Jamming-Resilient Physical-to-Virtual Communications in Digital Twin Edge Networks

Li Yang, *Member, IEEE,* Yifei Zou, *Member, IEEE,* Zuyuan Zhang, Peng Wang, Dongxiao Yu, *Senior Member, IEEE,* Anatolij Zubow, *Senior Member, IEEE,* Falko Dressler, *Fellow, IEEE,* Xiuzhen Cheng, *Fellow, IEEE*

Abstract—As an integration of digital twin and edge computing, the digital twin edge networks (DITENs) have been proposed in recent years to fill the gap between physical edge networks and digital systems. Meanwhile, the multi-access wireless environments in edge computing make it hard to provide ultra-reliable and low-latency communications for digital twin, especially when the jamming attacks can be launched by the adversaries. This paper studies the jamming-resilient physical-to-virtual communication (PTVC) problem in DITENs despite strong cooperative jamming. Note that the previous jamming models mainly focus on the jamming behaviors from individual adversaries and are restricted by the energy budget limitation and uniform jamming assumption. In this paper, we consider a more comprehensive jamming model, in which f adversaries can cooperatively launch their jamming attacks in totally k wireless channels with unlimited power budget and non-uniform jamming signals. Then, based on the new proposed (k, f)-cooperative jamming model, we show that $k > f$ is the necessary and also sufficient condition to solve the PTVC problem. On one hand, we prove that the PTVC problem is insoluble when $f \leq k$; on the other hand, two distributed algorithms are given as the solutions of the PTVC problem among n physical objectives and one sink node when $k > f$, the time complexity of which are $O(\frac{n \log n}{k(\log k - \log f)})$ and $O(\frac{n \log n}{\log k - \log f})$ based on the communication modes with/without acknowledgement, respectively. Both of the theoretical results and empirical simulations are conducted to show the resilience of our algorithms despite such a strong cooperative jamming model.

Index Terms—Digital twin edge networks, Physical-to-virtual communications, Cooperative jamming, Multiple wireless channels.

1 Introduction

As the integration of digital twin and edge computing, the Digital Twin Edge Networks (DITENs) have attracted lots of attention in the recent years [1]. By implementing the digital twins (DTs) on the edge side, the physical and virtual entities in the DITENs can be locally connected via single/multi-hop wireless communications. Such close connections between the physical and virtual entities help the digital twin systems to provide the real-time services with fast insight and low-latency to the users. Besides, the open-access feature of wireless channels in edge networks is friendly for the migration of twin objects and prototyping of physical objects, especially in some mobile and vehicular scenarios [2]. However, the open-access wireless environments in edge computing also open a door for the adversaries to fault the physical-to-virtual communications (PTVC) in digital twin by launching some jamming attacks, which can cause great damage to the DT system.

For example, we consider a DT system deployed on an industrial Internet of Things (IIoT) network that consists of multiple physical devices and a server on the sink node [3].

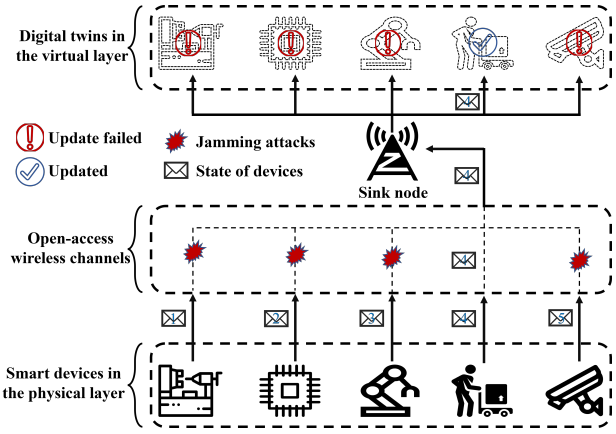


Fig. 1: Jamming attacks in the wireless channels that destroy the physical-to-virtual communications in DITENs.

With the running state vectors of IIoT devices constantly collected by the physical devices and aggregated to the sink node through the wireless channels, the digital twins of those devices are constructed and updated in the edge server, which supports the real-time decision making for the objectives in the physical layer, the accurate training and inference in the twin layer, the customized services in the services layer, and makes the DT far beyond the traditional computer-based simulations and analysis. Even though the wireless channels provide ubiquitous connections for the physical and virtual objectives, the physical-to-virtual communications become unreliable because the adversary can

- *L. Yang, Y. Zou (Corresponding Author), P. Wang, D. Yu and X. Cheng are with Institute of Intelligent Computing, School of Computer Science and Technology, Shandong University, Qingdao, 266237, P.R. China. E-mail: 202020632@mail.sdu.edu.cn, yfzou@sdu.edu.cn, wangpeng1102@163.com, dxyu@sdu.edu.cn, xzcheng@sdu.edu.cn.*
- *Z. Zhang is with the Electrical and Computer Engineering department at The George Washington University, 1918 F Street, NW Washington, DC 20052, E-mail: zuyuan.zhang@gwu.edu.*
- *A. Zubow and F. Dressler are with the School of Electrical Engineering and Computer Science, TU Berlin, Berlin, 10587, Germany. E-mail: {zubow, dressler}@ccs-labs.org.*

Manuscript received MM DD, YYYY; revised MM DD, YYYY.

easily jam the wireless channels due to their open-access feature [4], [5]. As is illustrated in Figure 1, by launching jamming attacks on the wireless channels, the adversaries can prevent the digital twins from obtaining real-time information from their physical objectives. After the virtual twins lose the association with their physical devices, the sequential attacks on those digital twins and physical objectives can be conducted, such as the forged packet injection and misconfigured policy enforcement in [6].

Currently, most of the DT works [7]–[9] are considered based on some reliable wireless environments. Whereas, as a common and critical phenomenon in the realistic wireless networks, the adversaries can fail the legitimate communications with a sufficiently large deliberate jamming signal, which is also termed as the jamming attacks. It has been proved in [10] that the disguised jamming attacks from the adversaries can reduce the communication capacity of a digital twin system to zero. In other words, if the physical-to-virtual communication protocols in DITENs are not specifically designed, it is easy for the adversaries to prevent the whole system from obtaining any valuable information from the physical layer.

To depict the jamming experienced by the physical devices in a wireless channel, there have been a series of jamming models proposed in Internet-of-Things (IoT) networks, such as the constant jamming model in [11], in which the jamming constantly occurs because of some natural faults, and the malicious jamming model in [12] to depict the various jamming behaviors from adversaries. In the malicious jamming model, to what extent can the jamming hinder the communications ups to the knowledge and capability of the adversaries. Two examples are the adaptive jamming model in [13] and the reactive jamming model in [14], [15]. In the previous one, the adversary knows the protocol and all the communication history. At the beginning of each round, it can make decisions on whether the current round should be jammed based on the history of the algorithm execution. Compared with the adaptive jamming model, the adversary in reactive jamming additionally knows the current network information when it makes the decisions.

Due to the destructiveness of the above malicious jamming on the communications in the wireless channel, the jamming-resilient algorithms under such jamming models are usually designed based on some additional restrictions, to prevent the adversary from overly potent. Two of the most important restrictions are the uniform jamming assumption and the power limited assumption in [13]–[16]. Specifically, the uniform jamming assumes that the power level of the jamming signal at each end device in a wireless channel is the same or similar so that the nodes in a same channel have the same jamming state, e.g., jammed or unjammed. And the power-limited jamming indicates that the adversary can only jam a fraction of time slots in every time window, which makes sure that there are always some unjammed time slots left for legitimate communications. Though these restrictions greatly facilitate jamming-resilient protocol design and performance analysis, an open question on the opposite side has arisen: *can the wireless communications still be resilient if the jamming in DITENs is non-uniform and span a very long time?* This question is realistic and significant currently. Because with the jamming technique rapidly developed, it is feasible for a group of adversaries to permanently jam a channel with a stable and sufficiently large energy source in edge computing [17], and inject signals with different strengths to different local areas in the network with the help of reconfigurable intelligent surface technique [18]. Besides, the previous works often consider the jamming signals from a single adversary. Whereas, a more realistic case should be multiple adversaries cooperatively jamming the communications in a wireless network.

In this paper, we answer the above question positively by proposing a more comprehensive cooperative jamming model and designing two distributed jamming-resilient physical-to-virtual communication algorithms in DITENs based on our proposed jamming model. Besides, we consider the reliable physical-to-virtual communication problem in DITENs, which contains n physical devices and one sink node[1] in a single hop wireless network. We consider a roundly-based communication in our problem, i.e., in each round, the physical devices can choose one of the channels to update its state to the sink node and the adversaries can cooperatively jam the multiple channels, which may fail a fraction of the legitimate communications. Our target is to design efficient distributed algorithms that can ensure the jamming-resilient physical-to-virtual communications in DITENs. Note that the hopping techniques [19], [20] and blind rendezvous-based strategies [21], [22] can be used to design jamming-resilient algorithms on multiple channels. Particularly, the channel hopping techniques focus on how the transmitters and receivers synchronously jump to a clean channel[2] when the current channel is jammed, in which the receivers might lose synchronization with the transmitters if too many channels are not clean. A general blind rendezvous problem requires that the secondary users are on a same channel in a same time slot, which is not occupied by the primary users. If the data packets want to be exchanged across multiple secondary users in the rendezvoused channel, existing strategies like the CSMA/CA-style procedure cannot guarantee the jamming-resilient contention resolution. While our paper focuses on how to find and use the residual clean channels, to guarantee the resilient communications from a distributed view.

Our main contributions can be summarized as follows:

- Compared with most of the previous jamming models with the uniform jamming restriction and energy budget constraint, in this paper, we propose a (k, f)-cooperative jamming model, in which the f adversaries are power unlimited with sufficient energy, and can cooperate with each other to arbitrarily launch their jamming attacks in k wireless channels. In our model, the jamming in each channel can be non-uniform and span permanently, which are more realistic and comprehensive settings than that in the previous ones. We hope that our jamming model would be helpful for the practical jamming-resilient algorithm design.
- Based on the (k, f)-cooperative jamming model, we consider the jamming-resilient physical-to-virtual

1. which can be a base station or a centralized server
2. a channel is clean if it contains no jamming signals.

communication problem in a digital twin edge network consisting of n end nodes and one sink node. Firstly, we show that such a problem cannot be solved when $k \leq f$. In the next, when $k > f$, two distributed jamming-resilient communication algorithms are presented with the time complexity of $O(\frac{n \log n}{k(\log k - \log f)})$ and $O(\frac{n \log n}{\log k - \log f})$ based on the communication modes with/without acknowledgement, respectively. Both of our algorithms can ensure the jamming-resilient physical-to-virtual communications with high probability[3] (w.h.p. for short).

Extensive simulations are conducted to evaluate the performances of our algorithms, which well corroborate our theoretical analysis. Even though this paper chooses the digital twin edge network as a typical scenario to discuss the motivation and contribution of their works, the jamming-resilient communication algorithms proposed in this paper can have broader applications in wireless networks, such as vehicular network and drone network, which are often deployed in open environments and have high requirement for real-time jamming-resilient communications.

Roadmap. The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 presents the network model, the jamming model, and the problem statement. The jamming-resilient physical-to-virtual communication algorithms are given in Section 4, followed by the analyses in Section 5, and simulation results in Section 6. Finally, Section 7 concludes our work.

