Investment Incentives in Near-Optimal Mechanisms*

Mohammad Akbarpour† Scott Duke Kominers‡
Shengwu Li§ Paul Milgrom¶

First posted: February 25, 2020
This version: March 14, 2021

Abstract

In many real-world resource allocation problems, optimization is computationally intractable, so any practical allocation mechanism must be based on an approximation algorithm. We study investment incentives in strategy-proof mechanisms that use such approximations. In sharp contrast with the Vickrey-Clark-Groves mechanism, for which individual returns on investments are aligned with social welfare, we find that some algorithms that approximate efficient allocation arbitrarily well can nevertheless create misaligned investment incentives that lead to arbitrarily bad overall outcomes. However, if a near-efficient algorithm "excludes bossy negative externalities," then its outcomes remain near-efficient even after accounting for investments. A weakening of this "XBONE" condition is necessary and sufficient for the result.

Keywords: Combinatorial optimization, Knapsack problem, Investment, Auctions, Approximation, Algorithms

JEL classification: D44, D47, D82

*We thank Moshe Babaioff, Peter Cramton, Michal Feldman, Matthew Gentzkow, Paul Goldsmith-Pinkham, Nima Haghpanah, Andy Haupt, John William Hatfield, Emir Kamenica, Zi Yang Kang, Eric Maskin, Ellen Muir, Rad Niazadeh, Noam Nisan, Amin Saberi, Mitchell Watt, and numerous seminar audiences for helpful comments. We thank Kevin Michael Li for his truly superb contributions as a research assistant. We thank Broadsheet Cafe for inspiration and coffee. Akbarpour and Kominers gratefully acknowledge the support of the Washington Center for Equitable Growth. Additionally, Kominers gratefully acknowledges the support the National Science Foundation (grant SES-1459912) and both the Ng Fund and the Mathematics in Economics Research Fund of the Harvard Center of Mathematical Sciences and Applications. Milgrom gratefully acknowledges support from the National Science Foundation (grant SES-1947514). All errors remain our own.

†Stanford University. Email: mohamwad@stanford.edu

‡Harvard University. Email: kominers@fas.harvard.edu

§Harvard University. Email: shengwu_li@fas.harvard.edu

¶Stanford University and Auctionomics. Email: milgrom@stanford.edu

1 Introduction

Many real-world allocation problems are too complex for exact optimization. For example, it is computationally difficult—even under full information—to optimally pack indivisible cargo for transport (Dantzig, 1957; Karp, 1972), coordinate electricity generation and transmission (Lavaei and Low, 2011; Bienstock and Verma, 2019), assign radio spectrum broadcast rights subject to legally-mandated interference constraints (Leyton-Brown et al., 2017), or find value-maximizing allocations in combinatorial auctions (Sandholm, 2002; Lehmann et al., 2006b).

Computational difficulty, however, does not obviate the need to solve allocation problems in practice. Instead, it has inspired research to identify fast approximation algorithms for allocation problems, as well as associated payment schemes that provide incentives for participants to report their input values truthfully. In the language of textbook economics, that research focuses on *short-run* analyses: it takes the values and resource constraints as fixed, omitting *long-run* considerations about parties' incentives to invest to create new assets and/or improve existing ones—or disinvest to cash in less valuable assets. But in resource allocation problems, investments can affect both what is feasible (such as when a larger plane is more difficult to schedule on a runway) and the values of the items being allocated (because a larger plane carries more passengers).

A bidder's incentives to make investments are shaped by the rules of the resource allocation mechanism. For instance, suppose a bidder can pay a cost to raise his value before participating in a Vickrey-Clarke-Groves (VCG) mechanism. In a VCG mechanism, if we raise one bidder's value while holding all else fixed, the change in that bidder's utility is equal to the change in social welfare. Thus, the VCG mechanism aligns the bidder's self-interest with social welfare: the investment is privately profitable if and only if it is socially optimal (Rogerson, 1992).

The efficiency of investments under VCG is already subject to several important caveats: First, it presumes that the bidder can forecast the prices he will face in the VCG mechanism. If a bidder mis-predicts prices, he may make investments that are *ex post* inefficient. Second, if multiple bidders make simultaneous investment decisions, there can be multiple equilibria of the investment game, some of which have inefficient investments.

But a novel problem arises in settings with computational difficulties: Finding VCG allocations and prices requires solving multiple optimizations, which are infeasible in problems we have described. This challenge is the main subject of this paper. As we show, mechanisms based on feasible algorithms can sometimes misalign participants' investment incentives with social welfare. This misalignment can occur even under full information and even when only one bidder may invest.

To illustrate misaligned incentives, consider the classic knapsack problem, in which there is a knapsack of fixed capacity and several indivisible bidders. Each bidder has a size and a value, and our aim is to maximize the sum of the values of packed bidders subject to the sum of their sizes not exceeding the knapsack's capacity. The auctioneer can see the sizes of the bidders but not their values, so she uses a truthful auction to elicit bidders' values for being packed, which she then takes as inputs to her algorithm. Since the knapsack problem is NP-hard, the auctioneer applies a fast algorithm—in this example, Dantzig's GREEDY algorithm—to the bids and sizes to determine which bidders to pack. This algorithm sorts bidders in decreasing order of value-per-unit-size and packs bidders in that order, designating them as "winning bidders" in the auction. The algorithm terminates when it encounters a bidder who does not fit. The associated truthful auction is a *threshold auction*. This means that each winning bidder pays an amount equal to his *threshold price*, which is the lowest value the bidder could report to win a space in the knapsack, given the bids of the other bidders; each losing bidder pays zero.

Suppose that the knapsack has capacity 20 and there are three bidders, whose values are 11, 11, and 12 and whose sizes are 10, 10, and 11. Since $\frac{11}{10} > \frac{12}{11}$, the GREEDY algorithm packs the first two bidders for a total value of 22, which is also the optimal packing. Next, we add an investment stage. Suppose that before the auction, the third bidder has an opportunity to increase his value from 12 to 14 at a cost of 1. From the bidder's perspective, the investment can be assessed like this: "If I invest, my value will be 14 and I will be packed. In fact, any value over 12.1 would result in my being packed ($\frac{11}{10} = \frac{12.1}{11}$), so 12.1 is my threshold price. If I invest, I will pay that threshold price of 12.1 plus my investment cost of 1, but my total cost of 13.1 is less than my value of 14 for a place in the knapsack. That's a good deal! I should invest." From a social welfare perspective, the investment is assessed differently. If the bidder invests, the packed value will be 14 and an investment cost of 1 will be incurred, for a welfare of just 13. With no investment, welfare would be 22, so the investment reduces welfare substantially.

We can extend the analysis of investment incentives to other allocation algorithms. Indeed, for any algorithm, there exists a pricing rule that incentivizes truthful reporting if and only if that algorithm is monotone (Lehmann et al., 2002; Saks and Yu, 2005). A simple application of the envelope theorem implies that, for each algorithm, any two payment rules for a bidder that make the associated mechanism strategy-proof differ by the addition of a term that does not depend on the bidder's values. It follows that any two strategy-proof mechanisms using the same allocation algorithm provide the same investment incentives, so we may regard investment incentives as depending on the algorithm alone. While we study general mechanisms in later sections, this Introduction focuses mostly on allocation problems in which each bidder is "packed" or not, so that the relevant mechanism can be described as a threshold auction.

We study a "long-run" resource allocation game consisting of two stages. In the first stage, one of the bidders can make a costly investment with foreknowledge of his threshold price.[1] Then, the second stage uses a strategy-proof mechanism based on some monotone allocation algorithm.

In this two-stage game, VCG mechanisms have the *efficient investment* property, meaning that for any investment technology, the net private returns to any investment are equal to net social welfare returns.[2] Not all strategy-proof mechanisms share this property. For instance, in the earlier example of the Greedy threshold auction, the third bidder's threshold price is 12.1, even though packing him results in a welfare loss of 22 for the other bidders.

So when do algorithms lead to strategy-proof mechanisms with the efficient investments property? We find that any strategy-proof mechanism with the efficient investments property is welfare-equivalent to a mechanism that maximizes social welfare over a (possibly restricted) set of allocations. Using any allocation algorithm outside this class creates *algorithmic externalities*, that is, when a bidder changes his value, the algorithm prescribes a different allocation, but the impact on other bidders' welfare is different from the change in the bidder's price.

Our benchmark result shows that any algorithm that ensures efficient investments must be able to maximize total welfare on a restricted set. In practice, state-of-the-art approximation algorithms often do not have this property, so efficient investments cannot be ensured. That fact leads us to adopt a combined standard for algorithm performance that weighs equally losses from failures of the algorithm to optimize and from any inefficient investments it may promote. We focus on worst-case performance in the two-stage game with investments — a measure that is compatible with the known performance guarantees for approximate allocation algorithms developed by computer scientists.[3]

Formally: fix any approximation algorithm that always delivers at least a fraction $\beta \in (0, 1)$ of the maximum welfare in a short-run allocation problem. The worst-case guarantee in the associated two-stage game with investments can never exceed β, because the investment technology might include no profitable positive-cost investments for the bidder.

[1] In the case of general mechanisms, we assume that bidders have foreknowledge of their prices.

[2] Net social welfare is the sum of the bidders' values for the outcome minus the investment cost, plus or minus a value or cost for the auctioneer.

[3] In the computer science literature, worst-case performance is the standard criterion for evaluating algorithms, for both approximations and run times. Indeed, complexity classes such as P and NP are defined with respect to worst-case run times (Cook, 1971). See Williamson and Shmoys (2011) or Vazirani (2013) for an overview of approximation algorithms.

But when is the guarantee for the two-stage game no worse than in the short-run problem? Specifically, if some *single bidder* has an investment opportunity, when does the *same* β-guarantee apply—for all investment technologies—in the corresponding two-stage game? A crucial step in answering these questions is to distinguish two kinds of algorithmic externalities: *threshold externalities* and *bossy externalities*.

Threshold externalities can arise when a bidder's value just crosses a threshold at which the bidder's own allocation changes. This crossing will typically change the bidder's payment and the allocations of other bidders. When the change in the bidder's payment is different from the change in the total value of others' allocations, that is a threshold externality. When there is a threshold externality, the algorithm's performance changes discontinuously at the threshold and the bidder's investment incentives there are not aligned with maximizing efficiency. The knapsack example offered earlier illustrates a threshold externality; the third bidder faces a threshold price of 12.1, but packing him reduces the welfare of the other bidders by 22, which leads him to make an inefficient investment.

The second kind of externality is bossy, arising when the change in a bidder's value does not alter his own allocation or price but does change the total value of others' allocations. An algorithm has a bossy *negative* externality if raising the value of a packed bidder or reducing the value of an unpacked bidder can decrease the total value of the allocation for the other bidders.[4] (Recall that every strategy-proof mechanism is monotone, which means that such a change in a bidder's value cannot change his own allocation.)

Our main result is that only bossy negative externalities can reduce an algorithm's worst-case investment guarantee. We prove this by decomposing the effect of any investment into the effects of crossing a threshold and moving beyond it. We then show that any loss caused by a threshold externality is already incorporated into the allocative guarantee β of the algorithm. Any bossy positive externality from an investment can only improve performance. So, if an algorithm excludes bossy negative externalities—a property we call *XBONE*—then the performance guarantee β' for its associated two-stage game with investment—the "long-run" problem—cannot be worse than the algorithm's guarantee β for the "short-run" allocation problem.

However, if an algorithm is not XBONE, its outcomes including investment can be much worse than its guarantee for the allocation problem. We show that for any $\beta < 1$, there exists a simple "satisficing" algorithm that attains the guarantee β in the short-run allocation problem, but has performance guarantee 0 in the two-stage game.

[4] We adapt the concept of a negative externality to approximate mechanisms as follows: An investment by bidder n that increases the value of his allocation by Δv_n and his price by Δp_n while reducing the total value allocated to others by Δv_{-n} has a *negative externality* if $\Delta v_{-n} - \Delta p_n > 0$.

Some standard allocation algorithms are XBONE, such as the Greedy algorithm, as well as the class of clock auction algorithms of Milgrom and Segal (2020), which includes the algorithm used for the 2017 Federal Communication Commission's broadcast incentive auction (Leyton-Brown et al., 2017).

Meanwhile, we establish that two textbook approximation algorithms are not XBONE: The canonical *Fully Polynomial-Time Approximation Scheme* for the knapsack problem, and the minimum spanning tree algorithm for the Steiner tree problem.

Worst-case performance is a pessimistic standard that may rely on unrealistically extreme examples. In practical applications, the potentially relevant instances share some special structure that leads to much better performance. Our theory accommodates and takes advantage of special structure using the concept of *sub-problems*, which are certain well-behaved subsets of the full problem. We show that if an algorithm is XBONE, then its long-run guarantee on every sub-problem is equal to its short-run guarantee on that same sub-problem. For example, in the knapsack problem, the Greedy algorithm generally has only a 0 worst-case guarantee, but for the sub-problem in which the item sizes are never more than a fraction α of the knapsack capacity, the short-run performance guarantee is $1 - \alpha$. The Greedy algorithm is XBONE, so for any investment technology, its long-run performance on this sub-problem is at least $1 - \alpha$.

Our XBONE condition is sufficient to preserve efficiency guarantees under investment, but it is not quite necessary. For a necessary condition, we say that an algorithm is *weakly XBONE* if it allows no bossy negative externalities except those arising from value decreases beginning below a bidder's Vickrey price. This yields a characterization theorem: An algorithm is weakly XBONE *if and only if* for every subproblem, its worst-case investment performance is the same as its worst-case allocation performance. When optimization is difficult, Vickrey prices are hard to compute and analyze, so we expect that the XBONE sufficient condition will often be easier to check.

We also briefly examine the case in which all bidders make simultaneous investment decisions. VCG mechanisms have the property that when all bidders can invest, there exists an equilibrium of the two-stage game in which the investment choices are efficient. We show similarly that for any non-bossy algorithm, there exists an equilibrium of the two-stage game that shares the algorithm's performance guarantee. However, some XBONE algorithms can fail to have such an equilibrium: with multiple investors, the performance guarantee can be worse in every equilibrium of the game with investments.

Our last finding concerns combinatorial auctions in which bidders' values for bundles are restricted (for tractability) to be fractionally subadditive. For that case, we show that if the investment cost function is isotone and supermodular, then for any XBONE algorithm, the long-run performance guarantee is again equal to the short-run performance guarantee.

1.1 Related work

Economists have studied *ex ante* investment in mechanism design at least since the work of Rogerson (1992), who demonstrated that Vickrey mechanisms induce efficient investment. Bergemann and Välimäki (2002) extended this finding in a setting with uncertainty, in which agents invest in information before participating in an auction. Relatedly, Arozamena and Cantillon (2004), studied pre-market investment in procurement auctions, showing that while second-price auctions induce efficient investment, first-price auctions do not. Hatfield et al. (2014, 2019) extended these findings to characterize a relationship between the degree to which a mechanism fails to be strategy-proof and/or efficient and the degree to which it fails to induce efficient investment. While like us, Hatfield et al. (2014, 2019) deal with the connection between (near-)efficiency at the allocation stage and (near-)efficiency at the investment stage, they use additive error bounds, rather than the multiplicative worst-case bounds that are standard for the analysis of computationally hard problems.[5]

Our paper is also not the first work to study investment incentives in an NP-hard allocation setting. Milgrom (2017) introduced a "knapsack problem with investment" in which the items to be packed are owned by individuals, and owners may invest to make their items either more valuable or smaller (and thus easier to fit into the knapsack). In the present paper, we reformulate the investment question in terms of worst-case guarantees and broaden the formulation to study incentive-compatible mechanisms for a wide class of resource allocation problems.

Lipsey and Lancaster (1956) explained that in economic systems that are not fully optimized, investments that violate optimality conditions can sometimes improve welfare by offsetting other distortions of the system. Our question is related to that of Lipsey and Lancaster (1956), but leads to a different analysis. We isolate *bossy* negative externalities as the only externalities that can degrade an allocation algorithm's long-run performance guarantee relative to its short-run guarantee. Other externalities associated with failures of optimization cannot have that effect.

By studying the investment problem in near-optimal mechanisms, our paper is naturally connected to a large literature, primarily in computer science, that considers computational complexity in mechanism design, and explores properties of approximately optimal mechanisms. Among these works are those of Nisan and Ronen (2007) and Lehmann et al. (2002). Nisan and Ronen (2007) showed that in settings where identifying the optimal allocation

[5] Tomoeda (2019) studies full implementation of exactly-efficient social choice rules with endogenous investment.

is an NP-hard problem, VCG-based mechanisms with nearly-optimal allocation algorithms are generically non-truthful, while Lehmann et al. (2002) introduced a truthful mechanism for the knapsack problem in which the allocation is determined by a greedy algorithm. In addition, Hartline and Lucier (2015) developed a method for converting a (non-optimal) algorithm for optimization into a Bayesian incentive compatible mechanism with weakly higher social welfare or revenue; Dughmi et al. (2017) generalized this result to multidimensional types. For a more comprehensive review of results on approximation in mechanism design, see Hartline (2016).

