Auctions for Structured Procurement

Matthew C. Cary* Abraham D. Flaxman† Jason D. Hartline‡ Anna R. Karlin§

Abstract

This paper considers a general setting for structured procurement and the problem a buyer faces in designing a procurement mechanism to maximize profit. This brings together two agendas in algorithmic mechanism design, frugality in procurement mechanisms (e.g., for paths and spanning trees) and profit maximization in auctions (e.g., for digital goods). In the standard approach to frugality in procurement, a buyer attempts to purchase a set of elements that satisfy a feasibility requirement as cheaply as possible. For profit maximization in auctions, a seller wishes to sell some number of goods for as much as possible. We unify these objectives by endowing the buyer with a decreasing marginal benefit per feasible set purchased and then considering the problem of designing a mechanism to buy a number of sets which maximize the buyer's profit, i.e., the difference between their benefit for the sets and the cost of procurement. For the case where the feasible sets are bases of a matroid, we follow the approach of reducing the mechanism design optimization problem to a mechanism design decision problem. We give a *profit extraction mechanism* that solves the decision problem for matroids and show that a reduction based on random sampling approximates the optimal profit. We also consider the problem of non-matroid procurement and show that in this setting the approach does not succeed.

1 Introduction

The design of protocols for resource allocation and in the Internet among parties with diverse and selfish interests has spawned a great deal of recent research at the boundary between economics, game theory, and theoretical computer science. In many settings, a natural way to assign resources to, or obtain goods and services from, such selfish agents is by means of an *auction*, where the agents submit bids to an auctioneer, who then chooses one or more winners and purchases their services (or sells them resources).

An important recent direction in this line of research has been to show that it is possible to maximize auctioneer profit (to within a constant factor) even in worst case settings. *Digital goods auctions* [9, 8] are the canonical example in this area. These results rely crucially on the flexibility that the auctioneer has in choosing the number of items to sell. The present paper explores the question of how this kind of flexibility can improve the auctioneer's profit in more complicated settings, specifically *structured procurement auctions*.

Consider, for example, *path auctions* [16, 1, 6, 12, 4, 18]. In this setting, selfish agents own edges of a publicly known network. An agent e can transmit data along her link at some cost $c(e)$ known only to her, and the auctioneer wants to hire a team of agents whose links form a path between two given nodes s and t (so that, for example, they route data on his behalf from s to t). Each agent submits a bid, and based on these bids, the auctioneer chooses a path and pays each selected agent e some amount p_e, according to the rules of the auction. The aim of each agent is to maximize her utility, the difference $p_e - c(e)$. The aim of the auctioneer is to minimize the total payment made.

In the very special case where the network is simply a set of parallel links connecting s and t, the truthful[1] and celebrated[2] VCG mechanism [20, 3, 10] reduces to simply choosing the cheapest edge and paying that edge the cost of the second cheapest edge. On the other hand, if paths can consist of multiple edges, as in the example of Figure 1, then not only the VCG mechanism but *any* truthful mechanism may overpay greatly to buy a single path, where this overpayment is measured relative

§Department of Computer Science, University of Washington, Seattle, WA 98195. karlin@cs.washington.edu.

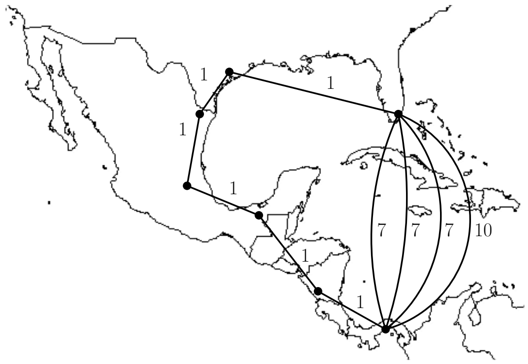
†Microsoft Research, Redmond, WA 98052. abie@microsoft.com.

‡Electrical Engineering and Computer Science, Northwestern U., Evanston, IL 60208. hartline@eecs.northwestern.edu. Work done while at Microsoft Research, Silicon Valley.

*Google Inc., Kirkland, WA 98033. cary@cs.washington.edu. Work done while at University of Washington.

[1] A desirable property in auction design is for it to be in each agent e's best interest to report her actual cost $c(e)$ as her bid, no matter how other agents bid. This property of *truthfulness* obviates the need for agents to perform complex computations or gather data about their competition, and at the same time simplifies the design and analysis of auction protocols as there is no need for assumptions about agents' knowledge of each other.

[2] Nobody ever mentions the VCG mechanism without first saying "celebrated". Just like nobody ever says Manuel Noriega without the prefix "Panamanian strongman".



k	Benefit	VCG cost	Profit
1	14	12	2
2	28	19	9
3	42	26	16
4	56	60	-4

Figure 1: Multi-Unit Procurement.

to the *second cheapest path* [1, 6, 12]. For example, in Figure 1, the leftmost path (from Florida to Panama) will be chosen by the VCG mechanism and will result in payments of 2 to each of the six edges in the path, so that the total payment (of 12) is much more than the cost of the second cheapest path. It is possible that procuring multiple paths could lower the per-path cost. In our example, buying two paths has a per-path cost of only $9^1/_2$, and three paths are even cheaper, $8^2/_3$ per path. This raises the question as to whether, like in the digital good auction problem, the freedom to choose the number of paths procured can alleviate the necessarily high over-payments in the single-path procurement problem.

To formalize this setting, let $B(k)$ be a function specifying the auctioneer's value for procuring k paths. For example, this may reflect the resale value for k paths. Then the auctioneer's profit, if he purchases k paths at a total price of P_k, is $B(k) - P_k$. One class of problems we consider in this paper is that of designing a truthful mechanism to achieve a target profit. For example, suppose the auctioneer of Figure 1, with a value of 14 per path (i.e., $B(k) = 14 \cdot k$), has a profit goal of 10. A prescient auctioneer could run the VCG mechanism specifically to procure three paths, and would make a profit of 16. The challenge is that absent foreknowledge, it is not clear how many paths to procure; and furthermore, a truthful mechanism that determines the number k of paths to procure on-the-fly will not generally be able purchase them with the same payments as the VCG mechanism for k paths.

A mechanism that solves this decision problem ("Is it possible to get a profit of R?") is called a *profit extractor* [5]. In this paper, our first contribution is to explore necessary and sufficient conditions on the structure of the procurement problem that ensure the existence of a truthful profit extractor. Profit extraction is interesting in its own right, but it is also an important subroutine in the design of mechanisms for solving the corresponding optimization problem, *profit maximization.* As in the case of the digital good auction problem (and classical optimization) a natural approach to solving an optimization problem is via reduction to the *decision problem*, that is, profit extraction.

