

# Media Access with Spatial Reuse for Cooperative Spectrum Sensing

Xiao Shao, Harry Leib

Department of Electrical and Computer Engineering, McGill University, Montreal, Canada

Email: xiao.shao2@mail.mcgill.ca, harry.leib@mcgill.ca

**How to cite this paper:** Shao, X. and Leib, H. (2017) Media Access with Spatial Reuse for Cooperative Spectrum Sensing. *Wireless Sensor Network*, 9, 205-237.  
<https://doi.org/10.4236/wsn.2017.97012>

**Received:** May 31, 2017

**Accepted:** July 28, 2017

**Published:** July 31, 2017

Copyright © 2017 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

The increasing interest for wireless communication services and scarcity of radio spectrum resources have created the need for a more flexible and efficient usage of the radio frequency bands. Cognitive Radio (CR) emerges as an important trend for a solution to this problem. Spectrum sensing is a crucial function in a CR system. Cooperative spectrum sensing can overcome fading and shadowing effects, and hence increase the reliability of primary user detection. In this paper we consider a system model of a dedicated detect-and-forward wireless sensor network (DetF WSN) for cooperative spectrum sensing with  $k$ -out-of- $n$  decision fusion in the presence of reporting channels errors. Using this model we consider the design of a spatial reuse media access control (MAC) protocol based on TDMA/OFDMA to resolve conflicts and conserve resources for intra-WSN communication. The influence of the MAC protocol on spectrum sensing performance of the WSN is a key consideration. Two design approaches, using greedy and adaptive simulated annealing (ASA) algorithms, are considered in detail. Performance results assuming a grid network in a Rician fading environment are presented for the two design approaches.

## Keywords

Cooperative Spectrum Sensing, Wireless Sensor Network, Media Access, Spatial Reuse

---

## 1. Introduction

With the rapid development of wireless technology and the growing number of innovative telecommunication services, the scarcity of radio spectrum resources has become a critical issue. Static allocation within the conventional spectrum management framework causes a bottleneck for efficient utilization of radio spectrum [1]. Relying on dynamic and opportunistic spectrum access technolo-

gies, Cognitive Radio (CR) systems offer a promising solution to spectrum shortage [2]. A typical example is the opportunistic use of the unused spectrum in TV broadcasting bands by IEEE 802.22 WRAN [3] systems. Another example is the concurrent operation of LTE and Wi-Fi in unlicensed bands via dynamic spectrum access techniques [4] [5].

An important function in CR systems, that makes possible the discovery of free channels for opportunistic use, is spectrum sensing. Single user sensing performance is often compromised by multipath fading, shadowing, and uncertainty of background noise [6]. Sensing performance, however, can be improved by using Cooperative Spectrum Sensing [7] [8]. In such a scheme local spectrum measurement are performed at each sensing node, and a final decision is made after fusing the information from all cooperating sensors. Cooperative spectrum sensing provides better spatial coverage as well as multi-user diversity gains that improves the detection reliability, and helps to reduce the sensitivity requirements of single detectors [9].

A common realization method of cooperative spectrum sensing integrates the sensing function into secondary user (SU) terminals. However, a sufficient number of adequately located SUs is required to obtain accurate detection results. Furthermore, the SU cannot transmit data during the sensing period [10] [11], reducing the user information transport efficiency, while still consuming battery power. An alternative approach is External Sensing, which relies on a dedicated wireless sensor network (WSN) for spectrum sensing. A dedicated WSN can be deployed and controlled by the cognitive service provider, and can aid an SU to access the licensed spectrum. Such a scheme can cope with the hidden primary user problem, and guarantees robustness against model uncertainty induced by fading and path loss [12]. Moreover, in such a scheme sensors neither need to be mobile nor battery-powered, and the cost, complexity and power consumption of SU terminals can be reduced.

In this paper, we consider a Detect-and-Forward (DetF) distributed WSN for cooperative spectrum sensing. The operation of this scheme does not rely on centralized control requiring a separate decision fusion center. Hence better flexibility and scalability can be provided by such an approach [13]. However, a proper media access control (MAC) protocol is required to coordinate transmissions between sensors in order to efficiently use resources and resolve contention. In an WSN, the most significant design constraint is usually the limited energy budget of a sensor node together with the requirement of network longevity [14] [15]. However, in our DetF WSN sensors are assumed to be fixed in the deployment region and do not depend on limited battery power. Thus, spectrum efficiency and interference, instead of energy efficiency, become the primary issues. Another special part of our MAC design for this DetF WSN is its performance as a cooperative spectrum sensing system.

More specifically, the contributions of this paper are three fold: 1) We propose a spatial reuse MAC protocol based on TDMA/OFDMA for exchanging sensing results that avoids primary conflicts and saves bandwidth resources. As design

approaches, we consider a greedy algorithm as well as an adaptive simulated annealing (ASA) technique. 2) We analyze the degradation introduced by reporting channel errors on cooperative spectrum sensing performance with the  $k$ -out-of- $n$  decision fusion rule. 3) For illustrative purposes we present numerical results, in terms of Receiver Operating Characteristics (ROC) curves, that demonstrate the spectrum sensing performance of such DetF WSN with the spatial reuse MAC protocol in a grid network. However, the proposed spatial reuse MAC protocol is not limited to a grid network only, and can be employed in any network structure with an associated cooperating partner selection scheme. The rest of this paper is organized as follows. Section 2 presents the system model and analyzes the performance of cooperative spectrum sensing using  $k$ -out-of- $n$  decision fusion with imperfect reporting channels. The designs of the MAC protocol by using the greedy algorithm and the ASA technique are presented in Section 3. Numerical performance results for a grid network are presented in Section 4, and the conclusions are drawn in Section 5.

## 2. System Model and Preliminaries of Cooperative Spectrum Sensing

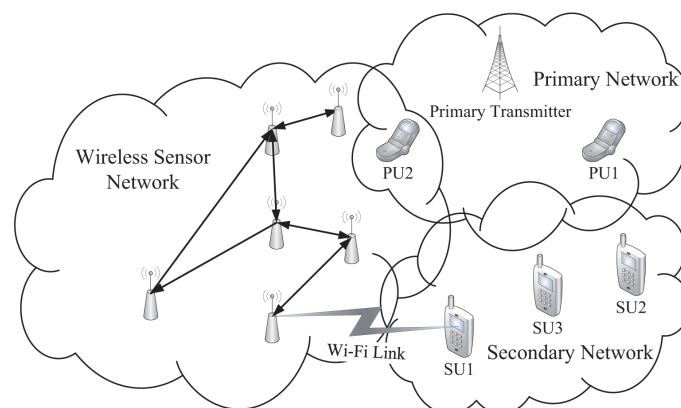
### (A) Network architecture

Consider a distributed DetF WSN illustrated in **Figure 1**, where the nodes have spectrum sensing as well as communication capabilities. The operation of this WSN can be divided into two stages: 1) Intra-WSN Sensing and 2) WSN-SU Handshaking. The first stage is summarized as follows:

a) *Measurement Phase*: A sensor measures the received signal over the channel of interest.

b) *Local Decision Phase*: A sensor makes a hard decision based on its measurement via a local spectrum sensing scheme [16].

c) *Communication Phase*: Sensor exchange their local decisions through dedicated reporting channels. The cooperation between sensors is determined by a partner selection scheme specified by an adjacency matrix  $\mathbf{R} = [R_{i,j}]$ . When  $R_{i,j} = 1 (i \neq j)$  then this indicates that there is cooperation from the partner  $s_i$  to recipient  $s_j$ . Otherwise,  $R_{i,j} = 0$ .



**Figure 1.** A WSN for cooperative spectrum sensing.

d) *Final Decision Phase*: A final decision is made by each sensor through combining the decisions gathered (including its own), according to a decision fusion rule.

e) The four phases above are repeated for another channel of interest.

In the second stage the sensing results are made available to SUs, which is a topic beyond the scope of this paper that focusses on spectrum sensing.

### (B) Channel model

All wireless channels in the system, including the reporting as well as the detecting channels, are modeled as time-invariant small-scale fading including path loss and additive interference.

#### Path loss model

We use the simplified model of ([17], p. 47) to characterize the variation in the received signal power with distance:

$$P_L = P_T / P_R = d^\mu / A \quad (1)$$

where the path loss  $P_L$  is defined as the ratio of the transmit power  $P_T$  over the received power  $P_R$ ,  $A$  is a constant which depends on the antenna characteristics and the average channel attenuation,  $d$  is the distance between the transmitter and receiver, and  $\mu$  is the path loss exponent.

#### Small scale fading model

We assume small scale Rician fading for each communication link ([18], p. 23), with probability density function (PDF) of the instantaneous signal-to-noise ratio (SNR)  $\gamma$  given by

$$f_{\text{Ric}}(\gamma) = \frac{K+1}{\bar{\gamma}} \exp\left[-K - \frac{(K+1)\gamma}{\bar{\gamma}}\right] I_0\left(2\sqrt{\frac{K(K+1)\gamma}{\bar{\gamma}}}\right) \quad (2)$$

where  $\gamma \geq 0$ ,  $K$  is the Rician fading parameter,  $\bar{\gamma}$  is the average received SNR, and  $I_0(\cdot)$  is the 0th order modified Bessel function of the first kind. When combining the effects of path loss and fading, the received SNR ([17], p.77) is  $\bar{\gamma} = \gamma_T A d^{-\mu}$ , where  $\gamma_T$  represents the transmit SNR. Our analysis can be applied also to other fading models, when (2) is replaced by the corresponding PDF. In addition, we ignore the impact of shadowing in this paper since the sensing nodes can be placed such that reporting channels are not impaired by this phenomenon. As far as the sensing channels are concerned, we assume that if a sensing node is shadowed, then it is disregarded.

#### Interference model

We assume an additive interference model, in which a link treats all the other on-going transmissions on the same channel as noise [19]. Let  $G_{i,j}$  be the channel gain,  $P_T^{(i)}$  the transmit power of  $s_i$ , and  $P_N$  the thermal noise power ( $P_N = N_0 W_c$  where the transmission bandwidth is  $W_c$ ). The received Signal to Interference plus Noise Ratio (SINR) at  $s_j$  is

$$\gamma_{i,j} = \frac{G_{i,j} P_T^{(i)}}{P_N + \sum_{k \in \mathcal{K} \setminus \{i\}} G_{k,j} P_T^{(k)}} \quad (3)$$

where  $\mathcal{K} \setminus \{i\}$  denotes the set of sensors, other than  $s_i$ , transmitting simultaneously on the same channel, and  $G_{i,j} = Ad_{i,j}^{-\mu}$  from (1).

**(C) Cooperative spectrum sensing through imperfect reporting channels**

We assume that each sensor employs a local energy detector, and a channel to be sensed has center frequency  $f_c$ . The received signal passes through an input bandpass filter of bandwidth  $W_D$  centered at  $f_c$ , then a squaring device and integrator measure the received energy over an observation interval  $T$ . Finally, the test statistic,  $\Lambda$ , formed by the integrator is compared to a threshold  $\lambda$ , to produce the local decision [20]. The probabilities of detection and false alarm over AWGN channels are given, respectively, by

$$P_{d,i} = \text{Prob}\{\Lambda_i > \lambda_i \mid \mathcal{H}_1\} = Q_u(\sqrt{2\gamma_i}, \sqrt{\lambda_i}) \tag{4}$$

$$P_{f,i} = \text{Prob}\{\Lambda_i > \lambda_i \mid \mathcal{H}_0\} = \frac{\Gamma(u, \lambda_i/2)}{\Gamma(u)} \tag{5}$$

where  $\lambda_i$  denotes the energy detection threshold,  $\gamma_i = \frac{P_{R,i}}{N_0 W_D}$  is the instantaneous received SNR of primary signal at  $s_i$  ( $P_{R,i}$  is the received primary signal power at  $s_i$ ,  $N_0$  is the noise power spectral density),  $u = TW_D$  is the time-bandwidth product which is assumed to be an integer for simplicity,  $\Gamma(\cdot)$ ,  $\Gamma(\cdot, \cdot)$  are the complete and upper incomplete gamma functions, and  $Q_u(\cdot, \cdot)$  is the generalized Marcum Q-function. With Ricean fading we need to average (4) and (5) over  $\gamma_i$  using (2), resulting in the average probabilities of detection and false alarm [21]:

$$P_{d,i} = \int_{\text{all}\gamma_i} Q_u(\sqrt{2\gamma_i}, \sqrt{\lambda_i}) f_{\text{Ric}}(\gamma_i) d\gamma_i, \tag{6}$$

$$P_{f,i} = \int_{\text{all}\gamma_i} \frac{\Gamma(u, \lambda_i/2)}{\Gamma(u)} f_{\text{Ric}}(\gamma_i) d\gamma_i = \frac{\Gamma(u, \lambda_i/2)}{\Gamma(u)}. \tag{7}$$

where  $P_{f,i}$  is the same as that over an AWGN channel as it is independent of received SNR  $\gamma_i$ . When  $u = 1$ , (6) has the form

$$P_{d,i} = Q_1\left(\sqrt{\frac{2K\bar{\gamma}_i}{K+1+\bar{\gamma}_i}}, \sqrt{\frac{\lambda_i(K+1)}{K+1+\bar{\gamma}_i}}\right)$$

where  $\bar{\gamma}_i$  is the average received SNR of the primary signal [21]. In the following analysis, we use  $u = 1$  for simplicity.

We assume that each sensing node gathers hard decisions from other partner nodes including its own decision, and uses the  $k$ -out-of- $n$  fusion rule to produce a final decision ([22], pp. 59-61). We define the partner set of a sensor  $s_i$  as  $\mathcal{J}^{(i)} = \{s_j \mid R_{j,i} = 1\}$ , and use  $n_i$  to denote the number of partners, i.e.  $n_i = |\mathcal{J}^{(i)}|$ , where  $|\cdot|$  represents the cardinality of a set. Sensor  $s_i$  declares  $\mathcal{H}_1$  (signal is present in the channel) when at least  $k$  out of the  $n_i$  decisions are  $\mathcal{H}_1$ . Otherwise, the output is  $\mathcal{H}_0$  (channel is free).

The channel from  $s_j$  to  $s_i$  is modeled as a binary symmetric channel (BSC) with cross-over probability  $\varepsilon_{j,i} \in [0, 0.5)$ , representing the uncoded BEP [8].

