Truthful Prioritization Schemes for Spectrum Sharing

Victor Shnayder, Jeremy Hoon, and David C. Parkes
{shnayder,parkes}@eecs.harvard.edu, jhoon@fb.com
School of Engineering and Applied Sciences
Harvard University

Vikas Kawadia
vkawadia@bbn.com
Raytheon BBN Technologies

Abstract—As the rapid expansion of smart phones and associated data-intensive applications continues, we expect to see renewed interest in dynamic prioritization schemes as a way to increase the total utility of a heterogeneous user base, with each user experiencing variable demand and value for access. We adapt a recent sampled-based mechanism for resource allocation to this setting, which is more effective in aligning incentives in a setting with variable demand than an earlier method for pricing network resources due to Varian and Mackie-Mason (1994). Complementing our theoretical analysis, which also considers incentives on the sell-side of the market, we present the results of a simulation study, confirming the effectiveness of our protocol in aligning incentives and boosting welfare.

I. INTRODUCTION

"I had to give my shirt to AT&T for my data plan," goes a recent TV commercial. With users now commonly owning several devices that need wireless broadband connectivity (smartphones, tablets, ebook-readers), the effective allocation of wireless bandwidth capacity is becoming a serious issue. High variability in data consumption further compounds the problem. According to a recent study by Nielsen Co. [1], the average data consumption was less than 300MB per month but the top 6% of users consume half of all wireless data.

Our focus is on dynamic (re)prioritization of access to wireless spectrum. Whereas a static allocation is inefficient because of high variability of demand, the goal in contemplating a market for prioritized access is to ensure that capacity is always in the hands of users with the most instantaneous value.

A further consideration is that the protocol promote stability, in the sense that incentives should be aligned with truthful reporting of value for network access. Achieving *incentive compatibility* in this sense is useful in promoting simple control logic for devices, when competing for access on behalf of users, and also in avoiding the churn and system overhead that would occur if devices could benefit from adapting their bids to the bids of others.

Our particular application focus is on making wireless broadband allocation more efficient, through the design of a protocol for auctioning wireless broadband bandwidth. Sellers in this market have wireless broadband (3G/4G) enabled devices with a wireless LAN radio such as 802.11 and a monthly data plan, and the ability to facilitate bandwidth sharing through tethering apps which essentially allow them to act as routers. Buyers have WiFi devices and pay the sellers to relay their data. While auctions can introduce an extra burden on users over simply charging a flat fee, they can greatly increase the efficiency of the system, with the complexity hidden through automated bidding agents.

We describe a competitive prioritization scheme where users submit bids and generate traffic to be prioritized for forwarding at the shared router. The scheme is incentive compatible for a setting in which each user has a fixed per-packet value over a fixed period of time (specific to the user). The scheme is presented for an 'open loop' demand model in which each user's next packet is generated to be sent after some arbitrary delay from the previous packet. The results also extend to a 'closed loop' model where the delay to the next packet occurs after the current packet is forwarded by the router, not generated by the user. In particular, a user can do no better than bidding truthfully and consuming bandwidth as soon as demand is realized, with no regret as to the outcome, whatever the future realization of demand.

Our approach adopts a recent innovation in computational mechanism design due to Babaioff et al. (BKS) [2], which provides incentive compatibility in domains where payments cannot be computed through "counterfactual" information. Counterfactual information is data about what an allocation would have been under alternate bids. This is not available in network settings, where only the demand consistent with the *actual* prioritization is available; e.g., it is not knowable what a low-priority user who attempted to send one packet over some period of time would have demanded if that packet had been forwarded.[1]

We contrast the proposed scheme with an early idea due to Varian and Mackie-Mason (VMM) [6]. VMM proposed a per-packet auction based on packets dropped from a router's queue. This is a myopic pricing scheme, and leads to a failure of incentive compatibility in environments with dynamic demand. In comparison, the approach enabled by BKS allows for payments that account for variable demand, estimating the actual externality imposed on other devices by the prioritization

