This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

Network Coding Techniques for Primary-Secondary User Cooperation in Cognitive Radio Networks

Athanasios Papadopoulos, *Student Member, IEEE,* Nestor D. Chatzidiamantis, *Member, IEEE,* and Leonidas Georgiadis, *Senior Member, IEEE*

***Abstract*—In this paper, we investigate transmission techniques for a fundamental cooperative cognitive radio network, i.e., a cognitive radio system where a Secondary user may act as relay for messages sent by the Primary user, hence offering performance improvement of Primary user transmissions, while at the same time obtaining more transmission opportunities for its own transmissions. Specifically, we examine the possibility of improving the overall system performance by employing network coding techniques. The objective is to achieve this while affecting Primary user transmissions only positively, namely: 1) avoid network coding operations at the Primary transmitter, hence avoiding increase of its storage requirements and keeping its complexity low, 2) keep the order of packets received by the Primary receiver the same as in the non cooperative case and 3) induce packet service times that are stochastically smaller than the packet service times induced in the non-cooperative case. A network coding algorithm is investigated in terms of achieved throughput region and it is shown to enlarge Secondary user throughput as compared to the case where the Secondary transmitter acts as a simple relay, while leaving the Primary user stability region unaffected. A notable feature of this algorithm is that it operates without knowledge of channel and packet arrival rate statistics. We further present a second network coding algorithm which increases the throughput region of the system under certain conditions on system parameters; however, the latter algorithm requires knowledge of channel and packet arrival rate statistics.**

***Index Terms*—Cognitive radio, Network coding, cooperative cognitive networks, Primary User, Secondary User, throughput region.**

I. Introduction

Cognitive radio networks (CRNs) received considerable attention due to their potential for improving spectral efficiency [2]. The main idea behind CRNs is to allow unlicensed users, known as Secondary users, to identify spatially or temporally available spectrum and gain access to the underutilized shared spectrum, while maintaining limited interference to the licensed user, also known as Primary user.

Initial designs of CRNs assumed that no interaction between Primary and Secondary users exists (see [3] and the references therein). Of particular interest are the works of [4], [5] which addressed the problem of optimal spectrum assignment to multiple Secondary users and presented resource allocation algorithms based on either the knowledge of Primary user transmissions obtained from perfect spectrum sensing mechanisms [4] or from a probabilistic maximum collision constraint with the Primary Users [5]. Furthermore, in this framework, an opportunistic scheduling policy was suggested in [6], which offered maximization of throughput utility for the Secondary users while providing guarantees on the number of collisions with the Primary user, as well.

By allowing cooperation between Primary and Secondary users in CRNs, cooperative CRNs have emerged. Cooperative CRNs have gained attention due to their potential of providing benefits for both types of users. Specifically, by allowing Secondary users to relay Primary user transmissions, the channel between the Secondary transmitter and Primary receiver is exploited, improving the effective transmission rate of the Primary channel which as a result becomes idle more often, hence providing more transmission opportunities to the Secondary users.

Due to their advantages cooperative CRNs have been studied in several research works. From an information theoretic perspective, cooperation between Secondary and Primary users at the Physical layer has been investigated in [7]. Advanced PHY-layer transmission techniques for cooperative CRNs have been presented in [8], [9]. Specifically, employing non-orthogonal multiple access (NOMA) techniques based on successive interference cancellation (SIC), PHY-layer transmission protocols were proposed, which exploit the merits of cooperation for both Primary and Secondary users.

Of particular interest are the works which conduct queuing theoretic analysis and transmission protocol design for cooperative CRNs [10]–[15]. A cooperation transmission protocol for CRNs where the Secondary user acts as a relay for Primary user transmissions was initially presented in [10], where the benefits of such cooperation for both types of users were investigated. In [11], cooperative CRNs with multiple Secondary users were investigated and advanced relaying techniques which involved advanced Physical layer coding between Primary and Secondary transmissions were suggested. Stationary transmission policies that allow simultaneous Primary and Secondary user transmissions were designed and optimized in [12], in terms of stable throughput region. Cooperation transmission policies which take into account the available power resources at the Secondary transmitter in order for the latter to decide whether to cooperate or not, have been presented in [13], [14]; in these works cooperation between Primary and Secondary users is treated in an abstract manner (when cooperation takes place, transmission success

The work of Athanasios Papadopoulos was funded by ELIDEK (August 2017-February 2019) and Onassis Foundation (October 2015-August 2017).

This work was presented in part at Global Information Infrastructure and Networking Symposium (GIIS 2018), Thessaloniki, Greece, [1].

A. Papadopoulos, N. D. Chatzidiamantis and L. Georgiadis are with the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, 54 124, Thessaloniki, Greece (e-mails:{athanapg,nestoras,leonid}@auth.gr).

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

probability is improved) without addressing in detail how this cooperation is effected. Moreover, a cooperative CRNs with an extra dedicated relay was investigated in [15] and the maximum Secondary user throughput for this setup was determined. It should be noted that the implementation of all these transmission algorithms for cooperative CRNs requires the modification of certain Primary User's parameters (such as Primary transmitted power, transmitted codewords, order of Primary transmitter packets received by Primary receiver) as compared to the non-cooperative case, in order for the cooperation between Primary and Secondary users to take place.

Network coding is a well known transmission technique that has been applied in CRNs as a means for enhancing the throughput of both Primary and Secondary users (see [16] and the references therein). However, only recently, the application of network coding between Primary and Secondary users packets has been proposed in [17]. While this technique has the potential to achieve capacity, the implementation presented in [17] leaves room for improvement since it misses a number of opportunities for transmitting network coded packets. Furthermore, the algorithm presented in [17] is frame based which implies that a) detailed knowledge of channel statistics is required to implement the algorithm, and b) the service times and delay characteristics of Primary user packets may be negatively affected as compared to the non-cooperative case.

In the current work we examine the possibility of employing efficient network coding techniques between Primary and Secondary users' packets to further improve the performance of cooperative CRNs. We propose a network coding algorithm that not only enhances the throughput region of Primary and Secondary users, but strictly improves the service times of the Primary user packets as well. In addition, the proposed transmission algorithm does not require any knowledge of statistical parameters for its operation. Furthermore, we present a modification of this algorithm that enhances the throughput region of the system, however, requires knowledge of channel statistics for its operation. Specifically, the contribution of the paper can be summarized as follows:

- We propose an algorithm that implements network coding in the cooperative cognitive radio network and study its performance in terms of Primary-Secondary throughput region. This setup imposes the requirement that the Primary transmitter implementation complexity is minimally affected, i.e., no network coding operations are imposed on the Primary transmitter. Moreover, the order of Primary channel packet reception is required to remain unaltered, while the service times of Primary packets must be improved compared to the case when no cooperation takes place. Under the aforementioned requirements, the proposed algorithm allows the Secondary transmitter to either act as relay for Primary transmitter packets or transmit network-coded packets (which enable the simultaneous reception by both the Primary and Secondary receivers). Employing queuing theory, the performance of the presented algorithm is evaluated and its throughput region is derived in closed form. A notable feature of the algorithm is that the only requirement for its operation is knowledge that the channel from Secondary transmitter to Primary receiver is better than the channel from Primary transmitter to Primary receiver.
- We examine the possibility whether it is possible to further increase the throughput region of the system by employing more sophisticated network coding techniques. To this end, we present a generalization of the proposed algorithm and show that significant improvements are observed for certain values of channel statistics. However, the implementation of this algorithm requires knowledge of channel statistics, as well as the arrival rate of Primary transmitter packets.

The remainder of the paper is organized as follows. Section II presents the system model. In Section III, two baseline algorithms are described, which will be used for comparison purposes. Section IV describes a transmission algorithm that is based on network coding. In Section V we present a generalization of the algorithm proposed in Section IV and show that it improves throughput region in certain cases. Numerical and simulation results are illustrated in Section VI, and, finally, concluding remarks are provided in Section VII.

II. System Model

We consider the four-node cognitive radio system model depicted in Figure 1. The system consists of two (transmitter, receiver) pairs (1,3), (2,4). Pair (1,3) - odd numbers- represents the primary channel. Node 1 is the primary transmitter who is the licensed owner of the channel and transmits whenever it has data to send to primary receiver, node 3. On the other hand, node 2 is the secondary transmitter; this node does not have any licensed spectrum and seeks transmission opportunities on the primary channel in order to deliver data to secondary receiver, node 4.

- *Time and unit of transmission model.* We consider the time-slotted model, where time slot $t = 0, 1, ...$ corresponds to time interval $[t, t+1)$; t and $t+1$ are called the "beginning" and "end" of slot t respectively. Information transmission consists of fixed size bits of packets whose transmission takes unit time. At the beginning of time slot t, a random number $A^1(t)$ of packets arrive at node 1 with destination node 3, thereafter called packets of session $(1, 3)$. These packets are stored in an infinite-size queue Q_1. We assume that the random variables $\left\{A^1(t)\right\}_{t=0}^{\infty}$ are independent and identically distributed (i.i.d.) with mean $\lambda_1 = \mathbb{E}\left[A^1(t)\right]$. Node 2 has an infinite number of packets destined to node 4, stored in queue Q_2, thereafter called packets of session $(2, 4)$. The latter assumption amounts to assuming that node 2 is overloaded and is made in order to simplify and clarify the presentation; with an apparent modification the algorithms presented still work and the results hold when packet arrive at node 2 randomly (see Appendix B).
- *Channel Model.* We consider the wireless broadcast channel, i.e., that transmissions by node i, $i \in \{1, 2\}$ may be heard by the rest of the nodes. We adopt the broadcast erasure channel model which efficiently models

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication. The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

communication at the MAC layer. In this channel model, a transmission by node i, $i \in \{1, 2\}$, may either be received correctly by or erased at each of the other nodes. Specifically, we make the following assumptions regarding the channel.

– *Erasure events.* We assume that reception/erasure events are independent across time slots, however, we allow for the possibility that they are dependent within a given time slot. Specifically, for a node $i \in \{1, 2\}$, let $\left\{Z_j^i(t)\right\}_{t=1}^{\infty}$ with $j \in \mathcal{N}_i$, $\mathcal{N}_1 = \{2, 3, 4\}$ and $\mathcal{N}_2 = \{3, 4\}$, be random variables denoting erasure events, taking values 1 (a packet transmitted by node i is received by node j) and 0 (a packet transmitted by node i is erased at node j). We assume that the set of random variables $\left\{Z_j^i(t)\right\}_{j\in\mathcal{N}_i,\ i\in\{1,2\}}$ are independent for $t = 0, 1, ...$; however, for given t, we allow for arbitrary dependence between the random variables in this set. We denote by $\epsilon_{\mathcal{S}}^i$, $\mathcal{S} \subseteq \mathcal{N}_i$ the probability that a packet transmitted by node i is erased at all nodes in set $\mathcal{S}$. For simplicity we omit the brackets when denoting specific sets in $\epsilon_{\mathcal{S}}^i$. For example, ϵ_{23}^1 is the probability that a transmission by node 1 is erased at nodes $2, 3$; the transmission may either be received correctly or erased at node 4. In the following, we assume that $\epsilon_{\mathcal{S}}^i < 1$.
– *Transmission scheduling*. We assume that simultaneous transmission of packets by both transmitters results in loss of both packets; hence, for useful transfer of information, only one of the transmitters must be scheduled to transmit at any given time.
– *Channel feedback.* Upon reception or erasure of a packet, a node sends respectively positive (ACK) or negative (NACK) acknowledgment on a separate channel, which is heard by the rest of the nodes.
– *Channel sensing.* We assume that the Secondary transmitter can sense whether the Primary transmitter is sending a packet on the channel.