Without loss of generality, in our work we choose binary phase shift keying (BPSK). The average BEP for BPSK in Rician fading over reporting channel  $s_j \rightarrow s_i$  is ([18], p. 126)

$$\varepsilon_{j,i} = \frac{1}{\pi} \int_0^{\pi/2} \frac{(1+K)\sin^2\phi}{(1+K)\sin^2\phi + \bar{\gamma}_{j,i}} \exp\left[-\frac{K\bar{\gamma}_{j,i}}{(1+K)\sin^2\phi + \bar{\gamma}_{j,i}}\right] d\phi \quad (8)$$

where  $K$  is the Rician parameter, and  $\bar{\gamma}_{j,i}$  is the average received SNR at  $s_i$  determined by the path loss as well as both AWGN and additive interference.

Hence from (3), 
$$\bar{\gamma}_{j,i} = \frac{Ad_{i,j}^{-\mu} P_T^{(i)}}{P_N + \sum_{k \in \mathcal{K} \setminus \{i\}} Ad_{k,j}^{-\mu} P_T^{(k)}}.$$

If an error occurs when  $s_j \in \mathcal{J}^{(i)}$  transmits its decision to  $s_i$ , a  $\mathcal{H}_1$  (or  $\mathcal{H}_0$ ) decision of  $s_j$  will be turned into  $\mathcal{H}_0$  (or  $\mathcal{H}_1$ ) when received by  $s_i$ . Thus, as shown in [8],  $\forall s_j \in \mathcal{J}^{(i)}$ , the equivalent probabilities of detection and false alarm received by  $s_i$  are

$$P_{d,j}^{(i)} \triangleq \text{prob}\{s_i \text{ receives } \mathcal{H}_1 \text{ from } s_j \mid \mathcal{H}_1\} = P_{d,j}(1 - \varepsilon_{j,i}) + (1 - P_{d,j})\varepsilon_{j,i}, \quad (9)$$

$$P_{f,j}^{(i)} \triangleq \text{prob}\{s_i \text{ receives } \mathcal{H}_1 \text{ from } s_j \mid \mathcal{H}_0\} = P_{f,j}(1 - \varepsilon_{j,i}) + (1 - P_{f,j})\varepsilon_{j,i}. \quad (10)$$

We assume that each sensor experiences i.i.d. Rician fading, and performs spectrum sensing independently. In this paper, we employ combinatorial notations as in [23]. We use  $\binom{\mathcal{J}^{(i)}}{m}$  to represent the set of all  $m$ -combinations of

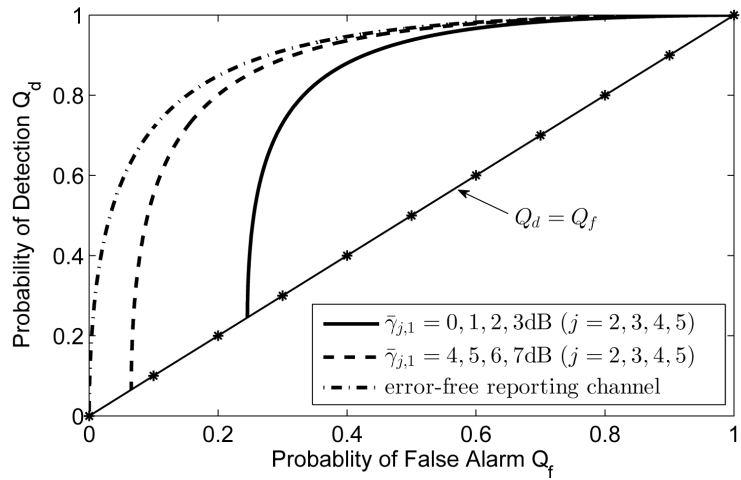
$\mathcal{J}^{(i)}$  having  $\binom{n_i}{m} = \frac{n_i!}{m!(n_i - m)!}$  members,  $\mathcal{J}_{m,l}^{(i)} \in \binom{\mathcal{J}^{(i)}}{m}, 1 \leq l \leq \binom{n_i}{m}$  represents one  $m$ -combination of  $\mathcal{J}^{(i)}$  ( $\mathcal{J}_{m,l}^{(i)}$  are arranged in lexicographic order), and  $\mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}$  denotes the relative complement of  $\mathcal{J}_{m,l}^{(i)}$  in  $\mathcal{J}^{(i)}$ . Then from ([16], Equation (11)) we obtain the probabilities of detection and false alarm at sensor  $s_i$  after  $k_i$ -out-of- $n_i$  decision fusion,

$$Q_{d,i}(k_i) = \sum_{m=k_i}^{n_i} \sum_{l=1}^{\binom{n_i}{m}} \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} P_{d,j}^{(i)} \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} (1 - P_{d,j}^{(i)}) \right) \quad (11)$$

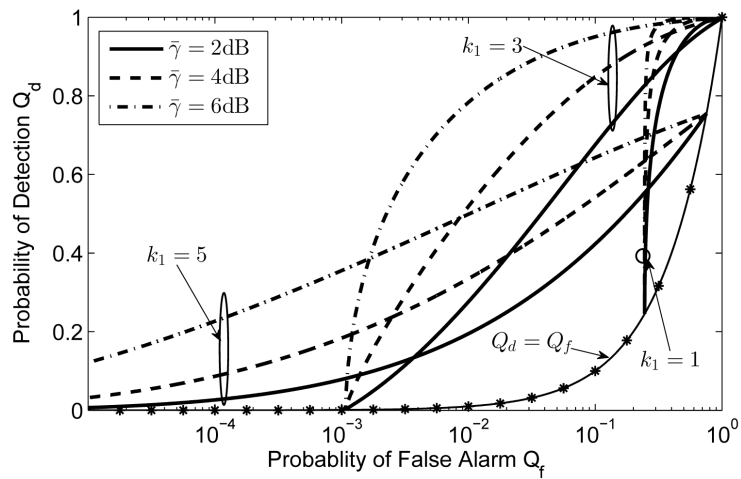
$$Q_{f,i}(k_i) = \sum_{m=k_i}^{n_i} \sum_{l=1}^{\binom{n_i}{m}} \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} P_{f,j}^{(i)} \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} (1 - P_{f,j}^{(i)}) \right) \quad (12)$$

where  $P_{d,j}^{(i)}$  and  $P_{f,j}^{(i)}$  are given in (9) and (10), and  $1 \leq k_i \leq n_i$  is  $s_i$ 's decision fusion threshold.

In **Figure 2** we present the impacts of reporting channel errors on cooperative spectrum sensing performance by illustrating ROC curves of  $Q_{d,i}(k_i)$  versus  $Q_{f,i}(k_i)$ . There are 5 sensors employing identical energy detection thresholds  $\lambda$ , and  $s_1$  makes a global decision. Thus, we omit the second subscript  $i$  in  $Q_{d,i}(k_i)$  and  $Q_{f,i}(k_i)$ . The Rician fading parameter is  $K = 7$ , and the average received SNR of the primary signal  $\bar{\gamma}$  is identical at each sensors. In **Figure 2(a)**, the vertical separation between curves indicates the degradation introduced



(a)



(b)

**Figure 2.**  $Q_d$  vs  $Q_f$  at  $S_1$  when  $n_1 = 5$ , and  $K = 7$ ,  $\bar{\gamma}$  is the identical received SNR of primary signal, and  $\bar{\gamma}_{j,1}$  ( $j \neq 1$ ) represents the average received SNR on each reporting channel. (a) OR rule with different sets of reporting channels, where  $\bar{\gamma} = 2$  dB, and  $k_1 = 1$ ; (b) Different  $\bar{\gamma}$  and decision fusion thresholds  $k_1$ , where  $\bar{\gamma}_{j,1} = 0, 1, 2, 3$  dB ( $j = 2, 3, 4, 5$ ).

by reporting channel errors for OR rule ( $k_1 = 1$ ), which is small in high  $Q_f$  range, and increases when  $Q_f$  decreases. Moreover, for ROC curves with erroneous reporting channels when  $Q_f$  decreases to a certain level  $Q_d$  will drastically decrease, and any  $Q_f$  lower than a threshold cannot be achieved. In **Figure 2(b)**, we fix the average received SNR on each reporting channel, and compare the ROC curves with different  $\bar{\gamma}$  and  $k_1$ . For a certain  $k_1$ , higher  $\bar{\gamma}$  results in better detection performance. However, the minimum values of  $Q_d$  and  $Q_f$  are no longer 0 as for error-free reporting channels. From (6), (7), (9) and (10), we have:

-When  $\lambda_j \rightarrow \infty$ ,  $P_{d,j} = P_{f,j} \rightarrow 0$  and thus

$$\min P_{d,j}^{(i)} = \min P_{f,j}^{(i)} = 0 - 2\varepsilon_{j,i} \times 0 + \varepsilon_{j,i} = \varepsilon_{j,i}$$

-When  $\lambda_j = 0$ ,  $P_{d,j} = P_{f,j} = 1$  and thus

$$\max P_{d,j}^{(i)} = \max P_{f,j}^{(i)} = 1 - 2\varepsilon_{j,i} \times 1 + \varepsilon_{j,i} = 1 - \varepsilon_{j,i}$$

**Property 1.** For a partner set  $\mathcal{J}^{(i)}$ , and decision fusion threshold  $k_i$ , if each  $P_{d,j}^{(i)}$  (or  $P_{f,j}^{(i)}$ ) assumes minimum value  $\varepsilon_{j,i}$ , then  $Q_{d,i}(k_i)$  (or  $Q_{f,i}(k_i)$ ) reaches minimum value. If each  $P_{d,j}^{(i)}$  (or  $P_{f,j}^{(i)}$ ) assumes maximum value  $1 - \varepsilon_{j,i}$ , then  $Q_{d,i}(k_i)$  (or  $Q_{f,i}(k_i)$ ) reaches maximum value.

*Proof:* Suppose that there are  $n_i$  Bernoulli trials, each of which has a success probability  $P_{d,j}^{(i)}$  (or  $P_{f,j}^{(i)}$ ). Then,  $Q_{d,i}(k_i)$  (or  $Q_{f,i}(k_i)$ ) is the probability that at least  $k_i$  of these  $n_i$  trials are successful, which is the complement of the Poisson-Binomial cumulative distributed function (CDF) ([24], Equation (8)). From Lemma 1 in **Appendix A**, we conclude that  $Q_{d,i}(k_i)$  (or  $Q_{f,i}(k_i)$ ) is monotonic non-decreasing in each  $P_{d,j}^{(i)}$  (or  $P_{f,j}^{(i)}$ ), proving Property 1.

Therefore, using (11) and (12) with  $P_{d,j}^{(i)}$  and  $P_{f,j}^{(i)}$  substituted by their minimum values  $\varepsilon_{j,i}$  for the lower bound and by their maximum values  $1 - \varepsilon_{j,i}$  for the upper bound, the minimum and maximum probabilities of detection and false alarm for a  $k_i$ -out-of- $n_i$  decision fusion rule at  $s_i$  are

$$Q_{d,i}^{[L]}(k_i) = Q_{f,i}^{[L]}(k_i) = \sum_{m=k_i}^{n_i} \sum_{l=1}^m \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} \varepsilon_{j,i} \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} (1 - \varepsilon_{j,i}) \right), \tag{13}$$

$$Q_{d,i}^{[U]}(k_i) = Q_{f,i}^{[U]}(k_i) = \sum_{m=k_i}^{n_i} \sum_{l=1}^m \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} (1 - \varepsilon_{j,i}) \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} \varepsilon_{j,i} \right). \tag{14}$$

The OR fusion rule is a special case when  $k_i = 1$ . Since  $\binom{n_i}{0} = 1$ ,  $\mathcal{J}_{0,l}^{(i)} = \emptyset$ , and  $\mathcal{J}^{(i)} = \{s_j \mid R_{j,i} = 1\}$ , from (13) and (14) with the convention  $\prod_{s_j \in \emptyset} (\cdot) = 1$  we get

$$Q_{d,i}^{[L]}(1) = Q_{f,i}^{[L]}(1) = 1 - \sum_{l=1}^{\binom{n_i}{0}} \left( \prod_{s_j \in \mathcal{J}_{0,l}^{(i)}} \varepsilon_{j,i} \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{0,l}^{(i)}} (1 - \varepsilon_{j,i}) \right) = 1 - \prod_{s_j \in \{s_j \mid R_{j,i} = 1\}} (1 - \varepsilon_{j,i}) \tag{15}$$

$$Q_{d,i}^{[U]}(1) = Q_{f,i}^{[U]}(1) = 1 - \sum_{l=1}^{\binom{n_i}{0}} \left( \prod_{s_j \in \mathcal{J}_{0,l}^{(i)}} (1 - \varepsilon_{j,i}) \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{0,l}^{(i)}} \varepsilon_{j,i} \right) = 1 - \prod_{s_j \in \{s_j \mid R_{j,i} = 1\}} \varepsilon_{j,i}. \tag{16}$$

Define the Achievable Range of  $Q_{d,i}(k_i)$  and  $Q_{f,i}(k_i)$  as

$$\Delta_{d,i}(k_i) \triangleq Q_{d,i}^{[U]}(k_i) - Q_{d,i}^{[L]}(k_i) \text{ and } \Delta_{f,i}(k_i) \triangleq Q_{f,i}^{[U]}(k_i) - Q_{f,i}^{[L]}(k_i). \tag{17}$$

From (13) and (14), we can see that  $Q_{f,i}^{[L]}(k_i) = Q_{d,i}^{[U]}(k_i)$ ,  $Q_{f,i}^{[U]}(k_i) = Q_{d,i}^{[L]}(k_i)$ , and thus  $\Delta_{f,i}(k_i) = \Delta_{d,i}(k_i)$ . Therefore, the following properties about  $Q_{d,i}^{[L]}(k_i)$ ,  $Q_{d,i}^{[U]}(k_i)$ , and  $\Delta_{d,i}(k_i)$  hold also for  $Q_{f,i}^{[L]}(k_i)$ ,  $Q_{f,i}^{[U]}(k_i)$  and  $\Delta_{f,i}(k_i)$ .

**Property 2.** For fixed  $\mathcal{J}^{(i)}$  and  $k_i$ ,  $Q_{d,i}^{[L]}(k_i)$ ,  $Q_{d,i}^{[U]}(k_i)$ , and  $\Delta_{d,i}(k_i)$ , only depend on  $\varepsilon_{j,i}$ .

Follows directly from (13), (14) and (17).

**Property 3.** For fixed  $\mathcal{J}^{(i)}$  and  $\varepsilon_{j,i}$ ,  $Q_{d,i}^{[L]}(k_i)$  and  $Q_{d,i}^{[U]}(k_i)$  decrease with increasing  $k_i$ .



