Cooperative Caching in Dynamic Shared Spectrum Networks

Dibakar Das Electrical, Computer and Systems Engineering
Rensselaer Polytechnic Institute
Troy, NY 12180
Email: dasd2@rpi.edu
Alhussein A. Abouzeid Electrical, Computer and Systems Engineering
Rensselaer Polytechnic Institute
Troy, NY 12180
Email: abouzeid@ecse.rpi.edu

Abstract

This paper considers cooperation between primary and secondary users in shared spectrum radio networks via caching. We first consider a network with one channel shared between a single macro (primary) base-station and multiple small (secondary) base-stations. Secondary base-stations can cache some primary files and thereby satisfy content requests generated from nearby primary users. For this cooperative scenario, we develop two caching and scheduling policies under which the set of primary and secondary request generation rates that can be supported is expanded from the case without cooperation. The first of these algorithms, Fixed Primary Caching Policy (FPCP), provides more gain in the set of supportable primary and secondary request generation rates. However under this algorithm primary packet transmissions from secondary base-stations do not have higher priority of access than that of secondary packets and thus might suffer in terms of delay. In the second algorithm, Variable Primary Caching Policy (VPCP), primary packet transmissions from the secondary base-stations have higher priority of access than that of secondary packets. We find that the set of primary and secondary request generation rate vectors for which all queues in the network are stable under each of these algorithms is greater than that under any non-cooperative algorithm. We conduct extensive simulations to compare the performance of both algorithms against that of an optimal non-cooperative algorithm. Finally, we extend the analysis to a network with multiple channels.

I. Introduction

With the huge increase in number of wireless devices, there has been an increasing demand for new spectrum. However, at any time, the licensed spectrum is often under-utilized [1]. This has prompted the widespread study of dynamic shared spectrum or cognitive radio networks. In such networks some secondary (i.e. unlicensed) users are allowed to opportunistically access and transmit on a given channel provided that the primary (i.e. licensed)

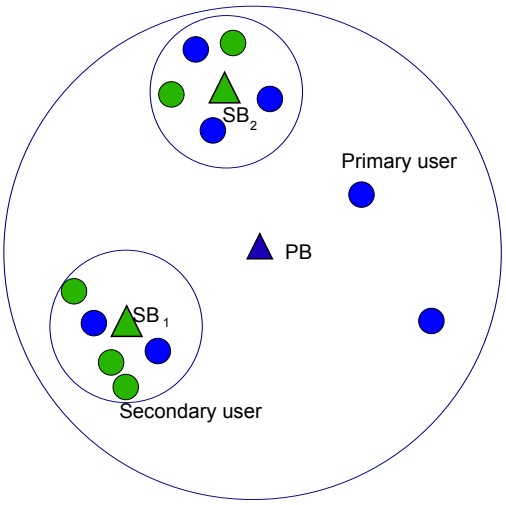


Fig. 1. A network with one primary base-station (PB) co-existing with two secondary base-stations: SB_1 and SB_2. Some primary users are located closer to the secondary base-stations than the primary base-station. The outermost circle indicates the transmission region of the primary base-station; the smaller circles indicate transmission range of secondary base-stations. For this network, number of secondary base-stations (M) is 2, total number of primary users ($N^{(p)}$) and secondary users ($N^{(s)}$) are 7 and 5 respectively, number of primary users within transmission range of SB_1 ($\phi_1^{(p)}$) is 2, number of secondary users within transmission range of SB_1 ($\phi_1^{(s)}$) is 3.

users of that channel are not active. Traditionally, primary and secondary networks have been assumed to be non-cooperative i.e. the primary and secondary users do not assist in each other's transmissions. However, if secondary users choose to assist transmission of primary users, then it may reduce the overall duration for which the primary user is active on that channel. This in turn can also benefit the secondary users because it increases their own transmission opportunities.

Cooperation between primary and secondary networks have been widely studied from a physical-layer perspective. Some of these works e.g. [2]–[4] study it as an information-theoretic problem. Other recent works e.g. [5]–[9] study network-layer aspects of cooperation such as queuing and prioritized scheduling. Our work belongs to those latter group of works wherein we consider joint caching and scheduling from a network-layer perspective.

Another solution that has been proposed to support increasing mobile network traffic is the use of base-stations with smaller coverage areas (commonly called small-cells). This leads to higher spatial re-use of the spectrum. However, limited capacity of backhaul links at the base-stations may reduce the impact of this approach [10]. As a result, caching popular files at nearby base-stations has been proposed (e.g. in [10]) to reduce back-haul usage as well as improve the delay performance for users. Caching is also an attractive practical solution since storage-capacity is relatively inexpensive compared to other network resources.

In this work we explore cooperation between primary and secondary networks via caching. We consider a primary network consisting of a single base-station that serves primary users. Co-existing with the primary network is a set of small secondary base-stations that serve secondary users. An example of such a network is shown in Fig.1. Both primary and secondary users request content from their respective base-stations where these requests

are queued. The base-stations serve these requests by transmitting packets corresponding to the requested files from their cache if the file is available in the cache. If the requested file is not currently present in the cache (i.e. if it is not 'cached'), then the base-station fetches this file, possibly after some delay, so as to satisfy the content request. We assume that content is fetched periodically and we refer to this period as the cache-refresh period. Secondary base-stations located closer to primary users may have better down-link channels than the primary base-station. Therefore, if some of the content requests from primary users are served via these secondary base-stations then it might free up spectrum resources to be used by the secondary users.

For the above network scenario we address the important problem of developing caching and scheduling policies with performance guarantees. In particular, we design algorithms under which a centralized network controller determines (a) which files to cache at the secondary base-stations in every cache-refresh period and (b) which file-requests to admit at a given base-station and schedule transmissions in each time slot. The goal of the network controller is to maintain stability of the queues i.e., to keep the length of all queues in the network bounded. Accordingly, our performance measure is the set of primary and secondary request generation rates for which every queue in the network is stable.

For a network with one channel we develop two algorithms: Fixed Primary Caching Policy (FPCP) and Variable Primary Caching Policy (VPCP), using Lyapunov-drift techniques. Both policies make decisions without knowledge of the request generation rates of secondary users. We find that the set of primary and secondary request generation rate vectors for which all queues in the network are stable under each of these algorithms is greater than that under any non-cooperative algorithm. Under the FPCP algorithm each secondary base-station caches only one secondary file in every period while filling up rest of the cache with primary files. However in FPCP, only primary packet transmissions from the primary base-station enjoy higher priority of channel access. In the VPCP algorithm however, requests from primary users queued at any secondary base-station are always served with higher priority than that from secondary users. Furthermore, in every cache refresh period, the network-controller dynamically varies the number of cached primary files while ensuring system stability. Hence the secondary base-stations can serve more than one type of secondary file requests in each cache refresh period. Simulation results in Section VII show that this algorithm, with proper selection of a penalty parameter, tends to have better delay performance than the FPCP algorithm when request generation rates are low. In such scenarios it is not necessary for system stability to cache as much primary files as possible in every secondary base-station. In Section VII we also show that the delay performance of VPCP is not guaranteed to be better than FPCP for any choice of the penalty parameter. However, the guaranteed stability region of the network under VPCP is less than that under the FPCP algorithm.

Finally, we consider a network with multiple orthogonal channels where all secondary base-stations are non-interfering to each other (i.e. they can simultaneously transmit when PB is not transmitting) and develop an algorithm for this scenario referred to as MultiChannel Caching Policy (MCCP).

There is a substantial body of literature on caching in non-cognitive wireless networks (e.g. [10]–[12] and the references therein). On the other hand, caching in cognitive networks has been studied recently in [13] and [14]. However they did not consider primary-secondary cooperation. Our periodic cache-refresh policy and the use of

Lyapunov drift to develop a scheduling policy is motivated by [15] and [16]. However, their results are not directly applicable in the cognitive network setting since here primary users have higher priority of channel access than secondary nodes.

The novelty of our work lies in showing that Lyapunov drift techniques can be used to design efficient joint cooperative caching and scheduling algorithms in dynamic shared spectrum networks. The difficulty of designing such algorithms lies in higher priority of channel access for primary users. While a simple Lyapunov drift-based algorithm tries to serve the queues with higher backlogs first, in our model the queues in the primary base-station would have to be served before that at the secondary base-stations even when it has relatively lower backlog. The difficulty of developing caching and scheduling algorithms is even higher when primary user requests are served with higher priority even at secondary base-stations, a scenario considered in the VPCP algorithm. Such complications resulting from higher priority of service for primary users are addressed in this work, and up to our knowledge, have not been addressed before.

The rest of the paper is organized as follows. In Section II we describe our system model for a single channel case. In Section III, we define an achievable capacity region for this network consisting of supportable primary and secondary request generation rate vectors. This region is used to measure the stability performance of the FPCP, VPCP and MCCP algorithms. In Section IV we present the FPCP algorithm and show that under this algorithm all queues are stable for every request generation rate vector within the achievable capacity region. In Section V we present the VPCP algorithm and find a guaranteed stability region for it. Section VI presents the multichannel network model and an achievable capacity region. In Section VI we also present the MCCP algorithm that stabilizes the network for all request generation vectors within the achievable capacity region. In Section VII we present simulation results. Section VIII concludes the paper.

II. System model

The network consists of a macro-cell wherein a single primary base-station PB serves $N^{(p)}$ primary users: PU_1,. . .,$\text{PU}_{N^{(p)}}$. It also contains M secondary small-cells $\text{SC}_1, \ldots, \text{SC}_M$ associated with small base-stations SB_1, . . . ,SB_M respectively. These base-stations together serve $N^{(s)}$ secondary users: SU_1,. . .,$\text{SU}_{N^{(s)}}$. Each secondary user is served by exactly one of those small base-stations. We consider a discrete time model. Every file request is served by successfully transmitting C packets of equal size. A base-station attempts a new packet transmission only at the beginning of a time slot; it can attempt at most one such transmission at any time slot. At the end of the time slot the transmission is either successful and the packet is successfully received by the desired user or the transmission fails and the packet needs to be re-transmitted at some other time slot. Next, we present details about the transmission model, interference model, caching and scheduling model.

A. Transmission Model for Primary and Secondary Base-stations

The primary and secondary base-stations transmit at fixed power. All secondary base-stations have identical transmission range. We assume that the probability of a successful transmission from PB to some primary users

is lower than that from adjacent secondary base-stations. This could arise due to a variety of reasons such as shadowing, or due to near-far effect, where some primary users happen to be closer to a secondary base-station than to a primary one. For simplicity, we assume non ideal transmissions from PB to primary users, but ideal transmission from secondary base-stations to users within its transmission range, because we are interested in studying the best case gains with secondary cooperation:

A1) At every time slot, a transmission from PB to any primary user succeeds with probability p (where $0 < p \leq 1$); a transmission from any secondary base-station to a user within its transmission range succeeds with probability 1.

Each primary user is served by at most one secondary base-station i.e. no two small-cells contain the same primary user. We denote the set of primary and secondary users in SC_i, $i \in \{1, \ldots, M\}$, as $\phi_i^{(p)}$ and $\phi_i^{(s)}$ respectively. All secondary users and base-stations are within the transmission range of PB and share a single channel.

B. Caching Model

Every time a secondary base-station fetches a set of files for caching it incurs some overhead cost. This cost is modeled by requiring that secondary base-stations only cache files periodically. A higher frequency of caching reflects a higher cost. A cache-refresh period is defined as the number of time slots between two successive caching events. A cache refresh period may also represent the frequency with which contents in files become outdated thereby requiring newer versions to be fetched. It consists of T time slots with the first caching event being at time slot $t = 1$. Henceforth, we will refer to the cache refresh period simply as a period whenever there is no confusion.

C. Content Requests

We consider the case where primary and secondary networks are different, and hence have statistically different content requirements [1]. This can occur, for example, if the small-cells serve an industrial or academic environment while users of the macro-cell are the general public who are typically interested in video content. We denote the library of files requested by primary users as $F^{(p)}$ with individual files in the set being denoted as $\text{F}_1^{(p)}$,…,$\text{F}_{|F^{(p)}|}^{(p)}$. Similarly we denote the library of files requested by secondary users as $F^{(s)}$ with individual files in the set being denoted as $\text{F}_1^{(s)}$,…,$\text{F}_{|F^{(s)}|}^{(s)}$. The sets $F^{(p)}$ and $F^{(s)}$ are mutually exclusive. Henceforth we will call files in $F^{(p)}$ and $F^{(s)}$ as primary and secondary files respectively. Similarly, we call packets corresponding to primary and secondary files as primary and secondary packets respectively. All files are of equal size. We denote the set of primary and secondary files cached by secondary base-station SB_k, $k \in \{1, \ldots, M\}$ at time slot τ as $H_k^{(p)}(\tau)$ and $H_k^{(s)}(\tau)$ respectively where $H_k^{(p)}(\tau) \in F^{(p)}$ and $H_k^{(s)}(\tau) \in F^{(s)}$.

Users request files according to a fixed popularity distribution. Given a primary user requests a file, the file $\text{F}_i^{(p)}$ is requested with probability $P_i^{(p)}$. Similarly, given a secondary user requests a file, the file $\text{F}_i^{(s)}$ is requested with probability $P_i^{(s)}$. Without loss of generality, we assume files are indexed according to their popularity i.e.,

[1]The case where the users have homogeneous content requirements can be studied in a similar fashion.

$P_i^{(p)} \geq P_{i+1}^{(p)}$ and $P_j^{(s)} \geq P_{j+1}^{(s)}$ for every $i \in \{1, 2, \ldots, |F^{(p)}| - 1\}$ and $j \in \{1, 2, \ldots, |F^{(s)}| - 1\}$. At every time slot, primary user PU$_m$, $m \in \{1, \ldots, N^{(p)}\}$, requests a file with probability $\lambda_m^{(p)}$. Similarly, at every time slot, secondary user SU$_l$, $l \in \{1, \ldots, N^{(s)}\}$, requests files with probability $\lambda_l^{(s)}$. All request generation processes are identical and independently distributed (iid) from time slot to time slot. We denote primary and secondary request generation rate vectors:$(\lambda_1^{(p)}, \ldots, \lambda_{N^{(p)}}^{(p)})^T$ and $(\lambda_1^{(s)}, \ldots, \lambda_{N^{(s)}}^{(s)})^T$ as $\boldsymbol{\lambda}^{(p)}$ and $\boldsymbol{\lambda}^{(s)}$ respectively.

D. Service Discipline for Primary Users

A file request from a primary user is served by either a secondary base-station (if the primary user is located in a small-cell and the associated secondary base-station contains the file) or by the primary base-station. Every base-station maintains *queues* for every file. Queues store packet requests that need to be satisfied in response to a file request. Whenever a base-station receives request for a primary file $f \in F^{(p)}$, C packet requests are queued at the queue for file f[2]. Within each queue packet requests are satisfied in a first-come first-serve (FCFS) manner.

In order to focus only on the effect of cooperative caching by the secondary network, we assume the primary base-station can cache all the $|F^{(p)}|$ primary files. On the other hand, the size of cache at each secondary base-station is finite. Each such cache can store at most B files where $0 \leq B \leq \min(|F^{(p)}|, |F^{(s)}|)$. A secondary base-station only admits a primary file request if the requested file is currently present in its cache. We indicate a request generated from PU$_i$, $i \in \{1, \ldots, N^{(p)}\}$, at time slot t for file f (where $f \in F^{(p)}$) by variable $A_{f,i}^{(p)}(t)$. This variable equals C if such a request is indeed generated at t; otherwise it equals 0.

We denote the queue length of primary file f (where $f \in F^{(p)}$) in SB$_k$, $k \in \{1, \ldots, M\}$, at time slot t as $U_{f,k}^{(p)}(t)$. If PU$_i$ is within transmission range of a secondary base-station SB$_k$ containing the file f requested by PU$_i$ at t, then SB$_k$ *may* admit this file request [3]; C packet requests are queued at SB$_k$ and $U_{f,k}^{(p)}(t)$ is incremented by C. Otherwise, this request will be served by PB and the length of the associated queue, denoted as $U_{\text{f,0}}^{(p)}(t)$, will be incremented by C. The system begins at time slot t=1 with all queues being initially empty.

On successful transmission of a packet corresponding to primary file f (where $f \in F^{(p)}$) from PB at t, $U_{\text{f,0}}^{(p)}(t)$ is decremented by 1. We denote the transmission rate offered to base-station PB and SB$_k$ for packets of file f at time slot t by the binary variables $\mu_{f,0}(t)$ and $\mu_{f,k}(t) \in \{0, 1\}$ respectively. Therefore the queue length of every primary file f at PB is updated as follows:

$$\begin{aligned} U_{\text{f,0}}^{(p)}(t+1) &= U_{\text{f,0}}^{(p)}(t) - \mu_{f,0}(t) I_{\text{PB}}(t) + \sum_{i:\text{PU}_i \notin \cup_{j=1}^{M} \phi_j^{(p)}} A_{f,i}^{(p)}(t) + \sum_{k=1}^{M} \sum_{i:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \left(1 - \hat{I}_{f,k}(t)\right) \\ &+ \sum_{k=1}^{M} \sum_{i:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \left(1 - \chi_{f,k}^{(p)}(t)\right) \hat{I}_{f,k}(t) \quad \forall f \in F^{(p)}, \end{aligned} \tag{1}$$

[2]Recall, each file request is served by successfully transmitting C packets.

[3]The admission control decision is taken by the network controller. The controller may decide not to admit a primary file request at a secondary base-station even when the requested file is present in the latter's cache.

where $I_{\text{PB}}(t)$ is an indicator variable representing whether the transmission from PB to the primary user at time slot t was successful or not, $\chi_{f,k}^{(p)}(t)$ is an admission-control variable representing whether SB_k (where $k \in \{1,\ldots,M\}$) admits request at time slot t for cached primary file $f \in H_k^{(p)}(t)$ and $\hat{I}_{f,k}(t)$ is an indicator variable representing whether SB_k (where $k \in \{1,\ldots,M\}$) contains a primary file f at time slot t or not. The variable $I_{\text{PB}}(t)$ equals 1 if the transmission is successful at t and is zero otherwise; $\chi_{f,k}^{(p)}(t)$ equals 1 if SB_k admits request for f at t and is zero otherwise. The variable $\hat{I}_{f,k}(t)$ equals 1 if SB_k contains file f at t and is zero otherwise. No transmission-rate is offered to PB for transmitting packets corresponding to an empty queue.

The queue length of cached primary files at the secondary base-stations are updated as follows:

$$U_{f,k}^{(p)}(t+1) = \max\left\{U_{f,k}^{(p)}(t) - \mu_{f,k}(t), 0\right\} + \sum_{i:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t)\chi_{f,k}^{(p)}(t) \quad \forall k \in \{1,\ldots,M\}, f \in H_k^{(p)}(t). \tag{2}$$

E. Service Discipline for Secondary Users

Requests from a secondary user is submitted to the unique secondary base-station associated with the user. The secondary base-station always admits such requests. Similar to the case of primary files, each secondary base-station maintains a queue for every secondary file. We indicate a request generated from SU_j, $j \in \{1,\ldots,N^{(s)}\}$, at time slot t for a secondary file f (where $f \in F^{(s)}$) by a variable $A_{f,j}^{(s)}(t)$. It equals C if such a request is indeed generated at t; otherwise it equals 0. For every $k \in \{1,\ldots,M\}$ we denote the queue length in SB_k at t of file f as $U_{f,k}^{(s)}(t)$. The length of this queue is incremented by C every time SB_k receives a request for file f. It is decremented by 1 every time SB_k transmits a packet of f. The queue is updated as,

$$U_{f,k}^{(s)}(t+1) = \max\left\{U_{f,k}^{(s)}(t) - \mu_{f,k}(t), 0\right\} + \sum_{j:\text{SU}_j \in \phi_k^{(s)}} A_{f,j}^{(s)}(t) \quad \forall k \in \{1,\ldots,M\}, f \in F^{(s)}. \tag{3}$$

We refer to the queues corresponding to primary and secondary files as primary and secondary queues respectively.

F. Interference Model

We represent the set of all base-stations that can transmit simultaneously by an *activation* vector of length M. An activation vector E is binary and its m'th, $m \in \{1,\ldots,M\}$, component corresponds to SB_m. It is set to 1 if SB_m is transmitting in that time slot; otherwise it is set to zero. The set of all feasible activation vectors is denoted as $\tilde{E}$. No secondary base-station can transmit when PB is transmitting.

