information, we borrow the idea of [19] to propose a modified MADDPG. In the proposed modified MADDPG, each SD maintains a local DDPG model that can distributively process multi-variety states to make high-dimensional action decisions, while a cooperative training method helps multiple SDs to learn the policies of other cooperators and avoid policy conflicts.

A. Structure of Local Model

Let us start with the structure of locally maintained model for individual SD in the modified MADDPG scheme. The DDPG models are applied to deal with the large state space and decide continuous actions for SDs. For an SD, its local DDPG model includes four networks: 1) the actor-network, 2) the critic-network, 3) the target critic-network, and 4) the target actor-network [20]. We denote all the parameters of an actor-network, a critic-network, a target actor-network, and a target critic-network of SD i as θ_i^{μ}, θ_i^{Q}, $\bar{\theta}_i^{\mu}$, and $\bar{\theta}_i^{Q}$, respectively. The critic-network is trained to estimate the Q-value $Q\left(\mathbf{s}, \mathbf{a}|\theta^{Q}\right)$ according to the input of action and state. Meanwhile, the actor-network is trained to generate a deterministic policy $\mu\left(\mathbf{s}|\theta^{\mu}\right)$ to decide the action with the maximum Q-value in the critic-network. Since both the actor-network and the critic-network being updated are also used in calculating the Q-value, this can potentially lead to training instabilities for highly nonlinear function approximations, i.e., neural networks. To solve that, the target actor-network $\bar{\mu}\left(\mathbf{s}|\theta^{\bar{\mu}}\right)$ and target critic-network $\bar{Q}\left(\mathbf{s}, \mathbf{a}|\theta^{\bar{Q}}\right)$ are created as time-delayed copies of their original networks. As these target networks slowly track the respective learned networks, it greatly improves stability in learning [20].

B. Cooperative Model Training in Modified MADDPG

The rationale behind our modified MADDPG is that, if an SD knows all the actions made by its symbiotic cooperators, the environment is relatively stationary even the policies change. Since the actions of other SDs are exploited in cooperative critic training, an SD can learn the approximate policies of its symbiotic cooperators and effectively utilize them in its own policy learning procedure. In this way, the modified MADDPG models can reduce policy conflicts, guide SDs to establish firm symbiotic relationships, and efficiently make symbiotic service decisions.

In this modified MADDPG, the action $\boldsymbol{a}_i^t = \mu_i(\boldsymbol{s}_i^t) + N$ is first decided by SD i's individual policy μ_i, where $\boldsymbol{s}_i^t$ is SD i's individual state, and N is the sampled noise for action exploration. Let $\mathcal{K}_i$ denote SD i together with its symbiotic cooperators. Once the actions are decided, SDs $\mathcal{K}_i$ will start the symbiotic service exchange. After that, the reward and next state of every SD will be generated. Different from the original DDPG, the replay buffer of an SD in the modified MADDPG includes the additional training information (i.e., policy, state, action, and reward) of its symbiotic cooperators. We denote the joint state at time t of $\mathcal{K}_i$ as $\tilde{\mathbf{s}}_i^t = (\mathbf{s}_1^t, \cdots, \mathbf{s}_{K_i}^t)$. Then, the cooperative training information tuple $\left(\tilde{\mathbf{s}}_i^t, \mathbf{a}_1^t, \cdots, \mathbf{a}_K^t, r_1^t, \cdots, r_{K_i}^t, \tilde{\mathbf{s}}_i^{t+1}\right)$ of the modified MADDPG is stored into the replay buffer $\mathcal{D}_i$ of SD i.

When sampling policies, the SDs refer to the cumulative weights of others, thus ensuring the selected policies are creditable. The joint policy of $\mathcal{K}_i$ is denoted as $\boldsymbol{\mu}_i = \{\mu_1, \cdots, \mu_{K_i}\}$, while $\bar{\boldsymbol{\mu}}_i = \{\bar{\mu}_1, \cdots, \bar{\mu}_{K_i}\}$ is the joint target policy. Since the local model of SD i is cooperatively trained by using the sample $\left(\tilde{\mathbf{s}}_i^m, \mathbf{a}_1^m, \cdots, \mathbf{a}_{K_i}^m, r_1^m, \cdots, r_{K_i}^m, \tilde{\mathbf{s}}_i^{m+1}\right)$

from buffer $\mathcal{D}_i$, the action-value function Q_i^{μ} is updated as

$$\begin{aligned} &Loss(\theta_i^Q) = \\ &\mathbb{E}_{\tilde{\mathbf{s}}^m,\mathbf{a}^m,\mathbf{r}^m,\tilde{\mathbf{s}}^{m+1}} \left[(Q_i^{\mu}(\tilde{\mathbf{s}}_i, \mathbf{a}_1, \cdots, \mathbf{a}_{K_i}) - y^m)^2 \right], \end{aligned} \tag{10}$$

where

$$\begin{aligned} &y^m = \\ &r_i^m + \gamma \bar{Q}_i^{\mu}(\tilde{\mathbf{s}}_i^{m+1}, \mathbf{a}_1^{m+1}, \cdots, \mathbf{a}_{K_i}^{m+1})|_{\mathbf{a}_j^{m+1}=\bar{\mu}_j(\mathbf{s}_j^m)}. \end{aligned} \tag{11}$$

$\bar{Q}_i^{\mu}$ is the target Q-value calculated in SD i's target critic-network by using output actions of all cooperator SDs' target actor networks, and γ is a discount factor. The actor is updated by using the sampled policy gradient

$$\begin{aligned} \nabla_{\theta_i^{\mu}} J = \mathbb{E}_{\tilde{\mathbf{s}},\mathbf{a}\sim D_i} \Big[&\nabla_{\theta_i^{\mu}} \mu_i(\mathbf{s}_i^m | \mathbf{a}_i^m) \\ &\nabla_{a_i} Q_i^{\mu} \left(\tilde{\mathbf{s}}_i^m, \mathbf{a}_1^m, \cdots, \mathbf{a}_{K_i}^m \right)|_{\mathbf{a}_i=\mu_i(\mathbf{s}_i^m)} \Big]. \end{aligned} \tag{12}$$

