Optimal Channel Assignment with Aggregation in Multi-channel Systems: A Resilient Approach to Adjacent-channel Interference

Gulnur Selda Uyanik[a], Mohammad J. Abdel-Rahman[b,*], Marwan Krunz[b]

[a]Department of Computer Engineering, Faculty of Computer and Informatics, Istanbul Technical University, Maslak, Istanbul, 34469, Turkey

[b]Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721, USA

Abstract

Channel assignment mechanisms in multi-channel wireless networks are often designed without accounting for adjacent-channel interference (ACI). To prevent such interference between different users in a network, guard-bands (GBs) are needed. Introducing GBs has significant impact on spectrum efficiency. In this paper, we present channel assignment mechanisms that aim at maximizing the spectrum efficiency. More specifically, these mechanisms attempt to minimize the amount of additional GB-related spectrum that is needed to accommodate a new link. **Similar to the IEEE 802.11n and the upcoming IEEE 802.11ac standards, our assignment mechanisms support channel bonding, and more generally, channel aggregation.** We first consider sequential assignment (i.e., one link at a time), and we formulate the optimal ACI-aware channel assignment that maximizes the spectrum efficiency as a subset-sum problem. An exact exponential-time dynamic programming (DP) algorithm, a polynomial-time greedy heuristic, and an ϵ-approximation are **presented** and compared. Second, considering a set of links (batch assignment), we derive the optimal ACI-aware exponential-

*Corresponding author

Email addresses: `seldauyanik@itu.edu.tr` (Gulnur Selda Uyanik), `mjabdelrahman@email.arizona.edu` (Mohammad J. Abdel-Rahman), `krunz@email.arizona.edu` (Marwan Krunz)

A preliminary version of this paper will be presented at the IEEE GLOBECOM 2013 Conference.

time assignment that maximizes the *network's* spectrum efficiency. The optimal batch assignment is compared with the sequential assignment. Results reveal that our proposed algorithms achieve considerable improvement in spectrum efficiency compared to previously proposed schemes.

Keywords: Channel assignment, dynamic programming, ϵ-approximate algorithms, greedy algorithms, integer programming, multiple subset-sum problem, spectrum efficiency, subset-sum problem.

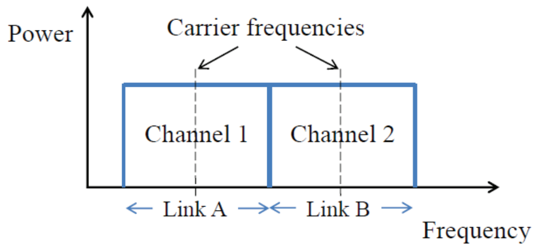
---

1. Introduction

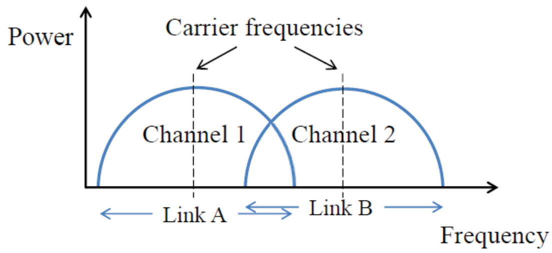
Adjacent-channel interference (ACI) is a form of power leakage that is attributed to imperfect filters and amplifiers in the radio device. The harmful impact of ACI on the throughput of IEEE 802.11a and IEEE 802.11n networks was demonstrated in [1] and [2], respectively. Most channel assignment algorithms in the literature do not account for ACI (see Figure 1(a)). Figure 1(b) shows the actual power spectral density of two channels in a practical communication system. To mitigate ACI, *guard-bands* (GBs) are needed between adjacent channels that belong to different links.

However, introducing GBs constrains the spectrum efficiency. In [3], the authors studied two models for utilizing GBs in a dynamic spectrum access (DSA) network: "GB reuse" and "no GB reuse". According to the "GB reuse" model, GBs can be shared by two different (interfering) links. In contrast, in the "no GB reuse" model, two adjacent transmissions require their own GBs. As explained in [3], the "GB reuse" model is suitable for discontinuous-OFDM (D-OFDM) systems, whereas the "no GB reuse" model is suitable for FDM-based systems. In this paper, we adopt the "GB reuse" model. This model is illustrated in Figure 1(c), where the same amount of GB is allocated between channels 1 and 2, irrespective of whether link B is active or not over channel 2. As shown later in this paper, the GB-aware (GBA) channel assignment algorithm in [3] for the "GB reuse" case does not achieve the maximum spectrum efficiency.

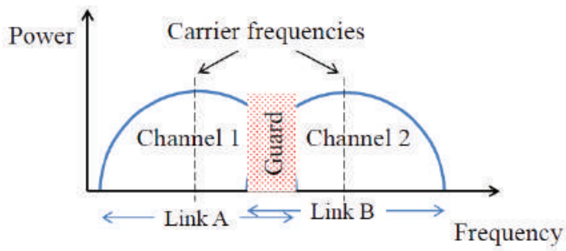
To support applications with high rate demands, the IEEE 802.11n and the upcoming IEEE 802.11ac standards have adopted the concept of channel bonding [4–8]. **This concept refers to the bundling of multiple adjacent channels, which can then be treated as a single frequency block whose data rate is approximately the sum of the data rates of the individual channels. By bonding two 20-MHz**



(a)



(b)



(c)

Figure 1: GBA channel assignment. (a) Ideal power spectral density, (b) power spectral density in a practical communication system, and (c) power spectral density under the "GB reuse" model.

channels, IEEE 802.11n supports a single 40 MHz channel [9]. In traditional single-input single-output (SISO) systems (e.g., IEEE 802.11a/b/g), channel bonding causes a reduction in the transmission range and a greater susceptibility to interference [10, 11]. However, with the incorporation of MIMO technology in IEEE 802.11n devices, the problems faced by SISO systems due to channel bonding can now be mitigated [12, 13]. In [5, 6], the authors conducted experimental studies in the 5 GHz band to characterize the behavior of channel bonding. They observed that ACI needs to be mitigated in order to perform intelligent channel bonding. The IEEE 802.11ac standard enhances the throughput beyond the IEEE 802.11n using an 80 MHz channel bonding technique [7, 8].

The concept of channel bonding can be extended to non-adjacent channels, and is referred to as *channel aggregation.* **For example, LTE-Advanced employs channel aggregation techniques, allowing 4G mobile operators to aggregate spectrum from non-adjacent bands to support links with high demands [14]. With channel aggregation, LTE-Advanced supports up to 100 MHz system bandwidth, with the potential of achieving more than 1 Gbps throughput for downlink and 500 Mbps throughput for uplink [15]. Implementation challenges of channel aggregation have been studied in [15, 16]. Recently, distributed channel aggregation has been studied in [17–19] in a game theoretic framework. The proposed schemes in [17–19] do not account for ACI. Although co-channel interference has been extensively studied in the context of distributed channel allocation [20, 21], ACI has been largely overlooked.**

Main Contributions–The main contributions of the paper are as follows:

1. We formulate and obtain the optimal (sequential) GBA channel assignment for a single link, adopting the "GB reuse" setting. The per-link channel assignment problem is formulated as a subset-sum problem (SSP) [22]. An exact exponential-time dynamic programming (DP) algorithm, a polynomial-time greedy heuristic, and an ϵ-approximation are **presented**.
2. We formulate and obtain the optimal GBA channel assignment for multiple links (batch approach), under the "GB reuse" setting.
3. We evaluate the exponential-time optimal sequential and batch assignment mechanisms and compare them with polynomial-time heuristics and

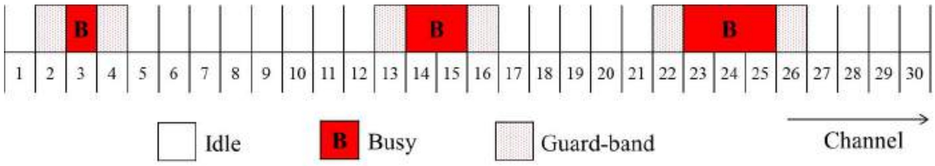


Figure 2: Spectrum status (channel assignment) at a given time instance.

ϵ-optimal approximations.

Paper Organization–The remainder of this paper is organized as follows. In Section 2, we present the system model followed by the problem statement. The single-link optimal channel assignment is explained in Section 3. Polynomial-time greedy and ϵ-approximate algorithms are also presented in the same section. In Section 4, we address the problem of optimal GBA channel assignment for multiple links. We provide an exponential-time exact algorithm along with an approximate sequential algorithm. We evaluate our assignment algorithms in Section 5. Section 6 gives an overview of related work. We provide directions for future research in Section 7. Finally, Section 8 concludes the paper.

2. Problem Statement

We consider a single-hop wireless network with a set of channels $\mathcal{M} = \{1, 2, \ldots, M\}$ and a set of links $\mathcal{L} = \{1, 2, \ldots, L\}$. Without loss of generality, we assume all channels to have the same bandwidth, denoted by W (in Hz). An available (unassigned) channel can be reserved as a GB, or assigned for data communication. All available channels support a common rate of r Mbps. **In Section 7, we provide directions for extending our work to a multi-rate setup**. Each link $j \in \mathcal{L}$ has a rate demand $d_j \stackrel{\text{def}}{=} \alpha_j r$ Mbps, where α_j is an integer between 1 and M. Given the current spectrum status, our objective is to satisfy the demands of one or more links in $\mathcal{L}$ while maximizing the spectrum efficiency (defined shortly). Figure 2 shows an example of a spectrum status.