A main requirement in this setup is that node 2 transmissions must either have no negative effect, or effect positively node 1 transmissions. In the simplest case this can be achieved if transmitter 2 sends data to receiver 4 only when transmitter 1 is idle. In this case, nodes 1 and 3 are effectively unaware of transmissions that take place between the secondary pair (2,4). However, if the erasure probability from node 2 to node 3 is smaller than the one from node 1 to node 3, i.e., $\epsilon_3^2 < \epsilon_3^1$, the possibility arises for improving the performance of both the primary and the secondary channel by cooperation. Specifically, node 2 may store packets sent by node 1 and erased at node 3 and then act as a relay to transfer these packets to node 3. Since $\epsilon_3^2 < \epsilon_3^1$, this transfer will take shorter time. As a result the throughput and packet delays for session (1,3) will improve and at the same time, as long as λ_1 is not very high, node 1 will be idle for a longer time and the throughput of packets for session (2,4) will also increase.

In this work we examine the possibility of improving further the throughput of packets of session (2,4) by allowing network coded transmissions by node 2. We propose a network coding

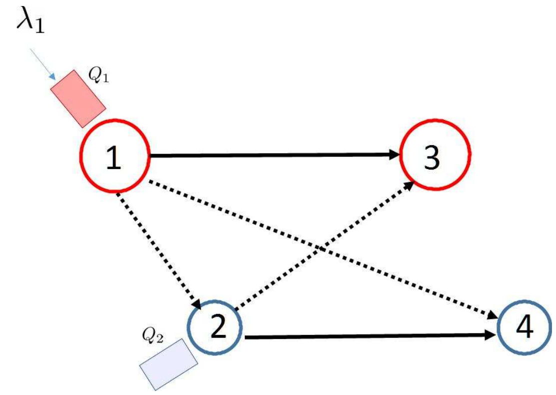


Fig. 1. The system model under consideration.

based algorithm according to which node 2 may transmit appropriate combinations of packets destined to nodes 3, 4 which result in increased throughput of packets of session (2,4). However, since node 1 is the owner of the communication channel, in order to ensure that session (1,3) transmissions are only positively affected, we impose the following requirements on the design of coding algorithms.

Algorithm Design Requirements

1) No coding operations takes place at transmitter node 1. Node 1 transmits its own packets based on the feedback received by nodes 2, 3, 4, but does not receive/process any of the packets transmitted by node 2.
2) The order of packet transmission of session (1,3) must be the same as in the case where no cooperation takes place.
3) The service time of each packet of session (1,3) (i.e. the time interval between the time the packet is at the head of the queue on node 1 and the time the packet is successfully received by node 3) must be "smaller" than the service time this packet would have if no cooperation took place. Specifically, we require that if S_l^{nc} (S_l^c) are the service times of the lth packet of session (1,3) when no cooperation (cooperation) takes place, then S_l^c is stochastically smaller than S_l^{nc}, that is,

$$\Pr\left(S_l^c \geq x\right) \leq \Pr\left(S_l^{nc} \geq x\right), \text{ for all } x \in [0, \infty).$$

An algorithm that satisfies all three requirements stated above will be called "admissible".

Requirements 2 and 3 above are imposed by the need to avoid negative effects on the performance of primary session (1,3) which has priority. It may seem at first that these restrictions may result in missing coding opportunities and therefore reduce the achievable throughput region of the system. However, in our recent work [18] where the information theoretic capacity of the system without imposing the restrictions of the items 2 and 3 was investigated, it was shown that the restrictions of the items 2 and 3 do not result in any loss system throughput.

A. Definitions and Preliminary Results

In the rest of this paper, for any storage element X we denote by $X(t)$ the number of packets in this element at time

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

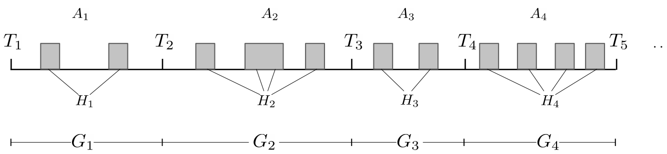


Fig. 2. The time-slotted model of a generic queue.

t.

A sequence of non-negative random variables $\{Y(t)\}_{t=0}^{\infty}$ is stable if it converges in distribution to a proper random variable, i.e, $\lim_{t\to\infty} \Pr(Y(t) > M) = f(M)$ for all $M \geq 0$, and $\lim_{M\to\infty} f(M) = 0$.

One objective of the performance analysis of the algorithms to be presented in the next sections is to determine the set of arrival rates λ_1 for which the number of primary session (1,3) packets *in the system* at time t, denoted by $Q_1^S(t)$[1], is stable. It will be seen that under the admissible algorithms discussed in this paper, $Q_1^S(t) = Q_1(t) + F_2(t)$ where $F_2(t)$ is a random variable taking values in $\{0,1\}$ and denotes the number of session (1,3) packets that may be located at node 2. Also, $Q_1^S(t)$ can be seen as the queue size of a discrete time queue where packets have independent identically distributed (i.i.d.) service times with general distribution with mean $\bar{S}_1$. Discrete time queues of this type have been studied in [19] where it is shown that $Q_1^S(t)$ is stable when

$$0 \leq \lambda_1 < \mu_1, \tag{1}$$

where $\mu_1 = \frac{1}{\bar{S}_1}$.

Moreover, the average length of the busy and idle periods of $Q_1^S(t)$ are given respectively by,

$$\bar{B}_1 = \frac{\lambda_1/\mu_1}{(1-\lambda_1/\mu_1)(1-q_0)}, \tag{2}$$

$$\bar{I}_1 = \frac{1}{(1-q_0)}, \tag{3}$$

where $q_0 = 1 - \Pr(A_1(t) = 0)$.

Let $R_i(t),\ i = 1,2$ be the number of packets of session $(i, i+2)$ received by node $i+2$ during time slot t. The throughput r_i of session $(i, i+2),\ i = 1,2$ is defined as

$$r_i = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} R_i(\tau). \tag{4}$$

It will be seen that for the algorithms discussed in this paper the limit in (4) exists.

The objective of the algorithms presented in the next section is to evaluate the maximum rate r_2 of session (2,4) packets that can be obtained for given λ_1 satisfying condition (1); under the latter condition, it is well known that it holds, $\lambda_1 = r_1$. The closure of the set of pairs (r_1, r_2) that can be obtained under an algorithm is called "throughput region" of the algorithm and is denoted by $\mathcal{R}$.

Next we present a *generic queuing system* that will be used for the performance analysis of the algorithms to be described in the next sections. Consider the slotted time system, depicted in Figure 2, with the following structure. There are random time instants $T_n,\ n = 1, 2, \ldots$ forming a renewal process, i.e., $G_n = T_{n+1} - T_n \geq 1$ are i.i.d. with finite expectation. A random number H_n of the slots in the time interval $[T_n, T_{n+1})$ are available for transmitting the packets that are in the queue when this interval starts; the rest of the slots are not available. Also, in the time interval $[T_n, T_{n+1})$ a random number A_n of packets arrives at the queue at various times; these packets are stored in an infinite size queue and can be served at or after slot T_n. The lth arriving packet needs a random number S_l of the available slots in order to be transmitted successfully. The random variables $\{A_n\}_{n=1}^{\infty}$, $\{H_n\}_{n=1}^{\infty}$, $\{S_l\}_{l=1}^{\infty}$ are i.i.d., and independent of each other, with finite expectations. Let r be the throughput of packets served by this queue. Using arguments similar to those in [20, Section 2] it can be shown that

$$\text{If } \mathbb{E}[A_1] \leq \frac{\mathbb{E}[H_2]}{\mathbb{E}[S_1]}, \quad \text{then } r = \frac{\mathbb{E}[A_1]}{\mathbb{E}[G_1]}. \tag{5}$$

$$\text{If } \mathbb{E}[A_1] > \frac{\mathbb{E}[H_2]}{\mathbb{E}[S_1]}, \quad \text{then } \ r = \frac{\mathbb{E}[H_2]}{\mathbb{E}[G_1]\,\mathbb{E}[S_1]}. \tag{6}$$

A special case of this system is the discrete time queue in [19] which is obtained by setting $T_n = n$, $G_n = 1$, and $H_n = 1$.

III. Baseline Algorithms

In this section we describe two baseline algorithms. The first involves no cooperation while in the second the secondary transmitter may be used as relay for session $(1,3)$ packets, but performs no network coding operations.

A. No Cooperation

The no cooperation algorithm, referred to in the following as Algorithm I, is very simple and requires no cooperation between the Primary and Secondary users. Its operation is described as follows:

Algorithm I

1) If Q_1 is nonempty, node 1 (re)transmits the packet at the head of Q_1 until it is received by node 3.
2) If Q_1 is empty, node 2 (re)transmits the head of Q_2 packet until it is received by node 4.

It can be easily seen that the throughput region of this algorithm is given by [1]

$$\mathcal{R}_{\mathbf{I}} = \left\{(r_1, r_2) \geq 0 : \frac{r_1}{1-\epsilon_3^1} + \frac{r_2}{1-\epsilon_4^2} \leq 1\right\}. \tag{7}$$

B. Simple Forwarding

The algorithms presented in this and the following sections are admissible when the channel from node 2 to node 3 is "better" than the channel from node 1 to node 3. Specifically we assume for the rest of this work that

$$\epsilon_3^1 \geq \epsilon_3^2. \tag{8}$$

While the algorithms to be presented are operational even if condition (8) is not satisfied, they are not admissible because they violate item 3 of Algorithm Design Requirements presented in Section II.

[1]That is a virtual queue that contains all the session (1,3) packets that exist in the system (located at either the Primary or Secondary transmitters).

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication. The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

In [11], the following algorithm, referred in what follows as Algorithm II, was presented:

Algorithm II

1) If Q_1 is nonempty, node 1 (re)transmits the packet at the head of Q_1 until it is received by either node 2 or node 3.
   a) If the packet is received by node 2 and erased at node 3, it is stored in a queue $Q_{2,\bar{3}}$ at node 2.
2) If Q_1 is empty and $Q_{2,\bar{3}}$ nonempty, node 2 (re)transmits the packet at the head of queue $Q_{2,\bar{3}}$ until it is received by node 3.
3) If Q_1 and $Q_{2,\bar{3}}$ are empty, node 2 (re)transmits the packet at the head of queue Q_2 until it is received by node 4.

This algorithm is not admissible since it violates items 2 and 3 of Algorithm Design Requirements presented in Section II. However, a slight modification makes this algorithm admissible. Specifically, in the next algorithm, referred to as Algorithm III, node 2 maintains a single-packet buffer B_2 (used for storing the packet received by node 1) and the following actions are taken:

Algorithm III

1) If Q_1 is nonempty and B_2 is empty, node 1 (re)transmits the packet at the head of Q_1 until it is received by either node 2 or node 3.
   a) If the packet is received by node 2 and erased at node 3, it is stored in buffer B_2 at node 2.
2) If B_2 is nonempty, node 2 (re)transmits the single packet in B_2 until it is received by node 3.
3) If Q_1 and B_2 are empty, node 2 (re)transmits the packet at the head of queue Q_2 until it is received by node 4.

The main difference of Algorithm III from Algorithm II is that if a session (1,3) packet is received by node 2 and erased at node 3, then node 2 starts re-transmitting immediately the packet instead of storing it in a buffer and transmitting it when Q_1 becomes empty. This modification makes the algorithm admissible. Indeed, items 1, 2 of Algorithm Design Requirements are obviously satisfied. Item 3 is also satisfied, as stated in the next proposition.