III. Achievable Capacity Region

In this section we describe an achievable capacity region Λ corresponding to the system model described in Section II. The region contains every primary and secondary request generation rate vector for which all queues in the network are stable considering only a *restricted* set of caching and scheduling algorithms. First we specify this restricted set of algorithms that satisfy a certain caching constraint; this constraint is useful for tractable analysis. Then we introduce a new term called the *primary availability matrix* which represents the set of primary files currently cached at each secondary base-station. Next, we obtain the set of feasible transmission rates for secondary

base-stations given the probability with which each primary availability matrix is used in each period. We use the set of feasible transmission rates for secondary base-stations to describe the region Λ by establishing stability constraint at each secondary base-station similar to the description of capacity region in [17]. Next, we find a simpler description of the achievable capacity region in Lemma 1. Finally, for the network in which all secondary base-stations are non-interfering we compare the achievable capacity region with the actual capacity region in Lemma 2. The actual capacity region is the set of primary and secondary request generation rate vector for which every queue in the network is stable considering all stationary caching and scheduling algorithms, not just the restricted ones. In the next section we will present the FPCP algorithm which stabilizes all queues for every request generation rate vectors in the interior of Λ.

In the description of the region Λ we only consider the set of caching and scheduling algorithms that satisfy the following constraint:

C1) Each secondary base-station has at least one secondary file in its cache in every slot.

We consider only this restricted set of algorithms instead of all caching and scheduling algorithms because it allows for tractable analysis. In particular, it allows us to determine the stability constraints at secondary base-stations without accounting for probability of the event: "the network controller offers transmission rate to a secondary base-station which has no cached secondary file". Obtaining probability of such events is cumbersome but is required to analyze stability performance of algorithms which do not satisfy constraint **C1**.

Now we introduce the term: primary availability matrix. A primary availability matrix is a binary matrix that indicates the availability of primary files in secondary base-stations at any given time slot. Each such matrix contains the same number of rows and columns as the total number of primary files and number of secondary base-stations respectively. The (l, j)'th component of a primary availability matrix $\mathbf{D}^{(p)}$ (where $l \in \{1, \ldots, |F^{(p)}|\}$, $j \in \{1, \ldots, M\}$), denoted as $D_{l,j}$, equals 1 if the file $\mathrm{F}_l^{(p)}$ is present at the cache of SB_j; otherwise it equals zero. Given a primary availability matrix $\mathbf{D}^{(p)}$, we indicate whether or not a primary file f (where $f \in F^{(p)}$) requested by PU_i, $i \in \{1, \ldots, N^{(p)}\}$, is present in a nearby secondary base-station by the binary variable $Q(i, f|\mathbf{D}^{(p)})$. In particular, for every primary file $\mathrm{F}_l^{(p)}$, the variable $Q(i, \mathrm{F}_l^{(p)}|\mathbf{D}^{(p)})$ equals 1 if there exists j such that (s.t.) $D_{l,j}^{(p)} = 1$ and $\mathrm{PU}_i \in \phi_j^{(p)}$; it is set to 0 otherwise. Since we consider only algorithms that satisfy constraint **C1** we consider only the set of primary availability matrices for which the sum of every column is less than B. We denote this set as $\tilde{D}^{(p)}$.

We find the set of feasible transmission rates for secondary base-stations, given a probability distribution over the set of primary availability matrices in $\tilde{D}^{(p)}$, in the following manner. Consider the vector $\boldsymbol{q} = (q_{\mathbf{D}^{(p)}})$ where the term $q_{\mathbf{D}^{(p)}}$ denotes the iid probability with which the primary availability matrix $\mathbf{D}^{(p)} \in \tilde{D}^{(p)}$ is selected in each period. For every $\mathbf{D}^{(p)}$ the term $q_{\mathbf{D}^{(p)}}$ should be non-negative and less than 1 (i.e. $0 \leq q_{\mathbf{D}^{(p)}} \leq 1$) and the sum of all $q_{\mathbf{D}^{(p)}}$ terms should equal 1 (i.e. $\sum_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} q_{\mathbf{D}^{(p)}} = 1$). Clearly, due to assumption A1, given a set of primary files cached at secondary base-stations, transmission opportunities for secondary base-stations are maximized when each secondary base-station admits all requests of a cached primary file. Then, the rate of primary packet transmissions by

all secondary base-stations, equals $C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\mathrm{F}_l^{(p)}|\mathbf{D}^{(p)})q_{\mathbf{D}^{(p)}}$. For a stable system the rate of primary packet transmissions by PB is the sum of rate of primary packet requests minus the rate of primary user requests served by secondary base-stations i.e., $\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q\left(k,\mathrm{F}_l^{(p)}|\mathbf{D}^{(p)}\right)q_{\mathbf{D}^{(p)}}$. The probability that PB is not transmitting in a given time slot is therefore, $\left\{1-\frac{\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q\left(k,\mathrm{F}_l^{(p)}|\mathbf{D}^{(p)}\right)q_{\mathbf{D}^{(p)}}}{p}\right\}$. We denote as $\Gamma(\boldsymbol{q})$ the set of feasible secondary transmission-rate vectors for a given vector $\boldsymbol{q}$. The set $\Gamma(\boldsymbol{q})$ is defined in average sense as the convex hull of all feasible activation vectors when PB is not transmitting, multiplied by the probability that PB is not transmitting, as

$$\Gamma(\boldsymbol{q})=\left\{1-\frac{\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q\left(k,\mathrm{F}_l^{(p)}|\mathbf{D}^{(p)}\right)q_{\mathbf{D}^{(p)}}}{p}\right\}\mathbf{conv}(\tilde{E}) \tag{4}$$

where **conv** of a set of vectors is the set of all possible convex combinations of its elements.

We use the set of feasible transmission rate vectors for secondary base-stations to define the achievable capacity region Λ. The region Λ is the set of all possible primary and secondary request generation rate vectors: $(\boldsymbol{\lambda}^{(p)},\boldsymbol{\lambda}^{(s)})$ for which there exists vector $\boldsymbol{q}$, $(R_1,\ldots,R_M)^T\in\Gamma(\boldsymbol{q})$ and variable π_0 s.t.

$$\frac{\pi_0}{p}\leq 1 \tag{5}$$

$$C\left\{\sum_{i:\mathrm{SU}_i\in\phi_j^{(s)}}\lambda_i^{(s)}+\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{k:\mathrm{PU}_k\in\phi_j^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}\lambda_k^{(p)}D_{l,j}^{(p)}q_{\mathbf{D}^{(p)}}\right\}\leq R_j\quad\forall 1\leq j\leq M, \tag{6}$$

where

$$\pi_0=C\left\{\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}-\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\mathrm{F}_l^{(p)}|\mathbf{D}^{(p)})q_{\mathbf{D}^{(p)}}\right\}. \tag{7}$$

The term π_0 in (7) represents the rate of successful packet transmissions by PB corresponding to the vector $\boldsymbol{q}$. The right hand side (RHS) of (7) is the product of the rate of primary user requests served by PB and C, the number of successful packet transmissions corresponding to every served file request. The constraint (5) is the stability constraint at PB: the rate of packet transmissions by PB cannot exceed 1. The inequality constraint (6) represents the stability requirement at secondary base-stations. The left hand side (LHS) represents total arrival rate into all the queues in a given secondary base-station; the RHS represents a feasible transmission rate offered to that base-station.

Remark 1: The region Λ contains the set of primary and secondary request generation rate vectors supportable without cooperation as the all-zero primary availability matrix (i.e. one whose every component is 0) belongs to the set $\tilde{D}^{(p)}$.

We now simplify the description of the achievable capacity region. Intuitively, we observe that secondary base-stations create maximum transmission opportunities for themselves by transmitting as much primary traffic as

possible. This means that the achievable capacity region Λ can be described by only considering policies in which each secondary base-station, in every period, caches exactly $B-1$ most popular primary files and admits all user requests for those files. We state this formally in Lemma 1.

Lemma 1: The region Λ can be defined by considering, in (5) and (6), only the vector $\boldsymbol{q}$ for which $q_{\boldsymbol{D}^*}=1$ and $q_{\boldsymbol{D}}=0$ for every primary availability matrix $\boldsymbol{D}\neq\boldsymbol{D}^*$ where

$$D^*_{n,j}=\begin{cases}1, & \text{if } 1\leq n\leq B-1, \quad 1\leq j\leq M\\ 0, & \text{otherwise.}\end{cases} \tag{8}$$

Proof: Proof is provided in Appendix A. ■

Finally, we compare the achievable capacity region Λ with the general capacity region Λ_{gen} for a network in which all secondary base-stations are non-interfering. The region Λ_{gen} consists of all primary and secondary request generation rate vectors for which the network is stable under any stationary scheduling and caching algorithm, and not just the restricted set of algorithms that satisfy constraint **C1**. Clearly, $\Lambda_{\text{gen}}\supseteq\Lambda$. In particular, we compare the total secondary request generation rate that can be maximally supported by each secondary base-station under the restricted set of algorithms and that under all stationary algorithms. We quantify the performance gap, measured in terms of supportable secondary request generation rates, between these two sets of policies as follows.

For a given $\boldsymbol{\lambda}^{(p)}$, let $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$ and $\lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{\lambda}^{(p)})$ denote the maximum total secondary request generation rate that can be supported (i.e. all queues in the network are stable) at secondary base-station SB_j, $j\in\{1,\ldots,M\}$, under the restricted set of policies and under all policies respectively. Then $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$ is lower bounded as follows.

Lemma 2: Consider a network in which all secondary base-stations are non-interfering. For any primary request generation rate vector $\boldsymbol{\lambda}^{(p)}$ which is present in Λ (i.e. there exists $\boldsymbol{\lambda}^{(s)}$ such that $\left(\boldsymbol{\lambda}^{(p)},\boldsymbol{\lambda}^{(s)}\right)\in\Lambda$) we have for every $j\in\{1,\ldots,M\}$,

$$\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)}) \;\geq\; \lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{\lambda}^{(p)})-(\frac{1}{p}-1)\sum_{k:\text{PU}_k\in\phi^{(p)}_j}\lambda^{(p)}_k P^{(p)}_B-\frac{\sum\limits_{\substack{n=1\\ n\neq j}}^{M}\sum\limits_{k:\text{PU}_k\in\phi^{(p)}_n}\lambda^{(p)}_k P^{(p)}_B}{p} \quad . \tag{9}$$

Proof: We first find the expression of $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$ using Lemma 1 and non-interfering nature of the secondary base-stations: any secondary base-station can transmit packet, independent of other secondary base-stations, whenever PB is not transmitting. Next, we obtain an upper bound of $\lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{\lambda}^{(p)})$ in a similar manner as we derived $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$. Comparing those two expressions of $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$ and $\lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{\lambda}^{(p)})$ we obtain (9). The detailed proof is provided in Appendix A. ■

IV. Fixed Primary Caching Policy

In this section we present FPCP, an algorithm that stabilizes all queues in the network for every primary and secondary request generation rate vector in the interior of the achievable capacity region Λ. The algorithm is

constructed using Lyapunov drift techniques that are widely used to develop efficient scheduling algorithms in communication networks. In order to construct the algorithm we first obtain expressions for the evolution of queues in each period. We then use Lemma 1 to simplify the expression for the evolution of primary queues. Then we apply the Lyapunov drift technique to find an upper bound of the conditional drift of a Lyapunov function of queue lengths[4]. We then minimize this upper bound to find the set of files cached at the secondary base-stations in each period. Given the set of files currently cached, in the FPCP algorithm we schedule packet transmissions according to a modified backpressure policy. We formally state the algorithm in Table 1 and analyze its stability performance in Theorem 1. Finally, we briefly discuss a drawback of the FPCP algorithm.

First we obtain expressions for evolution of the primary and secondary queues in each secondary base-station over any period. Suppose in the r'th period (where $r \in \{1, 2, \dots, \}$), under some caching and scheduling policy X, the network controller uses the primary availability matrix $D^{(p)} \in \tilde{D}^{(p)}$. Then the length of primary queues in secondary base-stations evolve as

$$\begin{aligned} U_{f,k}^{(p)}(t_{r+1,1}) \quad \leq \quad & \max\left\{U_{f,k}^{(p)}(t_{r,1}) - \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \mu_{f,k}^{X}(\tau), 0\right\} \\ & + \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \sum_{i:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(\tau) Q(i, f|D^{(p)}) \chi_{f,k}^{(p)}(\tau) \quad \forall k \in \{1, \dots, M\}, f \in F^{(p)}, \end{aligned} \tag{10}$$

where $t_{r,j}$ denotes the j'th time slot in r'th period ($j \in \{1, \dots, T\}$); $\mu_{f,k}^{X}(\tau)$ denotes the transmission rate offered to SB_k, under policy X, to transmit a packet of file f at time slot τ for every $f \in F^{(p)} \cup F^{(s)}$ and $k \in \{1, \dots, M\}$. Similarly, the secondary queues in each secondary base-station evolve as

$$U_{f,k}^{(s)}(t_{r+1,1}) \leq \max\left\{U_{f,k}^{(s)}(t_{r,1}) - \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \mu_{f,k}^{X}(\tau), 0\right\} + \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \sum_{i:\text{SU}_i \in \phi_k^{(s)}} A_{f,i}^{(s)}(\tau) \quad \forall k \in \{1, \dots, M\}, f \in F^{(s)}. \tag{11}$$

Next, we simplify the expression for primary queue evolution i.e. (10) by utilizing Lemma 1. Recall, according to Lemma 1, the achievable capacity region Λ can be described by considering only those policies under which, in every period, each secondary base-station caches exactly $B-1$ most popular primary files and always admits requests corresponding to those files. Therefore, in the construction of the FPCP algorithm we consider only those policies. Under any such policy X we observe, from (10), that the primary queue length in secondary base-stations evolve for every $r \in \{1, 2, \dots, \}$ and $k \in \{1, \dots, M\}$ as

$$U_{f,k}^{(p)}(t_{r+1,1}) \quad \leq \quad \begin{cases} \max\left\{U_{f,k}^{(p)}(t_{r,1}) - \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \mu_{f,k}^{X}(\tau), 0\right\} + \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \sum_{i:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(\tau), & \forall f \in \{\text{F}_1^{(p)}, \dots, \text{F}_{B-1}^{(p)}\} \\ 0, & \text{else.} \end{cases} \tag{12}$$

We refer to the aggregate of all primary queues in SB_k as the *aggregate primary queue* in SB_k for every $k \in \{1, \dots, M\}$. Clearly, the length of the aggregate primary queue in SB_k, denoted as $U_k^{(p)}(t)$, is the sum of the

[4]The k-slot conditional drift of a Lyapunov function of instantaneous queue lengths, $V(t)$ is $E[V(t+k) - V(t)|pmbU(t)]$ [17] where $pmbU(t)$ is a vector of queue lengths at time slot t.

length of all primary queues i.e., $U_k^{(p)}(t) = \sum_{f\in F^{(p)}} U_{f,k}^{(p)}(t)$. From (12) we observe that the aggregate primary queue in SB_k evolve as

$$U_k^{(p)}(t_{r+1,1}) \leq \max\left\{U_k^{(p)}(t_{r,1}) - \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\mu_k^{X,(p)}(\tau), 0\right\} + \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\sum_{i:\text{PU}_i\in\phi_k^{(p)}}\sum_{f=\text{F}_1^{(p)}}^{\text{F}_{B-1}^{(p)}} A_{f,i}^{(p)}(\tau) \quad \forall k\in\{1,\ldots,M\}, \tag{13}$$

where $\mu_k^{X,(p)}(\tau) \triangleq \sum_{f\in F^{(p)}}\mu_{f,k}^X(\tau)$, $k\in\{1,\ldots,M\}$, denotes the transmission rate offered to SB_k under policy X to transmit a packet from its aggregate primary queue at time slot τ.

We now apply the Lyapunov drift technique. In Lyapunov drift technique the conditional drift of a Lyapunov function of queue lengths is minimized. In particular, we consider the Lyapunov function of queue lengths at the beginning of each period, $L_1(r) \triangleq \sum_{k=1}^{M}\{\sum_{f\in F^{(s)}}(Z_{f,k}^{(s)}(r))^2 + (Z_k^{(p)}(r))^2\}$ where,

$$Z_{f,k}^{(s)}(r) \triangleq U_{f,k}^{(s)}(t_{r,1}) \quad \forall f\in F^{(s)}, k\in\{1,\ldots,M\} \tag{14}$$

$$Z_k^{(p)}(r) \triangleq U_k^{(p)}(t_{r,1}) \quad \forall k\in\{1,\ldots,M\}. \tag{15}$$

We define its conditional drift as,

$$\Delta_1(r) \triangleq E[L_1(r+1) - L_1(r)|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)], \tag{16}$$

where $\boldsymbol{Z}^{(s)}(r)$ and $\boldsymbol{Z}^{(p)}(r)$ denote the vector of secondary and aggregate primary queue lengths in all secondary base-stations at the beginning of the r'th period respectively. The conditional drift term depends on the caching and scheduling algorithm used. From (13) we obtain the following upper bound of $\Delta_1(r)$:

$$\begin{aligned}
\Delta_1(r) \leq\ & E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\{(\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\mu_{f,k}^X(\tau))^2 + (\sum_{i:\text{SU}_i\in\phi_k^{(s)}}\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}A_{f,i}^{(s)}(\tau))^2\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \\
& + E[\sum_{k=1}^{M}\{(\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\mu_k^{X,(p)}(\tau))^2 + (\sum_{f=\text{F}_1^{(p)}}^{\text{F}_{B-1}^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}}\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}A_{f,i}^{(p)}(\tau))^2\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \\
& - 2\sum_{k=1}^{M}\sum_{f\in F^{(s)}}E[Z_{f,k}^{(s)}(r)\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\{\mu_{f,k}^X(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}}A_{f,i}^{(s)}(\tau)\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \\
& - 2\sum_{k=1}^{M}E[Z_k^{(p)}(r)\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\{\mu_k^{X,(p)}(\tau) - \sum_{f=\text{F}_1^{(p)}}^{\text{F}_{B-1}^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}}A_{f,i}^{(p)}(\tau)\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)].
\end{aligned} \tag{17}$$

Since, in every period, the number of arrivals to each queue and number of packet transmissions by each base-station is upper bounded by some finite positive constant $\hat{T}$, therefore from (16) under any policy X we have,

$$\begin{aligned}
\Delta_1(r) \leq\ & 2\hat{T}^2M(1+|F^{(s)}|) - 2\sum_{k=1}^{M}\sum_{f\in F^{(s)}}E[Z_{f,k}^{(s)}(r)\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\{\mu_{f,k}^X(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}}A_{f,i}^{(s)}(\tau)\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \\
& - 2\sum_{k=1}^{M}E[Z_k^{(p)}(r)\sum_{\tau=t_{r,1}}^{t_{r+1,1}-1}\{\mu_k^{X,(p)}(\tau) - \sum_{f=\text{F}_1^{(p)}}^{\text{F}_{B-1}^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}}A_{f,i}^{(p)}(\tau)\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)].
\end{aligned} \tag{18}$$

In the FPCP algorithm we find the set of secondary files to be cached at each secondary base-station in the r'th period by minimizing the above upper bound of the conditional drift $\Delta_1(r)$ i.e. the RHS of (18). We perform this minimization over all polices X under which each secondary base-station caches exactly $B-1$ most popular primary file in every period. Therefore under the FPCP algorithm, for every strictly positive integer r, in the r'th period each secondary base-station SB_k (where $k \in \{1,\ldots,M\}$) can cache only one secondary file, denoted as $f_k^*(r)$. Since any secondary base-station can only transmit packets of a cached file, we obtain $f_k^*(r)$ by minimizing the RHS of (18) as,

$$f_k^*(r) \quad \in \quad \underset{f\in F^{(s)}}{\text{argmin}} -2\sum_{k=1}^{M} E[Z_{f,k}^{(s)}(r) \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \mu_{f,k}^{X}(\tau)|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \tag{19}$$

$$\text{i.e. } f_k^*(r) \quad \in \quad \underset{f\in F^{(s)}}{\text{argmax}} U_{f,k}^{(s)}(t_{r,1}) \quad \forall k = 1,2,\ldots,M. \tag{20}$$

Given the set of files cached at each base-station in any period, in the FPCP algorithm we schedule packet transmissions from base-stations according to a modified backpressure policy. At any time slot when PB is not transmitting, we schedule packet transmissions by solving a max-weight optimization problem of instantaneous queue lengths of only the cached files in all base-stations.

