EFFECT OF SOCIAL RELATIONS ON COOPERATIVE SENSING IN COGNITIVE RADIO NETWORKS

by

Mehmet Can Güven

B.S., Computer Engineering, Boğaziçi University, 2010

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University
2013

ACKNOWLEDGEMENTS

I would like to thank first and foremost to my family for the great support they have shown during my academic career as a whole. I am very happy and lucky to have them in my life.

During both my B.S. and M.S. studies, fellow academicians Suzan Bayhan and Gürkan Gür have been a great inspiration to me. They have provided most valuable help and insight about researching on Cognitive Radio Networks. They therefore have my sincere gratitude for all the contributions they have made to my research and me as a researcher in general. I especially thank Suzan for both the creative ideas she asserted during the writing process of this thesis and for her helpful comments to my work.

I would like to offer my sincere gratitude to my instructor and thesis advisor Fatih Alagöz, who has been a great tutor to me starting from the times I have first stepped foot into the academic life with my first paper to the SIU conference in 2010.

I also would like to thank TUBITAK, with their support to my finances during my studies, via the BIDEB-2210 M.S. scholarship program. I appreciate their policies of supporting successful B.S. students with scholarships during their M.S. studies.

Last but not the least, I would like to offer my deepest thanks to the other people that I have neglected to mention here by name. They all have supported my thesis in various ways and made it possible to develop, creatively resume and finish this work.

ABSTRACT

EFFECT OF SOCIAL RELATIONS ON COOPERATIVE SENSING IN COGNITIVE RADIO NETWORKS

Previous works in cognitive radio networks (CRNs) have shown that cooperation in sensing improves sensing reliability and in turn enhances the network throughput. Although this altruistic cooperative behavior is accepted as the default mode of operation, it may often be invalid under practical circumstances. In this thesis, we loosen this assumption and introduce a cooperative mode of operation conditioned on social relations between Cognitive Radios (CRs). Rather than taking CRs as wireless devices with no context, we associate each CR with its user that has some social relations, e.g. friendship, community, selfishness. Using these relations among CRs, we propose a social-aware cooperative sensing scheme (SAC) and analyze its effects on sensing performance. We examine that exploiting social metrics is highly beneficial for cooperative sensing in CRNs and a model with social relations embedded will fit better to the next decade's networking paradigm. Furthermore, we show in this thesis that a social aware sensing scheme outperforms a randomly cooperating candidate selecting scheme in terms of opportunity discovery, cooperation message overhead, and avoidance of cooperation with malicious users under various simulation scenarios.

ÖZET

SOSYAL İLİŞKİLERİN KAVRAMSAL RADYO AĞLARINDA MÜŞTEREK SEZİMLEMEYE ETKİLERİ

Kavramsal Radyo Ağlarındaki çalışmalar, kanal sezimlemede müşterek operasyonun sezimleme güvenilirliğini ve dolayısıyla ağ verimliliğini artırdığını göstermiştir. Fakat müşterek kanal sezimleme bu çalışmalarda varsayılan operasyon modu olarak kabul edilmektedir, ki bu varsayım birçok sebepten ötürü gerçekleşmeyebilir. Çünkü kavramsal radyolar batarya ile çalışan cihazlardır ve zaten nadir olan boş kanal fırsatlarını yakalamak için sezimleme yaparken değerli bir kaynak olan enerjilerini ve zamanlarını sadece müşterek sezimlemede yardım isteyen diğer kavramsal radyolara harcamak istemeyebilirler. Bu çalışmamızda biz bu varsayımı ortadan kaldırıp kavramsal radyolar arası sosyal ilişkilere dayandırılan bir müşterek sezimleme sistemi oluşturduk. Dolayısıyla kavramsal radyoları arkadaşlık bağı, sosyal grup ve bencillik gibi özellikleri olan kullanıcıları ile bağdaştırdık. Kavramsal radyolar arasındaki bu ilişkilere dayanarak, bu ilişkilerin farkında olan bir müşterek sezimleme planı geliştirdik ve bu planın sezimleme performansı üzerindeki etkilerini inceledik. Biz, sosyal metriklerin kullanıldığı bir sezimleme planının müşterek sezimlemeyi desteklediğine ve sosyal ilişkilerle iç içe bir modellemenin yeni çağın ağlarına daha uygun olacağına inanıyoruz. Çalışmamızda sosyal ilişkilerin farkında olan müşterek sezimleme planımızın, müşterek sezimleme için rastgele kavramsal radyo seçiminde bulunan bir modele göre, kanal fırsatları yakalama, haberleşme yükü ve saldırgan kullanıcılar ile müşterek sezimleme yapmama açılarından çeşitli benzetim senaryoları altında daha iyi bir performans sergilediğini gösterdik.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$1_{[X]}$	Indicator function taking value 1 if the boolean expression X is true, 0 otherwise
C	Community information of a Cognitive Radio
d_m	Malicious user density
e	Education information
$e_{default}$	Default sympathy value for an education information
e_{friend}	Sympathy value for a Cognitive Radio User's friend's education information
e_{main}	Sympathy value for a Cognitive Radio User's own education information
f_s	Sampling rate
H	The final decision on the spectrum occupancy state
H_0	Real spectrum occupancy state
$H(i)$	The sensing outcome of CR_i
L_c	Cooperation Score List of a CR
${L_c}^p$	Previous Cooperative Sensing Performance counterpart of the Cooperation Score List
${L_c}^r$	Most Recent Sensing Performance counterpart of the Cooperation Score List
L_f	Friend List information of a Cognitive Radio
L_s	Sympathy List information of a Cognitive Radio
${L_s}^i$	Sympathy List of CR_i
${L_s}^i(e)$	Education sympathy for education type e in the sympathy list of CR_i
${L_s}^i(pi)$	Personal interest sympathy for interest type pi in the sympathy list of CR_i
M_s	Modifier of selfishness
N_c	Number of Communities
N_{coop}	Maximum allowed number of Cognitive Radios for cooperative sensing

n_{coop}	Number of cooperations that a CR has performed with another CR, specific to each cooperated CR
n_{coop}^{min}	Minimum number of cooperations required before the cooperation score fully takes effect
N_{CR}	Number of Cognitive Radios in the system
N_e	Number of education types
N_g	Number of grids in the system
N_{pi}	Number of personal interest types
N_s	Number of similar personal interests
N_{sim}	Number of simulation repetitions run for statistical betterness
N_T	Number of timeslots in a single simulation run
$p_s(i)$	Probability of sensing of CR_i for a cooperation requesting Cognitive Radio
P_d	Probability of detection for local sensing
P_f	Probability of false alarm for local sensing
P_{idle}	Probability of idle for the primary user channel
pi	Personal interest
$pi_{default}$	Default sympathy value for a personal interest
pi_{friend}	Sympathy value for a Cognitive Radio User's friend's interest
pi_{main}	Sympathy value for a Cognitive Radio User's own personal interest
pi_{side}	Sympathy value for similar interests to a Cognitive Radio User's own personal interest
$s(j)$	Total Score of CR_j for another Cognitive Radio
s_c	Community Score of a Cognitive Radio
$s_c(j)$	Community Score of a Cognitive Radio for CR_j
s_e	Multiplier for the education score
s_f	Friendship Score of a Cognitive Radio
$s_f(j)$	Friendship Score of a Cognitive Radio for CR_j
s_{friend}	Score multiplier for a friend Cognitive Radio
s_{fof}	Score multiplier for a friend-of-friend Cognitive Radio
s_l	Cooperation Score of a Cognitive Radio

$s_l(j)$	Cooperation Score of a Cognitive Radio for CR_j
s_{pi}	Multiplier for the personal interest score
t_{CR}	Minimum sensing duration for a Cognitive Radio to sense the channel
$\mathcal{Q}(\cdot)$	Complementary distribution function of the standard Gaussian
α	Multiplier to prioritize history of cooperations over the last cooperation performance
α_c	Community Score multiplier for normalization
α_{ess}	The effect of score on the selfishness of a Cognitive Radio
α_f	Friendship Score multiplier for normalization
α_l	Cooperation Score multiplier for normalization
$\alpha_{correct}$	Sympathy list value multiplier for cooperating Cognitive Radios that have sensed the channel correctly
α_{error}	Sympathy list value multiplier for cooperating Cognitive Radios that have made a sensing error
α_{reject}	Sympathy list value multiplier for cooperation rejecting Cognitive Radios
β	Cooperation tendency of a Cognitive Radio or the default probability of sensing for a Cognitive Radio
γ_{CR}	Received Signal to Noise Ratio of the channel link associated with the Cognitive Radio
γ_g	Channel quality of a grid
λ_{coop}	Cooperation Threshold
μ_g	Channel quality distribution mean of a grid
σ_{CR}	Received Signal to Noise Ratio variance of a Cognitive Radio in a grid
σ_g	Channel quality distribution variance of a grid

LIST OF ACRONYMS/ABBREVIATIONS

AWGN	Additive White Gaussian Noise
CR	Cognitive Radio
CRN	Cognitive Radio Network
DSA	Dynamic Spectrum Access
DTN	Delay Tolerant Network
FC	Fusion Center
FCC	Federal Communications Commision
PSN	Pocket Switched Network
PU	Primary User
RAND	Social Unaware Cooperative Spectrum Sensing Scheme
SAC	Social-Aware Cooperative Spectrum Sensing
SNR	Signal to Noise Ratio

1. INTRODUCTION

There is an explosive growth in the amount of wireless devices in the world. This brings with it an equally fast growth in the demand for wireless spectrum. Most of these devices have to operate in the current unlicensed frequency bands due to the present regulation framework. Unfortunately, due to the same regulations, wireless devices have to use only the spectrum for which it has the license to access, and the majority of the spectrum has already been allocated to license holders for prolonged periods of time (tens of years) over large geographical regions. Since the unlicensed spectrum forms only a small portion of the wireless bands, the current regulations are inedaquate for supplying this increasingly growing demand for additional spectrum.

In other words, today's spectrum access policy, so called *static spectrum access*, dictates that the licensed spectrum is reserved for licensed users' exclusive use, leading to a false perception that the current spectrum is insufficient. However, the research performed by various organizations such as Federal Communications Commission (FCC) show that the spectrum, which is perceived to be scarce due to pre-allocations by license holders, is in fact underutilized. Furthermore, there have been measurement campaigns in various parts of the world [1] showing that the *static spectrum access* policy has also led to overcrowding of the spectrum at some portions of the spectrum as well as underutilization at others. This inefficiency needs to be resolved in order to meet the surging wireless data demand.

As a resolution to this problem, *dynamic spectrum access* (DSA), a more agile spectrum access paradigm that lets unlicensed users to use the licensed spectrum when its licensed users, also called *primary users* (PU), do not transmit, has emerged. In this opportunistic spectrum access concept, wireless devices can use the frequency bands that are not currently being used by its licensed owners, so called spectrum holes, but they have to vacate when the PUs access these bands. Like static spectrum access, the right to use the licensed band belongs to the PU. However, the DSA users, or in other words, *secondary users* (SU), can access the licensed spectrum by making sure that they do not disrupt the communications of the PUs. This protective mode of operation is achieved by *spectrum sensing*, the act of analyzing spectrum for characterizing an ongoing PU communication in the band of interest as well as locating the unused spectrum bands for SU access. In other words, when an SU finds an idle band, the SU accesses this channel and as long as the band remains unoccupied by PUs, continues utilizing it. After detecting the presence of a PU in the band though, the SU vacates the band and continues to analyze other bands by spectrum sensing to be able to find new opportunities.

Cognitive radios (CR) [2–5] present an elegant solution to this inefficiency by realizing DSA [6, 7]. CRs reuse the spectrum that lies idle but is reserved for licensed users' exclusive use in today's spectrum management policy. A CR operates as an intelligent wireless communication system that is aware of its surrounding environment and adapts to the changing radio conditions to achieve both highly reliable communications and efficient utilization of the radio spectrum [8]. There are many research topics about CRs that have drawn tremendous interest such as spectrum sensing, sensing scheduling, medium access and resource allocation, energy efficiency (since these devices are battery constrained), spectrum sharing among CRs, and security. As these works improve the concept in terms of cognitive capability, there still remains many challenges that lie ahead for making CRs viable in practical networks.

As we have mentioned before, DSA regulations assert that CRs have to operate without harming the PU communications. Previous works [9, 10] showed that rather than local sensing in which a single CR senses the spectrum by itself, cooperative sensing, in which a number of CRs all sense, enhances the PU detection reliability. Sensing performance is typically measured in terms of *the probability of detection* and the *probability of false alarm*, the former representing the probability that an active PU is detected when the spectrum is indeed occupied by the PU, whereas the latter representing the probability that the CR reports a PU existence when the spectrum is actually free. It is a regular practice to satisfy certain probabilities of detection and false alarm in local and cooperative sensing for assuring the quality of protection against harming the PU communications.

Although cooperative sensing improves spectrum sensing, it assumes all CRs are willing to sense for each other. However, since a CR is supposed to be a battery-operated wireless device which needs to use its energy efficiently, or due to security concerns for example, such an altruistic operation mode may not always be desirable and realistic. Instead, just like the human societies, cooperation willingness of a CR towards another should be established by the social ties of these two parties. For example, a CR may not sense for a stranger whereas it would for a CR that is a friend. In this thesis, we model this cooperation willingness as a function of social relations between CRs. Each CR is considered with its user that has some social relations, e.g. friendship, community and selfishness. These social relations are used to quantify the cooperativeness of a CR from the perspective of the CR that requests cooperation for sensing. We model people's biased opinion on others, e.g. prejudices or sympathy on certain social groups and interests via the *community score*, whereas the modeling of interactions between people are simulated by the *cooperation score*. Other examples include the cooperation tendency modeling of users via the *selfishness* parameter and the *friendship score*, denoting the degree of sympathy between people that are friends or a friend of a friend.

1.1. Key Contributions

Our contribution in this thesis is twofold. First, we model a CRN in a realistic context in which operation of CRs mimics the real human societies, e.g., nodes with social ties are more eager to assist the other asking for help. Apart from its practical importance, such a model is also quite novel in the sense that it integrates two emerging research domains, social networks and cognitive radio communications. To our knowledge, associating CRs with social relations and using these relations for a better sensing have not been explored in the literature.

The contributions of this thesis can be listed as follows:

- *New CRN model:* We propose a CRN model in which the CRs are coupled with their users in the behavioral context, mimicking the social networks of today. In this new model, CRs, like the users in many social networks today, have friends, affiliations and personal interests. The familiarity formed between users due to common communities, common friends (friend-of-friend), community knowledge (communities that a friend is in) and their previous interactions are integrated into the cooperation decision logic of CRs, as well as the users' inherent characteristics like selfishness towards cooperation. In Chapter 2 more information on how related works have used some of the concepts we have integrated into the CRN to make it simulate social network behavior will be given.

- *Social-aware Cooperative Spectrum Sensing:* We propose a spectrum sensing approach which utilizes the socially operating CRN proposed by us and chooses cooperative spectrum sensing candidates according to various social properties, e.g community, friendship and interactions. There are many benefits to smart selection of cooperation candidates, first and foremost being the higher probability of initiating cooperation due to the generally mutual nature of sympathy between people, the latter being the higher probability of getting correct sensing result in cooperation due to the exact same reason. A socially oblivious scheme cannot predict or gather historical information like people do about social relations and therefore faces the risk of choosing cooperation candidates that either might not accept cooperation or send erroneous, malicious or poor sensing results.

All in all, our concerns in this thesis are to both introduce a realistic and new CRN model which we believe fits better into the next decade's networking paradigm and to show that performance improvements in many aspects are possible with a social-aware cooperative spectrum sensing scheme and candidate selection.

1.2. Thesis Outline

First, in Chapter 2 we review the related works in the literature in order to provide an overview to the current contributions in the technical literature and identify where our contribution with this thesis resides.

Our contributions that we have previously mentioned have been explained in the chapters Proposed Scheme (Chapter 3) and Performance Analysis (Chapter 4).

In Chapter 3 we explain the system model that has been proposed for the research carried out in the thesis. The chapter elaborates on how the CRN under consideration is constructed, with the definition of the social CR, channel properties, mobility, community properties and assignment of social properties throughout the system. Afterwards, the explanations for the social-aware and social-oblivious schemes, how they operate, and how the cooperation between CRs are induced in these schemes are given. Finally, we explain how a specific negative social characteristic, maliciousness of users, is modeled in the system.

Chapter 4 first introduces how the simulations are performed in the system to evaluate certain scenarios and the effects which these scenarios create on the system. The performance evaluation metrics are then listed and explained so that our criteria for quantifying the success of the social-aware scheme over the social-oblivious scheme in a social CRN can be comprehended. We then provide the simulation parameters and define the simulation environment through the meaning of these parameters while offering a discussion on how the selection of the parameters are made. We conclude this section with our findings in the simulations, categorizing them under three scenarios, namely the effect of cooperation tendency, maliciousness, cooperation threshold and additionally, some cases which demonstrate the CR behavior, interactions between certain CRs in the system. These findings are thoroughly discussed and explained.

We summarize the key contributions, discuss the results of our thesis and conclude with a brief list of future directions in Chapter 5.

2. BACKGROUND AND RELATED WORK

2.1. Cooperative Sensing

The main idea behind cooperative sensing is to exploit the spatially different sensing data coming from spatially distributed CR users and therefore improve the sensing performance [9]. Various works [11–14] have proposed cooperative sensing methods to take advantage of spatial diversity. A combined decision from cooperative sensing, achieved by the sharing of sensing information amongst CR users, is more accurate compared to a decision obtained by local sensing [15]. Through both numerical results and analytically, it has been shown that cooperative sensing provides considerably higher spectrum capacity gains than local sensing [16]. *Cooperative gain* is the terminology used to denote this spatial diversity-induced performance improvement.

The sharing of the sensing data between the CRs that participate in cooperative sensing can be classified into three categories: centralized, distributed, and relay-assisted. Centralized cooperative sensing [17–19] is controlled and organized by the Fusion Center (FC), an entity which assumes a central role in the network. It works in three steps:

- FC decides on the spectrum band on which the cooperating CRs will perform sensing and sends an individual sensing instruction to all the cooperators.
- CR users that participate in the cooperation send their sensing results back to the FC for evaluation.
- FC combines these spatially different local sensing information from the cooperating CRs, decides on the sensing outcome and sends the cooperative decision back to the cooperators.

The sensing operation by the participating CRs is performed on the *sensing channel*, the channel where each CR has a physical link with the PU. The individual sensing results are sent via the *control channel*, the channel where each CR has a physical link to the FC. Finally cooperative sensing data fusion result is sent via the *reporting channel*, where the physical link formation is the same with the control channel. An interesting fact is that centralized cooperative sensing, along with centralized networks, can occur also in distributed CRNs. In the centralized network, an FC is typically a Cognitive Base Station (CBS), whereas in a distributed network, a CR assumes the FC role and organizes cooperation between the cooperating CRs by collecting/distributing sensing/result data from/to cooperating neighbors.

Main difference of distributed cooperative sensing [20] from centralized sensing is that it does not feature an FC in the decision process. In distributed cooperative sensing, the cooperating CRs handle the communication between themselves by themselves and try to converge on an agreed decision via iterations [9]. It works as follows:

- CRs agree on sensing and send their own sensing data to the cooperating neighbors.
- Each CR combines its own data with the received data to reach a decision according to a local criterion.
- As long as the criterion is not satisfied and a unified decision cannot be made, the cycle repeats.

Since the convergence of the decision is required in this type of cooperative sensing, it might take several iterations to be able to reach the final sensing decision.

The third scheme, relay-assisted cooperative sensing [10, 21, 22], is proposed due to the realization that the sensing channel and report channel are not perfect. This type of cooperative sensing features a mutual cooperative relationship between a CR that has a weak sensing channel and a CR that has a weak report channel. With relay-assisted cooperative sensing, this problem can be alleviated and cooperative sensing performance can be improved. In our thesis, we have used centralized spectrum sensing with the third step of cooperation removed from the process. Our system model features CRs that request cooperation and act as the fusion center itself. The need for distributing the result to the other CRs does not exist since we have modeled the system so that the cooperation assisting CRs do not access the channel and therefore do not need the sensing result at that timeslot. This decision was made to simplify the system from spectrum sharing concerns and focus on social CRN analysis.

