

© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Chen, Miaojiang, Anfeng Liu, Neal N. Xiong, Hongbing Song, and Victor C. M. Leung. "SGPL: An Intelligent Game-Based Secure Collaborative Communication Scheme for Metaverse over 5G and Beyond Networks." IEEE Journal on Selected Areas in Communications, 2023, 1–1.  
<https://doi.org/10.1109/JSAC.2023.3345403>.

<https://doi.org/10.1109/JSAC.2023.3345403>

Access to this work was provided by the University of Maryland, Baltimore County (UMBC) ScholarWorks@UMBC digital repository on the Maryland Shared Open Access (MD-SOAR) platform.

**Please provide feedback**

Please support the ScholarWorks@UMBC repository by emailing [scholarworks-group@umbc.edu](mailto:scholarworks-group@umbc.edu) and telling us what having access to this work means to you and why it's important to you. Thank you.

# SGPL: An Intelligent Game-based Secure Collaborative Communication Scheme for Metaverse over 5G and Beyond Networks

Miaojiang Chen, Anfeng Liu, Neal N. Xiong, *Senior member, IEEE*, Hongbing Song, *Fellow, IEEE*, Victor C. M. Leung, *Life Fellow, IEEE*

**Abstract**—Human-centric communication metaverse relies on the convergent integration of multiple existing technologies such as 5G and beyond networks, virtual reality, augmented reality, and digital twins, and thus their security vulnerabilities and vulnerability to interference attacks may also be inherited by the metaverse. In particular, existing security policies may be inefficient for communication interference problems encountered in multi-device collaborative computing in a metaverse 5G environment and lack adaptability to metaverse applications. In this paper, we propose a novel intelligent game anti-interference collaborative computing model that accurately describes the interference relationships among source devices, cooperating devices, and interferers in metaverse collaborative computing. We model the offensive and defensive confrontation between multiple metaverse devices as a Stackelberg game, where the source device is the leader, the collaborative computing device acts as the sub-leader, adjusts its antijamming strategy according to the source device's strategy to improve the source device's communication anti-jamming performance, and the jammer acts as the follower. We design an intelligent Stackelberg Game-theoretic Policy-based Learning (SGPL) algorithm for jamming resistance in metaverses over 5G and Beyond Networks, where the leaders (co-computing devices) update their training parameters using the total derivatives of the objective function, while the followers (i.e., jammers) update their training parameters using an independent gradient dynamics strategy. Loops are eased and convergence is accelerated by introducing differential dynamics into the leader training network to reflect the interaction structure of the critic and actor network layers. Finally, numerical results demonstrate the effectiveness of the proposed SGPL algorithm in metaverse anti-jamming countermeasures. The proposed SGPL algorithm has the potential to be generalized to other metaverse applications with multi-user attack and defense characteristics.

**Index Terms**—Metaverse, Human-centric communication, collaborative computing, anti-jamming, 5G and Beyond Networks.

## I. INTRODUCTION

THE metaverse is a digital living space with a new social system. Human-centric communication is achieved by meeting service requirements, whether necessary (e.g., a good communication environment) or desired (e.g., efficient computing), to achieve high-performance and low-cost communication goals. Human-centric communication metaverse is a highly developed virtual world that exists in parallel with the real world but reacts to the real world [1], [2], [3]. However, the metaverse has not yet brought completely independent new technologies, and is still dependent on the convergence and integration of many existing technologies such as 5G and beyond networks [4], Internet-of-Things [5], blockchain [6], [7], virtual reality [8], and digital twin [9].

Although human-centric communication in metaverse is a promising application technology, inherent security and privacy concerns have hindered its widespread deployment and adoption [10], [11]. From the management of massive data streams, attack interference in device communication, and inefficient optimization of intelligent algorithms, various security vulnerabilities and privacy issues continue to arise with the Metaverse. The fundamental reason is that since the metaverse integrates various network architectures and deep learning technologies as its foundation, its inherent vulnerabilities and inherent flaws may also be inherited by the metaverse, such as e-money theft, hijacking of wearable VR (AR) devices, information fraud, and other behaviors. In particular, metaverse builds virtual scenarios based on 5G edge networks, where users wear devices with AR/VR to achieve the collection and extraction of brain bioelectric signals, body range of motion, limb movement trajectories, facial expressions, and voice feature data, which will be extremely dependent on the collaborative computing technology in 5G networks. However, the inherent communication security characteristics of collaborative computing in 5G edge networks allow jammers to achieve interference with metaverse devices during communication by emitting jamming signals, thus threatening personal safety and even threatening the metaverse communication security by hijacking the channel.

In this paper, we focus on the anti-interference problem in the communication transmission of metaverse collaborative

Manuscript received xx, 2023; revised xx, 2023. This work was supported in part by the National Natural Science Foundation of China (62072475, 61772554). (Corresponding author: Anfeng Liu, N. Xiong.)

Miaojiang Chen is with School of Computer Science and Engineering, Central South University, Changsha 410083, China, Miaojiang Chen is also with the Guangxi Key Laboratory of Multimedia Communications and Network Technology, School of Computer, Electronics and Information, Guangxi University, Nanning 530004, China. (e-mail: miaojiangchen@foxmail.com).

Anfeng Liu is with School of Computer Science and Engineering, Central South University, Changsha 410083, China. (e-mail: afengliu@mail.csu.edu.cn).

Neal N. Xiong is Department of Computer Science and Mathematics, Sul Ross State University, Alpine, TX 79830, USA. (e-mail: xionгнаixue@gmail.com).

Houbing Song is with the Department of Information Systems, University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250 USA (e-mail: h.song@ieee.org).

V. C. M. Leung is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China, and also with the Department of Electrical and Computer Engineering, the University of British Columbia, Vancouver, BC V6T 1Z4, Canada (e-mail: vleung@ieee.org).

computing, where metaverse source devices and collaborative computing devices accomplish computational tasks by cooperating in an environment where malicious interference exists. A malicious attacker (jammer) learns the communication policy of a legitimate device through artificial intelligence techniques and then changes its jamming policy to block the channel communication of the legitimate device. During the continuous learning process, the jammer optimizes its jamming policy based on the experience of the communication policy of the legitimate device. On the other hand, the legitimate device senses the presence of the attacker through changes in the channel state and learns the optimal anti-interference channel strategy through DRL, e.g., using optimal power control to counteract the interfering signal, while weighing the anti-interference cost and the performance of collaborative computing. Note that previous anti-jamming techniques for 5G edge networks are based on deep Q-networks to implement attack and defense strategies; however, the value-based DRL algorithm cannot solve the problem of very slow convergence, or even failure to converge, due to channel state explosion. Therefore, it is crucial to improve the convergence speed of the training algorithm.

Based on the above analysis, we propose an intelligent SGPL algorithm for human-centric communication jamming resistance in metaverses over 5G and beyond networks to solve the high-dimensional channel explosion problem in the meta-universe multi-user dynamic channel anti-jamming game. Existing policy-based reinforcement learning algorithms use independent gradient dynamics to optimize the key and participants, which leads to ignoring the interaction structure between followers and leaders. Therefore, in our SGPL algorithm, the leader uses an implicit objective function to update the total derivatives, while the followers use individual gradient dynamics to update their parameters, thus enabling an iterative process that reflects the interaction structure to find the Stackelberg game equilibrium.

The main contributions of our researches are:

- 1) Previous human-centric Communication metaverse jamming models usually consider only two users: the defender (source user) and the attacker (jammer), which is difficult to work in metaverse collaborative computing with three types of attacking and defending users (i.e., source user, collaborative computing user, and jammer). Based on the inherent properties of collaborative computing, we propose a novel intelligent game-resistant metaverse collaborative computing model, which accurately describes the game relationship among source users, collaborative computing users and jammers. And we model the offensive and defensive confrontation scenario between the three users as a Stackelberg game, where the source user device is the leader, the collaborating user devices act as sub-leaders, they adjust their strategies according to the source user device's strategies to improve the anti-interference performance of the source user device, and the interferers act as followers to attack the source user. Subsequently, we analyze the policy optimization problem for the three meta-universe users during the game and prove the Stackelberg equilibrium among the three users.

- 2) We design an intelligent SGPL algorithm for jamming

resistance in metaverses over 5G and Beyond Networks, where the leaders (co-computing devices) update their training parameters using the total derivatives of the target and the follower (i.e., the jammer) updates its parameters using independent gradient dynamics. The previous gradient descent works are calculated separately using a independent gradient descent in a multi-layer network, so that the sub cycle would repeat a lot of times. In addition, using independent gradient dynamics to independently optimize multiple objectives will cause implicit neglect of the interaction structure between users. We introduce differential dynamics into the leader's network to update its parameters to reflect the interactive structure of critic and actor network layers, which can ease the cycle and accelerate convergence. Based on the above analysis, Stackelberg policy-based learning meta-framework is proposed. We finally prove the Stackelberg equilibrium of proposed SGPL algorithm.

- 3) We have conducted a large number of experiments to verify the performance of the proposed algorithm. Numerical results show that the anti-jamming performance of our proposed algorithm in the metaverse collaborative computing environment is better than that of SOTA algorithms.

The rest of our paper is organized as follows. We analyzed in detail the related work in section II. In Section III, we design a collaborative anti-jamming game system model. The SGPL algorithm with the Stackelberg equilibrium is derived in Section IV. Numerical results are presented in Section V. Finally, Section VI concludes our work.

## II. RELATED WORK

Due to the inherent nature of metaverse collaborative computing, metaverse devices are vulnerable to external jamming attacks or threats when performing tasks. In this section, we focus on reviewing the work related to anti-jamming.

There are five common types of collaborative computing interference in metaverse built based on 5G edge network technology: deceive [12], elimination [13], confrontation [14], hide [15], and avoidance [16]. To mitigate the collaborative computing interference, delay switching [17], power control [18], jamming detection and avoidance [19], and frequency hopping [20] are the preferable choices for metaverse anti-interference strategies. With the development of artificial intelligence technology, metaverse collaborative computing anti-interference based on deep reinforcement learning (DRL) has become a new research direction [21]. On the one hand, the combination of malicious interference and artificial intelligence has produced more efficient and complex interference patterns. On the other hand, deep reinforcement learning technology also provides new techniques to improve the anti-interference of metaverse collaborative computing. Therefore, DRL anti-interference techniques are a trend. In particular, DRL based on game theory is a very promising solution because it has the following advantages in terms of channel anti-interference.

- 1) The purpose of malicious jamming is to impede legitimate users' communications and degrade or interrupt their ability to communicate. However, for legitimate users, they

seek to maximize the elimination of various malicious interference attacks in order to achieve reliable communication quality of service. There are adversarial interactions between interferers and legitimate users, so game theory can simply and adequately describe the various types of interactions between interferers and illegitimate users.

2) For both the jammer and the communication user, it is difficult for both sides to obtain accurate information about each other. To make matters worse, the network environment for jamming attack and defense is dynamic because the time-varying channel is constantly changing. Therefore, such problems need to deal with both incomplete information and dynamic constraints. Deep reinforcement learning is an efficient technique in the interaction of incomplete information and dynamic environments.

For weak jamming, Navda *et al.* [15] innovatively proposed to use channel hopping to protect WIFI from external jamming. The results show that the channel hopping implemented by the real WIFI network can achieve 60% of the communication performance under the condition of jamming attacks. To improve the long-term blocking problem caused by channel hopping anti-jamming, Chang *et al.* [22] proposed two novel synchronous and asynchronous channel hopping strategies. Kotsiou *et al.* [23] proposed a distributed black-list frequency hopping technology to identify the interfered wireless channels and avoid using the interfered wireless channel to transmit information. Since then, MIMO-based anti-jamming strategies have been widely used in wireless communication scenarios with temporary jamming. Yan *et al.* [24] proposed a MIMO anti-jamming scheme for orthogonal frequency division multiplexing. First, analyze the impact of reactive jamming on user device, and then use the software defined radio signal to treat the jamming signal as noise and then eliminate it. However, the above work needs to obtain the channel state information of jammer and user device in advance. Then, to solve this problem, Yan *et al.* [25] propose a MIMO anti-jamming strategy that does not need to fully know the jamming channel, as long as the signal-to-noise ratio is large enough. Jamming suppression strategies are also widely concerned. Wu *et al.* [26] use non collaborative game to solve the channel jamming problem in spectrum sharing. Zheng *et al.* [27] propose a novel jamming mitigation game based on the characteristics of non spatial network scenarios. In addition, considering the characteristics of cyberspace, Xu *et al.* [28] propose a graphical game, and the MEC offloading decision problem is modeled as a collaborative game model. Unfortunately, the graphical game only considers the jamming between adjacent devices, but in practical applications, the cumulative weak jamming effect is equally important. In the MEC environment, the dense user environment may cause the cumulative jamming to exceed the threshold, thus forming strong jamming. To alleviate this problem, Sun *et al.* [29] propose a hypergraph anti jamming strategy to dynamically compete for channel resources and reduce channel jamming between users.

The above anti-jamming strategy does not consider external interference. Malicious interference from outside is one of the main threats to wireless networks, and it will greatly impair

the communication performance of user devices. To resist malicious jamming from jammers, Jia *et al.* [30] proposed a Bayesian Stackelberg game to implement the anti-jamming strategy under incomplete information. For the intelligent jamming attack in cognitive radio, Xiao *et al.* [31] proposed a game-based power control algorithm, which can achieve high signal to jamming noise ratio and network utility. Yu *et al.* [32] studied the anti-jamming game under multi-user and single jamming source, and deduced the Nash equilibrium. Xiao *et al.* [33] model the interaction between equipment and jammer as a game model, and predict the jamming state with the goal of maximizing their respective signal-to-noise ratio. Hanawal *et al.* [34] proposed a Markov game model against external jamming to achieve intelligent defense.

