Investment Incentives in Near-Optimal Mechanisms[*]

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

March 9, 2020

Abstract

In a Vickrey auction, if one bidder has an option to invest to increase his value, the combined mechanism including investments is still fully optimal. In contrast, for any $\beta < 1$, we find that there exist monotone allocation rules that guarantee a fraction β of the allocative optimum in the worst case but such that the associated mechanism with investments by one bidder can lead to arbitrarily small fractions of the full optimum being achieved. We show that if a monotone allocation rule satisfies a new property called XBONE and guarantees a fraction β of the allocative optimum, then in the equilibrium of the threshold auction game with investments, at least a fraction β of the full optimum is achieved. We also establish generalizations and a characterization, and show that some well-known approximation algorithms satisfy the XBONE property.

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

JEL classification: D44, D47, D82

---

[*] We thank Matthew Gentzkow, Paul Goldsmith-Pinkham, Andy Haupt, Emir Kamenica, Zi Yang Kang, Ellen Muir, Noam Nisan, Amin Saberi, and Mitchell Watt for helpful comments. 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 of 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. 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), to coordinate electricity generation and transmission (Lavaei and Low, 2011; Bienstock and Verma, 2019), to assign radio spectrum broadcast rights subject to legally-mandated interference constraints (Leyton-Brown et al., 2017), or to find a value-maximizing allocation in a combinatorial auction (Sandholm, 2002; Lehmann et al., 2006b).

Computational difficulty, however, does not obviate the need to solve allocation problems in practice. Hence, recent research in economics and computation has focused on identifying fast algorithms to find approximate solutions to hard problems and associated payment schemes that provide incentives for participants to report the input values truthfully. In the language of textbook economics, this 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 or improve existing ones or disinvest to cash in less valuable assets. In resource allocation problems, investments can affect both what is feasible (such as when an airline that chooses to use larger planes is more difficult to schedule on a runway) and the values of the items being allocated (because a larger plane carries more passengers).

Mechanisms based on fast algorithms can sometimes misalign participant's investment incentives with the objective of maximizing total welfare. To illustrate one such case, consider the classic knapsack problem in which the aim is to maximize the sum of the values of indivisible items placed into a knapsack of fixed capacity. Each item has a size and a value. Each also has a different owner and the owners bid in a truthful auction to buy space in the knapsack. The auctioneer can see the sizes of the items but not their values, so she uses the owners' bids instead of values to determine which items to pack. Since the knapsack problem is NP-hard, the auctioneer may apply a fast algorithm—in this example, Dantzig's GREEDY algorithm—to the bids and sizes to determine the winner. This algorithm sorts items in decreasing order of value-per-unit-size and packs items in that order, stopping when it encounters an item that does not fit. The associated truthful auction is a *threshold auction* in which each winning bidder pays an amount equal to its *threshold price*, which is the lowest value the bidder could report, given the bids of the other bidders, to win a space in the knapsack.[1]

---

[1] The truthfulness of this threshold auction and the ease of computing threshold prices for it were established by Lehmann et al. (2002).

Suppose that the knapsack has capacity 20 and there are three bidders, whose items have values 11, 11, and 12 and sizes 10, 10, and 11. Since $\frac{11}{10} > \frac{12}{11}$, the Greedy algorithm packs the first two items for a total value of 22, which is also the optimum for this problem. 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 my item will be packed. In fact, any value over 12.1 would result in my item 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.

In this paper, we study a long-run formulation in which the resource allocation mechanism consists of two stages, beginning with an investment stage in which one bidder can make a costly investment, while knowing the other bidders' values. The first-stage investment determines the investing bidder's value, which is then used by the second-stage algorithm to compute the final allocation. We limit attention to algorithms that are weakly monotone—precisely the algorithms that can be truthfully implemented by some auction mechanism (Nisan, 2000; Saks and Yu, 2005). Our central question is this: if an approximate algorithm delivers at least a fraction $\beta \in (0, 1)$ of the optimal welfare in an allocation problem, when does the same worst-case guarantee β apply—for all investment technologies—to the two-stage mechanism?

As a benchmark, consider participants' investment incentives when the algorithm is based on exact optimization. In that case, the truthful mechanism is the Vickrey-Clarke-Groves (VCG) auction, which provides each bidder with an incentive to report his values truthfully (Green and Laffont, 1977; Holmström, 1979). In a VCG auction, each participant's equilibrium payoff is equal to his contribution to total welfare. Consequently, for any investment cost function mapping bidder values into costs, the bidder's payoff-maximizing investment decision is the same as the one that maximizes total welfare.

