

Public Projects, Boolean Functions, and the Borders of Border's Theorem

PARIKSHIT GOPALAN, VMware Research

NOAM NISAN, Hebrew University and Microsoft Research

TIM ROUGHGARDEN, Stanford University

Border's theorem gives an intuitive linear characterization of the feasible interim allocation rules of a Bayesian single-item environment, and it has several applications in economic and algorithmic mechanism design. All known generalizations of Border's theorem either restrict attention to relatively simple settings or resort to approximation. This article identifies a complexity-theoretic barrier that indicates, assuming standard complexity class separations, that Border's theorem cannot be extended significantly beyond the state of the art. We also identify a surprisingly tight connection between Myerson's optimal auction theory, when applied to public project settings, and some fundamental results in the analysis of Boolean functions.

CCS Concepts: • **Theory of computation** → **Algorithmic mechanism design; Computational pricing and auctions;**

Additional Key Words and Phrases: Auctions, Border's theorem, revenue-maximization, interdependence, correlated values, optimal auctions, Myerson theory, prior-independence

ACM Reference format:

Parikshit Gopalan, Noam Nisan, and Tim Roughgarden. 2018. Public Projects, Boolean Functions, and the Borders of Border's Theorem. *ACM Trans. Econ. Comput.* 6, 3–4, Article 18 (November 2018), 21 pages.

<https://doi.org/10.1145/3274645>

1 INTRODUCTION

Let us start by considering the computational complexity of a basic problem in probability theory: characterizing the possible vectors of marginal probabilities in a probability space. Questions of this type have already been asked by George Boole in the 19th century (see [27]) under the name “conditions of possible experience.” Here is a simple but very relevant special case that we will focus on: For which vectors (p_0, p_1, \dots, p_n) does there exist a probability space with events E, X_1, \dots, X_n , with X_1, \dots, X_n independent with $\Pr[X_i] = 1/2$ for all $i = 1, 2, \dots, n$, such that $p_0 = \Pr[E]$ and $p_i = \Pr[E|X_i]$ for all $i = 1, 2, \dots, n$?

Noam Nisan was partially supported by Israeli Academy of Sciences under Grant numbers 230/10 ISF and 1435/14 ISF administered by the Israeli Academy of Sciences. Tim Roughgarden was supported in part by National Science Foundation NSF under Grant numbers CCF-1215965 NSF and CCF-1524062 NSF.

Authors' addresses: P. Gopalan, VMware, Inc., Palo Alto, CA; N. Nisan, Room A432, Rothberg building, School of Computer Science and Engineering, Hebrew University of Jerusalem; T. Roughgarden, Stanford University, 474 Gates Building, 353 Serra Mall, Stanford, CA 94305.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Association for Computing Machinery.

2167-8375/2018/11-ART18 \$15.00

<https://doi.org/10.1145/3274645>

The reader may pause for a second here and convince themselves that this is not a trivial question: For example, $\Pr[E] = 1/2$ and $\Pr[E|X_1] = \Pr[E|X_2] = 0.7$ is possible while $\Pr[E] = 1/2$ and $\Pr[E|X_1] = \Pr[E|X_2] = 0.8$ is not possible!

1.1 Relevance to Bayesian Mechanism Design

Why would anyone care about this problem? One motivation comes from mechanism design, specifically the problem of characterizing the set of feasible interim allocation rules. To explain, recall that, in a generic Bayesian mechanism design problem, a principal faces n strategic agents or players, each holding some private information, termed its *type*, where the tuple of types is distributed according to some known prior distribution. The mechanism must specify an (*ex post*) *allocation rule*: for each possible profile of agent types, an outcome chosen from some family of possible outcomes or, more generally, a probability distribution over outcomes. The description of a mechanism is thus naturally exponential in the number of agents, even if there are only two possible types per agent and two outcomes. This exponential description size is the first reason that mechanism design problems are difficult both mathematically and computationally. In particular, while it turns out that most mechanism design problems of interest are easily expressed as linear programs, computational efficiency does not follow due to the exponential size of these linear programs.

There is still hope, however. The goals and constraints of Bayesian mechanism design problems typically depend only on the marginals of the allocation rule, also known as the *interim allocations*: for each possible type of each agent, the average outcome over the types of the other players. This reduces the numbers of variables and constraints to be linear in the number of players rather than exponential. One would naturally hope that analysis in terms of the interim allocations helps the mathematical understanding of the problem in addition to computational tractability.

This is *almost* the case. The only rub is the combinatorial issue of which interim allocation rules are *feasible*: For which values of the interim allocation probabilities do there exist values of the (*ex post*) allocation probabilities with these marginals? Checking feasibility of an interim allocation rule is an instance of verifying the consistency of a collection of marginals, the problem described above.

Maskin and Riley [19] were the first to highlight the importance of characterizing feasible interim allocation rules; their motivation was to develop an analog of Myerson's characterization of optimal single-item auctions with risk-neutral bidders [22] for the case of risk-adverse bidders. Matthews [20] proposed an intuitive necessary condition and conjectured that it is sufficient for feasibility, and Border [2] proved this conjecture.¹ For further applications and interpretations in economics, see several other works [15, 17, 21].

Border's theorem also has computational implications. As a linear characterization that uses only "simple" linear inequalities, it implies that checking the feasibility of an interim allocation rule of a single-item auction is a coNP problem (assuming finite type spaces and explicitly given type distributions). In simultaneous and independent works, Alaei et al. [1] and Cai et al. [5] show that the problem is in fact in P. Alaei et al. [1] also gave a polynomial-size extended formulation of the feasible interim allocation rules (with polynomially many auxiliary variables and only polynomially many constraints).

To what extent can Border's theorem be generalized to other mechanism design problems? This question has been the focus of much of the recent work in algorithmic mechanism design since, as explained earlier, it lies at the heart of the efficient computational treatment of multiplayer

¹For the finite version of Border's theorem that we consider (see Section 2.3), there are also more combinatorial proofs [3, 7].

mechanism design challenges. The current state of knowledge, discussed in detail in Section 1.4, is that there are analogs of Border's theorem in settings modestly more general than single-item auctions, such as k -unit auctions with unit-demand bidders [1, 5], and that approximate versions of Border's theorem exist fairly generally [5, 6].

Can the state of the art be improved upon? Can we provide computationally useful exact extensions of Border's theorem?

1.2 Summary of Results: The Borders of Border's Theorem

The first main take-away from this article is that, under widely believed complexity assumptions, Border's theorem cannot be extended significantly beyond the state of the art. For example, our negative results imply the (conditional) impossibility of an exact Border's theorem even for the following extremely simple mechanism design setting.

Definition 1.1 (Boolean Public Project Problem). In the *Boolean public project* mechanism design problem, there are only two possible social outcomes: “yes” and “no” (e.g., build a bridge or not). Each of the n players has valuation 0 for “no.” Player i 's value for “yes” is either 0 or a_i ; the a_i 's are publicly known, but only player i knows which of 0 or a_i is its true value for “yes.” The distribution on the players' types is uniform, with all 2^n possibilities equally likely.

Key to our approach is the study of the computational complexity of the OPTREV problem: Given a description of a mechanism design problem—for each agent i and type t_i , the probability that i 's type is t_i —compute the maximum expected revenue obtained by a mechanism that is Bayesian incentive-compatible and interim individually rational. (See Section 2 for formal definitions.) We prove in Theorem 4.1 that the OPTREV problem is #P-hard² for Boolean public project problems. A similar result has been independently proved by Yang Cai (private communication). We note that the OPTREV problem is hard despite the setting's status as “completely solved” from a revenue-maximization perspective: Myerson's optimal auction theory [22] tells us exactly what the optimal auction is (always pick the outcome with the highest sum of “virtual values”), and this auction is trivial to implement (since virtual values are trivial to compute). Thus, while we know what to do (run Myerson's optimal auction) and it is computationally efficient to do it, it's intractable to compute (exactly) what our expected revenue will be!³

But, so what? Isn't identifying the optimal mechanism the problem we really care about? The point is this (Theorem 3.3): *A generalization of Border's theorem (defined formally in Section 3.1) would imply that the OPTREV problem is relatively tractable, formally within the complexity class P^{NP} .* Combining this result with our #P-hardness result for the OPTREV problem for Boolean public projects rules out an analog of Border's theorem for that setting unless $\#P \subseteq P^{NP}$, which is widely believed not to be the case.⁴

This impossibility result is not an artifact of the fact that Boolean public project settings are not “downward-closed.” For example, we can rule out a generalized Border's theorem (here and below, assuming $\#P \not\subseteq P^{NP}$) for the setting of single-minded bidders with known bundles even when all bundles have size 2 (Theorem 4.6). The same reduction rules out a Border's-type theorem for multi-item auctions with unit-demand bidders (Corollary 4.8). Analogous hardness results even apply to the mathematically well-behaved class of matroid environments, including graphical matroids (Theorem 4.9).

²In this article, all our hardness results are under general Turing reductions.

³Solving the OPTREV problem is nontrivial even when you know what the optimal mechanism is because the expectation is over the exponentially many type profiles.

