Self-enforcing Strategic Demand Reduction

Paul S. A. Reitsma[1], **Peter Stone**[2], **János A. Csirik**[3], **and Michael L. Littman**[4]

[1] Computer Science Department, Brown University, Box 1910, Providence, RI 02912
psar@cs.brown.edu — www.cs.brown.edu/~psar

[2] Department of Computer Sciences, University of Texas at Austin, Austin, TX 78712
pstone@cs.utexas.edu — www.cs.utexas.edu/~pstone

[3] AT&T Labs — Research, 180 Park Ave., Florham Park, NJ 07932
janos@research.att.com — www.research.att.com/~janos

[4] Department of Computer Science, Rutgers University, Piscataway, NJ 08854
mlittman@cs.rutgers.edu — www.cs.rutgers.edu/~mlittman

Abstract. Auctions are an area of great academic and commercial interest, from tiny auctions for toys on eBay to multi-billion-dollar auctions held by governments for resources or contracts. Although there has been significant research on auction theory, especially from the perspective of auction mechanisms, studies of autonomous bidding agents and their interactions are relatively few and recent. This paper examines several autonomous agent bidding strategies in the context of *FAucS*, a faithful simulation of a complex FCC spectrum auction. We introduce *punishing randomized strategic demand reduction* (PRSDR), a novel bidding strategy by which bidders can partition available goods in a mutually beneficial way without explicit inter-agent communication. When all use PRSDR, bidders obtain significantly better results than when using a reasonable baseline approach. The strategy automatically detects and punishes non-cooperating bidders to achieve robustness in the face of agent defection, and performs well under alternative conditions. The PRSDR strategy is fully implemented and we present detailed empirical results.

1 Introduction

Some of the largest auctions held involve hundreds of goods in hundreds of categories, scores of bidders, complicated rules, and everything being auctioned simultaneously over a hundred or more rounds. In these auctions, it is very difficult for humans to grasp all of the nuances regarding the effects of their strategic decisions. Computer aid could ease the burden of efficiently competing in these auctions, especially if the user need only input his or her values for the goods in the auction into a parameterized strategy.

Simple autonomous agents have started to appear as bidders in some auctions. However, they tend to be straightforward incremental bidders that raise the price of a single good up to a user's stated maximum (e.g. on eBay). Bidding agents that support dependent values among multiple interacting goods have been deployed in some experimental scenarios [5, 10, 8]. Here, we present agent bidding strategies in a large-scale and realistic auction scenario, namely the FCC Spectrum Auction Simulator, or *FAucS* [1].

Here, we build on our previous work in *FAucS*, in which we created *Knapsack* agents that optimized the set of goods they bid on given a budget constraint, but not taking into account the needs and strategies of other agents [1]. In this previous work we also created alternative agent strategies that outperformed the Knapsack agents, but that relied on the unrealistic assumption that agents knew each other's valuations of the goods with complete certainty.

In this paper, we present *punishing randomized strategic demand reduction* (PRSDR), a strategy by which cooperative agents can significantly outperform the Knapsack agents despite having highly uncertain knowledge regarding each others' goals and without any explicit inter-agent communication. The strategy is self-enforcing in that agents cannot benefit by defecting back to the Knapsack strategy.

PRSDR serves as an example of a general strategy for bidding in simultaneous multiple-round (SMR) auctions in which:

– there is room to bid below the expected sell prices; and in which
– there are few value interdependencies among goods.

The basic idea of PRSDR is that bidders bid for randomized subsets of their true desired goods while retaining the ability to effectively punish bidders who defect from this mutually beneficial strategy. The punishment takes the form of driving prices up in the markets of interest to a defecting bidder.

The remainder of the paper is organized as follows. An overview of *FAucS* and the general setup appears in Sections 2 and 3. Details of the PRSDR algorithm follow in Sections 4 and 5. Empirical results demonstrating its utility and robustness are presented in Section 6 along with a suggestion on how to improve the efficiency of FCC Spectrum auctions by a change of rules designed to inhibit communication-free cooperation. Section 7 relates our results to the game-theoretical construct of iterated bimatrix games and Section 8 concludes.

2 FCC Spectrum Auction Simulator

Our prototype implementation of PRSDR is tailored to the *FAucS* domain [1], a detailed and realistic simulator of the FCC Spectrum auctions. The goods available in the FCC spectrum auctions are a set of *licenses*, or blocks of spectrum; each in a *market*, or region of the United States. In this paper we focus on FCC Auction 35, in which licenses were 10 or 15 megahertz in size, and each of the 195 markets had between 1 and 4 licenses available. A total of 422 licenses and more than 80 bidders were involved.

To a first approximation, the rules of the auction are straightforward (official rules are presented in FCC document DA 00–2038[1]). All of the FCC spectrum auctions, including Auction 35, use an SMR system in which all goods are available at the same time, and bidding occurs in discrete rounds[2]. After each round, each bidder's bids are announced publicly. The *provisionally winning bids* and corresponding *provisional winners* are also announced: these are the highest bid received up to that point on each

[1] Available at http://wireless.fcc.gov/auctions/35/releases.html

[2] There are plans to move to a combinatorial auction setting in the future, but so far they have not been realized due in part to the complexity of the winner determination problem in combinatorial auctions [7].

license and the party that placed the bid (in case of a tie, the first bid submitted to the FCC's system wins). The auction ends immediately after the first round with no new activity. Each license is then sold to its provisional winner for a price equal to the provisionally winning bid.

As is customary in auctions, the FCC had anti-collusion rules forbidding bidders from explicitly exchanging information before and/or during the auction.

