Jamming-Resistant Collaborative Broadcast In Wireless Networks, Part I: Single-hop Networks

Liang Xiao
Xiamen University, China 361005
Email: lxiao@xmu.edu.cn

Huaiyu Dai
NC State University, Raleigh, NC 27695
Email: huaiyu_dai@ncsu.edu

Peng Ning
NC State University, Raleigh, NC 27695
Email: pning@ncsu.edu

Abstract—We propose a collaborative broadcast scheme for wireless networks, which is based on the Uncoordinated Frequency Hopping (UFH) technique and exploits the node cooperation to achieve higher communication efficiency and stronger jamming resistance. In the collaborative broadcast, nodes that already obtain the broadcast message help forward the message to other nodes. Relying on the sheer number of relay nodes, which grows with time surely, our scheme is fundamentally more powerful than most recent attempts for anti-jamming broadcast. Potential applications include emergency alert broadcast and distribution of key system information in the presence of jamming.

We provide three relay channel selection strategies for collaborative broadcast, analyze the corresponding successful packet reception rates for both synchronous and asynchronous scenarios, and present the corresponding cooperation gain. Simulation results in a practical setting show that our scheme significantly reduces broadcast delay and energy consumption against the most powerful jamming, – responsive-sweep jamming.

I. INTRODUCTION

Because of the broadcast nature of radio propagation, wireless networks are highly vulnerable to jamming attacks, where jammers aim at interrupting the ongoing legitimate information exchange by injecting replayed or faked signals into wireless media [1]. Jamming-resistant broadcast is important for many safety-critical applications such as emergency alert broadcast and navigation signal dissemination, and is critical for the distribution of important information such as the public key and system control information in wireless systems.

Jamming attacks are easily launched for wireless communications, and cannot be fully addressed through conventional cryptography. Spread spectrum techniques, including Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping (FH), have been commonly adopted to counteract jamming. One key vulnerability for these conventional anti-jamming techniques is the requirement of pre-shared secret keys (such as spreading codes in DSSS or frequency hopping pattern in FH) at the senders and legitimate receivers. This requirement suffers from scalability concerns, and may not even be feasible in the face of network dynamics and compromised receivers.

This problem was recognized recently, leading to a series of promising research efforts [2]–[8], including Uncoordinated FH (UFH) [2]–[5] techniques, Uncoordinated DSSS (UDSSS) [6], [7], and BBC [8]. In this paper, we further explore the UFH approach for jamming-resistant broadcast without pre-shared keys.

In UFH [2], a message is divided into multiple short packets, and each packet is transmitted over a randomly selected channel, independent of each other and only known to the sender. Such rapid channel switching over a large frequency range effectively thwarts the jamming attempts. On the down side, each packet has to be sent multiple times, due to the low communication efficiency resulting from the uncoordinated channel selection between the sender and the legitimate receivers. To this end, a BMA scheme was proposed to improve the communication efficiency by combining erasure coding [3]. Some additional efficient packet verification methods were proposed in [4].

In spite of all these efforts, the UFH-based techniques still suffer from low communication efficiency. The relative throughput of the original UFH compared with coordinated FH is only on the order of 10^{-3} for a spreading ratio of 200 [2], and the approaches in [3] only reduce the communication latency up to one-half. To address this problem, in [5], the USD-FH scheme was proposed to further improve the efficiency and robustness, where the hopping pattern is conveyed through UFH to allow message transmission through coordinated FH.

In this paper, we propose a Collaborative UFH-based Broadcast (CUB) scheme to achieve higher efficiency and stronger jamming resistance than existing anti-jamming broadcast schemes. The main idea is to allow the set of nodes that already receive the message to help broadcast, as all the nodes are expecting the same broadcast message. This process may start slowly, but as more and more nodes join the relaying, the broadcast accelerates much like an avalanche.

