Fast Distributed Backbone Construction Despite Strong Adversarial Jamming

Yifei Zou[†], Dongxiao Yu[*], Libing Wu[‡], Jiguo Yu[§], Yu Wu[¶], Qiang-sheng Hua[‖], Francis C.M. Lau[†]
[†]Department of Computer Science, The University of Hong Kong, Hong Kong, P.R. China
E-mail:{yfzou,fcmlau}@cs.hku.hk
[*] Institute of Intelligent Computing, School of Computer Science and Technology, Shandong University, P.R. China
E-mail: dxyu@sdu.edu.cn
[‡]Computer School of Wuhan University, Wuhan University, P.R. China.
E-mail: wu@whu.edu.cn
[§] School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), P.R. China
E-mail: jiguoyu@sina.com
[¶]School of Computer Science and Technology, Dongguan Univeristy of Technology, P.R. China
E-mail: wuyu@dgut.edu.cn
[‖]School of Computer Science and Technology, Huazhong University of Science and Technology, P.R. China.
E-mail: qshua@hust.edu.cn

***Abstract*—This paper studies jamming-resilient distributed backbone construction in multi-hop wireless networks. Specifically, a strong adversarial jamming model is proposed that captures the general jamming phenomena suffered by wireless communications. The jamming model is based on the realistic Signal-to-Interference-plus-Noise-Ratio (SINR) interference model, and is featured by local-uniformity, unrestricted energy budget and reactivity, which covers more jamming scenarios and is much closer to reality than existing jamming models. Under the strong adversarial jamming model, we propose a randomized distributed algorithm that can construct a backbone in $\mathscr{T}(O(\log n + \log R))$ rounds with high probability, where $\mathscr{T}(O(\log n + \log R))$ is the number of rounds in the interval from the beginning of the algorithm execution that contains $O(\log n + \log R)$ unjammed rounds for every node. This result is asymptotically optimal considering the trivial lower bound of $\Omega(\log n)$ for a successful transmission even without interference and jamming.**

I. Introduction

Jamming is a common and critical phenomenon in real wireless networks, which has attracted much attention from researchers in recent years. A node is jammed if the ambient noise at the node is too large relative to message reception. Jamming in wireless networks can be caused by the environment or jamming attacks. Once a node is jammed, transmissions to that node cannot be guaranteed anymore. It has been shown that the widely used IEEE 802.11 MAC protocol fails in delivering any message even when jamming persists for just a small fraction of time [3].

There have been many jamming-resilient algorithms [2], [15], [18], [19], [20], [21] proposed in previous works. Most of these works are based on traditional graph-based models, where the interference is simplified to a local and binary phenomenon. More recently, SINR-based jamming models were introduced [13], [14]. The SINR model depicts the accumulative and fading features of wireless interference. It defines that the interference fades with distance and can come from all simultaneously transmitting nodes, not just the nearby nodes. Hence, the SINR-based model reflects the wireless interference in a more precise manner than traditional graph-based models.

The adversarial approach of jamming modeling has been used in many existing works, such as [13], [14], [20], [21]. In these models, an adversary is set to cause jamming at the nodes. Some restrictions are usually added to the adversary so that they will not be overly potent, such as uniform jamming or limited energy budget. With uniform jamming, the jamming at the nodes must be the same; and with limited energy budget, the adversary can only jam a constant fraction of time slots in every time window. Though these restrictions can greatly facilitate jamming-resilient protocol performance analysis, jamming happening in reality can be dramastically different from these restrictions. Jamming can be bursty, and can span a long period of time, leaving only a small fraction of time slots to be available; and jamming can be localized, where the noises injected by the jammer can be different at different regions. Hence, it is necessary to propose a comprehensive jamming model that can cover the most general jamming scenarios, such that the devised algorithm can indeed be jamming-resilient in real operations.

In this work, we propose a strong adversarial jamming model which can cover more jamming scenarios and thus is closer to the reality than previous models. In our strong adversarial jamming model, the wireless interference is depicted by the SINR model. The jamming model is characterized by *local uniformity, unrestricted energy budget and reactivity*. Specifically, local uniformity requires that the jamming (the injected noise) at nodes within a local region to be the same, while the jamming in different regions is allowed to be different. The unrestricted energy budget means that there is no

[*] The corresponding author is Dongxiao Yu (dxyu@sdu.edu.cn).

budget restriction on the energy for jamming and the adversary can jam any round at will. Only some necessary unjammed rounds for communications are assumed. The adversary being reactive means that it can make decisions based on the history and the current state of the protocol execution. We believe this strong adversarial jamming model is comprehensive enough to cover most of the jamming scenarios in reality.

Under the proposed strong adversarial jamming model, we study the fundamental problem of backbone construction. For nodes in a given network, a connected dominating set (CDS) is a set of connected nodes, and for every node in the network, it is either in a CDS or one hop away from nodes in the CDS. A backbone is a CDS with the following additional constraints: (1) *constant degree*: each node in the CDS has a constant degree in the subgraph induced by the CDS; (2) *constant approximate diameter*: the diameter of the new communication graph constructed by connecting nodes outside the CDS with their designated one hop neighbors inside the CDS is asymptotically similar to that of the network; and (3) *constant approximate size*: the CDS is also asymptotically similar in size to the minimum CDS in the network. A backbone can greatly improve the efficiency of information exchange. When a backbone is constructed, the following two transmissions can be guaranteed: (a) the transmission between neighboring nodes in the backbone; and (b) the transmission between nodes outside the backbone and their designated one hop neighbors (usually called leaders) inside the backbone. Given such a backbone, many communication operations such as gossip, local broadcast, message broadcast, and one-to-one communication can then be implemented efficiently.

We target at jamming-resilient distributed solutions for backbone construction, as distributed solutions are more suitable for implementation in decentralized large-scale networks such as those used by Internet-of-Things. Our main contributions are summarized as follows. The performance of the proposed algorithm is guaranteed with high probability, i.e., with probability $1 - n^{-c}$ for some constant c, where n is the number of nodes in the network.

- We propose a strong adversarial jamming model which is more comprehensive (i.e., covers more jamming scenarios in reality) than existing models.
- We present an efficient randomized distributed algorithm for backbone construction. The algorithm adopts a simple transmission strategy for information exchange between neighbors, where the nodes transmit with a specified constant probability. Surprisingly, this simple oblivious strategy, which is not affected by the jamming strategy of the adversary, can fully make use of the spatial reuse provided by the SINR model and provide perfect communication efficiency. Specifically, the proposed algorithm can construct a backbone in T rounds with high probability, where T is measured from the beginning of the algorithm's execution and contains $O(\log n + \log R)$ unjammed rounds for every node; R is the communication range of nodes. Considering the lower bound $\Omega(\log n)$ for a successful tranmission even without interference and jamming [22] and in reality R is usually bounded by $poly(n)$, our algorithm is asymptotically optimal.
- We also show that the local-uniformity assumption is necessary in order to obtain asymptotically optimal solutions that are as efficient as our solution.

Roadmap. The remainder of the paper is organized as follows. Sec. II introduces the related work. Sec. III presents our strong adversarial jamming model. The backbone construction algorithm and performance analysis are covered in Sec. IV and Sec. V respectively. The necessity of the local-uniformity assumption is given in Sec. VI. Sec. VII presents the simulation results. Sec. VIII concludes the paper.

