

Towards Real-Time Dynamic Spectrum Auctions

Sorabh Gandhi, Chiranjeeb Buragohain*, Lili Cao, Haitao Zheng, Subhash Suri

Department of Computer Science

University of California, Santa Barbara, CA 93106 U.S.A

**Amazon.com, Seattle, WA 98106 U.S.A*

Main contact: Haitao Zheng, htzheng@cs.ucsb.edu

Abstract

In this paper, we propose a low-complexity auction framework to distribute spectrum in real-time among a large number of wireless users with dynamic traffic. Our design consists of a compact and highly-expressive bidding format, two pricing models to control tradeoffs between revenue and fairness, and fast auction clearing algorithms to achieve conflict-free spectrum allocations that maximize auction revenue. We develop analytical bounds on algorithm performance and complexity to verify the efficiency of the proposed approach. We also use both simulated and real deployment traces to evaluate the auction framework. We conclude that pricing models and bidding behaviors have significant impact on auction outcomes and spectrum utilization. Any efficient spectrum auction system must consider demand and spectrum availability in local regions to maximize system-wide revenue and spectrum utilization.

1. Introduction

Reliable and efficient spectrum access is vital for the growth and innovation of wireless technologies. Unfortunately, historical (and current) spectrum regulations allocate different spectrum to different technologies using long-term leasing to prevent interference among them. Over time, this has led to significant over-allocation and under-utilization of spectrum, which is slowing down wireless deployments. To realize efficient spectrum usage, we must migrate from the current static spectrum access to dynamic spectrum access.

One promising solution is spectrum trading that applies pricing based incentives to motivate users to sell and lease under-utilized spectrum. One particular form of trading mechanism is *auctions*, which is widely known for providing efficient allocation of scarce resources [3,10]. Sellers use auctions to improve revenue by dynamical pricing based on buyer demands, while buyers benefit from auctions since they assign resources to buyers who value them the most. Example auction systems include energy

markets [3], treasury bonds [2] and commercial goods [10].

In this paper, we consider a *dynamic spectrum auction* system where a seller auctions spectrum among users with dynamic demands while maximizing system revenue. User demands change over time and hence the seller must derive auction results in real-time. Figure 1 illustrates a general spectrum auction scenario where n buyers (wireless service providers) bid for spectrum from a seller (government agencies or spectrum owners) who auctions its spectrum periodically, *i.e.* every hour. These new types of auctions are significantly different from tractional auction systems, and face the following new challenges:

- *Constrained by radio interference.* Auction outcomes are constrained by radio interference. Buyers in close proximity interfere with each other and can not use the same spectrum; while well-separated buyers can reuse the same spectrum. Spectrum auctions need to explicitly account for the impact of interference when determining allocations and prices, making the corresponding op-

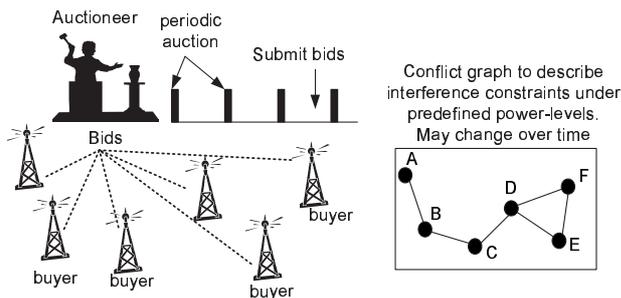


Fig. 1. A dynamic auction scenario. (left) An auctioneer auctions spectrum to buyers periodically. (right) A conflict graph illustrates the interference constraints among buyers.

timization problem NP-complete. Therefore, solving the auction becomes much more complex.

- *Supporting diverse demands.* Users have diverse demands that vary over time, and they expect to obtain and pay for what they need. These include both traditional long-term stable usage and short-term spontaneous usage to support bursty traffic. For example, occasional events like sports and conferences will create a spike in demand at a specific location for a short period of time.
- *Requiring online multi-unit allocations.* In spectrum auctions, users wish to obtain different amount of spectrum at their desired power level, and may be willing to pay differently depending on the assignment. The system needs a *bidding language* to allow buyers conveniently express their desire, and do so compactly. The system needs a fast *allocation* algorithm to distribute resources in real-time.

While there exists a critical need for dynamic spectrum auctions, we observe that existing solutions are not suitable to support real-time operations. In particular, existing solutions for multi-unit auctions apply combinatorial auctions as the most general framework [8]. These auctions require complex bid expression that grows exponentially with the size of goods, and apply complex allocation and pricing process that requires solving NP-hard problems. Hence, they are in general intractable and not suitable for real-time dynamic *hourly* auctions.

In this paper, we consider the problem of real-time dynamic spectrum auctions to distribute spectrum among a large number of buyers in a large geographic area. We focus on computational-efficient channel allocation/pricing algorithms to support large scale networks with real-time spectrum trading. While the problem is NP-hard, we show that by

restricting bids and radio interference constraints judiciously, we can design a practical and efficient auction system that is simple, scalable and yet provides powerful performance guarantees. Our work differs significantly from prior works on spectrum auctions [12,14,20] which assume small scale networks. We also perform extensive experiments to understand the impact of pricing models and bidding behaviors on spectrum utilization and revenue.

This paper makes four key contributions:

- *A compact and highly expressive bidding language – piecewise linear price-demand (PLPD).* Each buyer expresses its demand as the amount of spectrum desired at each particular per-unit price. PLPD can approximate a very broad class of demand curves with high accuracy. It allows bidders to express fairly sophisticated valuations in a single bid, and do so very compactly.
- *Pricing models to explore tradeoffs of revenue and fairness.* We investigate two pricing models, a simple uniform pricing model where all winners pay the same per-unit price, and a discriminatory pricing model where winners’ per-unit prices are different. While the decision of pricing model depends on the tradeoff between revenue and fairness, we focus on designing allocation algorithms for both models and exploring their impact on auction outcomes and user allocations.
- *Low-complexity allocation algorithms with analytical bounds.* While the revenue-maximizing auction problem is NP-hard, we propose low-complexity approximation algorithms to derive prices and allocations. Our algorithms are supported by strong theoretical bounds on performance and complexity. Our algorithms run in polynomial time (1 min for 3500 nodes using a 3.0 GHz processor with 1 GB RAM.) while the optimal solution takes exponential run time (4 hours for 80 nodes).
- *Extensive experiments and evaluations.* We perform extensive experiments to examine the proposed system, and explore the impact of bidding behavior, network topology and pricing model. Results show our algorithms run in real-time and produce near optimal solutions. We conclude that to maximize revenue and spectrum utilization, prices must be determined based on local demand and spectrum availability.

The rest of the paper is organized as follows. In Section 2 we describe the general model of spectrum auction, the impact of wireless interference and dis-

cuss some related work. In Section 3 we propose the auction framework and introduce our bidding language and pricing models. We describe auction clearing algorithms for both pricing models in Section 4 and derive theoretical bounds on performance and complexity in Section 5. We show experimental results in Section 6. We discuss in Section 7 several practical issues related to the proposed framework and conclude in Section 8.

2. Preliminaries and Related Work

This section briefly describe multi-unit auctions, existing solutions and challenges on the problem of spectrum allocation under interference constraints.

2.1. Background on Auctions

Auctions have been widely used to provide efficient allocation of scarce resources, including the sale of single-item indivisible goods, single-item, multi-unit bundles [9,22] and multi-item, multi-unit bundles [8] (*e.g.* bonds). A successful auction system must not only produce financial efficiency [16], but also provide (a) efficient bidding process and (b) fast clearing (or winner determination) process.

First, users express their preferences for various outcomes by submitting bids. There is often an inverse relation between the “expressiveness of the auction” and the computational complexity of determining the winners in the auction. Combinatorial auctions allow users to express their bids over arbitrary subsets of the goods, but are known to be intractable to solve optimally, or even approximately [21].

Next, given a set of bids, auctioneers use *auction-clearing* algorithms to compute the revenue-maximizing prices and allocations. Auction-clearing is simple for single-item single-unit auction: assign the item to the bidder with the highest bid. However, auctioning multi-unit items can be much more complex since multiple winners split the items. The complexity of clearing algorithms also depends on the complexity of bidding language. A comprehensive study of market clearing algorithms for single item, multiple-unit auctions was given by [22].

Auction outcomes and the corresponding clearing algorithms also depend on the pricing model used.

- *Uniform pricing.* The auctioneer determines a per-unit price and applies it to all winning bidders. The auction clearing problem here is to

determine a market-clearing price that maximizes the auctioneer’s revenue. Ebay multi-unit auctions [9] have been using this model.

- *Discriminatory pricing.* The auctioneer charges different prices to different bidders. While producing higher financial revenue, this model is also perceived as less “fair” to bidders than the uniform pricing model.

