Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 485345, 9 pages
doi:10.1155/2012/485345

Research Article

Optimal Adaptive Antijamming in Wireless Sensor Networks

Yanmin Zhu,[1,2] Xiangpeng Li,[1] and Bo Li[1,3]

[1] *Department of Computer Science and Engineering, Shanghai Jiao Tong University, Minhang, Shanghai 200240, China*
[2] *Shanghai Key Lab of Scalable Computing and Systems, Shanghai Jiao Tong University, Minhang, Shanghai 200240, China*
[3] *Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong*

Correspondence should be addressed to Yanmin Zhu, yzhu@cs.sjtu.edu.cn

Received 19 July 2012; Revised 12 October 2012; Accepted 8 November 2012

Academic Editor: Regina B. Araujo

Wireless sensor networks have been widely applied to various application domains such as environmental monitoring and surveillance. Because of reliance on the open transmission media, a sensor network may suffer from radio jamming attacks, which are easy to launch but difficult to defend. Attacked by jamming signals, a sensor network may experience corrupted packets and low network throughput. A number of defense techniques have been proposed. However, each defense technique is suitable for only a limited range of network and jamming conditions. This paper proposes an adaptive approach to antijamming for sensor networks by combining the strength of state-of-the-art antijamming techniques, which enables each node to adaptively select the optimal antijamming technique for different jamming conditions. The great challenge is that the sensor network undergoes varying jamming conditions over time. We address this challenge by formulating the antijamming problem of the sensor network as a Markov decision process and propose an efficient algorithm for computing the best antijamming strategy. By comprehensive simulation experiments, we demonstrate that a sensor network using the derived antijamming strategy can well defend from radio jamming attacks and in the meanwhile retain high energy efficiency.

1. Introduction

With the appealing characteristics of low-cost, easy to deploy, and unattended operation and the ability of withstanding harsh environmental conditions, wireless sensor networks have been implemented in a wide range of applications, such as environment monitoring [1] and event detection [2].

Sensor networks transmit wireless signals over the open shared media. This leaves a sensor network vulnerable to radio jamming attacks. In [3, 4], several jamming attacks have been explored, which corrupt control packets, such as RTS (Request-to-Send) and CTS (Clear-to-Send). The jammer just keeps sending packets like RTS to prevent transmission of legitimate packets. These methods are usually based on the statistics of packet transmission history and can cause severe damage to the sensor network with only modest overhead.

Thus, antijamming is enormously important for secure operation of sensor networks. As being well known, sensor nodes are typically powered by batteries and hence limited in power supply. This has been generally accepted as one of the crucial issues of the sensor network. Therefore, antijamming needs to be energy efficient.

A number of antijamming methods have been proposed, such as channel surfing [5], error correction codes and transmission power adjustment. However, these existing countermeasures of jamming attacks are usually suitable for a limited range of jamming conditions with varying operation cost. In the real world scenario, jamming attacks may be very different in nature and may change over time. In addition, radio signals are unstable as many factors may cause jamming signal attenuated in different ways for different environments. As a result, different nodes suffer different degrees of radio jamming. Thus, it is inefficient for a whole sensor network simply to apply a single antijamming technique. This may result in poor performance of antijamming and/or still suffer serious performance degradation of energy consumption.

As shown in Figure 1, sensor nodes in the jamming area may experience different degrees of jamming attack.

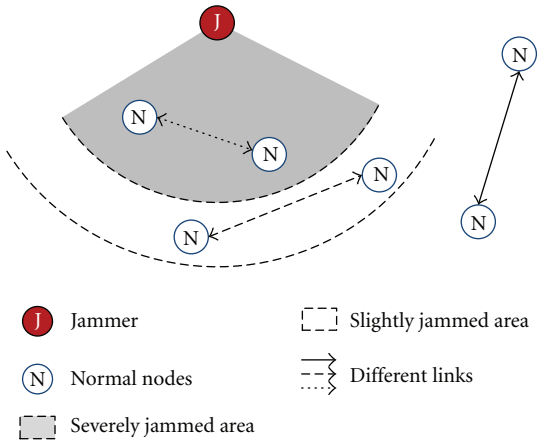

Figure 1: Illustration of varying jamming effects on different nodes.

Thus, the quality of different links varies. For ease of understanding, we show only two levels of jamming signal strength and result in two regions: severely jammed and slightly jammed. In the severely jammed area, the nodes must adopt a heavy antijamming technique in order to transmit a packet successfully. Such a technique usually requires a high energy consumption rate. In the slightly jammed area, the link may experience lower delivery ratio but is still able to deliver data. In this case, a light antijamming technique may be employed for improving the link quality.

This paper focuses on proposing an adaptive approach to antijamming for sensor networks, by combining the strength of different antijamming techniques. With this solution, a sensor node is able to select the best antijamming techniques for different jamming conditions. The challenge is that the sensor network may undergo different jamming conditions over time. As a result, an antijamming technique that is good for the present may produce poor performance after the network condition changes. Importantly, the overhead for a sensor node switching from one antijamming technique to another is usually nonnegligible. In this paper, we solve this challenging issue by formulating the antijamming problem of the sensor network as a Markov decision process and propose an algorithm for computing the best antijamming strategy. Conclusive performance results demonstrate that our approach with the derived antijamming strategy allows a sensor network to perform well against radio jamming attacks.