II. Related Work

There have been many works focusing on jamming-resilient communication, including mechanisms designed to avoid/detect jamming at the physical layer [10], [12], [23], coding strategies/channel surfing/spatial retreat [1], [5], [25] at the MAC layer against jamming, and hiding the transmission message from the adversary [24]. However, all these protocols can only handle *oblivious* jamming, which is fixed initially and is unrelated to the particular algorithm execution. Subsequently, stronger jamming models with adaptive adversary [2], [18], [21] and reactive adversary [13], [14], [19], [20] were proposed. If the adversary in the jamming model knows the protocol and all the communication history, and the adversary can make decisions based on the history of the algorithm execution, the adversary is called adaptive. If the adversary also knows the current network information and can instantly make a jamming decision based on that information, it is called reactive. Under the assumptions of adaptive adversary and that the adversary can only jam a constant fraction of the time steps uniformly, the problem of medium access control in a multi-hop network was studied [21]. With a similar assumption, jamming-resilient MAC protocols that can attain a constant competitive throughput were presented in [2], [18] for single-hop and multi-hop networks respectively. The result was then improved to the case of reactive jamming in a single-hop network in [19]. Under a similar reactive jamming setting, the problem of self-stabilizing leader election in a single-hop network was studied in [20]. All above results are derived under graph-based interference models. Due to the difficulty posed by the SINR model on distributed algorithm design, there are only a few works focusing on jamming-resilient protocols under this more realistic model. In [13], [14], distributed MAC protocols were presented under an SINR-based jamming model, where there is some contraint on the energy budget for the adversary such that it can only jam a constant fraction of time slots. Besides the energy budget constraint, jamming was set as totally uniform in previous works, i.e., all nodes in the network suffer from the same jamming in each round. Our jamming model considers the reactive adversary without an energy budget, as well as the local uniform assumption, which makes our model much more general than all previous ones. Additionally, [13], [14] show that the contention balancing techniques proposed in previous works will not work anymore if there is no energy budget constraint.

Backbone construction has also been extensively studied under the SINR model. However, all these works assume reliable channels for communication. In [8], the backbone construction for the case of spontaneous wakeup was considered, where the nodes are awake initially. Under the assumption that nodes know the location information, an algorithm was proposed with complexity of $O(\Delta \log^3 N)$ rounds, where Δ is the maximum node degree, and N is a linear estimate of n. The case of non-spontaneous wakeup was studied in [16], [17], and algorithms for backbone construction under different assumptions about the nodes' knowledge were presented. In the scenario where nodes do not have location information, Chlebus et al. presented algorithms for both spontaneous wakeup and non-spontaneous wakeup settings that can construct a backbone in $O(\Delta \log^7 N)$ and $O(n \log^2 N + \Delta \log^7 N)$ with high probability [6], [7]. Then deterministic algorithms under the same setting were given in [11]. In [9], the construction of a quasi-backbone which allows groups of nodes within certain distance to communicate was considered, and an algorithm with time complexity of $O(D \log^2 n)$ was proposed. To the best of our knowledge, there has been no proposal of any jamming-resilient backbone construction algorithm.

III. Models and Problem Definitions

We consider a network where n nodes are arbitrarily placed on a two dimensional Euclidean space, possibly in a worst-case fashion. Each node v has a unique identifier ID_v. For two nodes u and v, denote by $d(u,v)$ the Euclidian distance between them. The minimum distance between any pair of nodes is normalized to 1.

Communication Model. Initially, there is no prior structure in the network. The time is divided into synchronized time slots, each of which is the minimum time needed to send a message package depending on the message size. A node can communicate with other nodes via a shared channel. To make the designed algorithm applicable to both half-duplex and full-duplex transceiver equipped networks, we assume each node is equipped with a half-duplex transceiver, i.e., in each time slot, a node can transmit or listen, but cannot do both.

We assume that all nodes use the same transmission power, which is usually called *uniform* power assignment [9]. Clearly, the uniform power assignment is the easiest one to implement in real networks.

Simultaneous transmissions interfere with each other. We adopt the SINR model to depict the signal reception. In particular, a message sent by a node u can be correctly received by node v if and only if the following defined SINR rate $SINR(u,v)$ is above a hardware defined threshold β, which is larger than 1 usually.

$$SINR(u,v) = \frac{P/d(u,v)^\alpha}{\mathcal{N}(v) + \sum_{w \in W} \frac{P}{d(w,v)^\alpha}} \geq \beta, \qquad (1)$$

where α is the path-loss exponent normally between 2 and 6; $\mathcal{N}(v)$ is the ambient noise at node v, W is the set of nodes simultaneously transmitting with u, and $\sum_{w \in W} \frac{P}{d(w,v)^\alpha}$ is the interference experienced by v when u transmits.

Jamming model. Given a distance R_1, we say two nodes are R_1-*neighbors* if they are within distance R_1. Assume that the network is connected with respect to distance R[1], i.e., by connecting all pairs of R-neighbors, the obtained graph is connected. A pair of R-neighboring nodes are simply said to be neighboring.

To construct a backbone network, clearly, we need to ensure each node v can communicate with its R-neighbors. By the SINR formula, only when $\mathcal{N}(v) \leq \frac{P}{R^\alpha \beta}$, v can receive a message from its R-neighbor, and the equation holds only in an ideal scenario where there are no other simultaneous transmitters, which is impossible in multi-hop networks. Hence, to tolerate some interference, we set a *noise threshold* $N = \frac{P}{(1+\epsilon)^\alpha R^\alpha \beta}$, where ϵ is a positive constant. If in a timeslot, $\mathcal{N}(v) \geq N$, we say that the timeslot is *jammed* at v, and *unjammed* otherwise. In this work, the jamming variance on the shared channel is round-based, i.e., the jamming at every node is unchanging in each round, and a round may contain a constant number of slots as defined later.

To simply define the jamming on the shared channel, we assume there is an adversary which decides on the ambient noise $\mathcal{N}(v)$ of nodes v in each round. The adversary can define a jamming pattern with the following features.

- *Local-uniformity*: Uniform jamming globally is assumed in most previous work. Instead of that, we only need jamming to be uniform in each local region. Clearly, this local-uniform assumption on jamming is more realistic. More specifically, the whole network is divided into a grid; the adversary can determine the ambient noise at nodes in each cell; the ambient noise is the same for nodes in the same cell, but may differ for nodes in different cells.
  The division of the network area is as follows. Denote by $\mathcal{G}$ the grid obtained by the division, which consists of square cells of size $aR \times aR$, where a is a constant determined in the algorithm analysis. The division is in such a way that all cells are aligned with the coordinate axes: point $(0,0)$ is the grid origin. Each cell includes its left side without the top endpoint, and its bottom side without the right endpoint, and does not include its right and top sides. We say that (i,j) is the coordinate of the cell with its bottom left corner located at $(aR*i, aR*j)$, for $(i,j) \in \mathbb{Z}^2$. A cell with coordinate $(i,j) \in \mathbb{Z}^2$ is denoted as $g(i,j)$. For a node v locating at position (x,y) on the plane, we define its grid coordinate with respect to the grid $\mathcal{G}$ as the pair of integers (i,j) such that the point (x,y) is located in the cell $g(i,j)$ of grid $\mathcal{G}$ (i.e., $i*aR \leq x < (i+1)*aR$ and $j*aR \leq y < (j+1)*aR$). Let $\mathcal{N}_t^g$ be the ambient noise of cell g in a round t. Then for any node u, v in cell g, $\mathcal{N}(v) = \mathcal{N}(u) = \mathcal{N}_t^g$ in round t. In what follows, a round t is said to be *jammed*

[1] Note that R may not be a constant anymore after we normalize the minimum distance between nodes to 1.

for a cell g, if $\mathcal{N}_t^g > N$, and unjammed otherwise.

