Mobile Netw Appl
DOI 10.1007/s11036-012-0387-x

A New Cooperative Spectrum Sensing Scheme for Cognitive Ad-Hoc Networks

Yang Du · Hongxiang Li · Weiyao Lin · Lingjia Liu · Xudong Wang · Samee U. Khan · Sentang Wu

Abstract As the radio spectrum is becoming more and more crowded, cognitive radio has recently become a hot research topic to improve the spectrum utilization efficiency. It is well known that the success of cognitive radio depends heavily on fast and efficient spectrum sensing that is very difficult in practice. Toward this end, this paper introduces a new guard-resident cooperative spectrum sensing scheme for a cognitive ad-hoc network. In particular, we classify cognitive nodes as either *resident* or *guard* based on the spectrum neighbor decision and distributed boundary search. The guard nodes sense the spectrum and then inform the resident nodes that are greatly relieved from spectrum sensing about the radio environmental changes. The analysis and simulation results show that the proposed algorithm can significantly reduce the total spectrum sensing load without sacrificing the sensing accuracy.

Keywords cognitive radio · spectrum sensing · ad-hoc · multi-cell · distributed boundary search

1 Introduction

In order to improve the spectrum utilization, cognitive radio (CR) has recently gained significant attention from the wireless community [1]. In CR, within a tiered access hierarchy, the primary users retain preferential use rights; the secondary users may only use a primary channel when it is identified as unoccupied and must release such a channel whenever a primary user's transmission is detected. As is well known, the success of CR operation depends heavily on fast and efficient spectrum sensing [2]. This seemingly innocuous task can actually be quite difficult in practice due to the large variations in the dynamic range and bandwidth of signals to be detected. For example, in a large scale cognitive wireless sensor network, sensors are limited in size and complexity and the demanding spectrum sensing can quickly deplete the energy.

To achieve better performance, people proposed the cooperative spectrum sensing (CSS) concept where each single node collects individual sensing results from its neighbors and combines them to make a better decision [3]. The existing cooperative spectrum sensing research is mostly focused on how to combine sensing information collected by cooperative CR users and the optimizing sensing parameters [4, 5]. Reference [6] modeled the CSS problem as a nontransferable

Y. Du (✉) · S. Wu
School of Automation Science and Electrical Engineering,
Beihang University,
Beijing, China
e-mail: doyoung0.618@gmail.com

H. Li
Department of Electrical and Computer Engineering,
University of Louisville,
Louisville, KY, USA

W. Lin
Institute of Image Communication and Information Processing,
Shanghai Jiaotong University,
Shanghai, China

L. Liu
Department of Electrical Engineering and Computer Science,
University of Kansas,
Lawrence, KS, USA

X. Wang
UM-SJTU Joint Institute, Shanghai Jiao Tong University,
Shanghai, China

S. U. Khan
Department of Electrical and Computer Engineering,
North Dakota State University,
Fargo, ND, USA

Published online: 30 June 2012

coalitional game where the network of CR users could form cooperating coalitions and interact on whether to merge or split based on the comparison relation for improving their spectrum sensing performance. Reference [7] modeled the CSS problem as an evolutionary game where the payoff was defined as the throughput of a secondary user. Reference [8] proposed a fast spectrum sensing algorithm for a large network which required fewer than the total number of CRs in CSS while satisfying a given error bound. However, all existing CSS approaches put additional burden on neighboring nodes for constant spectrum sensing. Another major drawback of the existing CSS solutions is that most of them assume the cooperative nodes are subject to the same frequency exposure, few work consider the multi-cell primary network scenario where the neighboring CR nodes have exposure to different frequencies, leaving some open issues such as the well known *hidden node problem* [9] still unsolved.

In this paper, we consider a CR ad-hoc network (CRAHN), where the secondary network has ad-hoc connectivity (such as distributed multi-hop communication, self-organizing and dynamic network topology [10]). In CRAHN, the CR nodes generally have limited computation capability and thus constant spectrum sensing is not a suitable solution. *The key contribution of this paper is we derive a new guard-resident CSS method that can significantly reduce the overall spectrum sensing load without sacrificing the overall performance.*

The rest of the paper is organized as follows: Section 2 provides system model and the assumptions, followed by detailed discussion of the guard-resident CSS algorithms in Section 3. In Section 4, simulation results are presented. Finally, a conclusion is drawn in Section 5.

2 System model

In this section, we describe the system model and assumptions. Compared to other existing CSS models, our model has two distinct features: (1) the primary network has multiple frequency zones; (2) cognitive nodes have ad-hoc connectivity so that cooperation is not limited to geographic neighbors.

Consider a multi-cell TV broadcasting (or cellular downlink) primary network as shown in Fig. 1, where we assume no frequency reuse for adjacent cells to avoid inter-cell interference. We define a *frequency zone* as an area covered by the same primary transmission. Ideally, CR nodes within the same frequency zone should have the same spectrum sensing results. Figure 1 shows a three-cell primary network with seven frequency zones. The CR nodes with different densities are randomly distributed over the whole area. For any particular CR node, we define its *geometric neighbors as those* who have direct (one hop) connection with it. Note that a node's geometric neighbors may be located at different frequency zones, which is particularly true for those who are on cell edge.

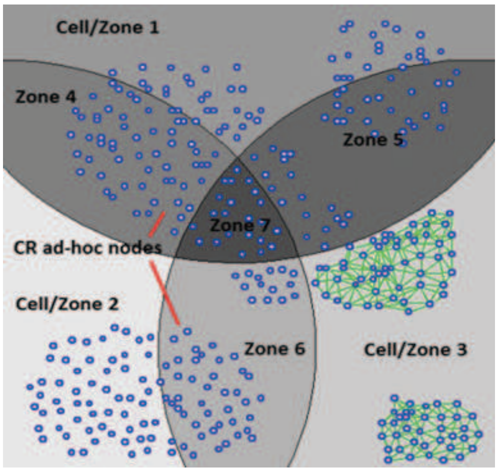


Fig. 1 Multi-cell TV broadcasting primary network

As we mentioned earlier, the benefits (increased sensing accuracy) of the existing CSS methods come at a price (increased sensing load). Furthermore, these methods are problematic for any cell edge CR user whose neighbors are from different frequency zones. On the other hand, we realize that cooperation between any two CR users is possible if they are connected (via single hop or multi hops) within the same frequency zone. Toward this end, we propose our new guard-resident CSS scheme. The basic idea is to classify each CR node as either *resident* or *guard*, where only the guard nodes constantly sense the spectrum and inform the resident nodes about the environmental changes. As shown by Fig. 2, the polygon formed by the guards becomes a safe zone such that any CR node within the safe

