Accepted Manuscript

Cooperation-based multi-hop routing protocol for cognitive radio networks

Arsany Guirguis, Mohammed Karmoose, Karim Habak, Mustafa El-Nainay, Moustafa Youssef

PII: S1084-8045(18)30086-9

DOI: 10.1016/j.jnca.2018.03.005

Reference: YJNCA 2086

To appear in: *Journal of Network and Computer Applications*

Received Date: 30 July 2017

Revised Date: 21 February 2018

Accepted Date: 9 March 2018

Cooperation-based Multi-hop Routing Protocol for Cognitive Radio Networks

Arsany Guirguis[a,*], Mohammed Karmoose[b], Karim Habak[c], Mustafa El-Nainay[a], Moustafa Youssef[d]

[a] *Computer and Systems Engineering, Alexandria University, Alexandria, Egypt*
[b] *Henry Samueli school of Engineering and Applied Sciences, UCLA, California, USA*
[c] *School of Computer Science, College of Computing, Georgia Tech., Atlanta, Georgia, USA*
[d] *Egypt-Japan Univ. of Sc. and Tech. (E-JUST), Alexandria, Egypt*

Abstract

In the scope of cognitive radio networks, typical routing protocols avoid areas that are highly congested with primary users, leaving only a small fragment of available links for secondary route construction. In addition, wireless links are prone to channel impairments such as multipath fading which renders the quality of the available links highly fluctuating. In this paper, we propose *Undercover*: a multi-hop routing protocol for cognitive radio networks in which we integrate the collaborative beamforming technique with layer 3 routing. Specifically, our protocol revisits a fundamental assumption taken by the state of the art routing protocols designed for overlay cognitive radio networks; this assumption is that secondary users cannot use the spectrum when primary users are using it. In *Undercover*, we allow a group of secondary users, each with a single antenna, to collaborate together and transmit in the regions of primary users activity. This is done through nulling out transmission at primary receivers via beamforming. Moreover, *Undercover* is designed to enhance the transmission quality at the secondary destinations whenever possible. To account for the excessive levels of interference typically incurred due to cooperative transmissions, we allow our protocol to be interference-aware. Thus, cooperative transmissions are penalized in accordance with the amount of negatively affected secondary flows. We evaluate the performance of our proposed protocol via NS2 simulations which show that our protocol can enhance the network goodput by a ratio reaches up to 250% compared to other state-of-art cognitive routing protocols with minimal added overhead.

Key words: Cognitive Radio Networks, Cooperative Communication, Routing Protocols

1. Introduction

Cognitive Radio Networks (CRNs) was proposed as a promising solution for the spectrum scarcity problem. This problem is increasing more and more and that is why CRNs are imminent to pervade into all fields of wireless communications. We backup this assertion with the following observations. First, with the inherent inefficiency of the static spectrum licensing policies [1] and proliferation of spectrum accessing mobile and connected devices [2], we are quickly heading towards a wireless spectrum crisis [3]. Second, the spectrum regulatory authorities are now working towards new regulations allowing for wireless spectrum reuse by unlicensed users [4]. These regulations allow the unlicensed secondary users (SUs) to access the spectrum as long as needs of the licensed primary users (PUs) are satisfied.

One of the emerging use cases of Cognitive Radios is 5G. 5G networks utilize the concept of Licensed-Assisted-Access (LAA) [5], which basically says that, in order to get very high rates (and due to spectrum scarcity), the network does not just rely on the licensed part of the spectrum, but also uses part of the unlicensed and shared spectrum. However, using the shared spectrum should be done carefully to avoid interference between different parties. For example, 5G should be able to provide cellular access to IoT networks, which are mainly private networks that consist of many devices which should co-exist. Since they are supposed to use the same shared spectrum as that of WiFi, then in such a scenario the WiFi access points could act as the primary users, while the IoT devices are secondary ones. In such a network, powerful nodes (which have multiple antennas for example) should use their capabilities to avoid interfering with the less powerful ones.

In order to use the Cognitive Radio (CR) in practice, many challenges need to be considered. Since one of the biggest goals is ensuring the integrity of the PU transmissions, all components of the CR cycle are developed such that this goal is attained [6]. For example, various spectrum sensing and sharing techniques have been developed, each with varying levels of complexity and efficiency[7]. In addition, different spectrum management policies have been envisioned which are better suited for different CR scenarios. For example, an overlay access policy allows

[*] Corresponding author

Email addresses: arsany.guirguis@alexu.edu.eg (Arsany Guirguis), mkarmoose@ucla.edu (Mohammed Karmoose), karim.habak@cc.gatech.edu (Karim Habak), ymustafa@alexu.edu.eg (Mustafa El-Nainay), moustafa.youssef@ejust.edu.eg (Moustafa Youssef)

a SU to access a spectrum only if a PU is not detected. In contrast, an underlay access policy would allow the coexistence of PU and SU transmissions provided that the level of interference incurred at the PU is not excessive. Developing such systems is therefore by no means an easy feat, and many practical challenges arise which makes designing such systems a challenging task [7, 8].

In the context of CRNs, communication links are established between SUs which do not necessarily have a direct communication link between them. One of the most challenging functionalities in CRNs is routing. Inherent to the nature of CRNs, a typical routing protocol must be able to provide the following: 1) the route from the source to the destination, 2) the wireless channels to be used along this route, while 3) adhering to the allowed interference limits imposed by the licensed PUs. Developing such protocols with adequate performance has therefore attracted the attention of a large number of researchers [9, 10], investigating various routing techniques and approaches [11, 12, 13]. One design approach to such protocols is cross-layer design for routing [14, 15]. By providing the routing functionality with information about the underlying physical and MAC layers, better-educated routing decisions could be made which promise to achieve better spectrum utilization and routing performance [16]. This is particularly true in the case of CRNs since interference management has been a primary topic of investigation in the literature of wireless communications [17].

However, one common assumption in most of the *existing CRN routing protocols* causes a critical limitation of the routing process and hinders CRNs to perform suitably for real-world applications. Specifically, the current state-of-art cognitive radio routing protocols assume that: *SUs inside the interference range of an active PU cannot utilize the PU licensed channel during its active periods, except with very low-power transmissions* [18]. This, in fact, is a needlessly limiting co-existence condition. For example, the United Nations announced International Telecommunication Union Radiocommunication (ITU-R) regulations indicate that [19]: "*stations of secondary service shall not cause harmful interference to stations of primary services to which frequencies are already assigned or to which frequencies may be assigned at a later date*". Thus, the mentioned assumption, followed by the existing routing protocols, is unnecessarily constraining. Such assumption leads to (1) wasting possible communication opportunities, (2) relying on relatively suboptimal routes due to PUs activities, and (3) being unable to construct routes in scenarios where PUs are highly dense and/or of high activity. The question then arises: "Can SUs *utilize* licensed channels that are occupied with *active* PUs, *without* interfering with them?"

A partial answer to this fundamental question is found in the literature of physical layer communications. Cooperative communications [20, 21] have been extensively studied as a mean to enhance the reliability of communication links. By allowing transmitters/receivers to cooperatively encode/decode transmitted/received data, communication links can be rendered less vulnerable to negative communication environments such as poor communication channels and excessive interference levels. One particular technique is Cooperative Beamforming [22], where precoding is employed cooperatively by a set of transmitters to null out transmission at particular directions of interest, while simultaneously combating poor channel conditions to increase the reliability of transmission links. Cooperative beamforming can, therefore, be well-fitted for overcoming the aforementioned assumption: by allowing SUs to cooperatively null-out transmission at the directions of active PUs while maintaining secondary transmission at the required level. Thus, physical layer communication mechanisms provide the means for *undercover* communication that enables SUs to communicate without interfering with PUs. Although being well-established in the literature, cooperative beamforming has been mainly considered in Cognitive Radio setups which involve **single-hop** transmission links[23, 24, 25, 26]. First steps have been taken towards developing multi-hop-based cooperative beamforming schemes in [27]. DZP [28] is the first to consider using cooperative beamforming in the route discovery process. However, it only utilizes cooperative beamforming to provide a ***route maintenance*** schemes that could be augmented on top of existing routing protocols, without fully integrating it into the routing process.

In this paper, we investigate more the possibility of penetrating the PUs regions without interfering with them. Specifically, we investigate how collaborative beamforming techniques can be employed in CRNs in a **multi-hop** context to enable concurrent primary and secondary transmissions in the same geographical area. We develop a cross-layer based layer 3 routing protocol, *Undercover*, that utilizes the available location information to route data towards their destinations. Our proposed protocol enables SUs to construct cooperative groups that employ collaborative beamforming to either enhance the attained secondary throughput or to null out reception at nearby PUs, thus allowing for simultaneous use of the spectrum by both PUs and SUs. Considering the elevated levels of interference commonly exhibited by cooperative transmissions, we allow our protocol to be interference-aware by penalizing the routing situations which may incur excessive interference on on-going secondary flows. Our proposed protocol is considered as a local one: each node is provided only with the necessary information about its neighbors and their wireless channel condition. This allows for a fully distributed implementation of the protocol. The introduced idea in this paper also applies to collaborative networks, which were the primary focus of the recent Spectrum Collaboration Challenge (SC2) proposed by DARPA in 2016 [29]. In this challenge, "*competitors will reimagine a new, more efficient wireless paradigm in which radio networks autonomously collaborate to dynamically determine how the spectrum should be used moment to moment*". Accordingly, nodes of a network have the incentive to collaborate with each other, as well as with the nodes of the other networks to share the spectrum in a fair manner.

We evaluate the performance of *Undercover* via NS2 simulations [30] under various network conditions. We compare *Undercover*'s achieved performance against other cognitive radio routing protocols. Our experiments show that *Undercover* achieves up to 250% increase in goodput compared to other protocols. Moreover, results show that the group selection process overhead can be controlled using our proposed heuristic; this allows using *Undercover* practically.

In summary, the contributions of this paper are as follows:

1. Designing *Undercover*, a multi-hop routing protocol in which we *integrate a cooperative communication scheme with the layer 3 routing process*. Through utilizing this scheme, primary and secondary transmissions can co-exist since SU transmissions are nulled in the PUs directions while fortified at other secondary destinations. For this, we propose a routing metric, for which the tradeoff between increasing throughput and decreasing the inter-route interference is studied and formulated.

2. Since the brute-force approach to search for the best neighbors to collaborate in a cooperative group has exponential complexity[1], we develop a heuristic algorithm for the practical implementation of our idea to control the search space of choosing those nodes, and hence control the group construction overhead. The proposed group selection heuristic provides candidate cooperative groups of SUs in a reduced amount of time that allows for practical usage. This is validated by our evaluations as shown in Section 6.

3. Evaluating *Undercover* performance using NS2 simulations against two other state-of-the-art CRNs routing protocols.

The rest of the paper is organized as follows. Section 2 presents some background material in addition to our related work. We then describe our system model and give an overview of the proposed routing protocol in Section 3. We present our routing objectives and propose our routing metric in Section 4. We then describe the whole routing process in a detailed way in Section 5. Section 6 evaluates our proposed routing protocol, and Section 7 concludes the paper.

