Simple Mechanisms for Welfare Maximization in Rich Advertising Auctions

Gagan Aggarwal
Google Research
gagana@google.com

Kshipra Bhawalkar
Google Research
kshipra@google.com

Aranyak Mehta
Google Research
aranyak@google.com

Divyarthi Mohan
Tel Aviv University
divyarthim@tau.ac.il

Alexandros Psomas
Purdue University
apsomas@cs.purdue.edu

Abstract

Internet ad auctions have evolved from a few lines of text to richer informational layouts that include images, sitelinks, videos, etc. Ads in these new formats occupy varying amounts of space, and an advertiser can provide multiple formats, only one of which can be shown. The seller is now faced with a multi-parameter mechanism design problem. Computing an efficient allocation is computationally intractable, and therefore the standard Vickrey-Clarke-Groves (VCG) auction, while truthful and welfare-optimal, is impractical.

In this paper, we tackle a fundamental problem in the design of modern ad auctions. We adopt a "Myersonian" approach and study allocation rules that are monotone both in the bid and set of rich ads. We show that such rules can be paired with a payment function to give a truthful auction. Our main technical challenge is designing a monotone rule that yields a good approximation to the optimal welfare. Monotonicity doesn't hold for standard algorithms, e.g. the incremental bang-per-buck order, that give good approximations to "knapsack-like" problems such as ours. In fact, we show that no deterministic monotone rule can approximate the optimal welfare within a factor better than 2 (while there is a non-monotone FPTAS). Our main result is a new, simple, greedy and monotone allocation rule that guarantees a 3 approximation. In ad auctions in practice, monotone allocation rules are often paired with the so-called *Generalized Second Price (GSP)* payment rule, which charges the minimum threshold price below which the allocation changes. We prove that, even though our monotone allocation rule paired with GSP is not truthful, its Price of Anarchy (PoA) is bounded. Under standard no-overbidding assumptions, we prove bounds on the a pure and Bayes-Nash PoA. Finally, we experimentally test our algorithms on real-world data.

1 Introduction

Internet Ad Auctions, in addition to being influential in advancing auction theory and mechanism design, are a half-a-trillion dollar industry [CF21]. A significant advertising channel is sponsored search advertising: ads that are shown along with search results when you type a query in a search box. These ads traditionally were a few lines of text and a link, leading to the standard abstraction for ad auctions: multiple items for sale to a set of unit-demand bidders, where each bidder i has a private value $v_i \cdot \alpha_{is}$ for the ad in position s, which has click-through rate α_{is}. However, when using your favorite search engine, you might instead encounter sitelinks/extensions leading to parts of the advertisers' website, seller ratings indicating how other users rate this advertiser or offers for

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

specific products. Advertisers can often explicitly opt in or out of showing different extensions with their ads. In fact, some extensions require the advertiser to provide additional assets, e.g. sitelinks, phone numbers, prices, promotion etc, and an ad cannot be shown unless this additional information is available. All these extensions/decorations change the amount of *space* the ad occupies, as well as affect the probability of a user clicking on the ad. The new and unexplored abstraction for modern ad auctions is now to select a set of ads that fit within the given total space.

In this paper, we study the problem of designing a *truthful* auction to determine the best set of ads that can be shown, with the goal of maximizing the social welfare. More formally, we consider the *Rich Ads* problem. In our model, each advertiser specifies a value per click v_i and set of rich ads. Each ad has an associated probability of click α_{ij} and a space w_{ij} that it would occupy if shown. The space and click probabilities are publicly known. Crucially, advertisers' private information is *not* single-dimensional. In addition to misreporting her value, there is another strategic dimension available: an advertiser can report only a subset of the set of ads available if the allocation under this report improves her utility. The open problem we address in this paper is whether there exist simple, approximately optimal and truthful mechanisms for Rich Ads.

Results and Techniques. The classic Vickrey-Clarke-Groves (VCG) mechanism is truthful and maximizes welfare for our setting, but it is computationally intractable: maximizing welfare is NP-complete (even without truthfulness) since our problem generalizes the KNAPSACK problem. It is also well known that coupling an approximation algorithm for welfare with VCG payments does not result in a truthful mechanism. And, maximal-in-range mechanisms [NR01], that optimize social welfare over a restricted domain, even though are one way around such situations, have limited use, since the range of possible outcomes (allocations) has to be committed to before seeing the bidders preferences (i.e., needs to be independent of bidders reports). For single parameter problems, Myerson's lemma [Mye81] can be used to obtain a truthful mechanism, as long as the allocation rule is monotone. The Rich Ads problem is not a single parameter problem, so this approach does not immediately work. However, similar to the inter-dimensional (or "one-and-a-half" dimensional) regime [FGKK16, DHP20, DGS$^+$20], we can extend Myerson's lemma to our domain. We show that an allocation rule that is monotone in the bid and the set of rich ads[1] can be paired with a payment rule to obtain a truthful mechanism.

Incentive issues aside, the Rich Ads problem is an extension of the KNAPSACK problem, called MULTI-CHOICE KNAPSACK: in addition to the knapsack constraint, we also have constraints that allow to allocate (at most) one rich ad per advertiser. As an algorithmic problem, this is well studied [SZ79, Law79]. The optimal fractional allocation can be derived using a simple *greedy* algorithm using the *incremental bang-per-buck* order. However, it turns out that the optimal (integral or otherwise) allocation, as well as other natural allocations, are not monotone. In fact, as we show, no deterministic (resp. randomized) monotone allocation rule can obtain more than half (resp. $11/12$ fraction) of the social welfare. In contrast, without the monotonicity constraint, there is an FPTAS for the MULTI-CHOICE KNAPSACK problem [Law79].

Our *main result* is providing an integral allocation rule that is monotone and obtains at least a third of the optimal (fractional) social welfare. Pairing with an appropriate payment function we get the following (informal) theorem.

Informal Theorem. *There exists a simple truthful mechanism, that can be computed in polynomial time, which obtains a* 3*-approximation to the optimal social welfare.*

To obtain this result, we first find an allocation of space amongst the advertisers. In contrast to the optimal fractional algorithm described above which allocates greedily using the incremental bang-per-buck order, our algorithm allocates greedily using an *absolute bang-per-buck* order. Crucially, the *space* allocated to each advertiser in this way is monotone, even though the expected number of clicks (i.e. the utility) of the bang-per-buck algorithm itself is *not* monotone. By post-processing to utilize this space optimally for each advertiser, we obtain an integral allocation that is monotone in the expected number of clicks. We prove that this allocation gives a two approximation to the optimal fractional welfare, minus the largest value ad. Finally, by randomizing between this integral allocation (with probability $2/3$) and the largest value ad (with probability $1/3$), we get a 3-approximation

[1] Here, an allocation rule is defined to be monotone in the set of rich ads if the expected clicks allocated to an advertiser can only increase when the advertiser reports a superset of rich ads.

to the optimal social welfare. Since the overall allocation rule is monotone, we can pair it with an appropriate pricing to get a truthful mechanism.

We proceed to further explore the merits of our monotone allocation rule by pairing it with the *Generalized Second Price (GSP)* payment rule, which charges each advertiser the minimum threshold (on their bid) below which their allocation changes. The overall auction is not truthful. However, we can analyze its performance by bounding its social welfare in a worst-case equilibrium. In particular, we consider the full information pure Nash equilibrium, where bidders best-respond to a profile of competitors bids, as well as the incomplete information Bayes-Nash equilibrium, where the bidders best-respond to a distribution of valuation draws and bids for the competitors. The corresponding pure Price of Anarchy (PoA) and Bayes-Nash Price of Anarchy are the ratios of the optimal social welfare to the welfare of the worst equilibrium. In either setting, we make the standard no-overbidding assumption, where bidders do not bid more than their value. This assumption is required as without it the PoA of even the single-item second price auction (which is truthful) can be unbounded.[2]

Informal Theorem. *There exists a simple mechanism with a monotone allocation rule, paired with the GSP payment rule, which under the no-overbidding assumption guarantees a pure Price of Anarchy (resp. Bayes-Nash PoA) of at most* 6 *(resp.* $\frac{6}{1-1/e}$*).*

We prove our PoA bounds by identifying a suitable deviation for each advertiser, and bounding the advertiser's utility in this deviation relative to the social welfare of the optimal integral allocation, our integral bang-per-buck allocation (in the equilibrium), and the largest value ad (in the equilibrium); as opposed to single-dimensional PoA bounds, the knapsack constraint in our setting introduces a number of technical obstacles we need to bypass. To prove a bound for the Bayes-Nash PoA, we combine techniques from our pure PoA bound with the standard smoothness framework [Rou15a]. In particular, the smoothness part of our argument is very similar to that of [CKK+15] for the Bayes-Nash PoA of the standard GSP position auction. Due to the specific form of the smoothness framework that we use, our bound also applies to mixed, correlated, coarse-correlated and Bayes-Nash with correlated valuations. Our PoA results can be found in Appendix E.

Finally, we provide an empirical evaluation of our mechanism on real world data from a large search engine. We compare performance of our mechanism with VCG and the fractional-optimal allocation that doesn't account for incentives. Our empirical results show that our allocation rule obtains at least 0.4 fraction of the optimal in the worst-case. However, there are many instances where our allocation rule is almost as good as VCG. In fact, the average approximation factor of our allocation rule is 0.97. Furthermore, our mechanisms are significantly faster than VCG, even with the Myersonian payment computation. We also empirically evaluate heuristic extensions of our algorithms when there is a bound on the total number of distinct rich ads shown.

Related work. Traditional sponsored search auctions have been studied extensively [AGM06, Var07, EOS07]. A number of recent works relax the traditional model of sponsored search auctions [CMSW20, Hum16] and introduce different versions of the "rich ads" problem [DSYZ10, CKSW17, GHLY19, HIK+18]; the specific model we study in this paper is new. [DSYZ10] are the first to formulate a rich ad problem where ads can occupy multiple slots. They analyze VCG and GSP variants for a special version of the rich ad problem where ads can be of only one of two possible sizes. They leave the problem addressed in this paper as an open problem for future work.

Much of the literature focuses on GSP-like rules (e.g., because the cost of switching from existing GSP to VCG can be high [VH14]). [CKSW17] consider the more general rich ad problem where there are constraints on number of ads shown and position effects in the click through rate. But their setting is still single-parameter — advertisers report a bid per click and cannot mis-report the set of rich ads. They provide a local search algorithm that runs within polynomial time and a generalized GSP like pricing to go with it. However, as opposed to our interest here, their auction is not truthful, nor do they give any approximation guarantees. [GHLY19] consider the optimization problem when the probability of click is submodular or subadditive in the size of the rich ad. They give an LP rounding based algorithm that provides a 4 approximation for submodular and a $\Omega(\frac{\log m}{\log \log m})$ approximation for subadditive, with respect to the social welfare assuming truthful bidding. They however do not provide a truthful payment rule, or any PoA guarantees. These works also focus on a single-dimensional setting (where the advertiser is strategic about its bid but the set of ads is publicly

[2] Consider, for example, the equilibrium where all bidders bid 0, except the lowest bidder, who bids infinity.

known). In contrast, we consider a multi-dimensional setting. Our simple and truthful mechanism also has a *monotone* allocation function, so we pair it with GSP as well.

The PoA of the GSP auction for text ads was studied in [CKKK11, LB10, LPL11]. Our PoA bounds use the smoothness framework introduced in [Rou15a], and later extended by [CKK+15] to show PoA bounds for GSP (as well as in [Rou15b] and [ST13] for more general use).

2 Preliminaries

Rich Ad Model. We introduce the following model for the rich ads auction problem. There is a set $\mathcal{N}$ of n advertisers and a universe of rich ads $\mathcal{S}$. Each advertiser $i \in \mathcal{N}$ has a private value per click v_i and a private set of rich ads $A_i \subseteq \mathcal{S}$.[3] We use $\mathbf{v} = (v_1, v_2, \ldots, v_n)$ to denote the vector of values per click and $\mathbf{A} = (A_1, A_2, \ldots, A_n)$ to denote the vector of sets of rich ads. For every advertiser $i \in \mathcal{N}$, each rich ad $j \in \mathcal{S}$ has a publicly known space w_{ij} and a publicly known probability of click α_{ij}.[4] We use v_{ij} for the value of rich ad j for advertiser i. If $j \notin A_i$, then the value of advertiser i is $v_{ij} = 0$; otherwise, $v_{ij} = \alpha_{ij} v_i$. An advertiser can be allocated only one of the ads from the set $\mathcal{S}$. Finally, there is a total limit W on the total space occupied by the ads. We assume without loss of generality that for each i and each j, $w_{ij} \leq W$, as any ad that is larger than W cannot be allocated integrally in space W. A (randomized or fractional) allocation $\mathbf{x} \in [0,1]^{n \times |S|}$ indicates the probability x_{ij} that ad j is allocated to advertiser i. An allocation is feasible if each advertiser gets at most one ad, i.e. $\sum_{j \in \mathcal{S}} x_{ij} \leq 1$ for all $i \in \mathcal{N}$, and the total space used is at most W, i.e. $\sum_{i \in \mathcal{N}, j \in \mathcal{S}} x_{ij} w_{ij} \leq W$. An allocation is integral if $x_{ij} \in \{0,1\}$, for all $i \in \mathcal{N}$ and $j \in \mathcal{S}$.

Our goal is to maximize social welfare. For an allocation $\mathbf{x} = Alg(\mathbf{v}, \mathbf{A})$, $SW(Alg(\mathbf{v}, \mathbf{A})) = \sum_{i,j} x_{i,j} v_{ij}$. We can write an integer program for the optimal allocation as follows, by introducing a binary variable $x_{ij} \in \{0,1\}$ for the allocation of advertiser $i \in \mathcal{N}$ and rich ad $j \in \mathcal{S}$. The objective is to maximize welfare $\sum_{ij} x_{ij} v_{ij}$, subject to a Knapsack constraint $\sum_i \sum_j w_{ij} x_{ij} \leq W$, and feasibility, i.e. $\sum_j x_{ij} \leq 1$ for all $i \in \mathcal{N}$ (expressing that each advertiser can get only one ad).

Mechanism Design Considerations. By standard revelation principle arguments, it suffices to focus on direct revelation mechanisms. Each advertiser $i \in \mathcal{N}$ reports a bid b_i and a set of rich ads $S_i \subseteq \mathcal{S}$. Similarly to many works in the inter-dimensional regime, e.g. [MV09, DW17], we assume that $S_i \subseteq A_i$, that is, an advertiser cannot report that they want an ad they don't have. Let $\mathbf{b} = (b_1, b_2, \ldots, b_n)$ to denote the vector of bids and $\mathbf{S} = (S_1, S_2, \ldots, S_n)$ to denote the vector of sets of rich ads. We use $b_{ij} = b_i \cdot \alpha_{ij}$ if $j \in S_i$ and $b_{ij} = 0$ otherwise, and refer to the rich ad using a (reported value, space) tuple (b_{ij}, w_{ij}). A mechanism selects a set of ads to show, of total space at most W, and charges a payment to each advertiser. Let $x_{ij}(\mathbf{b}, \mathbf{S})$ be the probability that ad j is allocated to advertiser i, and $p_i(\mathbf{b}, \mathbf{S})$ denote the expected payment of advertiser i. Let $\mathbf{x}_i(\mathbf{b}, \mathbf{S})$ be the allocation vector of advertiser i. We assume that for any valid allocation rule for $j \notin S_i$ $x_{ij}(\mathbf{b}, \mathbf{S}) = 0$. We slightly overload notation, and use $x_i(\mathbf{b}, \mathbf{S})$ to denote the expected number of clicks the advertiser will get; that is, $x_i(\mathbf{b}, \mathbf{S}) = \sum_{j \in S_i} x_{ij}(\mathbf{b}, \mathbf{S}) \alpha_{ij}$. If required we refer to the cost per-click $cpc_i(\mathbf{b}, \mathbf{S}) = p_i(\mathbf{b}, \mathbf{S}) / x_i(\mathbf{b}, \mathbf{S})$

Advertisers have quasi-linear utilities. An advertiser with value v_i and set A_i has utility $v_i x_i(\mathbf{b}, \mathbf{S}) - p_i(\mathbf{b}, \mathbf{S})$, when reports are according to $\mathbf{b}$ and $\mathbf{S}$. Let $u_i(v_i, A_i \to b_i, S_i; \mathbf{b}_{-i}, \mathbf{S}_{-i})$ be the utility of advertiser i when her true value and set of ads are v_i, A_i, but reports b_i, S_i, and everyone else reports according to $\mathbf{b}_{-i}, \mathbf{S}_{-i}$. For ease of notation we often drop $\mathbf{b}_{-i}, \mathbf{S}_{-i}$ when it's clear from the context. When the profile of true types is fixed, we drop (v_i, A_i) and use the notation $u_i(b_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})$.