In early research, Wu *et al.* [35] have heuristically used the channel shopping strategy to avoid jamming in the Markov decision model. This anti-jamming strategy based on reinforcement learning models the worst case as the Nash Equilibrium (NE) of the game to enhance the network anti-jamming performance. For high jamming environment, Min *et al.* [36] proposed a Colonial Blotto game approach to realize Nash equilibrium iteration. However, this algorithm cannot guarantee long-term power control under jamming. Gouissem *et al.* [37] used gradient descent iteration to achieve Nash equilibrium and realize jamming dynamic adaptive game. However, although Gouissem believes that the Nash equalization in the proposed system is unique, when the number of sub carriers in the channel increases, the above scheme will become non scalable.

However, the anti-jamming approaches mentioned above are based on an application scenario with two game users (i.e., jammers and defenders) and cannot be applied to a wider range of user types. In contrast, real-time rendering of the metaverse is a computationally intensive task with high resource requirements. Therefore, the metaverse is more applicable to edge co-computing application scenarios. Our proposed SGPL algorithm uses the metaverse edge co-computing model to not only solve the computationally intensive task of real-time rendering in metaverse vehicular networks, but also to resist malicious interference and ensure the privacy of metaverse users.

### III. SYSTEM MODEL AND DEFINITIONS

In this paper, we consider a human-centric communication metaverses collaborative computing attacked by intelligent jammer, where an adjacent user helps the source user offloading a task on single channel to the edge server. The Framework of anti-jamming is shown in Fig. 1, which includes anti-jamming game model and policy-based learning algorithm. Next, we give a specific anti-jamming game model. In the jamming game, the jammer transmits noise power to achieve interference, and then legitimate users can avoid interference in two ways: 1) select other channels that have not received interference to continue to complete their own transmission tasks; 2) The noise power of the jammer is offset by increasing the transmission power. Both the source user and the adjacent user are legitimate users, and they and the



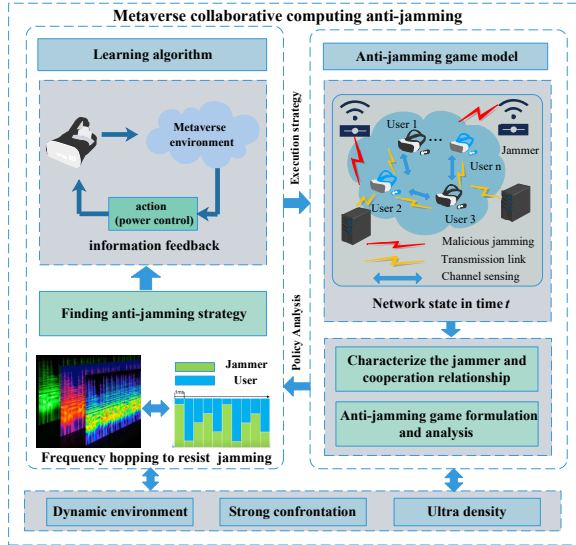


Fig. 1: Framework of game-theoretic learning anti-jamming metaverse collaborative computing.

jammer can intelligently adjust their own transmission power. Therefore, both legitimate users and jammers can approximate their maximum utility through intelligent power control. To help the source user achieve jamming free communication, the adjacent user modifies its own transmission power on the premise of obtaining the transmission power of the source user device. Furthermore, intelligent jamming determines how much jamming power is used to interfere with the channel according to the transmission power of legitimate users, thus damaging the task transmission performance of legitimate users. Obviously, the decision-making between three users to maximize their own interests can be modeled as a Stackelberg game model. Therefore, we model the anti-jamming power control process in the metaverse collaborative computing network as a Stackelberg game, in which each player obtains his own optimal power policy to achieve jamming (jammer) and anti-jamming (legitimate user). In this Stackelberg game, the source user device is a leader, the collaborating user device (i.e., adjacent users) is a vice leader, and intelligent jammer is a follower. We first give the game utility function of the source user, the collaborating user and the jammer. We define  $u = s, c, j$  as the source user device, the collaborating user device and jammer, respectively. Let  $P_u \in (0, P_{max})$  be the transmission power of three players and  $P_{max}$  is maximum transmission power supported by user devices,  $G_u$  is the fading channel gain. The user channel transmission rate is defined as:

$$\gamma = W \log \left( \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} \right), \quad (1)$$

where  $\sigma$  is the noise power,  $W$  is the bandwidth.

The target of the source user is to maximize its own data transmission rate. Therefore, we define a benefit function for the source user, that is, communication gain minus communication cost:

$$B_s(P_s, P_c, P_j) = \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} - C_s P_s, \quad (2)$$

where  $C_s$  is the transmission cost of user  $s$ .

The benefit function for the collaborating user device can be defined as:

$$B_c(P_s, P_c, P_j) = \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} - C_c P_c, \quad (3)$$

where  $C_c$  is the transmission cost of user  $c$ .

For an intelligent jammer, its purpose is to destroy the transmission benefit of legitimate users by transmitting jamming signals. Based on this, we can define the benefit function of the intelligent jammer as:

$$B_j(P_s, P_c, P_j) = -\frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} + C_c P_c + C_s P_s - C_j P_j, \quad (4)$$

where  $C_j$  is the transmission cost of intelligent jammer  $j$ .

In collaborative computing network, each legitimate user and jammer maximize their own benefits according to the transmission power strategy adopted by the opponent. Obviously, from the above model, we know that we can get the maximum benefit without considering the cost, as long as the transmitting power of legitimate users and jammers is high enough. However, in metaverse networks, the maximum transmission power of metaverse devices is  $P_{max}$  will be limited, and the high-power transmission signal of the jammer will also be easily captured and punished, so their transmission power has huge costs. Based on this, we obtain the optimal power control strategy:

$$P_s^* = \arg \max_{P_s \geq 0} B_s(P_s, P_c', P_j'), \quad (5)$$

$$P_c^* = \arg \max_{P_c \geq 0} B_s(P_s', P_c, P_j'), \quad (6)$$

$$P_j^* = \arg \max_{P_j \geq 0} B_s(P_s', P_c', P_j), \quad (7)$$

where  $P_u'$  is the predicted power of  $u$  in the Stackelberg game.

#### IV. OUR PROPOSED SGPL ALGORITHM

In metaverses collaborative computing networks, it is necessary to optimize the transmit power strategies of three different types of users. First, the source user needs to select the optimal transmission power, and then consider the impact of the source user's strategy on the intelligent jammer. In addition, the collaborative user needs to use the optimal strategy to complete the offloading to achieve the maximum benefit. Finally, after observing the transmitted signals of legitimate users, the intelligent jammer adopts the optimal jamming strategy, thus achieving Stackelberg equilibrium. Therefore, we need to prove that game strategies (5) - (7) have Stackelberg equilibrium points in the jamming game.

To obtain Stackelberg equilibrium, we first need to analyze the influence of source users on collaborative users and intelligent jammers.

To get the maximum benefit of the intelligent jammer, we need to solve the following problems:

$$\max_{P_j \geq 0} B_j(P_s, P_c, P_j) = \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} + C_c P_c + C_s P_s - C_j P_j. \quad (8)$$

**Lemma 1:** Let  $P_j^*$  be the optimal power strategy of intelligent jammer  $j$ , then

$$P_j^* = \begin{cases} 0, & G_s P_s + G_c P_c \leq \frac{C_j \sigma^2}{G_j}, \\ \frac{1}{G_j} \left[ \sqrt{\frac{G_j (G_s P_s + G_c P_c)}{C_j}} - \sigma \right], & \text{otherwise.} \end{cases} \quad (9)$$

*Proof 1:* First, according to benefit function  $B_j(P_s, P_c, P_j)$ , we have

$$\frac{\partial B_j(P_s, P_c, P_j)}{\partial P_j} = \frac{G_j (G_s P_s + G_c P_c)}{(N + G_j P_j)^2} - C_j, \quad (10)$$

$$\frac{\partial^2 B_j(P_s, P_c, P_j)}{\partial P_j^2} = \frac{-2G_j^2 (G_s P_s + G_c P_c)}{(\sigma + G_j P_j)^3}. \quad (11)$$

For Eq.(11),  $B_j(P_s, P_c, P_j)$  is a concave function w.r.t  $P_j$ . Let

$$\frac{G_j (G_s P_s + G_c P_c)}{(N + G_j P_j)^2} - C_j = 0, \quad (12)$$

we have

$$P_j' = \frac{1}{G_j} \left[ \sqrt{\frac{G_j (G_s P_s + G_c P_c)}{C_j}} - \sigma \right]. \quad (13)$$

If  $P_j' > 0$ , i.e.,  $G_s P_s + G_c P_c > \frac{C_j \sigma^2}{G_j}$ , then  $P_j^* = P_j'$ . If  $P_j' \leq 0$ , i.e.,  $G_s P_s + G_c P_c \leq \frac{C_j \sigma^2}{G_j}$ ,  $B_j$  decreases with  $P_j$ , yielding  $P_j^* = 0$ , then proof Eq. (9).

Similarly, for the collaborative user, according to Eq. (3), the optimization problem can be defined as:

$$\max_{P_c \geq 0} B_c(P_s, P_c) = \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} - C_c P_c, \quad (14)$$

*Lemma 2:* Let  $P_c^*$  be the optimal power strategy of collaborative user  $c$ , then

$$P_c^* = \begin{cases} 0, & W_1, \\ \frac{G_c C_j}{4G_j C_c^2} - \frac{G_s P_s}{G_c}, & W_2, \\ \frac{1}{G_c} \left( \frac{C_j \sigma^2}{G_j} - G_s P_s \right), & \text{otherwise.} \end{cases} \quad (15)$$

where

$$W_1 : P_s \geq \max \left( \frac{C_j \sigma^2}{G_s G_j}, \frac{G_c^2 C_j}{4G_s G_j C_c^2} \right), \text{ or } P_s < \frac{C_j \sigma^2}{G_s G_j} \text{ if } \frac{G_c}{\sigma} \leq C_c$$

$$W_2 : \frac{C_j \sigma^2}{G_s G_j} \leq P_s < \frac{G_c^2 C_j}{4G_s G_j C_c^2}, \text{ or } P_s < \frac{C_j \sigma^2}{G_s G_j} \text{ if } \frac{G_c}{\sigma} \geq 2C_c.$$

*Proof 2:* By adding  $P_j^*$  into Eq. (3), the benefit function of collaborative user is rewritten to

$$B_c(P_s, P_c) = \begin{cases} \left( \frac{G_c}{\sigma} - C_c \right) P_c + \frac{G_s}{\sigma} P_s, & P_c \leq Y_1, \\ \sqrt{\frac{C_j}{G_j} (G_s P_s + G_c P_c)} - C_c P_c, & P_c > Y_1, \end{cases} \quad (16)$$

where

$$Y_1 = \frac{1}{G_r} \left( \frac{C_j \sigma^2}{G_j} - G_s P_s \right).$$

Therefore,  $B_c$  is a linear function w.r.t.  $P_c \leq Y_1$ . When  $P_c > Y_1$ , we have

$$\frac{\partial B_c(P_s, P_c)}{\partial P_c} = \frac{G_c}{2} \sqrt{\frac{C_j}{G_j (G_s P_s + G_c P_c)}} - C_c, \quad (17)$$

$$\frac{\partial^2 B_c(P_s, P_c)}{\partial P_c^2} = \frac{-G_c^2}{4(G_s P_s + G_c P_c)} \sqrt{\frac{C_j}{G_j (G_s P_s + G_c P_c)}}. \quad (18)$$

According to Eq. (18),  $B_c$  is a concave function w.r.t  $P_c > Y_1$ , and maximized by

$$P_c' = \frac{G_c C_j}{4G_j C_c^2} - \frac{G_s P_s}{G_c}, \quad (19)$$

if  $P_c' \geq 0$ .

Considering two cases to find the optimal  $P_c$  to maximize  $B_c$ :

- 1)  $P_c \geq \frac{C_j \sigma^2}{G_s G_j}$  (i.e.,  $Y_1 \leq 0$ ):  $B_c$  is a concave function w.r.t.  $P_c > 0$ . When  $P_c' \leq 0$  (i.e.,  $P_s \geq \frac{G_c^2 C_j}{4G_s G_j C_c^2}$ ),  $B_c$  decreases w.r.t.  $P_c$ , and  $P_c^* = 0$ . Otherwise,  $B_c$  is maximized on  $P_c'$ , we have  $P_c^* = P_c'$ .
- 2)  $P_c < \frac{C_j \sigma^2}{G_s G_j}$  (i.e.,  $Y_1 > 0$ ):  $B_c$  is decreasing concave function w.r.t  $P_c > Y_1$ . When  $Y_1 \leq P_c'$  (i.e.,  $\frac{G_c}{\sigma} \geq 2\sigma$ ),  $B_c$  is increasing w.r.t.  $0 \leq P_c \leq Y_1$ . Therefore,  $B_c(P_s, P_c')$  is the maximum value of  $B_c$ , i.e.,  $P_c^* = P_c'$ . Note that when  $Y_1 > P_c'$ , if  $\frac{G_c}{\sigma} \geq C_c$ ,  $B_c$  increases w.r.t.  $P_c$  for  $0 \leq P_c \leq Y_1$  and  $P_c^* = Y_1$ . If  $\frac{G_c}{\sigma} < C_c$ ,  $B_c$  decreases w.r.t.  $P_c$ , therefore,  $P_c^* = 0$ . To sum up, proof Eq. (15).