- *Unrestricted energy budget*: We do not set any restriction on the energy budget for the jammer. The adversary can put enough ambient noise in each particular cell in an arbitrary round to disrupt the message reception. The only requirement is that in the considered time period, $\Omega(\log n)$ rounds are unjammed for each cell, or otherwise there will be no successful transmissions in the network as shown in [22], even if there is no jamming at all.
- *Reactivity*: The adversary is assumed to be reactive, i.e., it knows the history and the current state of the protocol execution, and can instantly make a jamming decision based on that information.

Knowledge and Capability of Nodes. All nodes wake up initially. Each node know the values of R, the SINR parameters α, β, and the jamming threshold N. Nodes do not need to know the number of nodes n in the network or the number of neighbors. The nodes can acquire the location information through some services, such as GPS. But physical carrier sensing is not needed, i.e., nodes cannot detect whether the channel is busy or not.

IV. Backbone Construction Algorithm

We give our jamming-resilient backbone construction algorithm in this section. Basically, the algorithm will elect leaders in each non-empty cell and connect these leaders to construct the backbone network by letting each leader know its neighboring leaders in the backbone. However, implementing this procedure presents some challenges. First, as there nearly is not any restriction on the jamming pattern, the adaptive contention balancing strategies used in previous work cannot be used, as a long interval of jammed rounds can make the nodes judge the contention level wrongly. Consequently, in these strategies, the nodes continuously decrease their transmission probability such that it will take an unacceptably long time to make a transmission. Second, the global interference defined in the SINR model is hard to bound in a distributed setting where the nodes even do not know their neighbors. To solve these problems, the nodes transmit with a fixed constant probability. Surprisingly, this simple oblivious strategy, which is not affected by the jamming strategy of the adversary, can fully make use of the spatial reuse provided by the SINR model and perfectly solve the problems posed by the harsh communication environment.

We next introduce the algorithm in details. Initially, the cells are colored using 9 colors as follows: the cell with grid coordinate (i, j) gets color $3 * (i \mod 3) + (j \mod 3)$. For each node v, the color of the cell in which v is located is called the color of v, denoted as $color(v)$. The coloring of cells generates a TDMA scheme for the algorithm's execution. Nodes in cells with the same color execute the algorithm together, to avoid the interference from nodes in nearby cells.

The algorithm execution is divided into rounds, and each round consists of 10 slots. The detailed pseudo-code of each round is given in Algorithm 1. In each round, there are two periods: Leader Election Period (LEP) and Leader Connection Period (LCP). The LEP, consisting of 9 slots each of which is used for the algorithm execution of nodes of a particular color, is used for nodes to compete for leadership. And in the LCP, which contains 1 slot, leaders connect with neighboring leaders to construct a backbone network.

Algorithm 1: Jamming-resilient Backbone Construction

```
Initialization: $state_v = \mathbb{A}$;
In each round, each node $v$ does:
LEP( );
LCP( );
LEP( ):
$Slot = 0$;
for $Slot < 9$ do
    if $state_v = \mathbb{A}$ and $Slot = color(v)$ then
        transmit the message $\mathcal{M}_v$ with probability $p_1$;
        if receive a message $\mathcal{M}_u$ from node $u$ in the same cell then
            $state_v = \mathbb{S}$;
    $Slot + +$;
LCP( ):
if $state_v = \mathbb{A}$ then
    transmit the message $\mathcal{M}_v$ with probability $p_2$;
    if receive a message $\mathcal{M}_u$ from node $u$ in a different cell then
        update the backbone neighbor list by the the updating rule;
else
    keep silent;
```

In the algorithm, each node v has two states: $state_v = \mathbb{A}$, the active state and $state_v = \mathbb{S}$, the silent or inactive state. All nodes are in state $\mathbb{A}$ initially. During each round, in LEP, the nodes execute the algorithm by the TDMA scheduling generated by the coloring. In particular, the nodes in color j execute the algorithm in the j-th slot of each round for $0 \leq j \leq 8$. In the assigned slot, the nodes in state $\mathbb{A}$ transmit with a specified constant probability p_1. The transmitted message $\mathcal{M}_v$ from a node v only contains v's ID and location information. If a node v in state $\mathbb{A}$ receives a message $\mathcal{M}_u$ from another node u in the same cell, then v becomes inactive and joins state $\mathbb{S}$, which means that the node will not join the backbone network. The above strategy, as shown later, ensures that for each cell, after $O(\log n + \log R)$ unjammed rounds, there will be exactly one leader elected with a high probability, which is the only alive node in the cell.

In the slot of LCP, each active node v transmits $\mathcal{M}_v$ with constant probability p_2 which will be given later. If node v receives a message $\mathcal{M}_u$ from another active node u in a different cell, v updates its backbone neighbor list with the following updating rule: *for node v in cell g_v, when receiving $\mathcal{M}_u$ from node u in cell g_u, if there is no connection between* *cell g_v and g_u in v's backbone neighbor list, v records edge (v, u) as the connection between cell g_v and g_u in its backbone neighbor list; if there is already an existing but different edge $(v, u^{'})$ between cell g_v and g_u, then v replaces edge $(v, u^{'})$ with (v, u) in its backbone neighbor list.* This updating rule ensures that after leaders are elected, the neighboring leaders can establish a connection.

To ease the analysis, we set the constant parameters in the algorithm as follows: $p_1 = (1 - \frac{1}{(1+\epsilon)^\alpha}) * (1 - 2^{1-\alpha/2})/(3 * 2^{\alpha+7}\beta)$, $p_2 = (\frac{(1+\epsilon)^\alpha}{(1+\epsilon/2)^\alpha} - 1)/(\frac{4\sqrt{2}a+2+\epsilon}{a^2} \cdot \frac{6\pi\beta(1+\epsilon)^\alpha}{(1+\epsilon/2)^{\alpha-1}} \cdot \frac{\alpha-1}{\alpha-2})$.

V. Algorithm Analysis

A. Analysis Overview

Denote by $\mathscr{T}(r)$ the minimum number of rounds from the beginning of the algorithm's execution during which there are r unjammed rounds for every non-empty cell. Basically, we first prove that after $\mathscr{T}(O(\log n + \log R))$ rounds, there will be exactly one leader elected in every non-empty cell, as shown in the following Lemma 1.

Lemma 1: After the algorithm executes for $\mathscr{T}(O(\log n + \log R))$ rounds, there is exactly one leader elected in each non-empty cell w.h.p..

Then after the leaders are elected, it will be shown that after $\mathscr{T}(O(\log n))$ rounds, each leader will send its message to neighboring leaders, and the leaders constitute a CDS. Denote t_{LEP} as the first round right after leaders have been elected in each non-empty cell.