The various issues that arise in uniform pricing versus discriminatory pricing models have been studied in diverse markets such as US treasury security auction [19], government bonds auction in UK [2], and electricity auctions in California [3,13]. For one time auctions, discriminatory pricing always generates more revenue. On the other hand, uniform pricing is simple, and provides “fairness” to bidders and promotes aggressive bidding [19]. However, uniform pricing is suspect to collusion among the bidders [4] and for an unsettled market, it might be more dangerous with respect to the amount of revenue it generates [19]. Because of these complex factors, we leave the choice of pricing model to auctioneers, and focus on designing efficient bidding language and fast clearing-algorithms for both models.

In addition to having different pricing models, auctions can be one-time or iterative. For one-time auctions, buyers input bids once, and auctioneer determines allocations and prices based on this input. In iterative auctions, the bidding proceeds in multiple rounds where buyers can change their bids from round to round based on market feedback.

2.2. Related Work on Spectrum Auctions

There are multiple complementary ways to design spectrum auctions, each applicable to different scenarios. First, the system can allocate/auction transmit power to minimize interference [12], while all buyers use the same spectrum band. Second, the system can allocate conflicting users with orthogonal channels to avoid interference, and compute appropriate prices and allocations to maximize system utility. Prior work in this category uses cellular network model. The work in [14] uses a demand responsive pricing framework, and applies iterative bidding to maximize social welfare for small scale networks. In [5], the authors propose the general problem in cellular systems and centralized heuristics for small scale networks. Ryan *et al.* [20] proposed a hybrid pricing model to reduce the frequency of auctions –

use simple auctions during peak period while applying a uniform price to all buyers during off-peak.

Interference Constraints in Spectrum Auctions. Spectrum auction differs from conventional auctions because it has to address radio interference. Given bids, the problem of auction-clearing becomes the problem of interference-constrained resource allocation. Next, we briefly discuss the impact of interference and the corresponding spectrum allocation problem.

We start from a sample scenario in Figure 1 where nodes A to F are wireless access points that provide network access for their associated users. Since A and B are located close to each other, their associated users will receive signals from both nodes. Signals from non-associated access points become interference and could disrupt communications. To avoid interference, A and B should not use the same spectrum frequencies. Assuming spectrum consists of M channels, we use F_A and F_B to represent the spectrum assigned to A and B , *i.e.*

$$F_A = \{s_1^A, s_2^A, \dots, s_M^A\}$$

where $s_k^A = 1$ if the k th channel is assigned to A , and otherwise 0. Let $f_A = |F_A|/M$, $f_B = |F_B|/M$ represent the *normalized* spectrum assigned to A and B , respectively. We can represent the interference constraint between A and B as

Interference Constraints: $F_A \cap F_B = \emptyset$, *i.e.* $s_k^A s_k^B = 0, \forall k \in [1, M]$; and $f_A + f_B \leq 1$.

Figure 1 shows the graphic interpretation of the constraints as a *Conflict Graph*. Vertices represent access points, and an edge exists between any two vertices if they conflict. Under interference constraints, we define the auction clearing problem as:

$$\text{Maximize } \sum_{i \in \text{bidders}} f_i p_i(f_i), \quad \text{subject to} \quad (1)$$

$$f_i \leq 1 \quad (2)$$

Interference Constraints (3)

where $p_i(f_i)$ represents the per-unit price that the bidder i pays if he obtains $f_i \cdot M$ unit of spectrum.

This problem is an integer programming problem because the model requires integer-valued variable; it is also non-linear because the objective function is quadratic. This problem is also known to be NP-hard. Jain *et al.* [15] were the first to study a class of related optimization problems and proposed an exponential time algorithm to solve it optimally. The works of [1,6,17] have provided polynomial time approximation algorithms with provable performance

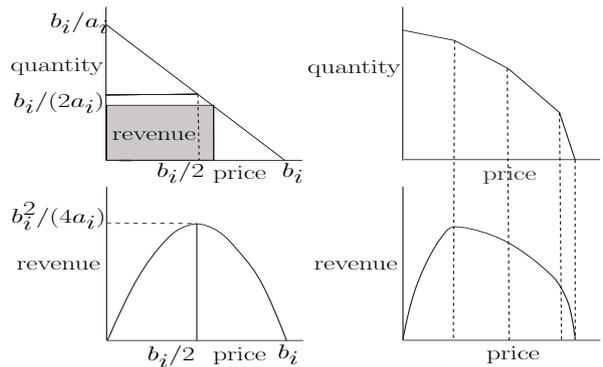


Fig. 2. On the left, linear demand curve (top) and the corresponding revenue generated (bottom) and on the right a concave piecewise linear demand curve (top) and the corresponding piecewise quadratic revenue function.

guarantees to maximize the total system throughput. Our work builds on existing works of [6,22] to solve spectrum auction problems that maximize revenue under interference constraints.

3. Spectrum Auction Framework

To support real-time dynamic spectrum trading, we propose a computational-efficient auction framework with simple and effective bidding and fast auction clearing algorithms. Specifically, buyers use a compact and yet expressive bidding format to express their desired spectrum usage and willingness to pay, while sellers execute fast clearing algorithms to derive prices and allocations under different pricing models. We present the proposed bidding formats and the corresponding optimization problems under different pricing models. We describe fast auction clearing algorithms in Section 4.

We begin by making several assumptions on the spectrum auction system. First, assuming each buyer bids for spectrum with specific but fixed power requirements, we focus on the problem of channel allocation only¹. The seller divides its spectrum into a large number of *homogeneous* channels with equal power limit and transmission bandwidth. We use centralized auction where the seller collects bids and auctions spectrum in single rounds periodically. In Section 7 we discuss extensions to heterogeneous channels, decentralized systems and iterative auctions, as well as practical mechanisms to acquire the

¹ Extensions to joint channel and power allocations are beyond the scope of this paper, and will be addressed in a future study.

knowledge of interference constraints. We model interference constraints using the widely-used *protocol interference model* [15], a succinct model to formulate the impact of interference in resource allocation problems. Using this model, we represent interference constraints as a conflict graph, shown in Figure 1. In Section 7 we will discuss practical considerations on realistic characterization of interference.

3.1. Piecewise Linear Price-Demand (PLPD) Bids

A good bidding language should provide expressive but concise bids. We propose to use piecewise linear demand curves that not only satisfy both requirements, but also lead to low-complexity clearing algorithms. A bidder i expresses the desired quantity of spectrum f_i at each per-unit price p_i using a *continuous concave piecewise linear* demand curve.

A simple example is linear demand curves

$$p_i(f_i) = -a_i f_i + b_i, \quad a_i \geq 0, b_i > 0, \quad (4)$$

where the negative slope represents *price sensitivity* of the buyer – as the per-unit price decreases, demands in general increase. When $a_i > 0$, the revenue produced by each bidder is a *piecewise quadratic* function of the price. Figure 2 shows the quantity $f_i(p_i)$, and the revenue generated $R_i(p_i)$ as a function of the price p_i :

$$f_i(p_i) = \frac{b_i - p_i}{a_i}, \quad 0 \leq p_i \leq b_i \quad (5)$$

$$R_i(p_i) = f_i(p_i)p_i = \frac{b_i p_i - p_i^2}{a_i} \quad (6)$$

For linear demand curves, the revenue is a quadratic function of price, with a unique maximum at $p_i = b_i/2$. Further,

$$p_i \rightarrow 0, \quad R_i(p_i) \rightarrow 0$$

$$p_i \rightarrow b_i, \quad R_i(p_i) \rightarrow 0.$$

We note that any *piecewise linear demand curve* can be expressed as a conglomeration of a set of individual linear pieces (see Figure 2). For ease of explanation, we will use linear demand curves to describe auction problems and solutions. However, our algorithms and proofs easily generalize to concave piecewise linear demand curves.

PLPD is suitable for dynamic spectrum auctions because of the following advantages. First, it is simple and yet highly expressive. PLPD can approximate any arbitrary continuous concave functions,

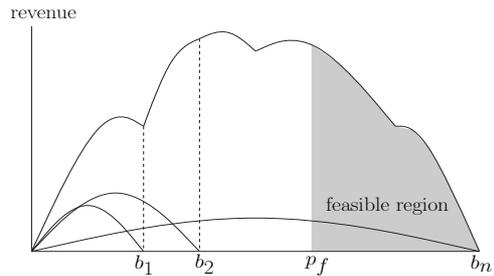


Fig. 3. The revenue as a function of clearing price p in the uniform pricing model.

and hence supports a broad class of demands. Second, using PLPD, the auction is single-shot with sealed-bid. This eliminates the need for iterative bid refinement as well as bidder feedback and signaling, and could prevent collusion among bidders. Since auctioneers no longer need to collect bids iteratively, the use of PLPD is particularly suitable for real-time auctions. Finally, PLPD produces (piecewise) quadratic revenue functions which significantly simplify the auction-clearing problem.