⁴Recall that P^{NP} denotes the problems that can be solved in polynomial time using an NP oracle. Toda's theorem implies that if $\#P \subseteq P^{NP}$, then the polynomial hierarchy collapses to P^{NP} .

Taken together, our negative results suggest that Border's theorem cannot be extended significantly beyond the cases already identified [1, 2, 5, 7] without resorting to approximation (as in Cai et al. [5, 6]). In particular, computationally useful Border's-type characterizations are apparently far rarer than computationally efficient characterizations of optimal auctions (as in Myerson [22]).

1.3 Summary of Results: Applications to Boolean Function Analysis

The second main take-away of this article is orthogonal to the first: There is a surprisingly tight connection between classical optimal auction theory, especially in the setting of Boolean public projects, and the analysis of Boolean functions.

To see this, here is yet another formulation of the problem stated at the beginning of the article, this time in the setting of the Fourier transform of Boolean functions. For a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, we will consider its *Chow parameters* (a.k.a. level 0 and 1 Fourier coefficients), which are defined as

$$\begin{aligned}\hat{f}(0) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x)]; \\ \hat{f}(i) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x)(-1)^{1+x_i}] \text{ for } i \in [n].\end{aligned}$$

We can convexify this set by considering all bounded functions $f : \{0, 1\}^n \rightarrow [0, 1]$. The space of Chow parameters for such functions is a convex polytope which we denote by C_n .

We observe that the space of feasible interim allocation rules for the Boolean public project setting (given by the conditional allocations) is essentially equivalent to the space of feasible Chow parameters for functions $f : \{0, 1\}^n \rightarrow [0, 1]$. This lets us reinterpret our hardness results for the setting of Boolean Public Projects in the language of Chow parameters as well: We can view the problem of maximizing revenue as maximizing a weighted sum of the form $\sum_{i=0}^n a_i \hat{f}(i)$ over C_n . Theorem 5.5 shows that this problem is #P-hard, and hence testing membership of a vector in C_n is also #P-hard. At this point, we can also return to the problem regarding marginal probabilities that this article started with and observe that this also implies its #P-hardness.

This equivalence is also useful in the converse direction and provides, in Lemma 5.4, a simple alternative stand-alone analysis of the OPTREV problem for the Boolean Public Project setting that is based on a simple analysis of Boolean functions rather than relying on Myerson's analysis of optimal auctions.

1.4 Related Work

Three recent and independent papers ask to what extent Border's theorem can be extended beyond the original setting of single-item auctions [2] and provide some positive results. The results in this article give senses in which their results are close to the best possible.

Alaei et al. [1] give an analog of Border's theorem for every single-parameter matroid environment in the form of an exponential-size set of linear inequalities that characterizes the feasible interim allocation rules. They illustrate some special cases, such as k -unit auctions, in which this characterization can be used to test the feasibility of a rule—and more generally optimize over the set of all feasible rules—in polynomial time. For general matroids, their linear characterization uses inequalities for which the right-hand side is #P-hard to compute (this follows from the reductions in the present work); thus, for general matroids, their characterization does not meet our notion of a “generalized Border's theorem” (Definition 3.1), and, indeed, by our Theorem 4.9 it cannot (unless the polynomial hierarchy collapses).

The contributions of Che et al. [7] relevant to the present work are similar in spirit to but technically different from those of Alaei et al. [1]: They give an analog of Border's theorem for a class of multi-unit environments involving “paramodular” constraints on the number of units each bid-

der can get.⁵ In general, this linear characterization uses inequalities that are computationally intractable to compute, but it includes some tractable special cases, such as when the upper and lower bounds of the different bidders are uncoupled.⁶

Cai et al. [5] also identify some computationally tractable extensions of Border's theorem, for example to the case of multi-item auctions with additive bidders (i.e., no feasibility constraints).⁷ Cai et al. [5, 6] also develop a theory of "approximate" Border's-type theorems that encompasses a much wider swath of mechanism design settings, including all of the concrete settings that we consider in this article. Such approximate theorems identify a set of linear inequalities with the property that every feasible solution is close (in ℓ_∞ norm, say) to a feasible interim allocation rule, and, conversely, Cai et al. [5, 6] show that approximate versions of Border's theorem are very useful for algorithmic mechanism design: Roughly, for an arbitrary constant $\epsilon > 0$, they enable the polynomial-time computation of a mechanism that is approximately Bayesian incentive compatible (up to an incentive of ϵ to misreport) and is approximately revenue-optimal (up to a loss of ϵ). Our results in this article imply that the approximation approach taken in Cai et al. [5, 6] is unavoidable in that no exact and computationally useful analog of Border's theorem can exist for most of the settings they consider (unless $\#P \subseteq P^{NP}$).

Our work shares some of the spirit of recent works that use complexity to identify barriers in mechanism design. For example, Roughgarden [28] uses complexity assumptions to rule out the existence of near-optimal Nash equilibria in simple auctions, while Roughgarden and Talgam-Cohen [29] use them to show the nonexistence of Walrasian equilibria in certain classes of markets with indivisible goods. For another example, Daskalakis et al. [10] consider a single-bidder multi-item (and hence multi-parameter) setting and prove that it is $\#P$ -hard to compute a description of the revenue-maximizing incentive-compatible mechanism.⁸ Note that this problem is *not* hard in most of the (single-parameter) settings that we consider, where the optimal mechanism is simple to compute and write down (it is just a "virtual welfare maximizer" with virtual values given by simple explicit formulas, as per Myerson [22]). What is hard for us is computing the expected revenue obtained by the (simple-to-describe) optimal mechanisms, not the mechanism design problem per se. Still more distant from the present work are previous papers on the intractability of computing optimal deterministic mechanisms in various settings [4, 8, 13, 26].

2 PRELIMINARIES

2.1 Mechanism Design Settings

We recall first the standard model of binary single-parameter mechanism design settings, with players' valuations drawn from a commonly known product distribution. Formally, a *single-parameter environment* \mathcal{E} consists of the following ingredients: (i) a player set $U = \{1, 2, \dots, n\}$; (ii) for each player i , a finite set V_i of possible nonnegative *valuations*; (iii) for each $i \in U$ and $v_i \in V_i$, a *prior* probability $f_i(v_i)$ that i 's valuation is v_i ; and (iv) a non-empty collection $\mathcal{F} \subseteq 2^U$ of *feasible sets*. For example (see also Section 4):

⁵Che et al. [7] define paramodularity as the upper bounds being submodular, the lower bounds being supermodular, and the two constraints being "compliant," meaning irredundant in a certain sense.

⁶Of course, Border's-type theorems like those in Alaei et al. and Che et al. [1, 7] are mathematically interesting even when they are not immediately computationally useful, and they do provide significant structural information about the constraints that define the set of feasible interim allocation rules.

⁷Both Alaei et al. [1] and Cai et al. [5] also give (different) compact formulations (i.e., polynomially many variables and constraints) of the feasible interim allocation rules for single-item settings.

⁸Daskalakis et al. [10] assume an additive bidder, with the prior distribution over valuations encoded in the natural succinct way.

- (1) A single-item auction (with n bidders) corresponds to an environment with $\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \dots, \{n\}\}$.
- (2) In a k -unit auction with unit-demand bidders, \mathcal{F} is all subsets of U with size at most k .
- (3) In a *public project* environment, $\mathcal{F} = \{\emptyset, U\}$.
- (4) In a *single-minded* environment, the set \mathcal{F} is described implicitly as follows. There is a set M of items and a subset $S_i \subseteq M$ desired by each bidder i . A set $F \subseteq U$ belongs to \mathcal{F} if and only if no desired bundles conflict: $S_i \cap S_j = \emptyset$ for all distinct $i, j \in F$. For example, if $|S_i| = 2$ for every i , then feasible sets correspond to the matchings in a graph with vertices M and edges corresponding to the S_i 's.

We also consider multi-parameter environments. Because our primary contributions are impossibility results, we confine ourselves to the simplest such environments (negative results obviously apply also to generalizations). A *multi-item auction* environment differs from a single-parameter environment in the following respects. First, there is also a set M of items. Second, a valuation \mathbf{v}_i of a bidder i is now a non-negative vector indexed by M . We restrict attention to additive bidders, meaning that the value that a player i derives from a set $S \subseteq M$ of items is just the sum $\sum_{j \in S} v_{ij}$. Third, \mathcal{F} is now a subset of $2^{U \times M}$, indicating which allocations of items to bidders are possible. For example, the standard setting of unit-demand bidders can be encoded by defining \mathcal{F} to be the subsets F of $U \times M$ in which, for every bidder i , there is at most one pair of the form (i, j) in F (and also at most one such pair for each j).

By a *setting*, we mean a family of mechanism design environments. For example, all single-item auction environments (with any number n of players, any valuation sets, and any prior distribution), all public project environments, all multi-item unit-demand auction environments, and the like.