Proposition 1. Algorithm III satisfies item 3 of Algorithm Design Requirements, i.e, if S_l^{nc} (S_l^c) are the service times of the lth packet of session (1,3) when no cooperation (cooperation) takes place, then S_l^c is stochastically smaller than S_l^{nc}.

Proof. The proof is given in Appendix A. □

Since the only difference between Algorithms II and III is the order in which packets are transmitted, the maximum stable arrival rate of primary session (1,3) and the maximum throughput of secondary session (2,4) packets are the same under both algorithms and as shown in [11] they are given by the formulas below:

$$0 \le r_1 < \frac{\left(1-\epsilon_3^2\right)\left(1-\epsilon_{23}^1\right)}{1-\epsilon_3^2+\epsilon_3^1-\epsilon_{23}^1} = \mu_1^{\mathbf{III}}, \tag{9}$$

$$0 \le r_2 \le \left(1 - r_1 \frac{1-\epsilon_3^2+\epsilon_3^1-\epsilon_{23}^1}{(1-\epsilon_3^2)\,(1-\epsilon_{23}^1)}\right)\left(1-\epsilon_4^2\right). \tag{10}$$

Hence the throughput region of Algorithm III is

$$\mathcal{R}_{\mathbf{III}} = \left\{(r_1, r_2) \ge 0 : \frac{1-\epsilon_3^2+\epsilon_3^1-\epsilon_{23}^1}{(1-\epsilon_3^2)\,(1-\epsilon_{23}^1)} r_1 + \frac{r_2}{1-\epsilon_4^2} \le 1\right\}. \tag{11}$$

IV. Proposed Network Coding Algorithm

In this section we propose an admissible scheduling algorithm that at appropriate times, depending on events occurring during system operation, transmits network-coded packets. The proposed algorithm is admissible and enhances the maximum throughput of secondary session (2,4), while leaving the maximum throughput of primary session (1,3) achieved by Algorithm III unaltered. For the operation of the proposed algorithm, the following structures are maintained at the nodes.

Algorithm IV

1) Two single-packet buffers at node 2, denoted as[2] $B^1_{2,\bar{3}\bar{4}}$ and $B^1_{2,\bar{3}4}$, for storing certain packets of session (1,3) transmitted by node 1 and received by node 2. Buffer $B^1_{2,\bar{3}\bar{4}}$ holds packets that are received by node 2 and erased at 3, 4 . Buffer $B^1_{2,\bar{3}4}$ holds packets that are received by nodes 2, 4 and erased at node 3. The operation of the algorithm ensures that these buffers hold at most one packet. Moreover, at most one of these buffers may be nonempty at the beginning of each time slot.
2) One infinite-size queue at node 2, denoted as $Q_{2,3\bar{4}}$, for storing packets of session (2,4) received by node 3 and erased at node 4.
3) One single-packet buffer and node 4, denoted as $B^1_{4,\bar{3}}$, for storing packets of session (1,3) erased at node 3 and received by node 4. The operation of the algorithm ensures that if buffer $B^1_{2,\bar{3}4}$ contains one packet, this packet is also stored in $B^1_{4,\bar{3}}$ at node 4.
4) One infinite-size queue at node 3, denoted as $Q^2_{3,\bar{4}}$, for storing packets of session (2,4) erased at node 4 and received by node 3. The operation of the algorithm ensures that the contents of $Q^2_{3,\bar{4}}$ are the same as those of $Q_{2,3\bar{4}}$.

Next we present the details of the operation of the algorithm, referred to as Algorithm IV. Depending on the status of a transmitted packet at each of the nodes (reception or erasure) various actions are taken by the nodes. Since each node sends (ACK, NACK) feedback that is heard by the rest of them, the nodes are able to perform the actions required by the algorithm. In addition the state of Q_1 (empty or nonempty) can be obtained by node 2 by sensing the channel.

- If Q_1, $B^1_{2,\bar{3}\bar{4}}$, $B^1_{2,\bar{3}4}$ are all empty, i.e., Q_1^S is empty, implying that there are no session (1,3) packets in the network, node 2 (re)transmits the packet at the head of Q_2 until it is received by at least one of the nodes 3, 4. If the packet is received by node 4, it is removed from Q_2. If the packet is received by node 3 and erased at node 4,

[2] For easy reference we use the following convention in the notation: storage element $X^i_{j,k\bar{l}}$ holds packets located at node j, that have been originated and transmitted by node i, are received by node k and erased at node l. If the superscript is missing, e.g., $X_{i,k\bar{l}}$ then $X_{i,k\bar{l}}$ holds packets that originated at node i.

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication. The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

it is removed from Q_2 and placed in $Q_{2,3\bar{4}}$; also, node 3 stores the packet in $Q^2_{3,\bar{4}}$. As will be explained shortly, the packets stored in $Q_{2,3\bar{4}}$ are candidates for network coding and are used by node 2 to form network-coded packets during the times that Q_1 is nonempty.

- If queue Q_1 is nonempty and buffers $B^1_{2,\bar{3}\bar{4}}$ and $B^1_{2,\bar{3}4}$ are empty, which implies that no packet of session (1,3) is stored at node 2, node 1 (re)transmits the packet at the head of Q_1 until it is received by at least one of the nodes 2, 3, say at time t. During this process, if node 4 receives the transmitted packet, it stores it in buffer $B^1_{4,\bar{3}}$. If at time t the transmitted packet is received by node 3, the packet at node $B^1_{4,\bar{3}}$ (if any) is removed. If at time t the packet is erased at node 3 and received by node 2, the packet is placed in $B^1_{2,\bar{3}\bar{4}}$ if $B^1_{4,\bar{3}}$ is empty (i.e., node 4 has not received the packet), and in $B^1_{2,\bar{3}4}$ otherwise; in this case, the packet is removed form Q_1 and node 2 starts the attempt to deliver the packet (stored in one of the buffers $B^1_{2,\bar{3}\bar{4}}$, $B^1_{2,\bar{3}4}$) until it is received by node 3 as described next. Observe that at time t only one of $B^1_{2,\bar{3}\bar{4}}$ and $B^1_{2,\bar{3}4}$ can be nonempty. Moreover, if $B^1_{2,\bar{3}4}$ is nonempty, $B^1_{4,\bar{3}}$ contains the same packet.
  - If $B^1_{2,\bar{3}\bar{4}}$ is nonempty (hence $B^1_{2,\bar{3}4}$ is empty), node 2 transmits the packet in $B^1_{2,\bar{3}\bar{4}}$ until it is received by at least one of the nodes 3, 4. At this time, the packet is removed from $B^1_{2,\bar{3}\bar{4}}$. Moreover, if the packet is erased at node 3 and received by node 4, it is moved to $B^1_{2,\bar{3}4}$ and it is also placed in $B^1_{4,\bar{3}}$. We see again that at time t only one of $B^1_{2,\bar{3}\bar{4}}$ and $B^1_{2,\bar{3}4}$can be nonempty.
  - If $B^1_{2,\bar{3}4}$ is nonempty (hence $B^1_{2,\bar{3}\bar{4}}$ is empty and $B^1_{4,\bar{3}}$ is nonempty) then,
    * if $Q_{2,3\bar{4}}$ is empty (hence $Q^2_{4,\bar{3}}$ is also empty), node 2 transmits the packet in $B^1_{2,\bar{3}4}$ until it is received by node 3, at which time the packet is removed from $B^1_{2,\bar{3}4}$ and $B^1_{4,\bar{3}}$.
    * if $Q_{2,3\bar{4}}$ is nonempty (hence $Q^2_{3,\bar{4}}$ is nonempty as well), then the opportunity for network coding arises. Indeed, let q_1 and q_2 be the packets stored in $B^1_{2,\bar{3}4}$ and $Q_{2,3\bar{4}}$ respectively. Packet q_1 is a session (1,3) packet, unknown to node 3 and received by node 4 (it is the packet stored in $B^1_{4,\bar{3}}$). Packet q_2 is a session (2,4) packet unknown to node 4 and received by node 3 (it is the packet stored in $Q^2_{3,\bar{4}}$). Hence node 2 sends packet $q = q_1 \oplus q_2$, where $\oplus$ denotes XOR operation, and if any node in $\{2, 4\}$ receives q, that node can decode the packet destined to it. For example, if node 3 receives packet p, then $q_1 = p \oplus q_2$.

Algorithm IV is admissible. In fact, the order and service times of session (1,3) packets are exactly the same as in Algorithm III. The only difference is that at certain times during which Q^S_1 is nonempty, some of these packets are network-coded with packets of session (2,4). This network coding operation does not alter the time the packet is delivered to node 3, but allows the increase of throughput for packets of session (2,4) by allowing for the possibility of simultaneous reception of packets by nodes 3, 4, using a single transmission by node 2.

A. *Performance Analysis of Network Coding Algorithm*

In this section we calculate the throughput region of Algorithm IV. We first provide an outline of the analysis. For a session (1,3) packet q, let t^q_s and t^q_r be respectively the time when node 1 starts transmitting the packet and the time node 3 receives it - note that according to the algorithm the packet may have been transmitted to node 3 by node 2. The "service time" of the packet is then $t^q_r - t^q_s$. Due to the operation of Algorithm IV and the statistical assumptions, all session (1,3) packets have the same distribution of service time. We denote by $\bar{S}^{\mathbf{IV}}_1$ the expected value of the service time of a session (1,3) packet achieved by applying Algorithm IV and provide a method for calculating it. The queue Q^S_1, discussed in Section II-A, consisting of all session (1,3) packets that are in the system (at node 1 or node 2), may be viewed as a discrete time queue with average packet service time $\bar{S}^{\mathbf{IV}}_1$. Based on this observation the maximum packet arrival rate λ_1 for which queue Q^S_1 is stable is given by

$$0 \le \lambda_1 = r_1 < \frac{1}{\bar{S}^{\mathbf{IV}}_1} \triangleq \mu^{\mathbf{IV}}_1. \qquad (12)$$

Next, given λ_1, we calculate the throughput for session (2,4) packets. For this, we observe that queue $Q_{2,3\bar{4}}$ is of the "generic type" discussed at the end of Section II-A, where T_n is the time when the nth busy period of queue Q^S_1 starts. The throughput of packets entering this queue and, hence, are to be delivered to node 4, can be determined through (5)-(6) after calculating the parameters involved in these formulas. The throughput of session (2,4) packets is then the sum of the throughput of packets entering $Q_{2,3\bar{4}}$ and the throughput of packets delivered by node 2 directly to node 4 during the times when queue Q^S_1 is empty.