For any  $k_{i,1} < k_{i,2}$ , using (13) and (14), we get

$$Q_{d,i}^{[L]}(k_{i,1}) - Q_{d,i}^{[L]}(k_{i,2}) = \sum_{m=k_{i,1}}^{k_{i,2}-1} \sum_{l=1}^{\binom{n_i}{m}} \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} \mathcal{E}_{j,i} \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} (1 - \mathcal{E}_{j,i}) \right) > 0,$$

$$Q_{d,i}^{[U]}(k_{i,1}) - Q_{d,i}^{[U]}(k_{i,2}) = \sum_{m=k_{i,1}}^{k_{i,2}-1} \sum_{l=1}^{\binom{n_i}{m}} \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} (1 - \mathcal{E}_{j,i}) \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} \mathcal{E}_{j,i} \right) > 0.$$

**Property 4.**  $\Delta_{d,i}(k_i) = \Delta_{d,i}(n_i - k_i + 1), 1 \leq k_i \leq n_i$ .

See **Appendix B** for proof.

**(D) Other system setup assumptions**

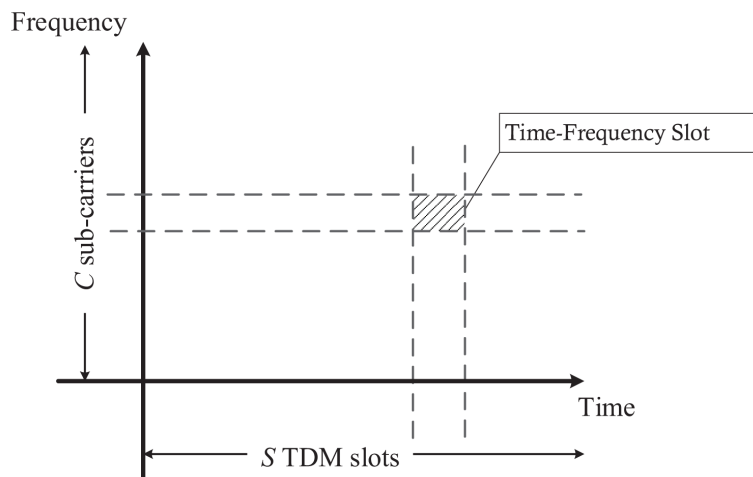
For network design purposes, we also make the following assumptions:

- Sensors are static after deployed.
- Each sensor employs an omni-directional antenna.
- A partner selection scheme has been devised to generate the adjacency matrix  $R$  for cooperation.
- Each sensor is in direct transmission range of its partners.
- The transmit power of each sensor is constant.
- Sensors communicate by broadcasting, and can dynamically switch to different sub-carriers.
- Sensors can transmit and receive at the same time on different sub-carriers.
- Sensors are synchronized ensuring time synchronous operation.

### 3. Spatial Reuse MAC Protocol Based on Hybrid TDMA/OFDMA

**(A) MAC Design Concept**

Our MAC protocol is based on TDMA/OFDMA, employing a frame of  $S$  time slots and  $C$  orthogonal sub-carriers as illustrated in **Figure 3**. An intuitive approach is to schedule each sensor into a particular time-frequency (T-F) slot [25]. Such a scheme is unrealistic since the number of required slots will increase linearly with the number of sensors. This problem can be solved by allowing a transmitter to share a slot with other far enough ones. The MAC protocol in



**Figure 3.** A MAC structure based on TDMA/OFDMA.

[26] reduces the number of required slots by dividing sensors into groups which can communicate simultaneously, employing spatial reuse for scheduling. The objective is to divide the  $N$  sensors into  $M$  separate sets, and assign one T-F slot to the sensors in the same set. The scheduling result is defined as an  $N$ -dimensional vector  $\mathbf{p} = (p_1, p_2, \dots, p_N)$ , where  $p_i \in \{1, 2, \dots, M\}$  represents the ID of the T-F slot assigned to  $s_i$ .

We assume that the T-F slots have identical frame lengths as well as sub-carrier bandwidths. Thus, we only need to care about the T-F slot sharing relationship between sensors, rather than a sensor's specific slot ID. For instance, if  $N = 6$  and  $M = 3$  the three scheduling schemes  $\mathbf{p} = (1, 1, 2, 3, 2, 1)$ ,  $\mathbf{p}' = (2, 2, 1, 3, 1, 2)$  and  $\mathbf{p}'' = (3, 3, 1, 2, 1, 3)$  are the same. To avoid such repetitions, we introduce the following ordering restriction for the scheduling vector  $\mathbf{p} = (p_1, p_2, \dots, p_N)$ :

$$\forall m_1 < m_2, m_1, m_2 = 1, 2, \dots, M : \min\{i \in \mathbb{Z}^+ \mid p_i = m_1\} < \min\{i \in \mathbb{Z}^+ \mid p_i = m_2\} \quad (18)$$

where  $\{i \in \mathbb{Z}^+ \mid p_i = m_1\}$  is the set containing the indices of  $p_i$  whose value is  $m_1$ , and  $\min\{i \in \mathbb{Z}^+ \mid p_i = m_1\}$  represents the smallest index. Thus, (18) means that if  $m_1 < m_2$ , the smallest index in  $\{i \in \mathbb{Z}^+ \mid p_i = m_2\}$  is less than the smallest index in  $\{i \in \mathbb{Z}^+ \mid p_i = m_1\}$ . For example, when  $\mathbf{p} = (1, 1, 2, 3, 2, 1)$ , we have  $\min\{i \in \mathbb{Z}^+ \mid p_i = 1\} = 1 < \min\{i \in \mathbb{Z}^+ \mid p_i = 2\} = 3 < \min\{i \in \mathbb{Z}^+ \mid p_i = 3\} = 4$ . So  $\mathbf{p} = (1, 1, 2, 3, 2, 1)$  satisfies condition (18).

When  $s_i$  and  $s_j$  are scheduled into the same slot, a primary conflict occurs when there is a communication link between them, or they have a common cooperating recipient, a situation to be avoided. We build an  $N \times N$  conflict relation matrix  $\mathbf{F}$  with components

$$F_{i,j} = \begin{cases} 1 & \text{if } (i \neq j) \wedge (R_{i,j} = 1 \vee R_{j,i} = 1 \vee \exists k : R_{i,k} = R_{j,k} = 1) \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

where  $\wedge$  and  $\vee$  denote the logical conjunction and disjunction,  $i \neq j$  means a sensor cannot have conflict with itself,  $R_{i,j} = 1 \vee R_{j,i} = 1$  means there is a communication link between two different sensors, and  $\exists k : R_{i,k} = R_{j,k} = 1$  means  $s_i$  and  $s_j$  have a common cooperating recipient  $s_k$ . Thus, when two different sensors  $s_i$  and  $s_j$  share the same T-F slot, (19) means that a primary conflict occurs between two different sensors  $s_i$  and  $s_j$  (i.e.  $F_{i,j} = 1$ ), if there exists a communication link from  $s_i$  to  $s_j$  (i.e.  $R_{i,j} = 1$ ) or a communication link from  $s_j$  to  $s_i$  (i.e.  $R_{j,i} = 1$ ) or a common cooperating recipient  $s_k$  of  $s_i$  and  $s_j$  (i.e.  $\exists k : R_{i,k} = R_{j,k} = 1$ ).

In other cases, there exists a secondary conflict, which can be permitted if the impact caused by mutual interference is acceptable. Given that  $s_j$  is a partner of  $s_i$ , i.e.  $R_{j,i} = 1$ , when  $s_j$  broadcasts its package, other sensors who share the same T-F slot with  $s_j$  become the interference sources. We define a distance matrix  $\mathbf{D}$ , whose element  $d_{i,j}$  represents the Euclidean distance between  $s_i$  and  $s_j$ . From (1) and (3), the average received SINR associated with the transmission from  $s_j$  to  $s_i$  is

$$\bar{\gamma}_{j,i}(\mathbf{p}) = \frac{Ad_{j,i}^{-\mu} P_{\text{tra}}^{(j)}}{P_{\text{noi}} + \sum_{k \in \{k \in \mathbb{Z}^+ | p_k = p_j, k \neq j\}} Ad_{k,i}^{-\mu} P_{\text{tra}}^{(k)}} \quad (20)$$

where  $\mathbf{p} = (p_1, p_2, \dots, p_N)$  represents the scheduling scheme, and using (8) the BEP is

$$\varepsilon_{j,i}(\mathbf{p}) = \frac{1}{\pi} \int_0^{\pi/2} \frac{(1+K)\sin^2\phi}{(1+K)\sin^2\phi + \bar{\gamma}_{j,i}(\mathbf{p})} \exp\left[-\frac{K\bar{\gamma}_{j,i}(\mathbf{p})}{(1+K)\sin^2\phi + \bar{\gamma}_{j,i}(\mathbf{p})}\right] d\phi. \quad (21)$$

**(B) Problem formulation**

The distance matrix  $\mathbf{D}$  is determined by the sensors' location, assumed to be fixed. Sensors should be spatially separated, to properly cover the target area. Analysis of the degradation on spectrum detection performance caused by correlated shadowing, showing that the distance between sensors should be larger than the decorrelation distance, can be found in [9] [13]. However, sensors cannot be deployed too far from each other to avoid a high BEP on each reporting channel and compromise the cooperative spectrum sensing performance as discussed in last section. Moreover, when the WSN is deployed by a service provider, the sensors may be installed in existing base stations, where the locations cannot be changed.

The adjacency matrix  $\mathbf{R}$  characterizes the cooperating relationship between sensors. If the distance between sensors is sufficiently large to avoid severe correlated shadowing, a sensor  $s_i$  always chooses the cooperating partners with the lowest reporting channel BEP first. As we cannot decide the scheduling of T-F slots before  $\mathbf{R}$  is determined, we do not consider the mutual interference between sensors, and only consider the thermal noise when setting up  $\mathbf{R}$ . If all the sensors use the same transmit power and experience identical multipath fading, a sensor always chooses the nearest sensors around as cooperating partners. Moreover, the number of partners of each sensor should be similar, such that the spectrum sensing performance does not vary significantly between different sensors. The matrix  $\mathbf{R}$  can be changed in the network design process, and in Section 4, we will discuss how to set up  $\mathbf{R}$  for a grid network.

Given a certain number of slots  $M$ , our spatial reuse MAC protocol aims at providing a partition of the  $N$  sensors to avoid any primary conflict, and minimize the degradation caused by the BEPs on communication links between sensors. We focus on maximizing the Achievable Range of the probability of detection or false alarm,  $\Delta_i(k_i, \mathbf{p}) = \Delta_{d,i}(k_i, \mathbf{p}) = \Delta_{f,i}(k_i, \mathbf{p})$ , as defined in (17), which characterizes the reporting error effect, not depending on the local sensing quality. The second argument  $\mathbf{p}$  is added because  $\varepsilon_{j,i}$  of (17) depends on  $\mathbf{p}$ . From (13), (14) and (17), we have

$$\Delta_i(k_i, \mathbf{p}) = \sum_{m=k_i}^{n_i} \sum_{l=1}^m \left( \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} \varepsilon_{j,i}(\mathbf{p}) \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} (1 - \varepsilon_{j,i}(\mathbf{p})) - \prod_{s_j \in \mathcal{J}_{m,l}^{(i)}} (1 - \varepsilon_{j,i}(\mathbf{p})) \prod_{s_j \in \mathcal{J}^{(i)} \setminus \mathcal{J}_{m,l}^{(i)}} \varepsilon_{j,i}(\mathbf{p}) \right). \quad (22)$$

The OR fusion rule ( $k_i = 1$ ) as well as AND fusion rule ( $k_i = n_i$ ) are special cases with same Achievable Range (from Property 4), which can be obtained from (15) and (16)

$$\Delta_i(1, \mathbf{p}) = \Delta_i(n_i, \mathbf{p}) = \prod_{s_j \in \{s_j | R_{j,i} = 1\}} (1 - \varepsilon_{j,i}(\mathbf{p})) - \prod_{s_j \in \{s_j | R_{j,i} = 1\}} \varepsilon_{j,i}(\mathbf{p}). \quad (23)$$

The network fusion factor  $\Omega$  is defined by  $k_i = \lceil \Omega n_i \rceil$ , where  $k_i$  is the fusion threshold at  $s_i$ ,  $n_i = |\{s_j | R_{j,i} = 1\}|$  is the number of partners of  $s_i$ , and  $\lceil \cdot \rceil$  represents the ceiling operator. Averaging  $\Delta_i(k_i, \mathbf{p})$  over sensor locations we obtain the Network Achievable Range of probability of detection or false alarm:

$$\bar{\Delta}(\Omega, \mathbf{p}) = \frac{1}{N} \sum_{i=1}^N \Delta_i(k_i, \mathbf{p}). \quad (24)$$

Therefore, with the adjacency matrix  $\mathbf{R}$  and distance matrix  $\mathbf{D}$  of a DetF WSN of  $N$  sensors, number of T-F slots fixed at  $M$ , and network fusion factor  $\Omega$ , the MAC protocol design can be formulated as the solution of the combinatorial optimization problem:

$$\max_{\mathbf{p}} \bar{\Delta}(\Omega, \mathbf{p}) \text{ or equivalently } \min_{\mathbf{p}} (1 - \bar{\Delta}(\Omega, \mathbf{p})) \quad (25)$$

s.t.

$$\mathbf{p} = (p_1, p_2, \dots, p_N), \text{ with } p_i \in \{1, 2, \dots, M\} \quad (26)$$

$$\forall (i, j) \in \{(i, j) | p_i = p_j\} : F_{i,j} = F_{j,i} = 0 \quad (27)$$

$$\forall m_1 < m_2, m_1, m_2 = 1, 2, \dots, M : \min \{i \in \mathbb{Z}^+ | p_i = m_1\} < \min \{i \in \mathbb{Z}^+ | p_i = m_2\}. \quad (28)$$

where (26) is the scheduling result, (27) is the constraint for primary conflicts, and (28) is the constraint to avoid repetition. If  $\mathbf{p}$  satisfies constraint (27), it is defined as a feasible solution. If it satisfies both (27) and (28), it is called a valid solution. Moreover, we define the cost function as

$$\zeta(\mathbf{p}) = 1 - \bar{\Delta}(\Omega, \mathbf{p}) \quad (29)$$

which is called the Network Achievable Range Loss of probability of detection or false alarm.

Essentially, such a MAC design is a combinatorial optimization problem, that is usually computationally intractable for practical applications [27]. Many combinatorial optimization problems of scheduling or channel assignment in wireless networks using similar models are too complicated for exact solution. In [28] [29] [30] [31] [32], efforts have been invested in developing techniques based on heuristic combinatorial optimization for such problems [33] [34]. Common heuristic methods for solving such problems include: *Tabu Search*, *Genetic Algorithms*, *Simulated Annealing*, *Ant System*, *Neural Networks*, [33]. Next we consider two approaches: Greedy Algorithm and Simulated Annealing.

