symmetry

MDPI

Article

User Classification in Crowdsourcing-Based Cooperative Spectrum Sensing

Linbo Zhai [1,2] and Hua Wang [1,*]

[1] School of Computer Science and Technology, Shandong University, Jinan 250100, China; zhai@mail.sdu.edu.cn

[2] Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Shandong Normal University, Jinan 250014, China

* Correspondence: wanghua@sdu.edu.cn

Academic Editor: Chi-Hua Chen

Received: 21 May 2017; Accepted: 3 July 2017; Published: 6 July 2017

Abstract: This paper studies cooperative spectrum sensing based on crowdsourcing in cognitive radio networks. Since intelligent mobile users such as smartphones and tablets can sense the wireless spectrum, channel sensing tasks can be assigned to these mobile users. This is referred to as the crowdsourcing method. However, there may be some malicious mobile users that send false sensing reports deliberately, for their own purposes. False sensing reports will influence decisions about channel state. Therefore, it is necessary to classify mobile users in order to distinguish malicious users. According to the sensing reports, mobile users should not just be divided into two classes (honest and malicious). There are two reasons for this: on the one hand, honest users in different positions may have different sensing outcomes, as shadowing, multi-path fading, and other issues may influence the sensing results; on the other hand, there may be more than one type of malicious users, acting differently in the network. Therefore, it is necessary to classify mobile users into more than two classes. Due to the lack of prior information of the number of user classes, this paper casts the problem of mobile user classification as a dynamic clustering problem that is NP-hard. The paper uses the interdistance-to-intradistance ratio of clusters as the fitness function, and aims to maximize the fitness function. To cast this optimization problem, this paper proposes a distributed algorithm for user classification in order to obtain bounded close-to-optimal solutions, and analyzes the approximation ratio of the proposed algorithm. Simulations show the distributed algorithm achieves higher performance than other algorithms.

Keywords: classification; crowdsourcing; sensing; distributed

1. Introduction

According to a Cisco report [1], wireless traffic has significantly increased over the last few years. This trend has led to spectrum scarcity. On the other hand, the licensed wireless spectrum is poorly utilized within the framework of a fixed wireless spectrum assignment policy. To increase the utilization of the wireless spectrum, cognitive radio technology has recently emerged. When licensed users (also called primary users, PUs) do not utilize the licensed wireless spectrum, cognitive radio allows unlicensed users (also called secondary users, SUs) to access the idle licensed wireless spectrum opportunistically [2]. Therefore, wireless spectrum sensing is the premise for opportunistic access of unlicensed users. To carry out spectrum sensing, it is necessary to model licensed users' activity. In [3], the authors provide a survey of licensed user activity models in cognitive radio networks, and show how these models are performed. When spectrum sensing is carried out, shadowing, multi-path fading, and other issues may result in a single user being assigned an incorrect sensing result. To improve the sensing accuracy, cooperative spectrum sensing has been proposed by multiple users [4]. Multiple

users attempt to sense the channel, and send their sensing reports to a fusion center (FC), which makes the final decision about the channel state.

Unfortunately, there may be some malicious mobile users, so cooperative spectrum sensing is vulnerable to malicious mobile users [5]. Malicious users will send false sensing information to disrupt cooperative spectrum sensing, or to gain unfair opportunity to access the channel. In [6], the authors propose the expectation maximization (EM) algorithm to determine the channel state and classify the users. In [7], SUs use energy detection and transmit signal strengths that they have sensed to the fusion center (FC). Based on these signal strengths, the FC decides whether the channel is being used by PUs or not. In [8], the authors propose a joint spectrum access and sensing framework to thwart malicious behaviors for both rational and irrational malicious users. Attack prevention is considered in collaborative spectrum sensing in [9]. SUs send their binary reports about the presence or absence of PUs to the FC. The FC uses the q-out-of-m (OR) rule to decide the channel state. There is another widely used method called reputation-based detection. In [10], the detection of the channel state and malicious radios is carried out in two steps. Firstly, the current channel state is decided based on the q-out-of-m rule. Then, a user will be identified as a malicious user if its past decisions during a certain period are different from the decisions of the FC, beyond a certain threshold. In [11], a reputation metric is defined based on two types of attackers: one kind of attacker sends busy reports when the channel is sensed to be idle, and the other kind of attacker sends idle reports when the channel is sensed to be busy. It is assumed that all honest users are known to the FC. In [12], the authors propose a method for spectrum sensing based on the autocorrelation of the received samples. The method is evaluated by practical experiments. The three most fundamental spectrum sensing techniques—i.e., energy detection-based, autocorrelation-based, and matched filter-based sensing—are examined and evaluated in [13]. Furthermore, compressive sensing has also been researched [14].