While this underlying mechanism is simple, there have been many subtle modifications and complex rules added over time. The two that are most important for the purposes of this paper are the rules governing allowable bids and eligibility constraints.

Allowable bids. Bids on a license that has received no bids in previous rounds must be of a predetermined minimum bid price. After a license has received bids, further bids can only increase the value of the current provisionally winning bid by 1 to 9 *bid increments*. A bid increment is a value between 10% and 20%, calculated by the FCC and increasing with bidding activity.

Eligibility Constraints. Each license is worth a certain number of *bidding units (BUs)*, correlated with the population of the market. Each round, a bidder has a specific *eligibility*, and may not bid on more BUs than that. A bidder must maintain bidding activity in order to maintain eligibility to bid in later rounds.[3]

FAucS models these rules and all others relevant to Auction 35 in their entirety.[4] It uses a client-server architecture with the server and the bidding agents (clients) all written in Perl and using TCP sockets to communicate with each other.

Typical auctions last between 100 and 150 rounds; Auction 35 lasted 101 rounds.

3 Simulation Overview

While the original FCC auction 35 had more than 80 bidders, only a few—about five—were of significant individual importance in terms of the licenses won. We designate a set of five agents as *strategic bidders* to emulate the presence of five national companies participating in Auction 35.[5] By virtue of their size, particularly estimated budget size, it is immediately apparent to all agents which bidders are strategic bidders.

[3] Despite the high degree of substitutability between goods in the goal representation bidders use, this activity rule is still very important. It forces bidders to show their hand earlier, rather than sniping at the end of the auction, as is a common tactic in other auctions [8, 6]. This requirement that bidders place bids significantly before prices reach their final levels also requires bidders to either risk winning a license they would rather not win by bidding without revealing their true desires until the last moment, or to bid according to their desires, thereby divulging enough information to allow all facets of PRSDR to function.

[4] The exception is that, as a simulation, *FAucS* breaks ties randomly. Analysis of the actual FCC auction data reveals that bidders tended to place their serious bids as early as possible, thus reducing tie-breaking essentially to random (depending on the particular connection bidders had to the FCC, the speed at which they clicked their mice, etc). For example, in the last 80% of Auction 35, fully 45% of the bids were placed in the first minute of the 40-minute bidding round. Thus, this is a reasonable simplification.

[5] AT&T Wireless, Cingular Wireless, Sprint PCS, Verizon Wireless, and Voicestream. These national companies were interested in winning most of the licenses and had the most money at stake. Thus, they had the most to gain by clever, strategic bidding.

The other 75 bidders in the original auction were budget-constrained regional companies with interests only in specific markets. The primary effect of these smaller bidders was to create competition in every market and drive up prices: they didn't win very much in the end. Accordingly, we have modeled the aggregate role of these bidders using five *secondary bidders* which are straightforward bidders with no budget constraints. These bidders each desire a license from each market but are willing to pay only about 75% as much as the strategic bidders for any one license. These secondary bidders ensure competition in every market until license prices are beyond the levels they are willing to pay, effectively setting the price floor for licenses at about 75% of strategic bidder valuations. This bidding approach served to raise prices to realistic levels without requiring us to model all 75 regional bidders explicitly.

3.1 Agent Utility

For the purposes of this paper, the utility function for all agents is *profit*, defined as the value to the agent of the set of licenses it won minus its expenditures. We assume no inter-market dependencies in market valuations. This hypothesis is a reasonable starting point for our explorations and can be partially justified by the existence of an (inefficient) aftermarket for licenses. There is also some evidence that human bidders in these auctions ignore intermarket dependencies.

Each agent has significantly different goals drawn from a realistic model based on analysis of real auction data, information from real bidders, and a Merrill Lynch analysis of the estimated theoretical values of particular markets [1]. These goals consist of *priorities* and *valuations* for each market.

Each agent's priority $p \in \{0, 1, 2\}$ for a given market is the number of licenses the agent would like to acquire from that market. A market's value to this agent increases with the size of the licenses in megahertz and the population of the market (*mhzpop*). This *Market Value* (*MV*) is the value to the agent (the maximum amount the agent is willing to pay) for a license in a market where it wants to acquire one more license (whether or not it has any already). If an agent wants two more licenses in a market, it is willing to pay an *Enhanced Market Value* or *EMV* of approximately 5% more for the first of those licenses, to try to ensure it gets at least one. Otherwise, if an agent wants no licenses in a market, any licenses in that market have no value to the agent. Profit earned is the difference between the value of a license to the agent and the price paid for that license, and the only effect of priority is to control these values.

For example, if the New York market had a population of 30,000,000, 10-mhz licenses, and agent 3 valued the market at \$5 million per mhzpop and had a priority of 2 in the market, it would be willing to pay \$1.575 billion for the first license, \$1.5 billion for the second, and \$0 for the third. Purchasing one license for \$1.2 billion would generate a profit of \$375 million.

These priorities introduce a simple type of inter-good dependency that model the real bidders' apparent interests, as expressed by one of the bidding teams from the original (real) auction. It is interesting to note that, while this value characterization may not be optimal since it may simplify away information about inter-good dependencies, it is nevertheless approximately what the real human bidders used. Our conversations with them suggested that the most probable explanation for using this simplified representation is that the auction is so tremendously complex that even highly skilled human bidders were unable to reason efficiently using the full complexity of all dependencies in the situation, and had to simplify the representation they used in order to make the problem tractable. One of our hopes in carrying out auction agent research is to relieve the human bidders of some of this complexity burden, allowing them to use richer goal representations. Potentially, this would improve the efficiency of the allocation of goods, leading to a consequent increase in value for the bidders, the auctioneer, and society at large through the more-efficient utilisation of goods.

