A REVEALED PREFERENCE APPROACH TO COMPUTATIONAL COMPLEXITY IN ECONOMICS

FEDERICO ECHENIQUE, DANIEL GOLOVIN, AND ADAM WIERMAN

ABSTRACT. Consumption theory assumes that consumers posses infinite computational abilities. Proponents of bounded rationality want to instead require that any model of consumer behavior incorporate computational constraints. In this paper, we establish that this requirement has no testable implications. Any consumption data set that is compatible with a rational consumer is also compatible with a rational and computationally bounded consumer (such a data set is rationalizable by a utility function that is easy to maximize over any budget set; specifically with a utility that can be maximized in strongly polynomial time). The result extends to data on multiple agents interacting in a market economy. We present sufficient conditions on observed market outcomes such that they are compatible with an instance of the model for which Walrasian equilibrium is easy to compute. Our result motivates a general approach for posing questions about the empirical content of computational constraints: *the revealed preference approach to computational complexity.* The approach complements the conventional worst-case view of computational complexity in important ways, and is methodologically close to mainstream economics.

Computational Complexity, Rationalization, Revealed Preference, Theory of the Consumer

1. INTRODUCTION

This paper presents a new perspective on the computational complexity of economic models. Many economic models can be criticised because they assume that agents or economies solve computationally hard problems. Perhaps the oldest, and most transparent, example of the critique is the idea of *bounded rationality*: economists assume that agents maximize utility, but

Key words and phrases. Computational Complexity, Strong Axiom of Revealed Preference, Revealed Preference, Theory of the consumer .

Echenique is affiliated with the Division of Humanities and Social Sciences; Golovin is affiliated with the Center for the Mathematics of Information at Caltech; Wierman is affiliated with Computers and Mathematical Sciences at Caltech. emails: {fede,dgolovin,adamw}@caltech.edu. This is the journal version of the paper presented at the ACM Electronic Commerce conference in 2011. The authors thank Vince Conitzer and Laurent Mathevet for detailed comments on an earlier draft.

utility maximization is a hard problem to solve. Therefore utility maximization is flawed as a prediction because it makes unrealistic assumptions on the computational ability of economic agents; the critique was formalized by Herbert Simon [26, e.g]; [16] is a recent exposition.

Our response to the critique is that it takes the theory too literally. Economists do not mean that agents actually maximize utility. Utility is not an observable quantity, so it has no positive empirical meaning to say that it is maximized. Instead, the content of the theory is to require that *observed* economic outcomes be consistent with utility maximizing behavior. We do not claim that computational constraints are irrelevant, but want to put them in a different light.

Our perspective on computational complexity is strongly motivated by the revealed preference literature in economics [24]. This literature seeks to understand the empirical implications of economic models: given data from an observed phenomenon, the task is to understand how generally a model is applicable (i.e., how large the class of explainable data is) and to determine what instance of the theory is revealed by the data (e.g., to determine the properties of the utility functions that are consistent with the data). Examples of such results include [1, 2, 27, 28, 21, 5, 4, 11, 6, 7]. Hal Varian [30] has an excellent survey of revealed preference.

Revealed preference theory, however, traditionally disregards computational issues (as do most economists, in general). Our proposed perspective on complexity changes this by adding to revealed preference theory the constraint that the instance revealed does not require agents to solve any computationally hard problems. Thus, the question becomes, does the addition of computational constraints limit the class of data explainable by the theory? The goal of this paper is to formalize this "revealed preference view" of computational complexity, and to begin to explore whether it leads to different conclusions than the standard "worst-case view."

This paper deals mostly with bounded rationality and the theory of the consumer. We consider data on consumption purchases, as is standard in revealed preference theory (see e.g. [27]). We show that such data is rationalizable if and only if it is rationalizable using a utility function that is tractable: easy to maximize.[1] Thus, the critique of bounded rationality has no empirical bite for consumption theory; it has no testable implications over and beyond the implications of rationality alone.

The meaning of our finding is that, while computational constraints are a priori an important factor that may limit the applicability of an economic

[1]The conference version of this paper, [12] establishes the result for either divisible or indivisible goods; the arguments used in each case is different. Here we extend the result to the mixed case, in which some goods are divisible and some are indivisible, using an approach that unifies the two previous arguments.

theory; they turn out to be *empirically* irrelevant, at least for the theory of the consumer. One can interpret this as a negative finding that establishes that computational constraints in consumption theory are not testable, or one can conclude that consumer theory is safe from the critique that it assumes unrealistic computational powers.

To be clear, utility maximization under computational constraints has empirical implications because utility maximization imposes empirical constraints. There are non-rationalizable data sets; data sets that violate the axioms of revealed preference. The point of our paper is that utility maximization under computational constraints has exactly the same empirical implication as does utility maximization in general.

In addition to results on the pure theory of consumption, this paper presents some preliminary results on Walrasian equilibria, on both the problems faced by the consumer and on the hardness of finding an equilibrium.

First, with respect to the consumer problem, it is easy to show that the results on the hardness of computing an optimum for the consumers extend to situations where the consumer is embedded in an economy. Given data on exchange economies, if the data is rationalizable using the notion of Walrasian equilibrium, then it is also rationalizable when consumers are required to solve easy problems.

Second, with respect to the problem of finding a Walrasian equilibrium, we present conditions on the data under which there is a rationalization that makes it easy to find a Walrasian equilibrium. These are conditions that ensure that there is a representative consumer. We argue (in an extension of some recent results on Fisher equilibria, see [18, 9]) that when there is a representative consumer of this kind, then it is easy to find Walrasian equilibria.

Note that our results for Walrasian equilibria only give sufficient conditions under which the data have a rationalization making it easy to find Walrasian equilibria. We do not know if that is always the case. For consumption theory, we have established that any rationalizable dataset is also rationalizable by an instance that is easy to compute, but we do not know if that is the case for Walrasian equilibria.

Finally, the paper presents a general formulation of our "revealed preference view" of computational complexity. We present a general framework in which one can talk of a theory being testable and easily computable, and where the testable implications of imposing computational constraints can be studied.

2. CONSUMER CHOICE THEORY

The revealed preference approach we propose is widely applicable, but consumer choice theory is a natural starting point since the theory is a basic building block for most of economics. It is the first topic covered in every introductory course in economics, and it remains a very active research field until this day: active for both empirical and theoretical research. Further, the revealed preference approach in economics was first proposed and developed for consumer choice theory.

Before introducing the model formally, we introduce some notation that will be used throughout the remainder of the paper. For integer n, we let $[n] := \{1, 2, \ldots, n\}$. Let $\mathbb{Z}_+$ and $\mathbb{R}_+$ denote the nonnegative integers and real numbers, respectively, and let $\mathbb{Z}_{++}$ and $\mathbb{R}_{++}$ and denote the set of positive integers and real numbers, respectively. For a binary relation R, we let R^* denote its transitive closure. A binary relation R is ***acyclic*** if $x\ R^*\ y$ and $y\ R^*\ x$ implies that $x = y$. All logarithms are base 2 unless otherwise noted. We denote a d-dimensional vector with all coordinates equal to c by $\mathbf{c}$ and denote the i^{th} basis vector, which is one at coordinate i and zero at the other $d-1$ coordinates, by $\hat{e}_i$.

We can now define the consumer choice problem as follows. We consider a single consumer and d different goods. A ***consumption bundle*** is a vector $x \in \mathcal{X}$, where $\mathcal{X}$ is the ***consumption space***. We consider three settings for the consumption space: (i) infinitely divisible goods, (ii) indivisible goods, and (iii) a mixed setting, where some goods are divisible and some are indivisible. In the case of infinitely divisible goods, each of the goods is available in perfectly divisible quantities, and so $\mathcal{X} = \mathbb{R}^d_+$. In the case of indivisible goods, the goods can not be divided, and thus $\mathcal{X} = \mathbb{Z}^d_+$. Strictly speaking, probably most goods are indivisible; but economists often model goods such as houses and cars as indivisible, and frequently-purchased goods (such as food items) as divisible. The third case is when some goods are divisible and some are indivisible: We shall simply allow $\mathcal{X}$ to be an arbitrary subset of $\mathbb{R}^d_+$. Note that as special cases one obtains the infinitely divisible and the divisible settings.

In each of the three cases, we denote by $\leq$ the natural partial order on $\mathcal{X}$: so $x \leq y$ iff $x_i \leq y_i$ for each coordinate $i \in [d]$. When $\mathcal{X} = \mathbb{R}^d_+$ or $\mathcal{X} = \mathbb{Z}^d_+$, $\leq$ generates a lattice $(\mathcal{X}, \leq)$.

An instance of the consumer choice problem is specified by a price vector $p \in \mathbb{R}^d_{++}$ and a consumer income $b \in \mathbb{R}_+$. Let $B(p, b) := \{x : x \in \mathcal{X},\ p \cdot x \leq b\}$ denote the budget set under prices p and budget b.

The standard economic theory of the consumer then postulates that the consumer behaves (i.e. chooses x) as if he solves the problem

$$\max \{u(x) : x \in B(p, b)\}\,. \tag{1}$$

To incorporate computational considerations, we make the distinction between *tractable* and *intractable* utility functions, using the standard demarcation of polynomial-time computability.