Lemma 2: After $\mathscr{T}(O(\log n))$ rounds since t_{LEP}, w.h.p., every leader sends its message to neighboring leaders, and the leaders constitute a CDS.

Then, by showing that the constructed CDS satisfies the properties of backbone, we have our final result.

Theorem 1: After $\mathscr{T}(O(\log n + \log R))$ rounds, the backbone can be constructed w.h.p.

Based on the above overview, we divide the the following analysis into three parts, to show Lemma 1, Lemma 2 and the properties of the CDS, respectively. Also, we assume the parameter $a = \min\{\frac{\sqrt{2}}{8}\epsilon, \frac{\sqrt{2}}{2}\}$ in the network division.

B. Analysis for Leader Election Period

In LEP, by the TDMA scheme, nodes in cells of the same color execute the algorithm together. Hence, we next consider the LEP execution of nodes in the cells of a particular color.

A node division is used to depict the network topology in each cell of the grid. Consider the network at the beginning of a round r. V^g is the set of nodes in cell g, which is divided into classes $\{V_i^g : i = 0, 1, \ldots, \log R\}$. For a node $v \in g$, let u be v's nearest neighbor in g if v has at least one neighbor in g. v is in class V_i^g for $0 \le i \le \log R - 1$ if $d(u, v) \in [2^i, 2^{i+1})$, or otherwise v is in class $V_{\log R}^g$. By the division of cells, it can be seen that if $V_{\log R}^g$ is non-empty, there is only one node in g. For set V_i^g at the beginning of round r, let $n_i^g(r) = |V_i^g|$. Furthermore, we use $V_{<i}^g$ to denote the sets of active nodes in classes V_j^gs for $j < i$. $n_{<i}^g(r)$ is defined as the number of active nodes correspondingly at the beginning of a round r.

Basically for an initially non-empty cell g, only one leader will be elected when all V_i^g for $i \in \{0, 1, \ldots, \log R - 1\}$ are reduced to empty. This is because when all V_i^g for $i \in \{0, 1, \ldots, \log R - 1\}$ are reduced to empty, there will be exactly one active node left in $V_{\log R}^g$.

We consider a particular non-empty cell g. The analysis for LEP consists of two parts: We first prove the reduction speed for the active nodes in g in an unjammed round, and then based on this result, analyze the reduction time for V_i^g to become empty in the strong adversarial jamming model.

We observe that with $p_1 = (1 - \frac{1}{(1+\epsilon)^\alpha}) * (1 - 2^{1-\alpha/2})/(3 * 2^{\alpha+7}\beta)$, in an unjammed round, a constant fraction of nodes in set V_i^g experience a limited interference from nodes in $\cup_{g \in \mathcal{G}} V_{\ge i}^g$; and with the assumption that $n_{<i}^g(r) \le \frac{1-(2^{1-\alpha/2})}{2} n_i^g(r)$, the interference from nodes in $\cup_{g \in \mathcal{G}} V_{<i}^g$ is also bounded. Thus in each unjammed round, a constant fraction of nodes in V_i^g receive messages and become inactive. Then, we get the result for the first part in the following Lemma 3, and the proof is omitted because of lack of space.

Lemma 3: For non-empty cell g in an unjammed round r, if for $i \in \{0, 1, \ldots, \log R - 1\}$, $n_{<i}^g(r) \le \varepsilon n_i^g(r)$ with $\varepsilon = \frac{1-(2^{1-\alpha/2})}{2}$, then with probability $1 - e^{\Omega(|V_i^g|)}$, γ fraction of nodes in V_i^g will become inactive at the end of round r, where $\gamma = \frac{p_1(1-p_1)}{8*(2s+5)^2}$.

Even with Lemma 3, it is still not easy to analyze the reduction of active nodes in cell g, because the reduction of $\{V_i^g\}$ may be influenced by the jamming round and continuously changes since some active nodes in $V_{<i}^g$ join V_i^g because their nearest neighbors become inactive. We need to show that even with these influences, each class V_i^g for $i \in \{0, 1, 2, \ldots \log R - 1\}$ finally reduces to empty, which means that there is exactly one active node left in g, and the reduction time is proved to be $\mathscr{T}(O(\log n + \log R))$ rounds.

Let $\gamma_1 = 1 - \gamma$ and $\gamma_2 = \gamma_1 + \rho/(1 - \rho)$ where ρ is a constant that will be fixed later. In what follows, we will try to upper-bound the number of active nodes in each class V_i^g by a series of vectors. In particular, we define $\{m_i^g(t) : t \ge 0 \; and \; 0 \le i \le \log R - 1\}$ as follows.

$$\forall t \ge 0 : m_i^g(t) = \begin{cases} n/\gamma_1 & t \le T_i \\ \lfloor m_i^g(t-1) * \gamma_2 \rfloor & t > T_i \end{cases}$$

Here $T_i = i * h$ and $h = \lceil \log_{\gamma_2} \rho \rceil$.

Our ensuing analysis consists of two parts. We first find a bound $\hat{T}$ such that all $m_i^g(\hat{T})$ becomes 0. Then define random events $\mathcal{E}(j)$ for $j \ge 0$: $\mathcal{E}(j)$ occurs when for some round r, $n_i^g(r) \le m_i^g(j)$ for all $i \in \{0, 1, \ldots, \log R - 1\}$. Hence when $\mathcal{E}(\hat{T})$ occurs, all classes V_i^g for $i \in \{0, 1, \ldots, \log R - 1\}$ become empty. So we only need to analysis when $\mathcal{E}(\hat{T})$ occurs.

By the definition of $m_i^g(t)$, obviously, $\hat{T} \in O(\log n + \log R)$. We next give a sufficient condition for $\mathcal{E}(j+1)$ to occur when $\mathcal{E}(j)$ has occurred. Let $\hat{m}_i^g(t+1) = \gamma_1 m_i^g(t)$.

Lemma 4: If in an unjammed round r for cell g, $\mathcal{E}(j)$ occurs and $n_i^g(r) \le \hat{m}_i^g(j+1)$, then $n_i^g(r+1) \le m_i^g(j+1)$.

Proof: We prove this lemma in two cases: (case 1) when $m_i^g(j) = n/\gamma_1$, $n_i^g(r+1) \le n < m_i^g(j+1) = n\frac{\gamma_2}{\gamma_1}$; (case 2) with $m_i^g(j) < n/\gamma_1$, because $\mathcal{E}(j)$ occurs, for $\forall i \in$

$\{0, 1, \ldots, \log R - 1\}$, $n_i^g(r) \leq m_i^g(j)$, and $\sum_{s=0}^{i-1} n_s^g(r) \leq \sum_{s=0}^{i-1} m_s^g(j) = m_i^g(j)\rho/(1-\rho)$, then,

$$\begin{aligned} n_i^g(r+1) &\leq n_i^g(r) + \sum_{s=0}^{i-1} n_s^g(r) \leq \hat{m}_i^g(j+1) + \sum_{s=0}^{i-1} m_s^g(j) \\ &\leq m_i^g(j)\gamma_2 - m_i^g(j)\rho/(1-\rho) + m_i^g(j)\rho/(1-\rho) \\ &= m_i^g(j+1) \end{aligned}$$

Combining the two cases, we finish the proof. ■

Lemma 5: If in an unjammed round r, $\mathcal{E}(j)$ occurs, then with probability at least $1 - e^{-\Omega(n_i^g(r))}$, $n_i^g(r+1) < m_i^g(j+1)$, where $i \in \{0, 1, \ldots, \log R - 1\}$.