There is also a large literature on greedy algorithms of the type we study here, which sort bidders based on some intuitive criteria and choose them for packing in an irreversible way; see Pardalos et al. (2013) for a review. Lehmann et al. (2002) studied the problem of constructing strategy-proof mechanisms from greedy algorithms; similarly, Bikhchandani et al. (2011) and Milgrom and Segal (2020) proposed clock auction implementations of greedy allocation algorithms.

Finally, our concept of an XBONE algorithm is closely related to the definition of a "bitonic" algorithm, introduced by Mu'Alem and Nisan (2008) to construct truthful mechanisms in combinatorial auctions. Bitonicity is defined for binary outcomes; with the restriction to binary outcomes, every XBONE algorithm is bitonic, but not vice versa.

2 Investment with binary outcomes

2.1 Model

We start our exposition with binary outcomes—each bidder is either "packed" or "unpacked," and we normalize the value of being unpacked to 0. We later generalize the main theorem to allow any finite number of outcomes for each bidder.

We consider three nested perspectives on the same situation. First is the allocation problem, in which our objective is total welfare and the values of the bidders are known to us. Second is the reporting problem, in which values are private information and we must elicit them via an incentive-compatible payment rule prior to allocation. Third perspective—our main contribution—is the investment problem, in which a bidder can make costly investments to change his value before reporting.

Proofs omitted from the main text are in Appendix A.

2.1.1 The allocation problem

We follow the standard approach in computer science, of assessing an algorithm's worst-case performance over some domain of instances. In words, an **instance** consists of a profile of bidder values and feasibility constraints. A bidder n has a value v_n for being packed. A **value profile** v is a vector that specifies, for each bidder, that bidder's value for being packed.

We define an **allocation problem** to be a collection of instances. An **algorithm** for a problem chooses a set of bidders to pack, subject to the feasibility constraints, with the objective of maximizing the sum of the values of the packed bidders. We now define those same objects formally, using the notation on which we will rely.

An **instance** (v, A) consists of:

1. a **value profile** $v \in (\mathbb{R}_0^+)^N$, for some set of **bidders** N, and
2. a set of **feasible allocations** $A \subseteq \wp(N)$.

An **allocation problem** is a collection Ω of instances such that the possible value profiles are products of intervals. More formally, for each set of feasible allocations A, there exists for each bidder $n \in N$ a closed interval $V_n^A \subseteq \mathbb{R}$ such that $\{v : (v, A) \in \Omega\} = \prod_n V_n^A$.

An **algorithm** x is a function that selects a feasible allocation for each instance $(v, A) \in \Omega$, that is, $x(v, A) \in A$.[6] We occasionally abuse notation and write $x_n(v, A)$ to denote an indicator function, equal to 1 if n is packed (*i.e.,* $n \in x(v, A)$) and 0 otherwise.

The **welfare** of algorithm x at instance (v, A) is

$$W_x(v, A) \equiv \sum_{n \in x(v,A)} v_n.$$

The **optimal welfare** at instance (v, A) is

$$W^*(v, A) \equiv W_{\text{OPT}}(v, A) = \max_{a \in A} \left\{ \sum_{n \in a} v_n \right\},$$

[6]In complexity theory, we often are not given the feasible allocations A directly, but instead only a description that implies which allocations are feasible. For instance, a description could specify the bidders' sizes and the capacity of the knapsack. In principle, algorithms for the knapsack problem could output different allocations for two instances with different item sizes but the same feasible allocations. Our formulation ignores this description-dependence, but we could easily accommodate it by specifying a function $\mathcal{A}$ from descriptions to feasible allocations, and defining an instance as consisting of a value profile v and a description d; none of our results would materially change with this adjustment.

where OPT is an algorithm that always achieves the maximum feasible welfare,

$$\mathrm{OPT}(v, A) \in \operatorname*{argmax}_{a \in A} \left\{ \sum_{n \in a} v_n \right\}.$$

In the knapsack problem and other cases of interest, optimization is NP-hard and it may be impractical to identify optimal solutions, even though fast algorithms may guarantee acceptable performance on some problems. The standard measure of algorithm performance is the worst-case guarantee, which is defined as follows.

Definition 2.1. For $\beta \in [0, 1]$, an algorithm x is a **β-approximation for allocation** if for all $(v, A) \in \Omega$, we have

$$\beta W^*(v, A) \leq W_x(v, A).$$

Our goal is to analyze whether and when the performance guarantee of an algorithm also applies to the long-run problem in which bidders' investments determine their values and their reports are the inputs to the algorithm.

We begin with the problem of truthful reporting, which is equivalently characterized as a problem of mechanism design.

2.1.2 The reporting problem

Given some allocation problem Ω, we next consider the corresponding **reporting problem**, which differs from the allocation problem because the algorithm can no longer directly input each bidder n's value v_n and must instead rely on each bidder's *reported* value $\hat{v}_n$. To elicit truthful value reports, we use a **mechanism** (x, p), which is a pair consisting of an algorithm x and a payment rule p that maps any reported instance $(\hat{v}, A)$ into an allocation $x(\hat{v}, A) \in A$ and a profile of payments $p(\hat{v}, A) \in \mathbb{R}^N$.

Definition 2.2. The mechanism (x, p) is **strategy-proof** if for all $(v, A) \in \Omega$ and all $n \in N$, we have

$$v_n \in \operatorname*{argmax}_{\hat{v}_n \in V_n^A} \{v_n x_n(\hat{v}_n, v_{-n}, A) - p_n(\hat{v}_n, v_{-n}, A)\};$$

that is, if reporting truthfully is always a best response for each $n \in N$.

In the reporting problem, the mechanism (x, p) might be chosen to (approximately) maximize welfare, subject to the additional constraint that (x, p) be strategy-proof.

Definition 2.3. For $\beta \in [0, 1]$, (x, p) is a **β-approximation for reporting** if x is a β-approximation for allocation and (x, p) is strategy-proof.

Given an algorithm x that is an β-approximation for allocation, when can we choose payments so that (x, p) is an β-approximation for reporting?

Definition 2.4. Algorithm x is **monotone (on Ω)** if, for all $(v, A) \in \Omega$ and $n \in N$, if $n \in x(v, A)$, then $n \in x(\tilde{v}_n, v_{-n}, A)$ for all $\tilde{v}_n \geq v_n$.

Definition 2.5. The **threshold price** for bidder n at instance (v, A) is

$$t_n^x(v, A) \equiv \inf\{\tilde{v}_n : n \in x(\tilde{v}_n, v_{-n}, A) \text{ and } (\tilde{v}_n, v_{-n}, A) \in \Omega\}.$$

For any monotone x, we define the **threshold auction** (x, p^x) to be the mechanism such that for all n and all (v, A),

$$p_n^x(v, A) = \begin{cases} t_n^x(v, A) & n \in x(v, A) \\ 0 & \text{otherwise;} \end{cases}$$

that is, a threshold auction uses a monotonic allocation rule and charges each bidder his threshold price in the case that he is packed, and charges 0 otherwise.

For any optimal algorithm OPT, the corresponding threshold auction $(\text{OPT}, p^{\text{OPT}})$ is the Vickrey-Clarke-Groves (VCG) auction. For other strategy-proof mechanisms, the following characterization is a special case of the well-known "taxation principle" of mechanism design. (Alternatively, see Myerson (1981).)

Proposition 2.1. *If x is monotone, then the threshold auction (x, p^x) is strategy-proof. Conversely, if (x, p) is strategy-proof then for all (v, A) and all n we have*

$$p_n(v, A) = p_n^x(v, A) + f(v_{-n}, A),$$

where $p_n^x(v, A)$ is the threshold auction price for n and f is a function that does not depend on v_n.

Corollary 2.1. If x is monotone and a β-approximation for allocation, then (x, p^x) is a β-approximation for reporting.

2.1.3 The investment problem

Finally, given some allocation problem Ω, we define the corresponding **investment problem**. For now, we focus on the investment decision of a single bidder; we consider the case in which multiple bidders may invest in Section 2.3.

We assume that one bidder $\iota \in N$ has an opportunity to change his value at a cost, with knowledge of his threshold price.

In the reporting problem, we require that each bidder be incentivized to report his value truthfully, regardless of his beliefs about the other bidders' values. In the investment problem, we instead require that a bidder with an investment opportunity be incentivized to make socially-beneficial investments, with full information about his threshold price.

Given these distinct assumptions about information, it is natural to ask: Is the true situation described by the reporting problem, or the investment problem? Our perspective is that these models capture different aspects of the same situation. When a mechanism is new, each bidder's value may be known by no one else, so we must ensure that these values are reported truthfully as inputs to the allocation algorithm. This short-run consideration is captured by the reporting problem. Over time, bidders may learn much more—a bidder may, for instance, use historical data to forecast his own threshold price. However, as time passes, bidders may also find opportunities to adjust their assets and technology given the prices they face. These gradual adjustments can affect the performance of the mechanism. This long-run consideration is captured by the investment problem.

Given an investor $\iota \in N$, an **investment** is a pair $(v_\iota, c_\iota) \in V_\iota^A \times \mathbb{R}$, specifying a value and a cost. An **instance** of the investment problem is a tuple $(I_\iota, v_{-\iota}, A)$, where $I_\iota \subseteq V_\iota^A \times \mathbb{R}$ is a set of feasible investments and $v_{-\iota} \in V_{-\iota}^A$. We restrict attention to instances that satisfy:

1. **Finiteness.** $|I_\iota| < \infty$.

2. **Normalization.** $\min \{c_\iota : (v_\iota, c_\iota) \in I_\iota\} = 0$.

Note that while n denotes a representative element of N, ι denotes the investor, so ι is only well-defined once we fix an instance of the investment problem.

As a baseline, we consider the investment problem under the VCG auction. For that auction, the total profits of the auctioneer and all the participants besides ι is an amount $f(v_{-\iota}, A)$ that does not depend on ι's report. Hence, ι's net profit is the total social welfare minus $f(v_{-\iota}, A)$. A consequence is that ι maximizes his own payoff by maximizing social welfare, which he does both by reporting truthfully and by choosing the social-welfare maximizing investment.

Definition 2.6. Mechanism (x, p) has **efficient investments** if for every investment instance $(I_\iota, v_{-\iota}, A)$ we have[7]

$$\operatorname*{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{v_\iota \cdot x_\iota(v_\iota, v_{-\iota}, A) - p(v_\iota, v_{-\iota}, A) - c_\iota\} = \operatorname*{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{W_x(v_\iota, v_{-\iota}, A) - c_\iota\}. \quad (1)$$

[7]It is natural to consider replacing "=" in (1) with "⊆". These definitions are equivalent.

Next we observe that having efficient investments is equivalent to a basic property of VCG mechanisms.

Lemma 2.1. *Mechanism (x, p) has efficient investments if and only if (x, p)* ***provides marginal rewards*** *in the sense that for any two allocation instances $(v_\iota, v_{-\iota}, A)$ and $(v'_\iota, v_{-\iota}, A)$,*

$$[v_\iota \cdot x_\iota(v_\iota, v_{-\iota}, A) - p(v_\iota, v_{-\iota}, A)] - [v'_\iota \cdot x_\iota(v'_\iota, v_{-\iota}, A) - p(v'_\iota, v_{-\iota}, A)] \\ = W_x(v_\iota, v_{-\iota}, A) - W_x(v'_\iota, v_{-\iota}, A). \quad (2)$$

Proposition 2.2. *Any VCG auction has efficient investments.*

We also obtain a partial converse to Proposition 2.2. In particular, we show that if a mechanism is strategy-proof and has efficient investment incentives, then it must act like a VCG auction restricted to a subset of the allocations.

We now introduce a notation for the welfare generated by selecting allocation a at value profile v:

$$w(a \mid v) \equiv \sum_{n \in a} v_n. \quad (3)$$

With the notation (3), note that we have $W_x(v, A) = w(x(v, A) \mid v)$.

Definition 2.7. Algorithm x has **range-efficient allocations** if for every A, there exists some set of allocations $R \subseteq A$ such that for every value profile $v \in V_n^A$,

$$W_x(v, A) = \max_{r \in R} w(r \mid v).$$

We now have three desiderata for mechanisms: range-efficient allocations, strategy-proofness, and efficient investments. Our next proposition states that any two of these together imply the third.[8]

Proposition 2.3. *For any mechanism (x, p) with monotone x:*

1. *If x has range-efficient allocations and (x, p) is strategy-proof, then (x, p) has efficient investments.*

2. *If x has range-efficient allocations and (x, p) has efficient investments, then (x, p) is strategy-proof.*

3. *If (x, p) is strategy-proof and has efficient investments, then x has range-efficient allocations.*

[8] None of the three, by itself, implies either of the other two. Clauses 1 and 2 of Proposition 2.3 are corollaries of Theorem 1 of Hatfield et al. (2019).

Proposition 2.2 states that the VCG auction induces any given bidder to make the socially optimal investment; Clause 3 of Proposition 2.3 states that any strategy-proof mechanism that has that property must yield the same welfare as choosing the exactly optimal allocation from a restricted set R. Nisan and Ronen (2007) find that any strategy-proof mechanism that provides marginal rewards (in the sense of Lemma 2.1) must have range-efficient allocations; together with Lemma 2.1 this yields Clause 3.[9]

Proposition 2.2 leaves open two ways in which VCG may fail to provide efficient investment incentives. First, a bidder may not know his threshold price and may forecast incorrectly whether he will be packed. Second, as we discuss further in Section 2.3, if multiple bidders make simultaneous investment decisions, then while each of them makes an investment that is socially optimal conditional on others' investments, there may be coordination problems that render the overall equilibrium socially inefficient.

To isolate the impact of using an approximately-optimal allocation rule, we shut down the information and coordination channels, and ask how a single agent's investment incentives are affected by the approximation.

As we have already seen, even with just one investing bidder under full information, the investment problem becomes subtle: under an approximately-optimal mechanism, there can be privately profitable investment opportunities that reduce social welfare. When does this possibility mean that investment performance is strictly worse than allocation performance?

Suppose we have some weakly monotone algorithm x that guarantees a β-approximation for allocation. Under what conditions does its corresponding threshold auction still yield a β-approximation in the investment problem?

When ι faces a threshold auction (x, p^x), his **utility** from investment (v_ι, c_ι) is

$$u_\iota(x, v_\iota, c_\iota, v_{-\iota}, A) \equiv v_\iota x_\iota(v_\iota, v_{-\iota}, A) - p_\iota^x(v_\iota, v_{-\iota}, A) - c_\iota.$$

We denote his **best responses** at instance $(I_\iota, v_{-\iota}, A)$ by

$$\text{BR}(x, I_\iota, v_{-\iota}, A) \equiv \underset{(v_\iota, c_\iota) \in I_\iota}{\text{argmax}} \left\{ u_\iota(x, v_\iota, c_\iota, v_{-\iota}, A) \right\}.$$

The **welfare** of algorithm x at instance $(I_\iota, v_{-\iota}, A)$ is then

$$\overline{W}_x(I_\iota, v_{-\iota}, A) \equiv \min_{(v_\iota, c_\iota) \in \text{BR}(x, I_\iota, v_{-\iota}, A)} \left\{ W_x(v_\iota, v_{-\iota}, A) - c_\iota \right\}; \tag{4}$$

[9] Theorem 3.2 and Corrollary 3.7 of Nisan and Ronen (2007). For completeness, we also provide a proof of Lemma 2.1 in Appendix A.

the **optimal welfare** at instance $(I_\iota, v_{-\iota}, A)$ is

$$\overline{W}^*(I_\iota, v_{-\iota}, A) \equiv \max_{(v_\iota, c_\iota)\in I_\iota} \{W^*(v_\iota, v_{-\iota}, A) - c_\iota\}.$$

Definition 2.8. For $\beta \in [0,1]$, algorithm x is a β**-approximation for investment** if for all investment instances $(I_\iota, v_{-\iota}, A)$,

$$\beta\overline{W}^*(I_\iota, v_{-\iota}, A) \leq \overline{W}_x(I_\iota, v_{-\iota}, A).$$

Proposition 2.4. *If x is a β-approximation for investment, then x is a β-approximation for allocation.*

Proof. Any instance of the allocation problem $(v_\iota, v_{-\iota}, A)$ is equivalent to the instance of the investment problem $(I_\iota, v_{-\iota}, A)$ in which the investment technology is the singleton $\{(v_\iota, 0)\}$. Thus, the investment problem embeds the allocation problem without investment as a special case. □

Our next result shows that without further structure, even if the allocation guarantee is very good, the investment guarantee can be arbitrarily poor.

Proposition 2.5. *Let Ψ be the set of instances such that $|N| = 2$, $v \in \mathbb{R}^2_+$, and $A = \wp(N)$. If $\Omega \supseteq \Psi$, then for all $\beta \in (0,1)$, there exists an algorithm x^β for Ω such that*

1. *x^β is monotone;*
2. *x^β is a β-approximation for allocation; and*
3. *for all $\beta' > 0$, x^β is not a β'-approximation for investment.*

Proposition 2.5 suggests that investment efficiency guarantees are (very) sensitive to relaxing full efficiency of the allocation rule—even independent of the known inefficiencies that arise in the presence of incomplete information and/or coordination failures.

Note also that the setting of Proposition 2.5 includes the knapsack problem, which we define in Section 2.2.2.

