Cooperative Delay-Constrained Cognitive Radio Networks: Throughput Maximization with Full-Duplex Capability Impact

Ali Gaber[1(✉)], El-Sayed Youssef[1], Mohamed R. M. Rizk[1], Mohamed Salman[2], and Karim G. Seddik[3]

[1] Department of Electrical Engineering, Alexandria University, Alexandria, Egypt
aligaber@alexu.edu.eg, dr.e.a.youssef@gmail.com, mrmrizk@ieee.org
[2] ECEE Department, University of Colorado, Boulder, CO, USA
mohamed.salman@colorado.edu
[3] ECNG Department, American University in Cairo, Cairo, Egypt
kseddik@aucegypt.edu

Abstract. In this paper, we study the problem of maximizing the secondary user (SU) throughput under a quality of service (QoS) delay requirement of the primary user (PU). In addition, we investigate the impact of having a full-duplex capability at the SU on the network performance, compared to the case of a half-duplex SU. We consider a cooperative cognitive radio (CR) network with multipacket reception (MPR) capabilities at the receiving nodes. In our proposed system, the SU not only exploits the idle time slots (i.e. when PU is not transmitting) but also chooses between cooperating or sharing the channel with the PU probabilistically. We formulate our optimization problem maximizing the SU throughput under a PU delay constraint; we optimize over the SU transmission modes' selection probabilities. The resultant optimization problem turns out to be a non-convex quadratic constrained quadratic programming (QCQP) optimization problem, which is, in general, an NP-hard problem. An efficient approach is devised to solve it and characterize the stability region of the network under a delay constraint on the PU. Numerical results, surprisingly, reveal that the network performance with a full-duplex SU is not always better than that of a half-duplex SU. In fact, we show that a full-duplex capability at the SU can adversely affect the stability performance of the network especially if the channel condition between the SU and the destinations is weaker than that between the PU and the destinations.

1 Introduction

Cooperative communication techniques have gained a lot of interest over the years due to the important key role they play in wireless communications [1]. In cooperative communications, intermediate nodes capture the source transmitted packets and contribute via cooperatively relaying them to the destination. In

A. Kliks et al. (Eds.): CrownCom 2019, LNICST 291, pp. 180–194, 2019.
https://doi.org/10.1007/978-3-030-25748-4_14

[2], the authors showed that a reasonable enhancement in the stable throughput region of the network can be achieved as a result of the existence of a cooperative relay node. On the other hand, cognitive radio (CR) has emerged as a powerful leading technology in alleviating the scarcity of available spectrum resources, which are relatively under-utilized. CR achieves efficient utilization of the spectrum while maintaining some primary users (PUs) QoS [3]. Recently, researchers started to integrate cooperative communications into CR networks, e.g., [4], by allowing the secondary users (SUs) to serve as relays for the PUs. As a result, the SUs use their available resources to transmit their own data as well as PUs' packets. It was shown that cooperation could, in general, enhance both the SU and the PU throughputs.

The authors in [5] and [6] have derived the stability region for a cooperative CR network consisting of one PU, one SU (with two queues, one for its own data packets and the other one to relay the PU packets), and a common destination. The authors in [5] considered a model in which the SU transmits only when the PU is inactive with firm priority in favor of the PU relaying queue. Assigning firm priority to the relaying queue can degrade the SU throughput, especially when the PU average packet delay becomes much smaller than the target delay constraint. To overcome this problem, the authors in [6] presented a model in which the SU can serve either its own data queue or the relaying queue according to some probability assigned to each queue (i.e., the SU randomizes the service between its own data queue and the relaying queue whenever it has access to the channel). In [5] and [6], optimizing the SU throughput and the average packet delay experienced by the PU and SU were studied, respectively, subject to network stability constraints. It was shown that cooperation is beneficial only when the channel between the SU and the common destination is better than that between the PU and the destination (same restriction as in [2]). Recently, the authors in [7] modified the scheme presented in [6] by allowing the SU to have an extra queue (battery queue). The authors then derived the stability region of the network subject to some energy harvesting constraints on the SU.

Another important aspect of our proposed framework is to consider the case when the SU has a full-duplex capability and characterize its effect on the system stability region. Most of the previous works on cooperative CR networks have assumed that the users are all half-duplex; this is because full-duplex communications were considered infeasible in the past due to the effect of self-interference. Recently, the authors in [8] presented feasible approaches that can achieve a drastic self-interference suppression. Using these approaches, the authors in [9] showed that the existence of a full-duplex relay can enhance the users' throughput. In [10], the model presented in [9] was extended assuming a full-duplex SU that has its own data queue in addition to the relaying queue, unlike [9] where the relay node is assumed not to have data of its own; the SU throughput was maximized subject to some stability constraints of the network. Note that the recent works presented in the context of cooperative CR, e.g., [6,7] and [11] consider only a half-duplex SU to simplify the analysis for their models.

Our main contributions in this paper are as follows.

