Jamming Sensor Networks: Attack and Defense Strategies

Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang, Rutgers University

Abstract

Wireless sensor networks are built upon a shared medium that makes it easy for adversaries to conduct radio interference, or jamming, attacks that effectively cause a denial of service of either transmission or reception functionalities. These attacks can easily be accomplished by an adversary by either bypassing MAC-layer protocols or emitting a radio signal targeted at jamming a particular channel. In this article we survey different jamming attacks that may be employed against a sensor network. In order to cope with the problem of jamming, we discuss a two-phase strategy involving the diagnosis of the attack, followed by a suitable defense strategy. We highlight the challenges associated with detecting jamming. To cope with jamming, we propose two different but complementary approaches. One approach is to simply retreat from the interferer, which may be accomplished by either spectral evasion (channel surfing) or spatial evasion (spatial retreats). The second approach aims to compete more actively with the interferer by adjusting resources, such as power levels and communication coding, to achieve communication in the presence of the jammer.

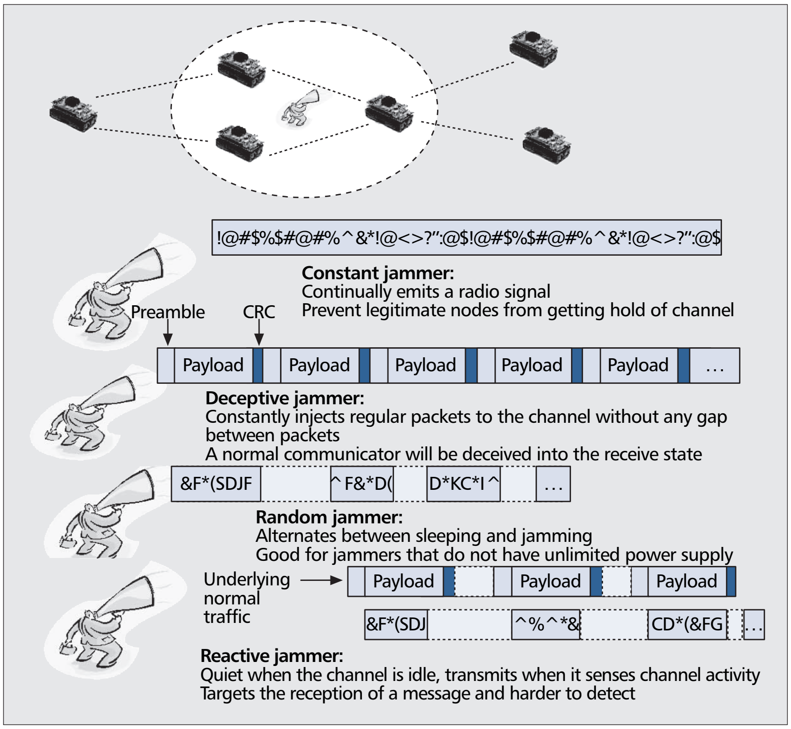
Securing sensor networks is a challenging task due to the limited resources associated with low-cost sensor hardware. The combination of the commodity nature of wireless technologies and an increasingly sophisticated user base means that adversaries are able to easily gain access to communications between sensor devices by purchasing their own device and running it in a monitor mode. Conventional cryptographic security mechanisms are being translated to the sensor domain in order to defend against attacks like packet injection and spoofing network-level control information. However, in spite of the progress being made to apply network security in the sensor realm, sensor networks will remain vulnerable to attacks that target their use of the wireless medium.

The wireless medium allows for radio interference attacks that target communications. Unlike traditional denial of service attacks, which are concerned with filling user domain and kernel domain buffers, jamming attacks exploit the shared nature of the wireless medium in order to prevent devices from communicating or receiving. Such attacks on the physical (PHY) layer have been known by the communications and radar community for some time, and there are numerous texts, such as [1, 2], which discuss the issues associated with these attacks. Typically, in the context of traditional communication systems, the objective of the jammer is to deny the reception of communications at the receiver using as little power as possible. In these systems jamming is usually addressed through spreading techniques, whereby resilience to interference is achieved by transmitting information using a bandwidth much larger than its required minimum bandwidth. Often, this spreading is also used to achieve multiple access, as in code-division multiple access (CDMA) cellular systems.

With the exception of some military systems, most commodity sensor and wireless networks do not employ sufficiently strong spreading techniques to survive jamming or to achieve multiple access. Instead, for reasons of cost, systems like the Berkeley MICA2, the Zigbee (e.g., MICAZ), and even 802.11 are based on a carrier sensing approach to multiple access. Because of their use of carrier sensing for medium access control (MAC), these systems are susceptible to a simple and severe jamming problem: an adversary can simply disregard the medium access protocol and continually transmit on a wireless channel. By doing so, he or she either prevents users from being able to commence with legitimate MAC operations, or introduces packet collisions that force repeated backoffs, or even jams transmissions. Such MAC and PHY layer security threats for wireless networks have been revisited recently by the Australian CERT [3], and will be a critical vulnerability for wireless sensor networks.