---

[1]Babaioff et al. [3] and Devanur et al. [4] first showed that the unavailability of counterfactuals impedes the design of truthful mechanisms, in the context of multi-armed bandit problems. Cavallo et al. [5] also study incentives in multi-armed bandit setting. We give a practical motivation for this work in the context of networking.

associated with a device.[2]

A new challenge in our environment is to ensure that buyers will *trust* the seller (i.e., the owner of the wireless broadband device) to faithfully follow the rules of the proposed auction. To address this, we introduce a *revenue pooling scheme*, which ensures that sellers have no profitable manipulations by making the mechanism appear to sellers as a variation on a first-price auction in which the sellers pay a fixed "tax rate" on their revenue. Our scheme is similar to the Hartline-Goldberg random-sampling approach for digital good auctions [12].

We present the results of a simulation study, which confirms that the mechanism achieves truthfulness and arbitrary approximations to allocative efficiency (allocating the shared resource to those who value them the most). In testing robustness to more realistic assumptions, we consider the behavior of the market when bid periods are not exactly aligned with market dynamics, examining the effect on performance. Finally, we examine the distributional effects of revenue pooling.

II. BASIC NETWORK MODEL

We consider a single seller with a router, and a set of n buyers (or users) $N = \{1, \ldots, n\}$. The users send packets to the router, which queues them and forwards them in some order. For simplicity, time is discretized into epochs, where an epoch is the time it takes to forward a packet. New packets from users arrive at the start of each epoch.

In each epoch, any new packets received by the router are added to the queue, and if the queue is non-empty, the router forwards a packet chosen according to a routing policy. We assume that if the router drops any packets from the queue, they are immediately resent by the corresponding user.

The seller sets a per-packet reserve price $r \geq 0$. This ensures the the per-packet revenue is at least r. The mechanism runs between start and end epochs $\{0, T\}$, and each user is active for some subinterval of $[0, T]$. The stochastic demand d_i of user $i \in N$ is modeled as a sequence of random variables d_{ij}, representing the delay between the sends of sequential packets. Even if many previous packets are still waiting to be forwarded, the buyer generates an additional packet to send.

Each user has a per-packet value v_i and realized utility at time T of,

$$u_i(T) = f_i(T)v_i - p_i, \tag{1}$$

where $f_i(t)$ is the total number of packets forwarded for user i by time t, and p_i is the total payment made by the user at the end of T epochs.

Each user submits a single bid $b_i \geq 0$ to the router.[3] On the basis of bids $b = (b_1, \ldots, b_n)$, a routing policy determines the order in which packets are forwarded at the router. Two

possible policies are *first-in first-out* (FIFO) and *highest-bid-first*. Under highest-bid-first, if one buyer has a higher bid than another, then the router always routes any queued packets from the first before any packets from the second. Given this, let $\mathcal{A}_i(b, d)$ denote the *allocation* to user i (itself a random variable, depending on the realized demand $d = (d_1, \ldots, d_n)$). This is the number of packets forwarded by the router for user i by time T. Let $\mathcal{A}(b, d) = (\mathcal{A}_1(b, d), \ldots, \mathcal{A}_n(b, d))$.

A. Fixed price and VMM

A baseline comparison is provided by FIXED, which charges a fixed price and uses a first-in first-out routing policy. This scheme is simple and incentive compatible, but can be very inefficient, as it is not responsive to variable demand or value.

The other comparison is provided by the VMM auction mechanism, in which uses a highest-bid-first routing policy. Payments in VMM are computed per-packet, by charging the owner of each forwarded packet the *immediate externality* imposed by the packet. Specifically, the price for a forwarded packet is *the value of the highest value packet that was dropped while the forwarded packet was in the queue.*

Some observations about the VMM mechanism:

- Longer queue length implies lower payments. In particular, if the queue length is one, then this is exactly a second-price auction among all arriving packets.
- VMM only charges for dropped packets, not for the delay imposed by a packet on other packets.
- The payment scheme in VMM is not incentive compatible in a setting with variable demand. Consider the following:

