Two Phase Spectrum Sharing for Frequency-Agile Radio Networks

Zhenhua Feng and Yaling Yang
Department of Electrical and Computer Engineering
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24060
Email: {zhenhua, yyang8}@vt.edu

Abstract—Modern frequency-agile radios are capable of dynamically changing the spectrum width and central frequency of its channels. Existing spectrum sharing algorithms often fail to exploit this characteristic to realize efficient spectrum utilization. In this paper, we present a theoretical framework that capitalize on the frequency agility of modern radios. We solve a joint spectrum sharing and end-to-end rate control problems for general wireless networks to achieve optimal spectrum efficiency with regard to network utility. Analytical and simulation results show the effectiveness of our design.

Index Terms—Spectrum Sharing, Congestion Control, cross-layer design, frequency-agile radio.

I. INTRODUCTION

Recent advances in frequency-agile radio technologies (e.g. software defined radio and cognitive radios) enable more flexible spectrum access through spectrum sensing and dynamic reconfiguration of the central frequency of communication channels. Recent development further shows that the amount/width of spectrum band of radio channels can also be configured dynamically. For example, in WiMax networks [4], users are allowed to use channel bandwidths that are multiples of 1.25MHz, 1.5MHz and 1.75MHz. In the 802.11n standard [5], channel bonding allows radios to form wider channel by bonding continuous smaller channels. In a recently published paper [6], a prototype radio that is capable of transmitting in four channel width of 5, 10, 20, and 40 MHz is developed from commodity Atheros-based Network Interface Card. In Virginia Tech, a prototype Multiband/Multimode Radio (MMR) is developed for public safety applications [7].

Despite the flexibility enabled by modern radios, most existing spectrum allocation works are still based on fixed channelization assumption where the center frequency and the bandwidth of wireless channels are predefined and programmed into radio firmware. This assumption of fixed channelization stems from traditional radio designs, where a radio is only designed for a specific communication system, such as a cellular system, a GPS system, a TV broadcasting system or a WLAN. Devices used in these systems work in predefined and stable spectrum ranges. Since the boundary of the available spectrum range is known to both system designers and device manufacturers, fixed channelization of this predefined spectrum range is appropriate for these systems. A pre-calculated channelization of the available spectrum and a carefully designed channel allocation scheme, e.g. [2][3], can hence be used to optimize the network performance.

However, in dynamic spectrum access networks, the boundary and characteristics of the available spectrum are known to neither system designers nor device manufacturers. Depending on the activities of primary users, a spectrum whitespace may be located in any spectrum range with any size. Therefore, a spectrum access scheme based on predefined channelization is often not flexible enough and leads to low spectrum efficiency [1]. The focus of this paper, hence, is to address this technical void by designing a practical channel spectrum adaptation algorithm to exploit the vastly improvable spectrum utilization by reconfiguring both channel spectrum width and central frequency.

Different from classic frequency scheduling in multichannel systems where only the channel number/central frequency of a radio is determined, spectrum sharing among frequency-agile radios requires reconfiguring both their channel spectrum width and central frequency. The challenge of our work comes from the fact that optimal channel allocation for general ad-hoc wireless networks is itself a difficult (generally NP-hard) problem widely researched in recent years. The joint allocation of channel width and frequency only add to the complexity. Hence, to obtain a computationally tractable solution, we resort to a divide-and-conquer approach. We divide the spectrum sharing problem into two subproblems: channel width allocation and link layer frequency scheduling. We model the channel width allocation (Phase I) as a joint optimization problem and derive a distributed algorithm (JSSRC) through dual decomposition. For the link layer frequency scheduling subproblem (Phase II), we propose an innovative timing window based spectrum access scheme (TWSR) that approximate the optimal spectrum width obtained in JSSRC.