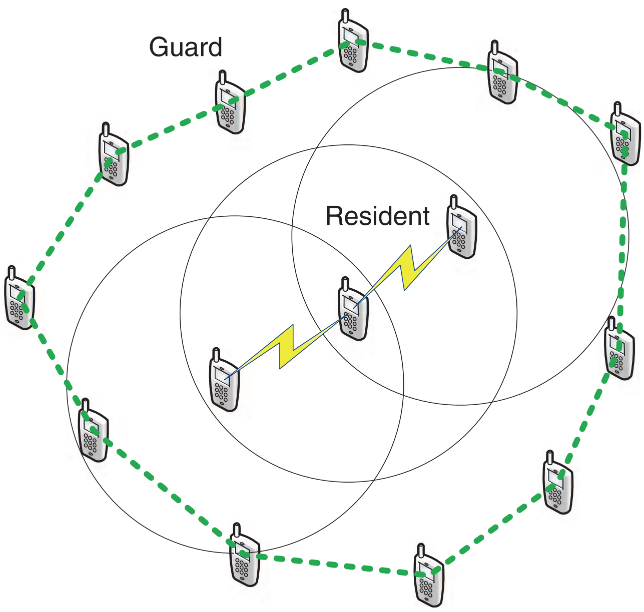


Fig. 2 The guard-resident scheme

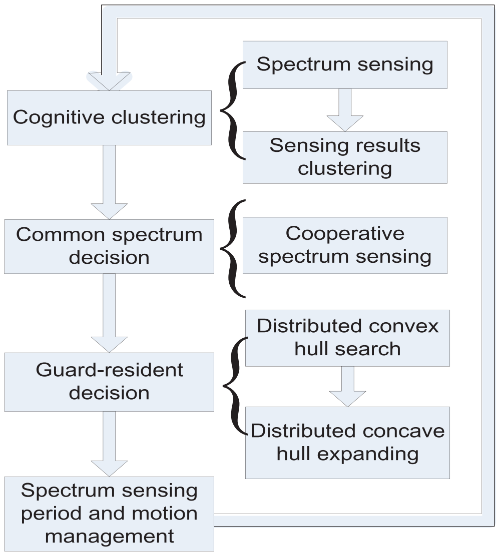


Fig. 3 Guard-resident scheme frame

zone will be greatly relieved from constant spectrum sensing.

In this work, we make the following assumptions: (1) Each CR node has no knowledge about the primary network, but it knows the direction of its geometric neighbor(s), which can be obtained by the positioning devices such as GPS or calculated from some "directional finding" algorithms [11]; (2) The CRAHN has a common control channel (CCC) that is dedicated to coordination and control information exchange among CR nodes [12].

3 Guard-resident scheme

The guard-resident cooperative spectrum sensing (GRCSS) scheme can be illustrated by the flow chart in Fig. 3. In this section, we'll explain it step by step.

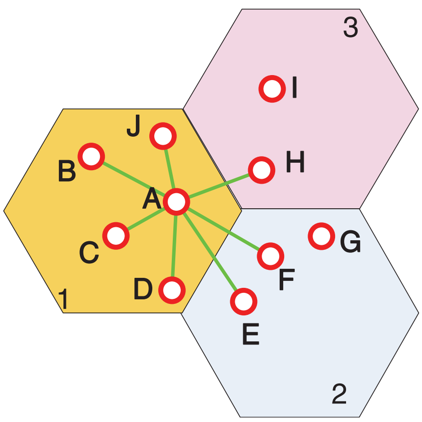


Fig. 4 Geometric neighbors and spectrum neighbors

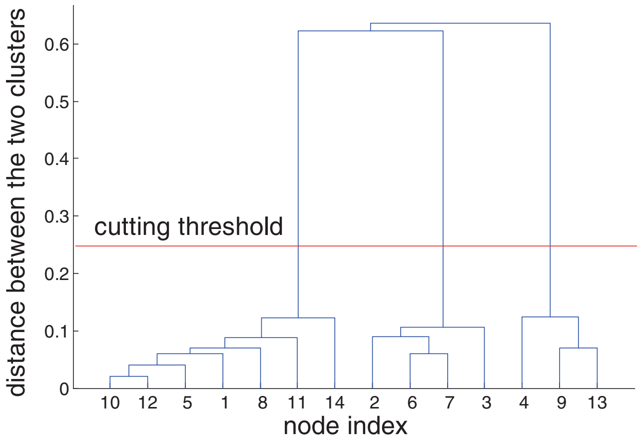


Fig. 5 The dendrogram of a hierarchical tree

3.1 Cognitive clustering

The goal of this step is to divide CR nodes into *clusters* such that nodes within the same cluster are fully connected and located in the same frequency zone. For example, in Fig. 1, there are two spectrum clusters in zone 3. As shown in Section 3.2, spectrum cluster is the basic unit to make guard-resident decision, i.e., each spectrum cluster will form a connected guard boundary to "protect" the inside residents.

Initially, all CR nodes sense the spectrum and exchange the sensing results with their geometric neighbors over CCC. According to the collected sensing results, each node recognizes its *spectrum neighbors* from *geometric neighbors* by cognitive clustering. As shown in Fig. 4, node *A* has seven geometric neighbors. Among them, node *B*, *C*, *D* and *J* are also spectrum neighbors of *A*.

Due to the noise and other imperfections, nodes in the same frequency zone may have different sensing results. Then the question is how to decide a node's spectrum

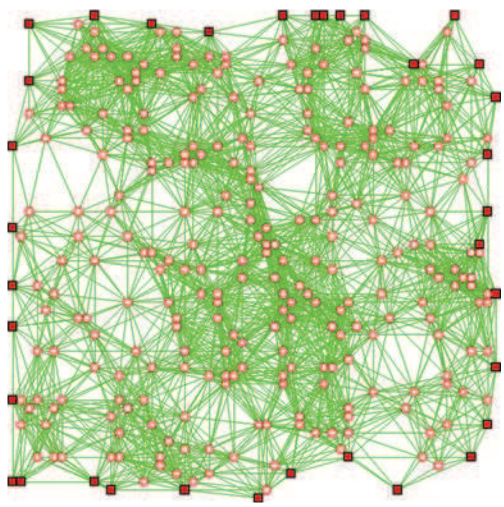


Fig. 6 A simulation of Guard-Resident Decision

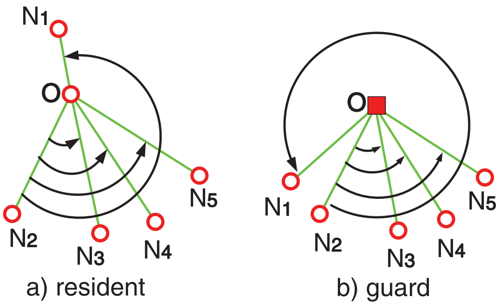


Fig. 7 Guard and resident illustration

neighbors with sensing errors. In this paper we introduce *cluster analysis* to partition the CR nodes into a certain number of clusters so that the sensing results in the same cluster are similar while those from different clusters are quite different. We aim to maximize both the cluster internal homogeneity and the external separation. Among many clustering algorithms, we choose hierarchical clustering algorithm (HCA) [13] because it doesn't need the prediction of the number of clusters and yields good performance in our cognitive clustering process compared with other clustering algorithms such as K-means and Fuzzy C-means [14].