The spectrum efficiency is defined as the fraction of the available spectrum that can be used for data communications. Let $h_{ij}, i \in \mathcal{M}$ and $j \in \mathcal{L}$, be a binary variable; $h_{ij} = 1$ if channel i is assigned to link j as a data channel, and zero otherwise. Let η_i be a binary variable indicating whether or not the ith channel is to be used as a

GB channel. Then, the network-wide spectrum efficiency, denoted by $\xi_{\mathbf{net}}$, is defined as follows:

$$\xi_{\mathbf{net}} \stackrel{\mathbf{def}}{=} \frac{\sum_{j=1}^{L}\sum_{i=1}^{M} h_{ij}}{\sum_{j=1}^{L}\sum_{i=1}^{M} h_{ij} + \sum_{i=1}^{M}\eta_i}. \tag{1}$$

Similarly, the per-link spectrum efficiency, denoted by $\xi_{\mathbf{link}}$, is defined as:

$$\xi_{\mathbf{link}} \stackrel{\mathbf{def}}{=} \frac{\sum_{i=1}^{M} h_i}{\sum_{i=1}^{M} h_i + \sum_{i=1}^{M}\eta_i} \tag{2}$$

where h_i is a binary variable that indicates whether or not channel i is assigned for data communication.

In this paper, we consider the following two problems.

Problem 1. Given an arbitrary link with a rate demand $d = \alpha r$ Mbps and given the current status of the M channels, find the optimal GBA channel assignment for this link **that maximizes $\xi_{\mathbf{link}}$ while satisfying the demand d.**

Problem 2. Given the set of links $\mathcal{L}$ and their associated rate demands, and given the current status of the M channels, find the optimal GBA channel assignment **that maximizes $\xi_{\mathbf{net}}$ while satisfying the link demands.**

The proposed assignment schemes support channel bonding and aggregation.

3. Optimal GBA Channel Assignment for a Single Link

Consider Problem 1. ξ_{link} in (2) can also be expressed as:

$$\xi_{\text{link}} = \frac{\alpha}{\alpha + \sum_{i=1}^{M}\eta_i}. \tag{3}$$

Equation (3) holds assuming the problem is feasible, i.e., there is a feasible assignment that can satisfy the link demand d. According to (3), in order to maximize ξ_{link}, the number of introduced GBs (i.e., $\sum_{i=1}^{M}\eta_i$) needs to be minimized.

Consider the spectrum status in Figure 2. Each set of consecutive idle channels is grouped into a "frequency block," as illustrated in Figure 3. Let $\mathcal{N}$ denote the set of idle frequency blocks, and let $N = |\mathcal{N}|$. Let $R_i \stackrel{\text{def}}{=} \beta_i r$ Mbps denote the rate supported by the ith block (IB_i), where β_i is an

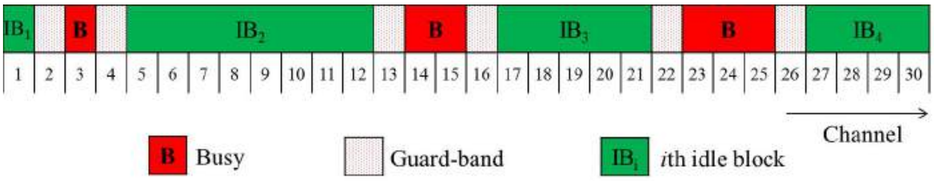


Figure 3: Set of idle blocks for the spectrum map in Figure 2.

integer between 1 and M. As justified in [23], we assume that one fixed-bandwidth GB channel on each side of a data transmission block is sufficient to prevent ACI, irrespective of the block size. This assumption is motivated by the results in [23], which showed that the main source of interference to any demodulated subcarrier are the nearest subcarriers of a neighboring frequency block that is assigned to another transmission. We remark that, in general, the difference in the transmission powers of two frequency-adjacent links impacts the required amount of GB between them. However, in this paper, we assume that this power difference is small, and one GB on each side of the frequency block is sufficient to prevent ACI. Next, we show that in order to minimize the number of introduced GBs (i.e., $\sum_{i=1}^{M}\eta_i$) and hence maximize ξ_{link}, channels need to be assigned on a per-block basis.

Theorem 1. Assigning channels on a per-block basis achieves the optimal **per-link spectrum efficiency.**

Proof. We show that assigning channels on a per-block basis introduces at most one additional GB. Consider the set of idle blocks $\mathcal{N}$. There are two cases to consider:

Case 1: $\exists \mathcal{B} \subseteq \mathcal{N}$ such that $\sum_{i\in\mathcal{B}} R_i = d$. This case is exemplified in Figure 4, where $d = 6$ Mbps can be met using $\mathcal{B} = \{\text{IB}_1, \text{IB}_3\}$ since $R_1 = 1$ Mbps and $R_3 = 5$ Mbps. In this case, the number of introduced GBs is zero (recall that we assume the "GB reuse" model). This is clearly an optimal assignment.

Case 2: $\nexists \mathcal{B} \subseteq \mathcal{N}$ such that $\sum_{i\in\mathcal{B}} R_i = d$.

In this case, let $\mathcal{B} \subset \mathcal{N}$ be the largest set such that $\sum_{i\in\mathcal{B}} R_i < d$. We assign the channels in $\mathcal{B}$ to this link. The unfulfilled $d - \sum_{i\in\mathcal{B}} R_i$ demand is then assigned to channels extracted from the beginning of one of the idle blocks in $\mathcal{N} \setminus \mathcal{B}$. Consider, for example, the spectrum status in Figure 2. Suppose that we need to assign channels to a new link with $d = 7$ Mbps. This demand cannot be exactly met by any combination of idle blocks. It can be satisfied using blocks IB_1 and IB_3, of rates 1 Mbps and 5 Mbps, and one

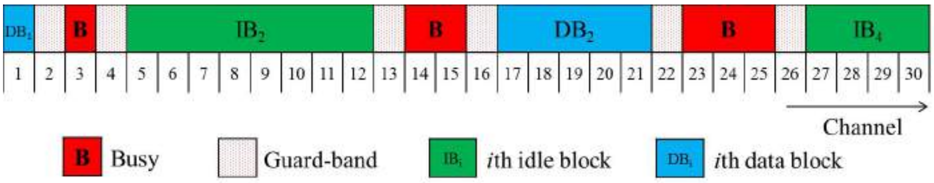


Figure 4: Channel assignment with no additional GBs ($d = 6$ Mbps).

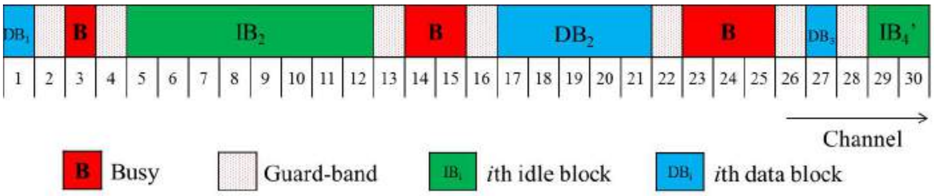


Figure 5: Channel assignment with one additional GB ($d = 7$ Mbps).

channel (channel 27) taken from the 4th idle block. As shown in Figure 5, this results in one additional GB, which is optimal because any other feasible assignment will introduce at least one GB (if there is an assignment with no new GBs, then this contradicts the assumption made in case 2). Hence, the total number of introduced GBs is either zero or one. ■

Having established that assigning channels on a per-block basis results in the optimal $\xi_{\mathbf{link}}$, Problem 1 can be re-stated as follows: Given the set of idle blocks $\mathcal{N}$, obtain a combination of idle blocks that either satisfies the link demand d or achieves the nearest rate to d. This is exactly the subset sum problem (SSP) [22], where "items" correspond to idle frequency blocks and the weights of these items correspond to the rates supported by the idle blocks. Let x_i be a binary variable; $x_i = 1$ if idle block i is to be assigned to the underlying link, otherwise, $x_i = 0$. Then, the optimal GBA channel assignment can be stated as follows:

Problem 1 (SSP):

$$\underset{\{x_i, i\in\mathcal{N}\}}{\text{maximize}} \left\{ \mathcal{R}_s \stackrel{\text{def}}{=} \sum_{i=1}^{N} R_i x_i \right\} \tag{4}$$

subject to:

$$\sum_{i=1}^{N} R_i x_i \leq d \tag{5}$$

$$x_i \in \{0,1\}, \forall i \in \mathcal{N}. \tag{6}$$

Let $\mathcal{R}_s^*$ denote the optimal solution for the SSP. From (5), $\mathcal{R}_s^* \leq d$. When $\mathcal{R}_s^* < d$, we augment the SSP with *a post-processing phase* to make up for the demand "deficit". As stated in Lemma 1 below, after executing the SSP, each of the remaining idle blocks for sure supports a data rate greater than $d - \mathcal{R}_s^*$. In the post-processing phase, we assign a portion of $(d - \mathcal{R}_s^*)/r$ channels[2] from any of the remaining idle blocks, starting from the beginning of the block. The assigned channels are followed by a GB, as shown in Figure 5.