We now proceed with the detailed analysis. Since as mentioned in Section IV the service times of session (1,3) packets (i.e., the service times of Q^S_1 packets) under Algorithm IV are the same as those induced by Algorithm III, we immediately conclude from (9) that

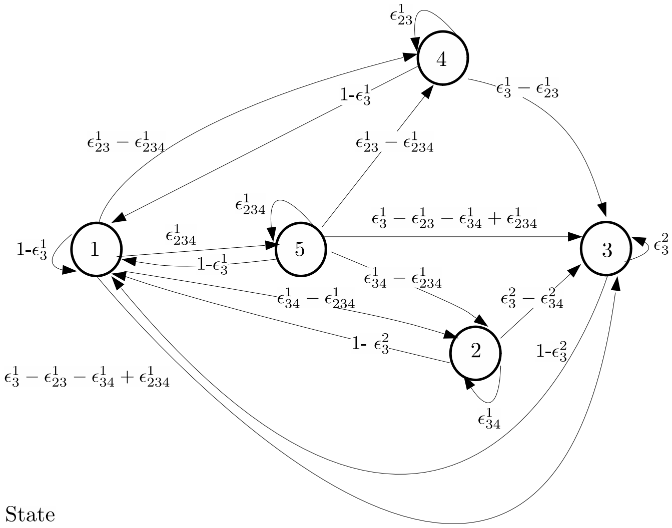
$$\mu^{\mathbf{IV}}_1 = \mu^{\mathbf{III}}_1 = \frac{\left(1-\epsilon^2_3\right)\left(1-\epsilon^1_{23}\right)}{1-\epsilon^2_3+\epsilon^1_3-\epsilon^1_{23}}. \qquad (13)$$

For the purposes of calculating the appropriate parameters of $Q_{2,3\bar{4}}$ needed in formula (44) we need to examine the service times of session (1,3) packets under Algorithm IV in more detail. From its operation it can be seen that the service time of a packet under Algorithm IV is equal to the (random) length of time needed for successive returns to state 1 of the Markov Chain described in Figure 3. In this figure, the formulas next to each arrow describe the transition probabilities of the Markov Chain. To see this, assume that node 1 begins transmission of a new packet from Q_1, hence the Markov Chain is in state 1. At this state:

- If the packet (sent from Q_1) is erased at node 3, and received by nodes 2, 4, an event with probability $\epsilon^1_3 - \epsilon^1_{23} - \epsilon^1_{34} + \epsilon^1_{234}$, then the packet is stored in buffers $B^1_{2,\bar{3}4}$ and $B^1_{4,\bar{3}}$ and node 2 begins transmission of the

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821



State

1 : Node 1 transmits new packet from Q_1.
2 : Node 2 transmits packet from buffer $B^1_{2,\bar{3}\bar{4}}$.
3 : Node 2 transmits (possibly network-coded) packet buffer $B^1_{2,\bar{3}4}$.
4 : Node 1 transmits packet from Q_1, received by node 4.
5 : Node 1 retransmits packet from Q_1, not received by any of nodes $\{2, 3, 4\}$.

Fig. 3. Markov chain describing service times of session (1,3) packets induced by Algorithm IV.

packet in $B^1_{2,\bar{3}4}$ in the next slot (note that if queue $Q_{2,3\bar{4}}$ is nonempty, the packet in $B^1_{2,\bar{3}4}$ is transmitted network-coded with the head of line of packet of $Q_{2,3\bar{4}}$) , i.e., the chain moves to state 3. At this state:

- If upon transmission by node 2 the packet is received by node 3, an event of probability $1-\epsilon_3^2$, the service time of the packet completes and we return to state 1.
- If upon transmission by node 2 the packet is erased at node 3, an event with probability ϵ_3^2, we remain at state 3.

- If the packet (sent from Q_1) is erased at nodes 3, 4 and received by node 2, an event with probability $\epsilon^1_{34} - \epsilon^1_{234}$, then the packet is stored in buffer $B^1_{2,\bar{3}\bar{4}}$ and node 2 begins transmission of the packet in $B^1_{2,\bar{3}\bar{4}}$ in the next slot, i.e., the chain moves to state 2. At this state:
  - If upon transmission by node 2 the packet is received is received by node 3, an event with probability $1 - \epsilon_3^2$, the service time of the packet completes and we return to state 1.
  - If upon transmission by node 2 the packet is erased at node 3 and received by node 4, an event with probability $\epsilon_3^2 - \epsilon_{34}^2$, the packet is removed from the buffer $B^1_{2,\bar{3}\bar{4}}$ and stored in buffers $B^1_{2,\bar{3}4}$ and $B^1_{4,\bar{3}}$; hence, the chain moves to state 3.
  - If upon transmission by node 2 the packet is erased at node 3 and 4, an event with probability ϵ_{34}^2, the chain remains at state 2.

Proceeding in a similar fashion we evaluate all the transition probabilities of the Markov Chain.

Let π_k be the steady-state probability that the Markov Chain represented in Figure 3 is in state $k \in \{1, 2, 3, 4, 5\}$. Let V^3

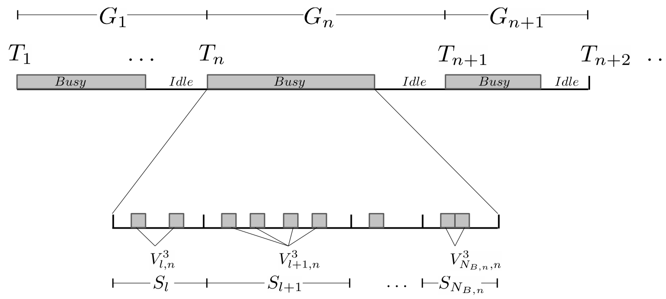


Fig. 4. Busy and idle periods of the Q_1^S queue.

be the number of visits to state 3 between two successive visits to state 1. It is known [21, page 161] that the following equalities hold.

$$\mu_1^{\mathbf{IV}} = \pi_1, \tag{14}$$

$$\mathbb{E}\left[V^3\right] = \frac{\pi_3}{\pi_1}. \tag{15}$$

We now concentrate on queue $Q_{2,3\bar{4}}$. This queue is of the generic type discussed at the end of Section II-A. Specifically, we identify T_n with the beginning of the n-th busy period of queue Q_1^S, as depicted in Figure 4. Packets arrive to queue $Q_{2,3\bar{4}}$ during the idle periods of Q_1^S when node 2 transmits a packet of session (2,4) that is erased at node 4 and received by node 3 (an event with probability $\epsilon_4^2 - \epsilon_{34}^2$). We denote the number of these packets that arrive to queue $Q_{2,3\bar{4}}$ during the n-th idle period of queue Q_1^S, with A_n^2 (it should be noted that the superscript refers to the session to which the packets belong, while the subscript refers to the busy period of the Q_1^S queue). Hence,

$$\mathbb{E}\left[A_n^2\right] = \mathbb{E}\left[A_1^2\right] = \bar{I}_1\left(\epsilon_4^2 - \epsilon_{34}^2\right). \tag{16}$$

Opportunities to transmit packets from $Q_{2,3\bar{4}}$ arise whenever buffer $B^1_{2,\bar{3}4}$ is nonempty, i.e., the Markov chain in Figure 3 is in state 3. Let $V^3_{l,n}$ be the number of times state 3 is visited during the service time, S_l, of the lth packet of session (1,3) in the n-th busy period of Q_1^S. The random variables $V^3_{l,n}$, $l = 1, 2, ..$ are i.i.d. and from the definition of the Markov Chain in Figure 3 it follows that their mean is

$$\mathbb{E}\left[V^3_{l,n}\right] = \mathbb{E}\left[V^3_{l,1}\right] = \frac{\pi_3}{\pi_1}. \tag{17}$$

Let $N_{B,n}$ be the number of session (1,3) packets served during the n-th busy period of Q_1^S. It is known [21] that

$$\bar{B}_1 = \bar{S}_1^{\mathbf{IV}}\mathbb{E}\left[N_{B,n}\right] = \frac{\mathbb{E}\left[N_{B,1}\right]}{\pi_1}, \tag{18}$$

where the last equality follows from (12) and (14).

The number of slots available for transmission of session (2,4) packets during the n-th busy period of Q_1^S is $H_n^2 = \sum_{l=1}^{N_{B,n}} V^3_{l,n}$. Using the fact that $N_{B,n}$ is a stopping time we obtain from Wald's equality [21], (17) and (18),

$$\mathbb{E}\left[H_n^2\right] = \mathbb{E}\left[V^3_{1,2}\right]\mathbb{E}\left[N_{B,2}\right] = \mathbb{E}\left[V^3_{l,1}\right]\mathbb{E}\left[N_{B,1}\right] = \pi_3\bar{B}_1. \tag{19}$$

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

The service time of the l-th packet of session (2,4) transmitted whenever buffer $B^1_{2,\bar{3}4}$ is nonempty and denoted by S^2_l, is geometrically distributed with parameter $1-\epsilon^2_4$, hence

$$\mathbb{E}\left[S^2_l\right] = \frac{1}{1-\epsilon^2_4}. \qquad (20)$$

Also, since $T_{n+1}-T_n$ is the sum of the lengths of the nth busy and nth idle period of queue Q^S_1, we have from (2), (3) and (14)

$$\begin{aligned}\mathbb{E}\left[G_1\right] &= \bar{B}_1+\bar{I}_1\\ &= \frac{\lambda_1/\pi_1}{\left(1-\lambda_1/\pi_1\right)\left(1-q_0\right)}+\frac{1}{\left(1-q_0\right)}\\ &= \frac{1}{\left(1-\lambda_1/\pi_1\right)\left(1-q_0\right)}. \qquad (21)\end{aligned}$$

Using (16), (19), (20) and (21) above in formulas (5)-(6) for the generic queue, after some algebra we obtain that the throughput of packets in queue $Q_{2,3\bar{4}}$, r_{busy}, is equal to

$$r_{busy} = \begin{cases}\left(\epsilon^2_4-\epsilon^2_{34}\right)\left(1-\frac{\lambda_1}{\pi_1}\right), & \text{if } \epsilon^2_4-\epsilon^2_{34} \le \frac{\frac{\pi_3\lambda_1}{\pi_1}\left(1-\epsilon^2_4\right)}{1-\frac{\lambda_1}{\pi_1}}\\ \frac{\pi_3}{\pi_1}\left(1-\epsilon^2_4\right)\lambda_1 & \text{if } \epsilon^2_4-\epsilon^2_{34} > \frac{\frac{\pi_3\lambda_1}{\pi_1}\left(1-\epsilon^2_4\right)}{1-\frac{\lambda_1}{\pi_1}}\end{cases} \qquad (22)$$

where the steady state probabilities $\pi_1,\ \pi_3$ can be calculated using the transition probabilities of the Markov Chain in Figure 3. In fact, from (13), (14) immediately have,

$$\pi_1 = \frac{\left(1-\epsilon^2_3\right)\left(1-\epsilon^1_{23}\right)}{1-\epsilon^2_3+\epsilon^1_3-\epsilon^1_{23}}, \qquad (23)$$

while calculation using the transition probabilities shows that

$$\pi_3 = 1-\frac{\left(\frac{1-\epsilon^1_{234}}{1-\epsilon^1_{23}}+\frac{\epsilon^1_{34}-\epsilon^1_{234}}{1-\epsilon^2_{34}}\right)}{\left(1-\epsilon^1_{234}\right)}\frac{\left(1-\epsilon^1_{23}\right)\left(1-\epsilon^2_3\right)}{\left(1+\epsilon^1_3-\epsilon^2_3-\epsilon^1_{23}\right)}. \qquad (24)$$

The throughput of session (2,4) packets transmitted during an idle period of queue Q^S_1, r_{idle}, is easily calculated as

$$r_{idle} = \frac{\bar{I}_1\left(1-\epsilon^2_4\right)}{\bar{B}_1+\bar{I}_1} = \left(1-\lambda_1/\pi_1\right)\left(1-\epsilon^2_4\right). \qquad (25)$$

Since the throughput of session (2,4) packets is $r_2 = r_{idle}+r_{busy}$, we conclude from (22), (23), (24) and (25) the following proposition.