Table 1: FPCP Algorithm

For every strictly positive integer r following steps are performed in the r'th period:

1) **Caching Scheme:** At the beginning of the period, every secondary base-station caches the $B-1$ most popular primary files. In addition, every secondary base-station SB_k (where $k \in \{1,\ldots,M\}$) caches the secondary file $f_k^*(r)$.
2) **Scheduling for PB:** PB retains highest priority of transmission in the network. At any time slot t it transmits a packet corresponding to the Head-of-Line (HOL) packet request at the queue of highest length (in case of ties, pick one arbitrarily). This packet request is removed from the queue if the transmission is successful. Mathematically, $\mu_{f,0}^{\text{FPCP}}(t) = 1$ if $U_{f,0}^{(p)}(t) > 0$ and $U_{f,0}^{(p)}(t) \geq U_{\bar{f},0}^{(p)}(t)$ for every $\bar{f} \in F^{(p)} - f$ and is 0 otherwise, where $\mu_{f,0}^{X}(t)$ denotes the transmission rate offered to PB under policy X to transmit a packet of file f at time slot t.
3) **Request Admission and Scheduling Policy at Secondary Base-stations:** Each secondary base-station SB_k, $k \in \{1,\ldots,M\}$, admits every request for a cached primary file from a primary user within its transmission-range i.e., for all t, $\chi_{f,k}^{(p)}(t)$ equals 1 if and only if $f \in \{\text{F}_1^{(p)},\ldots,\text{F}_{B-1}^{(p)}\}$. If PB is not transmitting at time slot $t_{r,j}$, then obtain the set of secondary base-stations that are allowed to transmit at $t_{r,j}$, denoted as $E_{\text{FPCP}}^*(t_{r,j})$, by solving the following max-weight problem:

$$E_{\text{FPCP}}^*(t_{r,j}) \quad \in \quad \underset{E\in\tilde{E}}{\text{argmax}} \left(\left(\max\left\{ U_k^{(p)}(t_{r,j}), U_{f_k^*(r),k}^{(s)}(t_{r,j}) \right\} \right)_{k=1}^{M} \right)^T E. \tag{21}$$

Suppose, according to $E_{\text{FPCP}}^*(t_{r,j})$ some SB_k is allowed to transmit at this time slot. Then we determine which packet to transmit from SB_k in the following manner.

We transmit the packet corresponding to the HOL packet request at the queue of secondary file $f_k^*(r)$ if $U^{(s)}_{f_k^*(r),k}(t_{r,j})$ is greater than or equal to $U_k^{(p)}(t)$. Otherwise transmit the packet corresponding to the HOL packet request at the aggregate primary queue in SB_k. If this queue is empty, transmit a dummy packet.
Mathematically, given k'th component of $E^*_{\text{FPCP}}(t_{r,j})$ is 1, $\mu_k^{\text{FPCP},(p)}(t_{r,j}) = 1$, $\mu^{\text{FPCP}}_{f_k^*(r),k}(t_{r,j}) = 0$ if $U_k^{(p)}(t) \geq U^{(s)}_{f_k^*(r),k}(t_{r,j})$; $\mu_k^{\text{FPCP},(p)}(t_{r,j}) = 0$, $\mu^{\text{FPCP}}_{f_k^*(r),k}(t_{r,j}) = 1$ otherwise.

The scheduling policy only uses knowledge of instantaneous queue lengths and not the request generation rates. The secondary base-stations do not serve queued primary packet requests with high priority over secondary ones.

It can be shown that FPCP stabilizes all queues in the network for every request generation rate vector in the achievable capacity region:

Theorem 1: FPCP stabilizes the network of queues for all ($\boldsymbol{\lambda}^{(p)}$, $\boldsymbol{\lambda}^{(s)}$)$\in$ Interior(Λ).

Proof: In order to prove Theorem 1, we first present an alternate policy ALT1 which, in every period, minimizes the conditional drift of the Lyapunov function L_1, among all policies that satisfy constraint **C1**. We then state and prove Lemma 3. In Lemma 3 we compare the values of an utility function that corresponds to the conditional Lyapunov drift of the system under FPCP and ALT1 respectively. We then use this result to compare the values of the aforementioned utility function under FPCP against that of a stationary caching and scheduling policy STAT1; STAT1 stabilizes all queues for every request generation rate vector in the interior of Λ. We complete the proof by applying Theorem 4.1 in [17] that connects stability performance of an algorithm to the conditional drift of a Lyapunov function under the algorithm. The detailed proof is provided in Appendix B. ■

One drawback of the FPCP algorithm is that it allows each secondary base-station to cache only one secondary file in every period. While this policy stabilizes every queue for all request generation rate vectors in the interior of Λ, our simulation results in Section VII show it can adversely affect the delay performance of the secondary users. This is more notable when primary request generation rates are low and it is not necessary to cache $B-1$ primary files at every secondary base-station in each period in order to stabilize the queues. As a result, in the next section we present the VPCP algorithm which dynamically varies the number of cached primary files at each secondary base-station in every period according to instantaneous secondary queue lengths. However, the guaranteed stability region under VPCP is lower than that under FPCP.

V. Variable Primary Caching Policy

In this section we present VPCP, an algorithm under which the network controller dynamically determines the number of primary files to be cached in each period. Furthermore, primary packet requests are satisfied ahead of secondary ones at every secondary base-station. We first provide an overview of the motivation behind construction of this algorithm; in the following subsections we discuss the construction methodology and present a guaranteed stability region under the algorithm.

The VPCP algorithm addresses the difficult problem of serving primary packets at each base-station ahead of secondary ones while also attempting to minimize the number of cached primary files and maintain stability of

all queues. The difficulty of the problem lies in the coupled nature of the evolution of primary and secondary queues in each base-station. In particular, scheduling decisions for secondary packets from any base-station depend on whether or not the primary queues in that base-station are empty. Evolution of the primary queues in turn depend on cooperative caching decisions since any base-station can only transmit packets of a cached primary file. If we view the primary queue lengths as the system state, then the problem essentially is a constrained Markov Decision Problem where the system state itself evolves according to the control decisions. One way to solve such problems efficiently is by the use of renewal frame based optimization techniques. Such techniques can solve certain constrained Markov Decision Problems without suffering from the pitfalls associated with conventional solutions to those problems such as requiring extensive knowledge of system dynamics or large convergence times [9]. Renewal frame based techniques (briefly discussed in Section V-A) are based on partitioning the time-line into distinct collection of time slots called "frames" and then making control decisions in the beginning of each new frame. Length of each frame is variable and depends on the control decisions taken at the beginning of the frame.

However, we cannot use the renewal frame based optimization techniques directly due to some restrictions imposed by the system model. In particular, in the renewal frame based methods, control decisions (e.g. caching) can be made only at the beginning of every variable length frame. On the other hand, according to our system model, caching decisions take place at the beginning of every fixed length cache-refresh period which may not occur at the beginning of a new frame.

In order to address the challenge of applying renewal frame method to this scenario, we construct the VPCP algorithm in two steps. First, as discussed in Section V-B, in each period we estimate, the number of primary files that would be cached at secondary base-stations by a renewal frame based caching and scheduling policy *had* the beginning of the period coincided with beginning of a new frame. This renewal frame based method makes caching decisions only at the beginning of a frame by minimizing the ratio of a drift plus penalty expression over a frame, to the expected length of the frame. Then, as discussed in Section V-C, we use those estimates to construct the VPCP algorithm by first determining the actual number of primary files cached at each secondary base-station in every period and then the scheduling policy. However, because of this two-step construction process, the VPCP algorithm is not an optimal renewal frame based optimization policy i.e. it does not provide any optimality guarantee on the average number of primary files cached at each secondary base-station.

In order to render tractable analysis of its stability performance, the VPCP algorithm is constructed under certain restrictions. First, under this algorithm only a fixed set of non-interfering secondary base-stations, denoted without loss of generality as $\text{SB}_1, \ldots, \text{SB}_G$ respectively (where $G \leq M$), cooperate with the primary network by caching primary files and, admitting and serving primary file requests. The other secondary base-stations $\text{SB}_{G+1}, \ldots, \text{SB}_M$ are referred to as non-cooperative secondary base-stations. Secondly, the policy VPCP is constructed such that it satisfies constraints **C1** (described in Section III) as well as two additional scheduling constraints described in Section V-B. In Section V-D we present a guaranteed stability region for this algorithm.

A. Renewal Frame Techniques for Cognitive Networks

In this subsection we give a brief overview of renewal frame based methods and how it is applied in our scenario.

In renewal frame based optimization methods, the timeline is partitioned into contiguous collection of time slots with each collection referred to as a frame [17]. Each frame is defined in terms of a system state; a new frame begins whenever the system state is refreshed. The length of each frame is variable and is determined by the control decision taken at the beginning of the frame. At the beginning of each frame, a controller selects a policy based on the average rewards collected at previous frames as well as the instantaneous values of some network parameters (such as queue length). The selected policy remains fixed during the entire frame. Rewards are utility function of some network parameter of interest that we want to optimize (such as average power consumption, throughput etc.) subject to some feasibility constraints (such as stability of queues, maximum transmission power etc.). Near-optimal control decisions are obtained by finding policies that minimize the ratio of a Lyapunov drift plus penalty expression, over a frame, to the expected frame-length. The drift term corresponds to the feasibility constraints while the penalty function corresponds to the utility function. Typically, such expressions represent a trade-off between satisfying the constraints versus achieving optimal value, with the extent of the trade-off determined by a constant penalty parameter.

We define the system state to be the sum of the length of primary queues in PB and all secondary base-stations, similar to the way primary user's channel occupancy process was used as system state in [9][5]. Defining the system state in this way is useful because scheduling decision for secondary packets depend on whether this system state is zero or not. The penalty function is the number of cached primary files at each secondary base-station. Each frame begins when the system state changes i.e. the sum of all primary queue lengths transitions from zero to a non-zero value. Fig. 2 shows an example of such a partition. Each frame consists of two distinct phases, one in which at least one of the primary queues in the entire network in non-empty, followed by one in which all primary queues in the network are empty.

B. Caching Decisions by a Renewal Frame Based Optimization Policy

In this subsection we estimate the number of primary files, $\hat{n}_k^*(r)$, that would be cached at each cooperative secondary base-station SB_k (where $k \in \{1, \ldots, G\}$) at the beginning of the r'th period (where $r \in \{1, 2, \ldots, \}$) under a renewal frame based optimization policy. The optimization policy tries to minimize the average number of primary files cached while maintaining stability of all queues. The policy also transmits primary packets with higher priority than secondary ones from each cooperative secondary base-station. The policy assumes a new frame began at time slot $t_{r,1}$, irrespective of the actual length of primary queues, and subsequent caching decisions take place only at the beginning of every frame. In the rest of this sub-section, for convenience of exposition, we assume those assumptions are indeed true and that the r''th frame (where $r' \in \{1, 2, \ldots, \}$) began at time slot $t_{r,1}$. We first obtain expressions for the evolution of secondary queues in each frame. Then we apply the Lyapunov drift

[5]Note that in [9] the authors did not study cooperative caching in cognitive radio networks.

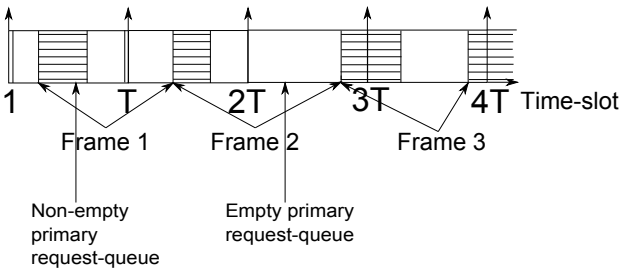


Fig. 2. Partition of time-line into frames. Shaded region shows time slots for which at least one primary queue in the network is non-empty. Each frame consists of one such shaded region followed by period for which all primary queues are empty. An arrow indicates beginning of a new period.

technique to find an upper bound of the conditional drift of a Lyapunov function of secondary queue lengths at the beginning of each frame. Next, we obtain a conditional drift plus penalty expression by adding to the conditional drift term, a penalty function representing the number of primary files cached at every secondary base-station. We then simplify the upper bound of this conditional drift plus penalty expression. Finally, we minimize the ratio of this upper bound to the expected frame length to obtain the $\hat{n}_k^*(r)$ variables. For tractable stability analysis, we solve the minimization problem approximately by considering only caching and scheduling policies that satisfy constraint **C1** and two additional scheduling constraints described later.

We obtain expressions for the evolution of secondary queues over the r''th frame for every $f \in F^{(s)}$ and $l \in \{1, \ldots, M\}$ as

$$U_{f,l}^{(s)}(\tilde{t}_{r'+1,1}) \leq \max\{U_{f,l}^{(s)}(\tilde{t}_{r',1}) - \sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1} \mu_{f,i}(\tau), 0\} + \sum_{i:\text{SU}_i \in \phi_l^{(s)}} \sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1} A_{f,i}^{(s)}(\tau) \tag{22}$$

where $\tilde{t}_{n,j}$ denotes the j'th, $j \in \{1, \ldots, \tilde{t}_{n+1,1} - 1\}$, time slot in the n'th, $n \in \{1, 2, \ldots, \}$, frame. We define $\tilde{Z}_{f,l}^{(s)}(n) \triangleq U_{f,l}^{(s)}(\tilde{t}_{n,1})$ for every $f \in F^{(s)}$, $l \in \{1, \ldots, M\}$ and $n \in \{1, \ldots, \}$.

We define the Lyapunov function of secondary queue lengths at beginning of each frame, $L_2(r') \triangleq \sum_{l=1}^{M} \sum_{f \in F^{(s)}} (U_{f,l}^{(s)}(\tilde{t}_{r',1}))^2$ and its conditional drift as,

$$\Delta_2(r') \triangleq E[L_2(r'+1) - L_2(r') | \tilde{\boldsymbol{Z}}^{(s)}(r')], \tag{23}$$

where $\tilde{\boldsymbol{Z}}^{(s)}(n)$ denotes the vector of secondary queue lengths in all secondary base-stations at the beginning of the n'th frame for every strictly positive integer n. From (22) and (23) we get,

$$\begin{aligned} \Delta_2(r') \quad \leq \quad & E[\sum_{l=1}^{M} \sum_{f \in F^{(s)}} \{(\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1} \sum_{i:\text{SU}_i \in \phi_l^{(s)}} A_{f,i}^{(s)}(\tau))^2 + (\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1} \mu_{f,l}(\tau))^2\} | \tilde{\boldsymbol{Z}}^{(s)}(r')] \\ - \quad & E[\sum_{l=1}^{M} \sum_{f \in F^{(s)}} \tilde{Z}_{f,l}^{(s)}(r') \sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1} \{\mu_{f,l}(\tau) - \sum_{i:\text{SU}_i \in \phi_l^{(s)}} A_{f,i}^{(s)}(\tau)\} | \tilde{\boldsymbol{Z}}^{(s)}(r')]. \end{aligned} \tag{24}$$

Note that, in any slot, the number of arrivals to any secondary queue and the number of packet transmissions by any secondary base-station is upper bounded by some finite positive constant $\tilde{T_2}$. Therefore we have,

$$\begin{aligned}\Delta_2(r') &\leq E[(\tilde{t}_{r'+1,1}-\tilde{t}_{r',1})^2|\tilde{\boldsymbol{Z}}^{(s)}(r')]2M|F^{(s)}|(\tilde{T_2})^2 \\ &- E[\sum_{l=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,l}^{(s)}(r')\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\{\mu_{f,l}(\tau)-\sum_{i:\mathrm{SU}_i\in\phi_l^{(s)}}A_{f,i}^{(s)}(\tau)\}|\tilde{\boldsymbol{Z}}^{(s)}(r')]. \end{aligned} \quad (25)$$

Next, we add to both sides of (25) a penalty term: $VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')]$, where V is a positive constant and $\hat{n}_k(r')$ denotes the number of primary files cached at SB_k in the r''th frame. The penalty parameter V reflects the trade-off between queue stability and number of primary files cached. High V implies less primary files should be cached and vice-versa. We thus obtain,

$$\begin{aligned}\Delta_2(r')+VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')] &\leq VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')] \\ &- E[\sum_{l=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,l}^{(s)}(r')\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\{\mu_{f,l}(\tau)-\sum_{i:\mathrm{SU}_i\in\phi_l^{(s)}}A_{f,i}^{(s)}(\tau)\}|\tilde{\boldsymbol{Z}}^{(s)}(r')] \\ &+ E[(\tilde{t}_{r'+1,1}-\tilde{t}_{r',1})^2|\tilde{\boldsymbol{Z}}^{(s)}(r')]2M|F^{(s)}|(\tilde{T_2})^2. \end{aligned} \quad (26)$$

Now we simplify the RHS of (26). Similar to [9] it can be shown that when $\boldsymbol{\lambda}^{(p)}$ is restricted to the set of primary request generation rate vectors that can stabilized even without cooperation, there exists finite non-zero constants W_0, W_1 and W_2 s.t. $W_0 \geq E[\tilde{t}_{r'+1,1}-\tilde{t}_{r',1}] > W_1$ and $E[(\tilde{t}_{r'+1,1}-\tilde{t}_{r',1})^2] < W_2$ for any policy that transmits queued primary packets with higher priority from each base-station. For this set of $\boldsymbol{\lambda}^{(p)}$ we obtain from (26),

$$\begin{aligned}\Delta_2(r')+VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')] &\leq VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')] \\ &- E[\sum_{l=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,l}^{(s)}(r')\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\{\mu_{f,l}(\tau)-\sum_{i:\mathrm{SU}_i\in\phi_l^{(s)}}A_{f,i}^{(s)}(\tau)\}|\tilde{\boldsymbol{Z}}^{(s)}(r')] \\ &+ 2W_2M|F^{(s)}|(\tilde{T_2})^2. \end{aligned} \quad (27)$$

The renewal frame based policy finds the number of primary files to cache at each base-station in the r''th frame, as well as the scheduling scheme to be used throughout the r''th frame, for which the ratio of the RHS in (27) to that of the expected frame-length is minimized:

$$\frac{VE[\sum_{k=1}^{G}\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\hat{n}_k(r')|\tilde{\boldsymbol{Z}}^{(s)}(r')]-E[\sum_{l=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,l}^{(s)}(r')\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\{\mu_{f,l}(\tau)-\sum_{i:\mathrm{SU}_i\in\phi_l^{(s)}}A_{f,i}^{(s)}(\tau)\}|\tilde{\boldsymbol{Z}}^{(s)}(r')]}{E[(\tilde{t}_{r+1,1}-\tilde{t}_{r,1})|\tilde{\boldsymbol{Z}}^{(s)}(r')]}$$

$$\text{i.e. } V\sum_{k=1}^{G}\hat{n}_k(r')-\frac{E[\sum_{l=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,l}^{(s)}(r')\sum_{\tau=\tilde{t}_{r',1}}^{\tilde{t}_{r'+1,1}-1}\mu_{f,l}(\tau)|\tilde{\boldsymbol{Z}}^{(s)}(r')]}{E[(\tilde{t}_{r+1,1}-\tilde{t}_{r,1})|\tilde{\boldsymbol{Z}}^{(s)}(r')]}. \quad (28)$$

Minimizing the expression in (28) reflects the tradeoff between stability of queues versus caching less primary files at secondary base-stations. Caching more primary files at each secondary base-station minimizes the fraction term in (28), while it increases the value of the penalty term: $V \sum_{k=1}^{G} \hat{n}_k(r')$.

In order to precisely define a guaranteed stability region for the VPCP algorithm (described in the next subsection), we minimize the expression in (28) by considering only caching and scheduling algorithms that satisfy the caching constraint **C1** and two scheduling constraints **C2** and **C3**:

> **C2)** At every time slot PB transmits a packet if and only if it has at least one non-empty primary queue and all cooperative secondary base-stations have only empty primary queues. No cooperative secondary base-station transmits a secondary packet whenever it has at least one non-empty primary queue.
>
> **C3)** No non-cooperative secondary base-station simultaneously transmits secondary packets when some cooperative secondary base-station is transmitting a primary packet.

Constraint **C2** is a priority constraint which ensures that primary packets are transmitted ahead of secondary packets from each cooperative secondary base-station. Furthermore, this constraint is also useful to construct a guaranteed stability region for the VPCP algorithm as it ensures that the events of primary packet transmission from cooperative secondary base-stations are independent from base-station to base-station. Constraint **C3** allows us to construct a guaranteed stability region for VPCP without calculating the probability of the event: "some non-cooperative secondary base-station is transmitting a secondary packet when some cooperative secondary base-station is simultaneously transmitting a primary packet". Calculating the probability of this event is cumbersome.

In Appendix C we find an approximate solution to the problem of minimizing the expression in (28) by considering only policies that satisfy constraint **C1**, **C2**, **C3** and under some additional assumptions. In particular, we obtain an estimate of the number of most popular primary files, $\hat{n}_k^*(r)$, that ought to be cached at each cooperative secondary base-station SB_k in the r'th period if the expression in (28) is minimized.