- We propose a delay-aware scheme that enables us to maximize the SU throughput subject to a QoS delay requirement of the PU. Our optimization problem turns out to be a non-convex quadratic constrained quadratic programming (QCQP) optimization problem which is NP-hard in general. An efficient approach is proposed to solve our optimization problem to characterize the stability region. The importance of our proposed framework arises from the emergence of numerous real-time applications that should be supported by cooperative CR networks e.g., video streaming, gaming, and other multimedia applications; these applications demand high throughput with strict delay requirements. This aspect is mostly neglected in all of the above-cited cooperative CR works [5–7,9,10], which mainly focused on enhancing certain performance subject to some network stability and/or energy harvesting constraints with no delay provisioning.
- We study the impact of having the full-duplex capability at the SU on the network performance from a queuing-theoretic perspective. The full-duplex capability enables the SU to decode the PU transmission while simultaneously transmitting over the channel. We, unexpectedly, show that having a full-duplex capability at the SU is not always beneficial. It can adversely affect the stability performance of the network and, in some cases, provide strictly inferior performance compared to the system with a half-duplex SU.

It is worth mentioning that optimizing network performance under PU delay constraints have been considered before in [11] for a different cooperation policy and for a simpler network configuration. More precisely, [11] considered only a half-duplex SU which simplifies the stability and delay analysis for their network. Moreover, [11] considered only the case in which the SU accesses the channel when the PU is sensed to be inactive (similar to [5,6]). This, in effect, reduces the number of optimization parameters, and hence, simplifies the optimization problem.

The rest of the paper is organized as follows. The system model is described in Sect. 2. Section 3 describes our cooperation policy and presents the delay and stability analysis. The problem formulation and the solution approach are presented in Sect. 4. The numerical results are presented in Sect. 5. Finally, conclusions are drawn in Sect. 6.

2 System Model

In this paper, we consider a full-duplex cooperative CR network, shown in Fig. 1, consisting of one PU (p), one SU (s) and two different destinations (d and $d^{'}$). The SU is a full-duplex node that can simultaneously receive a packet from the PU and transmit a packet to d or $d^{'}$. We also assume that the primary destination (d) has MPR capability, i.e., it can simultaneously decode multiple packets received from the PU and the SU relaying queue.

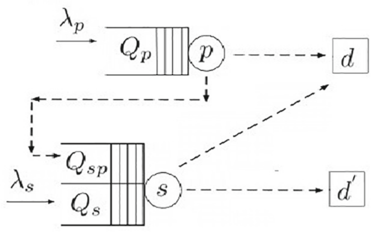

Fig. 1. The system model

Time is divided into slots; each slot has a fixed time duration, and for simplicity of presentation, we assume that every packet transmission takes only one slot. The SU is assumed to have two infinite queues Q_s and Q_{sp}. The queue Q_s is used to store the SU arriving packets, whereas the queue Q_{sp} is used to store the relayed packets received from the PU. The PU is assumed to have only one infinite queue Q_p used to store its arriving packets. The arrivals at the PU and the SU are considered to be Bernoulli processes with rates λ_p and λ_s, respectively. The arrival processes at both users are assumed to be independent.

The wireless channel between any two nodes (m, n), where $m \in \{p, s\}$ is the transmitting node and $n \in \{s, d, d^{'}\}$ is the receiving node, where $m \neq n$, is modeled as a stationary Rayleigh flat-fading channel. The channel gain is denoted by h_{mn}, where $\mathbb{E}(|h_{mn}|^2) = \rho_{mn}^2$. The channel gains, h_{mn}'s, are assumed to be constant within any given time slot but vary independently from one time slot to another. All users (nodes) are exposed to independent complex additive white Gaussian noise with unit variance and zero mean. The transmitted power from the PU or the SU is fixed and is denoted by P.

In our proposed scheme, we will have four different cases (modes of operation) for packet transmissions. The first case is when either the PU or the SU transmits alone, i.e, the non-transmitting node remains idle. In this case, the probability of successful decoding is given by:

$$f_{mn} = \mathbb{P}\{R < \tfrac{1}{2}\log_2(1 + P|h_{mn}|^2)\} = \exp\left(-\frac{2^{2R} - 1}{P\rho_{mn}^2}\right), \tag{1}$$

where $mn \in \{pd, ps, sd^{'}, sd\}$ and R is the transmission rate.

The second case is when the SU transmits a packet from Q_s simultaneously with the PU, i.e., each user is causing interference on the other user transmission. In this case, we let v_{mn}^I denote the probability that node n successfully decodes the packet transmitted from node m by considering the interference caused by node I transmission as noise, where $(mn, I) \in \{(pd, s), (sd^{'}, p)\}$.

The third case is when the SU transmits a relayed packet from Q_{sp} simultaneously with the PU, i.e., both nodes transmit primary packets to node d. In this case, the primary destination attempts decoding both transmissions by using its MPR capability (using successive interference cancellation). The probability of successful transmission, in this case, is given by

$$g_{mn}^{I} = u_{mn}^{I} + (1 - u_{mn}^{I}) v_{mn}^{I}, \tag{2}$$

where $(mn, I) \in \{(pd, s), (sd, p)\}$ and u_{mn}^{I} is the probability that node n successfully decodes both packets transmitted from nodes m and I. The derivations of u_{mn}^{I} and v_{mn}^{I} are omitted due to space limitations.