A mechanism is *truthful* if no advertiser has an incentive to lie, i.e. for any all $\mathbf{b}_{-i}, \mathbf{S}_{-i}$, $u_i(v_i, A_i \to v_i, A_i; \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq u_i(v_i, A_i \to b_i, S_i; \mathbf{b}_{-i}, \mathbf{S}_{-i})$, for all v_i, A_i, b_i, S_i, A mechanism is individually rational if in all of its outcomes, all agents have non-negative utility.

We are interested in auctions that are computationally tractable, truthful, individually rational, with the goal of maximizing the social welfare $SW(\mathbf{x}(\mathbf{b}, \mathbf{S})) = \sum_i v_i x_i(\mathbf{b}, \mathbf{S})$. Even ignoring truthfulness

[3] We expect the rich ads to be tailored to an advertiser, so we assume that $A_i \cap A_{i'} = \emptyset$, for all $i, i' \in \mathcal{N}$.

[4] Note that this safe to assume. The space consumed by a rich ad is evident when the rich ad is provided. The probability of click can be predicted by the platform (e.g. using machine learning models).

and individual rationality, the computational constraints rule out achieving the optimal social welfare; therefore, we seek approximately optimal mechanisms.

Definition 1. *A truthful mechanism $\mathcal{M}$ obtains an α factor approximation to the social welfare if $SW(\mathbf{x}_{\mathcal{M}}(\mathbf{v}, \mathbf{A})) \geq SW(\mathbf{x}_{OPT}(\mathbf{v}, \mathbf{A}))/\alpha$.*

Formal definitions of the Generalized Second Price and the Price of Anarchy are deferred to Appendix A (since all the instances where they are used are also in the appendix).

Optimal fractional allocations. The algorithmic problem is a special case of a well-known variation of the KNAPSACK problem, called MULTI-CHOICE KNAPSACK [SZ79]. The integer program for MULTI-CHOICE KNAPSACK is the same as the integer program above, except that the inequality constraint $\sum_j x_{ij} \leq 1$ is replaced with an equality. Our problem is easily reduced to the MULTI-CHOICE KNAPSACK problem by introducing a *null* ad with $(\alpha_{i0}, w_{i0}) = (0, 0)$.[SZ79] provide a characterization of the optimal fractional solution of MULTI-CHOICE KNAPSACK and provide a fast algorithm to compute the fractional optimal solution. As is colloquial in the KNAPSACK literature, we refer to the ratio $\frac{b_{ik}}{w_{ik}}$ as *Bang-per-Buck* and the ratio $\frac{b_{ij}-b_{ik}}{w_{ij}-w_{ik}}$ with $w_{ij} > w_{ik}$ as *Incremental Bang-per-Buck*. [SZ79] show that allocating ads in the incremental bang-per-buck order gives the optimal fractional solution. We state this standard algorithm in Appendix A. We refer to the solution constructed by this algorithm as OPT and use it as a benchmark in our approximation guarantees. We note a few more properties of OPT.

Fact 1 ([SZ79])**.** *In the optimal fractional allocation constructed by the algorithm, all advertisers except one have a rich-ad allocated integrally. Also for any advertiser i allocated space W_i^* in OPT, the allocation maximizes the value that advertiser i can obtain in that space.*

This fact also implies a 2-approximate integral allocation as follows. Construct an optimal fractional solution using the incremental bang-per-buck order. Let i' denote the advertiser that is allocated last: select the larger of the optimal fractional solution without i' and the highest value ad of i'.

3 Monotonicity and Lower Bounds

Monotonicity implies truthfulness. In single-parameter domains, Myerson's lemma provides a handy tool for constructing truthful mechanisms. One has to only construct a monotone allocation rule, and then the lemma provides a complementary payment rule such that the overall mechanism is truthful. We extend this approach to our particular multi-parameter domain. If the set of ads is fixed for each advertiser, then monotonicity in bid and Myerson-like payments imply truthfulness. We give constraints between the allocation rules for different sets of ads and show that they imply truthfulness everywhere using a local-to-global argument. We begin by defining monotonicity in our setting. An allocation rule is said to be monotone if it is monotone in each dimension of the buyer's preferences.

Definition 2. *An allocation rule $x(\mathbf{b}, \mathbf{S})$ is monotone in b_i, S_i for each i, if (1) For all $\mathbf{b}_{-i}, \mathbf{S}_{-i}, S_i, b_i' \geq b_i$; we have $x_i(b_i', S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq x_i(b_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})$, and (2) For all $\mathbf{b}_{-i}, \mathbf{S}_{-i}, b_i, S_i' \supseteq S_i$; we have $x_i(b_i, S_i', \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq x_i(b_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})$.*

As the following example shows, the optimal allocation rule is *not* monotone. The example also shows that monotonicity is not necessary for truthfulness (since VCG is truthful and optimal).

Example 1. *Consider two advertisers with two rich ads each. Both have value 1 and the rich ads have (value, size) $= (1, 1), (1 + \epsilon, 2)$. The space available is 3. In this case, the optimal integer solution chooses the smaller ad from one advertiser and the larger ad from other. However, if one of them removes their smaller option, they get the larger option deterministically.*

The next lemma (proof in Appendix B) shows that monotonicity is sufficient for truthfulness.

Lemma 1. *If a valid allocation rule $\mathbf{x}(\mathbf{b}, \mathbf{S})$ is monotone in b_i, S_i for each $i \in \mathcal{N}$, then charging payment $p_i(\mathbf{b}, \mathbf{S}) = b_i x_i(\mathbf{b}, \mathbf{S}) - \int_0^{b_i} x_i(b, \mathbf{b}_{-i}, \mathbf{S}) db$ results in a truthful auction.*

Lower bounds. Next, we illustrate the challenge in coming up with allocation rules which are monotone in the set of rich ads. We also prove that monotonicity rules out approximation ratios strictly better than 2 for deterministic mechanisms (and $12/11$ for randomized mechanisms).

As we've seen in Example 1, the algorithm that finds the optimal integer allocation is not monotone. The following example shows that simple algorithms such as selecting ads in the incremental bang-per-buck order, or the 2 approximation algorithm presented in Section 2 are not monotone either. Recall that bang-per-buck = b_{ij}/w_{ij} and incremental bang-per-buck = $\frac{b_{ij}-b_{ik}}{w_{ij}-w_{ik}}$ with $w_{ij} > w_{ik}$.

Example 2. *We have two advertisers A and B. A has two rich ads with (value, size) as (2,1),(3.5,3). B has one ad with (value, size)=(3,3).*

(i) Suppose $W = 4$. The incremental bang-per-buck algorithm picks $A : (2, 1)$, followed by $B : (3, 3)$. The rich ad $A : (3.5, 3)$ does not get picked over $B : (3, 3)$, despite having higher bang-per-buck, since it has smaller incremental bang-per-buck (of 0.75). On the other hand if A removes the rich ad $(2, 1)$, the algorithm picks $A : (3.5, 3)$ at the very beginning, and A obtains a higher value.

(ii) Suppose $W = 3.5$, the optimal fraction solution is $A : (2, 1)$ with a weight of 1.0 and $B : (3, 3)$ with weight $2.5/3$. The 2-approximation algorithm of Section 2 compares allocating just $A : (2, 1)$ or just $B : (3, 3)$, and chooses B. But if A removed $(2, 1)$, then the optimal fraction solution is $A : (3.5, 3)$ with a weight of 1.0 and $B : (3, 3)$ with weight $0.5/3$. The 2-approximation algorithm compares $A : (3.5, 3)$ and $B : (3, 3)$, and chooses A because it has higher value. Thus A gets a higher value by removing $(2, 1)$.[5]

The next theorem gives lower bounds on the approximation factor a monotone algorithm can achieve.

Theorem 1. *No monotone and deterministic (resp. randomized) algorithm has an approximation ratio better than $2 - \varepsilon$ (resp. $12/11 - \varepsilon$) for any $\varepsilon > 0$.*

Proof. There are two advertisers. Each advertiser has two rich ads: $(1, 1), (1 + \varepsilon, 2)$. The total space is 3. The optimal solution has value $(2 + \epsilon)$. To obtain an approximation better than $2 - \epsilon$, we must choose a small ad for at least one of the advertisers. Since the algorithm is monotone, when that advertiser does not provide the small ad, the algorithm cannot give them the larger ad. Thus when the advertiser does not provide the small ad, the algorithm must give this advertiser nothing, resulting in welfare at most $(1 + \varepsilon)$. See Appendix B for the proof for randomized algorithms. □

4 A Simple Monotone 3-Approximation

In this section we give our main result: a monotone algorithm that obtains a 3 approximation to the optimal social welfare. First, we give a fractional algorithm for allocating *space* to each advertiser i. Second, we show that optimally (and integrally) using the space given to each advertiser i gives a monotone allocation. Finally, we show that randomizing between the former algorithm and simply allocating the max value ad is a monotone rule that obtains a 3 approximation to OPT. Omitted proofs (and definitions) can be found in Appendix C. We show that the truthful payment function matching our allocation rule can be computed in polynomial time in Appendix C.3.

Monotone space allocation algorithm We start by giving a monotone algorithm for allocating space to each advertiser. This total space allocated is monotone in b_i and S_i, for all $i \in \mathcal{N}$. Our algorithm also provides an allocation of rich-ads to that space, but this allocation by itself may not be monotone, and may not provide a good approximation. Our algorithm, which we call ALG_B, works as follows (see Appendix C for a formal description). First, we order the ads in the bang-per-buck order. We iteratively choose the next ad in this order; let i be the corresponding advertiser, and j be the rich ad. We replace the previous ad of i with j, if this choice results in more space allocated to i. If there is not enough space we fractionally allocate j and terminate.

The following observation shows that any ads removed for not being-selected will not be used by the fractional-optimal solution as well. See Appendix A for the precise definition of dominated.

Observation 1. *Let $j' \in E_i$ be some ad removed from E_i in "step 2" of ALG_B for having space at most w_{ij}. Then either $j' = j$ or j' is a "dominated" ad.*

Next, we prove that ALG_B allocates space that is monotone in b_i and S_i, for all $i \in \mathcal{N}$.

[5] Even though we only seek integer monotone algorithms, this example shows that even the fractional allocation is not monotone: when A provides all ads its value is 2, but when A removes $(2, 1)$, its value is 3.5.

Theorem 2. *Let $\mathbf{x}(ALG_B)$ denote the allocation of ALG_B. Then, for all $i \in \mathcal{N}$, $b_i \leq b'_i$: $\sum_j w_{ij}x_{ij}(ALG_B(\mathbf{b}, \mathbf{S})) \leq \sum_j w_{ij}x_{ij}(ALG_B(b'_i, \mathbf{b}_{-i}, \mathbf{S}))$. Also, for all i and $S_i \subseteq S'_i$, the space W_i is monotone in S_i: $\sum_j w_{ij}x_{ij}(ALG_B(\mathbf{b}, \mathbf{S})) \leq \sum_j w_{ij}x_{ij}(ALG_B(\mathbf{b}, S_i, \mathbf{S}_{-i}))$.*

ALG_B is inefficient since the bang-per-buck allocation can change the relative order in which the advertisers are assigned space, generating sub-optimal outcomes.

Example 3. *Let M be a large integer. Let $W = M - 1$ and consider two advertiser A and B. A has two rich ads with (value, size) as $(1, 1), (1 + \varepsilon, M - 1)$. B has one rich ad with (value, size) = $(\frac{M-1}{M}, M - 1)$. Fractional OPT selects $A : (1, 1)$ fully and $B : (\frac{M-1}{M}, M - 1)$ fractionally with weight $(M - 2)/(M - 1)$ and obtains a social welfare $1 + \frac{M-2}{M}$. The bang-per-buck allocation in ALG_B selects $A : (1 + \varepsilon, M)$ fully and obtains social welfare $(1 + \varepsilon)$.*

While the space allocated is monotone in the bid and rich ads, the allocation itself may not be monotone, since ALG_B may allocate an advertiser a larger ad with lower value than another option; we provide an example in Appendix D.

Integral Monotone Allocation. The allocation generated by ALG_B can be fractional for one advertiser, and can be sub-optimal (but integer) for some of the other advertisers. In the following algorithm, we post-process to find the best *single ad* that fits in W_i.

Algorithm 1 (ALG_I)**.** *First, run ALG_B. Let W_i be the space allotted to advertiser i. Second, post-process to allocate the ad j with maximum value that fits in W_i, i.e. $j \in \operatorname{argmax}_{w_{ij} \leq W_i} b_{ij}$. Any remaining space is left unallocated.*

Observe that ALG_I is monotone, since (1) the space allocated by ALG_B is monotone, and (2) if the space allocated by ALG_B is larger, then the post-processing that allocates the highest value ad that fits in this space will also result in same or larger value. However, ALG_I alone might be an arbitrarily bad approximation: we provide an example in Appendix D.

Main result. Our main result is the following theorem.

Theorem 3. *The randomized algorithm that runs ALG_I with probability $2/3$, and otherwise allocates the maximum valued ad, is monotone in b_i and S_i, obtains a 3-approximation to the social welfare, and this approximation factor is tight.*

Proof. Let b_{max} denote the value of the maximum valued ad. We will first show that $2SW(\mathbf{x}(ALG_I)) + b_{max} \geq SW(\mathbf{x}_{OPT})$. Let $\text{Val}(\mathbf{x}, A, \vec{s}) = \sum_{i \in A}(\sum_{j \in \mathcal{S}} b_{ij} \cdot x_{ij}) \cdot \left(\frac{s_i}{\sum_{j \in \mathcal{S}} w_{ij}x_{ij}}\right)$ be the fraction of the social welfare of allocation $\mathbf{x}$, $SW(\mathbf{x})$, contributed by a subset of advertisers A for space $\vec{s}$. Let $\mathbf{x}^* = \mathbf{x}_{OPT(\mathbf{b},\mathbf{S})}$ denote an optimal fractional allocation. Let $W_i = W_i(ALG_B(\mathbf{b}, \mathbf{S}))$ and $W_i^* = W_i(OPT(\mathbf{b}, \mathbf{S}))$ denote the total space allocated to advertiser i in ALG_B and the optimal allocation $\mathbf{x}^*$, respectively. The space allocated in ALG_I is exactly the same as W_i. Recall Fact 1, that there is an optimal fractional allocation where at most one advertiser is allocated fractionally. Let i' be the advertiser whose allocation in $\mathbf{x}^*$ is fractional. There is also at most one advertiser in ALG_B whose allocation is fractional: the advertiser corresponding to the very last ad that is included; let i'' be this advertiser. We start by giving a series of technical claims. First, we bound the part of $SW(\mathbf{x}(OPT))$ contributed by advertisers who are allocated more space in ALG_B than in OPT. Let $\mathcal{I}$ denote the set of advertisers i with $W_i \geq W_i^*$.