There are two design parameters when applying HCA to our cognitive clustering: one is the *distance* among CR nodes and the other is the *threshold* for cutting the hierarchical tree. There are no fixed criteria for choosing the distance and the threshold because they depend on the specific application. For example, in Fig. 5, we have a hierarchical tree with three clusters (10, 12, 5, 1, 8, 11, 14), (2, 6, 7, 3) and (4, 9, 13) using the threshold of 0.25.

Note that the specific spectrum sensing technique [15, 16] is not the focus of this paper. For the convenience of the discussion, we use energy detection based spectrum sensing (EDSS) [17] technology to illustrate how to define the distance and threshold in HCA. The EDSS approach, which is especially suitable for wide-band spectrum sensing in practice [18], has the following two hypotheses:

$$\begin{cases} H_0 : Y = N \\ H_1 : Y = S + N \end{cases} \qquad (1)$$

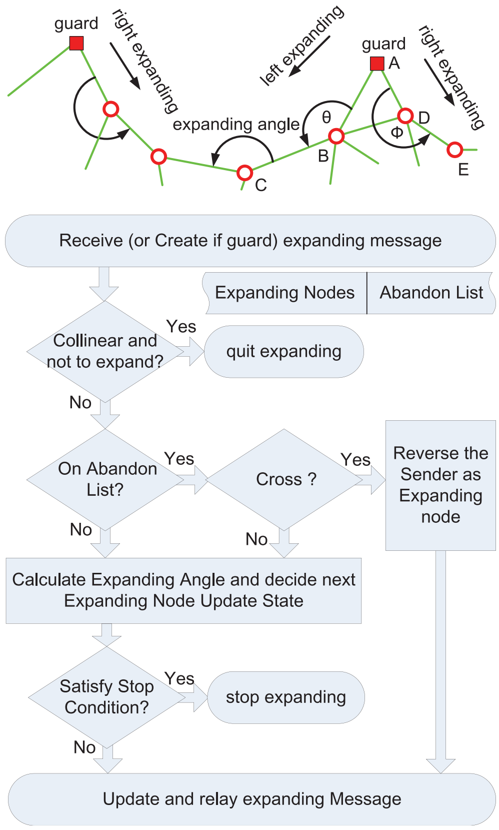


Fig. 9 Guard expanding procedure

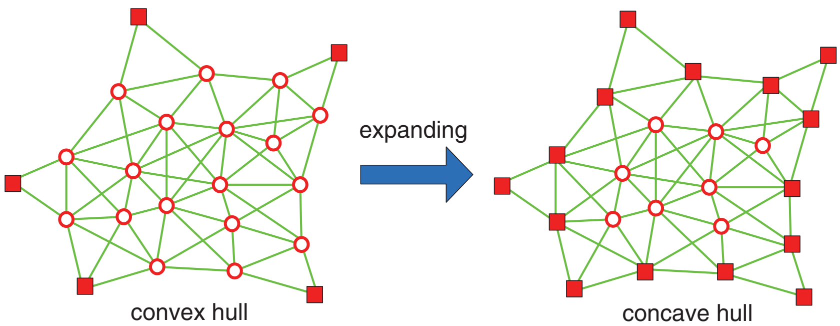
where Y is the overall sensed signal on a particular frequency channel; S is the primary signal to be detected; N is the additive white Gaussian noise (AWGN).

Assuming a node has *n-1* geometric neighbors and m channels to sense, we use the following n-by-m matrix X $\{x_{ij}\}_{n\times m}$ to denote the sensing results:

$$X = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 0 & 0 & 1 & 0 & \cdots & 0 \end{pmatrix}_{n\times m} \qquad (2)$$

where x_{ij} =1 or 0 means the channel j is sensed by node i as available or occupied. In X, each row vector represents the sensing results of a particular node. For any two nodes r and s, we use normalized Hamming distance (NHD) as the

Fig. 8 Distributed boundary search algorithm



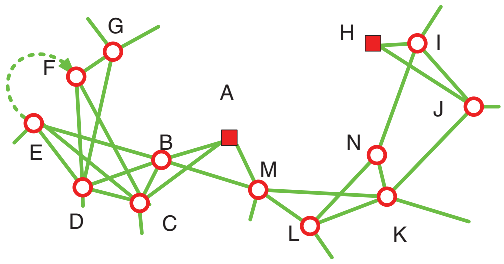


Fig. 10 The cross phenomenon

distance metric because it directly reflects the difference of sensing results:

$$d_{rs} = \frac{1}{m}\sum_{j=1}^{m} x_{rj} \oplus x_{sj} \tag{3}$$

Note that the symbol "⊕" is the mod operation, which can give erroneous result because each node may have detection errors. For node *i*, we denote the detection error rate for a particular channel *j* as $P_E(i,j)$. In order to maximize both the cluster internal homogeneity and the external separation, the *threshold* for cutting the hierarchical tree can't be either too large or too small. We denote the *threshold* as λ_{cut} and it should satisfy $\lambda_{min}<\lambda_{cut}<\lambda_{max}$.We have derived both λ_{min} and λ_{max} (derivation is omitted due to space limit):

$$\lambda_{\min} = 2E\{P_E\}(1 - E\{P_E\}) \tag{4}$$

$$\lambda_{\max} = (1 - 2E\{P_E\}(1 - E\{P_E\}))D_z + 2E\{P_E\} \times (1 - E\{P_E\})(1 - D_z) \tag{5}$$

where $E\{\cdot\}$ is expectation and D_z is the average frequency diversity rate of two adjacent frequency zones. The optimal threshold is given by $\lambda_{cut}=0.5\times(\lambda_{min}+\lambda_{max})$. Another question is whether or not we can always find a solution for λ_{cut}. Obviously, λ_{cut} always has a solution if $\lambda_{max}-\lambda_{min}\geq 0$. Plugging above results, we have

$$\lambda_{\max} - \lambda_{\min} = D_z(2E\{P_E\} - 1)^2 \geq 0 \tag{6}$$

Therefore, λ_{cut} always exists.