Although auction revenue and efficiency depend on buyer's social and financial strategy and their PLPD formats, we do not address mechanisms to compute the optimal PLPD curves. Instead, we assume that each buyer has its own curve, and focus on how to solve the auction-clearing problem given the bids. We perform experiments in Section 6 to explore the impact of various bidding behaviors, particularly aggressive versus conservative bidding. In Section 7, we also discuss extensions to iterative auctions where buyers adjust their PLPD bids iteratively based on market feedback.

3.2. Pricing Models and Auction-Clearing Problems

We now describe the auction clearing problem under both uniform and discriminatory pricing models. Note that when $a_i = 0$, the clearing problem becomes a classical weighted throughput maximization problems with good solutions [6,7,15]. In this paper, we assume the general cases where $a_i > 0$.

Uniform pricing – The auctioneer sets a clearing price p . Each bidder obtains a fraction of spectrum $f_i(p) = (b_i - p)/a_i$ and produces a revenue of $R_i(p) = (b_i p - p^2)/a_i$. Any bidder i with $b_i \leq p$ gets zero assignment. The optimization problem becomes to search for the revenue-maximizing price p . Without loss of generality, we assume that bidders 1 to n are labeled in increasing order of b_i , i.e. $b_1 \leq b_2 \leq b_3 \leq \dots \leq b_n$. And $b_0 = 0$. For a given price p ,

we compute the revenue $R(p)$ as:

$$R(p) = \sum_{i \in [1, n], b_i > p} R_i(p) = \sum_{i, b_i > p} \frac{b_i p - p^2}{a_i}.$$

Since each $R_i(p)$ is a quadratic function of p , the total revenue is a *continuous piece-wise quadratic* function as shown in Figure 3. Each of the quadratic piece has a parabolic shape.

The overall auction clearing problem becomes

$$\text{Maximize } \sum_{i \in [1, n], b_i > p} \frac{b_i p - p^2}{a_i}, \text{ subject to}$$

Interference Constraints (7)

$$f_i = \frac{b_i - p}{a_i}. \quad (8)$$

Discriminatory pricing – When the clearing prices are non-uniform and vary across i , the optimization problem becomes

$$\text{Maximize } \sum_{i=1}^n (-a_i f_i^2 + b_i f_i), \text{ subject to}$$

Interference Constraints (9)

$$-a_i f_i + b_i \geq 0, \quad f_i \geq 0 \quad (10)$$

3.3. The Optimal Clearing Algorithm

Both clearing problems are NP-hard. We first briefly describe an optimal solution with exponential run time complexity. We will use it as a benchmark for evaluating approximation algorithms.

Consider a single channel of the wireless spectrum. If we allocate this channel to any bidder, none of his neighbors in the conflict graph can be allocated this channel. Thus if we consider a maximal independent set of the conflict graph, then all bidders corresponding to the independent set can use the same channel simultaneously. Based upon this observation, Jain et al. [15] proposed an optimal algorithm to resolve interference conflicts: their approach results in a linear programming (LP) problem with an exponentially large number of constraints. Clearly solving such an LP requires exponentially large amount of time and hence not feasible for large number of bidders. We used a variant of this algorithm in our experiments to produce the optimal solution in order to compare the quality of our approximations. In the next section, we propose fast approximation algorithms to solve these problems in polynomial time.

4. Fast Auction-Clearing Algorithms

In this section, we propose centralized approximation algorithms to solve the auction clearing problems. We show that by judiciously relaxing the interference constraints, we can develop fast approximations to the original NP-hard clearing problems in polynomial time. Note that in this paper, we assume the auctioneer has global information on interference constraints and bids. We will discuss extensions to decentralized auction systems in Section 7.

4.1. Simplifying the Interference Constraints

The auction clearing problem is complex because the discrete interference constraints grow exponentially with the number of buyers. We propose to restrict the interference constraints and reduce them into a number of constraints that grow linearly with the number of buyers. The new constraints are stricter and hence lead to a feasible but sub-optimal solution. We show that analytically this sub-optimal solution can never be too far off from the optimal one.

To simplify the constraints, we assume that the spectrum is finely partitioned into a large number of channels. And each buyer i obtains a normalized allocation of $\{f_i : i = 1, 2, \dots, n\}$ where $f_i \leq 1.0$. For example, a 1MHz spectrum band is divided into 100 channels of 10kHz each. A buyer i with $f_i = 0.143$ will obtain $\lfloor 0.143 \times 100 \rfloor = 14$ channels. In practice this rounding down will lead to some loss of revenue. However, if the number of channels is significantly larger than the highest node degree in the conflict graph, the loss will not lead to undue reduction in revenue. Hence, in the following, f_i behaves as a continuous variable.

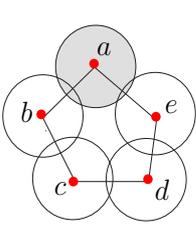
In the following, we refer to each buyer as a node in the conflict graph. We define a neighbor of a node i as any node that interferes with i and hence connects to i in the conflict graph.

Node-ALL Interference Constraints (NI) The simplest constraint is to restrict i and every neighbor of i to use different spectrum channels, *i.e.*

$$f_i + \sum_{j \in N(i)} f_j \leq 1, \quad i = 1, 2, \dots, n \quad (11)$$

where $N(i)$ represents the set of neighbors of i and n represents the total number of nodes.

While leading to simple interference free allocations, this constraint is more restrictive than neces-



Node	NI	NLI	OPT
a	{1}	{2, 3}	{1, 4}
b	{2}	{4, 5}	{2, 5}
c	{3}	{1, 2, 3}	{1, 3}
d	{4}	{4, 5}	{4, 2}
e	{5}	{1}	{3, 5}

Fig. 4. Example network, the conflict graph and the channel allocations by NI (Node-Interference), NLI (Node-L-Interference), and OPT (Optimal). There are a total of 5 channels.

sary. Using a sample topology, Figure 4 illustrates the channel allocation using NI where each node gets only one channel, although node a and d do not conflict with each other and can both use channel 4.

Node-L-Interference Constraints(NLI) We introduce a less restrictive constraint by imposing an order among nodes. By integrating the order in the allocation process, we can achieve much more efficient allocations than that using the NI constraints.

We define the concept of *left of* as: Given two nodes i and j , with location coordinates (x_i, y_i) and (x_j, y_j) , respectively, node i is said to be *left of* node j if either $x_i < x_j$, or $x_i = x_j$ and $y_i < y_j$.

Using this concept, the constraint becomes: the sets of channels assigned to i and all of its left-of neighbors should be different:

$$f_i + \sum_{j \in N_L(i)} f_j \leq 1, \quad i = 1, 2, \dots, n \quad (12)$$

where $N_L(i)$ is the set of neighbors of i lying to its left. Figure 4 compares the allocation results using NLI and NI, and the original constraints (OPT). We see that NLI achieves a more efficient channel allocation than NI.

In the following, we apply NLI constraint to develop approximation algorithms. We show that while it is still more restrictive than the original one, in both theory and practice, algorithms based on NLI produce near-optimal channel allocations in polynomial time. Further, NLI leads to the optimal solution when the conflict graph is a tree.

4.2. A Toy Example: Fixed Per-Unit Price Auctions

To illustrate our algorithm, we start from a simple model where each buyer pays a fixed per-unit price regardless of the allocated amount, *i.e.* $p_i(f_i) = b_i$,

$a_i = 0, \forall i$. We approximate this problem by using NLI as:

Maximize $\sum_i f_i b_i$, subject to

$$f_i + \sum_{j \in N_L(i)} f_j \leq 1 \quad (13)$$

$$0 \leq f_i \leq 1 \quad (14)$$

This is an optimization problem with linear constraints and a linear objective function and hence can be solved easily using linear programming (LP) in polynomial time. The quality of the solution produced by this LP is bounded by the following worst case error guarantee, proved by [6].

Lemma 1

$$R_{LP} \geq \frac{1}{3} R_{OPT}, \quad (15)$$

where R_{LP} is the revenue generated by solving the LP and R_{OPT} is the optimum possible revenue.

Simulation results reveal that this worst-case bound is almost never realized, and the LP solution is very close to the optimal [6].

The above simple example can be solved using linear programming because of its linear objective function. However, the general auction clearing problems are non-linear. Next, we design approximation algorithms for the general auction problems and derive theoretical bounds on the performance and complexity.

4.3. Clearing for uniform pricing (CAUP)

Under NLI, the optimization problem under uniform pricing model becomes

$$\text{Maximize } R(p) = \sum_{i \in [1, n], b_i > p} \frac{b_i p - p^2}{a_i} \quad \text{subject to} \quad (16)$$

$$f_i + \sum_{j \in N_L(i)} f_j \leq 1,$$

$$f_i = \frac{b_i - p}{a_i}. \quad (17)$$

The optimization is to find the optimal price p , which is a one-dimension search process. We propose a two-step solution: first find the feasible values of p subject to interference constraints and then search for the revenue-maximizing p .

