High-Dimensional Decision Theory*

Modibo K. Camara†

June 15, 2020

Abstract

In economic theory, rationality assumptions are used to make predictions about choice behavior. However, because life involves so many decisions, rationality in practice means solving a high-dimensional optimization problem. This can be computationally hard. In this paper, I introduce a model of high-dimensional choice under uncertainty, along with an axiom of computational tractability, to ask two questions. First, can tractability axioms, paired with rationality axioms, be used to obtain tighter predictions for choice behavior? I prove two representation theorems that provide an affirmative answer, by characterizing the set of tractable and rationalizable choice rules. In particular, one of the representations corresponds to a heuristic known as narrow choice bracketing. Second, when tractability constraints bind, will a self-interested decision-maker make rationalizable choices? Not necessarily. I show that for many intractable utility functions, no rationalizable algorithm obtains a constant approximation, yet every such algorithm is weakly dominated by an irrational algorithm that does.

Keywords: bounded rationality, choice under uncertainty, subjective expected utility, computational complexity, Hadwiger number

*I am grateful to Huck Bennett, Xiaoyu Cheng, Eddie Dekel, Jeff Ely, Faruk Gul, Jason Hartline, Peter Klibanoff, Kevin Leyton-Brown, Chuck Manski, Alvaro Sandroni, Eran Shmaya, Ludvig Sinander, Marciano Siniscalchi, and Lorenzo Stanca, in addition to several audiences at Northwestern.

†Department of Economics, Northwestern University. Email: modibokhane@gmail.com.

1 Introduction

Economists are concerned with understanding choice behavior. We are especially concerned with the following question: how does an individual's behavior change as the options she is presented with change? It is difficult to overstate the centrality of this question. A satisfactory answer is needed for any kind of counterfactual inference (e.g. in mechanism design).

To this end, Paul Samuelson and subsequent researchers developed an elegant, axiomatic theory of choice. The theory postulates that the individual's choices will be coherent, in the sense that she will avoid violating certain logical axioms. If these axioms hold, the individual behaves *as if* she were maximizing a utility function when outcomes are known (Samuelson 1938). If outcomes are random, she behaves as if she were maximizing her expected utility (von Neumann and Morgenstern 1944). If outcomes are unknown, she behaves as if maximizing her subjective expected utility with respect to some prior belief over the outcome (Savage 1954). Thus, decision theorists have established a tight link between optimization and these axioms.

I will refer to subjective expected utility (SEU) maximization as *rationalizable* choice.[1] Any behavior that is inconsistent with SEU – and therefore, in violation of some axiom – will be referred to as *irrational*. Notably, rationalizability imposes only weak restrictions on the individual's (Bernoulli) utility function.[2]

In this paper, I introduce an additional axiom to choice under uncertainty: computational tractability. That is, I consider the implications of computational constraints for an individual that faces many choices in her lifetime (or equivalently, a single high-dimensional choice). This assumption is adapted from the theory of computational complexity and, being suitable for computers, is quite weak when applied to humans. In this setting, I ask and answer two questions.

First, can computational tractability refine rationalizability? That is, can we obtain tighter predictions about choice behavior, via restrictions on the Bernoulli utility function? Moreover, can this refinement predict behavioral heuristics observed in practice? The answer is "yes". In a high-dimensional setting, SEU maximization is computationally intractable for most Bernoulli utility functions. Consider behavior that satisfies (a) rationalizability, (b) tractability, (c) monotonicity, and (d) invariance to relabelling of dimensions. I show that such behavior corresponds to the set of additively separable utility functions, and is observationally equivalent to a heuristic known as *narrow choice bracketing* in experimental economics (Read et al. 1999). Absent (d), behavior is still limited to a special class of utility functions, which I characterize using a novel application of a concept from graph theory, the Hadwiger number.

[1] There is no consensus in economics on what constitutes rationality, although it almost always refers to the exact optimization of some objective function. For example, there has been some discussion over whether ambiguity averse preferences should be regarded as rational (Al-Najjar and Weinstein 2009). However, it would be difficult to argue that SEU maximization is not our modal definition. For that reason, I do not view my definition as especially controversial. Furthermore, there is nothing unique to SEU maximization in this framework; one could easily undertake a similar analysis with other definitions of rationality. I suspect the results would have a similar flavor.

[2] A Bernoulli utility function refers to the "utility" in "expected utility"; this describes the agent's preferences over sure outcomes. When I refer to an agent's "utility function" without qualification, I mean the Bernoulli utility function, not the expected utility.

Second, in the presence of computational constraints, is rationalizability still compatible with self interest? That is, consider an individual is rational insofar as she wants to maximize SEU, but her objective is intractable. She has two options. First, she can maintain rationalizability while optimizing an alternate Bernoulli utility function[3]. Second, she can resort to choices that are not rationalizable but approximate her original utility function.

I demonstrate that, in terms of the agent's original utility function, the first option may be undesirable. That is, for an important class of risk-averse utility functions, I show that restricting attention to rationalizable behavior leads to a substantial loss of (approximate) optimality. The notion of a *rationalizability gap* quantifies this loss; not only does a gap exist, but it is unbounded. It follows that an analyst observing this individual's choice behavior should not expect it to be rationalizable, even though the individual is self-interested.

Insofar as high-dimensional choice is ubiquitous – after all, life is high-dimensional, as are many economic models – these results suggest that even highly-sophisticated agents may appear irrational to an analyst that does not take into account computational constraints. Indeed, given the naivety of narrow choice bracketing, sophisticated agents may be *especially* likely to appear irrational. This suggests the need for new definitions of rationality that allow for approximate optimization.

Running Example. An insurance company has n clients. The ith client approaches the firm with a contract h_i that the firm can accept or reject. Contract h_i specifies, as a function of an unknown state θ, the transfers z_i that the firm will receive from client i. Depending on the kind of insurance, θ could describe e.g. the spot price of some asset at a specified date, the rate of mortage default in the coming year, or the intensity of wildfires in the coming summer.

Negative transfers $z_i < 0$ correspond to net payments from the firm to the client. This would hold, for example, in any state where the insurance premium is smaller than the payout. The firm may or may not have pre-existing arrangements with the clients. In the latter case, rejecting contract h_i leads to transfers $z_i = 0$. Otherwise, rejection leads to some default contract.

In order to correctly manage its risk, the firm must solve a combinatorial optimization problem: for each client i, whether or not to accept the contract h_i. These decisions can be made simultaneously; the order in which clients arrive is not inherently relevant. Let us assume that the firm is profit-maximizing and cares about the sum $\pi_i = \sum_{i=1}^n z_i$ of transfers. Note that its utility function $\bar{u}$ might be nonlinear in π_i for various reasons: risk aversion, nonlinear taxation, debt obligations, managerial incentives, bankruptcy costs, etc. For concreteness, suppose that

$$\bar{u}_n(z) = \log\left(\alpha + \sum_{i=1}^n z_i\right) \tag{1}$$

where $\alpha > 0$ is a constant. Finally, the firm has a prior belief over the state space and evaluates a profile of contracts by their subjective expected utility $\mathrm{E}\left[\bar{u}_n(z)\right]$.

[3] Remember: rationalizable choice maximizes *some* utility function, not necessarily *your* utility function

High-Dimensional Choice. The insurance provision problem described above is clearly high-dimensional. There are n clients, where n could be arbitrarily large. Furthermore, we can pose the same question to the same firm when presented with any number $N > n$ clients. In my model, we will relate the n-dimensional choice with the N-dimensional choice by assigning some default value to the "missing" clients $n+1, \dots, N$. For the utility function $\bar{u}_n$ specified above, the default is obviously a transfer of zero. That is,

$$\bar{u}_n(z_1, \dots, z_n) = \bar{u}_N(z_1, \dots, z_n, 0, \dots, 0)$$

This construction relates a firm's preferences across dimensions, and may be regarded as an implicit coherence axiom. It is a minimal modification to the standard model that will allow us to formalize the sense in which an optimization problem becomes difficult as n increases.

Before proceeding with this example, however, let us take a step back. An obvious question has not been addressed. Why bother with high-dimensional choice at all? Or more broadly, why should economists concern themselves with complication?

One easy answer is that many of our models require agents to solve high-dimensional optimization problems.[4] Another easy answer is that, assuming decision theory has empirical content, life requires us to solve high-dimensional optimization problems.[5]

There is a more philosophical answer.[6] Economists have a tendency to discourage complication, under the conviction that simplification will highlight the essential features of a problem at hand. In contrast, computer science often embraces complication – or rather, certain kinds of complication – under the conviction that a problem may not be interesting until it becomes complicated. The results in this paper validate the latter perspective, without invalidating the former. They demonstrate that applying the right amount of stress to models of choice can lead to qualitatively new insights. Of course, applying the "right amount" of stress requires some tact. In his discussion of large worlds, Savage (1954) illustrates the problems that arise when too much stress is applied.

Computational Tractability. Let us return to our protagonist, the insurance company. The firm must decide, for each client i, whether to accept or reject the proposed contract h_i. To make these decisions, it relies on the ingenuity of its employees, as well as state-of-the-art computational re-

[4] Consider the insurance provision problem, or any number of financial problems that require investment decisions across multiple asset classes. Or the multiple-item monopolist's problem from the agent's perspective. Or games on networks, where edges define interactions. Or any number of dynamic models, from extensive-form games to DSGE in macroeconomics. Indeed, preferences over dynamic processes have already received significant attention from decision theorists (e.g. Kreps and Porteus 1978, Epstein and Zin 1989), albeit from a different lens.