Proposition 2. The throughput region of Algorithm IV, $\mathcal{R}_{\mathbf{IV}}$, is the set of throughput pairs (r_1, r_2) satisfying the following inequalities

$$\frac{1-\epsilon^2_3+\epsilon^1_3-\epsilon^1_{23}}{\left(1-\epsilon^2_3\right)\left(1-\epsilon^1_{23}\right)}r_1+\frac{r_2}{1-\epsilon^2_{34}} \le 1, \qquad (26)$$

$$\left(\frac{\epsilon^1_{34}-\epsilon^1_{234}}{\left(1-\epsilon^2_{34}\right)\left(1-\epsilon^1_{234}\right)}+\frac{1}{1-\epsilon^1_{23}}\right)r_1+\frac{r_2}{1-\epsilon^2_4} \le 1, \qquad (27)$$

$$r_i \ge 0,\ i\in\{1,2\}.$$

In Figure 5 we show the general form of the throughput regions of Algorithms I, III and IV. We see that when node 2 performs network coding, for the same arrival rate of session (1,3) packets, the throughput of Secondary session (2, 4) is increasing, adding in affect the area ABC to the throughput region of the system. We note that this is achieved without adding any additional complexity to the Primary transmitter. The Primary receiver has the additional complexity of storing received session (2,4) packets and performing simple decoding of network-coded packets; this seems an acceptable trade-off for the primary session (1,3), since as is seen in Figure 5, cooperation with the secondary session increases significantly the stability region of session (1,3) (from OD to OC).

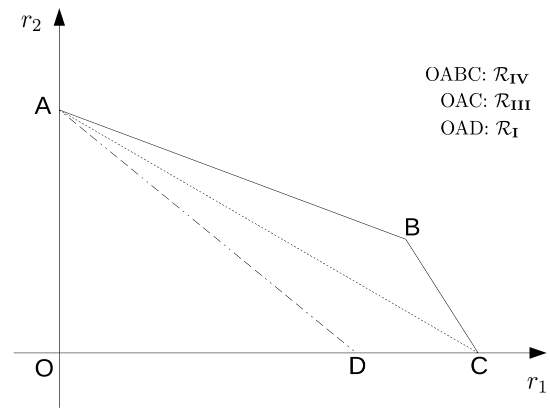


Fig. 5. The throughput regions of Algorithms I, III and IV.

V. An Algorithm with Increased Throughput Region

In this section we examine whether the throughput region of the system can be increased further by employing more sophisticated operations. The rationale is the following.

Consider the case where Primary transmitter (node 1) sends a session (1,3) packet p_1, and assume that this packet is received only by Secondary transmitter (node 2). According to Algorithms III and IV, node 2 will then act as relay for packet p_1. Note that for a given r_1, the increase in r_2 induced by Algorithm IV as compared to r_2 induced by Algorithm III, occurs because, during the attempt by node 2 to send packet p_1, it happens that this packet has already been received by node 4; so the possibility of network coding operation arises. However, if $\epsilon^1_{34} < \epsilon^2_{34}$, then it is more likely that packet p_1 is received by either node 3 or node 4 if it is re-transmitted by node 1. On the other hand, if the rate of session (1,3) packets, $r_1 = \lambda_1$, is close to point C in Figure 5 this re-transmission should be avoided since queue Q_1 will become unstable. Therefore, it seems that, in order to effect increase in the throughput of session (2,4) while maintaining admissibility of the algorithm, a compromise between the following two cases must be made: a) node 2 acts immediately as a relay of packet p_1 and b) node 1 keeps re-transmitting p_1 until received by either node 3 or node 4.

To effect this compromise, we modify Algorithm IV as follows. We introduce a parameter q, $0 \le q \le 1$. When node 1 transmits a packet p_1 that is seen only by node 2 (hence now the packet is stored in buffer $B^1_{2,\bar{3}\bar{4}}$) then p_1 remains in Q_1 and is transmitted by node 1 with probability q and by node 2 with probability $1-q$. In both cases, if p_1 is received by node 3, then it is removed from Q_1 and $B^1_{2,\bar{3}\bar{4}}$; if on the other hand it is erased at node 3 but received by node 4, then

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

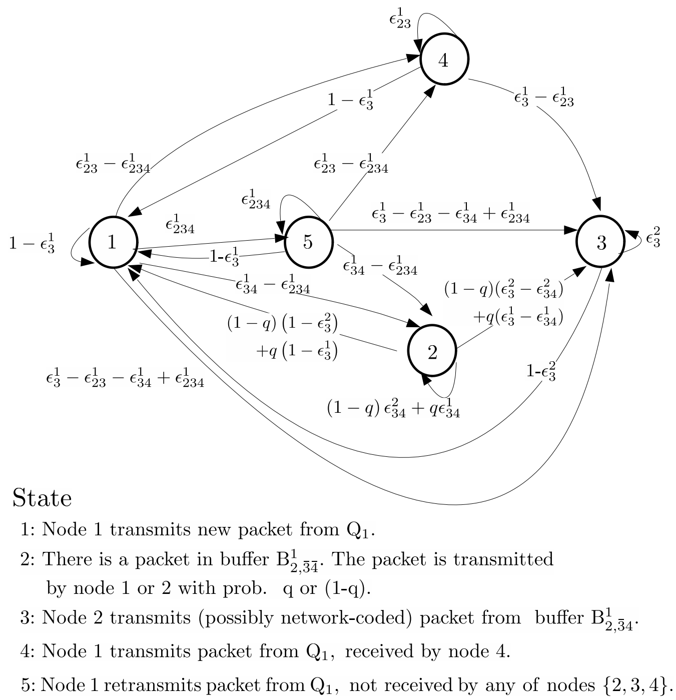


Fig. 6. The Markov chain describing service times of session (1,3) packets induced by Algorithm V.

the packet is removed from Q_1 and node 2 acts as relay for p_1 as in Algorithm IV. This algorithm is referred to in what follows as Algorithm V.

The service times of session (1,3) packets under Algorithm V may increase as compared to the service times of the packets under Algorithm IV, but it can be shown by employing arguments similar to those used in the proof of Proposition 1 that they remain stochastically smaller than the service times of Algorithm I that involved no cooperation.

For given q, the performance analysis of Algorithm V is similar to the performance analysis of Algorithm IV. The main difference is that the Markov Chain describing the service times of session (1,3) packets under Algorithm V is modified according to the Figure 6. Let $\pi_i(q), 1 \le i \le 5$, be the steady state probabilities of this Markov chain when parameter q is used. From algebraic calculations based on the transition probabilities of the Markov chain, it can be seen that,

$$\frac{1}{\pi_1(q)} = \frac{1+\epsilon^1_3-\epsilon^2_3-\epsilon^1_{23}}{(1-\epsilon^2_3)\,(1-\epsilon^1_{23})} + C_1\theta(q), \tag{28}$$

$$\frac{1-\pi_3(q)}{\pi_1(q)} = \frac{\epsilon^1_{34}-\epsilon^1_{234}}{(1-\epsilon^2_{34})\,(1-\epsilon^1_{234})} + \frac{1}{1-\epsilon^1_{23}} - C_2\theta(q), \tag{29}$$

where $C_1 = \frac{\epsilon^1_3-\epsilon^2_3}{1-\epsilon^2_3}\frac{\epsilon^1_{34}-\epsilon^1_{234}}{1-\epsilon^1_{234}}$, $C_2 = \frac{\epsilon^2_{34}-\epsilon^1_{34}}{1-\epsilon^2_{34}}\frac{\epsilon^1_{34}-\epsilon^1_{234}}{1-\epsilon^1_{234}}$ and

$$\theta\,(q) = \frac{q}{1-q\epsilon^1_{34}-(1-q)\,\epsilon^2_{34}}. \tag{30}$$

We can now state the following proposition which is analogous to Proposition 2.

Proposition 3. The throughput region of Algorithm V, $\mathcal{R}_{\mathbf{V}}$, is the set of pairs (r_1, r_2) satisfying the following inequalities

$$\left(\frac{1+\epsilon^1_3-\epsilon^2_3-\epsilon^1_{23}}{(1-\epsilon^2_3)\,(1-\epsilon^1_{23})} + C_1\theta(q)\right) r_1 + \frac{r_2}{1-\epsilon^2_{34}} \le 1, \tag{31}$$

$$\left(\frac{\epsilon^1_{34}-\epsilon^1_{234}}{(1-\epsilon^2_{34})\,(1-\epsilon^1_{234})} + \frac{1}{1-\epsilon^1_{23}} - C_2\theta(q)\right) r_1 + \frac{r_2}{1-\epsilon^2_4} \le 1, \tag{32}$$

$$0 \le q \le 1,\ r_i \ge 0,\ i = 1, 2.$$

The next proposition gives a non-parametric description of the throughput region of Algorithm V and provides a method for computing in closed form the appropriate parameter q for given erasure probabilities and rates $r_i,\ i = 1, 2$.

Proposition 4. The following hold.

1) If $\epsilon^1_{34} \ge \epsilon^2_{34}$ then $\mathcal{R}_{\mathbf{V}} = \mathcal{R}_{\mathbf{IV}}$.
2) If $\epsilon^1_{34} < \epsilon^2_{34}$ then $\mathcal{R}_{\mathbf{V}} = \mathcal{R}_{\mathbf{IV}} \cup \mathcal{R}_0$ where $\mathcal{R}_0$ is the set of rate pairs (r_1, r_2) satisfying the following inequalities.

$$\left(\frac{1+\epsilon^1_3-\epsilon^2_3-\epsilon^1_{23}}{(1-\epsilon^2_3)\,(1-\epsilon^1_{23})} + C_1\theta\,(q^*)\right) r_1 + \frac{r_2}{1-\epsilon^2_{34}} \le 1, \tag{33}$$

$$\left(\frac{\epsilon^1_{34}-\epsilon^1_{234}}{(1-\epsilon^2_{34})\,(1-\epsilon^1_{234})} + \frac{1}{1-\epsilon^1_{23}}\right) r_1 + \frac{r_2}{1-\epsilon^2_4} > 1, \tag{34}$$

$$\left(\frac{\epsilon^1_{34}-\epsilon^1_{234}}{(1-\epsilon^2_{34})\,(1-\epsilon^1_{234})} + \frac{1}{1-\epsilon^1_{23}} - \frac{C_2}{1-\epsilon^1_{34}}\right) r_1 + \frac{r_2}{1-\epsilon^2_4} \le 1, \tag{35}$$

$$r_i \ge 0,$$

where

$$r_1\theta\,(q^*) = \frac{\left(\frac{\epsilon^1_{34}-\epsilon^1_{234}}{(1-\epsilon^2_{34})(1-\epsilon^1_{234})} + \frac{1}{1-\epsilon^1_{23}}\right) r_1 + \frac{r_2}{1-\epsilon^2_4} - 1}{C_2}. \tag{36}$$

Proof: The proof can be found in Appendix C. ∎

According to Proposition 4, when $\epsilon^1_{34} \ge \epsilon^2_{34}$ the throughput regions of Algorithms IV and V coincide, hence there is no benefit in employing Algorithm V. When $\epsilon^1_{34} < \epsilon^2_{34}$, the throughput region of Algorithm V is strictly larger than the throughput region of Algorithm IV. In this case, it follows from Proposition 4 that $\mathcal{R}_{\mathbf{V}}$ is the region of non-negative $r_i,\ i = 1, 2$ that satisfy inequalities (26), (35) and (33) where $r_1\theta(q^*)$ is replaced by the right hand side of equality (36). Based on these inequalities, the parameter q^* needed by Algorithm V to operate, can be determined as follows:

1) For a given $r_1 > 0$, we find r_{2a} , r_{2b}, r_{2c} as solutions of (26), (35) and (33) respectively, where inequalities are replaced with equalities.
2) The throughput of session (2,4) packets, r_2 is then, $r_2 = \min\{r_a, r_b, r_c\}$.
3) Parameter $\theta(q^*)$ is determined as solution to equation (36).
4) Parameter q^* is determined using (30).