The two algorithms JSSRC and TWSR together provide a systematic treatment for the inherently cross-layer problem of spectrum sharing in wireless networks and comprise the main contribution of this work. Moreover, the derivation of the algorithms reveals some of the most fundamental parameters that are critical to network performance and spectrum efficiency, e.g., local buffer saturation level, wireless link capacity saturation level, number of contenders for the same spectrum band and number of links a specific link can interfere.

The rest of this paper is organized as follows. Section II describes the system model. In Section III, we state the objective of the spectrum width allocation as an optimization problem. Section IV introduces the dual decomposition based framework for phase I algorithm design. We propose and
provide an extensive study of the JSSRC Phase II algorithm TWSR in Section V. Simulation results are presented and discussed in Section VI. Finally, we give concluding remarks in Section VII.

II. SYSTEM MODEL

We consider a general wireless network modeled as a directed graph \( G(V, E) \), where \( V \) and \( E \) denote the set of nodes and the set of directed edges respectively. For two nodes \( i, j \in V \), an edge/link \( l \) from \( i \) to \( j \) exists only if node \( j \) can successfully receive and decode signals from \( i \), i.e., \( j \) is in the transmission range of node \( i \).

We use protocol interference model in our work. Under this model, link \( l \) can cause interference to link \( l' \) if and only if the receiver of link \( l \) is in the interference range of the transmitter of link \( l' \). The set of all links that can cause interference to link \( l \) is denoted as \( E^l(l) \).

There are a set \( S \) of active flow sessions in the network. Each flow \( s \in S \) has a session rate of \( y_s \). The set of links on the path of a session \( s \in S \) is denoted as \( L(s) \) and the set of sessions that are using link \( l \) is denoted as \( S(l) = \{s|l \in L(s)\} \). The set of all active links, denoted as \( L \), can be represented by the union of \( L(s) \), i.e., \( L = \bigcup_{s \in S} L(s) \). A link \( l \) is active if and only if \( l \in L(s) \) for some \( s \), i.e., \( l \) is on the path of a session \( s \). \( L \) is a subset of \( E \).

We define spectrum sharing policy as a set of quantities \( \{(b_{li}, \omega_{li})\}_{i \in L} \) which specifies the width \( b_{li} \) and central frequency \( \omega_{li} \) of spectrum bands of each link \( l \). The upper bound of the amount of spectrum that can be allocated to a single link is \( b_{ava} \in [0, \infty) \). The upper and lower bound for the central frequency of each link \( l \) is denoted as \( \omega_{l, max} \) and \( \omega_{l, min} \). In DSA networks, \( b_{ava} = [b_{ava}, b_{ava}, ..., b_{ava}] \) corresponds to the spans of the white spaces that are available to secondary users. \( \omega_{l, max} \) and \( \omega_{l, min} \) are the boundary of the white space that are available to link \( l \).

We assume that the transmission power of each radio is determined before the execution of JSSRC algorithm. This means that transmission power remains relatively constant during each execution cycle of JSSRC algorithm and changes at a slower rate than the convergence time of the JSSRC. The direct impact of these assumptions combined with the non-overlapping spectrum allocation policy is that we can treat the signal-to-noise ratio \( SNR_l \) at the receiver of a link \( l \) as a constant during the execution time of JSSRC. Consequently, the capacity of a link \( l \) becomes only the function of link \( l \)'s spectrum width \( b_{li} \), i.e., \( c_l = b_{li} \log(1 + SNR_l) \), where \( SNR_l \) is the signal-to-noise ratio (SNR) at the receiver of link \( l \).