For the source user, according to Eq. (2), the optimization problem can be defined as:

$$\max_{P_s \geq 0} B_s(P_s, P_c, P_j) = \frac{G_s P_s + G_c P_c}{\sigma + G_j P_j} - C_s P_s. \quad (20)$$

*Lemma 3:* Let  $P_s^*$  be the optimal power strategy of source user  $s$ , then

$$P_s^* = \begin{cases} \frac{G_s C_j}{4G_j C_c^2}, & W_3, \\ \frac{C_j \sigma^2}{G_s G_j}, & C_c \geq \frac{G_c}{\sigma}, \frac{G_s}{2\sigma} \leq C_s < \frac{G_s}{\sigma}, \\ 0, & \text{otherwise.} \end{cases} \quad (21)$$

where  $W_3 : C_c \leq \frac{G_c}{2\sigma}, \frac{C_s}{C_c} < \frac{G_s}{2G_c}$  or  $\frac{G_c}{2\sigma} < C_c < \frac{G_c}{\sigma}, C_s < \frac{G_s}{4\sigma}$  or  $C_c \geq \frac{G_c}{\sigma}, C_s < \frac{G_s}{2\sigma}$ .

*Proof 3:* We replace the variable  $P_c^*, P_j^*$  in Eq. (2), the benefit function of the source user device is rewritten as:

$$B_s(P_s) = \begin{cases} \sqrt{\frac{G_s P_s C_j}{G_j}} - C_s P_s, & P_s \geq \max(Z_1, Z_2), \\ -C_s P_s + \frac{C_j \sigma}{G_j}, & P_s < Z_1, C_c < \frac{G_c}{\sigma} < 2C_c, \\ \left( \frac{G_s}{\sigma} - C_s \right) P_s, & P_s < Z_1, \frac{G_c}{\sigma} \leq C_c, \\ \frac{G_c C_j}{2G_j C_c} - C_s P_s, & \text{otherwise,} \end{cases} \quad (22)$$

where  $Z_1 = \frac{C_j \sigma^2}{G_s G_j}$  and  $Z_2 = \frac{G_c^2 C_j}{4G_s G_j C_c^2}$ . Therefore,  $B_s$  is a linear function w.r.t.  $P_s < \max(Z_1, Z_2)$ . When  $P_s \geq \max(Z_1, Z_2)$ , then

$$\frac{\partial B_s(P_s)}{\partial P_s} = \frac{1}{2} \sqrt{\frac{G_s C_j}{G_j P_s}} - C_s, \quad (23)$$

$$\frac{\partial^2 B_s(P_s)}{\partial P_s^2} = -\frac{1}{4P_s} \sqrt{\frac{G_s C_j}{G_j P_s}}.$$

When  $P_s \geq \max(Z_1, Z_2)$ ,  $B_s$  is a concave function, which is maximized by  $\frac{G_c^2 C_j}{4G_j C_c^2}$ .

Considering three cases to find the optimal  $P_s$  to maximize  $B_s$ :

- 1)  $\frac{G_c}{\sigma} \geq 2C_c$  (i.e.,  $Z_1 \leq Z_2$ ):  $B_s$  is a decreasing linear function w.r.t  $P_s \in (0, Z_2)$ . If  $\frac{G_s}{C_s} \leq \frac{G_c}{C_c}$  ( $P_s' \leq Z_2$ ),  $B_s$

decreases with  $P_s$  and  $P_s > Z_2$ , then  $P_s^* = 0$ . If  $\frac{G_s}{C_s} > \frac{G_c}{C_c}$ ,  $B_s$  is maximized by  $P_s'$ . To find  $\max_{P_s' \geq 0} B_s$ , it need to compare  $B_s(0)$  with  $B_s(P_s')$ . If  $B_s(0) < B_s(P_s')$ ,  $P_s^* = P_s'$ , otherwise,  $P_s^* = 0$ .

2)  $C_c < \frac{G_c}{\sigma} < 2C_c$ :  $B_s$  decreases with  $P_s$ . If  $\frac{G_s}{C_s} \leq 2\sigma$  (i.e.,  $P_s' \leq Z_1$ ),  $B_s$  a decreasing concave w.r.t.  $P_s$ , thus  $P_s^* = 0$ . If  $\frac{G_s}{C_s} > 2\sigma$ ,  $B_s(P_s')$  is maximum value of  $B_s$ . When  $B_s(0) \geq B_s(P_s')$ , then  $P_s^* = 0$ , otherwise,  $P_s^* = P_s'$ .

3)  $\frac{G_c}{\sigma} < C_c$ : If  $\frac{G_s}{C_s} \leq 2\sigma$  (i.e.,  $P_s' \leq Z_1$ ),  $B_s$  is a decreasing concave, when  $\frac{G_s}{C_s} \leq \sigma$ ,  $B_s$  is decreasing w.r.t  $P_s$ , then  $P_s^* = 0$ , otherwise,  $P_s^* = Z_1$ . If  $\frac{G_s}{C_s} > 2\sigma$ ,  $B_s(P_s')$  is maximum value of  $B_s$  w.r.t  $P_s > Z_1$ , when  $\frac{G_s}{C_s} > \sigma$ ,  $B_s$  is increasing w.r.t.  $P_s \in (0, Z_1)$ , then  $P_s^* = P_s'$ .

Thus, the Stackelberg Equilibrium (SE) of the anti-jamming game in metaverse collaborative computing network is given by  $\{P_s^*, P_c^*, P_j^*\}$ . According to Eq. (22), as the leader, the source user will select the optimal transmission power according to the channel state of the jammer and the collaborative user. If the transmission cost of the legitimate user is too high (i.e.,  $C_c \geq \frac{G_c}{\sigma}$ ,  $C_s \geq \frac{G_s}{\sigma}$ ), or the channel status of the collaborative user is better, the source user equipment will terminate the transmission. Otherwise, the source user equipment will continuously adjust its optimal transmission strategy according to the channel status of the jammer and the collaborative user to maximize the benefits.

Next, we propose a Stackelberg Game-theoretical Policy-based Learning algorithm to solve the anti-jamming game model of section 3.

Let  $f_1(d_1, d_2)$  and  $f_2(d_1, d_2)$  be the objective functions that legitimate users and jammer want to maximize, respectively, where  $d_1 \in D_1 \subseteq \mathbb{R}^{d_1}$ ,  $d_2 \in D_2 \subseteq \mathbb{R}^{d_2}$  denote the decision variables (i.e., power  $P_u$  and channel gain  $G_u$ ), and  $D = (d_1, d_2) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$  is their joint strategy, and  $\mu$  is policy. The objective function of legitimate users and jammer is:

$$\min_{d_1} \left\{ -f_1(d_1, d_2) \mid d_2 \in \arg \min_{d_2' \in D_2} f_2(d_1, d_2') \right\}, \quad (24)$$

$$\min_{d_2} -f_2(d_1, d_2). \quad (25)$$

When the jammer chooses the best decision  $d_2^* \in \arg \min_{d_2' \in D_2} f_2(d_1, d_2')$ , the jammer's strategy is implicitly the function of legitimate users'. In deriving sufficient conditions for Eq. (24), users utilizes this information by:

$$\nabla f_1(d_1, d_2^*(d_1)) = \nabla_1 f_1(d) + (\nabla d_2^*(d_1))^T \nabla_2 f_1(d), \quad (26)$$

where  $\nabla d_2^*(d_1) = -(\nabla_2^2 f_2(d))^{-1} \nabla_2 f_2(d)$ .

Therefore,  $D = (d_1, d_2)$  is a local solution to Eq. (24), if  $\nabla f_1(d_1, d_2^*(d_1)) = 0$  and  $\nabla^2 f_1(d) > 0$ . The sufficient condition for the optimality of jammer is  $\nabla_2 f_2(d_1, d_2) = 0$  and  $\nabla_2^2 f_2(d) > 0$ .

The Stackelberg learning dynamics can be written as:

$$\begin{aligned} d_{1,k+1} &= d_{1,k} - \alpha_1 \nabla f_1(d_{1,k}, d_{2,k}), \\ d_{2,k+1} &= d_{2,k} - \alpha_2 \nabla_2 f_2(d_{1,k}, d_{2,k}), \end{aligned} \quad (27)$$

where  $\alpha_1, \alpha_2$  are the learning rate of legitimate users and jammer.

Let  $s_t$  denote the state (i.e., power and channel gain) and  $a_t$  denote the action (channel) at time  $t$ , respectively. We denote

reward at time  $t$  by  $r_t = r(a_t, s_t)$ , and cumulative rewards is  $R(v) = \sum_{t=0}^T \gamma^t r(a_t, s_t)$ , where  $\gamma \in (0, 1)$  is discount factor, and  $v = (s_0, a_0, \dots, s_T, a_T)$ . The  $Q$  function can be defined as:

$$Q^\mu(s_t, a_t) = \mathbb{E}_{v \sim \mu} \left[ \sum_{t'=t}^T \gamma^{t'-t} r(s_{t'}, a_{t'}) \mid s_t, a_t \right]. \quad (28)$$

The value function is defined as:

$$V^\mu(s_t) = \mathbb{E}_{v \sim \mu} \left[ \sum_{t'=t}^T \gamma^{t'-t} r(s_{t'}, a_{t'}) \mid s_t \right]. \quad (29)$$

Let  $\xi = s \sim \phi, a \sim \mu(\cdot \mid s)$ , the goal of our proposed algorithm is to find an optimal policy by:

$$\begin{aligned} J(\mu) &= \mathbb{E}_{v \sim \mu} \left[ \sum_{t=0}^T \gamma^t r(s_t, a_t) \right] \\ &= \int_v p(v \mid \mu) R(v) dv \\ &= \mathbb{E}_\xi [Q^\mu(s, a)], \end{aligned} \quad (30)$$

where  $p(v \mid \mu) = \phi(s_0) \prod_{t=0}^T \mu(a_t \mid s_t) T_p(s_{t+1} \mid s_t, a_t)$  and  $T_p$  is probability transfer function.

The parameter  $\vartheta$  can be updated by:

$$J(\vartheta) = \mathbb{E}_\xi [Q^\mu(s, a)]. \quad (31)$$

The optimization problem (31) is solved by:

$$\nabla_\vartheta J(\vartheta) = \mathbb{E}_\xi [\nabla_\vartheta \log \mu_\vartheta(a \mid s) Q^\mu(s, a)], \quad (32)$$

where  $\nabla_\vartheta$  is the derivative w.r.t.  $\vartheta$ .

Next, we use  $Q_c(s, a)$  (i.e., critic function) parameterized by  $c$  to approximate  $Q^\mu(s, a)$ . By written Eq (31) by  $Q_c(s, a)$ , the actor objective w.r.t.  $\vartheta$  can be defined as:

$$J(\vartheta, c) = \mathbb{E}_\xi [Q_c(s, a)]. \quad (33)$$

The optimization problem (33) is solved by:

$$\nabla_\vartheta J(\vartheta, c) = \mathbb{E}_\xi [\nabla_\vartheta \log \mu_\vartheta(a \mid s) Q_c(s, a)]. \quad (34)$$

The loss function can be defined as:

$$Loss(\vartheta, c) = \mathbb{E}_\xi [(Q_c(s, a) - Q^\mu(s, a))^2], \quad (35)$$

and the parameters are update by:

$$\begin{aligned} \vartheta &\leftarrow \vartheta + \alpha_\vartheta \nabla_\vartheta J(\vartheta, c), \\ c &\leftarrow c - \alpha_c \nabla_c Loss(\vartheta, c), \end{aligned} \quad (36)$$

where  $\alpha_c, \alpha_\vartheta$  are the learning rate of critic net and actor net, respectively.