Proof: We consider this in three cases: (case 1) with $m_i^g(j) = n/\gamma_1$, $\hat{m}_i^g(j+1) = n$ and (case 2) with $n_i^g(r) < \hat{m}_i^g(j+1)$, the lemma can be directly proved for case 1 and by Lemma 4 for case 2. We next consider (case 3) where $n_i^g(r) \geq \hat{m}_i^g(j+1)$ and $m_i^g(j) < n/\gamma_1$.

Because $\mathcal{E}(j)$ occurs and $m_i^g(j) < n/\gamma_1$, $n_{<i}^g(r) \leq m_{<i}^g(j) \leq m_i^g(j)\rho/(1-\rho)$. Then mathematically we have $n_{<i}^g(r) \leq n_i^g(r)\frac{\rho}{\gamma_1(1-\rho)}$. By setting ρ to be small enough to make sure $\rho/(1-\rho) < \gamma_1\varepsilon$, we obtain $n_{<i}^g(r) < \varepsilon n_i^g(r)$ from the above inequation. Then, in round r, by Lemma 3, with probability $1 - e^{-\Omega(n_i^g(r))}$,

$$\begin{aligned} n_i^g(r+1) &\leq \gamma_1 n_i^g(r) + \sum_{s=0}^{i-1} n_s^g(r) \leq \gamma_1 m_i^g(j) + \sum_{s=0}^{i-1} m_s^g(j) \\ &= \hat{m}_i^g(j+1) + \sum_{s=0}^{i-1} m_s^g(j) \leq m_i^g(j+1) \end{aligned}$$

■

We next bound the number of unjammed rounds between the time from $\mathcal{E}(j)$ to $\mathcal{E}(j+1)$. Assume that the algorithm's processing is divided into successive intervals $\mathbb{J}$, each of which consists of $\max\{\frac{2\tau}{a_1(1-\gamma_2)}, 2\tau\}$ unjammed rounds, where $\tau = \max\{m_i^g(j+1)/\hat{m}_i^g(j+1)\} > 1$ for $i \in \{0, 1, \ldots, \log R - 1\}$, $\hat{m}_i^g(j+1) > 0$ and a_1 is the constant behind the Ω notation in the probability guarantee in Lemma 5. Then

Lemma 6: If $\mathcal{E}(j)$ happens at the beginning round of interval $\mathbb{J}_a$, $\mathcal{E}(j+1)$ occurs at the beginning round of interval $\mathbb{J}_{a+1}$ with probability at least $1/2$.

Proof: For $i \in \{0, 1, \ldots, \log R - 1\}$, for the cases: $m_i^g(j) = 0$ or $n_i^g = 0$, it is easy to get $n_i^g \leq m_i^g(j) \leq m_i^g(j+1)$; for the case that $m_i^g(j) \geq n_i^g > 0$, the probability that n_i^g is larger than $m_i^g(j+1)$ after the interval is

$$\begin{aligned} e^{-2\tau n_i^g/(1-\gamma_2)} &\leq (1-\gamma_2)/(2\tau n_i^g) \leq (1-\gamma_2)/(2\tau \hat{m}_i^g(j+1)) \\ &\leq (1-\gamma_2)/(2m_i^g(j+1)). \end{aligned}$$

Then using union bound on the error probabilities given above for all is, the probability that there is at least one n_i^g larger than $m_i^g(j+1)$ after interval $\mathbb{J}_a$ is at most

$$\sum_{i=0}^{\log R-1} (1-\gamma_2)/(2m_i^g(j+1)) \leq \frac{1-\gamma_2}{2} \sum_{i=0}^{+\infty} \gamma_2^i \leq \frac{1}{2}$$

Hence, with probability at least $1/2$, $\mathcal{E}(j+1)$ occurs at the beginning round of $\mathbb{J}_{a+1}$, which completes the proof. ■

Now we are ready to prove Lemma 1 by combining the analysis in LEP together.

Proof for Lemma 1

Proof: By the definition of $\mathcal{E}(\hat{T})$, when $\mathcal{E}(\hat{T})$ occurs, all V_i^g, $i \in \{0, 1, \ldots, \log R - 1\}$ are reduced to empty, and each non-empty cell has exactly one leader elected. We bound the time when $\mathcal{E}(\hat{T})$ occurs by induction. Since $\mathcal{E}(0)$ always occurs, and by Lemma 6, after each interval, a new event $\mathcal{E}$ occurs with probability $1/2$. In expectation after $O(\hat{T})$ intervals, $\mathcal{E}(\hat{T})$ occurs. Using the Chernoff bound, it is easy to show that at most after $O(\log n + \log R)$ of intervals, $\mathcal{E}(\hat{T})$ occurs w.h.p.. Noting that each interval contains constant unjammed rounds, and $\hat{T} \in O(\log n + \log R)$, we get the conclusion that after $\mathscr{T}(O(\log n + \log R))$ rounds, each non-empty cell has one active node elected as leader with probability of $1 - \frac{1}{n^c}$, c is a constant larger than 1. With at most n non-empty cells in network, summing the error probability of each non-empty cell, we prove lemma 1. ■

C. Analysis for Leader Connection Period

As we have assumed, at round t_{LEP}, each non-empty cell has one and only one leader elected, which will run the following pseudo-code in LCP. In the following rounds, leaders connect with each other. Even though there may be some previous connections before t_{LEP}, according to the update rules, all the previous connections will be replaced by the later one after t_{LEP}. After t_{LEP} round, since the leaders have already been fixed, so the connections will also not change. We use the following lemmas to prove that $\mathscr{T}(O(\log n))$ rounds after t_{LEP}, each leader connects with all other leaders within distance $(1+\frac{\epsilon}{2})R$ w.h.p..

Lemma 7: For any leaders u, v within distance $(1+\frac{\epsilon}{2})R$, during $\mathscr{T}(O(\log n))$ rounds, u receives message from v w.h.p..

Proof: The following three Claims prove the lemma.

Claim 1: During $\mathscr{T}(O(\log n))$ rounds, there are $O(\log n)$ unjammed rounds for any leader u.

According to the definition of $\mathscr{T}(O(\log n))$, Claim 1 holds.

Considering a circle centered at leader u and with radius of $(1+\frac{\epsilon}{2})R$, we first consider the interference I_u^t experienced by u at round t from leaders outside the circle, and if $I_u^t < [\frac{(1+\epsilon)^\alpha}{(1+\epsilon/2)^\alpha} - 1]N$, we say u is in a *good* round.

Claim 2: In $\mathscr{T}(O(\log n))$ rounds, there are $O(\log n)$ *good* rounds for leader u.

Proof: We divide the whole network into rings $\{C_j\}$ for $j \geq 1$, where C_j is the ring with distance in the range $[j(1+\frac{\epsilon}{2})R, (j+1)(1+\frac{\epsilon}{2})R)$ from u. Note that each non-empty square cell with size $aR \times aR$ has one leader, so the number of leaders s_j in each ring C_j can be bounded.