[5] Even the coarsest representations of life are high-dimensional. Within the span of a few years, an individual may choose whether to buy a car, where to invest her savings, how to structure her debt, whether to make an expensive purchase, whether to start a family, etc. Whether these decisions are explicitly bound together is immaterial; they are connected implicitly by common sources of uncertainty. For example, the individual cannot not know her career trajectory, yet this is relevant to all the aforementioned decisions. Even if these phenomena are studied in isolation, it is important to understand how factors that arise primarily in high-dimensional settings, like complexity, will reflect themselves in a low-dimensional subsetting.

[6] Here, I am paraphrasing an insightful comment by Jason Hartline.

sources. Can we expect the firm to maximize SEU, i.e. $\mathrm{E}[\bar{u}(z)]$?

The reader may think to him or herself: "it depends on n". However, recall that the firm may be asked to make a choice with respect to any number n of clients. Clearly, for n sufficiently large, even enumerating all the options will be infeasible. Therefore, the answer is simply "no". Having understood this, consider a more tactful question: can we expect the firm to maximize SEU within time constraints that are growing at most polynomially in n?

The answer remains "no". The utility function $\bar{u}$ induces an NP-hard optimization problem (see theorem 1). That is, if the firm could maximize SEU in polynomial time, it could provide efficient solutions to a large class, NP-complete, of decision problems that are notoriously hard. Indeed, such solutions have eluded computer scientists for more than half a century. Their nonexistence is the content of a famous conjecture, $\mathrm{P} \neq \mathrm{NP}$.

More generally, time complexity is an important factor in high-dimensional choice. In response, this paper invokes an axiom of computational tractability. Formally, an agent's choice rule – a map from a menu of options to a choice from said menu – must be executable by a Turing machine with runtime that is at most polynomial in the size of the menu.[7] A Turing machine is an abstract model of computation used to study computational complexity.

As I will elaborate, this definition of tractability is remarkably permissive and relatively conservative. In decision theory, it has previously been employed in Apesteguia and Ballester (2010) and Echenique et al. (2011). Apesteguia and Ballester (2010) consider the problem of an analyst trying to rationalize a choice correspondence, in a model of choice under certainty that allows violations of WARP. They find that the analyst's problem is computationally hard in general. Echenique et al. (2011) consider a question similar to mine, in a classic model of consumer choice. In that setting, they show that tractability is toothless, insofar as any rationalizable choice correspondence is tractable.

This definition is also used routinely in algorithmic game theory. For example, a notable line of work studies the time complexity of finding Nash equilibria (see e.g. Daskalakis et al. 2009). Hardness results suggest that Nash equilibrium is unlikely to arise in complicated games, regardless of which introspective or learning process agents may employ. This approach is appealing precisely because intractability is a property of the problem itself, not a property of the solution.[8]

Representations and Dichotomies. In our running example, I have established that maximizing SEU is computationally intractable for the specified utility function $\bar{u}$. Nonetheless, suppose that the choice behavior of the firm is rationalizable. Specifically, suppose that the firm maximizes SEU with respect to a strictly increasing utility function $u \neq \bar{u}$. Under what conditions on u does the firm's choice rule satisfy tractability?

[7] Computer scientists should note that tractability is a property of the choice rule, and applies even if the choice rule is not explicitly optimizing some objective. Loosely, rationalizability says that the agent is exactly solving an optimization problem; imposing tractability in addition says she is exactly solving a tractable optimization problem.

[8] In contrast, the literature on bounded rationality often uses less permissive models of computation (e.g. Rubinstein 1986, Mandler 2015, Mandler et al. 2012, Jakobsen forthcoming). Of course, I have claimed that the existence of a choice-making algorithm with polynomial runtime is a particularly weak definition of tractability. It may be too weak for many applications.

A rationalizable choice rule is label-invariant if $u(z_1, \ldots, z_n) = u(z_{k_1}, \ldots, z_{k_n})$ for any permutation k. Suppose the firm restricts attention to label-invariant choice rules, a reasonable approach given that $\bar{u}$ is label-invariant and the order in which clients are presented seems immaterial. Then the firm will behave as if she were using a particular algorithm called narrow choice bracketing (NCB). Equivalently, the firm's *revealed* utility function u must be additively separable, i.e.

$$u(z) = \sum_{i=1}^{n} f(z_i) \tag{2}$$

for some strictly increasing function f. Absent label-invariance, the firm's behavior corresponds to a richer class of parameterized dynamic programming algorithms. The corresponding restrictions on u can be characterized in terms of the Hadwiger number of a graph G with n nodes, where edges represent pairwise violations of additive separability in u.

These results are presented in theorems 1 and 2, which demonstrate hardness via reductions from MAX2SAT and MIN2SAT. Respectively, these theorems assume P $\neq$ NP and the non-uniform exponential time hypothesis. In economics, these correspond to *representation theorems*: given a set of axioms, they characterize behavior as the solution to a restricted class of optimization problems. In computer science, these correspond to *dichotomy theorems*: they show that a collection of optimization problems can be split into two complexity classes (e.g. P and NP-hard), and identify which problems belong to each class. Notably, these results apply even when the underlying constraint satisfaction problem is trivial, as in the running example. For instance, they do not rely on budget constraints to make optimization difficult. They hold even if the feasibility of an agent's choice in one dimension is independent of her choice in every other dimension.

The Rationalizability Gap. The previous results establish that rationalizability is with loss of generality, in the sense that the revealed utility function u is restricted (2). Is it also with loss of optimality? That is, in this high-dimensional setting where tractability constraints bind, can the insurance company do better according to $\bar{u}$ by adopting an irrational choice rule? If so, how much better? The answers to these questions are "yes" and "unboundedly".

The approximation ratio $a \in [0, 1]$ of a tractable choice rule is a guarantee. For any set of contracts h, the acceptance/rejection decisions will obtain SEU that is at least an a-fraction of the optimal choice. Roughly speaking, the *rationalizability gap* $RG^n = a/b$ compares the best-possible approximation ratio a to the best approximation ratio b that can be obtained by a rationalizable choice rule.

For the utility function $\bar{u}$, a greedy algorithm obtains a $1/2$-approximation. In contrast, no rationalizable choice rule obtains a constant ϵ-approximation for any $\epsilon > 0$. In this case, the rationalizability gap is unbounded in n. Here is one implication, assuming tractability. For any rationalizable choice rule ϕ, there is an irrational choice rule that performs at least as well as ϕ on every set of contracts and, for any $\Delta > 0$ and sufficiently large n, outperforms ϕ by Δ on some set of n contracts. Therefore, it is in the best interest of the firm to be irrational, in the sense that her choice behavior will not be rationalizable.

This concept is inspired by the *revelation gap* of Feng and Hartline (2018). In auction theory, the revelation principle allows the auctioneer to restrict attention to direct mechanisms, where bidders simply report their valuations. In prior-independent settings, where the auctioneer lacks information about the bidder population, optimal auctions are infeasible and this result no longer applies. Feng and Hartline (2018) prove the existence of a revelation gap for single-item auctions with budgets, when the objective is welfare-maximization. Therefore, in an important class of problems, restricting attention to direct mechanisms is with loss of (approximate) optimality.

Like Feng and Hartline (2018), I ask whether an important result in microeconomics survives when the problem becomes more complicated. In both cases, exact optimization is infeasible. The notion of a gap allows us to quantify the loss of optimality associated with highly-regarded approaches like direct mechanisms and rationalizable choice. The gap will vary from problem to problem; the trick is to identify an important class of problems where the gap is positive or even large. Theorem 3 extends the aforementioned result – the unbounded rationalizability gap – to a larger class of utility functions $\bar{u}$ that satisfy a single-crossing condition and asymptotic sublinearity.[9]

Organization. The paper is organized as follows. I introduce the high-dimensional choice model in section 2, and formally define rationalizable and tractable choice. In sections 3 and 4, I present two representation theorems that characterize the intersection of rationalizability and tractability. In section 5, I consider two implications of these representation theorems: an axiomatic foundation for narrow choice bracketing, and an unbounded rationalizability gap for certain utility functions. Section 6 concludes.

2 Preliminaries

This section adapts a standard model of choice under uncertainty to a high-dimensional setting. Let Θ be a set of *states*. States describe an uncertain feature of the world, e.g. whether a recession occurs in the next year. Let Z be a set of *consequences*. Consequences describe outcomes that the agent cares about, e.g. money. An *act* $h : \Theta \to Z$ maps states to consequences. Acts describe an option available to the agent, with consequences that depend on the realized state, e.g. an insurance contract. A *menu* H describes a set of acts available to the agent.