The next contribution of this paper is to design a mechanism to maximize the auctioneers profit. For path auctions on a graph with s and t connected by a set of parallel links (or equivalently, the procurement version of the digital goods auction problem), there are known truthful reductions from the approximate profit maximization problem to the profit extraction problem. One such reduction [7] first randomly samples some of the agents to come up with an estimate R of OPT $= \max_k(B(k) - P_k)$, and then uses profit extraction on unsampled agents to try to extract a profit of R. We call an auction of this type a *random sampling profit extraction* auction. The success of this approach depends on the accuracy of the estimate OPT via random sampling, and on the existence of a truthful profit extractor. The second contribution of this paper is to identify a large class of problems for which random sampling provides a good estimate of OPT.

Combining these two results, this paper constitute a systematic development of our understanding of how broadly this paradigm for algorithmic mechanism design applies.

Results. We explore the paradigm of random sampling profit extraction auctions in the setting of a class of structured procurement problems often referred to as *hiring a team of agents* [1, 19, 12]. An auctioneer is intent on hiring a team of agents to perform a complex task. Each agent e can perform a simple task at some cost $c(e)$ known only to himself. Based on the agents' bids b_e, the auctioneer must select a *feasible set* – a set of agents whose combined skills are sufficient to perform the complex task – and pay each selected agent individually some amount p_e. In the absence of the agents' costs and bids, the problem is defined entirely by the *set system* of feasible sets. Two special cases of this have been studied extensively in the past [16, 1, 19, 6, 12, 4, 18, 2]: *path auctions*, discussed above, where the agents correspond to edges in a graph and the feasible sets are all s-t paths, and *spanning tree auctions*, where the agents again correspond to edges in a graph, and the feasible sets are spanning trees.

In this paper we generalize this procurement setting by considering the possibility of having the auctioneer purchase multiple disjoint feasible sets, obtaining a total profit equal to $B(k)$, his benefit for k sets, minus the payments he makes to procure those sets. The benchmark profit we will consider is $\mathrm{OPT} = \max_k(B(k) - \mathrm{VCG}_k)$ where VCG_k is the cost incurred by VCG for procuring k disjoint feasible sets. Our goal is to solve the mechanism design decision and optimization problems for this benchmark OPT.

Our main results are the following.

1. We give a natural profit extractor for the case that feasible sets are maximal independent sets in a matroid.

2. We show that for all set systems where feasible sets are not maximal independent sets in a matroid, this profit extraction technique does not give a truthful mechanism.

3. We show that for matroid set systems, the profit benchmark OPT, on a random sample, approximates the OPT on the full set.

4. Combining 1 with 3 we show that a random sampling profit extraction auction gives a truthful mechanism that approximates the profit benchmark OPT.

A theorem due to Karger [11] shows that if a matroid has k disjoint bases, and k is not too small, then a random sample of half the elements will have about $k/2$ bases. A significant challenge for proving 3, above, is in using this result is to show that the VCG payments on a sample are about half of what they would be in the full set. This constitutes the most technically challenging part of the paper, and requires understanding in detail the fine structure of "optimal replacements" (a.k.a., VCG payments) for unions of disjoint independent sets of a matroid. We expect that the technical lemmas that we prove in this context will be useful in tackling other problems involving matroids.

This paper is organized as follows. In Sections 2 and 3 we give preliminary definitions and review relevant material from mechanism design, procurement, and matroid theory. Our approach to profit maximization is via reduction to the decision problem. Our reduction approach is detailed in Section 2. In Section 4, we propose a solution to the decision problem, and we prove that this candidate solution does indeed solve the procurement decision problem if and only if we are trying to procure bases of a matroid. In Section 5 we prove the correctness of the random sampling based reduction for matroid procurement by showing that the optimal profit from a sample approximates that of the original set. We conclude in Section 6.

2 Mechanism Design Preliminaries

We are in a *binary single-parameter* agent setting considering *direct revelation* mechanisms. Agents correspond to elements of set $E = \{1, \ldots, N\}$. The auctioneer, or buyer, would like to purchase feasible sets from a set system $\mathcal{F}$ defined over 2^E. Let $\mathcal{F}_k$ be the set of feasible sets generated by taking the union of k disjoint feasible sets from $\mathcal{F}$. I.e., $E' \in \mathcal{F}_k$ iff $E' = \bigcup_j E'_j$ with $E'_j \cap E'_{j'} = \emptyset$ for $1 \le j, j' \le k$ and $E'_j \in \mathcal{F}$. A mechanism takes bids from each agent, $\mathbf{b} = (b_1, \ldots, b_N)$ and selects a set of winning agents S and payments $\mathbf{p} = (p_1, \ldots, p_N)$. Each agent incurs a private cost $c(e)$ of being selected, i.e., if $e \in S$, otherwise their cost is zero. We will consider only mechanisms where there are no payments made to unselected agents.[3] Each agent's objective is to maximize their *utility*, which is the difference between a payment made to them by the mechanism and their cost, i.e., $p_e - c(e)$ for $e \in S$ and zero otherwise.

A mechanism is *truthful* if each agent maximizes their utility by declaring a bid equal to their true cost irrespective of the actions of the other agents. A randomized mechanism is truthful if it is a randomization over deterministic truthful mechanisms. It is standard to show (see, e.g., [14]) that a truthful mechanism is characterized by a *threshold* that exists for each agent e when all other bids $\mathbf{b}_{-e}$ are held fixed. If e bids under this threshold, e is selected, and is paid that threshold.

The truthful Vickrey-Clarke-Groves (VCG) mechanism [20, 3, 10] is defined to:

1. Select agents: $S = \mathrm{argmin}_{S' \in \mathcal{F}} \sum_{e \in S'} c(e)$.

2. Make payments: $p_e = \min_{S' \in \mathcal{F} \,:\, e \notin S'} \sum_{e \in S'} c(e) - \sum_{e \in S \setminus \{e\}} c(e)$ for agent $e \in S$, zero otherwise.

Notice that the set that *maximizes the social welfare* is the one that minimizes the combined cost of its elements, and this is precisely the set selected by VCG. We denote the VCG mechanism that procures a set from $\mathcal{F}_k$ as VCG_k. For agents E we denote by $S_k(E)$ the cheapest cost feasible set in $\mathcal{F}_k$ (which by definition is the set procured by VCG_k) and by $\mathrm{VCG}_k(E)$ the total payments made by VCG_k.

