Consortium Blockchain-Based Spectrum Trading for Network Slicing in 5G RAN: A Multi-Agent Deep Reinforcement Learning Approach

Gordon Owusu Boateng, *Graduate Student Member, IEEE,* Guolin Sun, *Member, IEEE,* Daniel Ayepah Mensah, Daniel Mawunyo Doe, Ruijie Ou, and Guisong Liu

Abstract—Network slicing (NS) is envisioned as an emerging paradigm for accommodating different virtual networks on a common physical infrastructure. Considering the integration of blockchain and NS, a secure decentralized spectrum trading platform can be established for autonomous radio access network (RAN) slicing. Moreover, the realization of proper incentive mechanisms for fair spectrum trading is crucial for effective RAN slicing. This paper proposes a novel hierarchical framework for blockchain-empowered spectrum trading for NS in RAN. Specifically, we deploy a consortium blockchain platform for spectrum trading among spectrum providers and buyers for slice creation, and autonomous slice adjustment. For slice creation, the spectrum providers are infrastructure providers (InPs) and buyers are mobile virtual network operators (MVNOs). Then, underloaded MVNOs with extra spectrum to spare, trade with overloaded MVNOs, for slice spectrum adjustment. For proper incentive maximization, we propose a three-stage Stackelberg game framework among InPs, seller MVNOs, and buyer MVNOs, for joint optimal pricing and demand prediction strategies. Then, a multi-agent deep reinforcement learning (MADRL) method is designed to achieve a Stackelberg equilibrium (SE). Security assessment and extensive simulation results confirm the security and efficacy of our proposed method in terms of players' utility maximization and fairness, compared with other baselines.

Index Terms—blockchain, network slicing, resource trading, Stackelberg game, MADDPG, 5G.

1 Introduction

With the increasing proliferation of intelligent mobile devices currently, the fifth generation (5G) and beyond 5G (B5G) networks are expected to deliver ubiquitous connectivity, flexibility in resource management, and overall network security. Due to scarcity of wireless resource in the face of rapid increase in traffic demand, radio access technologies (RATs) and cognitive radio have been explored to meet end users' quality of service (QoS) demands. Such technologies have the disadvantages of increased costs, and complexity in resource management. As an alternative, wireless network virtualization (WNV) has emerged as a key technology for splitting the existing physical network infrastructure into isolated logical networks, often referred to as *network slicing (NS)* [1]. The goal of NS is to map virtualized networks onto a common physical infrastructure, while ensuring the QoS satisfaction of different users with versatile demands. Software defined networking (SDN) and network function virtualization (NFV) are the main enablers of NS, especially in radio access network (RAN) [2].

Although recent advances in NS promise a plethora of efficient resource sharing techniques, huge opportunities present critical challenges. The most critical aspects of NS are resource allocation, and isolation. Despite the above-mentioned challenges, there are still some open issues to be addressed in RAN research. Firstly, resource trading and sharing among entities in various levels of the RAN architectural design, suffers from data privacy leakages as a result of exposing sensitive information of traders [3]. In this regard, resource providers and buyers are unwilling to cooperatively share the scarce wireless resource for efficient utilization. Secondly, improper modeling of interactions between competing resource providers and buyers for resource provisioning may demotivate entities to share their resource with others. Information asymmetry between infrastructure providers (InPs) and mobile virtual network operators (MVNOs) in a typical RAN scenario, forbids the implementation of proper incentive mechanisms to promote resource trading [4]. Thirdly, a vulnerable time-variant resource trading negotiation and renegotiation encounters fairness issues when a few peers condone to fail fair orders matching [5]. This creates uneven grounds for entities to maximize their utilities in resource trading settings. Lastly, resource providers in a bid to maximize profits, may attempt to lease the same resource to different customers at the same time. This results in double-spending, which could cause interference among customers who desire to utilize the very same subleased resource at the same time [6].

With the current roll-out of 5G, most existing conventional security solutions cannot meet its accompanied stringent security requirements. Amongst the present emerging technologies, blockchain has shown promising traits of em-

- *Gordon Owusu Boateng, Guolin Sun, and Daniel Ayepah-Mensah, are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731-CHINA, and also with Intelligent Terminal Key Laboratory of Sichuan Province, Yibin, 644005-CHINA.*
- *Daniel Mawunyo Doe and Ruijie Ou are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731-CHINA.*
- *Guisong Liu is with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics (SWUFE), Chengdu 611130, CHINA.*
- *Corresp. Authors Email: guolin.sun@uestc.edu.cn, gliu@swufe.edu.cn.*

powering secure and robust key 5G applications. Blockchain is a distributed ledger technology (DLT) that was firstly used to perform cryptocurrency (*Bitcoin*) transactions [7]. The key features of blockchain such as immutability, decentralization, transparency, and privacy, make it ideal for mitigating security challenges in 5G networks. The integration of blockchain with 5G is therefore a solution for security challenges in the evolving wireless RAN. In spite of blockchain achieving great strides in supporting diverse vertical applications, pricing and demand prediction for resource management have bottlenecks in resource trading context. Entities involved in resource trading activities form strategies to maximize their own utilities, based on what role they play in the trading process. Game theory emerges as an analytical tool to control payoffs of both buyers and sellers, by modeling the interaction between entities, and achieving optimal trading strategies [8]. To this end, blockchain and game theory have been tailored for secure and transparent resource trading, while achieving optimal reaction strategies based on equilibrium analysis.

Many existing literatures have attempted to propose various methods for secure, and fair resource trading in 5G NS by combining blockchain, and game theory separately. The authors in [9] proposed a two-stage Stackelberg game framework for dynamic pricing and resource allocation in OFDMA virtualized wireless network. Taking a market model consisting of an InP and multiple MVNOs into account, a dual-based algorithm was proposed to maximize the InP's revenue. This work lacks clarity on the security of the resource trading activities in the wireless environment, which cannot be ignored in current research. The work in [6] presented a blockchain-based MVNOs creation via secure and transparent WNV technique. However, the techno-economic aspect of blockchain-related NS is vague, making it uncertain whether the said scenario is feasible in the real-world environment. Taking changes in network states at a specific time into account, reinforcement learning (RL) [10] is preferred to traditional optimization schemes for solving resource allocation problems in RAN. In [11], joint radio and computation resource optimization in edge network slicing was investigated. A utility maximization problem for the MVNO was formulated and solved. To account for the dynamic change of slice demands, a DRL-based algorithm was proposed for dynamic allocation. Technically, this work focuses on maximizing the utility of only the MVNOs, without considering the strategies of the InP and network service providers (NSPs) in the business network. Such a model violates standard interaction modeling policies for resource allocation in a layered slicing network, which provides an unfair ground for dynamic resource allocation.

Motivated by the above-mentioned limitations, the aim of this paper is to combine blockchain, game theory, and DRL, to design a hierarchical framework for secure spectrum trading and intelligent dynamic spectrum management in 5G RAN. We present a consortium blockchain network that supports smart contracts (SCs), to perform decentralized spectrum trading among spectrum providers and spectrum requesters. Most existing works assume spectrum is only leased from InPs to MVNOs for slice creation. However, beyond slice creation, an MVNO with redundant spectrum to spare can trade with its peers for efficient slice resource adjustment, and obtain incentives in return. In this light, we assume an MVNO who is in need of spectrum can borrow from other MVNOs who have redundant spectrum to spare. To create a fair level ground for spectrum trading, we formulate a three-stage multi-leader multi-follower (MLMF) Stackelberg game model for joint optimal pricing and demand prediction-based spectrum management in the layered network. In the first stage, the InPs (leaders) decide their unit prices. In the second stage, each seller MVNO (leader) decides its unit price. In the third stage, the buyer MVNOs (followers) determine the spectrum amount to purchase. We prove the existence and uniqueness of a Stackelberg equilibrium (SE), and propose a multi-agent deep deterministic policy gradient (MADDPG) algorithm, named *Stackelberg MADDPG* to obtain an optimal learning strategy, without any prior knowledge of the network environment. The main contributions of this paper are as follows:

1) We propose a hierarchical framework for consortium blockchain-empowered spectrum trading and DRL-based intelligent NS in 5G RAN.
2) We deploy a consortium blockchain network with hyperledger SC to ensure secure and transparent spectrum trading procedure between multiple spectrum providers and multiple spectrum requesters.
3) Considering proper interaction modeling for incentive maximization, we formulate a three-stage MLMF Stackelberg game to jointly consider dynamic pricing and demand strategies of sellers and buyers respectively, at each stage, for resource management. The sellers are leaders and buyers are followers in each stage of the game.
4) We prove the existence and uniqueness of the SE, and propose a novel MADDPG algorithm referred to as *Stackelberg MADDPG*, for joint optimal pricing and demand prediction strategies. We seek to maximize the utilities of all entities involved in the game.

The remainder of the paper is structured as follows: Section 2 discusses related works, and Section 3 presents the system model. In Section 4, we formulate the joint dynamic pricing and demand-based resource management problem, and propose an MADDPG-based algorithm to achieve the SE. Experimental results and analysis are presented in Section 5. Finally, we conclude this work in Section 6.

2 Related Works

It is projected that 5G and B5G will utilize the recent advances in WNV to expand the full potential of NS [1] [12]. *Da Silva et. al.* [12] discussed the impact of NS on 5G RAN by considering its architectural and protocol design, as well as the management framework for network functions (NFs). Due to the trustless nature of the wireless environment, blockchain technology is perceived as a promising cutting-edge solution to promote secure resource trading in sliced RANs, by ensuring privacy preserving and overall network security. *Nour et. al.* [13] proposed a blockchain-based brokering mechanism, which enables the NSP to lease resources from different resource providers for end-to-end slice creation, ensuring secure and anonymous transactions. In [14], blockchain and NS were adopted to propose *NSBchain*,

a novel network slice brokering mechanism for resource sharing. As an intermediary, the broker enables the InP to assign resources to MVNOs through SCs in a secure and automated manner. However, it is still yet to be explicitly proven whether the intermediate broker is secure and fair enough for supervising resource trading. The authors in [15] addressed spectrum sharing problem in multi operators' wireless communication networks by designing an SC-based multi-operator spectrum sharing (MOSS) scheme. This scheme eliminates the need for a trustless spectrum broker, while punishing malicious operators. *Rawat et. al.* [6] proposed a blockchain-enabled scheme for virtual wireless network creation, where primary wireless resource owners (PWROs) sublease their resources to MVNOs based on service level agreements (SLAs) between the two entities. The works discussed so far mainly focused on leveraging blockchain and WNV for secure resource sharing, but fail to set up an incentive mechanism to maximize the utilities of the buyers and sellers involved in the trading.

Game theory has emerged as a tool for thoroughly modeling the interactions among competing players in a trading environment, presenting an avenue to complement current research. *Chang et. al.* [4] based on contract theory to propose an incentive mechanism for managing radio resources in a virtualized wireless network. The authors in [9] proposed a two-stage Stackelberg game model for dynamic pricing-based resource allocation. In order to maximize users' utilities and MVNOs' revenue, the work in [16] formulated a two-stage Stackelberg game for the interaction between the users and MVNOs. In [17], the authors proposed a framework based on Stackelberg game model for resource allocation in virtualized wireless networks. An SE was derived by applying backward induction method. However, these works assume that the entities involved reveal their private information, which may affect fairness in real-world scenarios. Furthermore, iterative and analytic methods are centralized algorithms that are feasible only when all participants partially disclose their demand and pricing information.