Step I: find the feasible region of p subject to interference constraints. We use the following Lemma to simplify the search:

Lemma 2 There exists a unique price p^T where for any $p, p \geq p^T$, the channel allocation according to (17) will satisfy the constraints defined by (16), and for any $p, p < p^T$ results in allocations that violate the constraints.

Proof: Assume that the buyers (1 to n) are sorted by $b_i, b_1 \leq b_2 \dots \leq b_n$. When $p = b_n$, then $f_i = 0, \forall i$. Obviously this allocation is feasible. From (17), as the price decreases, buyers obtain more spectrum and could violate the constraints. If there is a price for which the constraints are violated, reducing the price further will only increase allocations and continue to violate the constraints. ■

Therefore, the feasibility region of p is $[p^T, b_n]$. To find p^T , we use binary search over all possible values of p ranging from 0 to b_n . Let $b_{j-1} \leq p^T < b_j$.

Step II: search for the revenue-maximizing p . We divide the feasible region of p into intervals $(p^T, b_j], (b_j, b_{j+1}], \dots, (b_{n-1}, b_n]$. Within each interval the revenue $R(p)$ is a quadratic function, as explained in Section 3.1. Since every quadratic function has a single maximum, finding the optimal p that maximizes the revenue function in a interval $[b_k, b_{k+1}]$ is straightforward. Hence, by finding the maximum of the revenue function over all feasible intervals we can find the optimal p .

4.4. Clearing for discriminatory pricing (CADP)

Using NLI, the problem becomes

Maximize $\sum_{i=1}^n (-a_i f_i^2 + b_i f_i)$, subject to

$$f_i + \sum_{j \in N_L(i)} f_j \leq 1, \quad (18)$$

$$-a_i f_i + b_i \geq 0, \quad f_i \geq 0 \quad (19)$$

We propose an approximation algorithm using separable programming [11], a special case of semi-definite programming. This method allows one to approximately solve a special class of non-linear programs using linear programming. In addition, semi-definite programming can be solved within an additive error of ε in time polynomial in n and $\log(1/\varepsilon)$.

Separable Programming. Separable programming depends upon the fact that the objective function R can be written as (i) a sum of separable terms, i.e. in every term only a single variable appears and (ii) each term is concave. Since each term in R is of the form $-a_i f_i^2 + b_i f_i$, both the above conditions are

satisfied. Consider the single term $R_i(f_i)$ as shown in Fig. 5 and an approximation of it by k piecewise linear segments. We call the approximate revenue function \hat{R}_i . We number the segments consecutively from left to right. Since the revenue function is concave, the slope of the segments are monotonically decreasing, i.e. $s_1 \geq s_2 \geq \dots \geq s_k$. Now we replace the single variable f_i and R_i with k new variables $\{g_\alpha\}$ as follows:

$$f_i = g_1 + g_2 + \dots + g_k \quad (20)$$

$$\hat{R}_i = s_1 g_1 + s_2 g_2 + \dots + s_k g_k \quad (21)$$

$$g_\alpha \leq u_\alpha \quad (22)$$

$$g_\alpha \geq 0 \quad (23)$$

$$\sum_{\alpha} u_\alpha = \frac{b_i}{a_i}. \quad (24)$$

Here we have replaced the variable f_i by k variables g_α each of which range between $[0, u_\alpha]$. As such, the above decomposition of f_i as a sum of g_α variables has too much freedom: for a given value of f_i , we can construct many decompositions each of which will give different values of \hat{R}_i . We need to impose an extra condition which will make sure that we end up with the correct \hat{R}_i . The condition is as follows: if $u_\beta > g_\beta > 0$ for any β , then $g_\alpha = u_\alpha$ for all $\alpha < \beta$ and $g_\alpha = 0$ for all $\alpha > \beta$. But, it turns out that *when we carry out the maximization, the solution always satisfies the above condition*. Interested readers can refer to [11] for additional details. As a result, the approximate LP that we have constructed above always leads to the correct optimum under maximization.

We will present the performance analysis and runtime complexity of both CAUP and CADP algorithms in Section 5. We conclude the present section with an algorithm for obtaining a feasible channel allocation to each user, given the outcome of our optimization problem, i.e. $\{f_i\}$.

4.5. Scheduling Spectrum Usages

Given spectrum allocations $\{f_i\}$, we need to schedule the actual usage patterns, i.e. the index of channels assigned to each buyer. We follow the *left of order* in the NLI constraints. We start from the leftmost node in the network and assign to it the initial portion of the spectrum. For every next node i , we examine the rightmost node lying to the left of i , referred to \mathcal{R}_i , and assign to i the portion of its allocated spectrum starting from where the

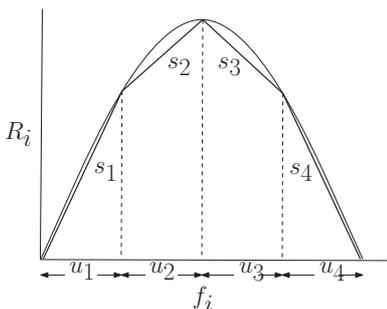


Fig. 5. Separable programming using piecewise linear segments to approximate a concave revenue function.

assignment of \mathcal{R}_i finishes. This schedule is *always feasible* because the constraint (16) – no node and its left neighbors can consume all the spectrum. This conclusion can be proved by induction, but in the interest of space its proof is omitted. We would like to note that this schedule in general assigns a continuous block of spectrum to each bidder, however, there are cases where a bidder may be allocated with two separate blocks of spectrum when the allocated spectrum falls on the boundary of the total spectrum range.

5. Analytical Performance

In this section, we provide analytical results on the performance and run-time complexity of both CAUP and CADP algorithms.

5.1. CAUP Performance and Complexity

Theorem 1 *CAUP solves the revenue maximization problem with concave piecewise linear demand curves and uniform clearing price, within an approximation factor of 3, i.e. $R_{CAUP} \geq \frac{1}{3}R_{OPT}$, in time $O(n \log n + n \log U)$. U represents the search range $U = b_n$.*

Proof: Suppose the set of optimal spectrum allocations is given by $\{f_i^*\}$ and the allocations found by our algorithm is $\{f_i\}$. The constraint in eqn. 12 ensures that the allocations $\{f_i^*/3\}$ are always feasible (proof in [6]). Hence

$$\sum_i R_i(f_i^*/3) \leq \sum_i R_i(f_i) \leq \sum_i R_i(f_i^*). \quad (25)$$

We also note $R_i(f_i^*) > 0 \Rightarrow R_i(f_i^*/3) > 0$, and

$$\frac{R_i(f_i^*)}{R_i(f_i^*/3)} = \frac{b_i f_i^* - a_i f_i^{*2}}{b_i f_i^*/3 - a_i f_i^{*2}/9} = 3 - \frac{6a_i f_i^{*2}}{3b_i f_i^* - a_i f_i^{*2}} \leq 3. \quad (26)$$

Thus

$$\frac{1}{3} \sum_i R_i(f_i^*) \leq \sum_i R_i(f_i) \leq \sum_i R_i(f_i^*). \quad (27)$$

Note here that if we use piecewise linear demand curves instead of linear demand curves, again we end up with the same guarantees. Any clearing price, which produces the maximum revenue (in the feasible region), lies uniquely on some segment of the piecewise linear demand curves, defined by a unique slope and y-intercept, and those can be substituted in for the b_i and a_i used above to get the same result.

Next we turn to the running time of our algorithm. Given a set $\{a_i\}$ and $\{b_i\}$, we can sort them in $O(n \log n)$ time. Then with some preprocessing which takes $O(n)$ time, we can evaluate $R(p)$ is $O(1)$ time for any given p . So if the size of the domain of the possible values of b_i is U , then the region of feasibility, i.e. p_f can be found in $O(n \log U)$ time by using binary search over p . Since the number of intervals is $O(n)$, the time required to find the maximum over all the intervals is $O(n)$. Thus the maximization problem defined above can be solved in $O(n \log n) + O(n \log U) + O(n) = O(n \log n + n \log U)$ time. All this analysis was for linear demand curves – for piecewise linear curves where each bid has a maximum of m segments, it is not difficult to see that the time required for this algorithm is $O(nm(\log U + \log(nm)))$. ■

Lemma 3 *When the conflict graph is a tree graph, CAUP produces the optimal solution to the revenue maximization problem.*

The proof can be found in [6].

5.2. CADP Performance and Complexity

Similarly, we show that the performance and complexity of CADP are also bounded.