2.2 Bayesian Incentive-Compatible Mechanisms

This section reviews the classical setup of Bayesian mechanism design problems, as in Myerson [22]. Fix a binary single-parameter environment, as defined in Section 2.1. A (direct-revelation) *mechanism* (\mathbf{x}, \mathbf{p}) comprises an *allocation rule* $\mathbf{x} : \mathbf{V} \rightarrow \{0, 1\}^n$ and a *payment rule* $\mathbf{p} : \mathbf{V} \rightarrow \mathbb{R}_+^n$, where $\mathbf{V} = V_1 \times \dots \times V_n$. The former is a map (possibly randomized) from each bid vector \mathbf{b} —with one bid per player—to a characteristic vector of a feasible set in \mathcal{F} ; the latter is a map (possibly randomized) from each bid vector \mathbf{b} to a payment vector \mathbf{p} , with one payment per player. For the questions we study, we can restrict attention to truthful mechanisms (via the Revelation Principle, e.g. [23]), and we henceforth use the true valuations \mathbf{v} in place of the bids \mathbf{b} .

A mechanism (\mathbf{x}, \mathbf{p}) and prior distribution \mathbf{F} over valuations together induce an *interim allocation rule*

$$y_i(v_i) = \mathbf{E}_{\mathbf{v}_{-i} \sim \mathbf{F}_{-i}}[x_i(v_i, \mathbf{v}_{-i})], \quad (1)$$

which describes the probability (over the randomness in \mathbf{v}_{-i} and any randomness in \mathbf{x}) that bidder i is chosen when it reports the valuation v_i . Similarly, the *interim payment rule*

$$q_i(v_i) = \mathbf{E}_{\mathbf{v}_{-i} \sim \mathbf{F}_{-i}}[p_i(v_i, \mathbf{v}_{-i})] \quad (2)$$

describes the expected payment made by i when it reports v_i . The pair (y, q) is the *reduced form* of the mechanism (\mathbf{x}, \mathbf{p}) . We sometimes call \mathbf{x} and \mathbf{p} *ex post* rules for emphasis.

A mechanism (\mathbf{x}, \mathbf{p}) for an environment is *Bayesian Incentive Compatible (BIC)* if truthful bidding is a Bayes-Nash equilibrium. We assume that bidders are risk-neutral and have quasi-linear utility (value minus payment) and can therefore use linearity of expectation to write succinctly the BIC condition in terms of the reduced form (y, q) of a mechanism:

$$v_i y_i(v_i) - q_i(v_i) \geq v_i y_i(v'_i) - q_i(v'_i) \quad (3)$$

for every bidder i , true valuation v_i , and reported valuation v'_i .

Similarly, we can express the *Interim Individual Rationality (IIR)* requirement, stating that truthful bidding leads to non-negative expected utility, by

$$v_i y_i(v_i) - q_i(v_i) \geq 0 \quad (4)$$

for every bidder i and true valuation v_i .

We can also write the seller's expected revenue

$$\sum_{i=1}^n \sum_{v_i \in V_i} f_i(v_i) \cdot q_i(v_i) \quad (5)$$

in terms of the reduced form of the mechanism. In the *optimal (revenue-maximizing) mechanism design problem*, the goal is to identify the BIC and IIR mechanism (\mathbf{x}, \mathbf{p}) that maximizes Equation (5).

Multi-item auction environments can be treated similarly, with an (ex post) allocation rule choosing a feasible set of \mathcal{F} for each valuation profile \mathbf{v} and $x_{ij}(\mathbf{v})$ now denoting whether or not bidder i receives the item j . (There is no need to keep track of a separate payment for each item received.) For example, because we assume that bidders are additive, the BIC constraints for a multi-item environment can be phrased in terms of the reduced form (\mathbf{y}, \mathbf{q}) of a mechanism by

$$\sum_{j \in M} v_{ij} y_{ij}(\mathbf{v}_i) - q_i(\mathbf{v}_i) \geq \sum_{j \in M} v_{ij} y_{ij}(\mathbf{v}'_i) - q_i(\mathbf{v}'_i) \quad (6)$$

for every bidder i , true valuation \mathbf{v}_i , and reported valuation \mathbf{v}'_i .

2.3 Border's Theorem for Single-Item Auction Environments

As discussed in the Introduction, there are several applications that rely on understanding when an interim allocation rule \mathbf{y} is induced by some ex post allocation rule \mathbf{x} . Such interim rules are said to be *feasible*.

Border's Theorem [2] characterizes interim feasibility for single-item auction environments. To review it, fix such an environment and assume without loss of generality that the valuation sets V_1, \dots, V_n are disjoint.⁹ To derive an obvious necessary condition for feasibility, consider an ex post allocation rule \mathbf{x} with induced interim rule \mathbf{y} . Fix for each bidder i a set $S_i \subseteq V_i$ of *distinguished* valuations. Linearity of expectation implies that the probability, over the random valuation profile $\mathbf{v} \sim \mathbf{F}$ and any coin flips of the allocation rule \mathbf{x} , that the winner of the item has a distinguished valuation is

$$\sum_{i=1}^n \sum_{v_i \in S_i} f_i(v_i) y_i(v_i). \quad (7)$$

The probability, over $\mathbf{v} \sim \mathbf{F}$, that there is a bidder with a distinguished type is

$$1 - \prod_{i=1}^n \left(1 - \sum_{v_i \in S_i} f_i(v_i) \right). \quad (8)$$

Since there can only be a winner with a distinguished type when there is a bidder with a distinguished type (and at most one winner, in any case), the quantity in Equation (7) can only be less than that in Equation (8). Border's theorem asserts that satisfying these linear (in \mathbf{y}) constraints,

⁹We can enforce this by thinking of each $v_i \in V_i$ as the pair $\{v_i, i\}$.

ranging over all choices of $S_1 \subseteq V_1, \dots, S_n \subseteq V_n$, is also a sufficient condition for the feasibility of an interim allocation rule y .

THEOREM 2.1 (BORDER'S THEOREM [2]). *In a single-item environment, an interim allocation rule y is feasible if and only if for every choice $S_1 \subseteq V_1, \dots, S_n \subseteq V_n$ of distinguished types,*

$$\sum_{i=1}^n \sum_{v_i \in S_i} f_i(v_i) y_i(v_i) \leq 1 - \prod_{i=1}^n \left(1 - \sum_{v_i \in S_i} f_i(v_i) \right). \quad (9)$$

3 GENERALIZED BORDER'S THEOREMS AND COMPUTATIONAL COMPLEXITY

3.1 Generalizing Border's Theorem

What do we actually mean by a "Border's-type theorem?" Since we aim to prove impossibility results, we should adopt a definition that is as permissive as possible. Border's theorem (Theorem 2.1) gives a characterization of the feasible interim allocation rules of a single-item environment as the solutions to a finite system of linear inequalities. This by itself is not interesting: Since the set is a polytope,¹⁰ it is guaranteed to have such a characterization (see, e.g., Ziegler [32]). The appeal of Border's theorem is that the characterization uses only the "nice" linear inequalities in Chow (9). Our "niceness" requirement is that the characterization uses only linear inequalities that can be efficiently recognized and tested. This is a weak necessary condition for such a characterization to be computationally useful.

Definition 3.1. A *generalized Border's theorem* holds for the mechanism design setting Π if, for every environment $\mathcal{E} \in \Pi$, there is a linear inequality system $\mathcal{L}(\mathcal{E})$ such that the following properties hold.

- (1) (Characterization) For every $\mathcal{E} \in \Pi$, the feasible solutions of $\mathcal{L}(\mathcal{E})$ are precisely the feasible interim allocation rules of \mathcal{E} .
- (2) (Efficient recognition) There is a polynomial-time algorithm that, given as input a description of an environment $\mathcal{E} \in \Pi$ and a linear inequality, decides whether or not it belongs to $\mathcal{L}(\mathcal{E})$.¹¹ Note that the description length of \mathcal{E} is polynomial in n , the $|V_i|$'s, and the maximum number of bits needed to describe a valuation or a prior probability.
- (3) (Efficient testing) There is a polynomial $p(\cdot)$ such that, for every $\mathcal{E} \in \Pi$, the natural encoding length of every inequality of $\mathcal{L}(\mathcal{E})$ is at most $p(\ell)$, where ℓ is the description length of \mathcal{E} . (The number $|\mathcal{L}(\mathcal{E})|$ of inequalities can still be exponential.) Thus, deciding whether or not a given point $\mathbf{x} \in \mathbb{R}^n$ satisfies a given inequality of $\mathcal{L}(\mathcal{E})$ can be done in time polynomial in the descriptions of \mathcal{E} and \mathbf{x} , just by computing and comparing the two sides of the inequality.

For example, consider the original Border's theorem, for single-item auction environments (Theorem 2.1). The recognition problem is straightforward: The left-side of (9) encodes the S_i 's, from which the right-hand side can be computed and checked in time polynomial in the description of \mathcal{E} . It is also evident that every inequality in Equation (9) has a polynomial-length description.

The characterization in Theorem 2.1 and the extensions in other works [1, 5, 7] have additional features not required or implied by Definition 3.1, such as polynomial-time separation oracles (and even a compact extended formulation in the single-item case [1]).¹² All of our impossibility results

¹⁰The set of ex post allocation rules is a polytope, and the feasible rules are the image of this polytope under a linear map (and hence also a polytope).

¹¹Note that this is a weaker assumption than requiring the efficient recognition of an arbitrary valid inequality.