For  $J(\vartheta, c)$  and  $Loss(\vartheta, c)$ , we need to solve two optimization problems:

$$\max_\vartheta \left\{ J(\vartheta, c^*(\vartheta)) \mid c^*(\vartheta) = \arg \min_c Loss(\vartheta, c) \right\}, \quad (37)$$

$$\min_c Loss(\vartheta, c). \quad (38)$$

The leader dynamics total derivative is given by:

$$\nabla_\vartheta J(\vartheta, c) - \nabla_{c\vartheta}^\top Loss(\vartheta, c) (\nabla_c^2 Loss(\vartheta, c))^{-1} \nabla_c J(\vartheta, c). \quad (39)$$

---

**Algorithm 1** The Stackelberg Game theoretical Policy-based Learning (SGPL) for anti-jamming

---

- 1: **Input:** channel gain  $G$ , Transmission power  $P$ , learning rate  $\alpha_c$ ,  $\text{learningrate}\alpha_\vartheta$ , Memory pool  $\mathcal{D}$ , probability  $p_\eta$ , discount factor  $\gamma$ .
  - 2: **Initialize:** the parameter of actor net  $\vartheta$ , the parameter of critic net  $c$ .
  - 3: **while** each episode **do**
  - 4:   Set the mixed policy:
 
$$\sigma \leftarrow \begin{cases} \epsilon\text{-greedy}(Q), & \text{with probability } p_\eta \\ \eta, & \text{with probability } 1 - p_\eta \end{cases}$$
  - 5:   The players (users and jammer) observe initial information state  $s_0$
  - 6:   **while** each time slot  $t$  **do**
  - 7:     The plays selected action  $a_t$  form  $\mathcal{D}$
  - 8:     Performing  $a_t$
  - 9:     Obtained reward  $r_t$ , transfer the next state  $s_{t+1}$
  - 10:    Store experience  $(s_t, a_t, r_t, s_{t+1})$  in  $\mathcal{D}$
  - 11:    **if**  $a_t$  is selected by  $\epsilon$ -greedy **then**
  - 12:      $\epsilon = \max(\epsilon_{\min}, \epsilon * \text{decay})$
  - 13:    **end if**
  - 14:    The loss function:
 
$$\nabla_\vartheta J(\vartheta, c) = \mathbb{E}_{s \sim \phi, a \sim \mu(\cdot|s)} [\nabla_\vartheta \log \mu_\vartheta(a|s) Q_c(s, a)].$$

$$\text{Loss}(\vartheta, c) = \mathbb{E}_{s \sim \phi, a \sim \mu(\cdot|s)} [(Q_c(s, a) - Q^\mu(s, a))^2]$$
  - 15:   **end while**
  - 16:   Use **Algorithm 2** to calculate gradient.
  - 17:   Use **Algorithm 3** to update total derivative parameter.
  - 18: **end while**
  - 19: **Output:** the optimal policy  $a \sim \mu(\cdot|s)$ .
- 

The follower dynamics total derivative is given by:

$$\nabla_c \text{Loss}(\vartheta, c) - \nabla_{\vartheta c}^\top J(\vartheta, c) (\nabla_\vartheta^2 J(\vartheta, c))^{-1} \nabla_\vartheta \text{Loss}(\vartheta, c). \quad (40)$$

To obtain a more accurate approximation, the leaders need to choose a best  $c^*(\vartheta) = \arg \max_{c'} \text{Loss}(\vartheta, c')$ . Furthermore, we have

$$\nabla_c J(\vartheta, c) = \mathbb{E}_\xi [\nabla_c Q_c(s, a)], \quad (41)$$

$$\begin{aligned} \nabla_c^2 \text{Loss}(\vartheta, c) = & \mathbb{E}_\xi [2 \nabla_c Q_c(s, a) \nabla_c^\top Q_c(s, a) \\ & + 2 (Q_c(s, a) - Q^\mu(s, a)) \nabla_c^2 Q_c(s, a)]. \end{aligned} \quad (42)$$

To compute  $\nabla_{\vartheta c} \text{Loss}(\vartheta, c)$  in Eq. (39), we obtain  $\nabla_\vartheta \text{Loss}(\vartheta, c)$  by Theorem 1.

*Theorem 1:* Given a Markov Decision Process and a policy-based parameters  $(\vartheta, c)$ , the gradient of  $\text{Loss}(\vartheta, c)$  w.r.t.  $\vartheta$  is given by:

$$\begin{aligned} \nabla_\vartheta \text{Loss}(\vartheta, c) = & \mathbb{E}_{v \sim \mu_\vartheta} [\nabla_\vartheta \log \mu_\vartheta(a_0 | s_0) \\ & (Q_c(s_0, a_0) - Q^\mu(s_0, a_0))^2 + \sum_{t=1}^T \gamma^t \nabla_\vartheta \log \mu_\vartheta(a_t | s_t) \\ & (Q^\mu(s_0, a_0) - Q_c(s_0, a_0)) Q^\mu(s_t, a_t)]. \end{aligned} \quad (43)$$

*Proof 4:* First, the critic's objective is:

$$\text{Loss}(\vartheta, c) = \mathbb{E}_\xi [(Q_c(s, a) - Q^\mu(s, a))^2]. \quad (44)$$

---

**Algorithm 2** Stochastic Gradient Descent with mini-batch

---

- 1: **Input:** learning rate  $\alpha_c, \alpha_\vartheta$ , Memory D.
- 2: **Initialize:** the parameter of actor net  $\vartheta$ , the parameter of critic net  $c$ , parameter  $\beta, \mu$ .
- 3: **while** each episode  $t$  **do**
- 4:   Select m mini-batch sample from memory D
- 5:   Calculate gradient for loss function  $\nabla_\vartheta J(\vartheta, c)$  and  $\text{Loss}(\vartheta, c)$ :

$$\nabla_\vartheta J(\vartheta, c) \leftarrow \frac{1}{m} \nabla_w \sum_i^m J(\vartheta, c)$$

$$\text{Loss}(\vartheta, c) \leftarrow \frac{1}{m} \nabla_w \sum_i^m L(\vartheta, c)$$

- 6:   Calculate square gradient:

$$\mathbb{E}[J^2]_t = \beta E[J^2]_{t-1} + (1 - \beta) J_t^2$$

$$\mathbb{E}[\text{Loss}^2]_t = \beta E[\text{Loss}^2]_{t-1} + (1 - \beta) \text{Loss}_t^2$$

- 7:   Update parameters  $\vartheta$  and  $c$ :

$$\vartheta_{t+1} = \vartheta_t - \frac{\alpha_\vartheta}{\sqrt{E[J^2]_t + \mu}} * J_t.$$

$$c_{t+1} = c_t - \frac{\alpha_c}{\sqrt{E[\text{Loss}^2]_t + \mu}} * \text{loss}_t.$$

- 8: **Output:** the optimal parameter  $\vartheta$  and  $c$ .
- 

The derivative of  $\text{Loss}(\vartheta, c)$  is:

$$\begin{aligned} & \nabla_\vartheta \text{Loss}(\vartheta, c) \\ = & \nabla_\vartheta \int_{s_0} \phi(s_0) \int_{a_0} \mu_\vartheta(a_0 | s_0) (Q_c(s_0, a_0) - Q^\mu(s_0, a_0))^2 da_0 ds_0 \\ = & \int_{s_0} \phi(s_0) \int_{a_0} \nabla_\vartheta \mu_\vartheta(a_0 | s_0) (Q_c(s_0, a_0) - Q^\mu(s_0, a_0))^2 da_0 ds_0 \\ & + \int_{s_0} \phi(s_0) \int_{a_0} \mu_\vartheta(a_0 | s_0) \nabla_\vartheta (Q_c(s_0, a_0) - Q^\mu(s_0, a_0))^2 da_0 ds_0 \\ = & \int_{s_0} \phi(s_0) \int_{a_0} \mu_\vartheta(a_0 | s_0) \nabla_\vartheta \log \mu_\vartheta(a_0 | s_0) (Q_c(s_0, a_0) - Q^\mu(s_0, a_0))^2 da_0 ds_0 \\ & + 2 \int_{s_0} \phi(s_0) \int_{a_0} \mu_\vartheta(a_0 | s_0) (Q^\mu(s_0, a_0) - Q_c(s_0, a_0)) \nabla_\vartheta Q^\mu(s_0, a_0) da_0 ds_0. \end{aligned} \quad (45)$$

According to Eq. (45), we need to calculate function  $Q^\mu(s_0, a_0)$ . According to Eq. (28-29), we have

$$Q^\mu(s_t, a_t) = \mathbb{E}_{v \sim \mu} \left[ \sum_{t'=t}^T \gamma^{t'-t} r(s_{t'}, a_{t'}) \mid s_t, a_t \right], \quad (46)$$

and

$$V^\mu(s_t) = \mathbb{E}_{v \sim \mu} \left[ \sum_{t'=t}^T \gamma^{t'-t} r(s_{t'}, a_{t'}) \mid s_t \right]. \quad (47)$$

$$\begin{aligned}
\nabla_{\vartheta} Q^{\mu}(s_0, a_0) &= \gamma \int_{s_1} T_p(s_1 | s_0, a_0) \nabla_{\vartheta} V^{\mu}(s_1) ds_1 \\
&= \gamma \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} (\nabla_{\vartheta} \mu_{\vartheta}(a_1 | s_1) Q^{\mu}(s_1, a_1) + \mu_{\vartheta}(a_1 | s_1) \nabla_{\vartheta} Q^{\mu}(s_1, a_1)) da_1 ds_1 \\
&= \gamma \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} \mu_{\vartheta}(a_1 | s_1) \nabla_{\vartheta} \log \mu_{\vartheta}(a_1 | s_1) Q^{\mu}(s_1, a_1) da_1 ds_1 \\
&\quad + \gamma^2 \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} \mu_{\vartheta}(a_1 | s_1) \int_{s_2} T_p(s_2 | s_1, a_1) \nabla_{\vartheta} V^{\mu}(s_2) ds_2 da_1 ds_1 \\
&= \gamma \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} \mu_{\vartheta}(a_1 | s_1) \nabla_{\vartheta} \log \mu_{\vartheta}(a_1 | s_1) Q^{\mu}(s_1, a_1) da_1 ds_1 \\
&\quad + \gamma^2 \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} \mu_{\vartheta}(a_1 | s_1) \int_{s_2} T_p(s_2 | s_1, a_1) \int_{a_2} \mu_{\vartheta}(a_2 | s_2) \nabla_{\vartheta} \log \mu_{\vartheta}(a_2 | s_2) Q^{\mu}(s_2, a_2) da_2 ds_2 da_1 ds_1 \\
&\quad + \gamma^3 \int_{s_1} T_p(s_1 | s_0, a_0) \int_{a_1} \mu_{\vartheta}(a_1 | s_1) \int_{s_2} T_p(s_2 | s_1, a_1) \int_{a_2} \mu_{\vartheta}(a_2 | s_2) \int_{s_3} T_p(s_3 | s_2, a_2) \nabla_{\vartheta} V^{\mu}(s_3) ds_3 da_2 ds_2 da_1 ds_1 \\
&= \gamma \int_v T_p(v_{1:1} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_1 | s_1) Q^{\mu}(s_1, a_1) dv_{1:1} \\
&\quad + \gamma^2 \int_v T_p(v_{1:2} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_2 | s_2) Q^{\mu}(s_2, a_2) dv_{1:2} \\
&\quad + \dots \\
&= \int_v \sum_{t=1}^T \gamma^t p(v_{1:T} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) Q^{\mu}(s_t, a_t) d\nu,
\end{aligned} \tag{48}$$

**Algorithm 3** The proposed total derivative parameter update algorithm

- 1: **Input:** learning rate  $\alpha_c, \alpha_{\vartheta}$ , follower unrolling steps  $m$ , hyperparameter  $\lambda$ ,
- 2: **while**  $k = 0, 1, 2, 3, \dots$  **do**
- 3:   **if** if actor is leader **then**
- 4:     update critic net and actor net in SGPL:

$$\begin{aligned}
\vartheta_{k+1} &= \vartheta_k + \alpha_{\vartheta, k} (\nabla_{\vartheta} J(\vartheta_k, c_{k,0}) \\
&\quad - (\nabla_{\vartheta}^{\top} Loss \circ (\nabla_c^2 Loss + \lambda I)^{-1} \circ \nabla_c J)(\vartheta_k, c_{k,0}))
\end{aligned}$$

$$c_{k,l+1} = c_{k,l} - \alpha_{c,k} \nabla_c Loss(\vartheta_k, c_{k,l}), \quad l \in [0, m-1]$$

$$c_{k+1,0} = c_{k,m}$$

- 5:   **if** if critic is leader **then**
- 6:     update critic net and actor net in SGPL:

$$\begin{aligned}
c_{k+1} &= c_k - \dot{\alpha}_{\vartheta, k} (\nabla_c Loss(\vartheta_k, c_k) \\
&\quad - (\nabla_{\vartheta}^{\top} J \circ (\nabla_{\vartheta}^2 J + \lambda I)^{-1} \circ \nabla_{\vartheta} Loss)(\vartheta_k, c_k))
\end{aligned}$$

$$\vartheta_{k,l+1} = \vartheta_{k,l} + \alpha_{c,k} \nabla_{\vartheta} J(\vartheta_{k,l}, c_k), \quad l \in [0, m-1]$$

$$\vartheta_{k+1,0} = \vartheta_{k,m}$$

7: end while

Therefore, we have

$$\begin{aligned}
\nabla_{\vartheta} Loss(\vartheta, c) &= \int_{s_0} \phi(s_0) \int_{a_0} \mu_{\vartheta}(a_0 | s_0) \nabla_{\vartheta} \log \mu_{\vartheta}(a_0 | s_0) (Q_c(s_0, a_0) \\
&\quad - Q^{\mu}(s_0, a_0))^2 da_0 ds_0 \\
&\quad + 2 \int_{s_0} \phi(s_0) \int_{a_0} \mu_{\vartheta}(a_0 | s_0) (Q^{\mu}(s_0, a_0) \\
&\quad - Q_c(s_0, a_0)) \nabla_{\vartheta} Q^{\mu}(s_0, a_0) da_0 ds_0 \\
&= \int_v T_p(v_0 | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_0 | s_0) (Q_c(s_0, a_0) - Q^{\mu}(s_0, a_0))^2 \\
&\quad + 2 \sum_{t=1}^T \gamma^t T_p(v_{0:t} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) (Q^{\mu}(s_0, a_0) \\
&\quad - Q_c(s_0, a_0)) Q^{\mu}(s_t, a_t) dv \\
&= \mathbb{E}_{v \sim \mu_{\vartheta}} \left[ \nabla_{\vartheta} \log \mu_{\vartheta}(a_0 | s_0) (Q_c(s_0, a_0) - Q^{\mu}(s_0, a_0))^2 \right. \\
&\quad \left. + \sum_{t=1}^T \gamma^t \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) (Q^{\mu}(s_0, a_0) - Q_c(s_0, a_0)) Q^{\mu}(s_t, a_t) \right].
\end{aligned} \tag{49}$$

Therefore, the Theorem 1 is proved.