2. Background and Related Work

The wireless spectrum is a broadcast medium where different radio waves can constructively or destructively collide and affect one another. The research community has thoroughly investigated utilizing these effects to alter the signal transmission characteristics by allowing multiple users to transmit carefully selected signals simultaneously and collaboratively [24, 20, 21]. These investigations have led to enhancing communication throughput [31], enabling simultaneous non-interfering transmissions [32], and reshaping the transmission beam [22].

There have been some attempts to exploit cooperative communication in the context of **conventional wireless networks**. These attempts were led by Khandani et al. who analyzed the energy saving benefits introduced by employing cooperative diversity [33]. However, despite the sound theoretical framework, their proposed algorithms are not suitable for real-time communication. With similar energy efficiency goals, other researchers proposed more real-time suitable mechanisms in static environments with wireless shadowing[34, 35] as well as in multi-path environments[36, 37]. Other researchers focused on enhancing the network throughput in case of having multiple concurrent flows [31, 38, 39].

The significant performance gains introduced by cooperative communication have motivated some researchers to employ its techniques in the context of **cognitive radio networks** [40, 41, 42]. For instance, Ding et al. use cooperative diversity to transmit data with higher capacity to maximize throughput [41]. In addition, Sheu et al. proposed a cooperative routing protocol in [42] that enhances the end-to-end throughput. Also, some approaches used cooperative communications in spectrum sensing [43]. Unfortunately, none of these approaches (1) offer new communication opportunities, (2) mitigate the effect of having active primary users on the secondary network, (3) address the inter-path interference problem which increases as a result of cooperative transmission.

One of the intensively studied cooperative communication techniques is cooperative beamforming. This technique relies on sending precoded versions of the same data to reshape the signal beam producing transmission nulls at certain spatial directions [22]. A direct consequence of employing cooperative beamforming is to allow for spatial multiplexing of concurrent transmissions of multiple nodes [21]. Fortunately, cooperative beamforming provides the means for hiding secondary user communication from primary users and avoiding interfering with primary user communication. This opportunity was considered by a small number of attempts, like [26, 44], which utilize beamforming by developing MAC layer protocols that maximize the received signal-to-noise ratio (SINR) among SUs with different power constraints and the QoS requirement of PUs. Moreover, these protocols deploy beamforming for **one** hop only.

Our previous work [28] considered using beamforming in the routing layer. However, we only proposed a route maintenance mechanism to alleviate the need for route re-establishment upon the detection of a PU which limits the usefulness of beamforming. Aktar et al., in [45], proposed

[1] In the brute-force technique, a node tests all possible groups, with all sizes, with its neighbors using all possible combinations. As a result, this algorithm has an exponential complexity. This is detailed in Section 5.

a routing protocol which utilizes beamforming. However, in this work, each node is assumed to be equipped with an antenna array (multiple antennas) and therefore is able to do beamforming locally. In contrast, we assume nodes with single antennas. Our intent behind this work is therefore to investigate the use of cooperative beamforming, and how it can be used to give conventional nodes beamforming capabilities while mitigating interference effects.

3. System Model

In this section, we present our system assumptions and then provide a brief overview on the proposed routing protocol.

3.1. System Assumptions

Throughout this work, we assume the presence of a CRN which consists of a set of stationary PUs, that are licensed to use the spectrum according to their data delivery requirements, and stationary SUs. We further assume that each SU knows its own location and the location of its direct neighbors. Both assumptions are common in many CRN scenarios such as TV white space-based CRNs[46]. Furthermore, a node can estimate its location using any of the current localization systems, including GPS [47] or cellular mobile-based systems [48]. Moreover, a sender can obtain the location information of the ultimate destination via out of band services that map node addresses to locations, or have it disseminated through the network. Knowing the location of the final destination is a common backbone assumption in most of current location-aided routing protocols [9, 13, 49]. We assume also that SUs are able to sense and detect PUs activities [50]. A set of stationary SUs are allowed to use the spectrum in a manner that does not jeopardize the integrity of the PU transmissions, with maximum transmission power of P_T for each SU. Since harmful interference should not exist at PUs and assuming that the primary network adopts an overlay transmission policy, a SU is not allowed to transmit except in two cases: 1) a PU is not currently active, in which case a SU is permitted to occupy the spectrum for a period which terminates with the reoccupation of a PU to the spectrum, or 2) concurrently with the PU only if a SU is able to employ cooperative beamforming. It is key to mention here that interference must be avoided at the receiving end of the primary link. We assume also a slow fading multipath wireless channel, in which a channel coefficient is constant over a period of T_c. A short T_c would require frequent estimation of the channel coefficients, while a long T_c alleviates this requirement.

Primary receiver detection is a standalone problem in the literature of CRNs, whose solution depends on the nature of PUs. A lot of research efforts were directed towards detecting transceivers, either passive [51] *e.g.*, TV or active [52, 53]. In addition, channel estimation between transceiver pairs cannot be implemented in a straightforward manner in CRNs. This is due to the fact that the secondary network does not necessarily know the structure of pilot signals employed by the primary network. In [28], primary receiver detection was done via overhearing and decoding reply messages sent by the receiving nodes, such as ARQ or CTS packets. In this work, we assume that PUs are separately identified by each of the SU nodes in the network. This gives the SU identifying information of the PU, such as the MAC address of its equipped NIC card. Moreover, we assume that some level of *offline* cooperation exists between SUs and PUs so that SUs can know the structure of pilot signals sent by PUs (*e.g.*, SUs are aware of the communication standard employed by the primary network). SUs are assumed to communicate control packets over a dedicated Common Control Channel (CCC) such as the 2.45GHz ISM band [54].

3.2. Routing Protocol Overview

We assume that our protocol will be deployed in a cognitive radio network that employs an overlay transmission policy. In this network, SUs are not allowed to simultaneously transmit with PUs except via cooperative beamforming. Our proposed algorithm operates in a greedy manner, as we describe next. When a node wishes to send a data packet, it broadcasts a route request (RREQ) packet to all of its neighbors and waits for replies. Neighboring nodes check their relative distance to the destination node, and only those closer to the destination than the source are considered as "potential" relay nodes and are eligible for sending a route reply. As per conventional greedy routing protocols, each potential relay considers all of its direct neighbors (other than the source node) as possible next hops for this relay node. However, instead of calculating **one** metric value to represent the link quality between the relay node and a particular next hop, it calculates **multiple** values of the routing metric; as different groups can assist the transmission from the relay to the next hop. Each metric value corresponds to a different transmitting cooperative group, where a cooperative group consists of the relay node, along with a subset of its direct neighbors (other than the next hop node). The relay node then chooses the highest routing metric (and the corresponding cooperative group) and sends it back to the requesting source node as a Route Reply (RREP) packet.

A source originally generating a RREQ waits for a timeout period during which it collects received RREP from neighboring nodes. Upon its termination, a node chooses the link with the highest link metric as the next hop, to which it sends an ACK packet. As the next hop receives the ACK packet, it knows that it has been decided for the forwarding of the source packet from the originating source. It then sends an Ack Reply (AREP) packet to the source. Finally, it receives the data packet and becomes responsible for disseminating it to the constructed cooperative group so that all participating nodes in this group can send it to the already chosen next hop. Such a node, in this case, is called the **group coordinator**. When the next hop receives the data packet, it repeats again the

Table 1: Mathematical Notations

Symbol	Description
P_T	Maximum transmission power for SUs
P_c	Received power by some node due to the transmission of a cooperative group
P_{int}	Interference limit of SUs
P_{FSP}	The level of interference caused by a transmitting node
h_{sd}	Channel coefficient between two nodes s and d
w_{ij}	Beamforming weights between two nodes i and j
σ_N^2	Variance of Gaussian noise that affects each node in the network
B	Available bandwidth for some particular link
C_{sd}	Achievable capacity between nodes s and d
C_{cd}	Achievable capacity between cooperative group c and node d
$\hat{C}_{ij}^{coop}$	Achievable capacity during cooperative transmission from i to j
C_{ij}^{wor}	The worst capacity achieved to deliver the packet from the group coordinator to any of the neighbors participating in some cooperative transmission
$\hat{C}_{ij}$	The effective capacity achieved through sending as a group
N_f	Number of on-going flows that are in the interference range of the cooperative group
N_n	Number of flow-carrying direct neighbors of all the participating nodes in the transmitting group
N_{min}	The value of N_n for the minimum allowable group size
d_r	Radius of the circle describing the effective interference range of some group
A	Area covered by nodes participating in some group
D_f	Density of the flows surrounding some group
D_n	Node density around some node based on the two-hop neighbor information
F_n	A rough estimate of the number of flows per node at nodes surrounding some group

same algorithm to find the next hop on the route. This procedure is repeated until the data packet reaches its final destination. At this point, a route is constructed where the next packets of the same flow follow the same route. Such a route is considered valid until the channel coefficients change (after a period T_c) or a failure is detected.

If a failure is detected in some route, due to any reason, the group coordinator node does the group search algorithm again to find a new collaborative group to relay its data. In other words, we use a generalized implementation of DZP [28] in which the size of the cooperative group can be of any number, not only two.

4. Mathematical Model

In this section, we present the proposed mathematical formulation of our proposed routing metric. First, we give a brief overview of the metrics that affect the routing decisions and the incentives behind them. Then, we describe the details of the mathematical model for each of them. Finally, the complete routing protocol metric is given.

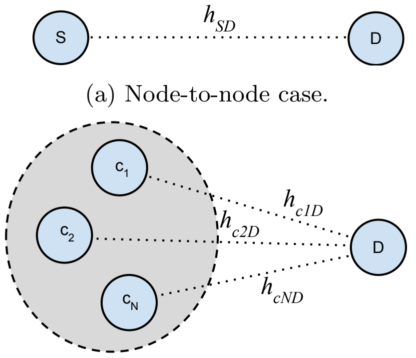
4.1. Overview

The proposed routing protocol aims at maximizing the achievable throughput across the network. Links along the route are chosen based on the maximum achievable capacity. However, in order to account for throughput calculations along the links, a node should be aware of the sources inflicting interference and therefore reducing the achievable capacity when transmitting to this node. A precise calculation of the interference level due to **all** transmitting sources is a difficult task, since it is possible for two or more nodes that are not in the interference range of the receiving node to inflict interference if they engage in a cooperative transmission phase. In other words, cooperating groups in the network can cause considerable interference levels at relatively distanced nodes that are unaware of this source of interference.

The incentive in adding the interference terms to the routing metric is to penalize the use of cooperative groups in a manner that is proportional to the amount of expected interference incurred at nearby concurrent flows. The proposed metric is a composite of two components: the first gives an estimate of the achievable throughput across the links, and the second captures the interference effect of using a cooperative group across the network. At this point, we make the assumption that the source node is **not** responsible for packet delivery to all the cooperating nodes in the next hop. It is only required to deliver the data packet to the *group coordinator* which is responsible for data dissemination among the collaborating nodes. Finally, Table 1 summarizes all mathematical notations that are used throughout this paper.

4.2. Capacity Calculations

In this section, we present the mathematical formulation of the achievable throughput across a link; this is the maximum possible number of bits that can be transmitted through a particular link. As shown in Figure 1, there are two possible links that can be utilized along a given route. Those are: 1) a node-to-node link and 2) a multi-node-to-node link. We calculate the maximum achievable throughput through utilizing each of these two types of links. The following discussion is primarily based on the basic wireless communication concepts that can be found in [20, 21]. In all cases, we assume that each node is affected by a thermal additive white Gaussian noise of variance σ_N^2.