In this article we survey issues related to jamming sensor networks by examining both the attack and defend sides of the problem. We present different jamming attack strategies that might be used against sensor networks. Later, we examine methods that can be employed by the sensor network in order to detect the presence of jamming. We illustrate that basic statistics alone (e.g., signal strength) are not sufficient for classifying the presence of a jammer, and more advanced detection methods are needed. We examine two strategies for coping with jamming. The first strategy involves avoiding the jammer in either the spectral or spatial sense, and can be achieved by changing channel allocations or, in mobile sensor networks, by moving nodes away from the jammer. The second strategy involves competing with the jammer by adjusting the transmission power levels and employing error correction in order to have more resilience against jamming. Finally, we present concluding remarks.

0890-8044/06/$20.00 © 2006 IEEE


!@#$%$#@#%^&*!@<>?":@$!@#$%$#@#%^&*!@<>?":@$
Constant jammer:
Continually emits a radio signal
Prevent legitimate nodes from getting hold of channel
Preamble
CRC
Payload
Payload
Payload
Payload
Payload
...
Deceptive jammer:
Constantly injects regular packets to the channel without any gap between packets
A normal communicator will be deceived into the receive state
&F*(SDJF
^F&*D(
D*KC*I^
...
Random jammer:
Alternates between sleeping and jamming
Good for jammers that do not have unlimited power supply
Underlying normal traffic
Payload
Payload
Payload
&F*(SDJ
^%^*&
CD*(&FG
...
Reactive jammer:
Quiet when the channel is idle, transmits when it senses channel activity
Targets the reception of a message and harder to detect

■ Figure 1. *Jamming attacks target a sensor's ability to transmit or receive packets. Different jamming models accomplish the objective of blocking communications through different strategies.*

Jamming Attacks

There are many different attack strategies an adversary can use to jam wireless communications [4–6], as depicted in Fig. 1. While it is impractical to cover all the possible attack models that might exist, in this article we review a wide range of jammers that have proven to be effective.

Constant jammer: The constant jammer continually emits a radio signal, and can be implemented using either a waveform generator that continuously sends a radio signal [7] or a normal wireless device that continuously sends out random bits to the channel without following any MAC-layer etiquette [4]. Normally, the underlying MAC protocol allows legitimate nodes to send out packets only if the channel is idle. Thus, a constant jammer can effectively prevent legitimate traffic sources from getting hold of a channel and sending packets.

Deceptive jammer: Instead of sending out random bits, the deceptive jammer constantly injects regular packets to the channel without any gap between subsequent packet transmissions. As a result, a normal communicator will be deceived into believing there is a legitimate packet and be duped to remain in the receive state. For example, in TinyOS, if a preamble is detected, a node remains in the receive mode, regardless of whether that node has a packet to send or not. Even if a node has packets to send, it cannot switch to the send state because a constant stream of incoming packets will be detected.

Random jammer: Instead of continuously sending out a radio signal, a random jammer alternates between sleeping and jamming. Specifically, after jamming for a while, it turns off its radio and enters a "sleeping" mode. It will resume jamming after sleeping for some time. During its jamming phase, it can behave like either a constant jammer or a deceptive jammer. This jammer model tries to take energy conservation into consideration, which is especially important for those jammers that do not have unlimited power supply.

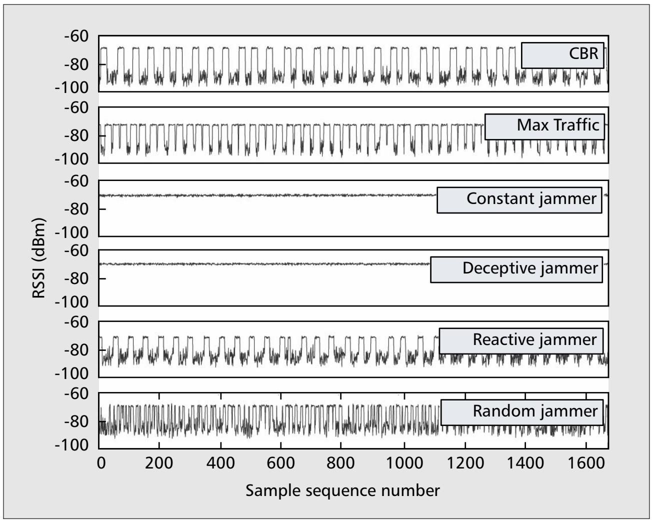
Reactive jammer: The three models discussed above are active jammers in the sense that they try to block the channel irrespective of the traffic pattern on the channel. Active jammers are usually effective because they keep the channel busy all the time. As we shall see in the following section, these methods are relatively easy to detect. An alternative approach to jamming wireless communication is to employ a reactive strategy. The reactive jammer stays quiet when the channel is idle, but starts transmitting a radio signal as soon as it senses activity on the channel. One advantage of a reactive jammer is that it is harder to detect.