In all of the aforementioned literature, centralized algorithms are implemented in the FC. That means that the FC receives the sensing results from users and makes a decision about the channel state based on these sensing results. However, the system may collapse if the FC is attacked by malicious users. Moreover, a centralized system is less flexible for users' dynamics. In order to overcome the faults of centralized algorithms, some literature has examined distributed algorithms. In [15], the authors focused on the performance analysis of cluster-based heterogeneous vehicular networks, and analysis models of intracluster and intercluster communications are designed based on a Markov queuing model. In [16], a new distributed and cooperative scheme was proposed to detect and classify incumbents and licensed shared-access licensees in a network. The authors proposed a clustering strategy for cooperative spectrum sensing, with the clustering considering the differences in underlying hidden Markov models associated with the detection of distinct licensed users.

In traditional classification, users are simply divided into honest ones and malicious ones. However, users should not be divided into just two classes (honest and malicious). There are two reasons for this. On the one hand, because shadowing, multi-path fading, and other issues may influence the sensing results, honest users may have different sensing results based on the area in which they are located. Based on the sensing results, honest users should be divided into several classes. On the other hand, as malicious users may act differently from one another, they should also be divided into several classes. Some malicious users are always yes, making other users decide that the channel is being used by PUs. Some malicious users are always no, making other users decide that the channel is idle. There may be another kind of malicious user trying to confuse other users under both channel states. Therefore, it is necessary to classify mobile users into more than two classes. Additionally, mobile users such as smartphones and tablets are being used more and more in our daily life. These mobile users can sense the wireless spectrum. Therefore, spectrum sensing tasks can be assigned to mobile users. This is called the crowdsourcing method.

Therefore, mobile users are assigned spectrum sensing tasks by the crowdsourcing method in this paper. Based on other mobile users' sensing reports, each mobile user makes a decision about the channel state independently. Since there may be some malicious mobile users that send false sensing

reports deliberately, for their own purposes, decisions about channel states will be influenced by false sensing reports. Therefore, it is necessary to classify mobile users in order to distinguish malicious users. According to sensing reports, mobile users should be divided into several classes, rather than only two classes (honest and malicious). Due to the lack of prior information on the number of user classes, this paper casts the problem of mobile user classification as a dynamic clustering problem that is NP-hard. The paper uses the interdistance-to-intradistance ratio of clusters as the fitness function for evaluating the clustering effect, and aims to maximize the fitness function. To cast this optimization problem, this paper proposes a distributed algorithm for user classification in order to obtain bounded close-to-optimal solutions, and analyzes the approximation ratio of the proposed algorithm. Simulations show that the distributed algorithm achieves higher performance than other algorithms.

In this paper, the problem of classifying mobile users is studied based on the sensing reports. The main contributions of this paper are summarized below.

- Considering that there may be several types of mobile user, the paper casts user classification as a dynamic clustering problem without prior information about network parameters. The fitness function is designed for the dynamic clustering problem.
- The paper proposes a distributed algorithm for user classification to obtain bounded close-to-optimal solutions. Each mobile user, rather than the FC, carries out the process of classification independently according to all sensing reports. Then, the approximation ratio of the proposed algorithm is analyzed with a Markov chain.

The rest of the paper is organized as follows. In Section 2, the system model of user classification is described. In Section 3, the paper proposes a distributed algorithm to solve the user classification, and analyzes the approximation ratio of the proposed algorithm. In Section 4, the proposed algorithm is evaluated by simulation results. Finally, conclusions are shown in Section 5.