Recent advances in RL show its ability to learn the stochastic policy of a dynamic environment without prior knowledge. *Wang et. al.* [11] presented a joint optimization of radio and computation resources to maximize the utility of MVNOs, using a single provider in multi-access edge computing (MEC)-enabled network slicing. A DDPG-based resource allocation algorithm was proposed to cater for the continuous action space. However, the pricing strategy of a single provider in the resource trading market depends solely on the supply-demand relationship between the multiple requesters and itself [18]. What happens is that such a single provider tends to be monopolistic (i.e., it imposes unrealistic high prices on the requesters), as competition among multiple providers is not considered. In this case, the requesters may be forced to either purchase resource at such high prices or not have enough resource to serve their subscribers. On the other hand, if there are multiple providers in the trading market, the price determination of a provider not only depends on the supply-demand relationship, but also it is affected by the remaining providers' pricing strategies [3]. For instance, if a provider sets its resource price lower than the others', more requesters are bound to purchase resource from the said provider, and such a provider may earn the largest market share. Therefore, the work in [3] applied RL technique to solve the incentive mechanism problem for multiple providers and multiple Internet of things (IoT) devices' computing service trading. An MLMF Stackelberg game formulation was presented for the pricing and demand problem among the players.

Although the above-mentioned literatures have investigated blockchain, game theory, and their unification with RL separately, we believe the joint integration of such disruptive technologies in 5G RAN slicing can unlock its full potential. Blockchain can ensure secure and transparent trading of spectrum, game theory can create a competitive market among spectrum providers and requesters to maximize their utilities, and RL can ensure the optimal pricing and demand prediction of providers and requesters in an automated manner. With this, we propose a techno-economic framework that hinges on blockchain, game theory, and RL for secure spectrum trading and autonomous RAN slicing.

3 System Model

3.1 System Architecture

We consider a multi-cellular time-synchronized OFDMA wireless network, where users of different QoS requirements coexist. The system architecture consists of the physical network and virtualized network, with a three-stage Stackelberg game framework to depict the interaction among the entities involved in spectrum trading. The system is made up of entities such as base stations (BSs)/virtual BSs (VBSs), block managers (BMs), users, MVNOs, a radio intelligent controller (RIC), a DRL agent, and a regulator. The role of SDN in WNV is to decouple the network operations into the control and data planes to allow for ease of network management and programmability via software programs. The RIC is a software-defined component of the O-RAN architecture, responsible for controlling and optimizing RAN functions [19] [20]. We adopt its role in O-RAN, which is in line with the vRAN controller [21], to suit our architecture. That said, the RIC performs the overall network optimization, while the BSs/VBSs perform signaling and user association.

Considering the entities in the system architecture, the RIC is owned by the regulator who acts as an administrator of the trading framework, and is responsible for the deployment of decentralized blockchain. The reason is that the InPs and MVNOs owning such a controller may result in partiality, since they may be tempted to manipulate it to their own benefits. The BSs/VBSs are equipped with blockchain functionalities, and communicate with other BSs/VBSs via BMs. BMs are trusted devices integrated in the BSs/VBSs to maintain the blockchain and distributed cryptographic keys of the entities. The DRL agent is deployed on the RIC to select optimal spectrum management decisions based on the changes in network states at a specific time e.g. change in spectrum demand and price. The reason for deploying the DRL agent on top of the RIC is for easy access to the data analytics for machine learning training to make spectrum optimization decisions.

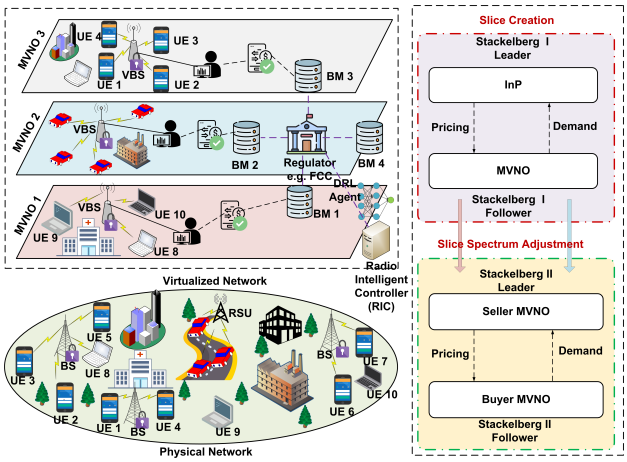


Fig. 1: System architecture.

The system architecture as shown in Fig. 1, is adopted from [22] where multiple InPs partition their substrate networks into slices and lease them to various MVNOs. Then, the MVNOs sublease the leased resources to their respective subscribers in order to satisfy their QoS requirements. To ensure effective MVNOs' spectrum utilization after slice creation, idle spectrum of underloaded MVNOs can be resold to overloaded MVNOs in need of spectrum, in a specific slicing period. We define the slicing period as the time interval after which spectrum of the MVNOs are adjusted, due to changes in spectrum demand and supply of the MVNOs. Setting the slicing period to a small timescale, e.g., seconds, could ensure better spectrum demand estimation, which may lead to better adaptation to the MVNOs' demands. However, this results in high computation and reconfiguration costs. Conversely, setting the slicing period to a large time-scale e.g., hours, may lead to gathering inaccurate spectrum demand of MVNOs, although computation and reconfiguration costs may be minimized. Therefore, we carefully set the slicing period to 20 minutes to strike a balance between better spectrum demand estimation and reconfiguration costs [23]. Since the wireless network is trustless, spectrum sharing among entities should be as secure and transparent as possible. Therefore, the BSs/VBSs are equipped with blockchain for secure spectrum trading, and autonomous slice spectrum adjustment. We decentralize the RIC by linking it to all entities in the blockchain network. Thus, every information on the RIC is copied into databases of all the other entities. In this way, external attacks on or single point of failure of the RIC may not result in data loss [24]. Since the DRL agent is deployed on the RIC, they are both automatically decentralized. We classify the system model into business model, virtualization model, network model, and utility model.

3.2 Business Model

We construct a three-layer hierarchical business model with InPs, seller MVNOs, and buyer MVNOs in the layers. The network service area is covered by a set of InPs denoted by $i \in \mathcal{I} = \{1, 2, ...I\}$, who own infrastructure and resource e.g. spectrum, BS. We denote a set of MVNOs who lease slices from the InPs by $j \in \mathcal{J} = \{1, 2, ...J\}$. The set $\mathcal{J}$ comprises a set of underloaded MVNOs (sellers) $m \in \mathcal{J}_{sell} = \{1, 2, ...M\}$, whose aggregated user demand is lower than their available spectrum, and overloaded MVNOs (buyers) $n \in \mathcal{J}_{buy} = \{1, 2, ...N\}$, whose aggregated user demand is higher than their available spectrum; i.e. $\mathcal{J}_{sell}, \mathcal{J}_{buy} \subseteq \mathcal{J}$. Thus, at slicing period t, a *j-th* MVNO can choose to participate as $m \in \mathcal{J}_{sell}$ or $n \in \mathcal{J}_{buy}$, depending on its trading strategy. From the business perspective, the InPs who own spectrum and infrastructure lease to MVNOs, to be managed independently. In a bid to minimize the cost of spectrum occupancy due to wastage, the MVNOs with abundant spectrum to spare can sublease to other MVNOs who lack spectrum. At slicing period t, an *m-th* seller MVNO decides to sell b_m amount of spectrum to an *n-th* buyer MVNO. To guarantee the QoS satisfaction of the users of the *m-th* seller MVNO, b_m^{req} fraction of spectrum is required to serve its users such that;

$$b_m^{req} \geq b_m^{tot} - b_m, \tag{1}$$

where b_m^{tot} is the total spectrum of the *m-th* seller MVNO. An *n-th* buyer MVNO decides to buy b_n amount of spectrum based on its demand d_n.

For spectrum trading among $\mathcal{I}$ InPs, $\mathcal{J}_{sell}$ seller MVNOs and $\mathcal{J}_{buy}$ buyer MVNOs, we deploy a consortium blockchain network managed by a regulator, who issues certificates to participants. The certificate qualifies entities to take part in spectrum trading activities. The regulator is linked with the InPs and MVNOs as a decentralized network, and is responsible for the deployment of blockchain. It should be noted that the regulator replaces a trustless spectrum broker who is liable to privacy disclosure and single point of failure. Different from the spectrum broker in previous studies, the regulator does not directly participate in the spectrum trading among buyers and sellers [15]. In other words, the regulator only deploys blockchain and calls SCs, but does not participate in mining or verification. In fact, the regulator could be a body such as the government or the federal communications commission (FCC). Each entity acquires transaction details in the tuple $\{Cert, SK, PK, Add\}$, where $Cert$ is the certificate that authorizes the entity to create a blockchain account. An entity is issued a private key SK to digitally sign transactions, and a public key PK for node identification. A wallet address Add will be used to track payment details of spectrum trading transactions [25].

3.3 Virtualization Model

WNV technology enables the physical network to be abstracted, partitioned, and isolated as virtualized networks or *slices*, and assigned to multiple MVNOs as network slice instances. The stages of WNV are presented as follows:

- *Abstraction stage*: The shared common physical infrastructure including the spectrum, is virtualized to simplify the provisioning of customized networks to the heterogeneous traffics.
- *Partitioning stage*: The virtualized infrastructure and spectrum are divided into logical networks and each logical network is referred to as a "slice". This ensures the coexistence of different virtualized networks mapped onto the same physical infrastructure.

- *Isolation stage*: The slices are logically separated from one another so as to avoid conflicts between coexistent virtualized networks.

Slices are assigned radio resource in spectrum form, with granular units of resource blocks (RBs). To avoid ambiguity, we refer to slice resource as spectrum in the following sequel. The main functionality of a slice is to provide differentiated and customized service to its subscribers.

User association is also virtualized to be slice specific in the same manner. Each user directly reports its channel state information (CSI) to the BS that it is attached to. In other words, the BSs collect user information such as user demand, user location, preferred service type, and traffic distribution from the users. Each slice estimates the aggregated resource demand of its subscribers, its available spectrum amount, spectrum demand and supply deficit, etc., as slice parameter states. For efficient spectrum management, the RIC observes the slice parameter states for slice-level spectrum optimization by leveraging AI techniques [26]. This information will be used by the DRL agent deployed on the RIC to make optimal spectrum management decisions. Different users with varied QoS requirements coexist in the network. Therefore, a slice classifier is tasked with sorting requested services into various slice queues, depending on which traffic flow belongs to which slice. The grouping of user requests follows a flow-QoS class mapping based on the QoS classifier index (QCI) table [27]. Then, the slicing controller decides the resource allocation based on demand, and spectrum availability. Note that one flow can be mapped to only one slice, but one or more flows of the same characteristics can be mapped to the same slice. To achieve isolation and proper resource allocation, a BS allocates granular units of spectrum in the form of physical RBs to users in its coverage area.

