Polynomial Sample Complexity for Blackbox Reductions in Mechanism Design with Additive Bidders

Arya Maheshwari, S. Matthew Weinberg, Eric Xue

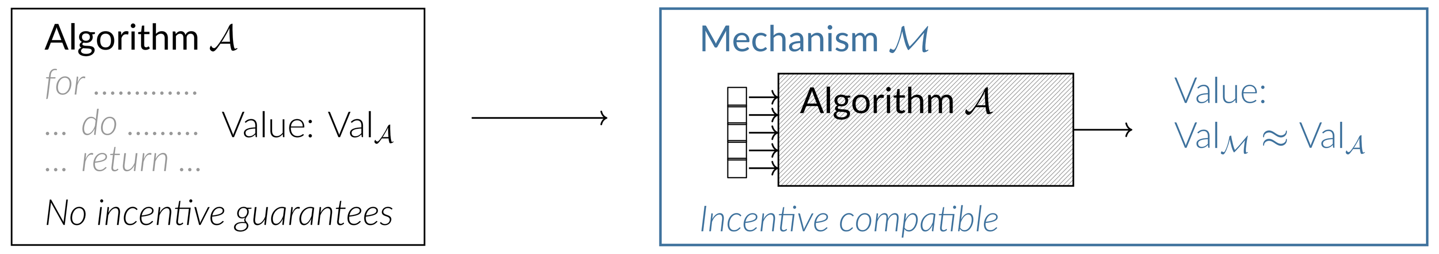
Princeton University

Background

Blackbox Reductions in Mechanism Design.

Say we have an algorithm $\mathcal{A}$ for an optimization task, but agents might misreport input to $\mathcal{A}$.

Can we design an incentive-compatible mechanism $\mathcal{M}$ that uses $\mathcal{A}$ as a blackbox and achieves (approximately) the same objective value?



We study such *reductions* from *Bayesian mechanism design* to Bayesian algorithm design, where bidders have types drawn independently from some underlying distribution D.

Sample Complexity.

In an ideal model, we would have full knowledge of the bidders' type distribution D. In reality, we might only have a limited number of samples from D to work with.

⚠️ **Efficient sample complexity** is a key desideratum for reduction problems:

Existing blackbox reduction procedures (e.g. [4, 2, 1]) all require an **exponential** number of samples ⇒ question: can we improve to **polynomial** sample complexity?

This was posed as the main open problem of [3], for instance, and remains open, *even under additional structural assumptions* on bidder valuations (e.g. additivity).

Model & Preliminaries

- One seller, n items, m (independent) bidders. Welfare maximization[a] setting.
- Bidder $k \in [m]$ has type drawn from $D^{(k)}$, with supp$(D^{(k)}) \subseteq [0,1]^n$.
- **Bayesian incentive compatibility** (BIC): Optimal for each bidder k to report truthfully *assuming* all other bidders do so (i.e. they report $t_j \sim D^{(j)}, \forall j \neq k$).
- Expected welfare: denoted Val$_\mathcal{A}(D)$ for algorithm $\mathcal{A}$, Val$_\mathcal{M}(D)$ for mechanism $\mathcal{M}$.

Model assumptions:

– **Additive valuations:** for allocation $\vec{x}$ and type $\vec{t}_k$, value $v_k(\vec{t}_k, \vec{x}) = \sum_{i\in[n]} x_i \cdot (t_k)_i$.
– **Independent items**: $D^{(k)} = \times_{i\in[n]} D_i^{(k)}$ is a product distribution over items $[n]$.

– **Sample access** to bidder distributions $D^{(k)}$, $\forall k \in [m]$.
– Unrestricted query access to underlying algorithm $\mathcal{A}$ *and its interim form* $\forall k \in [m]$.

Prior Work on Blackbox Reductions

- Hartline & Lucier (2010) initiate study in single-dimensional, welfare-max setting.
- Bei & Huang; Hartline et al. [4] (2011) extend to multi-dimensional setting.
- Daskalakis & Weinberg (2012) use similar techniques for related ε=BIC to BIC reduction for revenue max.
- Dughmi et al. [2] (2020) use *Bernoulli factories* for reductions in *fully-sample-based* (no interim form) welfare-max setting. Cai et al. [1] extend to revenue-max setting.

[a] With the goal of extending to revenue maximization in ongoing work.

Problem Statement

Input: (1) Sample access to distribution $D = \times_{k\in[m]} D^{(k)}$ over m bidders, each with additive valuations over n independent items. (2) Algorithm $\mathcal{A}$ (and its interim form).

Output: A BIC mechanism $\mathcal{M}$ with expected welfare Val$_\mathcal{M}(D)$ at least Val$_\mathcal{A}(D) - O(\varepsilon)$.

Question: Can we leverage the additional structure on the input valuations to obtain improved (i.e. polynomial) sample complexity?