In our work [4] we implemented the above four jammer models using Berkeley Motes that employ a ChipCon CC1000 RF transceiver and use TinyOS as the operating system. We bypassed the MAC protocol, so the jammer can blast on the channel irrespective of other activities taking place. We observed that the level of interference a jammer causes is governed by several factors, such as the distance between the jammer and a normal wireless node, the relative transmission power of the jammer and normal nodes, and the MAC protocol employed by normal nodes. The MAC protocol decides whether the channel is idle if the measured signal strength value is lower than a threshold. Many MAC protocols, such as the one in TinyOS release 1.1.1, use a fixed threshold value, while others, such as BMAC [8], adapt the threshold value based on the measured signal strength values when a channel is idle. As a result, these two different categories of MAC protocols decide the channel is jammed differently. We briefly summarize the results of our experiments. We had three parties, A, B, and X, where A and B are normal wireless nodes with A being the sender, B the receiver, and X is a jammer using one of our four models. More details of the experimental setup can be found in [4].

A jammer can interfere with normal communications between two legitimate communicators in two ways: preventing the sender from sending out packets, or preventing the receiver from receiving packets. Hence, we use the resulting packet send ratio (PSR) and packet delivery ratio (PDR) to measure the effectiveness of a jammer. Our experiments showed that all four jammers are quite effective in interfering with normal communications. The constant jammer can successfully prevent a node from sending out packets if that node employs a MAC protocol with a fixed threshold. Irrespective of the MAC protocol, the PDR is very poor, because the packets that manage to get sent out (e.g., where BMAC is employed) are corrupted anyway. The deceptive jammer, on the other hand, can completely block the send operations in any case because the sender will be forced to stay in receive mode all the time. The random jammer alternates between sleeping and jamming. While sleeping, the network operations will be normal; while jamming, it behaves just like a constant or deceptive jammer, depending on in which mode it operates. Finally, the reactive jammer does not affect the send ratio, but all the packets are corrupted, resulting in a zero PDR.

Detecting Jamming Attacks in Sensor Networks

Detecting jamming attacks is important because it is the first step toward building a secure and dependable wireless network. *Detecting radio interference attacks is challenging as it involves discriminating between legitimate and adversarial causes*



■ Figure 2. *RSSI readings as a function of time in different scenarios. RSSI values were sampled every 1 ms.*

of poor connectivity. In particular, legitimate scenarios for poor connectivity, such as congestion and device failures, may be difficult to differentiate from jamming.

There are several statistics that naturally lend themselves to detecting jamming, such as signal strength, carrier sensing time, and packet delivery ration. We will look at these different measurements and discuss how they are not effective in detecting a jamming attack. In order to repair the ability to detect a jamming attack, more sophisticated methods are needed, and we will discuss one possibility involving multimodal methods.

Basic Statistics

Signal Strength — One natural measurement that can be employed to detect jamming is signal strength. The rationale behind using this measurement is that the signal strength distribution may be affected by the presence of a jammer. Two natural approaches to detecting jamming using signal strength involve comparing average signal magnitude vs. a threshold calculated from the ambient noise levels, and classifying the *shape* of a window of signal samples.

In order to illustrate the effect a jammer would have on the received signal strength, we present results of several experiments conducted with the MICA2 Mote platform in Fig. 2. These experiments are described in more detail in [4]. In the first two experiments we have two Motes, a sender A and a receiver B, which are 30 in apart. The top two plots correspond to normal, or benign, traffic scenarios where the source A transmits at a constant bit rate (CBR) of 5.28 kb/s, while the second plot corresponds to A transmitting at its maximum send rate, a raw traffic rate of 6.46 kb/s. The bottom four plots correspond to four different jamming scenarios in which we introduced a jammer. Throughout these four jamming scenarios, A is a CBR source. Looking at raw time series data, it is clear that any statistic solely based on a moving average of the RSSI values would be hard pressed to discriminate between a normal traffic scenario and a reactive jammer scenario. Furthermore, the shape of the RSSI time series for normal traffic scenarios and the reactive jammer are too similar to rely on spectral discrimination techniques for discrimination. Further analysis of these methods and the difficulties associated with using signal strength readings may be found in [4]. Overall, these results suggest the following important observation: we may not be able to use simple statistics, such as average signal strength or energy, to discriminate jamming scenarios from normal traffic scenarios because it is not straightforward to devise a threshold that can separate these two scenarios.

Carrier Sensing Time — A jammer can prevent a legitimate source from sending out packets because the channel might appear constantly busy to the source, and hence it might seem possible to use carrier sensing time as a means to determine whether a device is jammed. In [4] the authors explored this possibility. We observed that using carrier sensing time is suitable when the following two conditions are true: the jammer is non-reactive or non-random, and the underlying MAC protocol determines whether a channel is idle by comparing the noise level with a fixed threshold. If these two conditions are true, carrier sensing time is an efficient way to discriminate a jammed scenario from a normal ill-functioning scenario, such as congestion, because the sensing time will be bounded, although large, in a congested situation, but unbounded in a jammed situation. Overall, carrier sensing time alone cannot be used to detect all the jamming scenarios.