3.4 Network Model

We define a set of users $k \in \mathcal{K} = \{1, 2, ...K\}$ distributed in the coverage area of multiple cellular networks. We assume that each user k exclusively subscribes to an MVNO, and each MVNO j serves a set of $\mathcal{K}_j$ users. The physical resource of InPs is of aggregated bandwidth $\mathcal{B}$ MHz, which consists of $w \in \mathcal{W} = \{1, 2, ...W\}$ set of RBs. An RB is the granular unit of bandwidth, with each RB having a bandwidth of b_w kHz in time and frequency domain. We further assume that a contiguous portion of the spectrum is allocated to each user. The transmit power of BS i is p_i watts and the number of RBs allocated to MVNO j is w_j. We assume a CSI-aware RB allocation where co-channel interference is controlled by powerful inter-cell interference cancellation (ICIC) techniques [28]. Based on the Shannon capacity theory, the expected instantaneous data rate of user k in slice j, who associates with BS i is expressed as [22];

$$r_{k_j,i} = w_k \cdot \mathcal{B} \cdot \log_2(1 + \Upsilon(k_j, i)), \tag{2}$$

where w_k denotes the number of RBs allocated to user k, and $\Upsilon(k_j, i)$ denotes the signal-to-interference-plus-noise-ratio (SINR) of user k in slice j associated with BS i. Therefore, the expected instantaneous data rate of slice j is the aggregated data rates of all the users in the slice, which is calculated as;

$$r_j = \sum_{k_j=1}^{K_j} (r_{k_j,i}). \tag{3}$$

Let $x_{k_j,i}$ be a binary association indicator, where $x_{k_j,i} = 1$ indicates user k_j associates with BS i, and $x_{k_j,i} = 0$ indicates otherwise. Then, the average achievable data rate of user k in slice j, who associates with BS i is calculated as;

$$\bar{r}_{k_j,i} = x_{k_j,i} \cdot r_{k_j,i}. \tag{4}$$

Combining the average achievable data rate of all the users in the slice, the average achievable data rate of slice j is expressed as;

$$\bar{r}_j = \sum_{k_j=1}^{K_j} (\bar{r}_{k_j,i}). \tag{5}$$

The value of $\bar{r}_j$ should exceed the minimum data rate requirement of slice j i.e. $\bar{r}_j \geq \bar{r_j}^{min}$.

We calculate the expected instantaneous delay of a packet in a queue from BS i to user k in slice j based on M/M/1 queuing theory [29] as;

$$\tau_{k_j,i} = \frac{1}{r_{k_j,i} - \lambda_{k_j,i}}, \tag{6}$$

where $\lambda_{k_j,i}$ denotes the packet arriving rate of user k in slice j who associates with BS i, and is measured in packets per second. The expected instantaneous delay of slice j is expressed as;

$$\tau_j = \sum_{k_j=1}^{K_j} (\tau_{k_j,i}). \tag{7}$$

The average delay of user k in slice j, who associates with BS i is calculated as;

$$\bar{\tau}_{k_j,i} = x_{k_j,i} \cdot \tau_{k_j,i}. \tag{8}$$

The average delay of slice j, combining the average delay of all the users in the slice, is expressed as;

$$\bar{\tau}_j = \sum_{k_j=1}^{K_j} (\bar{\tau}_{k_j,i}). \tag{9}$$

The value of $\bar{\tau}_j$ should not exceed the maximum delay requirement of slice j i.e. $\bar{\tau}_j \leq \bar{\tau_j}^{max}$.

3.5 Utility Model

In this sequel, we assume that the cost utility of buyers is directly proportional to their data rate or delay satisfaction demand. For instance, if the data rate demand of MVNO j is high, its cost utility will eventually be high and vice versa. On the contrary, the pricing of sellers is associated with the competition among sellers. The data rate satisfaction of user k in slice j is modeled as a sigmoid function and is expressed as;

$$\psi\left(\bar{r}_{k_j}\right) = \frac{1}{1 + e^{-\eta\left(\bar{r}_{k_j} - \bar{r}_{k_j}^{min}\right)}}, \tag{10}$$

where η is a constant that determines the shape of the satisfactory curve, and $\bar{r}_{k_j}^{min}$ is the minimum data rate requirement of user k in slice j. The data rate satisfaction of slice j is calculated as;

$$\psi\left(\bar{r}_j\right) = \sum_{k_j=1}^{K_j} \psi(\bar{r}_{k_j}). \tag{11}$$

The delay satisfaction of user k in slice j is defined as;

$$\psi\left(\bar{\tau}_{k_j}\right) = \frac{1}{1 + e^{-\eta\left(\bar{\tau}_{kj}^{max} - \bar{\tau}_{k_j}\right)}}, \tag{12}$$

where $\bar{\tau}_{k_j}^{max}$ is the maximum tolerant delay requirement of user k in slice j. The delay satisfaction of slice j is expressed as;

$$\psi\left(\bar{\tau}_j\right) = \sum_{k_j=1}^{K_j} \psi(\bar{\tau}_{k_j}). \tag{13}$$

3.5.1 Utility of InP

The InP earns its utility by leasing spectrum to the MVNOs for initial slice creation. The utility of an *i-th* InP is the revenue received from leasing spectrum to MVNOs minus its overhead cost c_{ovh}, where c_{ovh} results from electricity consumption, equipment maintenance cost, and hardware loss [30]. Since a *j-th* MVNO is plagued with selecting a specific InP for spectrum leasing, we express a likelihood ϕ_{ij} for the *j-th* MVNO choosing an *i-th* InP to trade spectrum with. Taking the spectrum demand d_j of a *j-th* MVNO and the unit price δ_i of an *i-th* InP into account, the revenue of the InP is given by;

$$\mathcal{R}_i(d_j, \delta_i) = \sum_{j \in \mathcal{J}} \delta_i \cdot [d_j(\cdot)], \tag{14}$$

where $(\cdot)$ is substituted with $\psi\left(\bar{r}_j\right)$ or $\psi\left(\bar{\tau}_j\right)$ from (11) or (13) respectively, depending on the QoS requirement of the slice j. The utility $\mathcal{U}_i(d_j, \delta_i)$ of the *i-th* InP is expressed as;

$$\mathcal{U}_i(d_j, \delta_i) = \phi_{ij}(\mathcal{R}_i(d_j, \delta_i) - \sum_{j \in \mathcal{J}} (c_{ovh} \cdot d_j)). \tag{15}$$

3.5.2 Utility of Seller MVNO

From the business perspective, an *m-th* MVNO with redundant spectrum in a slicing period, can sublease a portion of it to other N buyer MVNOs who lack spectrum to satisfy their subscribers' QoS requirements. In return, the *m-th* seller MVNO receives revenue to minimize spectrum cost, which is expressed as;

$$\mathcal{R}_m(d_n, \delta_m) = \sum_{n \in \mathcal{J}_{buy}} \delta_m \cdot [d_n(\cdot)], \tag{16}$$

where $[d_n(\cdot)]$ is the spectrum demand of the *n-th* buyer MVNO based on data rate satisfaction $\psi\left(\bar{r}_n\right)$ or delay satisfaction $\psi\left(\bar{\tau}_n\right)$, and δ_m is the unit price of the *m-th* seller MVNO. The cost involved in purchasing spectrum from the InP is $\mathcal{C}_m(d_m, \delta_i) = \sum_{m \in \mathcal{J}_{sell}} \delta_i \cdot [d_m(\cdot)]$. We define the utility $\mathcal{U}_m(d_n, \delta_m)$ of *m-th* seller MVNO as;

$$\mathcal{U}_m(d_n, \delta_m) = \phi_{mn}(\mathcal{R}_m(d_n, \delta_m) - \mathcal{C}_m(d_m, \delta_i)), \tag{17}$$

where ϕ_{mn} is the probability of the *n-th* buyer MVNO to select an *m-th* seller MVNO for spectrum trading.

3.5.3 Utility of Buyer MVNO

Considering the spectrum demand d_k of a customer subscribed to an *n-th* buyer MVNO, and the unit price δ_n of an *n-th* buyer MVNO, the revenue obtained by an *n-th* buyer MVNO can be calculated as;

$$\mathcal{R}_n(d_k, \delta_n) = \sum_{k=1}^{K} \delta_n \cdot [d_k(\cdot)], \tag{18}$$

where $(\cdot)$ is $\psi\left(\bar{r}_k\right)$ or $\psi\left(\bar{\tau}_k\right)$ from (10) or (12) respectively, depending on the QoS requirement of the user k. This is the revenue gained from reselling the subleased spectrum to users of the *n-th* buyer MVNO. As the amount of spectrum increases, the cost of purchasing the spectrum increases exponentially. The cost of acquiring spectrum to serve users in the *n-th* buyer MVNO is $\mathcal{C}_n(d_k, \delta_m) = \sum_{k=1}^{K} \delta_m \cdot [d_k(\cdot)]$. We denote the utility of the buyer MVNO for spectrum trading by $\mathcal{U}_n(d_k, \delta_m)$ and is calculated as;

$$\mathcal{U}_n(d_k, \delta_m) = \phi_{nm}(\mathcal{R}_n(d_k, \delta_n) - \mathcal{C}_n(d_k, \delta_m)). \tag{19}$$

4 Problem Formulation

At slicing period t, the blockchain procedure runs concurrently with the resource management process. That is, the spectrum of MVNOs can be readjusted after initial slice creation through real-time spectrum trading.

4.1 Consortium Blockchain for Spectrum Trading

The spectrum trading market in the wireless network environment is vulnerable to adversaries and attacks, commonly double spending attack and hacking attack. Double spending occurs when the same RB is subleased to different buyers at the same time, causing business friction. Hacking attack occurs when a malicious node manipulates the transaction records stored on the digital ledger. Blockchain has the potential of mitigating such security issues in the business ecosystem, thanks to its unique features such as decentralization, immutability, security, and traceability.

As shown in the proposed architecture in Fig. 1, the entire decentralized network maintains data stored on the incorruptible digital ledger, promoting manipulation-proof and value transfer among the trustless nodes. Nevertheless, blockchain technology has some drawbacks related to its implementation cost, authorization issues, and slow consensus process. Therefore, we deploy a (permissioned) consortium blockchain platform to perform decentralized and robust spectrum trading. Consortium blockchain is preferred to (permissionless) public blockchain because, it provides entity authorization, while nullifying monopoly [31]. In addition, we design a consensus mechanism with moderate implementation cost such as practical Byzantine fault tolerance (PBFT) [32] consensus algorithm. We prefer PBFT because, there is no need for extensive block mining like in Bitcoin's proof of work (PoW), ensuring higher energy efficiency and fast consensus process, and is best for small-scale networks as our scenario. The entities of the blockchain network are the InPs and MVNOs, who also act as validators to earn extra incentives if pre-selected for block verification and audit. We present the detailed operation of our consortium blockchain for spectrum trading as follows:

4.1.1 System Initialization

Entities in a permissioned blockchain are first required to be authenticated by the regulator and given a certificate $Cert$, in order to become legitimate operators in the network. Then, the regulator generates a unique PK, SK, and Add for each of the authorized entities as transaction details of the tuple $\{Cert,\ SK,\ PK,\ Add\}$, which is sent as a test transaction to the legitimate entity. We rely on elliptic asymmetric cryptography, and curve digital signature algorithm for system initialization [25]. The asymmetric cryptography of information integrity from a sender z during system initialization is expressed as [33];

$$dec_{PK_z}(sign_{SK_z}(hash(msg))) = hash(msg), \tag{20}$$

where $sign_{SK_z}$ represents the digital signature of sender z with private key SK_z, dec_{PK_z} is used to decode the signed data of sender z's public key, and $hash(msg)$ is the hash digest of message, msg.