Cooperative sensing, along with the many realized benefits that we have explained, brings with it a challenging task of sharing and combining the sensing results from the measurements of each CR. The shared information can be soft or hard decisions made by each cognitive device [23]. While soft information combination between CRs outperforms hard information-combining method according to the results in [23, 24], hard-decisions are found to perform in the same level with soft decisions when the number of cooperators is high [25]. Due to simplicity reasons and control channel assumptions, we use hard decisions in our system.

When hard decisions are used as a method for information combination, four different kinds of methods, AND, Majority, M-out-of-N, and OR logic can be used to perform cooperative sensing decision in the fusion center [26]. The AND logic requires all of the sensing results to be H_1 to be able to cooperatively decide on H_1, H_1 being the alternate hypothesis that states the current band has an active PU in it. OR logic on the other hand requires only the sensing outcome of the fusion center and another CR to be H_1 to be able to decide on the alternate hypothesis. When a majority of the CRs decide on H_1, the Majority rule outputs H_1. Finally, the M-out-of-N rule is a generic cooperative decision logic which decides on H_1 when M CRs out of N cooperators also that $H = H_1$. Substituting $M = N$, $M = 1$ and $M = \lceil \frac{N}{2} \rceil$ form the AND, OR, and Majority rules. Various cooperative sensing techniques and challenges are studied in [27–30].

Another challenge in cooperative spectrum sensing is the imperfect reporting channel. In [30], this challenge and its effect on the sensing performance is analyzed for hard cooperative decision logic. This scheme allows only the most favorable secondary users with peak reporting channel conditions for cooperation. The work can ensure both the maximization of detection probability and maintaining the desired false alarm probability via this cooperation set selection criteria based on the reporting channel conditions. In our work, we assume perfect communication and reporting channel conditions due to both works like this being present to alleviate the problem on hard cooperative decision logic, and because our work is closely coupled with choosing the cooperating nodes according to cooperation performance, which indirectly means that the sensing result being wrongly propagated can lead to an elimination of a cooperation candidate.

Selection of cooperating CRs for cooperative sensing is a key factor in its performance. A selection scheme can be utilized to achieve a better cooperative gain and reduce the cooperative messaging overhead. An exemplary situation is investigated in [25], where it has been shown that selecting independent CR users for cooperation can improve the robustness of sensing results for CRs that experience correlated shadowing-induced problems. The removal of malicious users from the cooperating CRs is another important fact, enhancing the security and the reliability of the network, making cooperation candidate selection a major performance effector.

It can safely be said that user selection and cooperative sensing overhead terms are strongly related when thought along the lines of energy efficiency, security and control channel bandwidth issues. There is always a tradeoff between the detection performance and one type of overhead [9]. Most of the user selection schemes that have been proposed [25, 31] target at addressing one or two of these issues. As this issue remains a challenge in the field of cooperative sensing, we are also using a cooperative user selection scheme based on satisfying the social score success criteria and do not put an effort to particularly overcome cooperative sensing overhead in multiple branches.

In cooperative sensing, when CRs share local sensing information with the other cooperating CRs or the FC, a reporting delay occurs. This means that cooperative sensing schemes have cooperation overhead because such delay does not exist in local spectrum sensing. For instance, in [32,33], there are methods proposed to alleviate this problem. In our work, we do not focus on the reporting delay and assume that it does not cause any extra overhead in our system.

2.2. Social Network Analysis

Social network analysis (SNA) [34, 35] has attracted a significant attention in many research areas, e.g. anthropology, biology, computer science, engineering, economics and information science [36]. Relationships among social entities, the patterns and what these relationships imply are the main focus of SNA. Social characteristics of users have also recently gained tremendous attention from the research community as it provides exciting angles in the design of systems by understanding the implications of these relations or their practical use. As the popularity of online social networks increase and new information technologies emerge and develop (e.g. E-commerce, mobile computing, distributed systems), the need to study social relationships and ties arises. SNA hence becomes an important tool in this study and may pioneer and create a new design approach of new protocols, policies or applications in various information systems.

Probably the most relevant research area that makes use of these characteristics is *pocket switched networks* (PSN) [37], a delay-tolerant network which routes data from one "pocket" to another without any infrastructure support. Performance of data routing is directly dependent on the movement of the nodes, which are the users of the pocket devices, which is obviously not random and has some patterns [38] depending on the social characteristics of the associated user. PSN routing can substantially be improved by forwarding data to more relevant nodes with the knowledge of social characteristics. These social characteristics are *friendship* and *community*, which are named as *positive social characteristics* and *selfishness*, named as *negative social characteristics* [36]. Please note that these characteristics are usually long term and can at least be assumed to be less volatile than node mobility. This makes exploiting these social characteristics beneficial in the enhancement of *delay-tolerant networks* (DTNs) in general. With the recent advances in SNA, several social-based DTN routing methods [39–44] have been proposed which put these observations (utilizing social characteristics such as community and centrality) to use and assist the relay selections. Since the usage of social characteristics as friendship, community and centrality can assist or improve the conditions or performance in a network, we name them as positive social characteristics.

On the other hand, the entities that control the mobile nodes in DTNs are highly rational (e.g. people, organizations) and therefore aim to maximize their own performance and resources. These nodes show selfish behavior during operation to conserve their limited resources, since doing otherwise and carrying or forwarding packets to other nodes consume both their energy and require computational resources. With these types of nodes that shy away from acting as relays for others, a significant negative impact is made to the network routing since routing in DTNs is a cooperative activity in nature. To alleviate this selfishness problem, several social-based DTN routing protocols [45–50] have been proposed, which recognize and discourage this type of behavior by using punishment policies on non-cooperative nodes or providing rewards to cooperative nodes throughout the network. Since they impact network performance in a negative way these characteristics are called negative social characteristics.

We can conclude that friendship and community are positive considerations as they can be beneficial for a cooperation scenario whereas selfishness is not a desirable property in a cooperative task, e.g. ad hoc routing. We facilitate the usage of these characteristics to define and react to social behavior of users in our thesis.

The social structure of a society can be represented as a *social graph* which is considered the most popular way to study the social relations in between and extract the social characteristics of these people. A social graph is basically a mapping of people which shows how they are related, in which vertices correspond to individuals and links are the relations between two individuals, a.k.a social ties. These ties between people can be interpreted in many forms, e.g. friendship, family, co-worker. Social graphs have been used in [51] and [52] for the analysis of online social networks, and terrorist networks. Furthermore, various social metrics like communality and centrality can easily be interpreted from social graphs, enabling usage of these metrics in social based approaches and therefore making social graphs important for social-based approaches.

Although we have listed many benefits to the construction of a social graph for discovering social metrics (e.g community, friendship), these graphs are sometimes either not available or hard to obtain due to privacy or security reasons. To remedy this information accessibility problem, the interactions and interests of people over wireless networks can be observed. From these observations, another way to represent the social relations can be built. The *contact graph*, in which links represent the interactions among entities [36], is a common alternative to the social graph. Contact graphs enable analysis and estimation of social metrics between people in a network like the social graphs. As packet forwarding happens when two mobile nodes come into contact with each other in DTNs, it is possible to generate a contact graph by using this previous contact information between nodes. In such a DTN contact graph, a mobile node is represented by a vertex and an edge represents the previous meetings between nodes. From there, since the existence of an edge between nodes mean that there has been a past encounter between these nodes, it can also help in determining the probability of the future encounters, i.e. nodes with few and far apart past encounters have a low probability to meet anytime soon. There are two ways to construct a contact graph, either separately for each single time slot of the past or as an information-heavy parameter set where each edge records the time, frequency and duration of past encounters.

Exploiting the underlying structure of these two graphs are beneficial for gaining understanding of the network and designing an application using them. For instance, two nodes who are friends have higher probability to contact compared to two randomly chosen people with no friendship ties, which can be used in PSN routing from a source to a specific destination [53]. People with similar interests are more likely to befriend and they behave similarly, contact more frequently, which is referred to as *homophily* [54]. In our thesis, we use this homophily phenomenon to assign friendship ties between two users according to their personal interests, and then use these friendship and friend-of-a-friend relations to determine one node's criteria to select the cooperation set and for evaluating if a node should respond to the other's cooperation request.

Analysis of the social graph or contact graph can uncover the communities in the network. A community can refer to different phenomena in different contexts. It is an important concept in both ecology and sociology [55–57]. Ecology defines community as an assembly of two or more populations of different species occupying the same geographical area whereas sociology's definition portray community as a group of interacting people living in a common location. Several works [55–59] study the interactions between species/people in communities at many spatial and temporal scales [36]. In general, a community is a group of entities that have higher relationship inside the community (i.e., high density of edges within them) compared to the other members of the network (i.e., a lower density of edges between groups) [60, 61]. Therefore, communities naturally reflect social relationship among people. In our thesis, different from its common use, we consider community as a group membership information that affects people's opinions. For example, people can build some prejudice or sympathy towards the members of a certain community and this feeling affects people's cooperation tendency in a positive (sympathy) or negative (prejudice) way. This kind of opinion formation is quite realistic and sometimes beneficial when it is impossible to interact with each member of the community directly and learn through experience [62]. Communities which might have shared members, i.e., overlapping communities [63], can be unrevealed via community detection algorithms [64]. In our thesis, we assume that each user is affiliated with a single community and the user states its community itself.

In a cooperative system, it is important that all participants act in a trustworthy manner. Hence, for an efficient operation, the participants should assess and track the trustworthiness of the others. An entity is not necessarily untrusted because of its bad intents but also because of its operation performance. For instance, a wireless device might be malfunctioning because of a problem in its hardware. Trust is built basically through three mechanisms: experience, recommendation, and knowledge [65]. We use this overloaded term - trust - as a way to assess the sensing performance or cooperation tendency of a user from the perspective of the user requesting cooperation. In our model, users built trust only through their experiences, i.e, cooperative sensing requests and their outcomes. In a sense, this approach interprets contact graphs as previous cooperation information and act as the memory of each CR user about their previous contacts with the other members of the society.

To the best of our knowledge, the only work on social CRs is [66] which investigates the propagation of channel occupancy state information from one CR to another via recommendation. Recall that recommendation is one way to acquire trust information about the recommended entity. In the considered system, each CR announces its favorite PU channel which is subject to change due to the temporal changes in the channel quality, to its neighbors. Change in the CR's favorite channel is modeled as an epidemic propagation in social networks but with a modification of infection, based on the physical distance between the infector and the victim CR.

3. PROPOSED SCHEME: SOCIAL-AWARE COOPERATIVE SPECTRUM SENSING

3.1. System Model

We consider a system which is divided into equal sized grids each having N_g CRs inside. The CRs seek for transmission opportunities at a single primary user (PU) frequency. The primary user's transmission in a grid is modeled as a two state finite state machine, where the states are *busy* and *idle*, denoting activity and inactivity times of the PU respectively. We have provided an illustration of the PU channel model in Figure 3.1, where the probability value of transitions from idle to idle/busy, from busy to busy/idle states are given and $P_{idle} + P_{busy} = 1$. The idle state corresponds to the "ON" state for the CRs, meaning that they cannot access the channel, and the busy state corresponds to the "OFF" state, during which one of the CRs in a grid can transmit. Each sensing CR can locally detect an existing PU in its grid with a probability P_d whereas give a false alarm with probability P_f. The CRs, when they have traffic, request cooperation from other CRs, and decide on whether the channel is busy or idle. Each CR can ask for cooperation in spectrum sensing to the other CRs in its transmission region. In our model, a CR can communicate only with the CRs that are in the same grid. When the channel state is determined idle by a CR via local or cooperative sensing, only the cooperation requesting or locally sensing CR transmits during that timeslot, meaning that spectrum sharing is not modeled in the system.

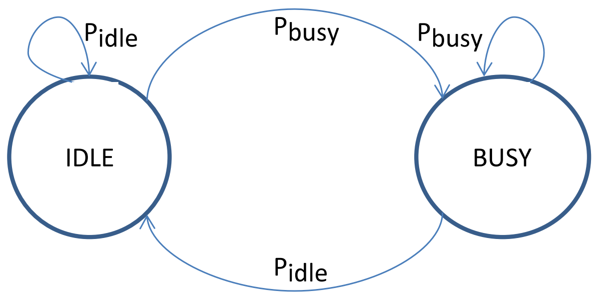

Figure 3.1. Primary user channel model.

Each grid in our system has a channel quality parameter γ_g with mean μ_g, variance σ_g and CRs in this grid also have a unique received SNR (γ_{CR}) with the mean as γ_g of the grid they are in and variance σ_{CR}. Therefore each CR in the grid have different received SNR for the link associated with them at the frequency of communication in the grid. Each CR calculates the minimum sensing time while sensing the channel in the grid to satisfy the probability of detection and false alarm requirements we have determined. The minimum sensing time is calculated as in Equation 3.1 [67]:

$$t_{CR} = \frac{\frac{1}{\gamma_{CR}^2}[\mathcal{Q}^{-1}(P_f) - \mathcal{Q}^{-1}(P_d)\sqrt{2\gamma_{CR}+1}]^2}{f_s} \tag{3.1}$$

where t_{CR} denotes the minimum sensing duration for a CR to sense the channel in the grid using energy detection for an AWGN channel under the given target performance parameters (P_d,P_f). In the equation, f_s is the sampling rate, γ_{CR} is the received SNR of the link associated with the CR in the grid and $\mathcal{Q}(\cdot)$ is the complementary distribution function of the standard Gaussian [67].

When modeling the system, since we are testing at the boundaries where the CRs will be forced to locally sense and be unable to cooperate under certain scenarios, we use static P_d and P_f values in determining the minimum time to sense. Therefore in cases where cooperative sensing occurs, although the cooperative sensing has a higher P_d and P_f value when this minimum sensing time is used, we are ensuring that the CRs individually satisfy the performance parameters by local sensing. For example, another approach to this would have been to keep the cooperative P_d same by decreasing the sensing time needed per CR, but we are refraining from this approach to ensure better opportunity detection.

In the system, the CRs are mobile and can move between grids randomly. More realistic mobility models are also possible. However, for keeping the system complexity controlled, we assume a uniform movement in all directions. This is in some way a disadvantage for our system since typical user movement is in actuality associated with social relations. In [44,53], nodes which contact each other frequently and have regular, long-lasting contacts are exploited in friendship determination, whereas in [54,68], it has been shown that members of the same communities are more likely to interact with each other than another randomly chosen community. We therefore could have exploited these findings by introducing a mobility model into our system where members of the same communities and friends interact more frequently with each other, resulting in better social-aware cooperation performance, but we have refrained from this because this would have made us unsure about where the benefit of social-aware cooperation lies. By keeping our mobility model simple, we both avoid introducing further complexity to our system and analyze the effects of social aware cooperative sensing on sensing performance without the aid of community member contact frequency. We believe that in real life, the results would be further improved because of the homophily phenomena and that our system covers an average-to-worst case scenario where members of the same communities do not interact as frequent as in the average real-life case.

A CR i denoted by CR_i is represented as a 5-tuple:

$$\mathrm{z} = < \mathrm{C}, \mathrm{L_f}, \mathrm{L_s}, \mathrm{L_c}, \beta > \tag{3.2}$$

where

- C is the *community* of this CR,
- $\mathrm{L_f}$ is the *friend list* representing the relationship of this CR to the others,
- $\mathrm{L_s}$ is the *sympathy list* representing the willingness of this CR to cooperate with a member of a specific community c_j,
- $\mathrm{L_c}$ is the *cooperation score* list showing the performance of each CR in previous cooperation requests initiated by this CR,
- β is the *cooperation tendency*[1] of this CR.

[1]We use cooperation tendency and selfishness interchangeably. The relation between these two terms is *cooperation tendency = 1 - selfishness*.

In our model, a CR refers to a wireless device with the social context of its user, e.g., social group information, personal interests. Each CR is a member of a community out of N_c communities. A community membership can be determined by various ways, such as interests, e.g., group of people that are fan of a certain sports club, or an affiliation, e.g., graduates of a university. Community membership also determines the friendship ties among users. Hence, in our model, we construct friendship list $\mathrm{L_f}$ of a CR such that CRs with similar interests are more likely to be friends due to the homophily phenomenon [60]. However, as users with totally different interests can also befriend, our model considers this fact in friendship assignment. CRs exchange this community information upon encounters, i.e., both are in the same grid.

In our system, for the sake of simplicity, community membership is modeled with education information and personal interests of users. We have determined N_{pi} different personal interest values and N_e different education information values, which, when assigned to CRs, denote their social group information. Although we are keeping the model simplistic by limiting the social group information to these two aspects of human social life, in order to add more intersection chances for communities, we also have introduced the similar interest concept into the frame. Out of N_{pi} different personal interests, each N_s personal interest is considered similar, for example if interest i is windsurfing, interest $i + k, k < N_s$ is kite-surfing. This adds a certain degree of extension to community coverage as it is in real life, people of similar music interests, though not differing in genre, can interact in the same concert but entirely different interests do not.

Our system assigns the education and personal interest values randomly to each CR. After forming the social group information for all CRs, the befriendment algorithm in the system tries to make friends from people with the same education and personal interest values, prioritizing education over interests and those over side-interests. Since the system has many different types of users with differing social group information, the remaining of the users in the system are randomly chosen as friends. This process is performed as follows:

- Iterate over the list of CRs, forming two hashmaps, with their keys as e and pi, values as "CRs sharing these e and pi values".
- Iterate over both the education hashmap and interest hashmap to match the CRs with the same education and interest values and befriend them with each other.
- Remove each befriended CR from the lists.
- Iterate over both the education hashmap and interest hashmap to match the CRs with the same education and similar interest values and befriend them with each other.
- Remove each befriended CR from the lists.
- Iterate over the education hashmap to befriend CRs with same education values with each other.
- Remove each befriended CR from the education list.
- Iterate over the personal interests hashmap to befriend CRs with same personal interest values with each other.
- Remove each befriended CR from the personal interest list.
- Iterate over the personal interests hashmap to befriend CRs with similar personal interest values.
- Remove each befriended CR from the personal interest list.
- Randomly assign the remaining CRs in both lists as friends.

It is shown that some social factors, such as political opinions, can help two strangers build trust between each other and cooperate. Taking this fact, we associate a sympathy list $\mathrm{L_s}$ with each CR to represent this CR's willingness to sense for another CR that is requesting cooperation. The sympathy value of CR_i for CR_j is determined by community and personal interests of these two CRs. At this point, it would be appropriate to clarify how the sympathy lists are initialized and some additional community concepts that are in our system. In initializing the sympathy lists we use the algorithm in Figure 3.1.

We make use of side interests, interests of friends and the education information of friends by assigning a sympathy value that is lower than the main (interest/education of self) value. This stems from the fact that similar interests are also attractive to

Require the knowledge of e_{main}, e_{friend}, $e_{default}$ and pi_{main}, pi_{friend}, pi_{side}, $pi_{default}$.

for all CR_i in System CRs **do**

  for all e in N_e education values in the system **do**

    if $e(CR_i) = e$ **then**

      $L_s{}^i(e) \leftarrow e_{main}$

    else if e is one of the friends' education of CR_i **then**

      $L_s{}^i(e) \leftarrow e_{friend}$

    else

      $L_s{}^i(e) \leftarrow e_{default}$

    end if

  end for

  for all pi in N_{pi} personal interest values in the system **do**

    if $pi(CR_i) = pi$ **then**

      $L_s{}^i(pi) \leftarrow pi_{main}$

    else if pi is one of the friends' interest of CR_i **then**

      $L_s{}^i(pi) \leftarrow pi_{friend}$

    else if $\lfloor \frac{pi(CR_i)}{N_s} \rfloor = \lfloor \frac{pi}{N_s} \rfloor$ **then**

      $L_s{}^i(pi) \leftarrow pi_{side}$

    else

      $L_s{}^i(pi) \leftarrow pi_{default}$

    end if

  end for

end for

Figure 3.2. Initializing $\mathrm{L_s}$ by using the community information of CRs.

the person and this person has some knowledge, and therefore trust to the interests or education information of a friend. This sympathy is higher when compared to an entirely unknown (default) education origin or personal interest.

In our system, while the friend list is static for the considered time period, in real life, as we continue our lives, we make and end friendships. However as this would increase complexity even further, we have refrained from dynamic friend lists. On the other hand, the sympathy list of CRs are dynamic. The L_s of a CR changes depending on the successful or failed cooperations with the CR under investigation. Thus, when a CR_j fails to cooperate with CR_i, the value of CR_j's personal interest and education information falls in ${L_s}^i$, and when the cooperation is successful, the values rise. Cooperation with a person therefore raises or degrades the value of the whole community that the cooperating CR is in.