The fourth case occurs when the SU exploits its full-duplex capability, i.e., it attempts decoding a primary packet while transmitting either a secondary or primary packet. Let f_{sd}^{dup} denote the probability that s decodes a primary packet from p in the full-duplex mode, and is given by

$$f_{ps}^{dup} = \mathbb{P}\left\{ R < \frac{1}{2}\log_2\left(1 + \frac{P|h_{ps}|^2}{1+Pg}\right)\right\} = \exp\left(-\frac{(2^{2R}-1)(1+Pg)}{P\rho_{ps}^2}\right), \tag{3}$$

where the scalar gain $g \in [0, 1]$ represents the effectiveness of self-interference cancellation [9,12]. For example, if $g = 0$ then self-interference is perfectly suppressed, and if $g = 1$ then no self-interference cancellation is considered. The details of the techniques utilized for self-interference cancellation are beyond the scope of this paper (the interested reader is referred to [8] and references therein). Note that to consider the case where the SU is a half-duplex node, we set $f_{sd}^{dup} = 0$ not $g = 1$. As $g = 1$ corresponds to the case of full-duplex mode with no self-interference cancellation which is different from the half-duplex mode.

3 Cooperation Policy and System Analysis

In this section, our proposed cooperation policy is introduced. Then, we provide our proposed scheme delay and stability analysis. Note that we assume ACK/NACK packet transmission from s, $d^{'}$, and/or d at the end of each time slot. These ACK/NACK packets are assumed to be received error-free at all nodes. For simplicity of presentation, the SU is assumed to be able to perfectly sense the existence or the absence of the PU. As a result of this sensing process, the cooperation policy can be split into two cases as follows:

3.1 Q_p Has Packets (An Active Primary User; A Busy Time Slot)

In this case, the SU could select one of two access decisions as follows.

- The SU decides to access the channel, causing interference to the PU, by sending a packet from Q_s or Q_{sp} with probabilities p_s^{busy} or p_{sp}^{busy}, respectively (unlike [5,6,11], where the SU always refrains from accessing the channel upon detecting a PU transmission). This, in turn, should result in better utilization of the PU channel. In this case, the SU can still decode the PU packet if d fails to decode it using its full-duplex capability. Note that, for a half-duplex SU, the SU will not be able to help the PU in this case.

– The SU decides to refrain from sending any packets and only listen to the PU transmission to help to relay it if the primary destination (d) does not succeed in decoding it. This occurs with probability $1 - p_{sp}^{busy} - p_s^{busy}$.

It should be noted that the PU is the owner of the spectrum, therefore, it does not have to provide any provisions to the SU; the PU just sends the packet on the top of Q_p provided that Q_p is nonempty. As a result, three possible scenarios would emerge:

– If d decodes the received packet successfully, then Q_p drops the packet irrespective of the decision taken by the SU.
– If the PU packet is decoded successfully by the SU, and at the same time d could not decode it, then the packet will be dropped from Q_p and stored at Q_{sp}.
– If both s and d were not able to decode the primary packet, then the PU keeps the packet in Q_p to attempt re-sending it in the next time slot.

3.2 Q_p Is Empty (An Inactive Primary User; An Idle Time Slot)

In this case, the SU could select one of two access decisions as follows.

– The SU transmits one packet from Q_{sp} with probability p_{sp}^{idle}. This packet will be dropped from Q_{sp} if the SU receives an ACK from d.
– The SU transmits one packet from Q_s with probability $p_s^{idle} = 1 - p_{sp}^{idle}$. This packet will be dropped from Q_s if the SU receives an ACK from the secondary destination ($d^{'}$).

Our model involves interaction among different queues. The stability and delay analysis of more than two interacting queues is, in general, a complex problem [13]. As a result, we resort to the dominant system approach to decouple the interaction among the system queues. This approach was used before in different contexts, e.g., [14], to derive sufficient stability conditions of a system of interacting queues. In our dominant system, the SU is assumed to send dummy packets when it chooses to transmit a packet from an empty queue. This has the effect of decoupling the interaction between the queues by decoupling the service rate for any queue from the number of packets in other queues. It is obvious that dominant system stability implies original system stability (as transmitting dummy packets can only degrade the system performance). Also, the average delay experienced by the packets in the dominant system is an upper bound on that of the original system.

Next, we present the details of the stability analysis of our system followed by the delay analysis. Loynes' theorem [15] is applied to examine the stability of each queue in the system. Loynes' theorem states that a queue is stable as long as its average arrival rate is strictly less than its average service rate if both the arrival and the service processes are stationary. It should be noted that the PU packets can be served by two queues, Q_p and Q_{sp}, and this should be taken into consideration while calculating the PU delay.

According to the scenarios illustrated above in Sects. 3.1 and 3.2, the PU packets' departure rate from Q_p can be expressed as [1]

$$\mu_p = p_s^{busy}(v_{pd}^s + (1 - v_{pd}^s) f_{ps}^{dup}) + p_{sp}^{busy}(g_{pd}^s + (1 - g_{pd}^s) f_{ps}^{dup}) \\ +(1 - p_s^{busy} - p_{sp}^{busy})(f_{pd} + (1 - f_{pd}) f_{ps}). \tag{4}$$

For Q_p to be stable, its arrival rate λ_p should be less than μ_p, i.e.,

$$\lambda_p < p_s^{busy}(v_{pd}^s + (1 - v_{pd}^s) f_{ps}^{dup}) + p_{sp}^{busy}(g_{pd}^s + (1 - g_{pd}^s) f_{ps}^{dup}) \\ +(1 - p_s^{busy} - p_{sp}^{busy})(f_{pd} + (1 - f_{pd}) f_{ps}). \tag{5}$$