2 Related Work

Jamming Models in Single/Multi-channel. Due to its significance for designing realistic jamming-resilient communication algorithms, several jamming models have been proposed in recent decades to depict the jamming attacks in single and multi-channel. Three of the typical jamming models in single channel include the constant jamming model [11], the adaptive jamming model [13], [23], and the reactive jamming model [14], [15], [24]–[26]. In the constant jamming model, the wireless channel is jammed with a constant probability in each communication round, which is a simple but efficient jamming approach [27]. In the adaptive jamming model, the adversary knows the protocol and all the communication history. At the beginning of each round, it can make decisions on whether the current round should be jammed based on the history of the algorithm execution. In the reactive jamming model, the adversary additionally knows the current network information when it makes the decisions. The other jamming models proposed in recent years include the disguised jamming in [28], the smart jamming in [29], [30], the follower jamming in [31], the strategic jamming in [32], and the hostile jamming in [33], most of which have the similar settings with adaptive and reactive jamming models, i.e., the jammer has some basic knowledge for the communication protocol and jams the channel according to its rule.

In the multi-channel scenario, the static jamming, random jamming, and adaptive jamming attacks across multiple channels are introduced in [34] and adopted in [35], respectively. In static jamming model, each static jammer permanently jams a fixed channel. With some detection strategy to find the jammed channels, the blind rendezvous problem is solved in [35]. In random jamming model, a random jammer transmits the jamming signals over randomly selected channels. In adaptive jamming model, each adaptive jammer is assumed to have all knowledge of the secondary user to guess and jam the channels that the second users are going to rendezvous. In [36], the static and reactive jamming models are considered with energy budget. Based on those two jamming models, the consensus problem is solved. Whereas, the collision and inference between legitimate devices are not considered in their communication problem. In [37], an access point is trying to deliver its message to a server via multiple orthogonal subcarriers despite the jamming attack of a jammer, i.e., it is a single-transmitter and single-jammer problem across multiple subcarriers. In [38], the multiple jammers prevent users from decoding the downlink communications from base station. Each jammer can statically jam multiple channels and equally allocates its constrained power. In [39], multiple legitimate users transmit their messages across multiple channels. A jammer can choose a channel to jam in each communication round. The sweep strategy and Q-learning based strategy are designed for the jammer. In Table 1, we listed the relevant jamming models in single channel and multiple channels, from which we can see that our jamming model is a more comprehensive and practical one.

Physical-to-Virtual Communication. As a fundamental issue in DITENs, physical-to-virtual communication has gained significant interest in recent years due to its real-time interaction between physical objects and virtual twins [1], [42], [43]. In general, the PTVC makes sure that the state and information of physical objects can be timely updated to the digital twins via wireless communication technologies, which support the accurate training/inference and real-time decision making in DITENs. Currently, the existing works in DITENs are mainly about high-level research based on reliable physical-to-virtual communications. For example, the research in [44] considers the implementation of DT on data, models, and services and addresses the internal module update and data/model integration problem. It assumes that the communications between the physical and virtual objects are stable, which is not a realistic assumption in the open-access wireless environment. The survey in [43] has pointed out that the interference and contention in the wireless channel may fail the physical-to-virtual communications in digital twin and should be carefully addressed.

In summary, most of the existing works in DITENs rely on reliable physical-to-virtual communications to design the high-level algorithms and applications, but few of them consider how to provide the reliable physical-to-virtual communications in an open-access wireless network, especially facing the jamming attacks from the adversaries. In this paper, we study the PTVC issue in complicated open-access edge scenarios. Compared with the existing jamming-resilient works, most of which are in the IoT scenarios, our adversary jamming model is stronger and more comprehensive for realistic open-access edge scenarios.

3. for some probability with $1 - n^{-\zeta}$ for some constant $\zeta > 1$.

TABLE 1: Comparison with existing jamming and similar models in single channel and multiple channels

Reference	Number of Channels	Settings of Jammers/Primary users			
		Number	Knowledge	Power Supply	Strengthen of Signal
[16], [23], [40]	Single	Single	Adaptive	Constrained	Uniform
[14], [15], [24], [26]	Single	Single	Reactive	Constrained	Uniform
[24]	Single	Single	Reactive	Constrained	Local Uniform
[13]	Multiple	Single	Adaptive	Constrained	Uniform
[41]	Multiple	Single	Static	Not mentioned	Non-Uniform
[39]	Multiple	Single	Adaptive	Not mentioned	Uniform
[21], [22]	Multiple	Multiple	Benign	Not applicable	Not applicable
[34]–[36], [38]	Multiple	Multiple	Static/Random/Adaptive	Constrained	Uniform
Our work	Multiple	Multiple	Reactive	Sufficient	Non-uniform

3 Model and Problem Statement

We consider a digital twin edge network containing n end devices and one base station as the sink node, all of which are arbitrarily placed in a two dimensional Euclidean space. The edge server is deployed on the base station, and all of the end devices are within the communication range of the base station, which consists of a single-hop edge computing network. Apart from the cellular technology, the edge device and end devices can also be connected via Wi-Fi and Bluetooth. The end devices and edge node are also termed as end nodes and edge node in the following of this paper. In our DITEN, the end nodes and sink node have a global clock and wake-up initially. Specifically, in each round, each end node endeavors to upload its message to the sink node through a shared wireless medium, while malicious adversaries can launch jamming attacks over the medium to prevent legitimate messages from being collected by the sink node. The jamming model for the adversaries, the detailed communication model for the end nodes and the sink node, and the problem statement are given in the following.

(k, f)-Cooperative Jamming Model. We consider our (k, f)-cooperative jamming model in the case that a group of f adversaries launches their jamming attacks on the k channels with the following features.

- *Reactive and cooperative adversaries:* There are f reactive and cooperative adversaries in our jamming model, each of which is equipped with a single radio. Each adversary knows the communication protocol, all the past history, current states of algorithm execution, and can arbitrarily choose a channel to jam at its own will in each round. We assume that the adversaries have the real-time and reliable communications with each other. At the beginning of each round, those adversaries can have a full discussion with each other and then cooperatively launch their jamming attacks on multiple channels. The jamming decisions of adversaries are roundly made and sustained until the current round ends, i.e., our model considers the roundly jamming.
- *Unlimited power with sufficient energy supply:* considering that the adversaries may have a strong and stable energy supply [17], we remove the energy budget limitation that is the most common restriction in the previous works. Thus, in our jamming model, the jamming signal in a channel can be sufficiently large so that no communication in the channel can succeed. Besides, in any interval, the number of jamming rounds for nodes in a channel can be sufficiently large until the whole interval is jammed, which differs from the restriction in [24] that for any long interval, there must be $\Omega(\log n)$ unjammed rounds left for legitimate communications.
- *Non-uniform jamming signals:* The previous works [16] require a uniform jamming format as the necessary condition for efficient algorithm design in single channel wireless network. Considering a reality that the cooperative adversaries can easily make the jamming signals non-uniform at each node [18], in our model, we assume that the strength of the jamming signal at a node v can be arbitrarily set by the adversaries when v is jammed.

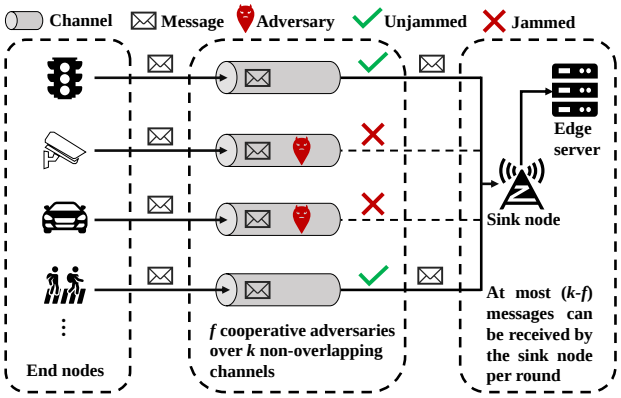


Fig. 2: Example of our (k, f)-cooperative jamming model.

We assume that these k channels are distributed over a wide range in the frequency domain, such that a single jammer is able to jam at most a single channel out of the k channel available in one round. Compared with the previous adaptive and reactive jamming models that consider the individual jamming attacks from a single adversary and have the uniform jamming and energy budget assumptions, our (k, f)-cooperative jamming model is more comprehensive and realistic, as shown in Figure 2.

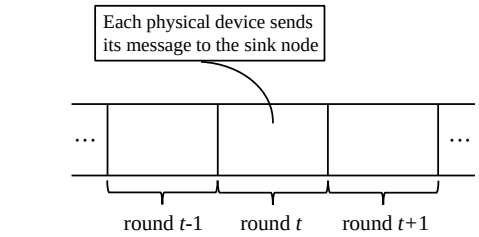
Communication Model based on Multiple Channels. All the end nodes in our digital twin edge network communicate through a shared medium divided into k non-overlapping channels. Similar with [45], we assume that only the simultaneous signals in a same wireless channel interfere with each other and the interference across multiple channels is not considered since the multiple channels

are not overlapped and are sufficiently discreted on the spectrum. Each of the end nodes is equipped with a single radio, while multiple radios are equipped on the sink node. Thus, in each round, the end nodes can choose one channel to transmit or listen to, but the sink node obtains all channels simultaneously.

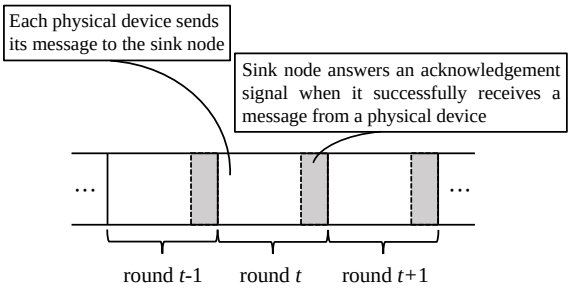
To capture the signal reception and interference generated by simultaneous transmissions in a same channel, a realistic and widely-adopted physical interference model, named as Signal-to-Interference-plus-Noise Ratio (SINR) model, is adopted. Particularly, a message sent by a node u to a node v can be correctly received if and only if (1) node u transmits and v listens in the same channel, and (2) the below defined SINR rate $SINR(u,v) \geq \beta$ holds.

$$SINR(u,v) = \frac{P_u/d(u,v)^\alpha}{\sum_{w \in Q_u} \frac{P_w}{d(w,v)^\alpha} + \sum_{j \in F_u} I(j,v) + N}, \quad (1)$$

where $d(u,v)$ is the Euclidean distance between nodes u and v, P_u is the transmission power of the node u, path-loss exponent $\alpha \in (2,6]$, and N is the ambient noise in environment. F_u is the set of adversaries that are in the same channel with u, and $I(j,v)$ is the interference from the adversary j and experienced by the node v, which is a positive value and can be arbitrarily set by j. Q_u is the set of other nodes simultaneously transmitting with u in the same channel, and we refer to $P_w/d(w,v)^\alpha$ as the interference caused by w at v. The hardware-defined threshold β is in usual larger than 1. A uniform power assignment [24] is assumed in our communication model, which means that all nodes have a fixed transmission power P. To satisfy our single-hop assumption in the DITENs, for each pair of nodes u, v, we have the assumption that $P > \beta N d(u,v)^\alpha$ according to Equation (1).