How much does the benchmark VCG analysis extend to other truthful mechanisms? In any truthful mechanism, the price a bidder must pay to acquire resources depends only on other participants' values; these prices play a crucial role in guiding investment decisions. If a bidder's price is too low, he may prefer to invest and become a winner even though that reduces total social welfare. Similarly, if a bidder's price is too high, he may fail to make an investment that would both make him a winner and increase total social welfare.

These are the ordinary externalities commonly found in economic models in which inaccurate prices lead to socially suboptimal decisions. However, in a mechanism associated with an approximate algorithm, there can also be a different kind of externality that arises if the algorithm is "bossy," meaning that a bidder can change his value in a way that alters another participant's outcome without affecting his own.[2]

We show by example that there are bossy algorithms for which allocative performance is arbitrarily close-to-optimal, but investment performance can be arbitrarily bad. More precisely, for any $\beta < 1$, there is an algorithm for the knapsack problem that guarantees at least a fraction β of the maximum value, but such that if one bidder can make an investment, then for any $\varepsilon > 0$, there are instances with performance less than ε of the social optimum. The algorithms that we identify, however, are bossy in a particular way.

We prove that if an algorithm excludes bossy negative externalities (a property we call "XBONE"), then that algorithm's performance guarantee for the "long-run" allocation problem with investment is just the same as its guarantee for the "short-run" problem without investment.

To describe XBONE for simple packing problems, suppose that we are given a value profile and feasibility constraints. An algorithm then outputs some set of packed bidders. If we raise the value of a packed bidder or lower the value of an unpacked bidder and then run the algorithm at the new value profile, the algorithm outputs a new packing. The algorithm is XBONE if the welfare of the new packing, assessed at the new values, is at least as high as the welfare of the old packing, assessed at the new values.

In practical applications, the instances that most often arise may have a known special structure that can improve the short-run performance guarantee. We formulate our theory to accommodate and take advantage of any such special structures. Given an allocation problem, we define well-behaved subsets of instances to be "sub-problems." We show that if an algorithm is XBONE, then on every sub-problem, its long-run guarantee is equal to its short-run guarantee.

For example, in the knapsack problem, the Greedy algorithm generally has only a 0 worst-case guarantee, but if we define a sub-problem with a knapsack capacity of 20 and item sizes of 10, 10 and 11, the short-run performance for any item values is always at least $\frac{11}{20}$ of the optimum. The Greedy algorithm is XBONE, so for any investment technology, the long-run sub-problem satisfies the same $\frac{11}{20}$ guarantee.

We also obtain a full characterization: an algorithm has equal short-run and long-run guarantees on every sub-problem *if and only if* it satisfies a slightly weakened version of

---

[2] Satterthwaite and Sonnenschein (1981) originally introduced the concept of non-bossiness.

XBONE.[3] We also identify some interesting XBONE algorithms—including a canonical Fully Poly-Time Approximation Scheme (FPTAS) for the knapsack problem, which for any $\epsilon > 0$, guarantees a value within a $(1 - \epsilon)$ factor of the maximum.

Our formal analysis of how externalities affect long-run performance guarantees treats positive investments, which increase value, differently from disinvestments, which reduce value. Given a profitable positive investment, we decompose its effect into two parts: first the investment may bring the value up to the bidder's threshold for the allocation and then it may increase the value strictly above that threshold. For any monotone algorithm and associated truthful auction, a winning bidder pays its threshold price, which is the lowest bid it could make and still be winning. Consequently, if a profitable investment raises the bidder's value just to his threshold, the investment must cost approximately 0. In that case, the long-run performance is equal to that of the related short-run problem in which the bidder's value is equal to his threshold value, so this long-run performance cannot be worse than the worst-case short-run performance.

Next, consider the further effect of increasing a bidder's value starting at or above the threshold. If the algorithm is non-bossy, this further change has no effect on the allocation and so cannot degrade the performance ratio. If the algorithm is bossy, then performance can be degraded only if the effect on the total value assigned to others—the externality—is negative, but XBONE excludes that possibility.

For the case of disinvestment, any change can also be decomposed into two steps. The value may first be reduced to the just below threshold value and then to a strictly lower value. A profitable disinvestment that reduces the bidder's value to just below its threshold cannot result in worse long-run performance than the short-run performance associated with the lower value. Any further reduction in the value does not change the bidder's allocation, so it cannot degrade long-run performance unless it causes a bossy negative externality, which XBONE excludes. Hence, with XBONE, no disinvestment can degrade the performance guarantee.

In summary, the message is simple: all negative externalities in an algorithm can reduce welfare given some investment cost function, but only bossy negative externalities can lead the algorithm's long-run performance to fall below its short-run guarantee. Because XBONE excludes such externalities, it leads to good long-run performance, even on sub-problems. We show that a similar analysis applies also to bidders with multi-dimensional values.