In Figure 7 we show the general form of the throughput regions of Algorithms IV and V. The throughput regions of Algorithms IV and V are the areas OABC and OADBC respectively. Furhtermore, the region $\mathcal{R}_0$ is the triangle ADB. The

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

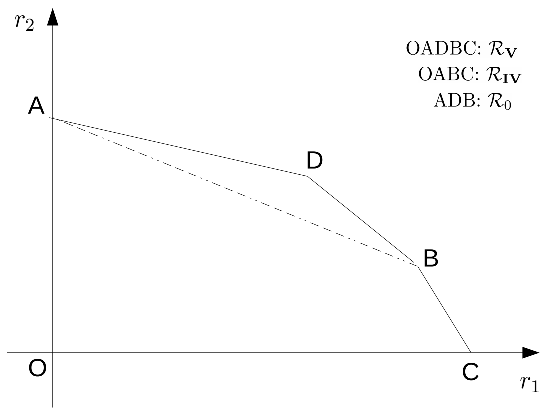


Fig. 7. The General Form of throughput Regions of Algorithms IV and V, when $\epsilon^1_{34} < \epsilon^2_{34}$.

lines AD, DB, BC represent (26), (35) and (33) respectively, where inequalities are replaced with equalities. Points on line AD are achieved using $q^* = 1$, points on DB (not including points D, B) using $0 < q^* < 1$, and points on BC using $q^* = 0$.

Algorithm V has strictly larger throughput region than Algorithm IV when $\epsilon^1_{34} < \epsilon^2_{34}$. However, as discussed above, Algorithm V needs parameter q^* to operate, which depends crucially on erasure probabilities and the arrival rate of session (1,3) packets. Hence in practice Algorithm IV may be preferable. However, Algorithm V has theoretical interest since it shows intricacies that arise even in simple cooperative systems.

VI. Numerical and Simulation Results

In this section we present numerical and Monte-Carlo simulation results that illustrate the performance of the proposed algorithms and validate the presented analysis. Specifically, assuming that $\epsilon^1_3 = 0.8$, $\epsilon^1_2 = \epsilon^2_3 = \epsilon^2_4 = 0.2$, $\epsilon^1_4 = 0.3$ and that all erasure events are independent, we investigate the proposed transmission algorithms in terms of throughput and average packet delays for both primary and secondary users' packets.

In Figure 8 we plot the throughput regions of algorithms I, III, IV and V, i.e., $\mathcal{R}_{\mathbf{I}}$, $\mathcal{R}_{\mathbf{III}}$, $\mathcal{R}_{\mathbf{IV}}$ and $\mathcal{R}_{\mathbf{V}}$, respectively. It is seen that the throughput regions $\mathcal{R}_{\mathbf{IV}}$ and $\mathcal{R}_{\mathbf{V}}$ coincide; this was expected, since for the selected erasure probability values it holds $\epsilon^1_{34} > \epsilon^2_{34}$ and according to Proposition 4, $\mathcal{R}_{\mathbf{IV}} = \mathcal{R}_{\mathbf{V}}$ in this case. Furthermore, the comparison between the algorithms illustrates that algorithms IV and V clearly provide the largest throughput region; the extra region ABC is added to the throughput region achieved by algorithm III. This is done by performing network coding actions at the Secondary transmitter, without adding any extra complexity to the Primary transmitter. In this example, the merits of cooperation for the Primary user are clearly illustrated; Algorithms III, IV and V improve the maximum throughput rate r_1 of Primary user from 0.2 to 0.45.

The performance of the presented transmission protocols in terms of average packet delays for Primary and Secondary users is shown in Figures 9 and 10. In the simulation setup, we consider that both queues Q_1 and Q_2 are initially empty,

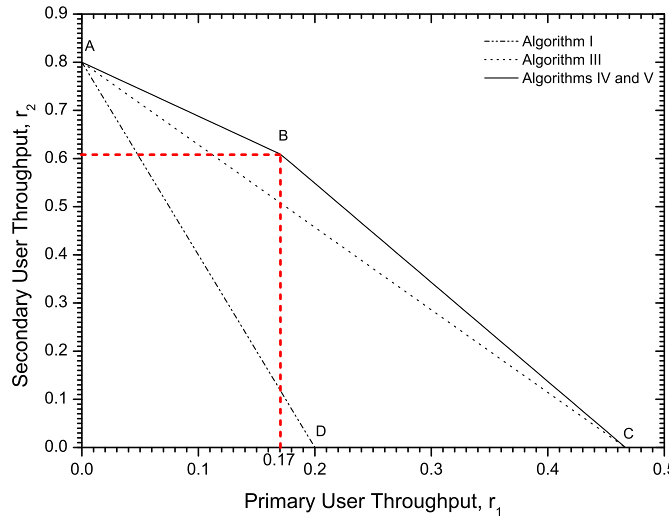


Fig. 8. Comparison of throughput regions of Algorithms I, III, IV and V.

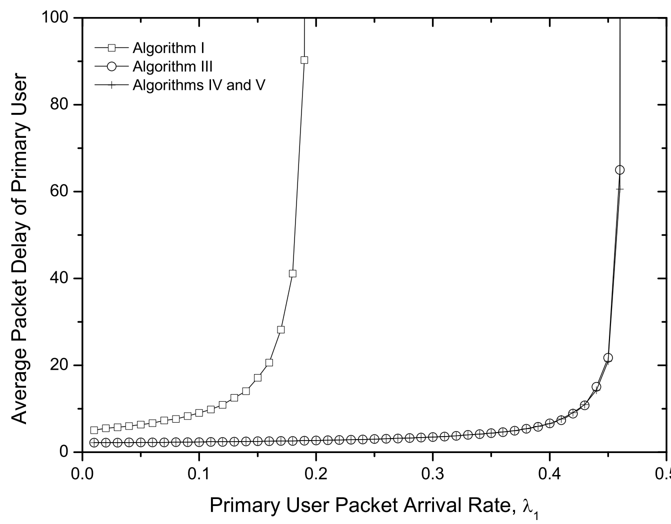


Fig. 9. Average packet delay of Primary user.

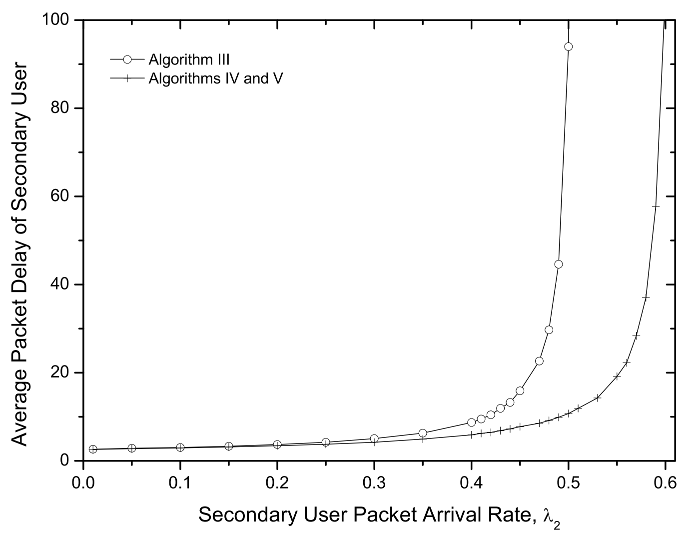


Fig. 10. Average packet delay of Secondary user, when $r_1 = 0.17$.

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

and assume that the arrivals in slot t are Bernoulli random variables with rates λ_1 and λ_2 respectively. Figure 9 illustrates the packet delays of Primary user when Algorithms I, III, IV and V are employed, measured in time slots. It is seen from the Figure 9 that cooperation significantly improves the average packet delay of Primary user. Moreover, when comparing Algorithm III with IV and V, it is shown that the performance of these transmission protocols is identical. However, the same does not happen when comparing the performance of these algorithms in terms of Secondary user's average packet delay; this is shown in Figure 10 where $\lambda_1 = 0.17$ is assumed. In this case, significant improvements in Secondary's packet delays by Algorithms IV and V are observed, especially for the values of λ_2 larger than 0.35. Furthermore, it can be observed from this figure that Q_2 is stable up to $\lambda_2 \approx 0.6$ (the average packet delay for $\lambda_2 = 0.6$ is 106 slots), which basically coincides with the value of r_2, when $r_1 = 0.17$ as shown by point B in Figure 8-numerically for $\lambda_1 = 0.17$ the maximum value of r_2 is 0.60952. As expected, the same behaviour is observed for all values of λ_1 that we tested.

Finally, the performance of Algorithm V was numerically investigated. Specifically, after an exhaustive numerical search over all possible values of erasure probabilities, ranging from 0.1 to 0.9, and assuming independent erasure events, it was found that the maximum relative difference between the throughput regions achieved by Algorithms V and IV (point D of Figure 7) was equal to 18.05% (when $\epsilon_3^1 = 0.75$, $\epsilon_2^1 = 0.1$, $\epsilon_3^2 = 0.7$, $\epsilon_4^2 = 0.9$, $\epsilon_4^1 = 0.6$). In addition, it was observed that a throughput region increase larger than 10% was observed for 0.3% of the different sets of erasure probabilities examined. Next we examined the case where erasure events for transmissions from the Primary and the Secondary transmitters to the two receivers are correlated. In this case, the maximum relative difference was found to be equal to 36,1% (when $\epsilon_3^1 = 0.75$, $\epsilon_2^1 = 0.1$, $\epsilon_3^2 = 0.7$, $\epsilon_{23}^1 = 0.075$, $\epsilon_{34}^1 = 0.55$, $\epsilon_{234}^1 = 0.055$, $\epsilon_4^2 = 0.9$, $\epsilon_{34}^2 = 0.7$, $\epsilon_4^1 = 0.55$), while throughput increase larger than 10% was observed for 9% of the different sets of erasure probabilities examined. These results imply that Algorithm IV can be considered as a reliable and more practical alternative to Algorithm V, since it does not require any knowledge of both the erasure probabilities of the channel and the arrival rate of packets at the Primary transmitter. However, as already mentioned, Algorithm V has important theoretical interest since its performance shows that for certain values of erasure probabilities there is room for further improvement in the system's throughput when more sophisticated operations take place.

VII. Conclusions

We proposed two algorithms for Primary-Secondary user cooperation in Cognitive Networks using network coding techniques. We analyzed the performance of the proposed algorithms. The results show that when compared to the case where the Secondary transmitter acts as a relay without performing network coding, significant improvement of the throughput of the secondary channel may occur. The first algorithm imposes no extra implementation requirements to the Primary transmitter apart from listening to the feedback sent by the Secondary transmitter. The Primary receiver has the additional requirement that it stores received packets intended for the Secondary receiver and it performs decoding of network-coded packets. In return though, the stability region and the service times of all Primary packets are significantly improved. A notable feature of this algorithm is that no knowledge of packet arrival rates to Primary transmitter and channel statistics is required. We next examined the possibility of increasing the throughput region of the system by more sophisticated techniques. We presented a second algorithm which is a generalization of the first and showed that this increase is possible in certain cases. However, in this case, knowledge of channel erasure probabilities, as well as the arrival rate of Primary transmitter packets are crucial for the algorithm to operate correctly.

An important issue that arises is to examine whether the throughput of the Secondary channel can be increased further by more sophisticated operations. In our recent work [18], we showed that the throughput region of the suggested algorithms coincides with the capacity region of the system for a large range of system parameters, and most of it for the rest of system parameters.