3.2 Uncertain Knowledge

In our preliminary experiments [1], agents had full knowledge of each others' utility functions; however, in reality, agents have only rough estimates of each others' utilities, arising from market research on that company's competitors. In this paper, we only consider strategies that are robust to uncertain knowledge.

We obtained an agent's estimate of the budget and MVs of other agents by taking the actual budget and MVs of those agents and randomly perturbing the values by up to 20% in either direction, separately for each value. Every priority for every other agent had approximately a 25% chance of being guessed incorrectly. Note that under this pessimistic uncertainty scheme, any set of priorities can be guessed for any agent. Particularly problematic is the interaction between this extreme randomness, the large variation in market size, and strategic bidders with relatively modest goals; simply guessing the priority of the largest market (New York) incorrectly for the strategic bidder with the smallest budget would add 50%–100% to the estimate of that bidder's overall desired value of licenses. This large uncertainty poses a significant problem for any agent strategy that makes use of knowledge about the other agents, requiring successful strategies to be robust to misinformation.

4 General Agent Strategy

Table 1 summarizes the agent bidding algorithm. The quantities in Steps 2 and 3 are:

Remaining eligibility: bidder's current eligibility minus the bidding units tied up in licenses of which it is provisional winner;
Remaining budget: bidder's total budget[6] minus the money tied up in licenses of which it is provisional winner;
Current values for markets: use the MV or EMV, depending on the agent's priority for the market and the number of licenses in the market for which the bidder is already provisional winner;
Current costs for each market: prices of the least expensive licenses in each market.

[6] We use strict budget limits to more accurately model the real auction; our conversations with a human bidding team revealed they were allocated a fixed block of money by their parent company, and were unable to spend more without extensive negotiation.

Table 1: High-level overview of the general agent bidding algorithm. Agents differ in operationalizations of Step 4

REPEAT (once per round) 1. Get market prices from server 2. Compute remaining budget and eligibility 3. Compute current values and costs of markets 4. *Choose desired licenses within constraints* 5. Submit bids to server at cheapest increment UNTIL game over

Once the set of desired licenses is determined, the agent bids at the 1-increment price for those and only those licenses (Step 5 in Table 1). All agents in this paper use this basic strategy; they only differ in how they choose desired licenses (Step 4). Note that in some implementations Step 5 is conducted in several stages throughout the execution of Step 4, but is otherwise unchanged.

4.1 Knapsack and Improvements

The baseline agent we used was the Knapsack agent. This agent was built on the realization that determining which licenses to bid on for maximum profit given a limited budget is very similar to the classic knapsack problem. The addition of BUs and eligibility costs for each license makes this problem more complex, however treating it as a knapsack problem usually yields optimal solutions [1].

While the Knapsack agent is extremely effective, it has one major weakness: it does not explicitly take into account the presence of other bidders. Our search for improved strategies is driven by two questions:

1. Can we develop a strategy that will beat a field of Knapsack agents?
2. Can we find a strategy that will outperform Knapsack when used by all agents?

The first question assumes that the other agents have not discovered a strategy superior to Knapsack, and so will all be Knapsack agents. Accordingly, it is necessary to adopt a strategy that, when faced with competing Knapsack agents, will generate more profit than when using Knapsack. Certain types of strategies (such as Budget Stretching [1]) which exploit the myopic nature of Knapsack initially appeared promising, allowing significant profit gains in the perfect-knowledge case. However, when we relaxed the unrealistic assumption of perfect knowledge, the gains all but disappeared—the strategies seemed too fragile to be effective in the more realistic setting.

In this paper, we focus on the second question, describing a strategy (PRSDR) which—when used by all agents—will significantly outperform Knapsack and additionally has the stability property that in an auction of all PRSDR agents, each agent is better off sticking with PRSDR than reverting back to Knapsack.

5 Randomized SDR

One simple strategy that can lead to an improvement in the utilities of all involved is Strategic Demand Reduction (SDR) [9]. In SDR, bidders avoid competing with each other for licenses, keeping the prices on those licenses low for all involved. Essentially, bidders using SDR allocate the goods available among themselves and then do not compete on goods they have not been allocated.

For example, consider two identical bidders with \$15 each bidding on two identical items worth \$10 each. Myopic greedy bidding, like Knapsack, would see each bidder obtain one item for \$8, since they can afford to pay \$7 for each item but \$8 for only one. Realizing this, the bidders can reduce their demands to one item each, ensuring both bidders will obtain an item for only \$1, earning \$9 of profit instead of \$2.

Since explicit communication is not permitted, the strategy needs implicit methods to allocate licenses [4]. Hence, it is necessary to use a variant of SDR that can allocate licenses dynamically and with only implicit communication.

The essential idea in *Randomized* SDR (RSDR) is to bid for any desired unclaimed license; recall that ties are broken randomly. The first strategic bidder to become the provisional winner of a license is said to *own* that license. This approach (shown in Table 2) provides step 4 of Table 1's general agent bidding algorithm.

Table 2: Description of the naive RSDR implementation of Step 4 in Table 1

4. Use Knapsack to bid for an optimal set of desired licenses that are not owned by another strategic bidder (i.e., don't steal licenses from strategic bidders).

An agent always takes any licenses it owns back from secondary bidders. The simulator reports a provisional winner for each license that has received a bid, ensuring that each license any agent desires will become owned by a strategic bidder, and hence can be acquired by that bidder without competition from other RSDR agents.

No profit above the valuations of the secondary bidders (the effective price floor) is wasted in figuring out this allocation; with this approach, the strategic bidders will determine their allocation while the secondary bidders are still active.