Claim 1. $\text{Val}(\mathbf{x}^*, \mathcal{I} \setminus \{i'\}, \vec{W}^*) \leq \text{Val}(\mathbf{x}(ALG_I), \mathcal{I} \setminus \{i'\}, \vec{W})$.

Let $\mathcal{K} = \mathcal{N} \setminus \mathcal{I}$ be the set of advertisers k with $W_k < W_k^*$. If $\mathcal{K} = \emptyset$ then $W_i = W_i^*$ for all i. We note that $b_i \cdot x_i(ALG_I) = b_i \cdot \mathbf{x}_i^*$ for all $i \neq i'$ by Claim 1. Also $b_{max} \geq b_i \cdot \mathbf{x}_{i'}^*$. Thus we get $SW(\mathbf{x}(ALG_I)) + b_{max} \geq SW(OPT)$. So for the rest of the proof we assume $\mathcal{K} \neq \emptyset$. We bound the portions of $SW(\mathbf{x}_{OPT})$ contributed by $k \in \mathcal{K}$ using the following claim.

Claim 2. *For all $k \in \mathcal{K} \setminus \{i'\}$ and $i \in \mathcal{N}$, we have $\frac{b_k \cdot \mathbf{x}_k^*}{W_k^*} \leq \frac{b_i \cdot \mathbf{x}_i(ALG_B)}{W_i}$.*

That is, advertisers that are allocated less space in ALG_I than in $\mathbf{x}^*$, must have lower bang-per-buck as otherwise their rich ad would be considered by ALG_I and they will be allocated more space.

Finally we bound the contribution of i', that is the ad allocated fractionally in OPT as follows:

Claim 3. *Let $(b_s, s), (b_\ell, \ell)$ with $s < l$ be the ads used in $\mathbf{x}^*_{i'}$, the optimal fractional allocation for i'. It holds that: (i) $\frac{b_\ell - b_s}{\ell - s} \leq \frac{b_i \cdot \mathbf{x}_i(ALG_B)}{W_i}$ for all $i \in \mathcal{N}$, (ii) if $s > W_{i'}$, then $\frac{b_s}{s} \leq \frac{b_i \cdot \mathbf{x}_i(ALG_B)}{W_i}$ for all $i \in \mathcal{N}$, and (iii) if $s \leq W_{i'}$ then $b_s \leq b_{i'} \cdot \mathbf{x}_{i'}(ALG_I)$.*

The proof of this claim is more involved. Intuitively we can bound the smaller of i''s allocated ads (if it is small enough) with the value i' obtains in ALG_I and the larger ad since it is allocated last has the lowest incremental bang-per-buck than any other advertiser's ad and hence the incremental bang-per-buck is also lower than the other ad's actual bang-per-buck.

We put all the claims together to bound $\text{Val}(\mathbf{x}^*, \mathcal{K} \cup \{i'\}, \vec{W}^*)$. If $s > W_{i'}$, then by Claims 2 and 3 we see that the bang-per-buck of $k \in \mathcal{K} \cup \{i'\}$ is less $b_i \cdot \mathbf{x}_i(ALG_B)/W_i$ for all i. Then the value for $\mathcal{K} \cup \{i'\}$ in OPT is less than the contribution of advertisers using the *same total space* in ALG_B .

$$\text{Val}(\mathbf{x}^*, \mathcal{K} \cup \{i'\}, \vec{W}^*) \leq \text{Val}(\mathbf{x}^*, \mathcal{K} \setminus \{i'\}, \vec{W}^*) + b_{i'} \cdot \mathbf{x}^*_{i'} \leq \text{Val}(\mathbf{x}(ALG_B), \mathcal{N}, \vec{W}) \quad (1)$$

If $s \leq W_{i'}$, then by Claims 2 and 3 we get that $b_s < b_{i'} \cdot \mathbf{x}_{i'}(ALG_I)$ and the bang-per-buck of the ads in ALG_B is higher than that of $K \setminus \{i'\}$ and $\frac{b_\ell - b_s}{\ell - s}$. Thus we have,

$$\begin{aligned}\text{Val}(\mathbf{x}^*, \mathcal{K} \cup \{i'\}, \vec{W}^*) &\leq \text{Val}(\mathbf{x}^*, \mathcal{K} \setminus \{i'\}, \vec{W}^*) + b_s + (W^*_{i'} - s)\frac{b_\ell - b_s}{\ell - s} \\ &\leq \text{Val}(\mathbf{x}(ALG_B), \mathcal{N}, \vec{W}) + b_{i'} \cdot \mathbf{x}_{i'}(ALG_I) \quad (2)\end{aligned}$$

Hence by putting both $\mathcal{I}$ and $\mathcal{K}$ together we get,

$$\begin{aligned}SW(\mathbf{x}_{OPT}) &= \text{Val}(\mathbf{x}^*, \mathcal{I} \setminus \{i'\}, \vec{W}^*) + \text{Val}(\mathbf{x}^*, \mathcal{K} \cup \{i'\}, \vec{W}^*) \\ &\leq \text{Val}(\mathbf{x}(ALG_I), \mathcal{I} \setminus \{i'\}, \vec{W}) + \text{Val}(\mathbf{x}(ALG_B), \mathcal{N}, \vec{W}) + b_{i'} \cdot \mathbf{x}_{i'}(ALG_I) \\ &\leq \text{Val}(\mathbf{x}(ALG_I), \mathcal{N}, \vec{W}) + \text{Val}(\mathbf{x}(ALG_B), \mathcal{N}, \vec{W}) \\ &\leq 2\text{Val}(\mathbf{x}(ALG_I), \mathcal{N}, \vec{W}) + b_{max}\end{aligned}$$

where the first equality is the definition of $SW(\mathbf{x}_{OPT})$, and the second inequality puts together Claim 1 and Equations (1), and (2). The third inequality holds because $\mathcal{I} \cup \{i'\} \subseteq \mathcal{N}$. The final inequality holds because $b_i \cdot \mathbf{x}_i(ALG_I) \geq b_i \cdot \mathbf{x}_i(ALG_B)$ for all $i \neq i''$, and $b_{i''} \cdot \mathbf{x}_{i''}(ALG_B) \leq b_{max}$.

Thus we have that $SW(\mathbf{x}_{OPT}) \leq 2\text{Val}(\mathbf{x}(ALG_I), \mathcal{N}, \vec{W}) + b_{\max}$. Therefore, running ALG_I with probability $2/3$ and allocating $b_{\max}$ with probability $1/3$ is a 3-approximation to $SW(\mathbf{x}_{OPT})$. Since our algorithm is randomizing between two monotone rules, it is monotone as well. We give an instance which shows that the approximation of the algorithm is at least 3 in Appendix C. □

5 Experiments

In this section we present some empirical results for our truthful mechanisms. The allocation rules we use for our theoretical results can be extended to obtain higher value. First we extend ALG_I to skip past a high bang-per-buck ad that does not fit in the remaining space. More precisely, recall that ALG_I calls ALG_B to get the space allocation. ALG_B stops when the ad being considered cannot be fit fully in the remaining space. The advertiser corresponding to this ad still gets allocated the remaining space, which is filled with the highest-value ad that fits in post-processing. In our modified version, we update ALG_B to skip past this large ad. This is equivalent to dropping step (3) of ALG_B (i.e., we keep going until we run out of ads), and running the rest of ALG_I as it is. We call this modified algorithm *GreedyByBangPerBuck*. For our theoretical result we also select the maximum value ad with probability $1/3$. In practice, this can be very inefficient. For our empirical evaluation, we extend this to continues to allocate as long as space is remaining. Similar to the GreedyByBangPerBuck, the algorithm skips past a high value but large ad that cannot fit, and continues allocating until the space or the set of rich ads runs out. We call this algorithm *GreedyByValue*. It is worth noting that these extensions do not improve the worst-case approximation ratio of Theorem 3: we include a brief proof in Appendix F. We implement our proposed randomized algorithm by flipping a coin with probability $2/3$ for each query and selecting the result of GreedyByBangPerBuck algorithm if it is heads and GreedyByValue otherwise. We call this *RandomizedGreedy*. As a baseline, we

Algorithm	ApproxECPM	time-msec
GreedyByBangPerBuck	0.9493	0.7363
GreedyByValue	0.9196	0.3242
RandomizedGreedy	0.9393 ± 0.0001	0.5983 ± 0.0007
IntOPT	1.0	6.9988

Table 1: Average performance of the algorithms compared to IntOPT. We report average approximation of eCPM relative to IntOPT and average running time in miliseconds. We report confidence intervals for the randomized algorithm by noting the average performance over all queries over 100 runs.

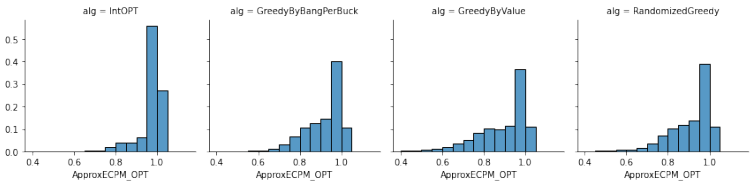

Figure 1: Histogram of approximation factor for IntOPT, GreedyByValue, GreedyByBangPerBuck and RandomizedGreedy compared to Fractional Optimal.

have implemented *IntOPT*, that uses brute-force-search to evaluate all possible allocations and compute the optimal integer allocation. This allocation paired with the VCG payment rule (which is computed by computing the integer OPT with advertiser i removed, and subtracting from that the allocation of all advertisers other than i in the optimal allocation) gives the VCG mechanism. Finally, we implement the incremental bang-per-buck order algorithm as *IncrementalBPB* to compute the fractionally optimal allocation.

We evaluated our algorithms from real world data obtained from a large search advertising company. The data consists of a sample of approximately 11000 queries, selected to have at least 6 advertisers each. All the space values for the rich-ads are integer. We use 500 as the space limit as that is larger than the space of any individual rich ad. Table 1 compares the average performance of these algorithms. Our algorithms are comparable to IntOPT, on average, but require a lot less time to run.

We first compare the approximation obtained by various algorithms to the fractional-optimal. In Figure 1, we see that GreedyByBangPerBuck and IntOPT obtain at least a 0.55 fraction of the fractional opt, while the approximation factor for GreedyByValue can be as low as 0.4. There is also a substantial amount of mass in the 1.0 bucket where integer-opt and fractional-opt coincide and the greedy algorithms also sometimes achieve that. Next we compare the approximation obtained by various algorithms to IntOPT.

In Figure 2 we see the approximation obtained by various algorithms compared to the IntOPT allocation. There are more queries where we obtain the optimal approximation, but the worst-case is still 0.6 for GreedyByBangPerBuck and 0.4 for GreedyByValue. For additional insight, we plot a heatmap to correlate the approximation factor obtained by GreedyByBangPerBuck and GreedyByValue with IntOPT as the baseline. In Figure 2, we compare the approximation factor of

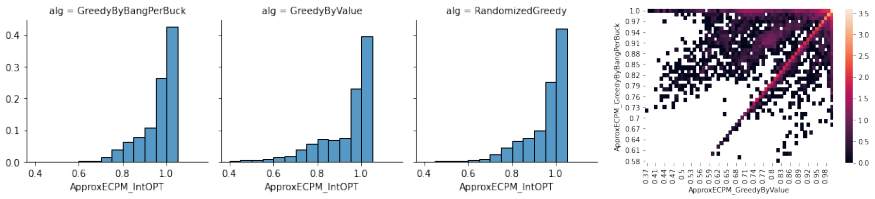

Figure 2: Histogram of approximation factor for GreedyByValue, GreedyByBangPerBuck and randomized Greedy compared to IntOPT. Height of each bucket represents the fraction of queries in the bucket. Last figure shows correlation of approximation factor relative to IntOPT for GreedyByValue, GreedyByBangPerBuck. Color-scale in the heat-map by log(number of queries) in bucket.

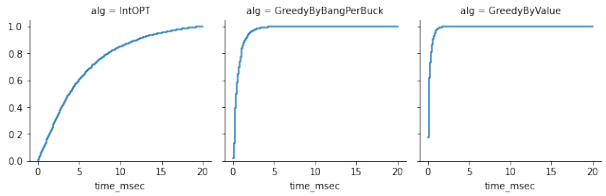

Figure 3: CDF of running time in milliseconds for for GreedyByValue, GreedyByBangPerBuck and IntOPT algorithms.

GreedyByValue and GreedyByBangPerBuck. Blank spaces in this plot correspond to not having any queries with a particular combination of approximation factors. We note that a lot of the queries have the same approximation factor for GreedyByValue and GreedyByBangPerBuck — indicating that RandomizedGreedy won't make mistakes. But GreedyByBangPerBuck more often has better approximation factor than GreedyByValue, so sticking to GreedyByBangPerBuck as the only heuristic might perform better. Finally, in Figure 3 we compare the clock-time of our allocation rules with that of the IntOPT allocation rule.

These allocation rules are monotone, so they can be paired with Myerson's payment rule as implied by Lemma 1. We evaluate the time required to compute the payments and relative revenue compared to VCG in Appendix G (Table 3). Since the focus of our paper is welfare maximization we do not consider reserve prices. A possible direction for future work is to study the trade-offs between welfare and revenue of our mechanism under different reserve prices. Additionally, in Appendix G we evaluate our algorithms with an added cardinality constraints and show that we obtain reasonable approximations to the optimal allocation.

Acknowledgements

The work of Divyarthi Mohan was partially supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement no. 866132) and by the Israel Science Foundation (grant no. 317/17). Alexandros Psomas is supported in part by an NSF CAREER award CCF-2144208, a Google Research Scholar Award, and by the Algorand Centres of Excellence program managed by Algorand Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of Algorand Foundation. Part of this work was done when Mohan and Psomas were employed at Google Research.

References

[AGM06] Gagan Aggarwal, Ashish Goel, and Rajeev Motwani. Truthful auctions for pricing search keywords. In Joan Feigenbaum, John C.-I. Chuang, and David M. Pennock, editors, *Proceedings 7th ACM Conference on Electronic Commerce (EC-2006), Ann Arbor, Michigan, USA, June 11-15, 2006*, pages 1–7. ACM, 2006.

[CF21] Ethan Cramer-Flood. Worldwide digital ad spending year-end update. *e-Marketer*, 2021.

[CKK+15] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, Maria Kyropoulou, Brendan Lucier, Renato Paes Leme, and Éva Tardos. Bounding the inefficiency of outcomes in generalized second price auctions. *J. Econ. Theory*, 156:343–388, 2015.

[CKKK11] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. On the efficiency of equilibria in generalized second price auctions. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 81–90, 2011.

[CKSW17] Ruggiero Cavallo, Prabhakar Krishnamurthy, Maxim Sviridenko, and Christopher A. Wilkens. Sponsored search auctions with rich ads. In Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich, editors, *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 43–51. ACM, 2017.

[CMSW20] Riccardo Colini-Baldeschi, Julián Mestre, Okke Schrijvers, and Christopher A. Wilkens. The ad types problem. In Xujin Chen, Nikolai Gravin, Martin Hoefer, and Ruta Mehta, editors, *Web and Internet Economics - 16th International Conference, WINE 2020, Beijing, China, December 7-11, 2020, Proceedings*, volume 12495 of *Lecture Notes in Computer Science*, pages 45–58. Springer, 2020.

[DGS+20] Nikhil R Devanur, Kira Goldner, Raghuvansh R Saxena, Ariel Schvartzman, and S Matthew Weinberg. Optimal mechanism design for single-minded agents. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 193–256, 2020.

[DHP20] Nikhil R Devanur, Nima Haghpanah, and Alexandros Psomas. Optimal multi-unit mechanisms with private demands. *Games and Economic Behavior*, 121:482–505, 2020.