Definition 1 (Tractable Utility Functions)**.** *A utility function $u : \mathcal{X} \to \mathbb{R}$ is* tractable *if and only if problem (1) can be solved in polynomial time*[2] *for all $p \in \mathbb{R}^d_{++}$ and $b \in \mathbb{R}_+$. Otherwise, u is said to be* intractable.

2.1. **Infinitely divisible goods.** We first discuss the case of infinitely divisible goods, which serves as a simple motivational example for the idea of a "revealed preference view" of computational complexity.

In the case of divisible goods, the consumption space is $\mathcal{X} = \mathbb{R}^d_+$. Thus, the standard economic theory of the consumer postulates that the consumer behaves (i.e. chooses x) as if solving the problem

$$\max \left\{ u(x) : x \in \mathbb{R}^d_+ \text{ and } p \cdot x \leq b \right\}. \tag{2}$$

We proceed to discuss the worst-case and the revealed preference view applied to this problem.

A worst-case view. From a worst-case perspective, it is immediately clear that if the utility function is allowed to be general, the consumer could be required to solve an arbitrary non-convex optimization, which is computationally intractable. (For completeness, we prove this rather basic fact for monotone utilities in Lemma 19 in Appendix B. A similar result is in [16].) Thus, without additional assumptions on the allowable utility functions, a worst-case view of computational complexity leads to a critique of the model as unrealistically assuming that consumers are solving a problem that is not computationally tractable. Taking a revealed preference view of computational complexity will lead to a different conclusion.

A revealed preference view. At its core, our revealed preference view is an empirical approach to computational complexity, thus it starts from data about observed behavior. Specifically, suppose that we have finitely many observations of the purchases made by a consumer at prices p^i and incomes b^i, $i = 0, \ldots, n$. That is, we have ***data*** of the form $\{(x^i, p^i, b^i)\}_{i=0}^n$.

We say that the data is ***rationalizable*** if there is a monotone increasing function $u(\cdot)$ such that x^i is a solution to (2) at prices p^i and income b^i. Then, revealed preference theory provides the following well-known result [1, 27],

[2] For continuous optimization problems, we take the notion of "solving in polynomial time" to mean obtaining an additive ϵ-approximation in time polynomial in the natural problem parameters and $\log(1/\epsilon)$. To be completely formal, we may also specify a precise model of computation, e.g., a RAM with word size $\Theta(\log n)$. However we feel that this level of formalism adds little in precision to our results and detracts from the main discussion, and so we avoid it in the remainder of the paper.

which says that the data results either from a consumer that is manifestly irrational or from a consumer whose behavior can be explained using a concave utility function:

Theorem 2. ***(Afriat's Theorem)*** *In the consumer choice problem with infinitely divisible goods, the data $\{(x^i, p^i, b^i)\}_{i=0}^n$ is rationalizable if and only if it is rationalizable by a monotone increasing and concave utility function.*

Though Afriat's Theorem does not explicitly address computational constraints, there exist computationally efficient approaches to solving continuous monotone concave maximization problems with convex feasible sets [3]. Thus, an immediate consequence of Afriat's Theorem is that a data set $\{(x^i, p^i, b^i)\}_{i=0}^n$ is either manifestly irrational or can be explained using a utility function for which it is computationally efficient to solve (2). So, in the case where goods are infinitely divisible, the constraint that consumers can solve (2) in a computationally efficient manner has no empirical consequences, since it does not impose any additional restriction on the applicability of the model beyond the restrictions imposed by the assumption of rationality.

Relating the worst-case and revealed preference views. The simple motivational example of infinitely divisible goods provides an illustration of the contrast between the worst-case and revealed preference views. This example highlights that the two views can lead to fundamentally different conclusions about the role of computation. However, observe that if an economic model is polynomial time computable in a worst-case sense, then it is immediate to see that computational restrictions will have no empirical consequences in the revealed preference view. So, the revealed preference view is weaker than the worst-case view.

The revealed preference view is weaker because the power of an adversary to choose a hard instance is limited. Instead of being able to specify a hard instance precisely (e.g., specify a hard utility function for the consumer), the instance must be specified via a data set. Since this interface is not precise, even if hard instances exist, they may not have empirical consequences in the revealed preference view if a data set specified by a hard instance can always be explained by an easy instance (e.g., all data sets are rationalizable via a monotone, concave utility function).

2.2. **Indivisible goods.** We now move to the computationally more interesting case of indivisible goods, i.e., $\mathcal{X} = \mathbb{Z}_+^d$. It is a realistic case, as many consumption goods are clearly indivisible. The standard economic theory of the consumer postulates that the consumer behaves (i.e., chooses x) as if solving the problem

$$\max \left\{ u(x) : x \in \mathbb{Z}_+^d \text{ and } p \cdot x \leq b \right\} \tag{3}$$

The analysis in this setting is significantly more involved than that for infinitely divisible goods. The main result, however, will be the same: that the consumer problem is computationally hard in a worst-case sense, but that computational constraints have no empirical consequences in the revealed preference view.

A worst-case view. As in the case of infinitely divisible goods, it is immediately clear that if the utility function is allowed to be general, the consumer could be required to solve an arbitrary discrete optimization, which is computationally intractable. In fact, optimization over a discrete space is intractable for seemingly simple utility functions: we show in Lemma 19 that solving (3) is NP-hard, even with *linear* utility functions. Thus, without additional assumptions on the allowable utility functions, a worst-case view of computational complexity leads to a critique of the model as unrealistically assuming that consumers are solving a problem that is not computationally tractable.

A revealed preference view. We now switch to a revealed preference view of computational complexity. As in the case of infinitely divisible goods, it provides a very different perspective than the worst-case view.

To define the revealed preference view, we must first define the form of the observed data about consumer behavior. We model the data (input) as a sequence of some number n of consumer choices $\{(x^i, p^i, b^i)\}_{i=0}^n$ where $x^i \in \mathbb{Z}_+^d$ is the consumer choice observed under prices p^i and budget b^i (i.e., under budget set $B^i := \{x : p^i \cdot x \leq b^i\}$) for each $i \geq 1$. We assume that if $x^i < y$ then $y \notin B^i$; so x^i is maximal in B^i. For technical reasons, we add a fictitious observation (x^0, p^0, b^0) with $x^0 = \mathbf{0}$, $p^0 = \mathbf{1}$, $b^0 = 0$; that is, a consumer with no money chooses the only feasible choice, namely nothing at all.

Adopting the language of revealed preference theory, we let $X = \{x^0, x^1, \ldots, x^n\}$ be the set of bundles purchased at any time in the input and observe that the data induces a revealed preference relation

$$R_0 := \left\{(x^i, x^j) : \; (x^i, x^j) \in X^2, \; x^i \in B^j\right\}.$$

As in the case of infinitely divisible goods, we say that the data $\{(x^i, p^i, b^i)\}_{i=0}^n$ is ***rationalizable*** if there is a monotone increasing function $u(\cdot)$ such that x^i is the unique solution to $\max\{u(x) : x \in B(p^i, b^i)\}$, for $i = 0, \ldots, n$.

Now, as in the case of indivisible goods, revealed preference theory provides a characterization of the set of data that is rationalizable. Specifically, the following well-known result (see, for example, [22] or [23]) highlights the relationship between rationalizability and the preference relation R_0:

Theorem 3. *In the consumer choice problem with indivisible goods, data* $\{(x^i, p^i, b^i)\}_{i=0}^n$ *is rationalizable if and only if the binary relation* R_0 *is acyclic.*

There are results analogous to Theorem 2 for infinitely divisible goods. The following result is due to [6] (and extended by [25]). Of course, concave utility makes no sense in the indivisible case, but supermodularity and submodularity are useful properties of utility one can potentially exploit. In fact, the use of super- and submodularity is promising because there are efficient algorithms for optimizing a function with these properties.

Theorem 4. *In the consumer choice problem with indivisible goods, the following statements are equivalent:*

- *The data* $\{(x^i, p^i, b^i)\}_{i=0}^n$ *is rationalizable.*
- *The data is rationalizable by a supermodular utility function.*
- *The data is rationalizable by a submodular utility function.*

At this point, however, the parallel with the case of infinitely divisible good ends. Unlike in the case of infinitely divisible goods, we can not immediately apply Theorem 4 to arrive at utility functions for which it is computationally efficient for consumers to solve (3). The reason is that solving (3) for either submodular or supermodular utility functions is not computationally efficient. Specifically, while there are efficient algorithms for optimizing a submodular function over a power set lattice, submodular maximization subject to a budget constraint is NP-hard. In fact, it is even hard to approximate to better than a $(1 - 1/e)$ factor [13]. Further, supermodular maximization subject to a budget constraint is likewise NP-hard [19], and extremely inapproximable under reasonable complexity assumptions, as we show in Theorem 20.

The main result of this paper is that, when data is rationalizable, it is possible to rationalize the data using a utility function for which solving (3) is computationally efficient. Specifically:

Theorem 5. *In the consumer choice problem with indivisible goods, data* $\{(x^i, p^i, b^i)\}_{i=0}^n$ *is rationalizable if and only if it is rationalizable via a tractable monotone utility function.*

The message in Theorem 5 is that computational constraints do not impose any further restrictions on the ability of consumer choice theory to explain data than are already imposed by the assumption of rationality. This is in stark contrast to the message suggested by the worst-case view of computational complexity for this model.

In order to prove Theorem 5, we must prove that the data is either not rationalizable, or there is a utility function rationalizing the data such that for any p and b, (3) can be solved in strongly polynomial time. That is, using only computation time polynomial in n, d, and the number of bits required to write down the prices and budgets.