(a) Communication mode without acknowledgement



(b) Communication mode with acknowledgement

Fig. 3: Communication modes.

According to the ability of sink node, the following two communication modes are considered. In the first one, the sink node is just a silent listener which cannot send any signal [46]. In the second mode, when the sink node successfully receives a message from a physical device in a round, it immediately answers with an acknowledgement signal at the end of this round [47]. This answering signal only consists of several-bits information, and does not occupy an additional round. The two modes are termed as communication without acknowledgement and with acknowledgement, as is illustrated in Figure 3. In general, the communication with acknowledgement helps to control the contention in the wireless channel but has a higher requirement for the ability of sink node. The communication without acknowledgement is more energy-saving for the sink node, but raises more challenges for the efficient algorithm design. In this paper, both of these two communication modes will be considered.

TABLE 2: Important Variables

Notation	Definition
V, n	Set and number of end nodes
$V(t), n(t)$	Set and order of active nodes at the beginning of the round t
$\hat{V}(t), \hat{n}(t)$	Set and order of active nodes at the end of the round t
u, v, w, s	end nodes u, v, w, and sink node s
$d(u,v)$	Euclidean distance between two nodes u and v
R	Maximum distance between any pair of nodes
F, f	Set and number of adversaries
K, k	Set and number of multi-channels
P_u	Transmission power of node u
$I(j,v)$	Interference experienced by v from the adversary j
N	Environmental ambient noise
α	Path-loss exponent in SINR model, $\alpha \in (2,6]$
β	Hardware-defined threshold in SINR model, $\beta \geq 1$
$\mathbb{A}, \mathbb{I}$	Active and inactive states
$\zeta_1, \zeta_2, \zeta_3$	Sufficiently large constants
c_0-c_9	Constants in the analysis of Algorithms 1 and 2
p	Transmission probability of nodes in Algorithm 1 and 2, $p = \frac{1-2^{1-\alpha/2}}{192 \cdot 2^{\alpha+2} \cdot \beta}$

Problem Statement. We investigate the jamming-resilient physical-to-virtual communication problem in DITENs. Initially, each of the end nodes in the DITEN holds a message containing its recent physical information for uplink delivery. In the following, end nodes endeavor to update their messages to the sink node s via multi-channels. While the adversaries cooperatively launch their jamming attacks, to prevent the updating process of the end nodes. As mentioned above, all the end nodes and adversaries are equipped with a single radio while the sink node has multiple radios. In each of the following rounds, (1) each end node can choose a channel to transmit or listen; (2) the adversaries are reactive and cooperative, each of which can arbitrarily choose a channel to jam; and (3) the sink node listens in all the k channels. We say an algorithm can solve the above PTVC problem within t rounds if by executing the algorithm, all the messages from the end nodes can be received by the sink node at least once after the round t, as is illustrated in Equation 2.

$$\forall v \in V, \exists 1 \leq t' \leq t, SINR(v,s) \geq \beta \quad (2)$$

Obviously, the smaller the value of t is, the more efficient the algorithm will be. Considering the edge computing environment, it is better for the algorithm to be distributed.

Knowledge and Capability of Nodes. We assume that the number of nodes n, the number of channels k, the number of adversaries f, and SINR parameters α, β are known by all the end and sink nodes for algorithm execution. If the exact values of those parameters are not available in

reality, the upper bounds that are constant times larger than the real values can be used, which will only increase the running time of our algorithm by a constant factor. The other information, such as the network topology, or location information is not required for each node. Each end node and adversary is equipped with a single radio, while the sink node is equipped with multiple radios so that it can obtain all channels simultaneously. Physical carrier sensing or collision detection is not needed in our algorithm. By normalizing the minimum distance between any pair of nodes to 1, we use R to denote the maximum distance between any pair of nodes. The important variables used in our paper are listed in Table 2 for reference.

4 Algorithm Design and Description

In this section, two distributed jamming-resilient algorithms are designed to solve the PTVC problem despite the (k, f)-cooperative jamming, based on the two communication modes with/without acknowledgement, respectively. To make sure the physical-to-virtual communication is resilient even in the worst case, we consider our algorithm design from the harshest assumption, in which the f adversaries arbitrarily choose f channels to jam with sufficiently large power in each round. In other words, in each round, the residual clean channels are arbitrarily determined by the adversary and are unknown to the end nodes. We say a channel is clean when no adversaries chooses to jam it.

Challenge and Solution. Intuitively, if the clean channels are known by all the end nodes, our problem can be reduced to a distributed contention resolution problem, in which n end nodes compete for the usage of the clean channels efficiently. Both of the statistical exponential back-off scheme in [40] and the leader election scheme in [48] can be used to solve this problem. However, the f adversaries in our jamming model are reactive so that they can hide the residual clean channels behind the f jammed channels in each round, which is nearly impossible for the end nodes to find out. An alternative approach is to equally allocate all the end nodes to the k channels to solve the contention resolution problem. However, the negative impact of jamming on algorithm execution is non-negligible and should be carefully addressed. Let's take the statistical exponential back-off scheme in [40] as an example, which solves the contention resolution problem by an adaptive scheme within $\Omega(\log n)$ steps in a non-jamming scenario. However, if all the end nodes in multiple channels adopt the backoff scheme simultaneously, only the adaptive schemes in the clean channels progress forward for one step in each round; while the adaptive schemes in the other f jammed channels are misled by the malicious jamming signals. Thus, the whole adaptive scheme goes backward from a global view and the contention resolution problem cannot be solved.

To address this issue, we design a jamming-resilient Leader Election and Broadcast (LEB for short) scheme that can be executed by nodes in multiple channels. By letting nodes give up the leader election when they receive messages from other nodes, we can make sure that our leader election process will only be delayed but not be misled by the jamming attack. The leader election processes are simultaneously executed in k channels to make sure that $O(k)$ leaders will be elected finally despite the jamming from f adversaries. Then, we let each elected leader randomly choose a channel to broadcast for sufficient rounds. By repeating the LEB scheme for multiple times, we make sure that all the end nodes are elected as the leader for at least once, and all the messages from the end nodes are received by the sink node resiliently with high probability.

Algorithm 1: Jamming-Resilient PTVC Algo. I

For each end node v:

for $\zeta_1 n$ *times* **do**
  $state_v \leftarrow \mathbb{A}$;
  for $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ *rounds* **do**
    `// Leader election starts`
    if $state_v = \mathbb{A}$ **then**
      randomly and uniformly choose a channel i from the total k channels;
      $X \sim B(p)$; `// Bernoulli trial`
      if $(X = 1)$ **then**
        transmit its message in the channel i;
      else
        listen in the channel i;
        if *receive a message* **then**
          $state_v \leftarrow \mathbb{I}$;
    else
      do nothing;
  for $\lceil \zeta_3 \frac{k \log n}{k - f} \rceil$ *rounds* **do**
    `// Leader broadcast starts`
    if $state_v = \mathbb{A}$ **then**
      randomly and uniformly choose a channel i from the total k channels;
      transmit its message with probability p in the channel i;
    else
      do nothing;
  if $state_v = \mathbb{A}$ **then**
    halt;

For the sink node s:

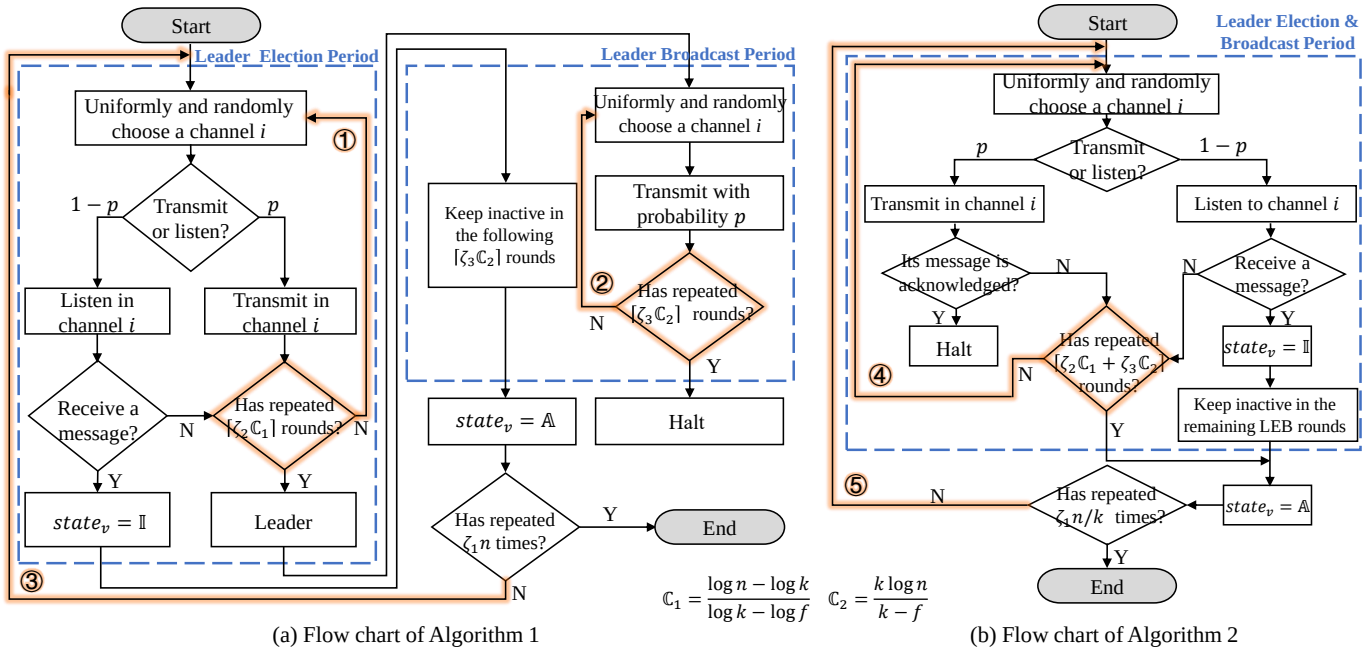
for $\lceil \zeta_1 n (\zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k - f}) \rceil$ *rounds* **do**
  listen to all channels;

Detailed Description. In this part, we show how our distributed jamming-resilient physical-to-virtual communication (JR-PTVC) algorithms work with our LEB scheme. Briefly, our JR-PTVC algorithm executes the LEB scheme for $\zeta_1 n$ times in the absence of acknowledgement. When assisted by the acknowledge mechanism, operating the LEB scheme for $\frac{\zeta_1 n}{k}$ times can fulfill the PTVC under the (k, f)-cooperative jamming model.