Remark: We focus on sample complexity. No claims about runtime.

Main Result

(Answer: Yes!) There is an **exactly-BIC** mechanism $\mathcal{M}$ that uses $\mathcal{A}$ and its interim form as a blackbox and achieves expected welfare **Val$_\mathcal{M}(\mathbf{D}) \geq$ Val$_\mathcal{A}(\mathbf{D}) - \mathbf{O}(\boldsymbol{\varepsilon})$** using **poly$(\mathbf{n}, \mathbf{m}, \frac{1}{\varepsilon})$ samples** from the bidders' valuation distribution D.

Existing Technique: Replica-Surrogate Matching

Replica-surrogate (R-S) matching: create a separate "interface layer" for each bidder $k \in [m]$ that wraps around $\mathcal{A}$ to guarantee BIC, while losing only $O(\varepsilon)$ welfare.

- **Surrogates**: Draw some number of *"surrogate types"* from $D^{(k)}$. Match bidder k to some surrogate that *will be inputted into $\mathcal{A}$ as a proxy, in place of t_k.*
- **Replicas**: Do the matching by drawing *"replica types"*, also from $D^{(k)}$, that will act as make-believe "competitors" against bidder k for being matched to surrogates.

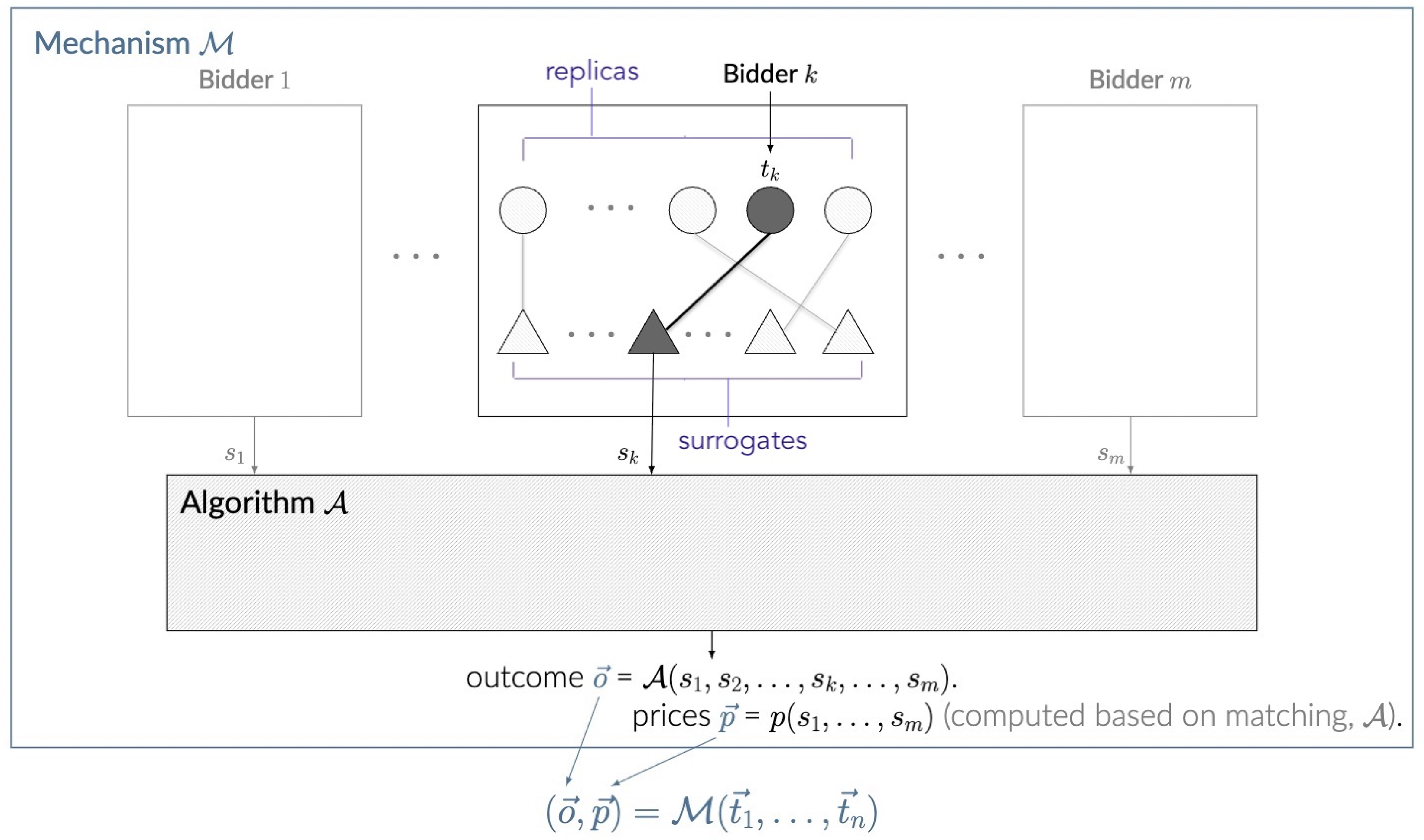


Figure 1. Visualization of R-S matching. Adapted from Figure 1 in Cai et al. [1].