Let Y_x^t be the random variable for leader x with value 1 if x transmits at round t and 0 otherwise. Denote by I_{xu}^t the interference at leader u caused by leader x at round t, and T_1 be the set of unjammed rounds for u during $\mathscr{T}(O(\log n))$ rounds. Then, omitting the mathematical calculation, we get

$$s_j \leq \pi(2j+1)(1+\frac{\epsilon}{2})\frac{4\sqrt{2}a + 2 + \epsilon}{2a^2}$$

$$E[I_u^t] == \sum_{j=1}^{\infty} \sum_{x \in S_j} E[I_{xu}^t] = [\frac{(1+\epsilon)^\alpha}{(1+\epsilon/2)^\alpha} - 1]N/4,$$

Based on the above analysis and the Chernoff inequality, with high probability, $\sum_{t \in T_1} I_u^t \leq |T_1|E|I_u^t| * 2 \leq |T_1|[\frac{(1+\epsilon)^\alpha}{(1+\epsilon/2)^\alpha} - 1]N/2$, i.e., at least half of the rounds in T_1 are *good* rounds for u. Otherwise, the inequality will be violated. Combining that $|T_1| = O(\log n)$, we prove this claim. ■

Claim 3: For any leader u and v within distance $(1+\frac{\epsilon}{2})R$, if u is in a *good* round, then u receives message from v with a constant probability.

Proof: Considering the circle centered at u and with radius of $(1+\frac{\epsilon}{2})R$, the interference from leaders outside the circle in a *good* round is already bounded. Let s_0 be the number of leaders inside the circle, then according to an area argument:

$$s_0 \leq \frac{\pi[(1+\frac{\epsilon}{2})R + \sqrt{2}aR)]^2}{(aR)^2} = \pi(\frac{\epsilon+2}{2a} + \sqrt{2})^2 \qquad (2)$$

Thus, other than leaders u, v, there are at most another $s_0 - 2$ leaders. So, with probability $p_2 * (1-p_2)^{s_0-1}$ that event "v broadcasts and all other nodes in the circle listen" happens. When this event happens, according to the SINR model,

$$SINR(v,u) \geq \frac{\frac{P}{((1+\frac{\epsilon}{2})R)^\alpha}}{N + [\frac{(1+\epsilon)^\alpha}{(1+\epsilon/2)^\alpha} - 1]N} = \beta. \qquad (3)$$

Thus, u receives message from v. ■

Combining Claims 1, 2 and 3, and applying a Chernoff bound, Lemma 7 is proved. ■

At round t_{LEP}, each non-empty cell has a leader elected. According to Lemma 7 and our update rule, during the following $\mathscr{T}(O(\log n))$ rounds after t_{LEP}, each pair of leaders within distance $(1+\frac{\epsilon}{2})R$ can successfully receive message from each other and are connected. Then, we are going to prove that the connected leaders construct a CDS.

Proof for Lemma 2

Proof: According to Lemma 1, all non-empty cells with size $aR \times aR$ have their leaders elected, so the leaders can construct a dominating set for the network. We then prove that the dominating set is connected. The proof is by contradiction. Otherwise, w.l.o.g., assume that the communication graph obtained by connecting each pair of leaders within distance $(1+\frac{\epsilon}{2})R$ contains two connected subgraphs G_1 and G_2.

Let V_1 and V_2 be the set of nodes in G_1 and G_2 respectively. Denote by V_1' (V_2') the union of V_1 (V_2) and the set of other nodes within distance $\sqrt{2}aR$ from nodes in V_1 (V_2). By the connectivity assumption in model, there must be two nodes $u' \in V_1'$ and $v' \in V_2'$, such that $d(u,v) \leq R$. Then by Lemma 1, there are two nodes $u \in V_1$ and $v \in V_2$ such that $d(u,u') \leq \sqrt{2}aR \leq \frac{\epsilon}{4}R$ and $d(v,v') \leq \sqrt{2}aR \leq \frac{\epsilon}{4}R$. Then

$$d(u,v) \leq d(u,u') + d(u',v') + d(v,v') \leq (1+\frac{\epsilon}{2})R. \qquad (4)$$

This means that G_1 and G_2 are connected, which contradicts the assumption. This completes the proof for Lemma 2.

D. Analysis for backbone properties ■

As we have given is Sec. I, a backbone is a CDS with (1) constant degree; (2) constant approximate diameter; and (3) constant approximate size;

Proof for backbone property (1), (2), (3).

Proof: Note that each leader connects with other leaders within distance $(1+\frac{\epsilon}{2})R$. The degree of a leader u w.r.t. to other leaders in CDS, denoted as $degree_u$, is defined as the number of other leaders connected with u. For a circle centered at u and with radius $(1+\frac{\epsilon}{2})R$, according to the area argument in Eqt. 2, $degree_u \leq \pi(\frac{\epsilon+2}{2a} + \sqrt{2})^2$. So, each leader in CDS has a constant degree, and property (1) is proved.

For any two nodes u, v within distance R, assume that nodes v, u are in cells g_v and g_u with corresponding leaders v_1 and u_1 respectively. Since a cell has the size of $aR \times aR$, $d(v,v_1) \leq \sqrt{2}aR \leq \frac{\epsilon R}{4}$, $d(u,u_1) \leq \sqrt{2}aR \leq \frac{\epsilon R}{4}$. Then $d(v_1,u_1) \leq d(v_1,v) + d(v,u) + d(u,u_1) \leq (1+\frac{\epsilon}{2})R$. Nodes u, v are connected to their corresponding leaders, and their leaders are also connected according to Lemma 7. So u, v are connected within three hops via their leaders. Then, any one hop connection in the network can be replaced by at most a three-hop connection in the new communication graph. Thus, the diameter of the new communication graph is at most three times larger than that of the network, and property (2) is proved.

Assume that the minimum CDS in the network contains ϖ nodes. Then according to the definition of CDS, circles centered at the ϖ nodes and with radius of R cover all nodes in the network, including the leaders. According to an area argument, a single circle with radius R consists at most $\frac{\pi[R+\sqrt{2}aR]^2}{(aR)^2} = \pi(\frac{1}{a} + \sqrt{2})^2$ leaders in the constructed CDS. Thus, there are at most $\pi(\frac{1}{a} + \sqrt{2})^2\varpi$ leaders in our constructed CDS, i.e., the constant factor between our constructed CDS and the minimum CDS is at most $\pi(\frac{1}{a} + \sqrt{2})^2$. The property (3) is proved. ■

With the above proofs, Theorem 1 is proved.

VI. Necessity of Local Uniformity

Theorem 2: If the jamming violates the local-uniform assumption, i.e., the noise injected by the adversary can be different for nodes in the same cell, then to get a high probability performance guarantee, any randomized algorithm for backbone construction needs $\mathscr{T}(\omega(\log n))$ time.

Theorem 2 can be proved by constructing a network in a cell with n nodes, and applying different jamming schedules on different nodes, the detail of which is omitted.