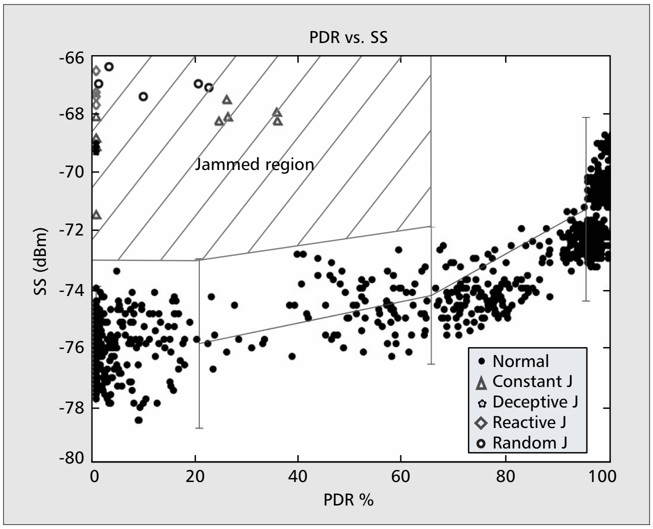
Packet Delivery Ratio — Similarly, PDR may be used to detect the presence of jamming, as the jammer can effectively corrupt transmissions, leading to a much lower PDR. Since a jamming attack will degrade the channel quality surrounding a node, the detection of a radio interference attack essentially boils down to determining whether the communication node can send or receive packets in the way it should have had the jammer not been present. More formally, let us consider the PDR between a sender and a receiver who are within radio range of each other, assuming that the network only contains these two nodes and that they are static. As noted earlier, an effective jammer results in a very poor PDR, close to 0, which indicates that PDR may be a good candidate in detecting jamming attacks. We would like to point out that a nonaggressive jammer, which only marginally affects the PDR, does not cause noticeable damage to network quality and does not need to be detected or defended against.

Next, we need to investigate how much PDR degradation can be caused by non-jamming normal network dynamics, such as congestion or failures at the sender side. Our studies in [4] showed that even in a highly congested situation where a raw traffic rate of 19.38 kb/s is offered to MICA2 radio whose maximum bandwidth capacity is 12.364 kb/s at a 100 percent duty cycle, the PDR measured by the receiver is still around 78 percent. As a result, a simple thresholding mechanism based on the PDR value can be used to differentiate a jamming attack, regardless of the jamming model, from a congested network condition. Although PDR is quite effective in discriminating jamming from congestion, it is not as effective for other network dynamics, such as a sender battery failure, or a sender moving out of a receiver's communication range, because these dynamics can result in sudden PDR drop in much the same way as a jammer does. Specifically, if the sender's battery drains out, it stops sending packets, and the corresponding PDR is 0 percent.

Consequently, compared to signal strength and carrier sensing time, PDR is a more powerful statistic in that it can be used to differentiate a jamming attack from a congested network scenario for different jammer models. However, it still cannot differentiate a jamming attack from other network dynamics that can disrupt communication between a sender and a receiver.

Advanced Detection Strategies

Rather than use such basic statistical methods, multimodal strategies, such as combining PDR with signal strength readings, appear to be promising. In a normal scenario with no



■ Figure 3. *(PDR, SS) measurements indicating the relationship between PDR and signal strength, and the (PDR, SS) values for different jammers. The shaded region is the jammed-region.*

interference, a high signal strength corresponds to a high PDR. However, if the signal strength is low (i.e., the strength of the signal is comparable to noise levels), the PDR will also be low. On the other hand, a low PDR does not necessarily imply a low signal strength: it may be that all of a node’s neighbors have died (perhaps from consuming battery resources or device faults), or the node is jammed. The key observation here is that in the first case, the signal strength is low, which is consistent with a low PDR measurement, while in the jammed case, the signal strength should be high, which contradicts the fact that the PDR is low.

Using these observations, we defined a multimodal consistency check. During a guaranteed time of noninterfered network operation, a table (*PDR*, *SS*) of typical packet delivery ratios and signal strength values are measured. From this data, one can calculate an upper bound for the maximum *SS* that would have produced a particular PDR value in a nonjammed scenario. Using this bound, the (*PDR*, *SS*) plane is partitioned into a benign region and a jammed region. To illustrate how such a detection scheme might operate, we present the results of our investigation, which was conducted using MICA2 Motes, in Fig. 3. We varied source-receiver separation for four different jammers. The PDR and SS readings were averaged over a sufficient time window to remove anomalous fluctuations (e.g. hardware-related or fading-related variations). We found the 99 percent *SS* confidence levels for different regions, and defined the jammed-region to be the region of (*PDR*, *SS*) that is above the 99 percent signal strength confidence intervals and whose PDR values are less than 65 percent. As can be seen in Fig. 3, the (*PDR*, *SS*) values for all jammers distinctly fall within the jammed region, suggesting that classification is feasible.

