

Utility-Maximizing Data Dissemination in Socially Selfish Cognitive Radio Networks

Hongxing Li*, Wei Huang*, Chuan Wu*, Zongpeng Li[†] and Francis C.M. Lau*

*Department of Computer Science, The University of Hong Kong, Hong Kong
Email: {hxli,whuang,cwu,fcmlau}@cs.hku.hk

[†]Department of Computer Science, University of Calgary, Canada
Email: zongpeng@ucalgary.ca

Abstract—In cognitive radio networks, the occupation patterns of the primary users can be very dynamic, which makes optimization (e.g., utility maximization) of data dissemination among secondary users difficult. Even under the assumption that all secondary users are fully collaborative, the optimization requires cross-layer decision making which is challenging. The challenge escalates if users are socially selfish, who prefer to relay data only to those other users with whom there are social ties. Such *social selfishness* of users translates into new constraints on network protocol design. There has been no study so far on the impact of social selfishness on data dissemination in cognitive radio networks. In this paper, we consider social selfishness of secondary users, and propose the design of a joint end-to-end rate control, routing, and channel allocation protocol which can maximize the overall throughput utility of multi-session unicast in cognitive radio networks. We give a distributed implementation of the protocol. Based on a Lyapunov optimization framework, we address social preferences of users using differentiated buffer sizes and relay rates for different data sessions, and apply back-pressure based transmission scheduling to achieve guaranteed utility optimality. A unique contribution of our Lyapunov optimization is that only a *finite-sized* buffer is required at each user node, which sets our design apart from other designs in existing literature where they assume infinite buffers. We investigate the optimality of our protocol and the impact of user social selfishness using both theoretical analysis and extensive simulations.

I. INTRODUCTION

Since its inception, cognitive radio network has been perceived as the next generation wireless network [1] to fundamentally enhance the spectrum efficiency of situations where the unlicensed spectrum is highly congested while the licensed spectrum is under-utilized [5]; such situations are already very common today. In a cognitive radio network, unlicensed users (or “secondary users”) are allowed to utilize the licensed spectrum for data transmission when the licensed user (or “primary user”) is not using it; this is done via dynamic spectrum access technologies for cognitive radios [2], [7].

When multiple data dissemination sessions (among secondary users) co-exist, a fundamental challenge in cognitive radio networks is to judiciously allocate the spectrum and schedule the transmissions such that the available spectrum is fully exploited to achieve the maximum network-wide throughput utility. This requires a cross-layer design, for optimal transport, network and MAC layer decisions to be made. At the transport layer, source nodes properly adjust

data injection rates for lower layers to handle; at the network layer, a relay node strategically decides the next-hop relay for each data session to be forwarded to; at the MAC layer, the available spectrum is carefully and dynamically allocated for transmission between pairs of nodes. All of these have to adapt to volatile channel occupancy patterns of the primary users while trying to maximize the end-to-end throughput utility among all data sessions.

To further add to the challenge, we can drop the usual assumption of fully collaborative data relay among the secondary users who in fact can behave selfishly during data relay. In real-world networks, e.g., civilian networks [8], [15], users have *social ties* at various strength levels. Naturally, a user would prefer helping others with whom there is a strong social tie, and less so for nodes with weak social ties. Such social selfishness of users complicates the design of efficient data dissemination protocols, especially when making routing decisions and link capacity allocation [13]. For example, a node with high link capacity and low hop count to the destination, which although appears favorable, may not constitute a good relay option if the node is not willing to assist in the data session. Therefore, traditional routing protocols, i.e., those based on link capacity or hop count, are no longer suitable in a socially selfish network.

In this paper, we consider social selfishness of secondary users in cognitive radio networks, based on which we design a joint end-to-end rate control, routing, and channel allocation protocol that can maximize the overall throughput utility of multiple unicast sessions. Our design is rooted in Lyapunov optimization theory [16], where utility maximization and network stability are achieved by back-pressure scheduling of transmissions among packet queues at the network nodes. We incorporate social selfishness of users in their transmission scheduling of packets belonging to different by-passing data sessions, according to the social ties between the users and the source/destination of each session. In particular, social preference of a user is novelly addressed by allocating differentiated buffer sizes and relay rates to different data sessions.

A salient contribution of our Lyapunov optimization is that *finite* buffer sizes are employed at each node with no-buffer-overflow guarantee. A number of back-pressure scheduling protocols [3], [4], [6], [20], [21], [23] have been proposed to achieve throughput maximization while adapting to the

dynamics of wireless networks. However, most are based on infinite node buffers, which is obviously an idealized and impractical assumption. Le *et al.* [11] have investigated optimal control of a wireless network with a finite buffer for each by-passing session per relay node, but an infinite buffer is still necessary at each source node in the worst cases. Neely [17] recently proposed an opportunistic scheduling protocol with bounded buffer size at each node for each data session, which would simply drop the packets should a buffer become full. In contrast, we demonstrate with rigorous proof that a finite buffer size without the possibility of buffer overflow suffices at each node using our protocol which can achieve global throughput utility maximization.

The contributions of this paper can be summarized as follows:

- ▷ We model social selfishness of users by assigning differentiated buffer sizes and relay rates allocated to data sessions of different source/destination pairs in a Lyapunov optimization framework for achieving throughput utility maximization in a cognitive radio network. To the best of our knowledge, this is the first work investigating the impact of social selfishness on protocol design in cognitive radio networks.
- ▷ We propose a back-pressure-style joint end-to-end rate control, routing, and channel allocation protocol for optimal multi-session unicast data dissemination, and give a distributed implementation for it. First time in the literature of back-pressure protocols, our protocol requires only a finite-sized buffer at each source or relay node with no buffer overflow, and is guaranteed to achieve an overall throughput utility that can be arbitrarily close to the ultimate optimum obtained when there is no constraint on buffer sizes.
- ▷ We demonstrate network stability and utility optimality of our protocol with rigorous theoretical analysis. Impact of social selfishness on throughput utility and end-to-end dissemination delay of different data sessions are further investigated using both case studies and empirical studies. An interesting discovery is that, contrary to intuition that larger buffers should be provisioned to preferred data sessions, allocating smaller buffers to them at nodes along their paths can actually lead to smaller end-to-end delay, without sacrificing throughput.

The remainder of the paper is organized as follows. We discuss related work in Sec. II and present the problem model in Sec. III. Detailed protocol design and theoretical performance analysis are presented in Sec. IV and Sec. V, respectively. The throughput utility of the protocol and the impact of social selfishness are evaluated with extensive simulations in Sec. VI. Finally, we conclude the paper in Sec. VII.

II. RELATED WORK

A. Utility Maximization with Back-pressure Protocol

Since the seminal work of Tassiulas *et al.* [20], back-pressure protocols for maximum-weight scheduling, which schedule links with the largest product of link capacity and

differential queue backlog, have been widely applied for utility maximization in multi-hop wireless networks [3], [4], [6], [21], [23]. It has been shown that optimal throughput can be achieved, however without any finite buffer guarantee.

Venkataramanan *et al.* [21] suggested a way to minimize the cumulative buffer utilization along the path of a unicast flow so as to reduce the end-to-end delay of the flow. However, there is no finite size guarantee for each buffer at the relay nodes. The challenge of using finite buffer in a back-pressure paradigm was not addressed until recently by Le *et al.* [11] and Neely [17] as we have mentioned in the introduction. In [11], the current queue size at each source node needs to be broadcast to all relay nodes, which can result in high communication overhead. Our protocol eliminates such broadcast overhead. In [17], the throughput utility is only compared with that of a T -slot lookahead policy which is an offline policy with perfect knowledge up to T slots into the future. Analysis is missing on how close the throughput utility approaches optimality.

For cognitive radio networks, Ding *et al.* [3], [4] have designed back-pressure protocols for routing with collaborative spectrum sensing, but without utility-optimality guarantee. Feng *et al.* [6] introduce a back-pressure routing protocol with primal-dual decomposition. No analysis of buffer size is presented. Xue *et al.* [23] propose a back-pressure throughput maximization protocol, under the constraints of bounded collision rates between secondary and primary users. The worst-case upper bound of buffer size at each node is provided, but the protocol cannot ensure that there will be no buffer overflow in situations where the buffer sizes are smaller than the upper bound. Our protocol provides that guarantee.

B. Social Selfishness in Network Protocol Design

Whereas assuming full collaborations among nodes is one extreme, the other extreme is assuming each network user is completely selfish. For the latter assumption, the literature has focused on incentive design, *e.g.*, [9], [22], [24], which is an orthogonal topic to our work¹. In this paper, we consider a new assumption of social selfishness, which may better capture user preferences in a real-world wireless network.