**(C) Spatial reuse by graph colouring and greedy algorithm**

We first form  $\mathbf{F}$  based on the given  $\mathbf{R}$ . Then all the sensors are considered as vertices, and an edge is drawn between  $s_i$  and  $s_j$  if  $F_{i,j} = 1$  or  $F_{j,i} = 1$ .

Thus, a simple undirected conflict graph is constructed, and finding a feasible solution  $\mathbf{p}$  can be viewed as assigning  $M$  “colours” to the vertices of this conflict graph, such that adjacent vertices have different colours. We choose the DSatur algorithm (a heuristic vertex colouring procedure) [35] to generate an initial solution. To explain this algorithm, we introduce two definitions:

- i) The degree is the number of edges incident to a vertex.
- ii) The saturation degree is the number of different colours in the neighbors of a vertex.

To start the DSatur algorithm we number the colours sequentially, and assign colour 1 to the vertex with the highest degree. In case of a tie, choose the one with the smallest sensor index. Select the next vertex with the highest saturation degree, and in case of a tie choose the vertex with the highest degree. Next, search the colours in ascending order, and find the first available one which hasn't been assigned to any neighbour of the current vertex. Assign this colour to the current vertex, and keep checking the remaining uncoloured vertices in such an order until all the vertices are coloured. Let  $M_0$  denote the number of colours used in DSatur algorithm. When  $M < M_0$  the problem of determining whether such an  $M$ -colouring exists is NP-complete [36], and thus we only consider the situation when  $M \geq M_0$ . Finally, we obtain a feasible initial solution  $\mathbf{p}$  by setting  $p_i$  to  $s_i$ 's colour index. This initialization method is applicable to any network form; nevertheless, if  $\mathbf{R}$  or  $\mathbf{F}$  have certain regular structures, we can find other more efficient deterministic algorithms. An example will be given in Section IV-B.

The initial generated solution may conflict with condition (28). So we perform the realignment procedure, which will be referred to as **REALIGN**( $\mathbf{p}$ ) throughout the rest of this paper, to transform a feasible solution  $\mathbf{p}$  into a valid solution:

- a) Initialization:  $x \leftarrow 0$ ,  $y \leftarrow 1$ , and  $\tilde{\mathbf{p}} \leftarrow \mathbf{p}$ ;
- b) Find the smallest index  $i$  satisfying  $\tilde{p}_i \neq 0$ , and assign the value of  $\tilde{p}_i$  to  $x$ ;
- c) For all the indices  $j \in \mathbb{Z}^+$  satisfying  $\tilde{p}_j = x$ , set  $p_j$  to  $y$ , and then set  $\tilde{p}_j$  to 0;
- d)  $y \leftarrow y + 1$ ;
- e) Repeat b) - d) until  $\tilde{\mathbf{p}} = \mathbf{0}$ , and then output  $\mathbf{p}$ , which is a valid solution.

Next we present the complete Greedy Algorithm. Let  $\mathcal{N}(\mathbf{p})$  denote the neighbourhood of a valid solution  $\mathbf{p}$ , which is defined as the set of valid solutions obtained from  $\mathbf{p}$  by performing a local change. The rule for neighbourhood generation can be chosen from a variety of existing methods [37]. For a current valid solution  $\mathbf{p}$ , our greedy algorithm changes the value of only one component to get a new solution  $\mathbf{p}'$ . If  $\mathbf{p}'$  violates the constraint of primary conflicts, we discard it. Otherwise, the realignment procedure is performed, and the outcome becomes one element of  $\mathcal{N}(\mathbf{p})$ . Starting from the initial solution, the greedy algorithm selects  $\hat{\mathbf{p}}$  from  $\mathcal{N}(\mathbf{p})$  resulting in the minimum value of the cost function  $\zeta(\mathbf{p}')$  of (29). If  $\zeta(\hat{\mathbf{p}}) < \zeta(\mathbf{p})$ , then  $\hat{\mathbf{p}}$  is accepted as the new valid solution for the next state, and this procedure is

repeated. If  $\zeta(\hat{\mathbf{p}}) \geq \zeta(\mathbf{p})$ , the algorithm is terminated, and  $\mathbf{p}$  is output as the final result. The complete procedure is summarized in **Figure 4**.

#### (D) Spatial reuse by adaptive simulated annealing

Despite its easy implementation and low complexity, the greedy algorithm cannot guarantee a globally optimal solution. Hence we propose an alternative approach using Adaptive Simulated Annealing (ASA) [38]. Simulated Annealing is a probabilistic method whose random search process mimics the physical annealing process [39]. It originates from the principles of thermodynamics associated with making a metal “freeze” into a crystalline structure at a minimum energy configuration, when cooled slowly from a state of high temperature. If its cooling process is well controlled, the algorithm can converge to a global optimum. The ASA algorithm permits an exponentially decreasing annealing schedule which greatly accelerates the optimization process compared with some traditional simulated annealing algorithms [40]. Moreover, the re-annealing procedure is introduced in ASA to adapt to changing sensitivities in the multi-dimensional parameter-space. More details on ASA are provided in **Appendix C**.

In our work, the components of  $\mathbf{p}$  take integer values, and interrelate with each other subject to the constraints induced by primary conflicts (27). Thus, the generating procedure in **Appendix C** needs to be modified. At the start of the  $(t+1)$ th state, we form a random permutation of the indices  $\{1, 2, \dots, N\}$ , denoted by a vector  $\sigma$ . Then at the  $a$ th step, we pick  $i = \sigma_a$ , and obtain the  $a$ th component of the temporary candidate solution,  $\check{p}_i^{(t+1)} = p_i^{(t)} + y_i(M-1)$ , where  $y_i \in [-1, 1]$  is a sample of a random variable  $Y_i$  with the PDF of (48). The range of the  $a$ th parameter is  $L_i \leq p_i^{(t)} \leq U_i$ , and if  $\check{p}_i^{(t+1)} \notin [L_i, U_i]$ , another sample  $y_i$  of  $Y_i$  is applied until  $\check{p}_i^{(t+1)} \in [L_i, U_i]$ . Next, we find the  $a$ th component of the candidate solution  $\hat{\mathbf{p}}^{(t+1)}$ :

---

#### Algorithm 1: Greedy Algorithm for Spatial Reuse

**Input:** Rician fading parameter:  $K$   
 Path-loss parameters:  $A$  and  $\mu$   
 Thermal noise power:  $P_{\text{noi}}$   
 Each sensor's transmit power:  $P_{\text{tra}}^{(i)}$   
 Number of sensors:  $N$   
 Number of T-F slots:  $M$   
 Adjacency and distance matrix  $\mathbf{R}$  and  $\mathbf{D}$

**Output:** A valid scheduling result  $\mathbf{p} \in \mathbb{R}^N$

- 1: Build the conflict matrix  $\mathbf{F}$  based on  $\mathbf{R}$
- 2: Obtain the initial solution  $\hat{\mathbf{p}}$  via vertex colouring, and conduct the *realignment* procedure  $\hat{\mathbf{p}} \leftarrow \text{REALIGN}(\hat{\mathbf{p}})$
- 3: **repeat**
- 4:    $\mathbf{p} \leftarrow \hat{\mathbf{p}}$
- 5:   Calculate the cost value  $\zeta(\mathbf{p})$
- 6:   Generate the *neighbourhood*  $\mathcal{N}(\mathbf{p})$  through 1-exchange and *realignment* procedure
- 7:    $\hat{\mathbf{p}} \leftarrow \arg \min_{\mathbf{p}' \in \mathcal{N}(\mathbf{p})} \zeta(\mathbf{p}')$
- 8: **until**  $\zeta(\hat{\mathbf{p}}) \geq \zeta(\mathbf{p})$
- 9: **return**  $\mathbf{p}$

---

**Figure 4.** Greedy algorithm for spatial reuse.

$$\hat{p}_i^{(t+1)} = \arg \min_{m \in \{1, 2, \dots, M\}} |\hat{p}_i^{(t+1)} - m|, \tag{30}$$

$$\text{s.t. } \forall j \in \{j \mid j \in \{\sigma_1, \sigma_2, \dots, \sigma_{a-1}\} \text{ and } \hat{p}_j^{(t+1)} = m\}: F_{i,j} = F_{j,i} = 0, \tag{31}$$

where  $\{j \mid j \in \{\sigma_1, \sigma_2, \dots, \sigma_{a-1}\} \text{ and } \hat{p}_j^{(t+1)} = m\}$  is the set of sensors scheduled into the  $m$ th slot before  $s_i$ , and thus (31) guarantees no primary conflict between  $s_i$  and any sensors scheduled before it. We generate the components of  $\hat{\mathbf{p}}^{(t+1)}$  according to the random order indicated by  $\sigma$ , and finally obtain a feasible candidate solution. If there is no valid value for  $\hat{p}_i^{(t+1)}$  at any step, the whole state will be started over with a new permutation vector  $\sigma$ . At the end of the generating procedure, we conduct the realignment procedure on  $\hat{\mathbf{p}}^{(t+1)}$  to obtain a valid solution.

In our work the acceptance and annealing procedures are the same as those described in **Appendix C**. However, we bypass the reannealing stage for temperature parameters, and only conduct this procedure on cost temperatures. This stage is optional, and since the components of  $\mathbf{p}$  are integers, there is no effective numerical method to calculate the derivatives  $f_i$  in (52). For a large numbers of parameters ( $N$ ) skipping this stage helps to significantly increase efficiency. We use the same initialization method of the greedy algorithm to get the initial value of  $\mathbf{p}$ . The ASA algorithm terminates when the predefined maximum number of generated states ( $N_{\text{gen}}^*$ ) or maximum number of accepted states ( $N_{\text{acc}}^*$ ) is reached. The ASA algorithm is summarized in **Figure 5**.

## 4. Performance Results for a Grid Network

### (A) Scenario setup

The sensors are assumed to be deployed on the grid of a target square region, with Euclidean distance between two nearest sensors fixed to  $L_0$ . We establish a Cartesian coordinate plane such that  $(0, 0)$  is in the centre of the square, and the coordinates of the sensors are  $x_i = \left(a + \frac{1 - \sqrt{N}}{2}\right)L_0$ ,  $y_i = \left(b + \frac{1 - \sqrt{N}}{2}\right)L_0$  where  $a, b \in \{0, 1, \dots, \sqrt{N} - 1\}$ ,  $i = 1 + a + b\sqrt{N}$ , and  $N$  is selected such that  $\sqrt{N}$  is an integer.

As to the partner selection scheme, the priority is given to the nearest sensors. We consider four adjacency matrix structures, and use the following cooperation levels to define them:

- i) CL0: No cooperation between sensors, and thus  $\mathbf{R}$  is an identity matrix.
- ii) CL2: A sensor selects itself and two other nearest sensors as its partners. If there are several sensors with the same distance, select in the ascending order of the sensor IDs.
- iii) CL4: A sensor selects all the sensors within  $L_0$  as its partners (including itself).
- iv) CL8: A sensor selects all the sensors within  $\sqrt{2}L_0$  as its partners (including itself).

**Algorithm 2:** ASA Algorithm for Spatial Reuse

---

**Input:** Rician fading parameter:  $K$   
 Path-loss parameters:  $A$  and  $\mu$   
 Thermal noise power:  $P_{\text{noi}}$   
 Each sensor's transmit power:  $P_{\text{tra}}^{(i)}$   
 Number of sensors:  $N$   
 Number of T-F slots:  $M$   
 Adjacency and distance matrices:  $\mathbf{R}$  and  $\mathbf{D}$   
 All the predefined ASA parameter values

**Output:** A *valid* scheduling result  $\mathbf{p} \in \mathbb{R}^N$

- 1: Build the conflict matrix  $\mathbf{F}$  based on  $\mathbf{R}$
- 2: Generate an initial solution  $\mathbf{p}$ , and calculate  $\zeta(\mathbf{p})$
- 3: Sample the parameter space to calculate the *initial cost temperature*  $T_{\text{cost}}(0)$ .
- 4:  $N_{\text{gen}} \leftarrow 0, N_{\text{acc}} \leftarrow 0$
- 5: **while**  $N_{\text{gen}} < N_{\text{gen}}^*$  and  $N_{\text{acc}} < N_{\text{acc}}^*$  **do**
- 6:   Form a random sorting vector  $\sigma$
- 7:   Generate the components of the *candidate* solution  $\hat{\mathbf{p}}$  according to the order indicated by  $\sigma$  (See Section 3(C) for details). Whenever a component fails to find any valid value, **go to** Step 6.
- 8:    $\hat{\mathbf{p}} \leftarrow \text{REALIGN}(\hat{\mathbf{p}})$
- 9:    $N_{\text{gen}} \leftarrow N_{\text{gen}} + 1$
- 10:   Conduct *acceptance* procedure represented by (49) to determine whether  $\mathbf{p} \leftarrow \hat{\mathbf{p}}$ .
- 11:   If  $\hat{\mathbf{p}}$  is accepted,  $N_{\text{acc}} \leftarrow N_{\text{acc}} + 1$
- 12:   Conduct *annealing* procedure to update  $T_i$  and  $T_{\text{cost}}$  as shown in (50) and (51).
- 13:   **if**  $N_{\text{gen}} = zm_{\text{gen}}$  or  $N_{\text{acc}} = zm_{\text{acc}} (z \in \mathbb{Z}^+)$  **then**
- 14:     conduct *reannealing* procedure to rescale the *cost temperature* as shown in (53)
- 15:   **end if**
- 16: **end while**
- 17: **return**  $\mathbf{p}$

---

**Figure 5.** ASA algorithm for spatial reuse.

We choose the COST-Walfish-Ikegami (COST-WI) model as the empirical path-loss model in the WSN, with parameters set for the Urban Macro scenario [41]. This model can be viewed as a simplified path-loss model in (1), with  $A = 10^{-3.018}$ ,  $\mu = 2.6$ , and  $20 \text{ m} \leq d \leq 5 \text{ km}$ . In addition, the average received SNR of the primary signal,  $\bar{\gamma}$ , is assumed to be identical at each sensor, which is a reasonable assumption when generating numerical or simulation results as in [8] [42]. Furthermore, each sensor employs the same transmit power  $P_{\text{tra}}$ , and energy detection threshold  $\lambda$ . The scenario setup parameters are summarized in Table 1.

The C-language implementation of the ASA algorithm is based on the source code provided by A. L. Ingber, which is available at <<http://www.ingber.com/#ASA>>. We made some modifications as described in Section 3(D), where we tuned the ASA parameters and options according to the specifics of our optimization problem, which are summarized in Table 2.