¹²Separation can be hard even when recognition and testing are easy (see, e.g., Grotschel et al. [16] for some examples in combinatorial optimization).

rule out analogs of Border's theorem that merely satisfy Definition 3.1, let alone these stronger properties.

A generalized Border's theorem implies that the problem of testing the feasibility of an interim allocation rule is in coNP. To prove that such a rule for an environment \mathcal{E} is not feasible, one simply exhibits an inequality of $\mathcal{L}(\mathcal{E})$ that the rule fails to satisfy: There is always such an inequality by Definition 3.1 (i), and verifying this failure reduces to the recognition and testing problems for $\mathcal{L}(\mathcal{E})$, which by Definition 3.1 (ii, iii) are polynomial-time solvable.

Formally, we define the MEMBERSHIP problem for a setting Π as: Given as input a description of an environment \mathcal{E} and an interim allocation rule y for it, decide whether or not y is feasible.

PROPOSITION 3.2. *If a generalized Border's theorem holds for the mechanism design setting Π , then the MEMBERSHIP problem for Π belongs to coNP.*

A generalized Border's theorem does not necessarily imply a separation oracle with analogous complexity since it does not provide a way to efficiently verify that a feasible interim allocation rule satisfies all of the (possibly exponentially many) inequalities of $\mathcal{L}(\mathcal{E})$.

3.2 Impossibility Results from Computational Intractability

We now forge a connection between the existence of generalized Border's theorems and the computational complexity of natural optimization problems. For a setting Π , the OPTREV(Π) problem is: Given a description of an environment $\mathcal{E} \in \Pi$, compute the expected revenue earned by the optimal BIC and IIR mechanism. The main result of this section shows that a generalized Border's theorem exists for a setting only when it is relatively tractable to solve exactly the OPTREV problem.

THEOREM 3.3. *If a mechanism design setting Π admits a generalized Border's theorem, then the OPTREV(Π) problem belongs to P^{NP} .¹³*

We later apply Theorem 3.3 in the form of the following corollary.

COROLLARY 3.4. *If the OPTREV(Π) problem is #P-hard, then there is no generalized Border's theorem for Π (unless the polynomial hierarchy collapses).*

In the next section, we prove that the OPTREV problem is #P-hard for many simple settings, ruling out the possibility of generalized Border's theorems for them (conditioned on $\#P \not\subseteq P^{NP}$).

PROOF OF THEOREM 3.3. Consider a setting Π in which a generalized Border's theorem holds and an instance of the OPTREV(Π) problem (a description of an environment $\mathcal{E} \in \Pi$). We compute the optimal expected revenue of a BIC and IIR mechanism via linear programming as follows.

The decision variables of our linear program correspond to the components of an interim allocation rule y and payment rule q . The number of variables is polynomial in n and $\max_i |V_i|$ and hence in the description of \mathcal{E} . The (linear) objective function is to maximize the expected seller revenue, as in Equation (5). The BIC and IIR constraints can be expressed as a polynomial number of linear inequalities, as in Equations (3) and (4), respectively. By assumption, the interim feasibility constraint can be expressed by a linear inequality system $\mathcal{L}(\mathcal{E})$ that satisfies the properties of Definition 3.1. Thus, the optimal objective function value of the linear program (LP) that maximizes Equation (5) subject to Equations (3), (4), and $\mathcal{L}(\mathcal{E})$ is the solution to the given instance of OPTREV(Π).

To solve (LP), we turn to the ellipsoid method [18], which reduces the solution of a linear program to a polynomial number of instances of a simpler problem (plus polynomial additional computation). The relevant simpler problem for us is a *membership oracle*: Given an alleged reduced

¹³Recall that P^{NP} denotes the problems that can be solved in polynomial time using an NP oracle (or, equivalently, a coNP oracle).

form (y, q) , decide whether or not (y, q) is feasible for (LP). Using Proposition 3.2 and the fact that there are only polynomially many constraints of the form seen in Equations (3) and (4), we have a coNP membership oracle. (The most common way to apply the ellipsoid method, on the other hand, involves a *separation oracle*: Given an alleged reduced form (y, q) , either verify that (y, q) is feasible for (LP) or, if not, produce a constraint of (LP) that it violates. Our assumption of a generalized Border's theorem for Π does not include also a separation oracle of the same complexity; hence, we use the version of Ellipsoid that is based on a membership oracle.)

For optimization over polytopes described by linear inequalities of bounded size, assuming that one knows a priori a feasible point (y_0, q_0) , the ellipsoid method can also be used to reduce the solution of a linear program to a polynomial number of invocations of a membership oracle (see Schrijver [30, P. 189]). The size bound on the defining linear inequalities is implied by the Efficient Testing condition in Definition 3.1. Computing a feasible point is trivial in our context: We can just consider a mechanism that outputs some constant outcome irrespective of players' types (with payments that are always zero) and the induced constant interim allocation rule.

We conclude that the linear program (LP) and hence the OPTREV problem can be solved using a polynomial number of invocations of a membership oracle and polynomial additional computation. Since the membership problem for (LP) belongs to coNP, the OPTREV problem belongs to P^{NP} . \square

What we have actually shown is a general Turing reduction from the OPTREV(Π) problem to the MEMBERSHIP problem for Π .

COROLLARY 3.5. *If the OPTREV(Π) problem is #P-hard, then so is the MEMBERSHIP problem for Π .*

More generally, the proof of Theorem 3.3 shows that a generalized Border's theorem allows an arbitrary linear function of the interim allocation and payment rules to be optimized over the space of BIC and IIR mechanisms in P^{NP} . For example, let OPTWEL(Π) be the problem of (given an environment $\mathcal{E} \in \Pi$) computing the maximum expected welfare achieved by a BIC and IIR mechanism.¹⁴ Since the expected welfare obtained by a mechanism can be written as $\sum_{i=1}^n \sum_{v_i \in V_i} f_i(v_i) v_i y_i(v_i)$ for a single-parameter environment or as $\sum_{i=1}^n \sum_{v_i \in V_i} f_i(v_i) \sum_{j \in M} v_{ij} y_{ij}(v_i)$ for a multi-item environment, we have the following corollary.

COROLLARY 3.6. *If the OPTWEL(Π) problem is #P-hard, then there is no generalized Border's theorem for Π (unless the polynomial hierarchy collapses).*

Remark 3.7. By the same reasoning as in the proof of Theorem 3.3, and under the same complexity assumption, #P-hardness of the OPTREV(Π) problem rules out any PH algorithm that recognizes the set of interim allocation rules for the setting Π , not just via a generalized Border's theorem.

4 COMPLEXITY OF COMPUTING THE OPTIMAL EXPECTED REVENUE AND WELFARE

This section shows that, in several simple settings, the OPTREV or the OPTWEL problem is #P-hard. Together with Corollaries 3.4 and 3.6, these results effectively rule out, conditioned on $\#P \not\subseteq P^{NP}$, significant generalizations of Border's theorem beyond those that are already known.

4.1 Preliminaries and Examples

Recall the definition of the OPTREV problem for a setting Π : Given a description of an environment $\mathcal{E} \in \Pi$, compute the maximum expected revenue obtained by any BIC and IIR mechanism. Even if

¹⁴The welfare-maximizing mechanism is, of course, the VCG mechanism (e.g., Nisan [23]), but, even knowing this, it is not generally trivial to compute its expected welfare.

one is handed the optimal mechanism on a silver platter, and this mechanism runs in polynomial time for every valuation profile, naive computation of its expected revenue requires running over the exponentially many valuation profiles. The question is whether or not there are more efficient methods for computing this expected value.

To develop context and intuition for the problem, we review the argument that the OPTREV problem for single-item auctions can be solved in polynomial time (see also Alaei et al. and Cai et al. [1, 5]).¹⁵

- (1) For each bidder i and possible valuation $v_i \in V_i$, compute the corresponding (*ironed*) *virtual valuation* $\varphi_i(v_i)$, as in Myerson [22].¹⁶ This can be done straightforwardly in time polynomial in the size of \mathcal{E} .
- (2) By Elkind and Myerson [14, 22], the (ex post) allocation rule \mathbf{x}^* of the optimal mechanism awards the good to the bidder with the highest positive virtual valuation (breaking ties lexicographically, say), if any, and to no one otherwise. The second step of the algorithm is to compute the interim allocation rule \mathbf{y}^* induced by \mathbf{x}^* . This can be done in polynomial time via a simple computation.¹⁷
- (3) By Elkind and Myerson [14, 22], the solution to the OPTREV problem equals $\sum_{i=1}^n \mathbf{E}_{v_i \sim F_i} [\varphi_i(v_i) y_i^*(v_i)]$; given the virtual valuations and the interim allocation rule, this quantity is trivial to compute in polynomial time.

4.2 Public Projects

Recall that, in a public project environment, there are only two outcomes: choose all players (“build the bridge”) or no player (“not”). We now show that the OPTREV problem is hard in such environments, even in the extremely simple case when each player is equally likely to have a zero valuation or a known positive valuation for the “yes” outcome.¹⁸

THEOREM 4.1. *The OPTREV problem is #P-hard for the public project setting, even when every player has only two possible valuations, and the valuation distribution is uniform.*

Theorem 4.1 follows immediately from Lemmas 4.4 and 4.5, given below. Section 5 shows how Theorem 4.1 can be usefully reinterpreted in the context of Boolean function analysis.