Once the cognitive clustering is finished, the traditional CSS will be implemented within each spectrum cluster over CCC, including the common spectrum decision shown in Fig. 3.

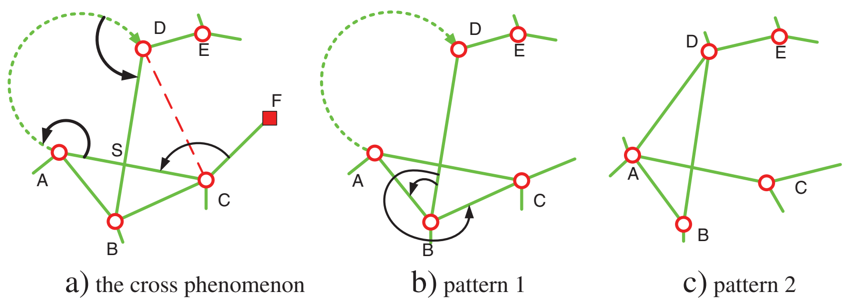
3.2 Guard-resident decision

The most important step in GRCSS is to make guard-resident decision for each spectrum cluster. Intuitively, the boundary nodes of each spectrum cluster can serve as the guards and "protect" the inside residents because the enclosed polygon has the advantage of detecting primary transmission from any direction. For example, Fig. 6 shows a spectrum cluster where the square and round nodes are marked as guard and residents respectively. It is a concave hull of the CR nodes. However, the challenge is how to determine the boundary nodes considering the ad-hoc nature of the network. A major contribution of this paper is that we derive an efficient distributed algorithm to find the connected boundary of any arbitrary spectrum cluster.

Note that in CRAHN, each node can only decide its status (guard or resident) according to the limited local information and the nodes on the boundary are supposed to sense the spectrum in our GRCSS, so the algorithm for concave hull search should be deterministic, distributed and low-cost. References [19–21] proposed some distributed boundary search methods without position information but they all assumed very dense node connectivity, which only has limited applications. The distributed boundary search algorithm we present in this paper assumes each node only has its neighbor's direction information, which is represented by the counter-clockwise angle from one edge to another (see Fig. 7).

The guard-resident decision contains two steps, the first is distributed convex hull search aimed to find a rough convex boundary and the second is distributed concave hull expanding aimed to expand some nodes as the boundary nodes for the final concave hull (Fig. 8).

Fig. 11 The decomposition of cross phenomenon



3.2.1 Distributed convex hull search

As shown in Fig. 7, the spectrum neighbor of node O is denoted by N_i, $i=1, 2, 3\ldots$ Select an arbitrary edge ON_j, the counter-clockwise angle from ON_j to ON_i is $\{\theta_i \mid i=1, 2, \ldots\}$, define:

$$\Delta\theta = \min\{\theta_i \cup 2\pi | \pi < \theta_i \leq 2\pi\} - \max\{\theta_i \cup 0 | 0 \leq \theta_i \leq \pi\} \quad (7)$$

Node O is called *guard* (boundary node) if $\Delta\theta > \pi$ (Fig. 7b). Otherwise, it is *resident* (interior node, Fig. 7a). For guard node, the spectrum neighbors whose indices achieve the "min" and "max" value in Eq. 7 are called the *left* and *right* spectrum neighbors respectively. For example, for the guard node O in Fig. 7b, N_1 and N_5 are its left and right spectrum neighbors.

3.2.2 Distributed concave hull expanding

To better protect the residents and facilitate information exchange, we need to expand the rough convex guard boundary obtained from Section 3.2.1 to make it fully connected. As shown in Fig. 9, guard node A first expands to both its left spectrum neighbor B and right spectrum neighbor D so that node B and D change their status from resident to guard. Then node B and D further expands the guard boundary to C and E, where node C is called the *left expanding node* of B and E is the *right expanding node* of D. The angel θ and ϕ are called *expanding angles* of the node B and D. The same procedure will continue till a stop condition is met.

During guard expanding procedure, how to choose the expanding nodes and stop condition is the key. There are three phenomena (along with their derivatives) that can cause guard expanding to stop unexpectedly. For the ease of discussion, we only use left expanding for illustration.

The cross phenomenon The cross phenomenon happens when the latter expanding route intersects the previous one. Figure 10 shows a guard expanding example with multiple route intersections, which can cause the unstoppable expanding or improper stop.

The general cross phenomenon can be represented by Fig. 11a. According to the minimum counter-clockwise angle, the left expanding route of node F will be *F-C-A—D-B* where edge CA intersects DB at point S. We assume that all nodes have the same communication radius R. Node C and D have no direct connection (otherwise, C would expand to D). There can be single or multiple hop connection between node A and D. In the following, we show that Fig. 11a can be further decomposed into two patterns (Fig. 11b and c).

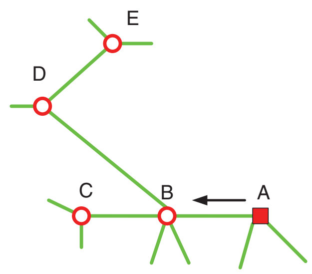


Fig. 12 The straight line phenomenon

$$\because \quad CS + DS > CD > R \text{ and } AC + BD <= 2R$$
$$\therefore \quad AS + BS < R$$
$$\because \quad AS + BS > AB$$
$$\therefore \quad AB < R$$

So we can conclude: (1) A and B must have one single hop connection; (2) If B and C don't have single hop connection, A and D must have single hop connection. Therefore, we get pattern 1 and pattern 2 as shown in Fig. 11b and c.

After the initial distributed convex hull search, each guard node sends different expanding messages to its left and right spectrum neighbors. The message includes additional information so that the spectrum neighbor who receives the message will execute the expanding procedure in Fig. 9. In particular, if this spectrum neighbor is abandoned, it will run the following cross detection algorithm to detect if there is a cross phenomenon.