The rest of this paper is organized as follows. In Section 2, we present the related work. The system model and problem formulation are described in Section 3. In Section 4, we elaborate the design of our MDP-based algorithm and the determination of the optimal antijamming strategy. We present some discussion issues on the design in Section 5. The evaluation results are discussed in Section 6. We finally conclude the paper in Section 7.

2. Related Work

Since radio jamming attacks have been recognized as a crucial issue in sensor networks, a plenty of research has been conducted. A good survey for radio jamming attacks and counter measures against radio jamming in sensor networks can be found in [6].

2.1. Jamming Attacks. In the literature, many possible jamming attacks have been studied or even implemented in the context of wireless sensor networks.

In [7], it is pointed out that jamming attacks can effectively cause a denial of service of either transmission or reception functionalities. These attacks can easily be accomplished by emitting a radio signal on a particular channel. Different jamming attacks may be posed against a sensor network.

In [8], the authors introduce four different jamming attack models. They explore various detecting techniques for jamming attacks in sensor networks. The packet delivery ratio (PDR) is used to classify a poor radio link, and then a consistency check is performed to make sure whether it is caused by jamming or not. This method is very efficient in identifying the adversary. Nevertheless, it does not provide effective countermeasures against jamming.

In [3], it shows that encrypting packets help to prevent jammers from launching attacks based on the content of the packets. However, temporal intervals of packets induced by the nature of the protocol may release patterns. Such patterns can be exploited by jammers even when packets are encrypted. The authors study packet interarrival times of three representative MAC protocols, S-MAC, LMAC, and B-MAC. And they develop several jamming attacks that allow a jammer to compromise S-MAC, LMAC, and B-MAC in a energy-efficient fashion.

In [9], the authors investigate a scenario where a radio jammer is strategic, meaning that the jammer controls the probability of jamming and the transmission range to maximize the damage to the sensor network in terms of the number of corrupted links. The jammer stops jamming attacks when it is detected by a monitoring node.

In [4], the authors analyze the effect of radio jamming against a sensor network. When the jamming is on the physical layer, the analysis shows that the loss is proportional to the jammer power. On the other hand, the jamming can be launched on the link layer or able to exploit the semantics of MAC-layer packet transmissions. For example, when CSMA/CA is used as the MAC protocol, the jammer could detect transmissions of RTS and CTS frames. Thus, the jammer can selectively jam those control frames.

In [10], reactive jamming is studied, by which a jammer only targets to jam packets already on the air. When jamming only selected packets, the risk of the jammer being detected by the sensor network is minimized. Previously, it was thought that reactive jamming is too challenging for an attacker to implement. The authors demonstrate that flexible and reliable software-defined reactive jamming is practical and easy to implement.

2.2. Antijamming Techniques. Due to the serious threats that may be caused by radio jamming attacks, a number of antijamming techniques have been proposed.

In [11], the paper focuses on the performance analysis of various error control codes in terms of BER performance and power consumption. The authors implement different error control codes using VHDL on FPGA and ASIC. In addition, the energy consumption for different error control codes is also measured. BER is the performance metric, evaluated by transmitting randomly generated data through a Gaussian channel. They found that binary-BCH codes with ASIC implementation are best suitable for wireless sensor networks. In the presence of jamming attacks, the channel condition becomes worse, and error control codes can help reduce BER.

One possible solution to cope with radio jamming is to adapt the transmission power of nodes with respect to the power of the jamming radio [12]. It is found that the effect of jamming upon source-receiver communications is not isotropic. The effect of jamming is studied by improving the transmission power on a testbed with Mica2 motes.

In [9], the author formulates the jamming issue as an optimization problem. By solving the optimization problem at both the network and jammers, they can control their probability to transmit the radio signals, so as to achieve the optimal jamming and defense effectiveness. This work studied the interaction between jammer and the nodes in the networks.

In [13], the author proposes a fast jamming detection algorithm in wireless sensor networks. It collectively evaluates the PDR in a given area instead of a pair of nodes, which allows the node detect jamming in a much faster way. In [14], the paper talked about the insider jamming. An attacker compromises some legitimate nodes to acquire cryptographic information and then jams the network. The solution of this paper is to determine the channels by the group key shared by all nodes. When insider jamming happens, the network will generate a new group key for the noncompromised nodes to protect the network from the insider jamming.

In [14], jamming attacks from insiders are studied. An attacker may gain the common cryptographic information through those compromised nodes and then launch jamming attacks. The paper then proposes a compromise-resilient antijamming scheme to deal with the insider jamming problem. According to the scheme, the physical channel used by a sensor network is determined by the group key shared by the sensor nodes.

A technique called channel surfing [15, 16] is developed to cope with the jamming interference, by which the sensor nodes change the communication channel when they detect jamming attacks. Two channel surfing methods are explored. One is coordinated channel switching, in which the entire sensor network changes the radio channel. The other is spectral multiplexing, in which the nodes in the jammed area change the radio channel and the nodes on the boundary act as relays.