Lemma 1. Let $\mathcal{C}$ be the set of assigned blocks that result from solving the SSP. If $\mathcal{R}_s^* < d$, then $R_i > d - \mathcal{R}_s^*, \forall i \in \mathcal{N} \setminus \mathcal{C}$.

Proof. We prove Lemma 1 by contradiction. Suppose $\exists i \in \mathcal{N} \setminus \mathcal{C}$ with $R_i \leq d - \mathcal{R}_s^*$. Then, this block will be selected by the solution to the SSP, because SSP selects a combination of idle blocks that achieves the nearest rate to d, and by assumption $\mathcal{R}_s^*$ is the optimal solution to the SSP. Hence, $i \in \mathcal{C}$, which leads to a contradiction. ■

Theorem 2. When augmented with the post-processing phase, SSP attains the optimal GBA channel assignment that achieves the maximum $\xi_{\textbf{link}}$.

Proof. There are two cases to consider.

Case 1: $\mathcal{R}_s^* = d$. In this case, no additional GBs will be introduced, which is optimal.

Case 2: $\mathcal{R}_s^* < d$. In this case, by Lemma 1 and Theorem 1, one new GB will be introduced, which is also optimal (there is no any other feasible assignment that results in a higher $\xi_{\textbf{link}}$). The reason is that by Lemma 1, any feasible assignment will introduce at least one additional GB. ■

[2]This number of channels is integer because both d and $\mathcal{R}_s^*$ are integer multiples of r.

$$R_s^*(i,\tilde{d}) = \begin{cases} R_s^*(i-1,\tilde{d}), & \text{if } \tilde{d} < R_i \\ \max\left(R_s^*(i-1,\tilde{d}), R_s^*(i-1,\tilde{d}-R_i)+R_i\right), & \text{if } R_i \le \tilde{d} \le d. \end{cases} \tag{7}$$

SSP is an NP-complete problem [22, 24, 25]. In the following subsections, we present exact and approximate algorithms for solving it.

3.1. Exact Algorithm based on Dynamic Programming (DP)

The idea behind the DP-based approach is as follows. For each subset of idle blocks, the algorithm finds the maximum achievable rate that is less than or equal to d. A pseudo-code of the DP-based exact channel assignment algorithm is shown in Algorithm 1 [25]. Consider a sub-instance of SSP, consisting of idle blocks $\mathrm{IB}_1, \ldots, \mathrm{IB}_{i-1}$ and rate demand $\tilde{d}$. If the rate supported by IB_i exceeds $\tilde{d}$ (i.e., $R_i > \tilde{d}$), then IB_i will not be included in the optimal assignment. Otherwise, IB_i will be included in the optimal assignment if this results in a better solution than excluding it. Let $R_s^*(i,\tilde{d})$ be the optimal solution value of the sub-instance of the SSP, consisting of idle blocks $\mathrm{IB}_1, \ldots,$ IB_i and demand $\tilde{d}$. Then, the recurrence relation is given by (7) (note that $R_s^*(N,d) \stackrel{\text{def}}{=} R_s^*$).

The DP-based algorithm correctly computes the optimal SSP solution. It runs in $\mathcal{O}(Nd)$ time**, so it is pseudo-polynomial** [25].

3.2. ϵ-approximate Algorithm

A pseudo-polynomial ϵ-approximate algorithm for SSP was developed in [24], and is shown here as Algorithm 2. This algorithm selects the combination of idle blocks that results in a total rate that is closest to d. In the ith iteration (see the 'for' loop in line 3 of Algorithm 2), the algorithm considers all combinations of i idle blocks. For each such combination, the algorithm stores their total rate in one of the elements of the ith list, denoted by l_i. List l_i is obtained by merging lists l_{i-1} and l_{i-1}, augmented with R_i, using the MERGE-LISTS function, which combines the two lists into one ascendingly ordered list with no duplicate elements. The addition operation in line 4 is a per-element addition operation. The approximate algorithm uses a function called TRIM, which trims the lists $l_i, i = 1, \ldots, N$, to reduce their lengths.

Algorithm 1 DP-based Exact Algorithm for SSP

Input: N, d, $N \times (d+1)$ array M
Initialize: $M[1, \tilde{d}] = 0, \forall \tilde{d} \in \{0, 1, \dots, d\}$
for $i = 1 : N$ **do**
  for $\tilde{d} = 0 : d$ **do**
    if $\tilde{d} < R_i$ **then**
      $M[i, \tilde{d}] \leftarrow M[i-1, \tilde{d}]$
    else
      $M[i, \tilde{d}] \leftarrow \max\left\{M[i-1, \tilde{d}], R_i + M[i-1, \tilde{d} - R_i]\right\}$
    end if
  end for
end for
Return: M

Algorithm 2 ϵ-approximate SSP Algorithm

Input: $\mathcal{N}$, d, ϵ, and q
$l_0 \leftarrow \emptyset$
for $i = 1 : N$ **do**
  $l_i \leftarrow$ MERGE-LISTS $(l_{i-1}, l_{i-1} + R_i)$
  $l_i \leftarrow$ TRIM $(l_i, \epsilon/2N)$
  Remove from l_i every element that is greater than q
end for
Let z^* be the largest element in l_N
Return: z^*

Basically, TRIM removes an element with value a from the list if there is another element with value b, such that $|a - b| \leq \delta$. In [24], δ is set to $\epsilon/2N$.

Note that the ϵ-approximate algorithm may return idle blocks with rates less than or equal to the remaining unsatisfied demand, i.e., there is some probability that $\exists$ an unassigned block i such that $R_i \leq d - \sum_{j=1}^{N} R_j \eta_j$. If $R_i = d - \sum_{j=1}^{N} R_j \eta_j$, then the ϵ-approximate algorithm can be turned into optimal by searching for such blocks and including them in the assignment. **The ϵ-approximate algorithm runs in $\mathcal{O}\left(\frac{1}{\epsilon} N^2 \ln d\right)$ time [24].**

Table 1: Complexity of various SSP algorithms.

Algorithm	Complexity
DP-based exact	$\mathcal{O}(Nd)$
ϵ-approximate	$\mathcal{O}\left(\frac{1}{\epsilon}N^2 \ln d\right)$
Greedy	$\mathcal{O}(N \log N)$

3.3. Greedy Scheme

The greedy approach starts with the set of idle blocks, sorted in a descending order of their data rates. It passes through the sorted list and adds the idle blocks sequentially as long as the total rate does not exceed the demand d. The complexity of the algorithm comes from the sorting phase and the traversal of the sorted array. This complexity is $\mathcal{O}(N \log N)$ if one uses a sorting algorithm with complexity $\mathcal{O}(N \log N)$ (e.g., merge sort algorithm).

It is to be noted that the above algorithms take as input the number of idle blocks N, which is typically much smaller than the total number of idle channels M. Therefore, the exact algorithm can be used to retrieve the optimal single-link assignment within a reasonable amount of time. **Table 1 lists the complexity of various SSP algorithms**.

4. Optimal GBA Channel Assignment for Multiple Links

Performing GBA channel assignment on a per-link basis is appropriate when link demands are to be considered sequentially, according to the times of arrival of requests. Alternatively, one may "batch" link demands and consider GBA channel assignment for multiple links. The *batch assignment* approach is expected to achieve higher network-wide spectrum efficiency.

In order to attain the network-wide optimal assignment in a distributed fashion, we follow the *access window (AW)* concept used in [26, 27], where each link broadcasts its rate demand in a given slot. Each link waits for a certain amount of time to collect the demands of other links in the network before executing the joint assignment problem. This time duration is called the access window (AW).

An intuitive way of modeling the optimal GBA channel assignment problem for multiple links is to use the multiple subset sum problem (MSSP) [28, 29]. More specifically, we consider a version of the MSSP with different capacities. Let x_{ij}, where $i \in \mathcal{N}$ and $j \in \mathcal{L}$, be a binary variable, which equals one if IB_i is assigned to link j, and zero otherwise. Then, the channel assignment for multiple links may be modeled as follows:

MSSP:

$$\underset{\{x_{ij}, i\in\mathcal{N}, j\in\mathcal{L}\}}{\text{maximize}} \left\{ \mathcal{R}_m \stackrel{\text{def}}{=} \sum_{j=1}^{L}\sum_{i=1}^{N} R_i x_{ij} \right\} \tag{8}$$

subject to:

$$\sum_{i=1}^{N} R_i x_{ij} \le d_j, \forall j \in \mathcal{L} \tag{9}$$

$$\sum_{j=1}^{L} x_{ij} \le 1, \forall i \in \mathcal{N} \tag{10}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{L}. \tag{11}$$

Several approximations and heuristic algorithms for the MSSP have been proposed in the literature (e.g., [30–32]). In the case of a single-link, SSP augmented with the post-processing phase achieves the maximum **per-link spectrum efficiency**, as proved in Theorem 2. However, in the case of multiple links, maximizing $\mathcal{R}_m$ in (8) does not necessarily achieve the maximum **network-wide spectrum efficiency**. Moreover, MSSP needs to be augmented with a more complicated post-processing phase, and even then, it does not result in the overall optimal assignment. To illustrate this, consider the following example of two links with demands $d_1 = 3$ Mbps and $d_2 = 7$ Mbps. There exists two idle blocks of sizes $\beta_1 = 2$ and $\beta_2 = 11$. MSSP will assign the first idle block to one of the links. Then, in the post-processing phase, either one channel will be assigned to the first link and seven channels to the second link, all taken from the second idle block, or three channels to the first link and five channels to the second link, all taken from the second idle block. In both cases, two additional GBs will be introduced. However, a better assignment with higher $\xi_{\mathbf{net}}$ can be achieved by assigning three channels to the first link and seven channels to the second link, all from the second idle block, without using the first idle block. In this case, only one additional GB is introduced.