According to Theorem 1, we can obtain  $\nabla_{\vartheta} Loss(\vartheta, c)$  by  $\nabla_c(\nabla_{\vartheta} Loss(\vartheta, c))$ . The critic net is used to approximate  $V^{\mu}(s)$ , then  $J(\vartheta, c) = \mathbb{E}_{v \sim \mu_{\vartheta}} [r(s_0, a_0) + V_c(s_1)]$  and  $Loss(\vartheta, c) = \mathbb{E}_{s \sim \phi} [(V_c(s) - V^{\mu}(s))^2]$ . Therefore,  $\nabla_{\vartheta} Loss(\vartheta, c)$  can be obtained by Theorem 2.

**Theorem 2:** Given a Markov Decision Process and a policy-based parameters  $(\vartheta, c)$ , if  $Loss(\vartheta, c) = \mathbb{E}_{s \sim \phi} [(V_c(s) - V^{\mu}(s))^2]$ , then  $\nabla_{\vartheta} Loss(\vartheta, c)$  is given by:

$$\mathbb{E}_{v \sim \mu_{\vartheta}} \left[ 2 \sum_{t=0}^T \gamma^t \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) (V^{\mu}(s_0) - V_c(s_0)) Q^{\mu}(s_t, a_t) \right]. \tag{50}$$

Therefore, we can obtained  $Q^{\mu}(s_0, a_0)$  by Eq. 48, where the last item of Eq. (48) is obtained by marginalizing and unrolling the entire Markov decision-making process  $\nu$ .

*Proof 5:* For  $Loss(\vartheta, c) = \mathbb{E}_{s \sim \phi} [(V_c(s) - V^\mu(s))^2]$ , we have

$$\begin{aligned} \nabla_{\vartheta} Loss(\vartheta, c) &= \int_{s_0} \phi(s_0) \nabla_{\vartheta} (V_c(s_0) - V^\mu(s_0))^2 ds_0 \\ &= 2 \int_{s_0} \phi(s_0) (V^\mu(s_0) - V_c(s_0)) \nabla_{\vartheta} V^\mu(s_0) ds_0. \end{aligned} \quad (51)$$

Based on Eq. (51), we can obtain  $\nabla_{\vartheta} V^\mu(s_0)$ , we have

$$\begin{aligned} \nabla_{\vartheta} V^\mu(s_0) &= \int_{a_0} \nabla_{\vartheta} \mu_{\vartheta}(a_0 | s_0) Q^\mu(s_0, a_0) \\ &+ \mu_{\vartheta}(a_0 | s_0) \nabla_{\vartheta} Q^\mu(s_0, a_0) da_0 \\ &= \int_v \mu_{\vartheta}(a_0 | s_0) (\nabla_{\vartheta} \log \mu_{\vartheta}(a_0 | s_0) Q^\mu(s_0, a_0) \\ &+ \sum_{t=1}^T \gamma^t T_p(v_{1:t} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) Q^\mu(s_t, a_t)) dv. \end{aligned} \quad (52)$$

Substituting Eq. (52) into Eq. (51), then

$$\begin{aligned} \nabla_{\vartheta} Loss(\vartheta, c) &= 2 \int_v \sum_{t=0}^T \gamma^t T_p(v_{0:t} | \vartheta) \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) (V^\mu(s_0) \\ &- V_c(s_0)) Q^\mu(s_t, a_t) dv \\ &= \mathbb{E}_{v \sim \mu_{\vartheta}} \left[ 2 \sum_{t=0}^T \gamma^t \nabla_{\vartheta} \log \mu_{\vartheta}(a_t | s_t) (V^\mu(s_0) - V_c(s_0)) Q^\mu(s_t, a_t) \right] \end{aligned} \quad (53)$$

which proof the Theorem 2.

Next, we analyze the convergence of the proposed algorithm. The legitimate users are the leaders and jammer is the follower. Therefore, the legitimate users and jammer updates with  $\alpha_{\kappa, \vartheta}$ ,  $\alpha_{c, \vartheta}$  and Stackelberg gradient dynamics:

$$\begin{aligned} \vartheta_{k+1} &= \vartheta_k + \alpha_{\vartheta, k} (\nabla J(\vartheta, c) + \nu_{\vartheta, k+1}), \\ c_{k+1} &= c_k - \alpha_{c, k} (\nabla_c Loss(\vartheta, c) + \nu_{c, k+1}), \end{aligned} \quad (54)$$

where  $\nu_{\vartheta, k+1}$ ,  $\nu_{c, k+1}$  are stochastic processes. The SGPL algorithm is shown in Algorithm 1.

*Proposition 1:*  $\nabla J : \mathbb{R}^k \rightarrow \mathbb{R}^{k_{\vartheta}}$ ,  $\nabla_c Loss : \mathbb{R}^k \rightarrow \mathbb{R}^{k_c}$  are Lipschitz, and  $\|\nabla J\| < \infty$ , such that  $\alpha_{\vartheta, c} = a(\alpha_c, k)$ ,  $\sum_k \alpha_{i, k} = \infty$  and  $\sum_k \alpha_{i, k}^2 < \infty$  for  $i \in I = \{c, \vartheta\}$ . The stochastic processes  $\nu_{i, k}$  are zero mean sequences, and function  $F_k = \varrho(\vartheta_s, c_s, \nu_{s, \vartheta}, \nu_{s, c})$ ,  $s < k$  are conditionally independent, almost surely  $\mathbb{E}[\|\nu_{i, k+1}\| | F_k] = 0$ , and  $\mathbb{E}[\|\nu_{i, k+1}\| | F_k] \leq m_i(1 + \|(\vartheta, c_k)\|)$  for  $m_i > 0$ .

According to Proposition 1, we obtain a convergence guarantee by Theorem 3.

*Theorem 3:* Given a Markov Decision Process and a policy-based parameters  $(\vartheta, c)$ , a Stackelberg game system  $(\vartheta, c) = (\nabla J(\vartheta, c), -\nabla_c Loss(\vartheta, c))$  have a Stackelberg equilibrium  $(\vartheta^*, c^*)$ , under Proposition 1, it exists a neighborhood  $M$  converge to  $(\vartheta^*, c^*)$  for  $(\vartheta_0, c_0) \in M$ .

*Proof 6:* For a differential Stackelberg equilibrium  $(\vartheta^*, c^*)$ , we have

$$\begin{bmatrix} \dot{\vartheta} \\ \dot{c} \end{bmatrix} = \begin{bmatrix} \nabla J(\vartheta, c) \\ -\nabla_c Loss(\vartheta, c) \end{bmatrix}. \quad (55)$$

The total derivative is given by

$$\begin{aligned} \nabla J(\vartheta, c) &= \nabla_{\vartheta} J(\vartheta, c) \\ &- \nabla_{c\vartheta}^\top Loss(\vartheta, c) (\nabla_c^2 Loss(\vartheta, c))^{-1} \nabla_c J(\vartheta, c). \end{aligned} \quad (56)$$

The individual gradient of follower is  $\nabla_c Loss(\vartheta, c)$ . Since leaders and followers have made unbiased estimates of their gradients and selected the learning rate according to Eq. (54), the Stackelberg gradient dynamics converges asymptotically to  $(\vartheta^*, c^*)$ . Note that each term in the total derivative is calculated as the expected value of the distribution of actions and states. For instance, given

$$J(\vartheta, c) = \mathbb{E}_{\xi \sim \mathcal{D}} [Q_c(s, \mu_{\vartheta}(s))], \quad (57)$$

$$Loss(\vartheta, c) = \mathbb{E}_{\xi \sim \mathcal{D}} [(Q_c(s, a) - (r + \gamma Q_0(s', \mu_{\vartheta}(s'))))^2], \quad (58)$$

we have Eq. (59) hold:

$$\begin{aligned} \nabla J(\vartheta, c) &= \nabla_{\vartheta} J(\vartheta, c) - \nabla_{c\vartheta}^\top Loss(\vartheta, c) (\nabla_c^2 Loss(\vartheta, c))^{-1} \nabla_c J(\vartheta, c) \\ &= \mathbb{E}_{\xi \sim \mathcal{D}} [\nabla_{\vartheta} Q_c(s, \mu_{\vartheta}(s))] \\ &- \mathbb{E}_{\xi \sim \mathcal{D}} \left[ \nabla_{c\vartheta} \left( (Q_c(s, a) - (r + \gamma Q_0(s', \mu_{\vartheta}(s'))))^2 \right)^\top \right. \\ &\quad \left. \left( \nabla_c^2 \left( (Q_c(s, a) - (r + \gamma Q_0(s', \mu_{\vartheta}(s'))))^2 \right) \right)^{-1} \nabla_c Q_c(s, \mu_{\vartheta}(s)) \right] \\ &\approx \mathbb{E}_{\xi \sim \mathcal{D}} [\nabla_{\vartheta} Q_c(s, \mu_{\vartheta}(s))] \\ &- \mathbb{E}_{\xi \sim \mathcal{D}} \left[ \nabla_{c\vartheta} \left( (Q_c(s, a) - (r + \gamma Q_0(s', \mu_{\vartheta}(s'))))^2 \right)^\top \right. \\ &\quad \left. \left( \mathbb{E}_{\xi \sim \mathcal{D}} \left[ \nabla_c^2 \left( (Q_c(s, a) - (r + \gamma Q_0(s', \mu_{\vartheta}(s'))))^2 \right) \right] \right)^{-1} \right. \\ &\quad \left. \mathbb{E}_{\xi \sim \mathcal{D}} [\nabla_c Q_c(s, \mu_{\vartheta}(s))] \right]. \end{aligned} \quad (59)$$

Therefore, the iterates  $(\vartheta_k, c_k)$  converge to  $(\vartheta^*, c^*)$ .

## V. PERFORMANCE ANALYSIS

TABLE I: Hyperparameters of SGPL algorithm

Parameters	Value
Learning rate	0.001
Explore start	0.8
Explore stop	0.01
Explore decay rate	0.0001
Discount factor	0.995
Memory size	3000
Episodes	15000
Batch size	128
Entropy regularization coefficient	0.1
Hidden layer 1 units	256
Hidden layer 2 units	128
Hidden layer 3 units	64

In this section, we performed several experiments to evaluate the performance of the proposed SGPL algorithm for anti-jamming in dynamic metaverses collaborative computing networks. The edge servers are 5, which are represented by the long-term evolution advanced evolved NodeBs. The macro cell NodeBs and the network service area are located at the same location. We assume that the jammer randomly appears in any network area, and its parameters are as follows: the path loss exponent is 5, jamming range radius is 1000m, the learning step is 0.07, available channels is 8, and the number

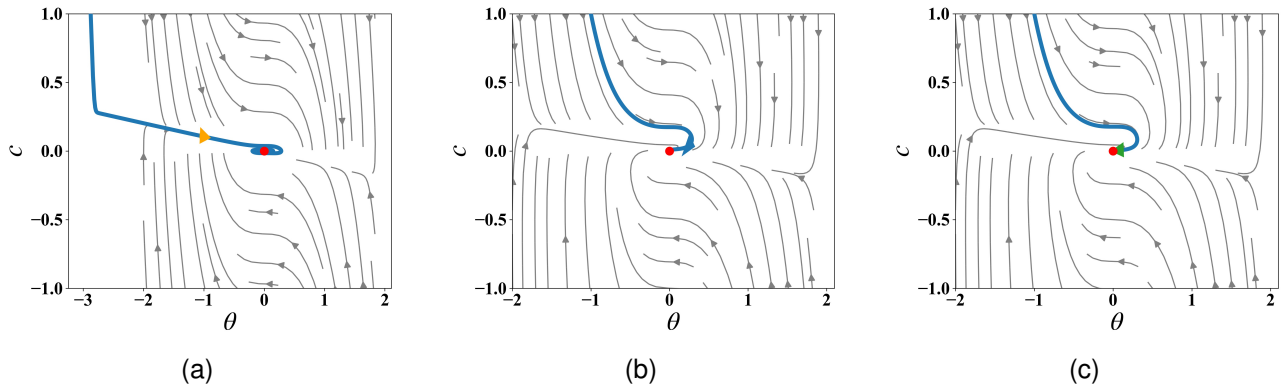


Fig. 2: Vector fields and trajectories of three gradient updates: (a) Individual gradient; (b) Stackelberg gradient; (c) Regularized Stackelberg gradient.

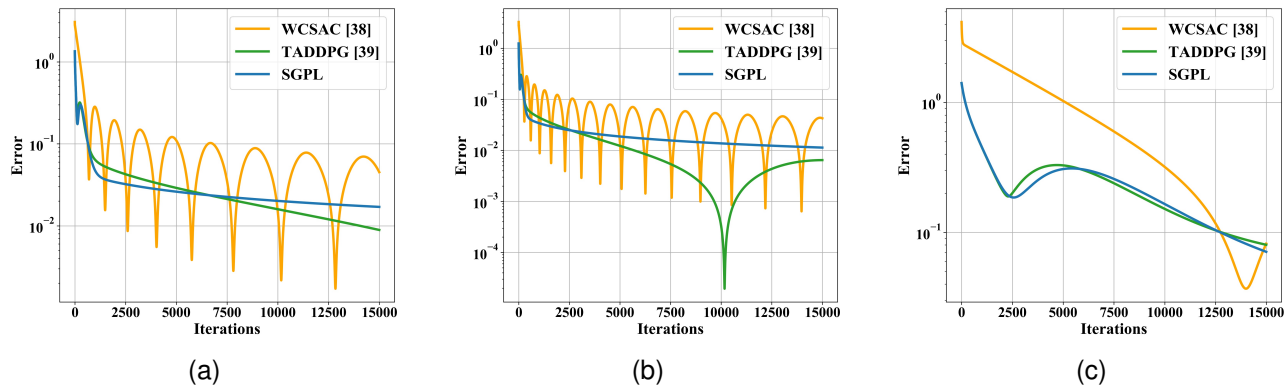


Fig. 3: Comparison of convergence error performance of three algorithms SGPL, TADDPG, WCSAC: (a) Critic learning rate  $\alpha = 0.01$ ; (b) Critic learning rate  $\alpha = 0.05$ ; (c) Critic learning rate  $\alpha = 0.0001$ .