2. The System Model

Mobile users are assigned the task of sensing the wireless channel. Then, they broadcast their sensing reports, reflecting the signal strength to others. Each mobile user, rather than a fusion center, makes a decision about the channel state independently, based on the sensing reports.

2.1. Behavior of Different Users

It is assumed that there are M mobile users in the wireless system. Each mobile user senses the channel to find whether the channel is being used by PUs or not. According to energy detection, the sensing outcome of each user follows a Gaussian distribution. For a user i, the sensing result $S(i)$ can be described as

$$S(i) - \begin{cases} \mathrm{N}(n_0, 2n_0^2/m) & H_0 \\ \mathrm{N}(n_0 + p_i, 2(n_0 + p_i)^2/m) & H_1 \end{cases} \quad (1)$$

where n_0 denotes the noise power, p_i denotes the signal strength of primary users received by user i, m denotes the number of samples, H_0 denotes that the channel is idle, and H_1 denotes that the channel is being used by primary users.

After users' sensing is complete, they will broadcast sensing reports consisting of multiple bits that denote the signal strength they have sensed. Different kinds of users will act differently. The details are described as follows.

For honest users, the sensing reports denoting the signal strength they have sensed will be broadcast without any modification. However, honest users may not all have the same sensing result after spectrum sensing. There are two reasons for this. As honest users may be in different locations, some factors, such as multipath fading, shadowing, and others, may influence the sensing results. If honest users employ different sensing technologies, they may also have different sensing results. Therefore, honest users can be divided into several kinds according to their sensing outcomes.

For malicious users, they modify their sensing results before broadcasting sensing results to others. For instance, one kind of malicious user is always yes, and may change the sensing results to a high value. This means that there are active primary users in the channel. Then, they broadcast sensing reports following modified sensing results to make other users decide that the channel is being used by PUs. Another kind of malicious user is always no, and may change the sensing results to a low value. This means there are no active primary users in the channel. Then, they broadcast sensing reports following modified sensing results to make other users decide that the channel is idle (unused by primary users). There may be the third kind of malicious users who try to confuse other users under both channel states.

In a word, each user broadcasts its sensing report with multiple bits reflecting signal strength, rather than only one bit denoting that the channel is idle or busy. When a user receives sensing reports from others, it needs to classify these reports and makes a decision about the channel state based on the credible reports.

As in Figure 1, there are many mobile users in the cognitive radio networks. Malicious users can be divided into two kinds. One kind of malicious user wants to make the channel state busy, while another kind of malicious user wants to make the channel state idle. Since honest users are in three different locations, they may have different sensing results because of multipath fading, shadowing, or other reasons. Therefore, honest users are divided into three classifications.

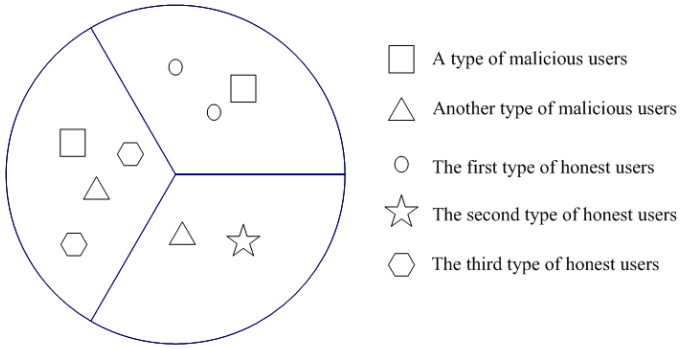

Figure 1. Several kinds of honest and malicious users.

2.2. Problem Formulation

According to user reports, each mobile user needs to distinguish different kinds of mobile users independently. This is a problem of clustering. The main goal of clustering is to maximize both homogeneity within the same cluster, and heterogeneity among different clusters. Without prior knowledge of the system, such as the number of mobile user classes, this becomes a dynamic clustering problem.