Theorem 2 *CADP solves the revenue maximization problem with concave piecewise linear demand curves and discriminatory clearing price, within an approximation factor $3(1 + 1/n)$, in polynomial time (depends on time required to solve the linear program).*

The detailed proof is listed in the Appendix.

Lemma 4 *When the conflict graph is a tree graph, CADP produces the optimal solution to the revenue maximization problem under discriminatory pricing. The proof can be found in [6].*

6. Experimental results

In this section, we conduct experiments to investigate the performance of the proposed auction framework. We consider the scenario described by Figure 1, where wireless service providers deploy their access points to serve their associated users (each access point is a buyer). To examine the impact of network topology, we use both randomly generated topologies and measured AP deployment traces to validate our results. We consider the following network topologies:

- *Random network.* We place users randomly in a unit square (normalized) area.
- *Clustered network.* We simulate a hotspot scenario by deploying a set of users densely in a small area of the random network. We use this topology to examine the impact of conflict degree on the rule performance.
- *Planned network.* We extract a set of actual AP deployments using data traces collected by Placelab. (<http://www.placelab.org/>).

We include the results using linear demand curves and maintain that piecewise linear curves lead to similar conclusions. We use the fixed power model and assume that every buyer wants to support users within a fixed radius (0.05 in our simulations). To produce the conflict graph, we use a simple distance-based interference model - any two access points conflict with each other if they are within 0.1 (twice the radius) distance of each other. While this assumption is used to produce the conflict graph, it does not limit the application of our approach to other general interference conditions. The maximum spectrum available at any location in the network is normalized to 1. All results shown are averaged over 5 random seeds. All the simulations are run in C++ on a 3.0 GHz processor with 1 GB of RAM.

We consider three types of bidding curves:

behavior	spec. vs. unit price	unit price vs. spec.
normal	$f(p) = -p + 1$	$p(f) = -f + 1$
conservative	$f(p) = -2p + 1$	$p(f) = 1/2(-f + 1)$
aggressive	$f(p) = -p/2 + 1$	$p(f) = 2(-f + 1)$

Note that the maximum per unit prices are 1, 1/2 and 2 for normal, conservative, and aggressive bidders respectively. Unless mentioned, all buyers are normal bidders for all experiments.

We use the following performance metrics:

- Revenue: $R = \sum f_i(p_i)p_i$.
- Spectrum utilization: $U = \sum f_i(p_i)$.
- Each buyer’s price p_i and channel assignment $f_i(p_i)$.
- Complexity in terms of algorithm execution time.

Using our experiments, we examine the performance of two pricing models, the performance of the proposed approximation algorithms compared to the optimal solutions, the impact of bidding behavior and node density, and finally the algorithm execution time.

6.1. Uniform vs. Discriminatory Pricing

We start by examining the performance of the proposed auction-clearing algorithms under two pricing models. We vary the network size from 0 to 1300, increasing the average conflict degree from 0 to 10. Results in Figure 6 show that both revenue R and spectrum utilization U grow with the network size, but the growth rate decreases with the network size since spectrum usages saturate at high node density.

At small network sizes (< 20), the difference between the revenue produced by the uniform and discriminatory pricing is small. As network size increases beyond 40, the discriminatory pricing model generates nearly twice the revenue (and spectrum utilization) as compared to uniform pricing. Under the uniform pricing model, the market-clearing price is determined by the maximum level of conflict in the network, *i.e.* the maximum node degree in the conflict graph. As the network size increases, the market-clearing price moves towards 1. Using the discriminatory pricing model, the seller charges buy-

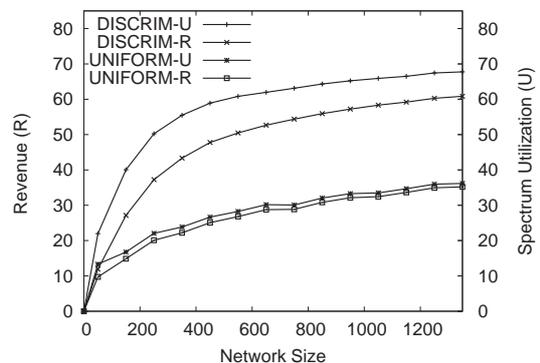
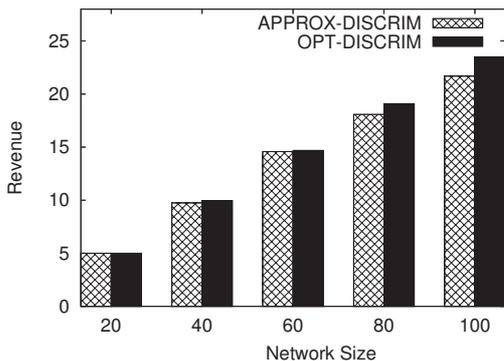


Fig. 6. Revenue and spectrum utilization for varying network sizes for both pricing models.

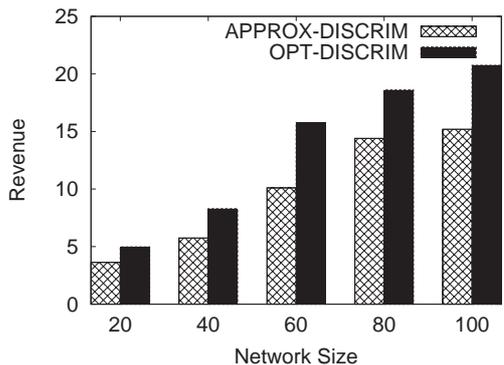
ers based on their local level of conflict, and hence results in more efficient use of spectrum.

6.2. Optimal vs. Approximation Algorithms

Next, using the discriminatory pricing model, we compare the performance of the approximation algorithm to the optimal solution. We use the randomized algorithm proposed in [15] to generate maximal independent sets for the linear program. We run the randomized algorithm for 200000 iterations for network sizes of 20 – 100 on a random deployment. Figure 7(a) shows this results for random network, it can be seen that amount of revenue generated by the approximation algorithm is within 10% of the optimal, for all network sizes. However, the amount of time taken for generating the random sets and solving the linear program for a network of 100 nodes, is 4 hours, which is more than 20000 times the time taken by the proposed algorithm.



(a) Random Network



(b) Planned Network

Fig. 7. Comparison of the optimal solution and the approximation algorithm under the discriminatory pricing model.

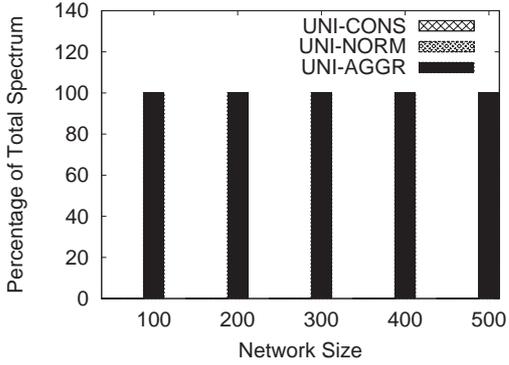
Next, we examine the revenue generated for planned networks. Figure 7(b) compares the perfor-

mance of the optimal and the proposed approximation algorithm. We see that the revenue produced by the optimal algorithm in the planned deployment is less than the optimal revenue in the random network (although they have the same number of users per area). In the planned network, nodes are placed in clusters and hence experience much higher level of interference than those in the random network. Next, we observe that the approximation algorithm produces roughly 70% of the optimal revenue, which is worse than that of the random network. This is because the node conflict degree is much higher and hence the loss due to Node-L-Interference approximation becomes more visible. We will further examine the impact of network topology in Section 6.4.

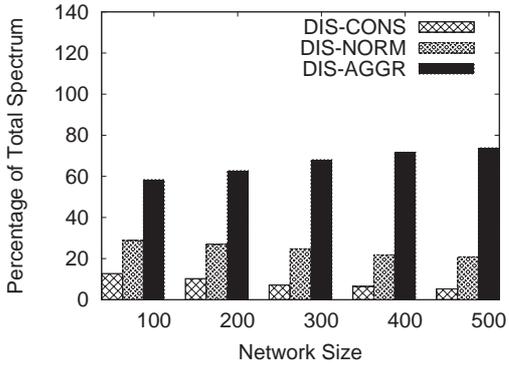
6.3. Impact of Bidding Behaviors

We now examine the impact of bidding behaviors on prices and allocations for both pricing models, using random topologies. In this experiment, buyers randomly choose their bidding curves as conservative, normal or aggressive, with equal probability. Figure 8 shows the percentage of spectrum allocated to different bidding categories. Under the uniform pricing model, aggressive buyers take over all the spectrum. Since the market-clearing price is high $p > 1$, conservative and normal buyers are completely cut off. While aggressive buyers also receive a large portion of the spectrum under the discriminatory pricing, they do not manage to exclude other buyers completely. This is because at small network sizes (or with low node density), there are not enough aggressive buyers to consume all the spectrum. As the network size increases, the level of contention increases and so does the price. As the price increases, conservative and normal users are slowly cut off from the auction while aggressive users start to dominate.