**(B) Features of the grid network**

We first build an undirected grid graph by considering all the sensors as



**Table 1.** Parameter settings for numerical results.

Parameter Name	Value
Minimum Distance between Sensors ( $L_0$ )	50m
Path-loss Constant ( $A$ )	$10^{-3.018}$
Path-loss Exponent ( $\mu$ )	2.6
Rician Fading Parameter ( $K$ )	7
Sensor's Transmit Power ( $P_{ts}$ )	100mW
Communication Subcarrier Bandwidth ( $W_c$ )	20 kHz
Noise Power Spectral Density ( $N_0/2$ )	$5 \times 10^{-16}$ W/Hz

**Table 2.** Key parameters and options of ASA.

Parameter or Option Name	Value
Maximum Number of Generated States ( $N_{gen}^*$ )	$10^7$
Maximum Number of Accepted States ( $N_{acc}^*$ )	$10^6$
Number of Cost Samples ( $N_{spl}$ )	5
Temperature Ratio Scale ( $\alpha_1$ )	$10^{-4}$
Temperature Anneal Scale ( $\alpha_2$ )	100
Cost Parameter Scale Ratio ( $\beta$ )	1.0
Generated Frequency Modulus ( $m_{gen}$ )	5000
Accepted Frequency Modulus ( $m_{acc}$ )	50
Initial Parameter Temperature ( $T_i(0)$ )	1.0
Parameter Quenching Factor ( $Q_i$ )	$N/10$
Cost Quenching Factor ( $Q_{cost}$ )	$N/10$
Include_Integer_Parameters	TRUE
Reanneal_Parameters	FALSE
QUENCH_PARAMETERS	TRUE
QUENCH_COST	TRUE
QUENCH_PARAMETERS_SCALE	FALSE
QUENCH_COST_SCALE	FALSE

vertices, with an edge between any two sensors whose Euclidean distance is  $L_0$ . The grid distance between  $s_i$  and  $s_j$  is defined as the number of edges in a shortest path connecting them in the grid graph.

-For CL2 and CL4, the grid distance between any partner-receiver pair is 1. If no two sensors within grid distance 2 share the same slot, no primary conflict

exists in the WSN.

-For CL8, the longest grid distance between any partner-receiver pair is 2. If no two sensors within grid distance 4 share the same slot, no primary conflict exists in the WSN.

If the T-F slots are viewed as “colours”, then for CL2 and CL4, a 2-distance colouring of the grid graph is a feasible solution, and for CL8, a 4-distance colouring is a feasible solution. The  $k$ -distance colouring is defined as a vertex colouring such that no two vertices lying at the grid distance less than or equal to  $k$  are assigned the same colour. The minimum number of colours necessary for the  $k$ -distance colouring of a grid graph, denoted by  $\chi_k$ , is given by [43]:

$$\chi_k = \begin{cases} \left( (k+1)^2 + 1 \right) / 2 & \text{if } k \text{ is even} \\ (k+1)^2 / 2 & \text{if } k \text{ is odd} \end{cases} \quad (32)$$

In summary, for the grid network with the partner selection scheme described above, if the longest grid distance between any partner--receiver pair is  $\kappa$ , then a  $2\kappa$ -distance colouring of the grid graph is a feasible solution (no primary conflict). The minimum number of colours (T-F slots) needed to complete the  $2\kappa$ -distance colouring is  $\chi_{2\kappa} = \left( (2\kappa+1)^2 + 1 \right) / 2$ . Thus, if  $M \geq \left( (2\kappa+1)^2 + 1 \right) / 2$ , we can guarantee that there is a scheduling solution without primary conflict. Then, we can set the minimum value of  $M$  (number of T-F slots) as  $\chi_{2\kappa}$ , which is 5 for CL2 and CL4, and 13 for CL8.

In addition, for the grid network, we introduce an alternative method to generate the initial valid solution, which is based on a deterministic approach of  $k$ -distance colouring:

- a) Find  $\hat{k} = \max \{ k \in \mathbb{Z}^+ \mid \chi_k \leq M \}$ , where  $\chi_k$  is calculated by (32).
- b) Obtain an initial solution  $\mathbf{p}$  by setting its components as follows:

$$p_i = \begin{cases} a + (\hat{k} + 1)b \bmod \left( (\hat{k} + 1)^2 + 1 \right) / 2 & \text{if } \hat{k} \text{ is even} \\ a + \hat{k}b \bmod (\hat{k} + 1)^2 / 2 & \text{if } \hat{k} \text{ is odd} \end{cases} \quad (33)$$

where  $a, b \in \{0, 1, \dots, \sqrt{N} - 1\}$ ,  $i = 1 + a + b\sqrt{N}$ , and “mod” represents the modulo operator.

- c) Realignment procedure:  $\mathbf{p} \leftarrow \text{REALIGN}(\mathbf{p})$

The scheduling  $\mathbf{p}$  of (33) is equivalent to a  $\hat{k}$ -distance colouring of the grid graph [43]. In addition,  $\hat{k} \geq 2\kappa$  because we have set  $M \geq \chi_{2\kappa}$ . Thus, the solution  $\mathbf{p}$  of step b) ensures that there is no primary conflict. The initialization by  $k$ -distance colouring is more concise and efficient than that via DSatur algorithm for a grid network.

### (C) Verification of the algorithms

This example shows how the greedy algorithm may be trapped in local minima, and how the ASA algorithm can reach the global optimum. We consider a small network with 9 sensors as illustrated in Section 4(A), and set  $M = 5$ . The adjacency matrix is constructed as

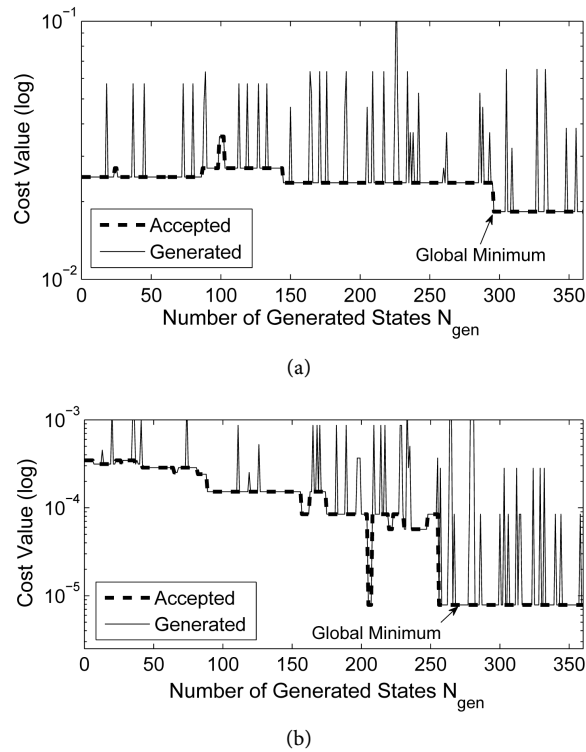
$$R = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

The number of partners of each sensor is  $n_i = 3$ , and we set the fusion factor  $\Omega$  to be 0.1 or 0.5. Then, the fusion threshold  $k_i$  of each sensor is  $k_i = \lceil \Omega n_i \rceil = 1$  or 2. When  $k_i = 3$ , the cost function (29) is the same as that of  $k_i = 1$ , and thus we do not need to use  $k_i = 3$ . Through exhaustive search, we find that there are 111 valid solutions out of 1935123 feasible solutions. When  $\Omega = 0.1$ , the global minimum is:  $\zeta(\mathbf{p}^*) = 0.01830289$ , with  $\mathbf{p}^* = (1, 2, 3, 3, 4, 5, 5, 2, 1)$  or  $\mathbf{p}^* = (1, 2, 3, 4, 5, 1, 3, 2, 4)$ . When  $\Omega = 0.5$ , the global minimum is:  $\zeta(\mathbf{p}^*) = 7.853152 \times 10^{-6}$ , with  $\mathbf{p}^* = (1, 2, 3, 4, 2, 5, 3, 2, 1)$ . The initial solution is obtained by  $k$ -distance colouring as in Section 4(B), which is  $\mathbf{p} = (1, 2, 3, 4, 5, 1, 2, 3, 4)$ . The greedy algorithm generates  $N(M - 1) = 45$  solutions to obtain the neighbourhood of a current solution, and accept the best solution from it. If the current solution is smaller than any of the neighbour solutions, the greedy algorithm terminates and outputs the current solution.

-When  $\Omega = 0.1$ , the initial solution is  $\zeta(\mathbf{p}) = 0.02491$ , with  $\mathbf{p} = (1, 2, 3, 4, 5, 1, 2, 3, 4)$ . Then the greedy algorithm generates the neighbourhood of this initial solution, and finds 8 new valid solutions. However, the initial cost value 0.02491 is smaller than any of the 8 neighbours, and the greedy algorithm gets stuck at this local minimum ( $0.02491 > 0.01830289$ ).

-When  $\Omega = 0.5$ , the initial solution is  $\zeta(\mathbf{p}) = 0.0003457747$ , with  $\mathbf{p} = (1, 2, 3, 4, 5, 1, 2, 3, 4)$ . This time the greedy algorithm gets stuck in the local minimum  $\zeta(\mathbf{p}) = 5.70172 \times 10^{-5}$ , with  $\mathbf{p} = (1, 2, 3, 4, 2, 5, 3, 1, 4)$  after 180 solutions are generated.

Next, we apply a relatively slow ASA algorithm with  $\alpha_1 = 0.01$ ,  $\beta = 0.25$ , and QUENCHing options turned off. Moreover, we adjust the generating procedure to change only one random component at each state, which results in the same neighbourhood structure as the greedy algorithm. At each state, the ASA algorithm generates a candidate solution, whose cost value is represented by the solid line in **Figure 6(a)** and **Figure 6(b)**, and the dark dash line shows the acceptance procedure illustrated in **Appendix C**. If the generated cost value is larger than the current one, the ASA algorithm allows to accept the worse cost value at certain points. This helps the ASA algorithm to escape from a local minimum. At the end, when  $\Omega = 0.1$  and  $\Omega = 0.5$ , the ASA algorithm reaches the following global minimum when  $N_{\text{gen}} = 296$  and  $N_{\text{gen}} = 205$ , which are the same as those obtained from exhaustive search:



**Figure 6.** The cost value (log) generated and accepted at each state of the relatively slow ASA algorithms for (a)  $\Omega = 0.1$ ; and (b)  $\Omega = 0.5$ .

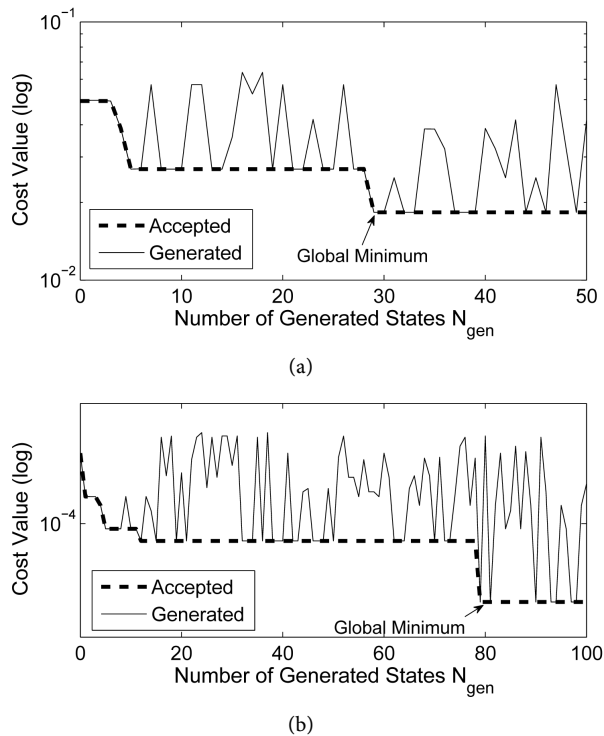
-When  $\Omega = 0.1$ ,  $\zeta(\mathbf{p}^*) = 0.01830289$ , with  $\mathbf{p}^* = (1, 2, 3, 4, 5, 1, 3, 2, 4)$ .

-When  $\Omega = 0.5$ ,  $\zeta(\mathbf{p}^*) = 7.853152 \times 10^{-6}$ , with  $\mathbf{p}^* = (1, 2, 3, 4, 2, 5, 3, 2, 1)$ .

We also apply the normal generating procedure illustrated in Section 3(C) with realignment and the ASA parameters summarized in Table 2. The cost values generated and accepted at each state are shown in Figure 7, in the same form as Figure 6. If  $\Omega = 0.1$ , it arrives at the global minimum 0.01830289 when  $N_{gen} = 29$ , and if  $\Omega = 0.5$ , it arrives at the global minimum  $7.853152 \times 10^{-6}$  when  $N_{gen} = 79$ , showing that the ASA algorithm is efficient when its parameters are well-tuned.