For Q_{sp}, the service rate, μ_{sp}, can be given as

$$\mu_{sp} = \frac{\lambda_p}{\mu_p} p_{sp}^{busy} g_{sd}^p + (1 - \frac{\lambda_p}{\mu_p}) p_{sp}^{idle} f_{sd}, \tag{6}$$

where $\frac{\lambda_p}{\mu_p}$ is the probability that Q_p is nonempty. Moreover, the arrival rate at Q_{sp}, λ_{sp}, is given by

$$\lambda_{sp} = \frac{\lambda_p}{\mu_p} \{ p_s^{busy} f_{ps}^{dup}(1 - v_{pd}^s) + p_{sp}^{busy} f_{ps}^{dup}(1 - g_{pd}^s) \\ +(1 - p_s^{busy} - p_{sp}^{busy}) f_{ps}(1 - f_{pd}) \}. \tag{7}$$

To satisfy the Q_{sp} stability requirement, λ_{sp} should be less than μ_{sp}, i.e.,

$$\lambda_p < \frac{C}{A - B + C} \mu_p, \tag{8}$$

where A, B, and C are given by

$$A = \{ p_s^{busy} f_{ps}^{dup}(1 - v_{pd}^s) + p_{sp}^{busy} f_{ps}^{dup}(1 - g_{pd}^s) \\ + (1 - p_s^{busy} - p_{sp}^{busy}) f_{ps}(1 - f_{pd}) \}, \\ B = p_{sp}^{busy} g_{sd}^p, C = p_{sp}^{idle} f_{sd}.$$

From (5) and (8), it can be easily shown that the PU arrival rate should satisfy the following condition for the system to be stable.

$$\lambda_p < \min\{\mu_p, \mu_r\}, \tag{9}$$

where μ_p is as in (4) and μ_r is given by

$$\mu_r = \frac{C}{A - B + C} \mu_p. \tag{10}$$

[1] Note that throughout the analysis presented in this paper, we consider a dominant system in which the SU transmits dummy packets if it selects to transmit from an empty queue. This has the effect of decoupling the service rates of each queue from the state of other queues.

Finally, for Q_s, the service rate μ_s can be shown to be given by

$$\mu_s = \frac{\lambda_p}{\mu_p} p_s^{busy} v_{sd'}^p + \left(1 - \frac{\lambda_p}{\mu_p}\right)(1 - p_{sp}^{idle}) f_{sd'}. \tag{11}$$

To satisfy the Q_s stability requirement, the arrival rate λ_s should be less than μ_s, i.e.,

$$\lambda_s < \frac{\lambda_p}{\mu_p} p_s^{busy} v_{sd'}^p + \left(1 - \frac{\lambda_p}{\mu_p}\right)(1 - p_{sp}^{idle}) f_{sd'}. \tag{12}$$

This completes our stability region characterization.

If a PU packet is delivered to its destination d via the SU, the packet will experience two queuing delays; the delay in Q_p and the delay in Q_{sp}. As a result, the average delay experienced by the PU packet can be expressed as follows:

$$D_p = \tau T_p + (1-\tau)(T_p + T_{sp}) = T_p + (1-\tau) T_{sp}, \tag{13}$$

where the average delays at Q_{sp} and Q_s are represented by T_{sp} and T_s, respectively. And τ is given by

$$\tau = \frac{(1 - p_s^{busy} - p_{sp}^{busy}) f_{pd} + p_s^{busy} v_{pd}^s + p_{sp}^{busy} g_{pd}^s}{\mu_p}, \tag{14}$$

and it represents the possibility that the PU packet is decoded successfully by d conditioned on that it was dropped from Q_p. The queues Q_{sp} and Q_s are discrete time M/M/1 queues with Bernoulli arrival processes and Geometric service rates. Therefore, by applying Pollaczek-Khinchine [16] and Little's law, T_p and T_{sp} can be expressed as

$$T_p = \frac{1 - \lambda_p}{\mu_p - \lambda_p}, \quad T_{sp} = \frac{1 - \lambda_{sp}}{\mu_{sp} - \lambda_{sp}}. \tag{15}$$

4 Problem Formulation and Solution Approach

In this section, we introduce our optimization problem. Our objective is to maximize the SU throughput λ_s subject to a PU QoS delay requirement. As a result, our optimization problem can be written, in an epigraph form [17, Chapter 4], as follows:

$$\begin{aligned} \max_{p_s^{busy}, p_{sp}^{busy}, p_{sp}^{idle}, \lambda_s} \quad & \lambda_s \\ \text{subject to} \quad & \lambda_s \le \mu_s, \\ & D_p \le \phi, \\ & 0 \le p_s^{busy} + p_{sp}^{busy} \le 1, \\ & 0 \le p_{sp}^{idle} \le 1, \\ & p_i^{busy} \ge 0 \quad i \in \{s, sp\}, \end{aligned} \tag{16}$$

where the stability of Q_s is guaranteed by the first constraint, while the PU QoS delay requirement is guaranteed by the second constraint. Introducing a PU delay constraint is stricter than a PU queue stability constraint and, hence, implies the stability of the primary queue, i.e., Q_p length is guaranteed not to grow to infinity. Consequently, there is no need for having an extra stability constraint of Q_p.