We firstly consider the case that the sink node is only a silent listener, and cannot acknowledge to messages from the end nodes. In general, our algorithm completes by repeating the LEB scheme for $\zeta_1 n$ times, and each LEB scheme consists of $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ rounds for leader election and $\lceil \zeta_3 \frac{k \log n}{k - f} \rceil$ rounds for leader broadcast, in which ζ_1, ζ_2 and

①: the line 4-14 of Algorithm 1 will be repeated for $\lceil \zeta_2 \mathbb{C}_1 \rceil$ times
②: the line 16-20 of Algorithm 1 will be repeated for $\lceil \zeta_3 \mathbb{C}_2 \rceil$ times
③: the Leader Election Period and the Leader Broadcast Period in Algorithm 1 will be repeated together for $\zeta_1 n$ times

④: the line 2-14 of Algorithm 2 will be repeated for $\lceil \zeta_2 \mathbb{C}_1 + \zeta_3 \mathbb{C}_2 \rceil$ times
⑤: the Leader Election & Broadcast Period in Algorithm 2 will be repeated for $\zeta_1 n/k$ times



$$\mathbb{C}_1 = \frac{\log n - \log k}{\log k - \log f} \quad \mathbb{C}_2 = \frac{k \log n}{k - f}$$

(a) Flow chart of Algorithm 1 (b) Flow chart of Algorithm 2

Fig. 4: The flow charts of JR-PTVC-I and JR-PTVC-II in terms of the end node.

Algorithm 2: Jamming-Resilient PTVC Algo. II

For each node v**:**
for $\frac{\zeta_1 n}{k}$ *times* **do**
  $state_v \leftarrow \mathbb{A}$; // LEB scheme starts
  for $\lceil \left(\zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k - f} \right) \rceil$ *rounds* **do**
    if $state_v = \mathbb{A}$ **then**
      randomly and uniformly choose a channel i from the k channels;
      $X \sim B(p)$;
      if $(X = 1)$ **then**
        transmit its message in the channel i;
        if *its message is acknowledged* **then**
          halt;
      else
        listen in the channel i;
        if *listen and receive a message* **then**
          $state_v \leftarrow \mathbb{I}$;
    else
      do nothing;

For the sink node:
for $\lceil \frac{\zeta_1 n}{k} \left(\frac{\zeta_2 (\log n - \log k)}{\log k - \log f} + \frac{\zeta_3 k \log n}{k - f} \right) \rceil$ *rounds* **do**
  Listen to all channels;
  if *receive a message from the end node* **then**
    acknowledge it;

ζ_3 are the sufficiently large constants. In our algorithms, an end node can be in two states: state $\mathbb{A}$ means that the node is active for leader competition; state $\mathbb{I}$ means that the node is inactive and has given up the leader election. When LEB scheme starts, all participating end nodes become active and transfer to state $\mathbb{A}$ initially. Then, in each round, an active node in state $\mathbb{A}$ randomly and uniformly chooses a channel to transmit its message with a constant probability p or listen otherwise. If it receives a message from other end nodes in the same channel, it gives up the leader competition, becomes inactive and transfers to state $\mathbb{I}$, and does nothing until the current LEB period ends. Notice that whether an active node can receive a message from another node is also affected by the collisions and interference in its selected channel. In our analysis part, we will show that the occurrence of collisions on each active node can be bounded and each node have some probability to receive messages from others in each communication round, until a leader is elected. After $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ rounds execution, as is proved in Lemma 1 of our analysis section, there will be at least 1 and at most $\Theta(k)$ active nodes left and then become the leaders. In the following $\lceil \zeta_3 \frac{k \log n}{k - f} \rceil$ rounds, each elected leader randomly chooses a channel to broadcast its message with probability p. As shown later, we will prove that all the leaders successfully disseminate their messages to the sink node with high probability. After a LEB period, the active leaders halt immediately and no longer participate in the following LEB periods since their messages have been received by the sink node. While the end nodes execute the LEB algorithms, the sink node always listens in all channels. The pseudocode of our distributed jamming-resilient physical-to-virtual communication algorithm (termed as JR-

PTVC-I) is given in Algorithm 1. Figure 4 (a) illustrates the flow chart of JR-PTVC-I in terms of the end nodes. In our analysis, we prove that setting $p = \frac{1-2^{1-\alpha/2}}{192 \cdot 2^{\alpha+2} \cdot \beta}$ is enough to handle all the worst cases with a high probability guarantee.

Secondly, we present algorithm JR-PTVC-II for jamming-resilient physical-to-virtual communication when the sink node can immediately acknowledge the messages from the end nodes. The pseudocode is given in Algorithm 2 and the flow chart is shown in Figure 4 (b). Similar to Algorithm 1, at the beginning of a LEB scheme, all end nodes that have not halted will be in active state. Then, in each round of the LEB scheme, each active node randomly and uniformly chooses a channel to transmit its message with probability p, or listen with probability $1-p$. If it transmits its message and receives an acknowledgement from the sink node, it immediately halts and no longer participates in the following LEB scheme. If it listens and receives a message from another end node in the same channel, it becomes inactive and turns into state $\mathbb{I}$. Within $\lceil(\zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k-f})\rceil$ rounds, as shown later in the Lemma 4, there would be $\Theta(k)$ active nodes successfully sending their messages to the sink node with high probability. In the Algorithm JR-PTVC-II, repeating the LEB scheme for $\frac{\zeta_1 n}{k}$ times is enough to make sure all the messages from the end nodes can be received by the sink node with high probability.

The Algorithm JR-PTVC-II differs from the algorithm JR-PTVC-I on the following points. The first point is that when the sink node receives a message from the end nodes, it immediately acknowledges the message. The second point is that when a leader receives the acknowledgement from the sink node, it can directly halt. With the help of the acknowledgement from the sink node, the end nodes only need to successfully broadcast once. Thus, repeating the LEB scheme for $\zeta_1 \frac{n}{k}$ times is enough to make sure all nodes have their messages received by the sink node, which is the third point.

Obviously, the running time of Algorithms JR-PTVC-I and JR-PTVC-II are $\lceil(\zeta_1 n(\zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k-f}))\rceil$ rounds and $\lceil(\frac{\zeta_1 n}{k}(\zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k-f}))\rceil$ rounds, respectively. Since n is sufficiently large than k and $\frac{k}{k-f} \in O(\frac{1}{\log k - \log f})$, their time complexity can be simplified as $O(\frac{n \log n}{\log k - \log f})$ and $O(\frac{n \log n}{k(\log k - \log f)})$. Theorem 1 is attached to show the performance of our algorithm and the proofs are given in the next section.

Theorem 1. *For the physical-to-virtual communications in the digital twin edge networks consisting of n end nodes, a sink node, k channels and f cooperative adversaries, we have*

- *$k > f$ is a necessary condition for designing the jamming-resilient communication algorithms in our (k, f)-cooperative jamming model;*
- *when $k > f$, the Algorithm JR-PTVC-I makes sure that all messages from the end nodes can be received by the sink node within $O(\frac{n \log n}{\log k - \log f})$ rounds w.h.p., if the sink node is only a silent listener;*
- *when $k > f$, the Algorithm JR-PTVC-II makes sure that all messages from the end nodes can be received by the sink node within $O(\frac{n \log n}{k(\log k - \log f)})$ rounds w.h.p., if the acknowledge mechanism can be provided by the sink node.*

5 Analyses of Our Algorithms

In this section, we first show that $k > f$ is a necessary condition for designing the jamming-resilient algorithms in our (k, f)-cooperative jamming model. Then, we prove the correctness of our algorithm JR-PTVC-I, by claiming that (1) at least one elected leader has its message successfully received by the sink node within a LEB period with high probability (2) repeating the LEB for $\zeta_1 n$ times is enough for all end nodes to have their messages received by the sink node. Thereafter, we show the correctness of our algorithm JR-PTVC-I. Armed with the acknowledgement setting, there are $\Theta(k)$ leaders had their messages received by the sink node within a LEB scheme w.h.p. Thus, the LEB scheme only needs to be repeated for $\frac{\zeta_1 n}{k}$ times.

Initially, $k > f$ is a trivial necessary condition for the jamming-resilient communication problem under the (k, f)-cooperative jamming model. In the following, we further show that $k > f$ is also the sufficient condition by proving the correctness of our algorithms when $k > f$.

5.1 Analysis of Algorithm 1

In the algorithm JR-PTVC-I, the LEB scheme is repeated for $\zeta_1 n$ times, each of which consists of $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ rounds for leader election and $\lceil \zeta_3 \frac{k \log n}{k-f} \rceil$ rounds for leader broadcast. In the following, we firstly use Lemma 1 to show that within the $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ rounds, there is at least one leader elected w.h.p. Secondly, all the elected leaders have their messages received by the sink node in the following $\lceil \zeta_3 \frac{k \log n}{k-f} \rceil$ rounds w.h.p., which is proved by Lemma 2. In other words, in each of the LEB scheme, there is at least one end node having its message received by the sink node and halted. Finally, in Lemma 3, we prove that repeating the LEB for $\zeta_1 n$ times is enough to guarantee the physical-to-virtual communications within n end nodes despite the (k, f)-cooperative jamming.

We consider an arbitrary LEB period starting from the round t_0. Let $t_1 = t_0 + \lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ and $t_2 = t_0 + \lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} + \zeta_3 \frac{k \log n}{k-f} \rceil$, i.e., the leader election period starts from the round t_0 and ends at the round t_1 and the leader broadcast period starts from the round $t_1 + 1$ and ends at the round t_2. Let $V(t)$ and $\hat{V}(t)$ be the set of active nodes at the beginning and the end of the round t. $n(t) = |V(t)|$ and $\hat{n}(t) = |\hat{V}(t)|$. Then, we have the following Lemmas for the leader election and leader broadcast periods.

Lemma 1. *When the leader election period ends, the number of active nodes is at least one and at most $c_0 k$ w.h.p., in which c_0 is a sufficiently large constant. In other words, $\hat{n}(t_1) \geq 1$ and $\hat{n}(t_1) \in O(k)$ w.h.p.*

Proof. Note that for each end node, it only gives up the leader competition when it receives the messages from other end nodes. Thus, there must be at least one active node left no matter how the algorithm was executed. Additionally, we show that the number of active nodes reduces with a factor of $c_1(\log k - \log f)$ in expectation for some constant c_1. Thus, within $\lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ rounds, the number of active nodes is within $[1, O(k)]$. Detailed technical proof can be found in the Appendix. □

Lemma 2. *When the number of active nodes is at least one and at most c_0k at the the beginning of leader broadcast period, all of them have their messages received by the sink node and halt at the end of round t_2 w.h.p.*

Proof. From Lemma 1, we have that $\hat{n}(t_1) \in [1, c_0k]$ w.h.p. Since those active nodes transmit their messages with the probability p in their selected channels and do nothing else. We have $n(t) = \hat{n}(t_1)$ for an arbitrary round $t \in [t_1+1, t_2]$. Assume that node v is an active node within the interval $[t_1+1, t_2]$. Let $\mathcal{E}_v(t)$ be the event that the message of v is received by the sink node at the round t and $Pr[\mathcal{E}_v(t)]$ is the corresponding probability. A sufficient condition for the event $\mathcal{E}_v(t)$ to hold is: v transmits in a clean channel and v is the only transmitter in the clean channel. Thus, we have

$$Pr[\mathcal{E}_v(t)] = p\frac{k-f}{k}(1-p)^{x_v(t)-1}, \tag{3}$$

in which $x_v(t)$ is the number of nodes that choose the same channel with v at round t.