In [17], a jammed area detecting and mapping service is developed. As a result, this service allows network applications to reason about the region as an entity. Evaluation results show that regions can be mapped in 1–5 seconds.

2.3. Summary. This paper complements the existing work of antijamming by combining the strength of different antijamming techniques and proposes an adaptive antijamming solution to wireless sensor networks.

3. Model and Problem Formulation

In this section, we first present the performance description and then introduce the antijamming techniques considered in this paper. Note that it is possible to include more antijamming techniques and the proposed algorithm is still valid.

3.1. Problem Description. In a wireless sensor network, there are a set of nodes, denoted as N and a set of a few jammers in the environment, denoted as J. For each node, there are K antijamming techniques available for different jamming conditions.

Since the jamming signal is not constant but varying over time, the node periodically evaluates the channel condition, $\varphi_n(t)$. The period is denoted as τ. Based on $\varphi_n(t)$, the node chooses the proper antijamming technique to deal with different jamming signal. For the node, each antijamming technique has different cost, C_k. Focusing on the energy efficient purpose, we evaluate this cost as a function of the energy that the node will consume in the next period. Consider

$$C_{k(t)} = f\left(E_{k(t)}(\tau)\right). \tag{1}$$

In the following period, the performance reward of the node using specific antitechnique is the improvement of the communication between the nodes. Consider

$$R_{k(t)} = (Q(t)) = Q(t+\tau) - Q(t), \tag{2}$$

where $R_{k(t)}$ is the performance reward, and $Q(t)$ is the communication quality.

The objective of the nodes is to choose the proper antijamming techniques, which considers both the reward and the cost. From the definition earlier, we know that $C_{k(t)}$ and $R_{k(t)}(Q(t))$ are different kinds of quantity. Thus, we make the objective by minimizing the product of cost and reward. We focus on a relatively long time period, denoted as T, so the nodes totally make $\lfloor T/\tau \rfloor$ decisions. Then, we formulate the overall objective as:

$$\min_{k(t_i)\in\Delta} \quad \sum_{i=0}^{T/\tau} R_{k(t_i)}(Q(t))C_{k(t_i)}. \tag{3}$$

3.2. Antijamming Techniques. There are a number of different antijamming techniques in the literature. Without loss of generality, this paper assumes that each sensor node is capable of applying three representative antijamming techniques.

3.2.1. Transmission Power Adjustment. With this technique, a sender node increases its transmission power, and thus

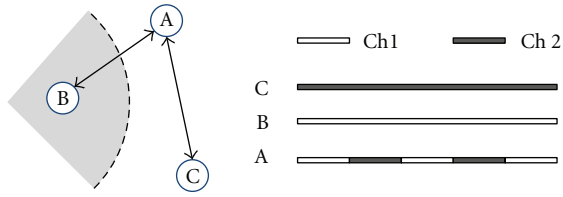

Figure 2: Illustration of the channel hopping technique.

increases the SNR at the receiver node [12]. This technique is suitable under a slight jamming condition, for example, at the periphery of the jamming area. In that area, the jamming signal is relatively weak, so the nodes usually only need to raise its transmission power by one or two levels. This technique introduces modest energy cost.

3.2.2. Error-Correcting Code. An error-correcting code [11] is used for correcting some error bits that occurred during transmission. Before transmission, the node encodes the packet. When the receiver has received the packet, the decoding process is capable of correcting some error bits by using the redundancy information contained in the encoded packet (under a certain condition, e.g., the number of error bits is smaller than a given threshold).

Applying error-correcting codes as an antijamming technique is energy efficient as it largely relies on computation and transmission of extra bits. Many error-correcting codes have been proposed. This paper chooses Reed-Soloman code as an example. Other codes can similarly work with our algorithm.

3.2.3. Channel Hopping. With this technique [16], a sensor node will change the working channel when it detects strong jamming signals in the current channel. As shown in Figure 2, node B in the shaded area is jammed. Node A is an intermediate node which works on two channels. It switches between the two channels, so it can keep the network connected. When it changes its working channel, it will notify its neighbor working on the same frequency immediately. The schedule of the intermediate nodes is shown in the right of Figure 2. When nodes B and C transmit periodically, and the schedule of Node A is appropriate, there will be no packet loss.

Thus, each sensor node has a set of available antijamming techniques, denoted as $\Delta = \{\delta_0, \delta_1, \delta_2, \delta_3\}$, representing null technique, transmission power adjustment, error-correcting code, and channel hopping, respectively.

4. Optimal Adaptive Antijamming

In this section, we present our algorithm in detail, which formulates the problem as a Markov decision process and obtains an optimal policy for antijamming.