C. Description of the VPCP Algorithm

In this subsection we use the $\hat{n}_k^*(r)$ variables to construct the VPCP algorithm. First we discuss the problem associated with simply caching $\hat{n}_k^*(r)$ most popular primary files at each cooperative base-station SB_k , $k \in \{1, \ldots, G\}$, in the r'th period. Next, we discuss the caching scheme used in the VPCP algorithm to resolve this problem. Finally, we state the VPCP algorithm in Table 2.

It is problematic to simply cache $\hat{n}_k^*(r)$ most popular primary files at each cooperative base-station SB_k , $k \in \{1, \ldots, G\}$, in the r'th period for every $r \in 1, 2, \ldots$. This is because the $\hat{n}_k^*(r)$ estimates were calculated assuming that the beginning of the r'th period, i.e. time slot $t_{r,1}$, coincided with the beginning of a new frame and without observing the actual length of each primary queue in the network at $t_{r,1}$. Hence, it is possible that at time slot $t_{r,1}$, the queue length in SB_k for some primary file f, not one of the $\hat{n}_k^*(r)$ most popular primary files, is non-zero. Not caching file f in the r'th period can lead to loss of transmission opportunities for SB_k since a cooperative secondary base-station can transmit a secondary packet only if all of its primary queues are empty.

In order to address the above problem of lost transmission opportunities due to lack of cached primary files, in the VPCP algorithm we ensure that in the r'th period, each cooperative secondary base-station caches every primary file corresponding to its non-empty primary queues. We guarantee this by having the network controller cache $n_k^*(r)$ most popular primary files and admit primary file requests at each SB_k, $k \in \{1, \ldots, G\}$, in the r'th period (for every $r \in \{1, \ldots, \}$) as follows:

1) Suppose no new frame began during the $(r-1)$'th period i.e. at least one primary queue in the network was non-empty for all time slots in the $(r-1)$'th period. Then, at time slot $t_{r,1}$, cache at least as many most popular primary files at SB_k as cached in the $(r-1)$'th period i.e., $n_k^*(r) = \max\{n_k^*(r-1), \hat{n}_k^*(r)\}$. For the very first period we have, $n_k^*(1) = \hat{n}_k^*(1)$.
2) Otherwise, suppose a new frame began during $(r-1)$'th period i.e. at some time slot $t_{r-1,j}$ (where $j \in \{1, 2, \ldots, T\}$) all primary queues in the network became empty for the first time in $(r-1)$'th period. Then from time slot $t_{r-1,j+1}$ till the end of the $(r-1)$'th period, SB_k only admits primary file requests corresponding to $\hat{n}_k^*(r-1)$ most popular primary files[6]. By admitting only those requests we ensure that only queues of $\hat{n}_k^*(r-1)$ most popular primary files can be non-empty in SB_k at $t_{r,1}$. At $t_{r,1}$, SB_k caches at least $\hat{n}_k^*(r-1)$ most popular primary files i.e $n_k^*(r) = \max\{\hat{n}_k^*(r-1), \hat{n}_k^*(r)\}$, if at least one primary queue is non-empty at $t_{r,1}$; otherwise, it caches $\hat{n}_k^*(r)$ most popular primary files i.e, $n_k^*(r) = \hat{n}_k^*(r)$.

After obtaining the set of cached primary files, the network controller fills rest of the cache at each secondary base-station with secondary files based on their instantaneous queue lengths.

Table 2 contains the details of the VPCP algorithm. In this algorithm, each cooperative secondary base-station serves queued primary file requests with higher priority than the secondary ones; the algorithm satisfies scheduling constraints **C2** and **C3**. When all primary queues are empty, secondary base-stations are scheduled according to a modified backpressure rule that schedules secondary base-stations based on their instantaneous secondary queue lengths.

Table 2: VPCP Algorithm

For every strictly positive integer r following steps are performed in the r'th period:

1) **Caching Policy:** At the beginning of the period, for every cooperative secondary base-station SB_k, $k \in \{1, \ldots, G\}$, compute $\hat{n}_k^*(r)$ and then $n_k^*(r)$. Each cooperative base-station SB_k caches the $n_k^*(r)$ most popular primary files; in addition, it caches the $B - n_k^*(r)$ secondary files with highest queue lengths in SB_k at $t_{r,1}$. Every non-cooperative secondary base-station SB_l, $l \in \{G+1, \ldots, M\}$, caches the B secondary files with highest queue lengths in SB_l at $t_{r,1}$.
2) **Request Admission and Scheduling Policy for Primary Users:** Suppose, at least one primary queue is

[6]Note that unlike FPCP the network controller may not admit primary file requests at a cooperative secondary base-station even if the requested file is present in its cache.

non-empty at $t_{r,1}$. Consider all time slots in the r'th period until either the end of the period or the first time slot in which all primary queues become non-empty, whichever is earliest. In these time slots each SB_k, $k \in \{1,\ldots,G\}$, admits requests corresponding to $n_k^*(r)$ most popular primary files from the $\phi_k^{(p)}$ primary users. Suppose at time slot t_{r,j^*} (where $1 < j^* \leq T$) all primary queues become empty for the first time. Then from time slots t_{r,j^*} until end of the period, SB_k admits requests corresponding to $\hat{n}_k^*(r)$ most popular primary files from the $\phi_k^{(p)}$ primary users. Mathematically, if such t_{r,j^*} exists then for every $k \in \{1,\ldots,G\}$,

$$\chi_{f,k}^{(p)}(t_{r,j}) = \begin{cases} 1, & \forall f \in \{\text{F}_1^{(p)},\ldots,\text{F}_{n_k^*(r)}^{(p)}\}, t_{r,1} \leq t_{r,j} < t_{r,j^*} \\ 1, & \forall f \in \{\text{F}_1^{(p)},\ldots,\text{F}_{\hat{n}_k^*(r)}^{(p)}\}, t_{r,j^*} \leq t_{r,j} \leq t_{r,T} \\ 0, & \text{else} \end{cases} \tag{29}$$

otherwise,

$$\chi_{f,k}^{(p)}(t_{r,j}) = \begin{cases} 1, & \forall f \in \{\text{F}_1^{(p)},\ldots,\text{F}_{n_k^*(r)}^{(p)}\}, t_{r,1} \leq t_{r,j} \leq t_{r,T} \\ 0, & \text{else.} \end{cases} \tag{30}$$

In any time slot $t_{r,j}$ (where $j \in \{1,\ldots,T\}$), SB_k transmits the packet corresponding to the HOL packet request at its primary queue of highest length; in case of ties, pick one arbitrarily. Mathematically for every $k \in \{1,\ldots,G\}$, $\mu_{f,k}^{\text{VPCP}}(t) = 1$, if $U_{f,k}^{(p)}(t) > 0$, $U_{f,k}^{(p)}(t) \geq U_{\bar{f},k}^{(p)}(t)$ for every $\bar{f} \in F^{(p)} - f$; it equals 0 otherwise.

If the primary queues in all secondary base-stations are empty and the queue in PB is non-empty, then identify the queue of highest length in PB and transmit the packet corresponding to its HOL packet request from PB. Mathematically, $\mu_{f,0}^{\text{VPCP}}(t) = 1$, if $U_{f,0}^{(p)}(t) > 0$, $U_{f,0}^{(p)}(t) \geq U_{\bar{f},0}^{(p)}(t)$ for every $\bar{f} \in F^{(p)} - f$ and $U_{f,k}^{(p)}(t) = 0$ for every $k \in \{1,\ldots,G\}$; it equals 0 otherwise.

3) **Scheduling Policy for Secondary Users:** Suppose, at time slot $t_{r,j}$ all the primary queues of cooperative base-station SB_k are empty but some other cooperative base-station is transmitting a primary packet. Then identify the queue of the cached secondary file with largest instantaneous backlog in SB_k (in case of ties, pick one arbitrarily); SB_k transmits the packet corresponding to the HOL packet request in this queue. In case the queue is empty, SB_k transmits a dummy packet. Mathematically, for every $k \in \{1,\ldots,G\}$, we have $\mu_{f,k}^{\text{VPCP}}(t_{r,j}) = 1$ if $f \in \underset{f_1 \in H_k^{(s)}(t_{r,j})}{\text{argmax}} U_{f_1,k}^{(s)}(t_{r,j})$, $\max\limits_{f_2 \in F^{(p)}} U_{f_2,k}^{(p)}(t_{r,j}) = 0$ and $\max\limits_{f_3 \in F^{(p)}} U_{f_3,l}^{(p)}(t_{r,j}) > 0$ for some $l \in \{1,\ldots,G\}$.

Otherwise, if all the primary queues are empty at time slot $t_{r,j}$, then schedule secondary packet transmission according to a modified backpressure rule. First find the activation vector $E_{\text{VPCP}}^*(t_{r,j})$ as

$$E_{\text{VPCP}}^*(t_{r,j}) \in \underset{E \in \bar{E}}{\text{argmax}} \left(\left(\max_{f \in H_l^{(s)}(t_{r,j})} U_{f,l}^{(s)}(t_{r,j}) \right)_{l=1}^{M} \right)^T E. \tag{31}$$

If base-station SB_l is scheduled to transmit according to $E_{\text{VPCP}}^*(t_{r,j})$, then SB_l transmits a packet corresponding to the HOL packet request in the secondary queue of highest length (in case of ties, pick one arbitrarily) i.e.

$\mu_{f,l}^{\text{VPCP}}(t_{r,j}) = 1$ if $f \in \underset{f_1 \in H_l^{(s)}(t_{r,j})}{\text{argmax}} U_{f_1,l}^{(s)}(t_{r,j})$. In case the queue is empty, transmit a dummy packet.

D. Guaranteed Stability Region

Next, we find a guaranteed stability region for the VPCP algorithm. First we construct this region, denoted as Λ^{VPCP}. Then, in Theorem 2 we show that VPCP algorithm indeed stabilizes all queues for every request generation rate vector within the guaranteed stability region.

The region Λ^{VPCP}, consists of all primary request generation rate vectors $\boldsymbol{\lambda}^{(p)}$ that can be satisfied without cooperation[7] and a set of secondary request generation rate vectors $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$. Mathematically, the region Λ^{VPCP} is defined as the set $\left\{(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)}) : \boldsymbol{\lambda}^{(p)} \in \Lambda_0^{(p)}, \boldsymbol{\lambda}^{(s)} \in \text{ Interior } \left(\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})\right)\right\}$ where $\Lambda_0^{(p)}$ denotes the set of primary request generation rate vectors that can be satisfied even in absence of cooperation. For a given primary request generation rate vector $\boldsymbol{\lambda}^{(p)}$, the set $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$ defines the set of all secondary request generation rate vectors that can be supported under caching and scheduling algorithms that satisfy constraints **C1**, **C2** and **C3**. The exact definition of the region $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$ is provided in Appendix C.

Theorem 2: Under the VPCP algorithm, all queues in the network are stable for all request generation rate vectors $(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)})$ in the set Λ^{VPCP}.

Proof: In the proof we consider two cases of secondary queue lengths at the beginning of any frame: one in which at least one secondary queue in the network is above a certain threshold and one in which all secondary queues are below that threshold.

For the first case we show that, for every $(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)})$ in Λ^{VPCP}, the conditional drift of the Lyapunov function L_2 under VPCP is upper bounded by a finite positive constant minus a weighted sum of secondary queue lengths at the beginning of the frame. In order to show this we compare the conditional drift of the Lyapunov function L_2, over a frame, under VPCP with that of a stationary policy STAT2. The policy STAT2 stabilizes every queue in the network for all request generation rate vectors in Λ^{VPCP}. The comparison consists of two steps. First we compare the drift under VPCP with that of an alternate algorithm ALT2 that minimizes the conditional drift term over a frame. Then we compare the drift under ALT2 with that under STAT2.

For the second case we show that the conditional drift of L_2 under the VPCP algorithm is upper bounded by a finite positive constant. We complete the proof by combining both cases and applying Theorem 4.1 in [17] that connects stability performance of an algorithm to the conditional drift of a Lyapunov function under the algorithm. The detailed proof is provided in Appendix D. ■

Clearly, the guaranteed stability region is greater than the capacity region without cooperation but is smaller than the achievable capacity region Λ. However, with respect to request generation rates from primary users, the guaranteed stability region under the VPCP algorithm remains the same as for the case without cooperation.

[7]Recall, in Section V-B while deriving the upper bound to the drift plus penalty expression in (27) we considered only those $\boldsymbol{\lambda}^{(p)}$ for which all queues are stable without cooperation. This allowed us to discard the contribution of the term $E[(\tilde{t}_{r'+1,1} - \tilde{t}_{r',1})^2]2M|F^{(s)}|(\tilde{T}_2)^2$ while minimizing the upper bound in (28) as for those $\boldsymbol{\lambda}^{(p)}$ vectors, this term is upper bounded by the constant $2W_2M|F^{(s)}|(\tilde{T}_2)^2$.

VI. Network with Multiple Channels

In this section we extend our analysis to a network where all base-stations can transmit on multiple orthogonal channels. For simplicity we consider the case where all secondary base-stations are non-interfering. First in Section VI-A we present the system model. Then in Section VI-B we describe an achievable capacity region for this model. In section VI-C we present the MCCP algorithm that stabilizes all queues in the network for all request generation rate vectors within the guaranteed stability region.

A. System Model

There are J orthogonal channels denoted as $c_1, \ldots, c_J$. Every base-station also has J antennas and can transmit on each channel simultaneously. Each user however has a single antenna and can only communicate with one base-station on a single channel at any given time slot. We assume the number of primary and secondary users in each small-cell is greater than the number of channels. Next, we present details about the transmission model, interference model, caching and scheduling model.

1) Transmission Model: All channels have identical quality. For each channel, the transmission model is same as that of the single channel network. Primary users are non-cognitive and do not have the capability to switch channels using software-defined radio. Hence, each primary user is associated with a unique channel i.e. it receives packets in all time slots only on this channel. Let γ_j denotes the set of primary users associated with channel c_j (where $\gamma_j \subseteq \{\text{PU}_1, \ldots, \text{PU}_{N^{(p)}}\}$ and $j \in \{1, \ldots, J\}$). However, each secondary user can receive packets from secondary base-stations on different channels in different time slots.

2) Interference Model: The interference model is slightly different from that in the single channel case as each base-station may transmit to multiple users at any time slot (on different channels) but each user can receive transmission on only one channel. Consequently, we define a link between base-station a (where $a \in \{\text{PB}, \text{SB}_1, \ldots, \text{SB}_M\}$) and user b (where $b \in \{\text{PU}_1, \ldots, \text{PU}_{N^{(p)}}, \text{SU}_1, \ldots, \text{SU}_{N^{(s)}}\}$), located within a's transmission range, on channel c (where $c \in \{c_1, \ldots, c_J\}$) as the 3-tuple (a, b, c). Note that since primary users can communicate on a fixed unique channel, the 3-tuple (a, b, c) is not a feasible link for a primary user $b \in \gamma_j$ (where $j \in \{1, \ldots, J\}$) if channel c is not c_j. A link (a, b, c) is said to be active if a transmits to b on channel c; otherwise it is inactive. The link (a, b, c) is active iff:

1) Base-station a does not transmit to user b on another channel i.e., $(a, b, \bar{c})$ is inactive for every $\bar{c} \in \{c_1, \ldots, c_J\} - c$.
2) Base-station a does not transmit to any other user on the same channel i.e., $(a, \bar{b}, c)$ is inactive for every user $\bar{b} \in \{\text{PU}_1, \ldots, \text{PU}_{N^{(p)}}, \text{SU}_1, \ldots, \text{SU}_{N^{(s)}}\} - b$ within a's transmission range.
3) If a is a secondary base-station, then PB is not transmitting on channel c.

We slight abuse of notation we use E to represent a feasible link activation vector whose every component corresponds to a unique link. We denote the set of all feasible activation vectors as $\tilde{E}$.

3) Service Discipline for Primary Users: The network controller queues user requests at either a secondary base-station or at the primary base-station in the same way as in the single channel model described in Section II. Different from the single-channel case, each base-station maintains queues for every (primary user, primary file) pair corresponding to each user within its transmission range. This is because, unlike in the single channel case, a base-station can use multiple antennas to transmit packets, possibly of the same file, to more than one user at a given time slot.

We denote the length of the queue at time slot t for the pair (PU_i, f) in SB_k, where $\text{PU}_i \in \phi_k^{(p)}$, $k \in \{1, \ldots, M\}$ and $f \in F^{(p)}$, as $U_{f,k,i}^{(p)}(t)$. We denote transmission rate offered to SB_k to transmit packet from this queue at time slot t as $\mu_{f,k,i}(t)$ where $\mu_{f,k,i}(t) \in \{0, 1\}$. Note that $\mu_{f,k,i}(t)$ equals 1 iff at t the link $(\text{SB}_k, \text{PU}_i, c_j)$ is active for some $j \in \{1, \ldots, J\}$ and $\mu_{\bar{f},k,i}(t) = 0$ for every primary file $\bar{f} \neq f$. The queue evolves as:

$$U_{f,k,i}^{(p)}(t+1) = \max\{U_{f,k,i}^{(p)}(t) - \mu_{f,k,i}(t), 0\} + \chi_{f,k}^{(p)}(t) A_{f,i}^{(p)}(t). \tag{32}$$

Similarly, we denote the length of queue at time slot t for the pair (PU_i, f) (where $i \in \{1, \ldots, N^{(p)}\}$ and $f \in F^{(p)}$) in PB as $U_{f,0,i}^{(p)}(t)$ and the transmission rate offered to PB to transmit packet from this queue at time slot t as $\mu_{f,0,i}(t)$ where $\mu_{f,0,i}(t) \in \{0, 1\}$. This queue evolves similar to the single channel case as:

$$\begin{aligned} U_{f,0,i}^{(p)}(t+1) &= U_{f,0,i}^{(p)}(t) - \mu_{f,0,i}(t) I_{\text{PB},i}(t) + \left(A_{f,i}^{(p)}(t) - \sum_{k:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \right) + \sum_{k:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \left(1 - \hat{I}_{f,k}(t)\right) \\ &+ \sum_{k:\text{PU}_i \in \phi_k^{(p)}} \hat{I}_{f,k}(t) A_{f,i}^{(p)}(t) \left(1 - \chi_{f,k}^{(p)}(t)\right) \end{aligned} \tag{33}$$

where the indicator variable $I_{\text{PB},i}(t) \in \{0, 1\}$ denotes whether there was a successful packet transmission from PB to PU_i at time slot t or not; it equals 1 if the transmission was successful and zero otherwise. The term $\left(A_{f,i}^{(p)}(t) - \sum_{k:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \right)$ represents request arrivals to the queue of pair (PU_i, f) in PB when PU_i is not within any small-cell. The term $\sum_{k:\text{PU}_i \in \phi_k^{(p)}} A_{f,i}^{(p)}(t) \left(1 - \hat{I}_{f,k}(t)\right)$ represents request arrivals to the queue of pair (PU_i, f) in PB when the primary file f is not present, at time slot t, in the cache of the secondary base-station containing PU_i within its transmission range. The term $\sum_{k:\text{PU}_i \in \phi_k^{(p)}} \hat{I}_{f,k}(t) A_{f,i}^{(p)}(t) \left(1 - \chi_{f,k}^{(p)}(t)\right)$ represents request arrivals to the queue of pair (PU_i, f) in PB when requests for the cached primary file f is not admitted, at time slot t, by the secondary base-station containing PU_i within its transmission range.

4) Service Discipline for Secondary Users: The network controller queues secondary user requests at a secondary base-station in the same way as in the single channel model. However, we maintain separate queues for every (secondary user, secondary file) pair. We denote the length of the queue at time slot t for the pair (SU_i, f) in SB_k (where $\text{SU}_i \in \phi_k^{(s)}$, $k \in \{1, \ldots, M\}$ and $f \in F^{(s)}$) as $U_{f,k,i}^{(s)}(t)$. As in the primary user case, we denote transmission rate offered to SB_k to transmit packet from this queue at time slot t as $\mu_{f,k,i}(t)$ where $\mu_{f,k,i}(t) \in \{0, 1\}$; $\mu_{f,k,i}(t)$ equals 1 iff at time slot t the link $(\text{SB}_k, \text{SU}_i, c_j)$ is active for some $j \in \{1, \ldots, J\}$ and $\mu_{\bar{f},k,i}(t) = 0$ for every

secondary file $\bar{f} \neq f$. The queue evolves as:

$$U^{(s)}_{f,k,i}(t+1) = \max\{U^{(s)}_{f,k,i}(t) - \mu_{f,k,i}(t), 0\} + A^{(s)}_{f,i}(t). \tag{34}$$

5) Caching Model: The caching model is same as that of the single channel case.