Theorems 3.3 and 4.1 immediately imply the following.

COROLLARY 4.2. *Unless the polynomial hierarchy collapses, there is no generalized Border's theorem for the public project setting.*

To prove Theorem 4.1, we first reformulate the OPTREV problem in the special case of an environment \mathcal{E} in which each bidder i is equally likely to have the valuation 0 or the valuation a_i . We show it is equivalent to the #P-complete problem of computing the Khintchine constant of a vector, which is defined as follows (cf., [12]).

Definition 4.3. For a vector $a \in \mathbb{R}^n$, define

$$K(a) = \mathbf{E}_{x \sim \{\pm 1\}^n} [|x \cdot a|]$$

to be the *Khintchine constant* for a .

¹⁵This is not surprising in light of Theorems 2.1 and 3.3.

¹⁶There is an analogous simple formula for the case of a discrete set of bidder valuations [14].

¹⁷For a bidder i and valuation v_i , $y_i^*(v_i)$ is 0 if $\varphi_i(v_i) \leq 0$, and otherwise is $\prod_{j \neq i} \Pr_{v_j} [\varphi_j(v_j) < \varphi_i(v_i)]$.

¹⁸Note that the OPTWEL problem is trivial to solve in the public project setting.

It is known that

$$\frac{\|a\|_2}{\sqrt{2}} \leq K(a) \leq \|a\|_2,$$

where the upper bound follows from Cauchy-Schwarz and the lower bound is the classical Khintchine inequality. We use the following #P-hardness result, which we prove in the appendix for completeness.

LEMMA 4.4. *Given a vector $a \in \mathbb{R}^n$, the problem of computing $K(a)$ is #P-complete, even for $a \in \mathbb{Z}^n$ with bit-length polynomial in n .*

Combining this with the next lemma, the proof of Theorem 4.1 is immediate.

LEMMA 4.5. *The optimal revenue of a BIC and IIR mechanism for the public projects problem is $K(a)/2$.*

We present two proofs of Lemma 4.5. Our first proof invokes Myerson's characterization of optimal auctions [22]. Our second proof is self-contained and uses an argument from the analysis of Boolean functions [24] and will be presented in Section 5.

PROOF OF LEMMA 4.5 (FIRST VERSION). The standard virtual valuations for our setting (see Elkind and Myerson [14, 22]) are $\varphi_i(0) = -a_i$ and $\varphi_i(a_i) = a_i$ for each bidder i . In a binary single-parameter environment, the revenue-maximizing auction always selects the outcome that maximizes the sum of the virtual valuations of the chosen players—the “virtual welfare”—and the expected revenue of this auction equals its expected virtual welfare [14, 22]. Translated to the current special case, the solution to the OPTREV problem is precisely

$$\mathbf{E}_{v \sim F} \left[\max \left\{ 0, \sum_{i=1}^n \varphi_i(v_i) \right\} \right] = \mathbf{E}_{x \sim \{\pm 1\}^n} [\max\{0, x \cdot a\}], \quad (10)$$

where $x \in \{\pm 1\}^n$ is chosen uniformly at random and $a = (a_1, \dots, a_n)$.

Since $(-x) \cdot a = -(x \cdot a)$, we have

$$\mathbf{E}_{x \sim \{\pm 1\}^n} [\max\{0, x \cdot a\}] = \frac{1}{2} \mathbf{E}_{x \sim \{\pm 1\}^n} [|x \cdot a|] = \frac{K(a)}{2},$$

completing the proof. \square

4.3 Single-Minded Environments

This section presents an impossibility result for a downward-closed setting (where, unlike in public project environments, every subset of a feasible set is again feasible). Recall that in a single-minded environment, each bidder i wants a publicly known subset S_i of goods, and the feasible outcomes are subsets of bidders with mutually disjoint bundles.

In the special case where each bundle S_i has size 2, we can view each player's bundle as an edge in a graph with t vertices. Denote the resulting graph by H . We allow parallel edges (i.e., players that desire the same bundle). The prior distribution is uniform over $\{0, 1\}^P$. Given a string $x \in \{0, 1\}^P$, we define a subgraph H_x of H by keeping edge i if $x_i = 1$. Having $x_i = 0$ indicates that player i has 0 valuation; $x_i = 1$ means her valuation is 1 for her bundle. A feasible allocation corresponds to a matching in H_x , and the maximum welfare is the size of a maximum matching. Thus, the OPTWEL problem in this setting amounts to computing the expected size of the maximum matching in a random edge subgraph of H .

THEOREM 4.6. *The OPTWEL problem is #P-hard for the single-minded bidder setting even when every player's bundle contains two items, every player has only two possible valuations, and the valuation distribution is uniform.*

PROOF. The proof is by reduction from the s - t connectivity problem in directed graphs. Formally, #stConnectivity is the following problem: Given a directed graph H with two distinguished vertices s, t , what is the probability that there is an s - t path in a random edge subgraph of H ? Valiant shows this problem is #P-complete [31].

Assume that G has vertex set $[n] \cup \{s, t\}$ and m edges. We construct a bipartite graph H where the vertex set is $L \cup R$, where $L = \{s\} \cup [n]$ and $R = [n] \cup \{t\}$. When we speak of an edge (i, j) in H , we always mean with $i \in L$ and $j \in R$. If there is a directed edge (i, j) in G , we add an edge between i on the left and j on the right in H . We call these *red edges*. For every $i \in \{1, \dots, n\}$, we add k parallel edges between the two copies of i (k will be a large polynomial in m, n). We call these *blue edges*.

Consider picking a random subgraph H' of H . Except with probability $m/2^k$ (over the choice of blue edges), at least one blue edge of the form (i, i) survives for every $i \in [n]$. Conditioned on this event, the maximum matching in H' has size at least n . Furthermore, a matching of size $n + 1$ exists if and only if the subgraph of G that corresponds to the surviving red edges has an s - t path P —the red edges corresponding to P match $s \in L$, $t \in R$ and both copies of all vertices internal to P (vertices not on P are matched with blue edges).

Let p denote the probability that a random subgraph G' of G contains an s - t path. Let the random variable M denote the size of the maximum matching in H' . The preceding observations imply that

$$\begin{aligned} \mathbb{E}_{H'}[M] &\leq p(n + 1) + (1 - p)n = n + p \\ \mathbb{E}_{H'}[M] &\geq \left(p - \frac{m}{2^k}\right)(n + 1) + (1 - p)n = n + p - \frac{mn}{2^k}. \end{aligned}$$

Note that p is always an integer multiple of $1/2^m$. Therefore, if we choose k sufficiently large (bigger than mn), then we can recover p by subtracting n from $\mathbb{E}_{H'}[M]$ and rounding up the remainder to the form $c/2^m$ for some integer c . \square

Combining Corollary 3.6 and Theorem 4.6 gives the following.

COROLLARY 4.7. *Unless the polynomial hierarchy collapses, there is no generalized Border's theorem for the single-minded bidder setting even when every player's bundle contains two items, every player has only two possible valuations, and the valuation distribution is uniform.*

4.4 Multi-Item Auctions with Unit-Demand Bidders

Recall that, in a multi-item auctions setting, there is a set M of goods and each bidder i has a valuation v_{ij} for each item $j \in M$. The set $\mathcal{F} \subseteq 2^{U \times M}$ describes the feasible allocations of items to bidders. In a multi-item auction with unit-demand bidders, the feasible sets \mathcal{F} correspond to the matchings of a complete bipartite graph G with vertex sets U and M . Given valuations \mathbf{v} , the welfare-maximizing allocation corresponds to the maximum-weight matching in G (with each edge weight w_{ij} equal to the valuation v_{ij}). Since the reduction in the proof of Theorem 4.6 produces a bipartite graph, the reduction also implies that the OPTWEL problem is #P-hard for the setting of multi-item auctions with unit-demand bidders. Corollary 3.6 then implies the following.

COROLLARY 4.8. *Unless the polynomial hierarchy collapses, there is no generalized Border's theorem for the setting of multi-item auctions with unit-demand bidders.*

4.5 Matroid Environments

Our final example shows that there are even simple matroid settings which do not admit generalized Border's theorems (unless $\#P \subseteq P^{NP}$). In a *graphical matroid* environment, bidders correspond to the edges of an undirected graph $G = (V, E)$. The feasible sets correspond to the acyclic subgraphs of G , so the welfare-maximizing outcome corresponds to a maximum-weight spanning

forest. Using the same valuation distributions as in Section 4.3, the OPTWEL problem becomes that of computing the expected size of a spanning forest—or, equivalently, the expected number of connected components—of a random subgraph G' of G .