We formulate the problem as a 4-tuple: (S, Δ, P, C), where S is the set of all the node states that describe the channel conditions, Δ is the set of antijamming techniques, $P_\delta(s, s')$ is the probability that the node state becomes s' after technique δ is performed at state s, and C is the cost of the antijamming technique. In the following, we introduce these four components in detail.

input:
PDR: Packet Delivery Ratio
RSSI: Received Signal Strength Indicator
CAS: Current Antijamming Strategy
Φ: The threshold of PDR for normal communication
K: The threshold of RSSI for weak link
output:
S: System State
main procedure:
if PDR $\geq \Phi$
return $S = \min(\Pi(\text{CAS}, \text{PDR}), \Gamma(\text{RSSI}))$
else if RSSI $< K$
return NormalState
else
return $S = \max(\Pi(\text{CAS}, \text{PDR}), \Gamma(\text{RSSI}))$
end if

Algorithm 1: System state determination.

4.1. System States. The system states denote the channel conditions $\varphi_n(t)$, which describe the different degrees of the jamming conditions.

In [8], the authors use PDR and RSSI (Received Signal Strength Indicator) to denote jamming signals with good accuracy. In our algorithm, however, the antijamming strategy affects the value of PDR. It will not be so accurate for using those two parameters to describe the current channel condition. We will use a method that considers the current antijamming strategy as well.

Considering the limited computing ability of the sensor nodes, it is important to reduce the complexity of the algorithm. Therefore, we use only four states, denoted as $S = \{0, 1, 2, 3\}$, which represent four different jamming conditions and correspond to the three antijamming strategies plus the case requiring no countermeasures. For the same reason, we use five levels for RSSI. As Algorithm 1 shows, functions $\Pi(\text{CAS}, \text{PDR})$ and $\Gamma(\text{RSSI})$ output the system state depending on the current countermeasure. For each countermeasure, it is suitable for a certain level of jamming, so there is a correspondence between states and countermeasures. Thus, when the PDR value is high enough, it indicates that the current antijamming strategy is effective. Then, function $\Pi(\text{CAS}, \text{PDR})$ outputs the corresponding state. For the transmitting power of the node is always the same, the RSSI value will be in a certain level of the normal case. When the jammer is present, the RSSI value will also rise. Under different jamming levels, the RSSI value will be different. Making this correspondence between RSSI and the jamming conditions, we realize function $\Gamma(\text{RSSI})$. There is a special interval of RSSI value, which is below the normal signal strength, meaning that the link is weak. In that case, the algorithm will return a normal state, because a low PDR value is not caused by radio jamming.

The smaller the value of the state is, the better the link condition is. When the PDR value is high, we prefer a light antijamming strategy. Thus, we choose the smaller state. When the PDR is not that good, for the sake of effectiveness, we choose a worse case as the system state.

4.2. Transition Probability. Since the state of nodes changes due to the varying jamming conditions, the transition probability of the states also describes the variation of jamming signals.

The transition probability is acquired by analyzing historical data. The nodes record number k_i^δ of the node reaching the state $i \in S$ and performing action δ. If the state changes to j at the current period, then the nodes add one to the variable $k_{i\to j}^\delta$ that keeps the total number of the state transitions from i to j for action δ. When $S(t-\tau) = i$ and $S(t) = j$, then the transition probability can be calculated as

$$\begin{aligned} P_\delta(S(t-\tau), S(t)) &= \frac{\Pr(S(t-\tau), S(t), \delta)}{\Pr(S(t-\tau), \delta)} \\ &= \frac{k_{i\to j}^\delta / k_i}{k_i^\delta / k_i} = \frac{k_{i\to j}^\delta}{k_i^\delta}, \end{aligned} \tag{4}$$

where k_i is the total number of nodes that reach the state i.

4.3. Cost of Antijamming Techniques. The cost function is about the energy consumption caused by the antijamming techniques.

4.3.1. Adjusting Transmission Power. The cost of increasing transmission power is easy to compute. It is the raised power multiplied by the packet transmission time,

$$C_I = \sum_{i=1}^{n} P_{\text{tx}} t_{\text{pkt}} = \sum_{i=1}^{n} \frac{P_{\text{tx}} L_{\text{pkt}}}{R_{\text{bits}}}, \tag{5}$$

where C_I is the cost of the increasing power action; the time of transmitting every packet is t_{pkt}; L_{pkt} and R_{bits} represent packet length and the bits transmission rate, respectively; n is the total number of packets within one period τ when the node is performing that technique.

4.3.2. Error-Correcting Codes. The energy consumed by this technique is the power used for the encoding and decoding process and transmitting the redundant bits. For the error-correcting codes has to be undertaken by both of the communicating nodes, the notification process also consumes extra energy.

As the notification process is all the same and will not change, therefore, the energy expended is a constant. With this information, we have the cost function of the error-correcting codes technique as

$$C_{\text{EC}} = \sum_{i=1}^{n} \left(\frac{P_{\text{tx}} L_{\text{EC}}}{R_{\text{bits}} + E_{\text{dec}}} \right) + E_{\text{noti}}. \tag{6}$$

L_{EC} is the length of the encoded packets. Since there are more bits than the normal packets, L_{EC} will be bigger than L_{pkt}. For this method, the nodes transmit the signals in a normal level of power. The first component of (6) is the energy for transmitting the packets. E_{dec} is the energy spent for decoding the packets or correcting the errors of the received signal. In [18] the author gives a method to calculate the computing energy cost.

E_{noti} is the energy expended for the notification process. Because this process is invariable, it just equals the multiplication of the normal power level and the time spent for this process.