B. Achievable Capacity Region

The achievable capacity region is the set of primary and secondary request generation rate vectors for which all queues in the network are stable under any caching and scheduling policy that satisfies constraint **C1**. It is a generalization of the achievable capacity region for the single channel system model in section III and is obtained similarly by establishing stability constraints at primary and secondary base stations. With slight abuse of notation we denote the achievable capacity region for multiple channels as Λ as well. We describe the construction of this region in Appendix E.

C. Description of the MCCP Algorithm

In this section we propose MCCP, an algorithm under which primary packet transmissions from a secondary base-station on any channel have higher priority over secondary packet transmissions on that channel. In other words, a secondary base-station transmits secondary packets on a given channel only if it has no queued primary packet requests yet to be served via that channel. The algorithm is constructed using Lyapunov drift techniques similar to the FPCP algorithm. In order to construct the algorithm we first obtain expressions representing evolution of secondary queues in each period. Following Lemma 1, in the construction of the MVPCP algorithm we consider only policies in which each secondary base-station caches exactly the $B-1$ most popular primary files in each period. Then we apply the Lyapunov drift technique to find an upper bound of the conditional drift of a Lyapunov function of secondary queue lengths at the beginning of each period. We then minimize this upper bound to find the set of secondary files to be cached at the secondary base-stations in each period. We minimize the upper bound by considering only policies that cache $B-1$ most popular primary files at every secondary base-station in each period and an additional scheduling constraint. We formally state the algorithm in Table 3 and analyze its stability performance in Theorem 3.

First we obtain expressions for the evolution of secondary queues over any period. The secondary queues in each secondary base-station SB_k evolve over the r'th period (where $r \in \{1, 2, \dots, \}$) under a caching and scheduling policy X, for every $k \in \{1, \dots, M\}$ as

$$U^{(s)}_{f,k,i}(t_{r+1,1}) \leq \max\left\{U^{(s)}_{f,k,i}(t_{r,1}) - \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} \mu^{X}_{f,k,i}(\tau), 0\right\} + \sum_{\tau=t_{r,1}}^{t_{r+1,1}-1} A^{(s)}_{f,i}(\tau) \quad \forall f \in F^{(s)} \text{ and } \text{SU}_i \in \phi^{(s)}_k, \tag{35}$$

where $\mu^{X}_{f,k,i}(\tau)$ denotes the rate offered to SB_k to transmit a packet, under policy X, from the queue corresponding to the pair (f,i) where $f \in F^{(s)}$ and $\text{SU}_i \in \phi^{(s)}_k$. Now similar to Lemma 1 we can show that even for the multichannel network, the achievable capacity region Λ can be described by only considering those policies under which, in every period, each secondary base-station caches exactly the $B-1$ most popular primary files and always

admits requests corresponding to those files. Therefore, in the construction of the MCCP algorithm we consider only those policies.

Next, we apply the Lyapunov drift technique. In particular, we consider the Lyapunov function of queue lengths at the beginning of each period, $L_3(r) \stackrel{\triangle}{=} \sum_{k=1}^{M} \sum_{i:\mathrm{SU}_i \in \phi_k^{(s)}} \sum_{f \in F^{(s)}} (Z_{f,k,i}^{(s)}(r))^2$ where ,

$$Z_{f,k,i}^{(s)}(r) \quad \stackrel{\triangle}{=} \quad U_{f,k,i}^{(s)}(t_{r,1}) \quad \forall f \in F^{(s)}, k \in \{1,\ldots,M\}, \mathrm{SU}_i \in \phi_k^{(s)}. \tag{36}$$

We define its conditional drift as,

$$\Delta_3(r) \stackrel{\triangle}{=} E[L_3(r+1) - L_3(r) | \boldsymbol{Z}^{(s)}(r)], \tag{37}$$

where $\boldsymbol{Z}^{(s)}(r)$ denotes the vector of secondary queue lengths in all secondary base-stations at the beginning of the r'th period.

Since the number of arrivals to each queue and the number of transmissions from each base-station in every period is upper bounded by a finite positive constant $\hat{T}_3$, therefore from (37) we have,

$$\Delta_3(r) \leq 2(\hat{T}_3)^2 M |F^{(s)}| N^{(s)} - 2 \sum_{k=1}^{M} \sum_{i:\mathrm{SU}_i \in \phi_k^{(s)}} \sum_{f \in F^{(s)}} E[Z_{f,k,i}^{(s)}(r) \sum_{\tau=t_{r,1}}^{t_{r,1}+T-1} \{\mu_{f,k,i}^{X}(\tau) - A_{f,k,i}^{(s)}(\tau)\} | \boldsymbol{Z}^{(s)}(r)]. \tag{38}$$

In the MCCP algorithm we determine the set of secondary files cached in the r'th period by finding a policy X that minimizes the RHS of (38). In the minimization problem, we only consider those policies X in which each secondary base-station caches $B-1$ most popular primary files in each period and admits requests for all those files. Therefore, for every strictly positive integer r, in the r'th period each secondary base-station SB_k (where $k \in \{1,\ldots,M\}$) can cache only one secondary file, denoted with slight abuse of notation as $f_k^*(r)$. Furthermore, in order to transmit primary packets with high priority from both PB as well as from secondary base-stations, we only consider policies X that satisfy the following scheduling constraint:

C4) PB transmits packet on any channel whenever it has non-empty primary queue for users associated with that channel. A secondary base-station does not transmit a secondary packet on any channel if it has non-empty queues for primary users associated with that channel.

A simplified expression for $f_k^*(r)$, derived by minimizing the RHS of (38), is provided in Appendix E.

Table 3: MCCP Algorithm

Following steps are performed in the r'th period for every strictly positive integer r:

1) **Caching Policy:** Each secondary base-station SB_k (where $k \in \{1,\ldots,M\}$) caches $B-1$ most popular primary files in every period and admits every request for a cached primary file from a primary user within its transmission-range i.e., for all $t \in \{1,2,\ldots,\}$, $\chi_{f,k}^{(p)}(t)$ equals 1 if and only if $f \in \{\mathrm{F}_1^{(p)},\ldots,\mathrm{F}_{B-1}^{(p)}\}$. At SB_k cache the secondary file $f_k^*(r)$ in r'th period.
2) **Scheduling Policy for Primary Users:** At any time slot t and for each $j \in \{1,\ldots,J\}$ find the queue in PB with highest queue length, among all primary queues corresponding to users associated with channel c_j. PB

transmits a packet corresponding to the HOL packet request from this queue on channel c_j. Mathematically, for some $\text{PU}_i \in \gamma_j$, $\mu_{f,0,i}(t) = 1$ if $U^{(p)}_{f,0,i}(t) > 0$ and $U^{(p)}_{f,0,i}(t) \geq U^{(p)}_{\bar{f},0,\bar{i}}(t)$ for every $\bar{f} \in F^{(p)} - f$ and $\text{PU}_{\bar{i}} \in \gamma_j - \text{PU}_i$; it is 0 otherwise.

If all such queues corresponding to channel c_j at PB are empty but that at some secondary base-station SB_k is non-empty, then SB_k transmits a packet on channel c_j. This transmission corresponds to the HOL packet request at the queue of highest length in SB_k, among all primary users associated with channel c_j. Mathematically, for some $\text{PU}_i \in \gamma_j$, $\mu_{f,k,i}(t) = 1$ if $U^{(p)}_{f,k,i}(t) > 0$ and $U^{(p)}_{f,k,i}(t) \geq U^{(p)}_{\bar{f},k,\bar{i}}(t)$ for every $\bar{f} \in F^{(s)} - f$ and $\text{PU}_{\bar{i}} \in \gamma_j - \text{PU}_i$; it is 0 otherwise.

3) **Scheduling Policy for Secondary Users:** At any given time slot τ in the r'th period, every secondary base-station SB_k (where $k \in \{1, \ldots, M\}$) transmits secondary packets on those channels in which there is no primary packet transmission by either PB or SB_k. Starting from c_1 to c_J identify the m'th such available channel (where $m \in \{1, \ldots, J\}$). On this channel, SB_k transmits a packet corresponding to the HOL packet request in the m'th longest queue of the cached secondary file $f^*_k(r)$ i.e., from the queue associated with pair $(\text{SU}_{\theta_{f^*_k(r),k,m}(\tau)}, f^*_k(r))$. The term $\theta_{f,k,m}(\tau)$, where $\theta_{f,k,m}(\tau) \in \{1, \ldots, N^{(s)}\}$, denotes the index of the secondary user with m'th highest queue length among all queues in SB_k associated with secondary file $f \in F^{(s)}$ at time slot τ.

For this network the MCCP algorithm stabilizes all queues in the network for every request generation rate vectors within the achievable capacity region.

Theorem 3: MCCP stabilizes the network of queues for all $(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)}) \in \text{Interior}(\Lambda)$.

Proof: We only provide an outline of the proof as it is similar to the proof of Theorem 1. First we observe that, there exists a caching and scheduling policy STAT3 in which each secondary base-station caches popular primary files and transmits primary packets in the same way as in MCCP. However under STAT3 each secondary base-station caches secondary files and schedules transmission of secondary packets according to a stationary distribution independent of secondary queue lengths. Moreover, the network is stable for all request generation rate vectors in Λ. Note that secondary packets transmission opportunities are same under both policies. We complete the proof by comparing the conditional drift of the Lyapunov function L_3 under both MCCP and STAT3 and then applying Theorem 4.1 in [17]. ■

VII. Simulation Results

In this section we observe the performance of the FPCP and VPCP algorithms through simulations in C-programming language for a single channel network. We consider a network with 3 small secondary base-stations that are non-interfering to each other. We assume there is one primary and secondary user in each of these 3 small cells; there is no other primary user in the network. We consider symmetric request generation: request generation rates by primary (and secondary) users in each small cell are equal. For convenience, we denote the sum of request generation rates of all primary users in the entire network as $\lambda^{(p)}$ and the request generation rates

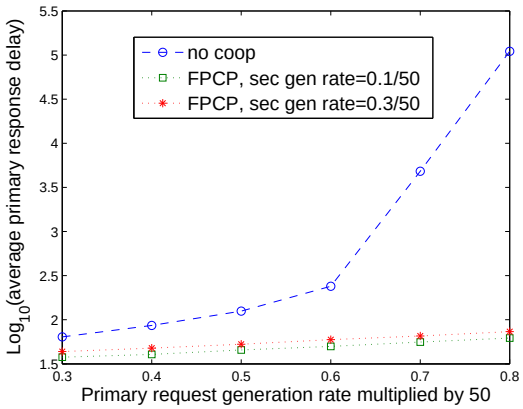


Fig. 3. Plot of base-10 logarithms of time-averaged primary response delay versus net primary request generation rate $50\lambda^{(p)}$, under no cooperation and the FPCP algorithm when $\lambda^{(s)}$ is $\frac{0.1}{50}$, $\frac{0.3}{50}$.

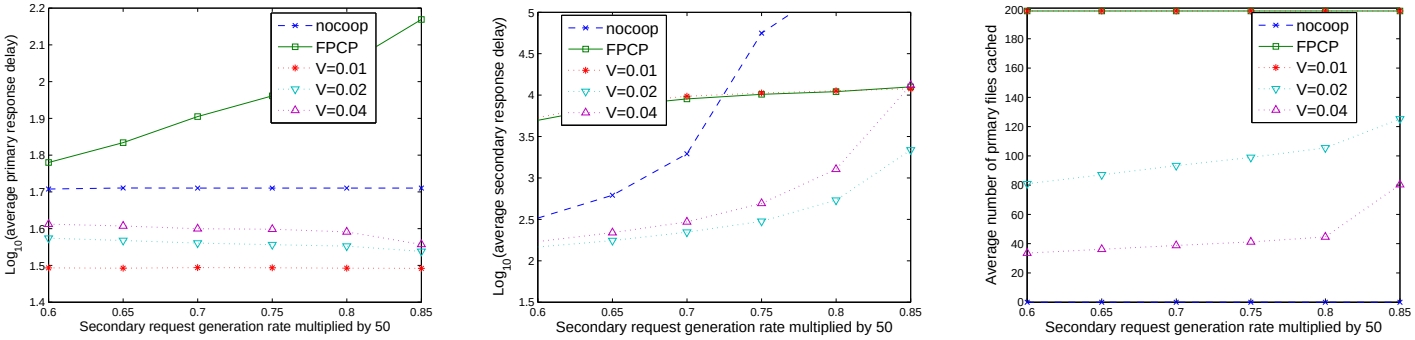
of every secondary user as $\lambda^{(s)}$ respectively. We use the following parameters: cache size (B) is 200, number of successful packet transmissions corresponding to a given file transmission (C) is 50, caching period (T) is 100, probability of successful transmission by primary base-station (p) is 0.7. Total number of primary and secondary files i.e., $|F^{(p)}|$ and $|F^{(s)}|$ respectively, are both equal to 400. Popularity of primary and secondary files have a Zipf distribution [8] with parameter 0.8. All simulations are run for 2,000,000 time slots.

For comparison we use a non-cooperative protocol similar to FPCP wherein all primary user requests are served by the primary base-station (PB). The algorithm is described as follows. At every time slot, PB transmits a packet corresponding to the queue in PB of highest length. A secondary base-station transmits only if all queues in PB are empty. Every secondary base-station caches B files with highest queue lengths at the beginning of every cache refresh period. In every time slot it transmits a packet corresponding to the queue of highest length, among the set of all queues whose files are currently cached in that base-station.

In Fig. 3 we compare the base-10 logarithm of time-averaged *response delay* for primary packets under both FPCP and the non-cooperative algorithm for different values of primary user request generation rates[9]. The response delay for every transmitted packet is measured as the time between generation of the corresponding file request by a user and the time slot when the packet is successfully transmitted to that user. Average primary (respectively secondary) response delay is obtained by averaging response delays of all primary (resp. secondary) packets that are transmitted during the simulation run-time. Plotting base-10 logarithm values allow us to view relatively high values of the time-averaged delay along with smaller values. Two values of $\lambda^{(s)}$: $\frac{0.1}{50}$ and $\frac{0.3}{50}$ are used; for each value of $\lambda^{(s)}$, the value of $\lambda^{(p)}$ is varied from $\frac{0.1}{50}$ to $\frac{0.8}{50}$. Note that maximum $\lambda^{(p)}$ that can be satisfied without

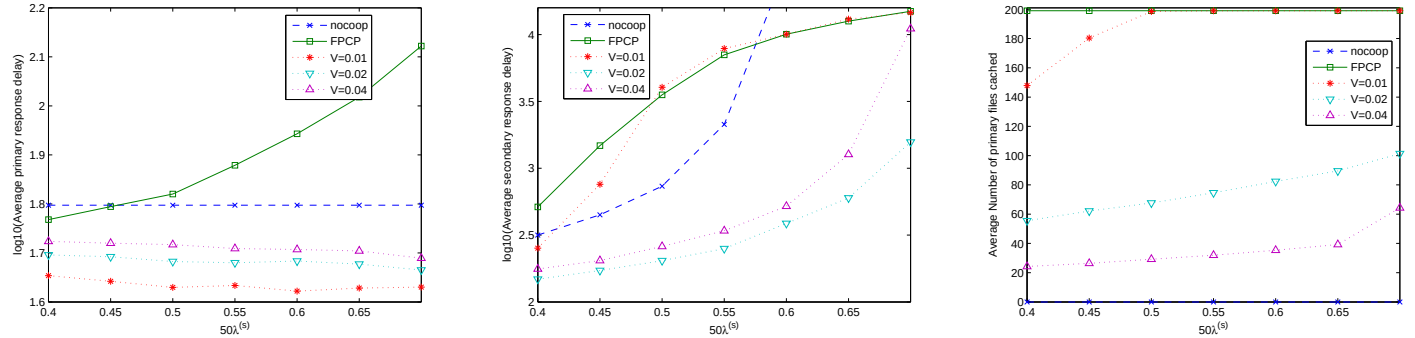
[8]Typically in works on caching, Zipf distribution is used to model the popularity distribution of files.

[9]Note that response delay is directly proportional to length of queues, by Little's law. Lower queue length implies lower response delay and vice-versa.



(a) Average primary response delay (b) Average secondary response delay (c) Average number of primary files cached

Fig. 4. Plot of base-10 logarithms of time-averaged primary and secondary response delay and the average number of primary files cached versus $50\lambda^{(s)}$ when $\lambda^{(p)}$ is $\frac{0.2}{50}$ under no cooperation, FPCP algorithm, and the VPCP algorithm for parameter V=0.01, 0.02 and 0.04.



(a) Average primary system delay (b) Average secondary system delay (c) Average number of primary files cached

Fig. 5. Plot of base-10 logarithms of time-averaged primary and secondary system delay and the average number of primary files cached versus $50\lambda^{(s)}$ when $\lambda^{(p)}$ is $\frac{0.2}{50}$ under no cooperation, FPCP algorithm, and the VPCP algorithm for parameter V=0.01, 0.02 and 0.04. A primary user located outside all small-cells request content at rate $\frac{0.1}{50}$.

cooperation is $\frac{0.7}{50}$; as a result, average delay is very high without cooperation for $\lambda^{(p)} = \frac{0.7}{50}$ and $\frac{0.8}{50}$. From Fig. 3 we observe that for these primary user request generation rates, average primary response delay is lowered under FPCP since the system is stable under FPCP. The difference in average delays is more pronounced as $\lambda^{(p)}$ increases or $\lambda^{(s)}$ decreases.

In Fig. 4 we plot base-10 logarithm of time-averaged primary and secondary response delay and the average number of primary files cached versus $\lambda^{(s)}$ when $\lambda^{(p)}$ is $\frac{0.2}{50}$. We plot these values for the case without cooperation, under the FPCP algorithm, and under the VPCP algorithm with different values of the penalty parameter V. From Fig. 4a we observe the primary users do not benefit under the FPCP algorithm as compared to the case of no cooperation. This is due to relatively high request generation rates by secondary users as compared to $\lambda^{(p)}$ because of which primary user requests are not served very often under the FPCP algorithm. However, under the VPCP

algorithm, average primary response delay is small as primary user requests are served with higher priority by secondary users over secondary user requests. Further, this delay improves with higher $\lambda^{(s)}$. This is because with higher $\lambda^{(s)}$ there is higher backlog of secondary user requests queued at each secondary base-station. As a result, each secondary base-station caches more primary files with increasing $\lambda^{(s)}$.

We observe from Fig. 4b that when $50\lambda^{(s)}$ is less than 0.75, average secondary response delay decreases under cooperation when FPCP or VPCP with penalty parameter $V = 0.01$ are used. This is because, as observed from Fig. 4c under both these algorithms 199 out of every 200 cache positions at each secondary base-station are filled by primary files[10]. This in turn reduces opportunities for secondary base-stations to satisfy multiple secondary file requests in a period. For higher $\lambda^{(s)}$ however, the system is unstable without cooperation and both these algorithms perform better than the case without cooperation. Comparing the average secondary response delay under VPCP with $V = 0.01$ and that under FPCP also shows that, it is not guaranteed that the average secondary response delay is lowered under VPCP for any value of V. However, for some choices of V (for example $V = 0.02$ in Fig. 4) the VPCP algorithm can lower average secondary response delay by striking a balance between the number of primary files cached versus system stability. We observe that, for this network, VPCP with $V = 0.02$ is beneficial for both primary and secondary users in terms of average response delay. For this value of V, the gain in secondary packet transmission opportunities in any period (compared to FPCP or VPCP with $V = 0.01$) due to higher number of cached secondary files, offsets the reduction in secondary packet transmission opportunities due to higher proportion of primary packets being transmitted by PB.

In Fig. 5 we study the behavior of the above algorithms when there are primary users which are outside all small-cells. We consider the network parameter used in Fig. 4; in addition, we consider one primary user, not within any small-cell, which requests contents at rate $\frac{0.1}{50}$. We observe similar results as in Fig. 4. In this case, VPCP with $V = 0.02$ outperforms FPCP and VPCP with V=0.01 and 0.04. In Fig. 4b and 5b, we do not plot average secondary response delay for the case without cooperation when $50\lambda^{(s)}$ is greater than 0.75 and 0.55 respectively because the secondary queues are unstable for those parameters.

VIII. Conclusion

In this work we studied cooperative caching in cognitive radio networks. Using Lyapunov drift techniques we proposed a caching and scheduling algorithm FPCP that increases the set of primary and secondary request generation rates that can be supported. We also proposed an alternative algorithm VPCP whereby secondary base-stations serve primary files requests with higher priority. In future we will extend this analysis to more general settings such as multiple secondary base-stations per primary user and multiple channels. Another interesting avenue of research is to find efficient values for the penalty parameter V in VPCP.