It can be easily seen that MSSP results in the optimal network-wide assignment only when there exists a block-based assignment that exactly satisfies the demands of all links. In this case, such block assignment is an optimal assignment. However, the optimal block-based assignment may not always exist. To obtain the network-wide optimal assignment for a general setting, the assignment needs to be performed on a per-channel basis instead of per-block basis. The network-wide optimal GBA channel assignment can be formulated as follows.

Problem 2:

$$\underset{\substack{x_{ij}, i\in\mathcal{M}, j\in\mathcal{L} \\ \eta_i, i\in\mathcal{M}}}{\text{maximize}} \left\{ \sum_{j=1}^{L}\sum_{i=1}^{M} x_{ij} - \frac{1}{M}\sum_{i=1}^{M}\eta_i \right\} \tag{12}$$

subject to:

$$\sum_{i=1}^{M} x_{ij} \le \alpha_j, \forall j \in \mathcal{L} \tag{13}$$

$$\sum_{j=1}^{L} x_{ij} \le 1, \forall i \in \mathcal{M} \tag{14}$$

$$\eta_i^{(\text{start})} = x_{(i+1)j} \wedge (1 - x_{ij}) \wedge (1 - \xi_i), \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{15}$$

$$\eta_i^{(\text{end})} = (1 - x_{ij}) \wedge x_{(i-1)j} \wedge (1 - \xi_i), \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{16}$$

$$\eta_i = \eta_i^{(\text{end})} \vee \eta_i^{(\text{start})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{17}$$

$$\eta_1 = \eta_1^{(\text{start})} \tag{18}$$

$$\eta_M = \eta_M^{(\text{end})} \tag{19}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{M}, \forall j \in \mathcal{L} \tag{20}$$

$$\eta_i \in \{0,1\}, \forall i \in \mathcal{M} \tag{21}$$

where '$\wedge$' and '$\vee$' denote the logical AND and OR operators, respectively. $\eta_i, i \in \mathcal{M}$, is a binary variable; $\eta_i = 1$ if channel i is a newly introduced GB, and zero otherwise. $\xi_i, i \in \mathcal{M}$, is a given data; $\xi_i = 1$ if channel i is an existing GB, and zero otherwise. $\eta_1^{(\text{start})}$ and $\eta_1^{(\text{end})}, i \in \mathcal{M}$, are additional auxiliary variables to simplify the formulation. Because $\frac{1}{M}\sum_{i=1}^{M}\eta_i < 1$, the first term in (12) always dominates the second term.

Constraint (15) ensures using a GB channel to the left of each assigned frequency block (i.e., before the start of the block) if there is no existing GB channel. Specifically, this constraint says that if channel $i+1$ is assigned to link j (i.e., $x_{(i+1)j} = 1$), channel i is not assigned to link j (i.e., $x_{ij} = 0$), and channel i is not an existing GB (i.e., $\xi_i = 0$), then channel i needs to be reserved as a GB channel (i.e., η_i needs to be set to 1). Similarly, Constraint (16) ensures using a GB channel to the right of each assigned frequency block (i.e., after the end of the block). Constraint (17) ensures that $\eta_i = 1$ if slot $i+1$ is the start of a frequency block or slot $i-1$ is the end of a frequency block. To simplify our formulation, we reformulate the constraints that contain logical operators (i.e., Constraints (15), (16), and (17)). Constraint (15) can be reformulated, and equivalently replaced by the following set of constraints:

$$\eta_i^{(\text{start})} \leq x_{(i+1)j}, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{22}$$

$$\eta_i^{(\text{start})} \leq 1 - x_{ij}, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{23}$$

$$\eta_i^{(\text{start})} \leq 1 - \xi_i, \forall i \in \mathcal{M} \setminus \{M\} \tag{24}$$

$$\eta_i^{(\text{start})} \geq x_{(i+1)j} + (1 - x_{ij}) + (1 - \xi_i) - 2, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{25}$$

$$\eta_i^{(\text{start})} \geq 0. \tag{26}$$

Similarly, constraint (16) can be reformulated as follows:

$$\eta_i^{(\text{end})} \leq 1 - x_{ij}, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{27}$$

$$\eta_i^{(\text{end})} \leq x_{(i-1)j}, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{28}$$

$$\eta_i^{(\text{end})} \leq 1 - \xi_i, \forall i \in \mathcal{M} \setminus \{1\} \tag{29}$$

$$\eta_i^{(\text{end})} \geq (1 - x_{ij}) + x_{(i-1)j} + (1 - \xi_i) - 2, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{30}$$

$$\eta_i^{(\text{end})} \geq 0. \tag{31}$$

Constraint (17) can be reformulated as follows:

$$\eta_i \geq \eta_i^{(\text{start})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{32}$$

$$\eta_i \geq \eta_i^{(\text{end})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{33}$$

$$\eta_i \leq \eta_i^{(\text{start})} + \eta_i^{(\text{end})}, \forall i \in \mathcal{M} \setminus \{1, M\}. \tag{34}$$

After reformulating constraints (15), (16), and (17), Problem 2 can be stated as follows.

Problem 2 (reformulated):

$$\underset{\substack{\{x_{ij}, i\in\mathcal{M}, j\in\mathcal{L}\\ \eta_i, i\in\mathcal{M}\}}}{\text{maximize}} \left\{ \sum_{j=1}^{L}\sum_{i=1}^{M} x_{ij} - \frac{1}{M}\sum_{i=1}^{M}\eta_i \right\} \tag{35}$$

subject to:

$$\sum_{i=1}^{M} x_{ij} \leq \alpha_j, \forall j \in \mathcal{L} \tag{36}$$

$$\sum_{j=1}^{L} x_{ij} \leq 1, \forall i \in \mathcal{M} \tag{37}$$

$$\eta_i^{(\text{start})} \leq x_{(i+1)j}, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{38}$$

$$\eta_i^{(\text{start})} \leq 1 - x_{ij}, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{39}$$

$$\eta_i^{(\text{start})} \leq 1 - \xi_i, \forall i \in \mathcal{M} \setminus \{M\} \tag{40}$$

$$\eta_i^{(\text{start})} \geq x_{(i+1)j} + (1 - x_{ij}) + (1 - \xi_i) - 2, \forall i \in \mathcal{M} \setminus \{M\}, j \in \mathcal{L} \tag{41}$$

$$\eta_i^{(\text{start})} \geq 0 \tag{42}$$

$$\eta_i^{(\text{end})} \leq 1 - x_{ij}, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{43}$$

$$\eta_i^{(\text{end})} \leq x_{(i-1)j}, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{44}$$

$$\eta_i^{(\text{end})} \leq 1 - \xi_i, \forall i \in \mathcal{M} \setminus \{1\} \tag{45}$$

$$\eta_i^{(\text{end})} \geq (1 - x_{ij}) + x_{(i-1)j} + (1 - \xi_i) - 2, \forall i \in \mathcal{M} \setminus \{1\}, j \in \mathcal{L} \tag{46}$$

$$\eta_i^{(\text{end})} \geq 0 \tag{47}$$

$$\eta_i \geq \eta_i^{(\text{start})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{48}$$

$$\eta_i \geq \eta_i^{(\text{end})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{49}$$

$$\eta_i \leq \eta_i^{(\text{start})} + \eta_i^{(\text{end})}, \forall i \in \mathcal{M} \setminus \{1, M\} \tag{50}$$

$$\eta_1 = \eta_1^{(\text{start})} \tag{51}$$

$$\eta_M = \eta_M^{(\text{end})} \tag{52}$$

$$x_{ij} \in \{0, 1\}, \forall i \in \mathcal{M}, \forall j \in \mathcal{L} \tag{53}$$

$$\eta_i \in \{0, 1\}, \forall i \in \mathcal{M}. \tag{54}$$

In the joint assignment for multiple links, each idle channel before the assignment will end up being in one of $L + 2$ states after the assignment: assigned to one of the L links, reserved as a GB, or left unassigned. Therefore, obtaining the optimal solution through an exhaustive search approach incurs an exponential complexity of $(L + 2)^I$, where $I = \sum_{i=1}^{N} \beta_i$ and $\beta_i = R_i/r$ (defined in Section 3). In the following subsection, we present an exponential-time exact algorithm for the batch assignment, followed by an approximate sequential assignment algorithm.