5.1 RSDR Algorithm

The expected net value to an agent when all agents use naive RSDR is directly related to the total value the agent desires and inversely related to the amount of competition in those desired markets, as is reasonable. However, due to the underlying randomness, it is possible for an agent to be unlucky and receive only a small amount of value.

For each agent, one can compute a *satisfaction* value, which is simply the total value to that agent of the licenses it owns divided by the total value to that agent of all of its goals. If the satisfaction of an agent is sufficiently below that of the other agents, it has been an *unlucky bidder*. If the satisfaction of an agent is low enough, the agent might actually receive less profit from RSDR than from Knapsack.

An unlucky agent can notice that it is such well before the end of the auction, since license allocations are largely determined early. Additionally, an agent can always start out using RSDR and then later switch to using Knapsack.[7] Thus, the unlucky agent would likely switch to Knapsack and earn more while lowering the profits of the other agents, which may be desirable if they are competitors. Accordingly, an unlucky bidder will defect from the RSDR scheme and go back to using Knapsack.

This risk of defection lowers the utility of RSDR, and so should be minimized. One simple method of avoiding such unlucky bidders is by introducing a process of *fairing*. The algorithm allows an unlucky bidder to randomly take licenses that it wants, regardless of ownership, until its satisfaction is no longer too low (Table 3). If this strategy drives another agent's satisfaction too low, the newly unlucky agent would then take licenses from others, until every agent has a reasonable amount. Since not all agents can be above average satisfaction, the threshold that determines an unlucky bidder must be less than the average; in practice, a value around 90% works well, although any threshold low enough to prevent profit loss due to spurious fairing would also suffice.

Table 3: Description of the RSDR implementation of Step 4 in Table 1

4.1.	Use Knapsack to bid for an optimal set of desired licenses that are not owned by a strategic bidder.
4.2.	Submit bids on randomly selected licenses until no longer an unlucky bidder. (i.e., only steal as much as is fair)

Table 3 shows Step 4 of the RSDR algorithm. In Step 4.2, ownership of a license can be taken just by bidding on that license and becoming provisional winner. This process greatly smoothes out the randomness of the scheme, but is not without risks of its own.

5.2 Cheater Detection

The fairing process allows agents to take licenses that other agents had considered their own. This behavior provides great temptation to cheat by deciding not to use RSDR and defecting to Knapsack instead: if the other agents are using RSDR, they will allow you to take any licenses you want, and won't compete for them.

To dissuade agents from cheating, there must be a punishment for it such that cheaters expect to make a *smaller* profit than when they don't cheat. Adding such punishments to RSDR makes a self-enforcing algorithm, called *Punishing* RSDR (PRSDR).

Before cheaters can be punished, they must be detected by observing which agents take licenses from other bidders when they shouldn't. In the *FAucS* domain, three observations are key to our detection algorithm:

1. A PRSDR agent only takes licenses owned by another agent through either fairing or by punishing a cheater.
2. Fairing only occurs when an agent is unlucky (below-average satisfaction rating).

[7] This is since all RSDR or PRSDR agents bid each round on at least enough licenses to attain the average satisfaction level, meaning such agents will always have enough eligibility to be able to switch to Knapsack without disadvantage.

3. A PRSDR agent punishes at most one of the five strategic bidders.

These points tightly circumscribe the situations in which a PRSDR agent will take licenses from another agent; it must either have low satisfaction (fairing) or only take from one agent (punishing). Accordingly, our agents identified cheaters as follows:

– If an agent has a high satisfaction in a round (at least 10% higher than average for that round) and yet it bids on licenses owned by two or more other agents, then it is showing evidence of cheating.
– If an agent shows evidence of cheating in five or more total rounds in the auction, then it is considered a cheater for the remainder of the auction.

The second requirement is a result of the agent's uncertain knowledge about each other's goals, and thus satisfaction levels. Since an agent that appears to have a high satisfaction may in fact not, evidence of cheating in a round is not proof that the agent is in fact a cheater. To reduce false positives, the satisfaction threshold for an agent to be considered a cheater must be greater than the average satisfaction for the round, and this evidence of cheating must be present for more than one round.

In practice, using a threshold of 10% over average satisfaction and requiring evidence of cheating in 5 total rounds led to good results. These settings correctly identified every cheater in all of 60 test runs, while incorrectly identifying a complying PRSDR agent as a cheater in none. Further experimentation indicated that good cheater detection is not particularly sensitive to these two parameters: values within a modest window of those above gave perfect or near-perfect ability to identify all cheaters. Thus, there is no requirement that all agents agree upon a precise set of parameters.

Due to this high accuracy and the minimal effect of the rare errors potentially caused by randomness, we assumed correct detection in our tests. While cheating was all-or-nothing in this implementation, one could envision a more careful cheater who attempted to stay hidden by cheating in moderation. However, any cheating that pushes the cheater's satisfaction over the threshold – usually 10% over the average – will be detected, leaving little room for hidden cheating even with this simple all-or-nothing algorithm. Moreover, the uncertain nature of the other agents' information about an agent means that any amount of cheating raises the risk of being flagged as a cheater, with more cheating causing more risk. In particular, even cheating by adhering to the PRSDR algorithm but with a higher fairing constant–say, insisting on having 100% of average satisfaction instead of the default 90%–raises the chance of being flagged as a cheater and severely punished. We leave the investigation of more sophisticated cheating and cheating detection strategies for future work.

5.3 PRSDR Algorithm

Once the cheater(s) have been identified, many methods of punishment are possible. Our work aimed for aggressive simplicity: if a cheater takes one of your licenses, make it your first priority to take it back and then keep that license away from the cheater (Table 4).