III. PROBLEM STATEMENTS OF PHASE I ALGORITHM

The objective of Phase I algorithm (JSSRC) is to distributively compute optimal spectrum width allocation for each link. This width allocation maximizes aggregate network utility if centralized algorithm are available. We associates each session \( s \in S \) with a utility function \( U_s(y_s) \). The objective of JSSRC then can be summarized as a maximization problem as

\[
\max_{\{y_s\} \in S} \sum_{s \in S} U_s(y_s),
\]

subject to

\[
\sum_{s \in S} y_s \leq c_l \quad \forall l \in L \tag{2}
\]

\[
c_l = b_{li} \log(1 + SNR_l) \quad \forall l \in L \tag{3}
\]

\[
b_l + \sum_{i \in E^l(l)} b_{li} \leq b_{ava} \quad \forall l \in L \tag{4}
\]

\[
b_l \geq 0, c_l \geq 0, y_s \geq 0, \quad \forall l \in L, s \in S \tag{5}
\]

In this formulation, Eq. (2) ensures that the aggregate rate of sessions that traverse link \( l \) does not exceed \( l \)'s link capacity \( c(l) \). Eq. (3) is the capacity constraints derived from information theory. Eq. (4) ensures that any allocation obtained from the formulation is feasible, i.e., there is no spectrum overlap between two active links that are in each other's interference range.

For ease of presentation, we derive a vector form of the primal problem in the following. Let \( b = [b_1,...,b_{|L|}]^T \) and \( c = [c_1,...,c_{|L|}]^T \) denote the spectrum width allocation vector and the link capacity vector respectively. Let \( N \) be a \( |L| \times |L| \) diagonal matrix where \( n_{ll} = \log(1 + SNR_l) \) and \( n_{lk} = 0 \) for \( l \neq k \). Let \( y = [y_1,...,y_{|S|}]^T \) denote the vector of session rates.

Define spectrum contention matrix \( A_{|L| \times |L|} \) as

\[
A_{ll} = \begin{cases} 
1 & l = l' \in E^l(l), \text{i.e., } l' \text{ can produce interference to link } l \text{'s receiver;} \\
0 & \text{otherwise.}
\end{cases}
\]

Define routing matrix \( R \) as

\[
R_{ls} = \begin{cases} 
1 & \text{link } l \in L \text{ is on the path of session } s \in S; \\
0 & \text{otherwise.}
\end{cases}
\]

We can obtain an equivalent but more compact form of the primal problem:

\[
\max_{y, b} \quad U(y) = \sum_{s} U(y_s) \tag{6}
\]

subject to

\[
R y \leq c \tag{7}
\]

\[
c = Nb \tag{8}
\]

\[
Ab \leq b_{ava} \tag{9}
\]

\[
y, b, c \leq 0 \tag{10}
\]

We use both the original and the compact formulations interchangeably whenever there is no confusion.

IV. THE DUAL FRAMEWORK OF PHASE I ALGORITHM

Note that in the primal formulation [1-5], the spectrum width allocation and the session rate control are coupled. Moreover, all session rates are coupled in the first constraint Eq. (2). Solving this primal optimization problem requires global information and centralized control of the transport and link layers of all links in a network. The difficulty in gathering all the information makes such centralized approach
impractical in a distributed multihop radio network. Therefore, in this section, we present a dual based approach to solve the problem in a distributed manner.

Consider the dual of the primal problem [1-4]

\[
\min_{u \geq 0} \ D(u) \tag{11}
\]

with partial dual function

\[
D(u) = \max_{b \geq 0} \ U(y) - u^T(Ry - c) \tag{12}
\]

subject to

\[
Ab \leq b^{ava} \tag{13}
\]

\[
c = Nb. \tag{14}
\]

Here, we relax the constraint Ry \(\leq c\) by introducing Lagrange multiplier \(u = [u_1, ..., u_{|L|}]^T\). The maximization problem in the partial dual function can be further decomposed into the following two subproblems

\[
D_1(u) = \max_{y \geq 0} \ U(y) - u^T Ry
\]

and

\[
D_2(u) = \max_{b \geq 0} \ u^T c
\]

subject to

\[
Ab \leq b^{ava}
\]

\[
c = Nb
\]

This decomposition successfully makes the joint spectrum sharing and rate control problem into three separate optimization problems: \(D_1, D_2\) and the master dual problem at Eq. (11). These three optimization problems are correlated through the Lagrangian multiplier \(u\). \(D_1(u)\) is a standard rate control problem which can be solved at the transport layer. \(D_2(u)\) is a link layer resource allocation problem that maximizes the weighted sum of link capacities with the Lagrangian multiplier as the weights. The master dual problem is an optimization problem regarding the Lagrangian multiplier \(u\). Each of these three problems can be solved using distributed iterative algorithms.