This scheme exploits the node cooperation to enhance both the efficiency and the security. Unless all the channels are simultaneously blocked (assumed impossible for a fairly large spreading ratio), it is always possible for some nodes to obtain the message through unjammed channels. These nodes then relay it across more channels to increase the success rate of reception. With time on its side, our scheme is fundamentally more powerful than most recent attempts for anti-jamming

broadcast. In addition, this approach is not restricted to UFH, and can be readily combined with other jamming counter measures.

In the first part of this paper, we study the collaborative broadcast in single-hop networks and explore the spectral diversity provided by the approach. In Part II [9], we extend this approach to multihop networks and provide protocol design solutions for multihop networks.

The remainder of the paper is organized as follows. In Section II we review related work, and in Section III we introduce the network model and jamming model. In Section IV, we propose the collaborative UFH-based broadcast scheme. We then give a detailed performance analysis for a snapshot scenario, for both synchronous and asynchronous cases, and provide the cooperation gain in Section V. Section VI further presents simulation results for our scheme in a more practical setting. Finally, we conclude in Section VII.

II. RELATED WORK

Jamming-resistant communication without pre-shared keys has been recently recognized as a crucial issue for wireless networks [2]–[12]. The BBC method in [8] was mainly devised for Ultra Wide Band (UWB) systems, where very short radio pulses with high power at specific slots are used as indelible marks. Meanwhile, various uncoordinated spread spectrum techniques, including Uncoordinated DSSS (UDSSS) and UFH, have also gained extensive research interests.

The UDSSS technique provides anti-jamming broadcast mechanism for DSSS-based systems, where each sender and receiver independently and randomly selects a spreading code sequence out of a large code set to transmit or receive [6]. Its randomized differential variation was proposed to improve the resistance against reactive jammers with strong computational power [7]. These techniques in general are still vulnerable to reactive jamming to some degree, with communication and computational overhead significantly higher than the UFH-based techniques.

The overview of the UFH technique has already been provided in the introduction and thus is omitted here. To the best of our knowledge, how to apply the UFH technique in a general multihop broadcast scenario is still an open problem. Efficient broadcast strategy has to be developed to resist jamming in wireless networks.

III. PROBLEM FORMULATION

A. Network Model

In this paper, we mainly consider a single-hop broadcast network, where all nodes are reachable in one hop. We assume that a source node intends to transmit a message to N identical randomly located nodes, utilizing the UFH mechanism to resist jamming. The broadcast message is divided into M short packets of the same length, each of which can be sent during one slot duration t_p. For simplicity, it is assumed that each node (including the source) sends or receives only one channel. Our results can be extended to the multi-channel cases and multihop networks as discussed in [9].

B. Jamming Model

As in [2], [3], we consider omniscient jammers with bounded computation and transmission capability, and focus on the jamming attacks, which have been shown more detrimental than other types of attacks (such as insertion and overwriting) [2]. It is assumed that a jammer needs to transmit on a channel for at least $t_{\bar{p}}$ time to effectively jam a packet.

The jamming attacks are usually categorized into two types: non-responsive and responsive. The difference between the two lies in whether a jammer attempts to detect the ongoing legitimate transmission before jamming. For non-responsive jamming, it is assumed that a jammer can block C_J channels simultaneously, and needs t_J time to switch the jamming channels. Since it takes only $t_{\bar{p}}$ time to jam a channel, it is in the interests of a jammer to jam as many channels as possible in each hop duration. The best she can achieve is $n_J C_J$, where $n_J = \frac{t_p}{t_{\bar{p}} + t_J}$, through the so-called sweep strategy [2]. For the responsive jamming, it is assumed that a jammer can sense C_s channels simultaneously, and needs t_s time to switch sensing channels. So in one hop duration, she can sense up to $n_s C_s$ channels, where $n_s = \frac{t_p - t_{\bar{p}} - t_J}{t_s}$, accounting for the time needed to launch an effective jamming once the transmission is detected.

In our study, we consider the most powerful jamming [2], [3], responsive-sweep, where a jammer conducts both non-responsive and responsive jamming independently and simultaneously. The corresponding jamming probability for a single source (without relay) depends on both the sensing probability and jamming probability,

$$p_J = \frac{n_s C_s + n_J C_J}{C}. \tag{1}$$

The analysis based on the responsive-sweep jamming provides a lower bound for the broadcast performance, and most conclusions can be easily adapted for other jamming types.