4.1. Exponential-time Exact Algorithm

We implement the optimal assignment of multiple links that results in the maximum assigned rate with the minimum number of introduced GBs by following an exhaustive search approach that benefits from some pruning rules. A tree is used for this exhaustive search, in which each node is represented by a state vector that contains the states of the I channels. Node i corresponds to a state vector $n_i = (s_1, s_2, \ldots, s_I)$, where $s_i \in \{D, G, 1, 2, \ldots, L\}$ represents the state of channel i with D means channel i is left idle, G means that is reserved as a GB, and $k \in \mathcal{L}$, means that it is assigned to link k. The depth of a node on the search tree represents the number of determined variables in that node, i.e., the states of the first i channels, $s_1, \ldots, s_i$, for all nodes of depth i are determined. To decrease the search space while ensuring the feasibility conditions, such as the required GBs for the assigned channels, we introduce the following pruning rules. Denote the current set of partially served links by $\mathcal{P}$. Then,

1. If idle channel i is at the beginning of an idle block, $s_i \in \{D, u\}$, where $u \in \mathcal{P}$.
2. If idle channel i is not at the beginning of an idle block, then,
   - if idle channel $i - 1$ has been assigned to link y (i.e., $s_{i-1} = y$), then,
     * if $y \in \mathcal{P}$, $s_i \in \{G, y\}$.
     * if $y \notin \mathcal{P}$, $s_i = G$.
   - if idle channel $i - 1$ has been reserved as a GB (i.e., $s_{i-1} = G$), then, $s_i \in \{D, u\}$, where $u \in \mathcal{P}$.
   - if idle channel $i-1$ has not been assigned (i.e., $s_{i-1} = D$), then, $s_i = D$.
3. Let A_i be the total number of assigned channels in node i located at depth t in the tree. If $A_i < A_{\text{best}} + t - I$, where A_{best} is the total number of assigned channels in the current best solution, then we do not branch

Table 2: Complexity of the multi-link assignment algorithms.

Algorithm	Complexity
Exact (batch)	$\mathcal{O}\left((L+2)^I\right)$
$\text{MSSP}_{\text{exact}}$	$\mathcal{O}\left(L^N\right)$
SEQ_{ASC}, SEQ_{DSC}, SEQ_{RND}	$\mathcal{O}\left(LN\log N\right)$

further from node i, because this will not improve the current best solution. There can be multiple solutions that result in the maximum total number of assigned channels. All these solutions will be recorded, and the one that introduces the minimum number of GBs will be selected.

Adding the above pruning rules reduces the running time of the brute force search significantly. However, the running time is still long, so we limit our simulations in Section 5.2 to small numbers of idle channels and links.

4.2. Approximate Sequential Assignment Algorithm

Given the high complexity of the exact algorithm, we instead propose assigning channels to links sequentially. Each link can be assigned using any of the algorithms proposed in Section 3. The fast greedy algorithm for SSP can be used to assign channels to each individual link. The links can be assigned in different orders. In here, we implement three different ordering approaches: start with the link that has the smallest demand (denoted by SEQ_{ASC}), start with the link with the largest demand (denoted by SEQ_{DSC}), or follow a random ordering of links (denoted by SEQ_{RND}). For the comparisons in Section 5.2, we have also implemented a version of the sequential assignment that uses the algorithm in [3] for each individual link assignment. This algorithm selects existing GBs and minimizes the number of assigned frequency blocks. **Table 2 shows the complexity of various multi-link assignment algorithms**.

5. Performance Evaluation

In this section, we evaluate the proposed channel assignment algorithms. All proposed algorithms were implemented in C++. In addition, we implemented the channel assignment scheme in [3], which we refer to as “Choose all existing GBs” in the legends of the simulation figures. In this scheme, the

Table 3: Simulation parameters for the single link assignment algorithms.

Parameter	Value
d	10 Mbps
p_{busy}	0.25
L	1
M	50

objective function is to minimize the number of assigned idle blocks that are required to meet a certain rate demand. This scheme selects all existing GBs. In the figures for multi-link assignment, "Choose all existing GBs" refers to a sequential assignment approach with a random order, where each link is assigned channels according to the scheme in [3]. Our results are averaged over 50 runs, and the 95% confidence intervals are indicated in the figures.

5.1. Single-link Assignment Algorithms

All single-link assignment algorithms are simulated in a common setup, shown in Table 3, and using a common spectrum status. p_{busy} in Table 3 is the probability that a given channel is already assigned to another link.

Figure 6 depicts the spectrum efficiency vs. p_{busy} for various single-link assignment schemes. As shown in this figure, SSP algorithms achieve higher $\xi_{\textbf{link}}$ than the scheme in [3]. This is attributed to the fact that SSP-based assignment schemes are per-block, so they inherently try to use existing GBs and avoid introducing any new GB, hence maximizing the $\xi_{\textbf{link}}$. As p_{busy} increases, the number of existing GBs increases. This improves the performance of the SSP-based schemes; because the sizes of idle blocks become smaller, which increases the chances of finding a subset of idle blocks whose sum rate is equal to the rate demand d. The performance of the scheme proposed in [3] also improves with p_{busy} because of the reduction in the sizes of idle blocks. The idle blocks selected by this scheme may not change with increasing p_{busy}, but the probability that the first and last channels of these blocks are existing GBs increases, which results in a higher $\xi_{\textbf{link}}$. As shown in Figure 6, the ϵ-approximate and greedy algorithms achieve comparable $\xi_{\textbf{link}}$ to the optimal DP algorithm. Figure 7 shows the $\xi_{\textbf{link}}$ vs. the rate demand d. SSP-based assignment algorithms outperform the one in [3] for all values of d.

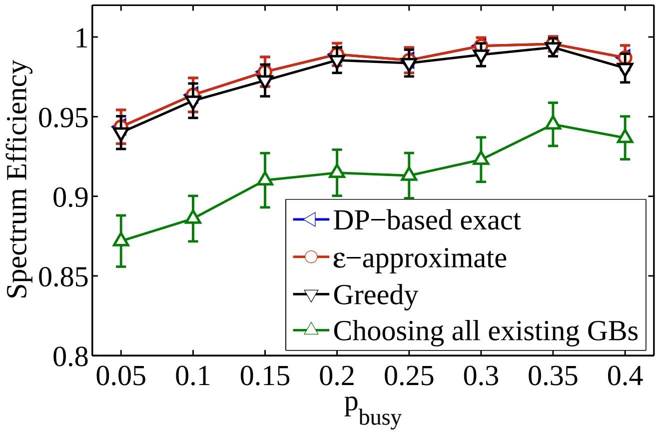


Figure 6: Spectrum efficiency $\xi_{\mathbf{link}}$ vs. p_{busy} (single-link assignment).

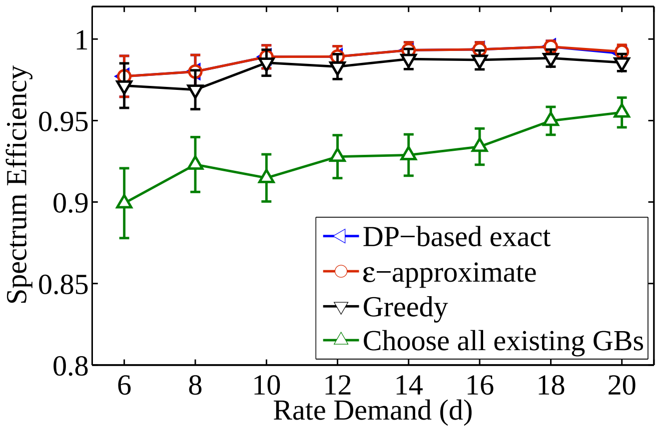


Figure 7: Spectrum efficiency $\xi_{\mathbf{link}}$ vs. d (single-link assignment).

The number of introduced GBs is depicted in Figure 8 for different values of p_{busy}. SSP-based algorithms introduce smaller numbers of GBs (≤ 1), which is consistent with the result in Theorem 2. Figure 9 shows the number of introduced GBs for different values of d. SSP-based assignment algorithms outperform the one in [3] for all values of d. When the channel availability decreases with increasing p_{busy}, the chance of not meeting the link demand increases. Figure 10 shows the fraction of the 50 runs that report infeasibility for different values of p_{busy}. The infeasibility ratio of all considered schemes can reach up to 0.45 when $p_{\text{busy}} = 0.4$. The infeasibility ratio is also shown for various values of d in Figure 11.

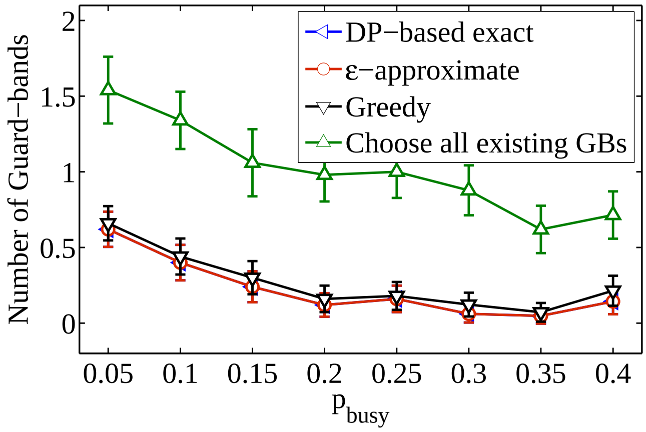


Figure 8: Number of introduced GBs vs. p_{busy} (single-link assignment).

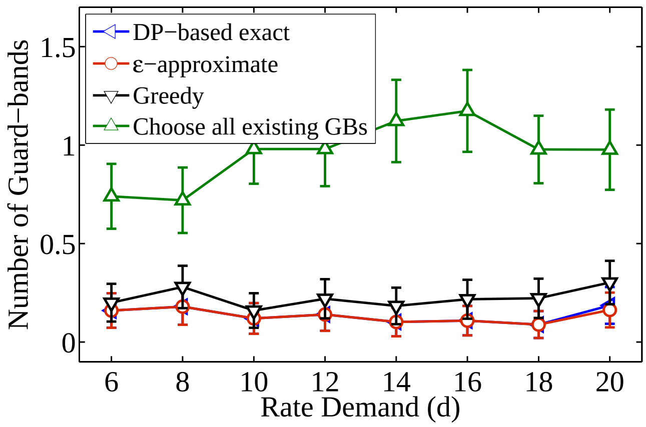


Figure 9: Number of introduced GBs vs. d (single-link assignment).