Proof of Proposition 2.5. We construct the algorithms x^β as follows:

$$x^\beta(v, A) = \begin{cases} \{1,2\} & \text{if } (v, A) \in \Psi \text{ and } \frac{v_1}{v_1+v_2} < \beta \\ \{1\} & \text{if } (v, A) \in \Psi \text{ and } \frac{v_1}{v_1+v_2} \geq \beta \\ \mathrm{OPT}(v, A) & \text{otherwise.} \end{cases}$$

By inspection, x^β is monotone and a β-approximation for allocation. Moreover, since Bidder 1 is always packed for instances in Ψ, Bidder 1's threshold price at such instances is 0.

Consider the investment technology $I_1 = \{(\gamma+\epsilon, \gamma), (0,0)\}$ for $\gamma, \epsilon > 0$. For any $(v, A) \in \Psi$, Bidder 1's best-response at investment instance (I_1, v_2, A) is to choose investment $(\gamma + \epsilon, \gamma)$. For large enough γ, however, x^β packs only Bidder 1, for total welfare ϵ. By contrast, the optimal benchmark chooses investment $(\gamma+\epsilon, \gamma)$ and packs both bidders, for total welfare $v_2 + \epsilon$. For all $\beta' > 0$, we can pick $v_2 > 0$ and small enough ϵ, so

$$\overline{W}_x(I_1, v_2, A) = \epsilon < \beta'(v_2 + \epsilon) = \beta' \overline{W}^*(I_1, v_2, A). \qquad \square$$

2.2 Results

For any given investment technology, a bidder may have multiple best choices and in (4) we have specified the welfare-minimizing one as the basis for our calculations. Our next result allows us to ignore this multiplicity by showing that an algorithm's investment approximation ratio over all instances is equal to its approximation ratio over just the instances with singleton best-responses.

Lemma 2.2. *If for all $(I_\iota, v_{-\iota}, A)$ such that $\mathrm{BR}(x, I_\iota, v_{-\iota}, A)$ is a singleton, we have*

$$\beta \overline{W}^*(I_\iota, v_{-\iota}, A) \leq \overline{W}_x(I_\iota, v_{-\iota}, A),$$

then x is a β-approximation for investment.

We now characterize the investor's best response facing any threshold auction. Specifically, we show that the bidder can find an optimal investment using the following procedure:

1. First, find the investment that would maximize his value net of cost.
2. Make that investment if the associated value net of cost is above the threshold price; otherwise, make a costless investment.

Lemma 2.3. *Given an instance $(I_\iota, v_{-\iota}, A)$, let $(v_\iota^\uparrow, c_\iota^\uparrow)$ denote an arbitrary element of $\operatorname{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{v_\iota - c_\iota\}$. Let $(v_\iota^\downarrow, c_\iota^\downarrow) \in I_\iota$ denote a costless investment ($c_\iota^\downarrow = 0$). For any monotone algorithm x:*

1. *if $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow, v_{-\iota}, A)$, then $(v_\iota^\uparrow, c_\iota^\uparrow)$ is a best-response for ι;*
2. *otherwise, $(v_\iota^\downarrow, c_\iota^\downarrow)$ is a best-response for ι.*

Proof. Let $\tau_\iota(v_{-\iota}, A)$ be the threshold price for ι. To reduce clutter, we suppress the dependence of u_ι, x_ι, and τ_ι on $(v_{-\iota}, A)$. To prove Clause 1, we suppose that $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow)$. Then $v_\iota^\uparrow - c_\iota^\uparrow \geq \tau_\iota$, and by x monotone, $\iota \in x(v_\iota^\uparrow)$. Thus,

$$u_\iota(v_\iota^\uparrow, c_\iota^\uparrow) = v_\iota^\uparrow - \tau_\iota - c_\iota^\uparrow \geq 0.$$

Take any $(v_\iota, c_\iota) \in I_\iota$. We want to prove that $u_\iota(v_\iota^\uparrow, c_\iota^\uparrow) \geq u_\iota(v_\iota, c_\iota)$. If $u_\iota(v_\iota, c_\iota) \leq 0$, then we are done. If $u_\iota(v_\iota, c_\iota) > 0$, then

$$u_\iota(v_\iota, c_\iota) = v_\iota - \tau_\iota - c_\iota \leq v_\iota^\uparrow - \tau_\iota - c_\iota^\uparrow = u_\iota(v_\iota^\uparrow, c_\iota^\uparrow),$$

where the inequality follows because $(v_\iota^\uparrow, c_\iota^\uparrow) \in \operatorname{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{v_\iota - c_\iota\}$.

Now, to prove Clause 2, we suppose that $n \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$. Take any $(v_\iota, c_\iota) \in I_\iota$. We want to prove that $u_\iota(v_\iota^\downarrow, c_\iota^\downarrow) \geq u_\iota(v_\iota, c_\iota)$. As $x_\iota(v_\iota^\uparrow - c_\iota^\uparrow) = 0$,

$$\tau_\iota \geq v_\iota^\uparrow - c_\iota^\uparrow \geq v_\iota - c_\iota.$$

Thus, we have $u_\iota(v_\iota, c_\iota) = \max\{v_\iota - \tau_\iota, 0\} - c_\iota \leq 0 \leq \max\{v_\iota^\downarrow - \tau_\iota, 0\} = u_\iota(v_\iota^\downarrow, c_\iota^\downarrow)$. □

We now state the key definition for our main theorem.

Definition 2.9. Algorithm x is **XBONE** (**eXcludes BOssy Negative Externalities**) if for any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$ of the allocation problem, if whenever either

1. $n \in x(v, A)$ and $\tilde{v}_n > v_n$, or
2. $n \notin x(v, A)$ and $\tilde{v}_n < v_n$,

we have

$$w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) \geq w(x(v, A) \mid \tilde{v}_n, v_{-n}). \tag{5}$$

If either of the two conditions in Definition 2.9 holds and x is monotone, then (5) is equivalent to the requirement that

$$\sum_{m \neq n} v_m \left[x_m(\tilde{v}_n, v_{-n}, A) - x_m(v, A)\right] \geq 0. \tag{6}$$

The left-hand side of (6) is the effect on other bidders' welfare caused by a change in bidder n's value. Since, under the identified conditions, there is no change in n's outcome or threshold price, this effect is a **bossy externality**; XBONE is thus, as the name suggests, the requirement that any such externality must be non-negative.

XBONE is equivalent to the requirement that if we raise the value of a packed bidder by some positive Δ, then the algorithm's welfare rises by at least Δ, and if we lower the value of an unpacked bidder, then the algorithm's welfare does not fall.[10]

XBONE algorithms can entail other kinds of externalities, as Section 2.2.2 illustrates, but excluding bossy negative externalities is sufficient to preserve the performance guarantee.

Theorem 2.1. *Assume that x is monotone. If x is XBONE and a β-approximation for allocation, then x is a β-approximation for investment.*

Proof. By Lemma 2.2, we can restrict attention to instances $(I_\iota, v_{-\iota}, A)$ with singleton best-responses. To reduce clutter, we suppress the dependence of x, W_x, $\overline{W}_x$, W^*, and $\overline{W}^*$ on $v_{-\iota}$ and A. Let $(v_\iota^\uparrow, c_\iota^\uparrow)$ denote an arbitrary element of $\text{argmax}_{(v_\iota, c_\iota)\in I_\iota} \{v_\iota - c_\iota\}$, and let $(v_\iota^\downarrow, c_\iota^\downarrow)$ denote a costless investment ($c_\iota^\downarrow = 0$).

By Lemma 2.3, there are two cases to consider. Either ι chooses $(v_\iota^\uparrow, c_\iota^\uparrow)$ and $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow)$, or ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$ and $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$. The next two inequalities below follow from the hypothesis that x is XBONE.

If ι chooses $(v_\iota^\uparrow, c_\iota^\uparrow)$ and $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow)$, then as x is XBONE,

$$\overline{W}_x(I_\iota) = W_x(v_\iota^\uparrow) - c_\iota^\uparrow \geq W_x(v_\iota^\uparrow - c_\iota^\uparrow).$$

If ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$ and $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$, then as x is XBONE,

$$\overline{W}_x(I_\iota) = W_x(v_\iota^\downarrow) - c_\iota^\downarrow = W_x(v_\iota^\downarrow - c_\iota^\downarrow) \geq W_x(v_\iota^\uparrow - c_\iota^\uparrow).$$

Let (v_ι^*, c_ι^*) be an element of $\text{argmax}_{(v_\iota, c_\iota)\in I_\iota} \{W^*(v_\iota) - c_\iota\}$, so that

$$\overline{W}^*(I_\iota) = W^*(v_\iota^*) - c_\iota^* = W^*(v_\iota^* - c_\iota^*) \leq W^*(v_\iota^\uparrow - c_\iota^\uparrow). \tag{7}$$

Thus, as x is a β-approximation for allocation, we have

$$\overline{W}_x(I_\iota) \geq W_x(v_\iota^\uparrow - c_\iota^\uparrow) \geq \beta W^*(v_\iota^\uparrow - c_\iota^\uparrow) \geq \beta \overline{W}^*(I_\iota).$$

This completes the proof of Theorem 2.1. □

2.2.1 Non-bossiness and XBONE

In mechanism design, a mechanism is said to be bossy if a bidder can affect other bidders' outcomes without affecting his own. Formally:

[10] Bitonicity, as defined by Mu'Alem and Nisan (2008), is a weaker requirement: if we raise the value of a packed bidder or lower the value of an unpacked bidder, then the algorithm's welfare does not fall.

Definition 2.10. Algorithm x is **non-bossy** if for all (v, A) and $\tilde{v}_n$, if $x_n(v, A) = x_n(\tilde{v}_n, v_{-n}, A)$, then $x(v, A) = x(\tilde{v}_n, v_{-n}, A)$, that is, if no bidder can affect other bidders' outcomes without affecting his own.

The following proposition shows that XBONE is weaker than non-bossiness.

Proposition 2.6. *If x is monotone and non-bossy, then x is XBONE.*

Proof. Take any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$ that satisfy the antecedent condition of Definition 2.9. As x is monotone, we have $x_n(v, A) = x_n(\tilde{v}_n, v_{-n}, A)$. Then, as x is non-bossy, we have $x(v, A) = x(\tilde{v}_n, v_{-n}, A)$. Thus, we see that

$$w(x(v, A) \mid \tilde{v}_n, v_{-n}) = w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}),$$

as desired. □

The converse of Proposition 2.6 is false; Example 2.2 in the next section exhibits a monotone XBONE algorithm that is bossy.

XBONE requires that for particular value changes for an individual that do not affect that individual's outcome, x should not pick less valuable outcomes for others. Non-bossiness is stronger: it requires that for *any* value change for an individual that does not affect that individual's outcome, x should not make *any* change in others' outcomes.

A standard technique for computationally hard problems is to run several candidate algorithms and select the best of their solutions. However, the resulting algorithm may be bossy, even if the candidate algorithms are non-bossy. By contrast, if the candidate algorithms are XBONE, then the resulting algorithm is XBONE.

Proposition 2.7. *Let X be a collection of XBONE algorithms. If y is an algorithm that at each instance $(v, A) \in \Omega$ outputs a welfare-maximizing allocation from the collection $\{x(v, A)\}_{x \in X}$, then y is XBONE.*

Proof. We consider any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$ satisfying the antecedent condition of Definition 2.9. Let $x \in X$ be such that $y(v, A) = x(v, A)$. As x is XBONE, we have

$$\begin{aligned} w(y(v, A) \mid \tilde{v}_n, v_{-n}) &= w(x(v, A) \mid \tilde{v}_n, v_{-n}) \\ &\leq w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) \\ &\leq w(y(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}), \end{aligned}$$

as desired. □

2.2.2 Application: Knapsack algorithms

The knapsack problem is a special case of the allocation problem introduced in Section 2.1.1. In the knapsack problem, there is a set of bidders, where an bidder n has value v_n and size s_n. The knapsack has capacity S. Without loss of generality, suppose no bidder's size is more than S. The set of feasible allocations is any subset of bidders $K \subseteq N$ such that $\sum_{n \in K} s_n \leq S$. As before, let A denote the set of feasible allocations and let a be an element of A.

The knapsack problem is NP-Hard (Karp, 1972); there is no known polynomial-time algorithm that outputs optimal allocations (Cook, 2006; Fortnow, 2009). Dantzig (1957) suggested applying a Greedy **algorithm** to the knapsack problem. Formally:

Algorithm 1 (Greedy)**.** Sort bidders by the ratio of their values to their sizes so that

$$\frac{v_1}{s_1} \geq \frac{v_2}{s_2} \cdots \geq \frac{v_{|N|}}{s_{|N|}}. \tag{8}$$

Add bidders to the knapsack one by one in the sorted order, so long as the sum of the sizes does not exceed the knapsack's capacity. When encountering the first bidder that would violate the size constraint, stop.

Although the Greedy algorithm performs well on some instances, including ones for which all bidders are small in relation to the capacity of the knapsack, its worst-case performance guarantee is 0, as illustrated by the following example.

Example 2.1. Consider a knapsack with capacity 1 and two bidders. For some arbitrarily small $\epsilon > 0$, let $v_1 = \epsilon$, $s_1 = \frac{\epsilon}{2}$, $v_2 = 1$, and $s_2 = 1$. The Greedy algorithm picks bidder 1 and stops, whereas the optimal algorithm picks bidder 2. Thus, Greedy's performance is no better than ϵ of the optimum.

There is a standard modification of the Greedy algorithm that improves the worst-case guarantee for the knapsack problem. Let us define the MGreedy **algorithm** as follows.

Algorithm 2 (MGreedy)**.** Run the Greedy algorithm. Compare the Greedy algorithm's packing to the the most valuable individual bidder; output whichever has higher welfare.

MGreedy's worst-case performance is much better than Greedy's:

Proposition 2.8. MGreedy *is a $\frac{1}{2}$-approximation for the Knapsack problem.*

Proof. For any instance ω, order the bidders by value/size as in (8). If GREEDY packs all bidders, then trivially $W^*(\omega) = W_{\text{MGreedy}}(\omega)$. Otherwise, let k be the lowest index of a bidder not packed by GREEDY and let K be the index of a bidder with maximum value. We have

$$\begin{aligned} W^*(\omega) \leq \sum_{n=1}^{k} v_n &= W_{\text{Greedy}}(\omega) + v_k \\ &\leq W_{\text{Greedy}}(\omega) + v_K \\ &\leq 2 \max \{W_{\text{Greedy}}(\omega), v_K\} \\ &= 2W_{\text{MGreedy}}(\omega). \end{aligned}$$

□

MGREEDY turns out to be bossy, as our next example shows.

Example 2.2. Consider the knapsack instance with capacity 10 and 3 bidders. $v_1 = 2$, $v_2 = 1$, $v_3 = 8$. $s_1 = s_2 = 1$, $s_3 = 9$. At this instance, MGREEDY packs just bidder 3. If we raise v_3 to 10, then MGREEDY instead packs bidder 1 and bidder 3. Thus, MGREEDY is bossy. However, this is a bossy **positive** externality; raising the value of a packed bidder by 2 has increased welfare by 4.

Proposition 2.9. *For the knapsack problem, the* GREEDY *algorithm and the* MGREEDY *algorithm are both XBONE.*

Proof. Consider the bidders sorted by the GREEDY algorithm as in (8), and suppose the GREEDY algorithm packs bidders 1 to K. If we raise the value of a packed bidder, then the GREEDY algorithm terminates at the same point, packing bidders 1 to K. If we lower the value of an unpacked bidder, then the GREEDY algorithm terminates no earlier than before, packing at least bidders 1 to K. Hence the GREEDY algorithm is XBONE.

The MGREEDY algorithm's output is equal to the welfare-maximizing selection from the outputs of two algorithms:

- the GREEDY algorithm, and
- the algorithm that selects the most valuable single item.

We have just shown that the GREEDY algorithm is XBONE. Meanwhile, the algorithm that selects the most valuable single item is monotone and non-bossy and so is XBONE by Proposition 2.6, as well. Thus, by Proposition 2.7, the MGREEDY algorithm is XBONE. □

For the example in the Introduction, the GREEDY and MGREEDY algorithms output the same packings. Hence, that example shows that there can be negative externalities under the

MGREEDY algorithm. In particular, an investment that causes the investor to be packed can increase the investor's utility but yield a reduction in social welfare. However, those negative externalities are not bossy, so they cannot undermine the MGREEDY algorithm's worst-case performance guarantee of $\frac{1}{2}$. Conversely, Example 2.2 shows that there can be bossy externalities under the MGREEDY algorithm, but because those bossy externalities are not negative, they, too, cannot undermine the worst-case performance guarantee.

For the knapsack problem, there is a *fully polynomial time approximation scheme (FPTAS)* that, for any $\epsilon > 0$, yields a $(1-\epsilon)$-approximation, and runs in polynomial time in both the number of items and $\frac{1}{\epsilon}$. The FPTAS rounds down the values, and uses dynamic programming to output an optimal allocation for the rounded instance.[11] Despite being optimal up to rounding, the FPTAS is not XBONE. In fact, the next result shows that the FPTAS is a 0-approximation for investment.

Proposition 2.10. *For the knapsack problem, the* FPTAS *is a* 0*-approximation for investment.*

We prove Proposition 2.10 in Appendix B.