We assume our buyer has *decreasing marginal benefit* per disjoint set from $\mathcal{F}$ procured. If $B(k)$ is the benefit for procuring k disjoint feasible sets from $\mathcal{F}$, then this assumption means that $B(\cdot)$ satisfies $B(k+1) - B(k) \le$

[3] This is a combination of the standard assumptions of *ex post individual rationality* and *no positive transfers*.

$B(k) - B(k-1)$. An interesting special case of decreasing marginal benefit is the case where the marginal benefit is constant, i.e., $B(k) = kB(1)$.

Suppose our buyer ran VCG_k on E to obtain outcome $S_k(E)$ with total payments $\mathrm{VCG}_k(E)$. Their profit would be $B(k) - \mathrm{VCG}_k(E)$. Our buyer would be especially happy if they happened to pick the k that maximized $B(k) - \mathrm{VCG}_k(E)$. This motivates Definition 2.1, below. Notice that this profit benchmark de-emphasizes frugality issues along the lines of "what is the overpayment for procuring k disjoint sets?" which has been considered extensively in algorithmic mechanism design literature (e.g., [1, 6, 12]) and places the emphasis instead on the orthogonal issue of "how do we determine how many sets to procure?" which is more in tune with the optimal auction design literature (e.g., for digital goods [9, 8]).

Definition 2.1. (OPT) *The* profit benchmark *for a set E and benefit function $B(\cdot)$ (and implicit set system $\mathcal{F}$ and agent costs $c(\cdot)$) is*

$$\mathrm{OPT}(E) = \max_k B(k) - \mathrm{VCG}_k(E).$$

We would like the mechanism to obtain a profit close to $\mathrm{OPT}(E)$ as defined above. The mechanism-design decision problem for objective OPT is to give a truthful mechanism, parameterized by a target profit R, that gives an outcome and payment with profit at least R whenever $R \leq \mathrm{OPT}(E)$. A solution to this decision problem is called a *profit extractor*. The following shows how a profit extractor can be used for the optimization problem.

Definition 2.2. (RSPE) *The* Random Sampling Profit Extraction *auction (RSPE) on E:*

1. *Randomly partition E into two parts E' and E''.*
2. *Compute the optimal benchmark on each part: $R' = \mathrm{OPT}(E')$ and $R'' = \mathrm{OPT}(E'')$.*
3. *Profit extract R'' from E' and R' from E''.*

Clearly, RSPE is truthful for bidders in E' (likewise for E'') as no bidder in E' can affect the value of $R'' = \mathrm{OPT}(E'')$ and because the profit extractor with R'' on E' is truthful. The profit of this auction is at least $\min(R', R'')$. Thus if there exists a profit extractor for OPT and the expected minimum of $\mathrm{OPT}(E')$ and $\mathrm{OPT}(E'')$ is a good approximation to $\mathrm{OPT}(E)$ then this reduction approach gives a good approximation [7].

3 Matroid Preliminaries

A *matroid* M is a set system $(E, \mathcal{I})$ such that if $I \in \mathcal{I}$, then for all $J \subset I$, $J \in \mathcal{I}$ (subset independence); and if $I, J \in \mathcal{I}$ with $|I| > |J|$, then there exists an $x \in I \setminus J$ such that $J \cup x \in \mathcal{I}$ (set augmentation). (For a comprehensive treatment, see e.g. [17]). The sets in $\mathcal{I}$ are called the *independent sets* of the matroid. A *base* of M is an independent set of maximal size. The set augmentation axiom implies that all bases are of the same size. The *rank*, $\rho(A)$, of a set of elements, $A \subseteq E$, is the size (number of elements) of the maximum independent set it contains.

It is well known that M_k, defined as the set system whose sets are the union of k disjoint independent sets in M is itself a matroid. We will abuse notation below and sometimes use M_k to denote the collection of sets in the matroid M_k. The *packing number* of a matroid M, which we will denote by K, is the maximum number of disjoint bases in M. We denote by $\rho_k(A)$ the rank of set $A \subset E$ in M_k (e.g., $\rho_1(\cdot) = \rho(\cdot)$.)

The following facts will be useful to us:

Fact 3.1. *If S and T are two independent sets of equal size in some matroid M, then there is a bijection $\pi : S \setminus T \to T \setminus S$ such that for any $e \in S \setminus T$, $(S \setminus e) \cup \pi(e)$ is an independent set of M.*

Fact 3.2. *Matroids have the* single exchange property*: if S is a minimum cost independent set of E in M of size i, then for any $e \in S$, there is a y such that $(S \setminus \{e\}) \cup \{y\}$ is a minimum cost size i independent set of $E \setminus \{e\}$ in M.*

We will use the following results due to Karger and Nash-Williams, respectively.

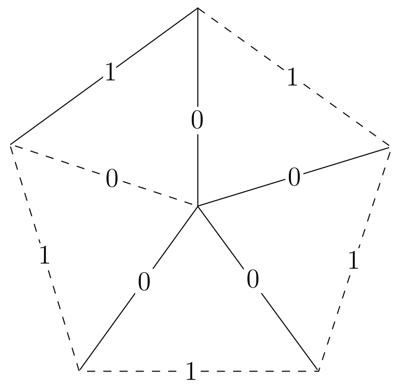
Lemma 3.1. *[11, Theorem A.7] For a matroid $M = (E, \mathcal{I})$ and set $E' \subset E$ obtained by sampling the elements of E independently with probability p, if K is the packing number of E in M, then with probability at least $1 - \rho(M) \cdot e^{-\varepsilon^2 pK/2}$ the maximum number of disjoint bases of M in E' is at least $pK(1-\varepsilon)$.*

Lemma 3.2. *[15] For any set $A \subset E$ of matroid $M = (E, \mathcal{I})$,*

$$\rho_i(A) = \min_{Y \subset A} \left[i \cdot \rho(Y) + |A \setminus Y| \right]. \qquad (3.1)$$

We will use the fact that the greedy algorithm (the algorithm which myopically adds the element of lowest cost such that the set selected remains independent) finds a base of minimum total cost. This indirectly implies a number of facts that we summarize below in Lemma 3.3. We omit the proofs.

Definition 3.1. *For an implicit matroid $M = (E, \mathcal{I})$, costs c, and any explicit set $A \subset E$ we define the following:*



The spokes of the wheel form a spanning tree, and when augmented with the rim will form two disjoint spanning trees as illustrated by the solid and dashed lines. However, the rim by itself is not a spanning tree.

Figure 2: The Wheel.