In addition to the general findings described above, we report two others. One concerns the equilibrium in investments when several bidders may invest. Even in a Vickrey auction, a Nash equilibrium can have inefficient investments due to a coordination failure among

---

[3] We de-emphasize the weakened version of XBONE because verifying it can itself be an NP-hard problem.

bidders, but for the Vickrey auction there is also a Nash equilibrium in which the investments preserve the efficiency of the mechanism. We show that if a monotone algorithm guarantees a β fraction of the optimum for all instances of the short-run problem and is (fully) non-bossy—a condition more restrictive than XBONE—then there exists an equilibrium of the long-run problem that has the same β guarantee. The second concerns combinatorial auctions in which the set of values is 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 that paper, like ours, deals with the connection between (near-)efficiency at the allocation stage and (near-)efficiency at the investment stage, it uses additive error bounds, rather than the multiplicative worst-case bounds that are standard for the analysis of computationally hard problems.

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) explain 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, 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 is an NP-hard problem, VCG-based mechanisms with nearly optimal allocations determined by heuristics 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[4].

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) study the problem of constructing strategy-proof mechanisms from greedy algorithms; similarly, Bikhchandani et al. (2011) and Milgrom and Segal (2020) propose clock auction implementations of greedy allocation algorithms.

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 finitely many outcomes.

We consider three nested perspectives on the same situation. First, the allocation problem, in which our objective is total welfare and the values of the bidders are known to us. Second, the reporting problem, in which values are private information and we must elicit them via an incentive-compatible payment rule prior to allocation. Third, the investment

---

[4] For a more comprehensive review of results on approximation in mechanism design, see Hartline (2016).

problem, in which a bidder can make costly investments to change his value before reporting.

2.1.1 The allocation problem

Next, we define an **allocation problem** to be a collection of instances. Intuitively, an **instance** consists of a profile of bidder values and feasibility constraints. A **value profile** v specifies, for each bidder, that bidder's value for being packed. 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. Here are the same definitions stated 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 $n \in N$ an interval $V_n^A \subseteq \mathbb{R}$ such that $\{v : (v, A) \in \Omega\} = \prod_n V_n^A$.

An **algorithm** x selects, for each instance $(v, A) \in \Omega$, a feasible allocation, that is, $x(v, A) \in A$.[5] We will occasionally abuse notation and write $x_n(v, A)$ to denote an indicator function, equal to 1 if $n \in x(v, A)$ 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

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

---

[5] 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 items' 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,

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

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 can guarantee acceptable performance on some problems. The standard measure of algorithm performance is the worst-case guarantee, which is characterized as follows.

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

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

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 the values of their assets 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 now consider the corresponding **reporting problem**, which differs from the allocation problem because the algorithm can no longer directly learn 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

$$v_n \in \underset{\hat{v}_n \in I}{\operatorname{argmax}} \left\{ v_n x_n(\hat{v}_n, v_{-n}, A) - p_n(\hat{v}_n, v_{-n}, A) \right\};$$

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

$$p_n^x(v, A) = x_n(v, A) t_n^x(v, A);$$

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 and we have*

$$x_n(v, A) = 0 \implies p_n(v, A) = 0,$$

then x is monotone and (x, p) is a threshold auction.

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

Given some allocation problem Ω, we now define the corresponding **investment problem**. The investment problem we consider can be interpreted as a long-run analysis, which complements the short-run analysis of reporting problems. In an investment problem, one bidder has an opportunity to change his value at a cost, and he does so with full information about the mechanism and about other bidders' values.

Formally, before the auction commences, one bidder $\iota \in N$ can invest to change his value. 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. **Finite.** $|I_\iota| < \infty$.
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.

We begin by studying investments in the case that the auction is a VCG auction. For VCG auctions, the total profits of the auctioneer and all the participants besides ι is an amount $f(v_{-\iota})$ that does not depend on ι's report. Hence, ι's net profit is the total social welfare minus $f(v_{-\iota})$. 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.

Proposition 2.2. *In the investment problem for a VCG auction, ι's payoff-maximizing investment choice also maximizes social welfare.*

Next, suppose we have some other monotone algorithm x that guarantees a β-approximation for allocation. Under what conditions does the 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

$$\mathrm{BR}(x, I_\iota, v_{-\iota}, A) \equiv \operatorname*{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{u_\iota(x, v_\iota, c_\iota, v_{-\iota}, A)\}.$$

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

$$\overline{W}_x(I_\iota, v_{-\iota}, A) \equiv \min_{(v_\iota, c_\iota) \in \mathrm{BR}(x, I_\iota, v_{-\iota}, A)} \{W_x(v_\iota, v_{-\iota}, A) - c_\iota\}; \tag{1}$$

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 2.6. 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 2.3. *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. $\square$

If x is a β-approximation for allocation, does that imply anything about its investment guarantee? Our next result shows that without further structure, the investment guarantee in our setting can be arbitrarily bad—even if the allocation guarantee is strong.