4.1.2 Validator Selection Based on Reputation

In a consortium blockchain network, not all entities are trusted to serve as validators. The regulator calculates the average reputation of each operator based on the feedback of *'opinions'* by other operators, using subjective logic model [34]. Subjective logic model is based on the history of interactions between entities, which is a probability of subjective beliefs. The subjective beliefs have three outcomes namely; belief (bel), disbelief $(disbel)$, and uncertainty $(uncert)$. Let us consider InP i and MVNO j, who interact during spectrum trading. The *'opinion'* of i to j can be expressed in terms of subjective logic as a vector $\mu_{(i\to j)} = \{bel_{(i\to j)}, disbel_{(i\to j)}, uncert_{(i\to j)}\}$. Similarly, the *'opinion'* of seller MVNO m to buyer MVNO n can be defined as $\mu_{(m\to n)}$. Note that $bel_{(i\to j)}, disbel_{(i\to j)}, uncert_{(i\to j)} \in [0, 1]$. The reputation $rep_{(i\to j)}$ of i to j can be expressed as;

$$rep_{(i\to j)} = bel_{(i\to j)} + \vartheta \cdot uncert_{(i\to j)}, \tag{21}$$

where ϑ indicates an effect of uncertainty for reputation. If the reputation exceeds a predefined minimum reputation threshold, then the node can be selected as a validator for the specific spectrum trading period.

4.1.3 Spectrum Trading Between Buyers and Sellers

In our proposed blockchain network, spectrum trading occurs in two levels. Firstly, $\mathcal{I}$ InPs act as the spectrum sellers to lease their spectrum to $\mathcal{J}$ MVNOs for slice creation. Secondly, underloaded MVNOs who have idle spectrum can join as a set of $\mathcal{J}_{sell}$ seller MVNOs, to sublease spectrum to a set of $\mathcal{J}_{buy}$ overloaded buyer MVNOs for slice spectrum adjustment. A spectrum buyer sends a request req containing its spectrum demand, to the nearest BS to be broadcast throughout the blockchain network. The spectrum sellers upon receiving the request, report their individual prices and available spectrum as response. Based on the supply-demand relationship of the entities, a pricing and demand prediction-based resource management problem is formulated. Each buyer and seller seeks to maximize its own strategy via an interest competition game, thanks to game theory. Depending on the outcome of the game, an SC is executed and the seller transfers the needed spectrum to the buyer for revenue in return. A detailed discussion on game theory is provided in the next subsection.

4.1.4 Block Creation and Broadcast

After successful spectrum exchange, one of the entities is selected as the leader node l to create a block for the current transaction. Based on the aggregated votes of all entities, the node with the highest vote becomes the leader for the block creation of the particular transaction until its acceptance onto the ongoing chain. The leader of each block need not be the same, as each block creation requires the nodes to vote on who becomes the leader. The selected leader records the transaction in a tamper-resistant block, encrypts, and digitally signs on it to guarantee block authenticity. Then, the leader broadcasts the block to all the nodes in the network for auditing.

4.1.5 Consensus Process

In order to incur moderate cost on block validation and confirmation, we deploy a lightweight PBFT consensus algorithm [32] for verifying the correctness of the block. Based on (21), the pre-selected nodes serve as validators to check for the correctness of the block, and report their audit results to the leader for analysis. By comparing the received block information, each validator sends a feedback to the leader. The leader analyzes the audit results of the validators and accepts the block, if consensus is reached. If the correctness of the block is approved by all validators, the leader broadcasts the accepted block, and the new block is added to the ongoing chain, which contains the hash of the previous block.

4.2 Three-Stage Stackelberg Game Modeling

The trading-based interactions among InPs, seller MVNOs, and buyer MVNOs can be modeled as a three-stage MLMF Stackelberg game, where each player ensures the maximization of its utility given other players' strategies. The leaders who are spectrum sellers move first by setting their unit prices. Then, the followers (spectrum buyers) accounting for the leaders' strategies, respond with their demands. Both leaders and followers can constantly adjust their strategies to earn more profit.

During the interaction between $\mathcal{I}$ InPs and $\mathcal{J}$ MVNOs for slice creation, an InP i acts first to set its unit price δ_i, and an MVNO j responds to the price by deciding its spectrum demand d_j. Therefore, the interaction between $\mathcal{I}$ InPs and $\mathcal{J}$ MVNOs can be formulated as an MLMF Stackelberg game, where InPs are the leaders and MVNOs are the followers. Similarly, during the interaction between $\mathcal{J}_{sell}$ underloaded seller MVNOs and $\mathcal{J}_{buy}$ overloaded buyer MVNOs, the seller m sets its unit price δ_m first, and the buyer n responds to set its demand d_n. The interaction between $\mathcal{J}_{sell}$ seller and $\mathcal{J}_{buy}$ buyer MVNOs can be formulated as an MLMF Stackelberg game, where seller MVNOs are leaders and buyer MVNOs are followers. We transform the three-stage Stackelberg game into an optimization problem as follows:

4.2.1 Stage I (InP Price Imposition)

The optimization goal of an InP is to maximize its utility in (15) by imposing appropriate price, which can be expressed as;

$$\underset{\delta_i}{max}\, \mathcal{U}_i(d_j, \delta_i), \tag{22}$$

$$s.t: \quad \sum_{j=1}^{J} d_j \leq \mathcal{B}_i^{max}, \tag{23}$$

where $\mathcal{U}_i(d_j, \delta_i)$ is the utility function of InP i, δ_i is the unit price vector with a price profile $\{\delta_i\}_{i\in\mathcal{I}}$ and d_j is the demand vector of MVNOs with a demand profile $\{d_j\}_{j\in\mathcal{J}}$. Constraint (23) states that the aggregated demand of the MVNOs cannot exceed the maximum bandwidth of the InP, $\mathcal{B}_i^{max}$.

4.2.2 Stage II (Seller MVNO Pricing)

Each seller MVNO is interested in optimizing its own utility in (17), by setting its pricing strategy to buyer MVNOs with the following optimization problem;

$$\underset{\delta_m}{max}\, \mathcal{U}_m(d_n, \delta_m), \tag{24}$$

$$s.t: \quad \delta_m \geq 0 \quad \forall m \in \mathcal{J}_{sell}. \tag{25}$$

It is noteworthy that an underloaded MVNO can suffer higher spectrum cost due to abundant spectrum leased from the InP. Therefore, each underloaded MVNO is motivated by the revenue it could gain from the resale of spectrum to overloaded MVNOs.

4.2.3 Stage III (Buyer MVNO Demand)

Given the spectrum price δ_m of the seller MVNO, the buyer MVNO seeks to maximize its utility in (19) by solving the following optimization problem;

$$\underset{d_k}{max}\, \mathcal{U}_n(d_k, \delta_m), \tag{26}$$

$$s.t: \quad \sum_{k=1}^{K} d_k \geq 0 \quad \forall k \in \mathcal{K}. \tag{27}$$

If the buyer MVNO subleases spectrum from the seller MVNO, it is able to satisfy the QoS requirements of its subscribers, and so its revenue increases.

We form the three-stage Stackelberg game from (22), (24), and (26) with an objective of finding an SE. An SE is the optimal outcome of the game, where neither of the players has an incentive to deviate from its strategy after considering the strategies of the other players.

Proposition 1: Given the optimal unit price of InP i as δ_i^* and the optimal demand of MVNO j as d_j^*, the SE is $(\delta^* = \{\delta_i\}_{i\in\mathcal{I}}, d^* = \{d_j\}_{j\in\mathcal{J}})$, if

1) For any $i \in \mathcal{I}$ given all MVNOs $j \in \mathcal{J}$ choose their optimal demands from InP i, and all the remaining InPs except InP i choose their optimal prices, then InP i chooses its optimal price δ_i^* to maximize its utility as $\mathcal{U}_i(\delta_i^*, \delta_{-i}^*, d^*(\delta)) \geq \mathcal{U}_i(\delta_i, \delta_{-i}^*, d^*(\delta))\forall i \in \mathcal{I}$.
2) For any MVNO $j \in \mathcal{J}$, given all InPs choose their optimal prices, MVNO j chooses its optimal demand d_j^* to maximize its utility $\mathcal{U}_j(d_j^*, \delta^*) \geq \mathcal{U}_j(d_j, \delta^*)\forall j \in \mathcal{J}$.

Proposition 2: We consider δ_m^* and d_n^* as the optimal unit price of the *m-th* seller MVNO and demand of the *n-th* buyer MVNO, respectively. Then, the point (δ_m^*, d_n^*) is the SE if for any (δ_m, d_n), $\delta_m \geq 0$, $d_n \geq 0$,

$$\mathcal{U}_m(\delta_m^*, d_n^*) \geq \mathcal{U}_m(\delta_m, d_n^*), \tag{28}$$

$$\mathcal{U}_n(\delta_m^*, d_n^*) \geq \mathcal{U}_n(\delta_m^*, d_n). \tag{29}$$

To verify the existence and uniqueness of the SE, we take the second order derivatives of the utility functions of InPs, seller MVNOs, and buyer MVNOs [35].

Lemma 1: *There is a unique equilibrium in buyer MVNO stage game.*

Proof: During buyer MVNO game, each buyer MVNO n determines the amount of spectrum to purchase, with the goal of maximizing its utility at given spectrum price δ_m. The buyer MVNO's utility function in (19) is continuous, and the second order derivative of the function is;

$$\frac{\partial^2 \mathcal{U}_n}{\partial {d_n}^2} = -2\mathcal{U}_n \frac{(\sum_{y\neq n} d_y)}{(\sum_{y\in N} d_y)^3}. \tag{30}$$

We can get $\frac{\partial^2 \mathcal{U}_n}{\partial {d_n}^2} \leq 0$, since $d_y \geq 0$ and $\mathcal{U}_n \geq 0$. Therefore, $\mathcal{U}_n$ is a strict concave function of variable d_n, and there exists

Algorithm 1 Blockchain-based Spectrum Trading Procedure

Initialize: The sets $\mathcal{I}$, $\mathcal{J}(\mathcal{J}_{sell}, \mathcal{J}_{buy})$, Regulator
Register: Register and authenticate $\mathcal{I}$, $\mathcal{J}(\mathcal{J}_{sell}, \mathcal{J}_{buy})$
for slicing period t **do**
  /Stackelberg game for trading and block creation/
  for all $\mathcal{I}$, $\mathcal{J}(\mathcal{J}_{sell}, \mathcal{J}_{buy})$ **do**
    Verify $Cert_{(i,m,n)}$ using batch verification
    if $Ver(Cert_i, Cert_m, Cert_n) = True$ **then**
      Set-up a three-stage Stackelberg game based on (22), (24), and (26)
      Execute SC for trading and create block
    else
      Terminate SC
    end if
  end for
  / Block verification and PBFT consensus /
  for validator leader l **do**
    Broadcast the current $BLK\ data$ to all nodes
    Select honest nodes for verification and audit based on reputation, using (21)
    for all validator nodes **do**
      if $validator\ Tx\ data$ = $BLK\ data$ **then**
        set $verify\ BLK$ = $True$
      else
        set $verify\ BLK$ = $False$
      end if
      Broadcast audit results to l for analysis
    end for
    Accept and add block to ongoing chain or discard block that fails verification
  end for
end for

a unique equilibrium in the buyer MVNO game.

