Research Article

Extracting Target Detection Knowledge Based on Spatiotemporal Information in Wireless Sensor Networks

Tian Wang, 1 Zhen Peng, 1 Cheng Wang, 1 Yiqiao Cai, 1 Yonghong Chen, 1 Hui Tian, 1 Junbin Liang, 2 and Bineng Zhong 1

1 College of Computer Science and Technology, Huaqiao University, Xiamen, Fujian 361021, China
2 School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi 530004, China

Correspondence should be addressed to Tian Wang; cs_tianwang@163.com

Received 18 November 2015; Accepted 24 January 2016

Academic Editor: Zhipeng Cai

Copyright © 2016 Tian Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks (WSNs) have been deployed for many applications of target detection, such as intrusion detection and wildlife protection. In these applications, the first step is to detect whether the target is present or not. However, most of the existing work uses the “simple disk model” as signal model, which may not capture the sensing environment. In this work, we utilize a more realistic signal model to describe sensing process of sensors. On the other hand, the “majority rule” is widely used to make the final decision, which may not obtain the true judgment. To this end, we utilize a more realistic signal model and also use a probabilistic decision model to make the final decision. Moreover, we propose a probabilistic detection algorithm in which all sensors’ local measurement values are fully used. This algorithm does not need any artificial threshold compared with traditional algorithms. It makes the most of spatiotemporal information to obtain the final decision. For the spatial perspective, sensors are distributed in different locations cooperating with each other. Meanwhile, for the temporal perspective, multitround subdecisions are fused. The effectiveness of the proposed method is validated by extensive simulation results, which show high detection probabilities and low false alarm probabilities.

1. Introduction

Wireless sensor networks (WSNs) generally consist of numerous inexpensive nodes. These nodes are common microelectromechanical systems (MEMSs) equipped with various kinds of sensors, which can sense the physical events and collect the environmental information [1]. Nowadays, WSNs have a lot of applications, such as target tracking [2], object localization [3], and data transmission [4]. Among them, target detection has been a typical application of WSNs.

A typical target detection scenario usually consists of an area of interest where many sensors are deployed. In this area, it is important to determine whether a specific target/event is present or not. Therefore, the goal of target detection is to accurately detect the presence of the target, namely, to achieve high detection probability [5] and low false alarm probability [6]. To this end, sensors in the area of interest usually need to complete two things: (1) sensing the environment situation and (2) determining whether the target is present or not.

For a single sensor, sensing the environment situation is largely related to its sensing ability. Generally, sensing radius is an essential character to measure the sensing ability. In some literatures, many researchers suppose the sensing area as a disk that if and only if the distance from the target to the sensor is within the disk radius then the target will be detected [7]. This kind of detection model is called a disk model, as the sensing area of a sensor is like a disk around it [8]. However, the disk model is too idealistic. First, since the signal emitted from the target is continuously attenuated with the increase in distance, it is hard to choose the appropriate radius to get a binary decision. Second, as there are background noises, the sensing radius should not be the same. In this paper, we utilize a more realistic signal attenuation model to describe the sensing process. In this model, the signal measured by sensors continuously attenuates with the increase in distance and the influence of noise is also considered. Therefore, signal measurements of sensors can be realistically modeled rather than a simple binary value.
Moreover, although a single sensor can detect whether a target is present in its vicinity, most of the existing methods aggregate several sensors' decisions to improve the detection quality [9]. Generally, a single sensor’s decision is called a local decision. A fusion center in the network collects local decisions from some sensors and then makes a global/final decision about the target’s presence [10]. Therefore, how to make the final decision is a critical issue for target detection. One of the widespread fusion methods is “majority rule” [11]. As its name suggests, the majority rule selects the same decision with the major part of all local decisions. For example, if more than a half of sensors determine that a target is present, the fusion center will correspondingly determine that the target is present. However, there are a couple of problems for the majority rule. The first is how to set a threshold for all sensors. A uniform threshold is not reasonable because a faraway sensor can hardly get the same signal strength as that of the sensor close to the target. It can be described as the saying that “the absolute fairness is unfair.” Second, if most of sensors located far away from the target, the final decision is likely to be 0 even if some adjacent sensors to the target indeed detect it. This kind of situation can be called the dilemma of democracy. In our previous conference paper [12], we designed a probabilistic algorithm based on adaptive threshold for every single sensor. In this paper, we further improve the probabilistic detection method by getting rid of these measurement thresholds at all. We also extend the spatial decision algorithm to spatiotemporal decision algorithm, which can make the most of the sensing measurement values and further improve the quality of detection.

To summarize, our main contributions in this work are listed as follows:

1. We propose a more objective and adaptive method to make decision fusion, which is based on a realistic model for all measurement values of sensors.
2. We make good use of the spatiotemporal information in our algorithm to output the final decision and improve the detection accuracy.
3. The proposed method is validated by extensive simulations which are implemented in NS-2 simulator. The results demonstrate high detection probabilities and low false alarm probabilities.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the system model and the problem definition. Section 4 presents the decision model and the solution for target detection. Simulation results are demonstrated in Section 5. Section 6 concludes this paper.