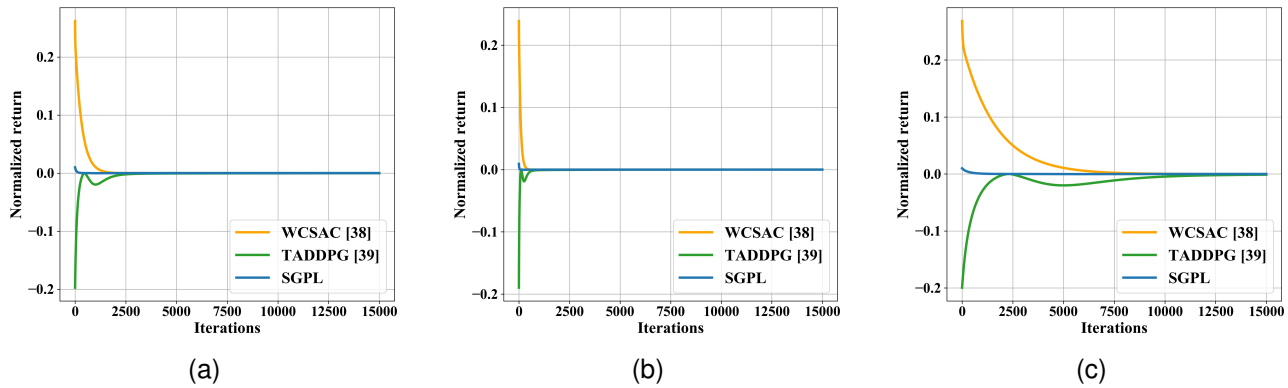


Fig. 4: Comparison of normalized return of the jammer of three algorithms SGPL, TADDPG, WCSAC: (a) Critic learning rate  $\alpha = 0.01$ ; (b) Critic learning rate  $\alpha = 0.05$ ; (c) Critic learning rate  $\alpha = 0.0001$ .

of metaverse users is 100. The hyperparameters are given in Table I.

First, we need to verify the convergence of Stackelberg gradient. According to Eq. (34), we verify the hidden structure of loss policy gradient. The algorithm of optimization goal is to reduce the cost of the game strategy of legitimate users and jammers by stackelberg gradient. Fig. 2(a) shows the

individual gradient dynamics vector field and convergence trajectory of players. Fig. 2(b) shows the Stackelberg gradient dynamics vector field and convergence trajectory of players. Fig. 2(c) shows the implicit map regularization stackelberg gradient dynamics vector field and convergence trajectory of players. It can be seen that there is an obvious cycle track in Fig. 2(a), which leads to the failure to ensure the reliability in



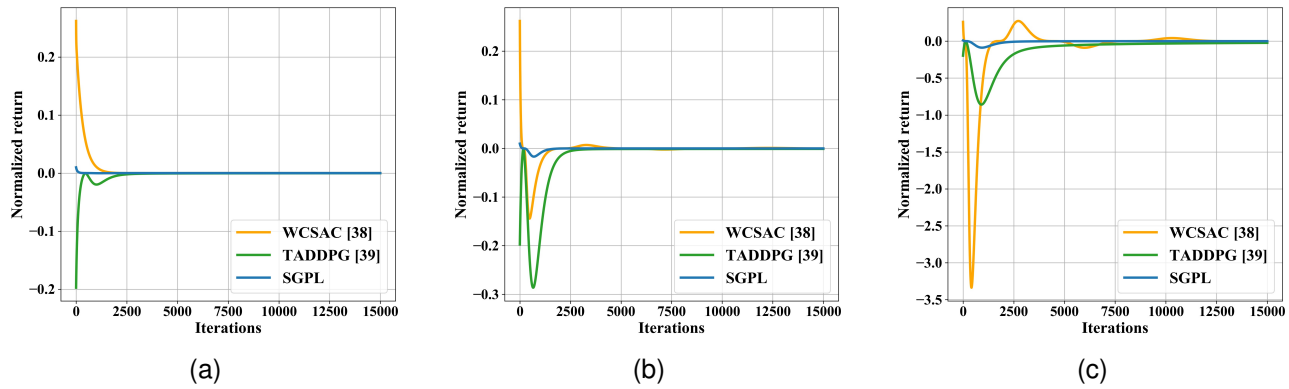


Fig. 5: Comparison of normalized return of the jammer of three algorithms SGPL, TADDPG, WCSAC: (a) Metaverse users = 10; (b) Metaverse users = 50; (c) Metaverse users = 100.

the training process. In the worst case, it is easy to be interfered by external noise information, which will further reduce the performance. In practical applications, compared with cyclic behavior, smooth and monotonous performance changes can improve the learning performance of the whole system. This is because the execution process of cyclic oscillation will have higher costs. On the other hand, as shown in Fig. 2(b)-2(c), our Stackelberg gradient dynamics converges to the equilibrium point  $(\vartheta^*, c^*) = (0, 0)$  directly and smoothly. This is because the Stackelberg gradient is used to optimize the jamming game, which alleviates the circular behavior and accelerates the convergence speed.

To further verify the convergence rate, we consider two policy-based SOTA algorithms Worst-Case Soft Actor Critic (WCSAC) [38] and Triplet-Average Deep Deterministic Policy Gradient (TADDPG) [39] as the comparison. We use convergence error  $\|c - c^*\| + \|\vartheta - \vartheta^*\|$  and normalized return of jammer as evaluation metric, which are show in Fig. 3 and Fig. 4. Among the three policy based algorithms, WCSAC has been unable to converge. Therefore, the jamming game in metaverses collaborative computing networks cannot be applied. TADDPG has the trend of convergence, but its convergence performance is not as good as our SGPL algorithm. According to Fig. 3 and Fig. 4, we see that the convergence speed of WCSAC and TADDPG is greatly affected by the critic learning rate, and too large or too small learning rate will lead to the inability of algorithms to converge. For the proposed SGPL algorithm, the more suitable critic learning rate is 0.01.

Fixing the learning rate to 0.01, in order to evaluate the algorithm's scalability and robustness, we conduct experiments using three different groups of metaverse users (i.e., metaverse users = 10, 50, 100). Fig. 5 shows that the algorithm converges fastest when the users is 10 and converges in 2500 iterations, when the users is 50, it takes 5000 iterations to converge, and when the users is 100, it takes 12,500 iterations to converge. This is because the proposed algorithm uses a multi-agent deep reinforcement learning game technique and thus requires high computational power as support during the training phase, and the required computational power increases with the size of the system users. Fortunately, AR/VR devices with average

computational power can also run easily when the model is trained and enters the application phase. In addition, the arithmetic performance of GPU servers in industrial applications is much higher than that of our existing servers. Therefore, the algorithm has good scalability and robustness, and it is feasible to support large-scale metadata scenarios in practical applications.

In Fig. 6 and Fig. 7, we add the entropy regularization of metaverse user and jammer's objection function. According to Fig. 6, metaverse users and jammers add entropy regularization to improve the convergence speed of Vector fields and trajectories. Fig. 7 shows that after the addition of entropy regulation, the error gap in the game results of metaverse users and jammers has gradually narrowed, and the two are in a state of equilibrium. According to Fig. 7(a)-Fig. 7(c), the entropy regulation coefficient can directly affect the training effect of entropy regulation. Too large or too small entropy regulation coefficient will cause large errors. In this experiment, the error is the smallest when the entropy regulation coefficient is equal to 0.1.

To demonstrate the performance of our proposed algorithm in the jamming game in metaverses collaborative computing networks, we use three game-based SOTA algorithms for comparison: Nash game (NG) [40], Zero-sum game (ZSG) [41], Stackelberg game (SG) [42]. As mentioned earlier, the game in jamming environment needs to control the cost. Therefore, the players need to reach the game equilibrium under the control of the lowest cost.

Next, we constrain the maximum power of the jammer, and test the benefits of different users under different maximum jamming powers in the collaborative computing game. We used 6 metaverse users (i.e., 3 source users and 3 collaborative users), 2 base stations, and 2 jammers in the anti-jamming game environment. According to Fig. 8, there are three different type results: 1) In Fig. 8(a) and Fig. 8(d), source user 1 has successfully avoided jamming, so the benefit has been increasing. At this time, the cooperation of collaborative users is rarely required, so the benefit of collaborative users is approximately unchanged. 2) In Fig. 8(b) and Fig. 8(e), the source user has suffered some interference, but the interference



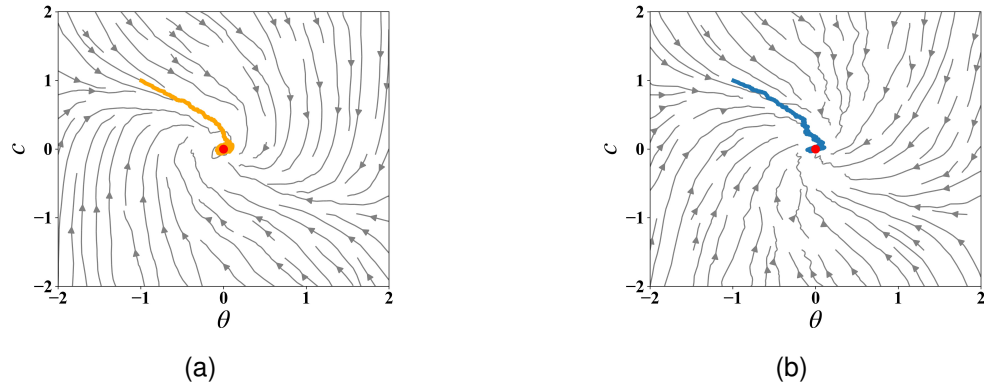


Fig. 6: Vector fields and trajectories of metaverse user and jammer: (a) Entropy regularization users gradient; (b) Entropy regularization jammer gradient;

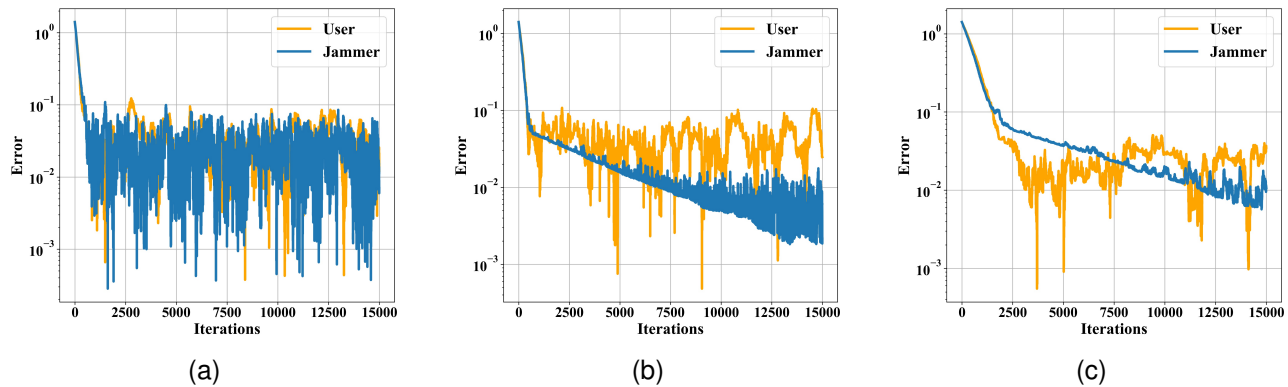


Fig. 7: Comparison of entropy regularization error of metaverse user and jammer: (a) Entropy regularization coefficient is 0.1; (b) Entropy regularization coefficient is 0.01; (c) Entropy regularization coefficient is 0.9.

can be offset by the transmission power. At this time, the source user chooses to forward some of the offloaded tasks to the collaborative users to reduce their energy consumption burden, so the revenue of the collaborative users rises slowly. 3) Fig. 8(c) and Fig. 8(f), the source user has been greatly interfered. Thus, with the increase of the maximum interference power, the revenue of the source user has been declining. At this point, in order to enable the task to continue to execute, the source user selects the collaborative user for data transmission, so the revenue of the collaborative user will increase rapidly.

In metaverse jamming defense systems, power consumption is an extremely important performance metric. The jammer interferes with the communication environment of legitimate user equipment by emitting high-power signals, blocking the completion of cooperative computing. The legitimate user devices need to offset the interfering signals by transmitting anti-interference power, and when the power consumption is too high, they can only abandon the channel and avoid the interference by frequency hopping. Fig. 9 shows the power consumption of different users in collaborative computing. Obviously, after training, the jammer needs extremely high power consumption to achieve jamming, which is uncontrollable cost for the jammer. In contrast, the source user device and the co-computing user can train the optimal anti-interference

strategy to achieve anti-interference by frequency hopping, so the power consumption is all within a manageable range.

We use the average channel utilization as the measures of the anti-jamming performance, which are shown in Fig. 10. If the performance of the user's anti-jammer game is improved, the average channel utilization will be high, and on the contrary, the average channel utilization will be low. As shown in Fig. 10, compared with other deep reinforcement learning algorithms, SGPL shows superior performance in the anti-jamming game. Users get the maximum transmission rate, indicating that users suffer the least interference. Note that WCSAC algorithm gets better throughput (i.e., transmission rate) than TADDPG algorithm, but its final result is always volatile, which indicates that the algorithm is not stable in the training process. Therefore, TADDPG algorithm may have worse performance due to this instability.