- *$S_k(A)$ is the maximal cheapest cost independent set of A in M_k,*
- *$\Delta_k(A) = S_k(A) \setminus S_{k-1}(A)$, the maximal cheapest cost set that whose union with $S_{k-1}(A)$ is independent in M_k, and*
- $\delta_k(A) = |\Delta_k(A)|$.[4]

When A is implicit from the context, we write S_k, Δ_k, and δ_k.

Lemma 3.3. *For matroid $M = (E, \mathcal{I})$ and $A \subset E$,*

1. *$S_k(A) \subseteq S_{k+1}(A)$,*
2. *$|\Delta_k(A)| \geq |\Delta_{k+1}(A)|$.*
3. *there is a decomposition of $S_k(A)$ into k disjoint independent sets $T_1, \ldots, T_k$ in M with $|T_i| = |\Delta_i(A)|$ for all i.*
4. *$\delta_k(\cdot)$ is monotone, i.e., for $A' \subset A$, $\delta_k(A') \leq \delta_k(A)$.*

Notice that it is not necessarily the case that the set Δ_k (which augments S_{k-1} to S_k) is independent. An example of disjoint spanning trees where Δ_2 is not independent is shown in Figure 2.

In Appendix A we prove the following lemma which describes more explicitly the structure of the matroid M_k as implied by Nash-Williams (Lemma 3.2). This lemma is one of the main building blocks in our subsequent proofs.

[4]Notice from this definition that $\rho_k(A) = \sum_{i=1}^{k} \delta_i(A)$.

Lemma 3.4. *For a matroid $M = (E, \mathcal{I})$, any set $A \subset E$, and any k, there exists a set Y such that:*

1. *$\rho(Y) \in \{\delta_{k+1}(A), \ldots, \delta_k(A)\}$,*
2. *$\rho_k(Y) = \rho(Y)k$ (so $\delta_i(Y) = \rho(Y)$ for $i \leq k$),*
3. *$Y \setminus S_k(Y) = A \setminus S_k(A)$ (for any costs c).*

The last condition implies that the dependent elements in M_k for Y and A are the same.

4 Profit Extraction for Procurement

The problem considered by this paper is in designing a truthful mechanism for approximating OPT. We approach this problem via reduction to the decision problem. We will consider the following algorithm as a candidate solution to the decision problem. This algorithm is a generalization of one given in [5] for the double auction problem which is based on a cost sharing mechanism due to Moulin and Shenker [13] that gives a profit extractor for the digital good auction problem [7].

Definition 4.1. (OPT-profit Extraction) *The* OPT-profit Extraction *algorithm with target R and input E works as follows:*

1. *Find the largest k such that the of $\mathrm{VCG}_k(E)$ satisfy $B(k) - \mathrm{VCG}_k(E) \geq R$.*
2. *If such a k exists, output $S = S_k$ and the VCG_k payments.*
3. *Otherwise, output $S = \emptyset$ and zero payments.*

It is easy to see that this algorithm gives a profit of at least R if and only if $R \leq \mathrm{OPT}(E)$. The following theorems show that this algorithm gives a truthful mechanism if and only if $\mathcal{F}$ is the set of bases of a matroid. The proof of the first theorem is standard, the proof of the second theorem is given in Appendix B.

Theorem 4.1. *The* OPT-*profit extractor is truthful for matroid set systems.*

Theorem 4.2. *The* OPT-*profit extraction algorithm does not give a truthful mechanism for non-matroid set systems.*

In the Appendix C we consider profit extractors for non-matroid set systems. We show that if we restrict attention to profit extractors that only extract profit when OPT is at least the target R then we can exhibit two non-matroid set systems, one for which a profit extractor exists and one for which none exists.

5 Random Sampling, Matroids, and VCG payments

As discussed in Section 2 the random sampling reduction to the decision problem requires that the value of OPT on a random sample of the elements in the ground set be close to that of the full set. In this section we prove that with high probability OPT of a random sample is a constant fraction of OPT on the full set. This shows that the Random Sampling Profit Extraction auction is a constant approximation.

The following two technical lemmas give upper and lower bounds on VCG payments, a.k.a., *replacement costs*, and enable our main theorem, below.

LEMMA 5.1. *Let $(E, \mathcal{F})$ be a set system whose feasible sets are the bases of a matroid M with packing number K. Let $m = \lfloor (1-\varepsilon)k/2 \rfloor$ for some constant $\varepsilon > 0$ and $K \geq k \geq \frac{8}{\varepsilon^2}\log n$, where $n = \rho(M)$. With probability $1 - 1/n$, the VCG_m payments for m disjoint bases in the sample E' satisfy:*

$$\mathrm{VCG}_m(E') \leq m\mathrm{c}(\Delta_k(E)).$$

LEMMA 5.2. *Let $(E, \mathcal{F})$ be a set system whose feasible sets are the bases of a matroid M with packing number K. The VCG_k payments for $k < K$ disjoint bases in E satisfy*

$$\mathrm{VCG}_k(E) \geq k \cdot \mathrm{c}(\Delta_k(E)).$$

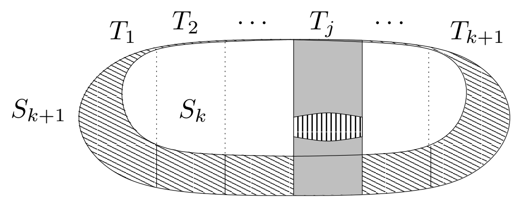
THEOREM 5.1. *Let $(E, \mathcal{F})$ be a set system whose feasible sets are the bases of a matroid M. Let k be the optimal number of bases to procure, i.e., $\mathrm{OPT}(E) = B(k) - \mathrm{VCG}_k(E)$. For any $\varepsilon > 0$, the RSPE procurement mechanism obtains a profit that is at least $\alpha = (1-\epsilon)/2$ of OPT with probability $1 - 2/n$, where $n = \rho(M)$, provided that $k \geq \frac{8}{\epsilon^2}\log n$.*

Proof. Sample to get E' and E'', and compute the optimal revenues $R' = \mathrm{OPT}(E')$ and $R'' = \mathrm{OPT}(E'')$. We claim that both R' and R'' are at least $\alpha\,\mathrm{OPT}$. To see this, observe that by Lemma 5.1, with probability at least $1 - 1/n$, $\mathrm{VCG}_{\alpha k}(E') \leq \alpha k\mathrm{c}(\Delta_k(E))$ and by Lemma 5.2, $k\mathrm{c}(\Delta_k(E)) \leq \mathrm{VCG}_k(E)$. Thus $\mathrm{VCG}_{\alpha k}(E') \leq \alpha\,\mathrm{VCG}_k(E)$. It is easy to show that if $B(\cdot)$ is marginally decreasing, then $B(\alpha k) \geq \alpha B(k)$, so with probability at least $1 - 1/n$, $R' = \mathrm{OPT}(E') \geq B(\alpha k) - \mathrm{VCG}_{\alpha k}(E') \geq \alpha B(k) - \alpha\,\mathrm{VCG}_k(E) = \alpha\,\mathrm{OPT}$. Similarly, with probability at least $1 - 1/n$, $R'' \geq \alpha\,\mathrm{OPT}$. Thus, by a union bound we see that the RSPE mechanism will obtain a profit of $\min\{R', R''\} \geq \alpha\,\mathrm{OPT}$ with probability at least $1 - 2/n$. □