Example. *Suppose that the queue length is one, and there is one buyer with value $3 per packet who wants to send one packet every epoch and a second buyer with value $2 per packet who wants to send a single packet (available to send in epoch one and available through the time period of interest to buyer 1). Now truthful bidding by buyer 1 will result in a charge of $2 for each packet, representing the per-epoch externality on the packet that buyer 2 keeps trying to send (recall that senders in our model are persistent, and will keep resending dropped packets). On the other hand, a bid of less than $2 would "flush" the packet of buyer 2 and then allow buyer 1 to send for the remaining epochs with payment $0. The tradeoff is to reduce the number of packets forwarded by 1 in return for a significant reduction in total payment.*

B. The BKS mechanism

The problem with VMM in the example is that it overestimates the externality because it does not have access to the information that buyer 2 only has a single packet to send. But this information is not available, since we insist that demand models are not described by users or known by the prioritization scheme.

The innovation of the BKS scheme, applied in the paper of Babaioff et al. [2] to a multi-armed bandits problem and some simpler settings, is to use "self-sampling," where a randomized perturbation to bids obviates the need for counterfactual

---

[2]Other approaches from dynamic mechanism design are unsuitable, either because they rely on counterfactual information [7] or rely on a probabilistic demand model [8]. VMM's work was continued in related approaches, such as the progressive second-price auction [9], and various follow-on work ([10], [11]). As with VMM, these papers do not achieve incentive alignment in dynamic settings. They also assume the existence of a trusted router.

[3]In our simulations, we also consider settings where users can bid more than once.

information. The idea is to obtain an estimate of the network resources that a user would have consumed at some lower bid as a side-effect of the randomization.

We describe the BKS scheme, adapted here to allow the seller to employ a *reserve price* $r \geq 0$, which is the minimal per-packet price a seller will accept.[4] The scheme is parameterized by $\mu \in (0, 1)$, which governs the probability of introducing a random perturbation into bids.

Definition 1 (BKS): Given an allocation rule $\mathcal{A}$ and a parameter $\mu \in (0, 1)$, the BKS procedure in our setting is:

1) *Upon arrival, each bidder i submits a per-packet bid $b_i \geq r$.*
2) *The mechanism computes a transformed bid $\tilde{b}_i$ for each i as:*
   a) *With probability $1 - \mu$, $\tilde{b}_i = b_i$*
   b) *Else, compute a reduced bid: pick $\gamma \in [0, 1]$ uniformly at random, and set $\tilde{b}_i = r + (b_i - r) \cdot \gamma^{1/(1-\mu)}$*
3) *For all T epochs, the router uses the allocation rule $\mathcal{A}$ applied to the transformed bids $\tilde{b}$ of the active users.*
4) *Given the realized quantity of packets forwarded, $a_i \geq 0$, for each user i, collect payment from user i as:*
   a) *Collect $b_i a_i$.*
   b) *If $\tilde{b}_i < b_i$, give a rebate $R_i = \frac{1}{\mu}(a_i(b_i - r))$. Otherwise, $R_i = 0$.*

This procedure does not use counterfactuals: bids are perturbed, used for prioritized routing, the total number of packets forwarded is observed, and payments are made with a rebate-adjusted "pay-your-bid" payment scheme.

Definition 2: A mechanism is truthful-in-expectation *if a risk-neutral buyer maximizes expected utility by bidding truthfully, whatever the bids of others and whatever realized demand, where the expectation is taken with respect to random coin flips of the mechanism.*

In achieving truthfulness-in-expectation, an essential property is that an allocation rule is *monotone*. This insists that for all users i, and all realized demand d, and all bid vectors $b = (b_1, \ldots, b_i, \ldots, b_n)$, that $\mathcal{A}_i(b', d) \geq \mathcal{A}_i(b, d)$, for $b' = (b_1, \ldots, b_i', \ldots, b_n)$ and $b_i' > b_i$. We will establish monotonicity for highest-bid-first prioritization. For now, we state the following result, due to Babaioff et al.:

Theorem 2.1 ([2]): Applying the BKS procedure with probability of perturbation μ to a monotone allocation rule results in a truthful-in-expectation mechanism.

In addition, the allocation is the same as in the original allocation rule with probability at least $1 - n\mu$, where n is the number of buyers.