It should be noted that D_p and μ_s, given in (13) and (11), are non-convex functions in the optimization parameters, which renders the overall optimization problems to be non-convex. Next, we go through a number of steps to solve this non-convex optimization problem. Note that, due to space limitations, we just provide a concise description of our proposed approach to solve the above non-convex optimization problem.

First, it should be noted that if we fix μ_p (make it constant) then μ_s from (11) becomes a linear function of the optimization parameters. Hence, the first constraint in the optimization problem in (16) becomes convex. Moreover, if we fix μ_p then the delay constraint, which is the second constraint in our optimization problem, becomes a quadratic function of the optimization parameters in the form of $\boldsymbol{p}^T\boldsymbol{A}\boldsymbol{p} + \boldsymbol{c}^T\boldsymbol{p} + d \leq 0$ where $\boldsymbol{p}$ is a vector that contains all the optimization parameters. Unlike the first constraint, the second one is, unfortunately, non-convex because the matrix $\boldsymbol{A}$ is not a positive semi-definite matrix as will be explained later. Hence, by fixing μ_p, we convert the first constraint to be linear and the second to be non-convex quadratic, while the rest of the constraints are already linear constraints. This form of optimization problems is called non-convex QCQP optimization problems. To solve this problem, we use the feasible point pursuit successive convex approximation (FPP-SCA) algorithm presented in [18], which solves the problem by linearizing the non-convex parts of the delay constraint. We use it, due to its advantages, illustrated in [18], over the other methods that can be used to solve QCQP optimization problems.

To find the optimum μ_p, we iterate over all possible values of μ_p and for each value we solve a non-convex QCQP optimization problem. For each value of μ_p, we solve for the maximum stable λ_s given μ_p. Finally, we find the maximum value of λ_s over all feasible μ_p's to be our solution. The values of feasible μ_p's range from λ_p to λ_p^m, which is the maximum feasible value of λ_p. The minimum value of μ_p is λ_p to guarantee the stability of the PU queue. The value of λ_p^m can be easily calculated by setting $p_{sp}^{idle} = 1$ and $p_s^{busy} = 0$ in (9) and optimizing only over p_{sp}^{busy} (which can be done easily via one-dimensional numerical search).

Based on the above, and for a given μ_p, we can easily see from (11) that the first constraint becomes a linear function in the optimization parameters. Define a 4-D vector $\boldsymbol{p} = [p_s^{busy}, p_{sp}^{busy}, p_{sp}^{idle}, \lambda_s]^T$; the second constraint can now be rewritten as $\boldsymbol{p}^T\boldsymbol{A}\boldsymbol{p} + \boldsymbol{c}^T\boldsymbol{p} + d \leq 0$ where $\boldsymbol{A}$, $\boldsymbol{c}$, and d can be directly obtained from (6), (7), and (13). It can be readily seen that this constraint is a non-convex quadratic constraint since $\boldsymbol{A}$ is an indefinite matrix. As mentioned above, we use FPP-SCA algorithm, presented in [18] which linearizes the non-convex parts of the constraints as shown next.

Using eigenvalue decomposition, we can express the matrix $\boldsymbol{A}$, which is an indefinite matrix, as $\boldsymbol{A}$=$\boldsymbol{A}^{+}$+$\boldsymbol{A}^{-}$, where $\boldsymbol{A}^{+} \succeq 0$ and $\boldsymbol{A}^{-} \preceq 0$. For any $\boldsymbol{z} \in R^{4\text{x}1}$, we have

$$(\boldsymbol{p}-\boldsymbol{z})^T\boldsymbol{A}^{-}(\boldsymbol{p}-\boldsymbol{z}) \leq 0, \tag{17}$$

$$\boldsymbol{p}^T\boldsymbol{A}^{-}\boldsymbol{p} \leq 2\boldsymbol{z}^T\boldsymbol{A}^{-}\boldsymbol{p}-\boldsymbol{z}^T\boldsymbol{A}^{-}\boldsymbol{z}. \tag{18}$$

With the aid of the above inequalities, the quadratic non-convex constraint ($\boldsymbol{p}^T\boldsymbol{A}\boldsymbol{p}+\boldsymbol{c}^T\boldsymbol{p}+d \leq 0$) can be replaced by the following convex constraint:

$$\boldsymbol{p}^T\boldsymbol{A}^{+}\boldsymbol{p}+2\boldsymbol{z}^T\boldsymbol{A}^{-}\boldsymbol{p}+\boldsymbol{c}^T\boldsymbol{p}+d \leq \boldsymbol{z}^T\boldsymbol{A}^{-}\boldsymbol{z}, \tag{19}$$

which relaxed the non-convex part of the constraint to a linear one. Now, our non-convex problem is converted into a convex one. Finally, we rewrite the optimization problem in (16) to that given in Algorithm 1, where $\boldsymbol{x}=[0,0,0,-1]^T$, $\boldsymbol{b}=[1,1,0,0]^T$ and $\boldsymbol{v},u,\boldsymbol{m},n$ can be obtained from (4).

Algorithm 1 finds the feasible solution that maximizes the throughput of the SU under a QoS delay constraint on the PU. Note that a slack variable (s) is used to ensure the feasibility of the approximated problem, and a penalty (Λ) is used to guarantee that the slack is mildly used.