(a) Node-to-node case.

(b) Multi-node-to-node case.

Figure 1: Existing options to transmit data. This is used to clarify throughput calculations.

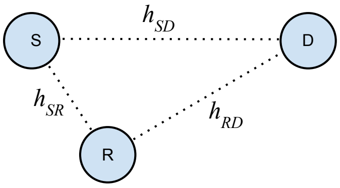


Figure 2: A scenario showing the difference between sending as a single node or as a group of two nodes.

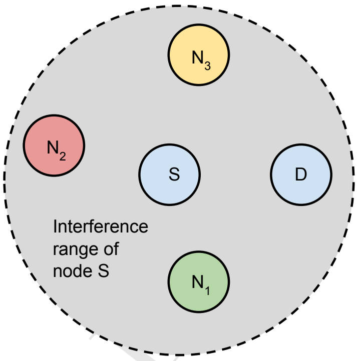


Figure 3: Illustrative example of interference effect.

4.2.1. Node-to-node link

Assume that nodes S and D are the transmitter-receiver pair of a simple link and that the transmitted signal is affected by slow multi-path fading. Let h_{sd} represents the multi-path channel coefficient between s and d. The achievable capacity between nodes S and D can be calculated as

$$\begin{aligned} C_{sd} &= B\log(1+\text{SINR}_{sd}) \\ \text{where, } \text{SINR}_{sd} &= \frac{P_T\|h_{sd}\|^2}{\sigma_N^2} \end{aligned} \tag{1}$$

and B is the available bandwidth for this particular link.

4.2.2. Multi-node-to-node link

Assume that nodes c_1 to c_N cooperatively send data to node d in the presence of a set of primary receivers $P = \{P_1, P_2, \dots, P_M\}$. It is important to note here that N should be *strictly larger* than M; the number of group members should be greater than the number of surrounding PUs. The cooperative group then applies the appropriate beamforming weights $\bar{w}_{cP} = [w_{c_1P} \dots w_{c_NP}]$ to null transmission at the primary receivers while maximizing the achievable throughput at node d. The achievable capacity in this case becomes

$$\begin{aligned} C_{cd} &= B\log(1+\text{SINR}_{cd}) \\ \text{where, } \text{SINR}_{cd} &= \frac{P_c}{\sigma_N^2},\ P_c = P_T\|\bar{w}_{cP}^H\bar{h}_{cd}\|^2 \end{aligned} \tag{2}$$

and x^H denotes the complex hermitian of a vector x, $\bar{h}_{cd} = [h_{c_1d}^H \dots h_{c_Nd}^H]^H$ is the channel coefficients vector between the nodes of the cooperative group c_1 to c_N and d, and P_c is the received power by node d due to the transmission of the cooperative group. Note that for appropriate transmission nulling, the following constraint must be satisfied

$$\bar{w}_{cP} \in \text{Null}(\mathbf{H}_P) \tag{3}$$

where Null(X) denotes the null space of matrix X and $\mathbf{H}_P$ is a matrix whose rows are $\bar{h}_{cP_j} = [h_{c_1P_j} \dots h_{c_NP_j}]$ for $j = 1, \dots, M$ where $h_{c_iP_j}$ is the channel coefficient between node i and primary receiver j. Note that the weights values can always be scaled so that the maximum transmission power constraint can be satified[2].

4.2.3. Effective Capacity

In this part, we formulate the maximum effective achievable capacity between nodes i and j. Consider the scenario in Figure 2 where, node S wants to deliver some data to node D with the help of node R as a cooperative pair noting that node R does not have the packet yet. Assume that the packet length is L and node S can send the packet to node R with capacity C_{SR}. Thus, the time needed to deliver the packet is $T_{SR} = L/C_{SR}$. After receiving, collaboration between S and R will take place, and the achievable capacity will be $\hat{C}_{SD}$; this quantity is greater than C_{SR} in almost all cases. Noting that the time needed to deliver the packet will be $T_{SD} = L/\hat{C}_{SD}$, the effective capacity in this case will be $\hat{C} = L/(T_{SD}+T_{SR}) = \hat{C}_{SD}C_{SR}/(\hat{C}_{SD}+C_{SR})$. Thus, we can see that the effective capacity term is constrained by the minimum of $\hat{C}_{SD}$ and C_{SR}. Hence, the effective capacity can be written as $\hat{C} \simeq \min(\hat{C}_{SD}, C_{SR})$. The preceding analysis can be readily extended to any multi-node-to-node transmissions from node i to j, for which the effective capacity is

$$\hat{C}_{ij} \simeq \min(\hat{C}_{ij}^{coop}, C_{ij}^{wor}) \tag{4}$$

where $\hat{C}_{ij}^{coop}$ is the achievable capacity during cooperative transmission from i to j, and C_{ij}^{wor} is the minimum capac-

[2] The sum of the absolute values of the beamforming weights is normalized to one. This means that each packet transmitted by our scheme consumes an amount of energy that is exactly equal to the amount it would have consumed if it was transmitted by any other scheme (if the maximum transmission power is used). Based on that, our protocol does not consume more energy to send a single packet compared to any other scheme of transmission.

ity achieved to deliver the packet to any of the neighbors participating in this cooperative transmission.

Discussion

As witnessed from the previous analysis, the process of disseminating the data packet to the group members prior to the transmission phase can be an enervating factor to the achievable effective capacity. However, two notes should be pinpointed: 1) despite the aforementioned obstacle, a cooperative group of such structure can yet provide better throughput than conventional point-to-point links. Consider the following scenario: $h_{SD} = 2$, $h_{SR} = 2\sqrt{2}$ and $h_{RD} = 2$. Assuming unity transmission power and noise variance, the achievable capacity through the point-to-point links S-D, S-R and R-D respectively are log(5), log(9) and log(5), according to Eq. (1). We can see that direct routes from S to D are all hindered by the bottleneck throughput of log(5). However, if nodes R and S are to cooperatively send the data to node D, then according to Eq. (4), a maximum throughput across this multipoint-to-point link is log(9), given that the achievable capacity while disseminating data is also log(9) in this case. In this example, despite having to report the packet to the cooperating node through a capacity-limited channel, the overall performance of the two-hop communication is superior to any of the point-to-point links available. 2) There are situations in which multiple nodes within the cooperative group are informed of the packet-to-be-delivered by overhearing the source transmission. This situation is commonly favorable because it alleviates the need for intra-communications among the cooperating nodes and therefore achievable capacity limits can be increased. Consider the previously mentioned scenario: if node R was to overhear the packet, then the throughput attained across the multipoint-to-point link, where S and R collaboratively send to D, is log(9) in contrast to log(5) attained over conventional routes. This means that the provided analysis gives the worst case scenario, while practical scenarios show the possibility of getting better values for network throughput.

4.3. Interference Calculations

In this section, we study the interference results from sending data through a cooperative group. This interference can be categorized into two types: (1) interference on the group itself due to its neighbors and (2) interference due to the group on the neighboring nodes.

4.3.1. Interference due to neighbors

As was previously mentioned, transmitting nodes in the same interference range are able to use the spectrum in a contending fashion, which is generally resolved by multiple access techniques such as CSMA/CA. The number of direct neighbors contending for the spectrum directly affects the average achieved capacity for a transmitting node. This is demonstrated in Figure 3 where node S wants to communicate with node D in the presence of three flow-carrying nodes in its interference range. Let C_{SD} be the achievable capacity along the direct link between the two nodes, S and D, in the presence of no interfering sources. Assuming the fair distribution of the time allocation for the medium among the four nodes, then the best achievable capacity between S and D in the presence of the interfering nodes is $C_{SD}/4$. The same argument holds in case a cooperative group is utilized instead of a single transmitting node. This concludes that the number of flow-carrying nodes in the interference range of the source node/group is inversely proportional to the average achievable capacity over the transmission link.

4.3.2. Auto-interference due to cooperative groups

A cooperative group inflicts a relatively higher level of interference on nearby nodes. Moreover, the effective interference range of a cooperative group is larger than that of a single node; this imposes extra difficulties in the design of an efficient interference-aware routing protocol. The reason for this is that conventional contention-handling protocols are oblivious to these out-of-range transmissions and thus have no control over them.

In order to alleviate this shortcoming, the proposed routing metric allows each transmitting node/cooperative group to keep track of the interference inflicted by its own transmission. Depending on the interference level, the use of this node/ group is penalized which affects the routing decision. A node should keep track of the on-going flows in its neighborhood; each node reports its witnessed flow IDs to its neighbors. Based on the received information from all neighbors along with neighbors known locations, a node can build a statistical model of the flow density in the surrounding area. We can estimate the area including the neighboring nodes by a general polygon whose vertices are the neighboring nodes. Assuming the obtained estimate area is A, the flow density is thus

$$D_f = \frac{\text{Total number of distinguished flows}}{A} \tag{5}$$

In order to determine an estimate for the number of on-going flows that are affected by the transmission of a cooperative group, the effective interference range of the group should be also estimated. For simplicity, we assume that the far transmissions are affected by Free-Space-Path (FSP) loss. The level of interference caused by a transmitting node with a power P_T at a node situated at distance d can be approximated by

$$P_{FSP} = P_T \frac{c}{d^2} \tag{6}$$

where c is the free space path loss and its value depends on the used frequency. Given an interference threshold of secondary nodes P_{int}, a cooperative group consisting of N nodes can then calculate its effective interference range which is defined as: *the geographical area in which a secondary node is -in the worst case- affected by an interference power greater than P_{int}, given the assumed FSP*

Table 2: Acceptable percentage increase in N_n for different group sizes relative to minimum allowable group size. The presented values are based on Rayleigh channel model with unit variance.

Group Size		2	3	4	5	6	7	8	9	10
Accepted	1 PU	85.6	148.8	197	236	268.9	298.2	321.7	342.2	362.1
% Inc. in	2 PU		36.3	75.5	111.5	145.4	174.1	199.1	220.7	241.8
N_n	3 PU			20.3	44.2	71.3	97.4	122.1	145.5	167.4

model. Assuming the worst case scenario (perfectly coherent addition of transmitted signals at any given point), the received power at any point (x, y, z) in space due to cooperative group transmission is given by

$$P_r(x,y,z) = cP_T \sum_{i=1}^{N} \frac{\|w_i\|^2}{d_i^2} \tag{7}$$

where d_i is the distance between node i in the group and the point (x, y, z). For ease of calculations, we approximate the effective transmission range of the cooperative group as if a single node situated in the center of the group, and transmitting with a power equal to $P_T \sum_{i=1}^{N} \|w_i\|^2$. Accordingly, the effective interference range is a circle of radius

$$d_r = \sqrt{\frac{cP_T \sum_{i=1}^{N} \|w_i\|^2}{P_{int}}} \tag{8}$$

This means that all nodes that are within d_r unit length from the center of the group are assumed to be affected by an interference greater than P_{int}; these are the affected SUs by the construction of this group. The expected number of affected on-going flows N_f can now be calculated by $N_f = D_f \times \pi d_r^2$.