BKS prove this by showing that the scheme obtains an unbiased sample of an integral that defines the payment rule

in the canonical approach of incentive-compatible mechanism design [13].

Let's revisit the earlier example of manipulation in the VMM scheme. Under BKS, when the first buyer's bid is not resampled, they pay \$3 per packet, and send some number of packets k. When their bid is resampled, the first buyer will have either k or $k-1$ packets forwarded, depending on whether the resampled bid was below \$2. This will result in a large rebate, and in expectation, the first buyer's payment will be essentially \$2. The exact value depends on μ.

III. THEORETICAL RESULTS

In this section, we summarize our results for buyer truthfulness and describe our approach to ensuring seller non-manipulability. We defer proofs to an extended version of the paper. We split our discussion into buy-side and sell-side.

A. Buy-side

In addition to truthful bidding, we are interested in establishing that the scheme is *demand revealing*, in that it is a weakly dominant strategy for a user not to introduce any additional delay between the packets sent over the network.

The main result is:

Theorem 3.1: The BKS mechanism with highest-bid-first routing is truthful-in-expectation and demand-revealing.

The proof is established by demonstrating monotonicity for highest-bid-first priority routing. For this, we start with the highest priority bidder, define sets of *available epochs* for each bidder, and can immediately show through an inductive argument that higher bids result in larger sets of available epochs. This holds for all realized demand and all bids by other users. We get the demand-revealing property because only later allocations but not earlier allocations are possible (and thus interesting) for a user, and delaying demand can only reduce a user's allocation because available epochs depend only on the bid and the bids of higher-priority buyers.

B. Sell-side

If implemented on an untrusted device, the seller can manipulate in various ways, for example by avoiding random perturbations, charging more than it should, or avoiding routing traffic when a bid has been down-sampled. Each one of these manipulations will increase the seller's revenue.

To preclude this, we propose a combination of cryptographic methods and incentive engineering, transforming the revenue stream to each seller by leveraging a pool of sellers.

The approach, ALIGN-TRUST, uses a trusted central server for accounting and other validation, logically inserting it between the buyer and seller. Buyers pay the center directly, and the center pools the revenue and then pays sellers.

ALIGN-TRUST relies on two properties: the system must ensure that sellers cannot tamper with bids, and that bid perturbation is done correctly. Buyers can cryptographically sign their bids to achieve bid integrity. A simple way to ensure correct bid perturbation is to require that it be done by the central server. This requires extra communication, but we hope

---

[4]Technical note: To support the reserve price, we use the *h-canonical self-resampling procedure* described by BKS, with $h(z, b) = r + z(b - r)$, which has distribution function $F_h(a, b) = (a - r)/(b - r)$. In Section 3.4, the BKS paper claims that $F_h = F_0$ for all h, where F_0 is the distribution function for the canonical resampling procedure, but F_0 doesn't satisfy their condition on F_h: $h(F_h(a, b), b) = a$ for all $a, b \in I, a < b$. $F_h(a, b) = (a - r)/(b - r)$ does satisfy this condition.

that this can be reduced via the use of pseudo-random number generators, leading to verifiable decentralized perturbations.

We are not assuming that our trusted central server is omnipotent: we rely on it to verify bid integrity and resampling, but do not assume that it can observe or enforce anything about the sellers' routing decisions. The incentives provided by ALIGN-TRUST will ensure that routing as specified is optimal for sellers, removing the sellers' incentives to reduce the allocation to buyers with resampled bids.

To ensure that following the highest-bid-first routing policy is optimal for sellers, our method pays sellers the resampled bid for each packet. Because the mechanism also pays rebates to the buyers, just paying resampled bids would leave the center with a deficit. We compensate by taxing the sellers a percentage of their revenue. The details ensure that sellers cannot reduce the tax rate they must pay, so they maximize revenue by following the protocol.

In ALIGN-TRUST, time is divided into *accounting periods*, perhaps a month long in practice. Each seller is required to pick one of a small set of reserve prices. Consider the pool of sellers that select the same reserve price. We apply the following system-wide payment mechanism for the auctions involving these sellers.