Mapping Jammed Areas

Following the detection of whether a node is jammed, it is desirable for the network to map out regions of the sensor network that are jammed. By having a map of jammed areas, network services can use this knowledge to influence routing, power management, and higher-layer planning. A protocol for mapping out the jammed regions of a sensor network was presented in [9]. In this article jamming detection is performed by monitoring channel utilization. Once the sensors observe that their channel utility is below a preset threshold, they conclude that they are jammed. Following detection, the jammed nodes bypass their MAC-layer temporarily and broadcast JAMMED messages, announcing the fact that they are jammed. These JAMMED messages will not be able to be received by other jammed neighbors. However, those neighbors on the boundary of the jammed region, but are not themselves jammed themselves, will be able to hear the JAMMED messages, though potentially at a higher error rate. Once non-jammed sensors receive JAMMED messages, they initiate the mapping procedure. These nonjammed nodes exchange and merge information describing which nodes they have witnessed as jammed, where those jammed sensors are located, along with neighbor information. By continuing the exchange of information regarding witnessed jammed nodes, the network will eventually be able to map out the boundary of a jammed area.

Evasion Defense Strategies

Security is a constant battle between the security expert and the clever adversary; therefore, we have chosen to take inspiration for our work from Sun Tze’s famous *The Art of War*:

> *He who cannot defeat his enemy should retreat.*

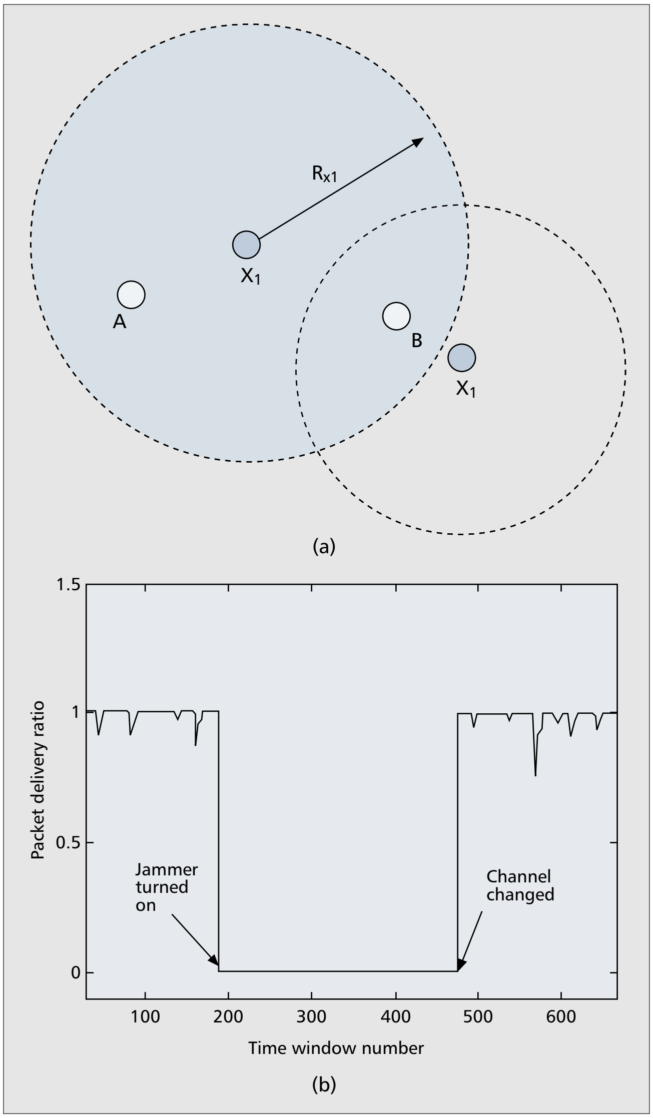
Translating this philosophy into the wireless domain, two strategies were recently proposed in [7] in order to defend against jamming attacks: channel surfing and spatial retreats. The underlying idea behind each strategy is to evade the interferer, in either the spectral or physical sense.

Channel Surfing

Channel surfing is motivated by frequency hopping modulation. Unlike frequency hopping, which is a PHY layer modulation method involving continual changing of the carrier frequency, the changing of frequencies in channel surfing is on demand and operates at the link layer. Let us examine a simple two-party communication scenario, as depicted in Fig. 4a, where adversary X_1 or X_2 has disrupted communication between A and B. In channel surfing, both A and B change their channel assignment to a new channel in order to avoid X’s interference. Changing channels in the two-party scenario is fairly straightforward. We have built a prototype using the Berkeley MICA2 platform, in which A and B detect whether they are jammed, switch to a new channel (channel assignment is done using a pseudo-random generator), and re-establish connectivity when they detect each other’s presence on the new channel [7]. Example results depicting the PDR for the two-party channel surfing prototype are presented in Fig. 4b.

Extending the notion of channel surfing to more general network scenarios, such as wireless LANs or ad hoc networks, is significantly more challenging. An initial outline of such channel surfing strategies was presented in [7]. Implementing these basic strategies, however, is a very difficult task as reliably coordinating multiple devices switching to new channels faces the usual challenges of distributed computing: asynchrony, latency, and scalability. To address these challenges, we propose the following channel surfing variations: *coordinated channel switching* and *spectral multiplexing*, discussed below.

In a coordinated channel switch, the entire network changes its channel to a new channel. In such a scheme, when a node detects that it is jammed, it switches channels and sends beacons to announce its presence on the new channel. Boundary nodes, which are not jammed but are neighbors of jammed nodes, will detect the absence of their neighbors on the original channel and probe the next channel to see if their neighbors are still nearby. If a node detects beacons on the new channel, it will switch back to the original channel and trans-



■ Figure 4. *a) Jammed two-party radio communication; b) PDR measurements from channel surfing prototype.*

mit a broadcast message informing the entire network to switch to the new channel.