Algorithm 1

For $\mu_p=\lambda_p:\delta:\lambda_p^m$
Initialization: set $i=0$ and $\boldsymbol{z}_0=\boldsymbol{0}$.
Repeat
1. solve

$$\begin{aligned} \lambda_s(\mu_p) \quad &= \min_{\boldsymbol{p}} \ \boldsymbol{x}^T\boldsymbol{p}+\Lambda s \\ \text{s. t.} \quad & \boldsymbol{v}^T\boldsymbol{\alpha}+u=\mu_p, \\ & \boldsymbol{m}^T\boldsymbol{p}+n \leq 0, \\ & \boldsymbol{p}^T\boldsymbol{A}^{+}\boldsymbol{p}+2\boldsymbol{z}_i^T\boldsymbol{A}^{-}\boldsymbol{p}+\boldsymbol{c}^T\boldsymbol{p}+d \leq \boldsymbol{z}_i^T\boldsymbol{A}^{-}\boldsymbol{z}_i+s, \\ & 0 \leq \boldsymbol{b}^T\boldsymbol{p} \leq 1, \\ & \boldsymbol{0} \preceq \boldsymbol{p} \preceq \boldsymbol{1}, \\ & 0 \leq s. \end{aligned}$$

2. Let $\boldsymbol{p}_k^*$ denote the optimal $\boldsymbol{p}$ obtained at the i-th iteration, and set $\boldsymbol{z}_{i+1}=\boldsymbol{p}_i^*$.
3. Set $i=i+1$.

until convergence
Return the maximum $\lambda_s(\mu_p)$.

To draw the stability region (λ_p versus λ_s) with the delay constraint on the PU, we vary λ_p from zero to λ_p^m and obtain the maximum stable throughput of the SU for each λ_p using Algorithm 1. Then, we get the convex hull for the obtained values. Note that we already know that the stable throughput point

$(\lambda_p, \lambda_s) = (0, f_{sd'})$ is in the stable region. This point corresponds to the case when $\lambda_p = 0$, and hence, the SU is free to transmit its own packets in all time slots achieving its maximum stable throughput $f_{sd'}$.

5 Numerical Results

In this section, we illustrate the impact of having a full duplex SU on the stability region under a PU delay constraint. We compare the stability region for three different configurations (a) full-duplex SU with perfect self interference cancellation $(g = 0)$, (b) full-duplex SU with no self interference cancellation $(g = 1)$, and (c) half-duplex SU such that $f_{ps}^{dup} = 0$. This comparison is shown for four different channel scenarios, i.e., different sets of channel gains variances ρ_{pd}^2 and ρ_{sd}^2, while fixing the rest of the simulation parameters. The fixed parameters are: $P = 10$, $R = 1$, $\rho_{p,s}^2 = 0.6$, $\rho_{s,d'}^2 = 0.8$, and $\rho_{p,d'}^2 = 0.3$.

For the first channel scenario, shown in Fig. 2, we choose high channel gains between $p-d$ and $s-d$ such that $\rho_{pd}^2 = 0.8$ and $\rho_{sd}^2 = 0.9$. Since there is already a good channel between the PU and its destination, most of the packets are successfully transmitted from the PU to d, and hence, the three configurations achieve almost the same maximum stable λ_p (relaying will not be that beneficial in this case). For small λ_p, the SU does not play an important role in delivering the PU packets, and hence, all the three configurations have a close performance. While, for high λ_p, the SU full-duplex capability helps to have faster delivery of the PU packets, allowing the SU to exploit more resources to send its own packets. Thus, as shown in Fig. 2, the stability region achieved by the full-duplex configuration is larger than that of the half-duplex.

In the second scenario, we choose low channel gain between $p-d$ $(\rho_{pd}^2 = 0.04)$ and high channel gain between $s-d$ $(\rho_{sd}^2 = 0.9)$. In this scenario, the SU plays an important role acting as a relay to deliver the PU packets because of the bad $p-d$ channel. It is clear from Fig. 3 that the full-duplex capability significantly improves the stability region compared to that of the half-duplex. The rationale behind this observation is that the full-duplex capability allows the SU to capture more PU packets in the relaying queue to account for the bad $p-d$ channel. In fact, the SU relaying rate dominates the service rate of the PU. On the other hand, the half-duplex capability reduces λ_{sp} as shown in (7) because the SU has to listen more to the PU transmissions, as a result of the bad $p-d$ channel, which reduces the opportunities available for the SU to send its own packets.

In the previous scenarios, we considered a good channel between s and d, which in turn allows the SU to deliver the relayed packets easily to d. Hence, the full-duplex capability was always advantageous and it enlarged the stability region. In the following two scenarios, we consider two other possibilities in which the SU suffers from bad channel to d.