A. Solving the subproblem \(D_1\)

To solve \(D_1\), note that \(D_1\) can be rewritten as

\[
D_1(u) = \sum_{s \in S} \left[ \max_{y \geq 0} U_s(y_s) - \sum_{l \in L(s)} u_l y_s \right]
\]

From the above equation, it can be seen that the optimal solution to \(D_1\) can be further decomposed into many small maximization problems that can be executed distributively at each source requiring only local information, i.e., source node of a session only needs to know the summation of \(u_l\) along the session’s path to calculate the optimal rate for this session. For each source, the unique maximizer is computed as

\[
y_s = \arg \max_{y \geq 0} \{U_s(y_s) - \sum_{l \in L(s)} u_l y_s\} \forall s \in S, \tag{15}
\]

B. Solving the subproblem \(D_2\)

The spectrum width allocation sub-problem \(D_2\) is a linear programming problem that determines the optimal amount of spectrum width allocated to each link. By substituting link capacity \(c\) with allocated spectrum width \(b\), the objective function of \(D_2\) is just a weighted sum of the elements of the spectrum width allocation vector \(b\), which is a linear function. The constraints are also linear functions. While solving such a linear programming problem requires only moderate computational power (polynomial time), directly calculating the spectrum width allocation requires global information about the lagrange multiplier \(u\), which is hard to implement in large networks. Hence, we design an efficient and distributed algorithm requiring only local information to solve \(D_2\) as follows.

Again, we use dual decomposition and subgradient method to obtain a distributed algorithm to solve problem \(D_2\). Consider the dual of \(D_2\)

\[
\min_{v \geq 0} \ L(v) \tag{16}
\]

with lagrangian dual function

\[
L(v) = \max_{b \geq 0} \ u^T Nb - v^T(AB - b^{ava}) \tag{17}
\]

It is worth noting that by solving (16), the primal optimal variables are not immediately available. This is because the dual problem may have multiple optimal points with the same dual objective. To remove this complication, we add a small regularization term in \(D_2\), i.e., a small quadratic term \(\sigma b^Tb\), and maximize

\[
u^T Nb - \sigma b^Tb.
\]

Here, \(\sigma\) is a very small positive constant. Subsequently, the dual problem of \(D_2\) changes to

\[
\min_{v \geq 0} \ L(v)
\]

with Lagrangian dual function

\[
L(v) = \max_{b \geq 0} \ u^T Nb - \sigma b^Tb - v^T(AB - b^{ava}) \tag{18}
\]

Using the gradient projection method, we can derive the iterative algorithms to solve the regularized dual problem of \(D_2\) as follows:

\[
b(m + 1) = \frac{1}{2\sigma} (Nu - A^Tv(m))^+
\]

\[
v(m + 1) = [v(m) + \beta (Ab(m + 1) - b^{ava})]^+
\]

where \(\beta\) is a positive stepsize.

The above iterative update algorithms for \(b_l\) and \(v_l\) only requires local information, e.g., \(v_l\) and \(b_l\) of links that can generate interference to link \(l\) and links that link \(l\) can interfere if transmit simultaneously using the same spectrum block.

The convergence of the above iterative algorithm to the optimal solution of \(D_2\) is well established [13]. Define \(\tilde{a}_1 = max_{l \in L} |E(l)|\). Define \(\tilde{a}_2\) as the largest number of constraints (inequalities) in (4) that any \(b_l\) may be involved in. The range of the stepsize with which the algorithm converges can be defined as in [12]:

\[
0 < \beta < \frac{4\sigma}{\tilde{a}_1 \cdot \tilde{a}_2}
\]
C. Solving the master dual problem

To solve the master dual problem in Eq. (11), we use an iterative algorithm based on the subgradient projection method.