2. Related Work

Target detection is an important application in WSNs and attracts a large number of scholars to conduct research in recent years. In earlier research, the disk model of sensors is widely employed, which means the sensing field of a sensor is like a disk around it and the sensor can detect a target only if the distance between them is less than the radius of the disk. Brass [13] discussed and analyzed search strategies and target detection capabilities of networks from other literatures in various models of sensors and targets, including mobile or stationary sensors and targets, under random or optimal placement, independent or globally coordinated search, and stealthy or visible sensors. It can be seen that all these detection strategies used the Boolean sensing model with a constant sensing radius. Lazos et al. [14] analytically evaluated the detection probability of mobile targets when some sensors were stochastically deployed to monitor an area of interest. They mapped the target detection problem to a line-set intersection problem and showed that the detection probability depended on the length of the perimeters of the sensing areas of the sensors but not on their shapes. Therefore, the authors employed an adjusted disk model that the sensing range of sensors can be changed within the limitation of perimeter. Medagliani et al. [15] addressed the problem of engineering energy-efficient target detection applications, taking into account some physical system parameters. They first derived an analytical framework to evaluate these system parameters, such as the probability of target detection missing, the notification transmission latency, and the network lifetime. Then, the authors showed how to optimally configure system parameters under realistic performance constraints. The sensing range of sensors can be adjusted with different environmental settings.

As WSNs for target detection get further studied, scholars gradually adopt more realistic sensing model rather than disk model according to sensors’ features. Wang et al. [16] considered the coverage problem for target detection applications in WSNs. Unlike conventional coverage problems assuming sensing regions are disks around sensors, they defined the sensing region according to detection constraints in terms of false alarm probability and missing probability and a cooperative detection scheme is proposed. However, the authors adopted the “OR rule” for final decision fusion; that is, the sink would determine that the target is present when any individual sensor detected the target locally. Zhou et al. [17] addressed the problem of detecting mobile targets with continuous movement. They classified these targets into two categories, which, respectively, are the rational targets having the knowledge of existing sensors and the blind targets whose traces are only straight lines. They investigated the detection performance and hoped to find a few critical positions to deploy additional sensors so that the freedom of targets could be limited. Therefore, their detection rule is simple that any sensor that has detected the target can indicate its presence. Tan et al. [18] exploited reactive mobility to improve the target detection performance, as mobile sensors in the network can collaborate with fixed sensors and move reactively. The detection consensus is based on a positive system decision when not less than a half of sensors in the group made their positive local decisions. Then, a sensor movement scheduling algorithm was developed to achieve near-optimal system detection performance under a given detection delay bound. Yi et al. [19] proposed a hierarchical decision fusion scheme to improve network-wide detection probability. In the proposed scheme, each sensor first makes its own local decision based on a signal attenuation model.
Then, a set of closest neighbor sensors makes a cluster-level sensor fusion to derive a new decision result. Finally, those sensors making a positive fusion result transmit their results to the network fusion center in order to make a final network-level decision. The fusion method is based on the number of positive decisions, which means that if the number of lower level positive decisions is not less than a certain threshold, the higher level decision produces a corresponding positive decision; otherwise, the higher level decision is negative. The problem for these strategies is that the final decision is always based on a certain number, such as one or a half of the total, of positive local decisions made by sensors. In fact, there are times when even only a less number of positive local decisions should lead to a positive final decision. Therefore, some more reasonably realistic indicators, rather than only a fixed number, should be set as the benchmark for making decisions.

The fusion strategy used in above literatures is called decision fusion where nodes send their local decision results to a fusion center to make a final decision [20]. Moreover, there is another kind of fusion strategy called value fusion where nodes just send their raw energy measurements to the fusion center which then makes a final decision according to its received measurements from different nodes [21]. Yuan et al. [22] presented fast sensor placement algorithms based on a probabilistic data fusion model. Through these algorithms, sensors are placed at optimal locations to achieve maximum detection performance. In the data fusion scheme, sensors send their energy measurements to the cluster head, which in turn compares the average of all measurements against a threshold. If the average is greater than the threshold, the cluster head determines that a target is present. Otherwise, it determines there is no target. The same author Yuan [23] also proposed another data fusion based collaborative detection scheme for achieving guaranteed accuracy for sensor deployment. A “density first” clustering algorithm was adopted to organize preselected surveillance locations into deployment units and to place sensors to cover these locations. The same scheme data fusion was employed in the algorithm. Yang and Qiao [24] studied barrier information coverage problem. They exploited collaborations and information fusion among neighboring sensors for intrusion detection. The objective is to reduce the number of active sensors, which are required to cover a barrier. They proposed a practical solution to identify the coverage set which contains only a small number of active sensors, and a virtual sensor can incorporate the sensed readings of sensors to make a value-fusion-based decision. The problem of these value fusion methods is how to determine the value of the threshold. Most of the authors of this kind of methods did not discuss the problem and leave the threshold as an experiential value.