**(D) ROC curves for different configurations**

We analyse the spectrum sensing performance of the grid WSN through ROC curves, which display  $Q_d$  vs  $Q_f$ . The quantities  $Q_d$  and  $Q_f$  represent the network probabilities of detection and false alarm averaged over sensor locations,  $Q_d = \frac{1}{N} \sum_{i=1}^N Q_{d,i}(k_i)$ , and  $Q_f = \frac{1}{N} \sum_{i=1}^N Q_{f,i}(k_i)$ , where  $Q_{d,i}$  and  $Q_{f,i}$  are defined in (11) and (12). We use the fusion factor  $\Omega$  to set the fusion threshold  $k_i$  of each sensor, as  $k_i = \lceil \Omega n_i \rceil$ , where  $n_i = \left| \left\{ s_j \mid R_{j,i} = 1 \right\} \right|$  is the number of partners of  $s_i$ . For the partner selection schemes introduced in Section 4(A), we have  $R_{i,i} = 1$ . Since there is no communication link when a sensor employs its own sensing result,  $\varepsilon_{i,i} = 0$ . Then when using the OR fusion rule, according to (16), the maximum value of  $Q_{d,i}(1)$  or  $Q_{f,i}(1)$  is



**Figure 7.** The cost value (log) generated and accepted at each state of the normal ASA algorithms for (a)  $\Omega = 0.1$ ; and (b)  $\Omega = 0.5$ .

$$Q_{d,i}^{[u]}(1) = Q_{f,i}^{[u]}(1) = 1 - \prod_{s_j \in \{s_j | R_{j,i} = 1\}} \varepsilon_{j,i} = 1 - \varepsilon_{i,i} \prod_{s_j \in \{s_j | R_{j,i} = 1, j \neq i\}} \varepsilon_{j,i} = 1 - 0 = 1. \quad (34)$$

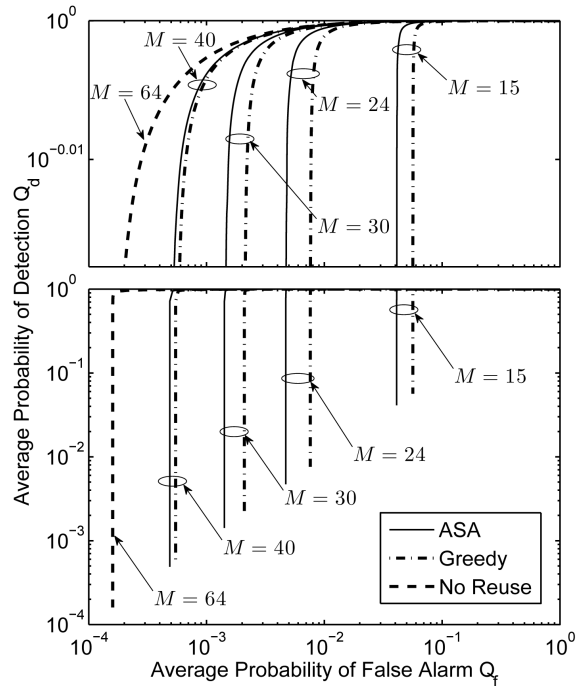
Thus, the maximum value of  $Q_d$  or  $Q_f$  when using OR rule is

$$Q_d^{[u]} = Q_f^{[u]} = \frac{1}{N} \sum_{i=1}^N 1 = 1.$$

In each case, we change one network parameter while fixing others, and use both greedy and ASA algorithms to realize spatial reuse of T-F slots. The  $k$ -distance colouring scheme is used for initialization methods with both algorithms. After obtaining the output of the protocol,  $\mathbf{p} = (p_1, p_2, \dots, p_N)$ , we calculate the BEP  $\varepsilon_{j,i}(\mathbf{p})$  of every communication link  $s_j \rightarrow s_i$  based on (21). Then we can calculate  $Q_d$  and  $Q_f$  using (6), (7), (9), (10), (11), and (12).

Different numbers of T-F slots

In **Figure 8** we present the ROC curves for different values of  $M$  with  $\bar{\gamma} = 10$  dB,  $N = 64$ ,  $K = 7$ , CL8, and  $\Omega = 0.1$  (OR fusion rule). As  $M$  increases, the sensors sharing one slot can be more spatially separated, resulting in lower BEPs on communication links. In turn, it helps increase  $Q_d$  with the same  $Q_f$ , and lowers  $Q_f^{[L]}$ , which can be seen by comparing the curves of the same line style. Comparing the curves of different line styles with the same  $M$ , we see that the difference between the ASA and greedy algorithms becomes more significant as  $Q_f$  decreases from 1 to  $Q_f^{[L]}$  of the greedy algorithm, where it is outperformed by the ASA algorithm. As  $M$  increases, the difference between the ROC curves of the two algorithms becomes smaller. When  $M = N = 64$ , no spatial reuse is needed, providing an upper bound to spectrum



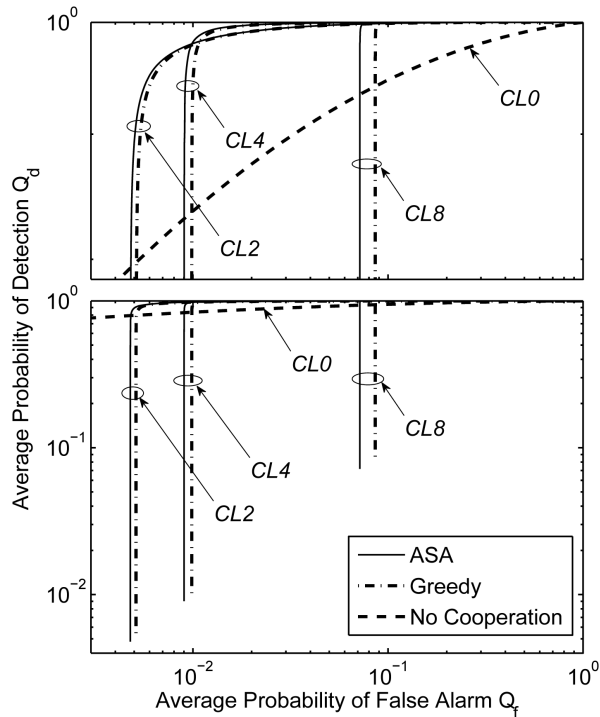
**Figure 8.** Average  $Q_d$  vs  $Q_f$  for different numbers of T-F slots  $M$  with  $\bar{\gamma} = 10$  dB,  $N = 64$ ,  $K = 7$ , CL8, and OR decision fusion rule.

sensing performance, that serves as a benchmark. At  $Q_f = 0.1$ , the ASA and the greedy algorithms with  $M = 15$  provide a value for  $Q_d$  that are nearly the same as the benchmark. At  $Q_f = 0.01$ , the difference between  $Q_d$  of the ASA and the greedy algorithms with  $M = 40$  and the benchmark can also be ignored. We can save at least 76.6% and 37.5% of T-F slots if the requirements of  $Q_f$  are 0.1 and 0.01 respectively.

#### Different cooperation levels

We evaluate the performance with different cooperation levels (defined in Section 4(A) in **Figure 9**, where  $N = 100$ ,  $M = 15$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB, and  $\Omega = 0.1$  (OR rule). No T-F slot is needed for CL0, acting as a benchmark for other cooperation levels. Comparing the curves of different line styles at the same cooperation level, we see that the difference between the ASA and greedy algorithms is mainly manifested in the  $Q_f$  range between their lower bounds. For example, using CL8, the difference between the curves when  $Q_f > 0.08608787$  is nearly negligible, and when  $0.07173913 \leq Q_f < 0.08608787$ , the ASA algorithm outperforms the greedy algorithm. Moreover, when the cooperation level increases, the difference between the values of  $Q_f^{[L]}$  increases.

Next, we focus on the curves of one algorithm at different cooperation levels. When the cooperation level increases, the existing partners for the lower cooperation level remain in the set  $\{s_j | R_{j,i} = 1\}$ , while new partners are added. Because the number of slots  $M$  is fixed, the interference caused by spatial reuse in the network remains nearly unchanged, and the BEP from an existing partner for the lower cooperation level varies very little. Then from (15), we see that  $Q_f^{[L]}$  increases with the cooperation level, which is confirmed by the curves in



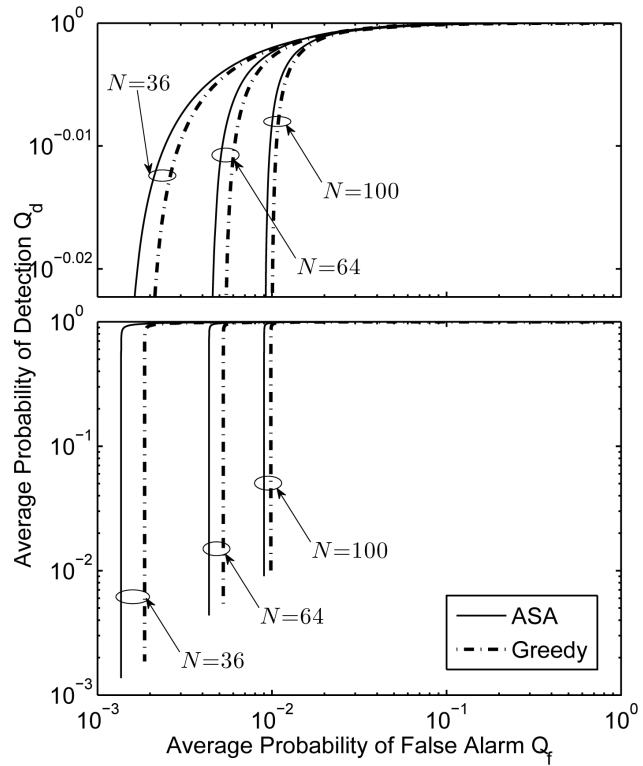
**Figure 9.** Average  $Q_d$  vs  $Q_f$  for different cooperation levels with  $N = 100$ ,  $M = 15$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB and  $\Omega = 0.1$  (OR decision fusion rule).

**Figure 9.** When  $Q_f$  is higher than  $Q_f^{[L]}$ , cooperation between sensors helps increase  $Q_d$  compared with CL0. However, the difference between CL2, CL4 and CL8 in the high  $Q_f$  range ( $Q_f \geq 0.1$ ) is very small. For the case shown in **Figure 9**, we can divide the  $Q_f$  axis into 4 intervals with boundaries close to  $Q_f^{[L]}$  at each cooperation level, and find the best choice of cooperation levels in each interval. Taking the ASA curves as an example, when  $0.0916 < Q_f < 1$ , CL2, CL4 and CL8 perform nearly the same. We see that CL4 is the best for  $0.00993 < Q_f < 0.0916$ , CL2 is the best for  $0.00478 < Q_f < 0.00993$ , and if the system requires  $Q_f < 0.00478$ , then CL0 is the only choice.

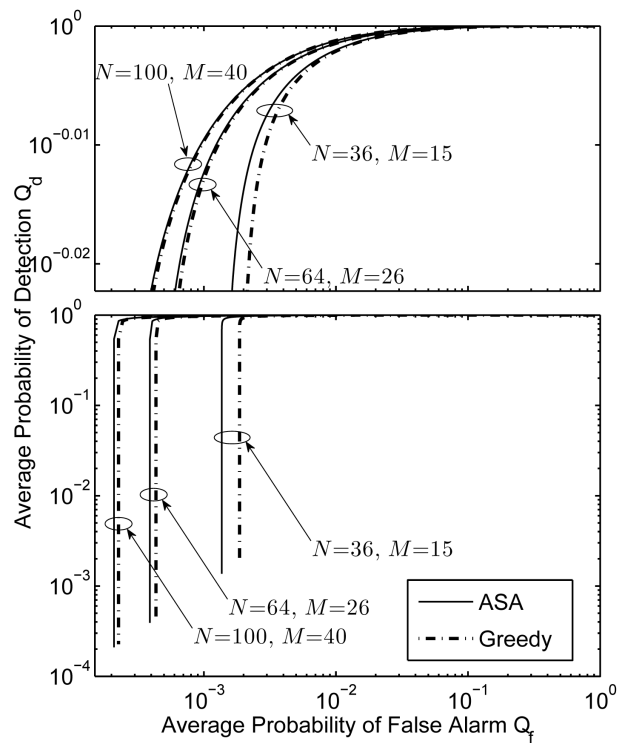
Different network sizes

Three networks sizes,  $N = 36, 64, 100$  are analyzed in **Figure 10** and **Figure 11**, with  $K = 7$ , CL4,  $\bar{\gamma} = 10$  dB, and  $\Omega = 0.1$  (OR rule). In **Figure 10**, we fix  $M = 15$ , and in **Figure 11** we fix  $M/N$  to 0.4. We see that when  $Q_f > 0.02$ , the difference between the two algorithms is very small. In relatively low  $Q_f$  range, for the same value of  $Q_f$ , the ASA algorithm results in higher  $Q_d$  compared to the greedy algorithm. For  $N = 36$  or  $N = 64$ , the ASA algorithm outperforms the greedy algorithm when  $Q_f < 0.01$ , and for  $N = 100$ , the difference is identifiable when  $Q_f < 0.02$ .

For CL4, a sensor located at the corner has 3 partners, the one on the edge has 4 partners, while any other sensor has 5 partners. When  $N$  is increased, the proportion of sensors on the edge  $4(\sqrt{N} - 1)/N$ , as well as that at the corner  $4/N$ , decreases, resulting in increasing average number of partners of each sensor. When  $M$  is fixed, larger  $N$  means an increasing average numbers of



**Figure 10.** Average  $Q_d$  vs  $Q_f$  for different numbers of sensors  $N$  with  $M = 15$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB, CL4, and  $\Omega = 0.1$  (OR decision fusion rule).



**Figure 11.** Average  $Q_d$  vs  $Q_f$  for different numbers of sensors  $N$  with  $M/N \approx 0.4$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB, CL4, and  $\Omega = 0.1$  (OR decision fusion rule).



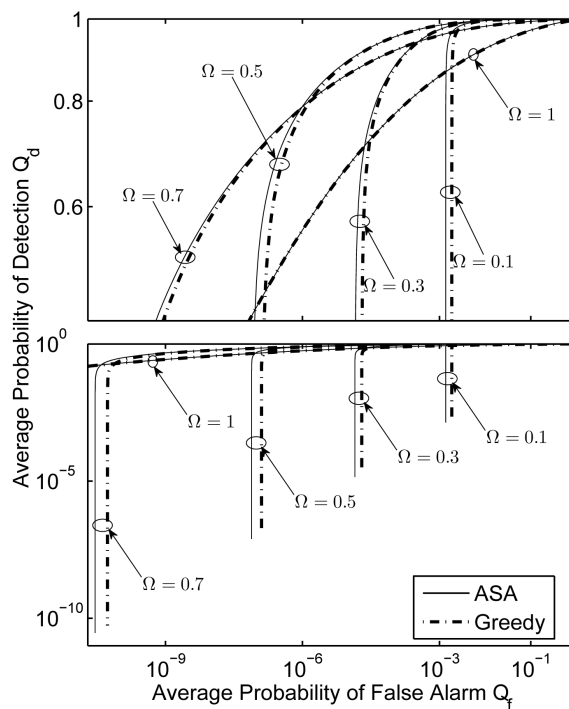
sensors sharing the same slot. Thus, the network interference level increases, resulting in higher average BEP of communication links. As shown in **Figure 10**, for relatively high  $Q_f$  the gain brought by more partners is counteracted by the impact of worse reporting channels. In addition, according to (15), more partners and higher BEPs of reporting channels lead to a higher  $Q_f^{[L]}$ . When the ratio  $M/N$  is kept constant, **Figure 11** shows that the performance improves with increasing  $N$ . This is because in a larger network, although the average number of sensors sharing one T-F slot remains almost unchanged, the average distance between them increases, which leads to a lower network interference level and lower average reporting BEP.