Let $\mathcal{H}$ be a collection of menus. This describes the universe of decisions that the agent may be faced with. For instance, one menu $H \in \mathcal{H}$ might consist of an insurance contract h with a

[9]This is not the first paper to question the normative foundations of rationality. For example, the literature on the evolution of preferences has established that natural selection can lead to irrational choice rules (see e.g. Robson 1996). Manski (2011) provides another important perspective: the agent's choices in counterfactual menus – menus other than the one she was presented with – cannot affect her utility. Therefore, whether the agent's counterfactual choices are coherent should have not affect our evaluation of her behavior as "good" or "bad". These arguments have substantially influenced this paper, but there are points of contrast. Unlike evolution of preferences, I do not presume knowledge of the process by which preferences are formed. Unlike Manski (2011), I provide a normative argument *against* rationality, and quantify the loss of optimality.

monthly premium of \$100 and a deductible of \$1000, as well as a contract h' with no deductible but a premium of \$200. Another menu H' might combine h with a contract h'' with a premium of \$100 and a 50% copay. In the end, an agent is presented with a single menu. However, a priori she could be presented with any menu $H \in \mathcal{H}$.

The agent's behavior is described by a choice correspondence that maps any menu $H \in \mathcal{H}$ to a subset of acts $h \in H$ in that menu. To be clear, the agent must always express a preference, but it does not need to be unique. For instance, an agent that is indifferent between two acts $h, h' \in H$ may set $\phi(H) = \{h, h'\}$. From a computational perspective, a choice correspondence ϕ describes a problem, which specifies the set of acceptable outputs $\phi(H)$ for any input $H \in \mathcal{H}$. If no efficient algorithm exists for a given ϕ, we may regard that behavior as implausible.

2.1 High-Dimensional Choice

The consequences $Z \subseteq \mathbb{R}^\infty$ are infinite sequences $z = (z_i)_{i=1}^\infty$ of real numbers. There is a *default* consequence $e \in Z$. A consequence z is called n-dimensional if $z_i = e_i$ for all $i > n$. Recall the running example: an n-dimensional consequence z represents transfers from n different clients. Hypothetical clients $i = n+1, \dots, \infty$ are associated with a default transfer of $e_i = 0$.

Treating consequences as infinite sequences allows us to deal with n-dimensional consequences for arbitrarily large n. But our interest is in finite-dimensional choice. That is why an n-dimensional consequence concatenates a finite sequence $z_1, \dots, z_n$ with entries $n+1$ and onward of the default consequence e. The existence of a default acknowledges that even in the absence of a meaningful choice $n+k$, there must be some physical outcome associated with dimension $n+k$.

Assumption 1 (Consequence Richness)**.** *The consequence space includes all n-cubes, i.e.*

$$\forall n \geq 1 : \quad \left[\underline{z}, \bar{z}\right]^n \times \prod_{i=n+1}^{\infty} \{e_i\} \subseteq Z$$

for some real numbers $\underline{z} < \bar{z}$*.*

An n-dimensional act h^n maps states to n-dimensional consequences. An n-dimensional menu H^n is a menu consisting exlusively of n-dimensional acts h^n. The collection $\mathcal{H}^n$ consists of n-dimensional menus. The agent could be presented with menus of varying dimensions n. That is,

$$\mathcal{H} = \bigcup_{n=1}^{\infty} \mathcal{H}^n$$

The following assumption ensures that the state space is sufficiently large.[10]

Assumption 2 (State Richness)**.** *The state space* $\Theta = [0, 1]$ *is the unit interval.*

[10] This is essential for the MAX/MIN2SAT reductions. This does not make it essential for the results, only the proofs. Whether the results extend to a finite state space is left as an open question.

It will be useful to focus on acts whose "effective" state space is finite.

Definition 1 (Regularity)**.** *A regular act h is characterized by an increasing sequence of states $\theta_0, \theta_1, \ldots, \theta_m$ where $\theta_0 = 0$ and $\theta_m = 1$. For any states θ, θ' that belong to the same interval $[\theta_{j-1}, \theta_j)$ for some integer $j = 1, \ldots, m-1$, the consequences $h(\theta) = h(\theta')$ are the same.*

The following assumption ensures that the agent expresses preferences over sufficiently diverse menus. Critically, I do not *require* the agent to solve complicated constrained optimization problems. It is well-known that even optimization under budget constraints is NP-hard (see the KNAPSACK problem). For other constraints, finding a feasible act may be hard (as in e.g. graph coloring), let alone finding an optimal one. The point here is that uncertainty in itself makes optimization difficult, even if we do not invoke constraints that bind feasible choices across dimensions.

Assumption 3 (Menu Richness)**.** *Each collection $\mathcal{H}^n$ includes all binary product menus of regular acts. Formally, a binary submenu H_i consists of at most two subacts $h_i : \Theta \to [\underline{z}, \bar{z}]$. Define*

$$H^n = H_1 \times H_2 \times \ldots \times H_n \times \prod_{i=n+1}^{\infty} \{e_i\}$$

Then the product menu $H^n \in \mathcal{H}^n$.

In addition, each collection $\mathcal{H}^n$ includes all binary menus of regular acts. Formally, if h, h' are n-dimensional regular acts then $\{h, h'\} \in \mathcal{H}^n$.

To be clear, menu richness does not assert that a given agent will be presented with every menu in this collection. Rather, it says that the agent should be capable of expressing a choice $\phi(H)$ in the hypothetical event that she is presented with some menu $H \in \mathcal{H}$.[11]

The following definition of rationalizability requires the agent to maximize her subjective expected utility (SEU) with respect to a uniform prior belief and some Bernoulli utility function.[12]

Assumption 4 (Rationalizability)**.** *There exists a continuous utility function $u : Z \to \mathbb{R}$ such that*

$$\forall H \in \mathcal{H} : \quad \phi(H) = \arg\max_{h \in H} \int_0^1 u(h(x))dx$$

In particular, all acts must be Lebesgue-measurable. This is always true for regular acts.

[11]Researchers will often be interested in models where the space of menus $\mathcal{H}'$ is smaller than $\mathcal{H}$. In that case, it is not enough to establish that optimization in $\mathcal{H}'$ is tractable. Rather, the researcher would also have to argue that evolution would prioritize optimization on menus $\mathcal{H}'$ over some subset of menus $\mathcal{H} \setminus \mathcal{H}'$ in which the agent does not optimize. Keep in mind that the hardness of a computational problem does not come from any particular instance (or menu); it comes from the need to simultaneously optimize over a large space of instances, despite using an algorithm that has a finite description and limited time. So any suboptimal algorithm will feature trade-offs.

[12]The fact that the prior is uniform is convenient but seems entirely inessential. I conjecture that any absolutely continuous prior with full support would suffice. What is essential to the proofs is that the prior does not have finite support, which undermines our assumption that the state space is rich. Finally, one could interpret this as a model of objective expected utility maximization where the state is known to be uniformly-distributed.

For an axiomatic justification of SEU, I refer the reader to the lecture notes by Kreps (1988), or the highly-influential paper by Anscombe and Aumann (1963). The original argument, of course, is due to Savage (1954). The underlying coherence axioms reflect logically-appealing restrictions on choice. For example, by transitivity, if the agent chooses $h = \phi(\{h, h'\})$ and $h' = \phi(\{h', h''\})$, then she must also choose $h = \phi(\{h, h'\})$. To be clear, I do not axiomatize SEU for high-dimensional choice. These references are meant to clarify why rationalizability is associated with SEU.

The following assumption restricts attention to settings (like insurance provision) where consequences are desirable. In terms of notation, let $z \geq \tilde{z}$ if $z_i \geq \tilde{z}_i$ for all dimensions i. Let $z > \tilde{z}$ if $z \geq \tilde{z}$ and there exists a dimension j where $z_j > \tilde{z}_j$.

Assumption 5 (Monotonicity)**.** *The utility functions u and $\bar{u}$ are strictly increasing. That is, $z \geq \tilde{z}$ implies $u(z) \geq u(\tilde{z})$ and $z > \tilde{z}$ implies $u(z) > u(\tilde{z})$.*[13]

This completes the choice-theoretic description of the model.

2.2 Axiom of Computational Tractability

A deterministic Turing machine (TM) is an abstract model of computation that takes in a string of characters and outputs another string. The precise definition will not be important here.[14] However, it *is* important to appreciate the apparent universality of this model. Consider the following discussion by Bernstein and Vazirani (1997):

> Just as the theory of computability has its foundations in the Church-Turing thesis, computational complexity theory rests upon a modern strengthening of this thesis, which asserts that any "reasonable" model of computation can be efficiently simulated on a probabilistic Turing machine (an efficient simulation is one whose running time is bounded by some polynomial in the running time of the simulated machine). Here, we take reasonable to mean in principle physically realizable.

In particular, if natural phenomena (e.g. neurological, social, or evolutionary processes) can be efficiently simulated by a model of computation, the strong Church-Turing thesis implies that impossibility results for TMs are binding on these natural phenomena as well.

The Cobham-Edmonds thesis further asserts a computation is tractable only if its runtime is at most polynomial in the length k of the input string. In other words, any algorithm whose runtime is superpolynomial (e.g. exponential) in k will take an unreasonable amount of time for moderately large k – regardless of how powerful the computer running it is. Clearly, the converse is not true. A computation that requires $k^{2^{100}}$ steps has time complexity $O(\text{poly}(k))$ but is clearly infeasible. Furthermore, even $O(k)$ time complexity can be challenging if the computer is not up to the task, as anyone who has struggled with addition can attest.

[13] The fact that u is strictly increasing (as opposed to weakly) is convenient. However, it is certainly not necessary to prove NP-hardness. For example, the reductions in this paper can easily be applied to the maximum $u(z) = \max\{z_i\}_{i=1}^{\infty}$ and the minimum $u(z) = \min\{z_i\}_{i=1}^{\infty}$.