[10]Recall, lower V implies higher number of primary files to be cached with exactly 199 primary files being cached when V is 0.

APPENDIX A

A. Proof of Lemma 1

We first show that it is sufficient, in description of Λ, to consider only those $\mathbf{D}^{(p)} \in \tilde{D}^{(p)}$ which has $B-1$ non-zero components in each column. Consider the set of primary availability matrices in which at least at one secondary base-station, less than $B-1$ primary files are cached. Let $\boldsymbol{D}'^{(p)}$ denote one such matrix in which there exists at least one column j with $\sum_{n=1}^{|F^{(p)}|} D'^{(p)}_{n,j} = B' < B-1$. Suppose $q_{\boldsymbol{D}'^{(p)}}$ is non-zero for some vector $\boldsymbol{q}$ that describes Λ. Then we can construct a new matrix $\boldsymbol{D}''^{(p)}$ by setting one of the zero components in j'th column of $\boldsymbol{D}'^{(p)}$, denoted as $\boldsymbol{D}'^{(p)}_{i,j}$, to 1. Then replacing $\boldsymbol{D}'^{(p)}$ with $\boldsymbol{D}''^{(p)}$ in (6)-(7) we observe that the L.H.S of (6) increases by $q_{\boldsymbol{D}'^{(p)}} \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} P_i^{(p)}$ while the R.H.S of (6) increases by $q_{\boldsymbol{D}'^{(p)}} \frac{\sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} P_i^{(p)}}{p}$. Therefore, (6) is still satisfied by replacing $\boldsymbol{D}'^{(p)}$ with $\boldsymbol{D}''^{(p)}$. Similarly, by extending $\boldsymbol{D}''^{(p)}$ until there are $B-1$ non-zero elements in each column of the extended matrix, we can show Λ can be obtained by considering only those primary availability matrices in $\tilde{D}^{(p)}$ with $B-1$ non-zero components in each column.

Next, we assume there is some $\boldsymbol{D}''^{(p)}$ which has $B-1$ non-zero components in each column but at least one component in $D''^{(p)}_{1,j}$,...,$D''^{(p)}_{B-1,j}$ is zero for some j. Suppose $q_{\boldsymbol{D}''^{(p)}}$ is non-zero for some vector $\boldsymbol{q}$ that describes Λ.. Therefore, by replacing $\boldsymbol{D}''^{(p)}$ with $\boldsymbol{D}^*$ in (6)-(7) we once again observe the L.H.S of (6) increases by less amount than the R.H.S. Hence, the region Λ can be achieved by replacing $\boldsymbol{D}''^{(p)}$ with $\boldsymbol{D}^*$. This proves the Lemma.

B. Proof of Lemma 2

1) Expression for $\lambda^{(s)}_{j,max,ach}(\boldsymbol{\lambda}^{(p)})$: We first find the expression of $\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)})$. When secondary base-stations are non-interfering, the stability constraint for secondary base-stations (6) can be decoupled and re-written for every $j \in \{1,\ldots,M\}$ as

$$\begin{aligned}\sum_{i:\text{SU}_i \in \phi_j^{(s)}} \lambda_i^{(s)} + \sum_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \sum_{l=1}^{|F^{(p)}|} P_l^{(p)} \lambda_k^{(p)} \mathbf{D}^{(p)}_{l,j} q_{\mathbf{D}^{(p)}} \quad &\leq \quad \frac{1}{C} - \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)}}{p} \\ &+ \quad \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} \sum_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} \sum_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(k, \mathrm{F}_l^{(p)} | \mathbf{D}^{(p)}) q_{\mathbf{D}^{(p)}}}{p}.\end{aligned} \tag{39}$$

Using Lemma 1 and (39) we obtain

$$\sum_{i:\text{SU}_i \in \phi_j^{(s)}} \lambda_i^{(s)} \quad \leq \quad \frac{1}{C} + (\frac{1}{p} - 1) \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} \sum_{l=1}^{B-1} P_l^{(p)} + (\frac{1}{p}) \sum_{\substack{n=1 \\ n \neq j}}^{M} \sum_{k:\text{PU}_k \in \phi_n^{(p)}} \lambda_k^{(p)} \sum_{l=1}^{B-1} P_l^{(p)} - \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)}}{p}. \tag{40}$$

Therefore,

$$\lambda^{(s)}_{j,\text{max,ach}}(\boldsymbol{\lambda}^{(p)}) \;=\; \frac{1}{C}+(\frac{1}{p}-1)\sum_{k:\text{PU}_k\in\phi_j^{(p)}}\lambda_k^{(p)}\sum_{l=1}^{B-1}P_l^{(p)}+(\frac{1}{p})\sum_{\substack{n=1\\n\neq j}}^{M}\sum_{k:\text{PU}_k\in\phi_n^{(p)}}\lambda_k^{(p)}\sum_{l=1}^{B-1}P_l^{(p)}-\frac{\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}}{p}. \quad (41)$$

2) Upper Bound of $\lambda^{(s)}_{j,max,ach}(\boldsymbol{\lambda}^{(p)})$: First we find upper bound of feasible secondary packet transmission rates for every secondary base-station. This upper bound is parametrized in terms of a probability distribution vector representing the probability with which each primary availability matrix is used in each period. The set of feasible primary availability matrices, denoted as $\hat{D}^{(p)}$ (where $\hat{D}^{(p)}\supseteq\tilde{D}^{(p)}$), are all those matrices in which sum of every column is less than or equal to B. Then we obtain a simpler expression for the upper bound of feasible secondary packet transmission rates by considering a smaller set of policies. Namely, we consider only policies which cache either B or $B-1$ most popular primary files at every secondary base-station in each period.

We obtain an upper bound of the feasible transmission rates for secondary base-stations as follows. Consider the vector $\boldsymbol{q}'=\left(q'_{\mathbf{D}^{(p)}}\right)$ where the term $q'_{\mathbf{D}^{(p)}}$ denotes the iid probability with which the primary availability matrix $\mathbf{D}^{(p)}\in\hat{D}^{(p)}$ is selected in each period. For every $\mathbf{D}^{(p)}$ the term $q'_{\mathbf{D}^{(p)}}$ should be non-negative and less than 1 (i.e. $0\leq q'_{\mathbf{D}^{(p)}}\leq 1$) and the sum of all $q'_{\mathbf{D}^{(p)}}$ terms should equal 1 (i.e. $\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}q'_{\mathbf{D}^{(p)}}=1$). Given this vector $\boldsymbol{q}'$ the rate of primary packet transmissions by SB_j in a stable system equals $C\sum_{k:\text{PU}_k\in\phi_j^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}$. Then, the rate of primary packet transmissions by all secondary base-stations equals $C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}$. For a stable system the rate of primary packet transmissions by PB is the total rate of primary packet requests minus the rate of primary packet transmissions by secondary base-stations i.e., $\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}$. The probability that PB is not transmitting in a given time slot is therefore, $\left\{1-\frac{\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}}{p}\right\}$. The proportion of time neither PB nor SB_j transmits primary packet therefore equals,

$$1-C\sum_{k:\text{PU}_k\in\phi_j^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}-\frac{\sum_{k=1}^{N^{(p)}}C\lambda_k^{(p)}-C\sum_{k=1}^{N^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,\text{F}_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}}{p}.$$

Now, since the rate of secondary packet transmissions by SB_j is upper bounded by the proportion of time neither PB nor SB_j transmits a primary packet, therefore given this probability vector $\boldsymbol{q}'$ the maximum supportable secondary request generation rate, denoted as $\lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{q}')$, is given as,

$$\lambda^{(s)}_{j,\text{max,gen}}(\boldsymbol{q}') \;\leq\; \frac{1-C\sum_{k:\text{PU}_k\in\phi_j^{(p)}}\lambda_k^{(p)}\sum_{\mathbf{D}^{(p)}\in\hat{D}^{(p)}}\sum_{l=1}^{|F^{(p)}|}P_l^{(p)}Q(k,F_l^{(p)}|\mathbf{D}^{(p)})q'_{\mathbf{D}^{(p)}}}{C}$$

$$- \quad \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} - \sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} \sum_{\mathbf{D}^{(p)} \in \hat{D}^{(p)}} \sum_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(k, \mathrm{F}_l^{(p)} | \mathbf{D}^{(p)}) q'_{\mathbf{D}^{(p)}}}{p}. \tag{42}$$

The stability constraint at PB is given as

$$\frac{C \left\{ \sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} - \sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} \sum_{\mathbf{D}^{(p)} \in \hat{D}^{(p)}} \sum_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(k, \mathrm{F}_l^{(p)} | \mathbf{D}^{(p)}) q'_{\mathbf{D}^{(p)}} \right\}}{p} \leq 1. \tag{43}$$

The set of all $(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)})$ for which there exists $\boldsymbol{q}'$ satisfying (42) and (43) contains the region Λ_{gen}.

Now we simplify the RHS of (42) as follows. Similar to the proof of Lemma 1 we can show that the region defined by (42) and (43) over all feasible $\boldsymbol{q}'$ vectors can be described by only considering a smaller set of vectors $\boldsymbol{q}'$. Namely, we consider only those $\boldsymbol{q}'$ vectors whose non-zero components correspond to those primary availability matrices that represent each secondary base-station caching either $B-1$ or B most popular primary files in each period. In other words, to describe the region Λ_{gen} we consider only those vectors $\boldsymbol{q}'$ in (42) and (43) for which $q'_{D^{(p)}} > 0$ only if for every $j \in \{1, \ldots, M\}$ either

$$D_{n,j}^{(p)} = \begin{cases} 1, & \text{if } 1 \leq n \leq B-1 \\ 0, & \text{otherwise,} \end{cases} \tag{44}$$

or

$$D_{n,j}^{(p)} = \begin{cases} 1, & \text{if } 1 \leq n \leq B \\ 0, & \text{otherwise.} \end{cases} \tag{45}$$

Then we can simplify (42) and obtain upper bound of $\lambda_{j,\text{max,gen}}^{(s)}(\boldsymbol{\lambda}^{(p)})$ as

$$\begin{aligned} \lambda_{j,\text{max,gen}}^{(s)}(\boldsymbol{\lambda}^{(p)}) \quad \leq \quad & \frac{1 - C \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} \left(\sum_{l=1}^{B-1} P_l^{(p)} + P_B^{(p)} \sum_{\mathbf{D}^{(p)} \in \hat{D}^{(p)}} q'_{\mathbf{D}^{(p)}} \right)}{C} \\ - \quad & \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} - \sum_{n=1}^{M} \sum_{k:\text{PU}_k \in \phi_n^{(p)}} \lambda_k^{(p)} \left(\sum_{l=1}^{B-1} P_l^{(p)} + P_B^{(p)} \sum_{\mathbf{D}^{(p)} \in \hat{D}^{(p)}} q'_{\mathbf{D}^{(p)}} \right)}{p} \end{aligned} \tag{46}$$

By simplifying the RHS of (46) we obtain

$$\begin{aligned} \lambda_{j,\text{max,gen}}^{(s)}(\boldsymbol{\lambda}^{(p)}) \quad \leq \quad & \frac{1}{C} + (\frac{1}{p} - 1) \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} \left(\sum_{l=1}^{B-1} P_l^{(p)} + P_B^{(p)} \right) \\ & + \frac{\sum_{\substack{n=1 \\ n \neq j}}^{M} \sum_{k:\text{PU}_k \in \phi_n^{(p)}} \lambda_k^{(p)} \left(\sum_{l=1}^{B-1} P_l^{(p)} + P_B^{(p)} \right)}{p} - \frac{\sum_{k=1}^{N^{(p)}} \lambda_k^{(p)}}{p}. \end{aligned} \tag{47}$$

From (47) and (41) we obtain for every $j \in \{1, \ldots, M\}$,

$$\lambda_{j,\text{max,ach}}^{(s)}(\boldsymbol{\lambda}^{(p)}) \quad \geq \quad \lambda_{j,\text{max,gen}}^{(s)}(\boldsymbol{\lambda}^{(p)}) - (\frac{1}{p} - 1) \sum_{k:\text{PU}_k \in \phi_j^{(p)}} \lambda_k^{(p)} P_B^{(p)} - \frac{\sum_{\substack{n=1 \\ n \neq j}}^{M} \sum_{k:\text{PU}_k \in \phi_n^{(p)}} \lambda_k^{(p)} P_B^{(p)}}{p} \quad . \tag{48}$$

APPENDIX B
PROOF OF THEOREM 1

For any policy X that caches exactly $B-1$ most popular primary files at each secondary base-station in every period, we define an utility function $\psi^X(r)$ (where $r=1,2,\ldots$) as

$$\psi^X(r) \triangleq \sum_{k=1}^{M} Z_k^{(p)}(r) E[\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1} \mu_k^{X,(p)}(\tau) | \boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] + \sum_{k=1}^{M} \sum_{f\in F^{(s)}} Z_{f,k}^{(s)}(r) E[\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1} \mu_{f,k}^{X}(\tau) | \boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]. \quad (49)$$

This utility function corresponds to the conditional drift of the Lyapunov function L_1; maximizing this utility function is equivalent to minimizing the conditional drift. Next, we present an algorithm ALT1 that maximizes $\psi^X(r)$ among the set of all policies Φ that perform the following.

1) Every secondary base-station caches exactly $B-1$ most popular primary files in each period.
2) In each time slot if the queue in PB is empty, first select an activation vector. If a secondary base-station is allowed to transmit according to this activation vector then it transmits a packet corresponding to one of the queues of cached files. If the queue is empty transmit a dummy packet.

The policy ALT1 performs the following in every period:

1) *Caching Policy:* Cache files and admit file requests in same way as FPCP.
2) *Scheduling at PB:* Transmit primary packets in same way as FPCP.
3) *Scheduling at Secondary Base-stations:* If at some time slot in r'th period, $t_{r,j}$, PB is not transmitting any file, obtain the set of secondary base-stations that can simultaneously, denoted as $E^*_{\text{ALT1}}(t_{r,j})$, as

$$E^*_{\text{ALT1}}(t_{r,j}) \in \underset{E\in\tilde{E}}{\operatorname{argmax}} \left(\left(\max\left\{ U_k^{(p)}(t_{r,1}), U^{(s)}_{f_k^*(r),k}(t_{r,1}) \right\} \right)_{k=1}^{M} \right)^T E. \quad (50)$$

Suppose, according to $E^*_{\text{ALT1}}(t_{r,j})$ some SB_k is allowed to transmit at time slot $t_{r,j}$. Each such SB_k transmits the packet corresponding to the HOL packet request at the aggregate primary queue in SB_k if $U_k^{(p)}(t_{r,1})$ is greater than or equal to $U^{(s)}_{f_k^*(r),k}(t_{r,1})$. Otherwise, SB_k transmits the packet corresponding to the HOL packet request at queue of file $f_k^*(r)$. In case the primary queue is empty, SB_k transmits a dummy packet.

Thus ALT1 is different from FPCP in that in ALT1 scheduling decisions at the secondary base-stations are made based on queue lengths in the beginning of the period while in FPCP they are made based on their instantaneous values.

We show the following.

Lemma 3: $\psi^{\text{FPCP}}(r) \geq \psi^{\text{ALT1}}(r) - K_1$ where $K_1 \geq 0$ is a finite constant independent of queue lengths.

Proof: Let $U_{f,k}^{X(s)}(t_{r,j})$, $U_k^{X,(p)}(t_{r,j})$ denote the secondary queue length of file f and primary queue lengths respectively in SB_k under policy X at time slot $t_{r,j}$ (where $j=1,\ldots,T$). First we note that location of transmission-opportunities for secondary base-stations are identical under both FPCP and ALT1. Note that for any policy X and

any $j = 1, \ldots, T$, we have,

$$U_k^{(p)}(t_{r,1}) - \hat{T} \leq U_k^{(p)}(t_{r,j}) \leq U_k^{(p)}(t_{r,1}) + \hat{T} \ \forall k \in \{1, \ldots, M\} \tag{51}$$

$$U_{f,k}^{(s)}(t_{r,1}) - \hat{T} \leq U_{f,k}^{(s)}(t_{r,j}) \leq U_{f,k}^{(s)}(t_{r,1}) + \hat{T} \ \forall k \in \{1, \ldots, M\}, \quad f \in F^{(s)} \tag{52}$$

as maximum number of arrivals to any queue and transmissions from any base-station in a period is upper bounded by $\hat{T}$.

Then for every $j \in \{1, \ldots, T\}$ we have,

$$E[\{\sum_{k=1}^{M} Z_k^{(p)}(r)\mu_k^{\text{FPCP},(p)}(t_{r,j}) + \sum_{k=1}^{M}\sum_{f\in F^{(s)}} Z_{f,k}^{(s)}(r)\mu_{f,k}^{\text{FPCP}}(t_{r,j})\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]$$

$$= E[\{\sum_{k=1}^{M} U_k^{(p)}(t_{r,1})\mu_k^{\text{FPCP},(p)}(t_{r,j}) + \sum_{k=1}^{M}\sum_{f\in F^{(s)}} U_{f,k}^{(s)}(t_{r,1})\mu_{f,k}^{\text{FPCP}}(t_{r,j})\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] \tag{53}$$

$$\geq E[\sum_{k=1}^{M} U_k^{\text{FPCP},(p)}(t_{r,j})\mu_k^{\text{FPCP},(p)}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]$$
$$+E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}} U_{f,k}^{\text{FPCP},(s)}(t_{r,j})\mu_{f,k}^{\text{FPCP}}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] - \hat{T}M(1 + |F^{(s)}|) \tag{54}$$

$$\geq E[\sum_{k=1}^{M} U_k^{\text{FPCP},(p)}(t_{r,j})\mu_k^{\text{ALT1},(p)}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]$$
$$+E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}} U_{f,k}^{\text{FPCP},(s)}(t_{r,j})\mu_{f,k}^{\text{ALT1}}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)] - \hat{T}M(1 + |F^{(s)}|) \tag{55}$$

$$\geq E[\sum_{k=1}^{M} U_k^{(p)}(t_{r,1})\mu_k^{\text{ALT1},(p)}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]$$
$$+E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}} U_{f,k}^{(s)}(t_{r,1})\mu_{f,k}^{\text{ALT1}}(t_{r,j})|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)TM(1 + |F^{(s)}|) \tag{56}$$

$$\tag{57}$$

$$= E[\{\sum_{k=1}^{M} Z_k^{(p)}(r)\mu_k^{\text{ALT1},(p)}(t_{r,j}) + \sum_{k=1}^{M}\sum_{f\in F^{(s)}} Z_{f,k}^{(s)}(r)\mu_{f,k}^{\text{ALT1}}(t_{r,j})\}|\boldsymbol{Z}^{(s)}(r), \boldsymbol{Z}^{(p)}(r)]. \tag{58}$$

Both relations (54) and (56) follow from (51) and (52); (55) follows because FPCP maximizes, among all policies X in Φ, the following expression for every time slot τ in r'th period:

$$\sum_{k=1}^{M} U_k^{(p)}(\tau)E[\mu_k^{X,(p)}(\tau)|(U_{f,k}^{(s)}(\tau))^T, (U_k^{(p)}(\tau))^T] + \sum_{k=1}^{M}\sum_{f\in F^{(s)}} U_{f,k}^{(s)}(\tau)E[\mu_{f,k}^{X,(s)}(\tau)|(U_{f,k}^{(s)}(\tau))^T, (U_k^{(p)}(\tau))^T].$$

Combining the above for all $j = 1, \ldots, T$ we prove the Lemma. ■

Proof of Theorem 1: For any $(\boldsymbol{\lambda}^{(p)},\boldsymbol{\lambda}^{(s)}) \in$ Interior(Λ) there exists some constant $\epsilon > 0$ s.t. $(\boldsymbol{\lambda}^{(p)} + \boldsymbol{\epsilon}^{(p)},\boldsymbol{\lambda}^{(s)} + \boldsymbol{\epsilon}^{(s)}) \in \Lambda$ where $\boldsymbol{\epsilon}^{(p)}$, $\boldsymbol{\epsilon}^{(s)}$ are vectors of lengths $N^{(p)}$ and $N^{(s)}$ respectively and whose each component is ϵ.

It can be easily shown that there exists a stationary policy STAT1 in Φ that stabilizes the network of queues without knowledge of queue lengths at the secondary base-stations, for all request generation rate vectors in Λ.

Since, $(\boldsymbol{\lambda}^{(p)}+\boldsymbol{\epsilon}^{(p)},\boldsymbol{\lambda}^{(s)}+\boldsymbol{\epsilon}^{(s)}) \in \Lambda$, therefore STAT1 stabilizes the network for this request generation vector as well.