Added up this tree part of the energy, we will get the total energy consumed by the error-correcting strategy.

4.3.3. Channel Surfing. The nodes also transmit their signals in normal power in this technique. As the error-correcting code strategy, the nodes have to notify the neighbors that it will change to another channel, for the neighbors undertaking the same strategy to keep the connectivity of the network. Secondly, when the nodes take this strategy, the intermediate nodes also need to send some packets when it switches to each channel. The total cost could be

$$C_{\text{CS}} = \sum_{i=1}^{m} \frac{P_{\text{tx}} L_{\text{noti}}}{R_{\text{bits}}} + \sum_{i=1}^{n} \frac{P_{\text{tx}} L_{\text{pkt}}}{R_{\text{bits}}} + E_{\text{noti}}. \tag{7}$$

E_{noti} is just like the error-correcting codes, for the notification process is all the same. In this strategy, the node has to inform the neighbors when it goes to a new channel. The $P_{\text{tx}} L_{\text{noti}} / R_{\text{bits}}$ is the energy expended to send those packets. m means the total channel switches. The energy consumed by data packets is the second part in (7).

4.4. Policy Determination. In the following, we use PDR to describe performance reward,

$$R_{\delta(t-\tau)}(Q(t-\tau)) = \text{PDR}(t) - \text{PDR}(t-\tau), \tag{8}$$

and we define:

$$\gamma_{i\delta} = 1 - \overline{R}_{\delta(t)}(S(t)), \tag{9}$$

$$\lambda_{i\delta} = \gamma_{i\delta} C_\delta, \tag{10}$$

where $\lambda_{i\delta}$ is the cost of technique δ at state i.

Because $\gamma_{i\delta}$ is a coefficient of the cost, (9) makes the more effective technique that has higher reward and less energy cost. It results in a smaller $\gamma_{i\delta}$ when the jamming is severe. Then, the cost $\lambda_{i\delta}$ will become less than the techniques with less energy cost, for the value $\gamma_{i\delta}$ of the lighter technique is probably greater than one. Therefore, the node is more likely to perform a more effective technique to guarantee a certain communication quality. On the other hand, when the jamming is not serious, a heavy technique does not gain so much reward that makes the cost less than the lighter ones. In such a case, the technique with less energy cost is preferred.

Based on the previous definitions, we devise the policy improvement algorithm to solve the MDP problem. The details of this algorithm are explained as follows.

We denote the total expected cost of the node beginning in state i and evolving for n periods by $\varepsilon_i^n(D)$, where D is the related policy. Then, we have

$$\varepsilon_i^n(D) = \lambda_{i\delta} + \sum_{j=1}^{M} P_\delta(i,j)\varepsilon_j^{n-1}(D), \quad \text{for } i = 1,2,\ldots,M, \tag{11}$$

where $\lambda_{i\delta}$ is the cost introduced in the first period. The second part of the equation is the cost of the next n period. There are M states in total. The long run expected average cost per unit time could be expressed as

$$\zeta(D) = \sum_{j=1}^{M} \pi_i \lambda_{i\delta}, \tag{12}$$

where π_i is the steady distribution of the states. We can have an approximate relationship when n is large as follows:

$$\varepsilon_i^n(D) \approx n\zeta(D) + \varepsilon_i(D). \tag{13}$$

We can consider $\varepsilon_i(D)$ as the effect on the total expected cost due to beginning in state i. After substituting (13) into (11), we get

$$\zeta(D) = \lambda_{i\delta} + \sum_{j=1}^{M} P_\delta(i,j)\varepsilon_j(D) - \varepsilon_i(D), \quad i = 1,2,\ldots,M. \tag{14}$$

The policy improvement algorithm starts by choosing an arbitrary policy D_n and set $\varepsilon_M(D_n) = 0$. Then, it solves (14) to $\zeta(D_n), \varepsilon_1(D_n), \varepsilon_2(D_n), \ldots, \varepsilon_M(D_n)$. We use $\varepsilon_i(D_n)$ to find another policy D_{n+1} such that for each state i,

$$\min_{\delta\in\Delta} \quad \lambda_{i\delta} + \sum_{j=1}^{M} P_{i,j}(\delta)\varepsilon_j(D) - \varepsilon_i(D), \tag{15}$$

where $\delta = d_i(D_{n+1})$. When D_{n+1} and D_n are identical, this iteration process will stop. Otherwise, it sets $n = n + 1$ and this process continues.

4.5. Algorithm Framework. The framework of the optimal antijamming algorithm is shown in Algorithm 2.

Considering the jamming pattern may change over time, each policy has an effective period, as shown in Algorithm 2. When the effective period runs up, the nodes will determine a new policy to make sure that the current policy is suitable for the jamming pattern. After the policy is acquired, the nodes begin to communicate with each other. The communication period allows a node to obtain PDR. The node then determines the state using the algorithm introduced before. Then, it updates the transition probabilities, which is for later policy determination. Before the next period of communication, the nodes choose a proper antijamming strategy based on the current policy, and it will send a notification to make sure the nodes are using the same antijamming strategy.

```
while (1) do
    while (PolicyEndureTime) do
        while (CommunicationPeriod) do
            Nodes Communicate;
        end while
        GetPDR();
        DetermineNodeState();
        UpdateTransitionProbabilities()
        ChooseStrategy(Policy, State);
        SendNotification();
    end while
    DeterminePolicy();
end
```

Algorithm 2: Antijamming algorithm framework.