Let \( y(u(n)) \) and \( c(u(n)) \) denote the maximizers of \( D_1 \) and \( D_2 \) given the value of \( u \) at the \( n_{th} \) iteration, i.e.,

\[
y(u(n)) = \arg\max_{y \geq 0} U(y) - u^T Ry \tag{21}
\]

and

\[
c(u(n)) = \arg\max_{b \geq 0} u^T c \quad \text{subject to} \quad Ab \leq b^{ava} \tag{22}
\]

Given \( y(u(n)) \) and \( c(u(n)) \), we can then use the subgradient method to solve the master dual problem in Eq. (11) by iteratively updating the Lagrangian multiplier \( u \) as follows:

\[
u(n + 1) = [u(n) + \gamma (Ry(u(n)) - c(u(n)))]^+ \tag{23}
\]

where \( \gamma \) is a positive scalar stepsize, \( (Ry(u(n)) - c(u(n)) \) is the subgradient of the dual function (Eq. (12)) at the \( n_{th} \) iteration, and \( ^+ \) denotes the projection to the set \( R^+ \) of non-negative real numbers.

The system of Eq. (15) , (19), (20) and (23) forms the mathematical base to design JSSRC Phase I algorithm for solving the dual problem (11).

The pseudo code for JSSRC Phase I algorithm is provided in Algorithm 1.

**Algorithm 1: JSSRC Phase I**

Input : \( G(V,E), A, \mathcal{R} \)

Output : \( \bar{b}, \bar{y} \)

begin

\[
1. \text{Initialize } (b,y) \text{ do}
2. \text{while } ||u(n+1) - u(n)||_2 > \delta_1 \text{ do}
3. \quad \text{Source rate update:}
4. \quad \quad \quad \ y_s \leftarrow [y_s + \alpha (U'_c(y_s) - \sum_{l \in E(l)} v_l)]^+
5. \quad \text{end}
6. \text{while } ||y(n+1) - y(n)|| > \delta_2 \text{ do}
7. \quad \text{Update spectrum width and } v_l \text{ as:}
8. \quad \quad \quad \ b_l \leftarrow \frac{1}{\sigma} (\log(1 + SNR_l)u_l - \sum_{l \in E(l)} v_l A_{il})^+, 
9. \quad \quad \quad \ v_l \leftarrow v_l + \beta \left( \sum_{l \in E(l)} A_{il} b_l - b_l^{ava} \right)^+. 
10. \quad \text{end}
11. \text{Update } u_l \text{ as:}
12. \quad \quad \ u_l \leftarrow u_l + \gamma (\sum_{s \in E(l)} y_s - c_l) 
13. \quad \text{end}
14. \text{output } (\bar{b}, \bar{y})
15. \text{end}
16. *\delta_1, \delta_2, \delta_3 \text{ are corresponding termination thresholds}
end

D. Convergence

If we assume the norm of the subgradients of the dual function is bounded, i.e., there exist a constant \( G \) such that \( ||\sum_{s \in E(l)} (y_s(u(n)) - c_l(u(n)))|| \leq G, \forall n, l. \) the convergence of the algorithm is guaranteed by Proposition 1. The proof is provided in our technical report [15].

**Proposition 1**: Let \( u^* \) denote an optimal spectrum contention price, the iterative solution system of Eq.(15), (19), (20) and (23) converges statistically to within \( \frac{\gamma G^2}{2} \) of the optimal value.

E. Understanding JSSRC Phase I algorithm

Before exploring any further, let us take a closer look at Phase I algorithm to gain some insight on its physical meaning.