The weights of the two target networks are updated by having them slowly track the learned actor-critic networks with a small learning rate $\alpha \ll 1$, shown as follows

$$\bar{\theta}_i^{\,Q} \leftarrow \alpha \theta_i^Q + (1-\alpha)\bar{\theta}_i^{\,Q}, \tag{13}$$

$$\bar{\theta}_i^{\,\mu} \leftarrow \alpha \theta_i^{\mu} + (1-\alpha)\bar{\theta}_i^{\,\mu}. \tag{14}$$

We present the full process of the modified MADDPG in Algorithm. 1.

C. Distributed Action Decision in the Modified MADDPG

After cooperative model training, with all local models pre-trained, efficient service exchange decisions are made in a real-time manner without performing the intensive computing process. The SDs can distributively decide the optimal actions according to the state-action values of their individual local networks. Under the blockchain-secured trusted environment, as malicious SDs are prevented from participating in service exchange, only trusted SDs take the optimal actions according to their individual local models. Therefore, the local state-action value $Q_i^{\mu}\left(\mathbf{s}_i^t, \mathbf{a}_i^t\right)$ of SD i can be interpreted as the contribution to the joint state-action value of the whole symbiotic network $Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t\}\right)$ [21]. The joint state-action value is a monotonically increasing function of each local state-action value. Therefore, we propose the following proposition for the optimal global action decision.

Proposition 2. *The global optimal service exchange of the symbiotic network can be achieved by individually deciding the local optimal actions of the trusted SDs in the proposed BIO-SD scheme, i.e.,*

$$\begin{aligned} Q^{\mu}\left(\tilde{\mathbf{s}}^t, \underset{\mathbf{a}_1^t\in\mathcal{A}_1}{\operatorname{argmax}} Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, \underset{\mathbf{a}_K^t\in\mathcal{A}_K}{\operatorname{argmax}} Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right) \right) = \\ Q^{\mu}\left(\tilde{\mathbf{s}}_t, \{\dot{\mathbf{a}}_1^t, \cdots, \dot{\mathbf{a}}_K^t\}\right). \end{aligned} \tag{15}$$

Proof: Assume $\{\dot{\mathbf{a}}_1^t, \cdots, \dot{\mathbf{a}}_K^t\}$ is the optimal joint action under the joint state $\tilde{\mathbf{s}}^t$. For all the possible joint action $\{\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t\}, \forall \mathbf{a}_i^t \in \mathcal{A}_i^t$, we have

$$Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\dot{\mathbf{a}}_1^t, \cdots, \dot{\mathbf{a}}_K^t\}\right) \geq Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t\}\right). \tag{16}$$

In our BIO-SD scheme, only trusted SDs approved by the proposed DAG-based blockchain can participate in coevolution. Therefore, each local state-action value of a trusted SD contributes to the joint state-action value, thus we have

$$\frac{\partial Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t\}\right)}{\partial Q_i^{\mu}\left(\mathbf{s}_i^t, \mathbf{a}_i^t\right)} \geq 0. \tag{17}$$

On the other hand,

$$\begin{aligned} Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\dot{\mathbf{a}}_1^t, \cdots, \dot{\mathbf{a}}_K^t\}\right) = \\ f\left(Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right)\right) \leq \\ f\left(Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, Q_i^{\mu}\left(\mathbf{s}_i^t, \dot{\mathbf{a}}_i^t\right), \cdots, Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right)\right) \\ = Q^{\mu}\left(\tilde{\mathbf{s}}^t, \mathbf{a}_1^t, \cdots, \mathbf{a}_{i-1}^t, \dot{\mathbf{a}}_i^t, \mathbf{a}_{i+1}^t, \cdots, \mathbf{a}_K^t\right), \end{aligned} \tag{18}$$

where $\dot{\mathbf{a}}_i^t = \operatorname{argmax}_{\mathbf{a}_i^t\in\mathcal{A}_i^t} Q_i^{\mu}\left(\mathbf{s}_i^t, \mathbf{a}_i^t\right)$. Then, the equation (18) is transformed as follows

$$\begin{aligned} Q^{\mu}\left(\tilde{\mathbf{s}}^t, \mathbf{a}_1^t, \cdots, \mathbf{a}_{i-1}^t, \dot{\mathbf{a}}_i^t, \mathbf{a}_{i+1}^t, \cdots, \mathbf{a}_K^t\right) \leq \\ Q^{\mu}\Bigg(\tilde{\mathbf{s}}^t, \underset{\mathbf{a}_1^t\in\mathcal{A}_1^t}{\operatorname{argmax}} Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, \underset{\mathbf{a}_i^t\in\mathcal{A}_i^t}{\operatorname{argmax}} Q_i^{\mu}\left(\mathbf{s}_i^t, \mathbf{a}_i^t\right), \\ \cdots, \underset{\mathbf{a}_K^t\in\mathcal{A}_K^t}{\operatorname{argmax}} Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right) \Bigg). \end{aligned} \tag{19}$$

According to (16) and (19), we prove Proposition 2. ■

The Proposition 2 leads to the following corollary.