[14] I refer the interested reader to any textbook on computational complexity (e.g. Arora and Barak 2009).

These two theses – strong Church-Turing and Cobham-Edmonds – constitute a leading view of our physical world.[15] This paper considers their implications for high-dimensional choice. Given that premise, the following axiom is the weakest possible restriction on behavior.

Assumption 6 (Tractability)**.** *There exists a TM M that satisfies the following properties.*

1. *For an input string that describes menu $H \in \mathcal{H}$, the output string describes $h \in \phi(H)$.*
2. *The runtime of M for input $H \in \mathcal{H}$ is at most polynomial in the description length of H.*

This definition of tractability is implicitly worst-case: the upper bound on the runtime is applied uniformly across all instances $H \in \mathcal{H}$. This is the correct definition. In axiomatic decision theory, a representation and its underlying axioms apply to *all* menus in a collection; indeed, researchers lean heavily on this fact in their proofs. In particular, a choice correspondence that maximizes SEU for "most" but not all menus will violate coherence axioms and is *not* rationalizable.[16]

3 Special Representation

Let H be a product menu of regular acts. The description length of H is increasing linearly in (a) the dimension n, (b) the size m of the effective state space, and (c) the number l of acts per submenu. However, the number of acts h in a product menu H is growing exponentially in n. Every possible combination of subacts in each dimension must be accounted for. This means that the brute-force algorithm – i.e. evaluate the payoff of every $h \in H$ and choose the best one – is impractical. Furthermore, the bottleneck is driven by the dimension n, not m or l per se.

In this section, we will see that the brute-force algorithm – although naive and impractical – is likely the best we can do, barring relatively strong restrictions on the utility function u. These restrictions are the content of representation theorems 1 and 2, which characterize the set of utility functions that are consistent with our tractability axiom. These results can also be thought of as dichotomy (or trichotomy) theorems: they partition a set of rationalizable choice correspondences ϕ into two (or three) complexity classes, based on properties of u. In the next section (5.2), I will present algorithms for SEU maximization when u satisfies the relevant properties.

3.1 Complexity Classes

Our results are only meaningful to the extent that several widely-believed conjectures are true. I will introduce these conjectures momentarily. Essentially, they assert that the best known algorithms for

[15]The main challenge to the strong Church-Turing thesis arises from the possibility of quantum computers, which can be formalized as quantum Turing machines (QTM) (Bernstein and Vazirani 1997). Although QTMs may be able to efficiently solve certain problems that are beyond the reach of TMs, it seems significantly less likely that they can efficiently solve NP-hard problems like 3SAT or MAX2SAT. Therefore, while I do not explicitly consider tractability from the perspective of a QTM, there is no obvious reason why the results should differ, except that I would replace conjectures like P $\neq$ NP with an appropriate quantum analog.

[16]To be clear, this irrational behavior appears reasonable and plausible a priori. The entire point of this paper is that alternative theories of choice – dealing with "approximate" or "frequent" optimization – are needed. But we cannot make this point effectively if we modify the existing theory at the same time that we are evaluating it.

certain fundamental problems are, indeed, the best possible. The evidence for these conjectures is straightforward: despite decades of effort by myriad computer scientists, the best known algorithms are essentially no better than brute-force search.[17]

First, note that a central task in computational complexity theory is categorizing problems according to their time complexity. Several complexity classes are defined in terms of *decision problems*, i.e. problems with a binary output. For example, "does graph G have a Hamiltonian path" is a decision problem.

- The class $\text{DTIME}(f(l))$[18] consists of decision problems that can be solved by algorithms with runtime that does not exceed $f(l)$ on any input string of length l.

- The class P consists of decision problems for which polynomial-time TMs exist, i.e.

$$\text{P} = \text{DTIME}(\text{poly}(n))$$

- The class NP[19] consists of decision problems for which polynomial-time verifiers exist. A verifier is a TM that, given a proposed solution, outputs "accept" if the solution is valid and "reject" otherwise. In the previous example, the verifier takes in a sequence p of nodes in G and determines whether p is a Hamiltonian path.

- The class NP-hard consists of problems that are "as hard as" any problem in NP. That is, given a polynomial-time algorithm for *any* NP-hard problem, we could devise a polynomial-time algorithm for *every* problem in NP. An algorithm that takes a solution to one problem and outputs a solution to another problem is known as a reduction.

- The class NP-complete is the intersection of NP and NP-hard. Cook (1971) proved that NP-complete is nonempty. Karp (1972) identified a number of additional NP-complete problems. The NP-complete problems are all polynomial-time reducible to one another.

It is immediate that $\text{P} \subseteq \text{NP}$. It is widely believed that $\text{P} \subsetneq \text{NP}$, but this has never been proven. The conjecture $\text{P} \neq \text{NP}$ captures the belief that NP-complete problems are, in fact, hard.

As defined, the complexity classes DTIME, P, and NP (but not NP-hard) refer to decision problems, but the choice correspondence ϕ is an optimization problem. One can define a decision problem for ϕ, i.e. "does there exist an act $h \in H$ that obtains SEU exceeding some threshold" and it is easy to verify that the decision problem is in NP. Nonetheless, I will abuse notation and interpret e.g. $\phi \in \text{P}$ in the natural way, i.e. the optimization problem ϕ is tractable.[20]

[17] Keep in mind that the definition of "better" is very specific, refering to the asymptotic rates of worst-case runtime. For better or worse, this is the language in which the conjectures are stated. In particular, this is the benchmark that – dare I say – complexity theorists have the greatest incentives to improve upon.

[18] The "D" indicates that we are restricting attention to "deterministic" TMs.

[19] The "N" indicates that we are allowing "non-deterministic" TMs. There are two equivalent definitions of NP: the one given above, and another that refers to non-deterministic TMs.

[20] The distinction between decision vs. optimization only affects my positive results. If the decision problem is NP-complete, then either both versions have polynomial-time algorithms or neither do (Arora and Barak 2009, theorem

It is also convenient to define the exponential time hypothesis (ETH), although we will not refer to this until later on. An important NP-complete problem is known as 3SAT (see appendix B.2 for more details about satisfiability problems). The ETH is a strengthening of P $\neq$ NP that claims there is no subexponential-time algorithm for 3SAT. Formally, it asserts

$$\text{3SAT} \notin \text{DTIME}\left(2^{o(n)}\right)$$

where n is the number of variables in a 3SAT instance.

3.2 Label-Invariant Case

The first representation theorem (1) applies to restricted set of choice correspondences. Consider a setting in which the consequences z_i are interchangeable across dimensions, even if the feasible choices H_i vary across dimensions. This is often a natural assumption if consequences are monetary. For instance, in our running example, the insurance company does not care whether it obtains z_i dollars from client i and z_j dollars from client j, or vice-versa.

Assumption 7 (Label-Invariance)**.** *The utility function u is invariant to finite permutations of z. That is, $u(z) = u(z_{k_1}, \dots, z_{k_n}, z_{n+1}, \dots)$ for any n and permutation k of $z_1, \dots, z_n$.*

Under label-invariance, the boundary between tractable and intractable choice is relatively easy to describe. Narrow choice bracketing (NCB), defined later on (10), is a simple heuristic for SEU maximization. Essentially, the agent optimizes in each dimension i, without considering her choices in dimensions $j \neq i$. The additively separable utility functions are precisely those for which NCB is without loss. As it turns out, NCB is effectively the only tractable behavior consistent with rationalizability, monotonicity, and label-invariance.

Definition 2 (Additive Separability)**.** *A utility function u is additively separable if there exist efficiently computable functons $f_i : \left[\underline{z}, \bar{z}\right] \to \mathbb{R}$ such that*

$$u(z) = \sum_{i=1}^{\infty} f_i(z_i)$$

for all consequences $z \in Z$. Naturally, if u satisfies monotonicity (5) then each f_i is strictly increasing. If u satisfies label-invariance (7) then $f_i = f_j$ for all i, j.

Theorem 1. *Assume richness (1, 2, 3), rationalizability (4), monotonicity (5), label-invariance (7). Then*

2.19). Here is a brief intuition. The decision problem only determines whether there exists a satisfactory act h^*, but the following algorithm can be used to identify h^*. By the Cook-Levin theorem, the input to an NP-complete decision problem can be reformulated as an instance of SAT, i.e. a boolean formula with variables $x_1, \dots, x_k$. If P = NP, then SAT can be solved in polynomial time. Set $x_1 = \texttt{true}$ to obtain a subformula, and determine whether the subformula is satisfiable. If yes, maintain that value. If no, set $x_1 = \texttt{false}$. Move on to x_2 and repeat. If the original formula was satisfiable, this will return the satisfying assignment, which can be used in turn to find h^*.

1. *Assume tractability (6). If* P $\neq$ NP*, then u is additively separable.*

2. *Restrict $\mathcal{H}$ to product menus of regular acts, without violating menu richness (3). If u is additively separable, then ϕ is tractable, i.e. $\phi \in$* P.

The proof is deferred to appendix B.2. For now, I provide a very high-level proof sketch. Suppose u is not additively separable. Assume for contradiction that ϕ is tractable. Let f be a boolean formula with n variables. I construct an n-dimensional menu H^f such that $\phi(H^f)$ gives a solution to the problem known as MAX2SAT on instance f. Thus, if ϕ can be solved in polynomial time, so can MAX2SAT. But MAX2SAT is NP-hard, so this contradicts P $\neq$ NP. Conversely, suppose u is additively separable. Then a heuristic called narrow choice bracketing solves ϕ in polynomial time without loss.