5. Discussions

In this section, we discuss two important design issues when designing the adaptive antijamming techniques for wireless sensor networks.

5.1. Integration of Multiple Antijamming Techniques. It is important for individual sensor nodes in a sensor network to work collaboratively as a whole. Since multiple antijamming techniques are equipped, sensor nodes in the network may be using different antijamming techniques. This may cause network disconnections to the sensor network, resulting in failed packet delivery. For example, when the technique of channel surfing is adopted, two neighboring nodes may be using different channels and make the link between the two nodes broken.

Thus, it is important to develop a coordination protocol which ensures that the network connectivity is maintained even the sensor nodes are using different antijamming techniques. The design of such a protocol is beyond the scope of the paper and is subject to future research.

5.2. Exploiting More Antijamming Techniques. In this paper, we have discussed three typical antijamming techniques. However, as mentioned in the Section of Related Work, there are many more antijamming techniques. Then, it is an important problem on what antijamming techniques should be equipped on sensor nodes. Note that the increasing number of equipped antijamming techniques may consume more resources such as memory and energy. More importantly, it may also increase the design complexity of the coordination protocol mentioned previously. Thus, the selection of the antijamming techniques should balance the advantage brought by more antijamming techniques and the overhead introduced.

6. Performance Evaluation

In this section we first present the performance evaluation metrics and the simulation setup and then discuss the evaluation results.

Table 1: Simulation parameters.

Parameter	Value
Region	40 m by 40 m
Propagation model	Large-scale path loss
Loss component n	4
Transmission power	−5 dBm
Adjusted power	−3 dBm
I_{tx}	13.9 mA
I_{rx}	19.7 mA
Voltage	3 v
Data packet length	50 Bytes
Ack packet length	20 Bytes
Transmission rate	250 Kb/s

6.1. Methodology and Simulation Setup. We have developed a discrete event driven simulator for simulating jamming attacks and antijamming operations of a sensor network. More specifically, we simulate the details of radio propagation of sensor nodes. The main simulations parameters are listed in Table 1.

We adopt the following simulation setting. The sensor nodes are uniformly distributed in a square area of 40 m by 40 m. We use the large-scale path loss model to describe the attenuation of a radio signal. The parameters of energy consumption are compatible with the CC2420 Chip. The nodes in the network transmit signals with the power of −5 dBm, and the jammer randomly chooses the power from (0, −1, −3, −5, −7, −10, −15, and −25) dBm. The raised power is −3 dBm, which is one level higher. The current for transmission is $I_{tx} = 13.9$ mA, and the current for reception is $I_{rx} = 19.7$ mA. The typical voltage is $V = 3$ v. The jammer with higher power means that the nodes near the jammer will suffer more severe damage. The jammer is placed in the middle of the network so that it jams as many nodes as possible. For the large scale path loss model, we set the value of the path loss exponent $n = 4$, and the reference distance $d_0 = 1$. The packet length L_{ACK} is 25 Bytes, and L_{data} is 50 Bytes. These values comply with the IEEE 802.15.4 Standard. They are convenient for the error-correcting codes, which is RS (31, 25, 3). The bit transmission rate R_{bits} is 250 Kb/s.

6.2. Packet Delivery Ratio. We first study the packet delivery ratio of different antijamming strategies. Figure 3 shows the effectiveness of the computed antijamming strategies in terms of packet delivery rate. As the normal condition, the nodes that are within 8 meters from the jammer have lost more than 50% packets, which make it difficult for effective communication in sensor networks. The strategy of raising the transmission power can improve the situation, but there are still a lot of nodes which severely suffer from packet loss. As BER affects PDR signifcantly, we can see that the curve of the strategy of using error-correcting Code is much better than the former two strategies. The channel surfing strategy lets the nodes work on another channel which experiences no jamming attack, so the nodes can communicate with each other properly. It is also effective and can maintain the communication quality.

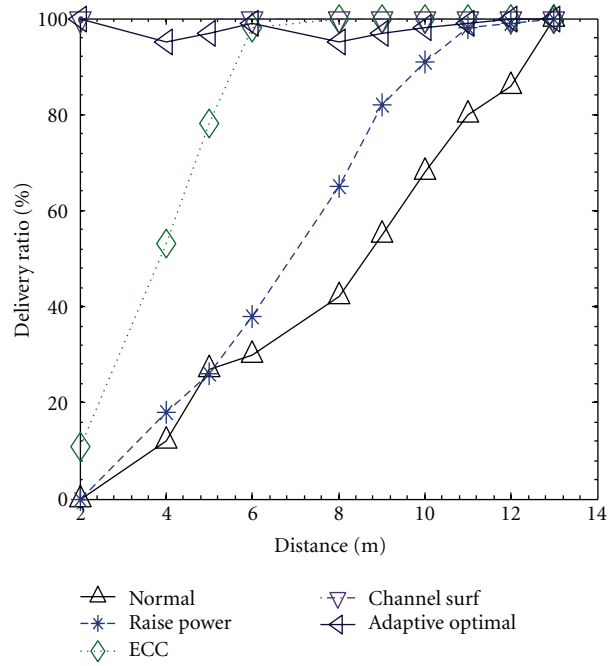

Figure 3: Delivery ratio of different strategies.