[DSYZ10] Xiaotie Deng, Yang Sun, Ming Yin, and Yunhong Zhou. Mechanism design for multi-slot ads auction in sponsored search markets. In Der-Tsai Lee, Danny Z. Chen, and Shi Ying, editors, *Frontiers in Algorithmics*, pages 11–22, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[DW17] Nikhil R Devanur and S Matthew Weinberg. The optimal mechanism for selling to a budget constrained buyer: The general case. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 39–40, 2017.

[EOS07] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, March 2007.

[FGKK16] Amos Fiat, Kira Goldner, Anna R. Karlin, and Elias Koutsoupias. The fedex problem. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, page 21–22, New York, NY, USA, 2016. Association for Computing Machinery.

[GHLY19] MohammadAmin Ghiasi, MohammadTaghi Hajiaghayi, Sébastien Lahaie, and Hadi Yami. On the efficiency and equilibria of rich ads. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 301–307. ijcai.org, 2019.

[HIK+18] Jason D. Hartline, Nicole Immorlica, Mohammad Reza Khani, Brendan Lucier, and Rad Niazadeh. Fast core pricing for rich advertising auctions. In Éva Tardos, Edith Elkind, and Rakesh Vohra, editors, *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 111–112. ACM, 2018.

[Hum16] Patrick Hummel. Position auctions with dynamic resizing. *International Journal of Industrial Organization*, 45:38 – 46, 2016.

[KP99] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *Annual symposium on theoretical aspects of computer science*, pages 404–413. Springer, 1999.

[Law79] Eugene L Lawler. Fast approximation algorithms for knapsack problems. *Mathematics of Operations Research*, 4(4):339–356, 1979.

[LB10] Brendan Lucier and Allan Borodin. Price of anarchy for greedy auctions. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 537–553. SIAM, 2010.

[LPL11] Brendan Lucier and Renato Paes Leme. Gsp auctions with correlated types. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 71–80, 2011.

[MV09] Alexey Malakhov and Rakesh V Vohra. An optimal auction for capacity constrained bidders: a network perspective. *Economic Theory*, 39(1):113–128, 2009.

[Mye81] Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

[NR01] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games Econ. Behav.*, 35(1-2):166–196, 2001.

[NR10] Uri Nadav and Tim Roughgarden. The limits of smoothness: A primal-dual framework for price of anarchy bounds. In Amin Saberi, editor, *Internet and Network Economics - 6th International Workshop, WINE 2010, Stanford, CA, USA, December 13-17, 2010. Proceedings*, volume 6484 of *Lecture Notes in Computer Science*, pages 319–326. Springer, 2010.

[Rou15a] Tim Roughgarden. Intrinsic robustness of the price of anarchy. *J. ACM*, 62(5):32:1–32:42, 2015.

[Rou15b] Tim Roughgarden. The price of anarchy in games of incomplete information. *ACM Trans. Economics and Comput.*, 3(1):6:1–6:20, 2015.

[ST13] Vasilis Syrgkanis and Eva Tardos. Composable and efficient mechanisms. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 211–220, 2013.

[SZ79] Prabhakant Sinha and Andris A. Zoltners. The multiple-choice knapsack problem. *Operations Research*, 27(3):503–515, 1979.

[Var07] Hal Varian. Position auctions. *International Journal of Industrial Organization*, 25(6):1163–1178, 2007.

[VH14] Hal R. Varian and Christopher Harris. The vcg auction in theory and practice. *American Economic Review*, 104(5):442–45, May 2014.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See "Results and Techniques" and Section 5

(c) Did you discuss any potential negative societal impacts of your work? [N/A]

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See technical sections (all but Section 5)

   (b) Did you include complete proofs of all theoretical results? [Yes] Proofs missing from the first 9 pages can be found in the appendix

3. If you ran experiments...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The data is proprietary.

   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]

   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We report error bars due to randomization of the algorithm and report distribution over a sample of queries.

   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We ran our algorithms on a single machine.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

   (a) If your work uses existing assets, did you cite the creators? [N/A]

   (b) Did you mention the license of the assets? [N/A]

   (c) Did you include any new assets either in the supplemental material or as a URL? [No] The assets are proprietary

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] The data does not contain any personally identifiable information or offensive content

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Missing Preliminaries

A.1 Fractional Optimal and the Incremental-bang-per-buck Algorithm

[SZ79] introduce the notion of *Dominated* and *LP dominated* ads and show that they are not used in the fractional optimal solution.

Definition 3 (Dominated/LP-dominated [SZ79])**.** *For an advertiser i, the rich ads of the advertiser can be dominated in two ways.*

- *Dominated: If two rich ads satisfy $w_{ij} \leq w_{ik}$ and $b_{ij} \geq b_{ik}$ then k is dominated by j.*
- *LP Dominated: If three rich ads j, k, l with $w_{ij} < w_{ik} < w_{il}$ and $b_{ij} < b_{ik} < b_{il}$, satisfy $\frac{b_{il}-b_{ik}}{w_{il}-w_{ik}} \geq \frac{b_{ik}-b_{ij}}{w_{ik}-w_{ij}}$, then k is LP-dominated by j and l.*

Moroever, [SZ79] showed that the fractional optimal solution can be obtained through the following *incremental-bang-per-buck* algorithm.

Algorithm 2.

- *Eliminate all the Dominated and LP-dominated rich ads for each advertiser.*
- *(Compute incremental-bang-per-buck) For each advertiser, allocate the null ad. Sort the remaining rich ads by space (label them $i1, \ldots ik, \ldots$). Construct a vector of scores $\frac{b_{ik}-b_{ik-1}}{w_{ik}-w_{ik-1}}$ for these.*
- *(Allocate in incremental bang-per-buck order) While space remains: select the rich ad (i, k) with highest score among remaining rich ads. If the remaining space is at least as much as the incremental space required ($w_{ik} - w_{i(k-1)}$), this new rich ad is allocated to its advertiser and it fully replaces previously allocated rich ad for the advertiser. Otherwise, allocate the advertiser fractionally in the remaining space. This fractional allocation puts a weight x on the previously allocated rich ad of the advertiser and $(1-x)$ on the newly selected rich-ad such that the fractional space equals remaining space plus previously allocated space of the advertiser.*[6]

We prove some lemmas about the optimal fractional solution. The following lemma provides a simple characterization of the advertisers' welfare depending on whether the optimal solution uses one or two rich ads fractionally.

Lemma 2. *Let W_i^* be the space allocated to advertiser i with a non-null allocation in an optimal solution to the* MULTI-CHOICE KNAPSACK *problem.*

There are two cases.

1. *The optimal allocation uses a single non-null rich ad with (value, size) (b_l, l) ($l = W_i^*$) in space W_i^*. The advertiser is allocated integrally and its value is b_l.*
2. *The optimal allocation uses two rich ads with (value, size) (b_s, s), (b_l, l), with $s < W_i^* < l$ and (b_l, l) not null, in space W_i^*. We have that $b_l > b_s$ and the advertiser's value (i.e. their contribution to the social welfare) is $b_s + \frac{b_l-b_s}{l-s}(W_i^* - s)$. If (b_s, s) is not null, $\frac{b_s}{s} \geq \frac{b_l}{l} \geq \frac{b_l-b_s}{l-s}$.*

Proof. Recall that the main purpose of the null ad is to help make the fractional allocation of advertiser i exactly equal to one. We can reason about the optimal allocation of advertiser i in space W_i without the null ad and bringing the null ad if i's allocation is less than one. Note that the null ad does not change the value or space occupied by advertiser i. The optimization problem for a single

[6] For more details refer to Lemma 2 in the appendix.

advertiser (without the null ad) is as follows. Let S be the set of rich ads for advertiser i.

$$\begin{aligned}
\max \sum_{k \in S} x_k b_k \\
s.t. \sum_{k \in S} x_k w_k \leq W_i^* \\
\sum_{k \in S} x_k \leq 1 \\
x_k \geq 0 \qquad \forall k \in S
\end{aligned}$$

This LP has $|S|$ variables and $|S| + 2$ constraints, so there exists an optimal solution with at most 2 non-zero variables.

Suppose there is only one non-zero variable. The optimal fractional solution is made of a single ad (b_l, l) with $s \geq W_i^*$. Suppose the advertiser is allocated $x \leq 1$. Then since $lx \leq W_i^*$, we have that $x \leq \frac{W_i^*}{l}$. Since the optimal fractional solution maximizes the advertiser i's value $b_l x$ in that space (as noted in Fact 1), we have that $x = \frac{W_i^*}{l}$ and the advertisers value is $b_l \frac{W_i^*}{l} = \frac{b_l}{l} W_i^*$. When $l = W_i^*$, $x = 1$ and the advertiser's allocation is integral. Otherwise, $x < 1$, we can set $(b_s, s) = (0, 0)$ with $x_s = 1 - x$ and the advertiser's value is still $\frac{b_l}{l} W_i^* = \frac{b_l - 0}{l - 0} W_i^* + 0$.

Suppose the optimal allocation uses two (non-null) rich ads (b_s, s), (b_ℓ, ℓ). Then both the knapsack and unit demand constraints must be tight. That is $x_s s + x_\ell \ell = 1$ and $x_s s + x_\ell \ell = W_i^*$. The solution to this system is to have $x_s = \frac{l - W_i^*}{l - s}$ and $x_l = \frac{W_i^* - s}{l - s}$. Both x_s and x_l are fractional if $s < W_i^* < l$. And the advertiser's value is

$$\begin{aligned}
& b_s \frac{l - W_i^*}{l - s} + b_l \frac{W_i^* - s}{l - s} \\
& = b_s + (b_l - b_s) \frac{W_i^* - s}{l - s} \\
& = b_s + \frac{b_l - b_s}{l - s} (W_i^* - s)
\end{aligned}$$

□

The following lemma is a niche property of the optimal fractional solution constructed by the incremental bang-per-buck order algorithm (Appendix A.1) that can be easily derived and that we use in our proofs.

Lemma 3. *Let i be the last advertiser that is allocated in the incremental bang-per-buck order algorithm and suppose it is allocated fractionally. Let $\mathbf{x}^* = \mathbf{x}(OPT)$ denote the optimal fractional allocation. Let $(b_s, s), (b_\ell, \ell)$ be the ads used in x_i^*. For all $j \neq i$, $\frac{b_\ell - b_s}{\ell - s} \leq \frac{b_j \cdot x_j^*}{W_j^*}$.*

Proof. Let j be an advertiser with $j \neq i$. Since $j \neq i$, by Fact 1 allocation of j in $\mathbf{x}^*$ is integral. Let (b_{jk}, w_{jk}) be the ad allocated to j. Let (b_{jt}, w_{jt}) be the ad that was previously allocated to j when (b_{jk}, w_{jk}) was considered. Since the incremental bang-per-buck is defined by sorting ads in increasing order of their space, $w_{jk} \geq w_{jt}$. Thus $\frac{b_{jk} - b_{jt}}{w_{jk} - w_{jt}}$ is the incremental bang-per-buck that allowed j to be selected and since i is the last advertiser we have that $\frac{b_l - b_s}{l - s} \leq \frac{b_{jk} - b_{jt}}{w_{jk} - w_{jt}}$.

To conclude the proof, we show that $\frac{b_{jk} - b_{jt}}{w_{jk} - w_{jt}} \leq \frac{b_{jk}}{w_{jk}}$. This is true if and only if $\frac{b_{jk}}{w_{jk}} \leq \frac{b_{jt}}{w_{jt}}$. We have that $0 \leq w_{jt} \leq w_{jk}$. We can conclude that $0 \leq b_{jt} \leq b_{jk}$, as otherwise, k is dominated by t. Finally, Suppose, $\frac{b_{jk}}{w_{jk}} > \frac{b_{jt}}{w_{jt}}$, then with simple rearrangement we can obtain that $\frac{b_{jk}}{w_{jk}} < \frac{b_{jt} - b_{jk}}{w_{jt} - w_{jk}}$ and k is LP-dominated by 0 and t. □

A.2 GSP Pricing and Price of Anarchy

In Sponsored Search, a popular pricing choice is the Generalized Second Price (GSP) [Var07, EOS07]. While this is classically defined for a position auction with k ads being selected, it can be defined for any allocation algorithm that is monotone in the bid.

Definition 4 (GSP). *For any allocation rule x_i that is monotone in bid, and a bid profile $(\mathbf{b}, \mathbf{S})$, the GSP per click price for advertiser i is the minimum bid below which the advertiser obtains a smaller amount of expected clicks: $cpc_i(\mathbf{b}, \mathbf{S}) = \arg\min_{b_i' : x_i(b_i', S_i, \mathbf{b}_{-i}, \mathbf{S}_i) = x_i(b_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})} b_i'$. Given a GSP cost-per-click, the GSP payment is the cost-per-click times expected number of clicks: $p_i(\mathbf{b}, \mathbf{S}) = x_i(\mathbf{b}, \mathbf{S}) cpc_i(\mathbf{b}, \mathbf{S})$.*

The mechanism that charges GSP prices may not be truthful. We can study the Price of Anarchy (PoA) [KP99] to understand the effective social welfare. The notion of Price of Anarchy captures the inefficiency of a pure Nash equilibrium. Fix a valuation profile $(\mathbf{v}, \mathbf{A})$. A set of bids $(\mathbf{b}, \mathbf{S})$ is a *pure Nash equilibrium*, if for each i, for any (b_i', S_i), $u_i(v_i, A_i \to b_i, S_i; \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq u_i(v_i, A_i \to b_i', S_i'; \mathbf{b}_{-i}, \mathbf{S}_{-i})$. The pure Price of Anarchy is the ratio of the optimal social welfare to the social welfare of the worst pure Nash equilibrium of the mechanism $\mathcal{M}$: pure PoA $= \sup_{\mathbf{v}, \mathbf{A},\ \text{pure Nash}(\mathbf{b}, \mathbf{S})} \frac{SW(OPT(\mathbf{v}, \mathbf{A}))}{SW(\mathcal{M}(\mathbf{b}, \mathbf{S}))}$.

We will also consider Bayes-Nash Price of Anarchy. Let $(\mathbf{v}, \mathbf{A})$ be drawn from a (possibly correlated) distribution $\mathcal{D}$, and D_i be the marginal of i in $\mathcal{D}$. A *Bayes-Nash equilibrium* is a (possibly randomized) mapping from value, set of rich ads (v_i, A_i) to a reported type $(b_i(v_i, A_i), S_i(v_i, A_i))$ for each i and $(v_i, A_i) \in Support(D_i)$ such that, for any b_i' and $S_i' \subseteq A_i$,

$$\mathbb{E}_{\mathbf{v}_{-i}, \mathbf{A}_{-i}, \mathbf{b}_{-i}, \mathbf{S}_{-i}}[u_i(v_i, A_i \to b_i, S_i; \mathbf{b}_{-i}, \mathbf{S}_{-i})] \geq \mathbb{E}_{\mathbf{v}_{-i}, \mathbf{A}_{-i}, \mathbf{b}_{-i}, \mathbf{S}_{-i}}[u_i(v_i, A_i \to b_i', S_i', \mathbf{b}_{-i}, \mathbf{S}_{-i})]$$

In the expression above the expectation is conditioned on v_i and over random draws of $\mathbf{v}_{-i}, \mathbf{A}_{-i}$ and the competitors bids $\mathbf{b}_{-i}, \mathbf{S}_{-i}$ drawn from the mapping $b_j(v_j, A_j), S_j(v_j, A_j)$ for each $j \neq i$. The *Bayes-Nash Price of Anarchy (PoA)* is the ratio of the optimal social welfare to that of the worst Bayes-Nash equilibrium of the mechanism $\mathcal{M}$.

$$\text{Bayes-Nash POA} = \sup_{\mathcal{D}, (\mathbf{b}, \mathbf{S})\ \text{Bayes Nash eq.}} \frac{\mathbb{E}_{(\mathbf{v}, \mathbf{A})}[SW(OPT(\mathbf{v}, \mathbf{A}))]}{\mathbb{E}_{(\mathbf{v}, \mathbf{A}), (\mathbf{b}, \mathbf{S})}[SW(\mathcal{M}(\mathbf{b}(\mathbf{v}, \mathbf{A}), \mathbf{S}(\mathbf{v}, \mathbf{A})))]}.$$

For the PoA bounds we also focus on equilibria that satisfy the *no overbidding* condition, i.e. equilibria where each bid profile $\mathbf{b}, \mathbf{S}$ satisfies $b_i \leq v_i$ for every advertiser i. Note that, the equilibrium condition still allows for deviations that overbid. However, by the definition of GSP overbidding is dominated. If an advertiser can obtain a higher expected number of clicks by bidding higher than their value, then their GSP cost-per-click will be larger than their value v_i, and the advertiser obtains negative utility. On the other hand if the expected number of clicks is unchanged, then the GSP cost-per-click is also the same and the utility is unchanged as well, thus no advertiser will be able to gain by overbidding.