Other standard optimization algorithms can also fail to be XBONE. For example, in Appendix C, we show that well-known minimum spanning tree algorithm for the Steiner tree problem is not XBONE.

2.2.3 A necessary and sufficient condition

Definition 2.9 is sufficient for approximation guarantees to persist under investment; however, it is not quite necessary. In this section, we show that the first half of the XBONE condition, which states that there is no bossy negative externality for positive investments, *is* necessary. The second half of the XBONE condition, which requires the same for disinvestments, is only necessary for values above the VCG price. We show that modifying XBONE to require only these components gives us a necessary and sufficient condition.

Definition 2.11. Algorithm x is **weakly XBONE** if for any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$ of the allocation problem, if

1. either $n \in x(v, A)$ and $\tilde{v}_n > v_n$,
2. or $n \notin x(v, A)$, $\tilde{v}_n < v_n$, and $t_n^{\text{OPT}}(v, A) < v_n$

then we have

$$w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) \geq w(x(v, A) \mid \tilde{v}_n, v_{-n}).$$

[11] We briefly explain this family of algorithms in Appendix B. For details, we refer interested readers to the textbooks by Williamson and Shmoys (2011, p. 65-68) or Vazirani (2013, p. 68-70).

Theorem 2.2. *Assume that x is monotone. If x is weakly XBONE and is a β-approximation for allocation, then x is a β-approximation for investment.*

Theorem 2.2 establishes that XBONE is not a necessary condition for worst-case guarantees to persist under investment, as weak XBONE is sufficient. However, in problems of interest there is no known fast method to compute the VCG threshold prices, since those prices are defined by the exact solution to the optimization problem. Thus, Clause 2 of Definition 2.11 may be intractable to verify.

Definition 2.12. For two problems Ω and Ω', Ω' is a **sub-problem** of Ω if $\Omega' \subseteq \Omega$.

If x is monotone and weakly XBONE on Ω, then x is monotone and weakly XBONE on any sub-problem Ω'; hence, we obtain the following corollary of Theorem 2.2.

Corollary 2.2. Suppose that x is monotone and is weakly XBONE on problem Ω. For any sub-problem Ω', if x is a β'-approximation for allocation on Ω', then x is a β'-approximation for investment on Ω'.

Next, we find that, under a mild technical condition, weak XBONE is necessary for the conclusion of Corollary 2.2. That is, weak XBONE comprises a maximal domain for allocative guarantees to extend to investment guarantees.

Theorem 2.3. *Assume x is monotone and a β-approximation for allocation on problem Ω for $\beta > 0$. Suppose that for all $\iota \in N$ and all $(v_{-\iota}, A)$, there exists a partition of V_ι^A into positive-length intervals such that $x(\cdot, v_{-\iota}, A)$ is measurable with respect to that partition.*

If x is not weakly XBONE, then there exists a sub-problem $\Omega' \subseteq \Omega$ and β' such that x is a β'-approximation for allocation on Ω', but not a β'-approximation for investment on Ω'.

How much can XBONE be relaxed, while still ensuring that an algorithm's allocative guarantees extend to investment? Theorem 2.3 provides an answer: the "upward" direction of XBONE cannot be relaxed at all, and the "downward" direction can only be relaxed below the VCG threshold price.

2.3 Allowing multiple investors

The analysis changes in two ways when multiple participants can make investments. The first change is made to acknowledge a possible *coordination problem* among the investors, which requires a different statement of the conclusion of the theorems. The second change arises because we use a condition stronger than XBONE to prove the new conclusion.

	2 invests	2 refrains
1 invests	$1,1$	$2,0$
1 refrains	$0,2$	$0,0$

Table 1: Payoffs induced for Bidders 1 and 2 under Example 2.4.

Formally, an instance of the multi-investor problem is a tuple (I, A), where $I = (I_n)_{n \in N}$ and $I_n \subseteq V_n^A \times \mathbb{R}$ is a set of feasible investments. We restrict attention to investment technologies that satisfy:

1. **Finiteness.** $|I_n| < \infty$.

2. **Normalization.** $\min \{c_n : (v_n, c_n) \in I_n\} = 0$.

With multiple investors, even VCG auctions can suffer from inefficient investments due to a coordination problem, as the following example illustrates.

Example 2.3. There is a knapsack with capacity 2, and three bidders, with sizes $s_1 = 2$, $s_2 = s_3 = 1$. Bidder 1 has the singleton technology $I_1 = \{(10, 0)\}$. Bidders 2 and 3 have the technology $I_2 = I_3 = \{(0, 0), (9, 1)\}$. It is socially optimal for Bidders 2 and 3 to both choose $(9, 1)$ and both be packed. However, if only one of them invests, then it is optimal to pack just Bidder 1. In the VCG auction $\left(\text{OPT}, p^{\text{OPT}}\right)$, there are two Nash equilibrium investment profiles. In one Nash equilibrium, no bidder invests. In the efficient Nash equilibrium, both Bidders 2 and 3 invest.

When multiple bidders can invest, XBONE does not imply that there exists an equilibrium that retains the original allocative guarantee, as we now illustrate.

Example 2.4. There is a knapsack with capacity 11, and three bidders, with sizes $s_1 = s_2 = 1$, $s_3 = 10$. Bidders 1 and 2 have the technology $I_1 = I_2 = \{(50, 48), (0, 0)\}$. Bidder 3 has the singleton technology $I_3 = \{(10, 0)\}$.

Facing the threshold auction for the MGreedy algorithm, it is a strictly dominant strategy for Bidder 1 to choose $(50, 48)$, and similarly for Bidder 2. Table 1 displays the induced payoffs. Then the MGreedy algorithm packs 1 and 2, yielding net welfare 4. By contrast, it is efficient for Bidder 1 to choose $(50, 48)$ and Bidder 2 to choose $(0, 0)$, and then to pack 1 and 3, for a net welfare of 12. Hence, no equilibrium obtains the MGreedy allocative guarantee of $\frac{1}{2}$.

However, if the algorithm is monotone, non-bossy and a β-approximation for allocation, then even with multiple investors, there is an equilibrium of the long-run problem that achieves the same performance.

Theorem 2.4. *Assume that x is monotone, non-bossy, and a β-approximation for allocation. For any instance of the multi-investor problem (I, A), there exists a Nash equilibrium $(\hat{v}, \hat{c})$ of the investment game facing threshold auction (x, p^x), such that*

$$W_x(\hat{v}, A) - \sum_{n \in N} \hat{c}_n \geq \beta \max_{(v,c) \in I} \left\{ W^*(v, A) - \sum_{n \in N} c_n \right\}.$$

The above results suggest that when several bidders invest simultaneously, this raises new challenges for the design of approximation algorithms. XBONE is no longer sufficient to preserve efficiency guarantees. Non-bossiness is sufficient, but many standard approximation algorithms are bossy.

3 Investment with multiple outcomes

The problems we studied in Section 2 were generalizations of the knapsack problem in which each bidder has two possible outcomes: being packed or not. We now extend our analysis to settings in which each bidder has many possible outcomes. This extension encompasses knapsack problems in which each participant can be packed with a large item or a small one, combinatorial auctions in which each bidder can win one of several packages, and many other problems.

3.1 Allocation problems with multiple outcomes

Let O denote a finite set of **outcomes**. Each bidder's **value** $v_n \in (\mathbb{R}_0^+)^O$ is a row vector, with element v_n^o denoting n's value for outcome o. We normalize the value of one outcome $\underline{o}$, $v_n^{\underline{o}} = 0$; this is n's value for "being unpacked." A **value profile** $v = (v_n)_{n \in N}$ specifies a value for each bidder.

An **allocation** $a = (a_n)_{n \in N}$ specifies an outcome $a_n \in O$ for each bidder n. It is convenient to represent a_n as a binary vector, with $a_n^o = 1$ if o is the outcome for bidder n, and 0 otherwise.

An **instance** (v, A) consists of a value profile v and a non-empty set of A of feasible allocations, such that for all $a \in A$, v's dimensions agree with a's dimensions.[12]

An **allocation problem** consists of a collection of instances, denoted Ω. For each A and n, let $V_n^A \subseteq \mathbb{R}^O$ denote the space of possible value vectors for bidder n. We assume a product structure: for all A, $\{v : (v, A) \in \Omega\} = \prod_n V_n^A$.

[12]With this formulation, it is without loss of generality for each bidder to have the same set of possible outcomes O. If some outcome is infeasible for bidder n, we can represent this by restricting A.

The welfare generated by selecting allocation $a \in A$ at instance (v, A) is

$$w(a \mid v) \equiv \sum_n a_n \cdot v_n.$$

As before, an algorithm x selects, for each instance $(v, A) \in \Omega$, a feasible allocation $x(v, A) \in A$; we denote n's outcome under x at (v, A) by $x_n(v, A)$. The **welfare** of algorithm x at instance (v, A) is

$$W_x(v, A) \equiv w(x(v, A) \mid v).$$

3.2 Reporting problems with multiple outcomes

A **mechanism** (x, p) consists of an algorithm x with $x(v, A) \in A$ and a payment rule p with $p(v, A) \in \mathbb{R}^N$. With multiple outcomes, it is less straightforward to characterize the strategy-proof mechanisms. A necessary condition is weak monotonicity of x.

Definition 3.1. Algorithm x is **weakly monotone** (**W-Mon**) if for any two instances (v_n, v_{-n}, A) and $(\tilde{v}_n, v_{-n}, A)$, we have

$$\tilde{v}_n \cdot x_n(\tilde{v}_n, v_{-n}, A) - \tilde{v}_n \cdot x_n(v_n, v_{-n}, A) \geq v_n \cdot x_n(\tilde{v}_n, v_{-n}, A) - v_n \cdot x_n(v_n, v_{-n}, A).$$

Proposition 3.1 (Lavi et al. (2003))**.** *If there exists p such that (x, p) is strategy-proof, then x is W-Mon.*

Moreover, when each V_n^A is convex, W-Mon is also a sufficient condition.[13]

Proposition 3.2 (Saks and Yu (2005))**.** *If for all n and A, the set of possible values V_n^A is convex, then if x is W-Mon, there exists p such that (x, p) is strategy-proof.*

When each V_n^A is convex, it follows that for any W-Mon x, the corresponding incentive-compatible payment rule p is essentially unique. The following Proposition is a corollary of the generalized envelope theorem (Milgrom and Segal, 2002, Corollary 1).

Proposition 3.3. *Suppose that for all n and A, the set of possible values V_n^A is convex. Then for any x, if (x, p) and $(x, \tilde{p})$ are both strategy-proof, then for any two instances (v_n, v_{-n}, A) and $(\tilde{v}_n, v_{-n}, A)$, we have*

$$p_n(v_n, v_{-n}, A) - p_n(\tilde{v}_n, v_{-n}, A) = \tilde{p}_n(v_n, v_{-n}, A) - \tilde{p}_n(\tilde{v}_n, v_{-n}, A).$$

[13] Bikhchandani et al. (2006) provide other domain assumptions such that W-Mon is sufficient.

Corollary 3.1. Let $\mathbf{0}$ denote a value vector with every element equal to 0. If $\mathbf{0} \in V_n^A$ and V_n^A is convex for all n and A, then for any W-Mon x, there is a unique payment rule p such that

1. (x, p) is strategy-proof, and
2. $p_n(\mathbf{0}, v_{-n}, A) = 0$ for all n, v_{-n}, and A.

Henceforth, we assume that each V_n^A is convex.

3.3 Investment problems with multiple outcomes

As before, we suppose that a bidder $\iota \in N$ has the opportunity to invest before reporting and allocation. An **investment** is a pair (v_ι, c_ι), with $v_\iota \in (\mathbb{R}_0^+)^O$ and $c_\iota \in \mathbb{R}$. An **investment instance** is a tuple $(I_\iota, v_{-\iota}, A)$, where $I_\iota \subseteq V_\iota^A \times \mathbb{R}$ is a set of feasible investments and $v_{-\iota} \in V_\iota^A$. We restrict attention to investment instances that satisfy:

1. **Finiteness.** $|I_\iota| < \infty$.
2. **Normalization.** $\min \{c_\iota : (v_\iota, c_\iota) \in I_\iota\} = 0$.

Given any W-Mon algorithm x, we suppose that ι faces a strategy-proof mechanism (x, p^x). We define u_ι, BR, $\overline{W}_x$, and $\overline{W}^*$ as before. Note that for convex V_ι^A, the particular choice of payment rule does not matter, because Proposition 3.3 implies that ι's best-responses are the same for all incentive-compatible payment rules.

3.4 Results for multiple outcomes

We now generalize our XBONE condition (Definition 2.9) and Theorem 2.1 to allow for more than two outcomes. Recall that Definition 2.9 involved starting from some instance (v, A) and then raising the value of a packed bidder or lowering the value of an unpacked bidder. The generalization below involves starting from some instance (v, A) and changing bidder n's value vector in a way that raises his marginal value for his current outcome $x_n(v, A)$ compared to any other outcome.

Definition 3.2. Algorithm x is **XBONE** if given any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$ such that

$$\text{for all outcomes } o, \quad \tilde{v}_n^{x_n(v,A)} - \tilde{v}_n^o \geq v_n^{x_n(v,A)} - v_n^o, \tag{9}$$

we have

$$w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) \geq w(x(v, A) \mid \tilde{v}_n, v_{-n}). \tag{10}$$

Note that by our normalization, $\tilde{v}_n^{\underline{o}} = v_n^{\underline{o}} = 0$, so condition (9) implies that $\tilde{v}_n^{x_n(v,A)} \geq v_n^{x_n(v,A)}$.

XBONE is a property of allocation algorithms—it is defined without reference to the payment rule. Nevertheless, when an algorithm x is paired with an incentive-compatible payment rule p, then the requirement that the algorithm x is XBONE can be restated in a way that associates the externality with the mechanism and corresponds closely to the conventional definition of externalities.

Proposition 3.4. *If (v, A) and $(\tilde{v}_n, v_{-n}, A)$ satisfy (9) and (x, p) is strategy-proof, then (10) is equivalent to the requirement that*

$$\underbrace{p_n(\tilde{v}_n, v_{-n}, A) - p_n(v, A)}_{\textit{change in } n\textit{'s payment}} + \underbrace{\sum_{m \neq n} v_m \cdot [x_m(\tilde{v}_n, v_{-n}, A) - x_m(v, A)]}_{\textit{effect on others' values}} \geq 0, \tag{11}$$

Moreover, if (x, p) is strategy-proof, then for almost all pairs $(v_n, \tilde{v}_n) \in \mathbb{R}^{2|O|}$, if v_n and $\tilde{v}_n$ satisfy (9), then we have $p_n(\tilde{v}_n, v_{-n}, A) - p_n(v, A) = 0$.

Expression (11) decomposes the effect of moving from v_n to $\tilde{v}_n$ into a change in n's payment and an effect on the total value allocated to other bidders. In total, the left-hand side is the net externality from the mechanism, that is, the portion of the effect on other participants that is not fully reflected in the price.[14] When condition (9) of the XBONE definition applies, changing n's report from v_n to $\tilde{v}_n$ while holding n's value fixed has no net effect on n's payoff. Thus, using a notion of bossy mechanisms based on payoffs rather than outcomes, (11) quantifies the impact of a *bossy* externality and requires it to be non-negative.

As before, XBONE allows us to carry over approximation guarantees for allocation into the investment problem.

Theorem 3.1. *Assume that x is W-Mon and that V_n^A is a product of one-dimensional intervals for all A and n. If x is XBONE and is a β-approximation for allocation, then x is a β-approximation for investment.*

Theorem 3.1 extends Theorem 2.1 to a much more general model that includes multiple outcomes. Almost everywhere, if a bidder's marginal value for his original outcome rises compared to every other outcome, then the bidder's outcome remains unchanged. If such a change affects others' outcomes, that is a bossy externality. Theorem 3.1 tells us that if the algorithm excludes bossy negative externalities, then the long-run problem inherits the worst-case guarantee from the short-run problem.

[14]In the mechanism design literature, the word "externality" is often used to refer just to the second term, but that is different from the traditional economic use of the word.

3.4.1 Proof of Theorem 3.1

As in the theorem statement, suppose that x is W-Mon, XBONE, and a β-approximation for allocation and suppose moreover that each V_n^A is a product of one-dimensional intervals. We define a **pivotal vector** $\overline{v}_\iota$ that plays a key role in the argument. For each outcome $o \in O$, the corresponding component of the pivotal vector is

$$\overline{v}_\iota^o = \max_{(v_\iota, c_\iota) \in I_\iota} \left\{ v_\iota^o - c_\iota \right\}. \tag{12}$$

As I_ι is normalized and V_ι^A is a product of one-dimensional intervals, we have $\overline{v}_\iota \in V_\iota^A$ by construction.

We begin by showing that the investor ι can find a best-response using the following simple procedure:

1. Construct the pivotal vector $\overline{v}_\iota$.

2. Check what outcome would occur if he reported the pivotal vector to the mechanism, this is $x_\iota(\overline{v}_\iota, v_{-\iota}, A)$.

3. Choose an investment that maximizes his value, net of costs, for $x_\iota(\overline{v}_\iota, v_{-\iota}, A)$.

The next lemma formalizes the procedure just described.