Corollary 1. *The optimal actions made by SDs in our proposed BIO-SD scheme outperform the actions made by SDs from an unsecured network, i.e.,*

$$\begin{aligned} Q^{\mu}\left(\tilde{\mathbf{s}}^t, \underset{\mathbf{a}_1^t\in\mathcal{A}_1}{\operatorname{argmax}} Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, \underset{\mathbf{a}_K^t\in\mathcal{A}_K}{\operatorname{argmax}} Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right) \right) > \\ Q^{\mu}\Bigg(\tilde{\mathbf{s}}^t, \underset{\mathbf{a}_1^t\in\mathcal{A}_1}{\operatorname{argmax}} Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, \underset{\mathbf{a}_\kappa^t\in\mathcal{A}_\kappa}{\operatorname{argmax}} Q_\kappa^{\mu}\left(\mathbf{s}_\kappa^t, \mathbf{a}_\kappa^t\right), \\ \cdots, \underset{\mathbf{a}_K^t\in\mathcal{A}_K}{\operatorname{argmax}} Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right) \Bigg). \end{aligned} \tag{20}$$

Proof: If malicious SD κ exists, it makes a negative contribution to the joint state-action value, we have $\frac{\partial Q^{\mu}\left(\tilde{\mathbf{s}}^t, \{\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t\}\right)}{\partial Q_\kappa^{\mu}\left(\mathbf{s}_\kappa^t, \mathbf{a}_\kappa^t\right)} < 0$. Similar to (19), we have

$$\begin{aligned} Q^{\mu}\left(\tilde{\mathbf{s}}^t, \mathbf{a}_1^t, \cdots, \dot{\mathbf{a}}_\kappa^t, \cdots, \mathbf{a}_K^t\right) > \\ Q^{\mu}\Bigg(\tilde{\mathbf{s}}^t, \underset{\mathbf{a}_1^t\in\mathcal{A}_1^t}{\operatorname{argmax}} Q_1^{\mu}\left(\mathbf{s}_1^t, \mathbf{a}_1^t\right), \cdots, \underset{\mathbf{a}_\kappa^t\in\mathcal{A}_\kappa^t}{\operatorname{argmax}} Q_\kappa^{\mu}\left(\mathbf{s}_\kappa^t, \mathbf{a}_\kappa^t\right) \\ \cdots, \underset{\mathbf{a}_K^t\in\mathcal{A}_K^t}{\operatorname{argmax}} Q_K^{\mu}\left(\mathbf{s}_K^t, \mathbf{a}_K^t\right) \Bigg). \end{aligned} \tag{21}$$

According to (15) and (21), Corollary 1 is obtained. ■

Overall, the deployment of the modified MADDPG enables SDs to make intelligent service exchange decisions that benefit all cooperators. These service exchanges are verified and recorded by the DAG-based blockchain, and then issued with trustworthiness weight. Importantly, the key junction between the DAG-based blockchain and the modified MADDPG lies in

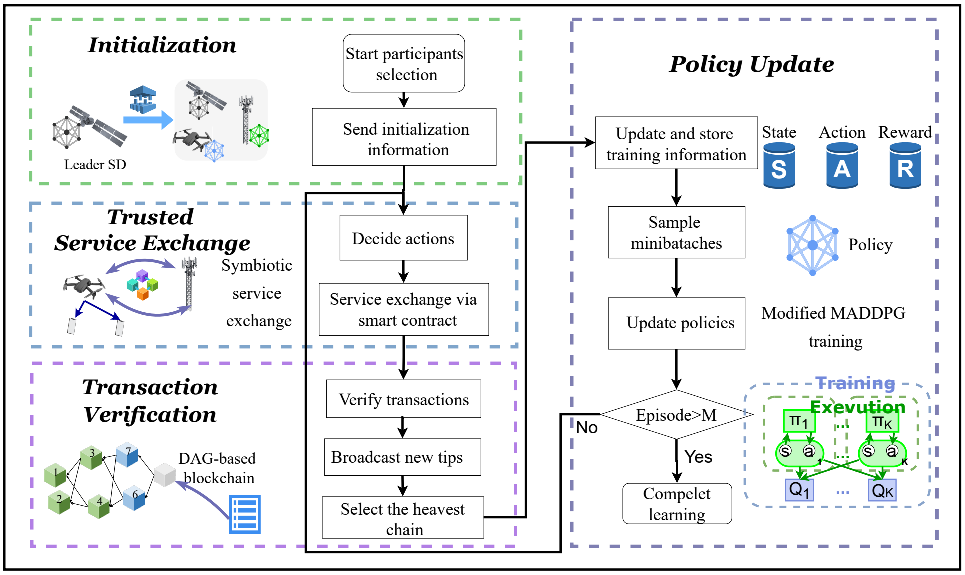


Fig. 4: Workflow of BIO-SD scheme.

that the DAG-based blockchain serves as a guide for the SDs to collectively train local models with trustworthy cooperators and achieve coevolution. This coevolution allows for updating robust and high-quality policies and making wiser decisions, even in the face of malicious attacks.

VI. Workflow of BIO-SD Scheme

As shown in Fig. 4, the BIO-SD scheme is composed of four processes: 1) initialization, 2) service exchange, 3) transaction verification and recording, and 4) policy updating.

A. Initialization

In the beginning, some pre-defined SDs start the SC participation verification process, as shown in Fig. 4. Then, other SDs send their information, like location, radio type, power supply type, network condition, etc., to the pre-defined SDs and request participation via signaling exchange. Subsequently, the SDs that meet the minimum requirements for service exchanges will receive the initialization information to participate in the DRL and blockchain systems. The BIO-SD initialization information includes the modified MADDPG local model structure, DAG-based blockchain transaction generation principle, reward calculation function, etc. Although the same information is shared with every SD, the training information (such as observations and rewards) generated by different SDs is distinct, leading to varying policies for local models in the modified MADDPG.