Technically, this is a straightforward corollary of theorem 2, one that uses the structure imposed by label-invariance to obtain a starker result. But pedagogically, the order is reversed: one should understand theorem 1 before theorem 2.[21]

4 General Representation

Absent label-invariance, choice behavior is much richer and the boundary between tractability and intractability is much less clear. In particular, it may not be obvious that there exists a useful vocabulary to describe that boundary. Nonetheless, theorem 2 obtains a nearly-complete characterization of choice correspondences that are rationalizable, tractable, and monotone.

To begin, consider the following counterexamples to theorem 1 sans label-invariance. These utility functions are tractable but not additively separable.

1. $u(z) = \sum_{i \text{ odd}} \sqrt{z_i + z_{i+1}}$
2. $u(z) = \sum_{i=1}^{\infty} \sqrt{z_i + z_{i+1}}$
3. $u(z) = z_1 \cdot \sum_{i=2}^{n} z_i$
4. $u(z) = \log\left(\sum_{i=1}^{100} z_i\right) + \sum_{i=101}^{\infty} z_i$

These examples have an important commonality. Suppose we fix z_{-ij} so that $u(z_i, z_j \mid z_{-ij})$ is effectively a function of z_i and z_j. For most pairs (i, j) of dimensions, this utility function is additively separable in (z_i, z_j). In example 1, this holds for all i, j where i is odd and $j \neq i + 1$ or where j is odd and $i \neq i + 1$. In example 2, this holds for all i, j where $|i - j| > 1$. In example 3, this holds for all i, j where $i, j \neq 1$. In example 4, this holds for all i, j where $i > 100$ or $j > 100$. The next two definitions formalize this idea.

[21] In both cases, the hardness proofs involve polynomial-time reductions from either MAX2SAT or MIN2SAT to ϕ. These reductions are thematically similar, but label-invariance puts the essential arguments into greater relief. On the other hand, label-invariance obscures the positive result (existence of efficient algorithms) by making it trivial.

Definition 3 (Pairwise Violation)**.** *A finite-dimensional consequence $z \in Z$ and quadruple $(a, b, c, d) \in [\underline{z}, \bar{z}]^4$ constitutes a (i, j)-pairwise violation of additive separability if*

$$\begin{aligned} &u\left(\ldots, z_{i-1}, a, z_{i+1}, \ldots, z_{j-1}, b, z_{j+1}, \ldots\right) + u\left(\ldots, z_{i-1}, c, z_{i+1} \ldots, z_{j-1}, d, z_{j+1}, \ldots\right) \\ &\neq u\left(\ldots, z_{i-1}, a, z_{i+1}, \ldots, z_{j-1}, d, z_{j+1}, \ldots\right) + u\left(\ldots, z_{i-1}, c, z_{i+1}, \ldots, z_{j-1}, b, z_{j+1}, \ldots\right) \end{aligned}$$

The utility function is (i, j)-pairwise additively separable if no (i, j)-violation exists.

Remark 1. *If u is (i, j)-pairwise additively separable then*

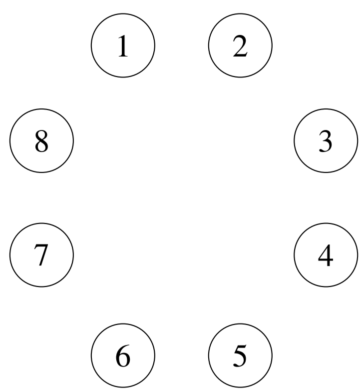
$$u(z) = \alpha(z_{-j}) + \beta(z_{-i}) \tag{3}$$

for some functions α, β.[22] *If u is label-invariant, it is sufficient to verify that there exist no violations with $a = b$ and $c = d$.*[23]

Definition 4 (Violation Graph)**.** *The violation graph G^n consists of n nodes. There is an edge between nodes i, j iff there exists an n-dimensional (i, j)-pairwise violation of additive separability.*

Example 1. *Here are some examples of violation graphs.*

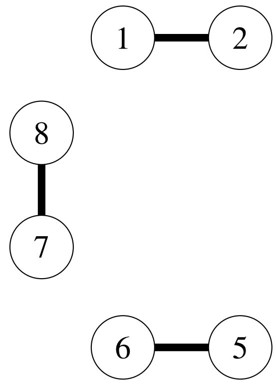
1. *The violation graph G^8 of $u(z) = \sum_{i=1}^{\infty} \sqrt{z_i}$ is empty, i.e.*



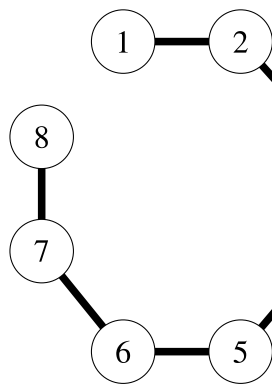
2. *The violation graph G^8 of $u(z) = \sum_{i=1}^{\infty} \sqrt{z_{2i-1} + z_{2i}}$ is*

[22] To see this, set $a = z_i$, $b = z_j$, $c = d = \underline{z}$ in the inequality (now equality) used to define violations.

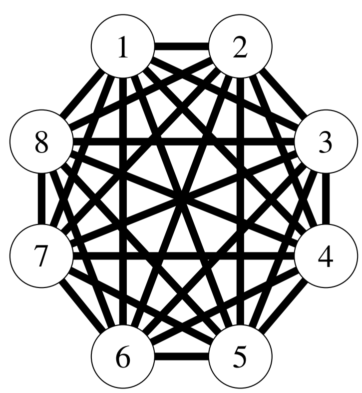
[23] To see this, set $z_i = a = b$ and $z_j = c = d$ in the same inequality (now equality). For a counterexample in the absence of label-invariance, consider $f(a, b) = a + b - \frac{1}{3}(a - b)^3$.



3. *The violation graph* G^8 *of* $u(z) = \sum_{i=1}^{\infty} \sqrt{z_i + z_{i+1}}$ *is*



4. *The violation graph* G^8 *of* $u(z) = \sqrt{\sum_{i=1}^{\infty} z_i}$ *is*



Loosely, I will argue that u induces a tractable choice correspondence ϕ if and only if the violation graph G is sparse. In particular, this clarifies the role that label-invariance played in theorem 1. If u satisfies label-invariance, then G^n is either a sequence of complete graphs or a sequence of empty graphs. If G^n is a sequence of empty graphs, then u is additively separable. By restricting attention to these extremes, we can easily distinguish between sparse and dense graphs, and therefore, between tractable and intractable utility functions.

In contrast, without label-invariance, any well-defined sequence G_n is consistent with *some* utility function. So we cannot avoid the difficult question: what is the right measure of sparsity? Fortunately, it turns out that the appropriate definition was formulated many decades ago by Hadwiger (1943), in his famous conjecture about the chromatic number of graphs.

4.1 Measuring Graph Sparsity

Consider an undirected graph $G = (V, E)$. The *order* of G is the number $|V|$ of nodes, while its *size* is the number $|E|$ of edges. The *degree* of node $i \in V$ is the number of nodes $j \in V$ such that $(i, j) \in E$ is an edge. Let $\delta(G)$ be the *minimal degree* of any node $i \in V$. The *average degree* $\mathrm{avg}(G) = |E|/|V|$ is the ratio of the graph's size to its order.

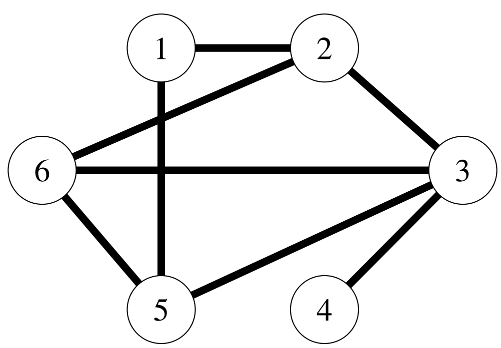
Definition 5 (Minor)**.** *Let G, H be graphs. Then S is a minor of G if it can be formed from G by repeatedly applying the following two operations:*

1. *Deleting a node i and all of its incident edges (i, j).*
2. *Contracting an edge (i, j). This operation deletes nodes i and j and replaces them with the combined node k. Furthermore, it replaces any edges (i, l) and (j, l) with a new edge (k, l).*

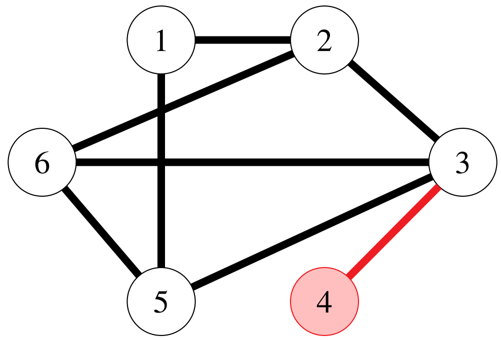
Definition 6 (Degeneracy)**.** *The* degeneracy $\mathrm{dgn}(G)$ *of a graph G is the largest number k such that $\delta(S) \leq k$ for all subgraphs S of G. Furthermore, the* contraction degeneracy $\mathrm{cdgn}(G)$ *is the largest number k such that $\delta(S) \leq k$ for all minors S of G.*