IV. COLLABORATIVE UFH-BASED BROADCAST

In this section, we propose a collaborative anti-jamming broadcast scheme based on UFH. There are C orthogonal channels, and the source randomly selects one of them for the transmission of each packet. The M packets are sent out sequentially and repeatedly. We exploit the node cooperation to improve both jamming resistance and broadcast efficiency, by allowing nodes that have successfully received the whole message to help relay it over multiple channels simultaneously. In the following, we discuss in more detail how relays may select the channels for rebroadcast, and how nodes that have yet to receive the message may adjust their receiving strategy. Due to the collaboration among the neighbor nodes, our scheme does not require the receivers switch channels at a much slower speed than the sender as in [2].

A. Relay Channel Selection

Assume there are ζ relay nodes in the network, we propose three strategies for relay channel selection as follows:

- Random Relay Channel selection (RRC): Similar to the source node, each relay node randomly and independently selects one channel out of the C channels for the transmission of each packet.
- Sweep Relay Channel selection (SwRC): The relays take non-overlapping channels for each packet transmission: the 1st relay node randomly selects one channel from C channels, and the ith relay node randomly chooses one out of the remaining $C - i + 1$ channels, for $1 < i \leq \zeta$.
- Static Relay Channel selection (StRC): The relays take fixed non-overlapping channels throughout the message broadcast process. For example, each relay may select a channel based on its (partial) ID (modulo some prime number) so that no overlapping is incurred.

Remark: The RRC strategy is amenable to distributed implement and has good scalability. One drawback of this strategy is that, sometimes some relay nodes (as well as the source node) may happen to select the same channel. This will lead to collision and failure of transmission; even with perfect synchronization and collaboration among the source and the relays such that the same packet is broadcast by all relays and the source at the same time, such overlap still leads to waste of energy and reduced opportunity. To evaluate the theoretical maximum of RRC, we also consider its idealized version SwRC, proposed to address the collision problem at the expense of communication overhead to accomplish such coordination.

In the StRC strategy, since a fixed set of relay channels are employed during the whole message broadcast process, it is reasonable to assume that the relay channels are (after some time) known to both the yet-to-inform receivers and jammers. At a first glance, this seems to be a dumb approach: the jammers will go ahead to jam these relay channels (even without sensing), so the hope of the receivers still lies in the UFH-based source node transmission. Actually this approach captures the essence of collaborative broadcast. As long as all the channels are not blocked simultaneously, the number of relays will increase with time. When the breaking point is reached so that the jammers can no longer block all relay channels, the communication efficiency will be boosted dramatically. As we will see soon, the statement holds even in the asynchronous relay case, due to the very low collision probability resulting from a large number of channels C. An alternative view is that, as the UFH-based source node already provides uncertainty in its channel selection to counteract jamming, the StRC strategy introduces certainty in the relay selection to improve the communication efficiency. Furthermore, this strategy is also easy to implement: it saves the efforts of channel switching, and requires little communication overhead for coordination. Note that for the StRC strategy, the source node could also avoid known relay channels to improve efficiency. In this study, we assume that the source node employs the same strategy for implementation simplicity and fair comparison.

B. Receiving Channel Selection

As in UFH, we assume that each receiver hops randomly and independently over the C channels, which is named Random Receiving Channel selection (RRxC). For the StRC relay strategy, we further propose an Adaptive Receiving Channel selection (ARxC) strategy. In ARxC, each receiver first sweeps (known) relay channels, one at a time, in a random order. If it finds out that all the relay channels are jammed, it switches to RRxC. Once it comes across a clear relay channel, the receiver restricts to the relay channels again. This process repeats.

In essence, a receiver first attempts to take advantage of the available relay channels. However, these relay channels are also known to the jammers, and it is definitely in the jammers' interest to first block them. In the face of such strong jamming, a receiver then switches back to RRxC. However, instead of continuously jamming the static relay channels, jammers may just spend enough energy to spoof the receivers away. We therefore allow a receiver to check for such scenarios and come back to relay channels.