Proposition 2.4. *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.*

Note that the setting of Proposition 2.4 includes the knapsack problem, which we define in Section 2.2.3.

Proof of Proposition 2.4. We construct the desired algorithm x^β:

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

By inspection, x^β is monotone and a β-approximation for allocation. Moreover, since bidder 1 is always packed for instances in Ψ, 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 $(v, A) \in \Psi$, 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 binary outcomes

For any given investment technology, a bidder may have multiple best choices and in (1) we have specified the welfare-minimizing one as the basis for our calculations. Our next result allows us to sidestep this multiplicity to simplify the analysis below. It states that an algorithm's investment approximation ratio over all instances is equal to its approximation ratio over instances with singleton best-responses.

Lemma 2.1. *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.*

Proof. 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. $\square$

We now characterize the investor's best response facing any threshold auction. In particular, 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 2.2. *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*.*

In Section 3.2, we prove a more general version of Lemma 2.2 (specifically, Lemma 3.1); here, we present an elementary argument for the binary outcome case.

Proof of Lemma 2.2. 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 $u_\iota(v_\iota, c_\iota) = \max\{v_\iota - \tau_\iota, 0\} - c_\iota \le 0 \le \max\{v_\iota^\downarrow - \tau_\iota, 0\} = u_\iota(v_\iota^\downarrow, c_\iota^\downarrow)$. ☐

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

$$w(a \mid v) \equiv \sum_{n \in a} v_n = \mathbf{1}_a \cdot v.$$

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

We now state the key definition for our main theorem.

Definition 2.7. 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

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

then we have

$$w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) - w(x(v, A) \mid \tilde{v}_n, v_{-n}) \ge 0. \tag{2}$$

For a monotone algorithm, the left-hand side of (2) is equal to

$$\sum_{m \neq n} v_m \left[x_m(\tilde{v}_n, v_{-n}, A) - x_m(v, A)\right], \tag{3}$$

which is the **bossy externality** associated with a change in bidder n's value; XBONE is the requirement that any such externality must be non-negative.

XBONE algorithms can entail other kinds of externalities, as Section 2.2.3 will illustrate, 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 is a β-approximation for allocation, then x is a β-approximation for investment.*

2.2.1 Proof of Theorem 2.1

By Lemma 2.1, 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 $\operatorname{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{v_\iota - c_\iota\}$, and let $(v_\iota^\downarrow, c_\iota^\downarrow)$ denote a costless investment ($c_\iota^\downarrow = 0$).

By Lemma 2.2, 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 $\operatorname{argmax}_{(v_\iota, c_\iota) \in I_\iota} \{W^*(v_\iota) - c_\iota\}$, so that

$$\overline{W}^*(I_\iota) = W^*(v_\iota^*) - c_\iota^* = W^*(v_\iota^* - c_\iota^*) \leq W^*(v_\iota^\uparrow - c_\iota^\uparrow). \tag{4}$$

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.

2.2.2 Non-bossiness and XBONE

XBONE is naturally weaker than non-bossiness.

Definition 2.8. 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.

Proposition 2.5. *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 2.7. 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. ☐

XBONE requires that for 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.

Proposition 2.6. *Let X be a collection of XBONE algorithms. If y is an algorithm that at each instance $(v, A) \in \Omega$ outputs a surplus-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 2.7. 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. ☐

2.2.3 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 items, where an item n has value v_n and size s_n. The knapsack has capacity S. Without loss of generality, suppose no item's size is more than S. The set of feasible allocations is any subset of items $K \subseteq N$ 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 items by the ratio of their values to their sizes so that

$$\frac{v_1}{s_1} \geq \frac{v_2}{s_2} \cdots \geq \frac{v_{|N|}}{s_{|N|}} \tag{5}$$

Add items 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 item that would violate the size constraint, stop.

Although Dantzig's Greedy algorithm performs well on some instances, including ones for which all items 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 items. 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 item 1 and stops, whereas the optimal algorithm picks item 2. Thus, Greedy's performance is no better than ϵ of the optimum.

There is a simple 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 item; output whichever has higher welfare.*

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

Proposition 2.7. MGreedy *is a $\frac{1}{2}$-approximation for the Knapsack problem.*

Proof. For any instance ω, order the items by value/size as in (5). If Greedy packs all items, then trivially $W^*(\omega) = W_{\text{MGreedy}}(\omega)$. Otherwise, let k be the lowest index of an item not packed by Greedy and let K be the index of an item 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 items. $v_1 = 2$, $v_2 = 1$, $v_3 = 8$. $s_1 = s_2 = 1$, $s_3 = 9$. At this instance, MGreedy packs just item 3. If we raise v_3 to 10, then MGreedy instead packs item 1 and item 3. Thus, MGreedy is bossy. This is a bossy *positive* externality; raising the value of a packed item by 2 has increased welfare by 4.