After the sensing process is completed, each mobile user broadcasts its sensing report consisting of multiple bits which denote the signal strength. When a mobile user receives sensing reports of other users, it divides all users into several clusters based on these reports. The main goal of clustering is to maximize both the homogeneity within the same cluster, and the heterogeneity among different clusters. Here, the paper uses the Euclidean distance of two users' sensing reports as the distance between two users. If the sensing reports of two users are quite similar, the two users may be in a same cluster. Otherwise, if the sensing reports of two users are quite different, the two users may be in different clusters. There may be multiple clustering configurations. Each clustering configuration corresponds to a division of all mobile users. The paper aims to obtain an optimized configuration to divide mobile users into several clusters to realize the maximum similarity in the same cluster and the maximum heterogeneity for different clusters.

Let M denote the number of mobile users in the system. For each mobile user, there is only a simple choice as to whether the user is to be a cluster center or not. A clustering configuration f is

a vector indicating that the choice of each mobile user, i.e., $f = \{f_1, f_2, \ldots, f_M\}$, where $f_i \in \{0, 1\}$, 0 denotes that the user is a cluster center, and 1 denotes that the user is not a cluster center. This paper defines F as the set of all feasible f. Given a clustering configuration f, the corresponding fitness function $Fit(f)$ is used to denote clustering effect which reflects both the homogeneity within the same cluster and the heterogeneity among different clusters. This paper aims to maximize $Fit(f)$ by choosing a proper clustering configuration f. $Fit(f)$ can be described as

$$Fit(f) = \frac{1}{N(f)} \sum_{i=1}^{N(f)} R_f(i) \tag{2}$$

where $N(f)$ denotes the number of clusters with the clustering configuration f, and $R_f(i)$ denotes the minimized interdistance-to-intradistance ratio of the ith cluster. The interdistance-to-intradistance ratio is used to measure the relationship between distance within a cluster and that among different clusters. Let e_j be the mean distance from users in the jth cluster to the center of the jth cluster, e_i be the mean distance from users in the ith cluster to the center of the ith cluster, and m_{ji} is the distance between the centers of the jth and ith clusters. Then, $m_{ji}/(e_i + e_j)$ is used to denote interdistance-to-intradistance ratio of the jth and ith clusters. The higher the interdistance-to-intradistance ratio is, the better the clustering effect. Under the clustering configuration f, the lowest interdistance-to-intradistance ratio is used to evaluate the clustering effect of configuration f. Therefore, $R_f(i)$ can be described as

$$R_f(i) = \min_{j \neq i} \frac{m_{ji}}{e_j + e_i} \tag{3}$$

To obtain optimized clustering, the paper aims to maximize $Fit(f)$ by choosing a proper clustering configuration f. The maximum fitness function can be derived as

$$Fit_m = \max_{f \in F} Fit(f) = \max_{f \in F} \frac{1}{N(f)} \sum_{i=1}^{N(f)} R_f(i) \tag{4}$$

3. A Distributed Algorithm

From an optimization perspective, clustering can be considered to be one particular kind of NP-hard problem [17]. Therefore, the optimization problem is hard to solve. In this section, a distributed algorithm is designed to implement dynamic clustering.

3.1. Algorithm Description

Initially, each user randomly chooses whether to be a cluster center or not. Then, they broadcast their roles. When a mobile user receives other users' roles, it obtains the current clustering configuration f. Then, each user senses the wireless spectrum and broadcasts its sensing report to other mobile users. When all mobile users have received other users' reports, each user chooses the nearest cluster center based on Euclidean distance in order to join the cluster, and calculates $R_f(i)$ ($i = 1, 2, \ldots, N(f)$) independently. Then, each mobile user generates a random number following exponential distribution, and its mean equals a positive constant C.