We are aware of only one paper, by Li *et al.* [13], which has made the same assumption as ours. They investigate routing design in socially selfish delay tolerant networks, where the probabilities that a node may forward traffic received from other nodes are differentiated. Unlike their work, we study a joint rate control, routing, and channel allocation scheme in a cognitive radio network, and we address social selfishness of users by differentiating both buffer sizes and targeted average relay rates.

III. PROBLEM MODEL

We now present the network model and the layers of the network stack under investigation using Lyapunov optimization, as well as our social selfishness model in the framework.

¹In the real world, the social ties tend to be fixed and stable. Thus, we design our protocol with social selfishness being a user demand, instead of designing incentive mechanisms to entice the users to collaborate.

A. Socially Selfish Cognitive Radio Network

We consider a cognitive radio network with a set of primary users V_P and a set of secondary users V_S . There are $|V_P|$ orthogonal spectrum channels $\mathcal{C} = \{c_1, \dots, c_{|V_P|}\}$, each subscribed by one primary user. The secondary users collectively constitute a multi-hop ad-hoc network. They distribute data flows to each other using the available channels that are not occupied by primary users at the time. There are M unicast data sessions among secondary users, denoted as set \mathcal{M} . Primary users are not involved in the data dissemination. Let s_m and d_m denote the source and destination of session $m \in \mathcal{M}$, respectively.

In the multi-hop network of secondary users, a source may directly transmit to a destination, if the later resides within the transmission range of the former and a spectrum channel is available to both nodes. Otherwise, a multi-hop route needs to be discovered to relay data packets.

Let E be the set of possible transmission links among secondary users, where link $e_{ij} \in E$ if node j is in the transmission range of node i . We consider a generic interference model. Let I denote the set of interference relations among potential transmissions in the network, which include two types of pairs: (1) $(e_{ij}, e_{kl}) \in I$ (with $e_{ij}, e_{kl} \in E$) denotes that transmission along link e_{ij} cannot be scheduled on the same channel concurrently with that along link e_{kl} ; (2) $(v_p, e_{ij}) \in I$ (with $v_p \in V_P$ and $e_{ij} \in E$) means that when primary user v_p is actively using its subscribed channel, transmission e_{ij} cannot simultaneously happen on the channel due to interference. We also assume that each secondary user is equipped with one radio only, such that it may either transmit or receive data on one channel at each time. Note that the generic interference model I subsumes most of the popular interference models in the literature, including the node-exclusive model and the k -hop ($k \geq 1$) interference model (used in [18] and the references therein).

Between each pair of secondary users i and j , a rational number $\rho_{ij} \in [0, 1]$ is given a priori and characterizes the strength of the social tie between the two users, where $\rho_{ij} = 1$ is strongest and $\rho_{ij} = 0$ means no tie at all². In multi-hop relay of data sessions, *social preference* of each *socially selfish* intermediate node n depends on its social relationship with the source and destination nodes of each by-passing data session m , indicated by $\rho_n^{(m)} = h(\rho_{n,s_m}, \rho_{n,d_m})$. $h(\cdot)$ is a non-negative non-decreasing function, and an example form $\theta\rho_{n,s_m} + (1-\theta)\rho_{n,d_m}$ with $\theta \in [0, 1]$ is used in our simulation in Sec. VI.

Table I summarizes the notations, for ease of reference.

B. Problem Model on Three Layers

We model the problems involved in enabling multi-session unicast at different layers of the network stack.

Transport Layer: End-to-end rate control is considered at the transport layer, at the source node of each data session. Suppose the system runs in a time-slotted fashion. In each

V_P	Set of primary users	V_S	Set of secondary users
\mathcal{C}	Set, orthogonal channels	\mathcal{M}	Set of unicast sessions
E	Set of links	I	Set, interference relations
M	# of data sessions	e_{ij}	Directed $i \rightarrow j$ link
$\mathbb{E}(\cdot)$	The expectation	$U(\cdot)$	Utility function
s_m	Source of session m	d_m	Destination of session m
ρ_{ij}	Social relation between user i and j		
$\rho_n^{(m)}$	Social relation between user n and data session m		
$A_m(t)$	Data arrival rate of session m in time slot t		
$A_{max}^{(m)}$	Maximum arrival rate of session m		
$r_m(t)$	Admissible data rate of session m in time slot t		
\bar{r}_m	Average admissible data rate of session m		
$\eta_m(t)$	Auxiliary variable of session m in time slot t		
$\bar{\eta}_m$	Average of auxiliary variable of session m		
$\mu_{ij}^{(m)}(t)$	Binary var: data session m is routed over e_{ij} in time slot t ?		
$\alpha_{ij}^{(c)}(t)$	Binary var: channel c is assigned to e_{ij} in time slot t ?		
$z_n^{(m)}$	Preset average relay rate of session m on user n		
$Q_n^{(m)}(t)$	Data queue of session m on user n in time slot t		
$q_n^{(m)}$	Buffer size for data queue $Q_n^{(m)}$ of session m on user n		
$Y_m(t)$	Transport virtual queue of session m in time slot t		
$G_n^{(m)}(t)$	Network virtual queue of session m on user n in time slot t		
V	User-defined constant weight in Lyapunov Optimization		
B	Quantity defined in Lyapunov Optimization in Sec. IV-C		

TABLE I
NOTATION TABLE.

time slot t , the application layer of source s_m injects data to the transport layer at rate $A_m(t) \in [0, A_{max}^{(m)}]$, where $A_{max}^{(m)}$ denotes the maximum data arrival rate for session m . Let $r_m(t) \in [0, A_m(t)]$ denote the admissible end-to-end data rate injected to the network layer of source s_m , such that congestion will not occur and *network stability*, to be defined shortly in Sec. III-D, is achieved. The traffic not admitted to network layer, *i.e.*, $A_m(t) - r_m(t)$, is discarded since it exceeds the network capacity and congestion would occur, if admitted.

Network Layer: Each secondary user $n \in V_S$ may receive data from multiple sessions (including the data session originated at itself), and makes routing decisions to forward them toward respective destinations. At the network layer of node n , a queue $Q_n^{(m)}$ is created to buffer data for session m , except at the destination node of session m where data is directly delivered to upper layers without buffering. Let $\mu_{ij}^{(m)}(t)$ be the amount of data of session m to be forwarded from node i to node j (where $e_{ij} \in E$) in time slot t . For every queue $Q_n^{(m)}$ at each node $n \in V_S - \{d_m\}$, $m \in \mathcal{M}$, we have the following regarding the queue size:

$$Q_n^{(m)}(t+1) = Q_n^{(m)}(t) - \sum_{e_{nj} \in E} \mu_{nj}^{(m)}(t) + \sum_{e_{in} \in E} \mu_{in}^{(m)}(t) + \mathbf{1}_{\{n=s_m\}} r_m(t), \quad (1)$$

where $\mathbf{1}_{\{n=s_m\}}$ is an indicator function defined as follows:

$$\mathbf{1}_{\{n=s_m\}} = \begin{cases} 1 & \text{if } n = s_m, \\ 0 & \text{otherwise.} \end{cases}$$

Without loss of generality, we assume that one unit of data can be transmitted from node i to node j in each time slot, *i.e.*, the capacity of any link $e_{ij} \in E$ is 1 if a transmission along the link is scheduled in a time slot. A similar assumption can also be found in [11], [23]. Therefore, we know that $\mu_{nj}^{(m)}(t)$ is either 0 or 1. In addition, $\mu_{nj}^{(m)}(t)$ cannot be larger than the size of queue $Q_n^{(m)}$ at time t . We derive

²How to derive and maintain the social ties is out of the scope of this paper.

$$\mu_{n_j}^{(m)}(t) \in \{0, \min\{Q_n^{(m)}(t), 1\}\}, \forall n, j \in V_S, n \neq d_m, \quad (2)$$

Note that we always have $\mu_{d_m}^{(m)}(t) = 0$ and $\mu_{nn}^{(m)}(t) = 0$.

MAC Layer: Based on the routing decisions from the network layer, a channel allocation and a link scheduling scheme is designed for the MAC layer, to schedule transmissions between nodes on each channel in each time slot.