THEOREM 4.9. *The OPTWEL problem is #P-hard for the graphical matroid setting even when every player has only two possible valuations and the valuation distribution is uniform.*

PROOF. We reduce from the #stConnectivity problem in undirected graphs, which is also #P-complete [31]. Given an instance G of #stConnectivity, let C_1 denote the expected number of connected components in a random subgraph of G . Derive H from G by adding the edge (s, t) (a second copy if it is already in G) and let C_2 denote the expected number of connected components in a random subgraph of H . Since

$$C_1 - C_2 = \frac{1-p}{2},$$

where p is the probability that s and t are connected in a random subgraph of G , we conclude that the #stConnectivity problem reduces to the OPTWEL problem in the graphical matroid setting. \square

COROLLARY 4.10. *Unless the polynomial hierarchy collapses, there is no generalized Border's theorem for the graphical matroid setting even when every player has only two possible valuations and the valuation distribution is uniform.*

5 CONNECTIONS TO BOOLEAN FUNCTION ANALYSIS

In this section, we reinterpret our results on the public projects problem in terms of Boolean function analysis and, conversely, provide a stand-alone analysis of the optimal-revenue Boolean public project mechanism based on Boolean function analysis.

5.1 From Boolean Public Projects to Boolean Functions

Let us go back to the revenue maximization problem, OPTREV, for the Boolean public project problem addressed in Section 4.2 and rederive the characterization of the optimal revenue from first principles. Recall that each of our n players has either value 0 or value a_i for the “positive outcome,” with both possibilities equally likely and independent of the others’ values. To convert this setting to a Boolean functions setting, let us indicate by $x_i = 0$ the case that the value of player i is 0 and by $x_i = 1$ the case that the value of player i is a_i . Our prior distribution of values for the Boolean public project setting is now translated to the uniform distribution over the Boolean hypercube $\{0, 1\}^n$.

Let us denote by $f(x_1, \dots, x_n) \in [0, 1]$ the probability of a positive outcome that our mechanism gives when the players’ values are according to indicators x_i . Let us further denote by

$$f^i(x_i) = \mathbf{E}_{x_{-i} \in \{0, 1\}^{n-1}} [f(x_i, x_{-i})] = \mathbf{E}_{x \in \{0, 1\}^n} [f(x) | x_i].$$

the interim allocation of player i with value indicated by x_i . Now let us denote by $p^i(x_i)$ the interim payment of player i with value x_i . Our incentive constraints and individual rationality constraints imply bounds on p^i in terms of the f^i and allow us to characterize exactly the maximum payments that are possible for a given interim allocation rule.

LEMMA 5.1. *The maximum possible interim payments for a BIC and IIR Mechanism for the Boolean Public Projects problem with interim allocations given by f^i with $f^i(1) \geq f^i(0)$ are precisely $p^i(0) = 0$ and $p^i(1) = a_i \cdot (f^i(1) - f^i(0))$. In particular, the optimal revenue among such mechanisms is exactly $\sum_i (a_i/2) \cdot (f^i(1) - f^i(0))$.*

PROOF. The individual rationality constraint for $x_i = 0$ immediately implies $p^i(0) \leq 0$ since, in that case, the player gets no utility from “the bridge.” Now let us focus on the incentive constraint

for the case $x_i = 1$: Reporting the truth will result in the bridge being built with probability $f^i(1)$ while lying, and reporting 0 will give a probability $f^i(0)$. Player i 's value for telling the truth is thus $a_i \cdot (f^i(1) - f^i(0))$ larger than his value from lying, and this difference is the maximum that the payment $p^i(1)$ can be larger than the payment $p^i(0)$ without his utility becoming lower, which would violate the incentive constraints.

We now only need to observe that indeed setting $p^i(0) = 0$ and $p^i(1) = a_i \cdot (f^i(1) - f^i(0))$ does yield an incentive-compatible and individually rational mechanism. Individual rationality means that, for $x_0 = 0$, the value and payment and thus also utility are 0; for $x_i = 1$, the value is $a_i f^i(1)$ and the payment $p^i(1)$ is lower, and thus the utility is positive. Incentives mean that, for $x_i = 0$, lying would not increase player i 's value, which is 0, but may only increase his payment; for $x_i = 1$, the payment $p^i(1)$ was chosen exactly so his utility for the truth will exactly match his utility from lying, $a_i \cdot f^i(0)$. \square

The constraint that $f^i(1) \geq f^i(0)$ for each i must be satisfied by every incentive-compatible mechanism since otherwise the incentive constraints for $x_i = 0$ would dictate $p^i(1) \geq p^i(0)$ while the incentive constraints for $x_i = 1$ would dictate $p^i(1) < p^i(0)$, a contradiction. We may, however, without loss of generality, optimize over all functions f , since one can always switch the roles of 0 and 1 by flipping an input bit. We thus have reduced the revenue maximization problem to the following problem on functions $f : \{0, 1\}^n \rightarrow [0, 1]$.

LEMMA 5.2. *The optimal revenue of the Boolean public project problem is given by*

$$\text{OPT} = \max_{f: \{0,1\}^n \rightarrow [0,1]} \sum_{i=1}^n \frac{a_i}{2} (\mathbf{E}_{x \sim \{0,1\}^n} [f(x)|x_i = 1] - \mathbf{E}_{x \sim \{0,1\}^n} [f(x)|x_i = 0]). \quad (11)$$

It turns out that characterizing the function f that maximizes this weighted sum of differences is not difficult using the basic tools of Boolean function analysis (and, of course, yields the same mechanism as Myerson's theorem implies). The rest of this section will continue coming up with this derivation in a leisurely way within the Boolean function context. As expected, this analysis will identify the optimal f to be threshold function (halfspace) $f(x) = \text{sign}^+(\sum a_i \cdot (-1)^{x_i+1})$ where $\text{sign}^+(z) = 1$ if $z \geq 0$ and $\text{sign}^+(z) = 0$ otherwise. The fact that this is a Boolean function (rather than taking fractional values) translates to the optimal mechanism being deterministic and the fact that it is a monotone function can be translated to an optimal mechanism that is truthful in dominant strategies (obtained, as usual, by setting critical payments).

5.2 The Chow Parameters of a Boolean Function

We refer to functions $f : \{0, 1\}^n \rightarrow [0, 1]$ as bounded functions and the subset of functions $f : \{0, 1\}^n \rightarrow \{0, 1\}$ as Boolean functions. Every bounded function can be viewed as a convex combination of Boolean functions. Given $f : \{0, 1\}^n \rightarrow [0, 1]$, we define its *Chow parameters* or degree-0 and degree-1 Fourier coefficients as¹⁹

$$\begin{aligned} \hat{f}(0) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x)]; \\ \hat{f}(i) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x)(-1)^{1+x_i}] \text{ for } i \in [n]. \end{aligned}$$

We refer to $(\hat{f}(0), \dots, \hat{f}(n))$ as the *Chow vector* of f . Let us define the set

$$C_n = \{(c_0, \dots, c_n) | \exists f : \{0, 1\}^n \rightarrow [0, 1] \text{ s.t. } c_i = \hat{f}(i) \text{ for } 0 \leq i \leq n\}.$$

¹⁹The reason we use $(-1)^{1+x_i}$ instead of the more common $(-1)^{x_i}$ is that $\hat{f}(i)$ being positive implies positive correlation between x_i and f by Equation (13).

Note that the space of feasible Chow vectors is a polytope, since it is convex, and has finitely many vertices corresponding to the Chow vectors of Boolean functions. Let us denote this polytope by C_n . While this polytope is natural in the context of Fourier analysis, we are not aware of prior work that studies it explicitly.

Note that

$$\begin{aligned}\hat{f}(0) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x)] \\ &= \frac{1}{2} \mathbf{E}_{x \sim \{0,1\}^n} [f(x) | x_i = 1] + \frac{1}{2} \mathbf{E}_{x \sim \{0,1\}^n} [f(x) | x_i = 0].\end{aligned}\quad (12)$$

$$\begin{aligned}\hat{f}(i) &= \mathbf{E}_{x \sim \{0,1\}^n} [f(x) (-1)^{1+x_i}] \\ &= \frac{1}{2} \mathbf{E}_{x \sim \{0,1\}^n} [f(x) | x_i = 1] - \frac{1}{2} \mathbf{E}_{x \sim \{0,1\}^n} [f(x) | x_i = 0].\end{aligned}\quad (13)$$

This lets us reinterpret our results regarding revenue maximization for the public projects problem in the language of Chow parameters. By comparing Equation (11) in the statement of Lemma 5.2 and Equation (13), we see that the problem of maximizing revenue is equivalent to maximizing the weighted sum of the Chow parameters of a function.

$$\text{OPT} = \max_{f: \{0,1\}^n \rightarrow [0,1]} \sum_{i=1}^n a_i \hat{f}(i). \quad (14)$$

We briefly explain why these parameters are of interest in the analysis of Boolean functions. Recall that $\text{sign}^+(z) = 1$ if $z \geq 0$ and $\text{sign}^+(z) = 0$ otherwise. We say that a function $h: \{0,1\}^n \rightarrow \{0,1\}$ is a halfspace if there exist real numbers a_0, \dots, a_n such that

$$h(x) = \text{sign}^+ \left(a_0 + \sum_{i=1}^n a_i (-1)^{1+x_i} \right).$$

We may assume that $a_0 + \sum_{i=1}^n a_i (-1)^{1+x_i}$ never vanishes on $\{0,1\}^n$. An elegant result of Chow [9] implies that the Chow parameters of a halfspace identify it uniquely in the set of all Boolean functions and, in fact, all bounded functions.