Cheaters are not considered to own licenses, and so any license that the other strategic bidders have any interest in will become owned by one of them. Thereafter, any

Table 4: Description of the full PRSDR implementation of Step 4 in Table 1

4.1.	Bid for any licenses stolen by a cheater. (i.e., don't let anyone steal more than is fair)
4.2.	Use Knapsack to bid for an optimal set of desired licenses that are not owned by a strategic bidder.
4.3.	Submit bids on randomly selected licenses until no longer an unlucky bidder.

license the cheater tries to take will provoke an immediate reaction from the agent perceived to be the legitimate owner of that license, who will not let the cheater have the license until it becomes too expensive to take back, at which point the cheater will receive only a small amount of profit from the license. This removes any advantages due to PRSDR from the cheater while still allowing other agents to profitably cooperate among themselves in other markets.

5.4 Algorithm Detail

Table 5 recaps PRSDR in full algorithmic detail. The table presents only Step 4 of Table 1; otherwise, an agent using PRSDR acts identically to a Basic Agent. The set of strategic bidders participating in this improved strategy is denoted as $\mathcal{B} = \{b_i\}|_{i=1..5}$. Without loss of generality, we use the convention that an agent always refers to itself as agent b_1, and all other strategic bidders are agents b_2 through b_N, where $N = |\mathcal{B}|$. Following this convention, X_i refers to the item X corresponding to agent i.

In Table 5, license selection covers the first half of Part 1 (lines 1–10) and all of Part 2. Part 1 handles internal state, while Part 2 computes the sets of available licenses in each market, which is simply those licenses that are not owned by a strategic bidder and whose price is less than this agent's value for the license. Part 3 implements fairing, and punishment is implemented in the second half of Part 1 (lines 11–16).

6 Empirical Results

In this section we present empirical results demonstrating the effectiveness PRSDR.

Our runs included only the largest 67 markets from Auction 35 (this is the subset of the top 100 U.S. markets that were available in Auction 35), which constitute a large majority of the value in the auction. This helped reduce run times to manageable levels. There were 163 licenses available in these markets. As described above, we used five strategic bidders and five aggregated secondary bidders, with priorities, market values, and budgets randomly selected from a constrained distribution so as to realistically represent the Auction 35 scenario. The secondary bidders are used to simulate realistic auction conditions; hence only strategic bidders are included in the results.

6.1 Homogeneous Strategy Results

Our initial experiment measured the potential gain when all agents use the PRSDR strategy compared to when they all use the Knapsack strategy. Aggregate results from

Table 5: PRSDR algorithm – steps 4 and 5 of general algorithm in Table 1

1. Initialize internal state
   - *bid queue* $\mathcal{Q} = \emptyset, \forall i$, *won list* $\mathcal{W}_i = \emptyset$
   - $\forall\, l \in \mathcal{L}$
     - $PW_l \equiv$ *provisional winner* of l
     - $V_{l,i} \equiv$ *value* of l to b_i given $\mathcal{W}_i$
     - $C_l \equiv$ *price* of l (minimum acceptable bid)
     - if $PW_l = 1$ (i.e., this agent is prov. winner)
       - deduct cost paid, BUs of l
       - add l to $\mathcal{W}_1$
     - $O_l \equiv$ *owner* of l
     - if $O_l \neq b_1$, add l to $\mathcal{W}_{O_l}$
     - if PW_l is a *cheater* and $O_l = b_1$, *lock* l
     - if l is *locked*, $PW_l \neq b_1$, and $V_{l,1} > C_l$
       - submit bid for l
       - deduct cost, BUs of l
       - add l to $\mathcal{W}_1$
     - else if $V_{l,1} < C_l$ unlock l
2. Bid for available licenses
   - $\forall$ markets m
     - $\mathcal{A}_m \equiv \{l \in m | V_{l,1} > C_l, O_l = \emptyset\}$
     - until added *Priority*(m) licenses from m:
       (a) add all $l \in m$ s.t. $PW_l = b_1$ to $\mathcal{W}_1$
       (b) add all $l \in m$ s.t. $O_l = b_1$, l unlocked to $\mathcal{Q}$
       (c) choose $l \in \mathcal{A}_m$ uniformly at random; add to $\mathcal{Q}$
   - $\mathcal{W}_1 = \mathcal{W}_1 \cup \mathcal{Q}$
   - bid on $\mathcal{Q}$ via Knapsack; $\mathcal{Q} = \emptyset$
3. Conduct randomized fairing
   - $\forall\, b_i \in \mathcal{B}$
     - estimate *satisfaction* $\mathcal{S}_i = \dfrac{\sum_{l \in \mathcal{W}_i} V_{l,i}}{\sum_{l \in \mathcal{L}} V_{l,i}}$
   - set $\mathcal{S}_{avg} = \frac{\sum_{b \in \mathcal{B}} \mathcal{S}_b}{|\mathcal{B}|}$
   - While $\mathcal{S}_1 < 90\%$ of $\mathcal{S}_{avg}$
     - choose some $l \in \mathcal{L}$ randomly
     - if $l \notin \mathcal{W}_1, V_{l,1} > C_l$
       - add l to $\mathcal{Q}$
       - add l to $\mathcal{W}_1$
       - recompute $\mathcal{S}_1$
   - bid on $\mathcal{Q}$ via Knapsack

120 runs of each of the Knapsack and PRSDR algorithms can be seen in Table 6. Profits are rounded to the nearest \$10 Million and cost is shown as a ratio of the amount of money paid to the agent's value for the licenses won.