We cannot use Theorem 4 to prove Theorem 5. Instead, we proceed by working with the demand function instead of the utility function. Specifically, to prove the result we construct an efficient algorithm $\mathcal{A} = \mathcal{A}_{\{(x^i,p^i,b^i)\}_{i=0}^n}$ to compute a ***demand function***[3] $\psi : \mathbb{R}^d_{++} \times \mathbb{R}_+ \to \mathbb{Z}^d_+$ such that

(i) the demand function explains the data, i.e., $\psi(p^i, b^i) = x^i$ for all $i \in [n]$, and
(ii) there is a monotone increasing utility function $u(\cdot)$ rationalizing the data, and for all $p \in \mathbb{R}^d_{++}$ and $b \in \mathbb{R}_+$ we have

$$\psi(p, b) = \arg\max \{u(x) \ : \ x \in B(p, b)\}.^4$$

We prove this fact as Corollary 18 in Appendix A. Interestingly, the algorithm does not explicitly construct a tractable utility function; rather, a utility function is implicit in the demand function which our algorithm computes.

A detailed description of the algorithm is postponed to the next section, where we discuss the mixed case. It turns out that the same algorithm works for mixed and indivisible goods.

We should, however, emphasize two points. First, that the algorithm is simple, and that it is simple to see that it runs in polynomial time. Second, that our result does not work by constructing a utility function. It works by showing that the algorithm is maximizing *some* utility function, but we never have the need to explicitly construct one.

2.3. **Mixed goods.** We turn now to the third case, the more general case of mixed goods. We let consumption space $\mathcal{X}$ be a subset of $\mathbb{R}^d_+$, and assume that some goods are available in discrete indivisible units, while other goods are always infinitely divisible. Specifically, we assume that $\mathcal{X} = \mathbb{Z}^{n_1}_+ \times \mathbb{R}^{n_2}_+$, with $n_1 + n_2 = d$. It is easy to accomodate more nuanced situations, but we assume that goods are partitioned in this way for expositional reasons.

We have until now described a consumer's behavior using a utility function, but the presence of mixed goods forces us to use the more primitive notion of a preference relation. A preference relation $\succeq$ on $\mathcal{X}$ is a complete and transitive binary relation (also known as a weak order). A utility function u defines a preference relation by $x \succeq y$ if and only if $u(x) \geq u(y)$, but there are preference relations that are not associated to any utility function. We want to steer clear of the technical issues involved in the existence of a utility

[3]That is to say, $\psi(p, b)$ is the choice of the consumer faced with budget set $B(p, b)$.

[4]Note the divergence from the more classical approach in revealed preference theory; e.g [1, 6]. Most studies of revealed preference construct a rationalizing utility. Here the approach is to construct a demand function, and then show that it is rationalizable.

function for a preference relation, so we shall simply focus on preference relations.[5]

The strict preference associated to $\succeq$ is a binary relation $\succ$ defined as $x \succ y$ whenever $x \succeq y$ and it is false that $y \succeq x$. A preference relation $\succeq$ is *monotonic* if, for all $x, y \in \mathcal{X}$, $x > y \Rightarrow x \succ y$. The monotonicity of $\succeq$ plays the role played by the monotonicity of u before.

The economic theory of the consumer postulates then that the consumer chooses $x \in \mathcal{X}$ such that $p \cdot x \leq b$ and such that $x \succeq y$ for all $y \in \mathcal{X}$ with $p \cdot y \leq b$. We say that x is *maximal* for $\succeq$ in the budget set defined by p and b. Observe that the consumer solves a utility maximization problem (1) whenever $\succeq$ is associated to a utility function. The notion of tractability can be extended from utility functions to preference relations. Recall that a utility function $u : \mathcal{X} \to \mathbb{R}$ is tractable if and only if problem (1) can be solved in polynomial time. We shall say that a preference relation is *tractable* if there is a polynomial time algorithm that finds a maximal x for the budget defined by any pair (p, b).

We have described how the divisible and indivisible cases present computationally hard problems from the worst-case perspective. The previous discussion implies that the same is true of the mixed case, as it is a more general model. We shall therefore focus on the revealed preference approach to the mixed model. In particular, we obtain the following result.

We say that the data is ***rationalizable*** if there is a monotone increasing preference relation $\succeq$ on $\mathcal{X}$ such that

$$\forall y \in \mathcal{X}(p^i \cdot y \leq b^i \text{ and } y \neq x^i \Rightarrow x^i \succ y).$$

Theorem 6. *In the consumer choice problem with mixed goods, data* $\{(x^i, p^i, b^i)\}_{i=0}^n$ *is rationalizable if and only if it is rationalizable via a tractable monotone preference relation.*

We prove Theorem 6 by introducing an efficient algorithm $\mathcal{A} = \mathcal{A}_{\{(x^i,p^i,b^i)\}_{i=0}^n}$ to compute a demand function. The properties of the algorithm were informally described in the section on discrete goods. In fact, the proof of Theorem 6 implies Theorem 5 as well.

In the remainder of this section, we explain the algorithm for computing a demand function from the data. As described in the pseudocode, the algorithm consists of two steps: (i) A preprocessing phase, $\mathcal{A}.\mathrm{preprocess}$, in which a ranking r on $X = \{x^0, x^1, \ldots, x^n\}$ is computed. (ii) A core phase, $\mathcal{A}.\mathrm{evaluate\_demand}$, that uses the ranking in order to evaluate the demand

[5]Utility functions, instead of preference relations, only matter in that their properties can be useful analytically. For example the differentiability of a utility function can be useful, as can concavity. In the mixed-goods case, these properties would not make sense, so there is no real reason for wanting a utility function in addition to a rationalizing preference relation.

function $\psi(p,b)$. Let $\operatorname{lex\,max} S$ denote the lexicographically maximum element of S.

Algorithm $\mathcal{A}$

$\mathcal{A}$.preprocess(Data $\{(x^i,p^i,b^i)\}_{i=0}^{n}$)
begin
- Let $X := \{x^0, x^1, \ldots, x^n\}$;
- Let $E_0 := \{(x,y) \in X^2 : x \neq y, x \leq y\}$ be the binary relation $\leq$ restricted to the elements of X;
- Let $E := E_0 \cup R_0$, and define a directed graph $G := (X, E)$;
- Perform a topological sort of G to get vertex ordering π;
- Output $r : X \to \mathbb{Z}_{++}$ defined such that $r(x) = i$ iff x is the i^{th} element according to π;

end

$\mathcal{A}$.evaluate_demand(Prices p, Budget b)
begin
- Obtain the rank function r computed during preprocessing;
- Compute $z(p,b) = \arg\max\{r(x) \ : \ x \in B(p,b) \cap X\}$ by iterating over the elements of X;
- Let $\psi(p,b) := \operatorname{lex\,max}\{x : x \in B(p,b), z(p,b) \leq x\}$, and compute $\psi(p,b)$ as follows.
- Set $z_0 := z(p,b)$;
- **for** $i = 1$ **to** d **do**
  - Compute $\delta_i := \max\{\delta : z_{i-1} + \delta \cdot \hat{e}_i \in B(p,b)\}$;
  - $z_i \leftarrow z_{i-1} + \delta_i \cdot \hat{e}_i$;
- **end**
- Output z_d; // Note $z_d = \psi(p,b)$

end

A first remark on Algorithm $\mathcal{A}$ is that, the use of $\operatorname{lex\,max}$ in $\mathcal{A}$.evaluate_demand may seem artificial. However, it should be noted that a variety of other functions that project $z(p,b)$ onto the boundary of $B(p,b)$ can be used instead and we have chosen $\operatorname{lex\,max}$ for convenience.

A second remark on Algorithm $\mathcal{A}$ is that, as a side effect, algorithm $\mathcal{A}$ is, in some sense,"learning" the consumer demand function. Indeed, as more data is observed the choice function determined by $\mathcal{A}$ is increasingly accurate. However, it is important to remember that the question being studied in this paper is quite different from whether a consumer demand function can be learned efficiently.

We defer the proof of the correctness of Algorithm $\mathcal{A}$ to Appendix A and discuss only the running time of the algorithm here. It is immediate to see that phase (ii), $\mathcal{A}.\text{evaluate\_demand}$, can be implemented in $\mathcal{O}(nd)$ time. Phase (i), $\mathcal{A}.\text{preprocess}$, can be implemented in $\mathcal{O}(n^2 d)$ time, since we may iterate over X^2, and test if each element belongs to $E_0 \cup R_0$ in $\mathcal{O}(d)$ time, and topological sort is well–known to have a linear–time implementation, i.e., one taking $\mathcal{O}(|X| + |E|)$ time, which in this case amounts to $\mathcal{O}(n^2)$. The total time is thus $\mathcal{O}(n^2 d)$.

2.4. **Dependence on the size of the data set.** A possibly surprising observation about the running time of algorithm $\mathcal{A}$ is that it depends on n, the size of the data set. One may reasonably take n to be a constant when evaluating the running time of the demand function ψ since it is the tractability with respect to p and b that is relevant when the choice function is evaluated for future demands (i.e., the data set is fixed for all future evaluations of the demand function). However, one may also reasonably ask whether the dependence of the running time on n is necessary or if it is an artifact of specific algorithm presented here.