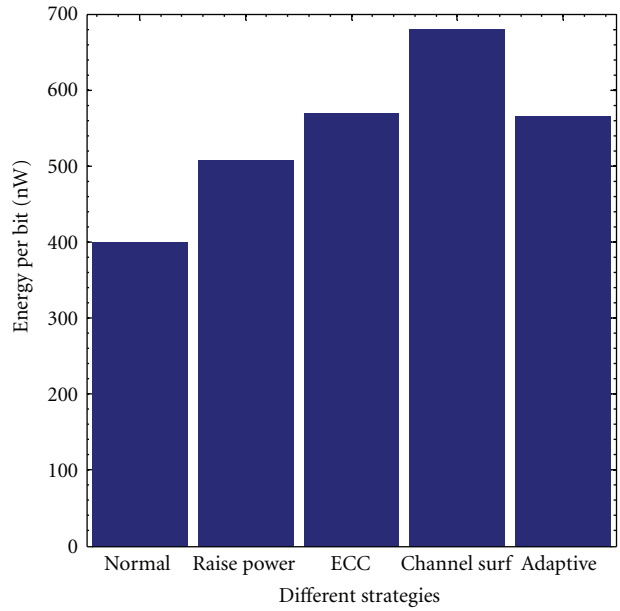

Figure 4: Energy consumption of different strategies.

6.3. Energy Efficiency. Next, we investigate the energy efficiency of different strategies. Figure 4 shows the energy consumption of the nodes. We evaluate the energy efficiency of the antijamming strategies by measuring the energy consumed by information bits excluding notification packets and coding redundancy. In the figure, we can find that the computed strategy by our algorithm expends 20% less energy

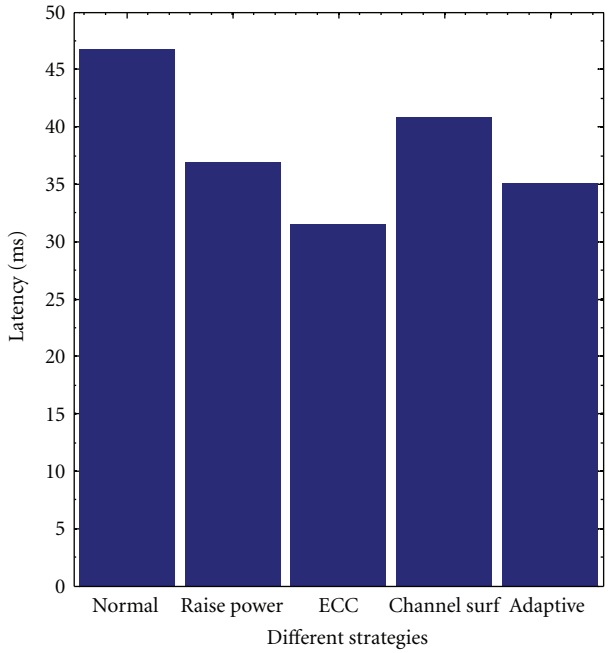

Figure 5: Latency performance of different strategies.

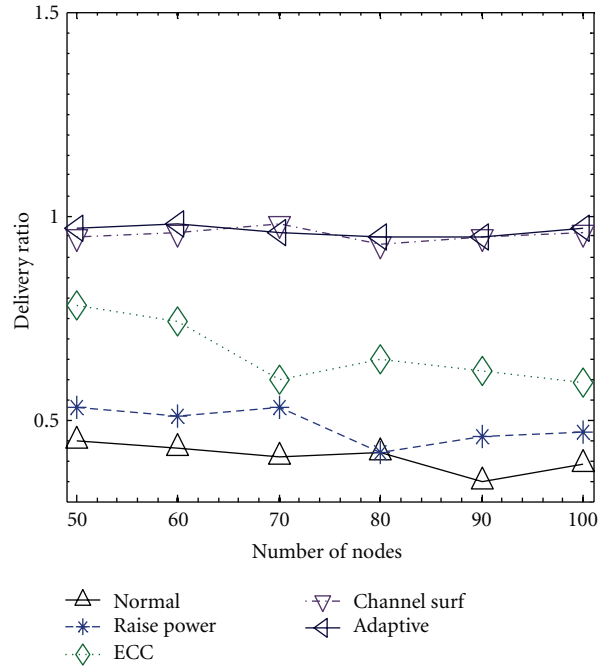

Figure 6: Delivery ratio verses number of nodes.

than the channel surfing strategy which is the most effective strategy.