B Proofs from Section 3

Proof of Lemma 1. We note that, for a fixed set of rich ads S_i' this is a single parameter setting (in the bids b_i). Since we use the same payment rule as Myerson, his result implies for any b_i, S_i' $u_i(b_i, \mathbf{b}_{-i}, S_i', \mathbf{S}_{-i}) \leq u_i(v_i, \mathbf{b}_{-i}, S_i', \mathbf{S}_{-i})$. Thus the mechanism is truthful in bids.

Next we show that reporting a smaller set of rich ads and the true value is not beneficial, i.e. $u_i(v_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \leq u_i(v_i, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})$, for any $S_i \subseteq A_i$. From the definition of the payment, $u_i(v_i, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) = \int_0^{v_i} x_i(b, S_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) db$. Since the allocation rule is monotone, we have $x_i(b, \mathbf{b}_{-i}, A_i, \mathbf{S}_{-i}) \geq x_i(b, \mathbf{b}_{-i}, S_i, \mathbf{S}_{-i})$; the claim follows.

Finally we can chain these two results to show that misreporting both bid and the set of rich ads will also not increase a buyer's utility. Let S_i be the reported set of rich ads, and fix $\mathbf{b}_{-i}$ and $\mathbf{S}_{-i}$. Recall that in our model the buyer can only report a subset of rich-ads, thus $S_i \subseteq A_i$. Further for any valid allocation rule $x_{ij} = 0$ for all $j \notin S_i$. Thus the utility when the true type is A_i is equal to the utility when the true type is S_i: $u_i(\nu_i, A_i \to \nu_i', S_i) = u_i(\nu_i, S_i \to \nu_i', S_i)$ for all ν_i, ν_i'. Putting all the previous claims together we have:

$$\begin{aligned}
u_i(v_i, A_i \to v_i, A_i) &\geq u_i(v_i, A_i \to v_i, S_i) \quad \text{(By local IC)} \\
&= u_i(v_i, S_i \to v_i, S_i) \\
&\geq u_i(v_i, S_i \to b_i, S_i) \quad \text{(By local IC)} \\
&= u_i(v_i, A_i \to b_i, S_i)
\end{aligned}$$

□

Proof of Theorem 1 continued. Consider an instance with two advertisers A, B, and rich ads $\{(1,1),(2-\varepsilon,2)\}$. The total space available is 3. The optimal allocation can randomly pick between $\{A:(1,1), B:(2-\varepsilon,2)\}$ and $\{A:(2-\varepsilon,2), B:(1,1)\}$, getting a total value of $3-\varepsilon$. Therefore, in the output of any algorithm in this instance, some advertiser, say B, obtains value $x \leq (3-\varepsilon)/2$. A randomized monotone allocation rule must ensure that B's value, if she hides $(1,1)$, is at most x. In that case, the algorithm will randomize between $\{A:(1,1), B:(2-\varepsilon,2)\}$ and $\{A:(2-\varepsilon,2)\}$, and it choose the first allocation with probability more than $\frac{x}{2-\varepsilon}$. The social welfare of this randomized allocation is at most $\frac{x}{2-\varepsilon}\cdot(3-\varepsilon)+(1-\frac{x}{2-\varepsilon})\cdot(2-\varepsilon) \leq (11/12)\cdot(3-\varepsilon)$. Recalling that OPT is $(3-\varepsilon)$ concludes the proof. □

C Proofs and Definitions from Section 4

C.1 Formal description of ALG_B

Algorithm 3 (ALG_B)**.** *Let w denote the remaining space at any stage of the algorithm; initialize $w = W$. Let E_i be the set of ads that are available to agent i and let $\mathcal{E}$ denote the set $\cup_{i=1}^{n} E_i$; initialize $E_i = S_i$. Let x_{ij} denote the fractional allocation of advertiser i for the rich ad j. The total space allocated to advertiser i is $W_i = \sum_{j\in\mathcal{M}} w_{ij}x_{ij}$.*

While the remaining space w is not zero:

1. *Let i be the advertiser whose rich ad (b_{ij}, w_{ij}) has the highest bang-per-buck, among all ads in $\mathcal{E}$. Let (b_{ik}, w_{ik}) be the ad previously allocated to i; $(b_{ik}, w_{ik}) = (0,0)$, if no previous ad exists.*
2. *Remove all ads of i (including ad j) with space at most w_{ij} from E_i.*
3. *If $w \geq w_{ij} - w_{ik}$, add $w_{ij} - w_{ik}$ to the total space allocated to advertiser i, which now becomes $W_i = w_{ij}$. Allocate rich ad j in that space, i.e. set $x_{ij} = 1$ (and $x_{ik} = 0$). Update $w = w - w_{ij} + w_{ik}$.*
4. *If $w < w_{ij} - w_{ik}$, add w to the total space allocated to advertiser i, that is, $W_i = w_{ik} + w$. Allocate rich ad j to i fractionally with $x_{ij} = W_i/w_{ij}$ (and set $x_{ik} = 0$). Update $w = 0$.*

C.2 Missing Proofs

Proof of Theorem 2. ALG_B allocates some ads that are later replaced. We refer to such ads as temporarily allocated.

We first prove that an advertiser i will not get allocated less space when bidding $b'_i > b_i$. For any agent $i \in \mathcal{N}$ and any $j \in S_i$, ALG_B temporarily allocates j if and only if the total space occupied by ads with higher bang-per-buck (counting only the largest such ad for each advertiser) is strictly less than W. For any ad $j \in S_i$, j's bang-per-buck b_{ij}/w_{ij} is increasing in the bid. Therefore, if $j \in S_i$ was temporarily allocated to i when reporting b_i, then j will definitely be temporarily allocated to i under $b'_i > b_i$. If j is the last ad to be allocated by the algorithm when reporting b_i (and thus might not fit integrally in the remaining space under bid b_i), it will only be considered earlier under $b'_i > b_i$, and therefore the remaining space (before allocating j) can only be (weakly) larger. Thus, in all cases, the space allocated to i under b'_i is at least as much as under b_i.

We next show that by removing an ad $k \in S_i$ the space allocated to i does not increase. Note that it is sufficient to prove monotonicity removing one rich-ad at a time. Monotonicity for removing a subset of rich-ads follows through transitivity. If no ad is allocated to i under S_i, then definitely the same holds for $S_i \setminus \{k\}$. Otherwise, let $j \in S_i$ be the final ad allocated to i under S_i. If k was never (temporarily or otherwise) allocated under S_i, then the allocation under $S_i \setminus \{k\}$ remains the same. Therefore, we focus on the case that k was allocated to i under S_i.

First, consider the case that $k \neq j$. Let ℓ be an ad (temporarily or otherwise) allocated under $S_i \setminus \{k\}$, but not under S_i (if no such ad exists, the claim follows). It must be that $b_{ik}/w_{ik} \geq b_{i\ell}/w_{i\ell}$ otherwise the algorithm under S_i would have temporarily allocated ℓ before k. Note, if $w_{i\ell} > w_{ik}$ and the algorithm under S_i did not allocate ℓ then it must be the case that space ran out before we got to ℓ. This implies that, also under $S_i \setminus \{k\}$, the space will run out before we get to ℓ. Hence $w_{i\ell} \leq w_{ik}$. At the time when ℓ is allocated to i (under $S_i \setminus \{k\}$), the total space allocated to bidders other than i

is weakly larger compared to the time when k is allocated to i (under S_i), while the space allocated to i is weakly smaller.

Second, consider the case that $k = j$ and i is not the last advertiser allocated by the algorithm. Removing k leads to a different last ad, say ℓ, for i under $S_i \setminus \{k\}$. If this ad was temporarily allocated under S_i, the claim follows. Otherwise, ℓ must have a lower bang-per-buck than k. If $w_{i\ell} \leq w_{ik}$, the claim follows. Otherwise, all advertisers who got allocated after i under S_i (we know this set is non-empty) have the opportunity to claim a (weakly) larger amount of space under $S_i \setminus \{k\}$, before ℓ is considered. Thus, the maximum amount of space i is allocated is w_{ik}.

Finally, suppose $k = j$, and i is the last advertiser allocated by the algorithm. let $W_{-i} = W - w_{ij}x_{ij}$ be the space allocated to advertisers other than i, under S_i. Since they are allotted this space *before* ad k is considered, the maximum space i can get is $W - W_{-i}$, her allocation under S_i. □

Proof of Theorem 3 continued. The following instance shows that the approximation of the algorithm is at least 3 (see Appendix C for the proof). Let M be a large integer. There are four advertisers named A, B, C, D. We describe the set of rich ads, bang-per-buck and incremental bang-per-buck for each advertiser in Table 2. Total space $W = 2M - \varepsilon$. We assume ties are broken in the order A, B, C, D (but the example can be constructed without ties).

	rich ads	bpb	ibpb
A	$(M, 1), (M + \varepsilon, M)$	$M, \frac{M+\varepsilon}{M}$	M, $\frac{\varepsilon}{M-1}$
B	$(1 + \varepsilon, 1), (M + \varepsilon, M)$	$1 + \varepsilon, \frac{M+\varepsilon}{M}$	$1 + \varepsilon, 1$
C	$(M - 1, M - 1)$	1	1
D	$(M + 2\varepsilon, 2M - \varepsilon)$	$\frac{M+2\varepsilon}{2M-\varepsilon}$	$\frac{M+2\varepsilon}{2M-\varepsilon}$

Table 2: Rich ads, Bang-per-buck(bpb) and Incremental Bang-per-bucks(ibpb) for each advertiser.

The fractional optimal solution can be constructed by allocating ads in the incremental-bang-per-buck order. The incremental bang-per-buck order is: $A : (M, 1), B : (1 + \varepsilon, 1), B : (M + \varepsilon, M), C : (M - 1, M - 1), D : (M + 2\varepsilon, 2M - \varepsilon), A : (M + \varepsilon, M)$. Any subsequent ad from the same advertiser fully replaces the previously selected ad. The allocation stops when the space runs out, so it will stop while allocating $C : (M - 1, M - 1)$ which will be allocated fractionally. The social welfare of the fractional optimal is $M + M + \varepsilon + \frac{(M-1-\varepsilon)}{M-1} \cdot (M - 1) = 3M - 1$.

ALG_B considers ads in the order $A : (M, 1), B : (1 + \varepsilon, 1), A : (M + \varepsilon, M), B : (M + \varepsilon, M), C : (M-1, M-1), D : (M+2\varepsilon, 2M-\varepsilon)$. Once again, any subsequent ad from the same advertiser fully replaces the previously selected ad. The algorithm stops when the space runs out. Thus the algorithm stops while allocating $B : (M + \varepsilon, M)$. ALG_B will allocate space of M to advertiser A and space $M - \varepsilon$ to advertiser B. ALG_I runs ALG_B and post-processes to find the best ad for the allocated space. Thus A is allocated $(M + \varepsilon, M)$ and B is allocated $(1 + \varepsilon, 1)$. The social welfare of ALG_I is $1+\varepsilon+M+\varepsilon = M+1+2\varepsilon$. The maximum value allocation will select $D : (M+2\varepsilon, 2M-\varepsilon)$. Thus the expected value of the algorithm that randomly chooses between ALG_I with probability $2/3$ and the largest value single ad with probability $1/3$ is $2/3(M+1+2\varepsilon)+1/3(M+2\varepsilon) = M+2/3+2\varepsilon$ and the ratio with the fractional optimal allocation is $\frac{3M-1}{M+2\varepsilon+2/3} = \frac{3-1/M}{1+\frac{2}{3M}+2\frac{\varepsilon}{M}}$. This ratio can be made arbitrarily close to 3 by choosing $\varepsilon = 1/M$ and M that is large enough. □

Now we prove all the claims from Section 4

Proof of Observation 1. We have that $\frac{b_{ij}}{w_{ij}} \geq \frac{b_{ij'}}{w_{ij'}}$ and $w_{ij} \geq w_{ij'}$. If we multiply these two inequalities we get: $b_{ij} \geq b_{ij'}$. If $w_{ij} = w_{ij'}$, then j' is dominated by j. Otherwise, we will show that j' is LP-dominated by j and 0. We have that $0 < w_{ij'} < w_{ij}$ and $0 \leq b_{ij'} \leq b_{ij}$. It remains to show, $\frac{b_{ij}-b_{ij'}}{w_{ij}-w_{ij'}} \geq \frac{(b_{ij'}}{w_{ij'}}$. This is equivalent to $\frac{b_{ij}}{w_{ij}} \geq \frac{b_{ij'}}{w_{ij'}}$, which is true. □

Proof of Claim 1. The post-processing step in ALG_I integrally allocates the best ad that fits in W_i for all advertisers $i \in \mathcal{N}$. Let $INT_i(w)$ denote best ad in S_i that fits in space w.

For all $i \in \mathcal{I} \setminus \{i'\}$, since $W_i \geq W_i^*$ and the optimal allocation $\mathbf{x}_i^*$ is integral, we get $b_i \cdot \mathbf{x}_i(ALG_I) = b_i \cdot \mathbf{x}_i(INT_i(W_i) \geq b_i \cdot \mathbf{x}_i(INT_i(W_i^*)) = b_i \cdot \mathbf{x}_i(OPT_i(W_i^*))$. Recall, by Lemma 2, we have $b_i \cdot \mathbf{x}_i(OPT_i(W_i^*)) = b_i \cdot \mathbf{x}_i^*$.

Thus we get $b_i \cdot \mathbf{x}_i^* \leq b_i \cdot \mathbf{x}_i(ALG_I)$ for all $i \in \mathcal{I} \setminus \{i'\}$. Further, by summing up the contributions of all $i \in \mathcal{I} \setminus \{i'\}$, we get $\text{Val}(\mathbf{x}^*, \mathcal{I} \setminus \{i'\}, \vec{W}^*) \leq \text{Val}(\mathbf{x}(ALG_I), \mathcal{I} \setminus \{i'\}, \vec{W})$. □

Proof of Claim 2. ALG_B consider ads in decreasing order of bang-per-buck, and moreover by Observation 1 it never "ignores" ads that are allocated in OPT. So, if $k \in \mathcal{K} \setminus \{i'\}$ had higher bang-per-buck in OPT than any advertiser i in ALG_B, then the space allotted for k in ALG_B would have been at least W_k^*; a contradiction. Note that this holds even if $i = i''$ is the last advertiser considered in ALG_B (who potentially gets a fractional allocation). □

Proof of Claim 3. From Lemma 2 we know that $b_\ell > b_s$ and $b_{i'} \cdot \mathbf{x}_{i'}^* = b_s + (b_\ell - b_s)\frac{W_{i'}^* - s}{\ell - s}$. Moreover if (b_s, s) is not the null ad, i.e. $s > 0$, $\frac{b_\ell - b_s}{\ell - s} \leq \frac{b_\ell}{\ell} \leq \frac{b_s}{s}$,

Suppose that $i' \in \mathcal{K}$. If $s > W_{i'}$, then by the same argument as Claim 2 we have that for all $i \in \mathcal{N}$ the bang-per-buck of every ad in ALG_B is at least $b_s/s \geq b_\ell/\ell \geq (b_\ell - b_s)/(\ell - s)$. If $s \leq W_{i'}$, we have $b_s \leq b_{i'} \cdot \mathbf{x}_{i'}(ALG_I)$. And, by the same argument as Claim 2, we have that $\frac{b_\ell - b_s}{\ell - s} \leq \frac{b_\ell}{\ell} \leq \frac{b_j \mathbf{x}_j(ALG_I)}{W_j}$ for $j \in \mathcal{N}$.