It turns out that there is an unavoidable dependence of any rationalization algorithm on the size of the data in (at least) two ways. First, when determining a demand function to rationalize the data, the algorithm must process every 'distinct' element of the data set, which highlights that the running time must be at least the size of the minimal representation of the data set. Second, the resulting demand function must at least encode each entry of the data set. Thus, the space needed to represent the demand function must also be bounded below by the size of the minimal representation of the data set. More formally, we can prove the following lower bounds on the running time of any algorithm for rationalizing the data set and for the space needed to represent the resulting demand function. The proofs are deferred to Appendix C. In this section we work under the assumption of indivisible goods.

Proposition 7. *Any algorithm that takes as input a data set with n data points, a price vector p, and a budget b and outputs $\psi(p, b)$ for a ψ which rationalizes the data set requires, in the worst case, $\Omega(n)$ running time on a RAM with word size $\Theta(\log n)$, even when there are only two goods.*

Proposition 8. *Any demand function ψ that rationalizes a data set with n data points requires $\Omega(n \log n)$ bits of space to represent, in the worst case, even when there are only two goods.*

These two propositions highlight that the dependence of the running time on the size of the data set is unavoidable. However, observe that the running time of algorithm $\mathcal{A}$ does not match the lower bound above, since it has a quadratic dependence on n. On the other hand, our preprocessing step is space-optimal, since it can be implemented using linear space by performing the topological sort in a lazy manner that generates the adjacently list of each vertex of G on the fly, and hence avoids the need to explicitly construct G. The running time of this lazy implementation is also $\mathcal{O}(n^2 d)$.

3. WALRASIAN EQUILIBRIUM THEORY

The second theory which we use to highlight our revealed preference approach to computational complexity is Walrasian general equilibrium theory. General equilibrium theory is perhaps the most widely applied model in mainstream microeconomics: it underlines most of modern macroeconomics, international trade, and financial economics. In this section, we flesh out some of the implications of our approach and contrast these implications with the worst-case view of computational complexity.

The basic model of general equilibrium assumes a finite number of goods and agents. Agents have preferences and are endowed with initial quantities of the goods. (We ignore production in our discussion.) The theory predicts which prices will emerge for the different goods, and what the quantities of the goods consumed by the agents will be. Specifically, the theory predicts that the prices and quantities conform to a Walrasian equilibrium: a situation where agents choose quantities optimally given prices, endowments, and the incomes derived from selling their endowments; and where supply equals demand for all goods simultaneously.

More formally, we consider the standard model of an exchange economy. Suppose there are d goods and m agents. Each agent i is described by a consumption space $\mathcal{X}_i \subseteq \mathbb{R}^d_+$, a monotone increasing utility function $u_i : \mathcal{X}_i \to \mathbb{R}$, and an endowment $\omega_i \in \mathcal{X}_i$. An economy is therefore described by $(\mathcal{X}_i, u_i, \omega_i)_{i=1}^m$. By changing the consumption space $\mathcal{X}_i$, we can work with either indivisible or divisible goods.

An ***allocation*** is a collection $\mathbf{x} = (x_i)_{i=1}^m$ with $x_i \in \mathcal{X}_i$, and such that $\sum_{i=1}^m x_i = \sum_{i=1}^m \omega_i$. A ***Walrasian equilibrium*** is a pair $(\mathbf{x}, p)$, where $\mathbf{x} = (x_i)_{i=1}^m$ is an allocation and $p \in \mathbb{R}^d_{++}$ is a price vector such that, for all i, x_i solves the problem

$$\max \left\{ u_i(x) : x \in \mathcal{X}_i,\ p \cdot x \leq p \cdot \omega_i \right\}.$$

A worst-case view. As discussed in the introduction, the computational complexity of Walrasian equilibrium has received some attention in recent years. On the positive side, it is known how to compute Walrasian equilibria in polynomial time if all agents have *linear* utility functions [17, 32]. Similarly, it is known how to compute a Fisher equilibria (a special case of Walrasian equilibria) for some special cases in the absence of the gross substitutability property [15, 10]. However, in general, finding a Walrasian equilibrium is known to be "hard." Specifically, computing Walrasian equilibria in instances where agents have utilities that are separable over the goods and concave piecewise linear for each good is PPAD-complete [8, 31].

A revealed preference view. For the remainder of the section, we contrast the above with the revealed preference view of computational complexity. To begin, we can describe the revealed preference formulation of Walrasian equilibrium as follows. Given data on prices and resources for an economy, we want to know if the data are rationalizable as Walrasian equilibrium outcomes. The rationalization involves specifying utility functions for the agents.

Specifically, we look at conditions under which a data set has a rationalization for which finding a Walrasian equilibrium is easy. Importantly, there are two considerations that are relevant: We study whether the utility functions yielded by the rationalization can be chosen so that (i) optimizing them is tractable, and (ii) the problem of finding a Walrasian equilibrium is easy. We address these two considerations in order in the next two sections.

3.1. **Tractable utilities.** We first address whether the rationalizing utilities can be chosen so that the agent optimization is tractable. Not surprisingly, since consumers are the building blocks of the Walrasian model, the results of Section 2 imply that: when data are equilibrium rationalizable, they are rationalizable with utility functions that are tractable.

To apply the results in Section 2 we assume that $\mathcal{X}_i = \mathbb{Z}^d_+$. We assume that we have data on prices, incomes and resources. Specifically, a ***data set*** is a collection $(p^k, (\omega^k_i)^m_{i=1})^K_{k=0}$. We follow [4] in assuming that individual consumption vectors are not observable. We say that a data set is ***rationalizable*** if there are utility functions u_i for each agent, and vectors x^k, $k = 1, \ldots, K$, such that (x^k, p^k) is a Walrasian equilibrium of the economy $(\mathcal{X}_i, u_i, \omega^k_i)^m_{i=1}$.[6]

Given the above setup, the results in Section 2 imply that in the case of indivisible goods (i.e., when $\mathcal{X}_i = \mathbb{Z}^d_+$), a data set $(p^k, (\omega^k_i)^m_{i=1})^K_{k=0}$ is either not rationalizable, or it is rationalizable using tractable u_i for all agents: utilities such that the problems

$$\max\left\{u_i(x) : x \in \mathbb{Z}^d_+,\ p \cdot x \leq p \cdot \omega_i\right\}$$

can be solved in strongly polynomial time. Further, from the discussion in Section 2 it is easy to see that a similar result holds for infinitely divisible goods (i.e., $\mathcal{X}_i = \mathbb{R}^d_+$) as a consequence of Afriat's Theorem.

[6]One can, instead, assume that only individual incomes are observable, and not individual endowments. The results are the same (see [4]).

However, it is important to realize that the results in this section do not address the complexity of *finding* a Walrasian equilibria. We have limited the discussion to the individual consumer's problem, not the collective problem of finding an equilibrium. The later question is clearly important, and it is the focus of the next section.

3.2. **Tractable equilibria.** Next, we consider whether rationalizing utilities can be chosen so that the problem of finding an equilibrium is tractable. This is arguably the most important computational problem that arises in the theory of general equilibrium. Though we will not resolve the question, the result in this section provides an important starting point. Specifically, we give a condition on data sets which ensures that there is a rationalization for which finding an equilibrium is easy. The condition relates to the existence of a "representative consumer;" a construct often used in economics. When there is a rationalization that allows for a representative consumer, we show that the problem of finding Walrasian equilibrium is tractable. Importantly, many empirical studies of consumption find results that are consistent with the existence of a representative consumer ([27, 29]).

Moving to the analysis, suppose that $\mathcal{X}_i = \mathbb{R}^d_+$. An exchange economy $(\mathcal{X}_i, u_i, \omega_i)_{i=1}^m$ is *equilibrium tractable* if for any $\omega \in \mathbb{R}^d_{++}$ there is a convex optimization program that returns a pair $(\mathbf{x}, p)$, where $\mathbf{x} = (x_i)_{i=1}^m$ is an allocation of ω and for all i, x_i solves the problem

$$\max \{u_i(x) : x \in \mathcal{X}_i,\ p \cdot x \leq b_i\};$$

where $\sum_i b_i = p \cdot \omega$.

Note that our notion of tractability is close in spirit to what we used for the consumer's problem (Problem (1)) when goods are infinitely divisible: we essentially reduce from convex optimization, and thus argue that finding a Walrasian equilibrium is easy. Of course, an algorithm for convex optimization will usually deliver an approximation to a solution, so we would obtain an approximate Walrasian equilibrium as well.

Suppose that there are n agents, and we want to test the theory that their consumption behavior is as if there were a (normative) representative consumer. The theory starts from the following primitives. First, a social welfare function $W : \mathbb{R}^n \to \mathbb{R}$ which is strictly monotonic and quasiconcave. Second, n utility functions $u_i : \mathbb{R}^L_+ \to \mathbb{R}$ which are strictly monotonic and quasiconcave. For given prices $p \in \mathbb{R}^L_{++}$ and budget $b > 0$ an allocation $(x_1, \ldots, x_n)$ is determined according to the maximization of $W(u_1(x_1), \ldots, u_n(x_n))$ subject to $(x_1, \ldots, x_n) \in \mathbb{R}^{md}_+$ and $p \cdot \sum_{i=1}^n x_i \leq b$.

A ***data set*** consists of a finite collection

$$(p^k, x_1^k, \ldots, x_n^k), k = 1, \ldots, K$$

of prices and allocations. A data set is ***rationalizable with a representative consumer*** if there is W and $u_1, \dots, u_n$ that would have generated, for each k the observed allocation $(x_1^k, \dots, x_n^k)$ at the observed prices p^k and income $p^k \cdot \sum x_i^k$.