Additionally, let $\mathcal{E}_v$ be the event that the message of v is received by the sink node at least once from the round t_1+1 to the round t_2, and $Pr[\mathcal{E}_v]$ is the corresponding probability. Then, we have

$$\begin{aligned} Pr[\mathcal{E}_v] &= 1 - \prod_{t=t_1+1}^{t_2} \left(1 - Pr\left(\mathcal{E}_v\left(t\right)\right)\right) \\ &\geq 1 - e^{-\sum_{t=t_1+1}^{t_2} Pr(\mathcal{E}_v(t))} \end{aligned} \tag{4}$$

The first inequality holds because for every $x_1, \cdots, x_n \in [0, \frac{1}{2}]$, $4^{-\sum_{i=1}^n x_i} \leq \prod_{i=1}^n (1-x_i) \leq e^{-\sum_{i=1}^n x_i}$.

According to the Equation (4), how likely the event $\mathcal{E}_v$ happened is determined by the value of $x_v(t)$ for the rounds $t \in [t_1+1, t_2]$, which is discussed in the following claim.

Claim 1. *$\sum_{t=t_1+1}^{t_2} x_v(t) \leq 2c_2\frac{k\log n}{k-f}$ w.h.p. for a sufficiently large positive constant c_2.*

The detailed proof of Claim 1 is given in the Appendix.

According to the AM-GM inequality[4] and equation (3), we get

$$\begin{aligned} &\sum_{t=t_1+1}^{t_2} Pr(\mathcal{E}_v(t)) = p\frac{k-f}{k}\sum_{t=t_1+1}^{t_2}(1-p)^{x_v(t)-1} \\ &\geq p\frac{k-f}{k}(t_2-t_1)\sqrt[t_2-t_1]{\prod_{t=t_1+1}^{t_2}(1-p)^{x_v(t)-1}} \\ &= p\frac{k-f}{k}(t_2-t_1)\sqrt[t_2-t_1]{(1-p)^{\sum_{t=t_1+1}^{t_2}(x_v(t)-1)}} \\ &\geq p\frac{k-f}{k}(t_2-t_1)\sqrt[t_2-t_1]{(1-p)^{(t_2-t_1)(2c_2-1)}} \\ &= p\frac{k-f}{k}(t_2-t_1)(1-p)^{2c_2-1} \\ &= p\zeta_3(1-p)^{2c_2-1}\log n. \end{aligned} \tag{5}$$

Substituting Equation (5) into Equation (4), we obtain

$$\begin{aligned} Pr[\mathcal{E}_v] &\geq 1 - e^{-\sum_{t=t_1+1}^{t_2} Pr(\mathcal{E}_v(t))} \\ &\geq 1 - e^{-p\zeta_3(1-p)^{2c_2-1}\log n} \quad \geq 1 - n^{-c_3}, \end{aligned} \tag{6}$$

where $c_3 = p\zeta_3(1-p)^{2c_2-1}$. Hence, for any node v that has been selected as the leader in LEB, it has its message successfully received by the sink node at least once when the LEB ends with a probability of $1-n^{-c_3}$. By taking a union bound on all leaders, we prove that for all elected leaders in the LEB period, their messages are received by the sink node at least once with a probability of $1-1/n^{c_3-1}$, which is still a high probability. □

Lemma 3. *After the LEB period is repeated for $\zeta_1 n$ times in the algorithm JR-PTCV-I, all end nodes have their messages received by the sink node at least once w.h.p.*

Proof. According to the results from Lemmas 1 and 2, we know that in each LEB period, there is at least one leader elected, and all the elected leaders have their messages successfully received by the sink node w.h.p.. Those elected leaders halt at the end of the LEB period, and those nodes whose messages have not been received by the sink node will participate in the following LEB periods. Thus, by setting constant ζ_1 sufficiently large, we can prove that repeating the LEB for $\zeta_1 n$ times is enough to guarantee the jamming-resilient physical-to-virtual communications from the n end nodes to the sink node despite the (k, f)-cooperative jamming. □

5.2 Analysis of Algorithm 2

As aforementioned in the algorithm JR-PTVC-II, the sink node can provide an acknowledgement for the received message, and the end nodes halt themselves when their messages are acknowledged. The halted nodes no longer transmit. Note that in the analysis of Algorithm 1, we prove that at least one sink node has its message received by the sink node in each LEB period and the LEB period has to be repeated for $\zeta_1 n$ times. In the Algorithm 2, with the help of acknowledge mechanism, we prove that at least $\Theta(k)$ sink nodes have their message received by the sink node in each LEB period and the LEB period only need to be repeated for $\zeta_1 n/k$ times, which is an improvement from the Algorithm 1. The detailed proof is given in the following.

Consider an arbitrary LEB period starting from the round t_0 and ending at the round t_2. $t_1 = t_0 + \lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} \rceil$ and $t_2 = t_0 + \lceil \zeta_2 \frac{\log n - \log k}{\log k - \log f} + \frac{\zeta_3 k \log n}{k-f} \rceil$. $n(t)$ and $\hat{n}(t)$ have been defined as the number of active nodes at the beginning and the end of the round t. Then, we have the following Lemmas for the leader election and broadcast periods.

Lemma 4. *For an arbitrary LEB period in JR-PTVC-II, if $n(t_0) \in \Omega(k)$, there are $\Theta(k)$ end nodes having their messages received by the sink node in expectation when the LEB period ends.*

Proof. This Lemma can be proved by the following three Claims. In Claims 2 and 3, we show that (1) there exists a round $t' \in [t_0, t_1]$ with $n(t') = \Theta(k)$ at least with a constant probability, and (2) none of the nodes will be active at the end of LEB with high probability, i.e., $\hat{n}(t_2) = 0$. Then, it can be seen that from the round t' to the round t_0, all the $n(t')$ active nodes either halt with their messages received by the sink node or become inactive due to receiving the messages from other end nodes. In Claim 4, we prove that the number of nodes that halts is at least the same magnitude with the number of nodes that become inactive in the interval $[t', t_2]$.

4. For any positive real number $a_1, a_2, \cdots, a_n$, $n \in \mathcal{Z}_+$, it holds $(a_1+a_2+\cdots+a_n)/n \geq \sqrt[n]{a_1a_2\cdots a_n}$

The Claims 2, 4, and 3 are given in the following, and the detailed proofs are attached in the appendix.

Claim 2. *For an arbitrary LEB period in JR-PTVC-II with $n(t_0) \in \Omega(k)$, at least with a constant probability there exists a round $t' \in [t_0, t_1]$ with $n(t') = \Theta(k)$.*

Claim 3. *For an arbitrary LEB period in JR-PTVC-II, none of the end nodes keeps active at the round t_2 with high probability, i.e., $\hat{n}(t_2) = 0$ w.h.p.*

In our algorithm JR-PTVC-II, a node no longer keeps active when it has its message received by the sink node or receives the message from an end node. Let $X_1(t)$ be the number of nodes that have their messages received by the sink node, and $X_2(t)$ be the number of nodes that receive the messages from the end nodes, at the round t respectively.

Claim 4. *In an arbitrary LEB period, in which $\hat{n}(t_2) = 0$, $t' \in [t_0, t_1]$ and $n(t') \in O(k)$, it holds that $E[\sum_{t=t'}^{t_2} X_1(t)] \geq c_4 E[\sum_{t=t'}^{t_2} X_2(t)]$ with high probability.*

Combining those results in Claims 2, 3, and 4 show that in the interval $[t', t_2]$, there are $\Theta(k)$ end nodes no longer keeps inactive, and at least constant fraction of them halt with their messages received by the sink node, which directly proves the Lemma 4. □

Lemma 5. *For an arbitrary LEB period in JR-PTVC-II, if $n(t_0) \in o(k)$, there are $c_5 n(t_0)$ active end nodes having their messages received by the sink node in expectation when the LEB period ends. c_5 is a constant smaller than 1.*

Proof. With the similar proof, we can see that Claims 3 and 4 also holds in Lemma 5 with $n(t_0) \in o(k)$. Then, we can prove that all the $n(t_0)$ nodes are inactive at the round t_2 and constant fraction of them have their messages received by the sink node. □

Lemma 6. *After the LEB period is repeated for $\frac{\zeta_1 n}{k}$ times in the algorithm JR-PTVC-II, all end nodes have their messages received by the sink node with high probability.*

Proof. Initially, there are n active nodes when the first LEB period is executed in the algorithm JR-PTVC-II. If we can prove that all the active nodes halt w.h.p. after the LEB period is repeated for $\frac{\zeta_1 n}{k}$ times, the Lemma 6 can be equally proved. From the Lemma 4, we know that when there are sufficient active nodes when a LEB period starts, i.e., $n(t_0) \in \Omega(k)$, $\Theta(k)$ active nodes will have their messages received by the sink node and halt in expectation during this LEB period. The Lemma 5 shows that when the number of active nodes is not sufficiently large compared with k when a LEB period starts, i.e., $n(t_0) \in o(k)$, at least constant fraction of active nodes halts in this LEB. In expectation, it takes $\Theta(\frac{n-k}{k})$ times LEB periods for n active nodes to reduce to $\Theta(k)$. Then, $\Theta(\log k)$ times LEB periods are needed for $\Theta(k)$ active nodes to reduce to 0. Since the reduction in Lemmas 4 and 5 are independent and occurs with a constant probability, by applying a Chernoff bound, we can prove that the number of active nodes reduces from n to 0 w.h.p. after the LEB period is repeated for $\Theta(\frac{n-k}{k} + \log k + \log n)$ times in the Algorithm HR-PTVC-II. Setting ζ_1 as a sufficiently large constant, we prove the correctness of Lemma 6. □

In the appendix, we discuss the similarity and difference of Algorithms 1 and 2.

6 Performance Evaluation

In this section, we investigate the empirical performance of our jamming-resilient physical-to-virtual communication algorithms: JR-PTVC-I without acknowledgement and JR-PTVC-II with acknowledgement, in different network sizes and jamming patterns. Specifically, we observe the success ratio of physical-to-virtual communications when our algorithms are executed with the jamming mode, the number of end nodes n, and the number of channels k varies. When a node v has its message received by the sink node, we say its physical-to-virtual communication is *successful*. The success ratio of the whole digital twin edge network is defined as the ratio of the successful physical-to-virtual communications to the total number of communications. Obviously, the success ratio increasing faster with running time indicates a more efficient jamming-resilient algorithm for PTVC problem and a shorter time delay for the messages from the end nodes to the sink node. When the success ratio obtains to 100%, all the messages from the end nodes are received by the sink node. Moreover, a comparative experiment with a previous multi-channel message aggregation work in [45] is conducted in our simulation, to show the efficiency and resilience of our algorithms despite the (k, f)-cooperative jamming.

Jamming Pattern. To simulate the jamming behaviors from the adversaries, both of the individual jamming (named "Indi.") in [16] and the cooperative jamming (named "Coop.") proposed in our model are considered in this simulation. Specifically, in the individual jamming, each of the adversaries has its own jamming target and decides its jamming policy without cooperation. While, in the cooperative jamming, the adversaries cooperatively jam the multi-channels for a same and global target, e.g., preventing the sink node to collect all the messages from the end nodes. We assume that the adversaries in edge environment have a sufficient and stable energy supply. Thus, when an adversary decides to jam a channel in both of the individual and cooperative jamming, a sufficiently large jamming signal will be launched and none of the communications in the channel succeeds. A randomized policy is considered in the individual jamming and the cooperative jamming. Specifically, in the individual jamming, each of the adversaries randomly, uniformly, and individually chooses a channel to jam in each round. No cooperation is conducted between the adversaries. While the f adversaries in the cooperative jamming will randomly and uniformly choose f channels to jam in each round. In other words, a channel will not be repeatedly jammed by multiple adversaries in cooperative jamming, which is more efficient than the individual jamming.