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 StandardFPTAS rounds down the values, and uses dynamic programming to output an optimal allocation for the rounded instance. For details, we refer interested readers to Williamson and Shmoys (2011, p. 65-68) or Vazirani (2013, p. 68-70).

Proposition 2.8. *For the knapsack problem, the* Greedy *algorithm, the* MGreedy *algorithm, and the* StandardFPTAS *all are XBONE.*

Proof. The Greedy algorithm is a monotone and non-bossy algorithm, and thus it is XBONE by Proposition 2.5.

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 2.5, as well. Thus, by Proposition 2.6, the MGreedy algorithm is XBONE.

The StandardFPTAS is monotone and non-bossy on the rounded instance. Moreover, changing one bidder's value does not affect the algorithm's output unless it changes the rounded instance. Therefore, the StandardFPTAS is monotone and non-bossy, and by Proposition 2.5 it 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 those bossy externalities are not negative, so they too cannot undermine the worst-case performance guarantee.

2.2.4 A weaker sufficient condition

Definition 2.7 is sufficient for approximation guarantees to persist under investment; it is not necessary, however—the following weaker condition will do.

Definition 2.9. 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 $\tilde{v}_n > v_n$ and $n \in x(v, A)$,
2. or $\tilde{v}_n < v_n$ and $n \notin x(v, A)$ and $n \in a$ for all $a \in \text{argmax}_{a' \in A} w(a' \mid v)$,

then we have

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

Intuitively, x is weakly XBONE if two conditions hold. First, if n is selected by x at v, then increasing his value increases the welfare achieved by x by at least an equal amount. This is the same as the first XBONE condition. Second, if n is not selected by x at v but is part of every optimal solution at v, then decreasing his value does not reduce the welfare achieved by x. This weakens the second XBONE condition, requiring it to hold only if the argmax condition is satisfied.

Clause 2 of Definition 2.9 equivalently requires that $\tilde{v}_n \leq v_n$ if $n \notin x(v, A)$ and for all $\epsilon > 0$ we have $W^*(v_n - \epsilon, v_{-n}, A) < W^*(v_n, v_{-n}, A)$.

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 optimal allocations. Thus, Clause 2 of Definition 2.9 may be intractable to verify.

Definition 2.10. 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'; thus, 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'.

We find that 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 $(v_{-\iota}, A)$, 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'.

2.3 Allowing multiple investors

We now show how to extend our framework and approach to a setting in which multiple bidders are able to invest.

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

When multiple investors simultaneously choose investments, even Vickrey auctions can suffer from inefficient investments due to a coordination problem, as the following example illustrates.

Example 2.3. Consider the knapsack problem. 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 the bidders invests, it will not be packed. In the Vickrey 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.

We do not know whether XBONE is enough, in general, to ensure that an efficient Nash equilibrium exists. However, if the algorithm is monotone and non-bossy and guarantees a fraction β in the short-run problem, 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*

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

3 Investment with multiple outcomes

The problems we studied in Section 2 were generalizations of the knapsack problem: each bidder is either packed or unpacked. We now generalize Theorem 2.1 to a setting in which each bidder can have more than two potential outcomes.

3.1 Allocation problems with multiple outcomes

Now, O denotes a finite set of **outcomes**. Each bidder's **value** $v_n \in \mathbb{R}^O$ 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 set of A of feasible allocations, such that for all $a \in A$, v's dimensions agree with a's dimensions. We assume that the allocation in which every bidder is unpacked is feasible.

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, $\{v : (v, A) \in \Omega\} = \prod_n V_n^A$.

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. 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, as in our setting each V_n^A is convex, W-Mon is also a sufficient condition.[6]

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.*

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. **Finite.** $|I_\iota| < \infty$.
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 $u_\iota(\cdot)$, $\text{BR}(\cdot)$, $\overline{W}_x(\cdot)$, and $\overline{W}^*(\cdot)$ as before. Note that for convex V_ι^A, the particular choice of payment rule does not matter—V_ι^A is path-connected, so by the envelope theorem, ι's best-responses are the same for all strategy-proof payment rules (Milgrom and Segal, 2002).

3.4 Results for multiple outcomes

We now generalize our XBONE condition and Theorem 2.1 to allow for more than two outcomes.