Lemma 3.1. *For any instance $(I_\iota, v_{-\iota}, A)$, it is a best-response for ι to choose (v_ι, c_ι) to maximize*

$$v_\iota^{x_\iota(\overline{v}_\iota, v_{-\iota}, A)} - c_\iota.$$

Proof. Bidder ι's best response corresponds to the maximization

$$\max_{(v_\iota, c_\iota) \in I_\iota} \left\{ v_\iota \cdot x_\iota(v_\iota) - p_\iota^x(v_\iota) - c_\iota \right\}. \tag{13}$$

As (x, p^x) is strategy-proof,

$$v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota)$$

is maximized by taking $\tilde{v}_\iota = v_\iota$; hence, we can rewrite the maximand in (13) to yield

$$\max_{(v_\iota, c_\iota) \in I_\iota} \max_{\tilde{v}_\iota} \left\{ v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota) - c_\iota \right\}. \tag{14}$$

Changing the order of maximization in (14) then gives us

$$\max_{\tilde{v}_\iota} \max_{(v_\iota, c_\iota) \in I_\iota} \left\{ v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota) - c_\iota \right\}.$$

Now, by our construction of $\overline{v}_\iota$, for all $\tilde{v}_\iota \in V_\iota^A$, we have

$$\max_{(v_\iota, c_\iota)\in I_\iota} \{v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota) - c_\iota\} = \overline{v}_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota), \tag{15}$$

as $x_\iota(\tilde{v}_\iota) \in O$. As (x, p^x) is strategy-proof, setting $\tilde{v}_\iota = \overline{v}_\iota$ maximizes the right-hand side of (15), and so also maximizes the left-hand side of (15). This reduces ι's problem to the maximization

$$\max_{(v_\iota, c_\iota)\in I_\iota} \{v_\iota \cdot x_\iota(\overline{v}_\iota) - p_\iota^x(\overline{v}_\iota) - c_\iota\} = \max_{(v_\iota, c_\iota)\in I_\iota} \{v_\iota \cdot x_\iota(\overline{v}_\iota) - c_\iota\} - p_\iota^x(\overline{v}_\iota). \tag{16}$$

Dropping the term in (16) that does not depend on (v_ι, c_ι) yields

$$\max_{(v_\iota, c_\iota)\in I_\iota} \{v_\iota \cdot x_\iota(\overline{v}_\iota) - c_\iota\},$$

which gives us Lemma 3.1. □

Lemma 3.2. *For any instance $(I_\iota, v_{-\iota}, A)$, we have*

$$\overline{W}^*(I_\iota, v_{-\iota}, A) = W^*(\overline{v}_\iota, v_{-\iota}, A).$$

Proof. We have

$$\begin{aligned}
\overline{W}^*(I_\iota, v_{-\iota}, A) &= \max_{(v_\iota, c_\iota)\in I_\iota} \max_{a\in A} \{w(a \mid v_\iota, v_{-\iota}) - c_\iota)\} \\
&= \max_{a\in A} \max_{(v_\iota, c_\iota)\in I_\iota} \{w(a \mid v_\iota, v_{-\iota}) - c_\iota)\} \\
&= \max_{a\in A} \{w(a \mid \overline{v}_\iota, v_{-\iota})\} \\
&= W^*(\overline{v}_\iota, v_{-\iota}, A).
\end{aligned}$$

□

Now, with Lemma 3.1 and Lemma 3.2, we can proceed with the proof of Theorem 3.1. By the same argument as in the proof of Lemma 2.2, we can restrict attention to proving the desired bound for instances with singleton best-responses. We let $(\hat{v}_\iota, \hat{c}_\iota) \in \mathrm{BR}(x, I_\iota, v_{-\iota}, A)$ denote ι's best-response.

We now prove that moving from $\overline{v}_\iota$ to $\hat{v}_\iota$ satisfies the antecedent condition of Definition 3.2: For all outcomes o, we have

$$\begin{aligned}\hat{v}_\iota^{x_\iota(\overline{v}_\iota)} - \hat{v}_\iota^o &= \left(\hat{v}_\iota^{x_\iota(\overline{v}_\iota)} - \hat{c}_\iota\right) - \left(\hat{v}_\iota^o - \hat{c}_\iota\right) \\ &\geq \max_{(v_\iota, c_\iota)\in I_\iota} \left\{v_\iota^{x_\iota(\overline{v}_\iota)} - c_\iota\right\} - \max_{(v_\iota, c_\iota)\in I_\iota} \left\{v_\iota^o - c_\iota\right\} \\ &= \overline{v}_\iota^{x_\iota(\overline{v}_\iota)} - \overline{v}_\iota^o,\end{aligned}$$

where the inequality follows from Lemma 3.1, given that $(\hat{v}_\iota, \hat{c}_\iota) \in \mathrm{BR}(x, I_\iota, v_{-\iota}, A)$ is a best response. Thus, as x is XBONE, we have that

$$W_x(\hat{v}_\iota) = w(x(\hat{v}_\iota) \mid \hat{v}_\iota) \geq w(x(\overline{v}_\iota) \mid \hat{v}_\iota). \tag{17}$$

Now, by our construction of the pivotal vector $\overline{v}_\iota$ in (12) and by Lemma 3.1, we have

$$\hat{v}_\iota^{x_\iota(\overline{v}_\iota)} - \hat{c}_\iota = \overline{v}_\iota^{x_\iota(\overline{v}_\iota)}$$

which implies

$$w(x(\overline{v}_\iota) \mid \hat{v}_\iota) - \hat{c}_\iota = w(x(\overline{v}_\iota) \mid \overline{v}_\iota) = W_x(\overline{v}_\iota). \tag{18}$$

Subtracting $\hat{c}_\iota$ from (17) and applying (18), we find that

$$W_x(\hat{v}_\iota) - \hat{c}_\iota \geq W_x(\overline{v}_\iota). \tag{19}$$

Combining the preceding steps, we see that

$$\overline{W}_x(I_\iota) = \overbrace{W_x(\hat{v}_\iota) - \hat{c}_\iota \geq W_x(\overline{v}_\iota)}^{(19)} \geq \overbrace{\beta W^*(\overline{v}_\iota)}^{\text{Lemma } 3.2}\!\underbrace{\phantom{W_x(\overline{v}_\iota) \geq \beta W^*(\overline{v}_\iota)}}_{\beta\text{-approx for allocation}} = \beta\overline{W}^*(I_\iota),$$

which shows that x is a β-approximation for investment, as desired.

3.5 Combinatorial auctions

Theorem 3.1 relies on each bidder's values for different outcomes having a product structure. In a combinatorial auction, an outcome consists of a bundle of goods and common assumptions in such analyses are incompatible with a product structure on the possible values of bundles. For instance, if a bidder's value function is additive, then knowing his value for each singleton bundle exactly pins down his value for the grand bundle. In such cases, Theorem 3.1 fails to apply. In this section, we develop an extension that accommodates a standard class of preferences for combinatorial auctions.

An **allocation instance** consists of:

1. a finite set of **bidders** N;
2. a finite set of **goods** G; and
3. for each $n \in N$, a **value function** $v_n : \wp(G) \to \mathbb{R}$.

We write v for a profile of value functions; (v, G) denotes an instance. An **allocation problem** Ω is a collection of allocation instances. An algorithm x selects for each (v, G) a bundle of goods, one for each bidder, $x(v, G) \in (\wp(G))^N$. We require that no good is allocated twice, that is, for all $n \neq n'$, we have $x_n(v, G) \cap x_{n'}(v, G) = \emptyset$.

Correspondingly, an **investment instance** consists of:

1. a **cost function** for the investing bidder, $c_\iota : V_\iota \to \mathbb{R}$, for some domain of value functions V_ι;
2. a profile of value functions for the other bidders, $v_{-\iota}$; and
3. a set of goods G.

As before, the investing bidder ι faces a strategy-proof mechanism (x, p^x), and chooses an investment $v_\iota \in V_\iota$.

When value functions are fully general, a bidder's preferences are described by $|\wp(G)|$ real numbers, and it is computationally infeasible even to approximate the optimum. Hence, we study allocation and investment under fractionally subadditive value functions. These are a canonical class of preferences, for which there are known fast algorithms with non-trivial guarantees (Nisan, 2000; Feige, 2009). The class includes all submodular functions, as well as all functions that have the gross substitutability property (Lehmann et al., 2006a; Paes Leme, 2017).

Definition 3.3. Value function $v_n(\cdot)$ is **additive** if there exists $\alpha \in (\mathbb{R}_0^+)^G$ such that for all $F \subseteq G$,

$$v_n(F) = \sum_{g \in F} \alpha_g.$$

In the case that a bidder's value function is additive with parameter vector α, we abuse notation, and use α to denote the value function itself.

Value function $v_n(\cdot)$ is **fractionally sub-additive (XOS)** if there exists a family of additive value functions $(\alpha^\ell)_{\ell \in L}$ such that for all $F \subseteq G$,

$$v_n(F) = \max_\ell \alpha^\ell(F).$$

We denote by XOS the set of all XOS value functions.

We restrict attention to allocation problems such that bidders can have any XOS preferences, that is, for all (v_{-n}, G),

$$\{v_n : (v_n, v_{-n}, G) \in \Omega\} = \mathsf{XOS}.$$

We restrict attention to cost functions c_ι such that, for each investment instance (c_ι, v_{-n}, G):

1. the investor's best-response set is non-empty;
2. the set of socially optimal investments is non-empty;
3. $V_\iota = \mathsf{XOS}$; and
4. if for all $F \subseteq G$, $v_\iota(F) = 0$, then $c_\iota(v_\iota) = 0$.

Definition 3.4. Cost function $c_\iota(\cdot)$ is **isotone** if for any $v_\iota, \tilde{v}_\iota \in V_\iota$, if $v_\iota(F) \geq \tilde{v}_\iota(F)$ for all $F \subseteq G$, then $c_\iota(v_n) \geq c_\iota(\tilde{v}_\iota)$.

Definition 3.5. For any $\alpha, \alpha' \in (\mathbb{R}_0^+)^G$, let $\alpha \vee \alpha' = (\max\{\alpha_g, \alpha'_g\})_{g \in G}$, and let $\alpha \wedge \alpha' = (\min\{\alpha_g, \alpha'_g\})_{g \in G}$. Cost function $c_\iota(\cdot)$ is **supermodular on additive valuations** if for any $\alpha, \alpha' \in (\mathbb{R}_0^+)^G$ we have

$$c_\iota(\alpha \vee \alpha') + c_\iota(\alpha \wedge \alpha') \geq c_\iota(\alpha) + c_\iota(\alpha').$$

We extend the definitions of W-Mon and XBONE to combinatorial auctions, by regarding each bundle of goods as an outcome.

Theorem 3.2. *Assume that x is W-Mon, and restrict c_ι to be isotone and supermodular on additive valuations. If x is XBONE and is a β-approximation for allocation, then x is a β-approximation for investment.*

Proof. Given some investment instance $(c_\iota, v_{-\iota}, G)$, let the pivotal value function $\overline{v}_\iota$ be defined by

$$\overline{v}_\iota(F) \equiv \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\}$$

for all $F \subseteq G$.

Lemma 3.3. *If c_ι is isotone and supermodular on additive valuations, then $\overline{v}_\iota \in \mathsf{XOS}$.*

We once again suppress the dependence of functions on $v_{-\iota}$ and G.

We now note that, by the same argument as in Lemma 3.1, in any instance $(c_\iota, v_{-\iota}, G)$, choosing $\hat{v}_\iota$ to maximize $v_\iota(x_\iota(\overline{v}_\iota)) - c_\iota(v_\iota)$ is a best-response for ι. And by the same argument as in Lemma 2.2, we can restrict attention to proving the bound for instances with singleton best-response sets.

By Lemma 3.3, $\overline{v}_\iota \in \mathsf{XOS}$. Thus, as x is a β-approximation for allocation, $W_x(\overline{v}_\iota) \geq \beta W^*(\overline{v}_\iota)$. Moreover, just as in the proof of Theorem 3.1, the fact that x is XBONE implies that

$$W_x(\hat{v}_\iota) - c_\iota(\hat{v}_\iota) \geq W_x(\overline{v}_\iota). \tag{20}$$

We then have

$$\overline{W}_x(c_\iota) = \overbrace{W_x(\hat{v}_\iota) - c_\iota(\hat{v}_\iota) \geq W_x(\overline{v}_\iota)}^{(20)} \underbrace{\geq \beta W^*(\overline{v}_\iota)}_{\beta\text{-approx for allocation}} \overbrace{= \beta \overline{W}^*(c_\iota)}^{\text{Lemma 3.2}},$$

which completes the proof. $\square$

4 Discussion

Market design analyses by economists frequently seek to maximize some objective exactly, which often leads to mechanisms that must compute optima in order to select allocations (and sometimes also to determine prices). Yet computing optima exactly is only possible for problems that are sufficiently small or have special structure; many other resource allocation problems are computationally intractable.

To get around computational challenges in market design applications, we have started using "nearly-optimal" mechanisms based on the large corpus of fast approximation algorithms developed by computer scientists. But then it is natural to ask: *What are the consequences of approximation for the larger economic systems in which our mechanisms are used? In particular, what happens to participants' investment incentives?*

The analysis in this paper suggests that the economic consequences of approximation can be subtle. With truthful implementation, nearly-optimal allocation rules can lead to arbitrarily bad long-run investment incentives. The key problem is that approximation algorithms may introduce a new type of externality, under which a bidder's investment that does not affect the bidder's own outcome may nevertheless cause the algorithm to select a different approximate optimum that reduces the value of other bidders' outcomes. Our XBONE condition rules out such bossy negative externalities, which ensures that short-run approximation guarantees apply equally to the long-run problem with investment. Notably, although we have defined bossy negative externalities in terms of a mechanism's allocation rule alone—without direct reference to the pricing rule—these externalities correspond exactly to the economic bossy negative externalities in the associated truthful mechanism, that is, to failures of the prices quoted to a bidder to capture the bossy part of that bidder's negative impact on others' welfare.

The analysis in this paper raises questions for further study. For example:

- We have so far focused on investment when the investor knows the prices he faces. How, if at all, does the analysis extend to cases in which those prices are unknown? What properties must an allocation algorithm have to retain its performance when a bidder can only guess about prices when he makes an investment decision?

- We have analyzed deterministic algorithms, but computer scientists sometimes recommend randomized algorithms. Does the analysis extend to randomized algorithms, with an appropriate generalization of XBONE?

- Does requiring an allocation algorithm to be XBONE raise significant new computational hurdles? Or is it possible to modify existing algorithms to satisfy the XBONE property? For example, given oracle access to some monotone allocation algorithm, is there a polynomial-time procedure that outputs a monotone XBONE allocation algorithm with a weakly better approximation ratio?

More broadly, replacing exact optimization with approximation can have many consequences beyond investment. For example, approximation can affect how participants understand mechanisms in practice, raise new opportunities for coordination or collusion, and/or influence post-auction resale markets. Given the close connection between monotone algorithms and truthful mechanisms, it seems possible—and important—to analyze how these and other economic properties correspond to properties of the underlying algorithms themselves.

References

Arozamena, L. and E. Cantillon (2004): "Investment incentives in procurement auctions," *Review of Economic Studies*, 71, 1–18.

Bergemann, D. and J. Välimäki (2002): "Information Acquisition and Efficient Mechanism Design," *Econometrica*, 70, 1007–1033.

Bienstock, D. and A. Verma (2019): "Strong NP-hardness of AC power flows feasibility," *Operations Research Letters*, 47, 494–501.

Bikhchandani, S., S. Chatterji, R. Lavi, A. Mu'alem, N. Nisan, and A. Sen (2006): "Weak Monotonicity Characterizes Deterministic Dominant-Strategy Implementation," *Econometrica*, 74, 1109–1132.

Bikhchandani, S., S. De Vries, J. Schummer, and R. V. Vohra (2011): "An ascending vickrey auction for selling bases of a matroid," *Operations Research*, 59, 400–413.

Clarke, E. H. (1971): "Multipart Pricing of Public Goods," *Public Choice*, 11, 17–33.

Cook, S. (2006): "The P versus NP problem," in *The Millennium Prize Problems*, ed. by J. A. Carlson, A. Jaffe, and A. Wiles, American Mathematical Society Providence, 87–104.

Cook, S. A. (1971): "The complexity of theorem-proving procedures," in *Proceedings of the third annual ACM symposium on Theory of computing*, 151–158.

Dantzig, G. B. (1957): "Discrete-variable extremum problems," *Operations research*, 5, 266–288.

Dughmi, S., J. D. Hartline, R. Kleinberg, and R. Niazadeh (2017): "Bernoulli factories and black-box reductions in mechanism design," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, 158–169.

Feige, U. (2009): "On maximizing welfare when utility functions are subadditive," *SIAM Journal on Computing*, 39, 122–142.

Fortnow, L. (2009): "The status of the P versus NP problem," *Communications of the ACM*, 52, 78–86.

Groves, T. (1973): "Incentives in Teams," *Econometrica*, 41, 617–631.

Hartline, J. D. (2016): "Mechanism design and approximation," *Book draft*, 122.

Hartline, J. D. and B. Lucier (2015): "Non-optimal mechanism design," *American Economic Review*, 105, 3102–24.