All mobile users count down the random numbers following exponential distribution. When the countdown of a mobile user expires, this mobile user may change its current role, i.e., from a non-center to a cluster center, or a from cluster center to a non-center, with the probability $p_{ff'}$, as described in (5).

$$p_{ff'} = \frac{1}{\exp\left(\frac{\beta}{N(f)} \sum_{i=1}^{N(f)} R_f(i)\right)} \tag{5}$$

where β is a positive constant. The mobile user changes its current role following the probability $p_{ff'}$, or stays in the role with the probability $1 - p_{ff'}$. If the mobile user stays in its role, it regenerates

a new random number following exponential distribution and counts down. If the mobile user changes its current role, a new clustering configuration f' appears. This mobile user broadcasts the new clustering configuration f' to other mobile users and generates a new random number following exponential distribution to start a new countdown process. When other users receive the new clustering configuration f', they calculate $R_{f'}(i)$ ($i = 1, 2, \ldots, N(f')$) and continue their countdown processes. When the countdown of a mobile user expires, the transition probability is calculated based on $R_f(i)$. This implementation is named the Role-Changing (RC) algorithm.

Each mobile user carries out the Role-Changing (RC) algorithm independently. The distributed algorithm is described as follows.

Algorithm 1: Role-Changing algorithm for user i

Input β
1: Mobile user i chooses its role randomly and broadcasts its role.
2: After user i receives other users' roles, it obtains the current clustering configuration f.
3: Mobile user i broadcasts its sensing reports and receives other users' reports.
4: Then, user i calculates $\sum_{i=1}^{N(f)} R_f(i)/N(f)$ independently.
5: User i generates a timer following exponential distribution and begins to count down.
6: When the timer expires, user i changes its role with $p_{ff'}$ or stay in its role with $1 - p_{ff'}$.
7: If user i changes its role, it broadcasts the new clustering configuration f'.
8: Other users calculate $\sum_{i=1}^{N(f\prime)} R_{f\prime}(i)/N(f\prime)$ under the new clustering configuration f'.
9: User i generates a new timer following exponential distribution and begins to count down. Then, it repeats step 6–9.

when a mobile user implements the RC algorithm, it can classify mobile users into several classes. It is assumed that there is one kind of honest user, with more members than any other kind. A mobile user will choose one class with the largest number of users as honest users. Then, the user makes a decision about the channel state based on these honest users' reports. Each user could calculate the average signal strength based on these honest users' reports. Let Th denote the pre-defined decision threshold. If the average signal strength is higher than the decision threshold, the user decides the channel is being used by primary users. Otherwise, the user decides the channel is idle.

3.2. Analysis of Approximation Ratio

The approximation ratio is defined as the maximum fitness function in the proposed algorithm to that in theory. Here, a Markov chain is used to describe the transition among clustering configurations in the system.

According to the Role-Changing (RC) algorithm, each clustering configuration f corresponds to a state in the system. Therefore, there are finite states in the system, and the number of states equals $|F|$ where F is the set of all feasible f. Each clustering configuration is reachable from any adjacent state by one-step transition. Then, the stationary distribution of the Markov chain is calculated.

Let M denote the number of mobile users in the system and $W_{ff'}$ denote the probability that the system moves to state f' from state f after count-down expiration. In the RC algorithm, a mobile user changes its role following the probability $p_{ff'}$ in (5). Therefore, $W_{ff'}$ can be obtained.

$$W_{ff\prime} = \frac{p_{ff\prime}}{M} \tag{6}$$

The probability $p_{ff'}$ denotes the probability that a mobile user will change its current role, while $W_{ff'}$ denotes the probability that the system moves to the state f' from the state f. When there are M users in the system, the probability that the user is chosen is $1/M$. Therefore, the probability that the system moves to state f' from state f is $p_{ff'}/M$. According to the RC algorithm, each mobile user counts

down following an exponential distribution that is memoryless. As each mobile user counts down with the rate $1/C$, the rate of the system count-down expiration is M/C. Then, the transition rate $q_{ff'}$ from state f to state f' can be obtained.

$$q_{ff\prime} = W_{ff\prime} M/C = p_{ff\prime}/C \tag{7}$$

Let p_f^* be the stationary distribution under state f. The detailed balance equation should be satisfied as follows:

$$p_f^* q_{ff\prime} = p_{f\prime}^* q_{f\prime f} \tag{8}$$

$$\sum_{f \in F} p_f^* = 1 \tag{9}$$

From (8) and (9), the stationary distribution p_f^* of the Markov chain can be obtained.

