On Designing Truthful and Optimal Spectrum Assignment Mechanisms for Dynamic Spectrum Access

Tingting Chen and Sheng Zhong
Department of Computer Science and Engineering
State University of New York at Buffalo, Buffalo, NY 14260-2000, U.S.A.
{tchen9, szhong}@buffalo.edu

***Abstract*—With the advances of cognitive radio technologies, Dynamic Spectrum Access (DSA) has received increasing attention. For dynamic spectrum access, a good spectrum assignment algorithm often considers the spectrum demands of different devices and assigns contiguous and variable bandwidth to them. However, if the DSA devices at a location belong to two or more entities, a selfish entity may want to over-claim its devices' demands, in order to obtain more spectrum for its own devices. Unfortunately none of the existing work has addressed this incentive problem in assigning contiguous and variable bandwidth to DSA devices.**

To solve this incentive problem, we propose two spectrum assignment mechanisms for DSA, one for single collision domain and one for multiple collision domains. We prove that our two mechanisms are truthful, i.e., they both stimulate each entity to submit the true demands of its own devices. We also prove that the mechanism for single collision domain is optimal in terms of satisfying spectrum demands, and our mechanism for multiple collision domains achieves constant-ratio approximation of optimal spectrum assignment. Extensive experimental evaluations of our mechanisms show that they have very good performance.

I. Introduction

With the advances of cognitive radio technologies, Dynamic Spectrum Access (DSA) has received increasing attention. Using cognitive radio devices, users can dynamically locate the unused spectrum and then appropriately configure the radios to operate in the spectrum, such that there is no (significant) interference to other users. With the capability of flexible radio configurations, DSA provides a promising resolution for the unsatisfied users to better utilize the scarce spectrum.

We notice that, for DSA, there may be a *problem of incentives*, when allocating contiguous and variable bandwidth of spectrum to devices based on their demands. Specifically, to assign the spectrum to DSA devices at a certain location, if the DSA devices at this location belong to two or more entities, a selfish entity may want to over-claim the demands of its devices, so that its own devices can be assigned more spectrum. Therefore, there needs to be a mechanism to stimulate the entities to submit the true spectrum demands of their own devices.

Consider, for example, an office building shared by two companies. Suppose that each company has three access points that need to access the aforementioned spectrum. Among the six access points, one has significantly higher demands for spectrum than all other five because it is located in a meeting room and thus is often used for video conferences. In this scenario, if every access point is assigned the same amount of spectrum, then clearly the one located in the meeting room will be penalized, and the quality of the related video conferences will become very poor. On the other hand, if each access point is assigned spectrum according to its demand, then each company will claim higher demands for its own access points, so that its own access points get more spectrum. Unless there is a mechanism to stimulate companies to make truthful claims, the selfish behavior of companies will lead to a competition of making false claims of spectrum demands, and very likely a failure of the spectrum assignment algorithm.

In this paper, we study the entities' incentive problem in contiguous and variable spectrum assignment of DSA. Our objective is to design spectrum assignment mechanisms that are *truthful* (i.e., stimulate entities to claim the true spectrum demands) and optimal in terms of satisfying spectrum demands. We emphasize that both our problem and our objective are realistic, because in practice it is hard to guarantee the DSA devices at each location all belong to the same entity. Consequently, the mechanisms we design for DSA spectrum assignment are of high practical importance.

It is not hard to see that the problem we consider here is similar to the well known *problem of commons* [10]. One possible solution to the problem of commons is to require a payment for use. Hence, for our problem, we also propose solutions based on payments. The payments we introduce are *not* payments of real money; they are payments of virtual money. Correspondingly, when more than one entity's devices need to have DSA simultaneously at the same location, the bandwidth assigned to each entity depends on how much virtual money the entity is willing to pay for each unit of spectrum. The more unit price it is willing to pay, the more bandwidth it is assigned. We will present the implementation of virtual money in Section V in details.

Our main results in this paper are two mechanisms for assigning contiguous and variable bandwidth of spectrum, in different settings of DSA. Our first mechanism, called SAS, is for <u>S</u>pectrum <u>A</u>ssignment in <u>S</u>ingle collision domain.

In a game theoretic model, we rigorously show that SAS is truthful. In terms of satisfying spectrum demands, we show that SAS is *optimal* (see Section II for the definition of optimality). Our second mechanism, called SAM, is for Spectrum Assignment in Multiple collision domains. SAM, is truthful just like SAS; moreover, SAM can be shown to achieve a *constant-ratio approximation* to the optimal spectrum assignment.

It is worth noting that our work is closely related to, but significantly different from the existing work on spectrum auctions [11], [23], [14], [12], [25]. In spectrum auctions, there is an existing spectrum owner who sells (part of) its unused spectrum. In our problem, there is no seller of the spectrum. Furthermore, spectrum auction only allows bidding using per-channel price, while in this paper we make sure that users are assigned a contiguous but variable bandwidth, depending on the devices demand.

Our contributions can be summarized as follows:

- We are the *first* to study the incentive problem in assigning contiguous but variable bandwidth of spectrum for DSA.
- For spectrum assignment in single collision domain, we present a mechanism SAS. We rigorously show that SAS is truthful and optimal.
- For spectrum assignment in multiple collision domains, we present a mechanism SAM that can be shown to be truthful and achieve a constant-ratio approximation to the optimal spectrum assignment.
- We have done extensive experiments to evaluate the our mechanisms and the results show that they have good performance.

The rest of this paper is organized as follows. In Section II, we present the technical preliminaries. In Sections III, we propose SAS. Section IV is dedicated to our mechanism for multiple collision domains, SAM. We present our evaluation results in Section VI. Finally, we briefly review related literature in Section VII and then conclude in Section VIII.

II. Technical Preliminaries

Suppose that there are K entities interested in DSA at a location. Each entity e has D_e DSA devices, e.g., Wi-Fi like access points, that need to be assigned spectrum. Just as in [16], [8], [5], we assume that there is a spectrum manager who is responsible for assigning spectrum to the devices from different entities. Our objective is to establish mechanisms that assign spectrum to such devices and determine the amount of virtual money that needs to be paid by each entity. (We use the phrase "spectrum assignment mechanism" to emphasize that it is not just an algorithm for assigning spectrum—it is also responsible for deciding the payments.)

Assume that the frequency spectrum that can be used for DSA at this location is (f_l, f_h), where $f_h - f_l = W$. Once we establish a spectrum assignment mechanism, this spectrum assignment mechanism is executed periodically to assign spectrum and determine payments.

For each device d of each entity e, denote by $(L_{e,d}, H_{e,d})$ the spectrum assigned to this device by the spectrum assignment mechanism. Let $w_{e,d} = H_{e,d} - L_{e,d}$. Clearly, $w_{e,d}$ stands for the bandwidth assigned to this device. There is a restriction: $B_\ell \le w_{e,d} \le B_u$, where B_ℓ and B_u are constants. These constants are decided by various factors, including the physical constraints, the FCC regulations, and the policies.

A spectrum assignment mechanism decides the spectrum assignment and payments based on the devices' spectrum demands. We do not assume each device has a fixed demand for spectrum, such that the entity owning this device is completely happy when the device is assigned this amount of spectrum or more, and is completely unhappy when the device is assigned less spectrum. In stead, we assume that, for each entity e, and for each device d of entity e, there is a valuation function $v_{e,d}()$. The input of this valuation function is $w_{e,d}$, the bandwidth assigned to device d. The output of this valuation function is entity e's valuation of this assigned bandwidth. Intuitively, $v_{e,d}()$ represents the device's spectrum demand across all possible levels. We adopt the standard assumption from the literature of economics [15] that every valuation function $v_{e,d}()$ is strictly increasing and quasi-concave. We also assume that there is a constant lower bound ζ for $v_{e,d}(B_\ell)$, i.e., for all e and all d, $v_{e,d}(B_\ell) \ge \zeta$.

Hence, in order to allow the spectrum assignment mechanism to make its decision, each entity e needs to first submit its valuation function set $V_e = \{v_{e,d}() | 1 \le d \le D_e\}$, which consists of the valuation functions of all devices of e. Then, the spectrum assignment mechanism uses these valuation function sets to compute $(L_{e,d}, H_{e,d})$ for each entity e and each device d of entity e. In addition, the mechanism also computes p_e for each entity e, where p_e is the payment entity e needs to make for its use of spectrum.