Interpretation: Replicas essentially **induce pricing rules** faced by bidder k for the surrogates, in a way that yields a mechanism $\mathcal{M}$ with the desired BIC and welfare properties:

1. **BIC**, via a key fact: if bidder k reports truthfully, the distribution of the surrogate matched to bidder k is $D^{(k)}$. Can then show matching is incentive compatible.
2. **Welfare approximation:** Val$_\mathcal{M}(D) \approx$ Val$_\mathcal{A}(D)$.

   For good welfare, it turns out that we need to be able to match a large-enough fraction of replicas to surrogates with *similar types*. In turn, this requires drawing *enough replicas and surrogates to cover the type space* (supp$(D^{(k)})$) *sufficiently well.*

   How well? Exponential-sized cover ⇒ exponentially-many replicas/surrogates. ☹️

Our Mechanism: Three Key Techniques

R-S matching upshot: Need exponentially-many surrogates to get good welfare.

1. Sampling by Products.

⚠️ *How to obtain exponentially-many surrogates from only polynomially-many samples?*

💡 Use item independence: sample N times from marginals D_i; take the N^n products.

For each item $i \in [n]$, draw $S_i = \{y_{i,1}, y_{i,2}, \ldots, y_{i,N} : y_{i,j} \leftarrow D_i, \forall j\}$, where $N = \text{poly}(n)$. Then, take exponential-sized **product set** S as replicas/surrogates from D!

$$S := \times_{i\in[n]} S_i = \{(s_1, \ldots, s_n) : s_i \in S_i\}, \text{ where } |S| = N^n = \exp(n).$$

But the draws are not i.i.d. This leads to e.g. incentives issue when including bidder's marginals before taking products: bidder's report now influences other replicas and thus prices ⇒ potential incentive to misreport.

2. Two-Phase Replica-Surrogate Procedure to Learn Approximate Prices.

⚠️ *How to fix incentive issue of a bidder's reported marginals affecting surrogate prices?*

💡 Decouple learning of prices from bidder report via separate "training" R-S phase.

Phase A: "Training" R-S matching.
- Draw *training replicas* $R^{(A)}$, *not including* bidder's report.
- Learn prices for surrogates S in R-S matching with $R^{(A)}$ as the replicas.

Phase B. "Real" R-S matching.
- Draw *real replicas* $R^{(B)}$, *including* bidder's type before taking products.
- Use prices from Phase A to run R-S matching on $(R^{(B)}, S)$ as final result.

Result: w.h.p., Phase A prices are *approximately good* for real Phase B matching. Proof uses marginal-wise concentration and leverages additivity of valuations.

3. Handling Small Errors/Failures with Discarding.

⚠️ *How to handle low-probability failures and small error due to approximate prices?*

💡 Discard allocations to remedy BIC in failure cases with only $O(\varepsilon)$ welfare loss.

Issue: What to do when prices from Phase A are *not* good for Phase B, for some bidder k? Bidder k's surrogate distribution wouldn't be $D^{(k)}$, ruining BIC for all bidders $j \neq k$.

Idea: If this happens, discard allocations for *every other bidder* $j \neq k$ to fix incentives.
- If other bidders get $\emptyset$ anyways, no incentive issues even if k's distribution looks off!

⇒ Preserve BIC + lose only $O(\varepsilon)$ welfare assuming small-enough discard probability.

Future Work

1. Extend to revenue maximization (a la Cai et al. [1]).
2. Generalize to valuation classes beyond additive.
3. Extend to *fully-sample-based* model via Bernoulli factories [2, 1].

Acknowledgements

This work was done as part of AM's senior thesis in the CS department of Princeton University. SMW was funded in part by an NSF CAREER award (CCF-1942497).

[1] Yang Cai, Argyris Oikonomou, Grigoris Velegkas, and Mingfei Zhao. An efficient ϵ-bic to bic transformation and its application to black-box reduction in revenue maximization. In *Proceedings of the Thirty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '21, 2021.

[2] Shaddin Dughmi, Jason Hartline, Robert D. Kleinberg, and Rad Niazadeh. Bernoulli factories and black-box reductions in mechanism design. *J. ACM*, 68(2), January 2021.

[3] Yannai A. Gonczarowski and S. Matthew Weinberg. The sample complexity of up-to-ε multi-dimensional revenue maximization. *J. ACM*, 68(3), March 2021.

[4] Jason D. Hartline, Robert Kleinberg, and Azarakhsh Malekian. *Bayesian Incentive Compatibility via Matchings*, pages 734–747. 2011.