4.4. Proposed Metric

Based on the what is discussed, the link metric between nodes i and j is formulated as follows:

$$LC_{ij} = \frac{\hat{C}_{ij}}{N_n + \beta(N_f - N_n)} \tag{9}$$

where C_{ij} is the maximum achievable capacity between node i and j among all possibilities of transmission (either a single transmitting source or possible cooperative groups), N_n is the number of flow-carrying direct neighbors of all the participating nodes in the transmitting group, N_f is the number of on-going flows that are in the interference range of the cooperative group (this is equal to 0 for a single transmitter), and β is a design parameter to alter the altruistic/egoistic behavior of the cooperative group.

4.5. Discussion

As *Undercover* is a CRN routing protocol, we should take care of some practical issues [8]. Sharma et al., in [7], offer solutions for a wide range of practical CRNs issues. These include, for example, channel uncertainty, and CR transceiver errors. As for channel uncertainty, one conservative solution is to increase the value of P_{int} to accommodate errors in estimating the channel interference. A more practical solution is to model channel uncertainty in such a case and control the value of P_T accordingly.

5. Implementation Details

This section gives some practical and implementation details of the whole process. First, we present some practical issues that we consider for our routing protocol. Then, we give the details of the information exchanged among the nodes to serve the routing process. Finally, we present the flowchart of our algorithm along with an example that highlights how *Undercover* works.

5.1. Practical Considerations

A node searches for the best link construction by calculating the link metric for all possibilities of cooperative transmissions. These possibilities are all based on the inclusion of direct neighbors of the potential relay node in a possible cooperative group. Assume the relay node has N direct neighbors, then calculating the metric for all possible combinations of groups is $\mathcal{O}(2^N)$. However, if we assume that the node calculating the link metric is in the interference range of M active PUs, then any allowed transmission should include at least $M + 1$ cooperative nodes that are also in the interference range of these PUs. In other words, the number of nodes participating in the cooperative group should be strictly higher than the number of surrounding PUs ($N > M$). In this case, the search space is decreased. In the next discussion, we study the statistical characteristics of the proposed metric, based on which we reduce the complexity of the link calculation algorithm via less-probable candidates elimination.

5.1.1. Average achievable capacity

In this section, we study the statistical behavior of the maximum achievable capacity. Accordingly, some cooperation possibilities that are less probable of attaining additional gains in link metric can be eliminated. Consider that the channel coefficients and, consequently, the beamforming weights presented in Equations (1) and (2) are random variables. Based on that, the achievable capacity of a particular link is also of a statistical nature.

In fact, based on the wireless fading model and the instances of the channel coefficients, the maximum achievable capacity of a node or cooperative group links can be

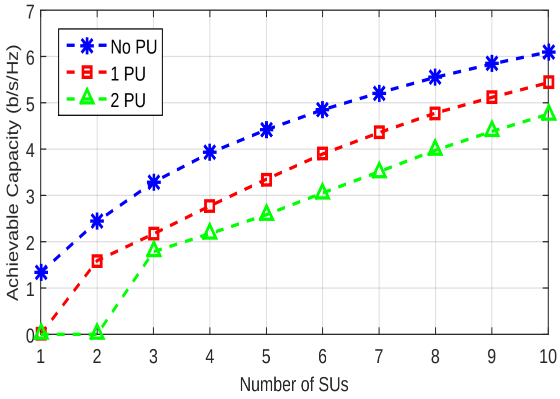


Figure 4: The effect of changing number of SUs and PUs on the average best achievable capacity assuming a Rayleigh fading model with unit variance.

foreseen to fall in certain regions with high probability. Conversely, relatively high levels of capacity are not likely to occur given a particular number of nodes in a cooperative group and a given number of active PUs. Figure 4 shows the average maximum achievable capacity based on the definitions given in Section 4.2 for different numbers of cooperating nodes and different numbers of active PUs, for a Rayleigh fading model with unit variance[3].

For example, considering the case where no PUs are available, it can be seen that moving from a single transmitting node to the case of two cooperating nodes nearly multiplies the average maximum achievable capacity of the link by a factor of 1.5.However, moving to a higher number of cooperating nodes is accompanied with less relative gains in the achievable capacity. A direct implication follows: considering the link metric construction in Eq. (9), the additional gain in the attained capacity by including an extra node in the cooperative group cannot be realized unless the accompanying increase in N_n is of less order.

Consider the scenario in Figure 5 in which all the presented nodes carry different data flows. Assume that node 0 calculates the link metric considering all possibilities of cooperation. Based on the previous discussion, using node 1 as a cooperative pair will limit the value of the link metric since it will increase the value of N_n by three in addition to the four originally added by node 0 - an increment of 75% in N_n. In contrast, using node 10 as a cooperative pair will not increase N_n beyond four. Therefore, using node 10 will provide better performance than using node 3. The same argument can be constructed for cooperative groups with a larger number of nodes. It is important to highlight that a node that is not relaying data for any of the existing flows is not affected by the incurred interference and, therefore, is not included in the calculations of N_n.

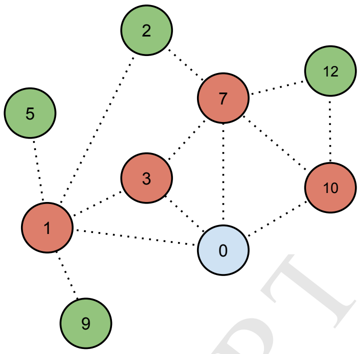


Figure 5: An example of network scenario that shows the effect of choosing different groups on the achievable capacity. Note that PUs are not shown here since the focus of this figure is on the secondary cooperative group. But, the same argument holds in the presence of PUs.

5.1.2. Elimination heuristic

Based on the mentioned discussion, our protocol is devised to control the search space by excluding the links that are less probable to score maximum link metrics. Each node estimates a local version of the node density based on the two-hop neighbor information it obtains from periodic hello packets as $D_n = N/A_N$, where N is the number of one-hop and two-hop neighbors of the node and A_N is the area of the polygon whose vertices are the farthest of these neighbors from the node. A rough estimate of the number of flows per node F_n can then be calculate mathematically as $F_n = D_f/D_n$. A node can now calculate an estimate of the number of flows that would be affected by the inclusion of N nodes in a group as $N \times F_n$. According to this estimate, and based on the capacity gains shown in Figure 4, a node then decides to exclude group formations that exceed a certain number of collaborating nodes. A node calculates the factor

$$\text{Factor} = \frac{N \times F_n - N_{min}}{N_{min}} \qquad (10)$$

where N_{min} is the value of N_n for the minimum allowable group size (*i.e.*, $N_{min} = M + 1$). Based on comparing this factor to the corresponding threshold in Table 2[4], the algorithm excludes groups of certain sizes from the search process. We calculate values of Table 2 as follows: for each row of the table, an entry under group size j is the percentage of increase in the achievable capacity relative to the achievable capacity of the minimum allowable group size. For example, the entry under group size seven in the second row is 174.1%. This means that the capacity achieved by groups of size seven is on average 174.1% greater than the capacity achieved by groups of size three (which is the minimum allowable group size). Thus, in this case, if the value of the calculated factor is higher than this threshold,

[3] Note that the achievable capacity, in this case, is measured by bits per second per Hertz. This value should be multiplied by the used bandwidth to convert it to bits per second (bps).

[4] Note that the values given in the table are for a particular channel model and statistics (Rayleigh channel model with unit variance). This would differ for other channel conditions. Values of this table depends on plotted values in Figure 4. Providing analytical expressions for the thresholds is a direct extension of this work.

this indicates that the capacity gain of using larger group size will be outweighted by the pernicious interference on the currently active flow. This result in rejecting groups of seven nodes in this case, putting an upper bound on the maximum group size.

5.2. Information Exchange

In order to allow for the calculation of the achievable capacities across links and the estimation of the on-going flows, a node should be provided periodically with the following information:

1. its direct neighbors and the channel coefficients between the node and each neighbor,
2. the ID of the on-going flows witnessed by each of the neighbors,
3. the primary receivers that are detected by each of the neighboring nodes and the estimated channel coefficients, and
4. its 2-hop neighbors and the channel coefficient between each of these neighbors and its intermediate 1-hop neighbors.

We allow nodes to send periodic "Hello" packets to their neighbors. A Hello packet consists of: 1) the ID of the generating node, 2) the IDs of the neighboring nodes, along with the channel coefficients between the node and each of its neighbors, 3) the IDs of the identified PUs, along with the estimated channel coefficients from the node to them, and 4) the IDs of the flows witnessed by the node[5]. Given this information, a node can calculate the three mandatory estimates (D_n, D_f and F_n) as discussed in Section 4 for link metric calculations.

We believe that *Undercover* can tolerate the loss of the information that are required to be exchanged among the SUs. If some information are missing, a node may not be able to construct all possible groups. Although this may lead to a sub-optimal route, we believe that *Undercover* can get the best route given the available information. The worst case happens when one sender node cannot construct any cooperative group. In this case, *Undercover* converges to LAUNCH. We believe that, even in this case, our protocol can get a route from the source to the destination. We also capture the performance of LAUNCH in Section 6.

5.3. Route Discovery

A source sends a Route Request (RREQ) packet which contains the IDs of the source and destination nodes. If closer to the destination, a neighboring node receiving the RREQ decides whether to send a Route Reply (RREP) packet or to discard the request according to the algorithm described in Figure 6. Denote this neighboring node with X. Then this algorithm is dissected into three phase:

- Potential Next Hops: First, $N_{\min}$ is set to $M + 1$ (where M is the number of primary users which have node X in their interference range). The algorithm sweeps over all possible next hops for X. If this next hop is closer to the destination, the algorithm moves to the second phase. Else, this next hop is discarded and the following next hop is tested.
- Maximum Group Size Determination: The maximum group size is determined such that the factor in Eq. (10) is less than the corresponding threshold from Table 2. At this state, the minimum and the maximum groups' sizes are obtained. The algorithm then creates a list of all possible groups which have a size within the allowable size range for the current potential next hop. Each of these groups then enters the last stage.
- Link Metric Calculation: The algorithm calculates the link metric in Eq. (9) for all possible transmission groups between node X and the possible next hop. The maximum metric for node X (and the corresponding transmission group) are then stored. Finally, the algorithm returns the maximum link metric, along with the ID of the neighbor node that corresponds to this metric value.

Once a node finishes the route reply algorithm, it sends a RREP packet which consists of 1) the source ID originating the RREQ, 2) the destination ID of the RREQ, and 3) the ID of the next node generating the RREP.

5.4. Example

Figure 7 shows a running example of our routing protocol. Suppose an intermediate node (node 0) has a data packet that it wishes to forward to a destination (node D). First, node 0 sends a RREQ to all its direct neighbors, which in this case are nodes 1 and 2 (Figure 7a). Each of these nodes applies the route reply algorithm to choose the best group to cooperate with. Consider node 1. With node 7 as a next hop, node 1 calculates the link metric for all possible transmission groups: group (1,2) (Figure 7b) and group (1,2,5) (Figure 7c). Node 1 repeats the calculations with node 9 as a possible next hop, with the only allowed group in this case: group (1,7) (Figure 7d). Node 2 applies the same algorithm. Upon finishing, all neighbors of node 0 send back their RREPs (Figure 7e). Finally, node 0 chooses one of these neighbors to be the next hop (Figure 7f) based on the link metric values that it has received. This routing process continues at each intermediate hop until the destination is reached.