B. Trusted Service Exchange

First, SDs receive the network access requests from UEs within their coverage via signaling. A known pilot signal sequence with a tiny data size is sent with the request and then the SDs can estimate the channel's current state by comparing the difference between received and known sequences [22]. Then, SDs decide on service exchange actions according to their individual policies and states. Subsequently, the DAG consensus execution process starts, where SDs begin to exchange services via the smart contract and cooperatively provide diverse network access services for UEs according to their service requirements, as shown in Fig. 4.

C. Transaction Verification

By using a gossip algorithm, several SDs broadcast their service exchange records as genesis tips to others. After that, verifier SDs can validate the old tips or unconfirmed transactions and broadcast their own tips until the heaviest chain selection stage starts. In the end, only the heaviest chain should be reserved while others are deleted as per the heaviest chain rule.

D. Policy Update

First, the cumulative weight of an SD is calculated based on its validated transactions' trustworthiness weights. Then, SDs obtain partial corresponding observations from their symbiotic cooperators along with their own observations. Meanwhile, SDs use the reward calculation function to access their action reward based on service quality benchmarks including required latency, required transmission rate, actual latency, and actual transmission rate. Next, training information like action, state, reward, and the next state of SDs and their cooperators is fetched via signaling exchange. SDs pack and store the training information into their replay buffer. Finally, SDs begin to cooperatively train and update their learning models by using individual and cooperator training information from the replay buffer, as indicated in Fig. 4. Once model convergence is achieved in the policy update process, there is no need for continuous model updates. SDs can effectively utilize pre-trained models to make real-time decisions regarding service exchanges.

VII. Simulation Results and Discussions

In this section, we conduct simulations to evaluate the performance of our proposed BIO-SD scheme. To make the simulation results more convincing, we consider a practical network scenario, i.e., a space-air-ground integrated network (SAGIN).

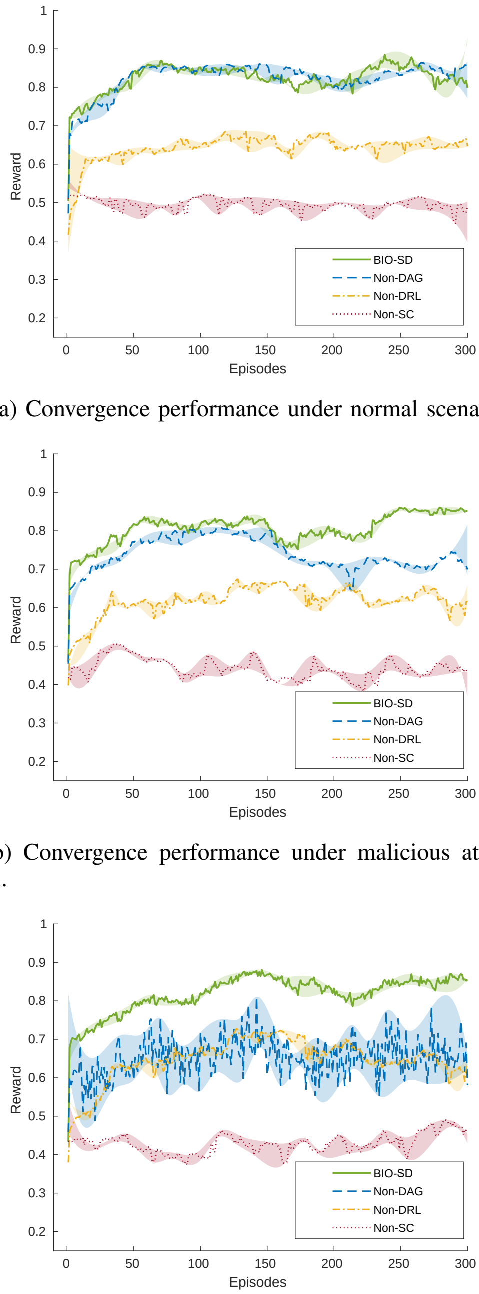
A. Simulation Settings

The simulation environment is designed based on a typical SAGIN environment [4], where 8 static base stations (BSs), 6 moving unmanned aerial vehicles (UAVs), and 3 low Earth orbits (LEOs) are deployed as SDs to provide services for 150 to 350 UEs. The LEOs are positioned within an orbit of 400 km above the Earth's surface, travel at a speed of 7.35 km/s, and their orbital period is set to 105 minutes [23]. Meanwhile, the UAVs operate at an altitude of 5 km, with a moving speed of 20 m/s and an angle of 180 degrees [23]. Additionally, the coverage radius of each ground BS is set to 200 m [24]. For channel models, a BS, a UAV, and an LEO are able to provide service with the downlink bandwidth of 20MHz, 20MHz, and 250 MHz, respectively. To simplify the service exchange problem, in this typical environment, the Doppler shift and short channel coherence time are not considered, while specific path loss coefficients are assigned to different links. The link between the BS/UAV and the receiver is characterized by a path loss coefficient of 3, and the link between the LEO and the receiver is associated with a higher path loss coefficient of 4 [4]. The dynamic service requirement of each individual UE is randomly generated within the transmission rate requirement range $[1, 100] Mbps$, and the latency range $[1, 100] ms$.

In the MADDPG-based local model, both actor-network and target actor-network have an input layer with 32 neurons and an 8-neural output layer, as well as 2 hidden layers with 256 neurons. We employ $ReLU$ as the activation function between the input layer and the two hidden layers, and $tanh$ as the activation function between the second hidden layer and the output layer. The critic-network and target critic-network have the same settings as those in the actor-network except that $ReLU$ is used as the active function between the second hidden layer and the output layer. The learning rates of the actor-network and the critic-network are set as 2.5×10^{-5} and 2.5×10^{-4}, respectively. The learning rate of both the target actor-network and target critic-network is set to 0.001.