Appendix A
Proof of Proposition 1

The proof of stochastic dominance can be done by explicitly calculating the relevant probabilities and then showing the required inequality. We can avoid cumbersome calculations, however, by resorting to a technique commonly used in this type of proofs. Specifically, let S_l^{nc}, S_l^c, $l = 1, 2, ...$ be the service times of packet l transmitted by node 1 under Algorithms I and III respectively (these service times are i.i.d. under both algorithms). To show stochastic dominance, we construct on the same probability space two random variables $\hat{S}^{nc}$ and $\hat{S}^c$ with the following properties: 1) It holds $\hat{S}^c \leq \hat{S}^{nc}$, 2) $\hat{S}^c$ and S_l^c have the same distribution, 3) $\hat{S}^{nc}$ and S_l^{nc} have the same distribution. The fact that S_k^c is stochastically smaller than S_k^{nc} follows then immediately from the inequality $\hat{S}^c \leq \hat{S}^{nc}$.

We now proceed with the construction of $\hat{S}^{nc}$ and $\hat{S}^c$. Consider on the same probability space a sequence $\left(\hat{Z}_2^1(t), \hat{Z}_3^1(t)\right)$, $t = 0, 1, ...$ of i.i.d pairs of random variables (for given t the pair $\hat{Z}_2^1(t), \hat{Z}_3^1(t)$ may be dependent), and a sequence $\theta(t)$, $t = 0, 1, ...$ of i.i.d. random variables, independent of $\left(\hat{Z}_2^1(t), \hat{Z}_3^1(t)\right)$, $t = 0, 1, ...$ All random variables take values either 0 or 1, with probabilities,

$$\Pr\left(\hat{Z}_3^1(t) = 0\right) = \epsilon_3^1, \ \Pr\left(\hat{Z}_2^1(t) = \hat{Z}_3^1(t) = 0\right) = \epsilon_{23}^1,$$
$$\Pr\left(\theta(t) = 0\right) = \frac{\epsilon_3^2}{\epsilon_3^1}, \ t = 1, 2, ...$$

Note that $\Pr\left(\theta(t) = 0\right)$ is indeed a probability because of (8). Let also

$$\hat{J}(t) = \begin{cases} \theta(t) & \text{if} \quad \hat{Z}_3^1(t) = 0 \\ \hat{Z}_3^1(t) & \text{if} \quad \hat{Z}_3^1(t) = 1. \end{cases} \tag{37}$$

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2980821, IEEE Transactions on Wireless Communications
This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TWC.2020.2980821

From (37) we see that $\hat{J}(t)$ takes values 0 or 1 and $\hat{J}(t) \geq \hat{Z}_3^1(t),\ t = 0, 1, 2....$ Moreover, $\hat{J}(t),\ t = 1., 2, ...$ are i.i.d, $\hat{J}(t)$ is independent of $\hat{Z}_3^1(\tau),\ \hat{Z}_3^2(\tau),\ \tau \neq t$, and

$$\begin{aligned} \Pr\left(\hat{J}(t) = 0\right) &= \Pr\left(\theta(t) = 0,\ \hat{Z}_3^1(t) = 0\right) \\ &= \Pr\left(\theta(t) = 0\right)\epsilon_3^1 \\ &= \epsilon_3^2. \end{aligned} \tag{38}$$

That is, the random variables $\hat{J}(t),\ t = 0, 1, ...$ are identically distributed to the random variables $Z_3^2(t),\ t = 0, 1, ...$ denoting erasure events defined in Section II.

Let $\hat{T}_{23}$ be the stopping time denoting the first time at least one of the random variables $\hat{Z}_2^1(t),\ \hat{Z}_3^1(t)$ takes the value 1, i.e.,

$$\hat{T}_{23} = \min_{t \geq 0}\{\hat{Z}_2^1(\tau) = \hat{Z}_3^1(\tau) = 0,\ \tau = 0, .., t-1, \\ \hat{Z}_2^1(t) + \hat{Z}_3^1(t) > 0\}. \tag{39}$$

Let $\hat{S}^{nc} \geq \hat{T}_{23}$ be the first time that the random variable $\hat{Z}_3^1(t)$ takes the value 1, and define $\hat{S}^c$ as follows. If at time $\hat{T}_{23}$ it holds $\hat{Z}_3^1(\hat{T}_{23}) = 1$ then $\hat{S}^c = \hat{T}_{23}$. Else $\hat{S}^c$ is the first time, $\hat{T}$, after $\hat{T}_{23}$ that the random variable $\hat{J}(t)$ becomes 1. Therefore it holds,

$$\hat{S}^c = \hat{T}_{23} + \mathbf{1}_{\{\hat{Z}_3^1(\hat{T}_{23})=0\}}(\hat{T} - \hat{T}_{23}), \tag{40}$$

where $\mathbf{1}_{\mathcal{A}}$ is the indicator function of event $\mathcal{A}$. Notice that the interval $\hat{T} - \hat{T}_{23}$ depends only on $\hat{J}(\hat{T}_{23} + t),\ t \geq 1$ and these variables are independent of $\hat{T}_{23}$, since $\hat{T}_{23}$ is a stopping time for the sequence $\left(\hat{Z}_3^1(t), \hat{Z}_3^2(t)\right)$. Note also that we can write

$$\hat{S}^{nc} = \hat{T}_{23} + \mathbf{1}_{\{\hat{Z}_3^1(\hat{T}_{23})=0\}}(\hat{S}^{nc} - \hat{T}_{23}), \tag{41}$$

where interval $\hat{S}^{nc} - \hat{T}_{23}$ depends only on $\hat{Z}_3^1(\hat{T}_{23} + t),\ t \geq 1$ From (40), (41) and since $\hat{J}(t) \geq \hat{Z}_3^1(t),\ m = 1, 2....$, it follows that, $\hat{S}^c \leq \hat{S}^{nc}$.

We now examine the service times of Algorithms I and III. For simplicity we omit the packet index l from S_l^{nc} and S_l^c. According to Algorithm 1, S^{nc} is the first time $Z_3^1(t)$ takes the value one, where $Z_j^i(t)$ are the random variables expressing erasure events defined in Section II. Since the random variables $Z_3^1(t)$ and $\hat{Z}_3^1(t),\ t = 1, 2, ...$ are identically distributed, it follows that S^{nc} and $\hat{S}^{nc}$ have the same distribution.

Let T_{23} be the first time at least one of the random variables $Z_2^1(t),\ Z_3^1(t)$ takes the value 1. From the operation of Algorithm 3 it follows that if $Z_3^1(T_{23}) = 1$ (the transmitted packet is received by node 3 at time T_{23}) then $S^c = T_{23}$. Else (the transmitted packet is received by node 2 and erased at node 3) S^c is the first time, T, after T_{23} that the random variable $Z_3^2(t)$ becomes 1. From the definitions it holds,

$$S^c = T_{23} + \mathbf{1}_{\{Z_3^1(T_{23})=0\}}(T - T_{23}), \tag{42}$$

where interval $S^{nc} - T_{23}$ depends only on the variables $Z_3^2(T_{23} + t),\ t \geq 1$.

According to (38), and the definitions, the set of random variables $\hat{T}_{23}, ,\ \hat{J}(\hat{T}_{23} + t),\ t \geq 1$ are identically distributed to the set $T_{23},\ Z_3^2(T_{23} + t),\ t \geq 1$. It then follows from (40) and (42) that S^{nc} and $\hat{S}^{nc}$ have the same distribution.

Appendix B
Random Packet Arrivals at the Secondary Transmitter

To simplify the analysis in this paper, we assumed that queue Q_2 at the secondary transmitter contains an infinite number of packets. Assume now that packets arrive at Q_2 according to an i.i.d. random process with rate λ_2 . In this case we can still apply the algorithms IV and V presented in this paper with the following addition.

Added instruction to algorithms IV and V

> If at some time t, Q_1^S and Q_2 are both empty, then a packet from $Q_{2,3\bar{4}}$ (if the queue is nonempty) is transmitted uncoded.

We are interested in determining the stability region of the system, i.e., the closure of the set of packet arrival rates (λ_1, λ_2) for which queues $(Q_1^S, Q_2, Q_{2,3\bar{4}})$ are stable under a given algorithm. It turns out that with the above mentioned modification, the stability region of an algorithm is the same as its corresponding throughput region. A detailed proof of this fact is based on regenerative theory [21]. To avoid lengthy but mostly standard arguments, we provide an outline of the proof below for the case of Algorithm IV.

According to Proposition 2, and using (23) and (24), we need to show that the queues $(Q_1^S, Q_2, Q_{2,3\bar{4}})$ are stable if and only if (λ_1, λ_2) where $\lambda_i \geq 0$ for $i \in \{1, 2\}$, satisfy the inequalities

$$\frac{1}{\pi_1}\lambda_1 + \frac{1}{1 - \epsilon_{34}^2}\lambda_2 < 1, \tag{43}$$

$$\frac{1 - \pi_3}{\pi_1}\lambda_1 + \frac{1}{1 - \epsilon_4^2}\lambda_2 < 1. \tag{44}$$

Assume that (λ_1, λ_2) satisfy the inequalities above. Inequality (43) implies that necessarily, $\lambda_1 < \pi_1 = \mu_1^{\mathbf{IV}}$, i.e., the arrival rate at Q_1^S is less than the departure rate, hence Q_1^S is stable.

Consider next Q_2. A packet is transmitted from (nonempty) Q_2 when Q_1^S is empty, and until at least one of nodes 3, 4 receives the packet, an event with probability $1 - \epsilon_{34}^2$. Since Q_1^S is stable, the probability that this queue is empty is equal to $1 - \lambda_1/\mu_1^{\mathbf{IV}} = 1 - \lambda_1/\pi_1$. Hence the packet service rate of Q_2 is $\left(1 - \epsilon_{34}^2\right)\left(1 - \lambda_1/\pi_1\right)$. By (43), we have $\lambda_2 < \left(1 - \epsilon_{34}^2\right)\left(1 - \frac{\lambda_1}{\pi_1}\right)$, hence Q_2 is stable and we conclude that for this queue the departure rate is equal to the arrival rate, i.e., $r_2 = \lambda_2$.

Consider now $Q_{2,3\bar{4}}$. Packets transmitted by Q_2 arrive at $Q_{2,3\bar{4}}$, when they are received by node 3 and erased at node 4, an event with probability $(\epsilon_4^2 - \epsilon_{34}^2)/(1 - \epsilon_{34}^2)$ - see Figure 11. Hence, the arrival rate at $Q_{2,3\bar{4}}$ is,

$$\lambda_{2,3\bar{4}} = \frac{\epsilon_4^2 - \epsilon_{34}^2}{1 - \epsilon_{34}^2}r_2 = \frac{\epsilon_4^2 - \epsilon_{34}^2}{1 - \epsilon_{34}^2}\lambda_2. \tag{45}$$

Packets are served from $Q_{2,3\bar{4}}$ either during the times when Q_1^S is nonempty (in which case the packets are sent encoded) or during times when Q_1^S and Q_2 are both empty (in which case, according to the instruction added above, the packets are sent uncoded). We evaluate the service rates of the packets in these two cases.

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

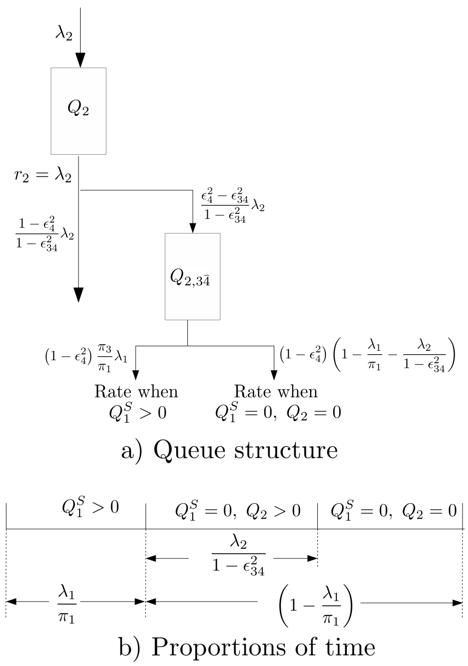


Fig. 11. Arrival and departure rates of Q_2 and $Q_{2,3\bar{4}}$.