Simulated DITEN. We simulate a digital twin edge network in a pipeline-based Industrial Internet-of-Things scenario. As shown in Figure 5, there are 5 production lines placed in a rectangular region with the size of $140m \times 140m$. The sink node is at the center of the square area and n end nodes are randomly and uniformly deployed in those 5 production line areas, including production lines 1 to 5.

TABLE 3: Simulation parameters

Parameters in SINR model	$\alpha = 3$	$\beta = 1.5$	$N = 1$
	$R = 200m$	$P = N\beta R^{\alpha}$	
Parameters in DITEN	$k \in \{f+1, 2f, 3f, 4f\}$		$f = 20$
	$n \in \{0.5, 1.0, 1.5, 2.0\} \cdot 10^3$		
Parameters in Algo. I and II	$\zeta_1 = 1.5$	$\zeta_2 = 40$	
	$\zeta_3 = 20$	$p = 0.1$	

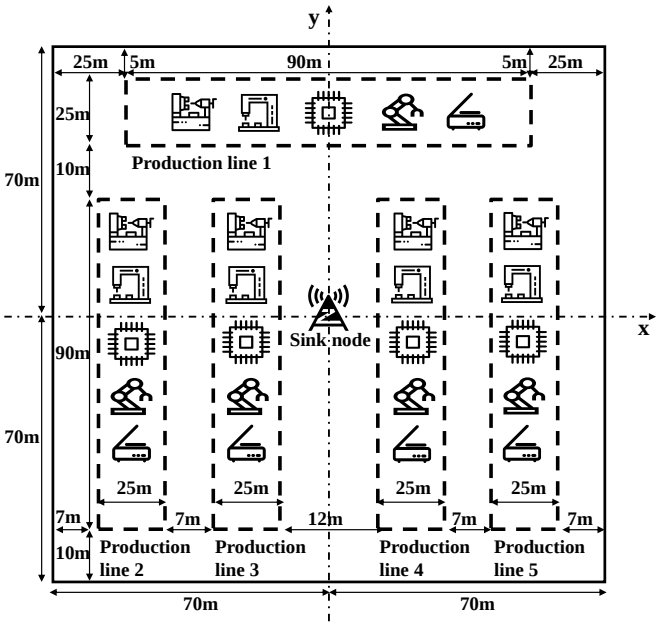


Fig. 5: Production lines 1-5 in our simulated DITEN.

Concretely, each production line works with streamlined operations. Besides, there exist some blank areas around the production lines supporting some other auxiliary applications, such as automatic guided vehicle walking. We set $R = 200m$ as the maximum transmission range of end/edge nodes. So, such a deployment makes sure a single hop environment. The number of adversaries is 20, i.e., $f = 20$. The number of end nodes n and the number of channels k vary within $[500, 2000]$ and $[21, 80]$, respectively. Table 3 gives the parameter settings in our simulation. Similar setting can also be found in [24]. Besides, our simulation is conducted by a Python programming language associated with a Python compiler, and executed on a Linux machine with Intel Xeon CPU E5-2670@2.60GHz and 128 GB main memory. Without loss of generality, for each reported result, we performed the simulation over 50 runs for an averaged result.

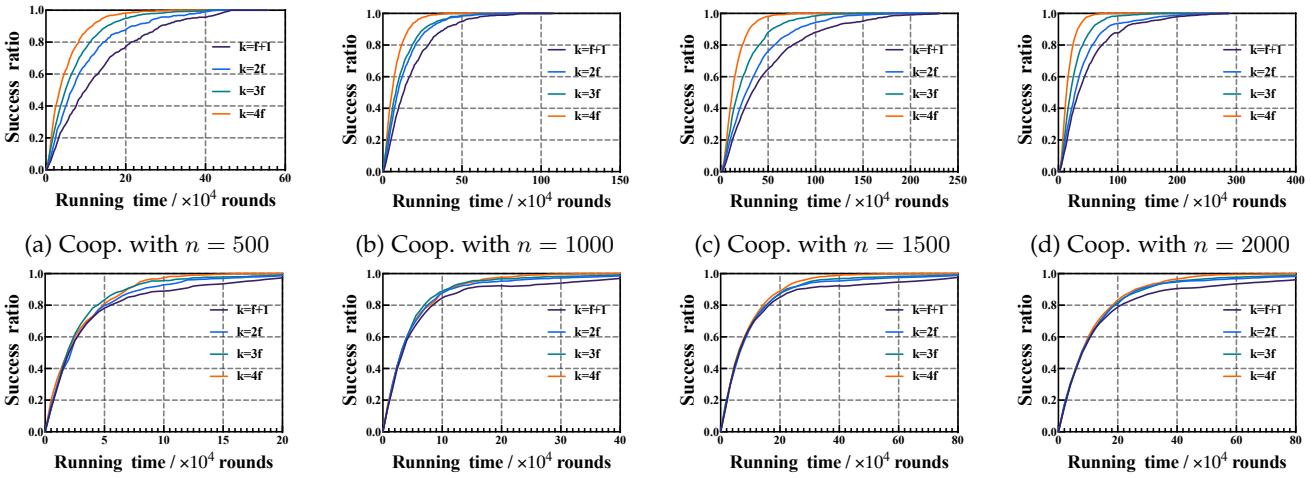
6.1 Success Ratio of our Algorithms

The success ratio of our algorithms, JR-PTVC-I and JR-PTVC-II under the cooperative jamming and individual jamming are illustrated in Figures 6 and 7, respectively. In each of the sub-figures, the x-axes and y-axes denote the running time and the success ratio of our algorithms, and four curves are given to illustrate the performance of our algorithm when the number of channels varies.

The success ratio of algorithm JR-PTVC-I under the cooperative jamming and the individual jamming is presented in Figure 6. By analyzing the curves in Figure 6, the jamming-resilience of our algorithm JR-PTVC-I and how the network parameters and jamming patterns impact the efficiency of our algorithm can be obtained.

- **Jamming-Resilience.** From each of the curves in Figure 6 with $n = 500, 1000, 1500, 2000$ and two jamming patterns, we can see that the success ratio of the DITEN gradually increases and finally reaches at 100% when the algorithm JR-PTVC-I is executed. This result verifies the jamming-resilience of our algorithm JR-PTVC-I in simulation. Even in the worst case with $n = 2000$ and $k = 21$ in Figure 6 (d), it takes about $4.5 \cdot 10^5$ rounds, $7.5 \cdot 10^5$ rounds and $2.9 \cdot 10^6$ rounds, for 60% nodes, 80% nodes and 100% nodes to succeed, respectively. Note that when $f = 20$ and $k = 21$, the 2000 nodes have to find the only 1 clean channel from the total 21 multi-channels in each round to update their messages to the sink node, which is already a very harsh condition.
- **The Impact of k.** By comparing all the curves with the same n and various k in Figure 6, we can see that when k gets larger, it takes less time for our algorithm to reach at 100% success ratio since there are more surplus channels combating the jamming attacks from the adversaries. For example, in Figure 6 (a), the success ratio reaches to 100% within $4.6 \cdot 10^5$, $4.2 \cdot 10^5$, $3.9 \cdot 10^5$, and $3 \cdot 10^5$ rounds when $k = 21, 40, 60$ and 80, respectively. This results well corroborates our theoretical analysis in Theorem 1, i.e., the time complexity of Algorithm JR-PTVC-I is $O(\frac{n \log n}{\log k - \log f})$, and indicates that increasing the spectrum resources can improve the efficiency of our algorithm on jamming-resilient PTVC problem.[5]
- **The Impact of n.** By further comparing the curves with the same k and various n in Figure 6, we can see that it takes a longer time for the success ratio to reach at 100% when n gets larger. This is because when there are more end nodes, it takes longer time for the end nodes to elect a leader, and for the sink node to receive all the messages from the end nodes. For instance, by comparing the curves with $k = 80$ and $n \in \{500, 1000, 1500, 2000\}$ in Figure 6 (a)-(d), the running times for the success ratio to reach at 100% are $3{\cdot}10^5$ rounds, $4{\cdot}10^5$ rounds, $7.5{\cdot}10^5$ rounds, and $8.5 \cdot 10^5$ rounds, respectively.
- **The Impact of Jamming Patterns.** Figure 6 (a)-(d) and Figure 6 (e)-(h) show the performance of algorithm JR-PTVC-I under the cooperative jamming and the individual jamming. Compared with the results in Figure 6 (a)-(d), the algorithm JR-PTVC-I in Figure 6 (e)-(h) has the better performance. For example, in Figure 6 (a) and (e) with $n = 500$ and $k = 21$, the success ratio reaches 100% within $4.6 \cdot 10^5$ and $2.2 \cdot 10^5$ rounds despite the cooperative and individual jamming patterns, respectively. Thus, even though our algorithm is considered for the cooperative jamming, we believe that it can have efficient performance in some other weaker jamming patterns.

5. With a wider spectrum, more sub-channels can be provided.



(a) Coop. with $n = 500$ (b) Coop. with $n = 1000$ (c) Coop. with $n = 1500$ (d) Coop. with $n = 2000$

(e) Indi. with $n = 500$ (f) Indi. with $n = 1000$ (g) Indi. with $n = 1500$ (h) Indi. with $n = 2000$

Fig. 6: Success ratio of JR-PTVC-I under the cooperative jamming and the individual jamming patterns.

Figure 7 shows the success ratio of JR-PTVC-II under cooperative jamming pattern and the individual jamming, from which we can see a similar tendency with the curves in Figure 6. That is to say, our JR-PTVC-II is also jamming-resilient in simulation. The detailed descriptions are given below:

- **Similar Jamming-Resilience.** From all the curves in Figure 7 with various k, n and jamming patterns, the success ratios eventually reach 100%, which indicate that our algorithm JR-PTVC-II is correct and jamming-resilient.
- **Similar Impacts of k and n.** By comparing the curves with the same n but various k and the curves with the same k but various n, we can see that algorithm JR-PTVC-II has a higher efficiency if more channels are provided, and it takes longer time for all physical-to-virtual communications to succeed when there are more end nodes. These results verify the time complexity $O(\frac{n \log n}{k(\log k - \log f)})$.
- **Similar Impact of Jamming Patterns.** By comparing the curves with the same n, k, but different jamming patterns, algorithm JR-PTVC-II uses less time to complete the physical-to-virtual communications despite the individual jamming, compared with its performance despite the cooperative jamming.
- **Difference on Efficiency.** The difference between the curves in Figure 6 and Figure 7 is that with the acknowledgement from the sink node, the JR-PTVC-II has higher efficiency on physical-to-virtual communications. For example, with $n = 500$, $k = 21$ and cooperative jamming in Figure 6 (a) and Figure 7 (a), the success ratio of algorithm JR-PTVC-II reaches to 100% within $2.3 \cdot 10^5$ rounds, which is about 2 times smaller than that of JR-PTVC-I in Figure 6 without acknowledgement.