Cooperation score list L_c acts as the memory of a CR by recording the results of previous cooperations. Since CRs are mobile, they can meet each other multiple times during the considered time period. L_c records both the summary of previous cooperative sensing performances (${L_c}^p$) for each CR as well as the most recent one (${L_c}^r$).

Finally, we model the selfishness of a CR via our cooperation tendency parameter β. What the individual selfishness value of a CR denotes for our system is that it is the most dominant parameter in what a CR's response to a sensing request will be. Although the social selfishness value, i.e., the friendship ties in our model also affect whether a CR will cooperate or not, this can only alter the original cooperation tendency inherent in a user. A highly selfish person may only cooperate with friends whereas a less selfish person may cooperate with people only sharing a side interest, which is the term we use for interests that are similar to the users' own interest, e.g the user likes wind-surfing and the cooperation candidate likes kite-surfing. Other aspects of the network that the selfishness parameter models are cases where a CR with low battery or high traffic load may reject cooperation requests by other CRs. All in all, in our model a CR adjust its selfishness value depending on the ties between itself and the CR requesting cooperation.

3.2. Social-Aware Cooperative Spectrum Sensing (SAC)

In this section, we will introduce Social-aware cooperative spectrum sensing (SAC). SAC operates in three steps: (1) cooperation set selection, (2) cooperative sensing, and (3) updating the score of the CRs in this cooperation set.

Step 1: Cooperation set selection When CR_i needs to transmit, first it must determine which CRs will be more eager to perform spectrum sensing for it, and in case a CR senses what its sensing accuracy will be. For example, some CRs may be willing to sense, but their performance might be low for some reasons, say low-performing hardware. Similarly, a CR with very high sensing performance should not be asked for sensing, if the social ties between these two CRs are not very strong. In short, cooperation set selection algorithm must consider these two aspects simultaneously. SAC determines set of collaborators of CR_i using the following algorithm:

- The CR calculates the friend score of the CRs in the same grid using $\mathrm{L_f}$. The score of CR_j denoted by $s_f(j)$ can be either 1 (friend), 0 (no friendship tie), and 0.5 (friend of a friend).
- The CR calculates the community score ($s_c(j) \in [0,1]$) of the other CRs using $\mathrm{L_s}$. This calculation is performed as follows:

$$s_c(j) = \alpha {L_s}^e(e_j) + (1-\alpha){L_s}^{pi}(pi_j) \tag{3.3}$$

  where e_j is the education information of CR_j, pi_j is the personal interest information of CR_j and $\alpha = 0.6$ to prioritize education information over personal interest sympathy in community scoring.
- In order to assess the sensing performance of the neighboring CRs, CR_i uses the cooperation score of $\mathrm{CR_j}$. Cooperation score $s_l(j)$ is calculated by weighted summation of $L_c^p(j)$ and $L_c^r(j)$. $L_c^p(j)$ accounts for the times that this CR has made an accurate decision. For example, if the CR has sensed the spectrum m

times, and out of these only r were accurate, then the $L_c^p(j)$ score is r/m. We calculate $s_l(j)$ as follows:

$$s_l(j) = \alpha L_c^p(j) + (1-\alpha) L_c^r(j) \tag{3.4}$$

where $\alpha = 0.6$ to prioritize all of the cooperation history over the most recent contact.

- The total score of CR_j is calculated as follows:

$$s(j) = \alpha_f s_f(j) + \alpha_c s_c(j) + \alpha_l s_l(j). \tag{3.5}$$

where α_f, α_c, and α_l, all in $[0, 1]$ and satisfying the equality $\alpha_f + \alpha_c + \alpha_l = 1$, represent the effect of friendship, community, and sensing performance of CR_j in assessing its attractiveness as a cooperating node from the perspective of CR_i. In order to avoid being very conservative against non-friends, CR_i calculates $s(j)$ by setting $\alpha_f = 0$ when they are not friends. The friendship scenario is therefore an added bonus to the total score and non-friends can also give a cooperation chance to each other. On the other hand, in order to avoid minor sensing errors in the beginning to affect cooperation in the long run, we have devised a gradually increasing α_l to simulate a learning period between CRs. For a CR_j that has not yet cooperated a certain number of times (${n_{coop}}^{min}$), CR_i sets $\alpha_l < \alpha_f$ and $\alpha_l < \alpha_c$. The α_l value increases from zero to its original value as the number of cooperations (n_{coop}) increases. Both of these special cases are necessary for avoiding improper low scoring of some CRs without giving them enough chance to cooperate. The algorithm used to determine the final score from the friend, community and cooperation scores is given in the algorithm in Figure 3.2.

- The CRs with scores that are above a certain cooperation threshold λ_{coop} are added to the cooperation candidate set. Among these candidates, N_{coop} with the highest scores are selected as the cooperating CRs for this requesting CR. A cooperation request is then sent to these CRs.

Require The knowledge of α_f, $s_f(j)$, α_c, $s_c(j)$, α_l, $s_l(j)$, n_{coop} and ${n_{coop}}^{min}$.

if $s_f(j) = 0$ **then**

  if $n_{coop} = 0$ **then**

    $\alpha_c \leftarrow 1$

    $\alpha_l \leftarrow 0$

    $\alpha_f \leftarrow 0$

  else if $n_{coop} < {n_{coop}}^{min}$ **then**

    $\alpha_l \leftarrow \alpha_l \cdot \frac{n_{coop}}{{n_{coop}}^{min}} \cdot \frac{1}{1-\alpha_f}$

    $\alpha_c \leftarrow (\alpha_c + \alpha_l \cdot (1 - \frac{n_{coop}}{{n_{coop}}^{min}})) \cdot \frac{1}{1-\alpha_f}$

    $\alpha_f \leftarrow 0$

  else

    $\alpha_l \leftarrow \alpha_l \cdot \frac{1}{1-\alpha_f}$

    $\alpha_c \leftarrow \alpha_c \cdot \frac{1}{1-\alpha_f}$

    $\alpha_f \leftarrow 0$

  end if

else if $n_{coop} < {n_{coop}}^{min}$ **then**

  $\alpha_l \leftarrow \alpha_l \cdot \frac{n_{coop}}{{n_{coop}}^{min}}$

  $\alpha_c \leftarrow \alpha_c + \alpha_l \cdot \frac{1}{2}(1 - \frac{n_{coop}}{{n_{coop}}^{min}})$

  $\alpha_f \leftarrow \alpha_f + \alpha_l \cdot \frac{1}{2}(1 - \frac{n_{coop}}{{n_{coop}}^{min}})$

end if

Calculate final score $s(j)$ as in Equation 3.5.

Figure 3.3. Final score s(j) calculation.

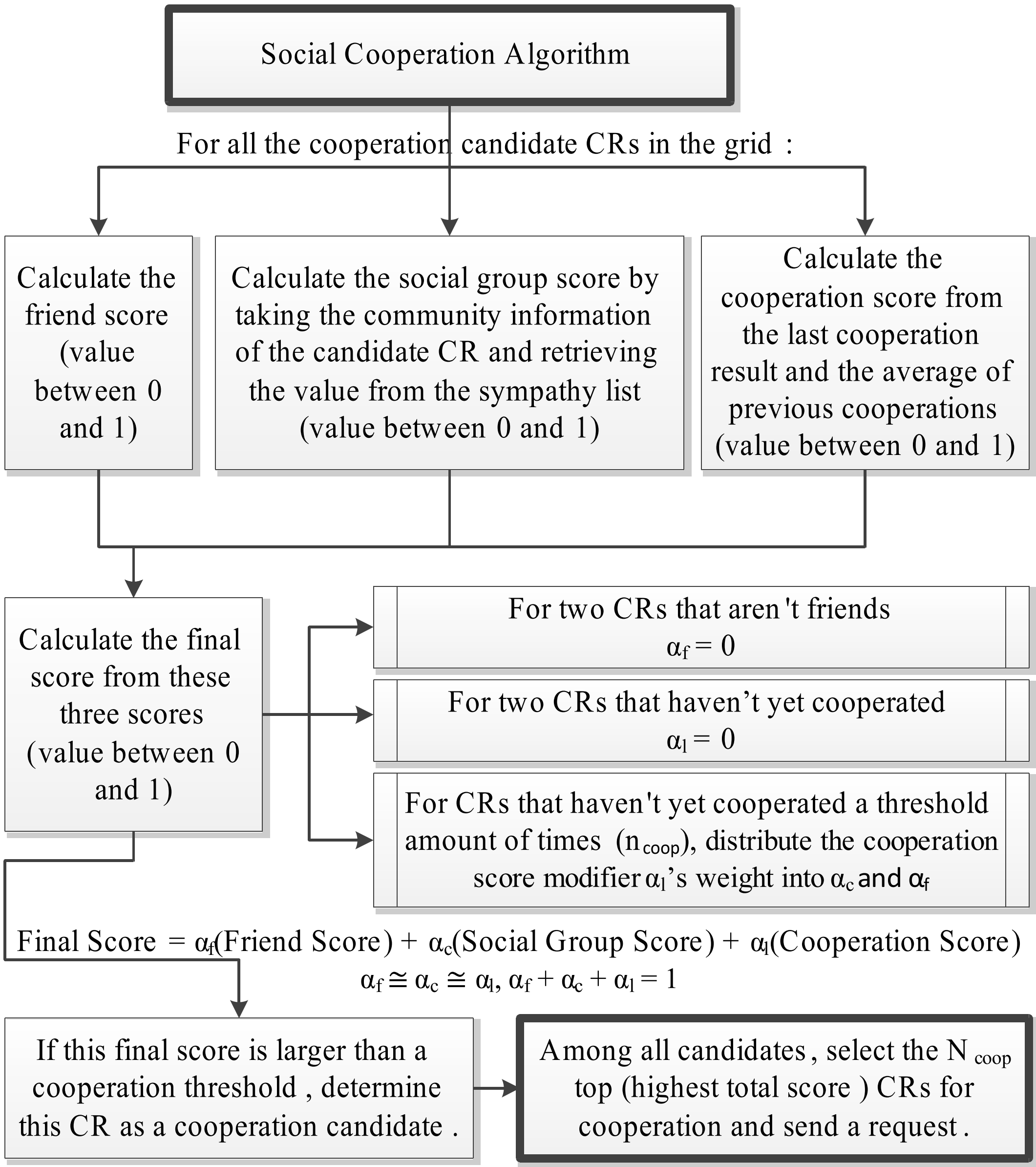

Figure 3.4. Flow chart for illustrating a basic summary of cooperation candidate selection.

Step 2: Cooperative Spectrum Sensing When CR_i requests cooperation from CR_j, CR_j decides whether to cooperate or not by a scoring system as in Equation 3.5 that additionally has selfishness incorporated into it. This mode of operation reflects the fact that people's selfishness is not a context-independent property, but changes depending on the other end of the communication. In this sensing scenario, if CR_j is a reputable agent, i.e. high score, then CR_j although having some natural selfishness, may increase its cooperation tendency. This tendency (new selfishness value) is represented as the probability of sensing for CR_i and denoted by $p_s(i)$. This probability is calculated as shown in the algorithm in Figure 3.2. Then, CR_j senses the spectrum with probability $p_s(i)$.

Require The knowledge of $s(j)$, cooperation threshold (λ_{coop}), effect of score on selfishness (α_{ess}) and β.

M_s is the modifier of selfishness.

if $s(j) - \lambda_{coop}$ is negative **then**

  $M_s = \frac{s(j)-\lambda_{coop}}{\lambda_{coop}}$

else

  $M_s = \frac{s(j)-\lambda_{coop}}{1-\lambda_{coop}}$

end if

Now we have a M_s value either in $[-1, 0]$ or $[0, 1]$ range.

$p_s(i) = \beta + \alpha_{ess} \cdot M_s$

if $p_s(i) > 1$ **then**

  $p_s(i) \leftarrow 1$

else if $p_s(i) < 0$ **then**

  $p_s(i) \leftarrow 0$

end if

Figure 3.5. Determining $p_s(i)$ from the natural cooperation tendency (β) of a CR.

A CR that refuses to cooperate sends a reject message to the requesting CR, whereas a cooperating CR starts sensing the channel and transmits the outcome of its sensing. When the CR receives a reject message or a sensing value result from other CRs, it decides on its transmission action via the majority logic as follows, where $K \leq N_{coop}$ is the number of cooperating CRs, $H(i)$ is sensing outcome of CR_i, and H is the final decision on the spectrum occupancy state:

$$H = \begin{cases} 1, & \text{if } \sum_{i=1}^{K} H(i) \geq \frac{K}{2} \\ 0, & \text{otherwise.} \end{cases}$$

Of course if the cooperation requesting CR receives a reject message from all the other CRs due to poor candidate selection, then the CR performs local sensing and decides on the spectrum occupancy state as follows:

$$H = \begin{cases} 1, & \text{if channel is busy} \\ 0, & \text{otherwise.} \end{cases}$$

If $H = 0$, CR_i initiates the transmission in the PU band. If $H = 1$, as the CRs conclude the existence of a PU, CR_i remains silent in the current time slot. Notice that our decision approach is conservative since it uses a greater than or equal to comparison for $H = 1$, meaning that for even number of cooperators, half of them deciding on a busy channel will result in a "remain silent" decision by the CRs.

Step 3: Updating the CR specific scores and community scores Let H_0 represent the real state of the PU channel and yield value 1 if the channel is occupied and 0, otherwise. For doing community and cooperation score updates, the CR that has requested cooperation from other CRs use the decisions those CRs have given (H=0, H=1 or rejected cooperation) to perform this operation. In case of spectrum access, e.g., $H = 0$ in Step 2, CR_i uses the ground truth about the spectrum: successful transmission (i.e., $H_0 = 0$) or a collision with the PU (i.e., $H_0 = 1$). On the other hand, if $H = 1$ in Step 2 and CR_i prefers to wait for the next time slot, the CR cannot know the ground truth. In the former, CR updates its last contact value and cooperation score for each CR ($L_c(j)$) that has cooperated according to the result of spectrum sensing. It then updates the sympathy list $\mathrm{L_s}$ by raising the social scores for corresponding communities of the cooperating CRs. If the sensing result was wrong, it updates these scores by decreasing them. In the latter, CR_i improves the score of the CRs that give the same decision as the cooperative outcome H, while degrading the scores of CRs that have conflicted with the majority. Please note that this approach assumes that there are less malicious users in the system, hence the majority represents an estimate of the ground truth. The sympathy list is updated as in the algorithm shown in Figure 3.2. As it can be seen from the algorithm, the education and personal interests of the user that is being cooperated with are individually updated according to the reject message or correct/wrong sensing results. The α_{reject} and α_{error} values are negative and degrade the sympathy, with $|\alpha_{reject}| < |\alpha_{error}|$ since an error is more costly for cooperation, whereas $\alpha_{correct}$ is positive and satisfies $|\alpha_{correct}| < |\alpha_{error}|$ since an error is perceived more punishing than the reward of a correct sensing outcome.

Require The knowledge of α_{reject}, α_{error} and $\alpha_{correct}$.
e is the education information of CR_j
pi is the personal interest value of CR_j
if CR_j has rejected CR_i **then**
  $L_s{}^i(e) = L_s{}^i(e) \cdot (1 + \alpha_{reject})$
  $L_s{}^i(pi) = L_s{}^i(pi) \cdot (1 + \alpha_{reject})$
else if CR_j has made an error in sensing for CR_i **then**
  $L_s{}^i(e) = L_s{}^i(e) \cdot (1 + \alpha_{error})$
  $L_s{}^i(pi) = L_s{}^i(pi) \cdot (1 + \alpha_{error})$
else
  $L_s{}^i(e) = L_s{}^i(e) \cdot (1 + \alpha_{correct})$
  $L_s{}^i(pi) = L_s{}^i(pi) \cdot (1 + \alpha_{correct})$
end if
if $L_s{}^i(e) > 1$ **then**
  $L_s{}^i(e) \leftarrow 1$
else if $L_s{}^i(e) < 0$ **then**
  $L_s{}^i(e) \leftarrow 0$
end if
if $L_s{}^i(pi) > 1$ **then**
  $L_s{}^i(pi) \leftarrow 1$
else if $L_s{}^i(pi) < 0$ **then**
  $L_s{}^i(pi) \leftarrow 0$
end if

Figure 3.6. Determining $L_s(j)$ from the latest cooperation with CR_j.

Finally, the cooperation score is updated as in (3.6, 3.7):

$$L_c^r(j) = \begin{cases} 1_{[H(i)=H]}, & \text{if } H = 1 \\ 1_{[H(i)=0]}, & \text{if } H = 0,\ H_0 = 0 \\ 1_{[H(i)=1]}, & \text{if } H = 0,\ H_0 = 1 \end{cases} \tag{3.6}$$

$$L_c^p(j) = \begin{cases} \dfrac{L_c^p(j) \cdot n_{coop} + 1_{[H(i)=H]}}{n_{coop} + 1}, & \text{if } H = 1 \\ \dfrac{L_c^p(j) \cdot n_{coop} + 1_{[H(i)=0]}}{n_{coop} + 1}, & \text{if } H = 0,\ H_0 = 0 \\ \dfrac{L_c^p(j) \cdot n_{coop} + 1_{[H(i)=1]}}{n_{coop} + 1} & \text{if } H = 0,\ H_0 = 1 \end{cases} \tag{3.7}$$

where $1_{[X]}$ is the indicator function taking value 1 if boolean expression X is true, 0 otherwise, and n_{coop} being the number of cooperation attempts between CR_i and CR_j. Thus, the most recent cooperation between requester CR_i and helper CR_j is simply recorded in $L_c^r(j)$ as 1 (successful sensing result) or 0 (sensing error). Then, this result is used to update the previous cooperation average value ($L_c^p(j)$) by adding the new result and taking the average over all cooperations.

There is one additional case in the calculation of $L_c^p(j)$, for when the CR_j rejects the cooperation request of CR_i. In this case, the n_{coop} is incremented to lower the average score. Hence, the cooperation is treated as a failure in terms of the previous cooperations average but the $L_c^r(j)$ remains the same. This is done to lower the severity of score punishment on CRs that do not harm the cooperation requesting CR by a sensing error.

The cooperation request replying and sympathy list updating processes have been briefly explained in a flow chart in Figure 3.7.

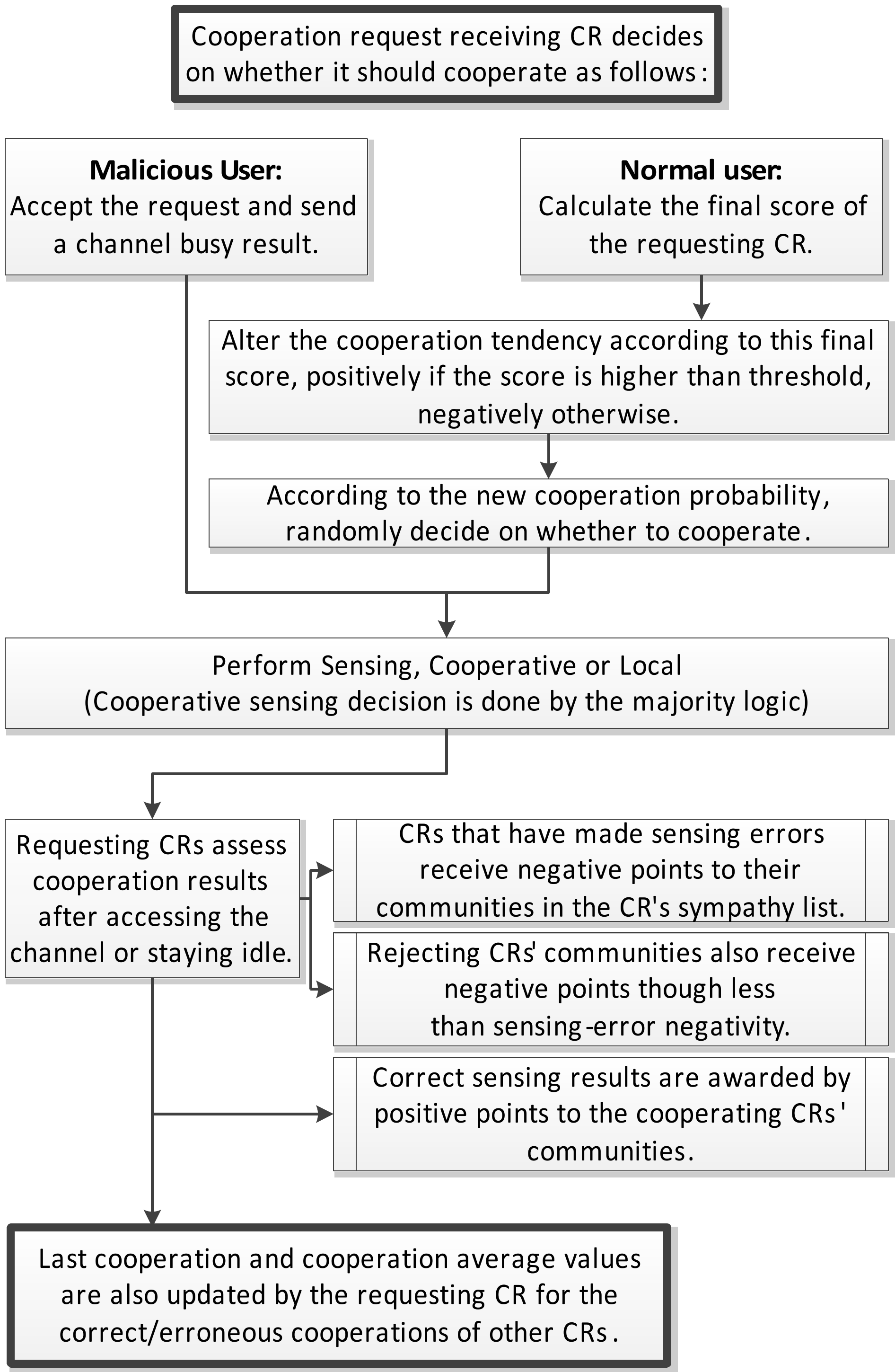

Figure 3.7. Flow chart for illustrating a basic summary of steps two and three of SAC.