V. PERFORMANCE ANALYSIS

In this section, we consider a snapshot scenario, and give some analytical results on the performance of our CUB scheme with different relay and receiving channel selection strategies. It is assumed that the source node always performs the UFH transmission over the C channels. We focus on one frequency hop duration (i.e., a time slot) in which the source node and exactly ζ relay nodes cooperate to broadcast a packet, in the presence of a responsive-sweep jammer[1] with jamming probability p_J. The following observation is useful for our following analysis.

Observation 5.1: The successful packet reception rate p_a, i.e., the probability that a packet is successfully received by some receiver determines the average overall broadcast delay.

Proof: Assume that M packets are broadcast to N identical and independent nodes. The probability for all the receivers to obtain the M packets during the first m rounds of transmission (of the packet sequence) is $(1 - (1 - p_a)^m)^{MN}$. Following the analysis in [2], the average broadcast delay in the number of frequency hops is

$$T_{avg} = M \sum_{m=0}^{\infty} \left[1 - (1 - (1 - p_a)^m)^{MN} \right], \qquad (2)$$

which monotonically decreases with p_a for the given setting. ∎

This observation holds for a constant successful packet reception rate p_a, i.e., the number of transmitters including the source node and relays is fixed. As we will see, in practice, the number of relays increases with time, and thus p_a also increases with it, significantly reducing the average overall broadcast delay.

Therefore, we can evaluate the broadcast performance through p_a, a function of jamming parameters and ζ. The

---

[1]For static relay channels, a sweep jammer is enough.

jamming-free broadcast performance can be simply derived by letting $p_J = 0$ in the following results. The benchmark performance without cooperative relay is also easily obtained by taking $\zeta = 0$.

A. Synchronous relay

We start with an ideal scenario where all $\zeta + 1$ transmitters are perfectly synchronized in the sense that the same packet is sent in the same hop. Hence there is no confliction for the packets transmitted on the same channel.

Lemma 5.2: With RRC and RRxC strategies, the successful packet reception rate is given by

$$p_a^{RRC} = \left(1 - \left(1 - \frac{1}{C}\right)^{\zeta+1}\right)(1 - p_J). \qquad (3)$$

Proof: The source node and the relay nodes send the same packet over channels randomly and independently chosen from C channels. The receiver can obtain the packet, if working on a channel that is clear from jamming with probability of $1 - p_J$, and is selected by at least one of the $\zeta + 1$ transmitters (with probability of $1 - \left(1 - \frac{1}{C}\right)^{\zeta+1}$). ∎

Lemma 5.3: With SwRC and RRxC strategies, the successful packet reception rate is given by

$$p_a^{SwRC} = \left(1 - \left(1 - \frac{1}{C}\right)\left(1 - \frac{\zeta}{C}\right)\right)(1 - p_J). \qquad (4)$$

Proof: The difference with the RRC strategy is that relay nodes work over a set of ζ non-overlapping channels randomly selected out of C channels. ∎

Lemma 5.4: With StRC and RRxC strategies, the successful packet reception rate is given by

$$p_a^{StRC,RRxC} = \begin{cases} \frac{1}{C}\left(\zeta - n_J C_J + 1 - \frac{\zeta}{C}\right), & n_J C_J < \zeta \\ \frac{1-p_J}{C}, & n_J C_J \geq \zeta. \end{cases} \qquad (5)$$

Proof: When knowing the relay nodes perform the StRC strategy, the jammer first blocks as many relay channels as possible, and continues to attack the non-relay channels to prevent the transmission from the source, if at all possible. Note that in the effort to block the static known relay channels, the sensing capability of the jammer does not help; it is solely determined by the jamming capability (i.e. the maximal transmission power of the jammer). Therefore, the turning point happens at $\zeta = n_J C_J$.