Hatfield, J. W., F. Kojima, and S. D. Kominers (2014): "Investment Incentives in Labor Market Matching," *American Economic Review Papers & Proceedings*, 104, 436–441.

——— (2019): "Strategy-Proofness, Investment Efficiency, and Marginal Returns: An Equivalence," Becker Friedman Institute Working Paper.

Karp, R. M. (1972): "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, ed. by R. E. Miller and J. W. Thatcher, Springer, 85–103.

Lavaei, J. and S. H. Low (2011): "Zero duality gap in optimal power flow problem," *IEEE Transactions on Power Systems*, 27, 92–107.

Lavi, R., A. Mu'Alem, and N. Nisan (2003): "Towards a characterization of truthful combinatorial auctions," in *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, 574–583.

Lehmann, B., D. Lehmann, and N. Nisan (2006a): "Combinatorial auctions with decreasing marginal utilities," *Games and Economic Behavior*, 55, 270–296.

Lehmann, D., R. Müller, and T. Sandholm (2006b): "The Winner Determination Problem," in *Combinatorial Auctions*, ed. by P. Cramton, Y. Shoham, and R. Steinberg, MIT Press, 297–318.

Lehmann, D., L. I. Oćallaghan, and Y. Shoham (2002): "Truth revelation in approximately efficient combinatorial auctions," *Journal of the ACM*, 49, 577–602.

Leyton-Brown, K., P. Milgrom, and I. Segal (2017): "Economics and computer science of a radio spectrum reallocation," *Proceedings of the National Academy of Sciences*, 114, 7202–7209.

Lipsey, R. and K. Lancaster (1956): "The General Theory of Second Best," *Review of Economic Studies*, 24, 11–32.

Milgrom, P. (2017): *Discovering Prices*, Columbia University Press.

Milgrom, P. and I. Segal (2002): "Envelope theorems for arbitrary choice sets," *Econometrica*, 70, 583–601.

——— (2020): "Clock auctions and radio spectrum reallocation," *Journal of Political Economy*, 128, 1–31.

Mu'Alem, A. and N. Nisan (2008): "Truthful approximation mechanisms for restricted combinatorial auctions," *Games and Economic Behavior*, 64, 612–631.

Myerson, R. B. (1981): "Optimal auction design," *Mathematics of Operations Research*, 6, 58–73.

Nisan, N. (2000): "Bidding and allocation in combinatorial auctions," in *Proceedings of the 2nd ACM Conference on Electronic Commerce*, 1–12.

Nisan, N. and A. Ronen (2007): "Computationally feasible VCG mechanisms," *Journal of Artificial Intelligence Research*, 29, 19–47.

Paes Leme, R. (2017): "Gross substitutability: An algorithmic survey," *Games and Economic Behavior*, 106, 294–316.

Pardalos, P. M., D.-Z. Du, and R. L. Graham (2013): *Handbook of Combinatorial Optimization*, Springer.

Rogerson, W. P. (1992): "Contractual solutions to the hold-up problem," *Review of Economic Studies*, 59, 777–793.

Saks, M. and L. Yu (2005): "Weak monotonicity suffices for truthfulness on convex domains," in *Proceedings of the 6th ACM Conference on Electronic Commerce*, 286–293.

Sandholm, T. (2002): "Algorithm for optimal winner determination in combinatorial auctions," *Artificial Intelligence*, 135, 1–54.

Tomoeda, K. (2019): "Efficient investments in the implementation problem," *Journal of Economic Theory*, 182, 247–278.

Vazirani, V. V. (2013): *Approximation Algorithms*, Springer Science & Business Media.

Vickrey, W. (1961): "Counterspeculation, auctions, and competitive sealed tenders," *The Journal of Finance*, 16, 8–37.

Williamson, D. P. and D. B. Shmoys (2011): *The Design of Approximation Algorithms*, Cambridge University Press.

A Proofs omitted from the main text

Proof of Lemma 2.1

Suppose (x, p) provides marginal rewards. By inspection of (1), (x, p) has efficient investments.

Suppose (x, p) does not provide marginal rewards. Let $(v_\iota, v_{-\iota}, A)$ and $(v'_\iota, v_{-\iota}, A)$ be a pair of allocation instances such that (2) does not hold. Consider the investment technology $I_\iota = \{(v_\iota, c_\iota), (v'_\iota, c'_\iota)\}$, such that

$$c_\iota - c'_\iota = [v_\iota \cdot x_\iota(v_\iota, v_{-\iota}, A) - p(v_\iota, v_{-\iota}, A)] - [v'_\iota \cdot x_\iota(v'_\iota, v_{-\iota}, A) - p(v'_\iota, v_{-\iota}, A)] \,.$$

We then have by construction that

$$\operatorname*{argmax}_{(\hat{v}_\iota, \hat{c}_\iota) \in I_\iota} \{\hat{v}_\iota \cdot x_\iota(\hat{v}_\iota, v_{-\iota}, A) - p(\hat{v}_\iota, v_{-\iota}, A) - \hat{c}_\iota\} = I_i \neq \operatorname*{argmax}_{(\hat{v}_\iota, \hat{c}_\iota) \in I_\iota} \{W_x\left(\hat{v}_\iota, v_{-\iota}, A\right) - \hat{c}_\iota\} \,,$$

so (x, p) does not have efficient investments.

Proof of Proposition 2.3

Lemma A.1. *If x is monotone and has range-efficient allocations, then (x, p) provides marginal rewards if and only if (x, p) is strategy-proof.*

Proof. Suppose x is monotone and has range-efficient allocations.

By x monotone and the envelope theorem (Milgrom and Segal, 2002), (x, p) is strategy-proof if and only if for all $v_\iota > v'_\iota$, all $v_{-\iota}$, and all A, we have that

$$[v_\iota \cdot x_\iota(v_\iota, v_{-\iota}, A) - p(v_\iota, v_{-\iota}, A)] - [v'_\iota \cdot x_\iota(v'_\iota, v_{-\iota}, A) - p(v'_\iota, v_{-\iota}, A)] = \int_{v'_\iota}^{v_\iota} x_\iota(t, v_{-\iota}, A) dt \quad (21)$$

Since x has range-efficient allocations, for all A, there exists $R_A \subseteq A$ such that for all v we have $W_x(v, A) = \max_{a \in R_A} w(a \mid v)$. Let $\tilde{x}(v, A)$ be an arbitrary selection from $\operatorname{argmax}_{a \in R_A} w(a \mid v)$.

Observe that for all $v_\iota > v'_\iota$, all $v_{-\iota}$, and all A, we have that

$$\int_{v'_\iota}^{v_\iota} x_\iota(t, v_{-\iota}, A) dt = \int_{v'_\iota}^{v_\iota} \tilde{x}_\iota(t, v_{-\iota}, A) dt = \max_{a \in R_A} \{w(a \mid v_\iota, v_{-\iota})\} - \max_{a \in R_A} \{w(a \mid v'_\iota, v_{-\iota})\}$$
$$= W_x(v_\iota, v_{-\iota}, A) - W_x(v'_\iota, v_{-\iota}, A) \quad (22)$$

where the first equality is by range-efficient allocations[15], the second equality is by the envelope theorem, and the third equality is by definition of R_A.

[15] Range-efficient allocations implies that $x_\iota(t, v_\iota, A) = \tilde{x}_\iota(t, v_\iota, A)$ almost-everywhere in $[v'_\iota, v_\iota]$.

Substituting (22) into (21) yields the marginal rewards condition, so (x, p) is strategy-proof if and only if (x, p) provides marginal rewards. $\square$

Together, Lemma 2.1 and Lemma A.1 prove Clause 1 and Clause 2 of Proposition 2.3.[16]

What remains is to prove Clause 3 of Proposition 2.3. We suppose that (x, p) is strategy-proof and has efficient investments. In the argument that follows, we fix some A and suppress the dependence on A henceforth.

Lemma A.2. *If (x, p) is strategy-proof and has efficient investments, then W_x is continuous in v.*

Proof. Suppose (x, p) is strategy-proof and has efficient investments. By Lemma 2.1, there exists some function ζ that does not depend on v_n, such that for all v_n and v_{-n}:

$$\begin{aligned} & W_x(v_n, v_{-n}) + \zeta(v_{-n}) && (23) \\ & = v_n \cdot x(v_n, v_{-n}) - p(v_n, v_{-n}) \\ & = \max_{\hat{v}_n} v_n \cdot x(\hat{v}_n, v_{-n}) - p(\hat{v}_n, v_{-n}) && \text{by } (x,p) \text{ strategy-proof} \end{aligned}$$

By the envelope theorem, the last line of (23) is 1-Lipschitz in v_n, so W_x is 1-Lipschitz in v_n. This argument holds for all n, so W_x is continuous in v. $\square$

We introduce a *modified range* for x, denoted $R \subseteq A$, that contains any r in the range of x such there exists v for which $r = x(v)$ and $w(r|v) \neq w(r'|v)$ for any $r' \neq r$.

Lemma A.3. *If (x, p) is strategy-proof and has efficient investments, then for any v, there exists an $r \in R$ such that $w(r|v) = W_x(v)$.*

Proof. Since W_x is continuous by Lemma A.2, for any $\epsilon > 0$ there exists a $\delta > 0$ such that if $|v - v'| < \delta$, then $|W_x(v) - W_x(v')| < \epsilon$. We can pick v' such that $|v'_n - v_n| < \frac{\min(\delta, \epsilon)}{|N|}$ for all $n \in N$ and—since the set of sums $\{\sum_{n \in J'} v'_n\}_{J \subseteq N}$ is finite—so that we have, for any $J, J' \subseteq N$,

$$\sum_{n \in J} v'_n = \sum_{n \in J'} v'_n \iff J = J'. \tag{24}$$

By $r = x(v') \in R$ by our choice of R, and

$$|w(r|v) - W_x(v)| \leq |w(r|v) - W_x(v')| + |W_x(v') - W_x(v)| \leq \epsilon + \epsilon. \tag{25}$$

[16] As noted in Footnote 8, these two clauses are essentially corollaries of Theorem 1 of Hatfield et al. (2019), an argument here adapts the Hatfield et al. (2019) approach to our setting.

Now, R is a finite set and ϵ can be arbitrarily small, so (25) proves the lemma. $\square$

Now, we show that

$$W_x(v) = \max_{r \in R} w(r|v). \tag{26}$$

To see (26), we assume for the sake of contradiction that there exists a v^0 such that

$$W_x(v^0) \neq \max_{r \in R} w(r|v^0).$$

By Lemma A.3, we know that there is some $r \in R$ such that $W_x(v^0) = w(r|v^0)$; hence, since $W_x(v^0) \neq \max_{r \in R}\{w(r|v^0)\}$, we must have

$$W_x(v^0) < \max_{r \in R}\{w(r|v^0)\}. \tag{27}$$

Both sides of (27) are continuous (the left side by Lemma A.2), so there exists an $\epsilon > 0$, such that $\|v - v^0\| < \epsilon$ implies that (27) holds for v. Therefore we can choose v so that (27) and (27) hold simultaneously.

Now, we let $r = x(v)$ and $r' \in \mathrm{argmax}_{r'' \in R}\, w(r''|v)$. Since $r' \in R$ there exists a $\tilde{v}$ and $\epsilon > 0$ such that $\|v' - \tilde{v}\| < \epsilon$ implies $r' = x(v')$. So we can choose v' so that $x(v') = r'$ and

$$\sum_{n \in J} v_n^* = \sum_{n \in J'} v_n^* \iff J = J'$$

for any v^* such that $v_n^* \in \{v_\iota, v_\iota'\}$.

We construct a new value profile v'' as follows:

$$v_n'' = \begin{cases} \max(v_n, v_n') & n \in x(v') \\ \min(v_n, v_n') & n \notin x(v'). \end{cases}$$

By weak monotonicity of x and since (24) holds at each step, we have $x(v'') = r'$. Now, we create a directed line segment from v'' to v and suppose that along it one encounters a value profile with allocation $r'' \neq r' \in R$ under x. Let $\hat{v}$ be the switching boundary for the new decision. Consider the linear functions $f(\tilde{v}) = w(r', \tilde{v}) - w(r'', \tilde{v})$ and $f_n(\tilde{v}_n) = (\mathbb{1}_{j \in r'} - \mathbb{1}_{j \in r''})\, \tilde{v}_n$. We have $f(v) > 0$ since $r' \in \mathrm{argmax}_{r \in R}\, w(r|v)$ and (24) holds at v. By construction, we have

$$f(v'') = \sum_{n \in N} f_n(v_n'') \geq \sum_{n \in N} f_n(v_n) = f(v) > 0.$$

So we must have $f(\hat{v}) > 0$ along the entire interval. But continuity of W_x requires that $f(\hat{v}) = 0$ which is a contradiction, so no such $r'' \neq r' \in R$ can arise along the line. Hence, we must have $x(v) = r'$, which contradicts our assumption that $x(v) = r \neq r'$. Thus, we have (26), as desired, establishing that (x, p) has range-efficient allocations.

Proof of Lemma 2.2

We prove the contrapositive: Suppose x is not a β-approximation for investment. Then there exists some $(I_\iota, v_{-\iota}, A)$ such that

$$\beta \overline{W}^*(I_\iota, v_{-\iota}, A) > \overline{W}_x(I_\iota, v_{-\iota}, A).$$

We now modify I_ι to ensure that ι's best-response is singleton. Let

$$(\hat{v}_\iota, \hat{c}_\iota) \in \underset{(v_\iota, c_\iota) \in \mathrm{BR}(x, I_\iota, v_{-\iota}, A)}{\operatorname{argmin}} \left\{ W_x(v_\iota, v_{-\iota}, A) - c_\iota \right\}.$$

For $\delta > 0$, let I_ι^δ be the investment technology produced by raising by δ the cost of all investments except $(\hat{v}_\iota, \hat{c}_\iota)$, and then re-normalizing the costs so that

$$\min \left\{ c_\iota : (v_\iota, c_\iota) \in I_\iota^\delta \right\} = 0.$$

Now $\mathrm{BR}(x, I_\iota^\delta, v_{-\iota}, A) = \{(\hat{v}_\iota, \hat{c}_\iota)\}$ by construction, making it a singleton. Moreover, in constructing I_ι^δ, each investment's cost has changed by no more than δ. Thus,

$$\overline{W}^*(I_\iota^\delta, v_{-\iota}, A) \geq \overline{W}^*(I_\iota, v_{-\iota}, A) - \delta$$
$$\overline{W}_x(I_\iota, v_{-\iota}, A) + \delta \geq \overline{W}_x(I_\iota^\delta, v_{-\iota}, A).$$

For small enough δ, we then have

$$\beta \overline{W}^*(I_\iota^\delta, v_{-\iota}, A) > \overline{W}_x(I_\iota^\delta, v_{-\iota}, A),$$

which completes the proof of the contrapositive.

Proof of Theorem 2.2

Proof. The proof of Theorem 2.1 established that

$$\overline{W}_x(I_\iota, v_{-\iota}, A) \geq \beta \overline{W}^*(I_\iota, v_{-\iota}, A) \tag{28}$$

in two cases:

1. ι chooses $(v_\iota^\uparrow, c_\iota^\uparrow)$ and $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow)$; and
2. ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$ and $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$.

To establish (28) under the assumption that x is weakly XBONE, we consider three cases:

1. ι chooses $(v_\iota^\uparrow, c_\iota^\uparrow)$ and $\iota \in x(v_\iota^\uparrow - c_\iota^\uparrow)$;

2a. ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$, $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$, and $v_\iota^\uparrow - c_\iota^\uparrow > t_n^{\mathrm{OPT}}(v_\iota^\uparrow - c_\iota^\uparrow, v_{-n}, A)$

2b. ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$, $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$, and $v_\iota^\uparrow - c_\iota^\uparrow \le t_n^{\mathrm{OPT}}(v_\iota^\uparrow - c_\iota^\uparrow, v_{-n}, A)$

When x is weakly XBONE, the same arguments as in the proof of Theorem 2.1 work for Case 1 and Case 2a. For Case 2b, $v_\iota^\uparrow - c_\iota^\uparrow \le t_n^{\mathrm{OPT}}(v_\iota^\uparrow - c_\iota^\uparrow, v_{-n}, A)$ implies that there exists a welfare-maximizing allocation at $(v_\iota^\uparrow - c_\iota^\uparrow, v_{-n}, A)$ such that n is not packed, and thus that $W^*(v_\iota^\downarrow, v_{-\iota}, A) = W^*(v_\iota^\uparrow - c_\iota^\uparrow, v_{-\iota}, A)$. Thus we conclude that

$$\overline{W}_x(I_\iota, v_{-\iota}, A) = W_x(v_\iota^\downarrow, v_{-\iota}, A) \ge \beta W^*(v_\iota^\downarrow, v_{-\iota}, A) = \beta W^*(v_\iota^\uparrow - c_\iota^\uparrow, v_{-\iota}, A) \ge \beta \overline{W}^*(I_\iota, v_{-\iota}, A),$$

where the last inequality follows by (7). □

Proof of Theorem 2.3

Definition A.1. $W_x(\cdot, v_{-\iota}, A)$ is **lower semi-continuous** at v_ι if for all sequences $\{v_\iota^k\}_{k=1}^\infty$ such that $v_\iota^k \to v_\iota$, we have

$$\liminf_{v_\iota^k \to v_\iota} \left\{ W_x(v_\iota^k, v_{-\iota}, A) \right\} \ge W_x(v_\iota, v_{-\iota}, A).$$

Lemma A.4. *Assume x is monotone and a β-approximation for allocation on problem Ω for $\beta > 0$. Assume $W_x(\cdot, v_{-\iota}, A)$ is lower semi-continuous at v_ι. If there exists $\tilde{v}_\iota$ such that (v, A) and $(\tilde{v}_\iota, v_{-\iota}, A)$ do not satisfy the requirements of Definition 2.11, then there exists a sub-problem $\Omega' \subseteq \Omega$ and β' such that x is a β'-approximation for allocation on Ω', but not a β'-approximation for investment on Ω'.*

Proof. Suppose we have some (v, A) and $\tilde{v}_\iota$ that do not satisfy the requirements of Definition 2.11. As usual, we will suppress the dependence of functions on $v_{-\iota}$ and A. Let

$$\Omega' = \{(v'_\iota, v_{-\iota}, A) : v'_\iota \in [\min\{v_\iota, \tilde{v}_\iota\}, \max\{v_\iota, \tilde{v}_\iota\}]\}$$
$$\overline{\beta} = \sup\{\beta' : x \text{ is a } \beta'\text{-approximation for allocation on } \Omega'\}.$$

It is straightforward to check that x is a $\overline{\beta}$-approximation for allocation on Ω'. As x is a β-approximation for allocation on Ω and $\Omega' \subseteq \Omega$, $\overline{\beta} \geq \beta > 0$. As x is not XBONE on Ω', x is not optimal on Ω', so $\overline{\beta} < 1$.