In this paper, sensors detect the target through a realistic attenuation signal model and can make a more reasonable decision about the presence of target based on a probabilistic decision model. In this new decision model, sensors do not need any measurement threshold at all. Therefore, the fusion center can objectively estimate the presence probability of the target and then make a final decision through a multiround method.

3. System and Problem Statements

3.1. Detection Model. We use the detection model based on the detection probability similar to literatures [18, 25]. The target itself emits a signal, for example, acoustic signal. A sensor can detect the target by measuring its energy. Meanwhile, the measurements of sensors are contaminated by background noise which is modeled as additive Gaussian noise. Assume that \( H_0 \) represents the hypothesis that the target is absent and that \( H_1 \) represents the other hypothesis that the target is present. Thus the energy measurement \( e_i \) of sensor \( i \) is given by

\[
H_0: e_i = e_n^i, \quad (1a)
\]
\[
H_1: e_i = e_s(d_i) + e_n^i. \quad (1b)
\]

Among them, \( e_n \) is the energy of noise to which sensor \( i \) is exposed, \( e_s(d_i) \) is the attenuated signal energy at the position of sensor \( i \), and \( d_i \) is the Euclidean distance between sensor \( i \) and the target. \( e_s(d_i) \) attenuates with the increasing of \( d_i \) and can be expressed as

\[
e_s(d_i) = \begin{cases} \frac{S_0}{(d_i/d_0)^k}, & \text{if } d_i > d_0 \\ S_0, & \text{if } d_i \leq d_0. \end{cases} \quad (2)
\]

Among them, \( d_0 \) is a constant as a reference factor, and \( S_0 \) is the signal energy measured within the distance \( d_0 \) to the source. \( k \) is an attenuation factor which is typically from 2 to 5. The noise energy \( e_n \) approximately satisfies the Gaussian distribution with mean equal to \( \mu \) and variance equal to \( \sigma^2 \). Therefore, the total signal energy value measured by sensor \( i \) follows a Gaussian distribution, which is given by

\[
H_0: e_i \sim \mathcal{N}(\mu, \sigma^2), \quad (3a)
\]
\[
H_1: e_i \sim \mathcal{N}(\mu + e_s(d_i), \sigma^2). \quad (3b)
\]

3.2. Network Model and Detection Problem

3.2.1. Network Model. The sensor network consists of a number of fixed sensor nodes, which are randomly deployed in a 2-dimensional area of interest. Assume that all sensors are homogeneous, which means they sense the same type of signal from the target based on the same signal model. After deployment, these sensors are stable, for their positions cannot be changed. All sensors know their own positions and have synchronized clocks [26]. Moreover, we assume that targets appear at some particular physical locations in this area with certain probabilities. These locations are referred to as target spots. The locations of target spots are known for two reasons: first, for most situations, the intruders, as targets, only appear in some specific locations such as the entrance or door [27]; second, after operating for a certain amount of time, the sensor network can identify some target spots where targets are likely to appear based on detection history [28]. After target spots are identified, the network is organized into clusters that each cluster monitors a target spot.
The clusters can be dynamically formed [29] when the network is initialized or when the target spots are changed, and one fixed sensor belongs to only one cluster. The cluster head can collect information from sensors belonging to the cluster [30]. We note that any suitable clustering protocol can be utilized here and no particular clustering method will be discussed in this paper. In fact, as every cluster performs detection separately with identical process, we narrow down our discussion to one cluster and target spot hereafter.

3.2.2. Problem Definition. There are $N$ sensors randomly deployed in a 2-dimensional $L \times L$ plane detection area. The target appears or disappears at the target spot and its signal sensed by sensors is modeled in Section 3.1. These sensors are required to detect the target when it is present. If the final decision made by sensors is 1 when the target is present, this kind of decision is called a true alarm. On the contrary, if the final decision is 1 when the target is actually absent, this kind of decision is called a false alarm. Then, the detection probability is the time of true alarm to the total time of target’s presence, and the false alarm probability is the time of false alarm to the total time of target’s absence. Therefore, our goal is to develop a method to improve the detection accuracy as much as possible, that is, to improve the detection probability and reduce the false alarm probability.

4. Target Detection Algorithm Based on Probabilistic Methods

4.1. Probabilistic Decision Model. This section introduces the method for detecting the target. According to (3a) and (3b), the probability density function (PDF) of getting the measurement $e_i$ for sensor $i$ is expressed as

$$H_0: p_f(e_i) = \frac{1}{\sqrt{2\pi \sigma^2}} \exp \left(-\frac{(e_i - \mu)^2}{2\sigma^2}\right), \quad (4a)$$

$$H_1: p_d(e_i) = \frac{1}{\sqrt{2\pi \sigma^2}} \exp \left(-\frac{(e_i - \mu - e_x(d_i))^2}{2\sigma^2}\right). \quad (4b)$$

At first, the local decisions are generated by every single sensor. Traditionally, the decision rule is based on Likelihood Ratio Test [31, 32]. That is, according to the measured value in Section 3.1, an individual sensor compares its signal energy measurement with a threshold $\lambda$. If it is greater than $\lambda$, the sensor determines 1 which means it detects the target; otherwise it determines 0 which means it does not. Generally, the probability of making a positive decision when the target is actually absent is called the false alarm probability, and the probability of making a positive decision when the target is indeed present is called the detection probability [33]. Let $P_F^i$ and $P_D^i$, respectively, represent the local false alarm probability and the detection probability of sensor $i$. According to (4a) and (4b), they can be expressed as

$$P_F^i = \Pr(e_i \geq \lambda \mid H_0) = \int_{\lambda}^{\infty} p_f(e_i) \, de_i,$$

$$P_D^i = \Pr(e_i \geq \lambda \mid H_1) = \int_{\lambda}^{\infty} p_d(e_i) \, de_i,$$  

where $\Pr(\cdot)$ is the probability function. It shows that sensors closer to the target can lead to higher detection probability.