THEOREM 5.3 (CHOW'S THEOREM [9]). *Let $h: \{0,1\}^n \rightarrow \{0,1\}$ be a halfspace. If $f: \{0,1\}^n \rightarrow [0,1]$ satisfies $\hat{f}(i) = \hat{h}(i)$ for $0 \leq i \leq n$, then $f(x) = h(x)$ for all $x \in \{0,1\}^n$.*

Chow's theorem is usually stated assuming that f is Boolean, but the proof [24, Theorem 5.1] also applies to the bounded case. Chow's argument does not give an algorithm to reconstruct a halfspace from its Chow parameters; this problem is known as the Chow parameters problem. It was solved recently by O'Donnell and Servedio [25] and subsequently improved in De et al. [11]. Both results start from approximations to the Chow parameters and return a halfspace that is close in Hamming distance to the target halfspace (with exactly the right Chow parameters).

Further motivation comes from the fact that, for monotone functions, $2\hat{f}(i)$ equals the influence of variable x_i , and hence $2 \sum_i \hat{f}(i)$ equals the average sensitivity of the function f . For n odd, let $\text{Maj}: \{0,1\}^n \rightarrow \{0,1\}$ denote the Majority function. It is known [24, Theorem 2.33] that for all Boolean functions,

$$\sum_{i=1}^n \hat{f}(i) \leq \sum_{i=1}^n \hat{\text{Maj}}(i) = \left(\sqrt{\frac{2}{\pi}} + o(1) \right) \sqrt{n}, \quad (15)$$

which implies that the Majority function has the highest average sensitivity among all monotone functions.²⁰ For more on Chow parameters and their significance, we refer the reader to O'Donnell [24, Chapter 5].

5.3 Optimization over C_n

At this point, we proceed with our analysis in the setting of Chow parameters which, on one hand, applies our hardness results to the problems of membership and optimization over the polytope C_n (Theorem 5.5) and, on the other hand, completes the promised stand-alone alternative proof of Lemma 4.5. By now, using Lemma 5.2 and the formulation in Equation (14), we have shown the equivalence of the revenue optimization problem for the Boolean public project setting to that of maximizing the weighted sum of the Chow parameters of a bounded function, equivalently maximizing a linear function over the polytope C_n .

Given $a = (a_0, \dots, a_n) \in \mathbb{R}^{n+1}$, it defines a linear objective function over C_n given by $a \cdot c$. To analyze the linear function corresponding to a vector $a \in \mathbb{R}^{n+1}$, we define the affine function $a(x) = a_0 + \sum_{i=1}^n a_i (-1)^{1+x_i}$ mapping $\{0, 1\}^n$ to \mathbb{R} . When $a_0 = 0$, the first part of Lemma 5.4 is essentially a restatement of Lemma 4.5, whereas the second part replaces the role of Myerson's theorem for identifying the optimal function in Section 4.5.

LEMMA 5.4. *Given $a \in \mathbb{R}^{n+1}$, we have*

$$\max_{c \in C_n} a \cdot c = \mathbb{E}_{x \sim \{0,1\}^n} [\text{sign}^+(a(x))a(x)] = \frac{1}{2}(K(a) + a_0). \quad (16)$$

Equality is attained at the Chow vectors corresponding to functions $f : \{0, 1\}^n \rightarrow [0, 1]$ such that $f(x) = \text{sign}^+(a(x))$ whenever $a(x) \neq 0$.

PROOF. Let $c = (\hat{f}(0), \dots, \hat{f}(n))$. We have

$$\begin{aligned} a \cdot c &= a_0 \hat{f}(0) + \sum_{i=1}^n a_i \hat{f}(i) \\ &= a_0 \mathbb{E}_{x \sim \{0,1\}^n} [f(x)] + \sum_{i=1}^n a_i \mathbb{E}_{x \sim \{0,1\}^n} [f(x)(-1)^{1+x_i}] \\ &= \mathbb{E}_{x \sim \{0,1\}^n} \left[f(x) \left(a_0 + \sum_{i=1}^n a_i x_i \right) \right] \\ &= \mathbb{E}_{x \sim \{0,1\}^n} [f(x)a(x)]. \end{aligned}$$

Since $0 \leq f(x) \leq 1$,

$$f(x)a(x) \leq \text{sign}^+(a(x))a(x).$$

If $a(x) \neq 0$, this holds with equality if and only if $f(x) = \text{sign}^+(a(x))$, whereas if $a(x) = 0$, $f(x)$ can take an arbitrary value in $[0, 1]$. This gives us

$$\max_{c \in C_n} a \cdot c = \mathbb{E}_{x \sim \{0,1\}^n} [\text{sign}^+(a(x))a(x)]$$

and characterizes the functions that achieve equality. To complete the proof, we just compute this expectation. This is a routine calculation which we defer to Appendix B. \square

THEOREM 5.5. *The problems of linear optimization over C_n and deciding membership in C_n are #P-hard.*

²⁰Going back to the Boolean public projects problem, this would be an estimate of the optimal revenue for the case where all $a_i = 1$.

PROOF. The hardness of linear optimization follows from Lemmas 5.4 and 4.4: If we can solve linear optimization efficiently, we can compute $K(a)$.

The hardness of membership is proved using a similar argument to Theorem 3.3. 0^{n+1} is a feasible point in C_n . Hence, the ellipsoid method can also be used to reduce linear optimization to a polynomial number of invocations of a membership oracle (see Schrijver [30, P. 189]). Hence, if we can decide membership in the polytope C_n , then we can solve linear optimization over C_n using the Ellipsoid algorithm, which we just showed is #P-hard. \square

This hardness result rules out a *nice* characterization of the polytope C_n , in the spirit of Definition 3.1. We believe this negative result is interesting in the context of the Chow parameters problem and sheds light on why the exact version of the problem (where the goal is to find a function whose Chow vector equals the input) is hard.

O'Donnell and Servedio [25] observe that the inverse problem of computing $\hat{f}(0)$ for a given halfspace is #P-complete, which implies that, given a target Chow vector, it is hard to verify if an input halfspace has exactly these Chow parameters. This can be viewed as evidence that the exact version of the Chow parameters problem is intractable.

Assume that we drop the requirement that the output be a halfspace and are willing to settle for any bounded Boolean function that has some compact representation, which lets us evaluate its Chow vector exactly. Can we now hope to solve the Chow problem exactly? Theorem 5.5 says this is unlikely since such an algorithm will allow us to test membership in C_n , run the algorithm, and check the function output by it.

Let us return to the question that we started this article with: For which vectors (p_0, p_1, \dots, p_n) does there exist a probability space with events E, X_1, \dots, X_n , with X_1, \dots, X_n independent and $\Pr[X_i] = 1/2$ for all $i \in [n]$, such that $p_0 = \Pr[E]$ and $p_i = \Pr[E|X_i]$ for all $i \in [n]$? Let x_i be the indicator of the event X_i . Define $f : \{0, 1\}^n \rightarrow [0, 1]$ by setting $f(x)$ to be the probability of E given x . This problem reduces to testing whether a vector of Chow parameters is feasible, which is #P-hard by Theorem 5.5.

APPENDIXES

A HARDNESS OF COMPUTING THE KHINTCHINE CONSTANT

PROOF OF LEMMA 4.4. Partition is a well-known NP-complete problem whose input consists of n integers w_1, \dots, w_n and the goal is to split the numbers into two parts so that their sum is equal. This is equivalent to asking if there exists $x \in \{\pm 1\}^n$ such that $w \cdot x = 0$ where $w = (w_1, \dots, w_n)$. The counting version, which we denote #Partition, is the problem of computing

$$\Pr_{x \in \{\pm 1\}^n} [w \cdot x = 0].$$

It is complete for #P. We show that this problem reduces to computing the Khintchine constant.

Given w , define the vectors

$$a_0 = (2w_1, \dots, 2w_n, 0), \quad a_1 = (2w_1, \dots, 2w_n, 1).$$

For $x \in \{\pm 1\}^n$, define $x^+ = (x, 1)$ and $x^- = (x, -1)$. We observe that

$$|a_0 \cdot x^+| + |a_0 \cdot x^-| = 4|w \cdot x|; \tag{17}$$

$$|a_1 \cdot x^+| + |a_1 \cdot x^-| = |2w \cdot x + 1| + |2w \cdot x - 1|. \tag{18}$$

When $w \cdot x = 0$, the right-hand side equals of Equation (17) equals 0, while that of Equation (18) equals 2.