Suppose that $i' \in \mathcal{I}$. Clearly, $s \leq W_{i'}^* \leq W_{i'}$, so we have $b_s \leq v_{i'} \cdot \mathbf{x}_{i'}(ALG_I)$. Let $k \in \mathcal{K}$ be some ad with $W_k^* > W_k$. By Lemma 3 we have $(b_\ell - b_s)/(\ell - s) \leq b_k \cdot \mathbf{x}_k^*/W_k^*$. Claim 2 gives us that $b_k \cdot \mathbf{x}_k^*/W_k^* \leq b_i \cdot \mathbf{x}_i(ALG_B)/W_i$ for all $i \in \mathcal{N}$. Thus we get $(b_\ell - b_s)/(\ell - s) \leq b_i \cdot \mathbf{x}_i(ALG_B)/W_i$ for all $i \in \mathcal{N}$. □

C.3 Computing Myerson Payments

Finally, we note that the truthful payment function matching our allocation rule (that gives the overall auction) can be computed in time that is polynomial in the number of advertisers and rich ads. Let ALG_{max} be the algorithm which simply allocates the maximum valued ad. To compute the payment for an advertiser i, we need to compute the expected allocation as a function of i's bid b_i. We can compute the expected allocation from ALG_I and ALG_{max} independently, and the final allocation is just $\frac{2}{3}x_i(ALG_I(b, \mathbf{b}_{-i}, \mathbf{S}) + \frac{1}{3}x_i(ALG_{max}(b, \mathbf{b}_{-i}, \mathbf{S})$. The payment is then $\frac{2}{3}p_i(ALG_I(\mathbf{b}, \mathbf{S})) + \frac{1}{3}p_i(ALG_{max}(\mathbf{b}, \mathbf{S}))$.

The payment for ALG_{max} is simple: Advertiser i's allocation is $\max_{j \in S_i} \alpha_{ij}$ if she is the highest value bidder, and zero otherwise. The expected payment p_i is the second highest value $\max_{i' \neq i, j \in S_{i'}} b_{i'j}$ if i is allocated and zero otherwise. The payment for ALG_I can be computed by identifying possible thresholds for i's bid where i's allocation changes, and computing expected allocation for these thresholds. Advertiser i's allocation can change *only* when his bang-per-buck for one of his ads is tied with that of another ad: there are therefore at most $O(n|\mathcal{S}|^2)$ such thresholds. Once we identify thresholds $t_1, t_2, ...$, the corresponding allocations can be computed by re-running the bang-per-buck allocation algorithm. Note that as i's bid changes, the relative order of rich ads does not change for any advertiser $j \in \mathcal{N}$. Thus, the new bang-per-buck allocation can be computed in $O(n|\mathcal{S}|)$ time. Once we have the thresholds $t_0 = 0, t_1, t_2, ...$ and the corresponding expected allocations $x_i(ALG_I(t_j, b_{-i}, \mathbf{S}))$, the final payment is $\sum_{j=1} (x_i(ALG_I(t_j, b_{-i}, \mathbf{S})) - x_i(ALG_I(t_{j-1}, b_{-i}, \mathbf{S}))\, t_j$.

D Examples

The following example shows that ALG_B might not be monotone.

Example 4. *Consider two advertisers A and B. A has two rich ads with (value,size) = $(2, 2)$ and $(1, 3)$, and B has one rich ad with value size $(0.5, 3)$. Let the total space $W = 3$. ALG_B will allocate $(1, 3)$. But if A removed $(1, 3)$, then the algorithm allocates $(2, 2)$ to A and $(0.5, 3)$ to B fractionally.*

The following example shows that ALG_I can be an arbitrarily bad approximation to the social welfare.

Example 5. *We have two advertisers A and B. A has one rich ad with (value, size) = $(\varepsilon, \varepsilon/2)$, and B has one rich ad ((M, M)). The total space available is $W = M$.*

Clearly, the optimal integer allocation is to award the entire space to B, to obtain social welfare $= M$. The fractional opt selects $A : (\varepsilon, \varepsilon/2)$ and $B : (M, M)$ with weight $M - \varepsilon/2$, obtaining social welfare $= M + \varepsilon/2$. We note that, in this instance the bang-per-buck algorithm also selects the fractional optimal allocation. At the same time, the integer allocation $\mathbf{x}(ALG_I)$, drops B, and only obtains social welfare $= \varepsilon$.

E Price of Anarchy Bounds for GSP

In this section we prove bounds on the pure and Bayes-Nash PoA when monotone algorithms are paired with the GSP payment rule. Note that unlike the previous section, in this section we bound relative to the optimal integer allocation which we denote as $IntOPT$.

We consider a mechanism $\mathcal{M}$ that runs ALG_I with probability $1/2$ and allocates the maximum value ad with probability $1/2$. The corresponding GSP payment for either allocation rule is charged depending on the coin flip. The bidders utility is modelled in expectation over the random coin flip that selects between the two allocation rules. We refer to ALG_I as bang-per-buck allocation and that of the maximum-value ad as the max-value allocation.

The following example illustrates that GSP paired with our allocation rules can be non-truthful.

Example 6. *Suppose there are two advertisers. Advertiser 1 has value $v_1 = 1$ and set of rich ads A_1 with one rich ad with (click probability, space): $(1/M, 1)$. Advertiser 2 has value $v_2 = 1$ and set of rich ads A_2 with two rich ads having (click probability, space): $(\varepsilon/M, 1), (1 + \varepsilon^2, M)$ respectively. Suppose total space $W = M$.*

In this example, truth-telling is not an equilibrium. Suppose both advertisers bid truthfully. The bang-per-buck order is $2 : (1 + \varepsilon^2, M), 1 : (1/M, 1), 2 : (\varepsilon/M, 1)$. Thus advertiser 2 gets allocated $(1 + \varepsilon^2, M)$ and pays cost-per-click $\frac{1}{M} \cdot \frac{M}{1+\varepsilon^2}$, which is the minimum bid below which 2's bang-per-buck $(1 + \varepsilon^2) \cdot bid/M$ is lower than advertiser 1's. In the max-value allocation advertiser 2 is allocated $(1 + \varepsilon^2, M)$ with the GSP cost-per-click of $1/M \cdot 1/(1 + \varepsilon^2)$. Advertiser 2's utility is

$$u_2(v_1, A_1, v_2, A_2) = \frac{1}{2}(1 + \varepsilon^2)\left(1 - \frac{1}{M} \cdot \frac{M}{(1 + \varepsilon^2)}\right) + \frac{1}{2}(1 + \varepsilon^2)\left(1 - \frac{1}{M} \cdot \frac{1}{(1 + \varepsilon^2)}\right) = \frac{1}{2}(1 + 2\varepsilon^2 - 1/M)$$

However, if advertiser 2 bids $\frac{1}{2}$, advertiser 2's utility is

$$u_2(1/2, A_1, v_2, A_2) = \frac{1}{2}(\varepsilon/M) + \frac{1}{2}(1 + \varepsilon^2 - 1/M)$$

In the second case, the calculation for max-value is the same. In the bang-per-buck allocation, advertiser A has higher bang-per-buck and is allocated first. The rest of the space is allocated to advertiser 2 which is filled with the smaller ad. The GSP cost-per-click is zero. The latter utility is higher for $\varepsilon < 1/M$.

We will use SW_{bpb} and SW_{max} to denote the social welfare of the bang-per-buck and max-value algorithms. Then, $SW_{\mathcal{M}}(\mathbf{b}, \mathbf{S}) = \frac{1}{2}SW_{bpb}(\mathbf{b}, \mathbf{S}) + \frac{1}{2}SW_{max}(\mathbf{b}, \mathbf{S})$. We use u_i^{max} and u_i^{bpb} to denote the utility of advertiser i in the max-value and bang-per-buck allocation respectively.

The key challenge is in bounding the social welfare of the bang-per-buck allocation. Recall that the bang-per-buck algorithm allocates in the order of b_{ij}/w_{ij}. Rich ads from an advertiser that occupy less space than a previously allocated higher bang-per-buck rich ad are not picked. As the algorithm continues, it might replace a previously allocated rich ad [7] of an advertiser with another one that occupies more space (but may have lower value). The algorithm stops, when the next rich ad cannot fit within the available space or the set of rich ads runs out. We also post-process each advertiser to choose a rich ad with the highest value that fits within allocated space.

[7] (We will refer to these ads as being temporarily allocated)

We will develop a little notation to make the argument cleaner. For each i and $j \in S_i$ we use (i, j) to denote this rich-ad. Without loss, we can assume that the space w_{ij} occupied by any rich-ad (i, j) is integer and the total space available W is also integer. Let's think of the algorithm as consuming space in discrete units. For each unit of space, we can associate with it the rich ad that was first allocated to cover that unit of space. For the k'th unit of space, let $i(k)$ as the advertiser the k'th unit of space is allocated to, and $j(k)$ the rich ad $j(k) \in S_{i(k)}$ that was (temporarily) allocated for advertiser $i(k)$ when the k'th unit of space was first covered. Note that $j(k)$ may not be the final rich ad allocated to $i(k)$. In general, the algorithm stops before all the space runs out, in particular, because the next rich ad in the bang-per-buck order is too large to fit in the remaining space. We associate this rich-ad with each of the remaining units of space. It helps to be able to identify $(i(k), j(k))$ as we know that the bang-per-bucks $b_{i(k)j(k)}/w_{i(k)j(k)}$ are non-increasing as k increases.

We use the following lemma to relate an upper bound to the payment any "small" ad has to pay and the equilibrium social welfare.

Lemma 4. *Consider an equilibrium profile of per click bids* $\mathbf{b}$ *and sets of rich ads* $\mathbf{S}$. *We assume the bids satisfy no overbidding* $b_i \leq v_i$ *for each* i. *Let* $k = \lfloor W/2 \rfloor + 1$, $(r, j) = (i(k), b(k))$, *and* $\beta = \frac{b_{rj}}{w_{rj}}$. *Then,* $\beta W \leq 2 \cdot SW_{bpb}(\mathbf{b}, \mathbf{S}) + 2 \cdot SW_{max}(\mathbf{b}, \mathbf{S})$.

Proof. Recall that we defined $(i(k), j(k))$ as the rich-ad allocated by ALG_B to the k'th unit of space. Then we let $k^* = \lfloor W/2 \rfloor + 1$, $(r, j) = (i(k^*), j(k^*)$ and $\beta = b_{rj}/w_{rj}$. Note that (r, j) may not be allocated integrally if it is bigger than the remaining space.

Case (i): (r, j) is allocated integrally
The rich-ads are also allocated contiguously. Let $K_0 = 0$ and $K_t = K_{t-1} + w_{i(K_t)j(K_t)}$. K_t is the cumulative units of space covered by the first t rich ads. For each $k \in \{K_{t-1}+1, K_{t-1}+2, \ldots, K_t\}$, $i(k) = i(K_t)$ and $j(k) = j(K_t)$. Then $\sum_{k=K_{t-1}+1}^{K_t} \frac{v_{i(k)j(k)}}{w_{i(k)j(k)}} = v_{i(K_t)j(K_t)}$.

$$
\begin{aligned}
\beta W &\leq 2\beta k^* = 2\frac{b_{i(k^*)j(k^*)}}{w_{i(k^*)j(k^*)}} k^* \\
&\leq 2\sum_{k=1}^{k^*} \frac{b_{i(k)j(k)}}{w_{i(k)j(k)}} \\
&\leq 2\sum_{k=1}^{k^*} \frac{v_{i(k)j(k)}}{w_{i(k)j(k)}} = 2\sum_{t=1} \sum_{k=K_{t-1}+1}^{K_t} \frac{v_{i(k)j(k)}}{w_{i(k)j(k)}} = 2\sum_{t=1} v_{i(K_t)j(K_t)} \\
&\leq 2SW_{bpb}(\mathbf{b}, \mathbf{S})
\end{aligned}
$$

In the first inequality, we use that $k^* > W/2$. Second inequality follows since, for each $k \leq k^*$, $b_{i(k)j(k)}/w_{i(k)j(k)} \geq \beta$. In the third step, we use the no-overbidding assumption. And the last inequality uses the fact that the temporary allocation $v_{i(K_i))j(K_i} \leq v_{i(K_i)\rho(i(K_i))}$ where $\rho(i)$ denotes the ad allocated to advertiser i in the bang-per-buck allocation.
Case(ii) (r, j) is not placed.
We have that $w_{rj} > W/2$ and $\beta \leq 2b_{rj}/W$. Hence,$\beta W \leq 2b_{rj} \leq 2SW_{max}(\mathbf{b}, \mathbf{S})$.

Combining the inequalities for the two cases, we have, $\beta W \leq 2SW_{bpb}(\mathbf{b}, \mathbf{S}) + 2SW_{max}(\mathbf{b}, \mathbf{S})$ □

The following lemma establishes a bound on the utility of an advertiser with rich ad (i, t) of size less than $W/2$ bidding at least $\beta\frac{w_{it}}{\alpha_{it}}$ where α_{it} is the probability of click for rich ad (i, t) . We use these deviations with the equilibrium condition to relate the social welfare of a bid-profile to that of a target optimal outcome.

Lemma 5. *Let* $k = \lfloor W/2 \rfloor + 1$, $(r, j) = (i(k), b(k))$, *and* $\beta = b_{rj}/w_{rj}$ *as defined above. Then for an advertiser i, with rich-ad* (i, t) *with* $w_{it} \leq W/2$ *bidding* $(y, \{t\})$ *with* $y = \min\{v_i, \beta w_{it}/\alpha_{it} + \varepsilon\}$, *for some* ε, $u_i^{bpb}(y, \{t\}, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq v_{it} - \beta w_{it}$.

We prove the following lemma, which covers Lemma 5, and is also used in the Bayes-Nash POA proof.

Lemma 6. *Let $k = \lfloor W/2 \rfloor + 1$, $(r, j) = (i(k), b(k))$, and $\beta = b_{rj}/w_{rj}$ as defined above. Then for an advertiser i, with rich-ad (i, t) with $w_{it} \leq W/2$ bidding (y, A_i) with $v_i \geq y \geq \beta \frac{w_{it}}{\alpha_{it}}$*

$$u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq v_{it} - y\alpha_{it}$$

Moreover, for some ε, $y = \min\{v_i, \beta w_{it}/\alpha_{it} + \varepsilon\}$,

$$u_i^{bpb}(y, \{t\}, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq v_{it} - \beta w_{it}.$$

Proof. First, we will argue that with any bid $b'_i \in (\beta \frac{w_{it}}{\alpha_{it}}, v_i]$, and set of rich ads $S'_i \subseteq A_i$ such that $t \in S'_i$, i will be allocated a rich ad of value at least v_{it}. This is tricky, because changing i's bid also changes the allocation of all advertisers allocated earlier in the bang-per-buck order. However, as long as the space occupied by all ads other than i when we get to (i, t) is at most $W/2$, there is sufficient space remaining to place (i, t). Since $b'_i > \beta \frac{w_{it}}{\alpha_{it}}$, $\frac{b'_i \alpha_{it}}{w_{it}} > \beta$, and (i, t) appears before (r, j) in bang-per-buck order. Hence the space allocated to others before (i, t) is at most $W/2$. Thus the bang-per-buck algorithm will allocate at least (i, t). Suppose the algorithm allocates (i, j) instead, then $w_{ij} \geq w_{it}$ and the post processing step guarantees that $v_{ij} \geq v_{it}$ and $\alpha_{ij} \geq \alpha_{it}$. Since the GSP cost-per-click is at most the bid b'_i we get

$$u_i^{bpb}(b'_i, S'_i, \mathbf{b}_{-i}, \mathbf{S}) \geq v_{ij} - b'_i \alpha_{ij} = \alpha_{ij}(v_i - b'_i) \geq v_{it} - b'_i \alpha_{it}$$

In the last step we use that $b'_i \leq v_i$.