Figure 9 compares the total revenue generated by different bidders under both pricing models. Using the uniform pricing model, we only show the revenue from aggressive bidders (UNI-AGGR) since they obtain all the spectrum. We observe that aggressive bidders under the discriminatory pricing model produce higher revenue than those under uniform pricing (reason same as in Section 6.1). Normal and conservative bidders contribute to a small portion of the revenue.



(a) Uniform Pricing



(b) Discriminatory Pricing

Fig. 8. Percentage of spectrum allocated to different bidder categories for uniform and discriminatory pricing models. UNI-CONS/NORM/AGGR: conservative, normal and aggressive bidders in uniform pricing model; DIS-CONS/NORM/AGGR: conservative, normal and aggressive bidders in discriminatory pricing model.

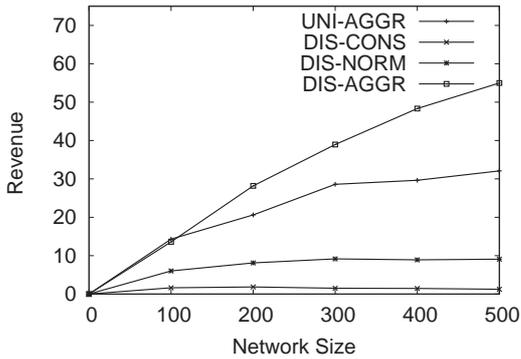


Fig. 9. Revenue generated by different bidders for uniform and discriminatory pricing algorithms.

6.4. Impact of Network Topology

In practice, wireless service providers might not position access points randomly over the area. They

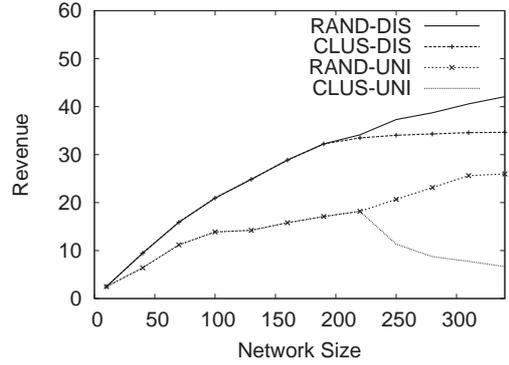


Fig. 10. Effect of clustering on the revenue. RAND and CLUS represent random deployments and clustered deployments.

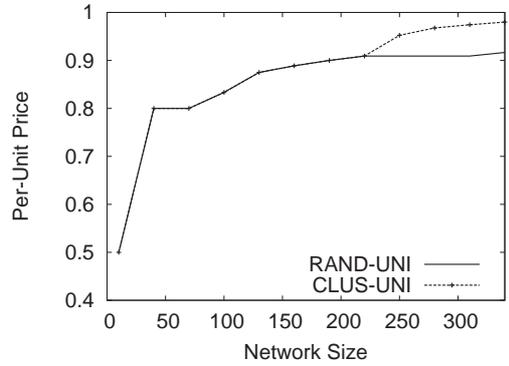


Fig. 11. Effect of clustering on the per-unit price (assuming the uniform pricing). RAND and CLUS represent random deployments and clustered deployments.

deploy many access points in areas with dense user populations, known as hotspots. We start from a simulated hotspot scenario by deploying a clustered network, illustrated by the leftmost figure in Figure 12. We initially deploy 200 nodes randomly on a unit square area, and then deploy the next k ($0 \leq k \leq 150$) nodes in a clustered region.

System Performance To examine the impact of clustering, we first compare the revenue of random and clustered deployments assuming normal bidders. Figure 10 shows the revenue under both pricing models for various network sizes. For network sizes of 200 or less, random and clustered deployments produce the exactly same topology and their revenue curves overlap.

An interesting observation is that the uniform and discriminatory pricing models respond very differently to clustering. Under the discriminatory pricing model, the revenue converges very fast to a constant value, corresponding to a fully utilization of spectrum inside the cluster. In contrast, under the

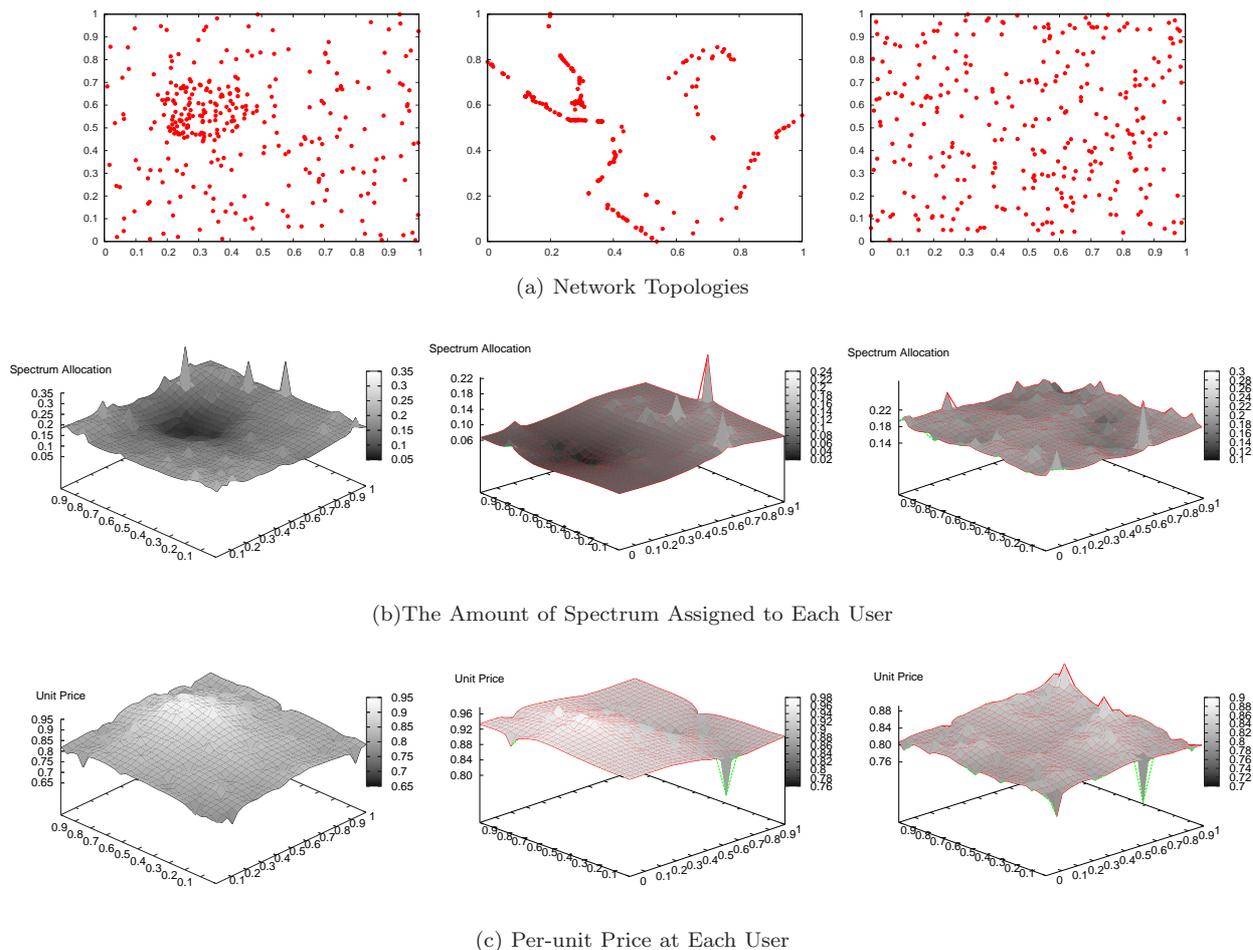


Fig. 12. The impact of network topology on per-user spectrum allocation and price, for clustered network (left column), planned network (middle column) and random network (right column), assuming discriminatory pricing and normal bidding.

uniform pricing model, the revenue drops with the clustering. This is because that the market-clearing price now is being governed by the maximum level of contention, *i.e.* the node density in the cluster. As k increases, the market-clearing price quickly rises to 0.99 ($k = 140$, network size = 340) (see Figure 11). Therefore, spectrum allocations (and revenue) at non-clustered regions drop drastically, and the degradation outweighs the improvement inside the cluster. Note that under random deployments, the market-clearing price under uniform pricing also increases with the node density. However, the increase is not drastic and is fully compensated for by the increase in network size.