Definition 7 (Hadwiger Number)**.** *The Hadwiger number* $\mathrm{Had}(G)$ *of a graph G is the order of its largest complete minor S (a.k.a. largest clique minor).*

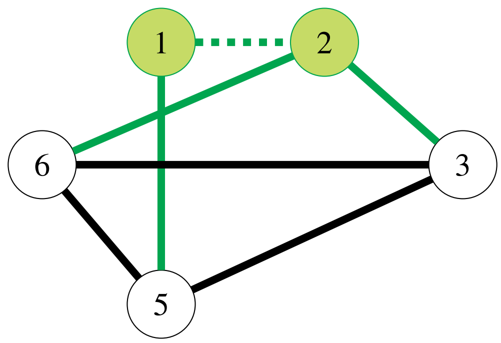
Example 2. *I will illustrate the Hadwiger number graphically. Let G be the following graph.*



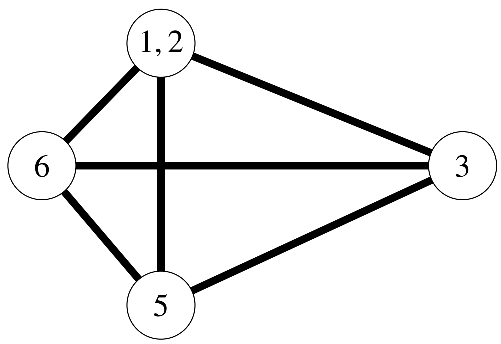
What is $\mathrm{Had}(G)$*? Start by deleting node 4.*



Contract the edge between nodes 1 and 2.



Obtain the following minor of graph G*. Note that it is complete.*



Remember, $\mathrm{Had}(G)$ *is the number of nodes in the largest complete minor. In fact, this is the largest complete minor, so* $\mathrm{Had}(G) = 4$.

It will be useful to define a sequence S_n of large clique minors of G_n.

Definition 8 (Violation Minor)**.** *Let the* violation minors S_n *be a sequence of complete graphs, where* S_n *is a minor of* G_n *and the order of* S_n *approximates the Hadwiger number, i.e.*

$$|S_n| = \Theta\left(\mathrm{Had}(G_n)\right)$$

When describing the minor, each node i *of* S_n *is associated with a subset* V_i *of the nodes of* G_n*, where nodes* $j \in V_i$ *were combined via edge contractions to form node* i.

At this point, I have defined several measures of graph sparsity. The next proposition describes the relationship between them.

Proposition 1. *Let f, g be functions of graphs. Let $f \preceq g$ if $f(G)$ is bounded above by a linear function of $g(G)$, up to log factors, and uniformly across all graphs G. This defines a partial order. The graph measures defined above can be totally-ordered:*

$$\delta(G) \prec \text{avg}(G) \prec \text{dgn}(G) \prec \text{cdgn}(G) \sim \text{Had}(G)$$

4.2 Non-Uniform Complexity

In order for a choice correspondence to be tractable, the violation graph must become very sparse, very quickly as the dimension increases. In particular, the Hadwiger number of G_n may increase at most log-polynomially in n. That is the content of my next result. To prove this, I require knowledge of the violation graph G_n for positive results, and knowledge of the violation minor S_n for hardness results. Both of these objects can be described with poly(n) space. However, it is unclear whether they can always be described in poly(n) time.

Fortunately, by adopting the language of non-uniform complexity, we can largely bypass this issue. In our setting, $\phi \in \mathrm{P/poly}$ means that for any n, there is an efficient algorithm for making choices from n-dimensional menus. One can describe this algorithm using at most poly(n) space. However, describing this algorithm will take more than poly(n) time, unless $\phi \in \mathrm{P}$. Nonetheless, fixing an arbitrarily large n, an agent with unlimited preprocessing time can quickly make choices $h \in \phi(H)$ when faced with n-dimensional menus H.

Next, I state this more formally in the definitions 9 and remark 2. Then, I state theorem 2.

Definition 9 (Advice)**.** *Let the sequence S_l of strings be* advice*. An S_l-advised TM takes in advice S_l and an input string of length l, before returning an output string. Let*

$$\text{DTIME}\left(f(l)\right)/S_l$$

be the class of problems that can be solved in $f(l)$ time by some S_l-advised TM. Define P/S_l *and* NP/S_l *similarly, as the analogs to the complexity classes P and NP for S_l-advised TMs.*

In a slight abuse of notation, when discussing choice problems we will refer to S_n-advised TMs as taking in advice S_n and an n-dimensional menu $H \in \mathcal{H}^n$.

Remark 2 (Non-Uniform Complexity)**.** *The complexity class* $\mathrm{P/poly}$ *consists of problems that can be solved in polynomial time, given some advice string S_l satisfying $|S_l| = O(\text{poly}(l))$. Formally,*

$$\mathrm{P/poly} = \bigcup_{|S_l| = O(\text{poly}(l))} \mathrm{P}/S_l$$

The conjecture $\mathrm{NP} \not\subset \mathrm{P/poly}$ *is stronger than* $\mathrm{P} \neq \mathrm{NP}$*, since $P \subset$* $\mathrm{P/poly}$*. Its failure would imply a partial collapse of the so-called polynomial hierarchy (Karp and Lipton 1980). Similarly, we can* *strengthen the ETH to the* non-uniform ETH*, which states*

$$3\text{SAT} \notin \text{DTIME}\left(2^{o(n)}\right)/\text{poly} \tag{4}$$

One can find several definitions of the non-uniform ETH, but we will stick to (4)*.*

4.3 General Case

I can now state my main theorem, followed by a corollary that is weaker but easier to interpret.

Theorem 2. *Assume richness* (1, 2, 3)*, rationalizability* (4)*, and monotonicity* (5)*. Then*

1. *Assume tractability* (6)*. If* $\text{NP} \nsubseteq \text{P}/H_n$*, then* $\text{Had}(G^n) = o(\text{poly}(n))$*. In particular, this holds under the conjecture* $\text{NP} \not\subset \text{P}/\text{poly}$*.*

2. *Assume tractability* (6)*. If* $3\text{SAT} \notin \text{DTIME}\left(2^{o(n)}\right)/H_n$*, then* $\text{Had}(G^n) = O(\log(\text{poly}(n)))$*. In particular, this holds under the non-uniform ETH.*

3. *Restrict* $\mathcal{H}$ *to product menus of regular acts, without violating menu richness* (3)*. If*

$$\text{Had}(G_n) = O(\log(\text{poly}(n))) \tag{$*$}$$

then $\phi \in \text{P}/G_n$*. In particular, this implies* $\phi \in \text{P}/\text{poly}$*.*

Corollary 1. *Assume richness* (1, 2, 3)*, rationalizability* (4)*, monotonicity* (5)*, tractability* (6)*, and the non-uniform ETH. Then the Bernoulli utility function u satisfies property* ($*$)*.*

Property ($*$) asserts that the violation graph becomes very sparse, very quickly as the dimension increases. To see how this property can be used to distinguish between tractable and intractable utility functions, see the following examples. It is also worth noting that, given a violation graph G^n associated with some utility function u, it is possible to verify property ($*$) in polynomial time (see appendix A). In that sense, it is not too difficult to translate this result into meaningful restrictions on the utility functions that an agent may possess. With that said, the strength of property ($*$) will become clearer in section 5.2, when I use it to prove the existence of a sizable rationalizability gap.

Example 3. *I revisit the examples from earlier.*

1. *Suppose the revealed utility function is*

$$u(z) = \sum_{i=1}^{\infty} \sqrt{z_i}$$

This is additively separable and label-invariant. Therefore, by my first (special representation) theorem, ϕ *is tractable.*

2. *Suppose the (revealed) utility function is*

$$u(z) = \sum_{i=1}^{\infty} \sqrt{z_{2i-1} + z_{2i}}$$

This is not label-invariant, but $\mathrm{Had}(G^n) = 2$*. Therefore, by my second (general representation) theorem,* ϕ *is tractable.*

3. *Suppose the (revealed) utility function is*

$$u(z) = \sum_{i=1}^{\infty} \sqrt{z_i + z_{i+1}}$$

This is not label-invariant, but $\mathrm{Had}(G^n) = 2$*. Therefore, by my second (general representation) theorem,* ϕ *is tractable.*

4. *Suppose the (revealed) utility function is*

$$u(z) = \sqrt{\sum_{i=1}^{\infty} z_i}$$

This is label-invariant but not additively separable. Therefore, by my first (special representation) theorem, ϕ *is not tractable, assuming that* $P \neq NP$*.*

Note that $\mathrm{Had}(G^n) = n$*. As expected, this violates property* $(*)$*.*

The proof is deferred to appendix B.3. For now, I provide a very high-level proof sketch. Suppose property $(*)$ fails. Assume for contradiction that ϕ is tractable. Take the largest complete minor S^n of G^n as advice. Let k be the order of S^n, i.e. the Hadwiger number of G^n. Let f be a boolean formula with k variables. I construct an n-dimensional menu H^f such that $\phi(H^f)$ gives a solution to the problem known as MAX2SAT on instance f. Since ϕ is solvable in poly-time and $k = \omega(\log n)$, MAX2SAT can be solved in subexponential time with advice S^n. Then 3SAT can be solved in subexponential time with polynomial advice, which contradicts non-uniform ETH.