Different decision fusion rules

The ROC curves for fusion factor values  $\Omega = 0.1, 0.3, 0.5, 0.7, 1$  are presented in **Figure 12**, where  $N = 64$ ,  $M = 15$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB and CL4 is used. For CL4, sensors at different locations in the network may have different numbers of partners  $n_i$ . For the 4 sensors in the corner  $n_i = 3$ , for the  $4(\sqrt{N} - 1) = 20$  sensors on the edge (but not in the corner)  $n_i = 4$ , and the other  $N - 4\sqrt{N} = 12$  sensors have  $n_i = 5$ . The decision fusion threshold  $k_i$  at each sensor is  $k_i = \lceil \Omega n_i \rceil$ , and we have

$$\text{For } n_i = 3, 4 \text{ or } 5, n_i + 1 - k_i|_{\Omega=0.7} = k_i|_{\Omega=0.3}, \text{ and } n_i + 1 - k_i|_{\Omega=1} = k_i|_{\Omega=0.1}. \quad (35)$$

Thus, according to Property 4,  $\Omega = 0.1$  and  $\Omega = 1$  (or  $\Omega = 0.3$  and  $\Omega = 0.7$ ) result in the same Achievable Range  $\Delta_i$  of each sensor in this scenario. Then based on (24) and (29),  $\Omega = 0.1$  and  $\Omega = 1$  have the same cost function, and



**Figure 12.** Average  $Q_d$  vs  $Q_f$  for different decision fusion factors  $\Omega = 0.1, 0.3, 0.5, 0.7$  and 1 with  $N = 64$ ,  $M = 15$ ,  $K = 7$ ,  $\bar{\gamma} = 10$  dB, and CL4.

so do  $\Omega = 0.3$  and  $\Omega = 0.7$ . Thus,  $\Omega = 0.1$  and  $\Omega = 1$  have the same scheduling result, and also  $\Omega = 0.3$  and  $\Omega = 0.7$  have the same scheduling result.

In **Figure 12**, we see that the separation between the ROC curves of the two algorithms becomes smaller as  $\Omega$  increases, and we can hardly differentiate the ROC curves when  $\Omega = 1$ . Comparing the curves for the same algorithm, we see that  $Q_f^{[L]}$  and  $Q_d^{[U]}$  decrease with increasing  $\Omega$ , following Property 3. For design purposes we have to choose  $\Omega$  according to the specific  $Q_f$  or  $Q_d$  requirements and the ROC curves. Consider the ROC curves for the ASA algorithm for example, when the required  $Q_f$  is in the range  $Q_f > 0.01$ , the ROC curves of  $\Omega = 0.1, 0.3$  and  $0.5$  are nearly the same, and result in higher  $Q_d$  than  $\Omega = 0.7$  or  $1$ . If the system requirement is  $Q_f < 0.001$ ,  $\Omega = 0.1$  can not achieve this  $Q_f$  and  $\Omega = 0.5$  is better than  $\Omega = 0.3$ . In this scenario, for  $10^{-6} < Q_f < 1$  we can always choose  $\Omega = 0.5$ , and  $Q_f = 1$  is the worst choice.

## 5. Conclusions

In this paper, we analysed cooperative spectrum sensing performance using the  $k$ -out-of- $n$  decision fusion rule with imperfect reporting channels, and derived the upper and lower bounds of probabilities of detection and false alarm. Then, we introduced a DetF distributed WSN for cooperative spectrum sensing, and proposed a spatial reuse MAC protocol based on TDMA/OFDMA. Two design approaches for the MAC protocol were considered: greedy and ASA algorithms. Finally, for a grid WSN, we discussed how to construct the adjacency matrix characterizing the cooperating relations between sensors. We also explain how to determine the minimum number of T-F slots, and get an initial valid solution via a  $k$ -distance colouring method.

Numerical results for a grid network in Section 4 show how the spectrum sensing performance is influenced by system parameters, and also present a comparison of the ASA and greedy algorithms. The ROC curves in **Figures 8-11** show that using the OR decision fusion rule, the difference between the curves of the two algorithms is very small in relatively high  $Q_f$  range, and the advantage of the ASA algorithm can be seen when  $Q_f$  is around and smaller than the lower bound  $Q_f^{[L]}$  of the greedy algorithm. The ROC curves in **Figure 12** show that the separation between the ROC curves of the ASA and greedy algorithms decreases as the decision fusion factor  $\Omega$  increases, and when  $\Omega = 1$  the two curves become virtually the same.

When comparing the ROC curves of one algorithm with different network settings, we can see the influence of system parameters. From **Figure 8** we see that our spatial reuse MAC protocol can save T-F slots without significant impact on the sensing performance at certain  $Q_f$ . We also see that the lower bounds  $Q_d^{[L]}$  and  $Q_f^{[L]}$  decrease when the number of T-F slots is increased. From **Figure 9** we see that cooperation between sensors helps improve the spectrum sensing performance when  $Q_f$  is higher than the lower bound  $Q_f^{[L]}$ . However, the lower bound  $Q_f^{[L]}$  increases with the number of partners of each sensor. For example, the difference between the ROC curves of CL4 and CL8 is

nearly negligible when  $Q_f > 0.07$ , and  $Q_f < 0.07$  is not achievable by CL8. Thus, we should not aim at increasing the number of partners of each sensor as much as possible. From **Figure 10** we see that if the number of T-F slots is fixed, a larger network size leads to an increased lower bound  $Q_f^{[L]}$ , and **Figure 11** shows that if the ratio  $M/N$  is fixed, a larger network size gives better sensing performance. In **Figure 12**, it is not a clear cut which fusion factor  $\Omega$  is the best in general. We can use the ROC curves to decide the value of  $\Omega$  according to the specific required  $Q_f$  and  $Q_d$ . For instance, if the system required  $Q_f$  is 0.001 then  $\Omega = 0.5$  is the best choice. In addition, if the number of T-F slots cannot be changed for a certain network size, adjusting the cooperation level or decision fusion threshold is necessary to improve the sensing performance subject to  $Q_f$  and  $Q_d$  requirements.

## References

- [1] Letaief, K. and Zhang, W. (2009) Cooperative Communications for Cognitive Radio Networks. *Proceedings of the IEEE*, **97**, 878-893. <https://doi.org/10.1109/JPROC.2009.2015716>
- [2] Akyildiz, I.F., Lee, W.-Y., Vuran, M.C. and Mohanty, S. (2006) NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey. *Computer Networks*, **50**, 2127-2159. <https://doi.org/10.1016/j.comnet.2006.05.001>
- [3] Stevenson, C.R., *et al.* (2009) IEEE 802.22: The First Cognitive Radio Wireless Regional Area Network Standard. *IEEE Communications Magazine*, **47**, 130-138. <https://doi.org/10.1109/MCOM.2009.4752688>
- [4] Abinader, F., Almeida, E.P., Chaves, F.S., Cavalcante, A.M., Vieira, R.D., Paiva, R.C., Sobrinho, A.M., Choudhury, S., Tuomaala, E., Doppler K., *et al.* (2014) Enabling the Coexistence of LTE and Wi-Fi in Unlicensed Bands. *Communications Magazine, IEEE*, **52**, 54-61. <https://doi.org/10.1109/MCOM.2014.6957143>
- [5] Zhang, H., Chu, X., Guo, W. and Wang, S. (2015) Coexistence of Wi-Fi and Heterogeneous Small Cell Networks Sharing Unlicensed Spectrum. *Communications Magazine, IEEE*, **53**, 158-164. <https://doi.org/10.1109/MCOM.2015.7060498>
- [6] Tandra, R. and Sahai, A. (2008) SNR Walls for Signal Detection. *IEEE Journal of Selected Topics in Signal Processing*, **2**, 4-17. <https://doi.org/10.1109/JSTSP.2007.914879>
- [7] Duan, D., Yang, L. and Principe, J.C. (2010) Cooperative Diversity of Spectrum Sensing for Cognitive Radio Systems. *IEEE Transactions on Signal Processing*, **58**, 3218-3227. <https://doi.org/10.1109/TSP.2010.2044612>
- [8] Atapattu, S., Tellambura, C. and Jiang, H. (2011) Energy Detection Based Cooperative Spectrum Sensing in Cognitive Radio Networks. *IEEE Transactions on Wireless Communications*, **10**, 1232-1241. <https://doi.org/10.1109/TWC.2011.012411.100611>
- [9] Mishra, S.M., Sahai, A. and Brodersen, R. (2006) Cooperative Sensing among Cognitive Radios. 2006 *IEEE International Conference on Communications*, Istanbul, 11-15 June 2006, 1658-1664. <https://doi.org/10.1109/ICC.2006.254957>
- [10] Han, Z., Fan, R. and Jiang, H. (2009) Replacement of Spectrum Sensing in Cognitive Radio. *IEEE Transactions on Wireless Communications*, **8**, 2819-2826. <https://doi.org/10.1109/TWC.2009.080603>
- [11] Shankar, S.N., Cordeiro, C. and Challapali, K. (2005) Spectrum Agile Radios: Utilization and Sensing Architectures. *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, 8-11 November 2005,

160-169.

- [12] Sahai, A., Mishra, S.M., Tandra, R. and Hoven, N. (2006) Sensing for Communication: The Case of Cognitive Radio. *Allerton Conference on Communication, Control, and Computing*, Allerton Park, Monticello, 27-29 September 2006, 1035-1037.
- [13] Akyildiz, I.F., Lo, B.F. and Balakrishnan, R. (2011) Cooperative Spectrum Sensing in Cognitive Radio Networks: A Survey. *Physical Communication*, **4**, 40-62. <https://doi.org/10.1016/j.phycom.2010.12.003>
- [14] Bachir, A., Dohler, M., Watteyne, T. and Leung, K.K. (2010) MAC Essentials for Wireless Sensor Networks. *IEEE Communications Surveys and Tutorials*, **12**, 222-248. <https://doi.org/10.1109/SURV.2010.020510.00058>
- [15] Demirkol, I., Ersoy, C. and Alagoz, F. (2006) MAC Protocols for Wireless Sensor Networks: A Survey. *IEEE Communications Magazine*, **6**, 115-121. <https://doi.org/10.1109/MCOM.2006.1632658>
- [16] Chaudhari, S., Lunden, J., Koivunen, V. and Poor, H.V. (2012) Cooperative sensing with Imperfect Reporting Channels: Hard Decisions or Soft Decisions? *IEEE Transactions on Signal Processing*, **60**, 18-28. <https://doi.org/10.1109/TSP.2011.2170978>
- [17] Goldsmith, A. (2005) *Wireless Communications*. Cambridge University Press, New York. <https://doi.org/10.1017/CBO9780511841224>
- [18] Simon, M.K. and Alouini, M.-S. (2005) *Digital Communication over Fading Channels*. John Wiley & Sons, Hoboken.
- [19] Iyer, A., Rosenberg, C. and Karnik, A. (2009) What Is the Right Model for Wireless Channel Interference. *IEEE Transactions on Wireless Communications*, **8**, 2662-2671. <https://doi.org/10.1109/twc.2009.080720>
- [20] Ghasemi, A. and Sousa, E.S. (2005) Collaborative Spectrum Sensing for Opportunistic Access in Fading Environments. *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, 8-11 November 2005, 131-136.
- [21] Digham, F.F., Alouini, M.-S. and Simon, M.K. (2003) On the Energy Detection of Unknown Signals over Fading Channels. *IEEE International Conference on Communications*, **5**, 3575-3579. <https://doi.org/10.1109/icc.2003.1204119>
- [22] Varshney, P.K. (1997) *Distributed Detection and Data Fusion*. In: Burrus, C., Ed., Springer, New York. <https://doi.org/10.1007/978-1-4612-1904-0>
- [23] Hong, Y. (2013) On Computing the Distribution Function for the Poisson Binomial Distribution. *Computational Statistics and Data Analysis*, **59**, 41-51. <https://doi.org/10.1016/j.csda.2012.10.006>
- [24] Fernandez, M. and Williams, S. (2010) Closed-Form Expression for the Poisson-Binomial Probability Density Function. *Aerospace and Electronic Systems, IEEE Transactions on*, **46**, 803-817. <https://doi.org/10.1109/TAES.2010.5461658>
- [25] Cidon, I. and Sidi, M. (1989) Distributed Assignment Algorithm for Multihop Packet Radio Networks. *IEEE Transactions on Computers*, **38**, 1353-1361. <https://doi.org/10.1109/12.35830>
- [26] Sagduyu, Y. and Ephremides, A. (2004) The Problem of Medium Access Control in Wireless Sensor Networks. *IEEE Wireless Communications*, **11**, 44-53. <https://doi.org/10.1109/MWC.2004.1368896>
- [27] Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A. and Protasi, M. (2012) *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer Science & Business Media, Berlin.

- [28] Maniezzo, V. and Carbonaro, A. (2000) An Ants Heuristic for the Frequency Assignment Problem. *Future Generation Computer Systems*, **16**, 927-935. [https://doi.org/10.1016/S0167-739X\(00\)00046-7](https://doi.org/10.1016/S0167-739X(00)00046-7)
- [29] Krumke, S.O., Marathe, M.V. and Ravi, S. (2001) Models and Approximation Algorithms for Channel Assignment in Radio Networks. *Wireless Networks*, **7**, 575-584. <https://doi.org/10.1023/A:1012311216333>
- [30] ElBatt, T. and Ephremides, A. (2004) Joint Scheduling and Power Control for Wireless Ad Hoc Networks. *IEEE Transactions on Wireless Communications*, **3**, 74-85. <https://doi.org/10.1109/TWC.2003.819032>
- [31] Alicherry, M., Bhatia, R., and Li, L.E. (2005) Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks. *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking*, 28 August-2 September, 2005, 58-72. <https://doi.org/10.1145/1080829.1080836>
- [32] Subramanian, A.P., Gupta, H., Das, S.R. and Cao, J. (2008) Minimum interference Channel Assignment in Multiradio Wireless Mesh Networks. *Mobile Computing, IEEE Transactions on*, **7**, 1459-1473. <https://doi.org/10.1109/TMC.2008.70>
- [33] Papadimitriou, C.H. and Steiglitz, K. (1998) *Combinatorial Optimization: Algorithms and Complexity*. Courier Corporation, North Chelmsford.
- [34] Blum, C. and Roli, A. (2003) Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys (CSUR)*, **35**, 268-308. <https://doi.org/10.1145/937503.937505>
- [35] Brelaz, D. (1979) New Methods to Color the Vertices of a Graph. *Communications of the ACM*, **22**, 251-256. <https://doi.org/10.1145/359094.359101>
- [36] Chaitin, G.J., Auslander, M.A., Chandra, A.K., Cocke, J., Hopkins, M.E. and Markstein, P.W. (1981) Register Allocation via Coloring. *Computer Languages*, **6**, 47-57. [https://doi.org/10.1016/0096-0551\(81\)90048-5](https://doi.org/10.1016/0096-0551(81)90048-5)
- [37] Avanthay, C., Hertz, A. and Zufferey, N. (2003) A Variable Neighborhood Search for Graph Coloring. *European Journal of Operational Research*, **151**, 379-388. [https://doi.org/10.1016/S0377-2217\(02\)00832-9](https://doi.org/10.1016/S0377-2217(02)00832-9)
- [38] Ingber, L., Petraglia, A., Petraglia, M.R., Machado, M.A.S., *et al.* (2012) Adaptive Simulated Annealing. *Stochastic Global Optimization and Its Applications with Fuzzy Adaptive Simulated Annealing*. Springer, New York, 33-62.
- [39] Bertsimas, D., Tsitsiklis, J., *et al.* (1993) Simulated Annealing. *Statistical Science*, **8**, 10-15. <https://doi.org/10.1214/ss/1177011077>
- [40] Szu, H. and Hartley, R. (1987) Fast Simulated Annealing. *Physics Letters A*, **122**, 157-162. [https://doi.org/10.1016/0375-9601\(87\)90796-1](https://doi.org/10.1016/0375-9601(87)90796-1)
- [41] Baum, D.S., Hansen, J. and Salo, J. (2005) An Interim Channel Model for Beyond-3G Systems: Extending the 3GPP Spatial Channel Model (SCM). *Vehicular Technology Conference (spring)*, **5**, 3132-3136. <https://doi.org/10.1109/vetecs.2005.1543924>
- [42] Unnikrishnan, J. and Veeravalli, V.V. (2008) Cooperative Sensing for Primary Detection in Cognitive Radio. *IEEE Journal of Selected Topics in Signal Processing*, **2**, 18-27. <https://doi.org/10.1109/JSTSP.2007.914880>
- [43] Fertin, G., Godard, E. and Raspaud, A. (2003) Acyclic and K-Distance Coloring of the Grid. *Information Processing Letters*, **87**, 51-58. [https://doi.org/10.1016/S0020-0190\(03\)00232-1](https://doi.org/10.1016/S0020-0190(03)00232-1)
- [44] Kuo, W. and Zuo, M.J. (2003) *Optimal reliability Modeling: Principles and Applications*, John Wiley & Sons, Hoboken.