As we have mentioned, our objective is to design spectrum assignment mechanisms. Ideally, a spectrum assignment mechanism should satisfy the following requirements:

- Truthfulness. Every entity has incentives to submit its true valuation function set. The formal definition of truthfulness requires a game theoretic model. So we leave the formal definition to Section II-A.
- No starvation. Every device involved is assigned some bandwidth.
- Conditional Maximization of Total Valuation. The total valuation of all assigned spectrum should be the maximum possible under the constraint that there is no starvation. As we will see, this is NP-hard for multiple collision domains. Consequently, for multiple collision domains, we loosen this requirement to allow approximations.

We say a mechanism is *optimal* if it guarantees no starvation and maximizes the total valuation under the constraint of no starvation. We say a mechanism is *constant-ratio approximately optimal* if it guarantees no starvation and achieves a constant-ratio approximation to the maximum total valuation under the constraint of no starvation. Hence, for single collision domain, our objective is that the mechanism should be optimal and truthful; for multiple collision domains, our objective is that the mechanism should be constant-ratio approximately optimal and truthful.

Because single collision domain and multiple collision domains are quite different in terms of spectrum assignment, we have designed two spectrum assignment mechanisms in Section III and IV, respectively. In these two sections, we assume that there is a *secure implementation* of virtual money, such that payments computed by the mechanism can be enforced.

A. Game Theoretic Model

To formally analyze the truthfulness of our mechanisms, we need to establish a game theoretic model.

We model the spectrum assignment as a strategic game, where the players are the involved entities. For each entity e, its action in the game is the valuation function set V_e it submits. The payoff for each entity e is decided by the action profile of all players, i.e., the profile of valuation function sets of all entities. We denote this profile by V. Formally, we have:

$$\mathsf{payoff}_\mathsf{e}(\mathsf{V}) = \sum_{1 \le \mathsf{d} \le \mathsf{D_e}} \mathsf{v_{e,d}}(\mathsf{w_{e,d}}) - \mathsf{p_e}. \tag{1}$$

Intuitively, this means that the payoff of entity e is equal to entity e's total valuation of the spectrum assigned to all its devices minus the payment it needs to make for its use of spectrum.

Given this game theoretic model, we can easily define truthfulness for a spectrum assignment mechanism: A mechanism is truthful if and only if it is a *dominant strategy equilibrium* (DSE) [18] for all entities to submit their true evaluation function sets. Intuitively, a DSE guarantees that every player of the game has incentives to play the strategies specified by the DSE regardless of other players' behavior. Below is our formal definition of truthfulness. (In this definition, we use V_{-e} to represent the profile of valuation function sets of all entities other than e. Similar notations are used throughout this paper.)

Definition 1. *A spectrum assignment mechanism is said to be truthful if it is a DSE for all entities to submit their true valuation function sets, i.e., for any entity e, assuming V_e^T is the true valuation function set of entity e, for any valuation function set V_e^A submitted by entity e, for any profile V_{-e} of valuation function sets submitted by all entities other than e,*

$$\mathbf{payoff}_e(V_e^T, V_{-e}) \ge \mathbf{payoff}_e(V_e^A, V_{-e}).$$

III. Single Collision Domain

In this section, we design and analyze a spectrum assignment mechanism, SAS, for the situation in which all DSA devices are in a single collision domain. Throughout this section, we restrict our attention to the case in which

$$\sum_{e=1}^{K} D_e \cdot B_\ell < W \le \sum_{e=1}^{K} D_e \cdot B_u.$$

The reason for this restriction is: If $W > \sum_{e=1}^{K} D_e \cdot B_u$, then we have a trivial mechanism that assigns each device with spectrum of width B_u and charges a flat rate for each device's spectrum usage. If $W < \sum_{e=1}^{K} D_e \cdot B_\ell$, then the requirement of no starvation cannot be achieved by any spectrum assignment mechanism. Hence, the case we focus on is the only case in which a non-trivial mechanism both exists and is needed.

A. Design of Mechanism

The design of SAS is based on two main ideas: greedy spectrum assignment and opportunity-cost-based payment.

Greedy Assignment The first idea is that we should greedily assign the spectrum such that the total valuation is maximized. Of course, we notice that we need to guarantee no starvation before we maximize the total valuation. So SAS first reserves a minimum bandwidth of B_ℓ for each device.

Then, the remaining bandwidth is assigned to devices. Assume that there is a small constant ϵ such that all assigned bandwidths (and the constants B_ℓ and B_u) must be multiples of ϵ.[1] Hence, we assign the remaining bandwidth in slices of size ϵ. Suppose the remaining bandwidth can be divided into N such slices. SAS assigns these N slices to the devices such that their total valuation is maximized.

To achieve this goal, we discretize each submitted valuation function $v_{e,d}()$ to obtain a sequence of valuations: $b_{e,d,1}$, $b_{e,d,2}$,…, $b_{e,d,(B_u-B_\ell)/\epsilon}$, where

$$b_{e,d,j} = v_{e,d}(B_\ell + j\epsilon) - v_{e,d}(B_\ell + (j-1)\epsilon).$$

Intuitively, each $b_{e,d,j}$ is a valuation of one slice of bandwidth of size ϵ. To be more precise, $b_{e,d,j}$ is entity e's valuation of the jth such slice assigned to its device d. Note that, because $v_{e,d}()$ is strictly increasing and quasi-concave,

$$0 < b_{e,d,1} \le b_{e,d,2} \le \ldots \le b_{e,d,(B_u-B_\ell)/\epsilon}$$

Next, we put together all such valuations of slices (for all entities e and all devices d), and pick the N largest from them. These N highest valuations correspond to the N slices SAS assigns to devices. For example, if $N = 3$ and the three largest valuations are $b_{e_1,d_1,1}$, $b_{e_2,d_2,1}$, and $b_{e_1,d_1,2}$, then we assign two slices of size ϵ to device d_1 of entity e_1, and one slice to device d_2 of entity e_2. The total bandwidth assigned to a device is equal to all these slices assigned to it plus B_ℓ.

Once the bandwidth assigned to each device is determined, it is easy to determine the spectrum assigned to this device: SAS starts the assignment from frequency f_l and assigns a continuous spectrum to each device, until frequency f_h is reached.

Opportunity-Cost-based Payment The second idea is that we should calculate the payment of each entity using the opportunity cost of its assigned spectrum. This opportunity cost of the assigned spectrum for entity e is calculated

[1] Since the precision of involved computing is limited, there must be such a small constant ϵ.

as follows. Imagine that SAS re-assigns the slices without considering entity e. It is easy to see that, the set of slices originally assigned to e's devices will be replaced by the same number of slices assigned to other entities' devices, and the assignment of other slices will remain unchanged. Consequently, the opportunity cost is equal to the total valuation of all the newly assigned slices that replace the slices originally assigned to e's devices.

Besides the opportunity cost we discuss above, SAS also charges each entity for its devices' use of the reserved minimum bandwidths of size B_ℓ. For each device, this charge is the same—ζ, the constant lower bound for $v_{e,d}(B_\ell)$.

Using the ideas above, we design SAS, the details of which are shown in Algorithm 1.

In SAS, $B'(1)$ denotes the 1st element of a sequence B'. Similar notations are used throughout this paper.

Note that when we write the details of SAS, we are *sloppy* with the sequences B, B', A, P, P_e. Specifically, each element of B should not only be a valuation $b_{e,d,j}$; it should also contain the corresponding index (e, d, j). Since the elements of B', A, P, P_e all originate from B, they should similarly contain the indices. We choose to be sloppy here in order to avoid too complex notations.

Algorithm 1 SAS: Spectrum assignment mechanism for single collision domain

INPUT: Available Space (f_l, f_h); valuation function set U_e from each entity e, parameter ζ, B_ℓ, B_u and a small number ϵ.
OUTPUT: Assigned spectrum for each device d of each entity e: $(L_{e,d}, H_{e,d})$, and price p_e for each entity e.