In order to make a final decision [34], a fusion center collects all local decisions in order to make a fusion. In general, one of the widespread fusion methods is the majority rule, which is simple and intuitive. For majority rule, if more than a half of sensors vote 1, the cluster head determines 1 which means the target is present; otherwise, the cluster head determines 0 to indicate there is no target. However, this decision rule as well as fusion method has several problems. First, the decision rule just output a binary result, either 1 or 0, based on a usually uniform threshold $\lambda$ for all sensors. However, the value of the threshold is hard to be consistently reasonable. For example, the measurement of a faraway sensor from the target is much more difficult to be enough to trigger the alarm, so it is unfair for these sensors to have the same threshold as that of those closer ones. In fact, the sensed signal is a continuous value. Even this value is less than $\lambda$, and important information may be implied in the value. For example, a strong signal means a higher presence probability of event than that of weak signal. Second, the majority-rule-based fusion method is inclined to miss a target. Generally, the closer the sensor to a target, the higher the detecting probability that it can achieve and vice versa. If the sensors far from the target make up the vast majority of all detecting sensors, they are likely to dismiss the presence of the target and then the final decision is also wrong. For example, as Figure 1 shows, there are 6 sensors to participate in detecting. When the target appears, sensors $a$, $b$, $c$, and $d$ make decision 0 while sensors $f$ and $e$ make decision 1. But with the majority rule the final fusion result is “0,” which is an obvious mistake.

4.2. Probabilistic Decision Method. In the following, we introduce a new decision method. This method fully utilizes the signal value sensed by sensors and extracts the useful information to make decisions. Let $A$ stand for the event that the target is present. And $B_i$ denotes events that sensor $i$ gets its present measurement value $e_i$, where $i = 1, 2, \ldots, n$ as there are $n$ sensors. We define an estimation probability $Q_0$ which stands for the probability for these $n$ sensors to get their own measurements under the hypothesis $H_0$ that the target is absent. Similarly, we define $Q_1$ standing for the probability for sensors to get their own measurements under the hypothesis $H_1$ that the target is present. Therefore, $Q_0$ and $Q_1$ can be expressed, respectively, as

$$Q_0 = P\left(B_1 \cap B_2 \cap \cdots \cap B_n \mid \overline{A}\right),$$  

$$Q_1 = P\left(B_1 \cap B_2 \cap \cdots \cap B_n \mid A\right).$$  

Rather than determining a measurement threshold to select only a certain part of sensors, we would like to take advantage of all sensors’ measurements to make the final decision. Under above definitions, if sensors get a set of measurements and the corresponding $Q_0$ is greater than $Q_1$, it is reasonable to believe that the target is absent in this time. On the contrary, if $Q_1$ turns to be greater than $Q_0$, it should be
determined that the target is present. Therefore, we define a testing function for global decision as

\[
T(Q_0, Q_1) = \begin{cases} 
0, & \text{if } Q_0 > Q_1 \\
1, & \text{if } Q_0 \leq Q_1,
\end{cases}
\]  

(7)

where 0 determines the decision that the target is absent and 1 determines that it is present.

Since events \( B_1, B_2, \ldots, B_n \) are independent, \( P(B_1 \cap B_2 \cap \cdots \cap B_n | A) \) and \( P(B_1 \cap B_2 \cap \cdots \cap B_n | \overline{A}) \) can be represented by \( \prod_{i=1}^{n} P(B_i | A) \) and \( \prod_{i=1}^{n} P(B_i | \overline{A}) \), respectively. The problem is how to calculate \( P(B_i | A) \) or \( P(B_i | \overline{A}) \). As the measurement value for sensor \( i \) follows the Gaussian distribution defined in formulas (3a) and (3b), the value of \( P(B_i | A) \) or \( P(B_i | \overline{A}) \) is equal to the product of its corresponding probability density and a tiny range \( \varepsilon \) of \( e_i \), which can be expressed as

\[
P(B_i | A) = p_f(e_i) \varepsilon, \quad (8a)
\]

\[
P(B_i | \overline{A}) = p_d(e_i) \varepsilon. \quad (8b)
\]

If we want to get the exact probability of event \( B_i \) under the situation of event \( A \), it is very hard to determine the size of \( \varepsilon \). Fortunately, we only need to compare between the sizes of \( Q_0 \) and \( Q_1 \) where \( \varepsilon \) can be lost from both sides of the inequation. Therefore, the testing function can be expressed as

\[
T(Q_0, Q_1) = \begin{cases} 
0, & \text{if } \prod_{i=1}^{n} P(B_i | \overline{A}) > \prod_{i=1}^{n} P(B_i | A) \\
1, & \text{if } \prod_{i=1}^{n} P(B_i | \overline{A}) \leq \prod_{i=1}^{n} P(B_i | A).
\end{cases}
\]  

(9)

Therefore, we present the global decision making algorithm in Algorithm 1.