6. Performance Evaluation

In this section, we evaluate the performance of our proposed routing protocol using a cognitive extension of NS2 [30],[55]. Table 3 summarizes the simulation parameters used in our evaluation. We model the PUs activity as an

[5] This can be obtained by the node by examining the header content of the packets belonging to the flow.

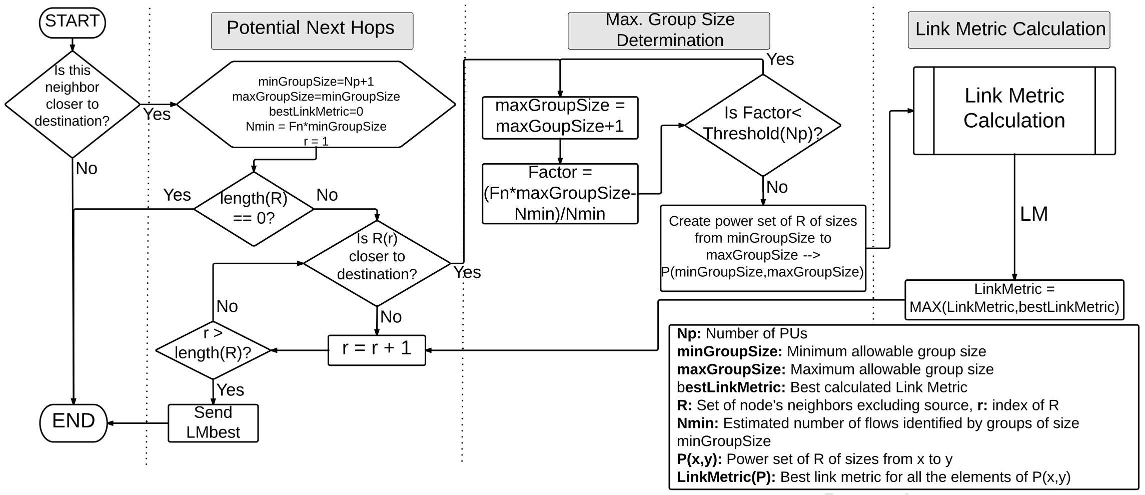
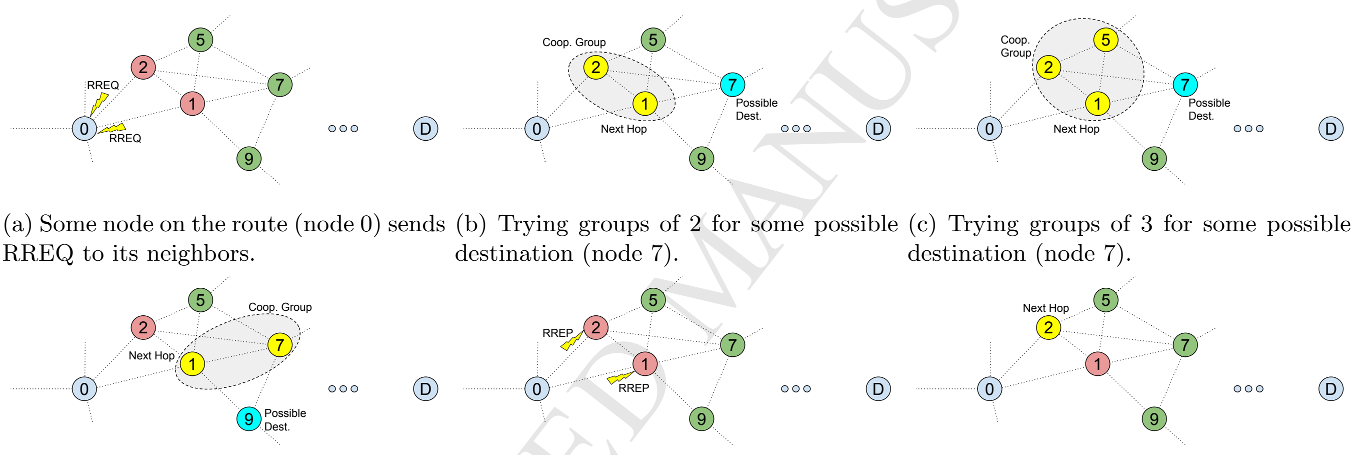

START
Is this neighbor closer to destination?
Yes
No
Potential Next Hops
minGroupSize=Np+1
maxGroupSize=minGroupSize
bestLinkMetric=0
Nmin = Fn*minGroupSize
r = 1
Yes
length(R) == 0?
No
Is R(r) closer to destination?
Yes
No
No
r > length(R)?
Yes
r = r + 1
Send LMbest
END
Max. Group Size Determination
maxGroupSize = maxGoupSize+1
Factor = (Fn*maxGroupSize-Nmin)/Nmin
Is Factor< Threshold(Np)?
Yes
No
Create power set of R of sizes from minGroupSize to maxGroupSize --> P(minGroupSize,maxGroupSize)
Link Metric Calculation
Link Metric Calculation
LM
LinkMetric = MAX(LinkMetric,bestLinkMetric)
Np: Number of PUs
minGroupSize: Minimum allowable group size
maxGroupSize: Maximum allowable group size
bestLinkMetric: Best calculated Link Metric
R: Set of node's neighbors excluding source, r: index of R
Nmin: Estimated number of flows identified by groups of size minGroupSize
P(x,y): Power set of R of sizes from x to y
LinkMetric(P): Best link metric for all the elements of P(x,y)

Figure 6: Flowchart of the route reply algorithm applied by each neighbor of the source node.



(a) Some node on the route (node 0) sends RREQ to its neighbors.

(b) Trying groups of 2 for some possible destination (node 7).

(c) Trying groups of 3 for some possible destination (node 7).

(d) Trying groups of 2 for some possible destination (node 9).

(e) All neighbors send their RREP to the sender node.

(f) Based on the RREPs, node 0 chooses node 2 to be the next hop.

Figure 7: *Undercover* routing scheme scenario. Although neither the PUs nor the SUs other flows are shown here, the routing scenario remains the same calculating the weights to null transmissions at PUs and reduce interference to other active flows.

Table 3: Experiments parameters.

Parameter	Value range	Nominal Value(s)
Number of PUs	2 - 16	4
Number of SUs	10 - 40	25
SU transmission range (m)	125	125
PU transmission range (m)	140	140
Number of connections	1 - 16	8
Frequency (GHz)	2.4	2.4
Effective bandwidth (Mbps)	1.5	1.5
Packet Size (byte)	128 - 1518	512
PU Activity(%)	0 - 100	20
Data Rate Per Source (kbps)	20 - 400	100
Dep. Area Side Length (m)	250 - 1000	250

ON-OFF process where the means of the exponentially distributed active and inactive periods are randomly chosen with the activity percentage shown in Table 3. PUs are uniformly distributed over the available grid. We assume the channel coefficients to be complex numbers that follow the Gaussian distribution with zero mean and unit variance [56, 57]. We assume that the SUs are randomly deployed using a uniform distribution across the grid. Each SU node is equipped with two radio interfaces, has omnidirectional antennas, and runs the IEEE 802.11 MAC protocol. The first radio is used for exchanging the control packets while the second one is used for exchanging data. The source and destination of each connection are selected randomly. We compare our protocol against existing protocols such as LAUNCH [13] and CAODV [58]. LAUNCH is a location-aided routing protocol that is designed to work in CRNs. CAODV is the cognitive extension of the popular AODV protocol [59]. We have chosen these two protocols as representatives for the local and global approaches of routing protocols of CRNs respectively[54].

6.1. Metrics

We evaluate *Undercover* using the following metrics:

1. Goodput: number of bits communicated successfully from the source to the destination per second.
2. Average end-to-end delay[6]: average time taken by packets to reach the destination from the source.
3. Routing overhead[7]: number of transmitted control packets in the routing phase.
4. Average group size: average number of nodes participating in the cooperative communication in case of using *Undercover*.
5. Routing Opportunities Gain: average number of groups such a node can construct to route through. This represents the new choices for the node to send the data through, and hence it is called "gain". Thus, Routing Opportunities gain is given by:

$$\text{Routing Opportunities Gain} = \frac{\text{Number of groups a node can construct}}{\text{Number of node's neighbors}} \tag{11}$$

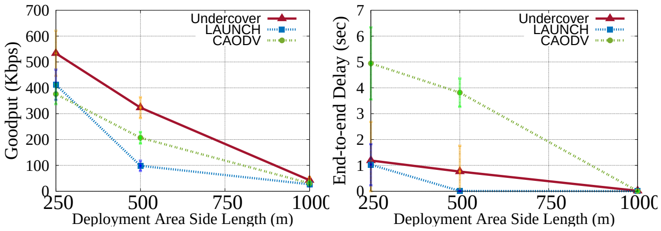
6.2. Experimental Results

In this section, we present the results we have got through the NS2 simulations. First, we evaluate the overall performance of *Undercover* as compared to LAUNCH and CAODV. Then, we investigate the performance and results of the group construction algorithm. Finally, we assess the routing overhead added by *Undercover*.

6.2.1. Protocol Performance

Changing Network Density.

- Changing Deployment Area Size: Figure 8 shows the effect of changing the deployment area size on the performance metrics. Increasing the square deployment area side length decreases the SUs density. As the SUs density decreases, both goodput and end-to-end delay decrease too. This happens since the available opportunities to relay data from the source to the destination decrease, decreasing the packet delivery ratio and hence the goodput. However, *Undercover* always outperforms CAODV and LAUNCH (in terms of goodput) since it has a higher probability of overcoming the presence of the surrounding PUs and relay data successfully. Finally, it is important to mention here that using the small area (250m×250m) for all of the next experiments was driven by the need to test *Undercover* in a dense environment to allow for groups formation.



(a) Goodput (b) Average End-to-end Delay

Figure 8: Effect of changing deployment area size on network performance.

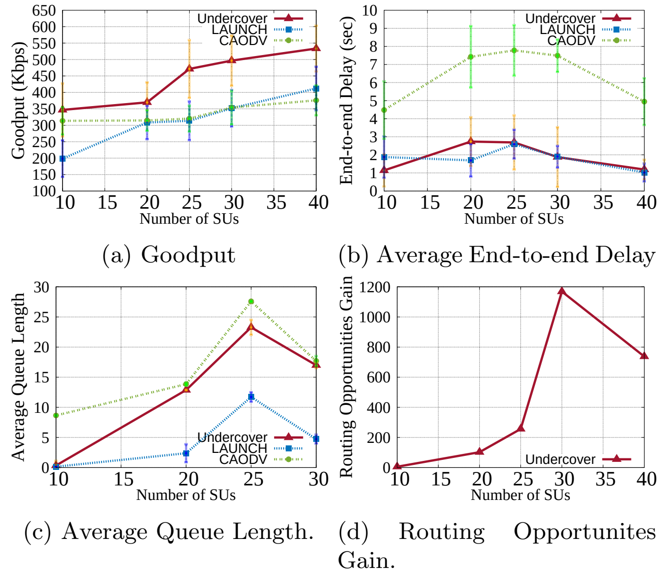
- Changing Number of SUs: Figure 9 shows the advantage of using *Undercover* over CAODV and LAUNCH in terms of the achieved goodput and average end-to-end delay as the SUs' density increases. Several conclusions can be drawn from this figure. Generally, goodput increases with the increase of SUs' density. This happens since better routes can be found as the number of SUs increases. Also, we can see that *Undercover* outperforms both of LAUNCH and CAODV in terms of goodput especially at the high density of SUs. This is due to the ability of *Undercover* to construct better and larger cooperative groups with this increase[8]. This leads to more reliable delivery of packets to the destination. Our last note for Figure 9a is that in high-density networks, LAUNCH beats CAODV since the former routing technique takes into consideration the minimum delay and the PU presence.