$W' = W - B_\ell \cdot \sum_{e=1}^{K} D_e$.
$N' = \frac{B_u - B_\ell}{\epsilon}$; $N = \frac{W'}{\epsilon}$.
for each entity e **do**
  $p_e = 0$.
  for each device d **do**
    for each slice j s.t. $1 \le j \le N'$ **do**
      $(L_{e,d}, H_{e,d}) = (0, 0)$.
      $b_{e,d,j} = v_{e,d}(B_\ell + j\epsilon) - v_{e,d}(B_\ell + (j-1)\epsilon)$.
    end for
  end for
end for
Compose a sequence B, using $b_{e,d,j}$ for all e, d, j.
$B' = sort(B)$. //ordering from largest to smallest.
$A = (B'(1), B'(2), \cdots, B'(N))$.
$P = (B'(N+1), B'(N+2), \cdots, B'(N+N'))$.
for each e **do**, $P_e = P \backslash \{b_{e,d,q} | \forall d, \forall q, \text{s.t.} b_{e,d,q} \in P\}$. **end for**
$s = f_l$.
for each e **do**
  $n_e = |\{b_{e,d,q} | \forall d, \forall q, \text{s.t.} b_{e,d,q} \in A\}|$.
  if $n_e > 0$ **then**
    for each d **do**
      $n_{e,d} = |\{b_{e,d,q} | \forall q, \text{s.t.} b_{e,d,q} \in A\}|$
      $w_{e,d} = n_{e,d} \cdot \epsilon + B_\ell$.
      $(L_{e,d}, H_{e,d}) = (s, s + n_{e,d} \cdot \epsilon + B_\ell)$
      $s = s + n_{e,d} \cdot \epsilon + B_\ell$.
    end for
  end if
  $p_e = \sum_{m=1}^{m=n_e} P_e(m) + D_e \cdot \zeta$.
end for

B. Analysis of Mechanism

Now we formally analyze SAS. We first prove the truthfulness of SAS. Then, we prove its optimality.

Theorem 2. *In single collision domain, SAS is truthful.*

Proof: Consider an arbitrary entity e. Given V_{-e}, the profile of valuation function sets submitted by all entities other than e, consider two possible strategies of entity e: The first strategy is that entity e submits its true valuation function set V_e^T, while the second is that entity e submits an arbitrary valuation function set V_e^A. Clearly, these two strategies may lead to different values of variables in our SAS mechanism, and thus different payoffs of entity e. For convenience, we use superscript T to denote the value of a variable when V_e^T is submitted, e.g., n_e^T is the value of n_e when V_e^T is submitted; correspondingly, we use superscript A to denote the value of a variable when V_e^A is submitted, e.g., n_e^A is value of n_e when V_e^A is submitted. Those values that are not affected by entity e's submitted valuation function set remain without either superscript, e.g., $b_{e',d,j}$ for $e' \neq e$. It is easy to get that

$$
\begin{aligned}
& \mathsf{payoff}_e(V_e^T, V_{-e}) \\
= & \sum_{d=1}^{D_e} v_{e,d}^T(w_{e,d}^T) - p_e^T \\
= & \sum_{d=1}^{D_e} v_{e,d}^T(n_{e,d}^T \cdot \epsilon + B_\ell) - \sum_{m=1}^{n_e^T} P_e^T(m) - D_e \cdot \zeta \\
= & \sum_{d=1}^{D_e} (v_{e,d}^T(n_{e,d}^T \cdot \epsilon + B_\ell) - v_{e,d}^T(B_\ell)) - \sum_{m=1}^{n_e^T} P_e^T(m) \\
& + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta \\
= & \sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}^T} (v_{e,d}^T(m\epsilon + B_\ell) - v_{e,d}^T((m-1)\epsilon + B_\ell)) \\
& - \sum_{m=1}^{n_e^T} P_e^T(m) + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta \\
= & \sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}^T} b_{e,d,m}^T - \sum_{m=1}^{n_e^T} P_e^T(m) + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) \\
& - D_e \cdot \zeta.
\end{aligned}
$$

Similarly,

$$
\mathsf{payoff}_e(V_e^A, V_{-e}) = \sum_{d=1}^{D_e} v_{e,d}^T(w_{e,d}^A) - p_e^A = \ \ldots
$$

$$
= \sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T - \sum_{m=1}^{n_e^A} P_e^A(m) + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta.
$$

Hence,

$$
\begin{aligned}
& \mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \\
& = (\sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}^T} b_{e,d,m}^T - \sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T) \\
& \quad - (\sum_{m=1}^{n_e^T} P_e^T(m) - \sum_{m=1}^{n_e^A} P_e^A(m)).
\end{aligned}
$$

Let ${B'}_e^T$ (resp., ${B'}_e^A$) be the subsequence of ${B'}^T$ (resp., ${B'}^A$) that consists of all elements $b_{e,d,j}$ for all d and all j.

By our SAS mechanism, clearly we have that

$$\sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} b_{e,d,m}^T = \sum_{m=1}^{n_e^T} {B'}_e^T(m). \quad (2)$$

On the other hand, $\sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T$ is the sum of n_e^A elements of ${B'}_e^T$. Let M_A be the set of indices for these elements. Then, we have $|M_A| = n_e^A$ and that

$$\sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T = \sum_{m\in M_A} {B'}_e^T(m). \quad (3)$$

Now let B'_{-e} be the sequence we obtain by removing all elements of ${B'}_e^T$ from ${B'}^T$. Note that when we remove all elements of ${B'}_e^A$ from ${B'}^A$, we get the same sequence B'_{-e}. It is not hard to see

$$\sum_{m=1}^{n_e^T} P_e^T(m) = \sum_{m=N-n_e^T+1}^{N} B'_{-e}(m); \quad (4)$$

$$\sum_{m=1}^{n_e^A} P_e^A(m) = \sum_{m=N-n_e^A+1}^{N} B'_{-e}(m). \quad (5)$$

Combining (2)(3)(4)(5), we get that

$$\begin{aligned}&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e})\\ =&(\sum_{m=1}^{n_e^T} {B'}_e^T(m) - \sum_{m\in M_A} {B'}_e^T(m))\\ &-(\sum_{m=N-n_e^T+1}^{N} B'_{-e}(m) - \sum_{m=N-n_e^A+1}^{N} B'_{-e}(m))\end{aligned}$$

We distinguish two cases:
Case A: $n_e^T \geq n_e^A$.

$$\begin{aligned}&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e})\\ =&(\sum_{m=1}^{n_e^A} {B'}_e^T(m) - \sum_{m\in M_A} {B'}_e^T(m))\\ &+\sum_{m=n_e^A+1}^{n_e^T} {B'}_e^T(m) - \sum_{m=N-n_e^T+1}^{N-n_e^A} B'_{-e}(m)\\ \geq& 0 + (n_e^T - n_e^A){B'}_e^T(n_e^T)\\ &-(n_e^T - n_e^A){B'}_{-e}(N - n_e^T + 1)\\ =&(n_e^T - n_e^A)({B'}_e^T(n_e^T) - {B'}_{-e}(N - n_e^T + 1))\end{aligned}$$

The inequality above is due to the fact that ${B'}_e^T$ and ${B'}_{-e}^T$ are both sorted from the largest to the smallest. From SAS, we can see that ${B'}_e^T$ has n_e^T elements in A^T, i.e., in the top N elements of ${B'}^T$. Hence,

$${B'}_e^T(n_e^T) \geq {B'}^T(N).$$

This implies that ${B'}_{-e}$ has $N - n_e^T$ elements in the top N elements of ${B'}^T$. Hence,

$${B'}_{-e}(N - n_e^T + 1) \leq {B'}^T(N).$$

Combining all the above three inequalities, we get that

$$\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \geq 0.$$

Case B: $n_e^T < n_e^A$. We partition M_A into two subsets $M_{A,1}$ ($|M_{A,1}| = n_e^T$) and $M_{A,2}$ ($|M_{A,2}| = n_e^A - n_e^T$), such that the elements with indices in $M_{A,1}$ are the largest n_e^T elements with indices in M_A.

$$\begin{aligned}&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e})\\ =&(\sum_{m=1}^{n_e^T} {B'}_e^T(m) - \sum_{m\in M_{A,1}} {B'}_e^T(m))\\ &-\sum_{m\in M_{A,2}} {B'}_e^T(m) + \sum_{m=N-n_e^A+1}^{N-n_e^T} B'_{-e}(m)\\ \geq& 0 - \sum_{m\in M_{A,2}} {B'}_e^T(m) + (n_e^A - n_e^T){B'}_{-e}(N - n_e^A + 1)\end{aligned}$$

Again, the inequality above is due to the fact that ${B'}_e^T$ and ${B'}_{-e}^T$ are both sorted from the largest to the smallest. Recall that ${B'}_e^T$ has n_e^T elements in the top N elements of ${B'}^T$. Hence, $\{{B'}_e^T(m)|m \in M_A\}$ has at most n_e^T elements in the top N elements of ${B'}^T$. Consequently, $\{{B'}_e^T(m)|m \in M_{A,2}\}$ has no element in the top N elements of ${B'}^T$, which implies that, for all $m \in M_{A,2}$,

$${B'}_e^T(m) \leq {B'}^T(N).$$

On the other hand, similar to Case A, ${B'}_{-e}$ has $N - n_e^T$ elements in the top N elements of ${B'}^T$. Since $N - n_e^A + 1 < N - n_e^T + 1$,

$${B'}_{-e}(N - n_e^A + 1) \geq {B'}^T(N).$$

Combining all the above three inequalities, we get that

$$\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \geq 0.$$

To summarize, for both Case A and Case B we have shown that

$$\mathsf{payoff}_e(V_e^T, V_{-e}) \geq \mathsf{payoff}_e(V_e^A, V_{-e}).$$

Hence, all entities submitting true valuation function sets is a DSE, which means SAS is truthful. ∎

Theorem 3. *(Optimality) In single collision domain, SAS achieves optimality.*

Proof: SAS clearly guarantees no starvation. So we only need to show that it maximizes the total valuation subject to the constraint of no starvation.