Moreover, we set out to further improve the accuracy of final decision by using multiple decisions among time dimension. In common, when a target is present, it will keep the same situation for some time rather than disappearing immediately. As a sensor has a comparatively much shorter working cycle and high sensing frequency [35], it can make multiround local decisions during this time. For those local decisions in every single round, a round decision can be produced to determine whether the target is present according to our probability-based algorithm. After that, a final decision can be made from these round decisions. The whole process is showed in Figure 2. During this process, the majority rule can be reutilized that if more than a half of round decisions are 1, the final decision will be 1 to determine that the target is present; otherwise, if more than a half of round decisions are 0, then the final decision will be 0 as well. Based on this temporal decision making method, the detection accuracy can be further improved, especially for reducing false alarms.

Therefore, we propose our target detection algorithm presented as Algorithm 2. When the network is initialized or when the target spot is changed, sensors around the target spot form a cluster and a cluster head \( C_H \) is selected as the fusion center. The network will use a default number as total decision rounds. In every round, the cluster head \( C_H \) makes a single round decision based on Algorithm 1. All decision results will be accumulated during the whole multiround process. After that, a final decision is concluded by \( C_H \). If more than a half of round decisions are 1, the final decision will be 1; otherwise, it will be 0. The number of multidecision rounds can be modified according to practical situations. In general, the total decision rounds can be set as a small odd
**Input:** local probability densities \( P(B_i | A) \) and \( P(B_i | \bar{A}) \), \( i = 1, 2, \ldots, n \); the number of total decision rounds \( T_R \).

**Output:** a final decision 0 or 1.

1. Sensors around the target spot form a cluster;
2. A cluster head \( C_H \) is selected; //\( C_H \) acts as a fusion center
3. Threshold of decision \( T_D := \left\lceil T_R / 2 \right\rceil + 1; //\text{Threshold for positive round decision}
4. Positive counter \( pc := 0; //\text{Record the number of positive round decisions}
5. for decision round iterator \( ri := 1 \) to \( T_R \) do
6. \( C_H \) gets the single round decision that
   \( g = \text{RoundDecision}(); \)
   //Which means a positive round decision
7. \( pc := pc + 1; \) //Record the number of positive round decisions
8. if \( pc \geq T_D \) then
   //Which means a positive final decision
9. break;
10. endif
11. end for
12. if \( pc \geq T_D \) then
13. \( C_H \) output the final decision 1;
14. else
15. \( C_H \) outputs the final decision 0;
16. end if

**Algorithm 2:** Probabilistic detection algorithm with spatiotemporal information.

4.3. Algorithm Analysis. It can be analyzed that the time complexity of Algorithm 1 is \( O(n) \) as there are \( n \) sensors. Then the time complexity of Algorithm 2 is \( O(T_R n) \). Because the value \( T_R \) is a constant, the final complexity of Algorithm 2 is \( O(n) \) as well. For limiting the local false alarm effectively, a property can be showed as follows.

**Theorem 1.** For sensor \( i \), if its attenuated signal energy \( e_s(d_i) \) from the target can be greater than 6 times the noise variance \( \sigma \), its local false alarm probability can be guaranteed not larger than about 0.1% and its local detection probability can be guaranteed not less than about 99.9%.

**Proof.** For sensor \( i \), it will make local decision 1 only if \( P(B_i | A) \geq P(B_i | \bar{A}) \) according to Algorithm 1. Figure 3 plots probability density functions (PDFs) of \( p_f(e_i) \) and \( p_d(e_i) \). Assume that when \( e_i \) is \( x \), those two curves intersect. When the target is absent, only if \( e_i \) is larger than \( x \) can sensor \( i \) make decision 1, which is a false alarm obviously. Then the local false alarm probability can be defined as the probability that \( e_i \) is larger than \( x \). According to the property of Gaussian distribution, when the target is absent, \( \text{Pr}(e_i > x) \) will be less than about 0.1% if \( x \) is larger than \( \mu + 3\sigma \) [36], where \( \mu \) and \( \sigma \) are the mean and the standard deviation of the noise, respectively. On the contrary, if the target is present, \( \text{Pr}(e_i > x) \) will be greater than about 99.9% under the same condition of \( x \). Then according to the relation of these two (PDFs), the relation between \( x \) and \( e_s(d_i) \) is

\[
\mu + e_s(d_i) - \mu = 2(x - \mu),
\]

\[
x = \frac{\sigma}{2} + \mu.
\]
Table 1: Simulation parameters.