6.4. Latency. We then study the latency performance of different antijamming strategies. The latency of a packet transmission is measured as the time between the instant of time when the node starts contending for the channel and the instant when the acknowledgement is received. In Figure 5, we can see that our adaptive antijamming strategy introduces a latency slightly longer than that of ECC. The channel surfing strategy introduces a much longer latency because it takes a long time for the nodes to change the communication channel. When no antijamming strategy is used in the sensor network, the latency is long because it needed many retransmissions before the packet is successfully delivered.

6.5. Scalability. We finally investigate the scalability of different antijamming strategies. In Figure 6, we show the packet delivery ratio when the number of nodes is varied from 50 to 100. We can see that as the number of nodes increases in the network, the delivery ratio of each antijamming scheme slightly drops since there is higher contention for accessing the media. However, we can see that the delivery ratio performance of our adaptive scheme only has a modest drop in packet delivery ratio when there are more sensor nodes in the network. This shows that our adaptive scheme is scalable to the increasing scale of the network.

7. Conclusion

We have presented the algorithm for selecting the best antijamming strategy for a sensor network, in which different sensor nodes may experience different degrees of jamming attacks. We propose an approach for combining the strength of several jamming countermeasures and allow a sensor node to adopt the best antijamming technique. Sensor nodes in the sensor network can adaptively change their antijamming methods as the jamming condition changes over time. The comprehensive simulation experiments have demonstrated that our algorithm achieves good performance in terms of successful delivery rate and at the meanwhile consumes slightly more energy.

Acknowledgments

This research is supported in part by MIIT of China (2009ZX03006-001-01), Shanghai Pu Jiang Talents Program (10PJ1405800), Shanghai Chen Guang Program (10CG11), NSFC (no. 61170238, 60903190, 61027009, 60970106, and 61170237), Doctoral Fund of Ministry of Education of China (20100073120021), 863 Program (2009AA012201 and 2011AA010500), HP IRP (CW267311), SJTU SMC Project (201120), and Program for Changjiang Scholars and Innovative Research Team in Universities of China (IRT1158, PCSIRT).

References

[1] Y. Liu, Y. He, M. Li et al., "Does wireless sensor network scale? A measurement study on GreenOrbs," in *Proceedings of the IEEE INFOCOM*, pp. 873–881, April 2011.

[2] M. Li, Y. Liu, and L. Chen, "Non-threshold based event detection for 3D environment monitoring in sensor networks," in *Proceedings of the IEEE ICDCS*, 2008.

[3] Y. W. Law, M. Palaniswami, L. V. Hoesel, J. Doumen, P. Hartel, and P. Havinga, "Energy-efficient link-layer jamming

attacks against wireless sensor network MAC protocols," *ACM Transactions on Sensor Networks*, vol. 5, no. 1, pp. 1–38, 2009.

[4] R. Negi and A. Perrig, "Jamming analysis of MAC protocols," Tech. Rep., 2003.

[5] W. Xu, W. Trappe, and Y. Zhang, "Channel surfing: defending wireless sensor networks from interference," in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 499–508, April 2007.

[6] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A survey on jamming attacks and countermeasures in WSNs," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 4, pp. 42–56, 2009.

[7] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *IEEE Network*, vol. 20, no. 3, pp. 41–47, 2006.

[8] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '05)*, pp. 46–57, May 2005.

[9] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attacks and network defense policies in wireless sensor networks," in *Proceedings of the 26th IEEE International Conference on Computer Communications (IEEE INFOCOM '07)*, pp. 1307–1315, May 2007.

[10] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, "Reactive jamming in wireless networks: how realistic is the threat?" in *Proceedings of the ACM WiSec*, 2011.

[11] G. Balakrishnan, Y. Mei, J. Yingtao, and K. Yoohwan, "Performance analysis of error control codes for wireless sensor networks," in *Proceedings of the 4th International Conference on Information Technology-New Generations (ITNG '07)*, pp. 876–879, April 2007.

[12] W. Xu, "On adjusting power to defend wireless networks from jamming," in *Proceedings of the 1st Workshop on the Security and Privacy of Emerging Ubiquitous Communication Systems*, pp. 1–6, 2007.

[13] K. Siddhabathula, "Fast jamming detection in wireless sensor networks," University of Texas, 2011.

[14] X. Jiang, W. Hu, S. Zhu, and G. Cao, "Compromise-resilient anti-jamming for wireless sensor networks," *Information and Communications Security*, vol. 6476, pp. 140–154, 2010.

[15] W. Xu, T. Wood, W. Trappe, and Y. Zhang, "Channel surfing and spatial retreats: defenses against wireless denial of service," in *Proceedings of the ACM Workshop on Wireless Security (WiSe '04)*, pp. 80–89, October 2004.

[16] W. Xu, W. Trappe, and Y. Zhang, "Defending wireless sensor networks from radio interference through channel adaptation," *ACM Transactions on Sensor Networks*, vol. 4, no. 4, article 18, 2008.

[17] A. D. Wood, J. A. Stankovic, and S. H. Son, "JAM: a jammed-area Mapping service for sensor networks," in *Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS '03)*, pp. 286–297, December 2003.

[18] P. Lettieri, C. Fragouli, and M. B. Srivastava, "Low power error control for wireless links," in *Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '97)*, pp. 139–150, September 1997.