3.3. Social-Unaware Cooperative Spectrum Sensing (RAND)

In this section we will introduce the *Social-unaware cooperative spectrum sensing scheme* (RAND), which represents the class of CR cooperative sensing methods present in the current literature to our knowledge, sensing scheme without any social ties or trust relationships embedded.

In RAND, there are two steps of operation. A CR simply selects its cooperation set randomly in Step 1, meaning that there is no calculation of friend, community scores etc. in the decision process. In SAC, step two had a social tie related selfishness alteration, where a CR could behave less selfish to a CR that is perceived reputable. In RAND, the selfishness is the sole determining factor for deciding whether a CR will respond positively to a cooperation request. Step 3 does not exist in RAND, since the CRs do not keep sympathy lists or previous cooperation history. In other words, a CR does not keep track of the cooperation willingness or trustworthiness of other CRs, facing the probability of choosing low-performing CRs over and over.

3.4. Malicious Users

To achieve a realistic model in which some users may exploit cooperative sensing benefits and give nothing in return to the overall network, we have created a malicious user model for the system.

A typical malicious user also has the characteristics of a CR (social group information, sympathy lists etc.) but behaves so that it accepts all requests but always responds with a busy PU channel reply to be able to transmit itself. This is a fairly simple model and we believe that more complex and advantageous malicious user models are also possible. However for keeping both the system complexity controlled and to be able to easily discern the malicious user behavior from simulation results, we have formed this approach, which can take advantage of social oblivious schemes.

In the system, we assign a malicious user density parameter d_m which denotes the fraction of the total CRs will behave as malicious users. When working with the SAC model, the CRs in the system behave according to social ties when both requesting and receiving cooperation. The malicious users keep the socially deciding request behavior from SAC, but they behave maliciously in accepting requests as previously described. Correspondingly in RAND, the malicious users choose their cooperation candidates randomly, the same way as a typical RAND CR does. We therefore keep the opportunity discovery efforts of malicious CRs in line with the other CRs in the system but also enable them to exploit the other users by giving them busy channel replies without spending any sensing time at all.

4. PERFORMANCE ANALYSIS

In this chapter, we show the performance improvement achieved by SAC compared to the "social-oblivious" scheme, RAND. We use several simulation scenarios and metrics for evaluating the performance of the two schemes. In addition to highlighting performance improvements, we also show results that support system correctness and show how the system in general and CRs in specific behave under different social conditions.

4.1. Simulations

In our simulations to evaluate the system, we have used 3 main scenarios to test the effects of these simulation conditions on system behavior and performance.

Scenario 1: Effect of increasing the cooperation tendency of CRs In this scenario, we increase the cooperation tendency from a very low value (selfish users) to a very high value (cooperative users) and inspect its effects on the system. We have used fixed λ_{coop} and d_m values for this scenario, using two different d_m values to be able to observe how the system behaves under a low malicious user population (low d_m), which is the usual case, and under a high d_m value, which is unusual to see in a realistic and working system, but is useful to consider as a security case.

Scenario 2: Effect of increasing the malicious user density in the system In this scenario, we increase the number of malicious users from none to a high value, keeping the system operational but at the same time affected by the malicious user population and inspect the effects this has on the system. We keep the λ_{coop} and β parameters fixed for this scenario, with the β being high to keep a realistic system state where the users are not both mostly selfish and malicious, focusing only on the effects of maliciousness on the system.

Scenario 3: Effect of increasing the cooperation threshold for candidate selection *in the system* In the simulations, we have increased the λ_{coop} for the cooperation set decision logic (step 1 of SAC) and observed how SAC behaves compared to RAND under a strict vs. loose selection of cooperating CRs. For this scenario we have both used fixed β and d_m values to inspect the system under realistic conditions, and used different β and d_m values to compare the ways SAC react to the change of λ_{coop} under different system states.

Here are the metrics that we have used to evaluate the system, and their succinct explanations as to what they are or measure:

- *Opportunity discovery ratio* is the ratio of the number of idle channels discovered to the total number of idle channels in the whole system. This metric is used to show the opportunity discovery success of the cooperations and local sensing done by users, and the overall opportunity discovery success rate.
- *Rejected request ratio* is the fraction of cooperation requests that are rejected by the CRs that were asked for cooperation.
- *Cooperation request returns* is the amount of requests that are accepted or rejected by the CRs that were asked for cooperation.
- *Probability of malicious user cooperation* shows the probability that a request will be sent to malicious CRs among all CRs.
- *Malicious user cooperation* is the amount of malicious users that have been co-operated with, or in other words, the amount of requests that have been sent to malicious users, since they accept every request.
- *Opportunity discovery number* is the quantified version of the opportunity discovery ratio. This metric is used to discern individual success of cooperations and local sensing (provided in discovery ratio metric) from the amount of opportunities discovered via cooperative versus local sensing. It also enables comparing the total opportunity discovery amounts of SAC and RAND.
- *Missed opportunity number* is the amount of missed opportunities, when a CR has given a false alarm and did not access the channel.
- *Sensing time spent* denotes the total sensing time spent on cooperative sensing

and local sensing separately. Note that malicious users, when cooperated with, don't contribute to the sensing time spent. The individual sensing times spent on sensing is calculated with Eq. 3.1 and added up in cooperative sensing.

- *Collisions with the PU due to local and cooperative sensing* is the percentage of collisions from the channel accesses made by CRs due to cooperative sensing or local sensing decision of the channel being idle.
- *Cooperation request per successful cooperative discovery* is the amount of requests sent by CRs to others divided by the amount of successful discoveries resulting from a cooperation. This excludes the opportunity discoveries by local sensing. A high ratio denotes that the cooperation requests have not been planned carefully whereas a low ratio means that the cooperation requests have been used to create successful opportunity discoveries.
- *Cooperation request per successful discovery* is similar to the previous metric, with the denominator denoting all of the discoveries made by the system. This metric denotes the overall success of the system, with the rejected cooperations not affecting the system as sensing cooperation requests to malicious users.
- *Malicious user cooperation R request per successful discovery* is the ratio of the amount of cooperation requests made by malicious users to the opportunity discoveries realized by them. This metric denotes the success of malicious users, with a low ratio meaning that they have exploited the system resources effectively.
- *Malicious cooperative opportunity discovery* is the opportunity discovery metric for malicious users in the system, denoting their success in accessing the spectrum.
- *Cooperative sensing errors* denotes the amount of erroneous sensing results in the system. It is used to show the sum of misdetections and false alarms that have occured. The results for errors are separated for cooperative and local sensing.

4.2. Simulation Environment and Parameters

We have compiled the simulation parameters used in the system in Table 4.1.

For the simulation scenarios that we have explained in Section 4.1, we have used an array of increasing d_m, β and λ_{coop} values. These values are provided below:

Table 4.1. Simulation Parameters.

Parameter(s)	**Value(s)**	**Parameter(s)**	**Value(s)**
d_m **(low)**	0.15	d_m **(moderate)**	0.5
d_m **(high)**	0.7	$e_{default}$	0.5
e_{friend}	0.75	e_{main}	1
f_s	1000	N_{coop}	3
n_{coop}	4	N_{CR}	100
N_e	20	N_g	20
N_{pi}	60	N_s	3
N_{sim}	20	N_T	10000
P_d	0.7	P_f	0.1
P_{idle}	$U(0.6, 0.7)$	$pi_{default}$	0.5
pi_{friend}	0.75	pi_{main}	1
pi_{side}	0.75	s_e	0.6
s_{friend}	1	s_{fof}	0.5
s_{pi}	0.4	α	0.6
α_c	0.3	α_{ess}	0.2
α_f	0.3	α_l	0.4
$\alpha_{correct}$	0.04	α_{error}	-0.06
α_{reject}	-0.04	β **(low)**	0.3
β **(moderate)**	0.5	β **(high)**	0.7
λ_{coop}	0.45	μ_g	$5dB$
σ_{CR}	$4dB$	σ_g	$1dB$

$d_m = \{0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6\}$
$\beta = \{0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0\}$
$\lambda_{coop} = \{0, 0.1, 0.2, 0.3, 0.4, 0.45, 0.475, 0.5, 0.525, 0.55, 0.6, 0.7, 0.8, 0.9, 1.0\}$

In each scenario, while an array is being used to simulate the effects of one parameter on the system, the others are being fixed at a value that fixes the other aspects of the system state. For example, in a simulation for the effect of cooperation threshold, a low cooperation tendency (β (low)) and a low malicious user density (d_m (low)) environment can be selected to inspect the effects of λ_{coop} in a selfish network. The parenthesis indicators near d_m and β (low, moderate, high) denote the scenario specific cooperation tendency and maliciousness used in simulations. In Section 4.3, when we are evaluating the results, we will denote which parameters are being used by either giving the parameter names or stating the network condition (e.g. low cooperation tendency, high maliciousness).

There are some parameters in the tables that need explanation as to why they were chosen as they are. In this thesis, first of all, our objective was to show that, a CRN with social properties embedded, its users with social context, its cooperative sensing decisions coupled with a social-aware algorithm, will be overperforming a social-oblivious scheme and will fit better into the next decade's networking paradigm. Since the objective was not maximizing throughput, minimizing energy costs, sensing time or some other sensitivity-demanding metric, we have picked uniform distributions, low standard deviations and didn't particularly optimize any parameter to a 0.001% sensitivity with thousands of Monte-Carlo simulations. Certain decisions on parameters like the λ_{coop} being equal to 0.45 is given via the results of simulations, but the choice of α_{friend} being 0.3 over 0.29, since the results do not particularly favor one over another in every scenario, is just made with this optimization requirement not being present in the work. As it can be seen, the system is a complex system with many parameters that can impact the system optimization. Thus, to keep the system complexity controlled, to analyze the particular effects of the simulation scenario parameters stated in Section 4.1 and to be able to show that a social-aware scheme outperforms a social-unaware one in a network without the altruistic cooperation-ready CRs or non-exploiting users, we have kept the parameter decisions simple.

There is one additional note we would like to make on the choice of the number of timeslots in a single simulation run (N_T) being 10000. During the simulations, we have observed that, when the simulations last 10000 timeslots, a CR meets and contacts another CR an average of six to eight times and contacts a friend CR an average of 50 to 60 times, with the number of contacts with other CRs ranged from zero to roughly 40 to 100 times. We found these values reasonable in a network of 20 grids and 100 CRs, in terms of CRs generating some social awareness of other CRs and communities towards the end of the simulation, and avoiding the community knowledge saturation. For example, when we increased the number of timeslots to 120000, the network started to converge on the cooperative sensing decisions given, with almost each CR knowing who the most valuable CR is and who not to send a request to. This condition we believe would not happen in a realistic scenario, because it is a rough equivalent of having contacted and gathered information about all the people using social networks (e.g. 1.2 billion facebook users). Thus, we have decided on the 10000 timeslot per simulation parameter.

For the simulations we have performed, we have developed a discrete event simulator in java. The event simulator is used in producing the results and recording of the performance metrics, as well as handling multiple repetitions for better data sampling.

4.3. Results and Performance Evaluation

In this section, we will provide and evaluate the simulation results we have compiled to explain the performance improvements of SAC over RAND under various scenarios, cases where SAC cannot definitively achieve better results than RAND and how the CRs behave with SAC in the system.

4.3.1. Cooperation Tendency

In this section we will evaluate the effect of increasing cooperation tendency of the CRs in the system. Selfishness, i.e., low cooperation tendency, degrades the cooperation performance as selfish users are more likely to reject the sensing requests. Hence, CRs, if they do not avoid these selfish users, will have to sense with less number of CRs, or sometimes alone. Therefore the sensing performance will converge to the performance of local sensing with an added cost of cooperation request overhead. Hence, appropriate selection of the cooperating set is paramount. In Figure 4.1, `cooperative` refers to the cases where the spectrum sensing is performed collaboratively with at least two CRs, and `total` covers both local and cooperative sensing. As Figure 4.1 shows, performance of cooperative sensing improves with increasing β for both schemes. However, SAC always attains higher performance than that of RAND. Under low cooperation tendency, SAC with 0.9 discovery ratio significantly overperforms RAND with 0.58 discovery success. Even for high cooperative modes (e.g., $\beta = 0.9$ or $\beta = 1.$), SAC attains 91% spectrum discovery whereas RAND can only discover 82% of the opportunities. This improvement is due to the selection of the CRs with the highest scores by SAC.

We demonstrate the cooperation overhead in Figure 4.2 as rejected requests. It is obvious that RAND wastes quite a lot of cooperation messages for asking the CRs that will not sense for this CR. Whereas SAC does not send as many cooperation messages in high selfishness scenarios to avoid rejection, network disturbance and cooperation message overhead. Although not quantified in our thesis, high amount of rejected requests can also indicate the waste of energy for a CR. Hence, we can roughly conclude that SAC has the potential to increase energy-efficiency of cooperative sensing. Furthermore, as the cooperation tendency increases, SAC starts to send more cooperation messages while keeping the rejected messages low, therefore maintaining a higher accepted to rejected request ratio. Figure 4.3 backs up this fact. The cooperation request returns is given as a ratio, acceptance or rejection per request in this figure, so the individual success of the requests that are sent can be seen. As it can be seen, in choosing the CRs that the requests are being sent to, SAC is better than RAND in

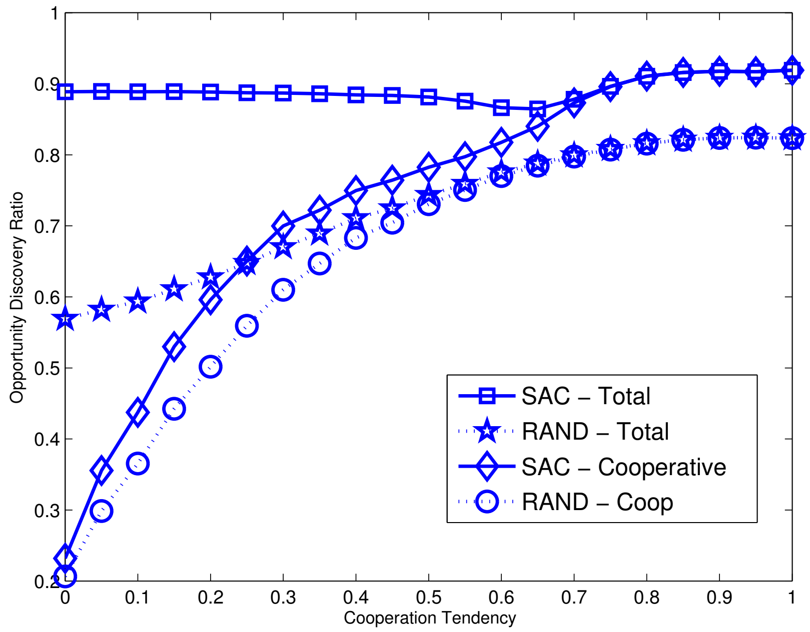

Figure 4.1. Effect of cooperation tendency β on opportunity discovery ratio.

low and high cooperation tendency values (between 0, 0.25 and between 0.55 and 1). As it was clearly seen in Figure 4.2, RAND shows spammy behavior and sends cooperation messages to three CRs in its sector no matter what. On the other hand, SAC doesn't send out as many cooperation messages until a cooperation tendency value of 0.85, meaning that it can almost be sure that the requests it sends will be accepted. What we can draw from these results as a conclusion is that the spammy behavior can outperform clever but sparse choosing of cooperators under medium cooperation tendency, though not by a large margin. However it should be noted that SAC can avoid sending many cooperation requests to CRs when the rejection ratio is high (0.77) due to very high selfishness ($\beta = 0$) and still do not increase the cooperation requests until the rejection ratio gets below 0.3 at $\beta = 0.6$.

In Figure 4.4 we have plotted the number of opportunities missed by a type of sensing, cooperative or local, in the system. The local plots show the amount of idle channels that have been sensed locally, due to either the decision of the sensing CR to not send any cooperation requests or being rejected by all the cooperation requested

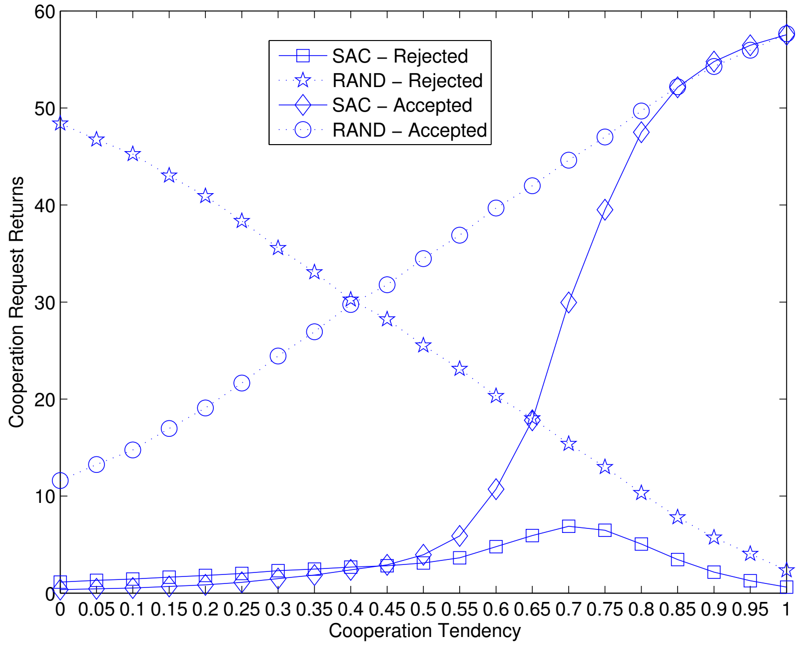

Figure 4.2. Effect of cooperation tendency β on cooperation request returns.

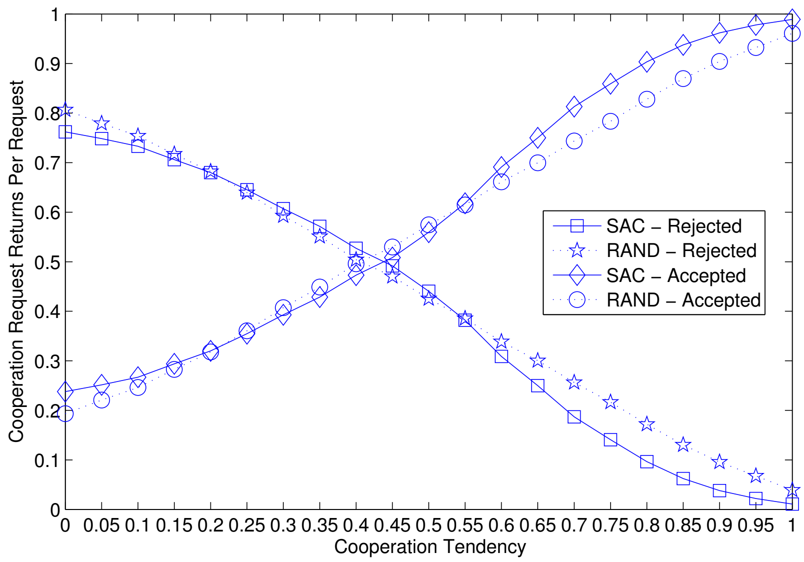

Figure 4.3. Effect of cooperation tendency β on cooperation request returns per request.