| Parameters                        | Values  |
|-----------------------------------|---------|
| Simulation times                  | 100     |
| Area size (m$^2$)                 | $30 \times 30$ |
| Number of sensors $n$             | 5       |
| Probability of target’s presence  | 5%      |
| Maximum target signal energy $S_0$ (dB) | 6     |
| Distance landmark for target signal energy $d_0$ (m) | 5      |
| Attenuation factor $k$            | 2       |
| Noise mean $\mu$ (dB)             | 0.2     |
| Noise variance $\sigma^2$ (dB$^2$) | 0.6    |

![Figure 3: An illustration of probability density functions.](image)

Setting $x > \mu + 3\sigma$ to limit the local false alarm probability less than about 0.1%, it turns out to be

$$x = \frac{e_x}{2} + \mu > \mu + 3\sigma, \quad e_x > 6\sigma.$$  

(11)

As the derivation process can be backwards equivalently, the proposition can be proved.

5. Performance Evaluation

In order to verify the effectiveness of our proposed methods, we conducted simulations with NS-2. In the simulation scenario, sensors are randomly deployed in an area of interest, which is $30 \times 30$ m$^2$. Some general parameters are listed in Table 1. The probability of target’s presence is set as 5%, and $S_0$ is 5 dB with $d_0$ as 5 m. For noise, $\mu = 0.01$ dB and $\sigma^2$ is equal to $2\mu$. For comparisons, we conducted two other target detection algorithms. The first is majority-rule-based algorithm, whose fusion decision rule is the majority rule. The second is the method without fusion; that is, the closest sensor to the target makes the final decision the same as its local decision. In our simulation, PD refers to our proposed algorithm, MD represents the majority-rule-based algorithm, and CD stands for the closest-based one.

Figure 4 showed the system detection probabilities achieved by PD, MD, and CD, respectively, when the number of sensors increased from 1 to 10. It could be seen that PD achieved the best performance among these three algorithms especially when there are fewer sensors. When the number of sensors increased, PD gradually achieved 100% in detection probability. Meanwhile, the performance of MD tended upwards with fluctuation. The reason is that when the number of sensors increases from odd number to even number, the number threshold actually does not increase. On the contrary, if the number increases from even to odd, the threshold will also increase. This shows that the setting of threshold can have a large impact on the performance of MD. For CD method, its detection probability also increased with the increase of the number of sensors. The reason is that it can have more opportunity to choose a better sensor when the number of sensors increases, and it can achieve a high detection probability when the number is large enough.

In Figure 5, the results of false alarm probability of these three algorithms were presented. It could be seen that PD achieved a relatively low false alarm probability. When the number of sensors increased, its false alarm probability decreased to almost 0. MD also tended downwards with fluctuation due to the discretely changed value of thresholds. For CD, its false alarm probability almost kept the same. The reason is that its probability is equal to the closest sensor’s local false alarm probability which is determined by the property of noise. As the number of sensors cannot influence the noise energy, its false alarm probability will not change in theory.

In Figure 7, it could be seen that noise variance also influenced the false alarm probability made by the three algorithms. Among them, CD had the largest increase which increased about 400% when the variance increased from 0.2
to 1.0. For MD, its false alarm probability did not increase in such a degree but also obviously. On the contrary, the performance of PD was not influenced much and also kept close to 0. The reason is that the use of multiround decision method can limit the false alarm probability effectively.

Figure 8 showed how the system detection probabilities were achieved by PD, MD, and CD, respectively, when the value of noise mean increased from 0.1 to 0.3. It could be seen that PD achieved the best performance among these three algorithms all the time. It may be strange to see that the performance of MD and CD was even better when the noise mean increased. However, this phenomenon is normal because a higher noise will make measurement values more easily reach the threshold, which makes sensors believe that the target is present. However, those decisions are likely to be false alarms as showed in Figure 9. In Figure 9, MD and CD tended to have higher false alarm probability when the noise mean increased. Therefore, the increase in the detection probability for these two algorithms in Figure 8 is just superficial phenomenon due to the larger noise cheating them. This can reflect the advantage of PD in some respects.

Figure 10 showed how PD, MD, and CD performed when sensors were limited to be deployed among a certain range of distance to the target. $d$ in it represents the landmark distance $d_{0}$. When sensors were deployed farther away from the target, the performance of these three algorithms became worse to more or less extent. PD still performed the best among them. When sensors were in $3d$ to $2d$ from the target, the detection probability of PD outperformed that of MD and PD by at least 60%. This shows that the performance of MD and CD depends much on the deployment situation of sensors in the network, which may be a disadvantage in some scenarios.
In Figure 11, it could be seen that the distance to the target could influence the false alarm probability of these three algorithms to different extent. PD and MD both achieved a higher false alarm probability when the distance increased. However, CD did not get influenced because its false alarm probability depends on the state of noise.

Figure 12 showed how PD performed with 3 sensors when the number of total rounds in our algorithm increased from 1 to 9 where only odd numbers were considered. As MD and CD do not have this spatiotemporal process, they were not included in this simulation. As shown in the figure, PD achieved better performance when the total rounds increased that its detection probability increased and its false alarm probability decreased. The results showed that the spatiotemporal information can truly improve the accuracy of final decisions because multiround decision can eliminate interference from noise and make the final decision more reasonable.