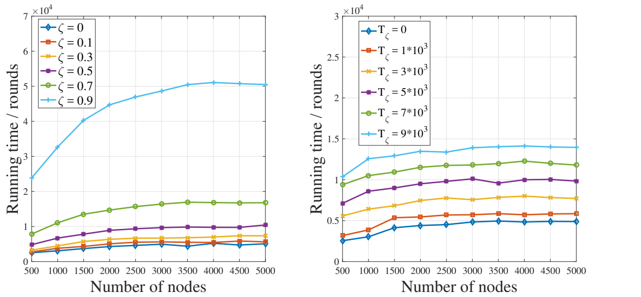
VII. Simulation Results

We investigate the empirical performances of our backbone construction algorithm in this section. Specifically, we investigate (i) the running time of the algorithm under different jamming patterns and different network sizes; and (ii) the properties of the constructed backbone, i.e., the degrees of nodes in the backbone, the diameter of new constructed communication graph and the size of backbone.

Jamming patterns. Similar to the simulation setting in [13], here we adopt two types of jamming patterns, Regular (or random) Jamming (REGJ) in which the adversary has a constant probability $\zeta \in [0,1)$ to jam the transmissions in each cell in every round, and Bursty Jamming (BURJ) in which the adversary randomly jams at most T_ζ rounds for each cell. The jamming parameters ζ in REGJ and T_ζ in BURJ reflect the jamming level, i.e., the larger ζ or T_ζ is, the heavier the jamming. The jamming in the cells is independently determined.

TABLE I: Parameters in simulation

Parameters	Value	Parameter	Value
n	$[500, 5000]$	ϵ	2
R	30m	p_1	0.05
p_2	0.002	α	3
ζ	$\{0, 1, 3, 5, 7, 9\} * 10^{-1}$	β	1.5
T_ζ	$\{0, 1, 3, 5, 7, 9\} * 10^3$		



(a). Running time in REGJ (b). Running time in BURJ

Fig. 1: Running time of our algorithm in two jamming patterns when network size and jamming level change.

Parameters. Basically, n nodes are randomly and uniformly distributed into a network area of size $300m \times 300m$. Each node has a uniform transmission range of $30m$. The setting of parameters is given in Table I. Over 20 runs of the simulation have been carried out for each reported result. All experiments are conducted on a Linux machine with Intel Xeon CPU E5-2670@2.60GHz and 64 GB main memory, implemented in C++ and compiled by the g++ compiler.

A. *Running time of our algorithm*

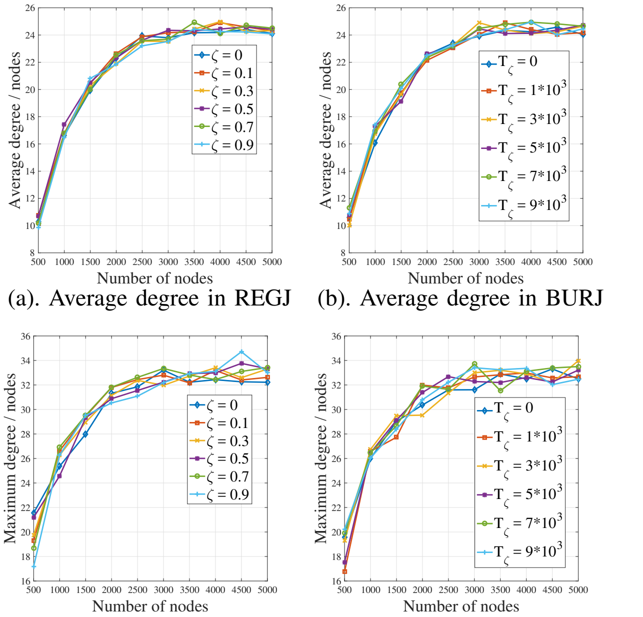
Under the two jamming patterns REGJ and BURJ, we investigate the running time of our backbone construction algorithm in reality when varying the network size and the jamming level. Specifically, in the simulation, the network size n varies from 500 to 5000. In regular jamming (REGJ), the jamming ratio is $\zeta \in \{0, 0.1, 0.3, 0.5, 0.7, 0.9\}$. In bursty jamming (BURJ), we have the jamming parameter $T_\zeta \in \{0, 1, 3, 5, 7, 9\} * 10^3$. The simulation results are illustrated in Fig. 1. In Fig. 1, the running times of our algorithm in the two jamming patterns when the number of nodes and jamming levels change are shown.

From Fig 1, we can find that as the number of nodes gets larger, the running time increases logarithmically. Also as the jamming ratio (number of jamming rounds) gets larger, the running time increases.

From the figures, it can also be found that our algorithm is insensitive to jamming in the sense that the number of unjammed rounds used by our algorithm is rarely affected by particular jamming in the network. Take Fig. 1(a) as an example. In the case of $n = 3000$, the running time is about $5 * 10^3$ rounds, which means our algorithm needs $5 * 10^3$ unjammed rounds to accomplish the task in a network without jamming. Then for the case with jamming ratio $\zeta = 0.9$, the running time is about $5 * 10^4$. In this case, there are $5 * 10^4 * (1 - \zeta) = 5 * 10^3$ unjammed rounds in expectation during the $5 * 10^4$ running time. So, the $4.5 * 10^4$ jamming rounds do not increase our algorithm's requirement for unjammed rounds and our algorithm mainly uses the remaining $5 * 10^3$ unjammed rounds to complete the task.

B. *Properties of constructed backbone*

We then verify the properties of the constructed backbone.



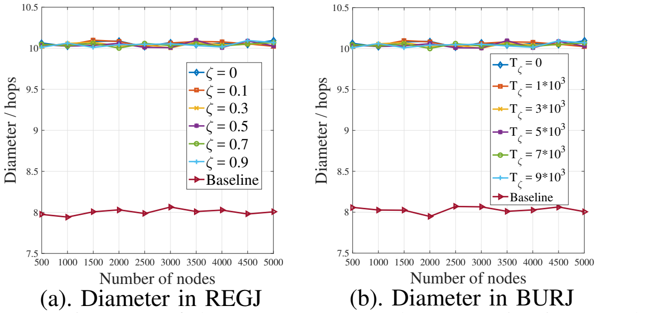
(a). Average degree in REGJ (b). Average degree in BURJ

(c). Maximum degree in REGJ (d). Maximum degree in BURJ

Fig. 2: Average/maximum degree of the backbone in two jamming patterns when network size and jamming level change.

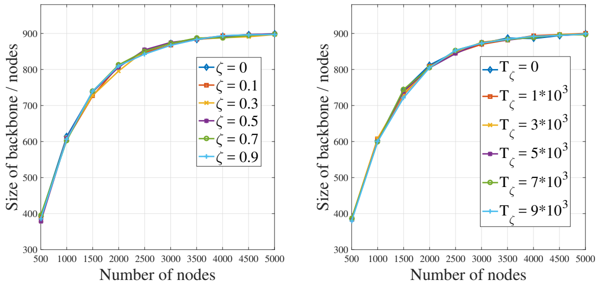
Degree of nodes in backbone. Fig. 2 illustrates the degrees of nodes in the constructed backbone for networks with different sizes and in different jamming situations. The x-axes in Fig. 2 represent the number of nodes in the network. The y-axes represent the average and maximum degree of nodes in the constructed backbone respectively in Fig. 2(a)(b) and Fig. 2(c)(d). From Fig. 2, we can see that (1) the average degree and maximum degree increase first when n gets larger and becomes stable subsequently; (2) the average degree and maximum degree of the constructed backbone are not large in reality, which are always smaller than 25 and 35 respectively; (3) the average (maximum) degrees are almost the same in different jamming situations. So, our constructed backbone is insensitive to jamming in terms of nodes' degrees.