CRs, and haven't been accessed because of a false alarm. The cooperative plot show the same but according to channels that have been cooperatively sensed. Total value shows how many of the opportunities in the system have been missed. From the values we can first list the following two observations:

- SAC misses less opportunities from RAND overall.
- SAC utilizes much more of local sensing in low cooperation tendency values as opposed to RAND, which maintains a high missed cooperative opportunity number, which constitutes most of its missed opportunities.

We have seen the opportunity discovery rate of the schemes in Figure 4.1. The amount of opportunities present in the system is constant for both schemes and therefore SAC is better at finding CRs which report the channel condition accurately or choosing local sensing over cooperative sensing when it would be beneficial, by not selecting malicious or overly selfish users.

Notice that for $\beta < 0.55$, SAC's total missed opportunities come from local sensing and for $\beta > 0.65$ cooperative and total missed opportunities start to converge with local sensing slowly moving out of the equation. This result can also be seen in Figure 4.5, which is a figure that shows the the total sensing time spent on local and cooperative sensing in the system. However we should state this fact, to be able to differentiate cooperations of two CRs over three and four CRs, we have added all the individual sensing times spent by the cooperating CRs, so think of this sensing time graphic in this light. The plot first of all shows the fact that SAC doesn't shift the weight into cooperative sensing until $\beta = 0.5$ and RAND starts from being overly rejected at low cooperation tendency to perform some amount of local sensing to increasingly performing cooperative sensing. As it can be seen, until $\beta = 0.75$ cooperative sensing time of SAC doesn't surpass RAND, which means until this point, even though SAC starts to prefer cooperative sensing ($0.55 < \beta < 0.75$), it doesn't send cooperation requests and perform cooperation with as many users as RAND does. This result shows that SAC can be picky in choosing the maximum number of cooperators (N_{coop}) until a certain cooperation tendency, after which all the requests are more probable to be accepted.

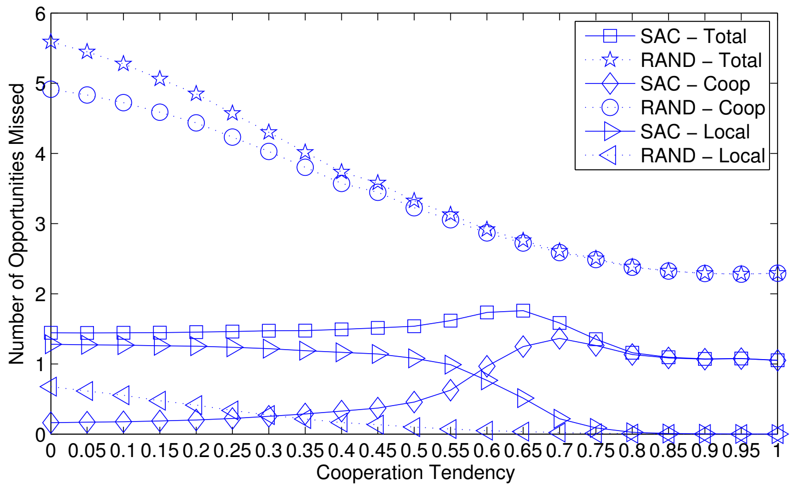

Figure 4.4. Effect of cooperation tendency β on the overall sensing time spent.

Figure 4.6 shows how SAC outperforms in avoiding the scarce amount of malicious users ($d_m = 0.15$) in the system. We have now explained that after a certain cooperation tendency, which ensures more cooperation requests will be accepted, SAC increases the amount of cooperation requests it sends to CRs and hence the cooperative sensing. In this figure it can be seen that the number of malicious users that are being cooperated with is very close to zero until $\beta = 0.6$ at which point this number starts to increase. However the number of malicious users that are cooperated with do not pass the value 2.5 even though we have shown with the previous figures that cooperative sensing of SAC, in terms of contributing CRs at very high β values surpass RAND. Thus, SAC is better at avoiding malicious users which RAND selects constantly in each selfishness value since it does random selection and malicious users accept every cooperation request.

<u>4.3.1.1. Special Case: A system with a high density of malicious users.</u> We have also run the simulations with $d_m = 0.7$ to be able to evaluate how the system behaves in reaction to increasing cooperation tendency under high maliciousness. In Figure 4.7, we can see that the opportunity discovery success in total for SAC is overperforming RAND by quite a large margin. This stems from the fact that RAND is constantly bumping

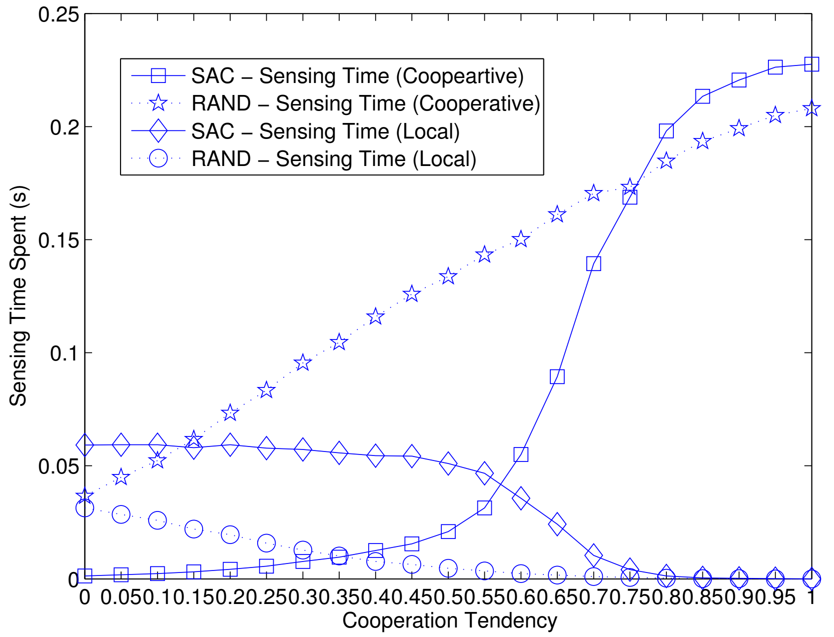

Figure 4.5. Effect of cooperation tendency β on sensing time spent.

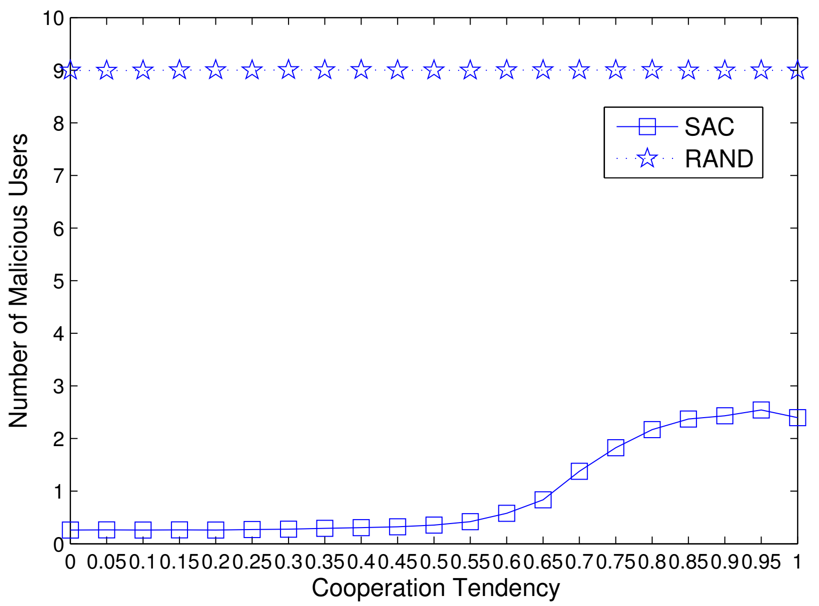

Figure 4.6. Effect of cooperation tendency β on malicious user cooperation.

into malicious CRs that are very highly probable to be selected for cooperation with the social-oblivious decision criteria of RAND. These malicious CRs are preventing the requesting CR from doing spectrum access. On the other hand, SAC starts with a 87% total opportunity discovery, declining to 78% as the very low cooperative opportunity discovery of SAC rises to 70%. This result points at SAC performing more local sensing in a selfish user crowd and increasing cooperative sensing requests as selfishness drops. However, SAC cannot reach neither the cooperative nor total opportunity discovery amounts of the low malicious case explained with Figure 4.1, since as the cooperation tendency increases, although the normal user base starts to respond to cooperation requests, SAC algorithm users cannot meet and identify every malicious CR in the system throughout the simulation length we have determined. Thus, SAC using CRs have the possibility of trying to send requests to malicious users, which are densely found in the system.

In Figure 4.8, we can see how RAND cooperates with malicious users in the $\beta = 0$ data point of the accepted request returns plot. RAND has sent 42 requests (70% of 60, total number of requests RAND sends) that have been accepted on average throughout the system. These accepted requests consist solely of malicious users, since with that selfishness value, no normal CR in the system accepts any request directed at itself. On the other hand, this figure once again shows how conservative SAC is in approaching the community formed up of malicious and selfish users. Let us once again state that SAC increases cooperative sensing requests after $\beta = 0.55$ threshold is reached and it keeps rejected requests low while making successful cooperation requests as cooperation tendency increases.

Cooperative sensing results plot in Figure 4.9 show how the correct sensing results for RAND correlate in trend with opportunity discovery plot of RAND. SAC on the other hand is seen to do too many cooperative sensing errors (63%) in low β because it sends the very few amount of cooperation requests to the malicious users in the system. As β becomes 1, SAC starts to find the normal CRs in they system as cooperation candidates and the cooperative sensing errors reach as low as 22%. This value of course is not as low as the 9% reached in the low maliciousness scenario we have previously

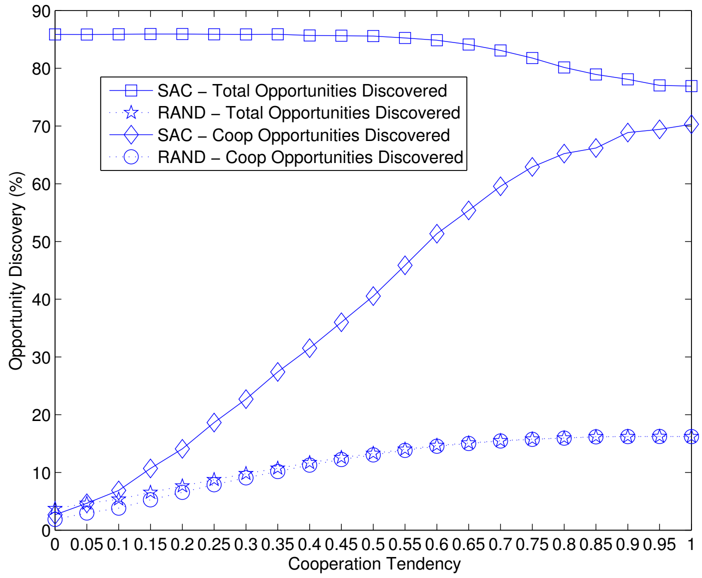

Figure 4.7. Effect of cooperation tendency β on opportunity discovery ($d_m = 0.7$).

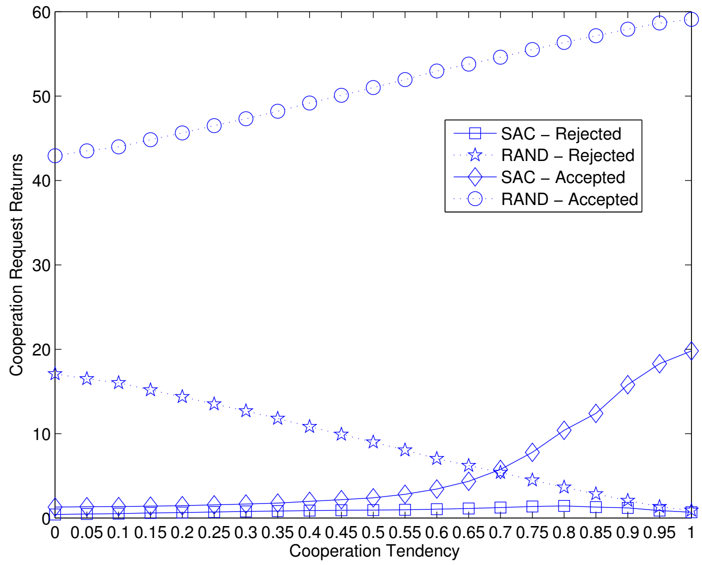

Figure 4.8. Effect of cooperation tendency β on cooperation request returns ($d_m = 0.7$).

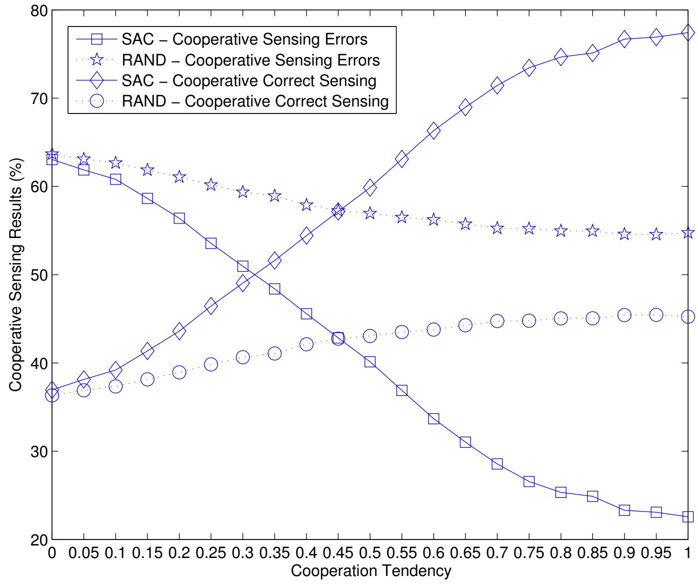

Figure 4.9. Effect of cooperation tendency β on cooperative sensing results ($d_m = 0.7$).

evaluated in the beginning of Section 4.3.1, because as we have mentioned before, it is not possible for the CRs to identify and avoid cooperation with all the malicious CRs in the system.

It can be interesting to note that human social behavior in this kind of situation (subsequently being conned by malicious users) could have been reducing cooperation or being more calculating against cooperation approaches in the real world. This sophisticated behavior might have circumvented the total opportunity discovery performance loss if we have incorporated into the social-aware algorithm, but we haven't overcomplicated SAC and it can still achieve performance improvements against a social-oblivious scheme.

4.3.2. Maliciousness

Next, we evaluate how SAC and RAND act under increasing malicious user density, the ratio of malicious users to the whole CR population. Please recall that malicious users always accept sensing requests but rather than performing sensing, always reply as $H(i) = 1$, i.e., channel is occupied. Hence, if a CR selects malicious users, then it will have less spectrum access opportunities as it tends to stay silent. Figure 4.10 illustrates how RAND and SAC interact with malicious CRs. In RAND, a CR sends a cooperation request to all malicious CRs while SAC achieves to avoid most of them. For example, for malicious user density of 0.6, RAND's probability of a malicious user cooperation is 0.6 whereas it is roughly 0.2 for SAC. The negative effect of this malicious CR interaction is visible in Figure 4.11. For low malicious CR density, RAND and SAC perform similarly, both with 0.9 opportunity discovery ratio. However, as maliciousness increases, RAND's performance degrades substantially while SAC is more robust. Total and cooperative lines overlap for RAND as CRs always perform cooperative sensing with N_{coop} others. On the other hand, these lines do not overlap in SAC since CRs in this scheme may perform local sensing in cases where none of the neighbor CRs have score above λ_{coop}. Because there is a learning period of a CR to build opinion about the other CRs, it is not surprising that SAC's performance also decreases. However, the performance degradation is minimal compared to RAND.

In Figure 4.12 we can see the numerical opportunity discovery results breakdown for all sensing types and in total. It can clearly be denoted that RAND's cooperative opportunity discoveries almost constitute the whole spectrum discovery of the scheme, whereas SAC, although showing similar decline in trend for cooperative sensing, does less cooperative opportunity discoveries only because of the increase in local opportunity discoveries and not mainly because of being exploited by malicious users, since the total number of opportunity discovery decline is only slight.

The collision percentage plot given in Figure 4.13, when combined with the findings from the number of opportunity plot, reveals that SAC induces more collisions in the system because of two facts:

- Collisions in each spectrum access after a local sensing decision is more than cooperative sensing induced collisions, roughly 8% more. Since SAC does increasingly more local sensing than RAND as maliciousness in the system rises,

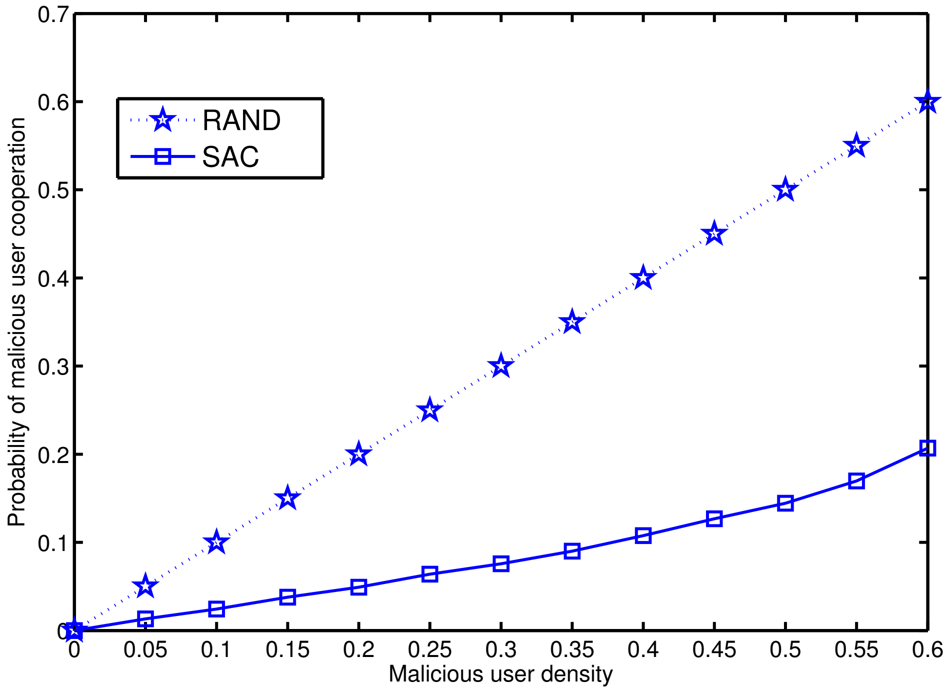

Figure 4.10. Effect of malicious user density on probability of malicious user cooperation.

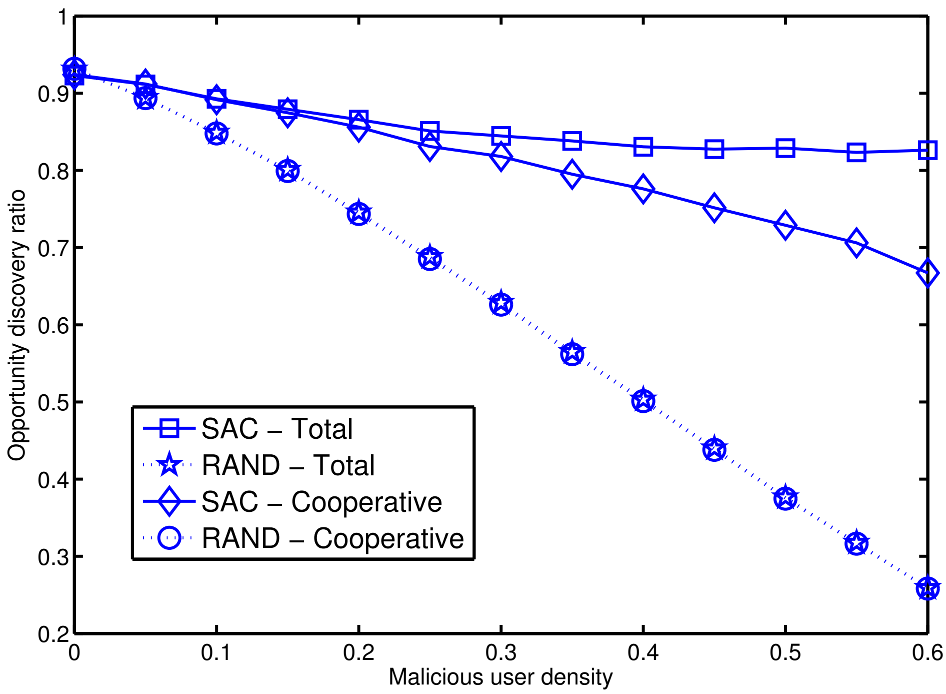

Figure 4.11. Effect of malicious user density on opportunity discovery ratio.