Performing a coordinated channel switch across an entire network incurs significant latency as the scale of the network increases; as a result, the network may be in an unstable phase where some devices are on an old channel while others are waiting on the new channel. To alleviate the latency problem, we can have only jammed regions switch channels, and have nodes on the boundary of a jammed region serve as relay nodes between different spectral zones.

Spatial Retreats

We now explore spatial retreats, in which jammed nodes try to evacuate from jammed regions. Spatial retreats are suitable for mobile sensor networks. Merely escaping from a jammed region is not sufficient, however, as a mobile adversary can move through the coverage area and cause large swaths of the sensor network to relocate. By doing so, an adversary can cause the network to become unevenly distributed, or even partitioned, thereby severing network communications.

Therefore, spatial retreat strategies should be robust to mobile jammers. In order to achieve this robustness, a spatial retreat strategy should have two phases:

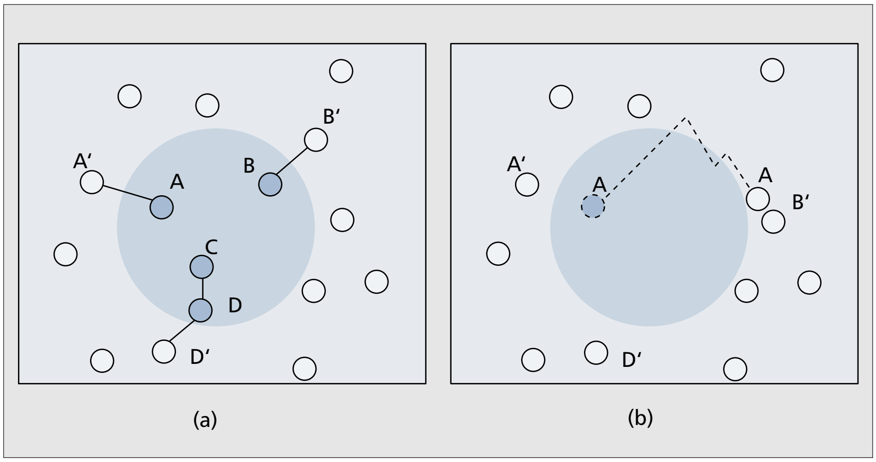
- *Escape phase*, in which the nodes located within the jammed area move to "safe" regions, and stay connected with the rest of the network
- *Reconstruction phase*, in which the mobile nodes move about to achieve uniform network coverage, thereby preventing the jammer from partitioning the network

Let us look at a robust escape strategy that has been developed. Suppose the network is connected before the jamming attack; that is, every node within the jammed area is connected with nodes outside via one hop or multiple hops. In the example shown in Fig. 5a, before the jamming attack, node A was directly connected with A', node B was directly connected with B', node D was directly connected with D', and C was connected with D' via D. After the jamming attack is detected, the nodes within the jammed area choose a random direction to evacuate. While moving, each node continuously runs the jamming detection algorithm until it leaves the jammed area. As soon as it leaves the jammed area, it tests whether there are some nodes within its radio range. If not, it moves along the boundary of the jammed area until it reconnects to the rest of the network. In Fig. 5a, if node A moves along the boundary, it will eventually arrive at a location that is between the location of A' and the original location of A, where it can reconnect to A'. The techniques that enable a node to move along the boundary of the jammed area (Fig. 5b) are discussed in detail in [10].

If the jammer is mobile, its movement may cause the network to become severely unbalanced, or even partitioned. As an example, in the two figures on the left in Fig. 6, we depict an initial network configuration (the top picture on the left), and then introduce a jammer that moves in the *y*-direction through the middle of the network. The result is a network that is severely partitioned (bottom left). In order to address this problem, we propose to apply the techniques of virtual forces and potential fields, which are popular methods for governing motion in robotic systems, to continuously repair the network topology, regardless of the jammer movement. To use virtual forces, we need three types of forces: the forces between the nodes, the force from the boundary of the region, and the force from the boundary of the jammed area. In order to define the boundary of the jammed region, one must use a jammed area mapping technique, such as the one proposed in [9]. By carefully defining these forces and their interplays, we can continually repair the network topology as presented in [10]. We now examine the behavior of our robust spatial retreat strategy by looking at an experiment involving a mobile jammer cutting across the network coverage area. Figure 6 illustrates the evolution of the mobile sensor network's topology as the jammer moves through the network, and the robust spatial retreat algorithm not only evacuates the jammed area but also repairs regions left empty by the mobile jammer.