Let $\Psi = \{w_{e,d}|\forall i, \forall d\}$ be the spectrum assignment result of SAS. Let $\Psi' = \{w'_{e,d}|\forall i, \forall d\}$ be the spectrum assignment result of an arbitrary different mechanism, such that there is no starvation. It is easy to get that

$$\begin{aligned}
&\sum_e \sum_{d=1}^{D_e} v_{e,d}(w_{e,d}) - \sum_e \sum_{d=1}^{D_e} v_{e,d}(w'_{e,d}) \\
=& \sum_e \sum_{\substack{1\le d\le D_e \\ w'_{e,d}=w_{e,d}}} (v_{e,d}(w_{e,d}) - v_{e,d}(w'_{e,d})) \\
&+ \sum_e \sum_{\substack{1\le d\le D_e \\ w'_{e,d}<w_{e,d}}} (v_{e,d}(w_{e,d}) - v_{e,d}(w'_{e,d})) \\
&+ \sum_e \sum_{\substack{1\le d\le D_e \\ w'_{e,d}>w_{e,d}}} (v_{e,d}(w_{e,d}) - v_{e,d}(w'_{e,d})) \\
=& \sum_e (\sum_{\substack{1\le d\le D_e \\ w'_{e,d}<w_{e,d}}} (v_{e,d}(n_{e,d}\cdot\epsilon + B_\ell) - v_{e,d}(n'_{e,d}\cdot\epsilon + B_\ell)) \\
&+ \sum_{\substack{1\le d\le D_e \\ w'_{e,d}>w_{e,d}}} (v_{e,d}(n_{e,d}\cdot\epsilon + B_\ell) - v_{e,d}(n'_{e,d}\cdot\epsilon + B_\ell))) \\
=& \sum_e (\sum_{\substack{1\le d\le D_e \\ n'_{e,d}<n_{e,d}}} \sum_{m=n'_{e,d}+1}^{n_{e,d}} b_{e,d,m} - \sum_{\substack{1\le d\le D_e \\ n'_{e,d}>n_{e,d}}} \sum_{m=n_{e,d}+1}^{n'_{e,d}} b_{e,d,m})
\end{aligned}$$

It is also clear that,

$$\sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}<n_{e,d}}} \sum_{m=n'_{e,d}+1}^{n_{e,d}} B'(N) = \sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}>n_{e,d}}} \sum_{m=n_{e,d}+1}^{n'_{e,d}} B'(N) \tag{6}$$

From SAS, we can see that

$$\sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}<n_{e,d}}} \sum_{m=n'_{e,d}+1}^{n_{e,d}} b_{e,d,m} \ge \sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}<n_{e,d}}} \sum_{m=n'_{e,d}+1}^{n_{e,d}} B'(N), \tag{7}$$

and that

$$\sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}>n_{e,d}}} \sum_{m=n_{e,d}+1}^{n'_{e,d}} b_{e,d,m} \le \sum_e \sum_{\substack{1\le d\le D_e \\ n'_{e,d}>n_{e,d}}} \sum_{m=n_{e,d}+1}^{n'_{e,d}} B'(N). \tag{8}$$

From equations (6) (7) and (8), we can obtain that

$$\begin{aligned}
&\sum_e \sum_{d=1}^{D_e} v_{e,d}(w_{e,d}) - \sum_e \sum_{d=1}^{D_e} v_{e,d}(w'_{e,d}) \\
=& \sum_e (\sum_{\substack{1\le d\le D_e \\ n'_{e,d}<n_{e,d}}} \sum_{m=n'_{e,d}+1}^{n_{e,d}} b_{e,d,m} \\
&- \sum_{\substack{1\le d\le D_e \\ n'_{e,d}>n_{e,d}}} \sum_{m=n_{e,d}+1}^{n'_{e,d}} b_{e,d,m}) \ \ge\ 0.
\end{aligned}$$

Therefore, SAS achieves optimality. ■

IV. Truthful and Approximately Optimal Mechanism for Multiple Collision Domains

In the previous section, we have proposed a spectrum assignment mechanism for single collision domain and proved its truthfulness and optimality. However, if we consider multiple collision domains, the problem becomes extremely challenging. In fact, even if all entities claim the true spectrum demands, there is little hope that we can achieve optimality.

Theorem 4. *In multiple collision domains, it is NP-hard to compute a spectrum assignment that maximizes the total valuation subject to the constraint of no starvation.*

We skip the proof of this theorem, which is similar to other NP-hardness results for spectrum assignment, e.g., [22]. Given this theorem, we have to weaken our objective for multiple collision domains. Our weakened objective is to find a mechanism that is both truthful and constant-ratio approximately optimal.

A. Design of Mechanism

Assume that the interference graph is given as input. We build SAM, a truthful and constant-ratio approximately optimal mechanism for multiple collision domains, based on two ideas: greedy pair-wise spectrum assignment and highest-conflicting-valuation payment.

Just like SAS, SAM also assigns spectrum in two steps: The first step is still to guarantee that there is no starvation. The second step is also a greedy assignment, but it differs from SAS in that it assigns spectrum slices in pairs. Below are more details of these two steps.

In the first step, SAM reserves a continuous spectrum of bandwidth B_ℓ for each device, such that there is no interference between any two devices (Algorithm 2 line 5-11). In other words, any two neighbors in the interference graph are assigned two spectra that do not overlap. (Hereafter, we often use "neighbor" to refer to a neighbor in the interference graph.) Also we make sure that the devices belonging to the same entity are not assigned adjacent spectra.

In the second step, SAM "grows" each device's spectrum under the four following restrictions:

(1) A device cannot be assigned any spectrum slice that overlaps with the spectrum reserved in the first step for any of its neighbors.
(2) The spectrum of a device must grow in *symmetric pairs* of spectrum slices. Specifically, suppose that, in the first step, SAM has reserved $(\mathsf{CF_{e,d}} - \frac{B_\ell}{2}, \mathsf{CF_{e,d}} + \frac{B_\ell}{2})$ for a device where $\mathsf{CF_{e,d}}$ is its center frequency. Then, in the second step, after assigning symmetric pairs of spectrum slices to this device, the device's spectrum can only grow to $(\mathsf{CF_{e,d}} - \frac{B_\ell}{2} - n\epsilon, \mathsf{CF_{e,d}} + \frac{B_\ell}{2} + n\epsilon)$, where $n \ge 0$ is an integer.
(3) The growth of a device's spectrum must start from the symmetric pair closest to its reserved spectrum in the first step, and gradually go to farther symmetric pairs. Once SAM decides not to assign a symmetric pair to the device, the growth must terminate. For example, for device (e, d), the growth of this spectrum goes in the order: $(\mathsf{CF_{e,d}} - \frac{B_\ell}{2} - \epsilon, \mathsf{CF_{e,d}} + \frac{B_\ell}{2} + \epsilon)$, $(\mathsf{CF_{e,d}} - \frac{B_\ell}{2} - 2\epsilon, \mathsf{CF_{e,d}} + \frac{B_\ell}{2} + 2\epsilon)$,... If SAM decides not to assign the symmetric pair $(\mathsf{CF_{e,d}} - \frac{B_\ell}{2} - 2\epsilon, \mathsf{CF_{e,d}} - \frac{B_\ell}{2} - \epsilon)$ and $(\mathsf{CF_{e,d}} + \frac{B_\ell}{2} + \epsilon, \mathsf{CF_{e,d}} + \frac{B_\ell}{2} + 2\epsilon)$ to the device, then the growth terminates and thus the spectrum slice pairs farther than this pair will not be assigned to this device.
(4) SAM decides whether to assign a symmetric pair of spectrum slices to a device based on the valuation of the

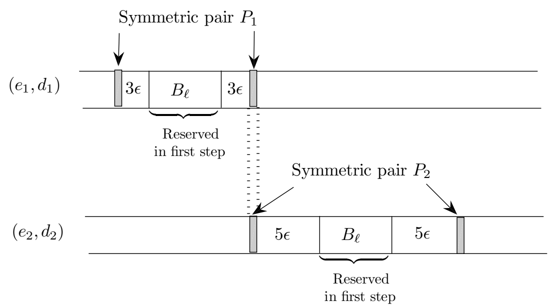

Fig. 1. Example for the 4th restriction.

spectrum slices pair (i.e., how much the valuation of spectrum can increase for this device if this pair is assigned). Specifically, a symmetric pair is assigned to a device only if this device's valuation of this pair of spectrum slices is greater than[2] its neighbors' valuations of all *conflicting pairs*. Here a conflicting pair is just a symmetric pair of spectrum slices for a neighbor (not for this device) that overlaps with this symmetric pair.