Lemma 2: *At given spectrum price δ_m, the optimal amount of spectrum purchased by buyer MVNO n, is calculated as;*

$$d_n{}^* = \frac{min \sum_{n \in \mathcal{J}_{buy}} \delta_m \cdot [d_k(\cdot)]}{[d_k(\cdot)]}, D_n. \tag{31}$$

Proof: At a given δ_m, buyer MVNO n decides d_n by making the second order derivative of (19) equal to 0 as;

$$\frac{\partial^2 \mathcal{U}_n}{\partial d_n^2} = -2\frac{(\sum_{y \neq n} d_y)\mathcal{U}_n}{(\sum_{y \in N} d_y)^3} = 0. \tag{32}$$

Note that $d_n \leq D_n$. Therefore, the lemma is proven. Similar steps can be followed to prove the existence and unique equilibrium of seller MVNO and InP games, respectively via the following;

$$\frac{\partial^2 \mathcal{U}_m}{\partial \delta_m^2} = -2\mathcal{U}_m \frac{\mathcal{C}_m}{{\delta_m}^2} \frac{n-1}{n} \leq 0. \tag{33}$$

$$\frac{\partial^2 \mathcal{U}_i}{\partial \delta_i^2} = -2\mathcal{U}_i \frac{c_{ovh}}{{\delta_i}^2} \frac{j-1}{j} \leq 0. \tag{34}$$

It is proven in [36] that backward induction can be used to achieve SE for the formulated game. However, this method of finding SE requires all nodes to disclose their private information. More so, this practice may affect the fairness of the game. In contrast, DRL approach learns the optimal policy without prior knowledge. Therefore, we design a DRL-based method for obtaining joint optimal pricing and demand strategies for autonomous resource management.

4.3 Multi-Agent DRL-based Utility Optimization for Spectrum Management

The purpose of DRL is to find the joint optimal pricing and demand strategies of the entities involved, that solves the proposed Stackelberg game given little or no information. Deploying a single agent only maximizes its own cumulative reward, which is not in line with our scenario involving multiple players. Contrarily, a multi-agent DRL system seeks to maximize the whole reward of all the agents in the system, considering other agents' strategies [37]. The optimal pricing and demand prediction problem can be modeled as a Markov decision process (MDP), which is expressed as a stochastic process. At time step t, an agent selects an action a^t given a state s^t, and obtains an immediate reward $r^t(s^t, a^t)$ based on a state transition probability $P(s^{(t+1)}|s^t, a^t)$. A detailed MDP formulation for resource management can be found in [22]. From Markov property, the policy π can be obtained by;

$$\mathcal{V}^{\pi}(s) = \mathbb{E}_{\pi}\left\{ r^t(s^t, a^t) + \gamma \sum_{s'} P\left(s' \mid s^t, a^t\right) \mathcal{V}^{\pi}\left(s'\right) \right\} \tag{35}$$

where $r^t(s^t, a^t)$ is the present reward, $\mathcal{V}^{\pi}(s)$ is the present utility, and $\mathcal{V}^{\pi}(s')$ is the future utility. The state-value function for an optimal policy based on the Bellman equation [10] is given as;

$$\mathcal{V}^{\pi^*}(s) = arg \max_{a^t \in A} \{\mathcal{V}^{\pi}(s)\}, \tag{36}$$

4.3.1 State-Action Mapping

In each time step t, the InPs, seller MVNOs, and buyer MVNOs take actions sequentially. We define the states, actions, and rewards of the MDP to match the optimal pricing and demand prediction strategies as follows:

State(s): At time step t in *Stage I*, InP i first sets a unit price δ_i^t based on the demand $[d_j^{t-1}]_{j \in \mathcal{J}}$ of MVNO j at the previous time step $t-1$. In *Stage II*, the seller MVNO m sets its unit price δ_m^t based on $[d_n^{t-1}]_{n \in \mathcal{J}_{buy}}$. In *Stage III*, the buyer MVNO n observes the unit price δ_m^t to determine its demand d_n^t.

Definition 1: The states of the *i-th* InP, *m-th* seller MVNO, and *n-th* buyer MVNO at time step t, are expressed as:

$$\begin{aligned} s_i^t &= [d_j^{t-1}]_{j \in \mathcal{J}} \\ s_m^t &= [d_n^{t-1}]_{n \in \mathcal{J}_{buy}} \\ s_n^t &= \delta_m^t \end{aligned} \tag{37}$$

Action(a): Given state s^t, a DRL agent performs action a^t based on the observations. Based on the actions selected by the players, the DRL agents effect the changes on the spectrum of each entity.

Definition 2: The actions set of the *i-th* InP, *m-th* seller MVNO, and *n-th* buyer MVNO are as follows:

$$\begin{aligned} a_i^t &= \delta_i^t \\ a_m^t &= \delta_m^t \\ a_n^t &= d_n^t \end{aligned} \tag{38}$$

For simplicity, the action set of each InP, seller MVNO,

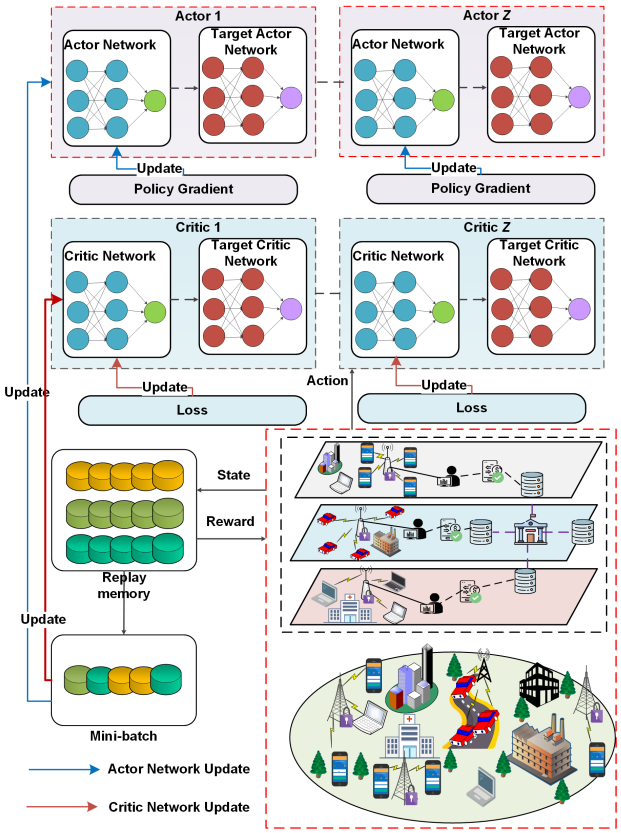


Fig. 2: MADDPG framework for Optimal Utility.

and buyer MVNO at time step t are defined as $a_i^t = \{1, 2, ..., 100\}$, $a_m^t = \{1, 2, ..., 50\}$, and $a_n^t = \{1, 2, ..., 50\}$, respectively. In this way, the players are able to dynamically adjust their actions to reflect their utilities.

Reward(r): We seek to obtain an immediate reward r^t that maximizes the utility of each player, without an incentive to deviate from the game.

Definition 3: The immediate reward of the *i-th* InP, *m-th* seller MVNO, and *n-th* buyer MVNO are as follows:

$$\begin{aligned} r_i^t &= \mathcal{U}_i^t(d_j^{t-1}, \delta_i^t) \\ r_m^t &= \mathcal{U}_m^t(d_n^{t-1}, \delta_m^t) \\ r_n^t &= \mathcal{U}_n^t(d_k^t, \delta_m^t) \end{aligned} \tag{39}$$

Accordingly, the system utility is the aggregated reward that the agents receive from the environment as; $r = \sum_{i=1}^{I}\sum_{m=1}^{M}\sum_{n=1}^{N}(r_i^t + r_m^t + r_n^t)$.

4.3.2 Stackelberg MADDPG for Optimal Policies

In order to achieve an SE, we consider a multi-agent learning method that can observe the dynamic environment and learn the strategies of other agents. Policy-based methods e.g. DDPG learn stochastic strategies effectively using function approximation, which solves the curse of dimensionality problem. Therefore, we deploy an MADDPG algorithm [38] named *Stackelberg MADDPG*, to achieve the SE of the formulated game. The DDPG architecture adopts an actor-critic approach that combines the gains of policy-based and value-based methods. By policy function, the actor generates an action given a state. The critic produces an action-value function and uses a loss function to criticize the actor's performance. Then, the actor uses DPG to approximate policies with the critic's output. DPG directly generates deterministic behavior policy, and avoids frequent action sampling. The critic updates the action-value function using gradient descent method [11].

The actor chooses an action a^t based on current state s^t and current policy π as;

$$a^t = \pi(s^t, \theta^\pi). \tag{40}$$

Based on the Bellman equation, the critic network calculates the target Q-value as;

$$y^t = r^t + \gamma \cdot Q^{'}(s^{'}, \pi^{'}, (s^{'}|\theta^{\pi\prime}), \theta^{Q\prime})). \tag{41}$$

Let π_i, π_m, and π_n be the set of policies for an *i-th* InP, *m-th* seller MVNO, and *n-th* buyer MVNO, respectively where $\pi_i = \{\pi_1, ..., \pi_I\}$, π_m=$\{\pi_1, ..., \pi_M\}$, and π_n=$\{\pi_1, ..., \pi_N\}$.

At *Stage I*, the critic network can be updated by minimizing the loss function as;

$$\begin{aligned} \mathcal{L}_i(Q_i) &= \mathbb{E}_{(s_i, a_i, r_i, s_i^{'})\sim\mathcal{D}_i}[(y_i - Q_i(s, a_i; \theta_i))^2] \\ y_i &= r_i + \eta \cdot Stackelberg Q_i(s^{'}), \end{aligned} \tag{42}$$

where $Stackelberg Q_i(s^{'}) = max_{a^{'}} Q_i(s^{'}, a_i^{'}, \theta_i)$ is the SE rewards under state $s^{'}$.