If $w \cdot x \neq 0$, then $|2w \cdot x| \geq 2$. We then use

$$|a + b| + |a - b| = 2 \max(|a|, |b|)$$

to conclude that

$$|a_1 \cdot x^+| + |a_1 \cdot x^-| = 4|w \cdot x| = |a_0 \cdot x^+| + |a_0 \cdot x^-|.$$

Finally, observe that we can write

$$K(a_i) = \mathbf{E}_{x \sim \{\pm 1\}^n} \left[\frac{|a_i \cdot x^+| + |a_i \cdot x^-|}{2} \right].$$

It follows that

$$\Pr_{x \in \{\pm 1\}^n} [w \cdot x = 0] = K(a_1) - K(a_0). \quad \square$$

B COMPLETING THE PROOF OF LEMMA 5.4

Consider the affine function $a(x) = a_0 + \sum_{i=1}^n a_i(-1)^{1+x_i}$ mapping $\{0, 1\}^n$ to \mathbb{R} . Our goal is to show that $\mathbf{E}_{x \sim \{0, 1\}^n} [\text{sign}^+(a(x))a(x)] = \frac{1}{2}(K(a) + a_0)$. Observe that

$$(2\text{sign}^+(a(x)) - 1)a(x) = |a(x)|.$$

Hence,

$$\begin{aligned} \mathbf{E}_{x \in \{0, 1\}^n} [|a(x)|] &= 2\mathbf{E}_{x \in \{0, 1\}^n} [\text{sign}^+(a(x))a(x)] - \mathbf{E}_{x \in \{0, 1\}^n} [a(x)] \\ &= 2\mathbf{E}_{x \in \{0, 1\}^n} [\text{sign}^+(a(x))a(x)] - a_0. \end{aligned}$$

We will now show that

$$\mathbf{E}_{x \in \{0, 1\}^n} [|a(x)|] = K(a),$$

which implies the claim. Observe that

$$\begin{aligned} K(a) &= \mathbf{E}_{y \in \{\pm 1\}^{n+1}} [|a \cdot y|] \\ &= \frac{1}{2} \mathbf{E}_{y \in \{\pm 1\}^n} \left[\left| a_0 + \sum_{i=1}^n a_i y_i \right| \right] + \frac{1}{2} \mathbf{E}_{x \in \{\pm 1\}^n} \left[\left| -a_0 + \sum_{i=1}^n a_i y_i \right| \right]. \end{aligned}$$

We claim that the two expectations on the right-hand side are in fact equal to each other, since

$$\mathbf{E}_{y \in \{\pm 1\}^n} \left[\left| -a_0 + \sum_{i=1}^n a_i y_i \right| \right] = \mathbf{E}_{y \in \{\pm 1\}^n} \left[\left| a_0 - \sum_{i=1}^n a_i y_i \right| \right] = \mathbf{E}_{y \in \{\pm 1\}^n} \left[\left| a_0 + \sum_{i=1}^n a_i y_i \right| \right],$$

using that $\sum_i a_i y_i$ is an even random variable. Hence, we get

$$\begin{aligned} K(a) &= \mathbf{E}_{y \in \{\pm 1\}^n} \left[\left| a_0 + \sum_{i=1}^n a_i y_i \right| \right] \\ &= \mathbf{E}_{x \in \{0, 1\}^n} \left[\left| a_0 + \sum_{i=1}^n a_i (-1)^{1+x_i} \right| \right] \\ &= \mathbf{E}_{x \in \{0, 1\}^n} [|a(x)|]. \end{aligned}$$

ACKNOWLEDGMENTS

We would like to thank Noga Alon, Yang Cai, Costis Daskalakis, Uri Feige, Gil Kalai, Nati Linial, Yuval Peres, Rocco Servedio, Li-Yang Tan, and Matt Weinberg for helpful discussions during various stages of this work. We thank Ilias Diakonikolas for useful pointers to the literature on Chow parameters.

REFERENCES

- [1] Saeed Alaei, Hu Fu, Nima Haghpahan, Jason D. Hartline, and Azarakhsh Malekian. 2012. Bayesian optimal auctions via multi- to single-agent reduction. In *Proceedings of the ACM Conference on Electronic Commerce (EC'12)*. 17.
- [2] K. C. Border. 1991. Implementation of reduced form auctions: A geometric approach. *Econometrica* 59, 4 (1991), 1175–1187.
- [3] K. C. Border. 2007. Reduced form auctions revisited. *Economic Theory* 31 (2007), 167–181.
- [4] Patrick Briest. 2008. Uniform budgets and the envy-free pricing problem. In *Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP)*. 808–819.
- [5] Yang Cai, Constantinos Daskalakis, and S. Matthew Weinberg. 2012. An algorithmic characterization of multi-dimensional mechanisms. In *Proceedings of the 44th Symposium on Theory of Computing Conference (STOC)*. 459–478.
- [6] Yang Cai, Constantinos Daskalakis, and S. Matthew Weinberg. 2012. Optimal multi-dimensional mechanism design: Reducing revenue to welfare maximization. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science (FOCS)*. 130–139.
- [7] Y.-K. Che, J. Kim, and K. Mierendorff. 2013. Generalized reduced form auctions: A network flow approach. *Econometrica* 81 (2013), 2487–2520.
- [8] Xi Chen, Ilias Diakonikolas, Dimitris Paparas, Xiaorui Sun, and Mihalis Yannakakis. 2014. The complexity of optimal multidimensional pricing. In *Proceedings of 25th Annual ACM-SIAM Symposium on Discrete Algorithms*. 1319–1328.
- [9] C. Chow. 1961. On the characterization of threshold functions. In *Proceedings of the Symposium on Switching Circuit Theory and Logical Design (FOCS)*. 34–38.
- [10] Constantinos Daskalakis, Alan Deckelbaum, and Christos Tzamos. 2014. The complexity of optimal mechanism design. In *Proceedings of 25th Annual ACM-SIAM Symposium on Discrete Algorithms*. 1302–1318.
- [11] Anindya De, Ilias Diakonikolas, Vitaly Feldman, and Rocco A. Servedio. 2014. Nearly optimal solutions for the Chow parameters problem and low-weight approximation of halfspaces. *Journal of the ACM* 61, 2 (2014), 11. DOI : <https://doi.org/10.1145/2590772>
- [12] Anindya De, Ilias Diakonikolas, and Rocco A. Servedio. 2013. A robust khintchine inequality, and algorithms for computing optimal constants in fourier analysis and high-dimensional geometry. In *Proceedings of the 40th International Colloquium on Automata, Languages, and Programming (ICALP'13), Part I*. Riga, Latvia. 376–387.
- [13] Shahar Dobzinski, Hu Fu, and Robert D. Kleinberg. 2011. Optimal auctions with correlated bidders are easy. In *Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC)*, Lance Fortnow and Salil P. Vadhan (Eds.). 129–138.
- [14] Edith Elkind. 2007. Designing and learning optimal finite support auctions. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 736–745.
- [15] Alex Gershkov, Jacob K Goeree, Alexey Kushnir, Benny Moldovanu, and Xianwen Shi. 2013. On the equivalence of Bayesian and dominant strategy implementation. *Econometrica* 81, 1 (2013), 197–220.
- [16] M. Grötschel, L. Lovász, and A. Schrijver. 1988. *Geometric Algorithms and Combinatorial Optimization*, 2nd edition. Springer.
- [17] S. Hart and P. J. Reny. 2014. Implementation of reduced form mechanisms: A simple approach and a new characterization. *Economic Theory Bulletin* (2014).
- [18] L. G. Khachiyan. 1979. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady* 20, 1 (1979), 191–194.
- [19] E. Maskin and J. Riley. 1984. Optimal auctions with risk-adverse buyers. *Econometrica* 52 (1984), 1473–1518.
- [20] S. A. Matthews. 1984. On the implementability of reduced form auctions. *Econometrica* 52 (1984), 1519–1522.
- [21] K. Mierendorff. 2011. Asymmetric reduced form auctions. *Economics Letters* 110 (2011), 41–44.
- [22] R. Myerson. 1981. Optimal auction design. *Mathematics of Operations Research* 6, 1 (1981), 58–73.
- [23] Noam Nisan. 2007. Introduction to mechanism design (for computer scientists). In *Algorithmic Game Theory*, Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani (Eds.). Cambridge University Press.
- [24] Ryan O'Donnell. 2014. *Analysis of Boolean Functions*. Cambridge University Press.
- [25] Ryan O'Donnell and Rocco A. Servedio. 2011. The Chow parameters problem. *SIAM Journal of Computing* 40, 1 (2011), 165–199. DOI : <https://doi.org/10.1137/090756466>
- [26] Christos H. Papadimitriou and George Pierrakos. 2011. On optimal single-item auctions. In *Proceedings of the 43rd ACM Symposium on Theory of Computing*. 119–128.
- [27] Itamar Pitowsky. 1994. George Boole's "conditions of possible experience" and the quantum puzzle. *The British Journal for the Philosophy of Science* 45, 1 (1994), 95–125.
- [28] Tim Roughgarden. 2014. Barriers to near-optimal equilibria. In *Proceedings of the 55th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 71–80. DOI : <https://doi.org/10.1109/FOCS.2014.16>
- [29] Tim Roughgarden and Inbal Talgam-Cohen. 2015. Why prices need algorithms. In *Proceedings of the 16th ACM Conference on Economics and Computation (EC)*. ACM Press, 19–36. DOI : <https://doi.org/10.1145/2764468.2764515>

- [30] A. Schrijver. 1986. *Theory of Linear and Integer Programming*. Wiley.
- [31] Leslie G. Valiant. 1979. The complexity of enumeration and reliability problems. *SIAM Journal of Computing* 8, 3 (1979), 410–421. DOI: <https://doi.org/10.1137/0208032>
- [32] G. M. Ziegler. 1993. *Lectures on Polytopes*. Springer.

Received February 2016; revised August 2017; accepted November 2017