We use a binary variable $\alpha_{n_j}^{(c)}(t)$ to indicate whether channel $c \in \mathcal{C}$ is allocated to transmission along link $e_{n_j} \in E$ in time slot t :

$$\alpha_{n_j}^{(c)}(t) = \begin{cases} 1 & \text{if } e_{n_j} \text{ is scheduled on channel } c \text{ in time slot } t, \\ 0 & \text{otherwise.} \end{cases}$$

The following constraints guarantee a feasible channel allocation and link scheduling scheme:

$$\sum_{c \in \mathcal{C}} \alpha_{n_j}^{(c)}(t) = \sum_{m \in \mathcal{M}} \mu_{n_j}^{(m)}(t), \forall e_{n_j} \in E, \quad (3)$$

$$\sum_{c \in \mathcal{C}} [\sum_{e_{n_j} \in E} \alpha_{n_j}^{(c)}(t) + \sum_{e_{in} \in E} \alpha_{in}^{(c)}(t)] \leq 1, \forall n \in V_S, \quad (4)$$

$$\alpha_{n_j}^{(c)}(t) + \sum_{(e_{kl}, e_{n_j}) \in I} \alpha_{kl}^{(c)}(t) \leq 1, \forall e_{n_j} \in E, c \in \mathcal{C}, \quad (5)$$

$$\alpha_{n_j}^{(c)}(t) \leq \mathbf{1}_{n_j}^{(c)}(t), \forall e_{n_j} \in E, c \in \mathcal{C}, \quad (6)$$

$$\alpha_{n_j}^{(c)}(t) \in \{0, 1\}, \forall e_{n_j} \in E, c \in \mathcal{C}, \quad (7)$$

where $\mathbf{1}_{n_j}^{(c)}(t)$ is an indicator function:

$$\mathbf{1}_{n_j}^{(c)}(t) = \begin{cases} 1 & \text{if channel } c \text{ is available to } e_{n_j} \text{ in time slot } t, \\ 0 & \text{otherwise.} \end{cases}$$

Constraint (3) states that the total amount of data transmission from node n to j on different channels, in one time slot, should be equal to the overall units of data from all sessions to be routed from n to j in that time slot. Inequality (4) models the primary interference constraint: a node can either transmit or receive on at most one channel in each time slot. Constraints (5) and (6) model the interference relations in I : the former indicates that transmission along link e_{n_j} should not be scheduled concurrently with that along any interfering link on the same channel; the latter guarantees that each link transmission can only be scheduled on an available channel. Channel c is available to link e_{ij} in time slot t if primary user v_p of the channel is not transmitting in the time slot or no interference exists between the transmissions from v_p and along e_{n_j} , i.e., $(v_p, e_{n_j}) \notin I$.

C. Social Preference in Routing

Socially selfish users $n \in V_S$ differentiate their resource allocation when relaying traffic for by-passing data sessions $m \in \mathcal{M}$, where n is neither the source ($n \neq s_m$) nor the destination ($n \neq d_m$).

Buffer space: A secondary user n provides a maximum buffer size of $q_n^{(m)} = f(\rho_n^{(m)})$ for by-passing data session $m \in \mathcal{M}$, where $f(\cdot)$ is a non-negative function that differentiates the buffer space allocated to sessions with different social relationships $\rho_n^{(m)}$ with user n . The goal is to provide better end-to-end throughput and delay performance for data sessions with

stronger social relationships. Interesting enough, we will show later in Sec. V and Sec. VI that smaller buffer sizes actually lead to better dissemination performance for a data session, in terms of lower end-to-end delay.

Note that $q_{d_m}^{(m)} = 0$, as there is no buffering requirement on the destination d_m of a session m . The buffer size $q_{s_m}^{(m)}$ at source s_m of session m is not constrained by the social relation $\rho_n^{(m)}$, as it is natural to favor a session originated from itself.

Average data relay rate: As data transmissions consume battery power, each secondary user n sets upper bounds for the average number of data units allowed for relay per time slot for by-passing data sessions $m \in \mathcal{M}, n \neq s_m$, and $n \neq d_m$.³ Let z_n denote the predefined upper bound of the average number of data units user n may relay for all by-passing sessions per time slot, which can be set according to the power consumption per transmission, the total battery capacity, and the frequency of battery charging at the user. For each of the sessions, an upper bound $z_n^{(m)}$ of average relay rate is set ($\sum_{m \in \mathcal{M}, n \neq s_m, n \neq d_m} z_n^{(m)} = z_n$), according to an increasing function of the social relationship $\rho_n^{(m)}$ between user n and session m . An example function $z_n^{(m)} = z_n \frac{\rho_n^{(m)}}{\sum_{m' \in \mathcal{M}, n \neq s_{m'}, n \neq d_{m'}} \rho_n^{(m')}}$ is used in our simulations.

We have that the average number of data units node n forwards for session m per time slot should be no larger than $z_n^{(m)}$ as follows

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E} \left(\sum_{e_{n_j} \in E} \mu_{n_j}^{(m)}(\tau) \right) \leq z_n^{(m)}, \quad \forall n \in V_S, m \in \mathcal{M}, n \neq s_m, n \neq d_m. \quad (8)$$

D. Network Stability and Capacity Region

Some important definitions and theorems are presented next from the literature on Lyapunov optimization [16], to be used in our protocol design and analysis.

Definition 1 (Queue and Network Stability [16]): A queue Q is *strongly stable* (or *stable* for short) if and only if

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(Q(\tau)) < \infty,$$

where $Q(\tau)$ is the queue size⁴ at time slot τ and $\mathbb{E}(\cdot)$ is the expectation. A network is *strongly stable* (or *stable* for short) if and only if all queues in the network are strongly stable.

Theorem 1 (Necessity and Sufficiency for Queue Stability [16]): For any queue Q with the following queuing law,

$$Q(t+1) = Q(t) - b(t) + a(t),$$

where $a(t)$ and $b(t)$ are the queue incoming rate and outgoing rate in time slot t , respectively, the following results hold:

Necessity: If queue Q is strongly stable, then its average incoming rate $\bar{a} = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(a(\tau))$ is no larger than the average outgoing rate $\bar{b} = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(b(\tau))$.

Sufficiency: If the average incoming rate \bar{a} is strictly smaller than the average outgoing rate \bar{b} , i.e., $\bar{a} + \epsilon \leq \bar{b}$ with $\epsilon > 0$, then queue Q is strongly stable.

³We assume no such rate limit is imposed on source s_m of a data session m for sending out its own data, i.e., enough power is provided at a source node for transmitting its own data.

IV. DYNAMIC UTILITY MAXIMIZATION ALGORITHM

In this section, we first describe the utility maximization problem and then give a cross-layer back-pressure protocol based on Lyapunov optimization [16].

A. Utility Maximization

Let \bar{r}_m denote the average end-to-end admissible data rate of unicast session m (i.e., throughput of session m), such that $\bar{r}_m = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(r_m(\tau))$. Let the vector of average end-to-end admissible rates of all sessions, $\bar{\mathbf{r}} = (\bar{r}_m, m \in \mathcal{M})$, denote the throughput of the network. Λ is the *capacity region* of the network, defined as the set of all vectors of admissible data rates $\bar{\mathbf{r}}$, for each of which there exists a routing and channel allocation algorithm to stabilize the network (Definition 1).

Let $U(\cdot)$ be a concave, differentiable, and non-decreasing utility function on throughput \bar{r}_m of each data session $m \in \mathcal{M}$. Our objective is to maximize the overall utility with guarantee of network stability and in the presence of social preference of the users.

$$\max \sum_{m \in \mathcal{M}} U(\bar{r}_m) \quad (9)$$

$$\text{s.t. } \bar{\mathbf{r}} \in \Lambda, \quad (10)$$

$$0 \leq r_m(t) \leq A_m(t), \forall m \in \mathcal{M}, t = 1, 2, \dots \quad (11)$$

Constraint (10) guarantees that the derived average admissible data rates in $\bar{\mathbf{r}}$ can achieve network stability, i.e., there exists a routing and channel allocation protocol that decides a set of feasible admissible data rates $r_m(t), \forall m \in \mathcal{M}$, in each time slot t (i.e., those satisfying constraints (2)(3)(4)(5)(6)(7)(8)), such that all queues are strongly stable in the network and each queue buffer is finite without overflow. Constraint (11) ensures that the admissible data rate of each session m in each time slot is non-negative and upper-bounded by the respective rate $A_m(t)$ at which data arrives at each source.

B. Introducing Virtual Queues

To derive a dynamic algorithm to solve the utility maximization problem, we apply Lyapunov optimization techniques. There are two issues that need to be resolved.

Issue 1: Since $r_m(t)$ is constrained by an arbitrary rate $A_m(t)$, it is hard to derive the expectation $\mathbb{E}(r_m(t))$, and thus the time averaged admissible rate of each session, $\bar{r}_m = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(r_m(\tau)), \forall m \in \mathcal{M}$.