Competition Strategies: Power Control and Code Throttling

An alternative to performing evasion strategies, where the sensor nodes try to evade the jammer in some sense, is to have the sensors attempt to compete against the jammer. In this case the objective should be for the sensors to improve the reliability of the reception of their packets. This requires, if a node detects it is jammed, that it will ignore the fact that it is jammed, and transmit its packet anyway. In order for nodes that are jammed, as well as nodes near jammed regions (as mapped out using techniques proposed by [9]), to compete against the jammer, they should adjust the coding and power of their communications at the lower layers. By employing a



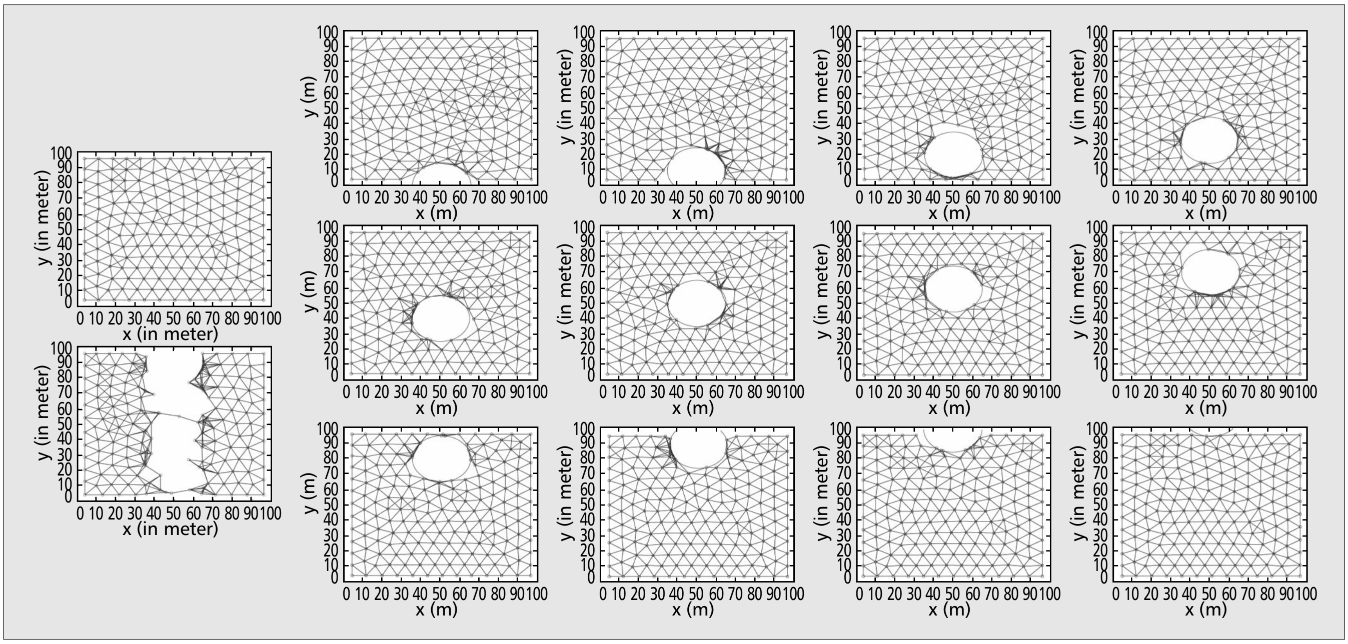
■ Figure 5. *Escaping from the jammed area: a) The network topology when the jamming attack occurs — the jammed area is highlighted by the shaded area; b) the dashed line marks the trace through which node A escapes from the jammed area and reconnects to the rest of the network.*

stronger error correcting code with a lower information rate, we reduce the overall throughput but increase the likelihood of packets successfully being decoded. We may also increase the transmission power employed by legitimate radio devices in order to allow reception to operate at higher signal-to-noise ratios. This strategy involves a return to looking at the problem using traditional communications theory, as well as methods employed by the electronic warfare community. However, there are some key differences that suggest this direction of research will not fall out immediately from traditional communications theory for anti-jamming, as described in [1, 2]. In particular, since we are operating in a system employing carrier sensing, it should be realized that using increased power levels introduces new problems as radio devices will have a larger radio coverage pattern, thereby increasing the likelihood of collisions and unintentional interference with other legitimate radio devices.

Additionally, there are several systems-level issues that need to be addressed when employing these methods. For example, the work of [11] provides a theoretical study into the use of low density parity check and Reed-Solomon codes to cope with jamming, but does not address critical systems integration issues. Here, applying the ECC must be done in such a way to protect the critical frame preamble and not just the payload. Furthermore, it is necessary to develop protocols that adaptively throttle code rates based upon a perceived threat level. In particular, the detection mechanisms described earlier should be modified in order to provide an estimate of the severity of a jamming threat. This parameter will serve as input into a protocol that will adaptively adjust power and code rate. Taken together, these issues just outlined suggest that an important direction for future exploration would involve looking at the viability of combining power control and code throttling in the presence of RF jamming.