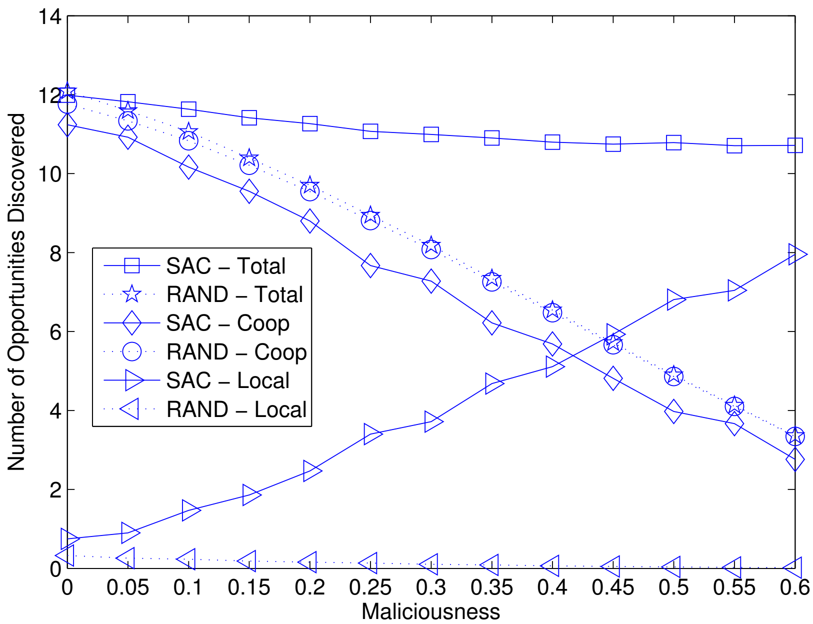

Figure 4.12. Effect of malicious user density on number of opportunities discovered.

these individual collision percentages will mean that SAC will have more collisions overall.

- RAND has less cooperative sensing induced collisions than SAC, due to RAND's cooperation with malicious users being more than SAC and this leading to the prevention of spectrum access by RAND's cooperative sensing. Less spectrum access results in less collisions.

From Figure 4.14 we can clearly see the acceptance rate of cooperation requests. Recall that the simulations for maliciousness scenario are performed with a β mean of 0.7, which is RAND's number of accepted requests per each request made at zero d_m value. It is clear that SAC chooses its cooperation candidates better since at this point SAC's acceptance rate is 0.81. As the amount of malicious users in the system increase, RAND's acceptance rate increases as a side result of malicious user's guaranteed acceptance policy to incoming requests. On the other hand, SAC roughly keeps the same acceptance ratio, although we believe this to be a reasonable result since we know that both the amount of malicious users cooperated with has increased and

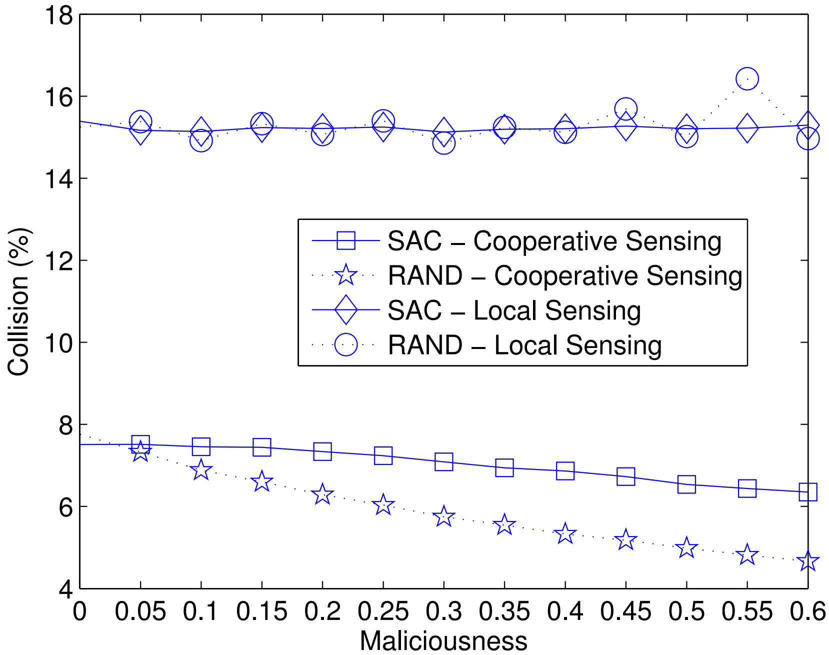

Figure 4.13. Effect of malicious user density on collision percentage per spectrum access.

the amount of cooperation requests have decreased (meaning less data set from which a CR chooses, generally leading to lower success in request returns in our simulations, see figures 4.8, 4.2.

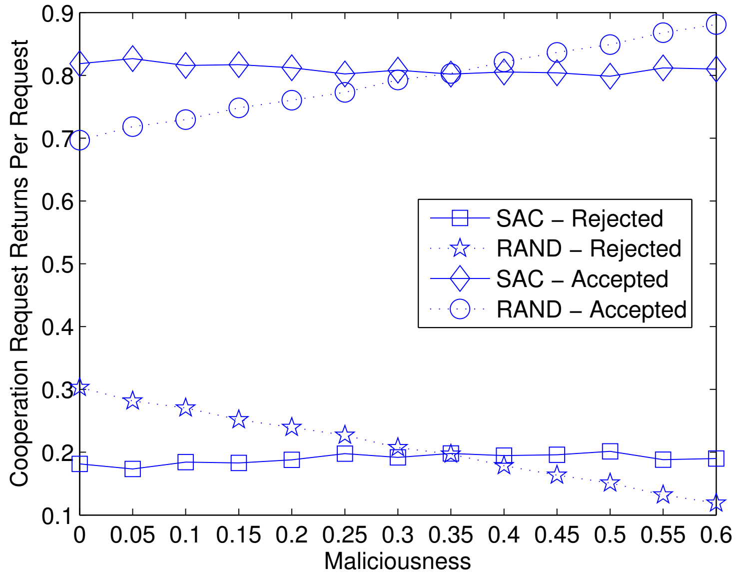

Figure 4.14. Effect of malicious user density on cooperation request returns per request.

Figure 4.15 shows the amount of cooperation requests sent to be able to discover a transmission opportunity. This plot provides results in line with the previous plots evaluated in this section, with RAND doing increasingly more cooperation requests per opportunity discovery than SAC with increasing maliciousness. SAC's local sensing opportunity discovery has also been reflected into this plot because request per discovery for total is increasingly lower than cooperative discoveries, which means that there are local discoveries contributing to the total line in the plot.

After inspecting what happens for the whole system, we will evaluate the results for the malicious users in the system in the figures 4.16 and 4.17. The first figure shows a plot of opportunity discovery by malicious users in both a quantitative way and as a percentage (malicious user success). The second figure shows the cooperation requests by malicious users per successful cooperative discovery under increasing maliciousness.

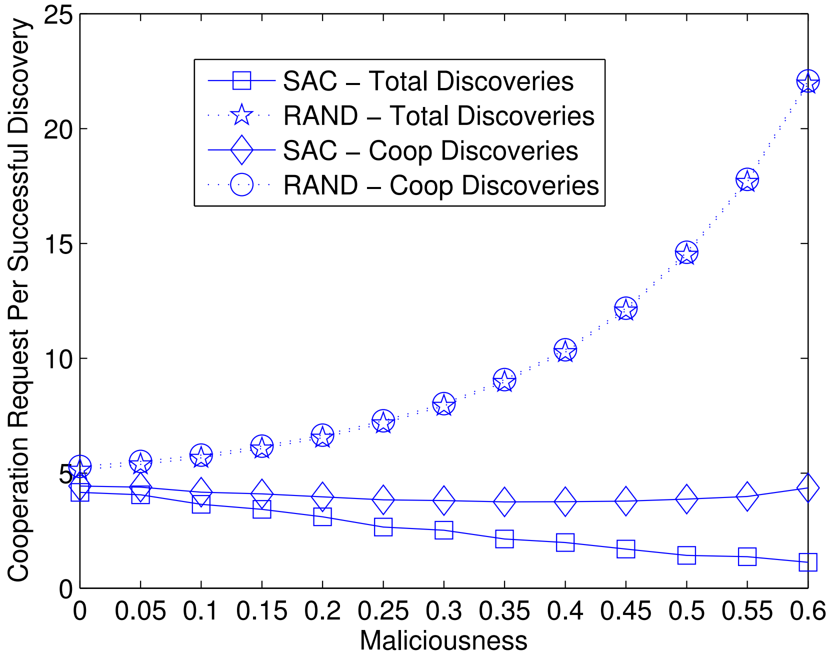

Figure 4.15. Effect of malicious user density on request per successful discovery.

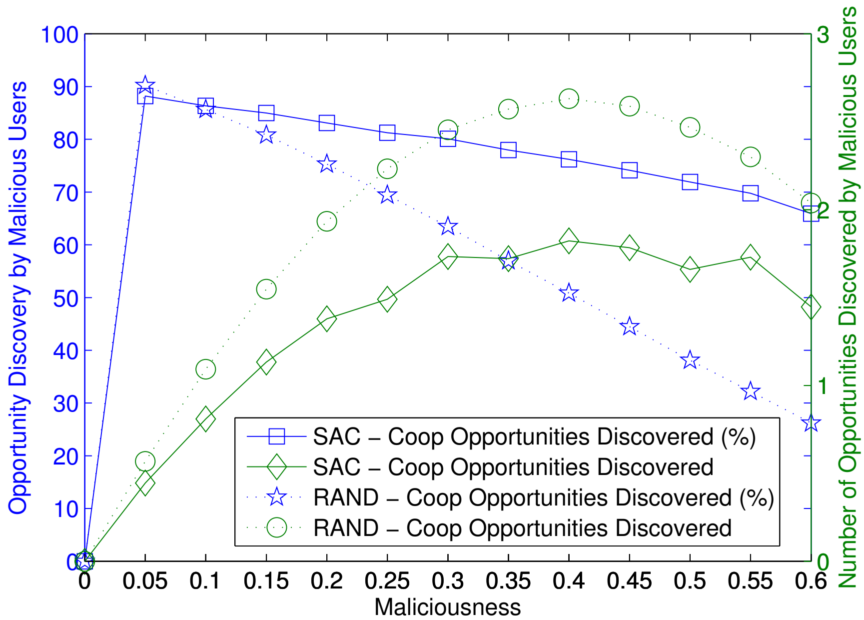

Figure 4.16. Effect of malicious user density on opportunity discovery ratio of malicious users.

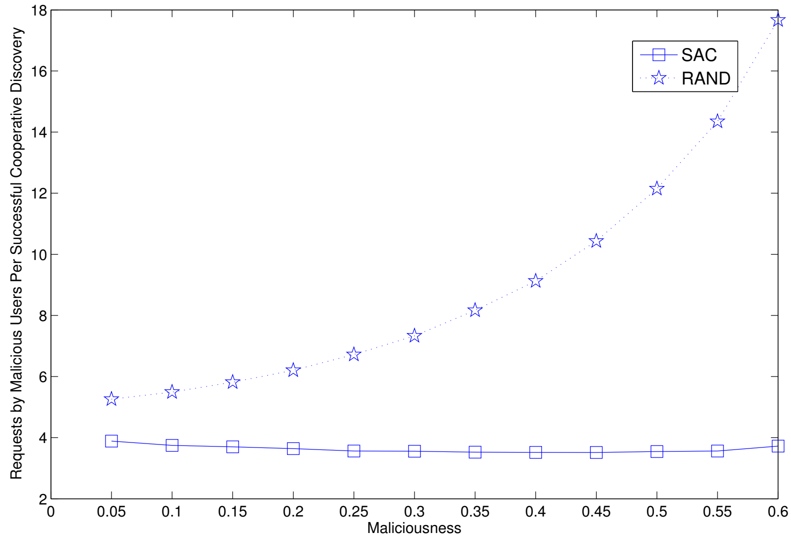

Figure 4.17. Effect of malicious user density on the cooperation request per successful discovery of malicious users.

What we can mainly draw from these figures is that when the system CRs operate with SAC, the malicious users have a high success in finding cooperators that accept their requests and in discovering opportunities by cooperative sensing. This success drops less with increasing maliciousness when compared to RAND, meaning that malicious users of SAC overperform RAND. This might seem as a bad thing for system performance in general, but we have provided the second Y-axis in Figure 4.16 to show that RAND has more numerical cooperative opportunity discoveries than SAC and in fact steals more of the total opportunity discoveries in the system. Thus, malicious users only cooperate more efficiently in SAC, but SAC still ensures that the opportunities present in the system are used more by normal users when compared to RAND, since the users develop awareness against malicious users and reject their cooperation requests due to low scores.

4.3.3. Cooperation Threshold

In Figure 4.18, we show the effect of cooperation threshold. Since SAC chooses the top scoring CRs that are above the threshold for cooperation, the threshold starts to become important as the selfishness of users or the amount of malicious users in the system increases, creating a need for filtering low scoring CRs. We model these cases in three scenarios. Scenario I represents the two negative characteristics where the malicious user density is high as well as the selfishness of users. Scenario II represents high malicious CR density but a low selfishness whereas Scenario III has high selfishness but low malicious user density. These three cases are analyzed to determine an appropriate λ_{coop} value for the system. When the cooperation threshold is low, opportunity discovery ratio is also low due to sending cooperation requests to selfish and malicious users, resulting in missed opportunities. However as λ_{coop} increases past the $0.4 - 0.45$ mark, the opportunity discovery ratio starts to converge to 0.9. On the other hand, as λ_{coop} gets higher, the tendency to perform local sensing instead of cooperative sensing increases. Therefore, both to encourage cooperation and to avoid missing cooperative sensing opportunities, we can set λ_{coop} to 0.45 in our model.

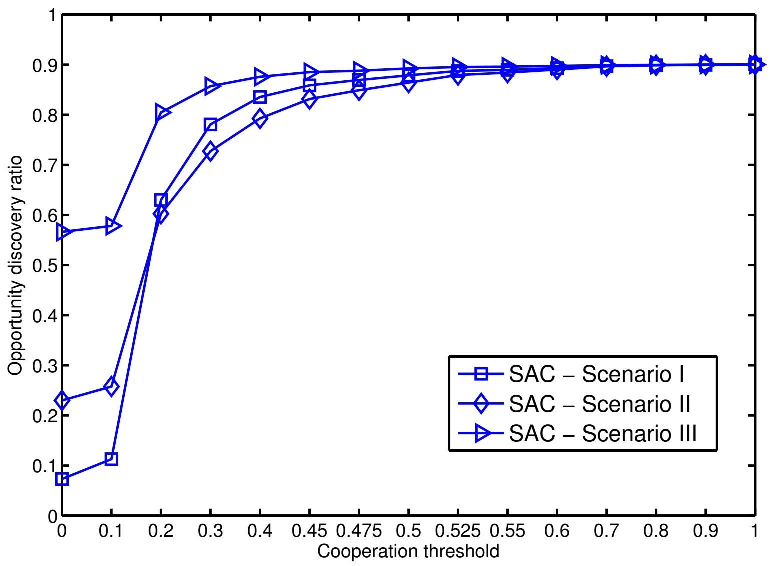

Figure 4.18. Effect of cooperation threshold on opportunity discovery ratio.

Furthermore, for the three scenarios, we can see that the effect of λ_{coop} on opportunity discovery is higher with high maliciousness values in comparison to high selfishness (from 0.22 in Scenario II vs. from 0.56 in Scenario III). This fact is due to opportunity discovery being remedied by local sensing in high selfishness values whereas a high maliciousness invites the low λ_{coop} using CRs to miss opportunities by trusting the malicious CR's busy sensing results.

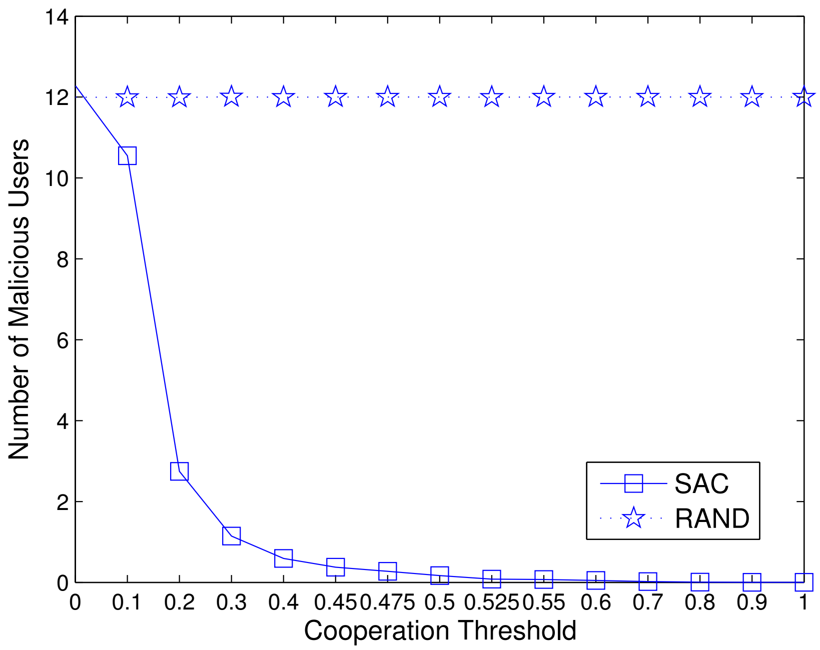

Figure 4.19. Effect of cooperation threshold on malicious user cooperation.

In Figure 4.19 we see the amount of malicious users cooperated versus cooperation threshold plot. The results in this figure is obtained with a single scenario, with $d_m(low)$ and $\beta(low)$ values, representing our standard simulation setup. We can easily determine the need for an appropriate cooperation threshold, since when $\lambda_{coop} = 0$, SAC is performing almost the same with RAND since it sends cooperation requests without filtering the candidates. The plot supports the decision for $\lambda_{coop} = 0.45$ since this value is roughly the point at which the number of malicious users cooperated with decreases substantially. We also know that although there are lower values achieved at higher thresholds, these result in less cooperation in the system, something we do not support.

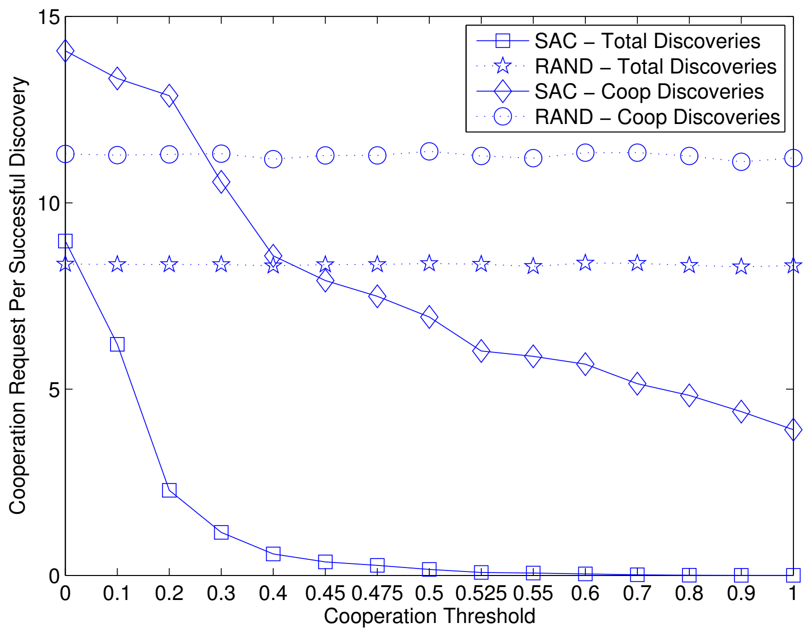

Figure 4.20. Effect of cooperation threshold on request per successful discovery.