6. Conclusion

In this paper, we introduced our improvement for target detection solutions in sensing and fusion aspects. First, rather than using a simple disk model, we utilized a more realistic signal model to describe the sensing environment. After that, we analyzed the defects of the majority rule which is widely used in decision fusion methods. We proposed a probabilistic detection method to make the temporary decision in a single round. As the probability density of measurements values is taken advantage of, no thresholds for measurement values of sensors are needed any more and all sensors can contribute to the decision making process. Moreover, we proposed an algorithm which makes the most of the spatiotemporal information to detect target more
accurately. For the spatial perspective, sensors are distributed in different locations and cooperate with each other. For the temporal perspective, decisions of several rounds are fused to output the final decision so that the detection accuracy can be improved further. Simulation results showed that our proposed algorithm can achieve a high detection probability and low false alarm probability even with interference of noise.

**Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

**Acknowledgments**

The above work was supported in part by grants from the National Natural Science Foundation (NSF) of China under Grant nos. 61572206, 61202468, 61370007, 61302094, 61305085, 61562005, 61572205, U1536115, and 51305142 and the Natural Science Foundation of Fujian Province of China under Grant nos. 2014J01240 and 2015J01257 and Promotion Program for Young and Middle-Aged Teacher in Science and Technology Research of Huaqiao University under Grant no. ZQN-PY308.

**References**

[1] L. M. L. Oliveira and J. J. P. C. Rodrigues, “Wireless sensor networks: a survey on environmental monitoring,” *Journal of Communications*, vol. 6, no. 2, pp. 143–151, 2011.

[2] Z. He, Z. Cai, S. Cheng, and X. Wang, “Approximate aggregation for tracking quantiles and range countings in wireless sensor networks,” *Theoretical Computer Science*, vol. 607, part 3, pp. 381–390, 2015.

[3] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, and Q. Liang, “Sparse target counting and localization in sensor networks based on compressive sensing,” in *Proceedings of the IEEE INFOCOM*, pp. 2255–2263, IEEE, Shanghai, China, April 2011.

[4] O. D. Incel, A. Ghosh, B. Krishnamachari, and K. Chintalapudi, “Fast data collection in tree-based wireless sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 1, pp. 86–99, 2012.

[5] Y. Xiao, H. Chen, K. Wu et al., “Coverage and detection of a randomized scheduling algorithm in wireless sensor networks,” *IEEE Transactions on Computers*, vol. 59, no. 4, pp. 507–521, 2010.

[6] M. Zhu, S. Ding, Q. Wu, R. R. Brooks, N. S. V. Rao, and S. S. Iyengar, “Fusion of threshold rules for target detection in wireless sensor networks,” *ACM Transactions on Sensor Networks*, vol. 6, no. 2, article 18, 2010.

[7] J. Gao, J. Li, Z. Cai, and H. Gao, “Composite event coverage in wireless sensor networks with heterogeneous sensors,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM ’15)*, pp. 217–225, IEEE, April 2015.

[8] B. Liu, O. Dousse, P. Nain, and D. Towsley, “Dynamic coverage of mobile sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 2, pp. 301–311, 2013.

[9] S. Cheng, Z. Cai, J. Li, and X. Fang, “Drawing dominant dataset from big sensory data in wireless sensor networks,” in *Proceedings of the 34th Annual IEEE International Conference on Computer Communications (INFOCOM ’15)*, pp. 531–539, IEEE, Hong Kong, April 2015.

[10] S. Maleki, A. Pandharipande, and G. Leus, “Energy-efficient distributed spectrum sensing for cognitive sensor networks,” *IEEE Sensors Journal*, vol. 11, no. 3, pp. 565–573, 2011.

[11] S. Atapattu, C. Tellambura, and H. Jiang, “Energy detection based cooperative spectrum sensing in cognitive radio networks,” *IEEE Transactions on Wireless Communications*, vol. 10, no. 4, pp. 1232–1241, 2011.

[12] T. Wang, Z. Peng, J. Liang et al., “Detecting targets based on a realistic detection and decision model in wireless sensor networks,” in *Wireless Algorithms, Systems, and Applications*, vol. 9204 of *Lecture Notes in Computer Science*, pp. 836–844, Springer, Berlin, Germany, 2015.

[13] P. Brass, “Bounds on coverage and target detection capabilities for models of networks of mobile sensors,” *ACM Transactions on Sensor Networks*, vol. 3, no. 2, Article ID 1240229, 2007.

[14] L. Lazos, R. Poovendran, and J. A. Ritcey, “Probabilistic detection of mobile targets in heterogeneous sensor networks,” in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN ’07)*, pp. 519–528, ACM, Cambridge, Mass, USA, April 2007.

[15] P. Medagliani, J. Leguay, G. Ferrari, V. Gay, and M. Lopez-Ramos, “Energy-efficient mobile target detection in wireless sensor networks with random node deployment and partial coverage,” *Pervasive and Mobile Computing*, vol. 8, no. 3, pp. 429–447, 2012.