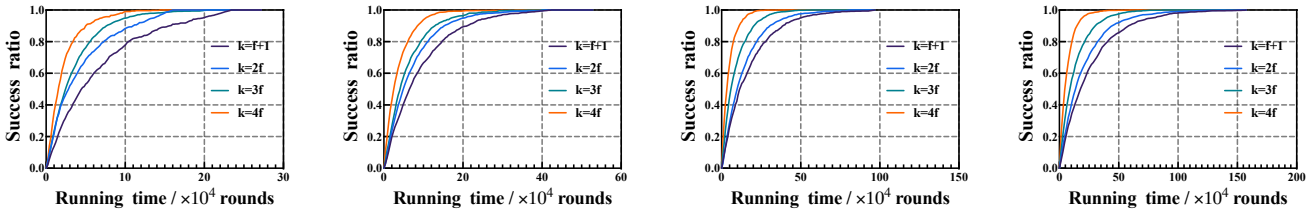
6.2 Comparison with a Previous Work

The observations on the success ratio of algorithms JR-PTVC-I and JR-PTVC-II have already shown the jamming-resilience of our algorithm. In the following, we show the efficiency of our algorithm by comparing it with a previous work in [45], which considers the data aggregation from physical devices to a sink node in ad hoc networks without considering the jamming attacks. In detail, the multi-channel message aggregation ("MCMA" for short) algorithm in [45] is designed based on the communication mode with acknowledgement. With the intra-cluster message aggregation in single channels and inter-cluster message aggregation across multiple channels, the messages from physical devices are aggregated to the sink node. Thus, the MCMA algorithm can be used to support the PTVC problem in DITENs and serves as a comparison. In comparison, we set $f = 20$ and $k = 21$.

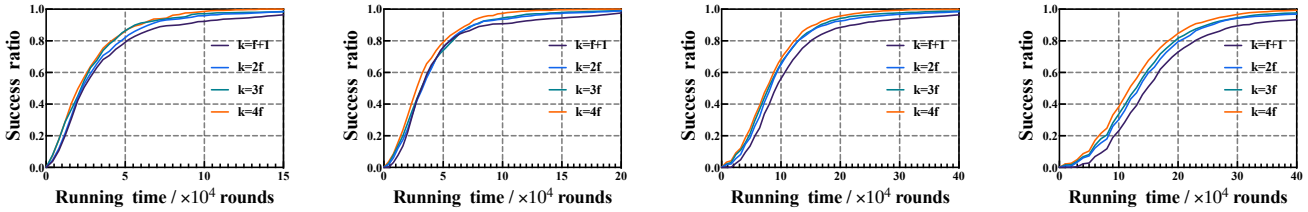
Figure 8 shows the success ratio of our algorithms JR-PTVC-I, JR-PTVC-II and the previous work MCMA with $n = \{0.5, 1.0, 1.5, 2.0\} \cdot 10^3$ and two jamming patterns. From the curves in Figure 8, it can be seen that our JR-PTVC-I and JR-PTVC-II are more efficient and resilient than the MCMA on message aggregation under different network sizes and jamming patterns. Specifically, in Figure 8 (b) with $n = 1000$, the JR-PTVC-II is the first one reaching at 100% success ratio within $2.2 \cdot 10^5$ rounds and $4.2 \cdot 10^5$ rounds despite the individual and cooperative jamming, respectively. The JR-PTVC-I follows within $4.6 \cdot 10^5$ rounds and $8.5 \cdot 10^5$ rounds. However, according to the curve in Figure 8 (b), there are still at least 50% communications not succeeded after the MCMA algorithm has been executed for $7 \cdot 10^5$ rounds. A similar tendency can be found in Figure 8 (a), (c), and (d).

6.3 Summary for Simulation Results

In this simulation, we investigate the success ratio of our algorithms JR-PTVC-I and JR-PTVC-II with jamming mode, the number of end nodes n, and the number of channels k vary. Besides, we compare our algorithms with a previous MCMA algorithm in [45], which is not specifically designed for jamming case. From the numerical results, we can see that (1) in all the simulated scenarios, JR-PTVC-I and JR-PTVC-II are jamming-resilient; (2) JR-PTVC-II is more efficient than JR-PTVC-I because the sink node additionally

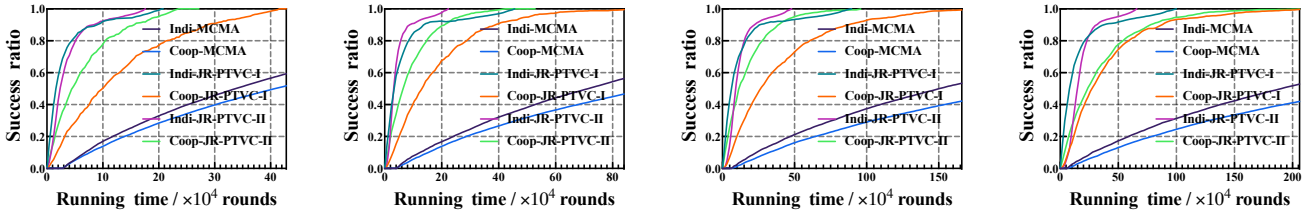


(a) Coop. with $n = 500$ (b) Coop. with $n = 1000$ (c) Coop. with $n = 1500$ (d) Coop. with $n = 2000$



(e) Indi. with $n = 500$ (f) Indi. with $n = 1000$ (g) Indi. with $n = 1500$ (h) Indi. with $n = 2000$

Fig. 7: Success ratio of JR-PTVC-II under the cooperative jamming and the individual jamming patterns.



(a) $n = 500$ (b) $n = 1000$ (c) $n = 1500$ (d) $n = 2000$

Fig. 8: Comparison among JR-PTVC-I, JR-PTVC-II, and MCMA in [45].

has an acknowledgement ability; (3) both of our algorithms are much faster and jamming-resilient than the compared algorithm in [45].

7 Conclusions

In this paper, we studied the physical-to-virtual communication problem in digital twin edge networks despite strong cooperative jamming. Note that most of the previous works consider the jamming attack from adversaries with the limited energy budget and uniform jamming constraint. In this paper, we proposed a new (k, f)-cooperative jamming model, in which f adversaries can cooperative jam total k channels with their energy budget and uniform jamming constraint removed. Thus, our new proposed jamming model is more comprehensive and realistic. Under the (k, f)-cooperative jamming model, we consider how to guarantee the jamming-resilient physical-to-virtual communications problem among n end nodes and a sink node, in which the digital twin of the end nodes are constructed. First of all, we prove that it is impossible to guarantee the reliable physical-to-virtual communications when $k \leq f$. Then, for the cases with $k > f$, two distributed algorithms are proposed with time complexities of $O(\frac{n \log n}{k(\log k - \log f)})$ and $O(\frac{n \log n}{\log k - \log f})$, for the communication modes with/without acknowledgment, respectively. Both of the two algorithms can guarantee the jamming-resilient physical-to-virtual communications in DITENs with high probability. We hope that our strong cooperative jamming model and the jamming-resilient communication schemes on multiple channels in this paper can shed some lights for the resilient protocol design in the wireless networks against jamming. Extending our research to a multi-hop and mobile scenario will be considered in future work.

Acknowledgement

This work was supported in part by the Federal Ministry of Education and Research (BMBF, Germany) within the 6G Research and Innovation Cluster 6G-RIC under Grant 16KISK020K as well as by the German Research Foundation (DFG) within the project ResCTC under grant DR 639/30-1, the Joint Key Funds of National Natural Science Foundation of China under Grant U24A20244, and Shandong Science Fund for Excellent Young Scholars (No.2023HWYQ-007).

References

[1] Y. Wu, K. Zhang, and Y. Zhang. Digital twin networks: A survey. *IEEE Internet of Things Journal*, 8(18):13789–13804, 2021.

[2] X. Yuan, J. Chen, N. Zhang, J. Ni, F. R. Yu, and V. C. M. Leung. Digital twin-driven vehicular task offloading and IRS configuration in the internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):24290–24304, 2022.

[3] J. Tan, X. Sha, B. Dai, and T. Lu. Wireless technology and protocol for iiot and digital twins. In *IEEE ITU Kaleidoscope: Industry-Driven Digital Transformation*, pages 1–8, 2020.

[4] T. X. Brown, J. E. James, and A. Sethi. Jamming and Sensing of Encrypted Wireless Ad Hoc Networks. In *IEEE/ACM MobiHoc*, pages 120–130, 2006.

[5] F. Klingler and F. Dressler. Jamming WLAN Data Frames and Acknowledgments using Commodity Hardware. In *IEEE INFOCOM Workshops*, pages 1015–1016, 2019.

[6] J. C. C. Chica, J. C. Imbachi, and J. F. Botero. Security in SDN: A comprehensive survey. *Journal of Network and Computer Applications*, 159:102595, 2020.

[7] D. Tan, V. Dang, O. A. Dobre, B. Canberk, and Q. Trung. Digital twin-aided intelligent offloading with edge selection in mobile edge computing. *IEEE Wireless Communications Letters*, 11(4):806–810, 2022.

[8] B. Li, Y. Liu, L. Tan, H. Pan, and Y. Zhang. Digital twin assisted task offloading for aerial edge computing and networks. *IEEE Transactions on Vehicular Technology*, 71(10):10863–10877, 2022.

[9] H. Liao, Z. Zhou, N. Liu, Y. Zhang, G. Xu, Z. Wang, and S. Mumtaz. Cloud-edge-device collaborative reliable and communication-efficient digital twin for low-carbon electrical equipment management. *IEEE Transactions on Industrial Informatics*, 19(2):1715–1724, 2023.

[10] T. Song, K. Zhou, and T. Li. CDMA system design and capacity analysis under disguised jamming. *IEEE Transactions on Information Forensics and Security*, 11(11):2487–2498, 2016.

[11] H. Pirayesh, P. K. Sangdeh, and H. Zeng. Securing zigbee communications against constant jamming attack using neural network. *IEEE Internet of Things Journal*, 8(6):4957–4968, 2021.

[12] Z. Ji, X. Guan, J. Tu, Q. Wu, and W. Yang. Robust trajectory and communication design in irs-assisted uav communication under malicious jamming. In *IEEE ICC Workshops*, pages 1023–1028, 2022.

[13] Q. Wang and M. Liu. Joint control of transmission power and channel switching against adaptive jamming. In *IEEE Allerton*, pages 909–916, 2013.

[14] A. Pourranjbar, G. Kaddoum, A. Ferdowsi, and W. Saad. Reinforcement learning for deceiving reactive jammers in wireless networks. *IEEE Transactions on Communications*, 69(6):3682–3697, 2021.

[15] G. Chen and W. Dong. Reactive jamming and attack mitigation over cross-technology communication links. *ACM TTransactions on Sensor Networks*, 17(1):1–25, 2020.

[16] A. Ogierman, A. Richa, C. Scheideler, S. Schmid, and J. Zhang. Competitive mac under adversarial sinr. In *IEEE INFOCOM*, pages 2751–2759, 2014.

[17] J. Guo, N. Zhao, F. R. Yu, X. Liu, and V. C. M. Leung. Exploiting adversarial jamming signals for energy harvesting in interference networks. *IEEE Transactions on Wireless Communications*, 16(2):1267–1280, 2017.

[18] X. Cheng, J. Shi, M. Sha, and L. Guo. Launching smart selective jamming attacks in wirelesshart networks. In *IEEE INFOCOM*, pages 1–10, 2021.