The policy gradient of the DPG objective function with respect to θ^{π_i} is given by;

$$\begin{aligned} \bigtriangledown_{\theta^{\pi_i}} J(\pi_i) = \mathbb{E}_{s, a_i\sim\mathcal{D}_i}[\bigtriangledown_{\theta^{\pi_i}} \pi_i(a_i, s_i) \bigtriangledown_{a_i} Q_i(s, a_i, \\ \theta^Q | a_i = \pi_i(s_i))], \end{aligned} \tag{43}$$

Algorithm 2 Stackelberg MADDPG-based Algorithm

Randomly initialize: Actor and critic evaluation networks with random weights θ^π and θ^Q, respectively
Initialize: Actor and critic target networks with weights $\theta^{\pi\prime} \leftarrow \theta^\pi$ and $\theta^{Q\prime} \leftarrow \theta^Q$, respectively
Initialize: Replay memory D and mini-batch $D^{'}$
for each episode **do**
  Set up the simulation environment
  for each decision step t, **do**
    for each agent **do**
      Observe state s^t based on (37)
      Execute **Algorithm 1**
      Select action a^t for exploration based on (40)
      Perform a^t, compute r^t and s^{t+1}
      Update resource pool at BS-level
      Store experience (s^t, a^t, r^t, s^{t+1}) in D
      Sample mini-batch of transitions from D
      Compute target value y^t using (41)
      Update critic network by minimizing loss $\mathcal{L}_i(Q_i)$ using (42)
      Update actor network by $\bigtriangledown_{\theta^{\pi_i}} J(\pi_i)$ using (43)
      Update target networks by soft update using (44)
    end for
  end for
end for

Finally, we update the target network of i, using soft update;

$$\begin{aligned} \theta^{\pi_i{'}} &\leftarrow \tau\theta^{\pi_i} + (1-\tau)\theta^{\pi_i{'}} \\ \theta^{Q_i{'}} &\leftarrow \tau\theta^{Q_i} + (1-\tau)\theta^{Q_i{'}} \end{aligned} \tag{44}$$

where τ denotes the learning rate. *Stage II* and *Stage III* follow a similar formulation to compute $\bigtriangledown_{\theta^{\pi_m}} J(\pi_m)$, $\mathcal{L}_m(Q_m)$, $\theta^{\pi_m{'}}$, $\theta^{Q_m{'}}$ for the *m-th* seller MVNO, and $\bigtriangledown_{\theta^{\pi_n}} J(\pi_n)$, $\mathcal{L}_n(Q_n)$, $\theta^{\pi_n{'}}$, $\theta^{Q_n{'}}$ for the *n-th* buyer MVNO.

The computational complexity of the MADDPG algorithm is expressed as $\mathcal{O}(\mathcal{G} \times |\mathcal{S}| \times |\mathcal{A}|)$, where $\mathcal{G}$ denotes the total number of agents, $\mathcal{S}$ denotes a state set, and $\mathcal{A}$ denotes an action set. Let the number of hidden layers be H and the dimension of the output be L. The complexity of each actor and critic network is $\mathcal{O}(|L|^2 H)$.

5 Performance Evaluation

5.1 Scenario Configuration

To evaluate the performance of our proposed *Stackelberg MADDPG* algorithm, we perform extensive simulations in a Python 3.6 environment with Tensorflow 2.0, running on a core i7 server of a 2.4GHz Intel Xeon CPU, and 16GB RAM. Given a coverage area of 500m $\times$ 500m, we consider 5 BSs that are 80m apart from one another, and 10 MVNOs with 50 users randomly distributed in each. The system bandwidth is set to 30MHz, which is divided into 150 RBs and each RB has a bandwidth of 180kHz. The BS transmit power is set to 46dBm, assuming negligible interference as a result of applying ICIC techniques [28]. To account for changing user traffic, random walk mobility model is preferred for modeling the user mobility [39].

TABLE 1: Simulation Parameters

Parameters and Units	Values
Number of BSs, $\mathcal{I}$	5
Number of MVNOs, $\mathcal{J}$	10
Number of users, $\mathcal{K}$	50 in each MVNO
System bandwidth, $\mathcal{B}$	30 MHz
Number of RBs, $\mathcal{W}$	150
Transmit power of BS, P_j	46 dBm
Network coverage area	500 m $\times$ 500 m
Noise power density, θ^2	-174 dBm/Hz
User distribution	Uniform
Slice minimum data rate ($\bar{r}^{min}$)	[Slice 1-4=500, Slice 5-7=10, Slice 8-10=15] kbps
Slice maximum delay ($\bar{\tau}^{max}$)	[Slice 1-4=100, Slice 5-7=10, Slice 8-10=100] ms
Number of validators	$\leq$ 15
Transactions per block	[5-10]KB
Minimum reputation threshold	0.5
Number of hidden layers(actor and critic)	4 (32 neurons in each)
Number of episodes	3000
Discount factor, γ_a, γ_c	0.9
Replay memory size, D	10^5
Mini batch size, D	128
Learning rate, τ_a, τ_c	0.01

As stated in ***Definition 2***, the action set of each InP, seller MVNO, and buyer MVNO at time step t are defined as $a_i^t = \{1, 2, ..., 100\}$, $a_m^t = \{1, 2, ..., 50\}$, and $a_n^t = \{1, 2, ..., 50\}$, respectively. Quantitatively, the maximum price of a unit spectrum (RB) is $\delta_{max} = 1$. Otherwise stated, all simulation results are averaged over a number of random independent runs. To implement the consortium blockchain, we set up a hyperledger Iroha platform [40] with SC on Ubuntu 16.04 LTS Bionic OS. Each of the MADDPG and DDPG models consists of a four-layer fully connected feed-forward neural network (NN) for each actor and critic network, with 32 neurons in each network. For the DQN and Dueling DQN algorithms, we configure a four-layer fully connected NN for each evaluation (prediction) network and target network, with 32 neurons in each network. We utilize ReLU activation function for the hidden layers, and $tanh(\cdot)$ to bound the actions. The replay memory and mini-batch sample sizes are set to 10^5 and 128, respectively. To optimize the loss, we adopt the *AdamOptimizer*. Simulation parameters are summarized in Table I.

5.2 Convergence Analysis

In this simulation, we evaluate the convergence of our proposed Stackelberg MADDPG algorithm compared with classical DDPG (DDPG) [11], dueling DQN [41], classical DQN (DQN) [22], and greedy approach (GA) [42], for the blockchain-empowered spectrum management system. We run the simulation for 3000 episodes, while averaging every 100 episodes for performance comparison. Fig. 3 shows the convergence on normalized system utility for the five algorithms. For the MADDPG algorithm, each agent selects an action independently to constitute the joint action policy of all the agents.

From Fig. 3, it can be observed that as the number of episodes increases, the normalized system utility increases until stability is reached. All five algorithms achieve convergence, which means that the optimal policy can be learned. The proposed Stackelberg MADDPG algorithm achieves the fastest convergence at approximately 500 episodes, and highest system utility of about 0.82. This is because, each agent observes the actions of other agents and makes a decision to maximize its utility, based on other agents' actions. The convergence performance of DDPG comes close

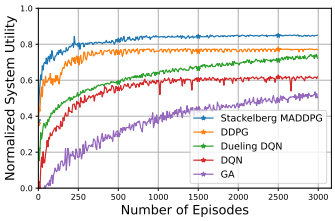


Fig. 3: Convergence analysis.

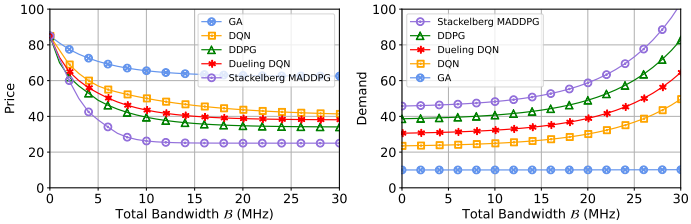
to that of the proposed algorithm with the reason being that, they both have a similar actor-critic framework. However, the DDPG algorithm deploys a single agent who maximizes its own cumulative reward. The next best performance is achieved by dueling DQN, with convergence at approximately 900 episodes and its system utility is about 0.62. This is because, dueling DQN has the ability of learning the important states and actions, without causing changes to the underlying DQN algorithm. Of the learning schemes, DQN achieves the slowest convergence and lowest utility at approximately 1000 episodes and 0.60, respectively, due to overestimation of its Q-values. Lastly, GA achieves the worst performance because, it does not consider future outcomes. We can conclude that the proposed Stackelberg MADDPG algorithm can best learn the optimal policy to maximize overall system utility, compared with the other baselines.

5.3 Analysis on Pricing and Demand Relationship

In this simulation, we evaluate the performance of the proposed Stackelberg MADDPG algorithm in terms of the pricing and demand relationship between the sellers and buyers in the trading market by comparing it with the baselines. We take into consideration Stackelberg I where 5 InPs trade spectrum with 10 MVNOs for slice creation. Fig. 4(a) and 4(b) show the pricing and demand trends of the five algorithms against the total bandwidth available for trading, respectively.

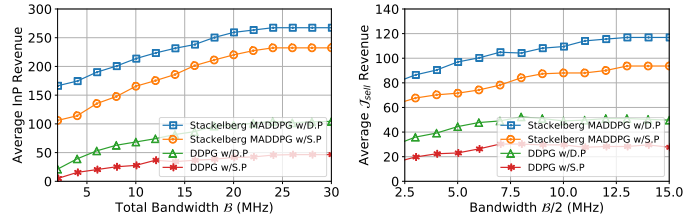
From Fig. 4(a), we observe that as the amount of bandwidth increases, the bandwidth price decreases in all the algorithms. For instance, with 5MHz bandwidth, the bandwidth price of Stackelberg MADDPG, DDPG, Dueling DQN, DQN, and GA are approximately 38, 50, 55, 59, and 70, respectively. With 30MHz bandwidth, the bandwidth prices decrease to about 25, 35, 43, 45, and 61, respectively. We observe this trend because with a small amount of bandwidth, the sellers set higher prices due to scarce resource. However, as the amount of bandwidth increases, the sellers have a large amount of goods to sell, so they lower their prices to stimulate consumption.

From Fig. 4(b), we observe that as the amount of bandwidth increases, the demand of MVNOs increases, with the proposed Stackelberg MADDPG algorithm achieving the highest demand followed by DDPG, dueling DQN, DQN, and GA in that order. The reason for this trend is that with an increasing bandwidth, the prices are reduced which motivates the MVNOs to buy bandwidth in order to earn more profits. For GA, there is no change in demand with increasing bandwidth amount probably due to lack of



(a) InPs' price vs. $\mathcal{B}$. (b) MVNOs' demand vs. $\mathcal{B}$.

Fig. 4: Pricing and demand relationship.



(a) InPs' revenue vs. $\mathcal{B}$. (b) $\mathcal{J}_{sell}$'s revenue vs. $\mathcal{B}/2$.

Fig. 5: Impact of static and dynamic pricing.

learning. In summary, we can conclude that the proposed algorithm is able to best match the pricing and demand strategies of the InPs and MVNOs due to its ability to find the SE which gives the optimal pricing and demand decisions.

5.4 Impact of Static and Dynamic Pricing

In this simulation, we evaluate the effect of dynamic pricing (D.P) and static pricing (S.P) on the proposed Stackelberg MADDPG algorithm, using DDPG as a benchmark. We plot the average revenue of the InPs against the total bandwidth in the system for Stackelberg I, in Fig. 5(a). In Fig. 5(b), we plot the average revenue of seller MVNOs against the total bandwidth of the MVNOs for Stackelberg II, assuming the sellers own half of the system bandwidth. Under D.P, the pricing profile of InPs and seller MVNOs are defined as $\delta_i = \{1, ..., 100\}$ and $\delta_m = \{1, ..., 50\}$, respectively. For S.P, the pricing profile is fixed at $\delta = 50$.