Conversely, suppose property $(*)$ holds. I construct a dynamic programming algorithm that generalizes choice bracketing. It is parameterized by a graph G. If G does not include an edge (i, j), the algorithm ignores any dependence in u between dimensions i and j. I show that, if this algorithm finds itself searching over a d-dimensional set, then there exists a complete minor of G with order d. Therefore, $d \leq \mathrm{Had}(G^n)$. Taking G^n as advice, set $G := G^n$. It is straightforward to show that this algorithm solves ϕ exactly. By property $(*)$, brute-force search over $\mathrm{Had}(G^n)$-dimensional sets takes $O(n)$ time, and the algorithm halts in $\mathrm{poly}(n)$ time.

5 Implications

In this section, I will consider two implications of my representation theorems: an axiomatic foundation for narrow choice bracketing and the existence of a rationalizability gap. To that end, it will be useful to distinguish between the agent's *revealed* utility function u and her *hedonic* utility function $\bar{u}$. The former (defined by assumption 4) corresponds to the utility function that rationalizes the agent's behavior ϕ. The latter corresponds to the agent's true objective function, supposing it exists. In the absence of computational constraints, it is natural to assume $u = \bar{u}$. In their presence, it is entirely plausible that $u \neq \bar{u}$.

I also restrict attention to product menus H of regular acts (the binary menus required by menu richness (3) are easy to recognize and handle using auxilliary algorithms).

5.1 Narrow Choice Bracketing

Theorem 1 can be viewed as a positive result. It establishes that narrow choice bracketing (NCB) is an implication of – rather than a challenge to – rationality in high-dimensional settings. This may be surprising, given that Read et al. (1999) proposed choice bracketing as a way to explain apparently irrational behavior observed in experiments. We refer to their definition:

> [Choice bracketing] designates the grouping of individual choices together into sets. A set of choices are bracketed together when they are made by taking into account the effect of each choice on all other choices in the set, but not on choices outside of the set. When the sets are small, containing one or very few choices, we say that bracketing is narrow, while when the sets are large, we say that it is broad. Broad bracketing allows people to consider all the hedonic consequences of their actions, and hence promotes utility maximization. Narrow bracketing, on the other hand, is like fighting a war one battle at a time with no overall guiding strategy, and it can have similar consequences.

In our setting, we identify NCB with the following heuristic algorithm.

Definition 10 (NCB Algorithm)**.** *Specify a utility function u. Given a product menu H, the* narrow choice bracketing *(NCB) algorithm proceeds as follows:*

1. *Iterate $i = 1, \ldots, n$.*
   (a) *Define*
   $$\hat{h}_i \in \arg\max_{h_i \in H_i} \int_0^1 u\left(h_i(x), \underline{z}, \ldots, \underline{z}\right) dx$$
2. *Return act $\hat{h}$, which belongs to the menu H by construction.*

For many objectives $\bar{u}$, the NCB algorithm will fail to optimize SEU. Of course, as Rabin and Weizsäcker (2009) observe, there are utility functions $\bar{u}$ for which NCB is without loss of optimality. In our setting, this occurs when $\bar{u}$ is additively separable. Furthemore, NCB can always be rationalized using a revealed utility function u that is additively separable.[24]

Assuming that the revealed utility function u exists, is strictly increasing, and is label-invariant, theorem 1 implied that the tractability is equivalent to additive separability of u. Furthermore, NCB is without loss of optimality if and only if u is additively separable. In that sense, tractability is observationally equivalent to NCB.

Having provided an axiomatic foundation for NCB, it is natural to ask whether this behavior is observed in practice. Indeed, there is considerable experimental evidence that choice bracketing is a feature of human decision-making (Andersen et al. 2018; Andreoni et al. 2018; Martin 2017; Rabin and Weizsäcker 2009; Read et al. 1999). Some evidence and interpretations tend towards framing effects, where the type of bracketing depends on the framing of the problem (Brown et al. 2017; Haisley et al. 2008; Moher and Koehler 2010). Others tend towards bounded rationality, where bracketing is a response to the complexity of combinatorial optimization (Brown et al. 2017; Koch and Nafziger 2019; Pennings et al. 2008; Simonsohn and Gino 2013; Stracke et al. 2017).

Finally, while the assumption of label-invariance is not appropriate for many settings, it may be more plausible in experimental settings where consequences z_i are monetary and dimensions $i = 1, \ldots, n$ correspond to different gambling problems (as in e.g. Rabin and Weizsäcker 2009).

5.2 The Limits of Rationality

Here, our goal is to better understand the limitations of rationality in the presence of time constraints. It is obvious from the representation theorems that optimizing certain objective functions is difficult. The best the agent can hope for is a tractable choice rule that is approximately optimal in some sense. However, it is not obvious whether this choice rule will be rationalizable or not. To be more precise, although optimization is infeasible, certain tractable choice rules may deliver better approximations to the optimum than others. The rationalizability gap quantifies how much better the agent can do if she is willing to use a choice rule that is not rationalizable.

Let $\bar{\phi}$ be the choice correspondence that is optimal for $\bar{u}$. Throughout this section, I make several assumptions for convenience. First, $\bar{u}$ can be efficiently computed in the sense of proposition 3. Second, $\bar{u}$ is monotone (5). Third, $\bar{u}^n(\underline{z}, \ldots, \underline{z}) = 0$. Fourth, $\underline{z} \leq 0$.

Definition 11 (Approximation)**.** *Let $\delta \in [0, 1]$. A choice correspondence ϕ is a $\delta(n)$-*approximation *for $\bar{\phi}$ if*

$$\forall H \in \mathcal{H}^n : \quad \int_0^1 \bar{u}\left([\phi(H)](x)\right) dx \geq \delta(n) \int_0^1 \bar{u}\left([\bar{\phi}(H)](x)\right) dx$$

The approximation ratio $\mathrm{APX}^n(\phi \mid \bar{u})$ *is at least $\Omega(\delta(n))$ if there exists a $\delta(n)$-approximation for $\bar{\phi}$.*

I define the rationalizability gap. Let Φ be the set of choice correspondences that are rich (1, 2, 3) and tractable (6). Let $\Phi_R \subseteq \Phi$ consist of $\phi \in \Phi$ that are rationalizable (4) and monotone

[24]The reason why NCB may be thought of as "irrational" is because it is rationalized by an additively separable u even though we believe that the true objective $\bar{u}$ is not additively separable. But from the perspective of revealed preference, rationalizability is what matters.

(5). Recall that theorem 2 characterized Φ_R in terms of violation graphs. In contrast, Φ contains a much richer expanse of irrational choice correspondences.

If exact optimization of hedonic SEU is tractable, then $\bar{\phi} \in \Phi_R$. Trivially, restricting attention to $\phi \in \Phi_R$ is without loss of optimality. However, if $\bar{\phi}$ is intractable, then $\bar{\phi} \notin \Phi$ and the agent's problem becomes more interesting. Should she exactly optimize an alternative utility function $u \neq \bar{u}$? Or should she resort to irrational choice in the hopes of obtaining a better approximation to her true objective $\bar{u}$? To answer this question, we must first quantify the (possible) loss of optimality associated with restricting attention to Φ_R.

A natural way to define the rationalizability gap is as the ratio of the best approximation ratio obtained by $\phi \in \Phi$ to the best approximation ratio obtained by $\phi \in \Phi_R$. Formally, this would correspond to

$$\frac{\sup_{\phi \in \Phi} \mathrm{APX}^n(\phi \mid \bar{u})}{\sup_{\phi \in \Phi_R} \mathrm{APX}^n(\phi \mid \bar{u})}$$

However, the reader may object to the use of approximation ratios to rank choice correspondences, on the grounds that the worst-case criterion is too pessimistic. For this reason, I strengthen the requirement. I say that a rationalizability gap exists if for every rationalizable choice correspondence, there exists an irrational choice correspondence that (a) weakly dominates it and (b) obtains a strictly better approximation ratio. In that sense, weak dominance is used to rank algorithms, and the worst-case approximation ratio is only used to quantify the gap.

To define this formally, I require additional notation. For a given choice rule ϕ, let $\Phi^*(\phi, \bar{u}) \subseteq \Phi$ be the set of tractable choice rules that perform at least as well as ϕ on any menu $H \in \mathcal{H}$. So, any choice correspondence in $\Phi^*(\phi, \bar{u})$ is clearly at least as good as ϕ from the perspective of an agent with hedonic utility function $\bar{u}$.

Definition 12 (Rationalizability Gap)**.** *The* rationalizability gap *is defined as*

$$\mathrm{RG}^n(\bar{u}) = \inf_{\phi \in \Phi_R} \frac{\sup_{\phi' \in \Phi^*(\phi, \bar{u})} \mathrm{APX}^n(\phi' \mid \bar{u})}{\mathrm{APX}^n(\phi \mid \bar{u})}$$

To show that the rationalizability gap is sizable (i.e. unbounded in n), I need to prove two claims. First, I will show that there exists a decent approximation algorithm for certain objectives $\bar{u}$; namely, the greedy algorithm. Afterwards, I will show that no rationalizable algorithm obtains a decent approximation, for certain objectives $\bar{u}$.