$$p_f^* = \frac{\exp(\frac{\beta}{N(f)} \sum_{i=1}^{N(f)} R_f(i))}{\sum_{f\prime \in F} \exp(\frac{\beta}{N(f\prime)} \sum_{i=1}^{N(f\prime)} R_{f\prime}(i))}, f \in F \tag{10}$$

The stationary distribution p_f^* can also denote the percentage of the time that the system is under the cluster configuration f. It happens to be the optimal solution of the problem expressed as in (11).

$$\begin{aligned} &\max_{p_f \geq 0} \sum_{f \in F} p_f \tfrac{1}{N(f)} \sum_{i=1}^{N(f)} R_f(i) - \tfrac{1}{\beta} \sum_{f \in F} p_f \log p_f \\ &s.t. \sum_{f \in F} p_f = 1 \end{aligned} \tag{11}$$

where β is a positive constant.

Using the stationary distribution p_f^* in (10), the optimal value of (11) is

$$\gamma = \frac{1}{\beta} \log(\sum_{f \in F} \exp(\frac{\beta}{N(f)} \sum_{i=1}^{N(f)} R_f(i))) \tag{12}$$

The aforementioned analysis means that the distributed algorithm can obtain the optimal value of the optimization problem in (11). Let $V(f) = \sum_{i=1}^{N(f)} R_f(i)/N(f)$. From (12), it is obtained that

$$\begin{aligned} \gamma &= \tfrac{1}{\beta} \log(\exp(\beta \max_{f \in F} V_f) \sum_{f \in F} \exp \beta(V_f - \max_{f \in F} V_f)) \\ &\leq \tfrac{1}{\beta} \log(\exp(\beta \max_{f \in F} V_f)|F|) \\ &= \max_{f \in F} V_f + \tfrac{1}{\beta} \log|F| \\ &= \max_{f \in F} \tfrac{1}{N(f)} \sum_{i=1}^{N(f)} R_f(i) + \tfrac{1}{\beta} \log|F| \end{aligned} \tag{13}$$

Let the approximation gap denote the difference between the maximum fitness function in the proposed algorithm and that in theory. Then, the approximation gap of the proposed algorithm can be obtained.

$$|\gamma - \max_{f \in F} \frac{1}{N(f)} \sum_{i=1}^{N(f)} R_f(i)| \leq \frac{1}{\beta} \log|F| \tag{14}$$

where $|F|$ denotes the number of all clustering configurations.

Therefore, the maximum approximation gap, meaning the upper bound of the approximation gap, can be derived. It can be obtained as

$$\frac{1}{\beta}\log|F| \tag{15}$$

According to Formulation (15), the approximation gap approaches zero when β approaches infinity. This means the distributed algorithm is more accurate with larger values of β.

4. Simulations

In this section, the results of the distributed algorithm are compared with those of the reputation-based classification [10]. With reputation-based classification, users are classified into two kinds of users (honest and malicious). The reputation method is related to the duration T. That means that the sensing reports in the duration T are collected to evaluate the reputation of users. Based on their reputation, the classification is carried out. The average solution is derived by running the algorithm 500 times. The parameters are described as follows. The sensing region is a circular region with a 100 m radius. The sensing region is equally divided into three sub-regions. All users are located in the region randomly, and employ energy detection. The sensing outcome of each user follows a Gaussian distribution. In each sub-region, the sensing results of mobile users may be different from mobile users' results in other sub-regions because of multipath fading and shadowing. Therefore, there are three kinds of honest users.

Let M denote the number of mobile users. β is set to 20 and 30. As M varies from 10 to 18, Table 1 shows the approximation gap of the proposed RC algorithm. As shown in Table 1, it describes the approximation gap of the proposed algorithm increases as the number of mobile users increases. When there are more mobile users, $|F|$, equaling 2^M, will increase. Therefore, the approximation gap increases as $|F|$ increases. In addition, it is also shown that the larger β is, the smaller the approximation gap is. This means the distributed algorithm is more accurate with larger values of β.