In the case $n_J C_J \geq \zeta$, all relay channels are blocked. They can also jam $p_J C - \zeta$ non-relay channels, and the channel sensing function does help. In this case, each receiver randomly selects one out of C channels, and can get the packet if that channel is a non-relay channel selected by the source node and free from jamming. Hence, $p_a = (1 - \frac{\zeta}{C})\frac{1}{C}(1 - p'_J)$, where p'_J is the probability that a non-relay channel is jammed. p'_J can be easily obtained by replacing $n_J C_J$ with $n_J C_J - \zeta$ and C with $C - \zeta$ in (1), and the result in (5) follows after simple calculation.

In the case $n_J C_J < \zeta$, all the non-relay channels and $\zeta - n_J C_J$ relay channels are clear from jamming. In this case,

the receiver can obtain the packet, if listening to either an unblocked relay channel, or a non-relay channel selected by the source node. Thus,

$$\begin{aligned} p_a &= \frac{\zeta}{C}\left(1 - \frac{n_J C_J}{\zeta}\right) + \left(1 - \frac{\zeta}{C}\right)\frac{1}{C} \\ &= \frac{1}{C}\left(\zeta - n_J C_J + 1 - \frac{\zeta}{C}\right). \end{aligned}$$

∎

Lemma 5.5: With StRC and ARxC strategies, the successful packet reception rate is given by

$$p_a^{StRC,ARxC} = \begin{cases} 1 - \frac{n_J C_J}{\zeta}, & n_J C_J < \zeta \\ \frac{1-p_J}{C}, & n_J C_J \geq \zeta. \end{cases} \qquad (6)$$

Proof: In the case $n_J C_J \geq \zeta$, ARxC and RRxC strategies behave the same. So let us focus on the remaining case $n_J C_J < \zeta$. Since the operations of the jammer and a receiver (adopting the ARxC strategy) are independent, the successful packet reception rate is equivalent to the probability that a randomly selected relay channel is unblocked: $1 - \frac{n_J C_J}{\zeta}$. ∎

For given jamming parameters, as the number of relay nodes rises from $n_J C_J$ to $n_J C_J + 1$, $p_a^{StRC,ARxC}$ increases from $\frac{1-p_J}{C}$ to $\frac{1}{n_J C_J + 1}$ [2]. In addition, it is clear that for the case $n_J C_J < \zeta$, $p_a^{StRC,ARxC}$ is greater than $p_a^{StRC,RRxC}$, as C is a large number. Hence the ARxC strategy has to be applied in company with StRC to exploit the benefits of the latter.

B. Asynchronous Relay

It is nontrivial to realize ideal synchronization in practical wireless networks. Hence we further consider the asynchronous scenario, where two transmissions on the same channel will interfere with each other and neither can go through. First, in the RRC strategy, a receiver can successfully obtain a packet, when listening to a channel selected by exactly one legitimate transmitter, either the source node or a relay node. Therefore,

$$p_a^{RRCAsyn} = (\zeta + 1)\frac{1}{C}\left(1 - \frac{1}{C}\right)^{\zeta}(1 - p_J). \qquad (7)$$

Next, for the case of SwRC, the transmission is successful, if either the source node or one of the relay nodes chooses the receiving channel. As the relay channels in SwRC are non-overlapping, we have

$$p_a^{SwRCAsyn} = \left(\frac{1}{C}\left(1 - \frac{\zeta}{C}\right) + \left(1 - \frac{1}{C}\right)\frac{\zeta}{C}\right)(1 - p_J). \qquad (8)$$

Finally, in the StRC strategy, ζ relay nodes always send signal on fixed non-overlapping relay channels, and never interfere with the source node over the non-relay channels, we have $p_a^{StRCAsyn} = p_a^{StRC} = \frac{1-p_J}{C}$ for $n_J C_J \geq \zeta$. On the other hand, when $n_J C_J < \zeta$, the successful packet may be influenced when the source node interferes with one of the

[2] $n_J C_J < C$ by our assumption

spare relay channels. For the receiver with the RRxC strategy, we have

$$p_{RRxC,a}^{StRCAsyn} = \frac{\zeta}{C}(1 - \frac{n_J C_J}{\zeta})(1 - \frac{1}{C}) + (1 - \frac{\zeta}{C})\frac{1}{C}. \quad (9)$$