  Concerning Figure 9b, we can see a bell-like shape with a peak at some point in the graph for all protocols. This behavior can also be observed in Figure 9c and can be attributed to the following reason: there are two competing factors that affect the queues length and hence the delay. The first one is the number of transmissions between the sender and the receiver which affects the queuing delay at each node on the route. This factor increases with the increase of the number of SUs as shown in Figure 9a. The second one is the advantage of finding better routes as the number of SUs increases. This decreases the total end-to-end delay as the probability to interfere with a PU decreases. At the first part of the graph, the first factor beats the second one. Thus, increasing

[6] It is important to note that the group construction time and overhead are included in this metric. Thus, using this metric, we can observe whether using beamforming has an advantage, in terms of total delay, or not.

[7] This metric can be used in evaluating the energy consumed by the routing protocol. As discussed in Section 4, the energy consumed to transmit a single packet by *Undercover* is the same that is used by any other scheme. Based on that, the energy consumed by any protocol depends only on the number of transmitted packets, which is defined by this metric.

[8] Groups of size *eight members* were attained in some experiments.



(a) Goodput

(b) Average End-to-end Delay

(c) Average Queue Length.

(d) Routing Opportunites Gain.

Figure 9: Effect of changing number of SUs on network performance.

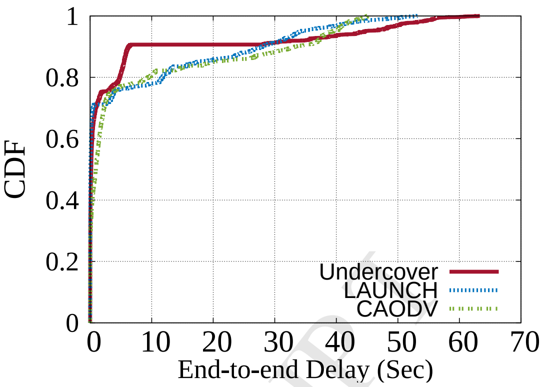


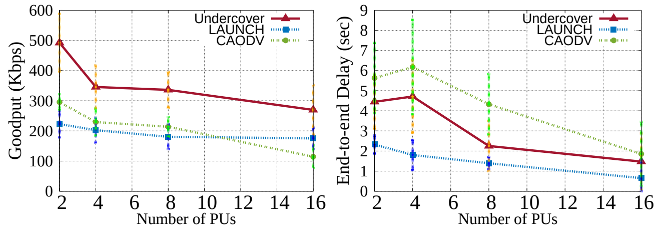
Figure 10: CDF of end-to-end delay when using the default parameters of table 3.

SUs' density increases the backoff delay and therefore the number of retransmissions due to the congestion at the MAC layer. This increases the queue length at each node (Figure 9c) increasing the total end-to-end delay. However, in the second part of the graph, the effect of the second factor dominates that of the first one. That is, the advantage of finding better routes (and hence getting a better experience for data delivery) dominates the counter effect of increasing queuing delay (due to SUs number increase). Therefore, queues lengths decrease at each node as shown in Figure 9c. This leads to the decrease of the average end-to-end delay as the SUs' density increases.

We can see that CAODV always has a higher delay than *Undercover* and LAUNCH. This happens since the last two protocols have the ability to react better with the presence of PUs, either by constructing cooperative groups or by the channel switching used by LAUNCH. On the other hand, the experienced delays for *Undercover* and LAUNCH are nearly equal in almost all cases since both protocols try to avoid interfering PUs by nulling transmission at them or by transmitting on different channels. Although the average end-to-end delay of *Undercover* seems to be higher than that of LAUNCH in some cases, we can get from Figure 10 a more detailed message. This figure draws the cumulative distribution function (CDF) of the delay experienced by all packets. This shows the distribution of the individual delay for each packet alone. We can see that in the case of using *Undercover*, a small portion of the packets suffers from the excessive delays introduced by route construction and that routes are stable enough for more than 90% of the packets. On the other hand, we can see that less than 80% of packets transmitted using LAUNCH have the same small delay. Thus, we can conclude that most of the packets routed using *Undercover* incur very small delay compared to LAUNCH even if the latter exhibits lower average end-to-end delay. According to our simulations, the groups' construction phase takes some time which makes about 10% of the packets on average have a high delay[9]. This conclusion can be applied too to the rest of the average end-to-end delay figures in this section.

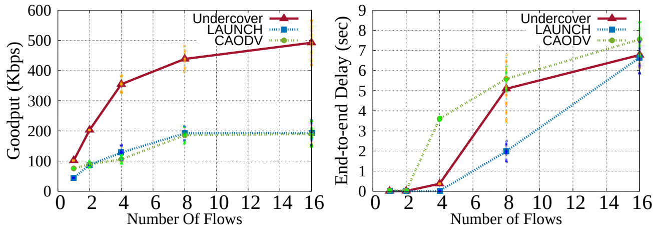
Figure 9d shows the effect of changing the number of SUs on the routing opportunities gain. This figure mainly compares between *Undercover* and LAUNCH since the former protocol converges to LAUNCH in the case of not using groups. Thus, Figure 9d shows the advantage of using cooperation with neighboring nodes. We can see that using cooperative groups gives more routing opportunities in all cases. This leads to the discovery of better and more stable paths, and this fact illuminates the value of nodes' collaboration used by *Undercover*. We can see that opportunities increase with the increase of the number of SUs since more nodes exist to cooperate with. However, The decrease in the opportunities gain curve (at very dense networks) is due to the following. As the number of SUs increase, an exponential number of possible cooperative groups exist. However, as described earlier in Section 5.1, our algorithm operates in a reasonable time by exploring only a fraction of these possible group. As the number of SUs increase, the explored fraction of groups (and consequently, routing opportunities) decreases, in contrast to LAUNCH which explores all possible routing opportunities (which involve single nodes). Both factors affect the routing opportunities gain according to Equation 11. Thus, the two factors collectively lead to the decrease of the opportunities gain.

[9] This fact opens a room of improvements which can be done as a future extension of the current work.



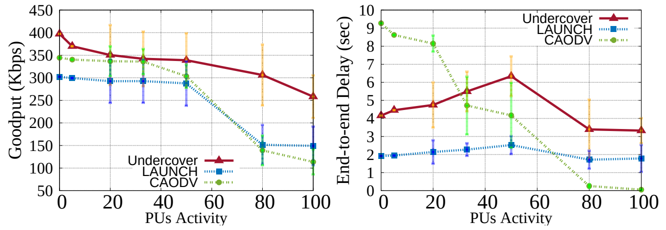
(a) Goodput (b) Average End-to-end Delay

Figure 11: Effect of changing number of PUs on network performance.



(a) Goodput (b) Average End-to-end Delay

Figure 13: Effect of changing number of flows on network performance.



(a) Goodput (b) Average End-to-end Delay

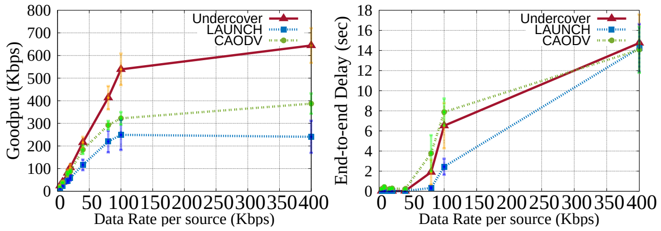
Figure 12: Effect of changing PUs activity on network performance.

Changing Number of PUs. Figure 11 shows the effect of changing the number of PUs on the goodput and the end-to-end delay for the three used protocols for comparison. It can be noted that the performance of the three protocols degrades as the number of PUs increases. Moreover, the performance of *Undercover* always outperforms that of LAUNCH and CAODV in terms of the achieved goodput (Figure 11a). Thanks to the constructed cooperative group, *Undercover* can send in many cases without interfering with PUs even if they exist and are active. However, the overhead of creating these groups in terms of construction time and interference to others can be shown in Figure 11b in which we can see that the delay for *Undercover* is higher than that of LAUNCH. We can note also in this figure that generally, the delay decreases as the number of PUs increases; this can be attributed to two factors. First, since a higher number of PUs results in a high packet loss ratio, SUs quickly empty their queues by dropping lost packets, and therefore an "artificial" decrease in queue length, and therefore average end-to-end delay, is exhibited. Second, as the number of PUs increases, only big groups (number of group members should be higher than the number of PUs) can be constructed. This limits the number of potential groups decreasing the construction time and the total end-to-end delay.

Changing PUs Activity. Figure 12 shows the effect of changing PUs activity on the performance metrics. We can conclude from Figure 12a that the goodput decreases with the increase of PUs activity. This happens since less number of packets are able to reach their destinations safely with the increase of PUs activity; this decreases the goodput with always upper hand for *Undercover* over other protocols. Moreover, we can see that *Undercover* beats other protocols in case of *zero* activity of PUs. In this case, the network is converted from Cognitive to Adhoc network *i.e.*, no PU traffic in this case. This means that CAODV converges to AODV and LAUNCH advantage in cognitive networks disappears. Cooperative groups constructed by *Undercover* in this case gives the source node the ability to send with a higher rate and hence achieving a higher goodput. Thus, Figure 12a shows the second goal of constructing cooperative groups: strengthing the transmission power to send a signal with a better quality. We can note some other facts from Figure 12b. Generally, the average end-to-end delay decreases with the increase of PUs activity. This happens due to delivering less number of packets and hence decreasing the congestion at MAC layer and the queue length at each node decreasing the total delay. We can note that at 100% activity, CAODV achieves the lowest time delay and the lowest goodput (highest loss rate). However, the delay increases at the first part of the graph for some protocols due to the time spent in constructing cooperative groups (to overcome PUs existence and activity) in the case of *Undercover* and the channel switching done by LAUNCH. But at high rates of PUs, the delay decreases for all protocols.

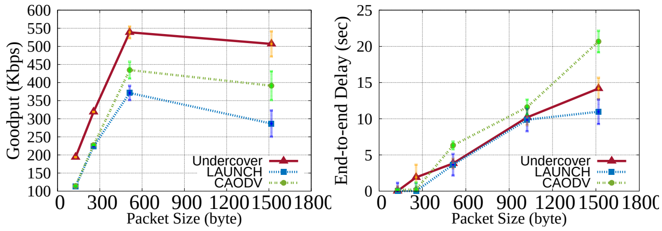
Changing Number of Flows. Figure 13 shows the effect of increasing the number of active connections on the performance metrics. From Figure 13a, we can note that *Undercover* has the best performance when there is a large number of flows in the network. Also, we can note that goodput saturates in the case of using LAUNCH and CAODV while still increasing (even with a decreased rate) in the case of using *Undercover*. This happens in the case of using LAUNCH and CAODV since there are not enough SUs to accommodate flows. The same saturation effect occurs with *Undercover*, but at a later point on the graph. This can be abstractly explained by thinking of networks operating with *Undercover* as larger networks with virtual SUs that correspond to groups. Also, increasing the number of flows increases the goodput as well[10]. This is due to

[10] This holds until the network is congested when the data generated fill the available bandwidth. At this points, the goodput saturates and then decreases.