Moreover, for the blockchain system, all the SDs are responsible for transaction generation and verification. An SD needs to validate at least 2 previous transactions before generating a new transaction. Each SD is assigned a unique ID as well as a private/public key pair. Two arrays are assigned for the local blockchain and transaction pool for each SD, which are assumed to have ample storage. The global blockchain is periodically updated, and all transactions in it can be accessed by any SD.

In addition, the simulations consider three scenarios: a normal scenario without any malicious SD, and two scenarios involving malicious attacks. Specifically, malicious attack A corresponds to an SD that ceases updating the local model and service exchange, while malicious attack B corresponds to an SD that generates fraudulent transactions and intentionally shares counterfeit policies.



(a) Convergence performance under normal scenario.

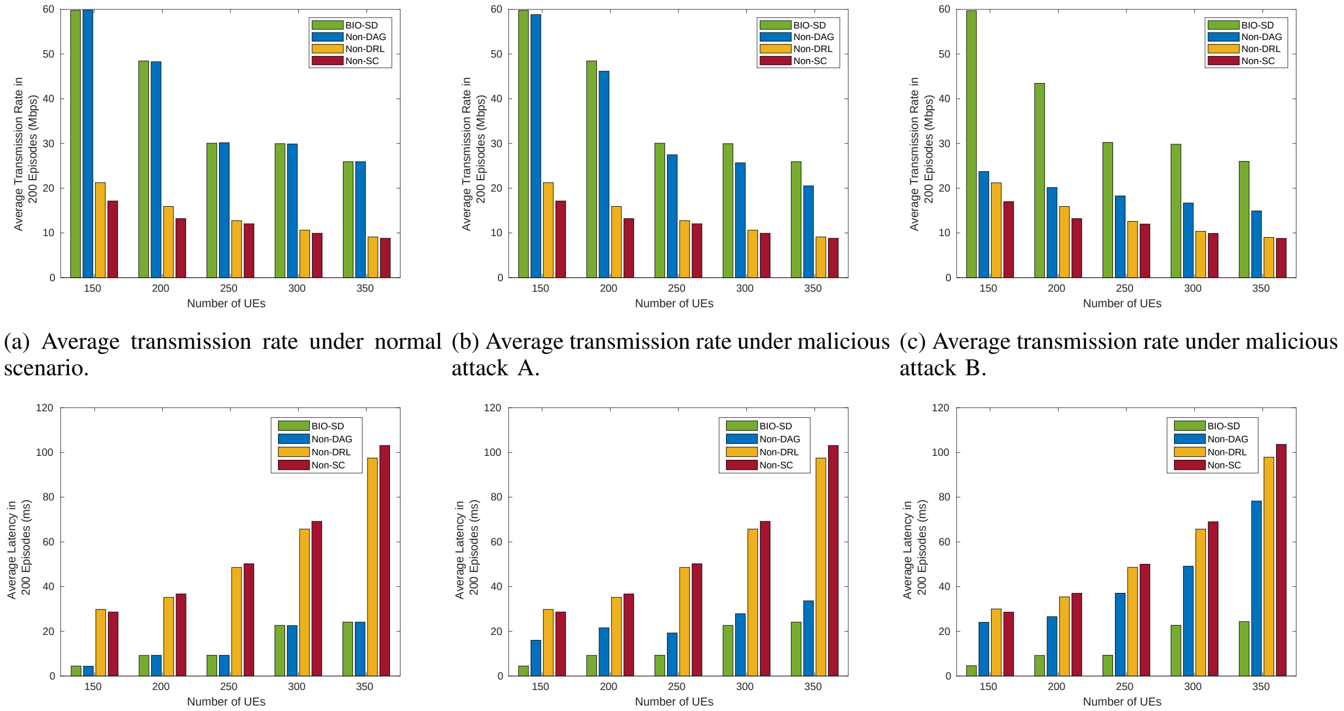
(b) Convergence performance under malicious attack A.

(c) Convergence performance under malicious attack B.

Fig. 5: Convergence performance for different schemes under different scenarios. (Normal scenario: No malicious attack. Malicious attack A: An SD ceases updating the local model and service exchange. Malicious attack B: An SD generates fraudulent transactions and intentionally shares counterfeit policies.)

B. BIO-SD Framework Performance Evaluation

In order to demonstrate the rationality of our proposed BIO-SD scheme framework, we first compare the BIO-SD with the



(a) Average transmission rate under normal scenario.

(b) Average transmission rate under malicious attack A.

(c) Average transmission rate under malicious attack B.

(d) Average latency under normal scenario.

(e) Average latency under malicious attack A.

(f) Average latency under malicious attack B.

Fig. 6: Service quality experienced by UEs for different schemes under different scenarios. (Normal scenario: No malicious attack. Malicious attack A: An SD ceases updating the local model and service exchange. Malicious attack B: An SD generates fraudulent transactions and intentionally shares counterfeit policies.)