And for the receiver with the ARxC strategy, we have

$$p_{ARxC,a}^{StRCAsyn} = (1 - \frac{n_J C_J}{\zeta})(1 - \frac{1}{C}). \quad (10)$$

It is clear through comparison with the results in Section IV.A that, when C is sufficiently large, p_a^{Asyn} only slightly degrades with respect to the synchronous counterpart p_a in all these cases. This means that the collaborative broadcast is robust against relay synchronization error. Therefore we will focus on the synchronous case in the following discussion for simplicity, while most results can be easily extended to the asynchronous case.

C. Cooperation Gain

It is apparent from Eq. (3)-(10) that the successful packet reception rate of the collaborative broadcast increases with the number of relay nodes ζ. We can define the cooperation gain with ζ relay nodes, as $G(\zeta) \triangleq p_a(\zeta)/p_a(0)$. For RRC and SwRC, by Eq. (3) and (4), the cooperation gains can be approximated by

$$G^{RRC}(\zeta) \approx G^{SwRC}(\zeta) \approx \zeta + 1, \quad (11)$$

when C is sufficiently large.

For the case of StRC, $p_a(0) = (1 - p_J)/C$. By Eq. (6), the cooperation gain for the case of the receiver with ARxC strategy is

$$G_{ARxC}^{StRC}(\zeta) = \begin{cases} \frac{C}{1-p_J}\left(1 - \frac{n_J C_J}{\zeta}\right), & n_J C_J < \zeta \\ 1, & n_J C_J \geq \zeta. \end{cases} \quad (12)$$
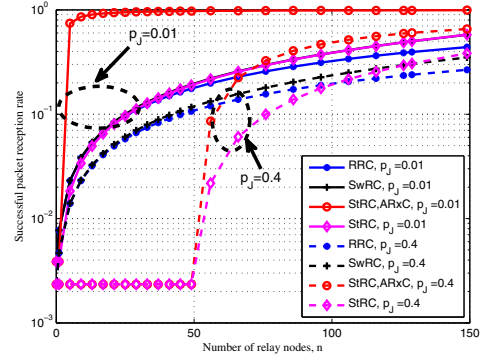
Similarly, by Eq. (5), the cooperation gain for the receiver using RRxC is given by

$$G_{RRxC}^{StRC}(\zeta) = \begin{cases} \frac{1}{1-p_J}\left(\zeta - n_J C_J + 1 - \frac{\zeta}{C}\right), & n_J C_J < \zeta \\ 1, & n_J C_J \geq \zeta. \end{cases}$$
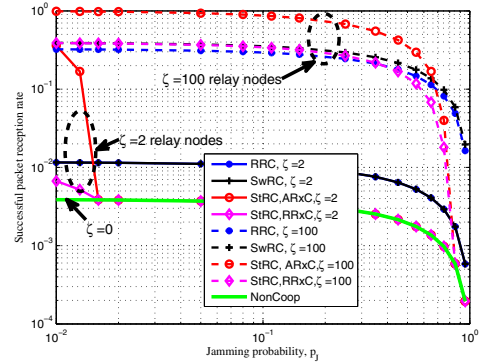
In contrast to RRC and SwRC where the cooperation gain grows roughly linearly with the number of relays, the cooperation gain for the StRC strategy is dichotomous: below the threshold $n_J C_J$, there is no cooperation gain; once the number of relay nodes passes the threshold, the cooperation rises dramatically, especially for the ARxC receivers. For instance, the cooperation gain with 40 relay nodes is 115.2 for the adaptive receiver, approximately 2.9 times greater than RRC or SwRC; meanwhile, it is as small as 19 for the RRxC receiver, for given $C = 256$, $p_J=0.2$ and $n_J C_J = n_s C_s$.