*Definition 3 (*ALIGN-TRUST*):* Consider an accounting period, and a set of sellers M with the same reserve price.

1) Charge each user the (rebated) BKS payment across all completed auctions for all sellers in M, crediting each seller the first-price revenue at the perturbed bids, without including the rebates. The rebates leave the center with a deficit.

2) Randomly split the sellers in M into two disjoint sets S_1, S_2. Let C_1^{out} denote the total credit above reserve price to sellers in S_1. Let C_1^{in} denote the total payment received from buyers that participated in auctions of sellers in S_1. Define C_2^{out} and C_2^{in} similarly for sellers in S_2. The difference between C^{out} and C^{in} in each set of sellers is that set's total deficit.

3) Define tax rate $tax_1 = \frac{C_2^{\text{out}} - C_2^{\text{in}}}{C_1^{\text{out}}}$ and $tax_2 = \frac{C_1^{\text{out}} - C_1^{\text{in}}}{C_2^{\text{out}}}$. Collect $tax_1 \cdot C_1^{\text{out}}$ from sellers in S_1, charging tax rate tax_1 uniformly across all sellers based on the above reserve credit made to them in step 1. Collect $tax_2 \cdot C_2^{\text{out}}$ from sellers in S_2, charging tax rate tax_2 uniformly across all sellers.

From the point of view of the sellers, ALIGN-TRUST transforms the mechanism into a first-price auction with a revenue tax. A seller will receive some share of the revenue achieved from forwarding packets at the perturbed bid values, with the share maximized by following the highest-bid-first prioritization scheme:

Theorem 3.2: With probability that tends to 1 exponentially quickly in the number of sellers with a given reserve price, ALIGN-TRUST applied to BKS allows no profitable manipulations of the routing policy by sellers and provides each seller with revenue at least its reserve price. The ALIGN-TRUST applied to BKS mechanism leaves incentives unchanged for
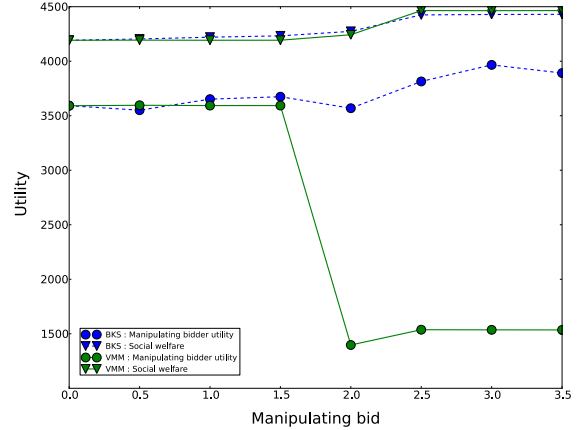

Fig. 1. Manipulation of VMM by Buyer 0.

buyers (= users). The total payment received and made to all sellers by the center is exactly balanced.

The proof relies strongly on our demand model: because the set of packets available to a seller does not change based on the routing policy, the seller maximizes revenue by forwarding the most valuable packets first, and our payment rule ensures that these are the packets with highest resampled bids.[5] We show exponential convergence by using Hoeffding's inequality to bound the difference between the payments in the two pools.

IV. SIMULATION RESULTS

We now present selected results from a simulation study, which confirm our theoretical analysis and examine the behavior of our scheme in more complex settings where our theoretical results do not apply. In particular, we examine VMM truthfulness, compare the mechanisms' economic efficiency, explore how BKS works under uncertainty, and measure the volatility and distributional effects of ALIGN-TRUST on seller revenues.

We use a custom, discrete event, packet-level simulator for single-seller bandwidth trading markets. Buyers place bids and send and receive data. The seller has a fixed size packet queue which supports either FIFO or strict priority routing: in BKS and VMM, the lowest priority packet gets dropped when the queue is full. In FIXED, the arriving packet is dropped.