For the subsequent proofs, recall that since M_k is a matroid for every k, and that the greedy algorithm finds the cheapest base of M_k, for all k simultaneously. For matroid $M = (E, \mathcal{I})$ and cost c, let $E(t)$ be the set of t cheapest elements of E. With E implicit, we extend our definition of S_k, Δ_k, and δ_k to let $S_k(t) = S_k(E(t))$, $\Delta_k(t) = \Delta_k(E(t))$, and $\delta_k(t) = \delta_k(E(t))$.

5.1 Proof of Lemma 5.1.

Lemma 3.1 shows that if we sample each element of S_k with probability $1/2$, then the sampled set will contain at least $m = \lfloor (1-\epsilon)k/2 \rfloor$ disjoint bases with high probability. The main challenge we face is to show that the VCG replacement costs for a base of M_m of in the sampled set is not too large. Our starting point is the following lemma.



This figure shows $S_k \subset S_{k+1}$. $T_1, \ldots, T_{k+1}$ partition S_{k+1}. The diagonally striped area is Δ_{k+1}. T_j is grey and $U_j = S_{k+1}\ T_j$ is the remaining area of S_{k+1}. R_j is the vertically striped area in $S_k \cap T_j$, which will be associated with $U_j \cap \Delta_{k+1}$.

Figure 3: The construction of Lemma 5.3.

LEMMA 5.3. *For matroid $M = (E, \mathcal{I})$ and cost* c*, for $A \subset E$ there are $k \cdot \delta_{k+1}(A)$ points in $S_k(A)$ whose total replacement cost is at most $k \cdot \mathrm{c}(\Delta_{k+1}(A))$.*

Proof. Fix the set A as implicit. Consider any decomposition of S_{k+1} into $k+1$ disjoint independent sets $T_1, \ldots, T_{k+1}$ of M. Define $U_j = S_{k+1} \setminus T_j$. By the decomposition of S_{k+1}, $U_j \in M_k$ for all j. Therefore, by the maximality of S_k, $|U_j| \leq |S_k|$. We will now perform $k+1$ rounds of point exchange, one between each U_j and S_k. First augment U_j with points of S_k to create U'_j with $|U'_j| = |S_k|$. Then use Fact 3.1 to associate each point of $U'_j \setminus S_k$ with a point of $S_k \setminus U'_j$. Note that $U'_j \setminus S_k \subset \Delta_{k+1}$, so that if $R_j = S_k \setminus U'_j$, each point of R_j has been replaced with an element of Δ_{k+1}. Furthermore, each point $y \in \Delta_{k+1}$ is used for replacement exactly k times, once for each j such that $y \notin T_j$.

Let $|T_j| = \delta_{k+1} + t_j$ with t_j representing the non-negative number of elements that are added to U_j to get U'_j. Let $R = \bigcup R_j$. R is precisely the elements of S_k that are not added to some U_j to get U'_j. Thus, $|R| = |S_k| - \sum_j^{k+1} t_j$. It is easy to see that this sum is exactly $k\delta_{k+1}$. Furthermore, each point $x \in R$ is involved in exactly one exchange, with U_j when $x \in T_j$. Hence the d points of Δ_{k+1} each replace k different points in R with total cost $k \cdot \mathrm{c}(\Delta_{k+1})$, as required. □

We can now get the main lemma we need to upper bound the replacement costs.

LEMMA 5.4. *For matroid $M = (E, \mathcal{I})$ and $m = \lfloor (1-\varepsilon)k/2 \rfloor$ for some constant $\varepsilon > 0$. Let t be the time that the i-th cheapest element v_i of Δ_k is added to Δ_k. For E' a random sample of $E(t)$, with probability $1 - n \cdot \exp(-\varepsilon^2 k/2)$, $S_{m-1}(E')$ has at least $(m-1) \cdot i$ elements that can each be replaced by an element of cost at most $\mathrm{c}(v_i)$.*

Proof. From the definition $i = \delta_k(t)$ and assume that $\delta_m(E') \geq i$ (we will show it shortly). Then we can apply Lemma 5.3 to show that $(m-1) \cdot \delta_m(E')$ elements of $S_{m-1}(E')$ can be replaced at total cost at most $(m-1) \cdot \delta_m(E') \cdot \mathrm{c}(v_i)$. However since none of these individual replacement costs are more than $\mathrm{c}(v_i)$, we clearly have at least $(m-1)i$ elements in $S_{m-1}(E')$ that can be replaced by an element of cost at most $\mathrm{c}(v_i)$.

Now we show that $\delta_m(E') \geq i$ with the necessary probability. Let Y be from Lemma 3.4 applied to $A = E(t)$ in M_{k-1} and let $R = S_{k-1}(Y)$. From the lemma, $\rho(R) \geq \delta_k(t) = i$ and $\rho_k(R) = k\rho(R)$. Let R' be the sampled portion of R. By applying Karger's theorem (Lemma 3.1) to the matroid $M|_{\rho(R)}$, the matroid whose bases are all independent sets of M of cardinality at most $\rho(R)$, we find that with the desired probability there is a subset S' of R' of cardinality $m \cdot \rho(R)$ that is independent in M_m. Moreover, since $\rho(S') \leq \rho(R') \leq \rho(R)$, it must be that $\rho(S') = \rho(R)$, otherwise it could not contain as many as $m \cdot i$ points that are independent in M_m. Thus, $\delta_m(S') = \rho(R) \geq i$. By the monotonicity of $\delta_k(\cdot)$ (Lemma 3.3) we have $\delta_m(E') \geq i$. □

Finally, we can put it all together to prove Lemma 5.1. Using a union bound, if $k \geq \frac{8}{\varepsilon^2} \log n$, we have that Lemma 5.4 holds with probability at least $1 - 1/n$ for all $1 \leq i \leq n$. Taking $i = n$, we have that $\mathrm{VCG}_{m-1}(E') \leq n(m-1)\mathrm{c}(v_n)$. Now considering $i = n-1$, we have at least $(n-1)(m-1)$ points in $S_{m-1}(E')$ that can be replaced with cost at most $\mathrm{c}(v_{n-1})$, so that $\mathrm{VCG}_{m-1}(E') \leq (n-1)(m-1)\mathrm{c}(v_{n-1}) + (m-1)\mathrm{c}(v_n)$. Induction shows that $\mathrm{VCG}_{m-1}(E') \leq \sum_{i=1}^{n} (m-1)\mathrm{c}(v_i) = (m-1)\mathrm{c}(\Delta_k(E))$, as required. This proves the lemma.