Case 1: $Q_1^S > 0$. When Q_1^S is nonempty the average number of transmission opportunities from $Q_{2,3\bar{4}}$ during the transmission time of session (1,3) packets is π_3/π_1 (see (15)); since a transmission from $Q_{2,3\bar{4}}$ is successful with probability $1-\epsilon_4^2$, the average number of *successful* transmissions from $Q_{2,3\bar{4}}$ per transmission of a session (1,3) packet is $(1-\epsilon_4)\pi_3/\pi_1$. Taking into account that Q_1^S is stable, we conclude that the proportion of time packets from $Q_{2,3\bar{4}}$ can be transmitted successfully (service rate of packets) is,

$$(1-\epsilon_4)\,\frac{\pi_3}{\pi_1}\frac{\mathbb{E}[N_{B,1}]}{\bar{B}_1+\bar{I}_1} = (1-\epsilon_4)\,\frac{\pi_3}{\pi_1}\lambda_1. \tag{46}$$

where the last equality follows from (2), (3) and (18).

Case 2: $Q_1^S = 0$ and $Q_2 = 0$. The proportion of time that (the stable) Q_1^S is empty is $1-\frac{\lambda_1}{\pi_1}$. Since Q_2 is stable and its transmission rate is $1-\epsilon_{34}^2$, the proportion of time during which packets from Q_2 are transmitted is $\frac{\lambda_2}{1-\epsilon_{34}^2}$. Since these packets are transmitted when Q_1^S is empty, the proportion of time left for transmission of packets from $Q_{2,3\bar{4}}$ during times that Q_1^S is empty, is $1-\frac{\lambda_1}{\pi_1}-\frac{\lambda_2}{1-\epsilon_{34}^2}$ (see Figure11) and the rate of successful reception of $Q_{2,3\bar{4}}$ packets by node 4 is

$$\left(1-\epsilon_4^2\right)\left(1-\frac{\lambda_1}{\pi_1}-\frac{\lambda_2}{1-\epsilon_{34}^2}\right). \tag{47}$$

Rearranging terms in (44) we conclude that,

$$\frac{\epsilon_4^2-\epsilon_{34}^2}{1-\epsilon_{34}^2}\lambda_2 < \left(1-\epsilon_4^2\right)\left(1-\frac{\lambda_1}{\pi_1}-\frac{\lambda_2}{1-\epsilon_{34}^2}\right)+\frac{\left(1-\epsilon_4^2\right)\pi_3}{\pi_1}\lambda_1. \tag{48}$$

From (45), (46), (47) and (48) we conclude that the arrival rate at $Q_{2,3\bar{4}}$ is smaller than its service rate (see Figure 11), hence this queue is also stable.

Conversely, if one of the inequalities (43), (44) is reversed, it can be shown that at least one of the queues $Q_1^S, Q_2, Q_{2,3\bar{4}}$, is unstable.

Appendix C
Proof of Proposition 4

Setting $q=0$ we see that inequalities (31), (32) imply (26), (27), hence we always have, $\mathcal{R}_{\mathbf{IV}} \subseteq \mathcal{R}_{\mathbf{V}}$. Let $(r_1,r_2) \in \mathcal{R}_{\mathbf{IV}}$. Observe that since $\epsilon_3^1 \geq \epsilon_3^2$, we have $C_1 \geq 0$, hence (26) implies (31). Assume now that $\epsilon_{34}^1 \geq \epsilon_{34}^2$. Then we have $C_2 \leq 0$, hence (27) implies (32), i.e., $\mathcal{R}_{\mathbf{V}} \subseteq \mathcal{R}_{\mathbf{IV}}$, and we conclude $\mathcal{R}_{\mathbf{V}} = \mathcal{R}_{\mathbf{IV}}$. Next, assume that $\epsilon_{34}^1 < \epsilon_{34}^2$, and let $(r_1,r_2) \in \mathcal{R}_{\mathbf{V}} \setminus \mathcal{R}_{\mathbf{IV}}$. Since as mentioned above (26) implies (31), inequality (34) must necessarily hold. Observe also that since $0 \leq q \leq 1$, and $\epsilon_{34}^1 < \epsilon_{34}^2$, it holds $0 \leq \theta(q) \leq \frac{1}{1-\epsilon_{34}^1}$, hence inequality (35) holds. Now, since $(r_1,r_2) \in \mathcal{R}_{\mathbf{V}}$ inequality (32) must be satisfied for some q. Notice that since (34) holds, we must have $q>0$. If this inequality is strict, we can reduce q, without altering inequality (31) (since as can be easily seen $\theta(q)$ is an increasing function of q). Hence we may select q^* so that (32) is satisfied with equality, which implies that (36) holds.

References

[1] A. Papadopoulos, N. Chatzidiamantis, and L. Georgiadis, “Network coding techniques in cooperative cognitive networks,” in *Proc. Global Information Infrastructure and Networking Symposium (GIIS 2018)*, 2018.
[2] S. Haykin, “Cognitive radio: Brain-empowered wireless communications,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
[3] Q. Zhao and B. Sadler, “A survey of dynamic spectrum access,” *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 79–89, May 2007.
[4] C. Peng, H. Zheng, and B. Y. Zhao, “Utilization and fairness in spectrum assignment for opportunistic spectrum access,” *ACM/Springer MONET*, vol. 11, no. 4, pp. 555–576, Aug. 2006.
[5] Y. Chen, Q. Zhao, and A. Swami, “Joint design and separation principle for opportunistic spectrum access in the presence of sensing errors,” *IEEE Trans. Inf. Theory*, vol. 54, pp. 2053–2071, Jan 2008.
[6] R. Urgaonkar and M. J. Neely, “Opportunistic scheduling with reliability guarantees in cognitive radio networks,” *IEEE Trans. Mobile Comput.*, vol. 8, pp. 766–777, Jan 2009.
[7] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, “Breaking spectrum gridlock with cognitive radios: An information theoretic perspective,” *Proc. IEEE*, vol. 97, no. 5, pp. 894–914, Jan 2009.
[8] L. Lv, J. Chen, Q. Ni, Z. Ding, and H. Jiang, “Cognitive non-orthogonal multiple access with cooperative relaying: A new wireless frontier for 5g spectrum sharing,” *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 188–195, Apr. 2018.
[9] L. Lv, Z. Ding, and J. Chen, “Application of non-orthogonal multiple access in cooperative spectrum-sharing networks over nakagami-m fading channels,” *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5510–5515, June 2017.
[10] O. Simeone, Y. Bar-Ness, and U. Spagnolini, “Stable throughput of cognitive radios with and without relaying capability,” *IEEE Trans. Commun.*, vol. 55, pp. 2351–2360, Jan 2007.
[11] I. Krikidis, J. Laneman, J. Thompson, and S. Mclaughlin, “Protocol design and throughput analysis for multi-user cognitive cooperative systems,” *IEEE Trans. Wireless Commun.*, vol. 8, pp. 4740–4751, Jan 2009.
[12] S. Kompella, G. D. Nguyen, C. Cam, J. Wieselthier, and A. Ephremides, “Cooperation in cognitive underlay networks: Stable throughput tradeoffs,” *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 1756–1768, Dec. 2014.
[13] R. Urgaonkar and M. Neely, “Opportunistic cooperation in cognitive femtocell networks,” *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 607 –616, April 2012.
[14] N. D. Chatzidiamantis, E. Matskani, L. Georgiadis, I. Koutsopoulos, and L. Tassiulas, “Optimal primary-secondary user cooperation policies in cognitive radio networks,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3443–3455, June 2015.

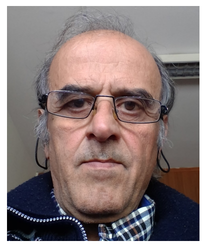
1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

[15] D. Kiwan, A. El Sherif, and T. ElBatt, "Stability analysis of a cognitive radio system with a dedicated relay," in *Proc. IEEE ICNC 2017*, 2017, pp. 750–756.
[16] A. Naeem, M. H. Rehmani, Y. Saleem, I. Rashid, and N. Crespi, "Network coding in cognitive radio networks: A comprehensive survey," *IEEE Commun. Surveys and Tut.*, vol. 19, no. 3, pp. 1945–1973, 2017.
[17] N. Li, L. K. Rasmussen, and M. Xiao, "Performance analysis of cognitive user cooperation using binary network coding," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7355–7369, August 2018.
[18] A. Papadopoulos, N. D. Chatzidiamantis, and L. Georgiadis, "Capacity and algorithms for a cognitive network with primary-secondary user cooperation," *arXiv preprint arXiv:1907.03444*, 2019. [Online]. Available: https://arxiv.org/abs/1907.03444
[19] H. Bruneel, "Performance of discrete-time queueing systems," *Computers & Operations Research*, vol. 20, no. 3, pp. 303–320, 1993.
[20] M. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.
[21] R. W. Wolff, *Stochastic Modeling and the Theory of Queues*. Englewood Cliffs New Jersey: Prentice Hall, 1989.

Athanasios Papadopoulos received the Diploma degree in electrical engineering from Aristotle University, Thessaloniki, Greece, in 2013 where he is currently a Ph.D student. His main research interests are in the area of Information Theory and Wireless Communications.

Nestor D. Chatzidiamantis (S'08, M'14) was born in Los Angeles, CA, USA, in 1981. He received the Diploma degree (5 years) in electrical and computer engineering (ECE) from the Aristotle University of Thessaloniki (AUTH), Greece, in 2005, the M.Sc. degree in telecommunication networks and software from the University of Surrey, U.K., in 2006, and the Ph.D. degree from the ECE Department, AUTH, in 2012. From 2012 through 2015, he worked as a Post-Doctoral Research Associate in AUTH and from 2016 to 2018, as a Senior Engineer at the Hellenic Electricity Distribution Network Operator (HEDNO). Since 2018, he has been Assistant Professor at the ECE Department of AUTH and member of the Telecommunications Laboratory.

His research areas span signal processing techniques for communication systems, performance analysis of wireless communication systems over fading channels, communications theory, cognitive radio and free-space optical communications.



Leonidas Georgiadis received the Diploma degree in electrical engineering from Aristotle University, Thessaloniki, Greece, in 1979, and his M.S. and Ph.D degrees both in electrical engineering from the University of Connecticut, in 1981 and 1986 respectively. From 1986 to 1987 he was Research Assistant Professor at the University of Virginia, Charlottesville. In 1987 he joined IBM T. J. Watson Research Center, Yorktown Heights as a Research Staff Member. Since October 1995, he has been with the Telecommunications Department of Aristotle University, Thessaloniki, Greece.

His interests are in the area of wireless networks, energy efficient communications, information theory and coding, distributed systems, routing,scheduling, congestion control, modeling and performance analysis.

Prof. Georgiadis has coauthored more that 100 journal and conference papers, one book on control of wireless networks and two book chapters. He is coauthor of six patent awards and has participated as a research coordinator or senior researcher in a number of European and National projects. In 1992, he received the IBM Outstanding Innovation Award for his work on goal-oriented workload management for multi-class systems and in 1994 the IBM Research Division Award for work on the architecture and design of broadband network systems (NBBS).

1536-1276 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Aristotle University of Thessaloniki. Downloaded on May 23,2020 at 07:51:39 UTC from IEEE Xplore. Restrictions apply.
Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.