Let $(\check{\epsilon}^k)_{k=1}^{\infty}$ denote a sequence such that $\check{\epsilon}^k > 0$ and $\lim_{k\to\infty} \check{\epsilon}^k = 0$. For all k, there exists $\check{v}_\iota^k \in [\min\{v_\iota, \tilde{v}_\iota\}, \max\{v_\iota, \tilde{v}_\iota\}]$ such that $(\overline{\beta} + \check{\epsilon}^k)W^*(\check{v}_\iota^k) > W_x(\check{v}_\iota^k)$. The sequence $\{\check{v}_\iota^k, W_x(\check{v}_\iota^k)\}_{k=1}^{\infty}$ is bounded. Thus, by the Bolzano–Weierstrass theorem, we can pick subsequences $(\epsilon^k)_{k=1}^{\infty}$ and $(v_\iota^k)_{k=1}^{\infty}$ such that both terms converge, where we denote $v_\iota^{\infty} = \lim_{k\to\infty} v_\iota^k$ and $\sigma_x^{\infty} = \lim_{k\to\infty} W_x(v_\iota^k)$. By continuity of $W^*(\cdot)$,

$$\lim_{k\to\infty} W^*(v_\iota^k) = W^*(v_\iota^{\infty}).$$

As for all k,

$$\overline{\beta} W^*(v_\iota^k) \leq W_x(v_\iota^k) \leq (\overline{\beta} + \epsilon^k) W^*(v_\iota^k),$$

it follows that $\overline{\beta} \lim_{k\to\infty} W^*(v_\iota^k) = \sigma_x^{\infty}$.

We will check four cases that are jointly exhaustive, and show that in each case x is not a $\overline{\beta}$-approximation for investment on Ω'.

Case 1: Suppose the first clause of Definition 2.11 is not satisfied, so there exists (v, A) and $\tilde{v}_\iota$ such that $\iota \in x(v, A)$, $\tilde{v}_\iota > v_\iota$, and $W_x(\tilde{v}_\iota, v_{-\iota}, A) - W_x(v_\iota, v_{-\iota}, A) < \tilde{v}_\iota - v_\iota$. Either $\sigma_x^{\infty} - W_x(v_\iota) < v_\iota^{\infty} - v_\iota$, or $W_x(\tilde{v}_\iota) - \sigma_x^{\infty} < \tilde{v}_\iota - v_\iota^{\infty}$.[17]

Case 1a: Suppose $\sigma_x^{\infty} - W_x(v_\iota) < v_\iota^{\infty} - v_\iota$.

If $v_\iota^{\infty} = v_\iota$, we have $\sigma_x^{\infty} - W_x(v_\iota) = \lim_{k\to\infty} W_x(v_\iota^k) - W_x(v_\iota^{\infty}) \geq 0$, where the inequality follows by lower semi-continuity, a contradiction. Thus, $v_\iota^{\infty} > v_\iota$.

Consider the binary investment technology $I_\iota^k = \big\{(v_\iota, 0), (v_\iota^k, v_\iota^k - v_\iota)\big\}$. Observe that

$$\overline{W}_x(I_\iota^k) \leq W_x(v_\iota^k) - (v_\iota^k - v_\iota)$$
$$\overline{W}^*(I_\iota^k) \geq W^*(v_\iota^k) - (v_\iota^k - v_\iota).$$

Hence,

$$\overline{\beta} \liminf_{k\to\infty} \overline{W}^*(I_\iota^k) \geq \overline{\beta} \left(\lim_{k\to\infty} W^*(v_\iota^k) - (v_\iota^{\infty} - v_\iota) \right) > \sigma_x^{\infty} - (v_\iota^{\infty} - v_\iota) \geq \limsup_{k\to\infty} \overline{W}_x(I_\iota^k).$$

Case 1b: Suppose $W_x(\tilde{v}_\iota) - \sigma_x^{\infty} < \tilde{v}_\iota - v_\iota^{\infty}$.

[17] Suppose not; then $\sigma_x^{\infty} - W_x(v_\iota) \geq v_\iota^{\infty} - v_\iota$ and $W_x(\tilde{v}_\iota) - \sigma_x^{\infty} \geq \tilde{v}_\iota - v_\iota^{\infty}$, so $W_x(\tilde{v}_\iota) - W_x(v_\iota) \geq \tilde{v}_\iota - v_\iota$, a contradiction.

Consider the binary investment technology $I_\iota^k = \big\{(v_\iota^k, 0), (\tilde{v}_\iota, \tilde{v}_\iota - v_\iota^k)\big\}$. Observe that

$$\begin{aligned}\overline{W}_x(I_\iota^k) &\leq W_x(\tilde{v}_\iota) - (\tilde{v}_\iota - v_\iota^k)\\ \overline{W}^*(I_\iota^k) &\geq W^*(v_\iota^k).\end{aligned}$$

Hence,

$$\overline{\beta} \liminf_{k\to\infty} \overline{W}^*(I_\iota^k) \geq \overline{\beta} \lim_{k\to\infty} W^*(v_\iota^k) = \sigma_x^\infty > W_x(\tilde{v}_\iota) - (\tilde{v}_\iota - v_\iota^\infty) \geq \limsup_{k\to\infty} \overline{W}_x(I_\iota^k).$$

Case 2: Suppose Clause 2 of Definition 2.11 is not satisfied, so that

1. $\iota \notin x(v, A)$;
2. $\tilde{v}_\iota < v_\iota$;
3. $t_\iota^{\mathrm{OPT}}(v, A) < v_\iota$; and
4. $W_x(\tilde{v}_\iota) - W_x(v_\iota) < 0$.

There are two cases to consider; either $v_\iota^\infty < v_\iota$ or $v_\iota^\infty = v_\iota$.

Case 2a: Suppose $v_\iota^\infty < v_\iota$. Consider the technology $I_\iota^k = \big\{(v_\iota^k, 0), (v_\iota, 0)\big\}$.

$$\begin{aligned}\overline{W}_x(I_\iota^k) &\leq W_x(v_\iota^k)\\ \overline{W}^*(I_\iota^k) &\geq W^*(v_\iota).\end{aligned}$$

Since $t_\iota^{\mathrm{OPT}}(v, A) < v_\iota$ and $v_\iota^\infty < v_\iota$, it follows that

$$W^*(v_\iota^\infty) < W^*(v_\iota).$$

Thus,

$$\overline{\beta} \liminf_{k\to\infty} \overline{W}^*(I_\iota^k) \geq \overline{\beta} W^*(v_\iota) > \overline{\beta} W^*(v_\iota^\infty) = \overline{\beta} \lim_{k\to\infty} W^*(v_\iota^k) = \sigma_x^\infty \geq \limsup_{k\to\infty} \overline{W}_x(I_\iota^k).$$

Case 2b: Suppose $v_\iota^\infty = v_\iota$. Let $I_\iota^k = \big\{(\tilde{v}_\iota, 0), (v_\iota^k, 0)\big\}$.

$$\begin{aligned}\overline{W}_x(I_\iota^k) &\leq W_x(\tilde{v}_\iota)\\ \overline{W}^*(I_\iota^k) &\geq W^*(v_\iota^k).\end{aligned}$$

By lower semi-continuity, we have

$$\sigma_x^\infty = \lim_{k\to\infty} W_x(v_\iota^k) \geq W_x\left(\lim_{k\to\infty} v_\iota^k\right) = W_x(v_\iota^\infty) = W_x(v_\iota).$$

Thus,

$$\overline{\beta}\liminf_{k\to\infty} \overline{W}^*(I_\iota^k) \geq \overline{\beta}\lim_{k\to\infty} W^*(v_\iota^k) = \sigma_x^\infty \geq W_x(v_\iota) > W_x(\tilde{v}_\iota) \geq \limsup_{k\to\infty} \overline{W}_x(I_\iota^k).$$

□

Now, under the hypotheses of Theorem 2.3, if we can find (v, A) and $(\tilde{v}_\iota, v_{-\iota}, A)$ that do not satisfy Definition 2.11, then we can find $\tilde{\tilde{v}}_\iota$ arbitrarily close to v_ι such that $(\tilde{\tilde{v}}_\iota, v_{-\iota}, A)$ and $(\tilde{v}_\iota, v_{-\iota}, A)$ do not satisfy Definition 2.11 and $W_x(\cdot, v_{-\iota}, A)$ is continuous at $\tilde{\tilde{v}}_\iota$. Lemma A.4 completes the proof.

Proof of Theorem 2.4

As before, let $(v_n^\uparrow, c_n^\uparrow)$ denote an arbitrary element of $\operatorname{argmax}_{(v_n,c_n)\in I_n}\{v_n - c_n\}$, and let $(v_n^\downarrow, c_n^\downarrow)$ denote a costless investment ($c_n^\downarrow = 0$). We suppress the dependence of functions on A.

Consider the allocation $x(v^\uparrow - c^\uparrow)$. We now construct an investment profile by requiring all bidders in this allocation to invest $(v_n^\uparrow, c_n^\uparrow)$, and all other bidders to invest $(v_n^\downarrow, c_n^\downarrow)$. Formally, let $(\hat{v}, \hat{c})$ be the investment profile such that, for all n,

$$(\hat{v}_n, \hat{c}_n) = \begin{cases} (v_n^\uparrow, c_n^\uparrow) & \text{if } n \in x(v^\uparrow - c^\uparrow) \\ (v_n^\downarrow, c_n^\downarrow) & \text{otherwise.} \end{cases}$$

Recall that the threshold price for bidder n at instance (v, A) is

$$t_n^x(v, A) = \inf\{\tilde{v}_n : n \in x(\tilde{v}_n, v_{-n}, A) = 1 \text{ and } (\tilde{v}_n, v_{-n}, A) \in \Omega\}.$$

Suppressing A, let $t^x(v)$ be the profile of threshold prices at (v, A).

Lemma A.5. *Let v^k be the value profile with the first $|N| - k$ elements equal to the corresponding elements of $v^\uparrow - c^\uparrow$, and the last k elements equal to the corresponding elements of $\hat{v}$. For all $k \in \{0, 1, \ldots, |N|\}$, $x(v^k) = x(v^\uparrow - c^\uparrow)$.*

Proof. We argue by induction. By definition, $x(v^0) = x(v^\uparrow - c^\uparrow)$. Suppose $x(v^k) = x(v^\uparrow - c^\uparrow)$. Moving from v^k to v^{k+1} either raises the value of a bidder in $x(v^k)$ or lowers the value of a bidder not in $x(v^k)$. Thus, as x is monotone and non-bossy, the $x(v^{k+1}) = x(v^k) = x(v^\uparrow - c^\uparrow)$; this proves Lemma A.5. ☐

Lemma A.6. *If x is monotone and non-bossy, then for all (v, A) and $\tilde{v}_n$, if*

1. *Either: $\tilde{v}_n \geq v_n$ and $x_n(v, A) = 1$*
2. *Or: $\tilde{v}_n \leq v_n$ and $x_n(v, A) = 0$*

then for all $m \neq n$ and all $\tilde{v}_m$ such that $x_m(\tilde{v}_m, v_{-m}, A) = x_m(v, A)$:

$$x_m(v, A) = x_m(\tilde{v}_n, \tilde{v}_m, v_{-\{nm\}}, A).$$

Proof. As x is non-bossy, we have

$$x_n(\tilde{v}_m, v_{-m}, A) = x_n(v, A).$$

By the previous equation and x monotone,

$$x_n(\tilde{v}_n, \tilde{v}_m, v_{-\{nm\}}, A) = x_n(\tilde{v}_m, v_{-m}, A).$$

By the previous equation and x non-bossy,

$$x_m(\tilde{v}_n, \tilde{v}_m, v_{-\{nm\}}, A) = x_m(\tilde{v}_m, v_{-m}, A).$$

which proves Lemma A.6. ☐

Lemma A.7. *If x is monotone and non-bossy, then $t^x_n(v^\uparrow - c^\uparrow) \geq t^x_n(\hat{v})$ for $n \in x(v^\uparrow - c^\uparrow)$ and $t^x_n(v^\uparrow - c^\uparrow) \leq t^x_n(\hat{v})$ for $n \notin x(v^\uparrow - c^\uparrow)$.*

Proof. We argue by induction. Let value profile v^k be as defined as in Lemma A.5. The inductive hypothesis is: $t^x_n(v^\uparrow - c^\uparrow) \geq t^x_n(v^k)$ for $n \in x(v^\uparrow - c^\uparrow)$ and $t^x_n(v^\uparrow - c^\uparrow) \leq t^x_n(\hat{v})$ for $n \notin x(v^k)$.

The hypothesis holds by definition for $k = 0$. Suppose it holds for some k. By Lemma A.5, $x(v^k) = x(v^\uparrow - c^\uparrow)$. Moving from v^k to v^{k+1} either raises the value of a bidder in $x(v^k)$ or lowers the value of a bidder not in $x(v^k)$. By the inductive hypothesis for k and Lemma A.6, $t^x_n(v^\uparrow - c^\uparrow) \geq t^x_n(v^k) \geq t^x_n(v^{k+1})$ for $n \in x(v^\uparrow - c^\uparrow)$ and $t^x_n(v^\uparrow - c^\uparrow) \leq t^x_n(v^k) \leq t^x_n(v^{k+1})$ for $n \notin x(v^\uparrow - c^\uparrow)$. Thus the inductive hypothesis holds for $k + 1$. This completes the proof of Lemma A.7. ☐

Lemma A.8. *$(\hat{v}, \hat{c})$ is a Nash equilibrium of the investment game (I, A) facing threshold auction (x, p^x).*

Proof. By Lemma 2.3, it suffices to check that bidders choosing $(v_n^\uparrow, c_n^\uparrow)$ cannot profitably deviate to $(v_n^\downarrow, c_n^\downarrow)$ and vice versa. (Recall that $c_n^\downarrow = 0$.)

Suppose that under $(\hat{v}, \hat{c})$, n plays $(v_n^\uparrow, c_n^\uparrow)$, so $n \in x(v^\uparrow - c^\uparrow)$. Then

$$\max\{v_n^\uparrow - t_n^x(\hat{v}), 0\} - c_n^\uparrow \geq \max\{v_n^\uparrow - t_n^x(v^\uparrow - c^\uparrow), 0\} - c_n^\uparrow \geq 0.$$

where the first inequality is by Lemma A.7 and the second inequality is by $n \in x(v^\uparrow - c^\uparrow)$. This implies:

$$\begin{aligned}\max\{v_n^\uparrow - t_n^x(\hat{v}), 0\} - c_n^\uparrow &= \max\{v_n^\uparrow - c_n^\uparrow - t_n^x(\hat{v}), 0\} \\ &\geq \max\{v_n^\downarrow - c_n^\downarrow - t_n^x(\hat{v}), 0\} = \max\{v_n^\downarrow - t_n^x(\hat{v}), 0\} - c_n^\downarrow.\end{aligned}$$

The left-hand side is n's utility from playing $(v_n^\uparrow, c_n^\uparrow)$ and the right-hand side is n's utility from playing $(v_n^\downarrow, c_n^\downarrow)$. Hence, n cannot profit by deviating to $(v_n^\downarrow, c_n^\downarrow)$.

Suppose that under $(\hat{v}, \hat{c})$, n plays $(v_n^\downarrow, c_n^\downarrow)$, so $n \notin x(v^\uparrow - c^\uparrow)$. Then we have

$$\max\{v_n^\uparrow - t_n^x(\hat{v}), 0\} - c_n^\uparrow \leq \max\{v_n^\uparrow - t_n^x(v^\uparrow - c^\uparrow), 0\} - c_n^\uparrow \leq 0 \leq \max\{v_n^\downarrow - t_n^x(\hat{v}), 0\} - c_n^\downarrow,$$

where the first inequality is by Lemma A.7 and the second inequality is by $n \notin x(v^\uparrow - c^\uparrow)$.