These results show that PRSDR gives a large benefit to each agent over Knapsack. The fourth column gives the ratio between the mean profit earned by that agent in the Knapsack runs—defined to be 1.00—and in the PRSDR runs. The mean of these ratios

Table 6: PRSDR vs. Knapsack results

Method	Agent	*Profit ($M)*	*Ratio*	*Cost*
Knapsack	1	980 (±170)	1.00	.82
	2	650 (±85)	1.00	.82
	3	830 (±91)	1.00	.84
	4	170 (±20)	1.00	.84
	5	550 (±96)	1.00	.86
PRSDR	1	1240 (±210)	1.26	.76
	2	820 (±83)	1.25	.77
	3	1300 (±290)	1.58	.74
	4	300 (±44)	1.78	.79
	5	930 (±240)	1.68	.76

was 1.51, so on average an agent saw more than a 50% increase in profit from using PRSDR instead of Knapsack. Summing over all agents, they earned 44% more profit when all agents used PRSDR than when they used Knapsack.

Note that the standard deviations of the profits are relatively large. Due to random tie-breaking and the highly discrete nature of the licenses being acquired, even simple Knapsack auctions tend to be quite variable; indeed, simply adding one license from one of the biggest markets can easily add 30% to the profit earned by an agent.

The *cost* column is an indication of the amount of potential profit the agents are earning. Recall that secondary bidders had valuations of approximately 75% those of the strategic bidders. Therefore, the strategic bidders were always forced to pay on average 75% or more of their values for all licenses. It can be seen that PRSDR preserved almost all of the remaining available profit, while Knapsack was not nearly as efficient.

6.2 Heterogeneous Strategy Results

The preceding results show the utility of all bidders using PRSDR or RSDR, which are equivalent in all situations without unilateral defection. This section addresses the question of cheaters. Table 7 presents the experiments with a single agent unilaterally defecting to Knapsack, both with RSDR (lacking countermeasures) and with the more robust PRSDR. The first two rows are repeated from Table 6 with the raw profit values removed for clarity and all results aggregated into per-agent averages.

Table 7: Defection and punishment results

Method	*Ratio*	*Cost*
Knapsack	1.00	.84
(P)RSDR	1.51	.76
RSDR Cheater	1.63	.76
RSDR Victims	1.22	.79
PRSDR Cheater	1.02	.83
PRSDR Enforcers	1.17	.81

Rows 3 and 4 reflect the case in which a single agent (the "cheater") defects to Knapsack while all other agents (the "victims") use (non-punishing) RSDR. The results are averaged over 120 runs with each of the 5 agents defecting in 24 of the runs. The results indicate that Knapsack is an effective defection from RSDR.

Note that the victims still do markedly better than by using Knapsack, even with a Knapsack-using cheater—the RSDR algorithm still works as usual in markets in which the cheater does not bid, and even in markets the cheater enters, competition is still kept low. Due to the high price floor brought about by the secondary bidders (75% of value), the cheater cannot afford to buy all the licenses it desires, so many are left for the other agents; in a scenario with a lower price floor, however, this would be less true and there would be more of a benefit to cheating and more of a loss for the victims.

The last two rows of Table 7 reflect the case in which the cheater still defects to Knapsack, but the other agents identify the cheater and use PRSDR against it. Results are averaged over 600 runs with each of the 5 agents defecting in 120 of the runs. In this case, almost all of the cheater's gains over Knapsack are removed by the punitive bidding. Meanwhile, the other agents lose only a small amount of profit by punishing the cheater (1.17 ratio vs. 1.22). Moreover, a lower price floor (under 75%) would likely erase or even reverse this loss of profit for enforcing punishment, since a lower price floor would allow an unchecked cheater to take more licenses and steal more of the profit. In addition, recent evidence suggests people are indeed willing to pay a small amount in order to punish a cheater [2]. Finally, alternative punishment strategies can reduce or remove this negative effect.

These results indicate that given the choice of using PRSDR or Knapsack, all agents do better if they all opt for PRSDR, and no individual agent can improve its profit by unilaterally defecting to Knapsack.

6.3 Additional Results

In this section, we examine the effects of more than one agent defecting to Knapsack simultaneously, and also examine the effects of varying the valuations of the secondary bidders. Table 8 summarizes the results.

Table 8: Robustness results

Method	*Price Floor*	*Ratio*	*Local Ratio*	*Cost*
Multiple Cheater	75%	1.03		.84
Multiple Enforcer	75%	1.01		.83
Knapsack	50%	1.70	1.00	.74
PRSDR	50%	3.42	2.02	.51
Knapsack	75%	1.00	1.00	.84
PRSDR	75%	1.51	1.51	.76
Knapsack	85%	0.68	1.00	.89
PRSDR	85%	0.81	1.19	.87

The first two rows show average results of 72 runs, each with two agents running Knapsack (the "cheaters") and the rest running PRSDR (the "enforcers"). In aggregate, the results are almost identical to scenarios where all agents use Knapsack.

The remaining rows of Table 8 show the results of varying the secondary bidder valuation from the 75% used in the preceding experiments. The values for the run with secondary bidder valuations at 75% are repeated from Table 7. All values in the "Ratio" column are compared to the defined 1.00 baseline set by Knapsack at 75% secondary bidder valuation level, while "Local Ratio" is the ratio between PRSDR and Knapsack at that particular secondary bidder valuation level.

Regardless of the price floor, PRSDR still preserves the available profit margin almost optimally, resulting in only fractionally more than the price floor being paid for the licenses acquired; by comparison, in the low price-floor (50%) case, Knapsack wastes almost half of the potential profit. At the higher price floor, PRSDR still outperforms Knapsack, but only marginally. In this scenario, there is little room for profit since once the secondary bidders drop out, the prices are very close to the agents' values, and at most one bid can be made above the 85% price floor before the license is too expensive.