Table 1. The approximation gap of the distributed Algorithm.

β \ M	10	12	14	16	18
20	0.34	0.41	0.48	0.55	0.62
30	0.23	0.27	0.32	0.36	0.41

It is assumed that there are two cases of malicious users. In the first case, there are two kinds of malicious users. One kind of malicious user always tries to make the channel state busy, and the other kind of malicious user always tries to make the channel state idle. This case is depicted in Figure 1. In the second case, there is only one kind of malicious user, where malicious users try to confuse the decision under both channel states.

The misclassification rate is defined as the ratio of mistakenly judging one kind of user to be another kind. The performance of the classification is shown in Figures 2 and 3. As shown in Figure 2, the misclassification rate varies with T when there are 20 users. Figure 3 depicts the estimation error regarding the channel state when there are 20 users. From Figures 2 and 3, it can be seen that the accuracy of estimation and user classification improves with T and the distributed algorithm (RC) outperforms the reputation-based method. In [10], the authors proposed the reputation-based method to classify users into two kinds. The method is not capable of classifying users into more than two classes. Therefore, there is no misclassification of reputation-based method about the first case in Figure 2.

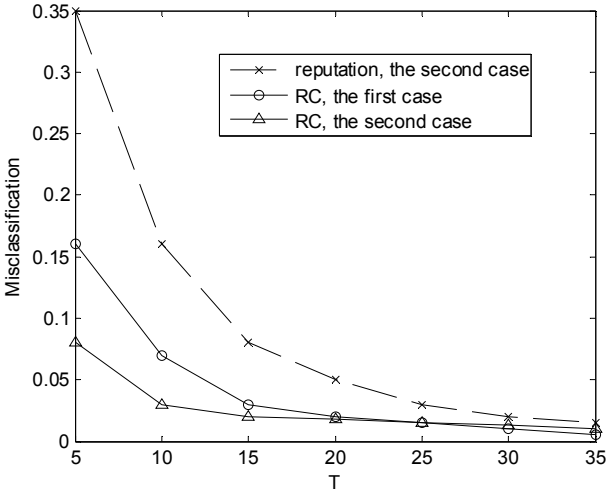

Figure 2. Misclassification as a function of T.

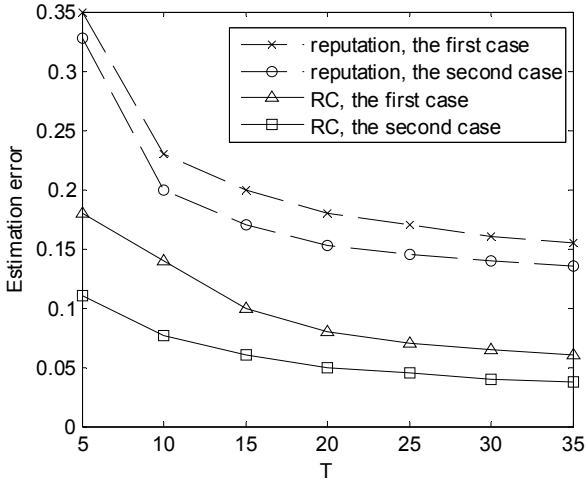

Figure 3. Estimation error as a function of T.

5. Conclusions

This paper studies user classification in order to distinguish malicious users. Due to the lack of prior information on the number of user classes, this paper casts the problem of mobile user classification as a dynamic clustering problem that is NP-hard. To carry out user classification, we designed a distributed algorithm to obtain bounded close-to-optimal solutions, and analyzed the approximation ratio of the proposed algorithm. Simulations show that the distributed algorithm achieves higher performance than other algorithms. In this paper, it was assumed that all users are always in the sensing area. Therefore, the number of users is invariable and the mobility of users is not considered. In the future, the case that users could enter and leave the sensing area should be studied.