5.2 Proof of Lemma 5.2.

Finally, we prove our lower bound on the VCG payments of OPT.

Recall that a *circuit* in a matroid is a dependent set that is independent after removing any element, and that if a element d is dependent on an independent set A, there is a unique circuit in $\{d\} \cup A$.

LEMMA 5.5. *Let v_j be the j-th least expensive element in Δ_k. Then the total number of elements in S_k that can be replaced by elements cheaper than v_j is at most $(j-1)k$.*

Proof. Let v_j be added to S_k just after time t. Let $E(t)$ be all elements at time t, and let $D(t) = E(t) \setminus S_k(t)$. $D(t)$ is the set of possible replacement elements at time t, and each element of $D(t)$ is dependent on $S_k(t)$ in M_k. In addition, as elements are ordered in increasing cost by time, $D(t)$ is also the set of all possible replacements of cost at most that of v_j for the final S_k. Note that no element of $D(t)$ can ever replace a element of S_k added at time greater than t, as that would contradict the correctness of the greedy algorithm.

Apply Lemma 3.4 with $A = E(t)$ to get a the set Y. Let $R = S_k(Y)$. From the lemma we have $\rho(R) \leq \delta_k(t)$, $D(t) \in Y$, $D(t)$ dependent on R in M_k, and $|R| = \rho(R)k$. Observe that the only elements from $S_k(t)$ that can be replaced by an element of $d \in D(t)$ are those in R. This follows from the fact that d is dependent on R which implies it forms a unique circuit in $R \cup \{d\}$. Only the elements of this circuit can be replaced by d and no others. By our choice of t, $\delta_k(t) = j-1$. So, $D(t)$ replaces at most $|R| \leq (j-1)k$ elements of $S_k(t)$. As $D(t)$ is the set of all possible replacements with cost at most that of v_j, this proves the lemma. □

The proof of Lemma 5.2 now follows quickly. Let $v_1, \ldots, v_n$ be the elements of $\Delta_k(E)$. For any $1 \leq j \leq n$, by Lemma 5.5, at least $(n-j+1)k$ elements in $S_k(E)$ have replacement cost at least $\mathrm{c}(v_j)$. Hence we can partition $S_k(E)$ into $L_1, \ldots, L_n$, where $|L_j| = k$ and the replacement cost for any $x \in L_j$ is at least $\mathrm{c}(v_j)$. Summing over all L_j proves the lemma.

6 Conclusions

We have presented a truthful mechanism for matroid procurement that approximates the optimal profit. Our mechanism uses random sampling in conjunction with profit extraction. We have also showed that our profit extractor is not truthful for set systems which are not matroids. This leaves open an interesting question with regard to profit extraction. "What is the structural characterization of the set systems for which profit extractors exist?"

It is worth noting that positive profit maximization results for the benchmark $\mathrm{OPT} = B(k) - \mathrm{VCG}_k(E)$ are compelling for matroid problems because $\mathrm{VCG}_k(E)$ is in some sense the best possible payment for procuring k bases (due to frugality results). However, even if positive results were to exist for this benchmark for paths, they would not be as compelling, since $\mathrm{VCG}_k(E)$ can be much more than "what we would like to pay" to procure k feasible sets. In essence, even if we knew the optimal k we might still not be happy just running VCG_k. Negative results, on the other hand, would be quite strong for such a loose profit benchmark.

References

[1] Archer, A., and Tardos, E. Frugal Path Mechanisms. In *Proc. 13th Symp. on Discrete Alg.* (2002), ACM/SIAM.

[2] Bikhchandani, S., de Vries, S., Schummer, J., and Vohra, R. Linear programming and Vickrey auctions. *IMA Volume in Mathematics and its Applications, Mathematics of the Internet: E-auction and Markets 127* (2001), 75–116.

[3] Clarke, E. H. Multipart Pricing of Public Goods. *Public Choice 11* (1971), 17–33.

[4] Czumaj, A., and Ronen, A. On the expected payment of mechanisms for task allocation. In *Principles of Distributed Computing* (2004), ACM.

[5] Deshmukh, K., Goldberg, A., Hartline, J., and Karlin, A. Truthful and Competitive Double Auctions. In *Proc. 10th European Symposium on Algorithms* (2002), Springer–Verlag.

[6] Elkind, E., Sahai, A., and Steiglitz, K. Frugality in path auctions. In *Proc. 15th Symp. on Discrete Alg.* (2004), ACM/SIAM.

[7] Fiat, A., Goldberg, A., Hartline, J., and Karlin, A. Competitive Generalized Auctions. In *Proc. 34th ACM Symposium on the Theory of Computing* (2002), ACM Press, New York.

[8] Goldberg, A., Hartline, J., Karlin, A., Saks, M., and Wright, A. Competitive auctions. *Games and Economic Behavior 55* (2006), 242–269.

[9] Goldberg, A., Hartline, J., and Wright, A. Competitive Auctions and Digital Goods. In *Proc. 12th Symp. on Discrete Alg.* (2001), ACM/SIAM, pp. 735–744.

[10] Groves, T. Incentives in Teams. *Econometrica 41* (1973), 617–631.

[11] Karger, D. R. Random sampling and greedy sparsification for matroid optimization problems. *Math. Programming 82*, 1-2, Ser. B (1998), 41–81. Networks and matroids; Sequencing and scheduling.

[12] Karlin, A., Kempe, D., and Tamir, T. Beyond VCG: Frugality in truthful mechanisms. In *Proc. of the 46th IEEE Symposium on Foundations of Computer Science* (2005).

[13] Moulin, H., and Shenker, S. Strategyproof Sharing of Submodular Costs: Budget Balance Versus Efficiency. *Economic Theory 18* (2001), 511–533.

[14] Myerson, R. Optimal Auction Design. *Mathematics of Operations Research 6* (1981), 58–73.

[15] Nash-Williams, C. S. J. A. An Application of Matroids to Graph Theory. In *Theory of graphs: International Symposium*, vol. 1966. Gordon & Breach, New York, 1967, pp. 264–265. Held at the International Computation Center in Rome, July.