Issue 2: Constraint (8), which formulates social selfishness of users in relay rate allocation, is defined on time averaged relay rates $\lim_{t \rightarrow \infty} \sup \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(\sum_{e_{nj} \in E} \mu_{nj}^{(m)}(\tau))$. How can we ensure the inequality by controlling the relaying rates $\mu_{nj}^{(m)}(t), \forall m \in \mathcal{M}$, per time slot?

Solution to issue 1: We derive a lower bound for $\bar{r}_m, \forall m \in \mathcal{M}$, by introducing a virtual queue Y_m at the transport layer of source node s_m of session m , as follows

$$Y_m(t+1) = \max\{Y_m(t) - r_m(t) + \eta_m(t), 0\}, \quad (12)$$

under the constraints

$$0 \leq r_m(t) \leq A_m(t), 0 \leq \eta_m(t) \leq A_{max}^{(m)},$$

where $\eta_m(t)$ is an auxiliary variable independent of $A_m(t)$, whose value in each time slot will be decided in our dynamic algorithm to be introduced shortly.

The rationale is that, if the stability of each virtual queue Y_m is guaranteed, we know from the necessity condition in Theorem 1 that $\bar{\eta}_m \leq \bar{r}_m$; therefore, by calculating expectation $\mathbb{E}(\eta_m(t))$ (which is calculable as long as $\eta_m(t)$ is independent of $A_m(t)$ and not arbitrarily assigned in $[0, A_{max}^{(m)}]$), and then the time average $\bar{\eta}_m = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(\eta_m(\tau))$, we are able to derive a lower bound $\bar{\eta}_m$ for \bar{r}_m .

Solution to issue 2: To fulfill users' social selfishness in relay rate allocation (constraint (8)), another virtual queue $G_n^{(m)}$ is introduced at the network layer of node n for each data session m it relays ($n \neq s_m$ and $n \neq d_m$). This queue records the relaying history for each data session at the node:

$$G_n^{(m)}(t+1) = \max\{G_n^{(m)}(t) + \sum_{e_{nj} \in E} \mu_{nj}^{(m)}(t) - z_n^{(m)}, 0\}. \quad (13)$$

According to the necessity condition in Theorem 1, if each virtual queue $G_n^{(m)}$ is stable, then the time averaged number of data units node n relays for session m per time slot, would not exceed the predefined upper bound $z_n^{(m)}$, i.e., constraint (8) is satisfied. Therefore, we can adjust relay rates $\mu_{nj}^{(m)}(t)$ in each time slot t to guarantee that the virtual queue is always stable, in order to satisfy constraint (8).

C. Dynamic Algorithm

To conclude, in our dynamic algorithm that solves the utilization maximization problem, three types of queues are needed, i.e., $Q_n^{(m)}$ ($\forall n \neq d_m, m \in \mathcal{M}$), Y_m ($\forall m \in \mathcal{M}$) and $G_n^{(m)}$ ($\forall n \neq s_m, n \neq d_m, m \in \mathcal{M}$). Let $\Theta(t) = [\mathbf{Q}, \mathbf{Y}, \mathbf{G}]$ be the vector of all queues in the system. Define our novel Lyapunov function as

$$L(\Theta(t)) = \frac{1}{2} \sum_{m \in \mathcal{M}} \sum_{n \neq d_m} \frac{q_{s_m}^{(m)}(Q_n^{(m)}(t))^2}{q_n^{(m)}} + \frac{1}{2} \sum_{m \in \mathcal{M}} (Y_m(t))^2 + \frac{1}{2} \sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} (G_n^{(m)}(t))^2. \quad (14)$$

The one-slot conditional Lyapunov drift is

$$\Delta(\Theta(t)) = L(\Theta(t+1)) - L(\Theta(t)). \quad (15)$$

According to the *drift-plus-penalty* framework in Lyapunov optimization [16], an upper bound for the following expression should be minimized in each time slot, with the observation of the queue states $\Theta(t)$, data arrival rates $A_m(t), \forall m \in \mathcal{M}$, and channel availability $\mathbf{1}_{nj}^{(c)}(t), \forall e_{nj} \in E, c \in \mathcal{C}$, such that a lower bound for $\sum_{m \in \mathcal{M}} U(\bar{\eta}_m)$ as well as $\sum_{m \in \mathcal{M}} U(\bar{r}_m)$ is maximized (see Chapter 5 of [16]):

$$\Delta(\Theta(t)) - V \sum_{m \in \mathcal{M}} U(\eta_m(t)).$$

Here, V is a user-defined constant that can be understood as the weight of utility in the expression.

By squaring the queuing laws (1), (12) and (13), we can derive the following inequality (the detailed steps can be found in our technical report [12]):

$$\Delta(\Theta(t)) - V \sum_{m \in \mathcal{M}} U(\eta_m(t)) \leq B - \Psi_1(t) - \Psi_2(t) - \Psi_3(t). \quad (16)$$

Here, $B = \frac{1}{2} [\sum_{m \in \mathcal{M}} \sum_{n \neq d_m} \frac{q_{s_m}^{(m)}}{q_n^{(m)}} \cdot ((\mathbf{1}_{\{n=s_m\}} A_{max}^{(m)} + 1)^2 + 1) + 2 \sum_{m \in \mathcal{M}} (A_{max}^{(m)})^2 + \sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} ((z_n^{(m)})^2 + 1)]$ is a constant value.

$\Psi_1(t)$, $\Psi_2(t)$, and $\Psi_3(t)$ are:

- Terms related to auxiliary variables $\eta_m(t)$:

$$\Psi_1(t) = V \sum_{m \in \mathcal{M}} U(\eta_m(t)) - \sum_{m \in \mathcal{M}} \eta_m(t) \cdot Y_m(t);$$

- Terms related to end-to-end rate control variables $r_m(t)$:

$$\Psi_2(t) = \sum_{m \in \mathcal{M}} r_m(t) \cdot [Y_m(t) - Q_{s_m}^{(m)}(t)];$$

- Terms related to routing decision variables $\mu_{n_j}^{(m)}(t)$:

$$\begin{aligned} \Psi_3(t) = & \sum_{m \in \mathcal{M}} \sum_{n \neq d_m} \left[\frac{q_{s_m}^{(m)}}{q_n^{(m)}} Q_n^{(m)}(t) \cdot \left[\sum_{e_{nj} \in E} \mu_{n_j}^{(m)}(t) - \sum_{e_{in} \in E} \mu_{in}^{(m)}(t) \right] \right. \\ & \left. + \sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} G_n^{(m)}(t) \cdot \left[z_n^{(m)} - \sum_{e_{nj} \in E} \mu_{n_j}^{(m)}(t) \right] \right. \end{aligned}$$

We then derive the following dynamic algorithm that observes queues at every time slot t and makes control decisions that maximize $\Psi_1(t)$, $\Psi_2(t)$, and $\Psi_3(t)$ at different layers of the network stack (thus minimizing the right-hand-side of the drift-plus-penalty bound in (16)), for maximization of joint utility $\sum_{m \in \mathcal{M}} U(\eta_m(t))$ (and thus lower bound for $\sum_{m \in \mathcal{M}} U(\bar{\eta}_m)$ and $\sum_{m \in \mathcal{M}} U(\bar{r}_m)$).

End-to-end Rate Control on Transport Layer: At the source node s_m of each session $m \in \mathcal{M}$, the admissible end-to-end data rate $r_m(t)$ is chosen by solving the following optimization problem:

$$\begin{aligned} \max_{r_m(t)} \quad & r_m(t) \cdot [Y_m(t) - Q_{s_m}^{(m)}(t)] \\ \text{s.t.} \quad & 0 \leq r_m(t) \leq A_m(t). \end{aligned} \quad (17)$$

Auxiliary variable $\eta_m(t)$ can be decided by solving:

$$\begin{aligned} \max_{\eta_m(t)} \quad & V \cdot U(\eta_m(t)) - Y_m(t) \cdot \eta_m(t) \\ \text{s.t.} \quad & 0 \leq \eta_m(t) \leq A_{max}^{(m)}. \end{aligned} \quad (18)$$

(17) is a linear optimization problem; (18) is a concave maximization problem with a linear constraint, since utility function $U(\cdot)$ is concave and differentiable. The maximum of the latter is achieved when the first order derivative of its objective function over $\eta_m(t)$ is zero. The optimal solutions can be derived as follows:

$$r_m(t) = \begin{cases} A_m(t) & \text{if } Y_m(t) > Q_{s_m}^{(m)}(t); \\ 0 & \text{otherwise} \end{cases}; \quad (19)$$

$$\eta_m(t) = \max\{\min\{U'^{-1}\left(\frac{Y_m(t)}{V}\right), A_{max}^{(m)}\}, 0\}, \quad (20)$$

where $U'^{-1}(\cdot)$ is the inverse function of $U'(\cdot)$, the first order derivative of $U(\cdot)$. Note that the above solutions are only related to local information at node s_m , *i.e.*, $A_m(t)$, $A_{max}^{(m)}$, $Y_m(t)$ and $Q_{s_m}^{(m)}(t)$, and can thus be derived in a fully distributed fashion.