By treating \( u_l \) as link price, the subproblem \( D_2(u) \) can be interpreted as maximizing the aggregated link layer utility through allocation of spectrum. Links with higher prices \( u_l \) are likely to be allocated with more spectrum as they generate more revenues. This makes sense since links with higher prices \( u_l \) in our design force those links that have heavier traffic and hence need more spectrum to alleviate its congestion. Due to interference among links in the network, when link \( l \) is allocated with more spectrum, the available spectrum for its neighboring links \( l \in E_l \) is decreased, which may increase the congestion level at link \( l \). Hence, the Lagrangian multiplier \( v_l \) is introduced as the spectrum contention price, which is the price that needs to be paid to link \( l \) for taking a unit of spectrum available to link \( l \) to some other link. In Eq. (20), JSSRC increases the contention price \( v_l \) of link \( l \) when link \( l \)’s local available spectrum \( b_l^{ava} \) (spectrum left by primary users) becomes smaller than the spectrum demands from the links in its neighborhood (s.t. \( \sum_{l \in E_C(i)} A_{il} v_l \leq b_l^{ava} \)), preventing these links in \( C(l) \) from demanding more spectrum from link \( l \). When there is still unallocated available spectrum at link \( l \) (s.t. \( \sum_{l \in E_C(i)} A_{il} v_l \leq b_l^{ava} \)), link \( l \) decreases its contention price \( v_l \) so that other interfering links are encouraged to use this unallocated spectrum.

V. JSSRC PHASE II ALGORITHM

If we assume a centralized spectrum broker, the scheduling problem is trivial. The optimal policy given by phase I algorithm JSSRC is achieved immediately when each node follows the command of the central broker. However, in large networks, such a central spectrum broker is not likely to exist. The links only have limited local information and thus prone to make "bad" decisions on spectrum selection. Hence, an optimal frequency scheduling may not be easily obtained. In the following, we first demonstrate a case where naive scheduling algorithm fails to generate a good policy and then propose a novel distributed Timing Window based Spectrum Sharing (TWSR) algorithm that closely approximate the optimal spectrum sharing policy without requiring centralized spectrum broker.
A. A case study of distributed link layer scheduling

Consider a simple network of 3 nodes in Figure [1]. Suppose there are two active one-hop sessions \( s_1 \) and \( s_2 \) from node 1 to 2 and 3 to 2, respectively. Each session requires 50MHz spectrum to transmit its traffic. Node 1 and 3 each has one radio and node 2 has two radios. Links (1,2) and (3,2) mutually interfere with each other. Suppose there is an available spectrum whitespace of 100MHz left by primary users and the spectrum whitespace starts from 2.4GHz. The optimal spectrum sharing policy should allocate each link a 50MHz wide spectrum. This is only possible if link (3,2) use either the first half block of the spectrum whitespace starting from 2.4GHz or use the second half block starting from 2.45GHz, while the other link uses the remaining half simultaneously. A centralized spectrum broker can easily realize such a spectrum allocation. However, when links make distributed spectrum reservation decision, link (3,2) may first reserve a 50MHz spectrum range starting from 2.425GHz without any trouble. However, this reservation left two 25MHz spectrum fragments and link (1,2) can only use one of the spectrum fragment due to the limited of only one radio at node 1. Hence, with the limitation on radios per link, distributed spectrum reservation may not result in optimal spectrum scheduling.

B. Timing Window based Spectrum Reservation

To address the above issue, we propose the Timing Window based Spectrum Reservation (TWSR) algorithm for link layer frequency scheduling. TWSR works in scenarios where there is only one dedicated radio at a node for each of its link. The spectrum scheduling produced by TWSR approximates the optimal spectrum scheduling.

Based on \( b^* \), TWSR produces an approximate spectrum scheduling \( \tilde{b} \) and \( \tilde{f} \). The design of TWSR is based on the observation that the more constraints in Eq (9) that a link \( l \) belongs to, the more difficult to find an appropriate spectrum fragment with size \( b_l \) for link \( l \). Hence, those links that are included in more number of constraints should be given higher priority in spectrum reservation.