User Performance To further examine the impact of clustering on buyer performance, in Figure 12 we plot the allocation and price for each buyer under the discriminatory pricing model for the clustered,

planned and random networks. In the planned network, nodes are even more clustered. We see that buyers in the cluster have significantly lower allocations and higher prices both in the planned and clustered networks. In the random network though, the price and allocations are nearly uniform. This shows that, to maximize revenue and spectrum utilization, pricing should depend on the conflict condition – price should be high at places with high demand and scarce resources. Note that there are several allocation spikes, which correspond to the nodes at sparse area with minimal conflicting neighbors – price should be low at places with low demand.

Impact of Bidding Behaviors The above observation triggers an interesting question: *how can a node in a clustered area obtain more spectrum?* In order to answer this question, we investigate the impact of bidding behavior on individual buyer's per-

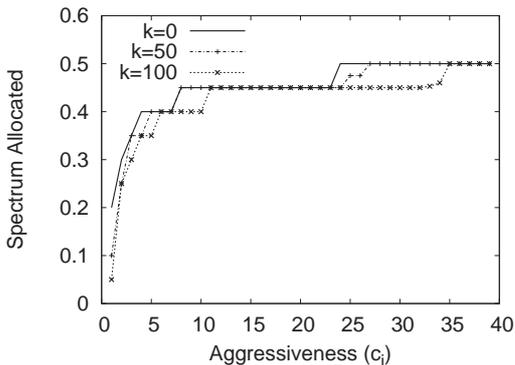


Fig. 13. The impact of bidding aggressiveness on individual buyers.

formance using the discriminatory pricing model.

We monitor a particular buyer's spectrum allocation while varying his bidding behavior. We consider the same clustering scenario, and pick a particular buyer i from the clustering area when $k = 0$. Next, we randomly add k nodes to the cluster. As k increases, the level of conflicts/competition around i increases. We model i 's bidding behavior using $f_i(p_i) = -p_i/c_i + 1$, or equivalently, $p_i(f_i) = -c_i f_i + c_i$ where c_i represents the bidding aggressiveness. The rest of the network nodes are normal bidders $c = 1$. Figure 13 shows i 's allocation f_i for various aggressiveness levels, for $k = 0$ (no clustering), $k = 50$ (mild clustering) and $k = 100$ (heavy clustering). We see that bidding aggressively ($c_i > 1$) as compared to neighboring nodes does bring in extra allocations (at higher prices). However, the benefit drops with the level of aggressiveness - the allocation curves flatten out. In order to have monopoly of the spectrum, the buyer has to pay significantly more per unit (depending on the density of the cluster) to obtain that last fraction of the spectrum.

6.5. Algorithm Complexity

In Figure 14 we compare the algorithm run times for varying network sizes. We see that the approximation algorithm under the uniform pricing model runs extremely fast (0.05 seconds for 3500 nodes), while the discriminatory pricing approximation algorithm requires less than 80 seconds for up to 3500 nodes. Earlier we had mentioned that the optimal solution runs 20000 slower (for a much smaller network size 20 – 100).

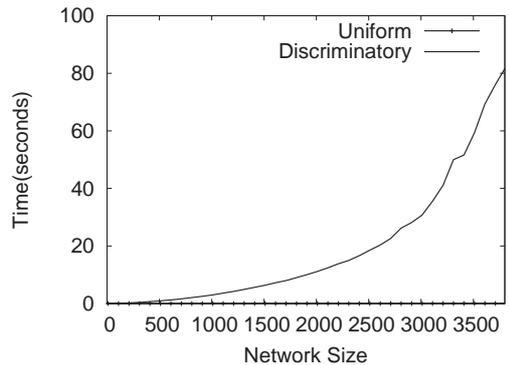


Fig. 14. Run-time comparison of the approximation algorithms.

7. Practical Considerations

In this section, we discuss the practical issues when implementing the proposed auction system.

7.1. Identifying interference constraints

The proposed auction system requires information on the interference constraints among buyers. There are multiple mechanisms to obtain this information. Using the scenario of access points based buyers, we list three complementary mechanisms.

- (i) The auctioneer (seller) perform network interference measurements to collect interference constraints. A similar mechanism is used in cellular networks to examine interference conditions among base stations.
- (ii) Individual access points scan radio signals to find interfering access points and report their findings to the auctioneer.
- (iii) Clients associated with access points sense radio signals and provide feedback on findings of interfering access points [18]. This mechanism has been shown to help refine the interference map.

To minimize the overhead in building conflict graphs, auctioneers can collect conflict information on all the candidate access points. In each round, the auctioneer constructs a conflict graph on the current buyers.

7.2. Decentralized auction systems

CAUP and CADP require a centralized server, which in practice might not always be available. In such a case, buyers send bids to local service points

who coordinate among themselves to derive allocations and prices. Decentralized systems have the advantage of allowing simple and scalable deployment and providing resilience against point failures.

To build a decentralized auction system, we apply the same bidding language and pricing models, and design coordination [7] based approximation algorithms. The basic concept is to let service points coordinate to apply local adjustments of allocations and prices to their associated buyers, and perform them recursively to improve the total revenue. The complexity of this approach depends on the algorithm complexity and the cost/delay of communications between local service points. We plan to study this system in a future paper.

7.3. Iterative bidding and heterogeneous channels

In iterative auctions, buyers submit bids in multiple rounds, and adjust bids based on market feedbacks. Auctioneers use clearing algorithms to derive prices and allocations and provide feedbacks. The challenge lies in simulating feedback and adjusting the bids accordingly. Also, in case of heterogeneous channels with different propagation properties and power limitations, the key issue is to define a standard price-quantity relationship. “Good” spectrum bands should cost more.

Both issues are important for practical spectrum auctions, and can be addressed by combining computational and non-computational (social behavior based) approaches. These are interesting problems in themselves, but we limit ourselves to homogeneous channels and single round bidding to investigate the absolute performance of the auction algorithms.

8. Conclusion and Future Work

We propose a spectrum auction framework to provide fast and efficient allocations of spectrum to wireless users. We propose a compact and expressive bidding language using piecewise linear price-quantity curves, two pricing models to address revenue and fairness, and low-complexity market-clearing algorithms to derive prices and allocations in real-time. We perform extensive experiments to verify the performance of the proposed system, and to explore the impact of bidding behaviors, pricing models and node clustering. We conclude that to maximize revenue and spectrum utilization, pricing

must be determined based on local demand and availability of resources. We summarize some practical issues and open problems in Section 7. We are currently working on extending our framework to address these issues.

Acknowledgement

The research of Sorabh Gandhi, Chiranjeev Buragohain and Subhash Suri was supported in part by National Science Foundation grants CNS-0626954 and CCF-0514738.

References

- [1] ALICHERY, M., BHATIA, R., AND LI, L. Joint channel assignment and routing for throughput optimization in multiradio wireless mesh networks. In *Proc. of Mobicom'05* (2005).
- [2] BINMORE, K., AND SWIERZBINSKI, J. Treasury auctions: Uniform or discriminatory? *Review of Economic Design* 5, 4 (2000), 387–410.
- [3] BORENSTEIN, S. The trouble with electricity markets: Understanding californias restructuring disaster. *Journal of Economic Perspectives* 16, 1 (2002).
- [4] BOURJADE, S. Strategic price discounting and rationing in uniform price auctions. *Journales Doctorales de l'ADRES* (2006).
- [5] BUDDHIKOT, M. M., AND RYAN, K. Spectrum management in coordinated dynamic spectrum access based cellular networks. In *Proc. of DySpan' 05* (November 2005).
- [6] BURAGOHAIN, C., SURI, S., TOTH, C., AND ZHOU, Y. Improved throughput bounds for interference-aware routing in wireless networks. In *UCSB Technical Report 2006-13* (2006).
- [7] CAO, L., AND ZHENG, H. Spectrum allocation in ad hoc networks via local bargaining. In *Proc. of SECON* (September 2005).
- [8] DE VRIES, S., AND VOHRA, R. V. Combinatorial auctions: A survey. *INFORMS Journal on Computing*, 3 (2003), 284–309.
- [9] EBAY. Multiple item auction <http://pages.ebay.com/help/buy/buyer-multiple.html>.
- [10] EBAY. <http://www.ebay.com>.
- [11] HILLIER, F., AND LIEBERMAN, G. *Introduction to Operations Research*. 2004.
- [12] HUANG, J., BERRY, R., AND HONIG, M. Auction mechanisms for distributed spectrum sharing. In *Proc. of 42nd Allerton Conference* (September 2004).
- [13] HUDSON, R. Analysis of uniform and discriminatory price auctions in restructured electricity markets. <http://certs.lbl.gov/pdf/ornl-pricing.pdf>.
- [14] ILERI, O., SAMARDZIJA, D., SIZER, T., AND MANDAYAM, N. B. Demand responsive pricing and competitive spectrum allocation via a spectrum server. In *Proc. of DySpan' 05* (November 2005).