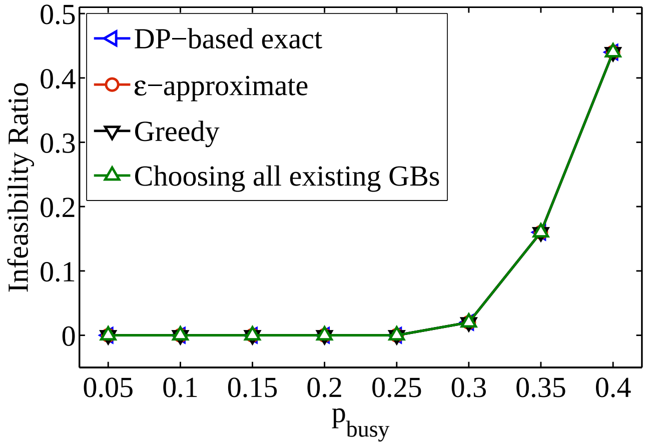


Figure 10: Infeasibility ratio vs. p_{busy} (single-link assignment).

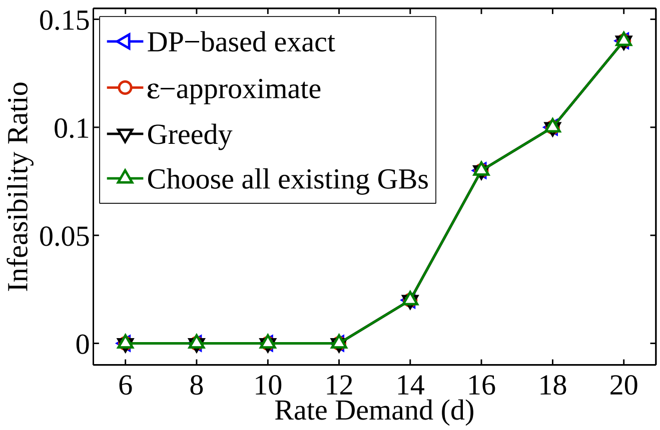


Figure 11: Infeasibility ratio vs. d (single-link assignment).

5.2. Multi-link Assignment Algorithms

First, we simulate the optimal and the heuristic sequential assignment algorithms using the parameters in Table 4. The rate demands are generated uniformly between d_{min} and d_{max}.

Table 5 shows the fraction of runs in which SEQ_{ASC}, SEQ_{DSC}, and SEQ_{RND} result in a sub-optimal solution, for different values of L. The performance

Table 4: Simulation parameters for the multi-link assignment algorithms (experiments vs. L).

Parameter	Value
p_{busy}	0.4
M	50
$d_{\min}$	1 Mbps
$d_{\max}$	5 Mbps

gap between the sequential greedy algorithms and the exact algorithm increases with L.

Table 5: Fraction of runs with sub-optimal results.

SEQ Alg.	$L=2$	$L=4$	$L=6$	$L=8$	$L=10$
SEQ_{ASC}	0.04	0.28	0.6	0.78	0.84
SEQ_{DSC}	0.20	0.34	0.18	0.22	0.20
SEQ_{RND}	0.08	0.34	0.46	0.48	0.48

Define the service ratio (SR) as follows:

$$\text{SR} = \frac{\sum_{j=1}^{L}\sum_{i=1}^{N} R_i x_{ij}}{\sum_{j=1}^{L} d_j}. \tag{55}$$

Figure 12 depicts SR vs. L for the optimal and sequential algorithms. The sequential greedy approaches achieve close-to-optimal SR, even when the number of sub-optimal runs in Table 5 is large. The number of introduced GBs and $\xi_{\mathbf{net}}$ are plotted in Figures 13 and 14, respectively. The performance of the three sequential algorithms depends on the states of the channels and link demands. This is the reason for the large intersecting confidence intervals. The average behavior shows that SEQ_{DSC} outperforms SEQ_{ASC} and SEQ_{RND} in terms of $\xi_{\mathbf{net}}$ and SR, especially for a large L.

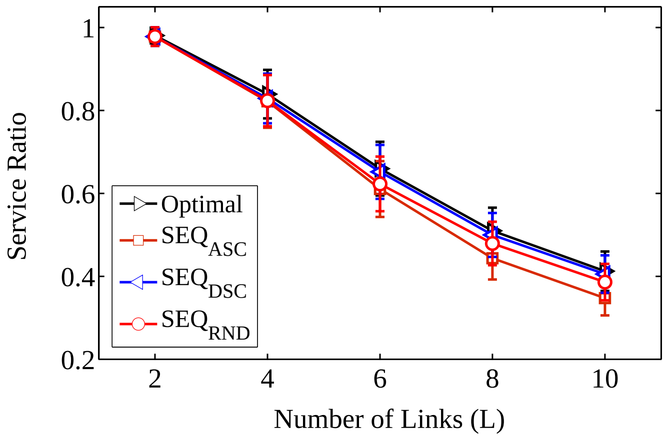


Figure 12: Service ratio vs. L (multi-link assignment).

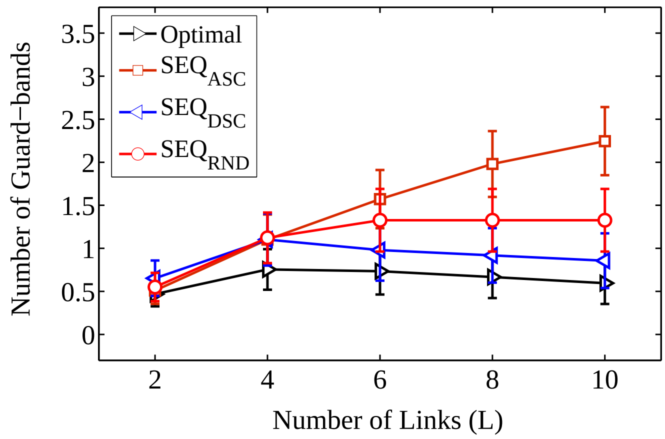


Figure 13: Number of introduced GBs vs. L (multi-link assignment).

Table 6: Simulation parameters for the multi-link assignment algorithms (experiments vs. p_{busy}).

Parameter	Value
L	10
M	150
d_{min}	2 Mbps
d_{max}	10 Mbps

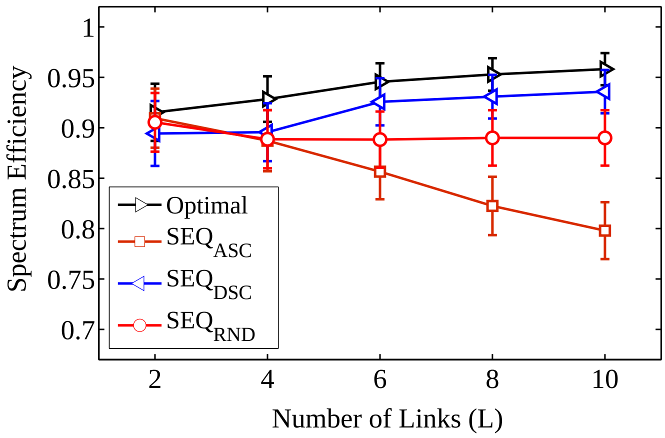


Figure 14: Spectrum efficiency vs. L (multi-link assignment).

Next, we simulate the exact MSSP and the heuristic sequential algorithms using the parameter values in Table 6. The rate demands are generated uniformly between d_{min} and d_{max}.

Figure 15 depicts SR vs. p_{busy}. SR decreases with p_{busy}. All algorithms achieve very close SRs, but they achieve different performance in terms of the number of introduced GBs and $\xi_{\mathbf{net}}$, as shown in Figures 16 and 17, respectively. MSSP achieves a better average performance than the sequential algorithms; because, even though it is not optimal, it assigns channels to links jointly considering all demands. MSSP and the sequential algorithms outperform the scheme in [3]. As shown in Figure 16, the inefficient performance of the scheme in [3] is more noticeable when p_{busy} is small, which leads to idle blocks of large sizes. Since the algorithm in [3] aims at minimizing the number of assigned blocks, larger blocks will be preferable over smaller blocks, which introduces more GBs and reduces $\xi_{\mathbf{net}}$. The increase in the number of introduced GBs also reduces the SR, given in (55).

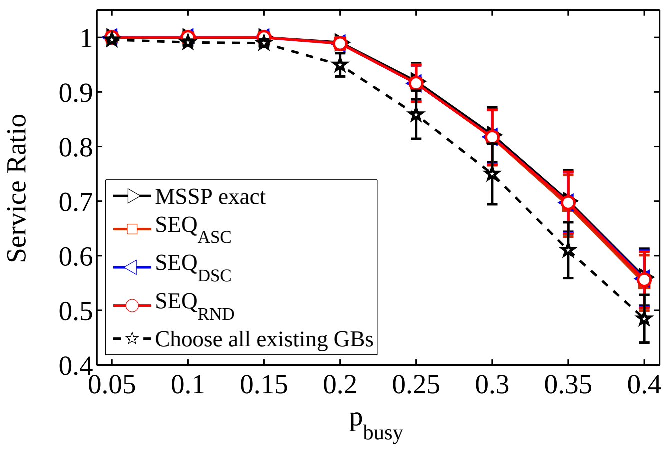


Figure 15: Service ratio vs. p_{busy} (multi-link assignment).