[19] C.-M. Chao and W.-C. Lee. Load-aware anti-jamming channel hopping design for cognitive radio networks. *Elsevier Computer Networks*, 184:107681, 2021.

[20] Y. Liu, Q. Zeng, Y. Zhao, K. Wu, and Y. Hao. Novel channel-hopping pattern-based wireless iot networks in smart cities for reducing multi-access interference and jamming attacks. *Springer EURASIP Journal on Wireless Communications and Networking*, 2021(1):1–19, 2021.

[21] L. Yu, H. Liu, Y.-W. Leung, X. Chu, and Z. Lin. Multiple radios for fast rendezvous in cognitive radio networks. *IEEE Transactions on Mobile Computing*, 14(9):1917–1931, 2015.

[22] M. A.-Rahman, H. Rahbari, M. Krunz, and P. Nain. Fast and secure rendezvous protocols for mitigating control channel dos attacks. In *IEEE INFOCOM*, pages 370–374, 2013.

[23] Q. Wang, P. Xu, K. Ren, and X.-Y. Li. Towards optimal adaptive ufh-based anti-jamming wireless communication. *IEEE Journal of Selected Areas in Communications*, 30(1):16–30, 2012.

[24] Y. Zou, D. Yu, L. Wu, J. Yu, Y. Wu, Q-S. Hua, and F. C. M. Lau. Fast distributed backbone construction despite strong adversarial jamming. In *IEEE INFOCOM*, pages 1027–1035, 2019.

[25] H. N. Van, D. T. Hoang, D. N. Nguyen, E. Dutkiewicz, and M. Mueck. Defeating smart and reactive jammers with unlimited power. In *IEEE WCNC*, pages 1–6, 2020.

[26] K. Ding, X. Ren, D. E. Quevedo, S. Dey, and L. Shi. Defensive deception against reactive jamming attacks in remote state estimation. *Elsevier Automatica*, 113:108680, 2020.

[27] E. Bayraktaroglu, C. King, X. Liu, G. Noubir, R. Rajaraman, and B. Thapa. On the performance of IEEE 802.11 under jamming. In *IEEE INFOCOM*, pages 1265–1273, 2008.

[28] Y. Liang, J. Ren, and T. Li. Secure and efficient ofdm system design under disguised jamming. In *IEEE ICNC*, pages 394–399, 2020.

[29] X. Lu, J. Jie, Z. Lin, L. Xiao, J. Li, and Y. Zhang. Reinforcement learning based energy efficient robot relay for unmanned aerial vehicles against smart jamming. *Springer Science China Information Sciences*, 65(1):1–13, 2022.

[30] C. Zhao, Q. Wang, X. Liu, C. Li, and L. Shi. Reinforcement learning based a non-zero-sum game for secure transmission against smart jamming. *Elsevier Digital Signal Processing*, 112:103002, 2021.

[31] L. Zhou, C. Zhang, Q. Zeng, X. Liu, and H. Wu. Optimal low-hit-zone frequency-hopping sequence sets with wide-gap for fhma systems under follower jamming. *IEEE Communications Letters*, 26(5):969–973, 2022.

[32] Y. Bai, S. Amin, X. Wang, and L. Jin. Securing signal-free intersections against strategic jamming attacks: A macroscopic approach. *arXiv preprint arXiv:2204.08187*, 2022.

[33] A. Jagannath, J. Jagannath, and A. Drozd. High rate-reliability beamformer design for $2\times$ 2 mimo-ofdm system under hostile jamming. In *IEEE ICCCN*, pages 1–9, 2020.

[34] Q. Wang, P. Xu, K. Ren, and X.-Y. Li. Delay-bounded adaptive ufh-based anti-jamming wireless communication. In *IEEE INFOCOM*, pages 1413–1421, 2011.

[35] J.-F. Huang, G.-Y. Chang, and J.-X. Huang. Anti-jamming rendezvous scheme for cognitive radio networks. *IEEE Transactions on Mobile Computing*, 16(3):648–661, 2017.

[36] A. Cetinkaya, K. Kikuchi, T. Hayakawa, and H. Ishii. Randomized transmission protocols for protection against jamming attacks in multi-agent consensus. *Elsevier Automatica*, 117:108960, 2020.

[37] M. Guizani, A. Gouissem, K. Abualsaud, E. Yaacoub, and T. Khattab. Combating jamming attacks in multi-channel iot networks using game theory. In *IEEE ICICT*, pages 469–474, 2020.

[38] Y. Bi, Y. Wu, and C. Hua. Deep reinforcement learning based multi-user anti-jamming strategy. In *IEEE ICC*, pages 1–6, 2019.

[39] I. Elleuch, A. Pourranjbar, and G. Kaddoum. A novel distributed multi-agent reinforcement learning algorithm against jamming attacks. *IEEE Communications Letters*, 25(10):3204–3208, 2021.

[40] B. Awerbuch, A. Richa, and C. Scheideler. A jamming-resistant mac protocol for single-hop wireless networks. In *ACM PODC*, pages 45–54, 2008.

[41] A. Gouissem, K. Abualsaud, E. Yaacoub, T. Khattab, and M. Guizani. Game theory for anti-jamming strategy in multi-channel slow fading iot networks. *IEEE Internet of Things Journal*, 8(23):16880–16893, 2021.

[42] S. Mihai, M. Yaqoob, H. V. Dang, W. Davis, P. Towakel, M. Raza, M. Karamanoglu, B. Barn, D. S. Shetve, R. V. Prasad, H. Venkataraman, R. Trestian, and H. X. Nguyen. Digital twins: A survey on enabling technologies, challenges, trends and future prospects. *IEEE Communications Surveys and Tutorials*, 24(4):2255–2291, 2022.

[43] L. U. Khan, Z. Han, W. Saad, E. Hossain, M. Guizani, and C. S. Hong. Digital twin of wireless systems: Overview, taxonomy, challenges, and opportunities. *IEEE Communications Surveys and Tutorials*, 24(4):2230–2254, 2022.

[44] Z. Jiang, Y. Guo, and Z. Wang. Digital twin to improve the virtual-real integration of industrial iot. *Elsevier Journal of Industrial Information Integration*, 22:100196, 2021.

[45] M. M. Halldórsson, Y. Wang, and D. Yu. Leveraging multiple channels in ad hoc networks. *Distributed Computing*, 32(2):159–172, 2019.

[46] G. De Marco, D. R. Kowalski, and G. Stachowiak. Deterministic contention resolution on a shared channel. In *IEEE ICDCS*, pages 472–482, 2019.

[47] M. A. Bender, T. Kopelowitz, W. Kuszmaul, and S. Pettie. Contention resolution without collision detection. In *ACM STOC*, pages 105–118, 2020.

[48] J. T. Fineman, S. Gilbert, F. Kuhn, and C. Newport. Contention resolution on a fading channel. In *ACM PODC*, pages 155–164, 2016.

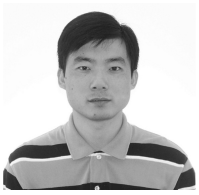
[49] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, New York, USA, 1995.

Li Yang received the M.S. degree from Fuzhou University, Fuzhou, China, in 2020. He is currently a Ph.D. student with the School of Computer Science and Technology, Shandong University, Qingdao, China, and a joint-PhD student at Paderborn University, Germany. His research interests include wireless networks and distributed computing.

Yifei Zou received the B.E. degree in 2016 from Computer School, Wuhan University, and the PhD degree in 2020 from the Department of Computer Science, The University of Hong Kong. He is currently an Assistant Professor with the school of computer science and technology, Shandong University. His research interests include wireless networks, ad hoc networks and distributed computing.

Zuyuan Zhang received the B.S. degree from the Shandong University, China in 2023. He is currently a first year Ph.D. student and a Research Assistant in Electrical and Computer Engineering department at The George Washington University. His research interests include Reinforcement Learning, Optimization and Game Theory.

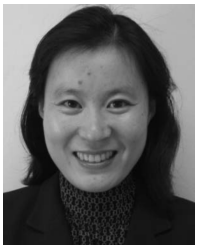
Peng Wang received his MS degree in engineering at the school of Control Science and Engineering of Shandong University in 2005. He is currently pursuing a Ph.D. degree in Computer Science from Shandong University. His research interests includes wireless network, edge computing, and distributed computing.



Dongxiao Yu received the BSc degree in 2006 from the School of Mathematics, Shandong University and the PhD degree in 2014 from the Department of Computer Science, The University of Hong Kong. He became an associate professor in the School of Computer Science and Technology, Huazhong University of Science and Technology, in 2016. He is currently a professor in the School of Computer Science and Technology, Shandong University. His research interests include wireless networks, distributed computing and graph algorithms.

Anatolij Zubow received the MSc and PhD degrees from the Department of Computer Science, Humboldt Universität zu Berlin, in 2004 and 2009, respectively. He is a senior researcher at the department of Electrical Engineering and Computer Science of Technische Universität Berlin. His research interest is on emerging wireless network architectures. Recently he is focusing mainly on coexistence of heterogeneous wireless technologies in unlicensed spectrum and high-performance WiFi networks. He has strong interest in prototyping, experimental work and testbeds. Dr. Zubow is an IEEE Senior Member. In the past, he had multiple research visits undertaken at the NEC Network Laboratories in Heidelberg, where he was working on future mobile networks.

Falko Dressler received his M.Sc. and Ph.D. degrees from the Dept. of Computer Science, University of Erlangen in 1998 and 2003, respectively. He is a full professor and Chair for Data Communications and Networking at the School of Electrical Engineering and Computer Science, TU Berlin. Dr. Dressler has been associate editor-in-chief for IEEE Trans. on Mobile Computing and Elsevier Computer Communications as well as an editor for journals such as IEEE/ACM Trans. on Networking, IEEE Trans. on Network Science and Engineering, Elsevier Ad Hoc Networks, and Elsevier Nano Communication Networks. He has been chairing conferences such as IEEE INFOCOM, ACM MobiSys, ACM MobiHoc, IEEE VNC, IEEE GLOBECOM. He authored the textbooks Self-Organization in Sensor and Actor Networks published by Wiley & Sons and Vehicular Networking published by Cambridge University Press. He has been an IEEE Distinguished Lecturer as well as an ACM Distinguished Speaker. Dr. Dressler is an IEEE Fellow as well as an ACM Distinguished Member. He is a member of the German National Academy of Science and Engineering (acatech). He has been serving on the IEEE COMSOC Conference Council and the ACM SIGMOBILE Executive Committee. His research objectives include adaptive wireless networking (radio, visible light, molecular communications) and embedded system design (from microcontroller to Linux kernel) with applications in ad hoc and sensor networks, the Internet of Things, and cooperative autonomous driving systems.



Xiuzhen Cheng received her M.S. and Ph.D. degrees in computer science from the University of Minnesota – Twin Cities in 2000 and 2002, respectively. She is a professor in the School of Computer Science and Technology, Shandong University. Her current research interests include cyber physical systems, wireless and mobile computing, sensor networking, wireless and mobile security, and algorithm design and analysis. She has served on the editorial boards of several technical journals and the technical program committees of various professional conferences/workshops. She also has chaired several international conferences. She worked as a program director for the US National Science Foundation (NSF) from April to October in 2006 (full time), and from April 2008 to May 2010 (part time). She received the NSF CAREER Award in 2004. She is Fellow of IEEE and a member of ACM.