[16] W. Wang, V. Srinivasan, K.-C. Chua, and B. Wang, “Energy-efficient coverage for target detection in wireless sensor networks,” in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN ’07)*, pp. 313–322, ACM, Cambridge, Mass, USA, April 2007.

[17] S. Zhou, M.-Y. Wu, and W. Shu, “Improving mobile target detection on randomly deployed sensor networks,” *International Journal of Sensor Networks*, vol. 6, no. 2, pp. 115–128, 2009.

[18] R. Tan, G. Xing, J. Wang, and H. C. So, “Exploiting reactive mobility for collaborative target detection in wireless sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 3, pp. 317–332, 2010.

[19] L. Li, Y. Deng, D. Ding, Z. Zou, and W. Wang, “Dcdf: a distributed clustered decision fusion scheme for target detection in wireless sensor networks,” in *Proceedings of the IEEE 17th International Conference on Computational Science and Engineering (CSE ’14)*, pp. 483–486, IEEE, Chengdu, China, December 2014.

[20] E. C. Y. Peh, Y.-C. Liang, Y. L. Guan, and Y. Zeng, “Cooperative spectrum sensing in cognitive radio networks with weighted decision fusion schemes,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 12, pp. 3838–3847, 2010.

[21] C.-T. Cheng, H. Leung, and P. Maupin, “A delay-aware network structure for wireless sensor networks with in-network data fusion,” *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1622–1631, 2013.

[22] Z. Yuan, R. Tan, G. Xing, C. Lu, Y. Chen, and J. Wang, “Fast sensor placement algorithms for fusion-based target detection,” in *Proceedings of the Real-Time Systems Symposium (RTSS ’08)*, pp. 103–112, Barcelona, Spain, November 2008.

[23] Z. Yuan and G. Wang, “Sensor deployment strategy for collaborative target detection with guaranteed accuracy,” in *Proceedings of the 4th International Conference on Mobile Ad-hoc and Sensor Networks (MSN ’08)*, pp. 68–71, IEEE, Wuhan, China, December 2008.
[24] G. Yang and D. Qiao, “Barrier information coverage with wireless sensors,” in *Proceedings of the IEEE INFOCOM*, pp. 918–926, IEEE, Rio de Janeiro, Brazil, April 2009.

[25] T. Wang, Z. Peng, Y. Chen, Y. Cai, and H. Tian, “Continuous tracking for mobile targets with mobility nodes in WSNs,” in *Proceedings of the International Conference on Smart Computing (SMARTCOMP ’14)*, pp. 261–268, IEEE, Hong Kong, November 2014.

[26] L. Schenato and F. Fiorentin, “Average TimeSynch: a consensus-based protocol for clock synchronization in wireless sensor networks,” *Automatica*, vol. 47, no. 9, pp. 1878–1886, 2011.

[27] M. Al Ameen, J. Liu, and K. Kwak, “Security and privacy issues in wireless sensor networks for healthcare applications,” *Journal of Medical Systems*, vol. 36, no. 1, pp. 93–101, 2012.

[28] T.-L. Chin and Y.-H. Hu, “Optimal detector based on data fusion for wireless sensor networks,” in *Proceedings of the 54th Annual IEEE Global Telecommunications Conference (GLOBECOM ’11)*, pp. 1–5, IEEE, Houston, Tex, USA, December 2011.

[29] W.-S. Jung, K.-W. Lim, Y.-B. Ko, and S.-J. Park, “Efficient clustering-based data aggregation techniques for wireless sensor networks,” *Wireless Networks*, vol. 17, no. 5, pp. 1387–1400, 2011.

[30] Z. Cai, G. Lin, and G. Xue, “Improved approximation algorithms for the capacitated multicast routing problem,” in *Computing and Combinatorics*, pp. 136–145, Springer, 2005.

[31] J. J. Fuchs, “The generalized likelihood ratio test and the sparse representations approach,” in *Image and Signal Processing*, vol. 6134 of *Lecture Notes in Computer Science*, pp. 245–253, Springer, Berlin, Germany, 2010.

[32] P. K. Sahu, S. R. Pal, and A. K. Das, “Likelihood ratio test,” in *Estimation and Inferential Statistics*, pp. 103–129, Springer, New Delhi, India, 2015.

[33] E. Yanmaz and H. Guclu, “Stationary and mobile target detection using mobile wireless sensor networks,” in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM ’10)*, pp. 1–5, IEEE, San Diego, Calif, USA, March 2010.

[34] R. Deng, J. Chen, C. Yuen, P. Cheng, and Y. Sun, “Energy-efficient cooperative spectrum sensing by optimal scheduling in sensor-aided cognitive radio networks,” *IEEE Transactions on Vehicular Technology*, vol. 61, no. 2, pp. 716–725, 2012.

[35] S. Cheng, Z. Cai, and J. Li, “Curve query processing in wireless sensor networks,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 11, pp. 5198–5209, 2015.

[36] B. E. Fristedt and L. F. Gray, *A Modern Approach to Probability Theory*, Springer, 2013.