Cross Detection Algorithm (left expanding)
1: $\theta_L \leftarrow$ find the minimum counter-clockwise angle from DB to BA_i
2: $\theta_R \leftarrow$ find the maximum counter-clockwise angle from DB to BA_i
3: if $\theta_R - \theta_L > \pi$
4: matches pattern 1
5: else if A_i is a single hop neighbor of D and $\theta_L < \pi$
6: matches pattern 2
7: else
8: no match
9: end if

The straight line phenomenon When it comes to the straight line issue in Fig. 12, node B and C are both in the radius of A. Node A, B, C are collinear, if the expanding is from A to C, B is missing to expand to D and the expanding will go wrong. To solve this problem, node A should send expanding message to both B and C because they both satisfy the minimum counter-clock angle criterion, i.e., they are both the left spectrum neighbors. Then node B and C themselves decide whether to be the expanding node. Here, node B will realize that C and A are on the two sides and C will realize that B and A are on the same side according to the direction information. So B decides to become the expanding node and C quits.

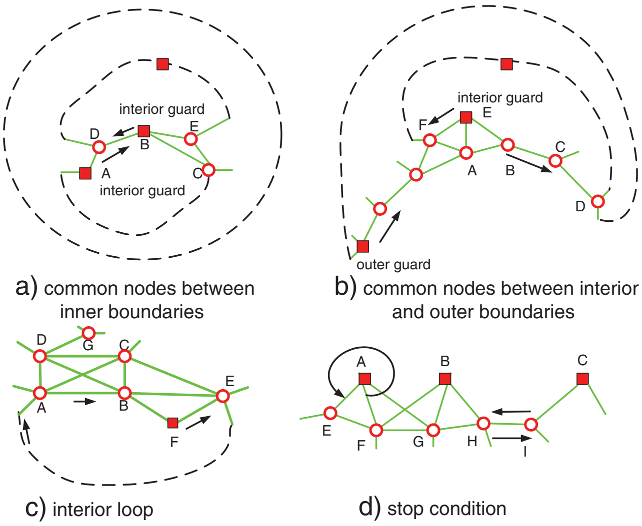


Fig. 13 The common nodes phenomenon and stopping condition

The interior guard phenomenon Because each node only knows its local information and isn't aware of the overall situation, it can cause the interior guard phenomenon and form the interior boundary (Fig. 13a and b). And the interior boundary becomes an interior loop if there is only one interior guard (Fig. 13c).

This phenomenon should be considered when decide stop conditions of the expanding. As shown in Fig. 13. In this case, the interior guards expanding form the inner boundary and the outer guards expanding form the outer boundary and they may have some nodes in common. So the expanding process may stop incorrectly when they expand to these common nodes if only according to the status (guard or resident) of the node. To solve this problem, we introduce the expanding

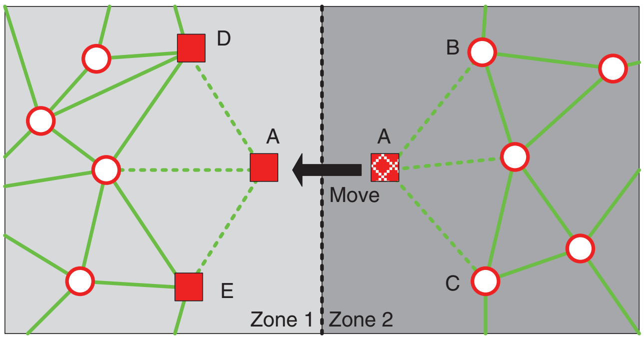


Fig. 15 The robustness for node mobility

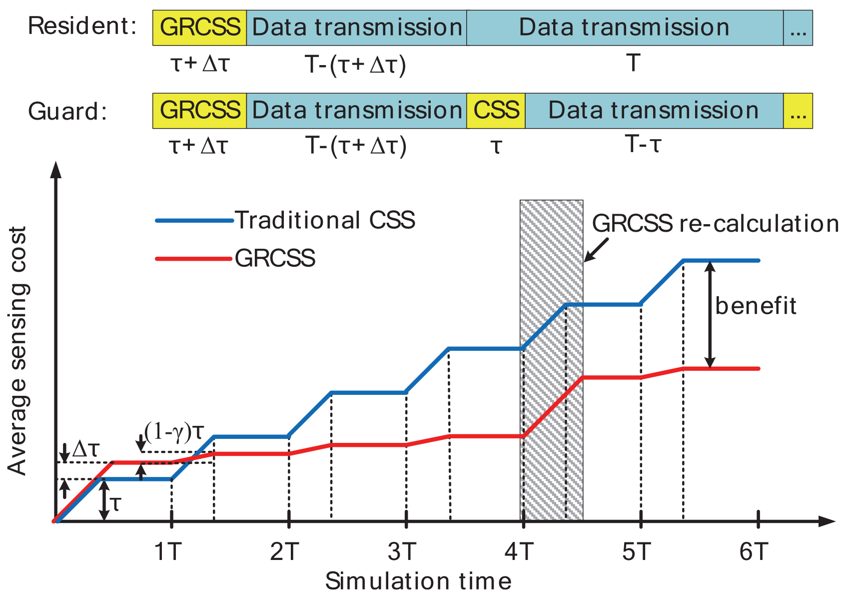
angle. The expanding stops if the node is guard and its expanding angle $\theta > \pi$ (stop condition (1)).

Considering the expanding goes to both left and right sides, we have stopping condition (2): The expanding stops if left and right expanding meet and have adverse directions. In Fig. 13d, node *H* runs right expanding to *I* and meanwhile *I* runs left expanding to *H*, so if *I* receives the right expanding message from *H* and the next left expanding node of *I* is just *H* then *I* stops to send expanding message and the two expanding processes stop.

3.3 Cost analysis and motion management

In order to timely detect any primary transmission, similar to traditional CR nodes, the guard nodes' spectrum sensing frequency (denoted as f_G) is usually high. On the other hand, when guards detect spectrum environment change and inform the residents, our GRCSS scheme also requires residents to sense the spectrum and re-calculate their status. For the convenience of discussion, we denote f_R as the average resident spectrum

Fig. 14 Illustration of the average sensing cost of GRCSS (T: time frame duration; τ: traditional CSS duration; Δτ: extra time duration of GRCSS at the beginning of the algorithms.)