Theorem 9. *If a data set is rationalizable with a representative consumer, then it has a rationalization that is equilibrium-tractable.*

Proof. Theorem 9 relies on the following convex program. Consider the function U defined by

$$U(z) = \sup\{W(u_1(z_1), \dots, u_n(z_n)) : (z_1, \dots, z_n) \in \mathbb{R}_+^{md}, \sum_{i=1}^{m} z_i = \omega\}.$$

Consider now the problem

$$\begin{array}{l} \min p \cdot z \\ \text{s.t.} U(z) \geq U(\omega) \end{array} . \tag{4}$$

Problem (4) is a convex program: the function U is quasiconcave, hence its upper contour sets are convex. The convex program finds a supporting hyperplane to the upper contour set of the function U at ω. It is now easy to see that a solution p^* to Problem (4) defines a Walrasian equilibrium. Let x^* solve the problem of maximizing $W(u_1(x_1), \dots, u_n(x_n))$ subject to $(x_1, \dots, x_n) \in \mathbb{R}_+^{md}$ and $p^* \cdot \sum_{i=1}^n x_i \leq p^* \cdot \omega$. Then it is easy to show that x_i^* solves the problem of maximizing $u_i(x_i)$ subject to $x_i \in B(p^*, p^* \cdot x_i^*)$. This is essentially a version of the second welfare theorem. Hence, by solving Problem (4) we obtain a Walrasian equilibrium. □

To this point we have not described the existence of a representative consumer in terms of the data. It is natural to inquire about a test for this property; that is, can the property be checked on data sets. The following proposition provides an answer. The proposition is outside of the scope of this paper, so we include it without proof.

Proposition 10. *A data set $(p^k, x_1^k, \dots, x_n^k), k = 1, \dots, K$ is rationalizable if and only if there are numbers U_i^k, for $i = 0, \dots, m$ and $k = 1, \dots, K$ such that*

$$U_i^l \leq U_i^k + \lambda_i^k p^k \cdot (x_i^l - x_i^k)$$
$$U_0^l \leq U_0^k + \lambda_0^k \sum_{i=1}^{n} \frac{U_i^l - U_i^k}{\lambda_i^k}$$

4. A GENERAL FORMULATION

In this paper we focus on two particular theories, consumer choice theory and general equilibrium theory; however it is clear that the revealed preference view of computational complexity can be studied much more broadly. In this section we present a general formulation of the revealed preference view which can be applied, for example, to test the impact of computational constraints on the empirical content of the theory of Nash equilibrium, the theory of stable matchings, and other economic theories.

To begin the general formulation, let $\mathcal{I}$ be a set of possible *inputs,* and $\mathcal{O}$ be a set of possible *outputs.* For example, in consumer theory, $\mathcal{I}$ is the set of all price-budget pairs (p, b), while $\mathcal{O}$ is the consumption space ($\mathcal{X}$).

We represent a *theory* as a collection $\mathcal{G} = \{g_\tau : \tau \in \mathcal{T}\}$ of correspondences, $g_\tau \subseteq \mathcal{I} \times \mathcal{O}$, which we index by a set $\mathcal{T}$ for convenience. For example, consumer choice theory is the collection $\{\psi_u : u \in U\}$ where U is the set of all monotone increasing utility functions $u : \mathcal{X} \to \mathbb{R}$ and $\psi_u(p, b)$ is the demand function obtained from maximizing u over the budget set defined by (p, b).

Classical revealed preference theory. When taking a revealed preference perspective on a theory, the goal is to test the theory using data. We define a *data set* D as a finite collection $D = (i_k, o_k)_{k=1}^n$ of pairs of inputs and outputs. The set of all possible data sets is $\mathcal{D} = (\mathcal{I} \times \mathcal{O})^*$, i.e., all finite sequences of pairs in $\mathcal{I} \times \mathcal{O}$.

A data set $(i_k, o_k)_{k=1}^n$ is *rationalizable* by the theory $\mathcal{G} = \{g_\tau : \tau \in \mathcal{T}\}$ if there is $\tau \in \mathcal{T}$ such that $o_k \in g_\tau(i_k)$ for all $k = 1, \dots, n$. In that case, we say that g_τ *rationalizes* the data.

Further, we define a *rationalization rule* $r : \mathcal{D} \to (\mathcal{I} \to \mathcal{O})$ such that for each rationalizable data set D, $r(D) = g_\tau$ for some $\tau \in \mathcal{T}$ such that g_τ rationalizes the data.

Finally, define the set of data sets rationalizable by theory $\mathcal{G}$ as

$$\mathcal{R}_\mathcal{G} = \{D \in \mathcal{D} : \exists \tau \in \mathcal{T} : g_\tau \text{ rationalizes } D\}.$$

Given this setup, some classic revealed preference questions are:

Is theory $\mathcal{G}$ refutable, i.e., Is $\mathcal{R}_\mathcal{G} \subsetneq \mathcal{D}$?
Is data D rationalizable by theory $\mathcal{G}$, i.e., Is $D \in \mathcal{R}_\mathcal{G}$?

However, these classic questions ignore computational constraints on the theory.

Adding computational constraints. Our revealed preference view of computational complexity can be fit into the above framework as follows.

Consider again the rationalization rule $r : \mathcal{D} \to (\mathcal{I} \to \mathcal{O})$. We now seek to understand the impact of imposing an additional constraint on $r(D) = g_\tau$

which rationalizes D — it must be "tractable." Formally, we say that $g_\tau(i) = [r(D)](i)$ is *tractable* if it is computable in time polynomial in the size of both D and i. We require it to be polynomial in the size of D for reasons discussed below (and in Section 2).

Define the set of data sets that are rationalizable using tractable instances of theory $\mathcal{G}$ as

$$\mathcal{E}_{\mathcal{G}} = \{D \in \mathcal{D} : \exists \tau \in \mathcal{T}.\ g_\tau \text{ rationalizes } D \textit{ and } g_\tau \text{ is tractable } \}.$$

Given this framework, the general version of the question posed by the revealed preference view of complexity is:

Do computational constraints have empirical consequences,
i.e., Is $\mathcal{E}_{\mathcal{G}} \subsetneq \mathcal{R}_{\mathcal{G}}$?

In general, the set $\mathcal{E}_{\mathcal{G}}$ constitutes a test for the joint hypotheses of assuming the economic theory $\mathcal{G}$, and requiring that the economic system not solve unrealistically hard problems. If $\mathcal{E}_{\mathcal{G}} \subsetneq \mathcal{R}_{\mathcal{G}}$ then one would seek to characterize $\mathcal{E}_{\mathcal{G}}$ so that actual tests of the joint hypotheses can be carried out.

One final remark about this formulation is that it may initially seem strange that the computational time of g_τ depends on size of D, but, as we saw in the case of consumer choice theory, this is necessary in order to ensure that g_τ rationalizes D.

In fact, we can generalize Propositions 7 and 8 for this general formulation as follows. For a theory $\mathcal{G}$, define

$$\mathcal{R}_{\mathcal{G}}\left(\{i_k\}_{k=1}^n\right) := \mathcal{R}_{\mathcal{G}} \cap \left\{\{(i_k, o_k)\}_{k=1}^n \ :\ \forall k,\ o_k \in \mathcal{O}\right\}$$

to be the set of rationalizable data sets on input sequence $\{i_k\}_{k=1}^n$. Then we have the following lower bounds on the running time of r and space required to represent $g_\tau \in \mathcal{G}$. The proofs are deferred to Appendix C.

Proposition 11. *Fix a theory $\mathcal{G}$ and $n \in \mathbb{N}$. The running time required to compute $[r(D)](i)$ for D of size n is*

$$\Omega\left(\frac{\max_{(i_1,\ldots,i_n)} \log_2 \left|\mathcal{R}_{\mathcal{G}}\left(\{i_k\}_{k=1}^n\right)\right|}{\log n}\right)$$

in the worst case on a RAM with wordsize $\Theta(\log n)$, where the max is taken over all input sequences $\{i_k\}_{k=1}^n$ of length n.

Proposition 12. *Fix a theory $\mathcal{G}$ and $n \in \mathbb{N}$. The space required to represent any $g_\tau \in \mathcal{G}$ rationalizing a rationalizable data set of size n is at least $\max_{(i_1,\ldots,i_n)} \log_2 \left|\mathcal{R}_{\mathcal{G}}\left(\{i_k\}_{k=1}^n\right)\right|$ bits in the worst case, where the max is taken over all input sequences $\{i_k\}_{k=1}^n$ of length n.*

These propositions highlight that there is an explicit tradeoff between the generality of the theory being tested and the running time of r and the space complexity of g_τ. The generality of the theory is measured by the quantity

$\max_{(i_1,\ldots,i_n)} \left|\mathcal{R}_{\mathcal{G}}\left(\{i_k\}_{k=1}^n\right)\right|$, which captures the notation that a more general theory can explain (rationalize) a larger class of data sets.

5. CONCLUDING REMARKS

The core idea presented by this paper is that there is a philosophical difference in how many computer scientists and economists view economic models, and that the lack of broad acceptance of worst-case complexity results in economics stems from the difference between these viewpoints. We posit that while computer scientists tend to think 'algorithmically' about economic models, economists tend to think 'empirically' about the models.