In Figure 1, we illustrate the fourth restriction using an example. Here entity e_1's device d_1 and entity e_2's device d_2 are neighbors. Device d_1 has a symmetric pair P_1 and device d_2 has a symmetric pair P_2. For P_1, P_2 is a conflicting pair. In order for SAM to assign P_1 to d_1, we must have that the valuation of P_1 is greater than the valuation of P_2, i.e., $v_{e_1,d_1}(B_\ell + 8\epsilon) - v_{e_1,d_1}(B_\ell + 6\epsilon) > v_{e_2,d_2}(B_\ell + 12\epsilon) - v_{e_2,d_2}(B_\ell + 10\epsilon)$.

Once the growths of all devices' spectra are completed, SAM calculates the payment each entity needs to make for its use of spectrum: This payment is equal to the sum of payments the entity needs to make for each symmetric pair of spectrum slices assigned to each of its devices, and for the spectrum reserved in the first step. For each pair assigned to a device, the amount of payment due is determined by the device's neighbors belonging to other entities. We consider the valuations of conflicting pairs from such neighbors and use the highest such valuation as the payment due.

In the example illustrated in Figure 1, assume that d_2 is the only neighbor of d_1 and that $e_1 \neq e_2$. Then the payment due for usage of P_1 is the valuation of P_2: $v_{e_2,d_2}(B_\ell + 12\epsilon) - v_{e_2,d_2}(B_\ell + 10\epsilon)$.

The entire SAM mechanism is shown in Algorithm 2.

Algorithm 2 SAM: Truthful and constant-ratio approximately optimal mechanism for multiple collision domains

INPUT: Available Space (f_l, f_h); valuation function set V_e from each entity e; the set of neighbors of (e, d): $\mathsf{Neighbr(e, d)}$; parameter ϵ, ζ, B_ℓ, B_u, Δ.
OUTPUT: Assigned spectrum for each device d of each entity e: $(L_{e,d}, H_{e,d})$, and price p_e for each entity e.

$s = \frac{W}{\Delta}$. $N' = \frac{B_u - B_\ell}{2\epsilon}$.
for each device (e, d) **do**
  for each integer x s.t. $1 \le x \le \Delta$ **do**
    if $\forall (e', d') \in \mathsf{Neighbr(e, d)}$, $f_l + (x - \frac{1}{2}) \cdot s \neq \mathsf{CF_{e',d'}}$ **and** $\forall d'' \neq d$, s.t., $|f_l + (x - \frac{1}{2}) \cdot s - \mathsf{CF_{e,d''}}| \neq \mathsf{s}$ **then**
      $\mathsf{CF_{e,d}} = f_l + (x - \frac{1}{2}) \cdot s$.
      $L_{e,d} = \mathsf{CF_{e,d}} - \frac{\mathsf{B}_\ell}{2}$; $H_{e,d} = \mathsf{CF_{e,d}} + \frac{\mathsf{B}_\ell}{2}$.
      Break.
    end if
  end for
  for each integer j s.t. $1 \le j \le N'$ **do**
    $b_{e,d,j} = v_{e,d}(B_\ell + 2 \cdot j\epsilon) - v_{e,d}(B_\ell + 2 \cdot (j-1)\epsilon)$.
  end for
end for
for each device (e, d) **do**
  for $(t = 1; t \le \frac{s - B_\ell}{\epsilon}; t{+}{+})$ **do**

$$\textbf{if } b_{e,d,t} > \max_{\substack{e' \neq e \\ (e',d') \in \mathsf{Neighbr(e,d)} \\ \& |\mathsf{CF_{e,d}} - \mathsf{CF_{e',d'}}| = \mathsf{s}}} b_{e',d',(\frac{s - B_\ell}{2\epsilon} - t + 1)}$$

$$\textbf{then } p_{e,d,t} = \max_{\substack{e' \neq e \\ (e',d') \in \mathsf{Neighbr(e,d)} \\ \& |\mathsf{CF_{e,d}} - \mathsf{CF_{e',d'}}| = \mathsf{s}}} b_{e',d',(\frac{s - B_\ell}{2\epsilon} - t + 1)}.$$

    else Break.
    end if
  end for
  $n_{e,d} = t - 1$.
  $L_{e,d} = L_{e,d} - n_{e,d}\epsilon$; $H_{e,d} = H_{e,d} + n_{e,d}\epsilon$.
end for
for each entity e **do**
  $p_e = \sum_{d=1}^{D_e} \sum_{m=1}^{n_{e,d}} p_{e,d,m} + D_e \cdot \zeta$.
end for

B. Analysis of Mechanism

We have the following theorems regarding the truthfulness and approximation to optimality of SAM.

Theorem 5. *In multiple collision domains, assuming that $\Delta > D_{max}$ and $\Delta > 2D'_{max} - 1$, where D_{max} is the maximum degree among all nodes in the interference graph and D'_{max} is an upper bound for a node's number of neighbors from the same entity, SAM is truthful.*

Proof: Consider an arbitrary entity e. Given V_{-e}, the profile of valuation function sets submitted by all entities other than e, consider two possible strategies of entity e, i.e., submitting its true valuation function set V_e^T and submitting an arbitrary valuation function set V_e^A. Notations such as V_e^T, n_e^T, V_e^A and n_e^A are defined similarly to in the proof of Theorem 2. (Recall that subscript T means the value is for the scenario that e submits V_e^T and subscript A means the value is for the scenario that e submits V_e^A.) Those values that are not affected by entity e's submitted valuation function set remain without either superscript, e.g., $b_{e',d,j}$ for $e' \neq e$.

[2]For simplicity of presentation, we assume there is no tie in valuations of symmetric pairs. If there is a tie, we can easily break the tie, e.g., by the identities of devices. However, we don't include these details in this paper because the notations and formulae would become too complicated.

We can easily get that

$$\begin{aligned}
&\mathsf{payoff}_e(V_e^T, V_{-e}) \\
=\ & \sum_{d=1}^{D_e} v_{e,d}^T(w_{e,d}^T) - p_e^T \\
=\ & \sum_{d=1}^{D_e} v_{e,d}^T(n_{e,d}^T \cdot \epsilon + B_\ell) - \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} p_{e,d,m}^T - D_e \cdot \zeta \\
=\ & \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} (v_{e,d}^T(m\epsilon + B_\ell) - v_{e,d}^T((m-1)\epsilon + B_\ell)) \\
& -\sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} p_{e,d,m}^T + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta \\
=\ & \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} b_{e,d,m}^T - \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^T} p_{e,d,m}^T + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta.
\end{aligned}$$

Similarly, we can obtain that

$$\begin{aligned}
&\mathsf{payoff}_e(V_e^A, V_{-e}) \\
=\ & \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T - \sum_{d=1}^{D_e}\sum_{m=1}^{n_{e,d}^A} p_{e,d,m}^A + \sum_{d=1}^{D_e} v_{e,d}^T(B_\ell) - D_e \cdot \zeta.
\end{aligned}$$

Hence,

$$\begin{aligned}
&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \\
=\ & \sum_{d=1}^{D_e}(\sum_{m=1}^{n_{e,d}^T} b_{e,d,m}^T - \sum_{m=1}^{n_{e,d}^A} b_{e,d,m}^T - (\sum_{m=1}^{n_{e,d}^T} p_{e,d,m}^T \\
& - \sum_{m=1}^{n_{e,d}^A} p_{e,d,m}^A))
\end{aligned}$$

Our mechanism states that each

$$p_{e,d,m} = \max_{\substack{e'\neq e \\ (e',d')\in\mathsf{Neighbr(e,d)} \\ \&|\mathsf{CF_{e,d}}-\mathsf{CF_{e',d'}}|=\mathsf{s}}} b_{e',d',(\frac{s-B_\ell}{2\epsilon}-m+1)}.$$

Since for each m, , $\frac{s-B_\ell}{2\epsilon} - m + 1$ is a fixed number and not affected by e's submitted valuation function set, $p_{e,d,m}$ remains the same regardless of whether e submits V_e^T or V_e^A.