If $v_i < \beta \frac{w_{it}}{\alpha_{it}}$, let us consider the deviation $(b'_i, \{t\})$ then by setting $b'_i = v_i$ we get $u_i^{bpb}(b'_i, A_i, \mathbf{b}_{-i}, \mathbf{S}) \geq 0 \geq v_{it} - \beta w_{it}$. This is because the GSP cost-per-click is at most the bid $b'_i \leq \beta \frac{w_{it}}{\alpha_{it}}$.

Next we consider a deviation $b'_i, \{(t)\}$ with $b'_i = \beta \frac{w_{it}}{\alpha_{it}} + \varepsilon < v_i$, for some $\varepsilon > 0$ such that (r, j) immediately follows (i, t) in bang-per-buck order. In this case b'_i is still sufficient for i to be allocated at least (i, t). The GSP payment of (i, t) is at most $\beta \cdot w_{it}$, as GSP payment is set by the bang-per-buck of (r, j) or lower. Thus,

$$u_i^{bpb}(b'_i, \{t\}, \mathbf{b}_{-i}, \mathbf{S}) \geq v_{it} - \beta w_{it}$$ □

Now we can prove the pure price of anarchy bound. In a pure PoA proof, we can consider deviations that depend on the other players bids which allows us to obtain a tighter analysis.

Theorem 4. *With the no-overbidding assumption, the pure PoA of the mechanism that selects using ALG_I with probability $1/2$, selects the maximum value ad with probability $1/2$ and charges the GSP payment in each case is at most* 6

Proof. Consider the integer optimal allocation $IntOPT$. For each i, let $\tau(i)$ denote the rich ad selected for advertiser i in $IntOPT$. If an advertiser i is not allocated in $IntOPT$, we set $\tau(i) = 0$ which indicates the null ad. Let $(\mathbf{b}, \mathbf{S})$ denote a pure Nash equilibrium bid profile. The allocation of the mechanism is composed of two parts - bang-per-buck allocation and max-value-allocation. Let $\gamma(i)$ refer to the max-value allocation for advertiser i, but note that $\gamma(i) = 0$, i.e. the null ad, for all but one ad. Let $\rho(i)$ denote the rich ad allocated to advertiser i in the bang-per-buck allocation.

First note that for any bid $b'_i \leq v_i$ and $S'_i \subseteq A_i$, $u_i^{bpb}(b'_i, S'_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq 0$ and $u_i^{max}(b'_i, S'_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq 0$. This is because the GSP cost-per-click is always less than the bid and the bid is less than value. Thus in either mechanism if the allocated rich ad is (i, t), the utility $\alpha_{it}(v_i - cpc_i) \geq 0$.

We first bound the social welfare in the bang-per-buck allocation for advertisers that obtain an ad of space $\leq W/2$ in the optimal outcome τ. Let (r, j) be defined as in Lemma 5 and let $\beta = \frac{b_{rj}}{w_{rj}}$. By Lemma 5, for each i with $w_{i\tau(i)} \leq W/2$, there exists an ε such that with $b'_i = \min\{v_i, \beta \frac{w_{i\tau(i)}}{\alpha_{i\tau(i)}} + \varepsilon\}$, $u_i^{bpb}(b'_i, \{\tau(i)\}, \mathbf{b}_{-i}, \mathbf{S}) \geq v_{i\tau(i)} - \beta w_{i\tau(i)}$. Thus we get,

$$\begin{aligned}
(pv_{i\rho(i)} + (1-p)v_{i\gamma(i)}) &\geq u_i(\mathbf{b}, \mathbf{S}) \\
&\geq u_i(b'_i, \{\tau(i)\}, \mathbf{b}_{-i}, \mathbf{S}) \\
&\geq pu_i^{bpb}(b'_i, \{\tau(i)\}, \mathbf{b}_{-i}, \mathbf{S}) \\
&\geq pv_{i\tau(i)} - p\beta w_{i\tau(i)}
\end{aligned}$$

Here, the first inequality uses the fact that the equilibrium payment is non-negative and the second is from the equilibrium condition. The third inequality follows from the fact that utility in max-value with GSP is non-negative. The last step is the estimation we have derived.

We have the above inequality for all i such that $w_{i\tau(i)} \leq W/2$. Note that there can be at most one advertiser with $w_{i\tau(i)} > W/2$, denote this advertiser as i^*. Let $T = \mathcal{N} \setminus \{i^*\}$. Summing the above inequality over all $i \neq i^*$ we get,

$$\begin{aligned} pSW(T,\rho) + (1-p)SW(T,\gamma) &= \sum_{i \neq i^*} (pv_{i\rho(i)} + (1-p)v_{i\gamma(i)}) \\ &\geq \sum_{i \neq i^*} (pv_{i\tau(i)} - p\beta w_{i\tau(i)}) \\ &\geq pSW(T,\tau) - p\sum_{i \neq i^*} \beta w_{i\tau(i)} \end{aligned} \tag{3}$$

Case 1: First we consider the case that there is no i^* with $w_{i^*\tau(i*)} > W/2$. Starting from equation (3), we can bound $\sum_i w_{i\tau(i)} \leq W$ and use Lemma 4. We have $pSW_{bpb}(\mathbf{b},\mathbf{S}) + (1-p)SW_{max}(\mathbf{b},\mathbf{S}) \geq pSW(IntOPT) - 2p(SW_{bpb}(\mathbf{b},\mathbf{S}) + SW_{max}(\mathbf{b},\mathbf{S}))$. Setting $p = 1/2$ and rearranging, we get that the price of anarchy = $SW(IntOPT)/SW_{\mathcal{M}}(\mathbf{b},\mathbf{S}) \leq 6$.

Case 2: If there is an ad i^* with $w_{i^*\tau(i^*)} > W/2$. Then we have that $\sum_{i \neq i^*} w_{i\tau(i)} < W/2$. We will prove the following inequality for i^*.

$$pv_{i^*\rho(i^*)} + (1-p)SW_{max}(\mathbf{b},\mathbf{S}) \geq (1-p)v_{i^*\tau(i^*)} \tag{4}$$

Let i^* deviate to bid truthfully. His utility on deviation,

$$\begin{aligned} u_{i^*}(v_{i^*}, A_{i^*}, \mathbf{b}_{-i^*}, \mathbf{S}_{-i^*}) &\geq (1-p)u_{i^*}^{max}(v_{i^*}, A_{i^*}, \mathbf{b}_{-i^*}, \mathbf{S}_{-i^*}) \\ &\geq (1-p)\max_{j \in A_{i^*}} v_{i^*j} - (1-p)\max_{i \neq i^*, j \in S_i} b_{ij} \\ &\geq (1-p)v_{i^*\tau(i^*)} - (1-p)\max_{i \neq i^*, j \in S_i} b_{ij} \end{aligned} \tag{5}$$

Here the first inequality uses the fact that with bid less than value, the utility in the bang-per-buck allocation with GSP cost-per-click is non-negative. The second inequality is because i^* may not be allocated when bidding v_i^* in which case the competing bid is larger than i^*'s value.

If i^* gets allocated in the max-value algorithm in equilibrium, $u_{i^*}(\mathbf{b},\mathbf{S})$ is at most $pv_{i^*\rho(i^*)} + (1-p)v_{i^*\gamma(i^*)} - (1-p)\max_{i \neq i^*, j \in S_i} b_{ij}$. Using the equilibrium condition with $u_{i^*}(\mathbf{b},\mathbf{S}) \geq u_{i^*}(v_{i^*}, A_{i^*}, \mathbf{b}_{-i^*}, \mathbf{S}_{-i^*})$, we get $pv_{i^*\rho(i^*)} + (1-p)v_{i^*\gamma(i^*)} \geq (1-p)v_{i^*\tau(i^*)}$. Equation (4) follows because $v_{i^*\gamma(i^*)} = SW_{max}(\mathbf{b},\mathbf{S})$.

If i^* does not get allocated in the max-value algorithm in the equilibrium, then $u_{i^*}(\mathbf{b},\mathbf{S})$ is at most $pv_{i^*\rho(i^*)}$. Using $(1-p)\max_{i \neq i^*, j \in S_i} b_{ij} \leq (1-p)SW_{max}(\mathbf{b},\mathbf{S})$, with the pure Nash equilibrium condition, and rearranging we get $pv_{i^*\rho(i^*)} + (1-p)SW_{max}(\mathbf{b},\mathbf{S}) \geq (1-p)v_{i^*\tau(i^*)}$, i.e., equation (4).

Then adding equations (3) and (4), we get

$$pSW_{bpb}(\mathbf{b},\mathbf{S}) + 2(1-p)SW_{max}(\mathbf{b},\mathbf{S}) \geq p(SW(IntOPT) - v_{i^*\tau(i^*)}) + (1-p)v_{i^*\tau(i^*)} - p\beta\frac{W}{2}$$

where we use $SW_{bpb}(\mathbf{b},\mathbf{S}) = SW(T,\rho) + v_{i^*\rho(i^*)}$, $SW_{max}(\mathbf{b},\mathbf{S}) \geq SW(T,\gamma)$, $SW(IntOPT) = SW(T,\tau) + v_{i^*\tau(i^*)}$, and $\sum_{i \in T} w_{i\tau(i)} < W/2$. By Lemma 4, we have $\beta W/2 \leq SW_{bpb}(\mathbf{b},\mathbf{S}) + SW_{max}(\mathbf{b},\mathbf{S})$. Thus,

$$pSW_{bpb}(\mathbf{b},\mathbf{S}) + 2(1-p)SW_{max}(\mathbf{b},\mathbf{S})) \geq pSW(IntOPT) - pSW_{bpb}(\mathbf{b},\mathbf{S}) - pSW_{max}(\mathbf{b},\mathbf{S}).$$

Setting $p = 1/2$, we get that $6SW(\mathbf{b},\mathbf{S}) \geq SW(IntOPT)$. So the price of anarchy is at most 6. □

Bayes-Nash PoA We also provide bounds on the Bayes-Nash Price of Anarchy when our allocation rule is paired with the GSP payment rule. Our proof technique is very similar to that of [CKK+15]. We borrow ideas from [CKK+15] and combine with techniques from our pure-PoA bound to prove a smoothness inequality, and obtain a bound on the Bayes-Nash PoA.

Theorem 5. *Under a no-overbidding assumption, the mechanism that runs ALG_I with probability $1/2$ and allocates to the maximum valued ad with probability $1/2$, and charges the corresponding GSP price has a Bayes-Nash PoA of $\frac{6}{1-1/e}$.*

We will prove the bound using the smoothness framework. Our proof approach is similar to that of [CKK+15] for proving bounds on the price of anarchy of the GSP auction in the traditional position auction setting. However the knapsack constraint and the randomized allocation rule create unique challenges in our setting that we have to overcome.

We recall the definition of (λ, μ) semi-smoothness, as defined as [CKK+15], that extends [Rou15a], [NR10].

Definition 5 ((λ, μ) semi-smooth games [CKK+15])**.** *A game is (λ, μ) semi-smooth if for any bid-profile $(\mathbf{b}, \mathbf{S})$, for each player i, there exists a randomized distribution over over bids b'_i such that*

$$\sum_i \mathbb{E}_{b'_i(v_i)}[u_i(b'_i(v_i), b_{-i}, \mathbf{S})] \geq \lambda SW(OPT(\mathbf{v}, \mathbf{A})) - \mu SW(Alg(\mathbf{b}, \mathbf{S}))$$

The following lemma from [CKK+15] shows that smoothness inequality of the above form provides a bound on the Bayes-Nash POA.

Lemma 7 ([CKK+15])**.** *If a game is (λ, μ)-semi-smooth and its social welfare is at least the sum of the players' utilities, then the Bayes-Nash POA is at most $(\mu + 1)/\lambda$.*

Thus, it only remains to prove the smoothness inequality. We prove that our mechanism is $(\frac{1}{2}(1-\frac{1}{e}), 2)$ semi-smooth, and hence obtain a Bayes-Nash POA bound of $6/(1 - \frac{1}{e}) \approxeq 9.49186$.

Theorem 6. *Under a no-overbidding assumption, the mechanism that runs ALG_I with probability $1/2$ and allocates to the maximum valued ad with probability $1/2$, and charges the corresponding GSP price in each is $(\frac{1}{2}(1 - \frac{1}{e}), 2)$ semi-smooth.*

Proof. Fix a valuations profile $(\mathbf{v}, \mathbf{A})$ Consider the integer optimal allocation with valuation $(\mathbf{v}, \mathbf{A})$ as OPT. For each i, let $\tau(i)$ denote the rich-ad selected for advertiser i in OPT. $\tau(i) = 0$ be the null ad with $\alpha_{i0} = 0$ and $w_{i0} = 0$ if advertiser i is not allocated in OPT.

Let $(\mathbf{b}, \mathbf{S})$ denote a bid profile. The allocation of the mechanism is composed of two parts - bang-per-buck allocation and max-value-allocation. Let $\rho(i)$ denote the rich ad allocated to advertiser i in the bang-per-buck allocation and $\gamma(i)$ denote the rich ad allocated to the advertiser i in the bang-per-buck allocation. If an advertiser is not allocated we refer to the null ad with $\alpha_{i0} = 0$ and $w_{i0} = 0$. We will use SW_{bpb} and SW_{max} to denote the social welfare of the bang-per-buck and max-value allocation algorithms. Then,

$$SW_{\mathcal{M}}(\mathbf{b}, \mathbf{S}) = pSW_{bpb}(\mathbf{b}, \mathbf{S}) + (1 - p)SW_{max}(\mathbf{b}, \mathbf{S}) = p\sum_i v_{i\rho(i)} + (1 - p)v_{m,\mu}.$$

Most of the difficulty in proving the smoothness inequality is in reasoning about what happens in the bang-per-buck allocation. As in Lemma 4, let $k^* = \lfloor W/2 \rfloor + 1$ and $(r, j) = (i(k^*), j(k^*))$ be the rich-ad that would be allocated the k^*'th unit of space. Let $\beta = b_r\alpha_{rj}/w_{rj}$.

For any advertiser i, consider the advertiser deviates to bid y drawn from a distribution on $[0, v_i(1 - 1/e)]$ with $f(y) = 1/(v_i - y)$. Then by Lemma 6, $u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq v_{i\tau(i)} - y\alpha_{i\tau(i)}$. If $\alpha_{i\tau(i)} \cdot y < \beta w_{i\tau(i)}$, we just lower bound the utility by 0.

$$\begin{aligned}
\mathbb{E}_{y\sim F}[u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i})] &\geq \mathbb{E}_{y\sim F}[\alpha_{i\tau(i)}(v_i - y)I(\alpha_{i\tau(i)} \cdot y \geq \beta w_{i\tau(i)})] \\
&= \int_0^{v_i(1-1/e)} \alpha_{i\tau(i)}(v_i - y)I(\alpha_{i\tau(i)} \cdot y \geq \beta w_{i\tau(i)}) \cdot \frac{1}{v_i - y} dy \\
&= \alpha_{i\tau(i)}v_i(1 - 1/e) - \beta w_{i\tau(i)}
\end{aligned}$$

We have the above inequality for every i, with $w_{i\tau(i)} \leq W/2$.