We consider the Lyapunov function $L_1(.)$ and the corresponding conditional drift $\Delta_1(.)$ defined in Section IV. Then for the r'th period, under the FPCP algorithm, the conditional drift $\Delta_1(r)$ satisfies,

$$\Delta_1(r) \leq 2\hat{T}^2M(1+|F^{(s)}|) - 2\sum_{k=1}^{M}\sum_{f\in F^{(s)}} E[Z_{f,k}^{(s)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{f,k}^{\text{FPCP}}(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}} A_{f,i}^{(s)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)]$$
$$-2\sum_{k=1}^{M} E[Z_{k}^{(p)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{k}^{\text{FPCP},(p)}(\tau) - \sum_{f\in F^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}} A_{f,i}^{(p)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)] \quad (59)$$

$$\leq K_1 + 2\hat{T}^2M(1+|F^{(s)}|) - 2\sum_{k=1}^{M}\sum_{f\in F^{(s)}} E[Z_{f,k}^{(s)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{f,k}^{\text{ALT1}}(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}} A_{f,i}^{(s)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)]$$
$$-2\sum_{k=1}^{M} E[Z_{k}^{(p)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{k}^{\text{ALT1},(p)}(\tau) - \sum_{f\in F^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}} A_{f,i}^{(p)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)] \quad (60)$$

$$\leq K_1 + 2\hat{T}^2M(1+|F^{(s)}|) - 2\sum_{k=1}^{M}\sum_{f\in F^{(s)}} E[Z_{f,k}^{(s)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{f,k}^{\text{STAT1},(s)}(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}} A_{f,i}^{(s)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)]$$
$$-2\sum_{k=1}^{M} E[Z_{k}^{(p)}(r)\sum_{\tau=t_{r,1}}^{t_{r,1}+T-1}\{\mu_{k}^{\text{STAT1},(p)}(\tau) - \sum_{f\in F^{(p)}}\sum_{i:\text{PU}_i\in\phi_k^{(p)}} A_{f,i}^{(p)}(\tau)\}|\boldsymbol{Z}^{(s)}(r),\boldsymbol{Z}^{(p)}(r)] \quad (61)$$

$$\leq K_1 + 2\hat{T}^2M(1+|F^{(s)}|) - 2T\epsilon\sum_{k=1}^{M}\sum_{f\in F^{(s)}} Z_{f,k}^{(s)}(r) - 2T\epsilon\sum_{k=1}^{M} Z_{k}^{(p)}(r). \quad (62)$$

The inequality (60) follows from Lemma 3. The inequality (61) follows because ALT1 maximizes $\psi^X(r)$ among all policies in Φ including STAT1. Therefore all queues are strongly stable by Theorem 4.1 in [17]. ■

APPENDIX C

A. *Minimizing the Expression in* (28)

We first discuss the issues in minimizing the expression in (28). Then we discuss the additional assumptions made to resolve those issues and find an approximate solution. Then we find an approximate solution by considering only policies that satisfy constraints **C1**, **C2** and **C3**. We obtain the solution in two steps. First, we obtain an expression for the denominator of the fraction term in (28) given a set of cached primary files. Second, we obtain an expression representing the minimum value of the numerator of the fraction term in (28) given a set of cached primary files. Combining the two expressions we approximately minimize the expression in (28).

Finding a caching and scheduling scheme that minimizes the expression in (28) is complicated as the number of primary files to be cached in each base-station at the beginning of the r''th frame (i.e. time slot $\tilde{t}_{r',1}$) depends not only on whether primary queues are empty but also on the primary file request event (i.e. what primary files

were requested and by which users) at time slot $\tilde{t}_{r',1}-1$. Minimizing the expression in (28) by calculating the probability of all possible primary file request events at $\tilde{t}_{r',1}-1$ is difficult.

In order to minimize the expression in (28) without calculating the probability of all possible primary file request events at $\tilde{t}_{r',1}-1$, we make some assumptions[11]. We assume:

1) The network controller delays queuing the primary requests generated at time slot $\tilde{t}_{r',1}-1$ to base-stations until the caching decision occurs at time slot $\tilde{t}_{r',1}$. At time slot $\tilde{t}_{r',1}$, the network controller first determines which files to cache at each base-station and then caches those files at corresponding base-stations. It then queues primary file requests that arrived at $\tilde{t}_{r',1}-1$, to either PB or cooperative secondary base-stations based on the new cache contents. Finally, the controller uses the updated queues to determine the scheduling policy at time slot $\tilde{t}_{r',1}$.
2) The network controller always admits requests of cached primary files at cooperative secondary base-stations.

Due to both assumptions we minimize the expression in (28) by only considering caching schemes that determine which primary files to cache, independent of the actual primary file request event at $\tilde{t}_{r',1}-1$.

Under the above assumptions, and considering only policies that satisfy constraints **C1**, **C2** and **C3**, we can find an approximate solution to (28) in terms of only the secondary queue lengths and the average rate of primary request generation processes.

Now we obtain an expression for the expected frame length i.e. the denominator of the fraction term in (28), $E[(\tilde{t}_{r+1,1}-\tilde{t}_{r,1})|\tilde{\boldsymbol{Z}}^{(s)}(r')]$. We denote the expected frame length, if $i_1,\ldots,\ i_G$ most popular primary files are cached at $\text{SB}_1,\ldots,\ \text{SB}_G$ respectively at the beginning of the frame, as $\kappa(i_1,\ldots,i_G)$. Due to constraint **C1**, i_k is less than B for every $k\in\{1,\ldots,G\}$. Each frame consists of three distinct segments.

1) The first segment consists of time slots in which all primary queues are empty. Clearly, duration of this segment is a geometric random variable with parameter equal to the total generation rate of primary user requests λ_{tot} where

$$\lambda_{\text{tot}} \quad = \quad \sum_{k=1}^{N^{(p)}} \lambda_k^{(p)} C \tag{63}$$

and its mean is therefore, $\frac{1}{\lambda_{\text{tot}}}$.
2) The second segment consists of time slots in which at least one primary queue in some cooperative secondary base-station is non-empty. Expected duration of this segment is the average proportion of time at least one secondary base-station is transmitting a primary packet multiplied by the expected length of the frame. Now, the proportion of the time each secondary base-station transmits a primary packet is independent of each other due to constraint **C2** and according to Little's law its mean equals $\hat{\nu}_k(i_k)$ for SB_k (for a stable system)

[11]We emphasize, these assumptions are only made to calculate the $\hat{n}_k^*(r)$ variables. Neither the VPCP algorithm nor the system model behaves according to these assumptions.

where

$$\hat{\nu}_k(i_k) \quad = \quad \begin{cases} \sum\limits_{j:\text{PU}_j \in \phi_k^{(p)}} \lambda_j^{(p)} \sum\limits_{n=1}^{i_k} P_n^{(p)} C, & \forall k \in \{1,\ldots,G\}, i_k \in \{1,\ldots,B-1\} \\ 0, & \text{otherwise.} \end{cases} \tag{64}$$

The expected proportion of time some secondary base-station transmits a primary packet, denoted as $\nu_0(i_1,\ldots,i_G)$, is therefore,

$$\begin{aligned} \nu_0(i_1,\ldots,i_G) \quad = \quad & \sum_{k_1=1}^{G} \hat{\nu}_{k_1}(i_{k_1}) - \sum_{k_1,k_2:1\le k_1<k_2\le G} \hat{\nu}_{k_1}(i_{k_1})\hat{\nu}_{k_2}(i_{k_2}) + \\ & \cdots + (-1)^{G-1} \sum_{k_1,k_2,\ldots,k_G:1\le k_1<k_2<\ldots<k_G\le G} \hat{\nu}_{k_1}(i_{k_1})\hat{\nu}_{k_2}(i_{k_2})\ldots\hat{\nu}_{k_G}(i_{k_G}) \\ & \forall i_k \in \{0,\ldots,B-1\}, k \in \{1,\ldots,G\}. \end{aligned} \tag{65}$$

3) The third segment consists of time slots in which all primary queues in the cooperative secondary base-stations are empty but at least one of the primary queues in PB is non-empty. Expected duration of this segment is the proportion of time PB transmits a primary packet multiplied by the expected length of the frame. Again, according to Little's law this expected proportion is obtained as $\frac{\lambda_{\text{tot}}-\lambda_{\text{rel}}(i_1,\ldots,i_G)}{p}$ where $\lambda_{\text{rel}}(i_1,\ldots,i_G)$ denotes the total rate of primary packet transmission by secondary base-stations given i_k most popular primary files are cached at SB_k for every $k \in \{1,\ldots,G\}$,

$$\lambda_{\text{rel}}(i_1,\ldots,i_G) \quad = \quad \sum_{k=1}^{G} \sum_{\text{PU}_j \in \phi_k^{(p)}} \sum_{n=1}^{i_k} P_n^{(p)} \lambda_j^{(p)} C \quad \forall i_m \in \{0,\ldots,B-1\}, m \in \{1,\ldots,G\}. \tag{66}$$

We denote as $\nu(i_1,\ldots,i_G)$ the average proportion of time some base-station (including PB) is transmitting a primary packet. Then,

$$\nu(i_1,\ldots,i_G) \quad = \quad \frac{\lambda_{\text{tot}} - \lambda_{\text{rel}}(i_1,\ldots,i_G)}{p} + \nu_0(i_1,\ldots,i_G) \quad \forall i_k \in \{0,\ldots,B-1\}, k \in \{1,\ldots,G\}. \tag{67}$$

By adding the expected duration of the three segments, noting that the sum equals $\kappa(i_1,\ldots,i_G)$ and then equating both sides we obtain the value of $\kappa(i_1,\ldots,i_G)$,

$$\kappa(i_1,\ldots,i_G) \quad = \quad \frac{\nu(i_1,\ldots,i_G)}{\lambda_{\text{tot}}(1-\nu(i_1,\ldots,i_G))} + \frac{1}{\lambda_{\text{tot}}} \quad \forall i_k \in \{0,\ldots,B-1\}, k \in \{1,\ldots,G\}. \tag{68}$$

Next, we obtain expression for the numerator of the fraction term in (28):
$-E[\sum\limits_{l=1}^{M} \sum\limits_{f\in F^{(s)}} \tilde{Z}_{f,l}^{(s)}(r') \sum\limits_{\tau=\tilde{t}_{r'+1,1}}^{\tilde{t}_{r'+1,1}-1} \mu_{f,l}(\tau) | \tilde{\boldsymbol{Z}}^{(s)}(r')]$. Given $i_1,\ldots,i_G$ most popular primary files are cached at $\text{SB}_1,\ldots,\text{SB}_G$ respectively, the minimum value of this term is obtained as,
$-\sum\limits_{k=1}^{G} \alpha_k(i_1,\ldots,i_G) U_{f_k^*(r),k}^{(s)}(t_{r,1}) - \frac{1}{\lambda_{\text{tot}}} \left(\left(U_{f_l^*(r),l}^{(s)}(t_{r,1}) \right)_{l=1}^{M} \right)^T E_{\text{VPCP}}^*(t_{r,1})$ where

$$\alpha_k(i_1,\ldots,i_G) \quad = \quad \kappa(i_1,\ldots,i_G)\{\nu_0(i_1,\ldots,i_G) - \hat{\nu}_k(i_k)\} \quad \forall i_m \in \{0,\ldots,B-1\}, m \in \{1,\ldots,G\} \tag{69}$$

$$E_{\text{VPCP}}^*(t_{r,j}) \quad \in \quad \underset{E\in\tilde{E}}{\text{argmax}} \left(\left(U_{f_l^*(r),l}^{(s)}(t_{r,j}) \right)_{l=1}^{M} \right)^T E \quad \forall j \in \{1,\ldots,T\}. \tag{70}$$

The term $-\sum\limits_{k=1}^{G} \alpha_k(i_1,\ldots,i_G) U_{f_k^*(r),k}^{(s)}(t_{r,1})$ represents the contribution of secondary packet transmissions from cooperative base-stations towards the aforementioned numerator term, during those time slots in which some

secondary base-station is transmitting a primary packet. This follows as the term $\alpha_k(i_1,\ldots,i_G)$ in (69) represents the expected duration of those time slots in the frame in which some cooperative secondary base-station other than SB_k is transmitting a primary packet. The term $-\frac{1}{\lambda_{\text{tot}}}\left(\left(U^{(s)}_{f^*_l(r),l}(t_{r,1})\right)_{l=1}^{M}\right)^T E^*_{\text{VPCP}}(t_{r,1})$ represents the contribution of secondary packet transmissions from all base-stations towards the aforementioned numerator term during those time slots in which no base-station is transmitting any primary packet.

Finally, combining the numerator and denominator terms we obtain the $\hat{n}^*_k(r)$ variables that minimize (28) for every $r \in \{1,2,\ldots,\}$ and $k \in \{1,\ldots,G\}$ as follows:

$$(\hat{n}^*_1(r),\ldots,\hat{n}^*_G(r))^T \in \underset{\{i_k:1\leq k\leq G, 0\leq i_k\leq B-1\}}{\text{argmin}} V\sum_{k=1}^{G} i_k - \frac{\sum_{k=1}^{G}\alpha_k(i_1,\ldots,i_G)U^{(s)}_{f^*_k(r),k}(t_{r,1}) + \frac{1}{\lambda_{\text{tot}}}\left(\left(U^{(s)}_{f^*_l(r),l}(t_{r,1})\right)_{l=1}^{M}\right)^T E^*_{\text{VPCP}}(t_{r,1})}{\kappa(i_1,\ldots,i_G)}. \quad (71)$$

For the special case when all secondary base-stations are non-interfering, (71) can be simplified and $\hat{n}^*_k(r)$ can be obtained separately for every $k \in \{1,\ldots,G\}$ as:

$$\hat{n}^*_k(r) \in \underset{i_k:0\leq i_k\leq B-1}{\text{argmin}} Vi_k - \frac{\hat{\nu}_k(i_k)}{p}\sum_{\substack{k'=1\\k'\neq k}}^{G} U^{(s)}_{f^*_{k'}(r),k'}(t_{r,1}) - \hat{\nu}_k(i_k)(\frac{1}{p}-1)U^{(s)}_{f^*_k(r),k}(r). \quad (72)$$

B. Definition of the Region $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$

We define the region $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$ by establishing stability constraint for secondary queues at each secondary base-station. The region defines the set of all request generation rate vectors that can be supported under algorithms that satisfy constraints **C1**, **C2** and **C3**. In describing the region $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$, we limit ourselves to stationary policies in which each secondary base-station caches exactly $B-1$ most popular primary files in each period and always admits requests for those files. This is because for algorithms that satisfy constraint **C1**, transmission opportunities for secondary base-stations are maximized when each cooperative base-station caches as many popular primary files as possible. Next, we obtain the set of feasible secondary packet transmission rates and use them to obtain the region $\Lambda(\boldsymbol{\lambda}^{(p)})$.

Next, we find the set of feasible secondary packet transmission rates at every secondary base-station given each cooperative base-station caches $B-1$ most popular primary files in each period and always admits requests for those files. First we note that the probability of the event: "all primary queues are empty" equals the ratio of average number of idle slots, $\frac{1}{\lambda_{\text{tot}}}$, and the average duration of a frame, $\kappa(B-1,\ldots,B-1)$ i.e. $\frac{1}{\lambda_{\text{tot}}\kappa(B-1,\ldots,B-1)}$. The set of all feasible average secondary transmission rate vectors, counting only those time slots when all primary queues are empty, is therefore given as $\frac{1}{\lambda_{\text{tot}}\kappa(B-1,\ldots,B-1)}\textbf{conv}(\tilde{E})$. Due to constraint **C3**, a cooperative secondary base-station can also transmit secondary packets simultaneously when another cooperative secondary base-stations transmits a primary packet. The probability of the event: "SB_k is not transmitting any primary packet but another cooperative base-station is transmitting a primary packet" can be calculated as $\nu(B-1,\ldots,B-1)-\hat{\nu}_k(B-1)$. Note that the term $\nu(B-1,\ldots,B-1)$ denotes the probability of the event: " at least one base-station is transmitting

a primary packet". Therefore, the feasible transmission rate for any cooperative secondary base-station SB_k is $R_k + \nu(B-1,\ldots,B-1) - \hat{\nu}_k(B-1)$ while that for any non-cooperative secondary base-station SB_l is R_l for some $(R_1, R_2, \ldots, R_M)^T \in \frac{1}{\lambda_{\text{tot}}\kappa(B-1,\ldots,B-1)}\textbf{conv}(\tilde{E})$.

Therefore, the set $\Lambda^{(s)}(\boldsymbol{\lambda}^{(p)})$ consists of all $\boldsymbol{\lambda}^{(s)}$ that satisfies

$$C \sum_{\text{SU}_i \in \phi_k^{(s)}} \lambda_i^{(s)} \leq \begin{cases} R_k, & \forall k \in \{G+1,\ldots,M\} \\ R_k + \nu(B-1,\ldots,B-1) - \hat{\nu}_k(B-1) & \forall k \in \{1,\ldots,G\}, \end{cases} \tag{73}$$

for some $(R_1, R_2, \ldots, R_M)^T \in \frac{1}{\lambda_{\text{tot}}\kappa(B-1,\ldots,B-1)}\textbf{conv}(\tilde{E})$. Note that the terms $\nu(.)$, $\hat{\nu}(.)$, λ_{tot} and $\kappa(.)$ are functions of the primary request generation rate vector $\boldsymbol{\lambda}^{(p)}$.

APPENDIX D
PROOF OF THEOREM 2

Proof: Since the conditional drift term in (71) decreases with increasing secondary queue lengths, there exists a finite constant K' s.t. if $\max_{f \in F^{(s)}} U_{f,k}^{(s)}(t_{r,1}) > K'$ for even one SB_k (where $k \in \{1,\ldots,M\}$) then (71) is minimized when $n_1 = n_2 = \cdots = n_G = B-1$. Let the first frame begin at time slot $\tilde{T}_0$.

We first consider the case when $\tilde{T}_0 < \infty$ and $\tilde{Z}_{f,k}^{(s)}(r) > K' + T$ for at least one $f \in F^{(s)}$ and SB_k. We will show that for this case the conditional drift of the Lyapunov function L_2 under VPCP is upper bounded by a finite positive constant minus a weighted sum of secondary queue lengths at the beginning of the frame. Later we will consider the case when $\tilde{T}_0 < \infty$ and $\tilde{Z}_{f,k}^{(s)}(r) \leq K' + T$ for every $f \in F^{(s)}$ and SB_k. In this case we will show that the conditional drift of L_2 under the VPCP algorithm is upper bounded by a finite positive constant. Combining both cases we prove Theorem 2. The case where $\tilde{T}_0 = \infty$ is equivalent to special case of scheduling for the general network without any primary request arrival and is thereby skipped.

A. $\tilde{T}_0 < \infty$ and $\tilde{Z}_{f,k}^{(s)}(r) > K' + T$ for at least one $f \in F^{(s)}$ and SB_k

Consider the set of policies, denoted as Φ_2, in which caching of secondary files occur only at the beginning of every frame and caching of primary files occur at beginning of every period. In particular, under each algorithm in Φ_2, at the beginning of every period the network controller caches $B-1$ most popular primary files at each cooperative secondary base-station; each secondary base-station admits requests for those files. Under each algorithm in Φ_2, at the beginning of every frame, the network controller caches one secondary file at each cooperative base-station and B secondary files at other secondary base-stations. Moreover, the scheduling scheme for policies in Φ_2 satisfying constraints **C2** and **C3**.