Unless stated otherwise, we use the following parameters, chosen to illustrate the properties of our scheme. Each run lasts 3000 simulated seconds. The seller's forwarding capacity is 1.5 packets per second. The seller's packet queue can store 10 packets. Buyers generate packets according to a Poisson distribution with a given mean.[6] A slight difference from our theoretical demand model is that when buyer packets get dropped, they are queued at the buyer to be resent quickly but not instantaneously. The μ parameter of BKS is 0.2. The

[5]See Section V for more about demand models.

[6]Note that if we could assume that demand was always Poisson, we could use statistical methods to estimate counterfactuals without using BKS. We use Poisson demand in our simulations, but do not allow our mechanism to rely on that knowledge.
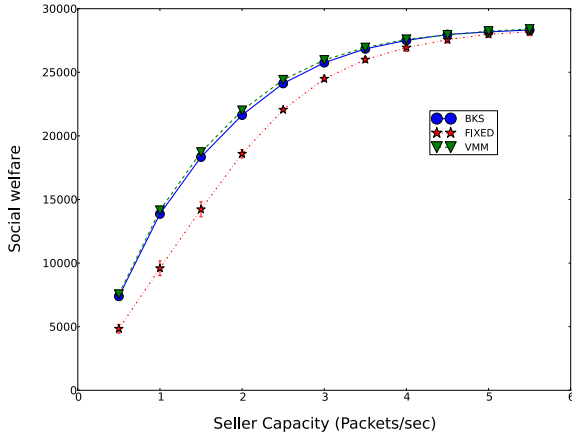
Fig. 2.   BKS provides truthfulness without compromising on efficiency.

Fig. 3.   Effect of manipulation on BKS when a buyer revises bids frequently.

reserve price is 0. Each plot shows an average over at least 100 simulation runs. In many of the simulations, we focus on the properties in regard to a single buyer, and emulate the total demand of everyone else via a second buyer.

A. Incentive properties of VMM

Building on the counter-example in Section II-A, Figure 1 confirms that VMM is not incentive-compatible, and that the manipulator's gain in utility and the drop in social welfare can be significant. In this experiment, there are two buyers. Buyer 1 has value 2 and bids truthfully. Buyer 0, the manipulating buyer, has value 3, and varies its bid as shown on the x axis. Buyer 0's average demand is 5 packets/sec and that of Buyer 1 is 1 packets/sec. The seller capacity is 5 packets/sec and the seller queue is has size 3. We show the average of 1000 runs of 300 simulated seconds each. The bottom two lines show buyer 0's utility: with VMM, when bidding less than buyer 1, buyer 0 pays 0, and improves utility. In contrast, underbidding does not help with BKS. The top two lines show social welfare, and confirm that manipulation noticeably harms efficiency.

B. BKS: Efficiency and incentive properties

In Figure 2, we compare the efficiency of BKS, VMM, and FIXED as seller capacity varies, with truthful bidding. Buyer 0's demand is 0.5 and buyer 1's is 1.5 packets/sec. BKS and VMM are significantly more efficient than FIXED under congestion, while BKS is almost as efficient as VMM. The slight drop in efficiency from VMM to BKS is due to the bid resampling in BKS. Thus, the BKS scheme provides truthfulness with almost no adverse effect on efficiency.