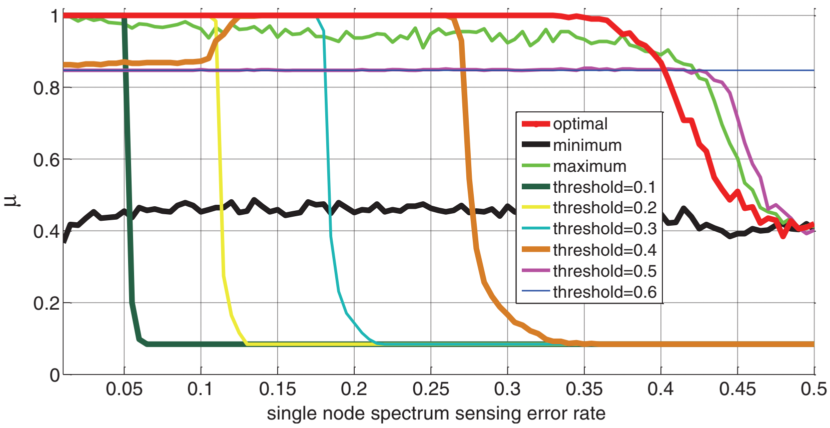


Fig. 16 μ of various clustering thresholds versus single node spectrum sensing error rate

sensing frequency. Because the spectrum environment usually changes slowly compared to f_G, it is reasonable to assume $f_G >> f_R$ for most practical applications. We also denote τ and $\tau + \Delta\tau$ as the time cost (algorithm running time) of the traditional CSS and our GRCSS scheme respectively such that $\Delta\tau$ is the extra cost of GRCSS per execution; L as the circumference of the graph; S as the area of the graph; R as the average distance between nodes; ρ as the nodes number per unit area. The free rate γ (the ratio of resident) can be estimated by

$$\gamma = 1 - L/S\rho R \tag{8}$$

Considering one single node on average, the sensing cost of GRCSS is:

$$f_G(1-\gamma)\tau + f_R(\tau + \Delta\tau) \tag{9}$$

And the traditional sensing cost is:

$$f_G\tau \tag{10}$$

Take EDSS for instance, in order to meet the probability of detection constraint [22], the number of samples required by Fast Fourier Transform (FFT) is on the order of O (1/SNR^2). The computation complexity of FFT is O (N_1logN_1) where N_1 is FFT size; the computation complexity of spectrum results clustering is O (N_2^2) where N_2 is the number of one node's geometric neighbors; the computation complexity of Guard-Resident Decision is O (N_3) where N_3 is the number of one node's spectrum neighbors. Usually N_1 is much larger than N_2 and N_3 so that $\Delta\tau$ is relatively small comparing to τ in the case of low f_R. Figure 14 illustrates the average sensing cost (in time) of the traditional CSS and our GRCSS scheme where T is the guard (or common CR node) spectrum sensing period. Obviously, the spectrum sensing cost can be significantly reduced (more benefits of GRCSS) in slow or moderate spectrum change environment.

Additionally, the cost of information exchange is not negligible in GRCSS. In particular, (1) when common control channel (CCC) is used for common CR nodes at the beginning of GRCSS, the cost is the time of exchanging the sensing results with geometric neighbors; (2) When CCC is used for guards cooperation, the cost is the time of exchanging the sensing results with other guards; (3) When CCC is used for guards to inform the residents of spectrum change, the cost is the time of broadcasting the notification to all residents. Note that the costs in (1) and (2) are also necessary in traditional CSS (where all CR nodes are considered as guards). Compared to the traditional CSS, our GRCSS scheme have some additional information exchange cost incurred in (3). Because our GRCSS focuses on applications with slow or moderate spectrum environment change, the guards inform residents less frequently so that the cost in (3) is small. For efficient guard protection, we do require that the distance (in terms of number of hopes) from any resident to guard(s) is reasonable to prevent long notification delay, which can be done by limiting the size of the spectrum cluster or adding some internal guards.

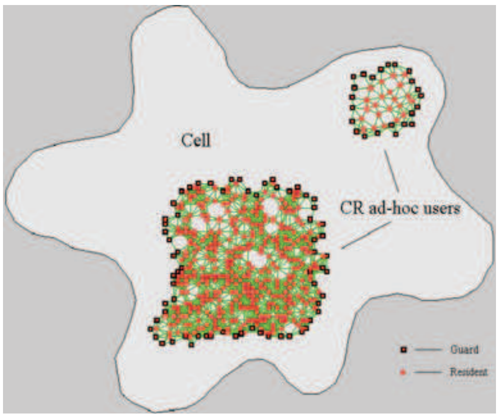


Fig. 17 Single cell scenario

Table 1 The free rate of single cell scenario

Color	red(small group)	red(large group)	total
Guard	23	66	89
Resident	22	389	411
Free rate γ	49 %	85 %	82 %

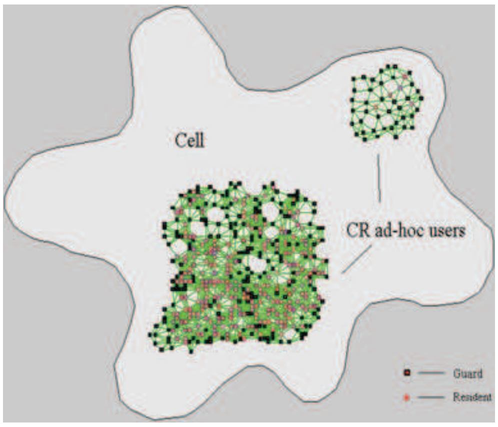


Fig. 18 Clustering error caused by detection error

Our algorithm is also robust to node mobility. As shown in Fig. 15, node *A* was a guard in zone 2 and moved to zone 1. It first informs its spectrum neighbors *B* and *C* before moving such that new boundary is formed in zone 2. When arriving at its new location in zone 1, node *A* will sense the spectrum and find the new spectrum neighbors such that a new boundary (*D-A-E*) is formed. Apparently, this automatic boundary recovery only involves a small number of local nodes and the overall network is not much affected.

4 Simulation

To evaluate the performance of the proposed GRCSS scheme, we consider a rectangular service area with dimensions 1000 m×1000 m. There are totally 100 frequency channels. The communication radius of the node is 30 m.

4.1 Accuracy of cognitive clustering

In the cognitive clustering process, each individual node executes HCA using the matrix *X* defined in Section 3.1.