(a) Goodput (b) Average End-to-end Delay

Figure 14: Effect of changing data rate per source on network performance.



(a) Goodput (b) Average End-to-end Delay

Figure 15: Effect of changing packet size on network performance.

the fact of successfully delivering more packets to the destination which increases the goodput value by definition. We can see that *Undercover* performance exceeds that of CAODV and LAUNCH due to the ability to deliver more packets for each single added flow, which is translated to an enhanced performance. From Figure 13b, we can see an increase in the values of delay with the increase in the number of active connections. This is due to the increase in the number of delivered packets to their destinations and hence, the increase of the delay due to congestion at the MAC layer.

Changing Data Rate Per Source. Figure 14 shows how *Undercover* beats other routing protocols for different data rates. System performance increases in terms of the goodput but degrades in terms of the end-to-end delay for higher data rates. Some of the observations mentioned previously can be noted also in this figure.

Changing Packet Size. Figure 15 shows the effect of changing the packet size on the performance metrics for all used protocols for comparison. In Figure 15a, we can see that the goodput increases to some value for packet size then decreases again. The goodput increase at the first part of the figure happens due to delivering more data to final destinations when increasing the packet size. This case is similar to that of increasing data rate observed in Figure 14. However, the goodput decreases after that since some packets cannot reach their destinations due to the introduced activity of PUs which preempt the sending process. Thus, these packets are lost and fewer data are then transmitted to their destinations safely. The last observation for Figure 15a is that: we can see that *Undercover* outperforms other protocols in terms of goodput when using any packet size. On the other side, we can note generally that the average end-to-end delay (Figure 15b) increases as the packet size increases. This happens since more time is needed as more bits are communicated.

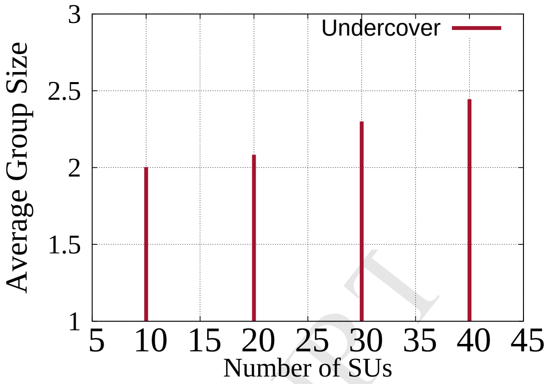


Figure 16: Average size of groups constructed by *Undercover*.

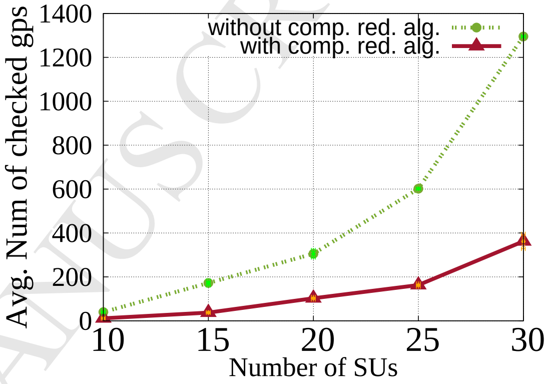


Figure 17: Average number of checked groups at each relay.

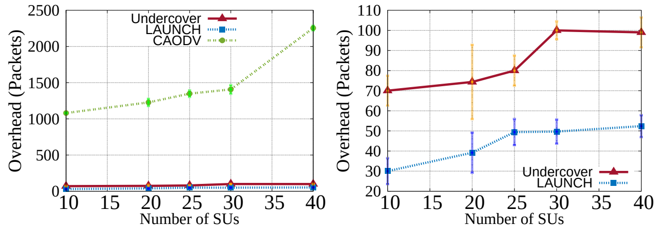
6.2.2. Group Construction

Average Goup Size. Figure 16 shows the average size of the groups constructed by *Undercover*. It can be noted that: as the number of SUs increases, the ability to construct larger groups increases. However, in almost all cases, *Undercover* prefers to construct small-size groups to decrease the interference to the least possible effect especially when the number of PUs in the region of node's transmission is low.

Average Number of checked groups. Figure 17 shows the average number of checked groups at each relay that applies the RREP algorithm. This figure shows the advantage of the proposed search window control heuristic that was mentioned in Section 5. We can see that without applying this heuristic, the number of checked groups per relay increases exponentially with the number of SUs in the topology. However, applying the mentioned techniques drastically reduce the number of groups being considered by the algorithm, thus reducing the time taken to check these groups.

6.2.3. Routing Overhead

Figure 18 shows the effect of increasing the number of SUs on the routing overhead. From Figure 18a, we can see that CAODV always has a higher overhead compared to both LAUNCH and *Undercover*. This is due to the



(a) Comparison between the three routing protocols

(b) Zoomed figure on the part of low overhead protocols

Figure 18: Effect of changing number of SUs on the routing overhead.

global routing approach used by CAODV which leads to having always a higher overhead compared to the local routing technique used by the other two protocols. Figure 18b compares between both LAUNCH and *Undercover* in terms of their added overhead packets. We can see that there is a nearly constant number of overhead packets added by *Undercover* to help it in the groups' construction process. Although the group construction leads to a higher total end-to-end delay (which is considered as a part of the protocol overhead), we note that, as in Figure 10, this added delay is experienced by only a small portion of packets.

7. Conclusion

In this paper, we proposed *Undercover*, a cross layering routing protocol that integrates physical layer techniques in the routing layer (layer 3). *Undercover* utilizes cooperative groups and uses beamforming to send data normally even if primary users exist and are active. This property leads to a better packet delivery ratio for *Undercover* than other protocols. Thus, the ability to send data simultaneously with the primary users opens a new degree of freedom that was not available before. Also, the collaboration between nodes is used to send signals in adhoc networks (the case when no primary users exist) with better qualities. Thus, although our protocol is designed mainly for Cognitive Radio Networks, it proves to be useful also in adhoc networks. *Undercover* is also designed to be an interference-aware protocol as it takes into consideration the interference that the constructed cooperative groups can inflict on other routes and vice versa. Moreover, search window control heuristic is proposed which aims at shrinking the search space for the potential group members. Evaluating *Undercover* is done using NS2 where the achieved goodput and the average end-to-end delay are observed. *Undercover* is compared against CAODV, which is a representative for the geographical protocols, and LAUNCH as an example from the location-aided routing protocols. *Undercover* achieves a goodput gain that reaches up to 250% compared to other protocols. Also, it shows to have a low overhead and a reasonable end-to-end delay. In addition, the search window control heuristic evaluation shows that it successfully reduces the time of searching for the best group so that the algorithm may be used practically.

Future directions include finding a mathematical model for values in table 2 and a way to improve the group construction time. In this work, we assume that PUs are stationary. One way to extend this work is to assume mobile PUs. One way to accomodate this change is to remember top k groups and choose one of them based on the PUs locations. Moreover, we assume some model of detecting PUs and sensing their activities. Exploring other models of doing this would be a good future direction too.

Acknowledgement

This work is supported by a grant from the Egyptian National Telecommunication Regulatory Authority (NTRA).

References

[1] F. S. P. T. Force, Report of the spectrum efficiency working group (2002).

[2] Facts and forecasts: Billions of things, trillions of dollars (2014 (accessed October, 2016)).

[3] P. Rysavy, Spectrum crisis?, Information Week Magazine (2009) 23–30.

[4] M. Matinmikko, M. Mustonen, D. Roberson, J. Paavola, M. Höyhtyä, S. Yrjölä, J. Röning, Overview and comparison of recent spectrum sharing approaches in regulation and research: From opportunistic unlicensed access towards licensed shared access, in: Dynamic Spectrum Access Networks (DYSPAN), 2014 IEEE International Symposium on, 2014, pp. 92–102.

[5] N. Rupasinghe, İ. Güvenç, Licensed-assisted access for wifi-lte coexistence in the unlicensed spectrum, in: Globecom Workshops (GC Wkshps), 2014, IEEE, 2014, pp. 894–899.

[6] S. Haykin, Cognitive radio: brain-empowered wireless communications, IEEE Journal on Selected Areas in Communications 23 (2) (2005) 201–220. doi:10.1109/JSAC.2004.839380.

[7] S. K. Sharma, T. E. Bogale, S. Chatzinotas, B. Ottersten, L. B. Le, X. Wang, Cognitive radio techniques under practical imperfections: A survey, IEEE communications surveys and tutorials.

[8] S. M. Dudley, W. C. Headley, M. Lichtman, E. Y. Imana, X. Ma, M. Abdelbar, A. Padaki, A. Ullah, M. M. Sohul, T. Yang, et al., Practical issues for spectrum management with cognitive radios, Proceedings of the IEEE 102 (3) (2014) 242–264.

[9] M. Youssef, M. Ibrahim, M. Abdelatif, L. Chen, A. V. Vasilakos, Routing metrics of cognitive radio networks: A survey, Communications Surveys & Tutorials, IEEE 16 (1) (2014) 92–109.

[10] S. Abdelaziz, M. ElNainay, Metric-based taxonomy of routing protocols for cognitive radio ad hoc networks, Journal of Network and Computer Applications 40 (2014) 151–163.

[11] I. Pefkianakis, S. H. Wong, S. Lu, Samer: spectrum aware mesh routing in cognitive radio networks, in: New Frontiers in Dynamic Spectrum Access Networks. DySPAN 2008. 3rd Symposium on, IEEE, pp. 1–5.

[12] K. R. Chowdhury, M. D. Felice, Search: A routing protocol for mobile cognitive radio ad-hoc networks, Computer Communications 32 (18) (2009) 1983–1997.

[13] K. Habak, M. Abdelatif, H. Hagrass, K. Rizc, M. Youssef, A location-aided routing protocol for cognitive radio networks, in: Computing, Networking and Communications (ICNC), International Conference on, IEEE, 2013, pp. 729–733.

[14] X. Zhou, L. Lin, J. Wang, X. Zhang, Cross-layer routing design in cognitive radio networks by colored multigraph model, Wireless Personal Communications 49 (1) (2009) 123–131.

[15] L. Ding, T. Melodia, S. N. Batalama, J. D. Matyjas, M. J. Medley, Cross-layer routing and dynamic spectrum allocation

in cognitive radio ad hoc networks, IEEE Transactions on Vehicular Technology 59 (4) (2010) 1969–1979.

[16] S. Shakkottai, T. S. Rappaport, P. C. Karlsson, Cross-layer design for wireless networks, IEEE Communications magazine 41 (10) (2003) 74–80.

[17] L. H. Grokop, Interference management in wireless networks: Physical layer communication strategies, MAC layer interactions, and high layer messaging structures, ProQuest, 2008.

[18] A. Goldsmith, S. A. Jafar, I. Maric, S. Srinivasa, Breaking spectrum gridlock with cognitive radios: An information theoretic perspective, Proceedings of the IEEE 97 (5) (2009) 894–914.