Joint Routing and Channel Allocation on Network and MAC Layers: We can reformulate $\Psi_3(t)$ into

$$\begin{aligned} \Psi_3(t) = & \sum_{m \in \mathcal{M}} \sum_{n \neq d_m} \sum_{e_{nj} \in E} \mu_{n_j}^{(m)}(t) \cdot \left[\frac{q_{s_m}^{(m)}}{q_n^{(m)}} Q_n^{(m)}(t) - \frac{q_{s_m}^{(m)}}{q_j^{(m)}} Q_j^{(m)}(t) \right] \\ & - \mathbf{1}_{\{n \neq s_m\}} G_n^{(m)}(t) + \sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} G_n^{(m)}(t) z_n^{(m)}, \end{aligned}$$

with the indicator function

$$\mathbf{1}_{\{n \neq s_m\}} = \begin{cases} 1 & \text{if } n \neq s_m, \\ 0 & \text{otherwise.} \end{cases}$$

At each user $n \in V_S$, routing decisions $\mu_{n_j}^{(m)}(t)$ for each session $m \in \mathcal{M}$ (where $n \neq d_m$) can be made based on joint routing and channel allocation by solving the following:

$$\begin{aligned} \max \quad & \Psi_3(t) \\ \mu_{n_j}^{(m)}(t), \alpha_{n_j}^{(c)}(t), \forall e_{nj} \in E, \forall m \in \mathcal{M}, \forall c \in \mathcal{C} \end{aligned} \quad (21)$$

s.t. Constraints (2),(3),(4),(5),(6),(7).

Note here we do not include the social selfishness constraint (8) in the above optimization, since it can be implicitly satisfied by the optimal solution of the above problem. The reason is that the solution guarantees stability of each virtual queue $G_n^{(m)}$ ($\forall n \neq s_m, n \neq d_m, m \in \mathcal{M}$), which is proved in Theorem 2 in Sec. V.

Problem (21) can be simplified into a pure channel allocation problem related only to variables $\alpha_{n_j}^{(c)}$, $\forall e_{nj} \in E, \forall c \in \mathcal{C}$, as follows. Define

$$w_{n_j}^{(m)}(t) = \frac{q_{s_m}^{(m)}}{q_n^{(m)}} Q_n^{(m)}(t) - \frac{q_{s_m}^{(m)}}{q_j^{(m)}} Q_j^{(m)}(t) - \mathbf{1}_{\{n \neq s_m\}} G_n^{(m)}(t), \quad (22)$$

representing the weight of $\mu_{n_j}^{(m)}(t)$ in $\Psi_3(t)$. Constraint (4) implies that in each time slot, each node n can transmit at most one data session on at most one channel to another node j where link $e_{nj} \in E$; therefore, we know $\sum_{m \in \mathcal{M}} \mu_{n_j}^{(m)}(t) = \sum_{c \in \mathcal{C}} \alpha_{n_j}^{(c)}(t) \leq 1$, based on constraint (3). To maximize the objective function $\Psi_3(t)$, on each link e_{nj} , only session \hat{m}_{nj} associated with the largest weight $w_{n_j}^{(m)}(t)$ should be scheduled where

$$\hat{m}_{nj} = \arg \max_m \{w_{n_j}^{(m)}(t)\}, \quad (23)$$

while all other sessions are not, *i.e.*, $\mu_{n_j}^{(\hat{m}_{nj})}(t) = \sum_{c \in \mathcal{C}} \alpha_{n_j}^{(c)}(t) \leq 1$, and $\mu_{n_j}^{(m)}(t) = 0, \forall m \neq \hat{m}_{nj}$. Therefore, constraint (2) can be automatically satisfied since $\mu_{n_j}^{(\hat{m}_{nj})}(t)$ must be a binary value, and $\mu_{n_j}^{(\hat{m}_{nj})} = 0$ when $Q_n^{(\hat{m}_{nj})}(t) = 0$ and then $w_{n_j}^{(\hat{m}_{nj})} \leq 0$.

Further eliminating $\sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} G_n^{(m)}(t) z_n^{(m)}$ that is not related to $\mu_{n_j}^{(m)}$'s, we can modify the objective function $\Psi_3(t)$ into:

$$\begin{aligned} \Psi_4(t) = & \sum_{e_{nj} \in E, n \neq d_{\hat{m}_{nj}}} \mu_{n_j}^{(\hat{m}_{nj})}(t) \cdot w_{n_j}^{(\hat{m}_{nj})}(t) \\ = & \sum_{e_{nj} \in E, n \neq d_{\hat{m}_{nj}}} \sum_{c \in \mathcal{C}} \alpha_{n_j}^{(c)}(t) \cdot w_{n_j}^{(\hat{m}_{nj})}(t). \end{aligned}$$

To conclude, the joint routing and channel allocation problem (21) can be reduced to the following:

$$\begin{aligned} \max \quad & \Psi_4(t) \\ \alpha_{n_j}^{(c)}(t), \forall e_{nj} \in E, \forall c \in \mathcal{C} \end{aligned} \quad (24)$$

s.t. Constraints (4),(5),(6),(7).

After solving the above channel allocation problem, the routing decision can be made as follows:

$$\mu_{n_j}^{(m)} = \begin{cases} \sum_{c \in \mathcal{C}} \alpha_{n_j}^{(c)} & \text{if } m = \hat{m}_{nj}, \forall e_{nj} \in E, \forall m \in \mathcal{M}. \\ 0 & \text{otherwise} \end{cases}$$

The channel allocation problem (24) is a 0-1 integer program. A centralized solution with $1 - \delta$ approximation ratio

Algorithm 1 Dynamic Utility Maximization Algorithm in Time Slot t

Input: $Q_n^{(m)}(t)$, $Y_m(t)$, $G_n^{(m)}(t)$, $A_m(t)$, $A_{max}^{(m)}$, V , $q_n^{(m)}$ ($\forall n \in V_S, \forall m \in \mathcal{M}$).

Output: $r_m(t)$, $\eta_m(t)$, $\alpha_{nj}^{(c)}(t)$, $\mu_{nj}^{(m)}(t)$ ($\forall n \in V_S, \forall m \in \mathcal{M}, \forall c \in \mathcal{C}, \forall e_{nj} \in E$).

- 1: **End-to-End Rate Control:** For each data session $m \in \mathcal{M}$, the end-to-end rate $r_m(t)$ and auxiliary variable $\eta_m(t)$ are decided at source s_m as

$$r_m(t) = \begin{cases} A_m(t) & \text{if } Y_m(t) > Q_{s_m}^{(m)}(t), \\ 0 & \text{otherwise} \end{cases},$$

$$\eta_m(t) = \max\{\min\{U'^{-1}(\frac{Y_m(t)}{V}), A_{max}^{(m)}\}, 0\}.$$

- 2: **Joint Routing and Channel Allocation:** From each link $e_{ni} \in E$, calculate

$$w_{nj}^{(m)}(t) = \frac{q_{s_m}^{(m)}}{q_n^{(m)}} Q_n^{(m)}(t) - \frac{q_{s_m}^{(m)}}{q_j^{(m)}} Q_j^{(m)}(t) - \mathbf{1}_{\{n \neq s_m\}} G_n^{(m)}(t),$$

$$\forall m \in \mathcal{M},$$

$$\hat{m}_{nj} = \arg \max_m \{w_{nj}^{(m)}(t)\}.$$

Derive channel allocation variable $\alpha_{nj}^{(c)}, \forall e_{ni} \in E, c \in \mathcal{C}$, by solving problem (24) with the branch-and-bound algorithm [10] or our distributed algorithm in Alg. 2.

Routing decisions are made as follows

$$\mu_{nj}^{(m)} = \begin{cases} \sum_{c \in \mathcal{C}} \alpha_{nj}^{(c)} & \text{if } m = \hat{m}_{nj}, \forall e_{ni} \in E, m \in \mathcal{M}. \\ 0 & \text{otherwise} \end{cases}$$

- 3: Update queues $Q_n^{(m)}(t+1)$, $Y_m(t+1)$, and $G_n^{(m)}(t+1)$ based on queuing law (1), (12), and (13), respectively.
-

can be obtained using the branch-and-bound method [10], where $\delta \in (0, 1)$ is the solution accuracy defined by users. In the following, we design a distributed algorithm to solve this problem.