Concluding Remarks

Due to the low-cost design of sensor nodes, and the ease with which they may be reprogrammed, sensor networks will be very susceptible to intentional radio interference attacks. This article has surveyed both the attack and defend side of jamming wireless sensor networks. Four different types of jamming devices, which involved bypassing MAC layer carrier sensing, have been discussed. We then turn to the problem of detecting the presence of jamming, where we have illustrated why simple statistics are not sufficient. Multimodal detection methods have recently been proposed as a means to circumvent this challenge. Following detection, it is desirable that the network can repair itself. Toward this end, two evasion strategies have been discussed: the first involving the sensor



■ Figure 6. *The two figures on the left illustrate that a mobile jammer can partition the network. The remaining figures depict the ability of a robust spatial retreats algorithm to repair the effect of a jammer passing through the coverage region.*

network adapting its operating frequencies, the second suitable for mobile sensor networks and involving nodes relocating themselves. A different defense strategy involves sensors trying to out-compete the jammer by employing error correcting codes and increasing the node transmission power. Both evasion and competition strategies are at an early stage of investigation by the community, and as these techniques mature an important area for study will be understanding and classifying the scenarios where one defense strategy is advantageous over another.

References

[1] B. Sklar, *Digital Communications: Fundamentals and Applications*, Prentice Hall, 2nd ed., 2001.

[2] J. G. Proakis, *Digital Communications*, McGraw-Hill, 4th ed., 2000.

[3] AusCERT, "AA-2004.02 — Denial of Service Vulnerability in IEEE 802.11 Wireless Devices," http://www.auscert.org.

[4] W. Xu *et al.*, "The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks," *MobiHoc '05: Proc. 6th ACM Int'l. Symp. Mobile Ad Hoc Net. and Comp.*, 2005, pp. 46–57.

[5] Y. Law *et al.*, "Link-Layer Jamming Attacks on S-Mac," Proc. 2nd Euro. Wksp. Wireless Sensor Networks, 2005, pp. 217–25.

[6] A. Wood and J. Stankovic, "Denial of Service in Sensor Networks," *IEEE Comp.*, vol. 35, no. 10, Oct. 2002, pp. 54–62.

[7] W. Xu *et al.*, "Channel Surfing and Spatial Retreats: Defenses Against Wireless Denial of Service," *Proc. 2004 ACM Wksp. Wireless Security*, 2004, pp. 80–89.

[8] J. Polastre, J. Hill, and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," *SenSys '04: Proc. 2nd Int'l. Conf. Embedded Networked Sensor Sys.*, 2004, ACM Press, pp. 95–107.

[9] A. Wood, J. Stankovic, and S. Son, "JAM: A Jammed-Area Mapping Service for Sensor Networks," *24th IEEE Real-Time Sys. Symp.*, 2003, pp. 286–97.

[10] K. Ma, Y. Zhang, and W. Trappe, "Mobile Network Management and Robust Spatial Retreats Via Network Dynamics," *Proc. 1st Int'l. Wksp. Resource Provisioning and Mgmt. in Sensor Networks*, 2005.

[11] G. Noubir and G. Lin, "Low-Power DoS Attacks in Data Wireless LANs and Countermeasures," *SIGMOBILE Mob. Comp. Commun. Rev.*, vol. 7, no. 3, 2003, pp. 29–30.

Biographies

KE MA (kemaru@winlab.rutgers.edu) received a B.S. degree in information science from Beijing University of Posts and Telecommunications, China, in 1998, and an M.Phil. degree in information engineering from the Chinese University of Hong Kong, China, in 2003. He is currently a Ph.D. student in the Department of Electrical and Computer Engineering at Rutgers University, New Brunswick, New Jersey. His current research focuses on optimization in sensor networks, especially for sensor networks that involve mobile nodes.

WADE TRAPPE [M] (trappe@winlab.rutgers.edu) received his B.A. degree in mathematics from the University of Texas at Austin in 1994, and a Ph.D. in applied mathematics and scientific computing from the University of Maryland in 2002. He is currently an assistant professor at the Wireless Information Network Laboratory (WINLAB) and the Electrical and Computer Engineering Department at Rutgers University. His research interests include wireless security, wireless networking, multimedia security, and network security. While at the University of Maryland, he received the George Harhalakis Outstanding Systems Engineering Graduate Student award. He is a co-author of the textbook *Introduction to Cryptography with Coding Theory*, Prentice Hall, 2001. He is the recipient of the 2005 Best Paper Award from the IEEE Signal Processing Society. He is a member of the IEEE Signal Processing and Communications Societies, and the ACM.

WENYUAN XU (wenyuan@winlab.rutgers.edu) received a B.S. degree in electrical engineering from Zhejiang University, Hangzhou, China , in 1998, and an M.E. degree in computer science and engineering from Zhejiang University, Hangzhou, China , in 2001. She is currently a fourth year Ph.D. student in WINLAB at Rutgers University. Her primary research interest is wireless network security, with an emphasis on DoS attacks in sensor networks.

YANYONG ZHANG [M] (yyzhang@winlab.rutgers.edu) received a Ph.D. in computer science and engineering from Penn State University in 2002. She is an assistant professor in the Department of Electrical and Computer Engineering at Rutgers University, and a faculty member at WINLAB. Her current research interests include sensor networks, sensor network security and privacy, and fault-tolerant sensor networks. She has received several NSF grants on these topics, including an NSF CAREER award. She has organized the Workshops on System Management Tools for Large-Scale Parallel Systems (2005, 2006), and the Workshop on Self-Healing, Adaptive and Self-Managed Systems (SHAMAN) 2002, and is on the technical committee for several conferences and workshops. She is a member of ACM.