D. Numerical Evaluation

Some numerical evaluations of Eq. (3)-(6) are provided in Fig. 1 to better illustrate the properties of these three types of collaborative broadcast. It is assumed that $n_s C_s = n_J C_J$ for the responsive-sweep jamming. As shown in Fig. 1 (a), the cooperation gain with RRC or SwRC is approximately

(a) $p_J = 0.01$ or $p_J = 0.4$

(b) $\zeta = 2$ or 100 relay nodes

Fig. 1. The probability for a node with either the default RRxC strategy or the ARxC strategy (only for StRC) to successfully receive a packet during a time slot, in the synchronous collaborative broadcast using the RRC, SwRC or StRC relay strategies over $C = 256$ channels, against the responsive-sweep jamming with jamming probability p_J.

proportional to ζ. It is also found that SwRC only slightly outperforms RRC. Intuitively, the probability of channel conflict is negligible for a large C. Therefore RRC is generally preferred over SwRC in practice. The advantage of StRC is obvious under weak jamming or large-scale cooperation, while its drawback is also obvious in the face of strong jamming. Furthermore, it is indicated by Fig. 1 (b) that the performance of both RRC and SwRC degrades graciously with the jamming strength. These results conform to our previous analysis.

VI. SIMULATION RESULTS

The previous section provides analytical results for a simplified "snapshot" scenario with a fixed number of relay nodes. However, in a practical collaborative broadcast, the number of relay nodes increases from zero at the beginning to around $N - 1$ in the end. As it is challenging to derive exact closed-form formulas for the performance of the practical dynamic collaborative broadcast process, we resort to simulations to evaluate the performance and compare our CUB scheme with the Noncooperative UFH-based Broadcast (NUB) scheme. In the simulation, the source node broadcasted $M = 7$ packets to N nodes over $C = 128$ channels, against J responsive-sweep jammers with $t_s = 10\mu s$, $t_J = 15\mu s$, $C_s = 1$, $C_J = 1$,
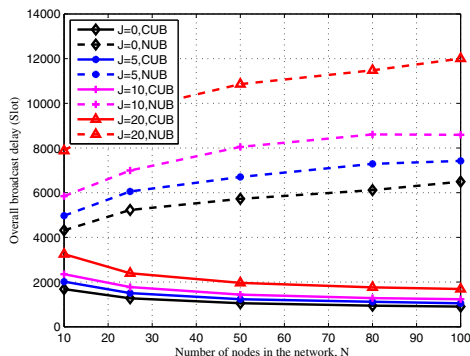
Fig. 2. Broadcast delay of the collaborative broadcast (CUB) and noncooperative broadcast (NUB), as a source node sending $M = 7$ packets to N nodes over $C = 128$ channels in presence of J responsive-sweep jammers, with $C_J = 1$, $C_s = 1$, $t_p = 40\mu s$, $t_{\bar{p}} = 5\mu s$, $t_J = 15\mu s$ and $t_s = 10\mu s$.

$t_p = 40\mu s$, and $t_{\bar{p}} = 5\mu s$ (thus $n_s = n_J = 2)^3$.

The advantage of CUB over NUB in terms of the average broadcast delay (from the beginning of broadcast till the time when all nodes in the network successfully receive the whole message) is clearly demonstrated in Fig. 2, where the average of 200 independent simulations is presented. For example, CUB takes only 14% of time to complete the broadcast, with $J = 20$ jammers and $N = 100$ nodes. Moreover, Fig. 2 shows an interesting phenomenon that as the network size N increases, NUS requires *longer* broadcast latency while CUB actually takes *shorter* delay. Hence, the saving in broadcast time by replacing NUB with CUB is more prominent in a larger network. Take the case with $J = 20$ jammers as an example. CUB can save 59% of the broadcast latency needed for NUS, with $N = 10$ nodes, which increases to 86%, as N rises to 100.

Let us take a closer look at the fact that CUB requires shorter latency as N rises. For example, with $J = 20$ jammers, the average broadcast delays are 3248 slots for $N = 10$ and 1692 slots for $N = 100$. Two main reasons account for this seemingly counter-intuitive phenomenon. First, in a larger network, the probability that a packet is received by at least one receiver is higher. In the case of one source and N independent receivers, this probability is given by $1 - (1 - 1/C)^N$, which goes from $1/C$ to 1 as N goes to infinity. In the above example, it takes on average only 1404 slots for the first 10 nodes of the 100-node network to receive the message. Second, as already mentioned, nodes that already receive the message turn into relays to continue their contribution in another domain.