In particular, an *algorithmic view* of the model assumes the model is fixed and literally true, and then proceeds to ask about the demands placed on the agents by the model. That is, it assumes that the agent is simply an implementation of the model and asks whether the agent can efficiently compute its decisions.

In contrast, an *empirical view* of the model takes the model as a tool for thinking about reality. One does not presume agents literally follow the model, only that the model provides a way to explain the observed behavior, i.e., the data. Thus, an empirical view of economic models is that they provide 'as-if' tools: economists postulate a model and claim the observable variables behave as if the model were true.

Note that the empirical view does not deny that computational issues should be studied. To the contrary, a model still loses credibility if the agents must solve computationally hard problems; however, worst-case complexity is no longer the relevant concept. Instead, the question is whether data from an observed phenomenon can always be explained by the theory with the additional constraint that agents are not required to solve computationally hard problems.

This paper presents a formulation of how to incorporate computational information into the empirical view; however the algorithmic view continues to be very relevant, of course. Many problems require the design of automated agents who participate in various economic interactions: in this case the utilities are hard-wired by the designer, they are not theoretical unknowns. Using classical economic jargon, one could say that the algorithmic view is relevant for normative economics while the empirical view is relevant for positive economics.

REFERENCES

[1] S. N. Afriat. The construction of utility functions from expenditure data. *International Economic Review*, 8(1):67–77, feb 1967.
[2] S. N. Afriat. Efficiency estimation of production functions. *International Economic Review*, 13(3):568–598, oct 1972.

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
[4] D. J. Brown and R. L. Matzkin. Testable restrictions on the equilibrium manifold. *Econometrica*, 64(6):1249–1262, November 1996.
[5] D. J. Brown and C. Shannon. Uniqueness, stability, and comparative statics in rationalizable Walrasian markets. *Econometrica*, 68(6):1529–1439, November 2000.
[6] C. P. Chambers and F. Echenique. Supermodularity and preferences. *Journal of Economic Theory*, 144(3):1004–1014, May 2009.
[7] C. P. Chambers, F. Echenique, and E. Shmaya. On behavioral complementarity and its implications. Forthcoming, *Journal of Economic Theory*, 2009.
[8] X. Chen, D. Dai, Y. Du, and S.-H. Teng. Settling the complexity of arrow-debreu equilibria in markets with additively separable utilities. In *FOCS '09: Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 273–282, Washington, DC, USA, 2009. IEEE Computer Society.
[9] B. Codenotti and K. Varadarajan. Computation of market equilibria by convex programming. In N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, editors, *Algorithmic Game Theory*. Cambridge University Press, 2007.
[10] N. R. Devanur, C. H. Papadimitriou, A. Saberi, and V. V. Vazirani. Market equilibrium via a primal–dual algorithm for a convex program. *J. ACM*, 55(5):1–18, 2008.
[11] W. E. Diewert and C. Parkan. Linear progamming tests of regularity conditions for production functions. In W. Eichhorn, R. Henn, K. Neumann, and R. Shephard, editors, *Quantitative Studies on Production and Prices*, pages 131–158. Physica-Verlag, Vienna, 1983.
[12] F. Echenique, D. Golovin, and A. Wierman. A revealed preference approach to computational complexity in economics. In *Proceedings of the ACM conference on Electronic Commerce*, 2011.
[13] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634 – 652, July 1998.
[14] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.
[15] D. Garg, K. Jain, K. Talwar, and V. V. Vazirani. A primal-dual algorithm for computing fisher equilibrium in the absence of gross substitutability property. *Theor. Comput. Sci.*, 378(2):143–152, 2007.
[16] I. Gilboa, A. Postlewaite, and D. Schmeidler. The complexity of the consumer problem and mental accounting. Mimeo: Tel-Aviv University, 2010.
[17] K. Jain. A polynomial time algorithm for computing an Arrow–Debreu market equilibrium for linear utilities. *SIAM J. Comput.*, 37(1):303–318, 2007.
[18] K. Jain and V. Vazirani. Eisenberg-gale markets: Algorithms and structural properties. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 364–373. ACM, 2007.
[19] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack problems*, chapter 12. Springer, 1 edition, 2004.
[20] S. Khot. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM J. Comput.*, 36(4):1025–1071, 2006.
[21] R. L. Matzkin. Axioms of revealed preference for nonlinear choice sets. *Econometrica*, 59(6):1779–1786, November 1991.
[22] M. K. Richter. Revealed preference theory. *Econometrica*, 34(3):635–645, jul 1966.

[23] M. K. Richter. Rational choice. In J. S. Chipman, L. Hurwicz, M. K. Richter, and H. F. Sonnenschein, editors, *Preferences, Utility and Demand*, pages 29–58. Harcourt Brace Jovanivic Inc., 1971.
[24] P. Samuelson. A note on the pure theory of consumer's behaviour. *Economica*, 5(17):61–71, 1938.
[25] K. Shirai. On the existence of a submodular utility function. Mimeo: Waseda University, 2010.
[26] H. A. Simon. Rationality in psychology and economics. *The Journal of Business*, 59:209–224, 1986.
[27] H. R. Varian. The nonparametric approach to demand analysis. *Econometrica*, 50(4):945–974, jul 1982.
[28] H. R. Varian. Non-parametric tests of consumer behaviour. *Review of Economic Studies*, 50(1):99–110, jan 1983.
[29] H. R. Varian. Non-parametric analysis of optimizing behavior with measurement error. *Journal of Econometrics*, 30:445–458, 1985.
[30] H. R. Varian. Revealed preference. In M. Szenberg, L. Ramrattan, and A. Gottesman, editors, *Samuelsonian economics and the twenty-first century*, pages 99–115. Oxford University Press, USA, 2006.
[31] V. V. Vazirani and M. Yannakakis. Market equilibrium under separable, piecewise-linear, concave utilities. In *Proc. Innovations in Computer Science*, pages 156–165, Beijing, China, January 2010.
[32] Y. Ye. A path to the Arrow–Debreu competitive market equilibrium. *Mathematical Programming*, 111(1):315–348, 2008.

APPENDIX A. PROOFS FOR THE CONSUMER CHOICE MODEL

In this section we prove Theorem 5 via a sequence of lemmas culminating in Theorem 17 and Corollary 18.

Lemma 13. *If R_0 is acyclic, then $\preceq := (R_0 \cup \leq)^*$ is acyclic.*

Proof. The proof follows from the definition of budget set. Here, we use the convention that $x\ R_0\ y$ is the same as $(x, y) \in R_0$, meaning x is revealed–preferred to y.

Now suppose by way of contradiction that R_0 is acyclic, but $\preceq$ contains a minimal cycle C. Since $\leq$ is transitive, any minimal cycle consists only of edges in X^2 (recall X is the set of consumption bundles observed in the data plus the zero vector). Otherwise C contains edges (x, y) and (y, z) for some $y \notin X$. Since these pairs cannot be in R_0, we have $x \leq y$ and $y \leq z$. Hence $x \leq z$ and we can replace (x, y) and (y, z) with (x, z), which violates the minimality of C.

Since R_0 is acyclic by assumption, C must contain an edge $(x, y) \notin R_0$. Because $(x, y) \in (R_0 \cup \leq)^*$ by assumption, this means $x \leq y$. However, since $x, y \in X$ from our earlier argument, and because the budget sets are downward–closed with respect to $\leq$, $x \leq y$ implies $(x, y) \in R_0$ by definition of R_0. This contradictions our earlier assumption that $(x, y) \notin R_0$. Hence no minimal cycle exists, and $\preceq$ is acyclic. □

Lemma 14. *Algorithm $\mathcal{A}$ computes a choice function ψ such that $\psi(p, b) \in B(p, b)$ for all p and b.*

Proof. For all valid p and b, $\mathbf{0} \in B(p, b) \cap X$, and all elements $x \in X$ have distinct ranks $r(x)$, so $z(p, b)$ and hence $x(p, b)$ are well defined, and by construction, the output $x(p, b) \in B(p, b)$. □

Lemma 15. *Let r be the ranking computed during preproccessing. Let $\preceq := (R_0 \cup \leq)^*$ and assume R_0 is acyclic. Fix $x, y \in X$. Then ranks are distinct, i.e., $x \neq y$ implies $r(x) \neq r(y)$, and the ranks are monotone in $\preceq$, i.e., $x \preceq y$ implies $r(x) \leq r(y)$.*

Proof. The first part, $x \neq y$ implies $r(x) \neq r(y)$, holds by construction since $r(\cdot)$ is the rank within a permutation π and no two elements have the same rank. Now suppose $x \preceq y$. Then by definition of $\preceq$ there is a path P in $(R_0 \cup \leq)^*$ from x to y. We claim there must be a path P' from x to y in the digraph $G = (X, E)$ constructed during preproccessing, since (i) all the edges in R_0 are present in E, and (ii) every maximal subpath of P starting and ending at points in X and consisting only of edges in $\leq$ can be traversed by a single edge in E_0 by the transitivity of $\leq$, and $E_0 \subseteq E$, and (iii) P can be decomposed into edges in R_0 and maximal subpaths of P starting and ending at points in X. Hence there is a path in G from x to y and so the topological sort must place x before y in π, which implies $r(x) \leq r(y)$. (Note that since R_0 is acyclic by assumption, $\preceq$ is acyclic by Lemma 13, which in turn implies G is acyclic, and thus there exists a topological ordering of its vertices.) □

Lemma 16. *Assume that R_0 is acyclic. The choice function ψ computed by $\mathcal{A}$ explains the data, i.e., $\psi(p^i, b^i) = x^i$ for all $i \in [n]$.*

Proof. Fix $i \in [n]$ arbitrarily. Recall B^i is the budget set faced by the consumer when x^i was chosen. Note $x^i \in B^i \cap X$, so it is considered in the $\arg\max$ in the computation of $z(p^i, b^i)$. We consider two cases, namely $z(p^i, b^i) = x^i$ and $z(p^i, b^i) \neq x^i$.