Overall, these results indicate that PRSDR is an effective and stable strategy in the auction scenario considered here.

6.4 Improved Auction Design

The PRSDR bidding strategy relies on the bidders' ability to exchange information via the auction mechanism, for example via their bids or knowledge of current provisional winners; this amounts to implicit collusion. Since the auctioneer — in this case, a powerful government — receives much of the same information as the bidders in terms of rough goal estimates and bid history, it is important to note that the auctioneer is not able to simply shut down PRSDR. Most important to note is that the auctioneer is bound by the auction rules just as strongly as the bidders, preventing it from arbitrarily levying punishments against bidders whose actions it does not like.

Moreover, especially since the auctioneer is highly unlikely to have information more precise than the very rough estimates available to the bidders, it is not at all clear that the auctioneer would even be able to conclude with any confidence that collusion was taking place. While the auctioneer can employ the same cheater-detection algorithm as the bidders, it is crucial to note that this algorithm can never provide proof than any agent is indeed using PRSDR, only that the agent may be using any number of possible strategies, including PRSDR, that lead to the observed bidding behaviour. Absence of evidence is not evidence of absence, so concluding a bidder is using a collusive strategy rather than some unknown non-collusive strategy is by no means a trivial problem. Accordingly, the efficiency of the auction may be improved by effecting rule changes that make it harder for bidders to exchange the information necessary to coordinate SDR.

One way to limit information exchange is to reduce the total number of bidding rounds. Our paper demonstrates that unrealistically low minimum starting bids make coordinating SDR easier by allowing exchanging signals via bids that are still unrealistically low and thus totally risk-free. A higher minimum bid (perhaps 40% of the license's presumed value) would permit agents to place far fewer bids on the license, thus limiting their ability to coordinate demand reduction using PRSDR or other methods. Weighing against this, the minimum bid cannot be too high as the auctioneer usually wants to ensure the sale of the item even if it overestimates its value. Moreover, PRSDR may be able to succeed even in this setting, as the higher minimum bid only lowers the risk-free rounds in which to do fairing but does not remove all opportunities.

A second possible way to prevent PRSDR is to restrict the information the auctioneer makes available to the bidders. For example, if agents are only told current prices and the licenses of which they are provisional winner, then PRSDR is impossible to implement: the bidders cannot tell which licenses are unowned or which were taken by a secondary bidder. On the other hand, there are good reasons to reveal information to the bidders from a theoretical perspective [3], as more information tends to result in a more efficient auction. Sealed-bid auctions also remove this information channel, but are more challenging for bidders and force the auctioneer to solve the winner determination problem [7]. Ultimately, in realistic complex scenarios there may be no perfect auction mechanism. Our research, as exemplified in this paper, focuses on improving agent bidding strategies given a fixed mechanism.

7 Discussion

In Section 4.1 we laid out one of the goals of this work as trying to "find a strategy that will outperform Knapsack when used by all agents." In Section 6 we demonstrated that PRSDR is such a strategy. However, it wouldn't be a practical strategy were it possible for an individual agent to improve its performance by unilaterally changing back to Knapsack while the others use PRSDR. Thus, we also demonstrated that PRSDR is self-enforcing.

This last property is the key property of Nash equilibria strategies in game theory. Indeed, one way of looking at this result is as an equilibrium strategy of a repeated bimatrix game as examined in [4][8]. A bimatrix game is defined by a pair of matrices M_1 and M_2 of the same size (same number of rows and same number of columns). At each stage, the players choose actions, a row i for the row player and a column j for the column player. The row player receives payoff $M_1[i, j]$ and the column player receives payoff $M_2[i, j]$. The objective for the players is to maximize their average payoff over an unbounded number of stages.

For example, consider the following 2×2 game.

$$M_1 = \begin{bmatrix} 3 & 0 \\ 5 & 1 \end{bmatrix}, M_2 = \begin{bmatrix} 3 & 5 \\ 0 & 1 \end{bmatrix}.$$

If the row player selects action 0 and the column player selects action 1, then the row player receives payoff 0 and the column player receives payoff 5.

This game is commonly known as the prisoner's dilemma. It roughly characterizes the interactions between the strategies RSDR (action 0) and Knapsack (action 1): If both

[8] Although the agents presented in this paper do not maintain state from run to run as is typically the case in an iterated scenario, the fact that each run consists of many rounds gives the agents a chance to respond to each others' strategies.

agents do RSDR, they are both better off than if they both do Knapsack; but knowing that the other agent is going to do RSDR, it is better to defect to Knapsack.

PRSDR in effect adds a third action possibility to the game such that if both players do it, there is no incentive to defect: it is a pure Nash equilibrium. The following 3×3 game roughly characterizes this situation, with action 0 representing RSDR, action 1 representing PRSDR, and action 2 representing Knapsack.

$$M_1 = \begin{bmatrix} 3 & 3 & 0 \\ 3 & 3 & 2 \\ 5 & 1 & 1 \end{bmatrix}, M_2 = \begin{bmatrix} 3 & 3 & 5 \\ 3 & 3 & 1 \\ 0 & 2 & 1 \end{bmatrix}.$$

These numbers are reflective of the data presented in Table 7[9]. The 3's characterize the fact that if all agents play PRSDR *or* RSDR, they all do equally well. It is only when one of the agents plays Knapsack that the payoffs deviate.