- [15] JAIN, K., PADHYE, J., PADMANABHAN, V., AND QIU, L. Impact of interference on multi-hop wireless network performance. In *Proc. of Mobicom'03* (2003).
- [16] KRISHNA, V. *Auction Theory*. Academic Press, 2002.
- [17] KUMAR, V. S. A., MARATHE, M. V., PARTHASARATHY, S., AND SRINIVASAN, A. Algorithmic aspects of capacity in wireless networks. In *Proc. of SIGMETRICS Conference* (2005).
- [18] MISHRA, A., BRIK, V., BANERJEE, S., SRINIVASAN, A., AND ARBAUGH, W. A client-driven approach for channel management in wireless LANs. In *Proc. of IEEE Infocom* (2006).
- [19] P. MALVEY, C. ARCHIBALD, S. F. Uniform price auctions : Evaluation of the treasury experience. <http://www.treasury.gov/offices/domestic-finance/debt-management/auctions-study/upas2.pdf>.
- [20] RYAN, K., ARAVANTINOS, E., AND BUDDHIKOT, M. M. A new pricing model for next generation spectrum access. In *Proc. of TAPAS* (August 2006).
- [21] SANDHOLM, T. Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence* 135, 1-2 (2002), 1–54.
- [22] SANDHOLM, T., AND SURI, S. Market clearability. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)* (2001).

9. Appendix: Proof of Theorem 2

In this section we show that the separable programming algorithm allows us to achieve a constant factor approximation to the revenue optimization problem with non-uniform pricing. We first prove our claim only for simple linear demand curves – the extension to piecewise linear demand curves is straightforward.

First we make the observation that the separable programming algorithm introduces a two-step approximation to the optimal revenue determination problem. The first approximation is due to the fact that in eqn. 12 we have introduced NODE-L-INTERFERENCE constraints. The next approximation we make is to replace the quadratic revenue function by a piecewise linear revenue as shown in Fig. 5. By an argument identical to the one that we made in the proof of Theorem 1, the NODE-L-INTERFERENCE constraints introduces an approximation factor of 3. In the rest of this section, we focus on bounding the error introduced by the piecewise linear approximation.

Before we prove the desired theorem, we shall prove a simple lemma. Recall that the range of values of f_i is $[0, b_i/a_i] \cap [0, 1]$. We divide this range to k equal segments, i.e. we set $u_\alpha = b_i/(ka_i)$. We now state a lemma which bounds relative error in R_i .

Lemma 5 *Given a function $R(x) = -ax^2 + bx$ and a piecewise linear approximation $\hat{R}(x)$ to it by segments each of which span a range δ along the x -axis, the maximum relative error in the approximation is $\delta a/b$.*

Proof: Consider a function $R(x) = -ax^2 + bx$ and a piecewise linear approximation to it $\hat{R}(x)$ with piecewise linear segments, each segment spanning a range $\delta \ll b/a$ along the x -axis. In other words, the segments span ranges $[0, \delta], [\delta, 2\delta], \dots$. Consider the range $[x, x + \delta]$ and a point inside the range $x + \lambda\delta, 0 \leq \lambda \leq 1$. Then the relative error at this point is

$$e(x, \lambda) = \frac{R(x + \lambda\delta) - \hat{R}(x + \lambda\delta)}{R(x + \lambda\delta)}. \quad (28)$$

We wish to find the maximum of this error. Since R is a quadratic function, we can expand it using Taylor series as follows:

$$R(x + \lambda\delta) = R(x) + \lambda\delta R'(x) + \lambda^2\delta^2 R''(x)/2 \quad (29)$$

$$\hat{R}(x + \lambda\delta) = R(x) + \lambda(R(x + \delta) - R(x)) \quad (30)$$

$$= R(x) + \lambda(\delta R'(x) + \delta^2 R''(x)/2). \quad (31)$$

This immediately gives us

$$e(x, \lambda) = \frac{\lambda\delta^2(\lambda - 1)R''/2}{R + \lambda\delta R' + \lambda^2\delta^2 R''/2}. \quad (32)$$

The numerator is independent of x , to find the maximum of $e(x, \lambda)$, we have to minimize the denominator. The denominator is $R(x + \lambda\delta)$, which is nothing but the all the revenue values varying with x and λ . We know that the revenue function for every node is quadratic and that its value is minimum around $x=0$ and $x=b/a$. Thus the two intervals in which we need to check for minimum value of $R(x + \lambda\delta)$ are $[0, \delta]$ and $[b/a - \delta, b/a]$ (it is easy to verify that in any other interval, the revenue value is more than the revenue values in these two intervals).

We consider the first of these two intervals, by replacing $x=0$ in eqn. 32 we get

$$e(0, \lambda) = \frac{\lambda\delta^2 a(1 - \lambda)}{\lambda\delta b - \lambda^2\delta^2 a} = \frac{1 - \lambda}{b/a\delta - \lambda}. \quad (33)$$

In the range $\lambda \in [0, 1]$, with $\delta \ll b/a$, the function $e(0, \lambda)$ is monotonically decreasing. Hence the maximum value of the error $\delta a/b$ and occurs as $\lambda \rightarrow 0$.

We now look at the value of error in eqn. 32 with x in the interval $[b/a - \delta, b/a]$, Simplifying the expression using $x=b/a - \delta$, we get ,

$$e(b/a - \delta, \lambda) = \frac{\lambda\delta^2 a(1 - \lambda)}{(1 - \lambda)(b\delta - a\delta^2(1 - \lambda))} \quad (34)$$

Simplifying further, we get

$$e(b/a - \delta, \lambda) = \frac{\lambda}{b/a\delta - (1 - \lambda)} \quad (35)$$

Again the maximum value of this error is $\delta a/b$ and occurs as $\lambda \rightarrow 1$. ■

Using **Lemma 5**, we obtain

$$\max_{f_i} \left(\frac{R_i(f_i) - \hat{R}_i(f_i)}{R_i(f_i)} \right) = \frac{b_i}{ka_i} \frac{a_i}{b_i} = \frac{1}{k}. \quad (36)$$

Thus

$$\hat{R} = \sum_i \hat{R}_i \geq \sum_i R_i \left(1 - \frac{1}{k} \right) = R \left(1 - \frac{1}{k} \right). \quad (37)$$

Combining this with the fact that NODE-L-INTERFERENCE constraints induces a 3-approximation, we arrive at the result we seek by setting $k = n$:

$$\frac{R_{\text{OPT}}}{R_{\text{LP}}} \leq 3 \left(1 + \frac{1}{n} \right). \quad (38)$$

This result was derived for linear curves. For piecewise linear demand curves, the separable programming approach will still work as long as the individual revenue function is concave. For an ordinary linear demand curve, the revenue function is an inverted parabola and hence is concave. For piecewise linear demand curves, let us consider a m -segment piecewise linear demand curve specified by the line segments $-a_1x + b_1, a_2x + b_2, a_3x + b_3, \dots, a_mx + b_m$. Here we have ordered the line segments starting from left to right. The concavity condition for the corresponding revenue is given by the following lemma.

Lemma 6 *Given a piecewise linear demand curve with monotonically decreasing slope, the revenue curve corresponding to the demand curve is always concave.*

Proof: Consider two adjacent segments $-a_kx + b_k$ and $-a_{k+1}x + b_{k+1}$ on the piecewise linear demand curve. The revenue curve corresponding to each of these segments is an inverted parabola and hence concave. The only point at which we need to establish concavity is the intersection point of these two segments, i.e. the point at which the demand curve switches from slope $-a_k$ to $-a_{k+1}$. The intersection point is given by

$$x = \frac{b_{k+1} - b_k}{a_{k+1} - a_k}. \quad (39)$$

Since $x > 0$ by construction and $a_{k+1} > a_k$ (monotonically decreasing slope), we arrive at the condition that $b_{k+1} > b_k$. If the revenue curves for

each of these segments are $R_k(x)$ and $R_{k+1}(x)$, then the concavity condition is given by

$$R'_{k+1}(x) - R'_k(x) \leq 0. \quad (40)$$

To show this, we evaluate the LHS of the above equation using the fact that

$$R'_k(x) = -2a_kx + b_k. \quad (41)$$

Then

$$\begin{aligned} R'_{k+1}(x) - R'_k(x) &= -2x(a_{k+1} - a_k) + b_{k+1} - b_k \quad (42) \\ &= -2(b_{k+1} - b_k) + (b_{k+1} - b_k) \quad (43) \\ &= -(b_{k+1} - b_k) < 0. \quad (44) \end{aligned}$$

The second line above follows from eqn. 39. ■

With the concavity condition established, we next turn to the error bound. With a maximum of m segments in each demand curve, Lemma. 5 will hold for piecewise linear demand curves by using $O(nm)$ approximation segments for each demand curve. Thus **Theorem 2** holds for linear demand curves as well as piecewise linear demand curves with monotonically decreasing slope.