[16] Nisan, N., and Ronen, A. Algorithmic Mechanism Design. In *Proc. of 31st Symposium on Theory of Computing* (1999), ACM Press, New York, pp. 129–140.

[17] Oxley, J. G. *Matroid Theory.* Oxford Science Publications. The Clarendon Press, Oxford University Press, New York, 1992.

[18] Ronen, A., and Tallisman, R. Towards generic low payment mechanisms for decentralized task allocation – a learning based approach. In *In proceedings of* The 7th International IEEE Conference on E-Commerce Technology (2005).

[19] Talwar, K. The Price of Truth: Frugality in Truthful Mechanisms. In *Proc. of 20th International Symposium on Theoretical Aspects of Computer Science* (2003), Springer.

[20] Vickrey, W. Counterspeculation, Auctions, and Competitive Sealed Tenders. *J. of Finance 16* (1961), 8–37.

A Proof of Lemma 3.4

Take Y to be a set which minimizes (3.1) for A (from Lemma 3.2). We prove the three parts of the lemma separately.

1. $\rho_k(Y) = k \cdot \rho(Y)$ (part 2 of lemma).

   If $Y = \emptyset$, then $\rho_k(A) = |A|$ and $A \in M_k$. If Y is nonempty, we argue that $\rho_k(Y) = k\rho(Y)$ by contradiction.

   If $\rho_k(Y) < k\rho(Y)$, then let T be a set which minimizes (3.1) for Y, and so we have $k\rho(T) + |Y \setminus T| < k\rho(Y)$. As $|A \setminus T| = |A \setminus Y| + |Y \setminus T|$, we have that $k\rho(T) + |A \setminus T| < k\rho(Y) + |A \setminus Y|$, contradicting the minimality of Y.

   If $\rho_k(Y) > k\rho(Y)$, then Y contains an independent set $U \in M_k$ of size exceeding $k\rho(Y)$. By definition, this means U can be decomposed into k disjoint independent sets of M with average size exceeding $\rho(Y)$. Thus there must exist a $Z \subset Y$ with $\rho(Z) > \rho(Y)$, a contradiction.

2. Let $D = A \setminus S_k(A)$ then $D = Y \setminus S_k(Y)$ (part 3 of lemma).

   Let $S = S_k(A)$, the maximal cheapest independent set in M_k, and $D = A \setminus S$, the set of dependent elements in M_k. Recall that $|S| = \rho_k(A)$ and add $|Y| - |S|$ to both sides of equation (3.1).

$$\begin{aligned} |Y| &= k\rho(Y) + |A| - |S| \\ &= k\rho(Y) + |D| \,. \end{aligned} \tag{1.2}$$

   Clearly,

$$|Y| = |S \cap Y| + |D \cap Y| \tag{1.3}$$

   It is easy to see that $|S \cap Y| \leq k\rho(Y)$ as

$$|S \cap Y| = \rho_k(S \cap Y) \leq \rho_k(Y) = k\rho(Y).$$

   Of course, $|D \cap Y| \leq |D|$. Therefore, the only way that equations (1.2) and (1.3) can hold is for $D = D \cap Y$ and $|S \cap Y| = k\rho(Y)$ (which also implies that $S \cap Y = S_k(Y)$).

3. $\rho(Y) \in \{\delta_{k+1}(A), \ldots, \delta_k(A)\}$ (part 1 of lemma).

   First, $\rho(Y) \geq \delta_{k+1}(A)$. To see this, recall that $S_{k+1}(A)$ can be partitioned into trees $T_1, \ldots, T_{k+1}$ with $|T_i| = \delta_i(A)$. From the previous arguments we can divide A in to S and D with $S = T_1 \cup \cdots \cup T_k$ and independent in M_k and D equal to the set of elements dependent on S in M_k. Clearly, then $T_{k+1} \subset D$. As shown above, $D \subset Y$ which implies that $\rho(Y) \geq \rho(D) \geq |T_{k+1}| = \delta_{k+1}(A)$.

   Second, $\rho(Y) \leq \delta_k(A)$. The above arguments imply that $\rho(Y) = \delta_k(Y)$. The fact that $\delta_k(A) \geq \delta_k(Y)$ follows from the monotonicity of $\delta_k(\cdot)$ (Lemma 3.3).

This completes the proof.

B Proof of Theorem 4.2

The theorem states that for any set system that is not a matroid and any marginally decreasing $B(\cdot)$, there is a set of private values c and a choice of R for which the profit extractor is not truthful. We first establish two claims, the proofs of which can be found in the full paper. Here we will be considering running the profit extraction algorithm on instances of set system with different agent costs. Let $S_k(E, \mathrm{c})$ (resp. $\mathrm{VCG}_k(E, \mathrm{c})$) be the cheapest cost feasible set (resp. VCG payments) for $\mathcal{F}_k$ for elements E and cost c.

Claim B.1. *There is a cost vector c, feasible sets A and B and distinct elements e and u in $A \setminus B$ such that $S_1(E, c) = A$ and $S_1(E \setminus u, c) = B$ for some integer cost vector c, with $e \notin B$. In other words, the best replacement set for u replaces e as well.*

Before the next claim, we modify the cost vector c. Let S be a union of two disjoint feasible sets that minimizes $|S \setminus (A \cup B)|$. Raising the costs of elements outside of $A \cup B \cup S$ does not change the properties of A and B, so we may change c so that the cost of any such element is very large; $C = 1 + K \cdot \big(|A| \cdot c(S) + c(A \cup B \cup S)\big)$ will suffice. The the following holds.

Claim B.2. *Under the cost vector c, $\mathrm{VCG}_k > k \cdot \mathrm{VCG}_1$ for all $k > 1$. In addition, $A = S_1(E)$ and $S = S_2(E)$.*

Armed with the cost vector from these two claims, we can now contradict truthfulness by showing that e can raise its bid and cause VCG_1 to go down. We will choose the revenue goal and benefit function so that if all elements bid truthfully, the buyer can nearly but not quite meet the goal. In particular, e will receive no utility from the canceled auction. However, e's overbidding and subsequent reduction in VCG_1 will cause the buyer to meet the goal at $k = 1$, and provide e with positive utility, and thus incentive to bid non-truthfully.