We now turn to a setting where our theory does not apply directly. We modify the simulated auctions to allows new bids to be submitted at regular intervals, where these intervals may not be correctly aligned with the underlying market dynamics. To simulate this, we allow a buyer's demand to change at any time. We define *bid duration* as the time period between bids imposed by the market design, and thus the time period over which BKS resampling is done. We study the affect of varying this bid duration.
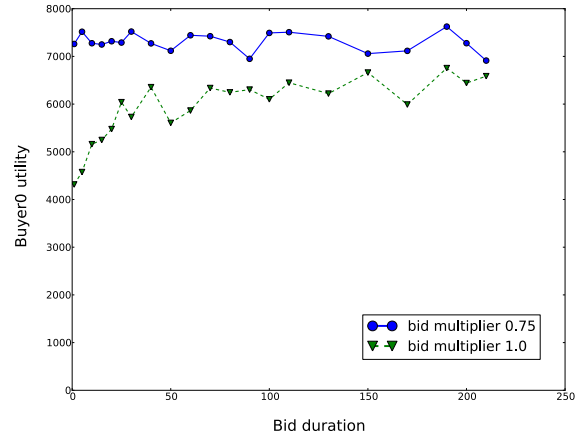
Figure 3 shows the effect of manipulation on BKS for various bid durations up to 225 seconds. Here, buyer 1 has constant value of 5 and demand of 1 packet/sec, and faces another buyer with value 4 but whose demand changes randomly between 0.5 and 1.5, every 10 seconds. Figure 3 shows buyer 1's utility for truthful bidding (solid line) and under-bidding (shaving his bid by 0.75, dotted line). For low bid durations, underbidding is profitable, because it avoids overpaying when buyer 2's demand is high (and since the bid duration does not correctly capture the bursty demand of buyer 2.) For longer durations, the mechanism is able to correctly estimate the externality imposed on buyer 2 by buyer 1. This demonstrates that setting a long enough bid revision time can mitigate opportunities for manipulation even when the sessions for bidders are no longer exactly aligned with the bid revision opportunity, as is assumed in the theory.[7]

C. Revenue pooling

In this section we confirm our theoretical results about revenue pooling in simulation, showing that the center will not have to run a loss while also demonstrating that pooling reduces the variance of seller revenue. Figure 4 shows the pooled revenue vs unpooled revenue for 200 sellers from four different settings: capacities 2.0 or 4.0, and buyers with total demand 3 or 4.5. These are indicated by different markers. Note the dramatic reduction in variance of pooled seller revenue for each seller category: without revenue pooling, the seller revenue varies from -10000 to almost 45000, despite each point being sampled from similar settings. However, the pooled revenue varies very little (less than 10000), thus resulting in a horizontal clustering of points for a given seller category. The sellers do not all end up with the same revenue—sellers with higher non-pooled revenue have higher final revenue as well.

---

[7]We have also examined the effect of bid duration when a buyer has dynamically varying per-packet value, and a good compromise can be struck between permitting frequent enough opportunity for value revision with long enough sample periods to assess externality correctly and avoid over-payments.
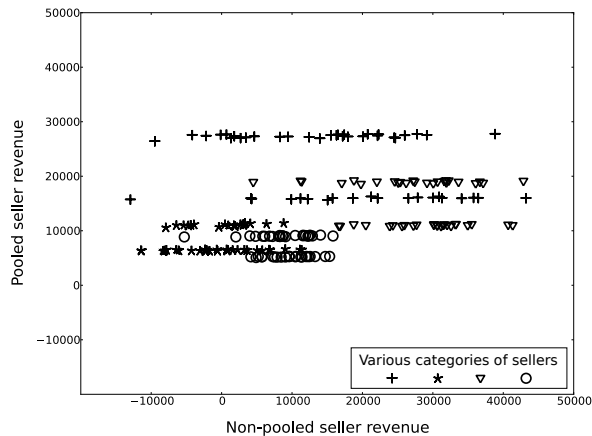
Fig. 4.   Effect of revenue pooling.

V. Discussion

We now discuss two technical issues. The first has to do with the availability of counterfactual allocations. In the simple open-loop demand model used in this paper, where packet arrivals do not depend on the router behavior, the router could in principle log all packet arrivals and compute counterfactual allocations for different bid vectors by replaying the log. BKS is still relevant for two reasons. First, logging and replaying the logs is impractical for space and performance reasons. More fundamentally, the availability of counterfactuals is an artifact of our model, which elided many real-world aspects of networks for analytic tractability. In realistic network settings with adaptive buyers and varying network conditions, counterfactual allocations are not available.

What would it take to apply BKS to such settings? One requirement is that the underlying allocation rule must be monotone. In routing, this depends on the combination of the selected routing rule and the demand model. With our simple demand model, it is easy to show monotonicity for the highest-bid-first routing rule, and we have preliminary results that suggest that many other natural routing rules also work. We have a proof that our results for buyer-side incentive compatibility extend to a closed-loop demand model where users wait until their previous packet was forwarded before generating the next packet, in the case where sellers continue to follow the intended routing policy. On the sell-side, we retain the alignment between social welfare and seller revenue obtained through revenue pooling in moving to a closed loop model. Although a seller with a good model of the future could benefit by deviating from the highest-bid-first routing policy, any benefit also accrues to market efficiency. Moreover, we conjecture that such a deviation would also retain monotonicity and thus buy-side incentives.