From Fig. 5(a) and 5(b), we observe that the revenue increases with increasing bandwidth, for both algorithms with S.P and D.P. It is clear that an increase in bandwidth implies more buyers need resource to serve their users and that the sellers obtain an increased revenue from the sale of their bandwidth. Towards the maximum bandwidth, the average sellers' revenue saturates as a result of buyers being satisfied and do not need extra acquisition of bandwidth. Therefore, the change in bandwidth does not affect revenue. For the same amount of bandwidth, Stackelberg MADDPG achieves higher average revenue than the DDPG algorithm under both S.P and D.P. This is because, Stackelberg MADDPG finds an SE where an optimal pricing is realized based on the buyers' demand strategies. Specifically, we observe in Fig. 5(a) that the Stackelberg MADDPG with D.P (Stackelberg MADDPG w/D.P) achieves a higher average InPs' revenue than Stackelberg MADDPG with S.P (Stackelberg MADDPG w/S.P). For instance, with 20MHz bandwidth, Stackelberg MADDPG w/D.P achieves InP revenue of about 250 whiles Stackelberg MADDPG w/S.P has revenue of about 220. Fig. 5(b) follows a similar trend as that in Fig. 5(a) i.e. Stackelberg MADDPG w/D.P achieves a higher average revenue of seller MVNOs than Stackelberg MADDPG w/S.P. We conclude that our proposed scheme with D.P can adjust sellers' prices efficiently to earn more revenue, compared with the other baseline algorithms.

5.5 Performance Analysis Based on Player Fairness

In this experiment, we evaluate the fairness of the trading market's players under the proposed Stackelberg MADDPG algorithm, and a baseline algorithm with a monopolistic InP. For simplicity in illustration, we consider only InPs as spectrum providers and all MVNOs as spectrum requesters, making the analysis suitable for Stackelberg I. Fig. 6 illustrates the performance of the proposed scheme and the baseline scheme, in terms of players' fairness.

From the figure, we observe that InPs in the proposed algorithm achieve a moderate average utility, whiles MVNOs achieve acceptable utility levels. With 20MHz bandwidth, the InPs and MVNOs achieve average utilities of approximately 200, and 98, respectively. This implies that each player constantly adjusts its utility to earn more profits without an incentive to deviate, creating a healthy and fair spectrum trading market. Under the baseline scheme, we see that the monopolistic InP achieves the highest average utility in the system. The monopolistic InP makes unreasonable profits because there is no competing InP to provide spectrum to the MVNOs. We can conclude that our proposed scheme maintains a fair and healthy market for all the players by keeping utilities at acceptable levels.

5.6 Impact of Varying Number of Spectrum Providers

In this simulation, we evaluate the effect of varied number of spectrum providers on the average utility of the spectrum requesters, using the proposed algorithm. We consider the spectrum trading activities in two-folds; between InPs and MVNOs in Stackelberg I, and between seller MVNOs and buyer MVNOs in Stackelberg II. For Stackelberg I, we consider 5 InPs as providers and 10 MVNOs as requesters. In Stackelberg II, we take 5 seller MVNOs as providers and 5 buyer MVNOs as requesters. Fig 7(a) shows the impact of varying InPs and Fig. 7(b) depicts the impact of varying seller MVNOs on the utilities of all MVNOs and buyer MVNOs, respectively.

From Fig. 7(a), we observe that as the number of InPs increases, the average utility of the MVNOs increases. This

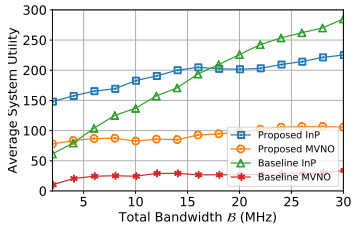
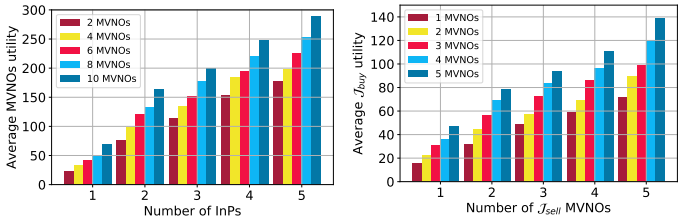
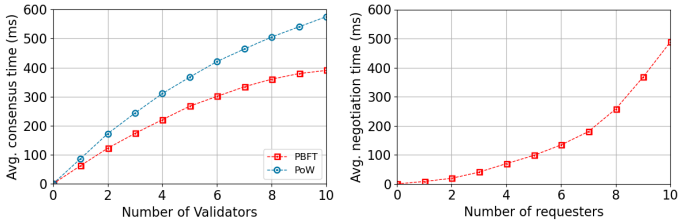


Fig. 6: Player fairness analysis.



(a) Impact of varying InPs. (b) Impact of varying $\mathcal{J}_{sell}$.

Fig. 7: Impact of changing no. of providers.



(a) Consensus time. (b) Negotiation time.

Fig. 8: Blockchain analysis.

is because, an increase in the number of spectrum providers implies an increase in spectrum for sale in the system. Therefore, the buyers are able to obtain the required amount of spectrum to serve their customers, which in turn increases their revenue. Also, an increasing number of MVNOs increases the average utility since an aggregated revenue of the MVNOs leads to increased utility. Specifically, we observe that with only one InP, the average utility of the MVNOs is very low. This is as a result of the one InP acting as a selfish player to create an unfair trading market environment, thereby forcing the MVNOs to incur much costs in purchasing spectrum to serve their customers.

In Fig. 7(b), we witness a similar trend as seen in Fig. 7(a). As the number of seller MVNOs increases, the average utility of the buyer MVNOs also increases. An increase in the number of spectrum providers necessarily means there are more options for the buyers to sublease spectrum from. With few buyers, the average utility is low while it rises as the number of buyers grows. However, a monopolistic seller (1 seller MVNO) affects the average utility of the buyers drastically due to its ability to force unfair prices on the buyers. We can conclude that the more the spectrum providers, the higher the average utility of the requesters.

5.7 Blockchain Performance Analysis

In this subsection, we seek to confirm the gains of the consortium blockchain platform by analyzing its performance on consensus time and negotiation time. The consensus time is the time needed by validators to vote, verify, and commit a transaction onto the block. Negotiation time is the time needed by the spectrum requesters to deliberate on the unit price and purchasing amount with the spectrum providers. We run this experiment taking the average of every 100 transactions to find the average consensus time and negotiation time. Fig. 8(a) and 8(b) show the performance of the consortium blockchain platform in terms of average consensus time and average negotiation time, respectively. For the performance on consensus time, we compare our PBFT consensus algorithm with Bitcoin's PoW [7]. We set the number of validators to at most 10.

From Fig. 8(a), we observe that as the number of validators increases, the average consensus time increases under both PBFT and PoW. However, PBFT takes less time to achieve consensus compared to PoW. This is so because, PoW is based on mathematical computation to mine the block and that it takes time to solve the puzzle, while PBFT is based on preselected nodes confirming the correctness of the block. With 2 validators, the time to reach a consensus is approximately 110ms under PBFT and 190ms under PoW. With 10 validators, the consensus time is at about 400ms for PBFT and 590ms for PoW. Thus, a higher number of validators involved in the PBFT consensus process means the leader will have to wait for all validators to submit their audit reports, analyze their reports and accept the block. However, the more the number of validators involved, the more robust the consensus mechanism.

To create a strong competition for acquiring spectrum, in Fig. 8(b), we consider 5 sellers as the spectrum providers and up to 10 spectrum requesters. We observe that as the number of requesters increases, the average time to negotiate transactions also increases. The reason for this trend is that the sellers will have to analyze the demands and QoS requirements of each requester. On the other hand, each requester has to analyze the pricing profile of each seller. Therefore, the negotiation time will eventually increase when the number of requesters increases.

5.8 Blockchain Security Assessment

Unlike conventional existing security solutions, the proposed method for secure spectrum trading relies on the consortium blockchain design, which has the ability to alleviate potential security issues in the trading environment. The blockchain-related security performance includes, but not limited to the following:

- *Non-reliance on a centralized third-party*: The entities in the consortium blockchain network trade spectrum in a P2P manner, without the need for a centralized trusted authority. Thus, all the entities are connected to the regulator in a decentralized manner, where encrypted transaction data is broadcast through the network and each entity has access to the data. In this case, centralized-based schemes' security threats such as single point of failure attack, is prevented since the data is distributed to all entities.
- *Preventing double spending*: Spectrum trading in the network can only materialize if the spectrum provider can prove beyond reasonable doubt that it truly owns the spectrum and that, it has not been allocated to any other entity in the network at the particular slicing period. The public history of transactions is cross-checked to ensure the ownership of the spectrum and its availability before the SC arbitrates. This solves the double-spending attack problem, and reduces business friction.
- *Privacy protection*: In the transaction process, the transaction data is encrypted with the private key of

the sender, which makes its impossible for the other entities to derive the raw transaction information. To preserve privacy, keys are generated randomly with strings of numbers so that it is mathematically impossible for an entity to guess the private key of the sender from its public key.

- *Transaction authentication*: To append a block to the ongoing chain, high-reputation pre-selected nodes publicly audit and verify the transactions and correctness of the block. Malicious nodes will have to race against time to tamper with transactions, which also comes with huge costs.

6 Conclusion

This paper proposed a novel hierarchical framework for a decentralized blockchain-enabled spectrum trading for slice creation and autonomous resource allocation in 5G RAN. Specifically, a consortium blockchain platform was developed to realize spectrum trading in two steps; trading between InPs and MVNOs for slice creation, and trading between underloaded MVNOs and overloaded MVNOs for slice spectrum adjustment. For a fair incentive mechanism, we formulated a three-stage Stackelberg game for the trading interactions among the players for joint optimal pricing and demand prediction strategies. Then, we designed a novel DRL-based algorithm named *Stackelberg MADDPG*, to achieve an SE for the formulated game. Security assessment showed that our proposed scheme ensures secure spectrum trading. Comprehensive simulation results analysis confirmed the performance gains of the proposed scheme in terms of convergence, fairness, and players' utility maximization. Future work will consider joint Stackelberg game and auction mechanism for spectrum trading.

Acknowledgments

This work is supported in part by the Natural Science Foundation of China under Grant No.61806040 and Grant No. 61771098; in part by the fund from the Department of Science and Technology of Sichuan Province under Grant No. 2020YFQ0025; and in part by the fund from Intelligent Terminal Key Laboratory of Sichuan Province under Grant No. SCITLAB-1018.

References

[1] C. Liang and F. R. Yu, "Wireless Network Virtualization: A Survey, Some Research Issues and Challenges," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 358–380, 2015.

[2] M. Richart, J. Baliosian, J. Serrat, and J.-L. Gorricho, "Resource Slicing in Virtual Wireless Networks: A Survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462–476, 2016.

[3] H. Xu, X. Qiu, W. Zhang, K. Liu, S. Liu, and W. Chen, "Privacy-Preserving Incentive Mechanism for Multi-Leader Multi-Follower IoT-Edge Computing Market: A Reinforcement Learning Approach," *Journal of Systems Architecture*, vol. 114, p. 101932, 2021.