In summary, the sketch of our dynamic joint end-to-end rate control, routing, and channel allocation algorithm is given in Algorithm 1.

Remarks: Algorithm 1 is a back-pressure style protocol since the joint routing and channel allocation always prefer scheduling transmissions along links with the largest differential queue backlog $w_{nj}^{(\hat{m}_{nj})}(t)$. In the differential queue backlog $w_{nj}^{(m)}(t)$ in Eqn. (22), $Q_n^{(m)}(t)/q_n^{(m)}$ and $Q_j^{(m)}(t)/q_j^{(m)}$ are the occupancy ratio of data buffers of session m at node n and j in time slot t , and $G_n^{(m)}(t)$, the cumulative number of transmitted data units for session m that exceeds the number allowed by relay rate z_n^m , can be understood as the deficit of relay capacity for session m at node n . The implication of the joint routing and channel allocation in Algorithm 1 is to prioritize transmissions from a relatively more congested node (with high buffer occupancy ratio $Q_n^{(m)}(t)/q_n^{(m)}$) with low relay-capacity deficit to a less congested node (with low buffer occupancy ratio $Q_j^{(m)}(t)/q_j^{(m)}$).

Algorithm 2 Distributed Channel Allocation Protocol at Node n in Time Slot t

Input: $Q_n^{(m)}(t)$, $G_n^{(m)}(t)$, $q_{s_m}^{(m)}$, and $q_n^{(m)}$ ($\forall m \in \mathcal{M}$)

Output: $\alpha_{nj}^{(c)}(t)$ ($\forall e_{nj} \in E, c \in \mathcal{C}$)

- 1: **Initialization**

- Initialize channel allocation variable $\alpha_{nj}^c(t) \leftarrow 0, \forall e_{nj} \in E, c \in \mathcal{C}$, and a candidate set of links to schedule $\mathcal{L}_n \leftarrow \emptyset$;
- Sense the spectrum and get the available channel set C_n ;
- Exchange queue sizes $Q_n^{(m)}(t)$, $\forall m \in \mathcal{M}$, and available channel set C_n with neighbors;
- Calculate and propagate weight $w_{nj}^{\hat{m}_{nj}}$ based on Eqn. (22) and (23), and commonly available channel set $C_n \cap C_j$ for each link $e_{nj} \in E$;

- 2: **Channel allocation**

- For each link $e_{nj} \in E$ and $C_n \cap C_j \neq \emptyset$, do:

- if $w_{nj}^{\hat{m}_{nj}} \geq \max_{e_{ik} \in E, (e_{ik}, e_{nj}) \in I, C_i \cap C_k \neq \emptyset} \{w_{ik}^{\hat{m}_{ik}}\}$
- update candidate link set $\mathcal{L}(t) \leftarrow \mathcal{L}(t) \cup \{e_{nj}\}$;

- If $\mathcal{L}(t) \neq \emptyset$ and $e_{nj} = \arg \max_{e_{ni} \in \mathcal{L}_n} \{w_{ni}^{\hat{m}_{ni}}\}$, randomly select an available channel $c \in C_n \cap C_j$ and allocate it to e_{nj} by setting $\alpha_{nj}^{(c)} = 1$; inform each neighbor and senders of interfering links about this channel allocation, and end the algorithm here and that at node j .

- 3: **Information update:** Upon receiving a channel allocation, update available channel set for each local link and inform the updates to sender of each interfering link.

- 4: The algorithm ends if either the available channel set is empty for each local link or node n is scheduled as a receiver by some other node; Otherwise, go to step 2.
-

D. Distributed Channel Allocation Algorithm

We next propose a distributed protocol to solve the channel allocation problem in (24)—Algorithm 2. Here we refer to node j with $e_{nj} \in E$ as a “neighbor” of node n , and each link $e_{nj} \in E$ as a “local link” of user n . Similar to the existing literature [3], [4], all the control messages can be passed over a Common Control Channel (CCC) defined on an unlicensed band available to all secondary users. Note that, unlike broadcasting the current queue size at each source, *i.e.* $Q_{s_m}^{(m)}(t), \forall m \in \mathcal{M}$, to all relay nodes in [11], our algorithm only needs the buffer size at each source, which is constant and can be disseminated to each relay node once for all.

The distributed protocol executed at each node greedily allocates available channels to its local links, each with a weight $w_{nj}^{(\hat{m}_{nj})}(t)$ that is largest among weights of all its interfering links. Each node n senses the spectrum and maintains a set of available channels over time. To carry out channel allocation to its local links, it calculates link weight $w_{nj}^{\hat{m}_{nj}}$ based on Eqn. (22) and (23) for each local link e_{nj} , and derives commonly available channels with destination node of each local link, using necessary information from neighbors.

A link e_{nj} satisfying the following three conditions will be chosen, and a randomly selected available channel from the commonly available channel set of node n and node j will be assigned for the transmission: (1) there is at least one

commonly available channel at node n and node j ; (2) $w_{nj}^{\hat{m}_{n,j}}$ is the largest among weights $w_{ik}^{\hat{m}_{i,k}}$ on all its interfering links e_{ik} , where $(e_{ik}, e_{nj}) \in I$; (3) $w_{nj}^{\hat{m}_{n,j}}$ is also the largest among weights on all those local links at node n , which have the largest weights among their respective interfering links as well.

After channel c is allocated to link e_{nj} , each neighbor of user n and the sender of each interfering link to e_{nj} are informed, which will exclude the channel from their available sets and further propagate the information. The algorithm ends here at nodes n and j for the current time slot, while each non-scheduled node will continue with the above until they are either scheduled or the available channel set for each local link becomes empty.

We will show in Sec. VI that the performance of the distributed protocol is in general close to that achieved by the centralized branch-and-bound channel allocation method.

V. PERFORMANCE ANALYSIS

We analyze the performance guarantee provided by our dynamic algorithm (Algorithm 1) with respect to utility optimality, network stability, finite buffer sizes, as well as performance differentiation among sessions as a result of users' social selfishness.

Lemma 1 (Finite buffer without overflow): For each data session $m \in \mathcal{M}$, define

$$Y_{max}^{(m)} \triangleq VU'(0) + A_{max}^{(m)}, \quad q_{s_m}^{(m)} \triangleq VU'(0) + 2A_{max}^{(m)} + 1.$$

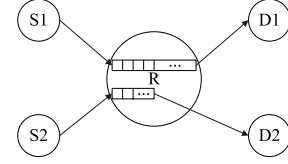
Queue sizes $Y_m(t)$, $Q_{s_m}^{(m)}(t)$, and $Q_n^{(m)}(t)$ (for all $n \neq s_m$ and $n \neq d_m$) are upper-bounded by buffer sizes $Y_{max}^{(m)}$, $q_{s_m}^{(m)}$, and any given non-negative $q_n^{(m)} = f(\rho_n^{(m)})$ (where $f(\cdot)$ is a non-negative function), respectively, without buffer overflow, i.e., $Y_m(t) \leq Y_{max}^{(m)}$, $Q_{s_m}^{(m)}(t) \leq q_{s_m}^{(m)}$, and $Q_n^{(m)}(t) \leq q_n^{(m)}$ in each time slot t .

The lemma can be proven by induction. The induction basis ($t = 0$) is trivial since all queues are empty at time slot 0. The induction steps for $Y_m(t)$ and $Q_{s_m}^{(m)}(t)$ can be derived based on the end-to-end rate control policy in Algorithm 1 and queuing laws (1) and (12), while the induction step for $Q_n^{(m)}(t)$ can be derived using the joint routing and channel allocation policy in Algorithm 1 and queuing law (1). The detailed proof can be found in our technical report [12], due to space limit.

Definition 2 (ϵ -optimum): The ϵ -optimal solution $\bar{r}^\epsilon = (\bar{r}_m^\epsilon, m \in \mathcal{M})$ is the optimal solution to the utility maximization problem modified from that in (9), without finite buffer requirement and replacing constraint (10) by $\bar{r} + \bar{\epsilon} \in \Lambda$ where $\bar{\epsilon} = (\epsilon, \dots, \epsilon)$ and $\epsilon > 0$.