We define the number of constraints link \( l \) belongs to in Eq. (9) as link \( l \)'s constraint degree \( q_l \). After link \( l \) obtains its new spectrum width \( b_l^* \) in Phase I, link \( l \) computes a timing window based on its constraint degree \( q_l \), where the lower bound of the time window is \( w^{lb}_l=\delta/q_l \) and the upper bound is \( w^{ub}_l=\delta/(q_l-1) \). Here, \( \delta \) is a scalar determined by timing granularity of the network. Link \( l \) then randomly picks a time between \( w^{lb}_l \) and \( w^{ub}_l \) to declare to its interfering links its spectrum reservation. The spectrum reserved by link \( l \) is picked as the lowest unreserved spectrum fragment that has size \( b_l \). If such a spectrum fragment does not exist, link \( l \) reserves the largest spectrum fragment available. The spectrum reservation process is a first come first serve process. A link \( l \) only reserves a spectrum fragment that is not previously reserved by its interfering links. This first in first service scheduling and non-overlapping declaration mechanism ensures that links that are more difficult to schedule declare their spectrum reservation earlier and, hence, have more chance to get exactly the spectrum size that they demand.

TWSR itself is a heuristic algorithm falling into the category of greedy packing problems. The spectrum allocation vector \( b \) generated by TWSR is generally a suboptimal version of \( b^* \). However, as shown in section VI, TWSR combined with the Phase I of JSSRC shows very impressive performance. Not only does it generate final results very close to optimal results, it also converges in many of the cases in a relatively short time limit (400 iterations). Note that the spectrum contention price \( u \) at the end of each master iteration (line 14 in Algorithm 1) is updated according to \( b \) generated by TWSR instead of the optimal value \( b^* \). As a result, the update in the master iteration can partially correct the approximation error made by \( b \). In Proposition 2, we give an upperbound on the approximation error of JSSRC with TWSR. It turns out that the combined approximation error of the iterative algorithm is upper bounded by how close the suboptimal schedule \( b \) are to the optimal schedule \( b^* \).

Proposition 2: Suppose at each iteration \( n \), the norm of the error between suboptimal \( b(u(n)) \) and \( b(u(n)) \) is bounded, i.e., \( ||b(u(n)) - b(u(n))||_2 \leq \epsilon \), then under the same assumption as in Theorem 1, JSSRC converges statistically to within \( \gamma \frac{\epsilon^2}{2} + \log(1 + SNR_{max}) \) of the optimal, where \( SNR_{max} = \max_i \{SNR_i\} \).

Proof: The proof is provided in our technical report [15].

VI. Numerical Results

To evaluate the performance of JSSRC, we run extensive simulations of JSSRC with TWSR in different topologies and network sizes. The number of sessions in a simulation ranges from 2 to 8. In all scenarios a node can interfere with any of its one-hop neighbors. In the simulation, we set step size for both master iterations (line 15 in Algorithm 1) and secondary iterations (line 6 and 11 in Algorithm 1) to 0.01. We compare JSSRC with TWSR to the optimal solution obtained by JSSRC with centralized algorithm (perfect scheduling).

We first show two representative scenarios to characterize the behavior of JSSRC.

In scenario 1, there are 4 active flow sessions in a 15 nodes network. We can see from Fig. [4], the convergence curve of JSSRC with TWSR shows big oscillation at the early iterations, then gradually smooths out and finally converges to the optimal. According to Proposition 2, this indicates that TWSR generates close to optimal schedules after enough number of iterations. JSSRC with TWSR also shows convergence speed as good as the JSSRC with unlimited radios per link (s.t. running only Phase I of JSSRC), which demonstrates the effectiveness of our design.