Finally, Figure 4.20 is the final supporter of our $\lambda_{coop} = 0.45$ claim. As it can be seen from the plot, with increasing λ_{coop} values, SAC's makes a better choice of CRs for requests and this result in higher discovery of opportunities per request. After the $\lambda_{coop} = 0.3$ mark, SAC outperforms RAND in cooperative discoveries per request. Again, increasing the λ_{coop} above 0.5 results in the decrease of cooperation attempts and increase in opportunity discoveries obtained by local sensing decisions, bringing the request per successful discovery in the system close to zero. This plot though, shows that the choice of 0.45 for λ_{coop} is valid but may not be optimal. However, we have chosen to use this value as the threshold in our simulations since our aim is to show the performance improvement potential of a social-aware scheme rather than optimization.

4.3.4. Cognitive Radio Behavior in System

In this section, we will delve into the community, friendship and cooperation score mechanism's effect on CR relationships. We will present some standard and specific examples showing a CR's interaction with different kind of other CRs, e.g. a friend, a malicious user, a user with moderate selfishness. This is important in understanding how the CRs individually behave against certain situations in the system.

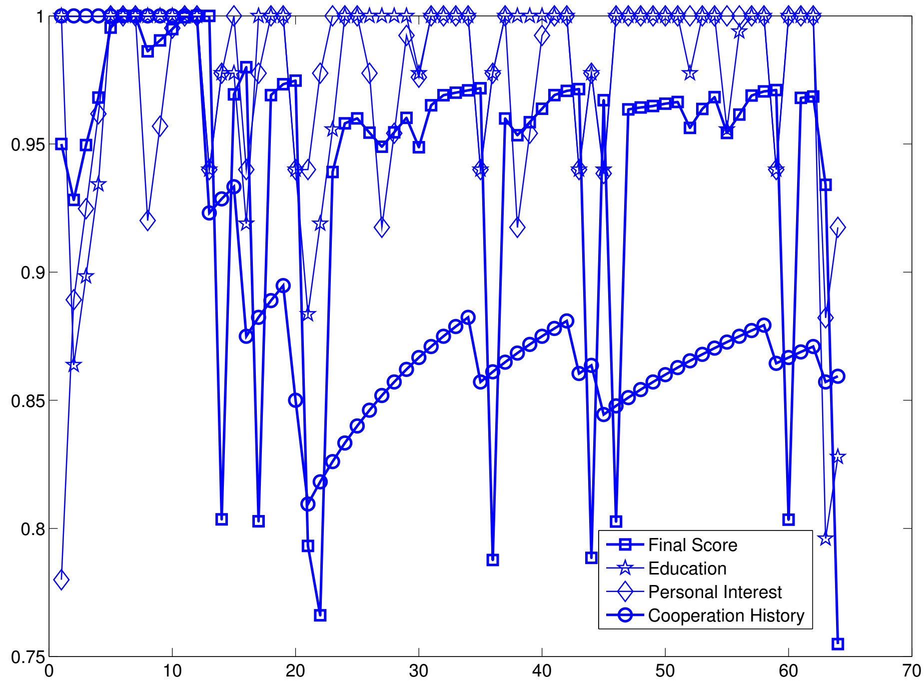

Figure 4.21. A CR with $\beta = 0.68$ cooperating with a friend CR with $\beta = 0.74$.

Before the first plot that we are going to explain in this section let us provide some preliminary information about how they are drawn. The x-axis is the number of cooperations that this CR has had with the target CR. The y-axis shows the score value (between 0 and 1) for the education, personal interest and final score at the time that this CR has evaluated the target CR as a cooperation candidate and sent a cooperation request. The y-axis also shows the cooperation history score, at the time when the target CR has cooperated (if the CR did not reject) with the requesting CR and the result of this cooperation has been evaluated as an error or a correct sensing result. Therefore the data points for cooperation history score are recorded after the cooperation whereas the other three scores are recorded before. This means that the cooperation in a previous data point will affect the final score of the next data point. Another important thing to note is that the cooperations may or may not happen in consecutive timeslots since the nodes are mobile and a CR can only sense in a consecutive timeslot with a very low probability. Therefore the e and pi sympathy values of the CR towards the e and pi values of the target CR can change between two interactions, not a result of the previous cooperation between the CRs.

In Figure 4.21, we start with a standard plot featuring a CR with a selfishness value not deviating much from the mean selfishness of the system with its friend who has a similar selfishness value. For easy reference, we will name the requesting CR as CR_i and the cooperated CR as CR_j from now on. If we take a look at the first cooperation between these two CRs, which is a successful cooperation, the initial sympathy values of CR_i for CR_j are ${L_s}^i(e_j) = 1$ and ${L_s}^i(pi_j) = 0.75$. Since the β values of these two CRs are also pretty suitable for cooperation, these CRs can be expected to choose each other for cooperation anytime they are in the same grid. This plot shows exactly that our predictions are accurate. The sympathy, cooperation averages and therefore the final score of CR_j always stays way above the 0.45 λ_{coop} and these CRs cooperate mostly successfully 64 times throughout the simulation run.

Figure 4.22 shows an interesting situation where CR_i with a low selfishness value has a friend CR_j with a moderate selfishness, $\beta = 0.51$, sympathy values being ${L_s}^i(e_j) = 1$ and ${L_s}^i(pi_j) = 0.75$. Since the friend has a moderate cooperation tendency, the final score boost by the friendship and community scores have been enough to keep the cooperation between them going, however the 60% success rate in cooperations is still pretty low compared to a standard friend we have discussed in Figure 4.21. Another interesting fact to note about the cooperations between these CRs is that the ${L_s}^i(e_j)$ has fallen significantly throughout the simulation (especially after 33rd interaction) but the rise in ${L_s}^i(pi_j)$ and the friendship bonus to the score has been enough to keep the final score high ($0.6 < s(j) < 0.8$) and keep the cooperations going.

In Figure 4.23 we see the cooperation struggles between two CRs with CR_j having a moderate selfishness value and therefore not being a preferable candidate for cooperation. The initial sympathy values of CR_j are ${L_s}^i(e_j) = 1$ and ${L_s}^i(pi_j) = 0.5$.

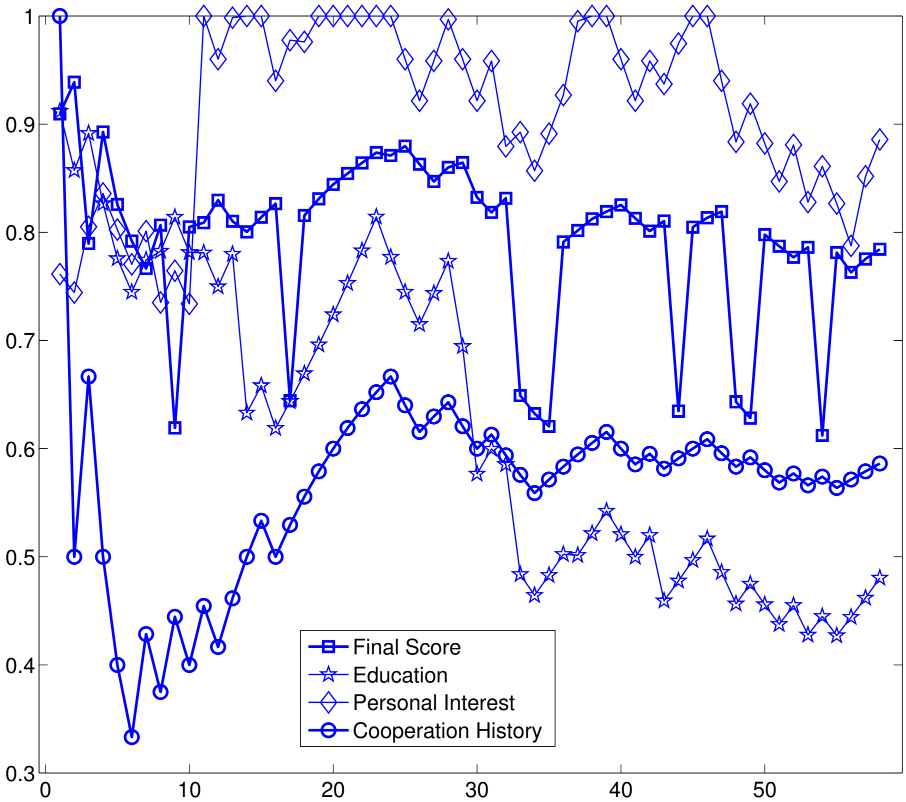

Figure 4.22. A CR with $\beta = 0.89$ cooperating with a friend CR with $\beta = 0.51$.

The high education sympathy keeps the final score of the CR high at the beginning of the simulations and since the sympathy for this education value doesn't go down throughout the simulation, this results in a harmful drag of communications between these two CRs. CR_i continues to send cooperation requests to this mostly rejecting CR due to reputable community perception. However the cooperations do not last for long as the score gradually decreases due to failed cooperations and these two CRs interact for 14 times in total.

Figure 4.24 shows another case where a half-preferable cooperation candidate with $\beta = 0.50$ being saved by high sympathy values. Notice that the communications between CR_i and CR_j result in final scores very close to λ_{coop} but the sympathy values staying high enough to save the cooperations with the help of a successful cooperation

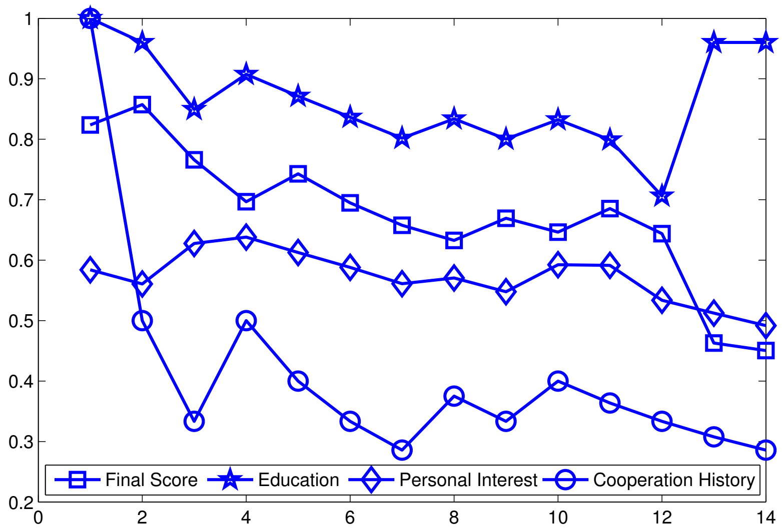

Figure 4.23. A CR with $\beta = 0.67$ cooperating with a CR with $\beta = 0.51$.

(e.g. cooperations 2, 6, 9, 15). Therefore, 40 interactions take place with 0.4 success rate, an example of how good community perception can support communications between two parties even though the results aren't beneficial.

Figure 4.25 shows the cooperation between two moderately selfish CRs, with CR_j having sympathy values of ${L_s}^i(e_j) = 1$ and ${L_s}^i(pi_j) = 0.5$, meaning that this CR has neither common interests nor a cooperation tendency suitable for being qualified as a good candidate. The two CRs first interact with an $s(j)$ value of 0.5, which is rejected by CR_j. Thanks to the rise in community perception of CR_i due to cooperations with other CRs and SAC system being tolerant to the learning period interactions until $n_{coop} = 4$ requests are made, these two CRs again cooperate with an $s(j)$ value of 0.46, this time successfully. This boosts the score of CR_j leading to another interaction, which fails due to rejection and after another failed cooperation at interaction number four, CR_i does not send any cooperation requests to CR_j throughout the simulation, therefore eliminating a bad cooperation candidate.

Figure 4.26 shows a faster severing of cooperations (when compared to Figure 4.25) between two CRs with moderate cooperation tendencies and no common

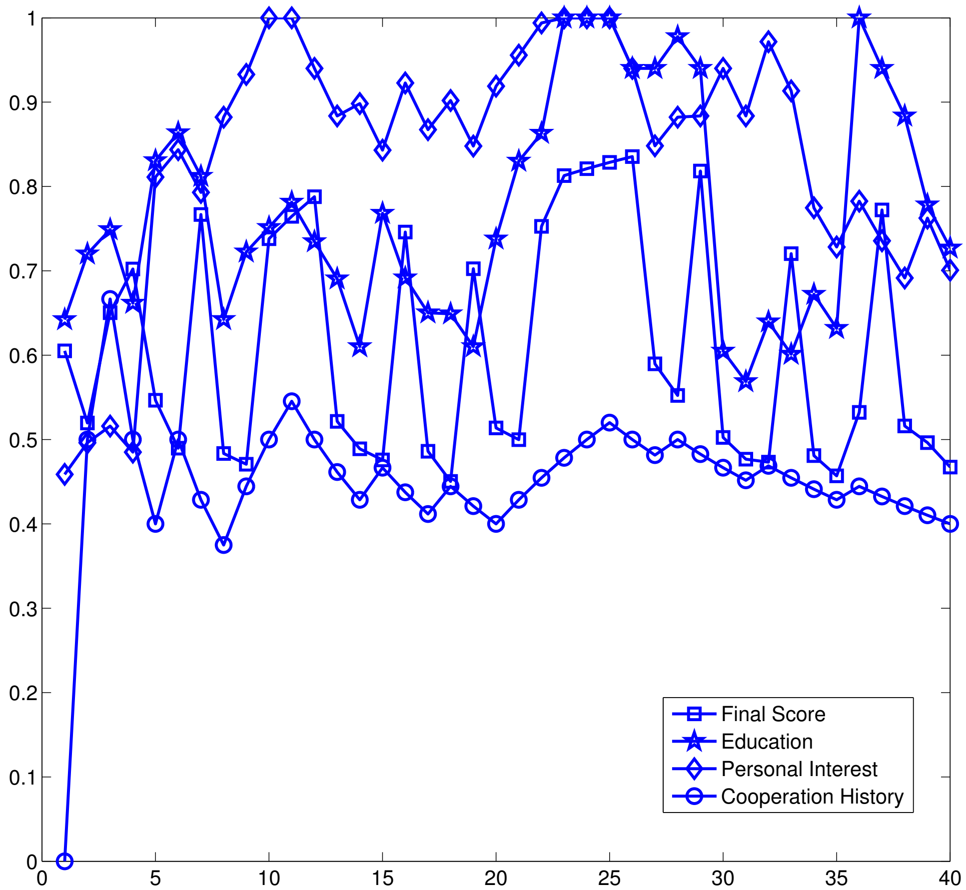

Figure 4.24. A CR with $\beta = 0.68$ cooperating with a CR with $\beta = 0.50$.

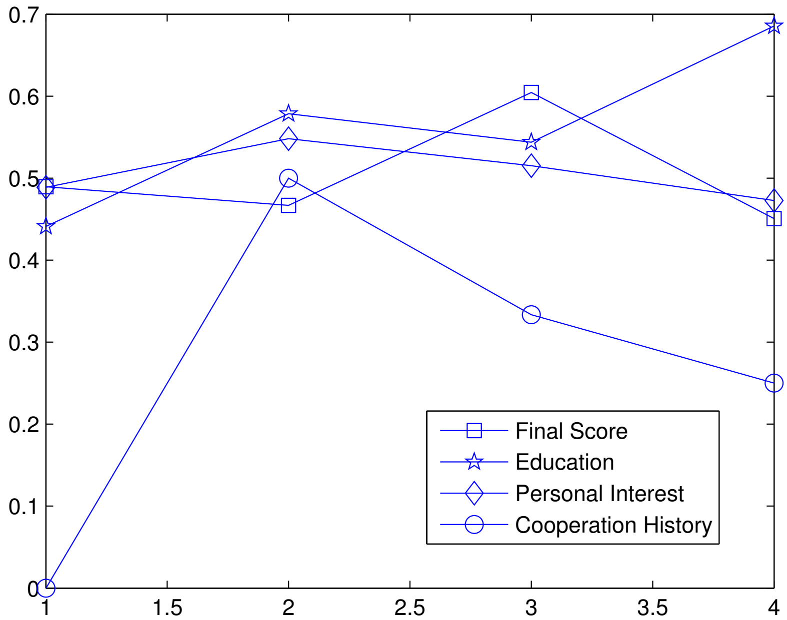

Figure 4.25. A CR with $\beta = 0.55$ cooperating with a CR with $\beta = 0.50$.

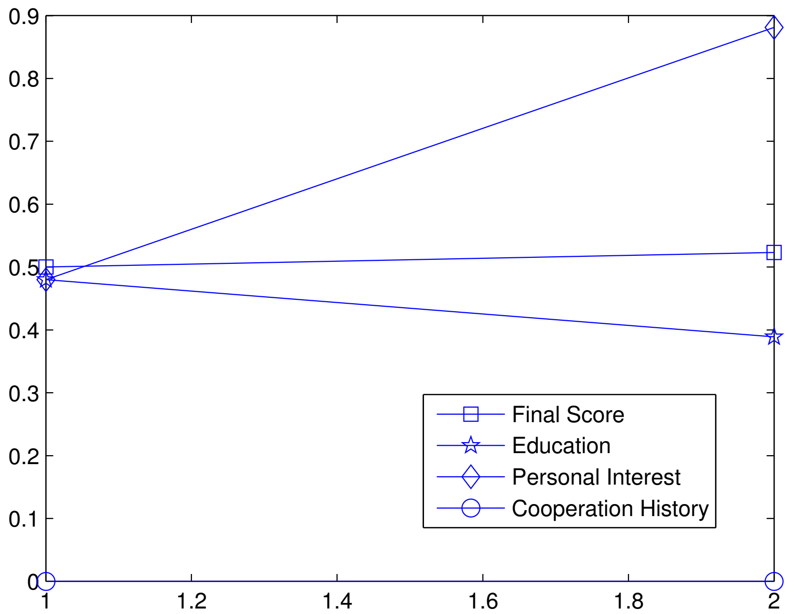

Figure 4.26. A CR with $\beta = 0.56$ cooperating with a CR with $\beta = 0.50$.

personal interests or education values. When we inspect the plot, we see that the first cooperation request of CR_i has been rejected by CR_j which has brought the $s(j)$ value below λ_{coop}. Then, the rise in the ${L_s}^i(pi_j)$ value has created another opportunity for cooperation, which also failed. Although the learning period for CR_i on CR_j haven't yet finished, CR_i does not send any more cooperation requests to CR_j throughout the whole simulation. Please note that since the score of the CR is already low, CR_i does not tend to cooperate with CR_j as long as there are $N_{coop} = 3$ candidates with better scores. Although it is important to not disregard a CR in the learning period, since it can have a high β value and a good cooperation candidate may be lost, in this case, the already low performing CR_j can be quickly cut off due to both the low scoring and finding other better CRs to cooperate.

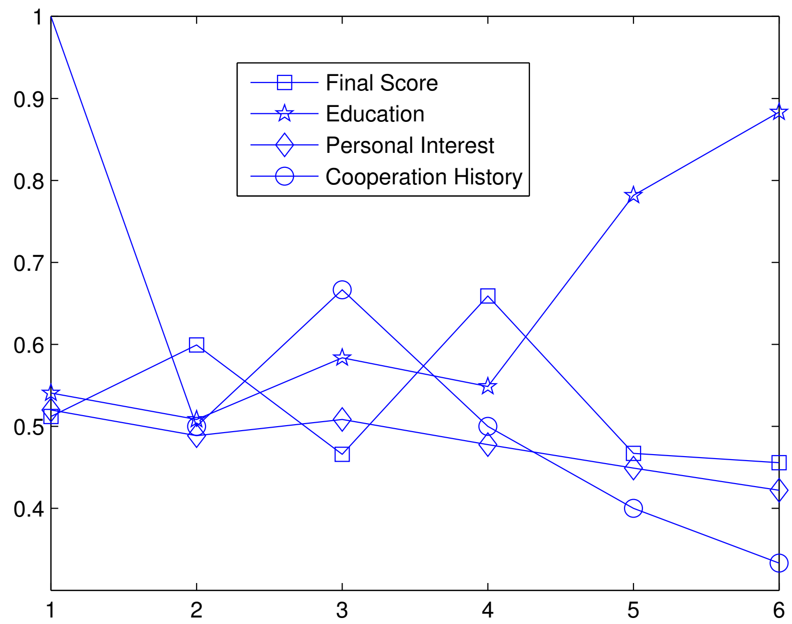

Figure 4.27. A CR with $\beta = 0.55$ cooperating with a malicious CR with $\beta = 0.64$.