Acknowledgments: This research was supported in part by Natural Science Foundation of Shandong Province, China (No. BS2015DX003), and in part by China Postdoctoral Science Foundation (No. 2014M561930). We received funds to cover the costs of publishing in open access.

Author Contributions: Linbo Zhai and Hua Wang conceived and designed the experiments; Linbo Zhai performed the experiments and analyzed the data; Linbo Zhai and Hua Wang wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cisco. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014, Tech.*; Cisco: San José, CA, USA, 2014.
2. Akhtar, F.; Rehmani, M.H.; Reisslein, M. White space: Definitional perspectives and their role in exploiting spectrum opportunities. *Telecommun. Policy* **2016**, *40*, 319–331. [CrossRef]
3. Saleem, Y.; Rehmani, M.H. Primary radio user activity models for cognitive radio networks: A survey. *J. Netw. Comput. Appl.* **2014**, *43*, 1–16. [CrossRef]
4. Yucek, T.; Arslan, H. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Commun. Surv. Tutor.* **2009**, *11*, 116–130. [CrossRef]
5. Fragkiadakis, A.; Tragos, E.; Askoxylakis, I. A survey on security threats and detection techniques in cognitive radio networks. *IEEE Commun. Surv. Tutor.* **2012**, *14*, 1–18. [CrossRef]
6. Soltanmohammadi, E.; Naraghi-Pour, M. Fast Detection of Malicious Behavior in Cooperative Spectrum Sensing. *IEEE J. Sel. Commun.* **2014**, *32*, 377–386. [CrossRef]
7. Min, A.; Shin, K.; Hu, X. Secure cooperative sensing in IEEE 802.22 WRANS using shadow fading correlation. *IEEE Trans. Mobile Comput.* **2011**, *10*, 1434–1447. [CrossRef]
8. Wang, W.; Chen, L.; Shin, K. Thwarting Intelligent Malicious Behaviors in Cooperative Spectrum Sensing. *IEEE Trans. Mobile Comput.* **2015**, *14*, 2392–2405. [CrossRef]
9. Duan, L.; Min, A.; Huang, J.; Shin, K. Attack prevention for collaborative spectrum sensing in cognitive radio networks. *IEEE J. Sel. Areas Commun.* **2012**, *30*, 1658–1665. [CrossRef]
10. Rawat, A.; Anand, P.; Chen, H.; Varshney, P. Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks. *IEEE Trans. Signal Process.* **2011**, *59*, 774–786. [CrossRef]
11. Penna, F.; Sun, Y.; Dolecek, L.; Cabric, D. Detecting and counteracting statistical attacks in cooperative spectrum sensing. *IEEE Trans. Signal Process.* **2012**, *60*, 1806–1822. [CrossRef]
12. Reyes, H.; Subramaniam, S.; Kaabouch, N.; Hu, W. A spectrum sensing technique based on autocorrelation and Euclidean distance and its comparison with energy detection for cognitive radio networks. *Comput. Electr. Eng.* **2015**, *52*, 319–327. [CrossRef]
13. Manesh, M.; Apu, M.; Kaabouch, N.; Hu, W. Performance evaluation of spectrum sensing techniques for cognitive radio systems. In Proceedings of the IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 20–22 October 2016.
14. Arjoune, Y.; Kaabouch, N.; Ghazi, H.; Tamtaoui, A. Compressive sensing: Performance comparison of sparse recovery algorithms. In Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 9–11 January 2017.
15. Zheng, Q.; Zheng, K.; Sun, L.; Leung, V.C.M. Dynamic Performance Analysis of Uplink Transmission in Cluster-Based Heterogeneous Vehicular Networks. *IEEE Trans. Veh. Technol.* **2015**, *64*, 5584–5595. [CrossRef]
16. Sobron, I.; Martins, W.A.; de Campos, M.L.R.; Velez, M. Incumbent and LSA Licensee Classification Through Distributed Cognitive Networks. *IEEE Trans. Commun.* **2016**, *64*, 94–103. [CrossRef]
17. Falkenauer, E. *Genetic Algorithms and Grouping Problems*; Wiley: New York, NY, USA, 1998.