Then $\forall m\ s.t.\ 1 \le m \le \min(n_{e,d}^T, n_{e,d}^A)$ we have

$$p_{e,d,m}^T = p_{e,d,m}^A$$

. We distinguish two cases.

Case A. $n_{e,d}^T \ge n_{e,d}^A$.

$$\begin{aligned}
&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \\
=\ & \sum_{d=1}^{D_e}(\sum_{m=n_{e,d}^A+1}^{n_{e,d}^T} b_{e,d,m}^T - \sum_{m=n_{e,d}^A+1}^{n_{e,d}^T} p_{e,d,m}^T) \\
=\ & \sum_{d=1}^{D_e}(\sum_{m=n_{e,d}^A+1}^{n_{e,d}^T} (b_{e,d,m}^T - p_{e,d,m}^T)) \\
\ge\ & 0
\end{aligned}$$

The inequality above is due to the fact that, for any $m \le n_{e,d}^T$, we must have $b_{e,d,m}^T \ge p_{e,d,m}^T$.

Case B. $n_{e,d}^T < n_{e,d}^A$.

$$\begin{aligned}
&\mathsf{payoff}_e(V_e^T, V_{-e}) - \mathsf{payoff}_e(V_e^A, V_{-e}) \\
=\ & \sum_{d=1}^{D_e}(-\sum_{m=n_{e,d}^T+1}^{n_{e,d}^A} b_{e,d,m}^T + \sum_{m=n_{e,d}^T+1}^{n_{e,d}^A} p_{e,d,m}^A) \\
=\ & \sum_{d=1}^{D_e}\sum_{m=n_{e,d}^T+1}^{n_{e,d}^A}(\max_{\substack{(e',d')\in\mathsf{Neighbr(e,d)} \\ \&|\mathsf{CF_{e,d}}-\mathsf{CF_{e',d'}}|=\mathsf{s}}} b_{e',d',(\frac{s-B_\ell}{2\epsilon}-m+1)} \\
& -b_{e,d,m}^T) \\
\ge\ & 0.
\end{aligned}$$

This inequality holds because $\forall m$ s.t., $m > n_{e,d}^T$,

$$b_{e,d,m}^T \le \max_{\substack{(e',d')\in\mathsf{Neighbr(e,d)} \\ \&|\mathsf{CF_{e,d}}-\mathsf{CF_{e',d'}}|=\mathsf{s}}} b_{e',d',(\frac{s-B_\ell}{2\epsilon}-m+1)}.$$

Therefore, it is a DSE for all entities to submit their true valuation function sets. ■

In addition to truthfulness, we can also prove the following theorem:

Theorem 6. *In multiple collision domains, suppose that* TOV *is the total valuation of the spectrum assigned to the devices computed by SAM, and* OPT *is the maximum total valuation under the constraint of no starvation, then there is a constant* $\Phi > 1$ *such that* $\mathsf{OPT} \le \Phi \cdot \mathsf{TOV}$.

Proof: Consider a system that has to be allocated spectrum with a total bandwidth of W. Let $w_{e,d}$ be the bandwidth allocated to device $\{e,d\}$ under the allocation by SAM, and let $\mathsf{TOV}_{e,d}$ be the valuation of spectrum assigned to device $\{e,d\}$ under the allocation by SAM. Then, we have

$$\mathsf{TOV}_{e,d} = \mathsf{v_{e,d}}(w_{e,d}).$$

Let $w'_{e,d}$ be the bandwidth assigned to device $\{e,d\}$ by the steps in line 4-15 of SAM. It is not difficult to see that $w'_{e,d} \ge B_\ell$. Let $w''_{e,d}$ be the bandwidth assigned to device $\{e,d\}$ by greedy assignment in line 16-25 of SAM. Since the valuation function is an increasing quasi-concave function, we can obtain that

$$\begin{aligned}
\frac{\mathsf{TOV}_{e,d}}{\mathsf{OPT}_{e,d}} &= \frac{\mathsf{v_{e,d}}(w_{e,d})}{\mathsf{v_{e,d}}(w_{e,d}^*)} \\
&= \frac{\mathsf{v_{e,d}}(w'_{e,d} + w''_{e,d})}{\mathsf{v_{e,d}}(w_{e,d}^*)} \\
&\ge \frac{\mathsf{v_{e,d}}(B_\ell + w''_{e,d})}{\mathsf{v_{e,d}}(W)}.
\end{aligned}$$

Define $\nu = \min_{e,d} v_{e,d}(B_\ell)$ and $\rho = \max_{e,d} v_{e,d}(W)$.

Let $w_{e,d}^*$ be the bandwidth allocated to the device d of entity e under the optimal allocation and let $\mathsf{OPT}_{e,d}$ be the valuation of spectrum assigned to the device d of entity e, under the optimal allocation.

Clearly, we have

$$\mathsf{OPT}_{e,d} = \mathsf{v_{e,d}}(w_{e,d}^*).$$

Now we assume there are K devices in the system, and consider the total valuation for the system.

$$\begin{aligned}\frac{\mathsf{TOV}}{\mathsf{OPT}} &= \frac{\sum_e \sum_d v_{e,d}(w'_{e,d} + w''_{e,d})}{\sum_e \sum_d v_{e,d}(w^*_{e,d})} \\ &\geq \frac{\sum_e \sum_d v_{e,d}(B_\ell)}{\sum_e \sum_d v_{e,d}(W)} \\ &\geq \frac{K \cdot \min_{e,d} v_{e,d}(B_\ell)}{K \cdot \max_{e,d} v_{e,d}(W)} \\ &= \frac{\nu}{\rho}.\end{aligned}$$

Thus, we have $\mathsf{OPT} \leq \frac{\rho}{\nu}\mathsf{TOV}$.

■

V. Implementation of Virtual Money

Recall that all our spectrum assignment mechanisms are based on virtual money. We propose that each DSA device should be preloaded with a constant amount of virtual money. Each device consumes the virtual money stores in it when it is assigned some spectrum to use. When the virtual money is used up in a device, if the device owner still needs to access more spectrum, she can choose to purchase additional virtual money using real money and reload some virtual money into the the device through the connection with the central bank of virtual money. (The device owner can also buy virtual money early, before using up the preloaded virtual money.) In this section, we present an approach to implement virtual money using reverse hash chains, which has low computational overheads but can provide a reasonable security guarantee.

Approach Using Reverse Hash Chains The main idea of our approach is to use a reverse hash chain. Recall that, each device is preloaded with an amount of virtual money. Suppose the amount of preloaded money is m. Let (k_{prv}, k_{pub}) be a pair of keys for the central bank of virtual money, where k_{prv} is the private key and k_{pub} is the public key. (We assume that k_{pub} is known to every device.) To implement the preloading of this amount of virtual money, we store a tuple $< r_i, H^m(r_i), S_{k_{prv}}() >$ in each device i, where r_i is a random number only known to the device, $H()$ is a well-known cryptographic hash function (e.g., SHA-512), and $S()$ is a digital signature algorithm. Intuitively, $(r_i,\ H(r_i),\ H^2(r_i),\ \ldots,\ H^m(r_i))$ forms a hash chain of length m, which represents the amount of preloaded virtual money. The device i does not need to keep the entire hash chain; in stead, it only keeps the head r_i, the tail $H^m(r_i)$, and the central bank's signature on the tail.

The first time a device i uses DSA, it broadcasts $< H^m(r_i), S_{k_{prv}}(H^m(r_i)) >$, i.e., the tail of the hash chain and the signature on the tail, to all other devices. Upon receipt of this message, each device verifies that the signature is valid using k_{pub}, and then makes a record of the received tail. Note that each device maintains a record of the current tail for each other device.

When device i needs to make a payment of μ ($\mu \leq m$), it does so by revealing to the public the subchain of length μ at the current tail, and then removing this subchain from its current chain. To be more precise, assuming the current tail kept by device i is $H^{m'}(r_i)$, device i simply broadcasts $H^{m'-\mu}(r_i)$ to all devices, and then replaces the current tail in its record with $H^{m'-\mu}(r_i)$.

Correspondingly, when device j receives hash value φ_i from device i for a payment of amount μ, assuming the current tail for device i in AP j's record is ϑ_i, device j needs to verify that $\vartheta_i = H^\mu(\varphi_i)$. After the verification, device j replaces the current tail for device i in its own record with φ_i.