In scenario 2, as shown in Fig. [3], there are 8 active sessions in a 16-node network. This network is considered a dense network with heavy load of traffic. The spectrum contention matrix is complex. From the convergence curve, we can see that JSSRC with TWSR still performs fairly well. It closely matches with the optimal curve. The final error is also well bounded from and very close to the optimal. The oscillation throughout the curve shows that TWSR has
difficulty in rendering the ultimate optimal scheduling. Instead, it oscillates around the optimal scheduling with a set of suboptimal scheduling solutions. To stop the oscillation we can either simply set a maximum number for iterations or add a stabilize in the dual objective function to stabilize the network.

Other results are summarized in Table I which shows the number of iterations for JSSRC-TWSR to converge and the error of JSSRC-TWSR, which is the gap between the stable point of JSSRC-TWSR and the optimal. We test the performance of JSSRC with 10 different settings of topologies and network sizes. The collective results of JSSRC with TWSR is very promising. The average error compared to the optimal is only 1.5%. According to Theorem 2, this error can be made smaller by improve the scheduling algorithm.

### Table I

<table>
<thead>
<tr>
<th>Group #</th>
<th>TWSR Aggregate Utility</th>
<th>TWSR No. of iterations</th>
<th>Optimal Aggregate Utility</th>
<th>Optimal No. of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.2209</td>
<td>159</td>
<td>3.2561</td>
<td>130</td>
</tr>
<tr>
<td>2</td>
<td>2.1285</td>
<td>223</td>
<td>2.1287</td>
<td>223</td>
</tr>
<tr>
<td>3</td>
<td>2.6354</td>
<td>192</td>
<td>2.6355</td>
<td>192</td>
</tr>
<tr>
<td>4</td>
<td>3.2027</td>
<td>400*</td>
<td>3.2561</td>
<td>368</td>
</tr>
<tr>
<td>5</td>
<td>1.2087</td>
<td>225</td>
<td>1.2087</td>
<td>209</td>
</tr>
<tr>
<td>6</td>
<td>2.3027</td>
<td>400*</td>
<td>2.3835</td>
<td>308</td>
</tr>
<tr>
<td>7</td>
<td>3.3049</td>
<td>342</td>
<td>3.3053</td>
<td>283</td>
</tr>
<tr>
<td>8</td>
<td>3.1342</td>
<td>400*</td>
<td>3.1349</td>
<td>294</td>
</tr>
<tr>
<td>9</td>
<td>3.7338</td>
<td>320</td>
<td>3.7344</td>
<td>302</td>
</tr>
<tr>
<td>10</td>
<td>0.7838</td>
<td>400*</td>
<td>0.7986</td>
<td>400*</td>
</tr>
</tbody>
</table>

### VII. CONCLUSION

We present a distributed algorithm JSSRC for joint spectrum sharing and end-to-end rate control for multihop cognitive radio networks. By taking advantage of cognitive radio's capability to dynamically reconfigure central frequency and channel width, JSSRC with perfect link layer scheduling is able to achieve the optimal spectrum sharing policy that maximizes transport layer utility. For practical implementation, we present a novel timing window based spectrum reservation scheme (TWSR) for link layer scheduling. Simulation results show that JSSRC algorithm with TWSR achieves good performance despite TWSR's heuristic nature. As a preliminary step towards a systematic approach to spectrum sharing among frequency-agile radio network, JSSRC can be extended in various directions.

First, to accommodate time-varying channels, a stochastic channel model can be used to describe the network. For example, if we assume the channel state is described by a finite state Markov Chain, JSSRC can be straightforwardly extended to maximize network utility over the average channel conditions. Second, JSSRC can be extended to work asynchronously. Actually, the dual decomposition is well suited for asynchronous design. Convergence and optimality analysis for asynchronous algorithm is at the center of this extension. In the future, we also plan to set up a DSA testbed to investigate practical aspects of JSSRC.

### ACKNOWLEDGMENT

This work was supported in part by the Institute for Critical Technology and Applied Science (ICTAS) and National Science Foundation Grant 0831865.

### REFERENCES