Diameter of new communication graph. Fig. 3 shows the diameter of the new communication graph, which is



(a). Diameter in REGJ (b). Diameter in BURJ

Fig. 3: Diameter of the new constructed communication graph in two jamming patterns when network size and jamming level change.



(a). Size of backbone in REGJ (b). Size of backbone in BURJ

Fig. 4: Size change of the constructed backbone in two jamming patterns when network size and jamming level change.

constructed by connecting nodes outside the backbone with their designated one hop neighbors inside the backbone, in networks of different sizes and having jamming situations. In Fig. 3, we use the diameter of the original network as the baseline. From the figures in Fig. 3, we can see that (1) the diameter of the new communication graph is very close to the diameter of the original network, which is at most two hops longer; (2) the diameters of the new communication graph in different jamming levels are almost the same. In other words, the algorithm is insensitive to jamming in terms of the diameter of the newly constructed communication graph.

Size of backbone. Fig. 4 illustrates the size of the constructed backbone, i.e., the number of nodes in the backbone, when n and the jamming level change. In Fig. 4, the x-axes and y-axes represent the number of nodes in the network and the size of the constructed backbone respectively. In Fig. 4, the sizes of backbones constructed for the cases of REGJ and BURJ are nearly the same. When n is 500, since the network is too sparse, a large fraction of nodes are by nature needed for the backbone construction. When the network becomes larger, the ratio of backbone size and network size gradually becomes smaller. When n is 5000, the backbone consists of 900 nodes.

C. Summary

The simulated results show that our algorithm can construct a backbone efficiently in a jamming environment. Furthermore, when the number of unjammed rounds used to construct a backbone is nearly unchanged for different jamming levels, our backbone construction algorithm is insensitive to the jamming in the network in terms of the running time and the various properties of the constructed backbone.

VIII. Conclusion

We presented a strong adversarial jamming model to portray general jamming phenomena in wireless networks. The proposed model, featured by local-uniformity, unrestricted energy budget and reactivity, can cover more jamming scenarios and is closer to reality than models in previous work. Under the strong adversarial jamming model, an efficient jamming-resilient backbone construction algorithm was given which can construct a backbone in $\mathscr{T}(O(\log n + \log R))$ rounds with a high probability guarantee. Extensive simulations reveal the efficiency of our algorithm in realistic situations.

The strong adversarial jamming model and the protocol designed in this work provide a useful base for further work in jamming-resilient distributed protocol design and analysis. It will also be interesting to devise specific efficient algorithms for other fundamental problems, such as broadcast and aggregation, under the proposed jamming model.

IX. Acknowledgements

This work is supported by National Natural Science Foundation of China (61602195, 61672321, 61771289, 61832012, 61373027, 61702198, 61672257, 61433012, U1435215), Science and Technology planning project of ShenZhen (JCYJ20170818112550194), and Fund of Hubei Key Laboratory of Transportation Internet of Things (WHUTIOT-2017A0011).

References

[1] G. Alnifie and R. Simon. A multi-channel defense against jamming attacks in wireless sensor networks. In *Q2SWinet*, 2007.

[2] B. Awerbuch, A. Richa, and C. Scheideler. A jamming-resistant mac protocol for single-hop wireless networks. In *PODC*, 2008.

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

[4] T. Brown, J. James, A. Sethi. Jamming and sensing of encrypted wireless ad hoc networks. In *MobiHoc*, 2006.

[5] J. Chiang, Y.-C. Hu. Cross-layer jamming detection and mitigation in wireless broadcast networks. In *MobiCom*, 2007.

[6] B.S. Chlebus, D.R. Kowalski, and S. Vaya. Distributed communication in bare-bones wireless networks. In *CoRR*, abs/1510.07357, 2015.

[7] B.S. Chlebus and S. Vaya. Distributed communication in bare-bones wireless networks. In *ICDCN*, 2016.

[8] T. Jurdziński and D.R. Kowalski. Distributed backbone structure for algorithms in the sinr model of wireless networks. In *DISC*, 2012.

[9] T. Jurdziński, D.R. Kowalski, M. Rozanski and G. Stachowiak. On Setting-Up Asynchronous Ad Hoc Wireless Networks. In *INFOCOM*, 2015.

[10] X. Liu, G. Noubir, R. Sundaram, and S. Tan. Spread: Foiling smart jammers using multi-layer agility. In *INFOCOM*, 2007.

[11] W.K. Moses Jr. and S. Vaya. Deterministic protocols in the SINR model without knowledge of coordinates. In *CoRR*, abs/1702.02455, 2017.

[12] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein. Using channel hopping to increase 802.11 resilience to jamming attacks. In *INFOCOM*, 2007.

[13] A. Ogierman, A.W. Richa, C. Scheideler, S. Schmid, J. Zhang. Competitive MAC under adversarial SINR. In *INFOCOM* 2014.

[14] A. Ogierman, A.W. Richa, C. Scheideler, S. Schmid, J. Zhang: Sade: competitive MAC under adversarial SINR. In *Distributed Computing*, 31(3): 241-254, 2018.

[15] A. Pelc and D. Peleg. Feasibility and complexity of broadcasting with random transmission failures. In *PODC*, 2005.

[16] S.P. Reddy, D.R. Kowalski, and S. Vaya. Multi-broadcasting under the SINR model. In *CoRR*, abs/1504.01352, 2015.

[17] S.P. Reddy and S. Vaya. Brief announcement: Multi-broadcasting under the sinr model. In *PODC*, 2016.

[18] A. Richa, C. Scheideler, S. Schmid, J. Zhang. A jamming-resistant mac protocol for multi-hop wireless networks. In *DISC*, 2010.

[19] A. Richa, C. Scheideler, S. Schmid, J. Zhang. Competitive and fair medium access despite reactive jamming. In *ICDCS*, 2011.

[20] A. Richa, C. Scheideler, S. Schmid, J. Zhang. Self-stabilizing leader election for single-hop wireless networks despite jamming. In *MobiHoc*, 2011.

[21] A.W. Richa, C. Scheideler, S. Schmid, J. Zhang. Competitive throughput in multi-hop wireless networks despite adaptive jamming. In *Distributed Computing* 26(3): 159-171, 2013.

[22] J. Schneider, R. Wattenhofer: What Is the Use of Collision Detection (in Wireless Networks)? In *DISC*, 2010.

[23] M.K. Simon, J.K. Omura, R.A. Schultz, and B.K. Levin. Spread Spectrum Communications Handbook. In *McGraw-Hill*, 2001.

[24] A. Wood, J. Stankovic, and G. Zhou. DEEJAM: Defeating energyefficient jamming in IEEE 802.15.4-based wireless networks. In *SECON*, 2007.

[25] W. Xu, T. Wood, and Y. Zhang. Channel surfing and spatial retreats: defenses against wireless denial of service. In *Workshop on Wireless Security*, 2004.