Fast Computing of Hash-Chain Tail The above approach for implementing virtual money requires frequent computing of hash-chain tails. For example, when device i needs to make a payment of μ, it needs to compute $H^{m'-\mu}(r_i)$. If $m' - \mu$ is large, it may take some time to compute $H^{m'-\mu}(r_i)$ from r_i. We propose a simple way to expedite this computation: For a constant LEN, device i should also be preloaded with $H^{\mathsf{LEN}}(r_i)$, $H^{2\cdot\mathsf{LEN}}(r_i)$, $H^{3\cdot\mathsf{LEN}}(r_i)$,..., in addition to the head and tail of the hash chain and the signature on the tail. In this way, when device i needs to compute $H^{m'-\mu}(r_i)$, the device only needs to compute it from $H^{\mathsf{M}\cdot\mathsf{LEN}}(r_i)$, where $\mathsf{M}\cdot\mathsf{LEN}$ is the largest multiple of LEN less than or equal to $m' - \mu$. Consequently, device i needs much less time to compute $H^{m'-\mu}(r_i)$.

VI. Evaluations

We evaluate our spectrum assignment mechanisms in various settings using GloMoSim [1]. We carry out three sets of experiments for different objectives.

- The first set of experiments evaluate how the payoff of an entity is affected by its possible cheating actions (in claiming its valuation function set). The results demonstrate that, when either SAS or SAM is used, entities' cheating actions never increase their own payoffs.
- The second set of experiments evaluate the total valuation of assigned spectrum in the system. The results demonstrate that, in a single collision domain, when a cheating entity appears, SAS can prevent the total valuation of assigned spectrum from decreasing; In multiple collision domains, SAM also achieves good efficiency in spectrum utilization.
- The third set of experiments are on the overheads introduced by the payment scheme. The results have confirmed the efficiency of our scheme.

A. Experiments Setup

The experiments are performed on a laptop with 2.0GHz Centrino CPU and 1.96GB RAM. We modify GloMoSim to enable the use of variable spectrum width, by setting the MAC layer parameters described in [3]. The payment scheme is implemented with SHA-512 from Cryptopp Library 5.2.1 [4].

Unless specified otherwise, we assume that 3 entities, each of whom has 2 devices, are randomly located in an area of 300×300 m^2 (for single collision domain experiments), or 600×600 m^2 (for multiple collision domains experiments). The transmission power of each device is 16dBm. The path loss is set to free space. In all experiments except those in Section VI-C, we assume that the available band is 48MHz in DTV whitespace (644MHz-692MHz). All traffic is single hop UDP flows that are always backlogged. We set the packet size to 1500 Bytes.

In our experiments, we assume each valuation function is in one of the following two forms:

$$v_{e,d}(w_{e,d}) = \begin{cases} \beta_{e,d} \log(1 + \gamma_{e,d} \cdot w_{e,d}) & \text{if } w_{e,d} < 1/\gamma_{e,d} \\ \beta_{e,d} \log 2 & \text{if } w_{e,d} \geq 1/\gamma_{e,d}. \end{cases} \tag{9}$$

$$v_{e,d}(w_{e,d}) = \begin{cases} \beta_{e,d}\sqrt{\gamma_{e,d} \cdot w_{e,d}}, & \text{if } w_{e,d} < 1/\gamma_{e,d} \\ \beta_{e,d}\sqrt{2} & \text{if } w_{e,d} \geq 1/\gamma_{e,d}. \end{cases} \tag{10}$$

The difference in devices' valuation functions is reflected by the difference in the values of $\beta_{e,d}$ and $\gamma_{e,d}$. When entities submit their valuation function sets, they may cheat by changing their values of $\beta_{e,d}$ and $\gamma_{e,d}$. When an entity is truthful, it should use the true values of $\beta_{e,d}$ and $\gamma_{e,d}$, denoted by $\beta^\star_{e,d}$ and $\gamma^\star_{e,d}$. We assume $\gamma^\star_{e,d} = 1/(N_{e,d}{*}1M)$, where $N_{e,d} * 1M$ is the spectrum demand of device (e, d). In experiments, we randomly set each $N_{e,d}$ as an integer in $[1, 20]$. We set $B_\ell = 6$MHz, $B_u = 40$MHz, $\epsilon = 1$MHz and $\zeta = 0.1$/MHz.

B. Truthfulness and Payoffs

In this set of experiments, we study the truthfulness of our mechanisms. In particular, we evaluate how the cheating behavior of entities affects their own payoffs. In each experiment, one random entity is picked to be the cheater; its claimed valuation function set has each $\beta_{e,d}$ (resp., $\gamma_{e,d}$) randomly chosen between 0 and $3\beta^\star_{e,d}$ (resp., $3\gamma^\star_{e,d}$).[3] We measure the payoff of the cheating entity in each experiment and also the same entity's payoff when the entity behaves honestly. The difference is the entity's payoff change for cheating. If the change is positive, then cheating benefits the entity; otherwise, cheating does not benefit.

Payoffs in SAS We perform the above experiments on SAS, with 1000 runs using valuation functions in the form of (9) and another 1000 runs using valuation functions in the form of (10). From Fig. 2 (a) we can observe that the payoff change when cheating is never positive. In other words, entities never benefit from, and usually lose for, cheating. The average payoff loss when cheating is 18.94. Similar observations can be made from Fig. 2 (b). In this case, the average payoff loss when cheating is 14.62. Overall, the truthfulness of SAS is verified.

Payoffs in SAM We also perform similar experiments on the two mechanisms for multiple collision domains. Fig. 3

[3]We have this random choice of cheater and cheater's action because it is hard to predict who will be the cheater and how the cheater will behave in reality. By repeating this experiment for many times, we hope that at least some of the randomly picked cheating actions will be consistent with real cheaters' actions in reality.

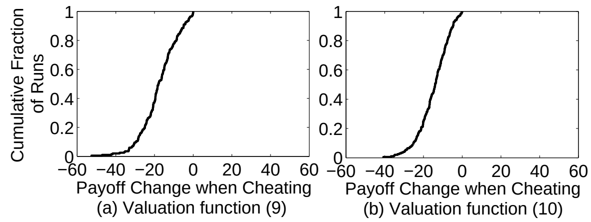

Fig. 2. Payoff change for SAS. It shows that the payoff change when cheating is never positive when SAS is used.

shows the results for SAM. We can see that, if SAM is used, an entity's cheating can never benefits itself (i.e., there is no positive payoff change for cheating). Consequently, the truthfulness of SAM is verified.

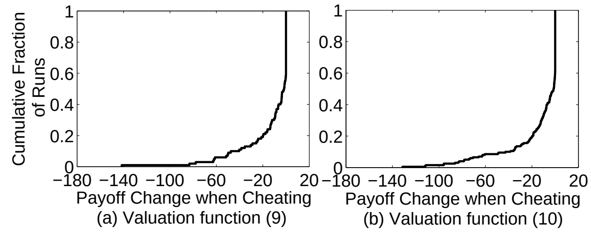

Fig. 3. Payoff change for SAM. It shows that there is no positive payoff change for cheating when SAM is used.

C. Total Valuation

The second set of experiments are to evaluate our two mechanisms in terms of satisfying spectrum demands.

For single collision domain, we measure the total valuation of assigned spectrum for SAS and compare it with the case that there is no payment scheme enforced in the system and one entity cheats in its submission of valuation functions. The result distributions shown in Fig. 4 demonstrate that SAS which achieves optimal spectrum utilization can significantly increase (7.62% on average) the total valuation of assigned spectrum with the presence of one cheating entity.

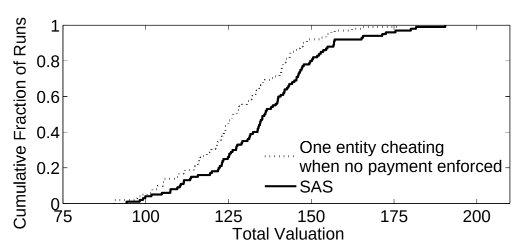

Fig. 4. Total valuation of assigned spectrum in single collision domains.

For multiple collision domains, we measure the total valuations of assigned spectrum for SAM in two different bands, the 2.4GHz ISM band and the DTV whitespaces, respectively. We assume that there are 80MHz available bandwidth in the 2.4GHz ISM band, and 48 MHz available bandwidth (644MHz-692MHz) in DTV whitespaces. Fig.

5 shows the distributions of total valuation of assigned spectrum of 100 runs, for our mechanism SAM. In the figure, we can see that, for both the 2.4GHz ISM band and the DTV whitespace, the total valuation of assigned spectrum in the system remains at a high level. Since the there are more bandwidth available in 2.4GHz ISM band, system-wide total valuation is higher than that of DTV whitespace.