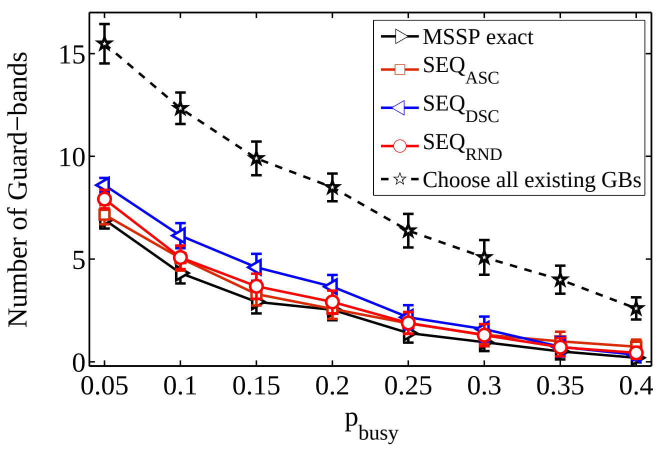


Figure 16: Number of introduced GBs vs. p_{busy} (multi-link assignment).

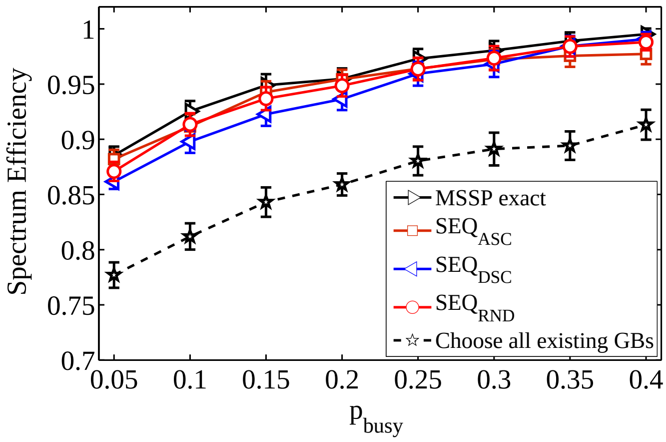


Figure 17: Spectrum efficiency vs. p_{busy} (multi-link assignment).

6. Related Work

To support applications with high rate demands, the IEEE 802.11n and the upcoming IEEE 802.11ac standards have adopted channel bonding [4–8]. By bonding two 20-MHz channels, IEEE 802.11n supports a single 40 MHz channel [9]. In [5, 6], the authors conducted experimental studies in the 5 GHz band over an IEEE 802.11n testbed to characterize the behavior of channel bonding. They observed that ACI needs to be mitigated in order to perform intelligent channel bonding. The IEEE 802.11ac standard enhances the throughput beyond the IEEE 802.11n using an 80 MHz channel bonding technique [7, 8]. In [7], the authors compared static and dynamic channel access schemes, applied to the IEEE 802.11ac standard. In the dynamic scheme, radios can switch between different bandwidths (20, 40, and 80 MHz), whereas in the static scheme radios tune to a fixed bandwidth. Several resource allocation schemes with channel bonding have been considered in [33–35] for OFDMA systems. However, none of the above schemes account for ACI through GBs.

The concept of channel bonding can be extended to non-adjacent frequency channels, and is referred to as channel aggregation. LTE-Advanced supports channel aggregation for 4G cellular networks by allowing mobile operators to aggregate spectrum from non-adjacent bands to support links with high demands [14]. Implementation challenges of channel aggregation were studied in [15, 16]. Recently, distributed channel aggregation was studied in [17–19] within a game-theoretic framework. In [17], the authors modeled the problem of distributed channel selection with aggregation as a stochastic game with incomplete information. They have shown that by adopting learning automata, the radios converge to a Nash equilibrium. A spatial spectrum sharing-based channel aggregation was studied in [18] from a game-theoretic perspective. In [18], the authors considered a model where an operator can access and aggregate the licensed spectrum of other operators upon payment of a certain fee. They modeled the channel aggregation problem as a pricing game. They related the pricing game to a power control game, and derived the Stackelberg equilibrium for the pricing and power optimization problem. In [19], the problem of dynamic inter-network channel aggregation was studied, where mobile operators decide whether to allow a portion of their spectrum to be used by other operators for a given duration. They modeled the problem as a Bayesian game with incomplete information. A distributed algorithm that approaches a neighborhood of a Bayesian Nash equilibrium was proposed. Although co-channel interference was extensively studied in the context of distributed channel allocation (e.g., [20, 21]), most existing works on channel allocation, including the schemes that support channel aggregation, do not account for ACI.

In [36], the amount of required GBs was determined based on the differences in the capacity limits of the used spectrum. A designated spectrum broker was used to manage spectrum sharing among different users with different priorities. In [37], a centralized adaptive GB configuration, called *Ganache*, was proposed to account for ACI. Ganache does not support channel aggregation. Our proposed channel assignment schemes support both channel bonding and aggregation, while mitigating ACI.

7. Future Research

Due to multi-path fading and shadowing, the channel quality in wireless networks is often uncertain and time-varying. In this case, it makes sense to model the channel quality (i.e., achievable rate) as a stochastic process. Each time the channel assignment is performed, the rates of various channels would be sampled from probability distribution functions. In this section, we provide directions for extending our sequential and batch channel assignment schemes to the case when the rates of various channels are treated as random variables.

7.1. Sequential Channel Assignment Under Uncertainty

We propose using stochastic programming techniques to formulate the channel assignment problem under uncertain channel rates. For deterministically known channel rates, the sequential channel assignment problem is given by Problem 1 ((4)- (6)). When the channel rates are random, the feasible region in Problem 1 becomes also random. Different stochastic optimization approaches have been proposed in the literature to deal with the uncertainty in the feasible region of an optimization problem [38]. One approach that we plan to pursue is the "chance constraint approach." In this approach, the constraints that include random variables are enforced to be satisfied with a probability greater than a given threshold. In Problem 1, when the rates are random, constraint (5) becomes random. Using a chance constraint, the link demand can be probabilistically satisfied. Moreover, the objective function in Problem 1 becomes random. We account for this randomness by replacing the channel rates by their expected values. Although the rates are random, their distributions are usually known prior to channel assignment. Adopting the chance constraint approach, the channel assignment problem under channel uncertainty can be formulated as follows:

Chance-constrained SSP:

$$\underset{\{x_i, i \in \mathcal{N}\}}{\text{maximize}} \sum_{i=1}^{N} \mathbb{E}[\tilde{R}_i] x_i \tag{56}$$

subject to:

$$\Pr\left\{\sum_{i=1}^{N} \tilde{R}_i x_i \geq d\right\} \geq \beta \tag{57}$$

$$x_i \in \{0,1\}, \forall i \in \mathcal{N} \tag{58}$$

where $\tilde{R}_i$ is the rate supported by the ith frequency block. $x_i, N, \mathcal{N}$, and d are as defined in Problem 1. Based on the distribution of the channel rates, the chance constraint in (57) can be reformulated. Further investigation of this problem is left for future research.

7.2. Batch Channel Assignment Under Uncertainty

One way to tackle the batch channel assignment problem under channel uncertainty is to generalize the above chance-constrained SSP formulation to multiple links. This can be done by introducing a chance constraint for each link demand, which leads to a chance-constrained MSSP formulation:

Chance-constrained MSSP:

$$\underset{\{x_{ij}, i\in\mathcal{N}, j\in\mathcal{L}\}}{\text{minimize}} \sum_{j=1}^{L}\sum_{i=1}^{N} \mathbb{E}[\tilde{R}_i]x_{ij} \tag{59}$$

subject to:

$$\Pr\left\{\sum_{i=1}^{N} \tilde{R}_i x_{ij} \geq d_j\right\} \geq \beta, \forall j \in \mathcal{L} \tag{60}$$

$$\sum_{j=1}^{L} x_{ij} \leq 1, \forall i \in \mathcal{N} \tag{61}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, j \in \mathcal{L}. \tag{62}$$

Recall that in the case of deterministic channel rates, MSSP does not always achieve the optimal (spectrum efficient) assignment. Therefore, chance-constrained MSSP may not be the optimal stochastic channel assignment scheme. Further investigation of this problem is left for future research.

8. Conclusion

In this paper, we proposed GBA channel assignment algorithms that account for ACI in multi-channel wireless networks with channel bonding/aggregation. Both single-link (sequential) as well as multiple links (batch) assignments were considered. For a single link, the optimal assignment problem was formulated as an SSP, and exact and approximate solutions were presented. We also obtained the optimal assignment for multiple links. To avoid the high complexity of the exact multi-link assignment algorithm, a polynomial-time sequential assignment was presented, which adopts a greedy strategy for each link. Our numerical results showed that the greedy sequential assignment achieves a near-optimal performance. The approximate greedy approach is still better than a previously proposed approach in [3, 39].