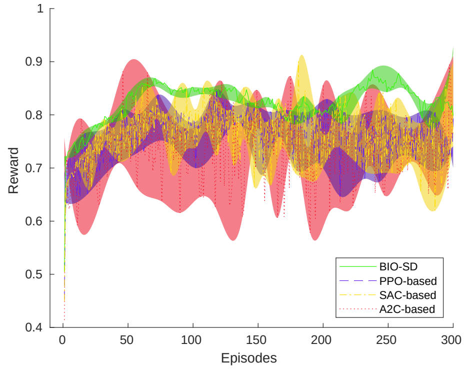


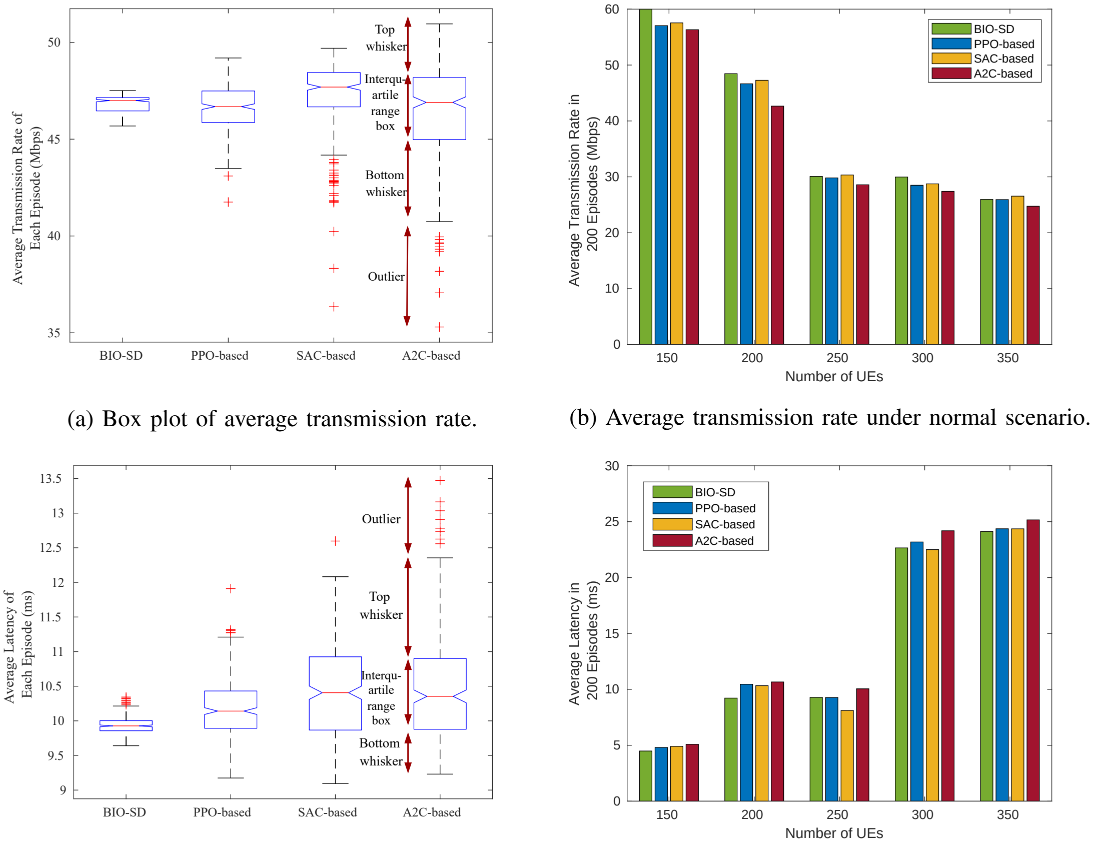
Fig. 7: Convergence performance for four DRL-based schemes.

following three benchmarks [2].

1) Non-DAG scheme: This scheme is identical to BIO-SD, except there is no DAG-based blockchain to secure a trusted service exchange environment (Benchmark for verifying the effectiveness of DAG-based blockchain).
2) Non-DRL scheme: It is similar to the BIO-SD scheme, except that the DRL-based service exchange policy is replaced by a heuristic algorithm-based conventional method that gives priority to SDs with low demand according to cumulative weights (Benchmark for verifying the effectiveness of DRL).
3) Non-SC scheme: There is no symbiotic service exchange in this benchmark, while resources are pre-allocated in this benchmark according to the throughput requirements of UEs in different SDs' coverage (Benchmark for verifying the effectiveness of SD).

We compare the reward convergence performance of the BIO-SD scheme with the three benchmarks when serving 200 UEs under both non-attack and malicious attack situations. These three benchmarks employ the same reward function as the BIO-SD scheme. According to Fig. 5(a), under the non-attack scenario, the reward of the BIO-SD and the non-DAG scheme quickly converge to a value above 0.8 after 60 episodes with a light vibration. Without the effective service exchange assisted by DRL, the reward of the non-DRL scheme converges to about 0.6 after 40 episodes and vibrates significantly, which is because the conventional scheme cannot efficiently and speedily adjust the policy over a high-dynamic network. Moreover, Fig. 5(b) and Fig. 5(c) evaluate the performance of the four schemes under different malicious attacks. We observe that the reward of the BIO-SD scheme converges to almost the same value as that in Fig. 5(b) and Fig. 5(c), while the reward for the non-DAG scheme reduces significantly in Fig. 5(b) and fails to converge in Fig. 5(c). This verifies that DAG-empowered schemes can efficiently avoid fake transactions and exclude malicious SDs from participating in service exchanges and model training.

[2] The performance evaluation of the learning approach will be explored later.



(a) Box plot of average transmission rate.

(b) Average transmission rate under normal scenario.

(c) Box plot of average latency.

(d) Average latency under normal scenario.

Fig. 8: Service quality experienced by UEs for four different DRL-based schemes.

Then, we examine the average transmission rate and average latency within 200 episodes provided by the four schemes under different UE numbers. Both non-attack and malicious attack scenarios are explored. It is observed that in Fig. 6(a) the average transmission rate of the BIO-SD scheme and the non-DAG scheme always outperform the other two non-intelligent schemes, although the average transmission rate of all four schemes decreases with the number of UEs. While comparing Fig. 6(a) and Fig. 6(b), we notice that the transmission rate of the non-DAG scheme is about $2-5Mbps$ lower than that of the BIO-SD scheme under malicious attacks, due to the lack of a trusted service exchange environment. Moreover, from Fig. 6(c), it is more obvious that the non-DAG scheme cannot provide a satisfying service as the BIO-SD scheme. Without the trusted environment secured by the DAG-based blockchain, the DRL policy updating and decision-making can be seriously impacted. Similar results can be observed about the average latency of the four schemes from Fig. 6(d), Fig. 6(e), and Fig. 6(f) which further verify the effectiveness of introducing blockchain and DRL.