Theorem 2 (Utility optimality): Given buffer size $q_{s_m}^{(m)} = VU'(0) + 2A_{max}^{(m)} + 1$ at source s_m of session m , and non-negative buffer size $q_n^{(m)} = f(\rho_n^{(m)})$ (where $f(\cdot)$ is a non-negative function) at each relay node n ($n \neq s_m, n \neq d_m$) of session m , $\forall m \in \mathcal{M}$, the overall utility achieved is within a constant gap $\frac{B}{V}$ from the ϵ -optimum, with Algorithm 1, i.e.,

$$\sum_{m \in \mathcal{M}} U(\liminf_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(r_m(\tau))) \geq \sum_{m \in \mathcal{M}} U(\bar{r}_m^\epsilon) - \frac{B}{V},$$



and all queues in the network are stable, and

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \sum_{m \in \mathcal{M}} \sum_{n \neq s_m, n \neq d_m} \mathbb{E}(G_n^{(m)}(\tau)) \leq \frac{B + V \sum_{m \in \mathcal{M}} (U(A_{max}^{(m)}) - U(\bar{r}_m^\epsilon))}{\epsilon C_2 (1 - \delta)},$$

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(Q_n^{(m)}(\tau)) \leq q_n^{(m)}, \quad \forall n \neq d_m, m \in \mathcal{M},$$

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(Y_m(\tau)) \leq VU'(0) + A_{max}^{(m)}, \quad \forall m \in \mathcal{M},$$

where $C_2 \in (0, 1)$ and $\delta \in (0, 1)$ are constants, and B, V are defined in Sec. IV-C.

The theorem is proved using Lyapunov optimization techniques. The complete proof can be found in our technical report [12], due to space limit.

Corollary 1: If buffer size $q_n^{(m)}$ at relay node n of session m ($\forall m \in \mathcal{M}, \forall n \neq s_m, n \neq d_m$) is further proportional to buffer size $q_{s_m}^{(m)} = VU'(0) + 2A_{max}^{(m)} + 1$ at the source s_m (i.e., $q_n^{(m)} = C_n^{(m)} \cdot q_{s_m}^{(m)}$ with constant $C_n^{(m)} \geq 0$), then B is a constant independent of V , and the overall utility achieved with our Algorithm 1 can be arbitrarily close to the maximum utility $\sum_{m \in \mathcal{M}} U(\bar{r}_m^*)$ achieved without finite buffer requirement, i.e., $\sum_{m \in \mathcal{M}} U(\lim_{t \rightarrow \infty} \inf \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}(r_m(\tau))) \rightarrow \sum_{m \in \mathcal{M}} U(\bar{r}_m^*)$, when $\epsilon \rightarrow 0$ and $V \rightarrow \infty$.

We next give two propositions on the impact of users' social selfishness on utility and end-to-end delay experienced by different sessions.

Proposition 1: If each relay node n differentiates the upper bound of average relay rate ($z_n^{(m)}$) set for different by-passing sessions m ($\forall m \in \mathcal{M}, \forall n \neq s_m, n \neq q_m$) as an increasing function of the social tie strength between n and m , sessions with stronger social ties with intermediate relays can achieve higher utility, when all the other parameters are the same.

Proposition 2: If each relay node n differentiates the buffer space ($q_n^{(m)}$) allocated to different by-passing sessions m ($\forall m \in \mathcal{M}, \forall n \neq s_m, n \neq q_m$) as a decreasing function of the social tie strength between n and m , sessions with stronger social ties with intermediate relays can achieve lower end-to-end delay while maintaining similar utility with other sessions, when all the other parameters are the same.

We briefly illustrate the propositions using a case study of the simple network in Fig. 1, and more detailed analysis can be found in our technical report [12]. There are one available channel, five secondary users, and two sessions from $S1$ to $D1$, and from $S2$ to $D2$, respectively. Suppose relay node R has stronger social tie with session 2, i.e., $\rho_R^{(1)} < \rho_R^{(2)}$; all other parameters with the two sessions are the same.

To illustrate proposition 1, suppose R only differentiates preset relay rate upper bounds but not buffer sizes for both

sessions, by $a_R^{(1)} < a_R^{(2)}$ and $a_R^{(1)} + a_R^{(2)} = \frac{1}{2}$. Based on Algorithm 1, the achieved average end-to-end data rates \bar{r}_1 and \bar{r}_2 are exactly $a_R^{(1)}$ and $a_R^{(2)}$, respectively. This shows that session 2 with stronger social tie with relay R is able to achieve higher throughput utility.

To illustrate proposition 2, suppose node R differentiates buffer sizes by $q_R^{(1)} > q_R^{(2)}$, but sets the same relay rate bounds $a_R^{(1)} = a_R^{(2)} = \frac{1}{4}$. Based on the back-pressure joint routing and channel allocation in Algorithm 1, the achieved average end-to-end rates are identical for both sessions. The average end-to-end delays are different, which are equivalent to the queuing delays in the respective buffers at R if we ignore transmission delays on the links. Since session 2 has smaller average queue length (analysis in [12]) but identical queue outgoing rate with session 1, its average queuing delay is smaller. We will further validate the performance impact of social selfishness under realistic network settings in Sec. VI.

VI. EMPIRICAL STUDY

We evaluate the throughput utility of our protocol and the impact of social relationships with discrete-event simulations under realistic settings. A cognitive radio network is simulated with several typical settings of the number of primary users (channels), the number of secondary users, and average number of neighbors per secondary user in its transmission range: 3, 10, 5; 5, 20, 5; 7, 50, 8; and 10, 100, 8. A primary user is active in a time slot following a Poisson distribution, with an average probability of 0.2. Sources and destinations of sessions are randomly chosen in the network. Data are injected at the sources following Poisson arrivals with average arrival rates $\bar{A}_m(t)$. The average relay rate for session m on user n is decided by $z_n^{(m)} = z_n \frac{\rho_n^{(m)}}{\sum_{m' \in \mathcal{M}, n \neq s_{m'}, n \neq d_{m'}} \rho_n^{(m')}}$ with $z_n = 0.5$ and $\rho_n^{(m)} = \theta \rho_{n,s_m} + (1 - \theta) \rho_{n,d_m}$, $\theta = 0.5$. In our default settings, the utility function is $U(\bar{r}_m) = \bar{r}_m$, $V = 2000$.

We set the social relationships among secondary users as follows: We first construct a social graph following a power law distribution of node degree with scaling exponent $k = 1.76$ [14], which is derived from the Bluetooth contact traces provided in [19]. Then we assign weight ρ_{ij} to the links in the social graph in three ways:

1) **Homogenous Social Relationship (HSR)**: The social relationship strengths are all the same, *i.e.*, $\rho_{ij} = 1$ for each pair of nodes with direct link in the social graph.

2) **Random Social Relationship (RSR)**: The social relationship ρ_{ij} on the links are uniformly randomly assigned values between $(0, 1]$.

3) **Social Relationship derived from the Traces (TSR)**: We calculate contact frequencies between devices from the traces in [19], normalize them to values within $(0, 1]$, and set social relationship ρ_{ij} between two nodes in our social graph following the distribution of normalized contact frequencies.

Note that in all three cases, users i and j without a direct link in the social graph is assigned a social relationship $\rho_{ij} = 0$. We compare the impact of the three social relationship patterns in the following experiments.

A. Utility with Centralized and Distributed Implementation

We compare the total throughput utility achieved on time averaged end-to-end rate (*i.e.*, $\sum_{m \in \mathcal{M}} U(\bar{r}_m)$), in cases where the centralized and distributed channel allocation algorithms stated in Algorithm 1 and 2 are used, respectively. We investigate a network with 10 secondary users, 5 sessions, and 3 channels as shown in Fig. 2(a), and a network with 20 secondary users, 10 sessions, and 5 channels as in Fig. 2(b), respectively. The average end-to-end rate is calculated after $t = 10,000$ rounds of algorithm execution. The buffer sizes are set following $q_n^{(m)} \propto (1/\rho_n^{(m)})$, with an average size of 10 data units. The average data arrival rates are calculated as $\frac{1}{t} \sum_{\tau=1}^t A_m(\tau)$. We can observe that the total utility achieved with the distributed algorithm is close to that achieved by the centralized one, under each social relationship distribution.

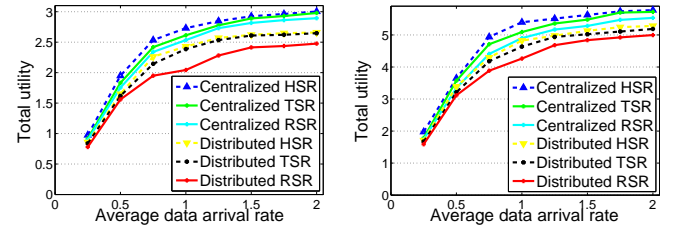


Fig. 2. Centralized vs. Distributed Algorithm on Utility.