We show how a CR identifies a typical malicious user in Figure 4.27. As it can be seen in the plot, initial community perception of CR_i for CR_j is ${L_s}^i(e_j) = 0.5$ and ${L_s}^i(pi_j) = 0.5$, making CR_j a poor candidate in score already. However CR_j is in this case lucky to have found two cooperation opportunities where the channel was busy and is able to keep the $s(j)$ value above λ_{coop}. However after three straight failed cooperations, CR_i identifies CR_j as a wrong cooperation result reporter and does not send any further requests throughout the simulation.

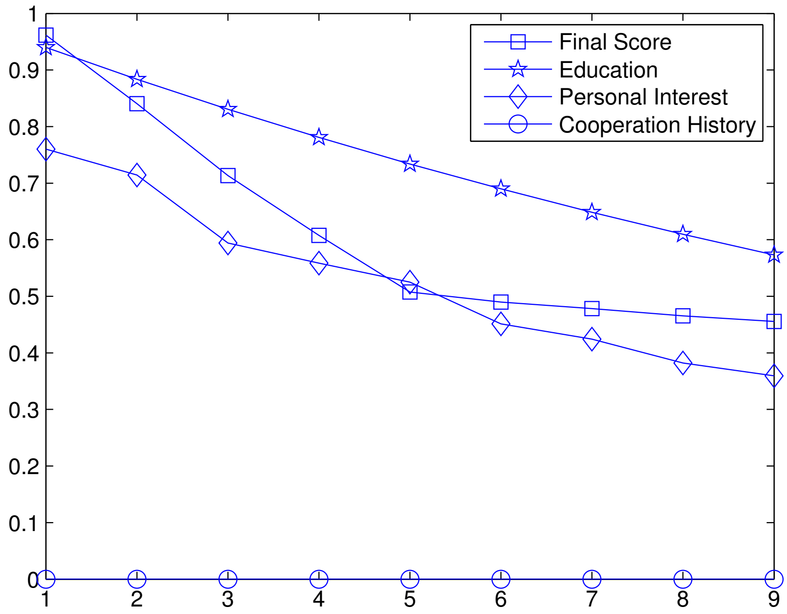

Figure 4.28. A CR with $\beta = 0.58$ cooperating with a malicious CR with $\beta = 0.86$.

Finally, we will present an interesting case which we have created on purpose to see how the system will behave under it. In Figure 4.28 we have plotted the case when CR_i and CR_j are friends but at the same time CR_j is a malicious user. What this plot shows us is that it requires the community perception of CR_i for CR_j to also go down for it to be able to stop requesting cooperations from CR_j. If other CRs of the same community with this malicious user were performing successful cooperations with CR_i, the cooperations would have dragged on with CR_i being harmed in the process. Although not likely to happen in a realistic case, this type of scenario shows a weakness in the scoring system we have determined for RAND and there is room for improvement. However we have many times stated that the algorithm is made simplistic and not optimized in terms of any metric (e.g throughput, cooperative candidate selection, energy efficiency) and as it can be seen in this case, only the community support can be a factor for the need to improve SAC, it can resolve the issue with this simplistic scoring mechanism if there is no community support to the malicious user.

5. CONCLUSION AND FUTURE DIRECTIONS

In this thesis, we have presented a cooperative sensing scheme that makes use of the social relations among the CRs that are in contact in order to select the set of cooperators. Almost all previous works on cooperative sensing assume that each CR is willing to cooperate with another CR that asks for cooperation. However, we criticize this assumption and propose a more realistic cooperative sensing scheme which models cooperation tendency according to various social ties among requesting and requested CRs. Besides these ties, i.e. friendship and community, sensing performance of each cooperating CR is also recorded as a kind of learning mechanism. The sensing performance record of a CR is later used in selecting proper cooperators. The model is basically formed upon a trust mechanism where CRs cooperate more readily or unwillingly based on experience, recommendation and knowledge.

Via simulations, we have shown that our proposal can distinguish malicious users in a network after a certain number of cooperations and avoid these users by eliminating them in the candidate cooperators set. Additionally, our scheme selects cooperators from the set of CRs that are more eager to cooperate. There are two main strengths of SAC:

- SAC makes an elimination in the candidate cooperators set to prefer local sensing over cooperative sensing if the cooperation request will only be causing harm in terms of either the overhead created or result reliability.
- SAC uses scores to rank cooperators amongst each other to select the best possible cooperators, increasing the chances of both being accepted and getting a correct sensing result.

On the other hand, there are two main weaknesses of SAC:

- Due to being a realistic social model, SAC features prejudice and sympathy for communities, which judges members of a community based on the interactions with the individual members.

- SAC, due to being conservative in its approach to cooperation, can tend to prefer local sensing in some average selfishness or maliciousness cases, while finding cooperation opportunities would have been possible with taking more cooperation overhead risk.

We believe that the first main weakness is also an active problem of societies and we do not over-concern ourselves with this problem. However the second problem can be assessed as an engineering problem in which cooperation requests may be sent to rejecting CRs or CRs with a fairly good cooperation average (over 0.5) even though the final score is below the cooperation threshold, trading off cooperation overhead for extra cooperation opportunities.

A first interesting direction is extending our model to various network models, e.g., Erdös-Renyi, Albert-Barabási, and exploit how the network structure affects sensing performance or cooperation behavior at CRs with different properties, e.g. higher degree vs. lower degree CRs. In our thesis, we have not considered any physical layer parameters for sensing, which restricted our model's design space and in turn limited the potential improvements over RAND.

In our system, CRs only have individual prevention mechanisms against overly selfish or malicious users. Another interesting addition to the system would be an encouragement/discouragement mechanism where cooperation in the network is promoted via certain rewards and selfish or malicious behavior is punished. We have encountered these types of work for DTNs during our related work review in Chapter 2. A mechanism like this will enable a quicker discovery of non-cooperative nodes by other CRs in the network, as well as promote cooperation in general, but will increase the general messaging overhead and create problems for the resource starved nodes by demoting them in the network.

We have used a random movement featuring mobility scheme in the system. However, due to the homophily phenomena, social users in real life would be more likely to meet with the members of the same community or people with which they have certain social ties, e.g. co-workers, family, friends. Designing a mobility model along the lines of exploiting the homophily concept, although being difficult to build an accurate society-mimicking model, can show that social-aware cooperative approaches are actually more beneficial in terms of increasing sensing performance, energy efficiency and reducing messaging overhead.

All in all, we believe that a more complete cooperative sensing scheme will show the real potential of a social aware sensing approach. We believe that modeling CR networks using social networks approach can raise many exciting questions as well as interesting CR operation policies.

REFERENCES

1. Patil, K., R. Prasad and K. Skouby, “A Survey of Worldwide Spectrum Occupancy Measurement Campaigns for Cognitive Radio”, *International Conference on Devices and Communications (ICDeCom)*, pp. 1–5, 2011.

2. Mitola III, J. and G. Maguire Jr, “Cognitive Radio: Making Software Radios More Personal”, *IEEE Personal Communications*, Vol. 6, No. 4, pp. 13–18, 1999.

3. Akyildiz, I., W. Lee, M. Vuran and S. Mohanty, “Next Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey”, *Computer Networks*, Vol. 50, No. 13, pp. 2127–2159, 2006.

4. Arslan, H., 2007, “Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems”, Springer, London.

5. Mitola., J., 2000, “Software Radios: Wireless Architecture for the 21st Century”, John Wiley and Sons Inc, New York, USA.

6. Zhao, Q. and B. Sadler, “A Survey of Dynamic Spectrum Access”, *Signal Processing Magazine, IEEE*, Vol. 24, No. 3, pp. 79–89, 2007.

7. III, J. M., “Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio”, Ph.D. Thesis, *Royal Institute of Technology, KTH*, Sweden, 2000.

8. Haykin, S., “Cognitive Radio: Brain-Empowered Wireless Communications”, *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 2, pp. 201–220, 2005.

9. Akyildiz, I. F., B. F. Lo and R. Balakrishnan, “Cooperative Spectrum Sensing in Cognitive Radio Networks: A Survey”, *Physical Communication*, Vol. 4, No. 1, pp. 40–62, 2011.

10. Ganesan, G. and L. Ye, "Cooperative Spectrum Sensing in Cognitive Radio, Part II: Multiuser Networks", *IEEE Transactions on Wireless Communications*, Vol. 6, No. 6, pp. 2214–2222, 2007.

11. Atapattu, S., C. Tellambura and H. Jiang, "Energy Detection Based Cooperative Spectrum Sensing in Cognitive Radio Networks", *IEEE Transactions on Wireless Communications*, Vol. 10, No. 4, pp. 1232–1241, 2011.

12. Unnikrishnan, J. and V. Veeravalli, "Cooperative Sensing for Primary Detection in Cognitive Radio", *IEEE Journal of Selected Topics in Signal Processing*, Vol. 2, No. 1, pp. 18–27, 2008.

13. Zhang, W., R. Mallik and K. Letaief, "Optimization of Cooperative Spectrum Sensing with Energy Detection in Cognitive Radio Networks", *IEEE Transactions on Wireless Communications*, Vol. 8, No. 12, pp. 5761–5766, 2009.

14. Lee, S. H., D. C. Oh and Y. H. Lee, "Hard Decision Combining-Based Cooperative Spectrum Sensing in Cognitive Radio Systems", *Proceedings of the International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 906–910, 2009.

15. Cabric, D., S. Mishra and R. Brodersen, "Implementation Issues in Spectrum Sensing for Cognitive Radios", *Conference Record of the 38th Asilomar Conference on Signals, Systems and Computers*, Vol. 1, pp. 772–776, 2004.

16. Leu, A. E., M. McHenry and B. L. Mark, "Modeling and Analysis of Interference in Listen-Before-Talk Spectrum Access Schemes", *Wiley International Journal of Network Management*, Vol. 16, No. 2, pp. 131–147, 2006.

17. Ghasemi, A. and E. Sousa, "Collaborative Spectrum Sensing for Opportunistic Access in Fading Environments", *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 131–136, 2005.

18. Visotsky, E., S. Kuffner and R. Peterson, “On Collaborative Detection of TV Transmissions in Support of Dynamic Spectrum Sharing”, *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 338–345, 2005.

19. Unnikrishnan, J. and V. Veeravalli, “Cooperative Sensing for Primary Detection in Cognitive Radio”, *IEEE Journal of Selected Topics in Signal Processing*, Vol. 2, No. 1, pp. 18–27, 2008.

20. Li, Z., F. Yu and M. Huang, “A Cooperative Spectrum Sensing Consensus Scheme in Cognitive Radios”, *IEEE INFOCOM*, pp. 2546–2550, 2009.

21. Ganesan, G. and Y. Li, “Cooperative Spectrum Sensing in Cognitive Radio, Part I: Two User Networks”, *IEEE Transactions on Wireless Communications*, Vol. 6, No. 6, pp. 2204–2213, 2007.

22. Zhang, W. and K. Letaief, “Cooperative Spectrum Sensing with Transmit and Relay Diversity in Cognitive Radio Networks - [Transaction Letters]”, *IEEE Transactions on Wireless Communications*, Vol. 7, No. 12, pp. 4761–4766, 2008.

23. Visotsky, E., S. Kuffner and R. Peterson, “On Collaborative Detection of TV Transmissions in Support of Dynamic Spectrum Sharing”, *Proceedings of the First International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, pp. 338–345, 2005.

24. Weiss, T. A., “A Diversity Approach for the Detection of Idle Spectral Resources in Spectrum Pooling Systems”, *Proceedings of the 48th International Scientific Colloquium*, pp. 37–38, 2003.

25. Mishra, S. M., A. Sahai and R. W. Brodersen, “Cooperative Sensing Among Cognitive Radios”, *Proceedings of the International Conference on Communications (ICC)*, Vol. 4, pp. 1658–1663, 2006.

26. Peh, E. and Y. C. Liang, "Optimization for Cooperative Sensing in Cognitive Radio Networks", *Proceedings of the Wireless Communications and Networking Conference (WCNC)*, pp. 27–32, 2007.

27. Zou, Q., S. Zheng and A. H. Sayed, "Cooperative Spectrum Sensing via Coherence Detection", *Proceedings of the 15th Workshop on Statistical Signal Processing (SSP)*, pp. 610–613, 2009.

28. Shen, J., T. Jiang, S. Liu and Z. Zhang, "Maximum Channel Throughput via Cooperative Spectrum Sensing in Cognitive Radio Networks", *IEEE Transactions on Wireless Communications*, Vol. 8, No. 10, pp. 5166–5175, 2009.

29. Huang, H., Z. Zhang, P. Cheng, A. Huang and P. Qiu, "Cooperative Spectrum Sensing in Cognitive Radio Systems with Limited Sensing Ability", *Springer Journal of Zhejiang University-Science C*, Vol. 11, No. 3, pp. 175–186, 2010.

30. Oh, D. C. and Y. H. Lee, "Cooperative Spectrum Sensing with Imperfect Feedback Channel in the Cognitive Radio Systems", *Wiley International Journal of Communication Systems*, Vol. 23, No. 6-7, pp. 763–779, 2010.

31. Sun, C., W. Zhang and K. B. Letaief, "Cooperative Spectrum Sensing for Cognitive Radios Under Bandwidth Constraints", *Proceedings of the Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, 2007.

32. Lo, B. and I. Akyildiz, "Reinforcement Learning-Based Cooperative Sensing in Cognitive Radio Ad Hoc Networks", *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 2244–2249, 2010.

33. Ghasemi, A. and E. S. Sousa, "Spectrum Sensing in Cognitive Radio Networks: the Cooperation-Processing Tradeoff", *Wireless Communications in Mobile Computing*, Vol. 7, No. 9, pp. 1049–1060, 2007.

34. Scott, J., 2000, "Social Network Analysis: A Handbook", SAGE Publications.

35. Wasserman, S. and K. Faust, 1994, “Social Network Analysis: Methods and Applications”, Cambridge University Press.

36. Zhu, Y., B. Xu, X. Shi and Y. Wang, “A Survey of Social-Based Routing in Delay Tolerant Networks: Positive and Negative Social Effects”, *IEEE Communications Surveys and Tutorials*, Vol. 15, No. 1, 2013.

37. Wang, S., M. Liu, X. Cheng and M. Song, “Routing in Pocket Switched Networks”, *IEEE Wireless Communications*, Vol. 19, No. 1, pp. 67–73, 2012.

38. Gonzalez, M. C., C. A. Hidalgo and A.-L. Barabasi, “Understanding Individual Human Mobility Patterns”, *Nature*, Vol. 453, No. 7196, pp. 779–782, 2008.

39. Hui, P. and J. Crowcroft, “How Small Labels Create Big Improvements”, *Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 65–70, 2007.

40. Daly, E. M. and M. Haahr, “Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs”, *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’07, pp. 32–40, ACM, New York, NY, USA, 2007.

41. P. Hui, J. C. and E. Yonek, “Bubble rap: Social-Based Forwarding in Delay Tolerant Networks”, *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc ’08, ACM, New York, NY, USA, 2008.

42. Gao, W., Q. Li, B. Zhao and G. Cao, “Multicasting in Delay Tolerant Networks: A Social Network Perspective”, *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’09, pp. 299–308, ACM, New York, NY, USA, 2009.

43. Zhang, Y. and J. Zhao, “Social Network Analysis on Data Diffusion in Delay

Tolerant Networks", *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '09, pp. 345–346, ACM, New York, NY, USA, 2009.

44. Bulut, E. and B. K. Szymanski, "Friendship Based Routing in Delay Tolerant Mobile Social Networks", *GLOBECOM'10*, pp. 1–5, 2010.

45. Shevade, U., H. H. Song, L. Qiu and Y. Zhang, "Incentive-Aware Routing in DTNs", *IEEE International Conference on Network Protocols*, pp. 238–247, 2008.

46. Mei, A. and J. Stefa, "Give2Get: Forwarding in Social Mobile Wireless Networks of Selfish Individuals", *IEEE 30th International Conference on Distributed Computing Systems (ICDCS)*, pp. 488–497, 2010.

47. Zhu, H., X. Lin, R. Lu and X. Shen, "A Secure Incentive Scheme for Delay Tolerant Networks", *Third International Conference on Communications and Networking in China*, pp. 23–28, 2008.

48. Zhu, H., X. Lin, R. Lu, Y. Fan and X. Shen, "SMART: A Secure Multilayer Credit-Based Incentive Scheme for Delay-Tolerant Networks", *IEEE Transactions on Vehicular Technology*, Vol. 58, No. 8, pp. 4628–4639, 2009.

49. Chen, B. B. and M. C. Chan, "Mobicent: A Credit-Based Incentive System for Disruption Tolerant Network", *Proceedings of the 29th conference on Information communications*, INFOCOM'10, pp. 875–883, IEEE Press, Piscataway, NJ, USA, 2010.

50. Li, Q., S. Zhu and G. Cao, "Routing in Socially Selfish Delay Tolerant Networks", *Proceedings of the 29th conference on Information communications*, INFOCOM'10, pp. 857–865, IEEE Press, Piscataway, NJ, USA, 2010.

51. Mislove, A., M. Marcon, K. P. Gummadi, P. Druschel and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks", *Proceedings of the 7th ACM*

SIGCOMM conference on Internet measurement, ACM, New York, NY, USA, 2007.

52. Krebs, V., "Mapping Networks of Terrorist Cells", *CONNECTIONS*, Vol. 24, No. 3, pp. 43–52, 2002.

53. Bulut, E. and B. Szymanski, "Exploiting Friendship Relations for Efficient Routing in Mobile Social Networks", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 23, No. 12, pp. 2254–2265, 2012.

54. McPherson, M., L. Smith-Lovin and J. M. Cook, "Birds of a Feather: Homophily in Social Networks", *Annual review of sociology*, pp. 415–444, 2001.

55. Hillery, G. A., "Definitions of Community: Areas of Agreement", *Rural Sociology*, Vol. 20, No. 2, pp. 111–123, 1955.

56. McMillan, D. W. and D. M. Chavis, "Sense of Community: A Definition and Theory", *Journal of Community Psychology*, Vol. 14, No. 1, pp. 6–23, 1986.

57. Perkins, D., P. Florin, R. Rich, A. Wandersman and D. Chavis, "Participation and the Social and Physical Environment of Residential Blocks: Crime and Community Context", *American Journal of Community Psychology*, Vol. 18, No. 1, pp. 83–115, 1990.

58. Newman, M. E. J., "Modularity and Community Structure in Networks", *Proceedings of the National Academy of Sciences*, Vol. 103, No. 23, pp. 8577–8582, 2006.

59. Okasha, S., "Altruism, Group Selection and Correlated Interaction", *The British Journal for the Philosophy of Science*, Vol. 56, No. 4, pp. 703–725, 2005.

60. Newman, M. E., "The Structure and Function of Complex Networks", *SIAM review*, Vol. 45, No. 2, pp. 167–256, 2003.

61. Easley, D. and J. Kleinberg, 2012, “Networks, Crowds, and Markets: Reasoning About a Highly Connected World”, Wiley Online Library.

62. Caverlee, J. e., “SocialTrust++: Building Community-Based Trust in Social Information Systems”, *IEEE CollaborateCom*, 2010.

63. Nguyen, N. P., T. N. Dinh, S. Tokala and M. T. Thai, “Overlapping Communities in Dynamic Networks: Their Detection and Mobile Applications”, *Proceedings of the 17th annual international conference on Mobile computing and networking*, pp. 85–96, ACM, 2011.

64. “Community Detection in Graphs”, *Physics Reports*, Vol. 486, No. 3–5.

65. Govindan, K. and P. Mohapatra, “Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey”, *IEEE Communications Surveys Tutorials*, Vol. 14, No. 2, pp. 279–298, 2012.

66. Li, H., C.-f. Chen and L. Lai, “Propagation of Spectrum Preference in Cognitive Radio Networks: A Social Network Approach”, *IEEE International Conference on Communications (ICC)*, pp. 1–5, 2011.

67. Liang, Y., Y. Zeng, E. Peh and A. Hoang, “Sensing-Throughput Tradeoff for Cognitive Radio Networks”, *IEEE Transactions on Wireless Communications*, pp. 1326–1337, 2008.

68. Okasha, S., “Altruism, Group Selection and Correlated Interaction”, *The British Journal for the Philosophy of Science*, Vol. 56, No. 4, pp. 703–725, 2005.