Case 1: First consider the case where every advertiser has $w_{i\tau(i)} \leq W/2$ in OPT. Then we can sum over all i and use the equilibrium condition to obtain a single smoothness inequality.

$$\begin{aligned}
&\sum_i \mathbb{E}_{y\sim f} u_i(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \\
&\quad = p\sum_i \mathbb{E}_{y\sim f} u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) + (1-p)\sum_i \mathbb{E}_{y\sim f} u_i^{max}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \\
&\quad \geq p\sum_i \mathbb{E}_{y\sim f} u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \\
&\quad \geq p\sum_i [\alpha_{i\tau(i)} v_i(1-1/e) - \beta w_{i\tau(i)})] \\
&\quad \geq p(1-1/e)SW(OPT) - p\beta W
\end{aligned}$$

In the last step, we bound $\sum_i w_{i\tau(i)} \leq W$. If $p = 1/2$, by Lemma 4 we have $p\beta W \leq 2p(SW_{bpb}(\mathbf{b}, \mathbf{S}) + SW_{max}(\mathbf{b}, \mathbf{S}) = 2SW_{\mathcal{M}}(\mathbf{b}, \mathbf{S})$. And we have a smoothness inequality with parameters $(1/2(1-1/e), 2)$.

Case 2: Otherwise suppose OPT has an advertiser with $w_{i\tau(i)} > W/2$. Note that OPT can have at most one advertiser with $w_{i\tau(i)} > W/2$. Denote this advertiser as i^*. We consider the utility of an advertiser i^* as he deviates to bid y drawn from distribution f on $(0, v_{i^*}(1-1/e))$ with $f(y) = 1/(v_{i^*} - y)$. With a bid of y, the bidder will definitely be allocated in the max-value algorithm if $\alpha_{i^*\tau(i^*)} y \geq SW_{max}(\mathbf{b}, \mathbf{S})$. Note that this is a loose condition, $(i^*, \tau(i^*))$ may not be the most valuable rich ad for i^*, and if i^* is allocated in the bid profile the threshold to win might be lower than $SW_{max}(\mathbf{b}, \mathbf{S})$. But we will use this weaker condition. Again, recall that the GSP cost-per-click will be at most the bid y. Then we have,

$$\begin{aligned}
\mathbb{E}_{y\sim f}[u_{i^*}^{max}(y, A_{i^*}, \mathbf{b}_{-i^*}, \mathbf{S}_{-i^*})] &\geq \mathbb{E}_{y\sim f}[\alpha_{i^*\tau(i^*)}(v_{i^*} - y) I(\alpha_{i^*\tau(i^*)} y \geq SW_{max}(\mathbf{b}, \mathbf{S}))] \\
&\geq \int_0^{v_{i^*}(1-1/e)} \alpha_{i^*\tau(i^*)}(v_{i^*} - y)\frac{1}{(v_{i^*} - y)} I(\alpha_{i^*\tau(i^*)} y \geq SW_{max}(\mathbf{b}, \mathbf{S}))dy \\
&\geq \int_{\frac{SW_{max}(\mathbf{b},\mathbf{S}))}{\alpha_{i^*\tau(i^*)}}}^{v_{i^*}(1-1/e)} \alpha_{i^*\tau(i^*)} dy \\
&\geq (1-1/e)\alpha_{i^*\tau(i^*)} v_{i^*} - SW_{max}(\mathbf{b}, \mathbf{S})
\end{aligned}$$

We can combine all the inequalities to obtain a single smoothness inequality.

$$\begin{aligned}
&\sum_i \mathbb{E}_{y\sim f} u_i(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \\
&\quad = p\sum_i \mathbb{E}_{y\sim f} u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) + (1-p)\sum_i \mathbb{E}_{y\sim f} u_i^{max}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \\
&\quad \geq p\sum_{i\neq i^*} \mathbb{E}_{y\sim f} u_i^{bpb}(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) + (1-p)\mathbb{E}_{y\sim f} u_{i^*}^{max}(y, A_{i^*}, \mathbf{b}_{-i^*}, \mathbf{S}_{-i^*}) \\
&\quad \geq p\sum_{i\neq i^*} [\alpha_{i\tau(i)} v_i(1-1/e) - \beta w_{i\tau(i)})] + (1-p)[(1-1/e)\alpha_{i^*\tau(i^*)} v_{i^*} - SW_{max}(\mathbf{b}, \mathbf{S})] \\
&\quad \geq p(1-1/e)SW(OPT) - p\beta W/2 + (1-2p)(1-1/e)v_{i^*\tau(i^*)} - (1-p)SW_{max}(\mathbf{b}, \mathbf{S})
\end{aligned}$$

Here in the last step we bound $\sum_{i\neq i^*} w_{i\tau(i)} < W/2$.

By Lemma 4, we obtain an upper bound of $2SW_{bpb}(\mathbf{b}, \mathbf{S}) + 2SW_{max}(\mathbf{b}, \mathbf{S})$ on βW. Setting $p = 1/2$,

$$\sum_i \mathbb{E}_{y\sim f} u_i(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq \frac{1}{2}(1-\frac{1}{e})SW(OPT) - \frac{1}{2}(SW_{bpb}(\mathbf{b}, \mathbf{S}) + SW_{max}(\mathbf{b}, \mathbf{S})) - \frac{1}{2}SW_{max}(\mathbf{b}, \mathbf{S}).$$

Recall that $SW_{\mathcal{M}}(\mathbf{b}, \mathbf{S}) = \frac{1}{2}SW_{bpb}(\mathbf{b}, \mathbf{S}) + \frac{1}{2}SW_{max}(\mathbf{b}, \mathbf{S})$. Hence, the smoothness inequality

$$\sum_i \mathbb{E}_{y\sim f} u_i(y, A_i, \mathbf{b}_{-i}, \mathbf{S}_{-i}) \geq 1/2(1-1/e)SW(OPT) - 2SW(\mathbf{b}, \mathbf{S}),$$

follows, and we obtain a bound on the Bayes-Nash POA of $6/(1-1/e) \approxeq 9.49186$. □

F Proofs from Section 5

We briefly sketch a proof that the new "heuristically better" algorithms are still monotone.

Lemma 8. *GreedyByBangPerBuck is monotone in both b_i and S_i.*

Proof. Recall that in GreedyByBangPerBuck we do not stop immediately when encountering an ad that doesn't fit, but instead we will continue until we run out ads or knapsack space. We will refer to this as the bang-per-buck allocation. Finally, we do a post-processing step as usual.

Let W_i denote the space allocated to agent i before the post-processing step for each i. Since the post-processing step uses the space W_i optimally, it is enough to show that the bang-per-buck allocation is space monotone. That is, we will show that W_i is monotone in b_i and S_i (without worrying about the post-processing step). Also, wlog we can assume that the algorithm continues to consider ads in bang-per-buck order until we run out of ads (that is, if knapsack is full we will keep going without allocating anything else).

Clearly, for any $b'_i > b_i$, the advertiser will not get allocated lesser space when bidding b'_i. This is because, when reporting b'_i, all the ads of i now have higher bang per buck (than compared to reporting b_i). So any ad $j \in S_i$ that was allocated (temporarily or otherwise) under b_i will still be allocated in b'_i. In particular, the space available to j is weakly higher under b'_i.

We next show that by removing any ad $j \in S_i$ the space allocated to i does not increase. Note that, by transitivity, it is sufficient to prove monotonicity for removing one ad at a time. We see that by removing ads we can only increase the amount of space allocated to other advertisers. This is because the algorithm will continue until we run out of space. First, if j was never (temporarily) allocated by the algorithm then either j was dominated or there was not enough space to allocate j, in either case nothing changes under $S_i \setminus \{j\}$.

First consider the case where j was the final ad allocated under S_i. This implies that either j was the largest of i, or no ad $k \in S_i$ larger than j (that is, $w_{ik} > w_{ij}$) had enough space available. In either case, by removing j the space allocated to i can only be lesser, since no such larger ad k (if any) will have enough space available.

Next we consider the case where j was temporarily allocated and finally a larger ad k was allocated to i instead. Then under $S_i \setminus \{j\}$ can only have weakly lesser space available to k. To see why this is true, consider "time step" at which j was temporarily allocated. Note that j being allocated does not reduce the space available to advertiser i as subsequent rich ads of i replace j and can use the space previously allocated to j. If $w - w_{ij}$ is the space available to other agents at this time step, then by removing j there is at least $w - w_{ij}$ space available for other agents at this time step (when j is not available). Also while j being allocated does cause the algorithm to drop other rich ads of lower bang-per-buck and lower size, since the size of these rich ads is necessary smaller they cannot make more space available under $S_i \setminus \{j\}$. Hence we see that the space allocated to i is weakly lesser under $S_i \setminus \{j\}$.

□

Lemma 9. *GreedyByValue is monotone in both b_i and S_i.*

Proof. Recall that in GreedyByValue we continue allocating ads in value order (while only allocating at most one ad per advertiser) until we run out of ads. That is, we skip past ads that do not fit in the available space and continue considering ads in value order. Once an agent i get allocated we ignore all their remaining ads.

Suppose i changes her bid from b_i to $b'_i > b_i$. This implies, the value of all of i's ads increased. Therefore, her allocation under b'_i can only be better.

Similarly by dropping an ad $j \in S_i$ the allocation of i can only be worse. If j was not allocated under S_i then nothing changes because either j didn't fit (in which case we will still continue) or a better ad was allocated before j and will be allocated under $S_i \setminus \{j\}$.

If j was allocated, then j was the highest valued ad of i that fit. That is, all ads of i with higher value than j (if any) were considered before j and did not fit. Thus they will still not be allocated even under $S_i \setminus \{j\}$.

Algorithm	ApproxECPM	ApproxPayment	time-msec
GreedyByBPB-Myerson	0.9493	0.66	3.2914
GreedyByValue-Myerson	0.9196	1.00	1.6087
RandomizedGreedy-Myerson	0.9393 ± 0.0001	0.7758 ± 0.0007	2.7316 ± 0.0037
VCG	1.0	1.0	30.8287

Table 3: Average performance of the mechanisms compared to VCG. We report average approximation of eCPM and payment relative to VCG and average running time in miliseconds. We report confidence intervals for the randomized mechanism.

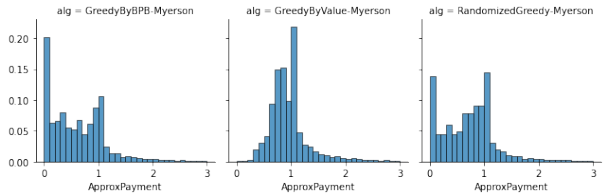

Figure 4: Histogram of ratio of revenue with MyersonPaymentRule for GreedyByValue, GreedyByBangPerBuck (GreedyByBPB) and RandomizedGreedy compared to VCG

□

G Further Empirical Evaluation

Comparison with Myerson Payment rule We compare the revenue performance of our allocation algorithms when paired with truthful payment rules.

We pair IntOPT with the VCG payment rule gives the VCG mechanism. The VCG payment for advertiser i is computed by computing the brute-force integer OPT with the advertiser i removed and subtracting from that the allocation of all advertisers other than advertiser i in the optimal allocation. For our monotone mechanisms we compute Myerson payments as implied by Lemma 1. We compute the Myerson payment for advertiser i, by first computing the GSP cost per click at the submitted bid, setting a new-bid equal to GSP cost-per-click - ϵ, and rerunning the allocation algorithm. This procedure is repeated until GSP cost-per-click or the allocation of advertiser i is equal to zero.

We first compare the revenue performance of the four different truthful mechanisms that we have. Since these mechanisms are truthful, revenue can be compared without having to reason about equilibrium. Note however that we do not set reserve prices and reserve prices can be set and tuned differently to fully compare the revenue from these mechanisms. In Figure 4, the revenue from GreedyByBangPerBuck and GreedyByValue can be both higher and lower than VCG. GreedyByBangPerBuck tends to have lower revenue on average. This is probably due to the bang-per-buck allocation — large value ads might also occupy larger space and have lower bang-per-buck. Thus, even if a large value ad is used to price smaller ads that are selected, since the bang-per-buck is small the payment for the smaller ad is still small. We might also make better trade-off between revenue and efficiency by stopping the algorithms early.

In Figure 5, we compare the running time for computing truthful payments in each mechanism. We note that all the implementations can be further optimized and the choice of programming language can influence the running time as well. The results here are from mechanisms implemented in Python. We see that GreedybyBangPerBuck(GreedyByBPB) and GreedyByValue run much faster than VCG. The greedy allocation rules themselves are much faster than brute-force OPT (see Table 1), but the truthful payment rule computation for the Greedy algorithm requires more recursive calls to the Greedy allocation rule than that for VCG, this can be further optimized if required.

Empirical results with cardinality constraint We also implemented our algorithm with the cardinality constraint. Suppose there is a limit of $k = 4$ distinct advertisers to be shown. This changes the optimization problem and the greedy-incremental-bang-per-buck algorithm no longer produces the optimal allocation. We compare the performance of simple greedy algorithms with Myerson payment rule with that of the VCG mechanism that computes the optimal allocation. The algorithm

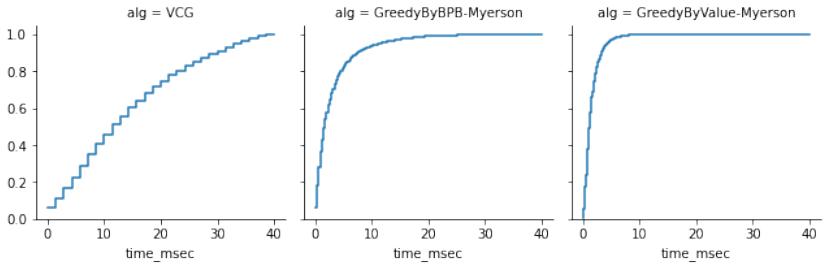

Figure 5: Histogram of ratio of running time in milliseconds for for GreedyByValue-Myerson, GreedyByBPB-Myerson and VCG mechanisms. We clip running time larger than 50 in the last bin.

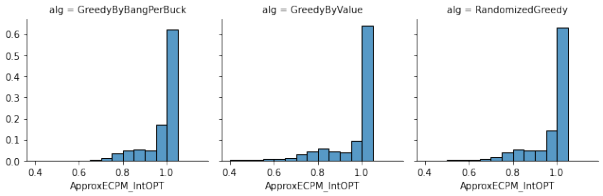

Figure 6: Approximation Factor of algorithms relative to IntOPT (a) Histogram of approximation factor for GreedyByValue, GreedyByBangPerBuck and randomized Greedy compared to IntOPT with a cardinality constraint of 4 ads

for computing optimal allocation recurses on subsets of ads and can be easily extended to track the cardinality constraint. The algorithm that allocates greedily by value of the rich-ad, will allocate the highest ad that fits within the available space and this algorithm can be stopped as as soon as k distinct advertisers have been selected. To obtain the best social welfare using the greedy by bang-per-buck heuristic, more care is required. We cannot stop as soon as k distinct advertisers are selected, instead we can improve social welfare further by replacing previously selected ads. Thus we extend our GreedyByBangPerBuck algorithm such that if the cardinality constraint is reached, it replace existing ad of the same advertiser if present (in this case the cardinality is unaffected) or replaces allocated ad of the advertiser that has the lowest value among all allocated advertisers.

In Figure 6, we compare the approximation factor of our greedy algorithms with cardinality constraint relative to the optimal integer allocation. We find that the worst-case approximation factors are still 0.6 and 0.4 for the GreedyByBangPerBuck and GreedyByValue algorithms, but 60% of the queries have an approximation of 1.0.