In the third scenario, we choose high channel gain between $p-d$ $(\rho_{pd}^2 = 0.8)$ and low channel gain between $s-d$ $(\rho_{sd}^2 = 0.04)$. Figure 4 shows that the three configurations achieve almost the same maximum stable λ_p because there is already a good channel between the PU and its destination and most of the

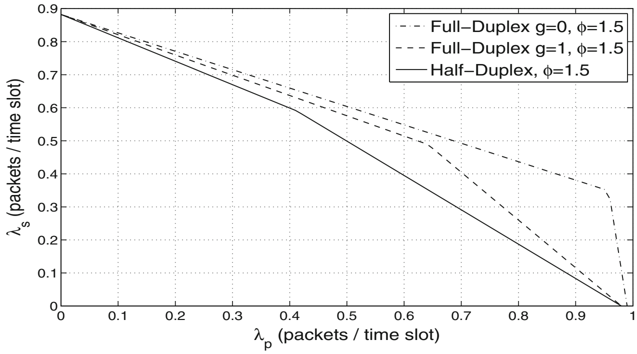

Fig. 2. High channel gain *p*-*d* and high channel gain *s*-*d*.

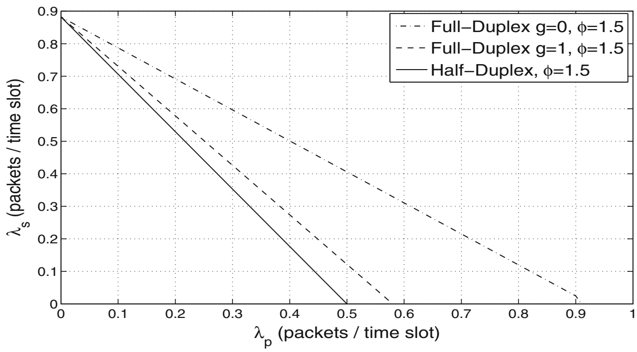

Fig. 3. Low channel gain *p*-*d* and high channel gain *s*-*d*.

packets are successfully transmitted from the PU to the destination on the direct channel. However, none of the three configurations is strictly better over the whole feasible range of λ_p. For small λ_p, the half-duplex configuration provides the best performance, while for large λ_p the full-duplex configuration is the best. The rationale behind this is that for small PU arrival rate λ_p, Q_{sp} is dominating the stability conditions; refer to (9). Hence, having only half-duplex capability reduces the arrival rate of Q_{sp} (7) which, in turn, preserves the stability of Q_{sp} for a wider range of λ_p than that for the full-duplex case. As we increase λ_p, it is clear that full-duplex is becoming the best configuration. This happens because, for large λ_p, Q_p is dominating the stability conditions; refer to (9). Consequently, having the full-duplex capability at the SU increases the service rate of Q_p (4) which, in turn, preserves the stability of Q_p for larger values of λ_p.

In the fourth scenario, we choose low channel gain between $p-d$ ($\rho_{pd}^2 = 0.04$) and low channel gain between $s-d$ ($\rho_{sd}^2 = 0.04$). Figure 5 shows that the half-duplex configuration is strictly better than the full-duplex configuration. For this scenario, when the SU has a full-duplex capability, most of the PU packets have to be delivered to the destination through the relying queue Q_{sp} due to the good

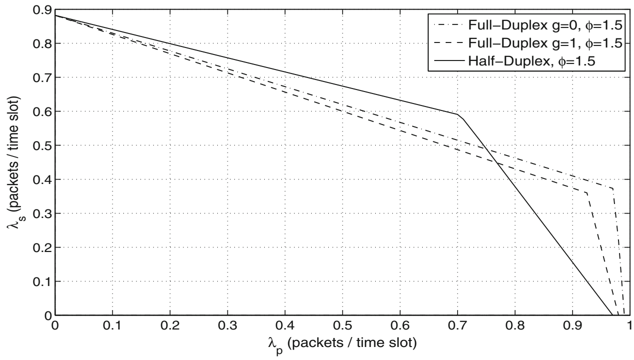

Fig. 4. High channel gain *p*-*d* and low channel gain *s*-*d*.

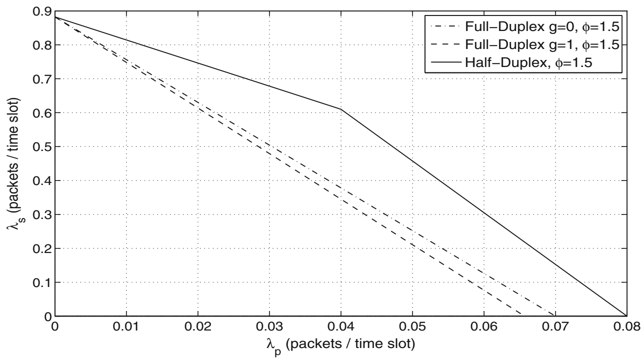

Fig. 5. Low channel gain *p*-*d* and low channel gain *s*-*d*.

channel condition between p and s and bad channel condition between p and d. However, the SU cannot relay these packets due to the bad channel condition $s - d$ causing congestion of the packets at the relaying queue Q_{sp}. On the other hand, having only half-duplex SU reduces the accumulation of packets at Q_{sp} by reducing the arrival rate of Q_{sp} which, in turn, enlarges the stability region. This clearly shows that having a full-duplex capability at the SU is not always beneficial.

Finally, Fig. 6 compares the performance of our adopted FPP-SCA algorithm, presented in Algorithm 1, to the exact results obtained from the exhaustive grid-search based solution. This is performed, without taking the convex hull, for the case where the channel $p - d$ is low and that between $s - d$ is high. Figure 6 Demonstrates that an almost identical performance is achieved by both for the entire range of λ_p. Hence, this shows the efficacy of the adopted FPP-SCA algorithm.