Definition 3.2. Algorithm x is **XBONE** if for any two instances (v, A) and $(\tilde{v}_n, v_{-n}, A)$, if for all outcomes o:

$$\tilde{v}_n^{x_n(v,A)} - \tilde{v}_n^o \geq v_n^{x_n(v,A)} - v_n^o, \tag{6}$$

then

$$w(x(\tilde{v}_n, v_{-n}, A) \mid \tilde{v}_n, v_{-n}) \geq w(x(v, A) \mid \tilde{v}_n, v_{-n}). \tag{7}$$

XBONE is a property of allocation algorithms—it does not depend on the payment rule. However, when an algorithm x is paired with an incentive-compatible payment rule p, then the requirement that x is XBONE can be restated in a way that illuminates the role of externalities.

---

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

Proposition 3.3. *For strategy-proof (x, p), Eq. (7) is equivalent to the requirement that*

$$\underbrace{p_n(\tilde{v}_n, v_{-n}, A) - p_n(v, A)}_{\textit{change in } n\textit{'s payment}} + \underbrace{\sum_{m \neq n} v_m \cdot [x_m(\tilde{v}_n, v_{-n}, A) - x_m(v, A)]}_{\textit{bossy externality}} \geq 0, \tag{8}$$

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 the requirements of Definition 3.2, then we have $p_n(\tilde{v}_n, v_{-n}, A) - p_n(v, A) = 0$.

Equation (8) decomposes the effect of moving from v_n to $\tilde{v}_n$ into a change in n's payment and an externality imposed on other bidders. Thus, Proposition 3.3 implies that if (x, p) is strategy-proof and x is XBONE, then for certain changes in n's value, any negative externality imposed on other bidders is offset by a rise in n's payment. In any strategy-proof mechanism, the first term in (8) is equal to 0 almost everywhere, so that if x is XBONE then its bossy externality must be weakly positive almost everywhere—closely corresponding to (3). Moreover, the first term in (8) is uniformly equal to 0 in the binary-outcome case, because raising the value of a packed bidder (or lowering the value of an unpacked bidder) does not affect that bidder's payment.

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 negative bossy externalities, then the long-run problem inherits the worst-case guarantee from the short-run problem.

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 $o \in O$, the corresponding component of the pivotal vector is

$$\overline{v}_\iota^o = \max_{(v_\iota, c_\iota) \in I_\iota} \{v_\iota^o - c_\iota\}. \tag{9}$$

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 this procedure.

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

$$\max_{(v_\iota, c_\iota)\in I_\iota} \left\{ v_\iota \cdot x_\iota(v_\iota) - p_\iota^x(v_\iota) - c_\iota \right\}. \tag{10}$$

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 (10) to yield

$$\max_{(v_\iota, c_\iota)\in I_\iota} \max_{\tilde{v}_\iota} \left\{ v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota) - c_\iota \right\}. \tag{11}$$

Changing the order of maximization in (11) 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

$$\max_{(v_\iota, c_\iota)\in I_\iota} \left\{ v_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota) - c_\iota \right\} = \overline{v}_\iota \cdot x_\iota(\tilde{v}_\iota) - p_\iota^x(\tilde{v}_\iota), \tag{12}$$

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 (12). This reduces ι's problem to the maximization

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

Dropping the term in (13) 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 2.1, 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

$$W_x(\hat{v}_\iota) = w(x(\hat{v}_\iota) \mid \hat{v}_\iota) \geq w(x(\overline{v}_\iota) \mid \hat{v}_\iota). \tag{14}$$

Now, by our construction of the pivotal vector $\overline{v}_\iota$ in (9) 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

$$w(x(\overline{v}_\iota) \mid \hat{v}_\iota) - \hat{c}_\iota = w(x(\overline{v}_\iota) \mid \overline{v}_\iota) = W_x(\overline{v}_\iota). \tag{15}$$

Subtracting $\hat{c}_\iota$ from (14) and applying (15), we find that

$$W_x(\hat{v}_\iota) - \hat{c}_\iota \geq W_x(\overline{v}_\iota). \tag{16}$$

Combining the preceding steps, we see that

$$\overline{W}_x(I_\iota) = \overbrace{W_x(\hat{v}_\iota) - \hat{c}_\iota \geq \underbrace{W_x(\overline{v}_\iota)}^{(16)} \geq \overbrace{\beta W^*(\overline{v}_\iota)}^{\text{Lemma 3.2}}}_{\beta\text{-approx for allocation}} = \beta\overline{W}^*(I_\iota),$$

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 many standard classes of value functions do not have a product structure on 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. Consequently, Theorem 3.1 has limited applicability for combinatorial auctions. 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 allocation 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. $V_\iota = \mathsf{XOS}$.

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 $F \subseteq G$.

We first derive an analog of Lemma 3.1 for the combinatorial auction setting.