Next, we use more SOTA algorithms, namely Worst-Case Soft Actor Critic (WCSAC) [38], Soft Actor Critic (SAC) [43], Trust region policy optimization (TRPO) [44], Triplet-Average Deep Deterministic Policy Gradient (TADDPG) [39], to evaluate the system performance. Fig. 11 provides the performance comparison results of expected normalized network capacity (NNC) and average energy consumption (EC) under different conditions. For convenience, we assume that all user devices

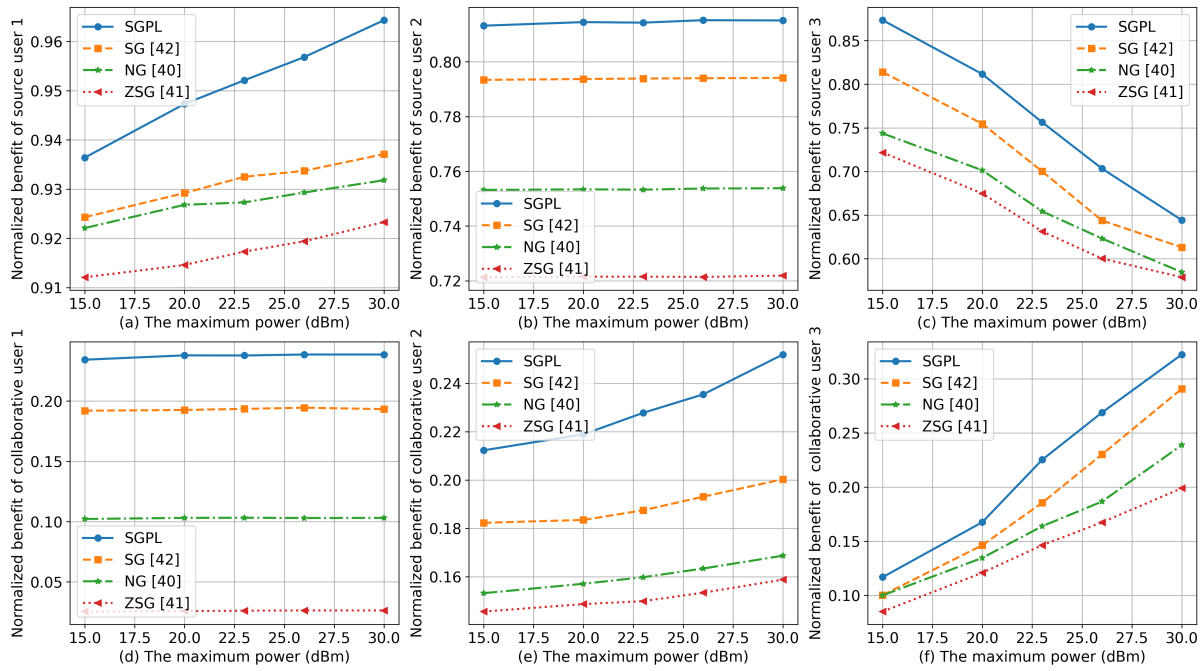


Fig. 8: Anti-jamming game benefit of each user in metaverses collaborative computing network: (a) Source user 1; (b) Source user 2; (c) Source user 3; (d) collaborative user 1; (e) collaborative user 2; (f) collaborative user 3;

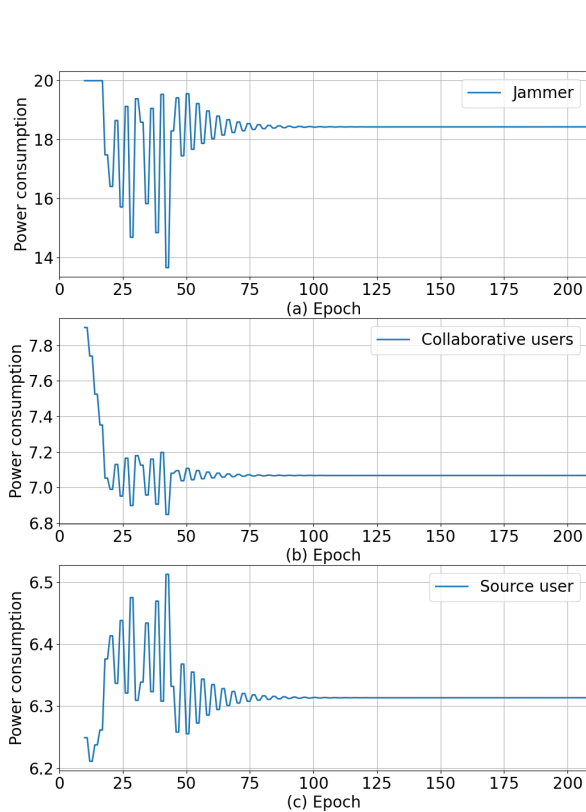


Fig. 9: Performance comparison of average power consumption of SGPL, WCSAC, TADDPG in anti-jammer game.

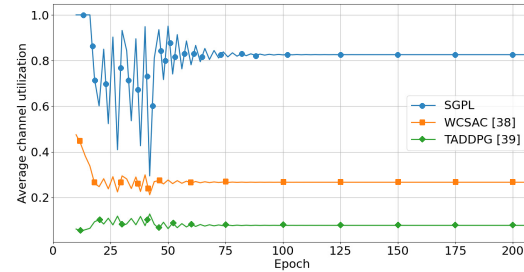


Fig. 10: Performance comparison of average channel utilization of SGPL, WCSAC, TADDPG in anti-jammer game.

have the same activity probability. According to Fig. 11, it can be seen that the proposed SGPL shows superior network performance on both NNC and EC indicators. Note that the NNC performance will increase with the activity probability. This is because with the activity of metaverse users, it is more difficult for the jammer to predict the communication situation of legitimate users, resulting in jamming failure. The increasing number of metaverse users will also strengthen the cooperation between users, making the anti-interference performance more powerful. On the contrary, the increase in the number of jammers will cause communication interference and affect the network performance. The limitation of the proposal is that the proposed algorithm uses multi-agent deep reinforcement learning technology, so in the training phase, high computing power is required as support, and as the scale of the system increases, the required computing power will also increase.

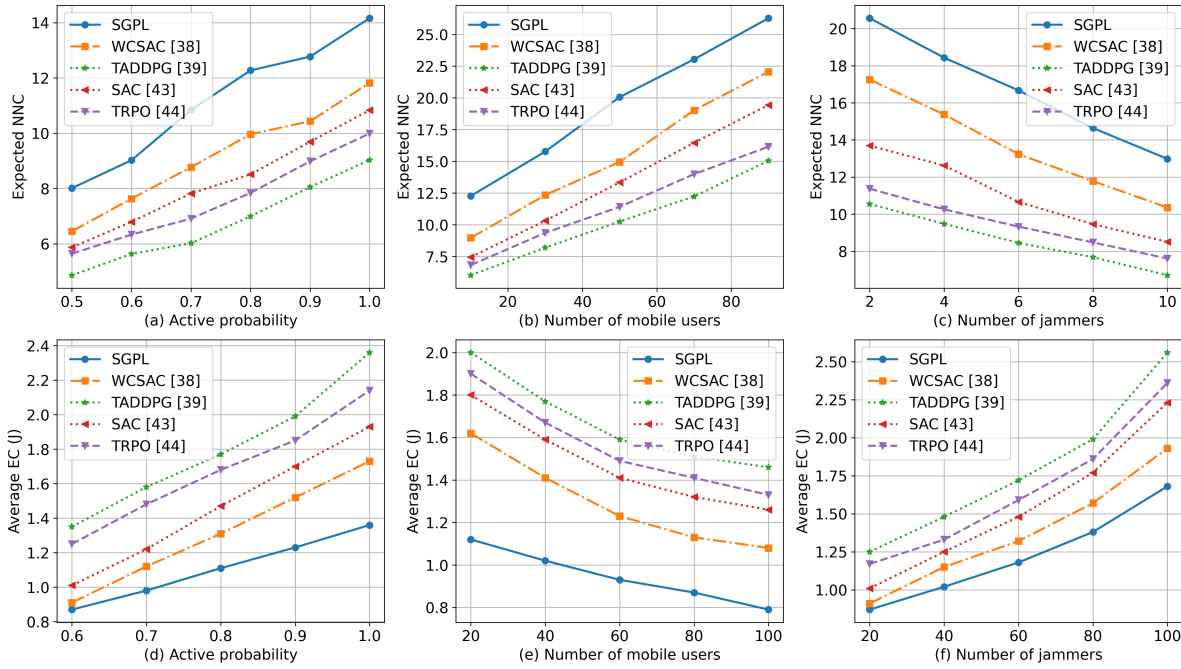


Fig. 11: Performance comparison of different algorithms in the anti-jammer game.

## VI. ENGINEERING APPLICATIONS

Metaverse introduces 5G edge computing, AI, and chips that can provide a high-performance computing portal for the computationally intensive tasks of smart vehicles. As shown in Fig. 12, in the metaverse intelligent vehicle network, vehicle users render the real-world driving environment through 5G metaverse technology to achieve an immersive driving experience. However, real-world real-time rendering tasks are computationally intensive and difficult to be completed efficiently on a single resource-limited mobile device. Therefore, metaverse mobile devices can offload metaverse rendering tasks to neighboring mobile user devices through collaborative edge computing techniques. In the metaverse collaborative computing model, idle resources from vehicles are collected to complete labor-intensive computations. However, due to the inherent property that metaverse collaborative computing is vulnerable to attacks, the vehicle device is susceptible to malicious interference from outside during task offloading computation, which affects the security of metaverse usage. Our proposed SGPL algorithm uses the metaverse edge collaborative computing model to not only solve the computational power demand problem of real-time rendering computation-intensive tasks in the metaverse vehicle network, but also resist malicious interferers and ensure the privacy security of metaverse users. Note that data is a major challenge that constrains the successful application of the proposed algorithm to the real world. The data on which the algorithm in the experiment relies does not fully reflect the real world. Therefore, obtaining access to meaningful, high quality real interaction data is critical for the practical application of the algorithm. In the absence of uniform, standardized, high-quality real-world data, it is difficult for the proposed algorithms to be truly useful.

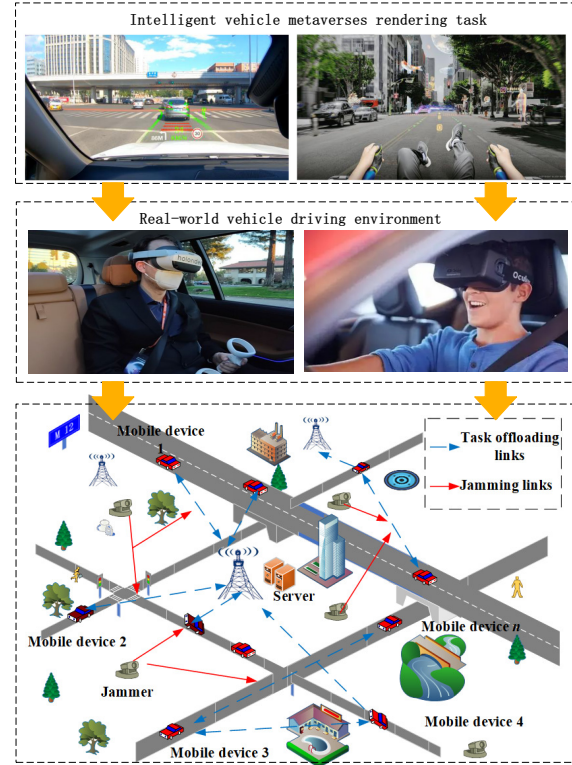


Fig. 12: Engineering Applications of SGPL.

## VII. CONCLUSION AND THE FUTURE WORK

In this paper, we propose an intelligent SGPL for jamming resistance in human-centric communication metaverses over 5G and beyond network based on the characteristics of metaverse collaborative computing networks, namely, the high

dynamics of time-varying channels, the strong antagonism between legitimate users and jammers, and the super density of user devices. We use the Stackelberg gradient dynamics model to solve the problem that channel state information has a huge amount of data, the amount of computation increases dramatically, and it is difficult to find Nash equilibrium when there are many game users. By developing accurate strategy gradient dynamics, it can reach local Stackelberg equilibrium, thus providing local convergence guarantee for the jamming game. More specifically, we use the Stackelberg dynamic equation rather than a single gradient descent to accurately capture jamming, thus reducing the jamming game cycle and speeding up the convergence of the algorithm. Finally, numerical results demonstrate the effectiveness of the proposed SGPL algorithm. In the future, we are ready to research more efficient and lower cost intelligent anti-jamming algorithm. Although SGPL improves the game speed, we can see that jamming channel information may be correlated at different times. We use the time-space correlation training model to capture the historical information of jammers, excavate additional jamming laws and knowledge, make prediction in advance to avoid jamming behavior, and realize the intelligent strategy of active anti-jamming.