C. *Learning Approach Performance Evaluation in BIO-SD*

In the following, to examine the performance of the learning approach exploited in the BIO-SD, we compare it with the following three DRL-based schemes, in which all parts except for the learning approach are the same as in the BIO-SD scheme.

1) A2C-based scheme: This benchmark exploits the vanilla actor-critic (A2C) model [25] instead of DDPG, where A2C local models are maintained by SDs independently in a non-cooperative distributed training mode.
2) PPO-based scheme: This scheme uses proximal policy optimization (PPO)-based local models [26] to replace the DDPG-based local models, while the rest parts are the same as BIO-SD, including the cooperative training mode.
3) SAC-based scheme: It replaces the DDPG local model with the soft actor-critic (SAC)-based learning model [27] in local model training, where additional training information from cooperator SDs is used as the input.

Fig. 7 compares the convergence of the reward for the four schemes under the non-attack scenario with 200 UEs. We observe that the BIO-SD outperforms the other three schemes in terms of both convergence speed (within about 60 episodes) and vibration in reward. This is because the BIO-SD avoids conflicts of interest among SDs by using training information from cooperator SDs to train local models. Additionally, the modified MADDPG of the BIO-SD scheme selects accurate actions according to a deterministic policy, which might reduce the high variance in the gradient.

Fig. 8(a) compares the average transmission rate of each episode attained by the four schemes when serving 200 UEs with dynamic service requests. The transmission rate requirement of each UE is independently and randomly generated in each episode according to a normal distribution with a mean of $50Mbps$. The interquartile range box in Fig. 8(a) represents the distribution of the middle 50% of the transmission rate, and the two whickers are the ranges for the bottom 25% and the top 25%, while outliers denote the remaining tail data. Fig. 8(a) shows that the BIO-SD scheme has a narrow interquartile box with the range of $46-47Mbps$. Additionally, the BIO-SD scheme's whickers are the shortest with no outliers. Therefore, it means that the BIO-SD always serves UEs on demand by exchanging resources among SDs, thus providing the average transmission rate of every episode close to the mean of the required rate. Meanwhile, the average transmission rate in all 200 episodes under different UE numbers are evaluated in Fig. 8(b). Although the proposed BIO-SD does not significantly improve the average transmission rate compared with the other three schemes in Fig. 8(a), it shows an attractive stability when satisfying different service requests, as demonstrated in Fig. 8(c). Similar to Fig. 8(a), Fig. 8(c) about average latency in each episode of the four schemes verifies that the BIO-SD scheme can stably maintain the desired average latency when satisfying dynamic service requests. Meanwhile, Fig. 8(d) demonstrates the average latency in 200 episodes of the BIO-SD, which outperforms the other three schemes with a slight average latency reduction in the majority of scenarios. The results further indicate that the modified MADDPG of the proposed BIO-SD is effective and robust.

VIII. Conclusion

In this paper, we proposed the BIO-SD scheme to efficiently manage resources in a symbiotic ecosystem via secure service exchange. A DAG-based blockchain is presented to secure a trusted service exchange environment and a modified MADDPG-based approach is developed to make optimal decisions. Simulations are conducted in specific SAGIN network scenarios with and without malicious nodes, where numerical results demonstrate the robustness and service quality improvement of our proposed BIO-SD scheme. In general, we expect this work to be a pioneer for breaking the boundaries of heterogeneous devices' resources in symbiotic networks through the interplay of DRL and blockchain.

References

[1] R. Cheng, Y. Sun, L. Mohjazi, Y. Liu, Y.-C. Liang, and M. Imran, "Intelligent Resource Management in Symbiotic Radio under a Trusted Coevolution," in *IEEE ICC 2023 - IEEE International Conference on Communications (ICC)*, 2023.

[2] Y.-C. Liang, R. Long, Q. Zhang, and D. Niyato, "Symbiotic Communications: Where Marconi Meets Darwin," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 144–150, 2022.

[3] T. Hewa, G. Gür, A. Kalla, M. Ylianttila, A. Bracken, and M. Liyanage, "The Role of Blockchain in 6G: Challenges, Opportunities and Research Directions," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*. IEEE, 2020, pp. 1–5.

[4] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-Air-Ground Integrated Network: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2714–2741, 2018.

[5] D. Bloembergen, K. Tuyls, D. Hennes, and M. Kaisers, "Evolutionary Dynamics of Multi-Agent Learning: A Survey," *Journal of Artificial Intelligence Research*, vol. 53, pp. 659–697, 2015.

[6] A. M. Seid, H. N. Abishu, Y. H. Yacob, T. A. Ayall, A. Erbad, and M. Guizan, "Blockchain-based Resource Trading in Multi-UAV-assisted Industrial IoT Networks: A Multi-agent DRL Approach," *IEEE Transactions on Network and Service Management*, 2022.

[7] R. Cheng, Y. Sun, Y. Liu, L. Xia, D. Feng, and M. A. Imran, "Blockchain-Empowered Federated Learning Approach for an Intelligent and Reliable D2D Caching Scheme," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 7879–7890, 2021.

[8] R. Cheng, Y. Sun, L. Mohjazi, Y.-C. Liang, and M. Imran, "Blockchain-assisted intelligent symbiotic radio in space-air-ground integrated networks," *IEEE Network*, vol. 37, no. 2, pp. 94–101, 2023.

[9] W. Sun, L. Wang, P. Wang, and Y. Zhang, "Collaborative Blockchain for Space-Air-Ground Integrated Networks," *IEEE Wireless Communications*, vol. 27, no. 6, pp. 82–89, 2020.