In the first case, we have $\psi(p^i, b^i) = x^i$, as x^i is a maximal element of the budget set with respect to the lattice ordering $\leq$, and hence the evaluation of the $\operatorname{lex}\max$ in the computation of $x(p^i, b^i)$ is over the singleton set $\{x^i\}$, and so the output of $\mathcal{A}$ is $x(p^i, b^i) = x^i$.

We next argue that the second case cannot occur. Suppose by way of contradiction that $z(p^i, b^i) \neq x^i$. Then there must be some $x^j \neq x^i$ such that $r(x^i) < r(x^j)$ and $x^j \in B^i$, by the definition of $z(p^i, b^i)$. However, the fact that $x^j \in B^i$ and x^i was chosen in the i^{th} observation implies $(x^j, x^i) \in R_0$ and hence $x^j \preceq x^i$, where $\preceq := (R_0 \cup \leq)^*$ as before. Lemma 15 then implies $r(x^j) \leq r(x^i)$, contradicting our earlier deduction that $r(x^i) < r(x^j)$. □

We introduce R_ψ, the revealed preference relation defined by $\psi(\cdot)$:

$$R_\psi := \{(x,y) \ : \ \exists p, b \, . \, x \in B(p,b) \setminus \{y\} \text{ and } y = \psi(p,b)\} \, . \tag{5}$$

Note $\psi(p^i, b^i) = x^i$ for all $i \in [n]$ implies $R_0 \subset R_\psi$.

Theorem 17. *Algorithm $\mathcal{A}$ computes a choice function ψ such that if R_ψ is the preference relation revealed by ψ and R_0 is acyclic, then ψ explains the data and $(R_\psi \cup \leq)^*$ is acyclic.*

Proof. We prove that ψ explains the input in Lemma 16. Suppose, by way of contradiction, that R_0 is acyclic and yet there is a minimal cycle C in the induced binary relation $(R_\psi \cup \leq)^*$. Note that $\preceq$ is acyclic by Lemma 13.

Suppose that C consists of k edges $\{(y^i, y^{i+1}) : 1 \leq i \leq k\}$, where we define $y^{k+1} := y^1$ for convenience. We can suppose without loss of generality that each edge (y^i, y^{i+1}) is in R_ψ; the proof of this fact is similar to the proof of Lemma 13. Let $p(y^i)$ and $b(y^i)$ denote the prices and budget corresponding to the choice of y^i, so that $\psi(p(y^i), b(y^i)) = y^i$. Let $B(y^i) := B(p(y^i), b(y^i))$ denote the corresponding budget set. Let $z^i := z(p(y^i), b(y^i))$ be the intermediate point computed when running $\mathcal{A}$ with input $(p(y^i), b(y^i))$ and the ranking $r(\cdot)$. We consider two cases, namely $z^i = z^j$ for all $i, j \in [k]$, or there exists i, j with $z^i \neq z^j$.

We start with the case that $z^i = z^j$ for all $i, j \in [k]$. By construction,

$$y^i \equiv x(p(y^i), b(y^i)) = \operatorname{lex\,max} \left\{ x \ : \ x \in B(y^i), z^i \leq x \right\}$$

for all i. Since $(y^i, y^{i+1}) \in C$, we know $y^i \in B(y^{i+1})$, i.e., y^i is affordable for the budget set for which the consumer purchases y^{i+1}. Since $z^i = z^{i+1}$, this implies y^i is in the set over which the algorithm computes the $\operatorname{lex\,max}$ when running on input $(p(y^{i+1}), b(y^{i+1}))$, namely $\{x \ : \ x \in B(y^{i+1}), z^{i+1} \leq x\}$. Since y^{i+1} was selected, y^i is lexicographically less than or equal to y^{i+1}, which we denote by $y^i \leq_{\text{Lex}} y^{i+1}$. Since i was arbitrary,

$$y^1 \leq_{\text{Lex}} y^2 \leq_{\text{Lex}} \cdots \leq_{\text{Lex}} y^k \leq_{\text{Lex}} y^{k+1} = y^1.$$

Since lexicographic order is a partial order, we infer $y^1 = y^2 = \cdots = y^k$, contradicting the assumption that C is a cycle.

Moving on to the second case, suppose there exists i, j with $z^i \neq z^j$. Then there exists $i \in [k]$ with $z^i \neq z^{i+1}$. Recall that since $(y^i, y^{i+1}) \in C$, $y^i \in B(y^{i+1})$. Clearly, $y^{i+1} \in B(y^{i+1})$. Note that for all x and y such that $x \leq y$, and for all prices p and budgets b, we have $y \in B(p,b)$ implies $x \in B(p,b)$. Since by construction $z^i \leq y^i$ and $z^{i+1} \leq y^{i+1}$, we infer $z^i, z^{i+1} \in B(y^{i+1})$. From this fact, the definitions of z^i and z^{i+1}, and the fact that distinct points in X have distinct ranks (as stated in Lemma 15), we can infer $r(z^i) < r(z^{i+1})$.

Of course, for any j, if $z^j = z^{j+1}$ then $r(z^j) = r(z^{j+1})$. Hence for all $j \in [k]$, either $r(z^j) = r(z^{j+1})$ or $r(z^j) < r(z^{j+1})$, so that for all

$j \in [k]$, $r(z^j) \le r(z^{j+1})$, and moreover there is at least one strict inequality: $r(z^i) < r(z^{i+1})$. Altogether these facts yield

$$r(z^1) \le \cdots \le r(z^i) < r(z^{i+1}) \le \cdots \le r(z^{k+1}) = r(z^1)$$

contradicting the transitive property of $\le$. □

Corollary 18. *A data $\{(x^i, p^i, b^i)\}_{i=0}^n$ is rationalizable if and only if algorithm $\mathcal{A}$ computes a demand function ψ such that ψ explains the data and is generated by a monotone utility function.*

Proof. By Theorem 3, Lemma 13, and Theorem 17, a data $\{(x^i, p^i, b^i)\}_{i=0}^n$ is rationalizable if and only if the constructed ψ is such that $(R_\psi \cup \le)^*$ is acyclic. Acyclicity implies (by a version of Szpilrajn's lemma) that $(R_\psi \cup \le)^*$ has an extension to $\mathbb{Z}_+^d$. Since $\mathbb{Z}_+^d$ is countable, the extension trivially satisfies the standard order-denseness condition for a utility representation. The utility representation rationalizes ψ by definition of R_ψ; and it is monotone because it represents an extension of $\le$. □

APPENDIX B. HARDNESS PROOFS

In this section we prove various hardness and inapproximability results cited in Section 2. These results show that the consumer is faced with solving an intractable problem according to a worst-case view of computational complexity in the consumer choice model. These results are not surprising or difficult but are included for completeness of presentation.[7]

We start by proving the hardness of maximizing utility subject to a budget constraint under monotone utility functions.

Lemma 19. *Maximizing utility subject to a budget constraint is NP-hard for monotone utility functions, both for divisible and indivisible goods.*

Proof. For indivisible goods, the problem is NP-hard even for linear utility functions, since the problem

$$\arg\max \{u \cdot x \ : \ p \cdot x \le b, \ x \in \mathbb{Z}_+\}$$

is precisely the UNBOUNDED KNAPSACK problem, a classic NP-hard problem [14].

For divisible goods, we may fix arbitrary nonnegative constants $v_1, \ldots, v_d$ and define the utility function as $u(x) = \sum_{i=1}^d v_i \mathbf{1}\{x_i \ge 1\}$, where $\mathbf{1}\{P\}$ is the indicator variable for predicate P, i.e., $\mathbf{1}\{P\} = 1$ if P is true, and $\mathbf{1}\{P\} = 0$ otherwise. Note u is monotone. The resulting problem,

$$\arg\max \{u(x) \ : \ p \cdot x \le b, \ x \in \mathbb{R}_+\},$$

[7] If the reviewers are aware of previous work that has already proven these results, we will happily remove these proofs.

is again equivalent to UNBOUNDED KNAPSACK. The less-approximable DENSEST-k-SUBGRAPH problem can likewise be encoded: given $G = (V, E)$ set $u(x) = \sum_{(u,v)\in E} \mathbf{1}\{x_u \geq 1 \text{ and } x_v \geq 1\}$, $b = k$, and $p = \mathbf{1}$. □

Next, We provide an inapproximability result for maximizing a monotone supermodular function f subject to a cardinality constraint. A special case of this problem called QUADRATIC KNAPSACK has long been known to be NP-hard [19], via a simple reduction from DENSEST-k-SUBGRAPH.

Despite much effort, accurate bounds on the best approximation factor obtainable for DENSEST-k-SUBGRAPH by an efficient algorithm under reasonable complexity assumptions have eluded all attempts to date. We can, however, exploit a construction due to Khot to obtain the following result.