[4] Z. Chang, D. Zhang, T. Hämäläinen, Z. Han, and T. Ristaniemi, "Incentive Mechanism for Resource Allocation in Wireless Virtualized Networks with Multiple Infrastructure Providers," *IEEE Transactions on Mobile Computing*, vol. 19, no. 1, pp. 103–115, 2020.

[5] Z. Xie, R. Wu, M. Hu, and H. Tian, "Blockchain-Enabled Computing Resource Trading: A Deep Reinforcement Learning Approach," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020, pp. 1–8.

[6] D. B. Rawat and A. Alshaikhi, "Leveraging Distributed Blockchain-based Scheme for Wireless Network Virtualization with Security and QoS Constraints," in *2018 International Conference on Computing, Networking and Communications (ICNC)*, 2018, pp. 332–336.

[7] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *Cryptography Mailing list at https://metzdowd.com*, 03 2009.

[8] Z. Liu, N. Cong Luong, W. Wang, D. Niyato, P. Wang, Y.-C. Liang, and D. In Kim, "A Survey on Applications of Game Theory in Blockchain," *arXiv e-prints*, p. arXiv:1902.10865, Feb. 2019.

[9] T. M. Ho, N. H. Tran, S. M. Ahsan Kazmi, and C. S. Hong, "Dynamic Pricing for Resource Allocation in Wireless Network Virtualization: A Stackelberg Game Approach," in *2017 International Conference on Information Networking (ICOIN)*, 2017, pp. 429–434.

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018. [Online]. Available: http://incompleteideas.net/book/the-book-2nd.html

[11] Z. Wang, Y. Wei, F. R. Yu, and Z. Han, "Utility Optimization for Resource Allocation in Edge Network Slicing Using DRL," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[12] I. da Silva, G. Mildh, A. Kaloxylos, P. Spapis, E. Buracchini, A. Trogolo, G. Zimmermann, and N. Bayer, "Impact of Network Slicing on 5G Radio Access Networks," in *2016 European Conference on Networks and Communications (EuCNC)*, 2016, pp. 153–157.

[13] B. Nour, A. Ksentini, N. Herbaut, P. A. Frangoudis, and H. Moungla, "A Blockchain-Based Network Slice Broker for 5G Services," *IEEE Networking Letters*, vol. 1, no. 3, pp. 99–102, 2019.

[14] L. Zanzi, A. Albanese, V. Sciancalepore, and X. Costa-Pérez, "NSBchain: A Secure Blockchain Framework for Network Slicing Brokerage," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.

[15] S. Zheng, T. Han, Y. Jiang, and X. Ge, "Smart Contract-Based Spectrum Sharing Transactions for Multi-Operators Wireless Communication Networks," *IEEE Access*, vol. 8, pp. 88 547–88 557, 2020.

[16] D. B. Rawat, "Game Theoretic Approach for Wireless Virtualization with Coverage and QoS Constraints," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2017, pp. 601–606.

[17] T. D. Tran and L. B. Le, "Stackelberg Game Approach for Wireless Virtualization Design in Wireless Networks," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[18] T. D. Tran, L. B. Le, T. T. Vu, and D. T. Ngo, "Stackelberg Game-Based Network Slicing for Joint Wireless Access and Backhaul Resource Allocation," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[19] A. Garcia-Saavedra and X. Costa-Pérez, "O-RAN: Disrupting the Virtualized RAN Ecosystem," *IEEE Communications Standards Magazine*, vol. 5, no. 4, pp. 96–103, 2021.

[20] I. Chih-Lin, S. Kuklinski, T. Chen, and L. Ladid, "A Perspective of O-RAN Integration with MEC, SON, and Network Slicing in the 5G Era," *IEEE Netw.*, vol. 34, pp. 3–5, 2020.

[21] M. Kist, J. F. Santos, D. Collins, J. Rochol, L. A. Dasilva, and C. B. Both, "AIRTIME: End-to-end Virtualization Layer for RAN-as-a-Service in Future Multi-Service Mobile Networks," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2020.

[22] G. Sun, G. O. Boateng, D. Ayepah-Mensah, G. Liu, and J. Wei, "Autonomous Resource Slicing for Virtualized Vehicular Networks With D2D Communications Based on Deep Reinforcement Learning," *IEEE Systems Journal*, vol. 14, no. 4, pp. 4694–4705, 2020.

[23] H. Zhang and V. W. S. Wong, "A Two-Timescale Approach for Network Slicing in C-RAN," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6656–6669, 2020.

[24] P. K. Sharma, S. Singh, Y.-S. Jeong, and J. H. Park, "Distblocknet: A Distributed Blockchains-Based Secure SDN Architecture for IoT Networks," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 78–85, 2017.

[25] N. Z. Aitzhan and D. Svetinovic, "Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 840–852, 2018.

[26] A. Papa, M. Klugel, L. Goratti, T. Rasheed, and W. Kellerer, "Optimizing Dynamic RAN Slicing in Programmable 5G Networks," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[27] M. Mamman, Z. M. Hanapi, A. Abdullah, and A. Muhammed, "Quality of Service Class Identifier (QCI) Radio Resource Allocation Algorithm for LTE Downlink," *PLOS ONE*, vol. 14, no. 1, pp. 1–22, 01 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0210310
[28] K. Wang, F. R. Yu, and H. Li, "Information-Centric Virtualized Cellular Networks With Device-to-Device Communications," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 11, pp. 9319–9329, 2016.
[29] S. M. Ross, *Introduction to Probability Models*, 6th ed. San Diego, CA, USA: Academic Press, 1997.
[30] H. Yao, T. Mai, J. Wang, Z. Ji, C. Jiang, and Y. Qian, "Resource Trading in Blockchain-Based Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3602–3609, 2019.
[31] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani, "Applications of Blockchains in the Internet of Things: A Comprehensive Survey," *IEEE Communications Surveys Tutorials*, vol. 21, no. 2, pp. 1676–1717, 2019.
[32] M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance," in *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, ser. OSDI '99. USA: USENIX Association, 1999, p. 173–186.
[33] Z. Su, Y. Wang, Q. Xu, M. Fei, Y.-C. Tian, and N. Zhang, "A Secure Charging Scheme for Electric Vehicles With Smart Communities in Energy Blockchain," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4601–4613, 2019.
[34] J. Kang, R. Yu, X. Huang, M. Wu, S. Maharjan, S. Xie, and Y. Zhang, "Blockchain for Secure and Efficient Data Sharing in Vehicular Edge Computing and Networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2019.
[35] Y. Fan, Z. Jin, G. Shen, D. Hu, L. Shi, and X. Yuan, "Three-Stage Stackelberg Game Based Edge Computing Resource Management for Mobile Blockchain," *Peer-to-Peer Networking and Applications*, vol. 14, pp. 1–15, 05 2021.
[36] J. Qiu, D. Grace, G. Ding, J. Yao, and Q. Wu, "Blockchain-Based Secure Spectrum Trading for Unmanned-Aerial-Vehicle-Assisted Cellular Networks: An Operator's Perspective," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 451–466, 2020.
[37] L. Busoniu, R. Babuska, and B. De Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, 2008.
[38] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6382–6393.
[39] Kuo-Hsing Chiang and N. Shenoy, "A Random Walk Mobility Model for Location Management in Wireless Networks," in *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications. PIMRC 2001. Proceedings (Cat. No.01TH8598)*, vol. 2, 2001, pp. E–E.
[40] F. Muratov, A. Lebedev, N. Iushkevich, B. Nasrulin, and M. Takemiya, "YAC: BFT Consensus Algorithm for Blockchain," *ArXiv*, vol. abs/1809.00554, 2018.
[41] N. Van Huynh, D. Thai Hoang, D. N. Nguyen, and E. Dutkiewicz, "Optimal and Fast Real-time Resource Slicing With Deep Dueling Neural Networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1455–1470, 2019.
[42] S. Najeh, H. Besbes, and A. Bouallegue, "Greedy Algorithm for Dynamic Resource Allocation in Downlink of OFDMA System," 10 2005, pp. 475 – 479.

Gordon Owusu Boateng received his Bachelor degree in Telecommunications Engineering from the Kwame Nkrumah University of Science and Technology (KNUST), Kumasi-Ghana, West Africa, in 2014 and his master degree in Computer Science in University of Electronic Science and Technology of China (UESTC), where he is currently pursuing his Ph.D. degree. From 2014 to 2016, he worked under sub-contracts for Ericsson (Ghana) and TIGO (Ghana). Till now, Gordon has co-authored over 20 scientific papers in conferences and journals. He is also a member of the Mobile Cloud-Net Research Team-UESTC. His interests include Mobile/Cloud Computing, 5G Wireless Networks, Data Mining, D2D communications, Blockchain, Game Theory, and SDN.

Guolin Sun received his B.S., M.S. and Ph.D. degrees all in Comm. and Info. System from the University of Electronic Science and Technology of China the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2000, 2003 and 2005 respectively. Since he finished his Ph.D. study in 2005, Dr. Guolin has got eight years industrial work experiences on Information and Communication Techniques (ICT) research and development for LTE, Wi-Fi, Internet of Things, Cognitive radio, Localization and navigation. Before he joined the UESTC, as an Associate Professor on Aug. 2012, he was with Huawei Technologies, Stockholm, Sweden. Till now, Dr. Guolin Sun has filed over 40 patents, and published over 70 scientific conference and journal papers, acts as TPC member and keynote speakers of many conferences. His general research interest is artificial intelligence, network virtualization, edge computing, blockchain techniques, resource management and vehicle networks.

Daniel Ayepah-Mensah received his Bachelor in Computer Engineering from Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana in 2014 and his master degree in Computer Science in University of Electronic Science and Technology of China (UESTC), where he is currently pursuing his Ph.D. degree. From 2014 to 2017, he worked as a software developer. He is also a member of the Mobile Cloud-Net Research Team – UESTC. His interest includes generally wireless networks, big data and cloud computing.

Daniel Mawunyo Doe received the bachelor's degree in computer engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2018. He is currently pursuing the M.Sc. degree in computer science and engineering with the University of Electronic Science and Technology of China (UESTC). From 2019 to 2021, He is a member of the intelliGame Team - UESTC. His general research interests include game theory, federated learning, wireless networks, big data, and cloud computing.

Ruijie Ou received his B.S degree and M.S degree in Computer Science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2012 and 2014, respectively. He began to study for his Ph.D. degree in 2017. Ruijie Ou has participated in 4 projects and published 3 journal papers.Before 2018, he was also a counselor in the School of Computer Science and Engineering, UESTC. Now, he is a secretary of Graduate Work Department and the assistant to the dean of Yibin Park of UESTC, Yibin,China . His research interests include pattern recognition, neural networks, and machine learning.

Guisong Liu received the B.S. degree in mechanics from Xi'an Jiao Tong University, Xi'an, China, in 1995, and the M.S. degree in automatics and the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, in 2000 and 2007, respectively. He was a Visiting Scholar with Humbolt University, Berlin, Germany, in 2015. Before 2021, he was a Professor with the School of Computer Science and Engineering, the University of Electronic Science and Technology of China. He is currently a Professor and the Dean of the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu. He has filed over 20 patents, and published over 70 scientific conference and journal papers. His research interests include pattern recognition, neural networks, and machine learning.