[1] V. Angelakis, S. Papadakis, V. Siris, A. Traganitis, Adjacent channel interference in 802.11a is harmful: Testbed validation of a simple quantification model, IEEE Communications Magazine 49 (3) (2011) 160–166.

[2] A. Zubow, R. Sombrutzki, Adjacent channel interference in IEEE 802.11n, in: Proceedings of the IEEE WCNC Conference, 2012, pp. 1163–1168.

[3] H. Bany Salameh, M. Krunz, D. Manzi, Spectrum bonding and aggregation with guard-band awareness in cognitive radio networks, IEEE Transactions on Mobile Computing, 2013.
URL http://doi.ieeecomputersociety.org/10.1109/TMC.2013.11

[4] D. Skordoulis, Q. Ni, H.-H. Chen, A. Stephens, C. Liu, A. Jamalipour, IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs, IEEE Transactions on Wireless Communications 15 (1) (2008) 40–47.

[5] L. Deek, E. Garcia-Villegas, E. Belding, S.-J. Lee, K. Almeroth, The impact of channel bonding on 802.11n network management, in: Proceedings of the ACM CoNEXT Conference, 2011.

[6] L. Deek, E. Garcia-Villegas, E. Belding, S. Lee, K. Almeroth, Intelligent channel bonding in 802.11n WLANs, IEEE Transactions on Mobile Computing, 2013.

[7] M. Park, IEEE 802.11ac: Dynamic bandwidth channel access, in: Proceedings of the IEEE ICC Conference, 2011, pp. 1–5.

[8] O. Bejarano, E. Knightly, M. Park, IEEE 802.11ac: from channelization to multi-user MIMO, IEEE Communications Magazine 51 (10) (2013) 84–90.

[9] IEEE 802.11n: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput, ANSI/IEEE Std 802.11n-2009, 2009.

[10] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, P. Bahl, A case for adapting channel width in wireless networks, in: Proceedings of the ACM SIGCOMM Conference, 2008, pp. 135–146.

[11] K. Pelechrinis, I. Broustis, T. Salonidis, S. V. Krishnamurthy, P. Mohapatra, Design and deployment considerations for high performance MIMO testbeds, in: Proceedings of the ACM WICON Conference, 2008.

[12] E. Gelal, K. Pelechrinis, I. Broustis, S. Krishnamurhty, S. Mohammed, A. Chockalingam, S. Kasera, On the impact of MIMO diversity on higher layer performance, in: Proceedings pf the IEEE ICDCS Conference, 2010, pp. 764–773.

[13] D. Halperin, W. Hu, A. Sheth, D. Wetherall, 802.11 with multiple antennas for dummies, ACM SIGCOMM Computer Communication Review 40 (1) (2010) 19–25.

[14] G. Yuan, X. Zhang, W. Wang, Y. Yang, Carrier aggregation for LTE-advanced mobile communication systems, IEEE Communications Magazine 48 (2) (2010) 88–93.

[15] D. Bai, C. Park, J. Lee, H. Nguyen, J. Singh, A. Gupta, Z. Pi, T. Kim, C. Lim, M.-G. Kim, I. Kang, LTE-advanced modem design: challenges and perspectives, IEEE Communications Magazine 50 (2) (2012) 178–186.

[16] K. Pedersen, F. Frederiksen, C. Rosa, H. Nguyen, L. Garcia, Y. Wang, Carrier aggregation for LTE-advanced: functionality and performance aspects, IEEE Communications Magazine 49 (6) (2011) 89–95.

[17] I. Macaluso, L. DaSilva, L. Doyle, Learning Nash equilibria in distributed channel selection for frequency-agile radios, in: Workshop on Artificial Intelligence for Telecommunications and Sensor Networks, 2012.

[18] Y. Xiao, T. Forde, I. Macaluso, L. DaSilva, L. Doyle, Spatial spectrum sharing-based carrier aggregation for heterogeneous networks, in: Proceedings of the IEEE GLOBECOM Conference, 2012, pp. 2591–2596.

[19] Y. Xiao, C. Yuen, P. Di Francesco, L. DaSilva, Dynamic spectrum scheduling for carrier aggregation: A game theoretic approach, in: Proceedings of the IEEE ICC Conference, 2013, pp. 2672–2676.

[20] C. Wu, K. Chowdhury, M. Di Felice, W. Meleis, Spectrum management of cognitive radio using multi-agent reinforcement learning, in: Proceedings of the International Conference on Autonomous Agents and Multiagent Systems: Industry Track, 2010, pp. 1705–1712.

[21] D. Leith, P. Clifford, Convergence of distributed learning algorithms for optimal wireless channel allocation, in: Proceddings of the IEEE Conference on Decision and Control, 2006, pp. 2980–2985.

[22] S. Martello, P. Toth, Knapsack problems: algorithms and computer implementations, John Wiley & Sons, 1990.

[23] I. Trigui, M. Siala, H. Boujemaa, Optimized pulse shaping for OFDM multi-user communications over doubly dispersive channels, in: Proceedings of the International Symposium on Signal Processing and Its Applications, 2007, pp. 1–4.

[24] T. H. Cormen, C. Stein, R. L. Rivest, C. E. Leiserson, Introduction to Algorithms, 2nd Edition, McGraw-Hill Higher Education, 2001.

[25] J. Kleinberg, E. Tardos, Algorithm Design, Addison-Wesley, 2005.

[26] A. Muqattash, M. Krunz, POWMAC: a single-channel power-control protocol for throughput enhancement in wireless ad hoc networks, IEEE Journal on Selected Areas in Communications 23 (5) (2005) 1067–1084.

[27] H. Bany Salameh, M. Krunz, Adaptive power-controlled MAC protocols for improved throughput in hardware-constrained cognitive radio networks, Ad Hoc Networks Journal 9 (7) (2011) 1127–1139.

[28] S. Martello, P. Toth, A bound and bound algorithm for the zero-one multiple knapsack problem, Discrete Applied Mathematics 3 (4) (1981) 275–288.

[29] D. Pisinger, An exact algorithm for large multiple knapsack problems, European Journal of Operational Research 114 (1999) 528–541.

[30] D. Bertsimas, R. Demir, An approximate dynamic programming approach to multidimensional knapsack problems, Management Science 48 (4) (2002) 550–565.

[31] C. Chekuri, S. Khanna, A PTAS for the multiple knapsack problem, in: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 2000, pp. 213–222.

[32] A. Caprara, H. Kellerer, U. Pferschy, A 3/4-approximation algorithm for multiple subset sum, Journal of Heuristics 9 (2003) 99–111.

[33] H. Zhu, J. Wang, Chunk-based resource allocation in OFDMA systems - part I: chunk allocation, IEEE Transactions on Communications 57 (9) (2009) 2734–2744.

[34] H. Zhu and J. Wang, Chunk-based resource allocation in OFDMA systems - Part II: Joint chunk, power and bit allocation, IEEE Transactions on Communications 60 (2) (2012) 499–509.

[35] L. Lei, V. Angelakis, D. Yuan, Performance analysis of chunk-based resource allocation in wireless OFDMA systems, in: IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2012, pp. 90–94.

[36] O. Yu, E. Saric, A. Li, Dynamic control of open spectrum management, in: Proceedings of the IEEE WCNC Conference, 2007, pp. 127–132.

[37] L. Yang, B. Y. Zhao, H. Zheng, The spaces between us: Setting and maintaining boundaries in wireless spectrum access, in: Proceedings of the ACM MobiCom Conference, 2010, pp. 37–48.

[38] P. Kall, S. W. Wallace, Stochastic Programming, John Wiley and Sons, 1994.

[39] H. Bany Salameh, M. Krunz, D. Manzi, An efficient guard-band-aware multi-channel spectrum sharing mechanism for dynamic access networks, in: Proceedings of the IEEE GLOBECOM Conference, 2011, pp. 1–5.

Gulnur Selda Uyanik received her B.Sc. degree in Computer Engineering from Istanbul Technical University, Turkey, in 2008. She received her M.Sc. degree in 2011 and is currently a Ph.D. student in the same major and university. She was a visiting scholar in Advanced Networking Laboratory at University of Arizona, USA between August 2012 - January 2013. Currently, she is a teaching and research assistant in the Department of Computer Engineering, Istanbul Technical University. She received the Best Paper Award from AICT-2009. Her research interests are in the areas of wireless sensor networks, modeling and spectrum management in cognitive radio networks.

Mohammad J. Abdel-Rahman received the M.S. degree in electrical engineering from Jordan University of Science and Technology, Jordan, in 2010 and the B.S. degree in communication engineering from Yarmouk University, Jordan, in 2008. He is currently working toward the Ph.D. degree in electrical and computer engineering at the University of Arizona, where he is a research assistant in the advanced networking lab. His current research interests are in wireless communications and wireless networking, with emphasis on wireless cognitive radio networks. He serves as a reviewer for several international conferences and journals. He is a member of the IEEE.

Marwan Krunz is a Professor of Electrical and Computer Engineering at the University of Arizona. He has been the UA site director for "Connection One." His research interests include computer networking and wireless communications with a focus on distributed radio resource management in wireless and sensor networks, protocol design, and secure communications. He has served on the editorial boards for the IEEE Transactions on Network and Service Management, IEEE/ACM Transactions on Networking, IEEE Transactions on Mobile Computing, and Computer Communications Journal, and as a TPC chair for several international conferences. He is a fellow of the IEEE.