VII. CONCLUSION

We have analyzed the anti-jamming broadcast in wireless networks, and proposed a collaborative broadcast scheme that

is based on uncoordinated frequency hopping and node cooperation. This scheme utilizes the collaborative communication technique and exploits frequency (channel) diversity to resist jamming and enhance communication efficiency. We have presented three relay channel selection strategies, RRC, SwRC and StRC for the collaborative broadcast, and analyzed their cooperation gain in terms of the successful packet reception rate in single-hop wireless networks. Both RRC and SwRC provide a cooperation gain proportional to the number of relay nodes, between which RRC is generally preferred due to its amenability to simple distributed implementation. The StRC strategy substantially further improves the cooperation gain under weak jamming relative to the collaboration scale. In addition, the proposed collaborative broadcast scheme has been shown to be robust against relay timing error.

Simulation results show that the collaborative broadcast is robust against responsive-sweep jammers, the most powerful jamming type in the UFH system. Compared with the non-collaborative UFH-based broadcast scheme, the proposed scheme takes only 14% of time to complete the broadcast, with 20 jammers in a 100-node wireless network. The average broadcast delay can be even less for the single-hop network with a larger size, due to increased multiuser diversity and spectral diversity. In the second part [9], we extend the study to the scenario of multihop networks and explore the spatial diversity that is provided by the approach.

REFERENCES

[1] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proc. MobiHoc*, 2005.
[2] M. Strasser, S. Capkun, C. Popper, and M. Cagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," in *Proc. IEEE symposium on security and privacy*, 2008.
[3] M. Strasser, C. Popper, and S. Capkun, "Efficient uncoordinated FHSS anti-jamming communication," in *Proc. MobiHoc*, 2009.
[4] D. Slater, P. Tague, R. Poovendran, and B. Matt, "A coding-theoretic approach for efficient message verification over insecure channels," in *Proc. ACM Conference on Wireless Network Security (WiSec'09)*, 2009.
[5] A. Liu, P. Ning, H. Dai, Y. Liu, and C. Wang, "USD-FH: Jamming-resistant wireless communication using frequency hopping with unco-ordinated seed disclosure," in *Proc. 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, 2010.
[6] C. Popper, M. Strasser, and S. Capkun, "Jamming-resistant broadcast communication without shared key," in *Proc. USENIX Security Symposium*, 2009.
[7] Y. Liu, P. Ning, H. Dai, and A. Liu, "Randomized differential DSSS: jamming-resistant wireless broadcast communication," in *Proc. IEEE Infocom*, 2010.
[8] L. Baird, W. Bahn, M. Collins, M. Carlisle, and S. Butler, "Keyless jam resistance," in *Proc. IEEE Information Assurance and Security Workshop*, 2007, pp. 143–150.
[9] L. Xiao, H. Dai, and N. Peng, "Jamming-resistant collaborative broadcast in wireless networks, part II: Multihop networks," in *Proc. IEEE Globecom 2011, to appear*.
[10] T. Jin, G. Noubir, and B. Thapa, "Zero pre-shared secret key establishment in the presence of jammers," in *Proc. ACM international symposium on Mobile ad hoc networking and computing*, 2009.
[11] L. Lazos amd S. Liu and M. Krunz, "Mitigating control-channel jamming attacks in multi-channel ad hoc networks," in *Proc. ACM WiSec*, 2009.
[12] A. Liu, P. Ning, H. Dai, Y. Liu, and C. Wang, "Defending DSSS-based broadcast communication against insider jammers via delayed seed-disclosure," in *Proc. IEEE Annual Computer Security Applications Conference (ACSAC)*, 2010.

---

[3] It is equivalent to a powerful jammer with combined capability of $C_S = J$ and $C_J = J$. Our simulation may represent a scenario where normal nodes are compromised.