Note that in this game, both agents playing the middle action (reflecting PRSDR) is a pure Nash equilibrium with each agent getting a payoff of 3, while in the prisoner's dilemma the only pure Nash equilibrium is when both agents play action 1 (Knapsack), giving them each a payoff of 1. In the larger (3-action) game, unlike in prisoner's dilemma, the Nash equilibrium strategy is also pareto optimal, meaning that no agent can achieve a better result without harming the other. This demonstrates the potential utility of adding a "punish" action to the space of possible actions.

Ideally, we would like to conclude that a version of PRSDR is a an equilibrium strategy within the full *FAucS* domain. However, identifying equilibrium strategies in such a complex scenario is far from straightforward. The preceding analysis applies only for the three strategies considered in this paper: Knapsack, RSDR, and PRSDR. The possibility remains open that there exist other strategies that outperform the general PRSDR strategy.

8 Conclusions and Future Work

The general PRSDR strategy applies to any simultaneous multiple-round (SMR) auctions in which there is room to bid below the expected sell prices and in which their are not complex value interdependencies among the goods for sale. By this strategy, bidders tacitly agree to reduce demand provided that others appear to be doing the same. While the full details of PRSDR as presented in this paper are specific to the FCC spectrum auction scenario, the goal of this work has been both to introduce the general strategy and to implement it fully in a complex, realistic auction scenario.

There is some evidence that strategies related to PRSDR actually do emerge in practice. PRSDR is essentially a randomized scheme for SDR in that bidders allow

[9] The small cost agents pay to punish a cheater is arguably acceptable [2], and is likely an artifact of our particular implementation, as it utilised a particularly aggressive punishment scheme that concentrated on powerful deterrence over short-term profit maximization. Moreover, a lower small-bidder price floor would allow a cheater to steal more licenses, likely worsening the loss victims suffer and making punishment more attractive. Accordingly, this game-theoretic analysis concentrates on the general characteristics of the scenario.

a random process to influence their winnings. Roth and Ockenfels [6] argue that the common practice of late-bidding (sniping) in eBay auctions is evidence that people use similar randomized strategies.

As we examined only a small part of the strategy space, we make no claims that PRSDR should or would emerge in real-life spectrum auctions. However, given that the Knapsack strategy is relatively effective against unknown strategies, and hard to beat unilaterally if all other agents adopt it, it is significant to note that a stable, robust, and more profitable strategy exists.

More than creating a bidding strategy for immediate use, this work examines interesting and promising ideas, the concepts behind which may be applied in future work to define directly-usable bidding strategies or countermeasures to such strategies. In particular, PRSDR demonstrates that useful implicit communication is possible even in realistic settings where significant work has been devoted to the problem of preventing communication. Our work suggests general methods both for exploiting this available communications channel and for blocking information transfer in this means. PRSDR as given is indeed usable, however, so the question remains how this algorithm would be employed, especially given the restrictions on communication present in many real auctions. While PRSDR relies on assumptions about the other bidders to be effective, the necessary assumptions are limited to the following:

1. All bidders are actively seeking means to maximize their profit.
2. They are likely to have heard of or discovered PRSDR on their own and realize that PRSDR leads to more profit than competitive bidding.
3. They are open to trying a rational profit-seeking strategy like PRSDR.

As noted, bidders can, with no harm to themselves, abandon PRSDR partway through if not enough others are using it. Hence, starting out using PRSDR has little to no risk and great potential reward: it is worth giving others the chance to use PRSDR so everyone can win. If enough of the biggest bidders in an auction independently decide to try starting with PRSDR, there is the potential for them to improve their profits without ever having had a bit of communication outside of the bids they place.

Our ongoing research focuses on continuing to explore the strategy space of this particular auction mechanism while generalizing our results to other scenarios.

References

1. J. A. Csirik, M. L. Littman, S. Singh, and P. Stone. FAucS: An FCC spectrum auction simulator for autonomous bidding agents. In L. Fiege, G. Mühl, and U. Wilhelm, editors, *Electronic Commerce: Proceedings of the Second International Workshop*, pages 139–151, Heidelberg, Germany, 2001. Springer Verlag.
2. E. Fehr and S. Gachter. Altruistic punishment in humans. *Nature*, 415:137–140, 2002.
3. P. Klemperer. Auction theory: A guide to the literature. *Journal of Economic Surveys*, 13(3):227–86, July 1999.
4. M. L. Littman and P. Stone. Implicit negotiation in repeated games. In *Proceedings of The Eighth International Workshop on Agent Theories, Architectures, and Languages (ATAL-2001)*, August 2001.

5. J. A. Rodriguez-Aguilar, F. J. Martin, P. Noriega, P. Garcia, and C. Sierra. Towards a test-bed for trading agents in electronic auction markets. *AI Communications*, 11(1):5–19, 1998.
6. A. E. Roth and A. Ockenfels. Last-minute bidding and the rules for ending second-price auctions: Evidence from eBay and Amazon auctions on the internet. *American Economic Review*, 2001. To appear. Available at `http://www.economics.harvard.edu/~aroth/alroth.html`.
7. T. Sandholm. Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence*, 135:1–54, 2002.
8. P. Stone and A. Greenwald. The first international trading agent competition: Autonomous bidding agents. *Electronic Commerce Research*, 2002. To appear.
9. R. J. Weber. Making more from less: Strategic demand reduction in the FCC spectrum auctions. *Journal of Economics and Management Strategy*, 6(3):529–548, 1997.
10. M. P. Wellman, P. R. Wurman, K. O'Malley, R. Bangera, S.-d. Lin, D. Reeves, and W. E. Walsh. A trading agent competition. *IEEE Internet Computing*, 5(2):43–51, March/April 2001.