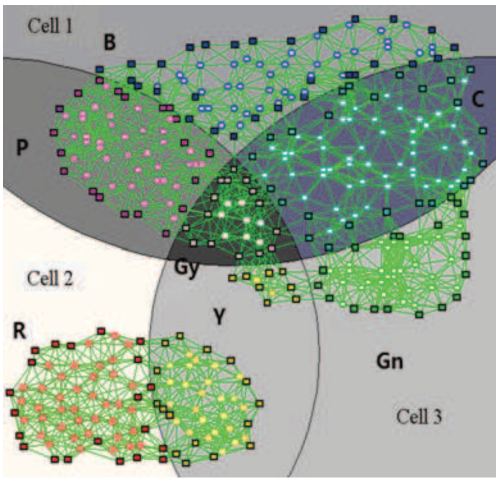


Fig. 19 Multi-cell scenario

Table 2 The free rate of multi-cell scenario

Color	Purple (P)	Blue (B)	Cyan (C)	Gray (Gy)	Yellow (Y)	Green (Gn)	Red (R)	total
Guard	20	29	25	15	24	19	21	153
Resident	37	39	54	8	27	31	39	235
Free rate γ	65 %	57 %	68 %	35 %	53 %	62 %	65 %	61 %

Consider $C=\{C_i \mid i=1, 2, \ldots, c\}$ are the clustered groups, where set C_i contains all the grouped nodes in group *i*; $P=\{P_j \mid j=1, 2, \ldots, v\}$ is the real partition where only the connected nodes within the same frequency zone fall in the same set P_j. We denote the number of nodes in a set by *number*(·) and the number of nodes that are clustered correctly in P_j by

$$N_j = \max_{i=1 \text{to} c}\{number_i(P_j \cap C_i)\} \tag{11}$$

where $max\{\cdot\}$ denotes the first maximum value.

Then the *accuracy of cognitive clustering* can be expressed as $\mu = E\{(1/n)\sum_{j=1}^{v} N_j\}$ where $E\{\cdot\}$ means the expectation to all the nodes.

As we can see in Fig. 16, by using the *optimal threshold* derived in Section 3.1, μ is close to *1* when detection error rate is less than 0.33 (the NHD between any two cells). It is nearly a straight line when the threshold is beyond 0.6, because all the nodes tend to be classified as one spectrum cluster with such high threshold and the value of μ depends on the distribution of the nodes. Meanwhile if the threshold is very low or the detection error rate is very high, each node will be assigned as a single spectrum cluster and the value of μ again depends on the distribution of the nodes.

4.2 Single cell scenario

As shown in Fig. 17, we scatter 500 nodes in a given area. The average single node spectrum sensing error rate is set as high as 10 %. The optimal threshold in Section 3.1 is used for cognitive clustering After running our GRCSS scheme, overall 82% of the nodes become residents (hollow round node) (see Table 1), which means the majority of the nodes are relieved from constant spectrum sensing.

When the average single node spectrum sensing error rate goes up to 40 % (some worst case scenario), we show the results in Fig. 18, where many nodes make incorrect clustering decisions.

4.3 Multi-cell scenario

As shown in Fig. 18, the 100 frequency channels are evenly allocated to three cells with overlap but no frequency reuse. The whole area forms seven different spectrum zones and the NHD between every two adjacent zones is at least 0.33; the average single node spectrum sensing error rate is 10 %. The result is shown in Fig. 19 (Different colors represent different spectrum sensing results) and Table 2. By GRCSS, the whole CRAHN recognizes the primary frequency environment and forms different spectrum clusters. The guard nodes in each spectrum cluster monitor the environment changing by forming an enclosed polygon and the inside resident nodes are relieved from heavy sensing task during the primary spectrum coherent time period.

Obviously, from the computation point of view, larger node density yields better performance. On the other hand, we also want to control the size of the spectrum cluster to make sure communications are effective within the same spectrum cluster.

5 Conclusion

This paper proposed a new guard-resident cooperative spectrum sensing method based on clustering analysis theory and distributed boundary search. We grouped the nodes into two types: guard nodes and resident nodes. The guard nodes will constantly sense the spectrum and inform the environmental changes to their residents. Within the coherent time period, the area formed by the guards becomes a safe zone and the residents can be greatly relieved from spectrum sensing task. The analysis and simulation results suggest that the proposed scheme can reduce the total sensing load of the CRAHN significantly.

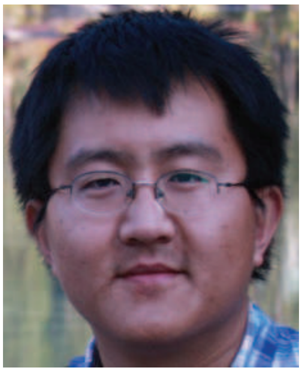
Acknowledgments This work was supported in part by the US National Science Foundation (#1032567), the University of Louisville Research Initiation Grant. The work of Weiyao Lin was supported by the National Natural Science Foundation of China (#61001146). The work of Xudong Wang was supported by the National Natural Science Foundation of China (No. 61172066), Shanghai Pujiang Scholar Program (10PJ1406100), and the MOE Program for New Century Excellent Talents.

References

1. Haykin S (2005) Cognitive radio: brain-empowered wireless communications. IEEE J Sel Area Comm 23(2):201–220
2. Ghasemi A, Sousa ES (2008) Spectrum sensing in cognitive radio networks: requirements, challenges and design trade-offs. IEEE Comm Manag 46(4):32–39
3. Cabric D, Mishra SM, Brodersen RW (2004) Implementation issues in spectrum sensing for cognitive radios. Proc 38th Asilomar Conf Signals Syst Compu 1:772–776
4. Liang YC, Zeng Y, Peh E, Hoang AT (2008) Sensing-throughput tradeoff for cognitive radio networks. IEEE Trans Wireless Comm 7(4):1326–1337
5. Quan Z, Cui S, Sayed AH (2008) Optimal linear cooperation for spectrum sensing in cognitive radio networks. IEEE J Sel Topics Signal Process 2(1):28–40
6. Saad W, Han Z, Basar T, Debbah M, Hjørungnes A (2011) Coalition formation games for collaborative spectrum sensing. IEEE Trans Veh Technol 60(1):276–297
7. Beibei Wang KJ, Liu R, Charles Clancy T (2010) Evolutionary cooperative spectrum sensing game: how to collaborate? IEEE Trans Comm 58(3):890–900
8. Zhang W, Mallik RK, Letaief KB (2009) Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks. IEEE Trans Wireless Comm 8(12):5761–5766
9. Krenik W, Batra A (2005) Cognitive radio techniques for wide area networks. Proc., 42nd Conf. Design Automation 409–412
10. Akyildiz IF et al (2006) NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey. Compu Netw 50 (13):2127–2159
11. Krizman KJ, Biedka TE, Rappaport TS (1997) Wireless position location: fundamentals, implementation strategies and sources of error. Proc 47th Conf Veh Technol 2:919–923
12. Akyildiz IF, Lee W-Y, Chowdhury KR (2009) CRAHNs: cognitive radio ad hoc networks. Ad Hoc Netw 7(5):810–835
13. Hastie T, Tibshirani R, Friedman J (2009) Elements of statistical learning: data mining, inference and prediction, 14.3.12, Hierarchical clustering. Springer Series in Statistics, pp 520–527
14. Rui Xu, Wunsch D II (2005) Survey of clustering algorithms. IEEE Trans Neural Netw 16(3):645–678
15. Yucek T, Arslan H (2009) A survey of spectrum sensing algorithms for cognitive radio applications. IEEE Comm Surveys & Tutorials 11(1):116–130
16. Haykin S, Thomson DJ, Reed JH (2009) Spectrum sensing for cognitive radio. IEEE Proc 97(5):849–877
17. Urkowitz H (1967) Energy detection of unknown deterministic signals. IEEE Proc 55(4):523–531
18. Ma J, Li GY, Juang BH (2009) Signal processing in cognitive radio. IEEE Proc 97(5):805–823
19. McLurkin J, Demaine ED (2009) A distributed boundary detection algorithm for Multi-Robot systems. IEEE Conf. Intelligent Robots and Syst. (IROS): 4791–4798
20. Fekete SP, Kaufmann M (2005) A. Kroeller, and N. Lehmann. A new approach for boundary recognition in geometric sensor networks. Proc 17th Can Conf Comp Geom:82–85
21. A. Kroeller, S. P. Fekete, D. Pfisterer and S. Fischer (2006). Deterministic boundary recognition and topology extraction for large sensor networks. Proc. 7th annual ACM-SIAM symposium on Discrete Algorithm, 1000-1009.
22. Sahai A, Hoven N, Tandra R (2004) Some fundamental limits on cognitive radio. BWRC Cognitive Radio Workshop:1662–1671