Theorem 20. *Fix any constant $c \geq 1$, and let $\alpha(n) = 2^{n^c}$. There is no polynomial time $\alpha(n)$-approximation algorithm for maximizing a monotone supermodular function f subject to a cardinality constraint, unless NP has randomized subexponential time algorithms, i.e., unless NP $\subseteq \cap_{\epsilon>0}$BPTIME $\left(2^{n^\epsilon}\right)$, where $n = \log_2 |\operatorname{dom}(f)|$ is the size of the ground-set V.*

Proof. Fix a constant $d \geq 1$ and a DENSEST-k-SUBGRAPH input (G = (V,E), k), and define $f(S) := |E \cap \binom{S}{2}|$ for a subset of vertices S. Let $\tau(x) := 2^{x^d}$ and let $F(S) := \tau(f(S))$ where $\tau(x) := 2^{x^d}$. It may be easily verified that f is monotone supermodular, hence $g \circ f$ is as well for any monotone increasing and convex function g, such as τ. Thus F is monotone supermodular. For now, assume that there is a distribution $\mathcal{D}$ on DENSEST-k-SUBGRAPH instances such that no polynomial time algorithm can distinguish instances with $\max_{|S|\leq k} f(S) \leq x$ from those with $\max_{|S|\leq k} f(S) \geq \alpha x$, when they are drawn from $\mathcal{D}$. Then it must be hard to distinguish instances with $\max_{|S|\leq k} F(S) \leq 2^{x^d}$ and those with $\max_{|S|\leq k} F(S) \geq 2^{(\alpha x)^d}$, because they correspond to exactly the same distinction. Note also that all integers in the range of F can be written with polynomial in n bits, so the problem size (which we take to be $|V| + \log F(V)$, where V is the ground set) increases only by a polynomial factor when going from f to F. It follows that polynomial-time α-inapproximability for the first problem implies polynomial-time β-inapproximability for the latter, with $\beta := 2^{(\alpha x)^d}/2^{x^d}$. It remains to establish a lower bound on β.

To lower bound β, it is useful to understand possible values for x and α. Khot [20] (*c.f.* section C.2) proves the existence of a distribution $\mathcal{D}$ with $x = \Omega(n)$ and $\alpha = (1 + \epsilon)$, under the complexity-theoretic assumption NP $\not\subseteq \cap_{\epsilon>0}$BPTIME $\left(2^{n^\epsilon}\right)$. Define constant $\delta > 0$ so that $x = \delta n$ in Khot's hard input distribution. Then $\log_2(\beta) = (\delta n)^d \left((1 + \epsilon)^d - 1\right)$. Setting $d \geq \frac{1}{\log_2(1+\epsilon)}$ ensures $\log_2(\beta) \geq (\delta n)^d$. Hence we have an inapproximability factor of $2^{(\delta n)^d}$ for arbitrarily large constant d, which is equivalent to an inapproximability factor of 2^{n^c} for arbitrarily large constant c. $\square$

APPENDIX C. PROOFS OF SPACE COMPLEXITY AND RUNNING TIME LOWER BOUNDS

We provide proofs for Propositions (7), (8), (11), and (12) in the reverse order of their appearance. We do so because we find it convenient to start with the proofs for general theories and then proceed to the proofs for the theory of the consumer.

of Proposition 12. Any g_τ that rationalizes data $D := \{(i_k, o_k)\}_{k=1}^n$ must encode a function $h : \{i_k : 1 \leq k \leq n\} \to \mathcal{O}$ that maps i_k to o_k for all k, since $g_\tau(i_k) = o_k$ by the assumption that g_τ rationalizes D. Fix an input sequence $\{i_k\}_{k=1}^n$. There are $N := |\mathcal{R}_\mathcal{G}(\{i_k\}_{k=1}^n)|$ possible functions h that map the elements of $\{i_k\}_{k=1}^n$ to outputs such that the resulting sequence $\{(i_k, h(i_k))\}_{k=1}^n$ is rationalizable. Hence g_τ encodes one of N states, which requires at least $\log_2 N$ bits in the worst-case. Since $\{i_k\}_{k=1}^n$ is an arbitrary input sequence of length n, we may maximize over such sequences to obtain the claimed worst-case bound. $\square$

of Proposition 11. Fix theory $\mathcal{G}$, and let

$$N := \max_{(i_1,\ldots,i_n)} |\mathcal{R}_\mathcal{G}(\{i_k\}_{k=1}^n)|.$$

Let $\{i_k\}_{k=1}^n$ be an input sequence achieving $|\mathcal{R}_\mathcal{G}(\{i_k\}_{k=1}^n)| = N$. Without loss of generality, we assume the inputs i_k are distinct, otherwise we may remove duplicates to obtain a smaller input sequence $\{i'_k\}_{k=1}^{n'}$ with $|\mathcal{R}_\mathcal{G}\left(\{i'_k\}_{k=1}^{n'}\right)| = N$. Fix a RAM with wordsize $w = \Theta(\log n)$, and assume there is a deterministic algorithm $\mathcal{A}^*$ in this machine model which computes a function $f : \mathcal{D} \times \mathcal{I} \to \mathcal{O}$ in worst case time $t = t(|D|, |i|)$ such that $i \mapsto f(D, i)$ is a demand function rationalizing D. Fix a data set $D \in \mathcal{R}_\mathcal{G}(\{i_k\}_{k=1}^n)$, and define o_k for $k \in [n]$ such that $D = \{(i_k, o_k)\}_{k=1}^n$. Since we may permute the data points (i_k, o_k) in D arbitrary, this implies that for any $i \in \{i_k\}_{k=1}^n$ this algorithm $\mathcal{A}^*$ can infer the value of $f(D, i)$ after reading *any* $q := t\left\lceil \frac{w}{s} \right\rceil$ data points, where s is the number of bits to encode a data point. By the distinctness of the inputs, $s \geq \log_2(n)$, so that $q = \mathcal{O}(t)$. Since $\mathcal{A}^*$ can infer the value of $f(D, i)$ for all $i \in \{i_k\}_{k=1}^n$ using the same q data points, it can compute D given $\{i_k\}_{k=1}^n$. Hence, qw bits suffice to identify which element of $\mathcal{R}_\mathcal{G}(\{i_k\}_{k=1}^n)$ is the actual data set, which implies $qw \geq \log_2 N$. Therefore, we obtain $\log_2 N \leq qw = \mathcal{O}(tw)$, which may be rearranged to yield the claimed bound $t = \Omega\left(\frac{\log_2 N}{w}\right)$. $\square$

of Proposition 8. By Proposition 12, it suffices to exhibit a sequence $\{i_k\}_{k=1}^n$ such that $\log|\mathcal{R}_{\mathcal{G}}(\{i_k\}_{k=1}^n)| = \Omega(n \log n)$, which we will now proceed to do.

Suppose we are in the indivisible goods case. Recall an input in this theory consists of a price vector p and a budget b, and an output is a consumption bundle. We suppose $p^k = \mathbf{1}$ and $b^k = k$ for all $k \in [n]$. Next, we count the number of rationalizable data sets whose inputs consist of such p^k and b^k, i.e., $N := |\mathcal{R}_{\mathcal{G}}(\{i_k\}_{k=1}^n)|$ where $i_k := (p^k, b^k)$ for all k.

We claim $N = (n+1)!$. To show this, we show that for all k with $1 \leq j < n$, for any choice of $x^1, \ldots, x^{j-1}$ such that $\left\{(x^k, p^k, b^k)\right\}_{k=0}^{j-1}$ is rationalizable, there are $j+1$ choices for x^j such that $\left\{(x^i, p^i, b^i)\right\}_{i=0}^{j}$ is also rationalizable. So, we proceed by induction on j. The base case, $j = 1$ is trivial, since the consumer can buy one unit of either of the two goods. For the induction case, we claim that any consumption bundle $(x_1, x_2) \in \mathbb{Z}_+^2$ such that $x_1 + x_2 = j$ yields a rationalizable extension of the data. By Theorem 3, it suffices to prove that the new data point $(x^j, p^j, b^j) := ((x_1, x_2), \mathbf{1}, j)$ does not introduce a cycle into the induced preference relation R_0. However this is clearly impossible since $(x^j, x^i) \notin R_0$ for all $i < j$. Since the number of solutions $(x_1, x_2) \in \mathbb{Z}_+^2$ to $x_1 + x_2 = j$ is $j+1$, this concludes the induction step. Hence we get the rather trivial recurrence $N(j) = (j+1)N(j-1)$ with $N(1) = 2$, which has solution $N := N(n) = (n+1)!$. Finally, we note that $\log_2(n+1)! = (1 - o(1))n \log_2 n$, which completes the proof. □

of Proposition 7. Proposition 7 claims that any function $f : \mathcal{D} \times \mathcal{I} \to \mathcal{O}$ such that $i \mapsto f(D, i)$ is a demand function rationalizing D implies f takes worst-case $\Omega(|D|)$ time, where $|D|$ is the number of data points (i.e., pairs (i_k, o_k)) in D. At a high level is this due to the simple observation that f must read all of D in the worst case. We can prove it by applying Proposition 11, using the construction of Proposition 8 which gives an input sequence with $\log|\mathcal{R}_{\mathcal{G}}(\{i_k\}_{k=1}^n)| = \Omega(n \log n)$. Combining the two immediately yields the claimed lower bound of $\Omega(n)$. □