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 2.1, 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

$$W_x(\hat{v}_\iota) - \hat{c}_\iota \geq W_x(\overline{v}_\iota). \tag{17}$$

We then have

$$\overline{W}_x(c_\iota) = \overbrace{W_x(\hat{v}_\iota) - c_\iota(\hat{v}_\iota) \geq W_x(\overline{v}_\iota)}^{(17)} \geq \overbrace{\beta W^*(\overline{v}_\iota)}^{\text{Lemma 3.2}} = \beta \overline{W}^*(c_\iota),$$
$$\underbrace{W_x(\overline{v}_\iota) \geq \beta W^*(\overline{v}_\iota)}_{\beta\text{-approx for allocation}}$$

which completes the proof. $\square$

4 Discussion

Standard economic models typically assume exact optimization. But in practice many allocation problems can at best be optimized approximately—and hence, heuristic algorithms are widespread in real-world markets. What are the economic consequences of replacing optimal allocation rules with approximations?

The preceding analysis suggests that the consequences can be subtle. Nearly-optimal allocation rules can lead to arbitrarily bad long-run investment incentives, even under truthful implementation. The key problem is that approximation algorithms introduce a new type of externality, under which a bidder's investment may bossily change other bidder's outcomes by causing the algorithm to select a different approximate optimum. Ruling out bossy externalities is sufficient for short-run approximation guarantees to persist in the long-run under investment. Notably, the potential for bossy externalities can be detected from a mechanism's allocation rule alone, without direct reference to the pricing rule.

Here are some open questions raised by our work:

- Our analysis so far has focused on investment under full information. How, if at all, does the analysis extend to incomplete information? What properties must an allocation algorithm have to retain its performance when a bidder invests without knowing the values of the other bidders? Can the relevant information be elicited in advance through an appropriate choice of mechanism?

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

- How hard is it to generate good investment incentives? Does requiring an allocation algorithm to be XBONE raise new computational hurdles? In particular, 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 has many consequences beyond investment. For example, approximation can affect how participants understand mechanisms in practice, can raise new opportunities for coordination or collusion, and can influence post-auction resale markets. We hope that the methods we have introduced here can be adapted to examine the economic implications of approximation algorithms in these and other contexts.

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.

Dantzig, G. B. (1957): “Discrete-variable extremum problems,” 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.

Green, J. and J.-J. Laffont (1977): “Characterization of satisfactory mechanisms for the revelation of preferences for public goods,” *Econometrica*, 427–438.

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.

Holmström, B. (1979): "Groves' scheme on restricted domains," *Econometrica*, 1137–1144.

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, 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.

Satterthwaite, M. A. and H. Sonnenschein (1981): “Strategy-proof allocation mechanisms at differentiable points,” *Review of Economic Studies*, 48, 587–597.

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 Theorem 2.2

Proof. The proof of Theorem 2.1 established that

$$\overline{W}_x(I_\iota, v_{-\iota}, A) \geq \beta \overline{W}^*(I_\iota, v_{-\iota}, A) \tag{18}$$

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 (18) 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 $\iota \in a$ for all $a \in \operatorname{argmax}_{a \in A} \left\{ w(a \mid v_\iota^\uparrow - c_\iota^\uparrow, v_{-\iota}) \right\}$;

2b. ι chooses $(v_\iota^\downarrow, c_\iota^\downarrow)$, $\iota \notin x(v_\iota^\uparrow - c_\iota^\uparrow)$, and there exists $a \in \operatorname{argmax}_{a \in A} \left\{ w(a \mid v_\iota^\uparrow - c_\iota^\uparrow, v_{-\iota}) \right\}$ such that $\iota \notin a$.

When x is weakly XBONE, the same arguments as in the proof of Theorem 2.1 work for Case 1 and Case 2a. Meanwhile, we observe that in Case 2b:

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

where the last inequality follows by (4). □

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

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

Lemma A.1. *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 2.9, 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 2.9. 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 $\{\check{v}_\iota^k, W_x(\check{v}_\iota^k), W^*(\check{v}_\iota^k)\}_{k=1}^\infty$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 all three converge, where we denote $v_\iota^\infty = \lim_{k\to\infty} v_\iota^k$, $\sigma_x^\infty = \lim_{k\to\infty} W_x(v_\iota^k)$, and $\sigma_{\text{OPT}}^\infty = \lim_{k\to\infty} W^*(v_\iota^k)$. 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 $\overline{\beta}\sigma_{\text{OPT}}^\infty = \sigma_x^\infty$.

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 2.9 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 $W_x(\tilde{v}_\iota) - \sigma_x^\infty < \tilde{v}_\iota - v_\iota^\infty$.