The left-hand side is n's utility from deviating to $(v_n^\uparrow, c_n^\uparrow)$ and the right-hand side is n's utility from playing $(v_n^\downarrow, c_n^\downarrow)$. Hence, n cannot profit by deviating to $(v_n^\uparrow, c_n^\uparrow)$; this proves Lemma A.8. □

Lemma A.9. *If x is monotone, non-bossy, and a β-approximation for allocation, then*

$$W_x(\hat{v}, A) - \sum_{n \in N} \hat{c}_n \geq \beta \max_{(v,c) \in I} \left\{ W^*(v, A) - \sum_{n \in N} c_n \right\}. \tag{29}$$

Proof. Let (v^*, c^*) be a profile of investments that attains the maximum on the right-hand side of (29). By Lemma A.5, $x(\hat{v}) = x(v^\uparrow - c^\uparrow)$. Recall that, by construction,

$$(\hat{v}_n, \hat{c}_n) = \begin{cases} (v_n^\uparrow, c_n^\uparrow) & \text{if } n \in x(v^\uparrow - c^\uparrow) \\ (v_n^\downarrow, c_n^\downarrow) & \text{otherwise.} \end{cases}$$

Hence,

$$W_x(\hat{v}) - \sum_{n\in N} \hat{c}_n = w(x(\hat{v}) \mid \hat{v}) - \sum_{n\in N} \hat{c}_n = w(x(v^\uparrow - c^\uparrow) \mid \hat{v}) - \sum_{n\in N} \hat{c}_n = W_x(v^\uparrow - c^\uparrow)$$
$$\geq \beta W^*(v^\uparrow - c^\uparrow) \geq \beta W^*(v^* - c^*) \geq \beta \left(W^*(v^*) - \sum_{n\in N} c_n^* \right);$$

this proves Lemma A.9. □

Combining Lemmata A.8 and A.9 completes the proof.

Proof of Proposition 3.4

As in many of our other arguments, here we suppress the dependence of x on v_{-n} and A, as doing so will not introduce confusion.

By our choice of $\tilde{v}_n$ (in particular, by (9), with $o = x_n(\tilde{v}_n)$), we have

$$\tilde{v}_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] \geq v_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] . \tag{30}$$

We have assumed that (x, p) is strategy-proof, so—by Proposition 3.1—x is W-Mon. W-Mon implies that

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] . \tag{31}$$

Combining (31) and (the negative of) (30) yields

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] = v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] . \tag{32}$$

Now, as (x, p) is strategy proof, we know that $\tilde{v}_n$ cannot profitably imitate v_n and vice versa, which implies:

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq p_n(\tilde{v}_n) - p_n(v_n) \tag{33}$$
$$v_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] \geq p_n(v_n) - p_n(\tilde{v}_n). \tag{34}$$

Now, from (33) and (the negative of) (34) we obtain

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq p_n(\tilde{v}_n) - p_n(v_n) \geq v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] . \tag{35}$$

Combining Eq. (32) and Eq. (35), we find that

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] = p_n(\tilde{v}_n) - p_n(v_n). \tag{36}$$

Finally, by the definition of w, we have

$$\begin{aligned}
&w(x(\tilde{v}_n \mid \tilde{v}_n) - w(x(v) \mid \tilde{v}_n) \\
&= \tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] + \sum_{m \neq n} v_m \cdot [x_m(\tilde{v}_n) - x_m(v_n)] \\
&= p_n(\tilde{v}_n) - p_n(v_n) + \sum_{m \neq n} v_m \cdot [x_m(\tilde{v}_n) - x_m(v_n)] \, ,
\end{aligned}$$

where the last equality follows from (36); this completes the proof of the first claim.

Now, we observe that $p_n(\tilde{v}_n) - p_n(v_n) \neq 0$ implies, by (36), that $x_n(\tilde{v}_n) \neq x_n(v_n)$. We then have from (32) that

$$\tilde{v}_n^{x_n(\tilde{v}_n)} - \tilde{v}_n^{x_n(v_n)} = v_n^{x_n(\tilde{v}_n)} - v_n^{x_n(v_n)},$$

which holds for a measure-zero set of pairs $(v_n, \tilde{v}_n)$ when $x_n(\tilde{v}_n) \neq x_n(v_n)$. Thus, we see that $p_n(\tilde{v}_n) - p_n(v_n) = 0$ almost everywhere.

Proof of Lemma 3.3

We begin with a general lemma on submodular functions.

Lemma A.10. *Let $q : \wp(G) \to \mathbb{R}_0^+$ be a non-negative submodular function, i.e. for all $F', F'' \subseteq G$:*

$$q(F' \cup F'') + q(F' \cap F'') \leq q(F') + q(F'').$$

For all $F \subseteq G$, there exists an additive value function $\alpha^* : G \to \mathbb{R}_+$ such that $\alpha^*(F) = q(F)$ and for all F', $\alpha^*(F') \leq q(F')$.

Proof. All submodular functions are fractionally sub-additive (Lehmann et al., 2006a). Thus, there exists a family of additive value functions $(\alpha^l)_{l \in L}$ such that for all F', $q(F') = \max_l \alpha^l(F')$.

Fix some arbitrary F. Let $\alpha^* \in \text{argmax}_{\alpha^l : l \in L} \left\{ \alpha^l(F) \right\}$. $\alpha^*(F) = q(F)$, and for all F', $\alpha^*(F') \leq q(F')$. □

Now, we can develop the proof of Lemma 3.3: For any $F \subseteq G$, let

$$v_\iota^F \equiv \operatorname*{argmax}_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\}$$

By $v_\iota^F \in \mathsf{XOS}$, there exists a family of additive value functions $(\alpha^l)_{l \in L}$ such that $v_\iota^F = \max_{l \in L} \alpha^l$. Let $\tilde{\alpha}^F = \operatorname{argmax}_{\alpha^l : l \in L} \left\{\alpha^l(F)\right\}$. We now define another additive value function α^F as follows:

$$\alpha_g^F \equiv \begin{cases} \tilde{\alpha}_g^F & \text{if } g \in F \\ 0 & \text{otherwise.} \end{cases}$$

By c_ι isotone,

$$\max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\} \leq \tilde{\alpha}^F(F) - c_\iota(\tilde{\alpha}^F) \leq \alpha^F(F) - c_\iota(\alpha^F).$$

$\alpha^F \in \mathsf{XOS}$, so

$$\max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\} = \alpha^F(F) - c_\iota(\alpha^F).$$

The next step is to define, for each set of goods F, an additive value function $\overline{\alpha}^F$ that divides the cost $c_\iota(\alpha^F)$ appropriately across the various goods in F.

For any F, F', let $\alpha^{F \triangleright F'}$ be the additive value function defined by:

$$\alpha_g^{F \triangleright F'} \equiv \begin{cases} \alpha_g^F & \text{if } g \in F' \\ 0 & \text{otherwise.} \end{cases}$$

Fix some arbitrary F. Let $q^F : \wp(G) \to \mathbb{R}$ be the function defined by

$$q^F(F') \equiv \alpha^{F \triangleright F'}(F') - c_\iota(\alpha^{F \triangleright F'})$$

(for all F'). As c_ι is supermodular on additive valuations, the function $q^F(\cdot)$ is submodular. Moreover, by submodularity of q^F, it follows that for all F' we have:

$$q^F(F') + q^F(G \setminus F') \geq \underbrace{q^F(F' \cup (G \setminus F'))}_{= \alpha^F(F) - c_\iota(\alpha^F)} + \underbrace{q^F(F' \cap (G \setminus F'))}_{=0}. \tag{37}$$

Moreover, we have

$$\begin{aligned} q^F(G \setminus F') &= \alpha^{F \triangleright (G\setminus F')}(G \setminus F') - c_\iota(\alpha^{F \triangleright (G\setminus F')}) \\ &= \alpha^{F \triangleright (G\setminus F')}(F) - c_\iota(\alpha^{F \triangleright (G\setminus F')}) \\ &\leq \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\} \\ &= \alpha^F(F) - c_\iota(\alpha^F). \end{aligned}$$

Rearranging terms in (37) yields

$$q^F(F') \geq \alpha^F(F) - c_\iota(\alpha^F) - q^F(G \setminus F') \geq 0.$$

Thus, q^F is a non-negative submodular function. By Lemma A.10, we can find an additive value function $\overline{\alpha}^F$ such that $\overline{\alpha}^F(F) = q^F(F)$ and for all F', $\overline{\alpha}^F(F') \leq q^F(F')$.

We assert now that the maximum of the family of additive value functions so constructed is exactly equal to the pivotal value function $\overline{v}_\iota$, that is, for all F,

$$\max_{F' \in \wp(G)} \left\{ \overline{\alpha}^{F'}(F) \right\} = \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\} \equiv \overline{v}_\iota(F).$$

By construction, for all F,

$$\overline{\alpha}^F(F) = q^F(F) = \alpha^F(F) - c_\iota(\alpha^F) = \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\}.$$

which implies that for all F,

$$\max_{F' \in \wp(G)} \left\{ \overline{\alpha}^{F'}(F) \right\} \geq \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\}.$$

Also by construction, for all F and F',

$$\overline{\alpha}^{F'}(F) \leq q^{F'}(F) = \alpha^{F' \triangleright F}(F) - c_\iota(\alpha^{F' \triangleright F}) \leq \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\},$$

which implies that for all F,

$$\max_{F' \in \wp(G)} \left\{ \overline{\alpha}^{F'}(F) \right\} \leq \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\}.$$

Thus, for all F,

$$\max_{F' \in \wp(G)} \left\{ \overline{\alpha}^{F'}(F) \right\} = \max_{v_\iota \in \mathsf{XOS}} \{v_\iota(F) - c_\iota(v_\iota)\} \equiv \overline{v}_\iota(F);$$

we conclude that $\overline{v}_\iota \in \mathsf{XOS}$.

B The knapsack FPTAS and Proof of Proposition 2.10

Here, we first explain the FPTAS briefly, and then prove that the algorithm is not XBONE. The key to the FPTAS is that when values are restricted to a discrete grid $\{0, \delta, 2\delta, \ldots\}$, we can solve for the exact optimum in

$$O\left(|N|S\frac{\sum_n v_n}{\delta}\right)$$

time using dynamic programming. The FPTAS rounds down the original values and runs a dynamic program to find the optimum (Williamson and Shmoys, 2011).

Consider an instance of the knapsack problem with $|N|$ bidders and some given $\epsilon \in (0,1)$. Let $\overline{v} = \max_n\{v_n\}$ be the highest bidder value. The FPTAS with parameter ϵ proceeds as as follows:

1. Let $K = \frac{\epsilon\overline{v}}{|N|}$.
2. For each bidder n, let $v'_n = \left\lfloor \frac{v_n}{K} \right\rfloor$.
3. With these rounded values v'_n, using dynamic programming, find the optimal packing $\hat{N}$.
4. Output $\hat{N}$.

For any specified $\epsilon > 0$, the FPTAS with parameter ϵ is a $(1-\epsilon)$ approximation for the knapsack problem, and runs in time $O(\frac{|N|^3}{\epsilon})$.

We now show that the FPTAS is a 0-approximation for investment. Consider an instance with knapsack capacity $2|N|-3$ and $|N|$ bidders, with values and sizes as follows:

$$\{(100, 2|N|-4), (99,2), (99,2), \ldots, (99,2), (1,1)\}.$$

Take any $\epsilon > 0$ and let us run the FPTAS with parameter ϵ. For large enough $|N|$, the FPTAS packs all bidders but the first one, and the total value will be $99(|N|-2)+1$.

Now suppose the last bidder can raise his value to $\frac{100|N|}{\epsilon}$ at a cost of $\frac{100|N|}{\epsilon} - 1$. This is (weakly) profitable. After the last bidder's investment, the FPTAS will round values to:

$$\left\{(1, 2|N|-4), (0,2), (0,2), \ldots, \left(\left\lfloor \frac{|N|}{\epsilon} \right\rfloor, 1\right)\right\}.$$

Therefore, only items 1 and $|N|$ will be packed. The total welfare net of investment costs is $100 + \frac{100|N|}{\epsilon} - (\frac{100|N|}{\epsilon} - 1) = 101$, whereas the first-best welfare including investment is $99(|N| - 2) + 1$. Hence, we can choose $|N|$ large enough that welfare under the FPTAS at this investment instance is an arbitrarily small fraction of the first-best, which proves Proposition 2.10.[18]

C Steiner tree

The Steiner tree problem is a classic problem with applications in package delivery and optimal routing. The input to the problem is a connected, undirected graph $G = (V, E)$, where each edge has a weight, and a set $V^* \subseteq V$ of nodes selected as *terminals*. The goal is to find a weight-minimizing connected subgraph of G which contains all the terminals. It is well-known that this problem is NP-Complete (Karp, 1972).

The Steiner tree problem has a classic 2-approximation that builds on the minimum spanning tree (MST) of the graph G (Vazirani, 2013, pp. 27–28). The MST-based 2-approximation algorithm for the Steiner tree problem, MST-Steiner, works in three steps:

1. Construct a weighted graph G' from the original graph G in the following way: The set of nodes of G' are the terminal nodes of G. For any two nodes t_1 and t_2 in G', let the weight of the edge between them be equal to the total weight of the shortest path between the two in G.

2. Find a minimum spanning tree in G'.

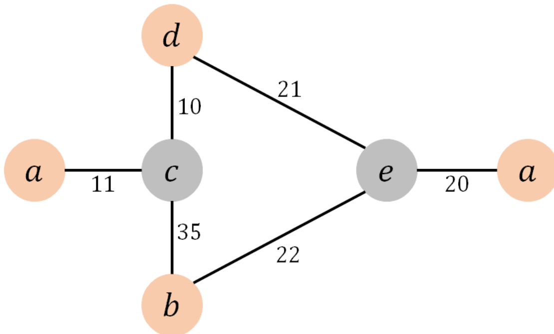
3. Recover the shortest paths in the original graph G that represent the edges in the minimum spanning tree just constructed; then remove edges if necessary to ensure that the output is a tree.

We have stated XBONE for value-maximization problems, but it naturally generalizes to cost-minimization problems such as this one, which can be interpreted as procurement auctions. In particular, each edge in G is a bidder, each bidder's value is the negative of the weight, and the packed bidders are the edges that are selected by the algorithm.

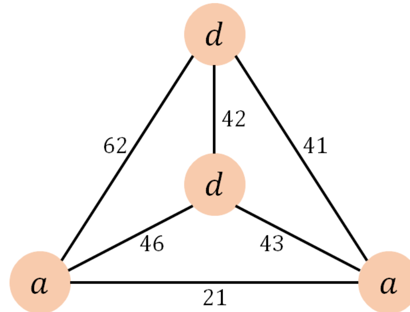
Proposition C.1. MST-Steiner *is not XBONE.*

Proof. We prove Proposition C.1 by example. The graph in Figure 1a represents the initial graph G. Nodes $\{a, b, d, f\}$ are the terminal nodes. The graph in Figure 1b is G', which we

[18] In particular, the last bidder was packed in both cases, but raising his value reduced the total welfare of the other bidders from $99(|N| - 2)$ to 100, which is a bossy negative externality.

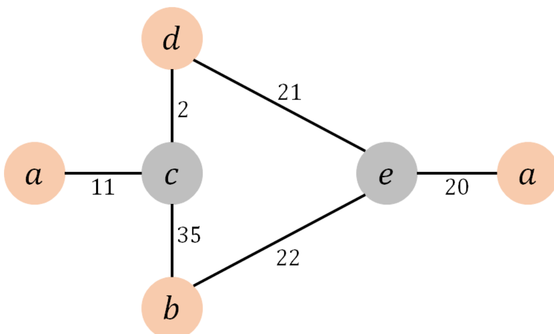


(a) Graph G

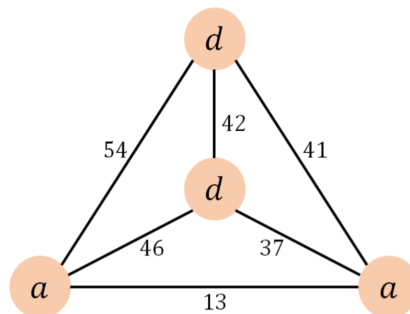


(b) Graph G' constructed from G

Figure 1: Steiner tree instance before investment



(a) Post-investment graph G



(b) Graph G' constructed from post-investment G

Figure 2: Steiner tree instance after investment

have constructed from G. The MST of G' includes edges ad, df, and bf. These correspond to ac, cd, de, ef, and be, which together comprise a Steiner tree in the original graph G. The total weight of this Steiner tree is 76.

Now suppose we introduce an investment that reduces the weight of cd from 10 to 2 as pictured in Figure 2a (this is equivalent to *increasing* the value of that edge in the corresponding maximization problem). Applying the same algorithm (as illustrated in Figure 2b) leads to choosing ac, cd, bc, de, and ef, with a total weight of 91. In particular, we reduced the weight of a packed edge cd, and the total weight incurred from the edges other than cd has risen by 13, which is a bossy negative externality. Thus, MST-Steiner is not XBONE. □