Next, we compare the conditional drift under VPCP with that under an algorithm ALT2 that caches secondary files and schedules transmissions so as to maximize $E[\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1} \sum_{k=1}^{M} \sum_{f \in F^{(s)}} \tilde{Z}_{f,k}^{(s)}(r)\mu_{f,k}^{X}(\tau)|\tilde{\mathbf{Z}}^{(\mathbf{s})}(\mathbf{r})]$ among all policies X in Φ_2. Note that when $\tilde{Z}_{f,k}^{(s)}(r) > K' + T$, the length of r'th frame remains the same under both ALT2 and VPCP as both have $B-1$ most popular primary files in the caches of all cooperative secondary base-stations during the

duration of this frame. Furthermore, the secondary packet transmission opportunities are same under both policies as well. Assume the length of queue of file f at beginning of r'th frame, $U^{(s)}_{f,k}(\tilde{t}_{r,1})$ is given for every $k \in \{1,\ldots,M\}$ and $f \in F^{(s)}$. Since the number of arrivals and departures from any secondary queue in every time slot is upper bounded by a constant $\tilde{T_2}$ (where $\tilde{T_2} = CN^{(s)}$) we have for all $\tau \in [\tilde{t}_{r,1},\ldots,\tilde{t}_{r+1,1}-1]$,

$$\min\{U^{\text{ALT2},(s)}_{f,k}(\tau), U^{\text{VPCP},(s)}_{f,k}(\tau)\} \geq U^{(s)}_{f,k}(\tilde{t}_{r,1}) - \tau\tilde{T_2} \tag{74}$$

$$\max\{U^{\text{ALT2},(s)}_{f,k}(\tau), U^{\text{VPCP},(s)}_{f,k}(\tau)\} \leq U^{(s)}_{f,k}(\tilde{t}_{r,1}) + \tau\tilde{T_2}. \tag{75}$$

Let $\mathbf{U^{(s)}}(\tau)$ denote the vector of secondary queue lengths at time slot τ; clearly, $\tilde{\mathbf{Z}}^{(s)}(r) = \mathbf{U^{(s)}}(t_{r,1})$. Let $\boldsymbol{\mu}^{\Phi,(s)}(\tau)$ denote the vector of secondary packet transmissions at time slot τ under policy Φ. We have for every τ in r'th frame,

$$(\tilde{\mathbf{Z}}^{(s)}(r)^T\boldsymbol{\mu}^{\text{VPCP},(s)}(\tau) \geq (\mathbf{U}^{\text{VPCP},(s)}(\tau))^T\boldsymbol{\mu}^{\text{VPCP},(s)}(\tau) - \tau M|F^{(s)}|\tilde{T_2} \tag{76}$$

$$\geq (\mathbf{U}^{\text{VPCP},(s)}(\tau)^T\boldsymbol{\mu}^{\text{ALT2},(s)}(\tau) - \tau M|F^{(s)}|\tilde{T_2} \tag{77}$$

$$\geq (\tilde{\mathbf{Z}}^{(s)}(r))^T\boldsymbol{\mu}^{\text{ALT2},(s)}(\tau) - 2\tau M|F^{(s)}|\tilde{T_2}. \tag{78}$$

The relation (76) and (78) follow from (75) and (74) respectively. The relation (77) follows from the definition of VPCP. Summing over all τ we obtain

$$E[\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}(\tilde{\mathbf{Z}}^{(s)}(r))^T\boldsymbol{\mu}^{\text{VPCP},(s)}(\tau)|\tilde{\mathbf{Z}}^{(s)}(r)] \geq E[\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}(\tilde{\mathbf{Z}}^{(s)}(r))^T\boldsymbol{\mu}^{\text{ALT2},(s)}(\tau)|\tilde{\mathbf{Z}}^{(s)}(r)]$$
$$-E[(\tilde{t}_{r+1,1}-2\tilde{t}_{r,1})^2M|F^{(s)}|\tilde{T_2}|\tilde{\mathbf{Z}}^{(s)}(r)]. \tag{79}$$

Now consider the Lyapunov function $L_2(r) = \sum_{k=1}^{M}\sum_{f\in F^{(s)}}(\tilde{Z}^{(s)}_{f,k}(r))^2$ for any $r \in \{1,2,\ldots,\}$, and its conditional drift $\Delta_2(r) = E[L_2(r+1) - L_2(r)|\tilde{\mathbf{Z}}^{(s)}(r)]$. When $\tilde{Z}^{(s)}_{f,k}(r) > K' + T$ for at least one $f \in F^{(s)}$ and SB_k, the conditional drift $\Delta_2(r)$ under VPCP satisfies

$$\Delta_2(r) \leq 2E[(\tilde{t}_{r+1,1}-\tilde{t}_{r,1})^2M|F^{(s)}|(\tilde{T_2})^2|\tilde{\mathbf{Z}}^{(s)}(r)]$$
$$-E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}^{(s)}_{f,k}(r)\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}\{\mu^{\text{VPCP}}_{f,k}(\tau) - \sum_{i:\text{SU}_i\in\phi^{(s)}_k}A^{(s)}_{f,i}(\tau)\}|\tilde{\mathbf{Z}}^{(s)}(r)] \tag{80}$$

$$\leq 4M|F^{(s)}|W_2(\tilde{T_2})^2 - E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}^{(s)}_{f,k}(r)\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}\{\mu^{\text{ALT2}}_{f,k}(\tau) - \sum_{i:\text{SU}_i\in\phi^{(s)}_k}A^{(s)}_{f,i}(\tau)\}|\tilde{\mathbf{Z}}^{(s)}(r)]. \tag{81}$$

The inequality (80) follows as number of packet transmissions and departures from any queue in any time slot is bounded. The inequality (81) follows from (79).

For any secondary request generation rate vector $\boldsymbol{\lambda}^{(s)} \in \text{Interior}(\Lambda(\boldsymbol{\lambda}^{(p)}))$ there exists $\epsilon > 0$ s.t. $\boldsymbol{\lambda}^{(s)} + \boldsymbol{\epsilon} \in \Lambda(\boldsymbol{\lambda}^{(p)})$. For secondary request generation rate-vector $\boldsymbol{\lambda}^{(s)} + \boldsymbol{\epsilon}$ there exists a stabilizing stationary policy STAT2 in Φ_2 that selects the secondary file to cache at each frame and secondary packet transmissions, independent of queue length, and under which all queues are stable. Now since ALT2 maximizes $\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,0}-1}\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}^{(s)}_{f,k}(r)E[\mu^{X}_{f,k}(\tau)|\tilde{\mathbf{Z}}^{(s)}(r)]$

among all policies X in Φ_2 of which STAT2 is one, we obtain from (81)

$$\Delta_2(r) \leq 4M|F^{(s)}|W_2(\tilde{T}_2)^2 - E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,k}^{(s)}(r)\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}\{\mu_{f,k}^{\text{STAT2}}(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}}A_{f,i}^{(s)}(\tau)\}] \quad (82)$$

$$\leq 4M|F^{(s)}|W_2(\tilde{T}_2)^2 - \epsilon W_1\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,k}^{(s)}(r). \quad (83)$$

B. $\tilde{T}_0 < \infty$ and $\tilde{Z}_{f,k}^{(s)}(r) \leq K' + T$ for every $f \in F^{(s)}$ and SB$_k$

For this case, the conditional drift under VPCP policy over the r'th frame is,

$$\Delta_2(r) \leq 2E[(\tilde{t}_{r+1,1}-\tilde{t}_{r,1})^2 M|F^{(s)}|(\tilde{T}_2)^2|\tilde{\mathbf{Z}}^{(s)}(r)]$$
$$-E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,k}^{(s)}(r)\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}\{\mu_{f,k}^{\text{VPCP}}(\tau) - \sum_{i:\text{SU}_i\in\phi_k^{(s)}}A_{f,i}^{(s)}(\tau)\}|\tilde{\mathbf{Z}}^{(s)}(r)] \quad (84)$$

$$\leq 2M|F^{(s)}|W_2(\tilde{T}_2)^2 + E[\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\sum_{i:\text{SU}_i\in\phi_k^{(s)}}\tilde{Z}_{f,k}^{(s)}(r)\sum_{\tau=\tilde{t}_{r,1}}^{\tilde{t}_{r+1,1}-1}A_{f,i}^{(s)}(\tau)|\tilde{\mathbf{Z}}^{(s)}(r)] \quad (85)$$

$$\leq 2M|F^{(s)}|W_2(\tilde{T}_2)^2 + (K'+T)M\tilde{T}_2|F^{(s)}|E[\tilde{t}_{r+1,1}-\tilde{t}_{r,1}|\tilde{\mathbf{Z}}^{(s)}(r)] \quad (86)$$

$$\leq K_3 \quad (87)$$

where K_3 is a finite constant since $E[\tilde{t}_{r+1,1}-\tilde{t}_{r,1}], K'$ are bounded. Combining (83) and (87) we get for every r,

$$\Delta_2(r) \leq 4M|F^{(s)}|W_2(\tilde{T}_2)^2 + K_3 - \epsilon' W_1\sum_{k=1}^{M}\sum_{f\in F^{(s)}}\tilde{Z}_{f,k}^{(s)}(r). \quad (88)$$

where $\epsilon' = \min(\epsilon, \frac{4W_2(\tilde{T}_2)^2}{(K'+T)W_1})$. Therefore all queues are strongly stable by Theorem 4.1 in [17].

■

APPENDIX E

A. Achievable Capacity Region for Multichannel Network

The achievable capacity region Λ is obtained by establishing stability constraints at primary and secondary base stations.

1) Stability Constraint at PB: Consider the vector $\boldsymbol{q} = (q_{\mathbf{D}^{(p)}})$ where the term $q_{\mathbf{D}^{(p)}}$ denotes the iid probability with which the primary availability matrix $\mathbf{D}^{(p)} \in \tilde{D}^{(p)}$ is selected in each period. Clearly, for every $\mathbf{D}^{(p)}$ the term $q_{\mathbf{D}^{(p)}}$ should be non-negative and less than 1 (i.e. $0 \leq q_{\mathbf{D}^{(p)}} \leq 1$) and the sum of all $q_{\mathbf{D}^{(p)}}$ terms should equal 1 (i.e. $\sum_{\mathbf{D}^{(p)}\in\tilde{D}^{(p)}} q_{\mathbf{D}^{(p)}} = 1$). Given a vector $\boldsymbol{q}$ we obtain the probability of packet transmissions from PB to a primary user as follows. We then use it to obtain stability constraint at PB for a given $\boldsymbol{q}$.

Given a primary availability matrix $\mathbf{D}^{(p)}$ is used with probability $q_{\mathbf{D}^{(p)}}$ the proportion of time PB transmits to any primary user on a given channel equals the ratio of rate of primary packet transmissions from PB to that user and the probability of a successful transmission from PB, p. The former term is simply the packet request generation

rate of that primary user minus rate of packet transmissions by an adjacent secondary base-station (if it exists) to that user. Mathematically, we denote the proportion of time PB transmits to PU_i (where $i \in \{1,\ldots,N^{(p)}\}$) on channel c_j (where $j \in \{1,\ldots,J\}$) given $\boldsymbol{q}$ as $\pi_{i,j}(\boldsymbol{q})$. Then we have,

$$\pi_{i,j}(\boldsymbol{q}) = \begin{cases} \frac{C}{p}\lambda_i^{(p)} \left(1 - \sum\limits_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} \sum\limits_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(i, \text{F}_l^{(p)} | \mathbf{D}^{(p)}) q_{\mathbf{D}^{(p)}}\right) & \text{if } \text{PU}_i \in \gamma_j \\ 0, & \text{otherwise.} \end{cases} \tag{89}$$

In (89) the term $C\left(1 - \sum\limits_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} \sum\limits_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(i, \text{F}_l^{(p)} | \mathbf{D}^{(p)}) q_{\mathbf{D}^{(p)}}\right)$ denotes the rate of successful primary packet transmissions from PB to PU_i on channel c_j. Since each successful primary packet transmission takes $\frac{1}{p}$ slots on average the RHS of (89) is the proportion of time PB transmits to PU_i on channel c_j.

The proportion of time PB does not transmit on a given channel c_j, denoted as $\pi_{0,j}(\boldsymbol{q})$, is therefore

$$\pi_{0,j}(\boldsymbol{q}) = 1 - \sum_{i=1}^{N^{(p)}} \pi_{i,j}(\boldsymbol{q}). \tag{90}$$

For stability of queues at PB we must have $\pi_{0,j}(\boldsymbol{q})$ to be non-negative for all j i.e.,

$$\pi_{0,j}(\boldsymbol{q}) \geq 0. \tag{91}$$

2) Stability Constraints at Secondary Base-stations: First we find the set of feasible secondary packet transmission rates and then use them to obtain stability constraint at secondary base-stations.

We first obtain the set $\Gamma(\boldsymbol{q})$ denoting the set of all feasible average transmission rate vectors that can be offered to queues in secondary base-stations, given the primary availability matrices are selected according to vector $\boldsymbol{q}$. The set of links associated with secondary base-stations that are active simultaneously in each time slot depends on PB's transmission activity in each channel. Given one set of transmission activities, the set of average transmission rate vectors that can be offered to queues in secondary base-stations can be obtained as the convex hull of all feasible activation vectors. Let $\xi(i_1,\ldots,i_J)$ denote the set of feasible activation vectors in $\tilde{E}$ given PB is transmitting to PU_{i_j} on channel c_j (where $i_j \in \{1,\ldots,N^{(p)}\}$, $j \in \{1,\ldots,J\}$); with slight abuse of notation we denote no transmission from PB on channel c_j by setting i_j as 0. Therefore, $\Gamma(\boldsymbol{q})$ is obtained as

$$\Gamma(\boldsymbol{q}) \quad = \quad \sum_{i_1 \in \{0,\ldots,N^{(p)}\}} \cdots \sum_{i_J \in \{0,\ldots,N^{(p)}\}} \left(\prod_{j=1}^{J} \pi_{i_j,j}(\boldsymbol{q}) \textbf{conv}\left(\xi(i_1,\ldots,i_J)\right)\right). \tag{92}$$

Let R denote one vector in $\Gamma(\boldsymbol{q})$. For every $\text{PU}_m \in \phi_k^{(p)}$ and $\text{SU}_n \in \phi_k^{(s)}$ we denote the component corresponding to link $(\text{SB}_k, \text{PU}_m, c_j)$ and $(\text{SB}_k, \text{SU}_n, c_j)$ as $R_{k,m,j}^{(p)}$ and $R_{k,n,j}^{(s)}$ respectively (where $k \in \{1,\ldots,M\}$, $m \in \{1,\ldots,N^{(p)}\}$, $n \in \{1,\ldots,N^{(s)}\}$, $j \in \{1,\ldots,J\}$).

Given vector $\boldsymbol{q}$, the queues for PU_m at any secondary base-station SB_k, where $\text{PU}_m \in \phi_k^{(p)}$, are stable if there exists some $R \in \Gamma(\boldsymbol{q})$ for which the transmission-rate offered to PU_m is greater than or equal to required packet transmission rate i.e.,

$$C\lambda_m^{(p)} \sum_{\mathbf{D}^{(p)} \in \tilde{D}^{(p)}} \sum_{l=1}^{|F^{(p)}|} P_l^{(p)} Q(m, \text{F}_l^{(p)} | \mathbf{D}^{(p)}) q_{\mathbf{D}^{(p)}} \quad \leq \quad \sum_{j=1}^{J} R_{k,m,j}^{(p)}. \tag{93}$$

Similarly, the stability constraint for queues corresponding to $\text{SU}_n \in \phi_k^{(s)}$ is,

$$C\lambda_n^{(s)} \leq \sum_{j=1}^{J} R_{k,n,j}^{(s)}. \tag{94}$$

The region Λ is the set of all request generation vectors $(\boldsymbol{\lambda}^{(p)}, \boldsymbol{\lambda}^{(s)})$ for which there exists probability vector $\boldsymbol{q}$ and transmission rate generation vector $R \in \Gamma(\boldsymbol{q})$ such that (89), (91), (93) and (94) are satisfied for every primary and secondary user.

Note that for a single channel, i.e. $J = 1$, the region Λ reduces to the region in Section III.

B. Obtaining the Expression for $f_k^*(r)$

Note that since all secondary base-stations are non-interfering, the $f_k^*(r)$ variables obtained by minimizing the RHS of (38) can be equivalently obtained as,

$$f_k^*(r) \in \underset{f \in F^{(s)}}{\text{argmin}} -2 \sum_{i:\text{SU}_i \in \phi_k^{(s)}} \sum_{f \in F^{(s)}} E[Z_{f,k,i}^{(s)}(r) \sum_{\tau=t_{r,1}}^{t_{r,1}+T-1} \mu_{f,k,i}^{X}(\tau) | \boldsymbol{Z}^{(s)}(r)]. \tag{95}$$

In order to solve the minimization problem in (95) we first find the expected proportion of time for which each SB_k can transmit secondary packets simultaneously on at least m (where $1 \leq m \leq J$) channels, denoted as $y_{k,m}$. For policies that cache $B-1$ primary files in each period and satisfy constraint **C4**, the term $y_{k,m}$ is obtained as,

$$y_{k,m} = \sum_{n=m}^{J} (-1)^{n-m} \sum_{k_1,\ldots,k_n : 1 \leq k_1 < k_2 \ldots < k_n \leq J} \nu_{k,k_1} \nu_{k,k_2} \cdots \nu_{k,k_n} \forall m \in \{1,\ldots,J\} \tag{96}$$

where

$$\nu_{k,j} = 1 - \frac{C(\lambda_{\text{tot},j} - \lambda_{\text{rel},j})}{p} - \sum_{i:\text{PU}_i \in \phi_k^{(p)} \cap \gamma_j} \lambda_i^{(p)} \sum_{m=1}^{B-1} P_m^{(p)} C \quad \forall j \in \{1,\ldots,J\} \tag{97}$$

$$\lambda_{\text{tot},j} = \sum_{k:\text{PU}_k \in \gamma_j} \lambda_k^{(p)} C \quad \forall j \in \{1,\ldots,J\} \tag{98}$$

$$\lambda_{\text{rel},j} = \sum_{k=1}^{M} \sum_{i:\text{PU}_i \in \phi_k^{(p)} \cap \gamma_j} \sum_{m=1}^{B-1} P_m^{(p)} \lambda_i^{(p)} C \quad \forall j \in \{1,\ldots,J\} \tag{99}$$

and $\theta_{f,k,m}(\tau)$ (where $\theta_{f,k,m}(\tau) \in \{1,\ldots,N^{(s)}\}$) denotes the index of the secondary user with m'th highest queue length among all queues in SB_k associated with secondary file f at time slot τ. The term $\nu_{k,j}$ represents expected proportion of time SB_k can transmit secondary packets on channel c_j, $\lambda_{\text{tot},j}$ represents total rate of packet transmission to primary users associated with channel c_j and $\lambda_{\text{rel},j}$ represents total rate of packet transmission from secondary base-stations to primary users associated with channel c_j. Clearly, $f_k^*(r)$ is then obtained for every $k \in \{1,\ldots,M\}$ as:

$$f_k^*(r) \in \underset{f \in F^{(s)}}{\text{argmax}} \sum_{m=1}^{J} y_{k,m} U_{f,k,\theta_{f,k,m}(t_{r,1})}(t_{r,1}). \tag{100}$$

REFERENCES

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

[2] I. Maric, R. Yates, and G. Kramer, "Capacity of interference channels with partial transmitter cooperation," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3536–3548, 2007.

[3] I. Marić, A. Goldsmith, and G. Kramer, "On the capacity of interference channels with one cooperating transmitter," *European Transactions on Telecommunications*, vol. 19, no. 4, pp. 405–420, 2008.

[4] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3037–3063, 2005.

[5] O. Simeone, Y. Bar-Ness, and U. Spagnolini, "Stable throughput of cognitive radios with and without relaying capability," *IEEE Trans. Commun.*, vol. 55, no. 12, pp. 2351–2360, 2007.

[6] I. Krikidis, N. Devroye, and J. Thompson, "Stability analysis for cognitive radio with multi-access primary transmission," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 72–77, 2010.

[7] A. Fanous and A. Ephremides, "Stable throughput in a cognitive wireless network," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 3, pp. 523–533, 2013.

[8] A. El-Sherif, A. Sadek, and K. Liu, "Opportunistic multiple access for cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 704–715, 2011.

[9] R. Urgaonkar and M. Neely, "Opportunistic cooperation in cognitive femtocell networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 3, pp. 607–616, 2012.

[10] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *Proc. of IEEE Infocom*, 2012, pp. 1107–1115.

[11] N. Golrezaei, A. G. Dimakis, and A. F. Molisch, "Wireless device-to-device communications with distributed caching," in *IEEE International Symposium on Information Theory Proceedings (ISIT)*, 2012, pp. 2781–2785.

[12] N. Golrezaei, A. F. Molisch, and A. G. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," in *IEEE International Conference on Communications (ICC)*, 2012, pp. 7077–7081.

[13] J. Zhao, W. Gao, Y. Wang, and G. Cao, "Delay-constrained caching in cognitive radio networks," in *Proc. of IEEE Infocom*, 2014, pp. 2094–2102.

[14] J. Zhao and G. Cao, "Spectrum-aware data replication in intermittently connected cognitive radio networks," in *Proc. of IEEE Infocom*, 2014, pp. 2238–2246.

[15] M. Amble, P. Parag, S. Shakkottai, and L. Ying, "Content-aware caching and traffic management in content distribution networks," in *Proc. of IEEE Infocom*, 2011, pp. 2858–2866.

[16] N. Abedini and S. Shakkottai, "Content caching and scheduling in wireless broadcast networks with elastic and inelastic traffic," in *WiOpt*, May 2011, pp. 125–132.

[17] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.