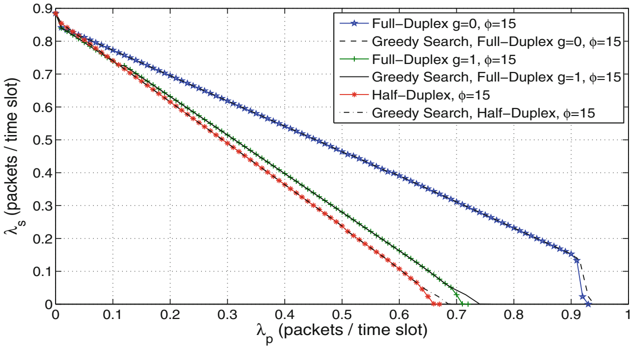

Fig. 6. Comparison between our adopted FPP-SCA algorithm and the greedy search algorithm.

6 Conclusion and Future Work

We have studied cooperative CR networks with the target of maximizing the SU throughput subject to a PU QoS delay constraint. We have also studied the impact of having a full-duplex SU on the network performance compared to having a half-duplex SU. We formulated an optimization problem to maximize the SU throughput subject to a PU delay constraint, which was shown to be non-convex. We proposed to solve the problem by iterating over a set of non-convex QCQP optimization problems and using the FPP-SCA algorithm to solve each iteration. Unexpectedly, our numerical results have revealed that having a full-duplex capability at the SU is not always beneficial and it can adversely affect the stability region of the network in some scenarios. In our future work, we will consider comparing the performance of our proposed cooperation policy to that of the other cooperation and no-cooperation policies presented in the same context. In addition, we will investigate the impact of the SU sensing errors on our system performance. We will also consider finding the optimal trade-off between the PU delay and the SU throughput.

References

1. Tao, X., Xu, X., Cui, Q.: An overview of cooperative communications. IEEE Commun. Mag. **50**(6), 65–71 (2012)
2. Sadek, A.K., Liu, K.R., Ephremides, A.: Cognitive multiple access via cooperation: protocol design and performance analysis. IEEE Trans. Inf. Theory **53**(10), 3677–3696 (2007)
3. Wang, B., Liu, K.R.: Advances in cognitive radio networks: a survey. IEEE J. Sel. Topics Sig. Process. **5**(1), 5–23 (2011)
4. Letaief, K.B., Zhang, W.: Cooperative communications for cognitive radio networks. Proc. IEEE **97**(5), 878–893 (2009)
5. Rong, B., Ephremides, A.: Cooperative access in wireless networks: stable throughput and delay. IEEE Trans. Inf. Theory **58**(9), 5890–5907 (2012)

6. Ashour, M., El-Sherif, A.A., ElBatt, T., Mohamed, A.: Cognitive radio networks with probabilistic relaying: stable throughput and delay tradeoffs. IEEE Trans. Commun. **63**(11), 4002–4014 (2015)
7. Abd-Elmagid, M.A., ElBatt, T., Seddik, K.G., Ercetin, O.: Stable throughput of cooperative cognitive networks with energy harvesting: finite relay buffer and finite battery capacity. IEEE Trans. Cogn. Commun. Netw. **4**(4), 704–718 (2018)
8. Sabharwal, A., Schniter, P., Guo, D., Bliss, D.W., Rangarajan, S., Wichman, R.: In-band full-duplex wireless: challenges and opportunities. IEEE J. Sel. Areas Commun. **32**(9), 1637–1652 (2014)
9. Pappas, N., Ephremides, A., Traganitis, A.: Stability and performance issues of a relay assisted multiple access scheme with MPR capabilities. Comput. Commun. **42**, 70–76 (2014)
10. ElAzzouni, S., Ercetin, O., El-Keyi, A., ElBatt, T., Nafie, M.: Full-duplex cooperative cognitive radio networks. In: 2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), pp. 475–482. IEEE (2015)
11. Elmahdy, A.M., El-Keyi, A., ElBatt, T.A., Seddik, K.G.: Optimizing cooperative cognitive radio networks performance with primary QoS provisioning. IEEE Trans. Commun. **65**(4), 1451–1463 (2017)
12. Ramirez, D., Aazhang, B.: Optimal routing and power allocation for wireless networks with imperfect full-duplex nodes. IEEE Trans. Wirel. Commun. **12**(9), 4692–4704 (2013)
13. Tsybakov, B.S., Mikhailov, V.A.: Ergodicity of a slotted aloha system. Problemy peredachi informatsii **15**(4), 73–87 (1979)
14. Krikidis, I., Devroye, N., Thompson, J.S.: Stability analysis for cognitive radio with multi-access primary transmission. IEEE Trans. Wirel. Commun. **9**(1), 72–77 (2010)
15. Loynes, R.M.: The stability of a queue with non-independent inter-arrival and service times. In: Mathematical Proceedings of the Cambridge Philosophical Society, vol. 58, pp. 497–520. Cambridge University Press (1962)
16. Jackson, J.R.: Jobshop-like queueing systems. Manag. Sci. **50**(Suppl_12), 1796–1802 (2004)
17. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
18. Mehanna, O., Huang, K., Gopalakrishnan, B., Konar, A., Sidiropoulos, N.D.: Feasible point pursuit and successive approximation of non-convex QCQPs. IEEE Sig. Process. Lett. **22**(7), 804–808 (2015)