### Appendix A

Consider  $N$  independent Bernoulli trials, where for the  $m$ th trial  $X_m$  the probability of success is  $\text{Prob}\{X_m = 1\} = p_m$  and  $\text{Prob}\{X_m = 0\} = 1 - p_m$ . The probability of at least  $k$  successes in these  $N$  trials is the complement of Poisson-Binomial CDF ([24], Equation (11)),

$$Q(k) = 1 - \frac{1}{N+1} \sum_{n=0}^N \left\{ \frac{1 - \exp[-j2\pi nk/(N+1)]}{1 - \exp[-j2\pi n/(N+1)]} \prod_{m=1}^N (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\}. \quad (36)$$

In addition, the Poisson-Binomial probability mass function (PMF) is ([24], Equation (5)),

$$P(k) = \frac{1}{N+1} \sum_{n=0}^N \left\{ \exp[-j2\pi nk/(N+1)] \prod_{m=1}^N (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\}. \quad (37)$$

**Lemma 1.** For a certain series of  $N$  Bernoulli trials, and fixed value of  $k$ , the complement of Poisson-Binomial CDF,  $Q(k)$  of (36) is monotonic non-decreasing in each  $p_m, m = 1, 2, \dots, N$ .

*Proof:* We obtain the partial derivative of  $Q(k)$  in (36) with respect to  $p_l, l = 1, 2, \dots, N$ ,

$$\begin{aligned} \frac{\partial Q(k)}{\partial p_l} &= -\frac{1}{N+1} \sum_{n=0}^N \left\{ \frac{1 - \exp[-j2\pi nk/(N+1)]}{1 - \exp[-j2\pi n/(N+1)]} (\exp[j2\pi n/(N+1)] - 1) \right. \\ &\quad \times \left. \prod_{1 \leq m \leq N, m \neq l} (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\} \\ &= \frac{1}{N+1} \sum_{n=0}^N \left\{ (\exp[-j2\pi nk/(N+1)] - 1) \exp[j2\pi n/(N+1)] \right. \\ &\quad \times \left. \prod_{1 \leq m \leq N, m \neq l} (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\}. \end{aligned} \quad (38)$$

Then we consider a certain series of  $N$  Bernoulli trials where  $\text{Prob}\{X_l = 1\} = p_l = 1$ , and use  $P^{[p_l=1]}(k)$  to denote the Poisson-Binomial PMF in this case. Based on (37) we have

$$\begin{aligned} P^{[p_l=1]}(k) &= \frac{1}{N+1} \sum_{n=0}^N \left\{ \exp[-j2\pi nk/(N+1)] \exp[j2\pi n/(N+1)] \right. \\ &\quad \times \left. \prod_{1 \leq m \leq N, m \neq l} (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\}. \end{aligned} \quad (39)$$

When  $k = 0$ , from (39) we have

$$P^{[p_l=1]}(0) = \frac{1}{N+1} \sum_{n=0}^N \left\{ \exp[j2\pi n/(N+1)] \prod_{1 \leq m \leq N, m \neq l} (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\} \quad (40)$$

Based on (39) and (40), we get

$$\begin{aligned} P^{[p_l=1]}(k) - P^{[p_l=1]}(0) &= \frac{1}{N+1} \sum_{n=0}^N \left\{ (\exp[-j2\pi nk/(N+1)] - 1) \exp[j2\pi n/(N+1)] \right. \\ &\quad \times \left. \prod_{1 \leq m \leq N, m \neq l} (1 - p_m + p_m \exp[j2\pi n/(N+1)]) \right\}. \end{aligned} \quad (41)$$

The right-hand side of (41) is exactly the same as that of (38), and hence

$$\frac{\partial Q(k)}{\partial p_l} = P^{[p_l=1]}(k) - P^{[p_l=1]}(0). \tag{42}$$

When  $p_l = 1$ , the  $l$ th trial is successful with probability 1, and hence  $P^{[p_l=1]}(0) = 0$ . Thus,  $\frac{\partial Q(k)}{\partial p_l} = P^{[p_l=1]}(k) - P^{[p_l=1]}(0) = P^{[p_l=1]}(k) \geq 0, l = 1, 2, \dots, N$ .

## Appendix B

### Proof of Property 4

Suppose that there are  $n_i$  independent Bernoulli trials, and the success probability of each trial is  $\varepsilon_{j,i}$ . Then  $Q_{d,i}^{[L]}(k_i)$  in (13) is the probability that at least  $k_i$  trials are successful:

$$Q_{d,i}^{[L]}(k_i) = \text{Prob}\{\text{Number of successes} \geq k_i\}, \tag{43}$$

and  $Q_{d,i}^{[U]}(k_i)$  in (14) can be viewed as the probability that at least  $k_i$  of these  $n_i$  trials are failed,

$$\begin{aligned} Q_{d,i}^{[U]}(k_i) &= \text{Prob}\{\text{Number of failures} \geq k_i\} \\ &= \text{Prob}\{\text{Number of successes} \leq n_i - k_i\} \\ &= 1 - \text{Prob}\{\text{Number of successes} \geq n_i - k_i + 1\} \end{aligned} \tag{44}$$

where (44) is also given in ([44], pp. 251-253). Hence from (43) and (44)

$$Q_{d,i}^{[U]}(k_i) = 1 - Q_{d,i}^{[L]}(n_i - k_i + 1). \tag{45}$$

Then, based on (45) and the definition of  $\Delta_{d,i}(k_i)$  in (17), we have

$$\Delta_{d,i}(k_i) \triangleq Q_{d,i}^{[U]}(k_i) - Q_{d,i}^{[L]}(k_i) = 1 - Q_{d,i}^{[L]}(n_i - k_i + 1) - Q_{d,i}^{[L]}(k_i) \tag{46}$$

and

$$\begin{aligned} \Delta_{d,i}(n_i - k_i + 1) &\triangleq Q_{d,i}^{[U]}(n_i - k_i + 1) - Q_{d,i}^{[L]}(n_i - k_i + 1) \\ &= 1 - Q_{d,i}^{[L]}(n_i - (n_i - k_i + 1) + 1) - Q_{d,i}^{[L]}(n_i - k_i + 1) \\ &= 1 - Q_{d,i}^{[L]}(k_i) - Q_{d,i}^{[L]}(n_i - k_i + 1) = \Delta_{d,i}(k_i). \end{aligned} \tag{47}$$

## Appendix C

### Details of ASA

The ASA algorithm consists of three major stages: generating, acceptance, and annealing [38]. In a  $N$ -dimensional parameter space with the  $l$ th parameter having the range  $[L_l, U_l]$ , assuming that  $\mathbf{p}^{(t)}$  is the  $t$ th saved point, the generating procedure aims to obtain a new candidate point  $\hat{\mathbf{p}}^{(t+1)}$  for the  $(t+1)$ th state, whose components are  $\hat{p}_i^{(t+1)} = p_i^{(t)} + y_i(U_i - L_i)$ , where  $y_i \in [-1, 1]$  is a sample of a random variable  $Y_i$  with PDF

$$g_i(y_i; T_i(t_i)) = \frac{1}{2(|y_i| + T_i(t_i)) \ln(1 + 1/T_i(t_i))}, \tag{48}$$

$T_i(t_i)$  is the parameter temperature, and  $t_i$  is the parameter annealing

index. If  $\hat{p}_i^{(t+1)} \notin [L_i, U_i]$ , a new  $\hat{p}_i^{(t+1)}$  is generated with another sample  $y_i$  of  $Y_i$ , until  $\hat{p}_i^{(t+1)} \in [L_i, U_i]$ .

The acceptance procedure determines whether the candidate point  $\hat{\mathbf{p}}^{(t+1)}$  is accepted as the new saved point for the next state. When the cost function is denoted by  $\zeta(\cdot)$ , and  $u \in [0, 1]$  is a sample of a uniformly distributed random variable, the acceptance procedure is

$$\mathbf{p}^{(t+1)} = \begin{cases} \hat{\mathbf{p}}^{(t+1)} & \text{if } \exp\left[-\left(\zeta(\hat{\mathbf{p}}^{(t+1)}) - \zeta(\mathbf{p}^{(t+1)})\right) / T_{\text{cost}}(t_{\text{cost}})\right] > u \\ \mathbf{p}^{(t)} & \text{if } \exp\left[-\left(\zeta(\hat{\mathbf{p}}^{(t+1)}) - \zeta(\mathbf{p}^{(t+1)})\right) / T_{\text{cost}}(t_{\text{cost}})\right] \leq u \end{cases} \quad (49)$$

where  $T_{\text{cost}}(t_{\text{cost}})$  is the cost temperature, and  $t_{\text{cost}}$  is called cost annealing index.

The annealing procedure for the parameter temperature  $T_i(t_i)$  is

$$T_i(t_i) = T_i(0) \exp(-c_{\text{para}} t_i^{Q_i/N}). \quad (50)$$

The initial parameter temperature  $T_i(0)$  is usually set to 1 by users. The parameter annealing index  $t_i$  is initially set to 0, and increased by 1 every time a new candidate point is generated. The parameter quenching factor for the  $i$ th parameter is  $Q_i$ , which is set to 1 for normal ASA. When  $Q_i > 1$ , the ‘‘quenching’’ option is turned on to speed up the search in a large parameter space [38]. The parameter temperature scale is

$$c_{\text{para}} = -\ln(\alpha_1) \exp(-\ln(\alpha_2)/N),$$

where  $\alpha_1$  and  $\alpha_2$  are called the temperature ratio scale and temperature anneal scale respectively.

The annealing procedure for the cost temperature  $T_{\text{cost}}(t_{\text{cost}})$  is

$$T_{\text{cost}}(t_{\text{cost}}) = T_{\text{cost}}(0) \exp(-c_{\text{cost}} t_{\text{cost}}^{Q_{\text{cost}}/N}). \quad (51)$$

The ASA algorithm first generates  $N_{\text{spl}}$  states and calculates the cost function of each state, where  $N_{\text{spl}}$  is named number of cost samples. Then, the initial cost temperature  $T_{\text{cost}}(0)$  takes the average of the absolute values of the sampled cost functions. The cost annealing index  $t_{\text{cost}}$  is initially set to 0 and then increased by 1 every time a new candidate point is accepted. The cost quenching factor  $Q_{\text{cost}}$  is similar to  $Q_i$ . The cost temperature scale  $c_{\text{cost}}$  is set as  $c_{\text{cost}} = \beta c_{\text{para}}$ , where  $\beta$  is an ASA parameter named cost parameter scale ratio.

An optional reannealing procedure will periodically rescale  $T_i(t_i)$  and  $T_{\text{cost}}(t_{\text{cost}})$ . This procedure takes place every  $m_{\text{gen}}$  (generated frequency modulus) candidate points generated, or every  $m_{\text{acc}}$  (acceptance frequency modulus) candidate points accepted. Assuming that this reannealing procedure is conducted after  $t_i$  states are generated,  $T_i(t_i)$  is reannealed as

$$T_i(t_i) \leftarrow \left( \max_{1 \leq j \leq N} \{f_j\} / f_i \right) T_i(t_i), \text{ with } f_i = \left| \frac{\partial \zeta(\mathbf{p})}{\partial p_i} \right|_{\mathbf{p}^{(t)}} \quad (52)$$



where  $p^{(t)}$  is the best cost value found as of the current state. If  $T_i(t_i) < T_i(0)$  then  $t_i$  is reset as  $t_i \leftarrow \left( \frac{|\ln T_i(0) - \ln T_i(t_i)|}{c_{\text{para}}} \right)^{N/Q_i}$ . Moreover,  $T_{\text{cost}}(0)$  is re-scaled as

$$T_{\text{cost}}(0) \leftarrow \min \left\{ T_{\text{cost}}(0), \max \left\{ \left| \zeta(p^{(t-1)}) \right|, \left| \zeta(p^{(t)}) \right|, \left| \zeta(p^{(t-1)}) - \zeta(p^{(t)}) \right| \right\} \right\}. \quad (53)$$

When  $T_{\text{cost}}(0)$  gets its new value, the current cost temperature is reset as

$$T_{\text{cost}}(t_{\text{cost}}) \leftarrow \min \left\{ T_{\text{cost}}(0), \max \left\{ T_{\text{cost}}(t_{\text{cost}}), \left| \zeta(p^{(t-1)}) - \zeta(p^{(t)}) \right| \right\} \right\}. \quad (54)$$

Thus, according to (51),  $t_{\text{cost}}$  after reannealing is

$$t_{\text{cost}} \leftarrow \left( \frac{|\ln T_{\text{cost}}(0) - \ln T_{\text{cost}}(t_{\text{cost}})|}{c_{\text{cost}}} \right)^{N/Q_{\text{cost}}}$$



**Submit or recommend next manuscript to SCIRP and we will provide best service for you:**

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact [wsn@scirp.org](mailto:wsn@scirp.org)