[19] I. R. Regulations, Resolutions 5.444 b, 5.444 c (2012).

[20] A. Goldsmith, Wireless communications, Cambridge university press, 2005.

[21] D. Tse, P. Viswanath, Fundamentals of wireless communication, Cambridge university press, 2005.

[22] B. D. Van Veen, K. M. Buckley, Beamforming: A versatile approach to spatial filtering, IEEE assp magazine 5 (2) (1988) 4–24.

[23] X. Chen, H.-H. Chen, W. Meng, Cooperative communications for cognitive radio networks—from theory to applications, Communications Surveys & Tutorials, IEEE 16 (3) (2014) 1180–1192.

[24] X. Tao, X. Xu, Q. Cui, An overview of cooperative communications, IEEE Communications Magazine 50 (6) (2012) 65–71.

[25] M. Karmoose, A. Sultan, M. Youssef, Stability analysis in a cognitive radio system with cooperative beamforming, in: Wireless Communications and Networking Conference (WCNC), IEEE, 2013, pp. 637–642.

[26] T. Yi, L. Guo, K. Niu, H. Cai, J. Lin, W. Ai, Cooperative beamforming in cognitive radio network with hybrid relay, in: 19th International Conference on Telecommunications (ICT), IEEE, 2012, pp. 1–5.

[27] B. Wang, K. R. Liu, Advances in cognitive radio networks: A survey, IEEE Journal of selected topics in signal processing 5 (1) (2011) 5–23.

[28] M. Karmoose, K. Habak, M. ElNainay, M. Youssef, Dead zone penetration protocol for cognitive radio networks, in: Wireless and Mobile Computing, Networking and Communications (WiMob), 9th International Conference on, IEEE, 2013, pp. 529–536.

[29] DARPA, Spectrum Collaboration Challenge, https://spectrumcollaborationchallenge.com/, [Online; accessed 7-June-2017] (2016).

[30] S. F. S. McCanne, NS network simulator, http://www.w3schools.com/browsers/browsers_os.asp.

[31] S. Lakshmanan, R. Sivakumar, Diversity routing for multi-hop wireless networks with cooperative transmissions, in: IEEE SECON, 2009, pp. 1–9.

[32] C. J. Collier, R. J. Murray, Transmission system for sending two signals simultaneously on the same communications channel, uS Patent 5,073,899 (Dec. 17 1991).

[33] A. E. Khandani, E. Modiano, J. Abounadi, L. Zheng, Cooperative routing in wireless networks, in: Advances in Pervasive Computing and Networking, Springer, 2005, pp. 97–117.

[34] A. E. Khandani, J. Abounadi, E. Modiano, L. Zheng, Cooperative routing in static wireless networks, IEEE Transactions on Communications 55 (11) (2007) 2185–2192.

[35] F. Li, K. Wu, A. Lippman, Energy-efficient cooperative routing in multi-hop wireless ad hoc networks, in: 25th IEEE International Performance, Computing, and Communications Conference, IPCCC 2006., pp. 8–pp.

[36] A. Ibrahim, Z. Han, K. R. Liu, Distributed energy-efficient cooperative routing in wireless networks, IEEE Transactions on Wireless Communications 7 (10) (2008) 3930–3941.

[37] G. Jakllari, S. V. Krishnamurthy, M. Faloutsos, P. V. Krishnamurthy, O. Ercetin, A cross-layer framework for exploiting virtual miso links in mobile ad hoc networks, IEEE Transactions on Mobile Computing 6 (6) (2007) 579–594.

[38] S. Lakshmanan, R. Sivakumar, Proteus: Multiflow diversity routing for wireless networks with cooperative transmissions, Mobile Computing, IEEE Transactions on 12 (6) (2013) 1146–1159.

[39] S. Sharma, Y. Shi, Y. T. Hou, H. D. Sherali, S. Kompella, Cooperative communications in multi-hop wireless networks: Joint flow routing and relay node assignment, in: INFOCOM, Proceedings, IEEE, 2010, pp. 1–9.

[40] Q. Zhang, J. Jia, J. Zhang, Cooperative relay to improve diversity in cognitive radio networks, IEEE Communications Magazine 47 (2) (2009) 111–117.

[41] L. Ding, T. Melodia, S. N. Batalama, J. D. Matyjas, Distributed routing, relay selection, and spectrum allocation in cognitive and cooperative ad hoc networks, in: 2010 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON), pp. 1–9.

[42] J.-P. Sheu, I.-L. Lao, Cooperative routing protocol in cognitive radio ad-hoc networks, in: IEEE Wireless Communications and Networking Conference (WCNC), 2012, pp. 2916–2921.

[43] I. F. Akyildiz, B. F. Lo, R. Balakrishnan, Cooperative spectrum sensing in cognitive radio networks: A survey, Physical communication 4 (1) (2011) 40–62.

[44] J. Liu, W. Chen, Z. Cao, Y. J. A. Zhang, Cooperative beamforming for cognitive radio networks: a cross-layer design, IEEE Transactions on Communications 60 (5) (2012) 1420–1431.

[45] A. M. Akhtar, L. De Nardis, M. R. Nakhai, O. Holland, M. G. Di Benedetto, A. H. Aghvami, Multi-hop cognitive radio networking through beamformed underlay secondary access, in: Communications (ICC), 2013 IEEE International Conference on, IEEE, 2013, pp. 2863–2868.

[46] FCC, White Spaces.

[47] P. Enge, P. Misra, Special issue on GPS: The Global Positioning System, Proceedings of the IEEE.

[48] M. Ibrahim, M. Youssef, CellSense: An Accurate Energy-Efficient GSM Positioning System, Vehicular Technology, IEEE Transactions on 61 (1) (2012) 286–296.

[49] C.-I. Bădoi, V. Croitoru, R. Prasad, Ipsag: an ip spectrum aware geographic routing algorithm proposal for multi-hop cognitive radio networks, in: Communications (COMM), 2010 8th International Conference on, IEEE, 2010, pp. 491–496.

[50] A. Guirguis, R. Guirguis, M. Youssef, Primary user-aware network coding for multi-hop cognitive radio networks, in: Global Telecommunications Conference (GLOBECOM), IEEE, 2014.

[51] B. Wild, K. Ramchandran, Detecting primary receivers for cognitive radio applications, in: First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005., IEEE, 2005, pp. 124–130.

[52] G. Hattab, M. Ibnkahla, Enhanced pilot-based spectrum sensing algorithm, arXiv preprint arXiv:1409.6392.

[53] R. Tandra, A. Sahai, Snr walls for signal detection, IEEE Journal of selected topics in Signal Processing 2 (1) (2008) 4–17.

[54] A. Guirguis, M. Ibrahim, K. Seddik, K. Harras, F. Digham, M. Youssef, Primary user aware k-hop routing for cognitive radio networks, in: Global Telecommunications Conference (GLOBECOM), IEEE, 2015.

[55] Cognitive radio extension., [Online]. Available: http://stuweb.ee.mtu.edu/ljialian/.

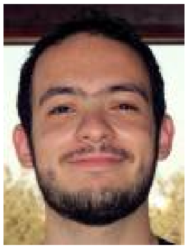
[56] K. Gomadam, V. R. Cadambe, S. Jafar, et al., Approaching the capacity of wireless networks through distributed interference alignment, in: Global Telecommunications Conference., IEEE, 2008, pp. 1–6.

[57] K. Yu, B. Ottersten, Models for mimo propagation channels: a review, Wireless Communications and Mobile Computing 2 (7) (2002) 653–666.

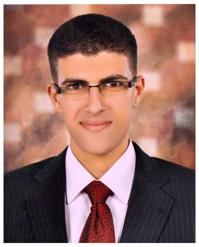
[58] A. S. Cacciapuoti, C. Calcagno, M. Caleffi, L. Paura, Caodv: Routing in mobile ad-hoc cognitive radio networks, in: Wireless Days (WD), IFIP, IEEE, 2010, pp. 1–5.

[59] C. Perkins, E. Belding-Royer, S. Das, Ad hoc on-demand distance vector (aodv) routing, Tech. rep. (2003).

Arsany Guirguis is a Research Assistant at Computer and Systems Department, Faculty of Engineering, Alexandria University, Egypt. He received his B.Sc. and M.Sc. in Computer and Systems Engineering from Alexandria University, Egypt in 2014 and 2017 respectively. His research interests include cognitive radio networks, routing protocols and cooperative routing. Arsany is a recipient of the Certificate of Honor for being ranked second in College of Engineering and first on Computer and Systems Department, Alexandria University in 2014. He has also received the Best Graduation Project Award from the same department in 2014.



Mohammed Karmoose received the BS and MS degrees in electrical engineering from the Faculty of Engineering in Alexandria University in Egypt in 2009 and 2013 respectively. He is currently pursuing his Ph.D degree in electrical engineering at UCLA in California, USA. He was a part of the CRN research group in E-JUST in Egypt as a Graduate Research Assistant from 2011 to 2014. He received the Annual Tribute Ceremony award for top-ranked students in Alexandria University in the years 2005 to 2009. He received the Electrical Engineering Department Fellowship from UCLA for his first year of Ph.D in 2014/2015. His research interests are detection and estimation, cooperative caching and wireless communications.



Karim Habak is a Research Assistant and Ph.D Student at the School of Computer Science, Georgia Institute of Technology. He received his M.Sc. degree in Computer Science and Engineering from Egypt-Japan University of Science and Technology (E-JUST), Egypt in 2012 and a .B.Sc. in Computer and Systems Engineering from Alexandria University, Egypt in 2010. His research interests include bandwidth aggregation, cognitive radio networks, sensor networks, and pattern recognition. Karim is the recipient of the Best Computing and Information Technology Research Program of the Year award of the 2013 Third Qatar Foundation Annual Research Conference.

Dr. ElNainay is an Associate Professor of the Computer and Systems Engineering department at Alexandria University, Egypt. He is also the associate director of the Virginia Tech-Middle East and North Africa (VT-MENA) program for administration and research and adjunct faculty at Virginia Tech. He received his B.Sc. and M.Sc. in Computer Science from Alexandria University in 2001 and 2005 respectively and his Ph.D. in Computer Engineering from Virginia Tech in 2009. His research interests include wireless and mobile networks, cognitive radio and cognitive networks, and software testing automation and optimization. He is the receipt of ICDT best paper award. He is a Senior IEEE member and served as reviewer and TPC member for various international conferences and journals.

Moustafa Youssef is a Professor at Egypt-Japan University of Science and Technology (E-JUST) and Founder & Director of the Wireless Research Center of Excellence, Egypt. His research interests include mobile wireless networks, mobile computing, location determination technologies, pervasive computing, and network security. He is an associate editor for the ACM TSAS, a previous area editor of the ACM MC2R and served on the organizing and technical committees of numerous prestigious conferences. Prof. Youssef is the recipient of the 2003 University of Maryland Invention of the Year award, the 2010 TWAS-AAS-Microsoft Award for Young Scientists, the 2012 Egyptian State Award, the 2013 and 2014 COMESA Innovation Awards, the 2013 ACM SIGSpatial GIS Conference Best Paper Award, among many others. He is also an ACM Distinguished Scientist and an ACM Distinguished Speaker.