Yang Du received the B.E. degree in electrical engineering from Beihang University, Beijing, China in 2005 and now is a Ph.D. student of Beihang University. He was a joint Ph.D. student of Beihang University, China and North Dakota State University, USA during 2009-2011 supported by the China Scholarship Council. His research interests include wireless ad-hoc networks, networked control system and the flocking control system.

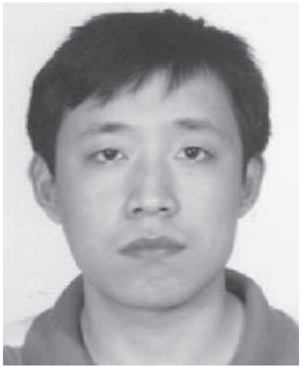
e-mail:doyoung0.618@gmail.com



Hongxiang Li received the B.S. degree from Xi'an Jiaotong University, China in 2000, the M.S. degree from Ohio University in 2004 and the Ph.D. degree from University of Washington, Seattle in 2008, all in electrical engineering.

Now he is an Assistant Professor with the Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY. His research interests include broadband mobile communications and wireless networks.

e-mail: hongxiangli@gmail.com



Weiyao Lin received the B.E. degree from Shanghai Jiao Tong University, China, in 2003, the M.E. degree from Shanghai Jiao Tong University, China, in 2005, and the Ph.D. degree from the University of Washington, Seattle, USA, in 2010, all in electrical engineering. Since 2010, he has been an assistant professor at the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University. His research interests include video processing, machine learning, computer vision and video coding & compression.

Lingjia Liu (S'03-M'08) received the B.S. degree from Shanghai Jiao Tong University, China and the Ph.D. degree from Texas A & M University, all in Electrical Engineering.

He is currently working as an Assistant Professor in the Electrical Engineering and Computer Science Department at the University of Kansas (KU). Prior to joining the EECS at KU, he spent more than 3 years in Dallas Technology Laboratory of Samsung Electronics leading Samsung's work on downlink multi-user MIMO, Coordinated multipoint (CoMP) transmission, and Heterogeneous Networks for 3GPP LTE/LTE-Advanced standards.

His general research interests lie in the areas of wireless communication systems, statistical signal processing, queueing theory, and information theory, with emphasis on delay-sensitive and energy-efficient communication over wireless networks.

Xudong Wang received his B.E. degree in Electric Engineering and his first Ph.D. degree in Automatic Control in 1992 and 1997, respectively, from Shanghai Jiao Tong University, Shanghai, China.

He is currently with UM-SJTU Joint Institute, Shanghai Jiao Tong University. He is also an affiliate faculty member with the Electrical Engineering Department at the University of Washington. Since he received the Ph.D. degree in Electrical and Computer Engineering from Georgia Institute of Technology in August 2003, he has been working as a senior research engineer, senior network architect, and R&D manager in several companies.

His research interests include low-power radio architecture and protocol suite, deep-space network architecture and protocols, cognitive/software radios, LTE-A, wireless mesh networks, cross-layer design, wireless sensor networks, and ultra-wideband networks. He holds several patents on wireless networking technologies and most of his inventions have been successfully transferred to products. Dr. Wang is an editor for Elsevier Ad Hoc Networks and ACM/Kluwer Wireless Networks. He was also a guest editor for several journals. He was the demo co-chair of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MOBIHOC 2006), a technical program co-chair of Wireless Internet Conference (WICON) 2007, and a general co-chair of WICON 2008. He has been a technical committee member of many international conferences and a technical reviewer for numerous international journals and conferences. Dr. Wang is a senior member of IEEE and was a voting member of IEEE 802.11 and 802.15 Standard Committees.

Samee Ullah Khan is Assistant Professor of Electrical and Computer Engineering at the North Dakota State University, Fargo, ND, USA. Prof. Khan has extensively worked on the general topic of resource allocation in autonomous heterogeneous distributed computing systems. As of recent, he has been actively conducting cutting edge research on energy-efficient computations and communications. A total of 118 (journal: 42, conference: 55, book chapter: 12, editorial: 6, technical report: 3) publications are attributed to his name. For more information, please visit: http://sameekhan.org/.

Sentang Wu received the B.E. degree and M.E. degree from Northwestern polytechnical University, Xi'an, China in 1986 and 1988, both in Electrical Engineering and the Ph.D. degree from National Aviation University, Kiev, Ukraine, in 1992 both in Dynamics and Ballistics & Aircraft motion control.

He is now a Professor with the School of Automation Science and Electrical Engineering, Beihang University, Beijing, China, as well as the specialist of the State Administration of Science Technology and Industry and Equipment Department of the Navy.

His research interests include aircraft guidance, navigation and control, the theory of stochastic system and the networked control system under wireless ad-hoc networks.