B. Impact of Social Relationship between Sessions and Relays

We compare utility and end-to-end delay experienced by sessions with different levels of social ties with relays, in a network with 50 secondary users, 20 sessions, and 7 channels, and a network with 100 secondary users, 40 sessions, and 10 channels, respectively. The average data arrival rate of each session is 1.5 units of data per time slot. The buffer sizes are set following $q_n^{(m)} \propto (1/\rho_n^{(m)})$, with an average size of 10 data units. The end-to-end delay experienced by each session is calculated as the average number of time slots taken for a unit of data to travel from the source to the destination. We use social relationships from the traces in these experiments.

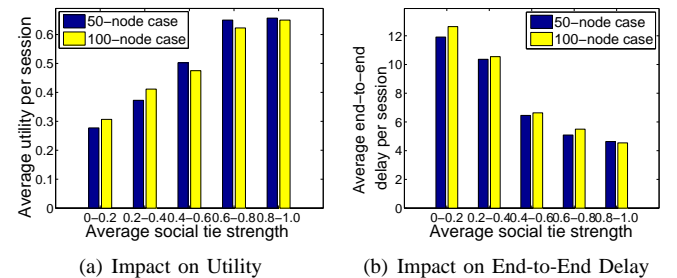


Fig. 3. Performance of sessions with different social tie strengths with relays.

We calculate the average social tie strength for each session between itself and relay nodes, group sessions based on their average social tie strength, and compare the average utility and end-to-end delay per session among different groups in Fig. 3. We observe that sessions of stronger social relations achieve higher throughput utility and lower end-to-end delay. This verifies the effectiveness of our buffer size and relay rate differentiation methods, as stated in propositions 1 and 2 in Sec. V.

C. Impact of Buffer Size Differentiation Methods

We explore the utility and end-to-end delay experienced by different sessions when the following two buffer size differentiation methods are applied: (1) buffer size at each node is proportional to its strength of social relationship with a by-passing session m , i.e., $q_n^{(m)} \propto \rho_n^{(m)}$, with an average size of 10 data units; (2) buffer size is inverse proportional to its strength of social relationship with a session, i.e., $q_n^{(m)} \propto (1/\rho_n^{(m)})$, with an average size of 10 data units. In this set of experiments, there are 100 secondary users, 40 sessions, and 10 channels in the network. The average data arrival rate of each session is 1.5 units of data per time slot.

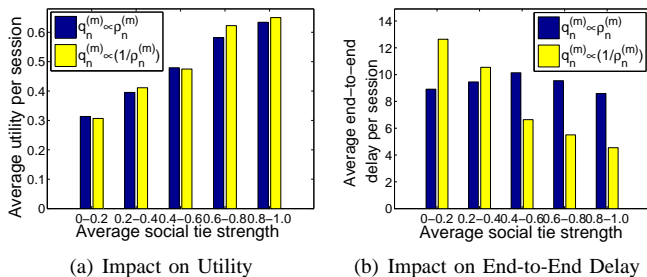


Fig. 4. Performance with different buffer size differentiation methods.

In Fig. 4, we observe that with both ways of differentiating buffer sizes, the average utility per session achieved is similar, and increases with the increase of the sessions' social tie strength. On the other hand, if the buffer sizes are smaller when the social ties are stronger ($q_n^{(m)} \propto (1/\rho_n^{(m)})$), lower end-to-end delay is experienced by sessions with stronger social ties with relays; if the buffer sizes are proportional to the social tie strengths ($q_n^{(m)} \propto \rho_n^{(m)}$), delay experienced by sessions with stronger social ties is not apparently smaller. Therefore, setting smaller buffer sizes for sessions with better social tie strengths is a more suitable way to reflect users' social preference, validating our analysis in Sec. V.

VII. CONCLUDING REMARKS

This paper addresses throughput utility maximization among multiple unicast sessions in a cognitive radio network under the constraint of social selfishness of the participants. A joint end-to-end rate control, routing, and channel allocation protocol as well as its distributed implementation are proposed that can achieve utility optimality with network stability guarantee using Lyapunov optimization techniques. We novelly model social selfishness of users via differentiated buffer sizes and relay rates allocated at each relay node for data sessions. The differentiation is based on the different social relationships the relay has with the sources and destinations of those data sessions. Another unique contribution of our Lyapunov optimization is that only finite buffer is needed at each node with no-buffer-overflow guarantee, which is in sharp contrast to the common assumption of infinite buffers in the literature. The utility optimality and the impact of social selfishness are studied with both rigorous theoretical analysis and extensive simulations. It will be our future work to investigate the impact of social selfishness on data dissemination in other types of wireless networks.

ACKNOWLEDGEMENTS

This project is partially supported by Hong Kong RGC GRF grants 714009E and 714311.

REFERENCES

- [1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey. *Computer Networks*, 50(13):2127 – 2159, 2006.
- [2] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. A Survey on Spectrum Management in Cognitive Radio Networks. *IEEE Communications Magazine*, 08:40–48, April 2008.
- [3] L. Ding, T. Melodia, S. Batalama, and M. J. Medley. ROSA: Distributed Joint Routing and Dynamic Spectrum Allocation in Cognitive Radio Ad Hoc Networks. In *Proc. of ACM MSWiM'09*, 2009.
- [4] L. Ding, T. Melodia, S. N. Batalama, and J. D. Matyas. Distributed routing, relay selection, and spectrum allocation in cognitive and cooperative ad hoc networks. In *Proc. of IEEE SECON'10*, 2010.
- [5] FCC. Notice of proposed rule making and order. *ET Docket No. 03-222*, Dec. 2003.
- [6] Z. Feng and Y. Yang. Joint transport, routing and spectrum sharing optimization for wireless networks with frequency-agile radios. In *Proc. of IEEE INFOCOM'09*, 2009.
- [7] S. Haykin. Cognitive Radio: Brain-Empowered Wireless Communications. *IEEE Journal on Selected Areas in Communications*, 23(2):201 – 220, February 2005.
- [8] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Growcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *Proc. of ACM SIGCOMM'05 Workshops*, 2005.
- [9] J. J. Jaramillo and R. Srikant. Darwin: Distributed and adaptive reputation mechanism for wireless ad-hoc networks. In *Proc. of ACM MOBICOM'07*, 2007.
- [10] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14:699–719, 1966.
- [11] L. B. Le, E. Modiano, and N. B. Shroff. Optimal control of wireless networks with finite buffers. In *Proc. of IEEE INFOCOM'10*, 2010.
- [12] H. Li, W. Huang, C. Wu, Z. Li, and F. C. M. Lau. Utility-maximizing data dissemination in socially selfish cognitive radio networks. Technical report, The University of Hong Kong, <http://i.cs.hku.hk/~hxli/social-cog.pdf>, 2011.
- [13] Q. Li, S. Zhu, and G. Cao. Routing in Socially Selfish Delay Tolerant Networks. In *Proc. of IEEE INFOCOM'10*, 2010.
- [14] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and Analysis of Online Social Networks. In *Proc. of IMC*, 2007.
- [15] M. Motani, V. Srinivasan, and P. Nuggehalli. Peoplenet: Engineering a wireless virtual social network. In *Proc. of ACM MOBICOM'05*, 2005.
- [16] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan&Claypool Publishers, 2010.
- [17] M. J. Neely. Opportunistic Scheduling with Worst Case Delay Guarantees in Single and Multi-Hop Networks. In *Proc. of IEEE INFOCOM'11*, 2011.
- [18] G. Sharma, R. R. Mazumdar, and N. B. Shroff. On the complexity of scheduling in wireless networks. In *Proc. of ACM MOBICOM'06*, 2006.
- [19] V. Srinivasan, A. Natarajan, and M. Motani. CRAWDAD data set nus/bluetooth (v. 2007-09-03). Downloaded from <http://crawdadd.cs.dartmouth.edu/nus/bluetooth>, Sept. 2007.
- [20] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Trans. Autom. Control*, 37:1936–1948, Dec. 1992.
- [21] V. J. Venkataramanan, X. Lin, L. Ying, and S. Shakkottai. On scheduling for minimizing end-to-end buffer usage over multihop wireless networks. In *Proc. of IEEE INFOCOM 2010*, 2010.
- [22] B. Wang, Y. Wu, and K. J. R. Liu. Game theory for cognitive radio networks: An overview. *Computer Networks*, 54:2537–2561, Oct. 2010.
- [23] D. Xue and E. Ekici. Guaranteed opportunistic scheduling in multi-hop cognitive radio networks. In *Proc. of IEEE INFOCOM'11*, 2011.
- [24] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *Proc. of IEEE INFOCOM'03*, 2003.