## REFERENCES

- [1] M. Deveci, A. R. Mishra, I. Gokasar, P. Rani, D. Pamucar, and E. Özcan, "A decision support system for assessing and prioritizing sustainable urban transportation in metaverse," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 2, pp. 475–484, 2023.
- [2] D. Van Huynh, S. R. Khosravirad, A. Masaracchia, O. A. Dobre, and T. Q. Duong, "Edge intelligence-based ultra-reliable and low-latency communications for digital twin-enabled metaverse," *IEEE Wireless Communications Letters*, vol. 11, no. 8, pp. 1733–1737, 2022.
- [3] Y. Jiang, J. Kang, D. Niyato, X. Ge, Z. Xiong, C. Miao, and X. Shen, "Reliable distributed computing for metaverse: A hierarchical game-theoretic approach," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 1, pp. 1084–1100, 2023.
- [4] A. Hossain and N. Ansari, "5g multi-band numerology-based tdd ran slicing for throughput and latency sensitive services," *IEEE Transactions on Mobile Computing*, vol. 22, no. 3, pp. 1263–1274, 2023.
- [5] W. Cui, C. Liu, W. Yang, and L. Cai, "I-talk: Reliable and practical superimposed signal decoding without power control," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4269–4281, 2021.
- [6] Z. Hong, S. Guo, and P. Li, "Scaling blockchain via layered sharding," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 12, pp. 3575–3588, 2022.
- [7] X. Luo and P. Li, "Learning-based off-chain transaction scheduling in prioritized payment channel networks," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 12, pp. 3589–3599, 2022.
- [8] N. T. Banerjee, A. J. Baughman, S.-Y. Lin, Z. A. Witte, D. M. Klaus, and A. P. Anderson, "Side-by-side comparison of human perception and performance using augmented, hybrid, and virtual reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 4787–4796, 2022.
- [9] P. Bellavista, C. Giannelli, M. Mamei, M. Mendula, and M. Picone, "Application-driven network-aware digital twin management in industrial edge environments," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7791–7801, 2021.
- [10] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 319–352, 2023.
- [11] J. Han, M. Yang, X. Chen, H. Liu, Y. Wang, J. Li, Z. Su, Z. Li, and X. Ma, "Paradefender: A scenario-driven parallel system for defending metaverses," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–10, 2022.
- [12] M. Chen, W. Liu, N. Zhang, J. Li, Y. Ren, M. Yi, and A. Liu, "Gpds: A multi-agent deep reinforcement learning game for anti-jamming secure computing in mec network," *Expert Systems with Applications*, vol. 210, p. 118394, 2022.
- [13] Z. Shen, K. Xu, X. Xia, W. Xie, and D. Zhang, "Spatial sparsity based secure transmission strategy for massive mimo systems against simultaneous jamming and eavesdropping," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3760–3774, 2020.
- [14] Y. Xu, Y. Xu, X. Dong, G. Ren, J. Chen, X. Wang, L. Jia, and L. Ruan, "Convert harm into benefit: A coordination-learning based dynamic spectrum anti-jamming approach," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13 018–13 032, 2020.
- [15] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein, "Using channel hopping to increase 802.11 resilience to jamming attacks," in *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*. IEEE, pp. 2526–2530, 2007.
- [16] M. Chen, M. Yi, M. Huang, G. Huang, Y. Ren, and A. Liu, "A novel deep policy gradient action quantization for trusted collaborative computation in intelligent vehicle networks," *Expert Systems with Applications*, vol. 221, p. 119743, 2023.
- [17] R. D. Halloush, "Transmission early-stopping scheme for anti-jamming over delay-sensitive iot applications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7891–7906, 2019.
- [18] D. Goeckel, S. Vasudevan, D. Towsley, S. Adams, Z. Ding, and K. Leung, "Artificial noise generation from cooperative relays for everlasting secrecy in two-hop wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 10, pp. 2067–2076, 2011.
- [19] P. Wang, E. Cetin, A. G. Dempster, Y. Wang, and S. Wu, "Gnss interference detection using statistical analysis in the time-frequency domain," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 1, pp. 416–428, 2017.
- [20] M. Chen, A. Liu, W. Liu, K. Ota, M. Dong, and N. N. Xiong, "Rdrl: A recurrent deep reinforcement learning scheme for dynamic spectrum access in reconfigurable wireless networks," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 2, pp. 364–376, 2021.
- [21] C. Chen, M. Song, C. Xin, and J. Backens, "A game-theoretical anti-jamming scheme for cognitive radio networks," *IEEE network*, vol. 27, no. 3, pp. 22–27, 2013.
- [22] G.-Y. Chang, W.-H. Teng, H.-Y. Chen, and J.-P. Sheu, "Novel channel-hopping schemes for cognitive radio networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 2, pp. 407–421, 2012.
- [23] V. Kotsiou, G. Z. Papadopoulos, D. Zorbas, P. Chatzimisios, and F. Theoleyre, "Blacklisting-based channel hopping approaches in low-power and lossy networks," *IEEE Communications Magazine*, vol. 57, no. 2, pp. 48–53, 2019.
- [24] Q. Yan, H. Zeng, T. Jiang, M. Li, W. Lou, and Y. T. Hou, "Mimo-based jamming resilient communication in wireless networks," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, pp. 2697–2706, 2014.
- [25] Q. Yan, H. Zeng, T. Jiang, M. Li, W. Lou, and Y. T. Hou, "Jamming resilient communication using mimo interference cancellation," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 7, pp. 1486–1499, 2016.
- [26] Y. Wu and D. H. Tsang, "Distributed power allocation algorithm for spectrum sharing cognitive radio networks with qos guarantee," in *IEEE INFOCOM 2009*. IEEE, pp. 981–989, 2009.
- [27] J. Zheng, Y. Cai, Y. Xu, and A. Anpalagan, "Distributed channel selection for interference mitigation in dynamic environment: A game-theoretic stochastic learning solution," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4757–4762, 2014.
- [28] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 2, pp. 180–194, 2012.
- [29] Y. Sun, Q. Wu, Y. Xu, Y. Zhang, F. Sun, and J. Wang, "Distributed channel access for device-to-device communications: A hypergraph-based learning solution," *IEEE Communications letters*, vol. 21, no. 1, pp. 180–183, 2017.
- [30] L. Jia, F. Yao, Y. Sun, Y. Niu, and Y. Zhu, "Bayesian stackelberg game for anti-jamming transmission with incomplete information," *IEEE Communications Letters*, vol. 20, no. 10, pp. 1991–1994, 2016.
- [31] L. Xiao, T. Chen, J. Liu, and H. Dai, "Anti-jamming transmission stackelberg game with observation errors," *IEEE communications letters*, vol. 19, no. 6, pp. 949–952, 2015.
- [32] L. Yu, Q. Wu, Y. Xu, G. Ding, and L. Jia, "Power control games for multi-user anti-jamming communications," *Wireless Networks*, vol. 25, no. 5, pp. 2365–2374, 2019.
- [33] L. Xiao, J. Liu, Q. Li, N. B. Mandayam, and H. V. Poor, "User-centric view of jamming games in cognitive radio networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2578–2590, 2015.



- [34] M. K. Hanawal, M. J. Abdel-Rahman, and M. Krunz, "Joint adaptation of frequency hopping and transmission rate for anti-jamming wireless systems," *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2247–2259, 2016.
- [35] Y. Wu, B. Wang, K. J. R. Liu, and T. C. Clancy, "Anti-jamming games in multi-channel cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp. 4–15, 2012.
- [36] M. Min, L. Xiao, C. Xie, M. Hajimirsadeghi, and N. B. Mandayam, "Defense against advanced persistent threats: A colonel blotto game approach," in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2017.
- [37] A. Gouisse, K. Abualsaud, E. Yaacoub, T. Khattab, and M. Guizani, "Towards secure iot networks in healthcare applications: A game theoretic anti-jamming framework," *IEEE Internet of Things Journal*, pp. 1–14, 2022.
- [38] Q. Yang, T. D. Simão, S. H. Tindemans, and M. T. Spaan, "Wcsac: Worst-case soft actor critic for safety-constrained reinforcement learning," in *AAAI*, pp. 10639–10646, 2021.
- [39] D. Wu, X. Dong, J. Shen, and S. C. Hoi, "Reducing estimation bias via triplet-average deep deterministic policy gradient," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 11, pp. 4933–4945, 2020.
- [40] A. Garnaev, A. P. Petropulu, W. Trappe, and H. V. Poor, "A jamming game with rival-type uncertainty," *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5359–5372, 2020.
- [41] Y. Li, L. Shi, P. Cheng, J. Chen, and D. E. Quevedo, "Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2831–2836, 2015.
- [42] Z. Shen, K. Xu, and X. Xia, "Beam-domain anti-jamming transmission for downlink massive mimo systems: A stackelberg game perspective," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2727–2742, 2021.
- [43] 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, pp. 1861–1870, 2018.
- [44] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*. PMLR, pp. 1889–1897, 2015.



**Miaojiang Chen** received the Ph.D. degree in computer science from Central South University in 2023. He is currently an Assistant Professor of School of Computer and Electronic Information, Guangxi University, China. He has published several journal and conference papers in the IEEE transactions on network science and engineering, Knowledge-Based Systems, Expert Systems with Applications, International Journal of Intelligent Systems, Computer Network, etc., and he also serves reviewer of the top-tier conferences and journals, e.g., International

Conference on Machine Learning (ICML), IEEE transactions on industrial informatics, Expert Systems with Applications, Neural Networks, Applied Soft Computing, etc. His major research interests include deep reinforcement learning, Internet of Things, edge computing, transfer learning, optimization.

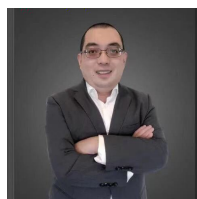


**Anfeng Liu** received the M.Sc. and Ph.D. degrees from Central South University, China, in 2002 and 2005, respectively, both in computer science. He is currently a professor of the School of Information Science and Engineering, Central South University, China. His major research interest is wireless sensor networks, Internet of Things, information security, edge computing and crowdsourcing. Dr. Liu has published 4 books and over 100 international journal and conference papers, among which there are more than 30 ESI highly-cited papers.



**Neal N. Xiong (S'05–M'08–SM'12)** is current a Professor (pendind, will be post in Oct 2023) at Department of Computer Science and Mathematics, Sul Ross State University, Alpine, TX 79830, USA. He received his both PhD degrees in Wuhan University (2007, about sensor system engineering), and Japan Advanced Institute of Science and Technology (2008, about dependable communication networks), respectively. Before he attended Sul Ross State University, he worked in Georgia State University, Northeastern State University, and Colorado Technical University (full professor about 5 years) about 15 years. His research interests include Cloud Computing, Security and Dependability, Parallel and Distributed Computing, Networks, and Optimization Theory.

Dr. Xiong published over 200 international journal papers and over 100 international conference papers. Some of his works were published in IEEE JSAC, IEEE or ACM transactions, ACM Sigcomm workshop, IEEE INFOCOM, ICDCS, and IPDPS. He has been a General Chair, Program Chair, Publicity Chair, Program Committee member and Organizing Committee member of over 100 international conferences, and as a reviewer of about 100 international journals, including IEEE JSAC, IEEE SMC (Park: A/B/C), IEEE Transactions on Communications, IEEE Transactions on Mobile Computing, IEEE Trans. on Parallel and Distributed Systems.



**Houbing Song (IEEE Fellow)** received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, in August 2012, and the M.S. degree in civil engineering from the University of Texas, El Paso, TX, in December 2006. He has served as an Associate Technical Editor for IEEE Communications Magazine (2017-present), an Associate Editor for IEEE Internet of Things Journal (2020-present), IEEE Transactions on Intelligent Transportation Systems (2021-present), and IEEE Journal on Miniaturization for Air and Space Systems (J-MASS) (2020-present), and a Guest Editor for IEEE Journal on Selected Areas in Communications (J-SAC), etc. Dr. Song is an IEEE Fellow, an ACM Distinguished Member, and an ACM Distinguished Speaker. Song is a Highly Cited Researcher identified by Clarivate™ (2021, 2022) and a Top 1000 Computer Scientist identified by Research.com. He received Research.com Rising Star of Science Award in 2022 (World Ranking: 82; US Ranking: 16). Dr. Song was a recipient of 10+ Best Paper Awards from major international conferences, including IEEE CPSCOM-2019, IEEE ICII 2019, IEEE/AIAA ICNS 2019, IEEE CBDCOM 2020, WASA 2020, AIAA/IEEE DASC 2021, IEEE GLOBECOM 2021 and IEEE INFOCOM 2022.



**Victor C. M. Leung (Life Fellow, IEEE Fellow)** is currently a Distinguished Professor of computer science and software engineering with Shenzhen University, Shenzhen, China. He is also an Emeritus Professor of electrical and computer engineering and the Director of the Laboratory for Wireless Networks and Mobile Systems, The University of British Columbia (UBC), Vancouver, Canada. His research is in the broad areas of wireless networks and mobile systems. He has published widely in archival journals and refereed conference proceedings in these areas; several of his papers have won Best Paper Awards. He is a fellow of the Royal Society of Canada, Canadian Academy of Engineering, and Engineering Institute of Canada. He was a recipient of the 1977 APEBC Gold Medal, Natural Sciences and Engineering Research Council of Canada Postgraduate Scholarships from 1977 to 1981, a 2012 UBC Killam Research Prize, IEEE Vancouver Section Centennial Award, the 2017 Canadian Award for Telecommunications Research, and the 2018 IEEE TCGCC Distinguished Technical Achievement Recognition Award. He has coauthored articles that won the 2017 IEEE ComSoc Fred W. Ellersick Prize, the 2017 IEEE Systems Journal Best Paper Award, the 2018 IEEE CSIM Best Journal Paper Award, and the 2019 IEEE TCGCC Best Journal Paper Award. He is named in the current Clarivate Analytics list of "Highly Cited Researchers."