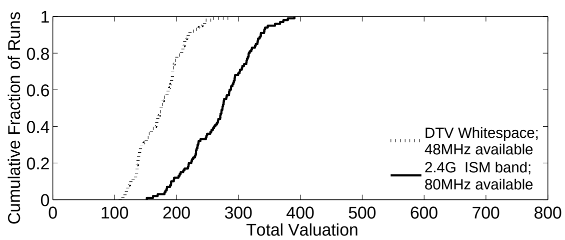

Fig. 5. Total valuation of assigned spectrum in multiple collision domains.

D. Computational Overhead

In this set of experiments, we evaluate the computational overhead introduced by our payment scheme. In particular, we distinguish two types of computational overhead, namely the overhead for computing the hash value in order to make a payment and the overhead for verifying a payment, and evaluate both of them. In these experiments, the amount of preloaded virtual money is 1000; $\mathsf{LEN} = 100$ for fast computing of hash-chain tails; the key length is 1024 bits.

For the first type of overhead, we measure the average amounts of time for a device to compute a payment using different methods: the basic method of directly computing the hash value from the head of the hash chain, and the fast hash-chain tail computation method given in Section V. The results in Fig. 6 show that the basic method is pretty fast, but the fast hash-chain tail computation method is even faster. We also observe that for the basic method, making later payments is faster than making earlier payments. The reason is that the length of the hash chain decreases over time, and thus making later payments requires fewer numbers of hashing.

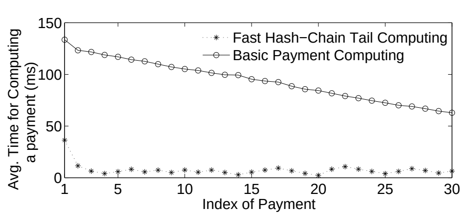

Fig. 6. Average computational overhead for making a payment.

We also evaluate the overhead for verifying a payment. From the results of 100 runs, we find that the average time for verifying a payment is 3.50 ms, and the standard deviation is about 0.32ms.

VII. Related Work

DSA has been studied extensively [13], [22], [3], [8], [16], [2]. In the KNOWS [22], [21], [17] project, the concept of time-spectrum block is introduced and close-to-optimal central and distributed spectrum allocation algorithms [22] are proposed. In [16], Moscibroda et al. design algorithms to assign dynamic channel width that matches the traffic load. Their results show that load-aware dynamic spectrum allocation can significantly improve the spectrum utilization. Another important recent contribution [5] in DSA is on DTV whitespaces. In this work, in addition to providing some basic design rules and an architecture, Deb et al. also present a demand-based dynamic spectrum allocation algorithm that achieves high performance.

Our work differs significantly from the above works, because we study the incentive problem, which is not considered in any of the above works.

Incentives are also considered in existing works on spectrum auctions [11], [24], [23], [14], [12], [25]. In [23], Zhou et al. propose a truthful and computationally efficient auction scheme; in [25], Zhou and Zheng make an important improvement by considering the incentives of the spectrum seller. Another truthful spectrum auction scheme is presented in [12] for generating more revenue from the auctions. Although our work may appear to be similar to spectrum auctions in some aspects, there are fundamental differences in the settings. Spectrum auctions are on *licensed* access to the spectrum, where the seller is the license holder, and the bidders must purchase the spectrum they want to access from the seller. In contrast, in the scenario we consider, there is no seller of spectrum. Spectrum auction sell spectrum in units of channels and thus only allows bidding using per-channel price, but our work guarantee that users are assigned a contiguous but variable bandwidth, depending on the users demand.

There are also a number of works on non-cooperative channel assignment problem in wireless networks [9], [19], [6], [20], [7]. For multiple radio devices, Felegyhazi et al. [6] introduce a strategic game model and obtain elegant theoretical results. After this work, Wu et al. [20] propose a solution based on strictly dominant strategies, and Gao et. al. [7] obtain interesting results in multi-hop networks. As we have mentioned, all these works are on assignment of fixed-width channels, rather than on assignment of general spectrum.

VIII. Conclusion and Future Work

In dynamic spectrum access, there may be an incentive problem if the DSA devices are owned by multiple entities. To solve this problem, we propose two spectrum assignment mechanisms with provable properties, for single collision domain and multiple collision domains, respectively.

To summarize, we are the first to study this important problem. So, there are many possible ways to further improve our mechanisms, e.g., better approximation mechanisms for multiple collision domains, or better methods to implement the mechanisms.

REFERENCES

[1] Ucla parallel comupting laboratory. information systems simulation library. http://pcl.cs.ucla.edu/projects/glomosim/.

[2] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh. White space networking with wi-fi like connectivity. In *Proc. of ACM SIGCOMM*, 2009.

[3] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl. A case for adapting channel width in wireless networks. In *Proc. of ACM SIGCOMM 2008*.

[4] W. Dai. Crypto++5.5.2. available at http://www.eskimo.com/ weidai/ cryptlib.html.

[5] S. Deb, V. Srinivasan, and R. Maheshwari. Dynamic spectrum access in dtv whitespaces: Design rules, architecture and algorithms. In *Proc. of ACM MOBICOM*, 2009.

[6] M. Felegyhazi, M. Cagalj, S. S. Bidokhti, and J.-P. Hubaux. Non-cooperative multi-radio channel allocation in wireless networks. In *Proc. of IEEE INFOCOM*, 2007.

[7] L. Gao and X. Wang. A game approach for multi-channel allocation in multi-hop wireless networks. In *Proc. of ACM MOBIHOC*, 2008.

[8] R. Gummadi and H. Balakrishnan. Wireless networks should spread spectrum based on demands. In *Proc. of ACM HotNets-VII*, 2008.

[9] M. M. Halldorsson, J. Y. Halpern, and L. E. L. etc. On spectrum sharing games. In *Proc. of ACM PODC*, 2004.

[10] G. Hardin. The tragedy of the commons. *Science*, 162(3859):1243, 1968.

[11] J. Huang, R. Berry, and M. Honig. Auction-based spectrum sharing. *Mobile Networks and Applications*, 11(3), 2006.

[12] J. Jia, Q. Zhang, Q. Zhang, and M. Liu. Revenue generation for truthful spectrum auction in dynamic spectrum access. In *Proc. of ACM MOBIHOC*, 2009.

[13] T. Kamakaris, M. Buddhikot, and R. Iyer. A case for coordinated dynamic spectrum access in cellular networks. In *Proc. of IEEE DySPAN*, 2005.

[14] G. Kasbekar and S. Sarkar. Spectrum auction framework for access allocation in cognitive radio networks. In *Proc. of ACM MOBIHOC*, 2009.

[15] D. M. Kreps. *A course in microeconomic theory*. Princeton University Press, 1990.

[16] T. Moscibroda, R. Chandra, Y. Wu, S. Sengupta, P. Bahl, and Y. Yuan. Load-aware spectrum distribution in wireless lans. In *Proc. of IEEE ICNP*, 2008.

[17] S. Narlanka, R. Chandra, P. Bahl, and J. I. Ferrell. A hardware platform for utilizing the tv bands with a wi-fi radio. In *Proc. of IEEE LANMAN*.

[18] M. J. Osborne. *A course in game theory*. The MIT Press, 1994.

[19] R. W. Thomas, R. S. Komali, L. A. DaSilva, and A. B. MacKenzie. Joint power and channel minimization in topology control: A cognitive network approach. In *Proc. of IEEE CogNet Workshop*, 2007.

[20] F. Wu, S. Zhong, and C. Qiao. Globally optimal channel assignment for non-cooperative wireless networks. In *Proc. of IEEE INFOCOM*, 2008.

[21] Y. Yuan, P. Bahl, R. Chandra, P. A. Chou, J. I. Ferrell, T. Moscibroda, S. Narlanka, and Y. Wu. Knows: Kognitiv networking over white spaces. In *Proc. of IEEE DySPAN*, 2007.

[22] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, and Y. Wu. Allocating dynamic time-spectrum blocks in cognitive radio networks. In *Proc. of ACM MobiHoc*, 2007.

[23] X. Zhou, S. Gandhi, S. Suri, and H. Zheng. ebay in the sky: strategy-proof wireless spectrum auctions. In *Proc. of ACM MOBICOM*, 2008.

[24] X. Zhou, S. Mettu, H. Zheng, and E. M. Belding. Traffic-driven dynamic spectrum auctions. In *Proc. of IEEE Workshop SDR*, 2008.

[25] X. Zhou and H. Zheng. Trust: A general framework for truthful double spectrum auctions. In *Proc. of IEEE INFOCOM*, 2009.