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

If $v_\iota^\infty = v_\iota$, then by lower semi-continuity, we have $\sigma_x^\infty - W_x(v_\iota) \geq 0$, 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,

$$\overline{\beta} \lim_{k\to\infty} \overline{W}^*(I_\iota^k) \geq \overline{\beta}\,(\sigma_{\text{OPT}}^\infty - (v_\iota^\infty - v_\iota)) > \sigma_x^\infty - (v_\iota^\infty - v_\iota) \geq \lim_{k\to\infty} \overline{W}_x(I_\iota^k).$$

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

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

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

Hence,

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

Case 2: Suppose Clause 2 of Definition 2.9 is not satisfied, so that

1. $\iota \notin x(v, A)$;
2. $\tilde{v}_\iota < v_\iota$;
3. for all $\epsilon > 0$, we have $W^*(v_\iota - \epsilon) < W^*(v_\iota)$; 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 $I_\iota^k = \big\{(v_\iota^k, 0), (v_\iota, 0)\big\}$.

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

As for all $\epsilon > 0$ we have $W^*(v_\iota - \epsilon) < W^*(v_\iota)$, it follows that

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

Thus,

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

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

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

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,

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

□

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 2.9, 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 2.9 and $W_x(\cdot, v_{-\iota}, A)$ is continuous at $\tilde{\tilde{v}}_\iota$. Lemma A.1 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.2. *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.2. □

Lemma A.3. *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.3. ☐

Lemma A.4. *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.2. 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.2, $x(v^k) = x(v^\uparrow - c^\uparrow)$. 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.3, $t^x_n(v^\uparrow - c^\uparrow) \geq t^x_n(v^k) \geq t^x_n(v^{k+1})$ 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.4. ☐

Lemma A.5. *$(\hat{v}, \hat{c})$ is a Nash equilibrium of the investment game (I, A) facing threshold auction (x, p^x).*

Proof. By Lemma 2.2, it suffices to check that bidders choosing $(v^\uparrow_n, c^\uparrow_n)$ cannot profitably deviate to $(v^\downarrow_n, c^\downarrow_n)$ and vice versa. (Recall that $c^\downarrow_n = 0$.)

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

$$\max\{v^\uparrow_n - t^x_n(\hat{v}), 0\} - c^\uparrow_n \geq \max\{v^\uparrow_n - t^x_n(v^\uparrow - c^\uparrow), 0\} - c^\uparrow_n \geq 0.$$

where the first inequality is by Lemma A.4 and the second inequality is by $n \in x(v^{\uparrow} - c^{\uparrow})$. This implies:

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

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.4 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.5. □

Lemma A.6. *If x is monotone, non-bossy, and a β-approximation for allocation, then*

$$W_x(\hat{v}, A) - \sum_{n \in N} \hat{c}_n \geq \beta \max_{(v,c) \in I} \left\{ W^*(v, A) - \sum_{n \in N} c_n \right\}. \tag{19}$$

Proof. Let (v^*, c^*) be a profile of investments that attains the maximum on the right-hand side of (19). By Lemma A.2, $x(\hat{v}) = x(v^{\uparrow} - c^{\uparrow})$. 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.6. □

Combining Lemmata A.5 and A.6 completes the proof.

Proof of Proposition 3.3

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 (6), with $o = x_n(\tilde{v}_n)$), we have

$$\tilde{v}_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] \geq v_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] \,. \tag{20}$$

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

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \,. \tag{21}$$

Combining (21) and (the negative of) (20) yields

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] = v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \,. \tag{22}$$

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

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq p_n(\tilde{v}_n) - p_n(v_n) \tag{23}$$

$$v_n \cdot [x_n(v_n) - x_n(\tilde{v}_n)] \geq p_n(v_n) - p_n(\tilde{v}_n). \tag{24}$$

Now, from (23) and (the negative of) (24) we obtain

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \geq p_n(\tilde{v}_n) - p_n(v_n) \geq v_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] \,. \tag{25}$$

Combining Eq. (22) and Eq. (25), we find that

$$\tilde{v}_n \cdot [x_n(\tilde{v}_n) - x_n(v_n)] = p_n(\tilde{v}_n) - p_n(v_n). \tag{26}$$

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 (26); 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 (26), that $x_n(\tilde{v}_n) \neq x_n(v_n)$. We then have from (22) 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.7. *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 $\alpha^* \in \operatorname{argmax}_{\alpha^l : l \in L} \left\{\alpha^l(F)\right\}$. $\alpha^*(F) = q(F)$, 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:

$$q^F(F') + q^F(G \setminus F') \geq \underbrace{q^F(F' \cup (G \setminus F'))}_{=\alpha^F(F) - c_\iota(\alpha^F)} + \underbrace{q^F(F' \cap (G \setminus F'))}_{=0}. \tag{27}$$

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 (27) 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.7, 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,

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

which implies that for all F,

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

Also by construction, for all F and F',

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

which implies that for all F,

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

Thus, for all F,

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

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