VI. Conclusions

We have outlined an approach to prioritized bandwidth access in a dynamic environment. We argued that counterfactuals are often unavailable in this setting, making it a natural

domain for applying the sampling methods from BKS [2], which had previously been primarily applied to multi-armed bandit settings. The approach aligns incentives for a simple model of fixed per-packet value and known arrival-departure intervals, and suggests a general protocol that allows periodic bid revision. Through random bid perturbations, a correct estimate of the externality imposed by one user on the rest of the system can be made, addressing a systematic over-payment problem with myopic packet-level auctions. A revenue pooling method addresses incentive problems on the sell side. In future work, we will complete a comprehensive simulation study of the robustness and efficiency properties of the scheme in different competitive environments and for different demand models, and consider competition between sellers.

References

[1] "Quantifying the mobile data tsunami and its implications," http://blog.nielsen.com/nielsenwire/online_mobile/quantifying-the-mobile-data-tsunami-and-its-implications/.

[2] M. Babaioff, R. D. Kleinberg, and A. Slivkins, "Truthful mechanisms with implicit payment computation," in *Proceedings of the 11th ACM conference on Electronic commerce*, ser. EC '10. New York, NY, USA: ACM, 2010, pp. 43–52. [Online]. Available: http://doi.acm.org/10.1145/1807342.1807349

[3] M. Babaioff, Y. Sharma, and A. Slivkins, "Characterizing truthful multi-armed bandit mechanisms: extended abstract," in *Proceedings of the 10th ACM conference on Electronic commerce*, ser. EC '09. New York, NY, USA: ACM, 2009, pp. 79–88. [Online]. Available: http://doi.acm.org/10.1145/1566374.1566386

[4] N. R. Devanur and S. M. Kakade, "The price of truthfulness for pay-per-click auctions," in *Proceedings of the 10th ACM conference on Electronic commerce*, ser. EC '09. New York, NY, USA: ACM, 2009, pp. 99–106. [Online]. Available: http://doi.acm.org/10.1145/1566374.1566388

[5] R. Cavallo, D. C. Parkes, and S. Singh, "Optimal coordinated planning amongst self-interested agents with private state," in *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI'06)*, Cambridge, MA, 2006, pp. 55–62.

[6] H. Varian and J. Mackie-Mason, "Pricing the internet," *Public Access to the Internet*, 1994.

[7] M. Hajiaghayi, R. Kleinberg, M. Mahdian, and D. Parkes, "Online auctions with re-usable goods," in *Proc. 6th ACM Conf. on Electronic Commerce (EC'05)*, 2005, pp. 165–174.

[8] D. C. Parkes and S. Singh, "An MDP-Based approach to Online Mechanism Design," in *Proc. 17th Annual Conf. on Neural Information Processing Systems (NIPS'03)*, 2003.

[9] A. Lazar and N. Semret, "The progressive second price auction mechanism for network resource sharing," *8th International Symposium on Dynamic Games, Maastricht, The Netherlands*, 1998.

[10] P. Maillé *et al.*, "The progressive second price mechanism in a stochastic environment," *Publication interne-IRISA*, Jan 2003. [Online]. Available: http://cat.inist.fr/?aModele=afficheN&cpsidt=14858910

[11] J. Lee and B. Szymanski, "An analysis and simulation of a novel auction-based pricing mechanism for network services," RPI, Tech. Rep., 2005.

[12] A. V. Goldberg and J. D. Hartline, "Competitive auctions for multiple digital goods," in *In Proc. 9th European Symposium on Algorithms*, 2001, pp. 416–427.

[13] R. B. Myerson, "Optimal auction design," *Mathematics of Operations Research*, vol. 6, pp. 58–73, 1981.