Note that the chosen cost vector is integral, and that $c(e)$ is at least one less than its threshold to be included in the optimal set. Hence if e bids $c(e) + 1/2$, e will remain in the optimal set. Let c' be this cost vector, that is, $c'(e') = c(e')$ for $e' \neq e$ and $c'(e) = c(e) + 1/2$. Furthermore, as the bid of e is increased, e will still be excluded from $S_1(E \setminus u, c')$. Define $p_1(v, c)$ to be the VCG_1 payment to v under c. Then $p_1(u, c) - p_1(u, c') = \big(c(B) - c(A) + c(u)\big) - \big(c'(B) - c'(A) + c(u)\big) = -1/2$, so that the payment to u decreases. For any other element $v \in A$, the threshold for e to be in $S_1(E \setminus v, c)$ is an integer, so $e \in S_1(E \setminus v, c)$ if and only if $e \in S_1(E \setminus v, c')$, and so $p_1(v, c) - p_1(v, c')$ is either 0 or $-1/2$. Summing these payment differences over all $v \in A$ thus gives $\mathrm{VCG}_1(c') \leq \mathrm{VCG}_1(c) - 1/2$.

Now choose $L_0 > 1$ such that $\mathrm{VCG}_1(c') < L_0$ and $L_0 \cdot k < \mathrm{VCG}_k(c)$ for all $k \geq 1$; such an L_0 exists as c is an integer cost vector and $\mathrm{VCG}_k(c) > k \cdot \mathrm{VCG}_1(c)$ for all $k > 1$. If the resale function were $B_0(k) = L_0 \cdot k$, then choosing a revenue target R as $R = B_0(k) - \mathrm{VCG}_1(c) + 1/4$ would suffice. If all elements bid their values, revenue R cannot be extracted, as $B(k) - \mathrm{VCG}_k(c) < 0$. On the other hand, e raises her bid to $1/2$, then revenue at least $R + 1/4$ can be extracted at $k = 1$. Hence there is incentive for e to overbid. Given $B(\cdot)$ from the lemma of the statement, choose L so that $B(k) \leq L \cdot k$; such an L exists as $B(\cdot)$ is marginally decreasing. Then scaling the costs of all elements by L/L_0 gives the required cost vector. This completes the proof.

C Profit Extraction on Non-Matroids

Define a *strict* profit extractor to be one that extracts profit R if and only if $R \leq \mathrm{OPT}$ (usually the "only if" is not required.

LEMMA C.1. *There exists non-matroid set system for which there is a truthful strict profit extractor.*

Proof. Define a set system $\mathcal{F} = \{\{e, f\}, \{g\}\}$. Note that it is possible to procure at most one set from $\mathcal{F}$ with the VCG mechanism, so that the benefit function is expressed simply as a single number B. Given target profit R, our proposed profit extractor buys from agent g if an only if $\mathrm{OPT}(\{e, f, g\}) \geq R$. This gives two constraints on the region of allocation:

1. When $c(e)+c(f) > c(g)$ then OPT meets the target R when $c(e) + c(f) < B - R$.

2. When $c(e)+c(f) < c(g)$ then OPT meets the target R when $c(g) \leq \frac{1}{2}(c(e) + c(f) + B - R)$.

Notice that this region is monotone for g (See Figure 4). That is if for some values of $c(e)$, $c(f)$, and $c(g)$, the mechanism buys from g then for lower values of g the mechanism continues to buy from g. Furthermore, the threshold bid for g is given by $(c(e) + c(f) + B - R)/2$.

By construction this mechanism allocates if and only if $\mathrm{OPT}(\{e, f, g\}) \geq R$. It remains to show that the our profit is at least R whenever we allocate. The payment of g when we allocate is $p_g = (c(e) + c(f) + B - R)/2$. We show that $B - p_g > R$ whenever we are in the region of allocation.

$$\begin{aligned} B - p_g &= B - \tfrac{1}{2}(c(e) + c(f) + B - R) \\ &= \tfrac{1}{2}(B - c(e) - c(f) + R) \end{aligned}$$

However, in the region of allocation $c(e)+c(f) < B-R$ so $R < B - c(e) - c(f)$, so $B - p_g \geq R$. □

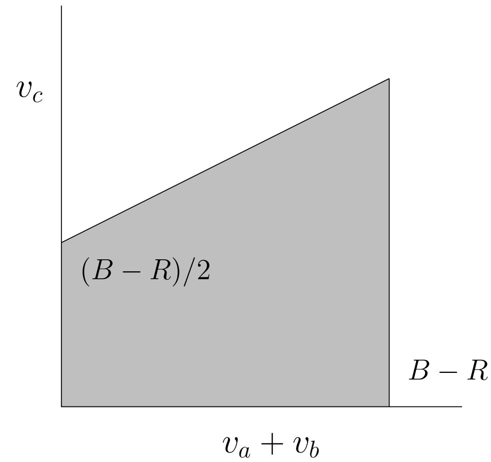


Figure 4: Allocation region for Lemma C.1.

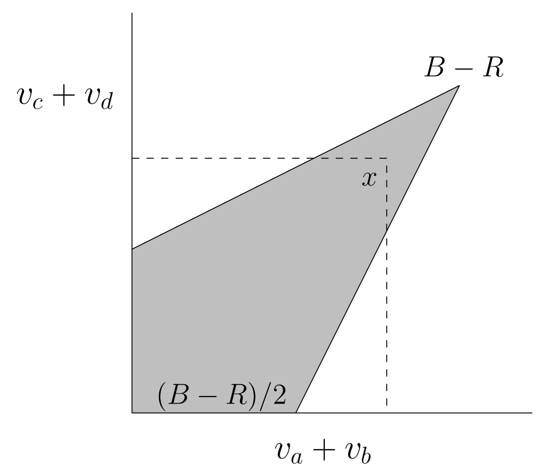


Figure 5: Allocation region for Lemma C.2.

LEMMA C.2. *There exists a non-matroid set system and benefit such that there is no truthful strict profit extractor for* OPT.

Proof. Let $\mathcal{F} = \{\{e, f\}, \{g, h\}\}$. As before, it is only possible for the VCG mechanism to procure one set. If $c(e)+c(f) < c(g)+c(h)$, then $\mathrm{VCG}_1 = 2(c(g)+c(h)) - (c(e)+c(f))$, and conversely for $c(g)+c(h) < c(e)+c(f)$. A strict profit extractor must produce revenue over the allocation region defined by $B - 2(c(g)+c(h)) + (c(e) + c(f)) > R$ and $B - 2(c(e) + c(f)) + (c(g) + c(h)) > R$, as shown in Figure 5.

Consider the point x at $\left(\frac{5}{6}(B-R), \frac{5}{6}(B-R)\right)$. It cannot be allocated to $\{g, h\}$, as then the allocation would not be monotone as $c(g)+c(h)$ varies, and neither can it be allocated to $\{e, f\}$. Since x is in the region of allocation, no truthful strict extractor exists. □