[10] N. Kato, Z. M. Fadlullah, F. Tang, B. Mao, S. Tani, A. Okamura, and J. Liu, "Optimizing Space-Air-Ground Integrated Networks by Artificial Intelligence," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 140–147, 2019.

[11] P. S. Bithas, E. T. Michailidis, N. Nomikos, D. Vouyioukas, and A. G. Kanatas, "A Survey on Machine-Learning Techniques for UAV-Based Communications," *Sensors*, vol. 19, no. 23, p. 5170, 2019.

[12] M. Lapan, *Deep Reinforcement Learning Hands-On: Apply Modern RL Methods, with Deep Q-Networks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More*. Packt Publishing Ltd, 2018.

[13] J. Cui, Y. Liu, and A. Nallanathan, "Multi-Agent Reinforcement Learning-Based Resource Allocation for UAV Networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 729–743, 2019.

[14] Y. Du, L. Han, M. Fang, J. Liu, T. Dai, and D. Tao, "Liir: Learning Individual Intrinsic Reward in Multi-Agent Reinforcement Learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[15] J. Xie, K. Zhang, Y. Lu, and Y. Zhang, "Resource-Efficient DAG Blockchain with Sharding for 6G Networks," *IEEE Network*, 2021.

[16] L. Cui, S. Yang, Z. Chen, Y. Pan, M. Xu, and K. Xu, "An Efficient and Compacted DAG-based Blockchain Protocol for Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4134–4145, 2019.

[17] H. Pervez, M. Muneeb, M. U. Irfan, and I. U. Haq, "A Comparative Analysis of DAG-Based Blockchain Architectures," in *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, 2018, pp. 27–34.

[18] S. Popov, "The Tangle," *White paper*, vol. 1, no. 3, 2018.

[19] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[20] 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.

[21] Z. Wang, J. Zong, Y. Zhou, Y. Shi, and V. W. Wong, "Decentralized Multi-Agent Power Control in Wireless Networks with Frequency Reuse," *IEEE Transactions on Communications*, vol. 70, no. 3, pp. 1666–1681, 2021.

[22] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, "A survey on 3gpp heterogeneous networks," *IEEE Wireless communications*, vol. 18, no. 3, pp. 10–21, 2011.

[23] F. Tang, H. Hofner, N. Kato, K. Kaneko, Y. Yamashita, and M. Hangai, "A Deep Reinforcement Learning-Based Dynamic Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN)," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 276–289, 2021.

[24] M. Liu, G. Gui, N. Zhao, J. Sun, H. Gacanin, and H. Sari, "UAV-Aided Air-to-Ground Cooperative Nonorthogonal Multiple Access," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2704–2715, 2019.

[25] J. Peters and S. Schaal, "Natural Actor-Critic," *Neurocomputing*, vol. 71, no. 7-9, pp. 1180–1190, 2008.

[26] C. Yu, A. Velu, E. Vinitsky, Y. Wang, A. Bayen, and Y. Wu, "The Surprising Effectiveness of PPO in Cooperative, Multi-Agent Games," *arXiv preprint arXiv:2103.01955*, 2021.

[27] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1861–1870.

Runze Cheng received the B.Eng. degree from the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China, in 2019, and the M.Sc. degree from the Department of Electrical and Electronic Engineering, University of Nottingham, UK, in 2020. He is currently pursuing his Ph.D. degree with James Watt School of Engineering, University of Glasgow, UK. His research interests include resource management in wireless communication, distributed machine learning, blockchain system, and semantic communication.

Yao Sun is currently a Lecturer with James Watt School of Engineering, the University of Glasgow, Glasgow, UK. Dr. Sun has extensive research experience in wireless communication area. He has won the IEEE IoT Journal Best Paper Award 2022, and IEEE Communication Society of TAOS Best Paper Award in 2019 ICC. His research interests include intelligent wireless networking, semantic communication and wireless blockchain system. Dr. Sun is a senior member of IEEE.

Yijing Liu received her B.S. degree in college of communication and information engineering from Chong Qing University of Post and Telecommunications, Chongqing, China, in 2017. She is currently pursuing the Ph.D. degree at the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China. Her current research interests include next generation mobile networks, network slicing, and distributed machine learning.

Ying-Chang Liang received the B.S. and Ph.D. degrees from Jilin University, Changchun, China. He served as a Postdoctoral Fellow with Tsinghua University, Beijing, China, and latter with Nanyang Technological University, Singapore, and the University of Maryland at College Park, College Park, MD, USA. He was a Professor with The University of Sydney, Sydney, NSW, Australia, a Principal Scientist and a Technical Advisor with the Institute for Infocomm Research, Singapore, and a Visiting Scholar with Stanford University, Stanford, CA, USA. He is currently a Professor with the University of Electronic Science and Technology of China, Chengdu, China, where he leads the Center for Intelligent Networking and Communications. His research interests include wireless networking and communications, cognitive radio, symbiotic communications, dynamic spectrum access, Internet of Things, artificial intelligence, and machine-learning techniques.

Muhammad Ali Imran (M'03, SM'12) Fellow IEEE, Fellow IET, Senior Fellow HEA is a Professor of Wireless Communication Systems with research interests in self organised networks, wireless networked control systems and the wireless sensor systems. He heads Autonomous System and Connectivity resource division at University of Glasgow and is the Dean University of Glasgow, UESTC. He is an Affiliate Professor at the University of Oklahoma, USA and a visiting Professor at 5G Innovation Centre, University of Surrey, UK. He has over 20 years of combined academic and industry experience with several leading roles in multi-million pounds funded projects. He has been a consultant to international projects and local companies in the area of self-organised networks. He has been interviewed by BBC, Scottish television and many radio channels on the topic of 5G technology.