5.2.1 The Greedy Algorithm

Next, I describe an irrational greedy algorithm. This can be thought of as a modified version of the NCB algorithm. Instead of optimizing in each dimension myopically with respect to the her beliefs, the agent allows her beliefs to change across dimensions. The agent puts less probability on states where the agent has already secured good outcomes by her choices in previous dimensions. Thus, the agent's behavior is not consistent with SEU maximization for a fixed belief.

Definition 13 (Greedy Algorithm)**.** *Specify a utility function u. Given a product menu $\boldsymbol{H}$, the algorithm proceeds as follows:*

1. *Initialize the act $\hat{h}$ by setting $\hat{h}_i(\theta) = \underline{z}$ for all dimensions i and states θ.*
2. *Iterate $i = 1, \ldots, n$.*
   (a) *Redefine*
   $$\hat{h}_i \in \arg\max_{h_i \in H_i} \int_0^1 u\left(h_i(x), \hat{h}_{-i}(x)\right) dx$$
3. *Return act $\hat{h}$, which belongs to the menu $\boldsymbol{H}$ by construction.*

This generalizes Johnson's (1974) approximation algorithm for MAXSAT. Loosely, MAXSAT corresponds to the utility function $\bar{u}(z) = \max_i z_i$. However, the algorithm performs well for a much larger class of utility functions, including a variety of risk-averse utility functions where

$$\bar{u}(z) = g\left(\sum_{i=1}^{n} z_i\right)$$

where g is strictly increasing and concave.

Definition 14 (SCC)**.** *The utility function $\bar{u}$ satisfies the* single-crossing condition *(SCC) if*

$$u(z) - u(z') \geq u(z + z'') - u(z' + z'')$$

for any finite-dimensional consequences $z, z', z'' \in Z$ where $z'' \geq \vec{0}$.

Proposition 2. *Let $\bar{u}$ satisfy the SCC (14). Then the greedy algorithm is $1/2$-approximation for $\bar{\phi}$.*

5.2.2 The Rationalizability Gap

I claim that the rationalizability gap is unbounded if (a) $\bar{u}$ satisfies the SCC, (b) $\bar{u}^n(1, \ldots, 1)$ is sublinear, and (c) 3SAT is computationally hard. For intuition, suppose we were to restrict attention to hedonic utility functions of the following form

$$\bar{u}(z) = g\left(\sum_{i=1}^{n} z_i\right)$$

In this case, sublinearity roughly says that $g(n) = o(n)$, i.e. g is growing at a sublinear rate. This is true of most strictly concave functions g, but not all. For example, if the derivative $g'(x)$ converges to a constant, then g may be strictly concave but asymptotically linear.

The following theorem provides three ways of formalizing this claim based on different interpretations of earlier results (theorem 1 and corollary 2).

Theorem 3. *Let $\bar{u}$ satisfy the SCC (14). The rationalizability gap is unbounded, i.e.* $\mathrm{RG}^n(\bar{u}) = \omega(1)$*, if any of the following conditions hold:*

1. $\mathrm{NP} \not\subset \mathrm{P/poly}$ *and there exists some* $\epsilon > 0$ *such that* $\bar{u}^n(\bar{z}, \ldots, \bar{z}) = O(n^{1-\epsilon})$.
2. *The non-uniform ETH holds and* $\bar{u}^n(\bar{z}, \ldots, \bar{z}) = o\left(\frac{n}{\log(\mathrm{poly}(n))}\right)$.
3. $\mathrm{P} \neq \mathrm{NP}$, $\bar{u}^n(\bar{z}, \ldots, \bar{z}) = o(n)$*, and* ϕ *satisfies label-invariance* (7).

The proof is deferred to the appendix. For now, I will provide a high-level sketch. First, we already showed that if $\bar{u}$ satisfies the single-crossing condition, the irrational greedy algorithm ψ guarantees a $1/2$-approximation. Therefore, every $\phi \in \Phi_R$ is weakly dominated by an tractable, irrational alternative that guarantees at least half of the optimal SEU. This alternative is quite simple: choose the better of $\phi(H)$ and $\phi_G(H)$.

Next, we want to show that if $\bar{u}$ is sublinear, there is no $\epsilon > 0$ such that any rationalizable $\phi \in \Phi_R$ guarantees a constant ϵ-approximation. Let G^n be the revealed violation graph of ϕ. By the general representation, $\mathrm{Had}(G^n) = o(\mathrm{poly}(n))$ if NP $\not\subset$ P/poly. This implies that G^n has a subpolynomial chromatic number. I construct a menu H where (a) ϕ appears to be narrowly bracketing dimensions associated with nodes of a certain color, and (b) this causes ϕ to severely underperform.

6 Conclusion

These results call into question the credibility of economic models that invoke exact optimization, but they also point to new opportunities for decision theory. First, there is a need for credible and useful definitions of approximate optimization. Extant definitions may not be ideal for characterizing human behavior. Second, high-dimensional choice is an underexplored frontier, where existing theory can make new predictions (narrow choice bracketing), old intuitions can be overturned (the rationalizability gap), and entirely new questions can be posed.

This draft is preliminary. Some priorities for further work include (a) allowing non-uniform prior distributions, (b) relaxing the monotonicity of the Bernoulli utility function, (c) seeing whether these results can be used to motivate separable time preferences and models of discounting; (d) seeing whether these results can be used to alleviate a curse of dimensionality when using choice data to estimate an agent's Bernoulli utility function; (e) showing that my theorems still hold even if you relax tractability to the existence of a polynomial-time approximation scheme (PTAS).

References

Alon, N., Lingas, A., & Wahlén, M. (2007). Approximating the maximum clique minor and some subgraph homeomorphism problems. *Theoretical Computer Science*, *374*(1), 149–158.

Andersen, S., Cox, J. C., Harrison, G. W., Lau, M. I., Rutström, E. E., & Sadiraj, V. (2018, December). Asset integration and attitudes to risk: theory and evidence. *The Review of Economics and Statistics*, *100*(5), 816–830.

Andreoni, J., Gravert, C., Kuhn, M. A., Saccardo, S., & Yang, Y. (2018, November). *Arbitrage or narrow bracketing? on using money to measure intertemporal preferences* (Working Paper No. 25232). National Bureau of Economic Research.

Anscombe, F. J. & Aumann, R. J. (1963). A definition of subjective probability. *The Annals of Mathematical Statistics*, *34*(1), 199–205.

Apesteguia, J. & Ballester, M. A. (2010). The computational complexity of rationalizing behavior. *Journal of Mathematical Economics*, *46*(3), 356–363.

Arora, S. & Barak, B. (2009). *Computational complexity: a modern approach*. Cambridge University Press.

Bernstein, E. & Vazirani, U. (1997). Quantum complexity theory. *SIAM Journal on Computing*, *26*(5), 1411–1473.

Brown, J. R., Kapteyn, A., Luttmer, E. F., Mitchell, O. S., & Samek, A. (2017, December). *Behavioral impediments to valuing annuities: evidence on the effects of complexity and choice bracketing* (Working Paper No. 24101). National Bureau of Economic Research.

Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the third annual acm symposium on theory of computing* (pp. 151–158). STOC '71. Shaker Heights, Ohio, USA: ACM.

Daskalakis, C., Goldberg, P. W., & Papadimitriou, C. H. (2009). The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, *39*(1), 195–259.

Echenique, F., Golovin, D., & Wierman, A. (2011). A revealed preference approach to computational complexity in economics. In *Proceedings of the 12th acm conference on electronic commerce* (pp. 101–110). EC '11. San Jose, California, USA: Association for Computing Machinery.

Epstein, L. G. & Zin, S. E. (1989). Substitution, risk aversion, and the temporal behavior of consumption and asset returns: a theoretical framework. *Econometrica*, *57*(4), 937–969.

Feng, Y. & Hartline, J. D. (2018, October). *An end-to-end argument in mechanism design (prior-independent auctions for budgeted agents)*. Los Alamitos, CA, USA: IEEE Computer Society.

Hadwiger, H. (1943). Über eine klassifikation der streckenkomplexe. *Vierteljahrsschr Naturforsch, Ges. Zurich*, *88*, 133–142.

Haisley, E., Mostafa, R., & Loewenstein, G. (2008, August). Myopic risk-seeking: the impact of narrow decision bracketing on lottery play. *Journal of Risk and Uncertainty*, *37*(1), 57–75.

Jakobsen, A. M. (forthcoming). A model of complex contracts. *American Economic Review*.

Johnson, D. S. (1974). Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, *9*(3), 256–278.

Karp, R. M. (1972). Reducibility among combinatorial problems. In R. E. Miller, J. W. Thatcher, & J. D. Bohlinger (Eds.), *Complexity of computer computations: proceedings of a symposium on the complexity of computer computations* (pp. 85–103). Boston, MA: Springer US.

Karp, R. M. & Lipton, R. J. (1980). Some connections between nonuniform and uniform complexity classes. In *Proceedings of the twelfth annual acm symposium on theory of computing* (pp. 302–309). STOC '80. Los Angeles, California, USA: Association for Computing Machinery.

Koch, A. K. & Nafziger, J. (2019). Correlates of narrow bracketing. *The Scandinavian Journal of Economics*, *121*(4), 1441–1472.

Kohli, R., Krishnamurti, R., & Mirchandani, P. (1994, May). The minimum satisfiability problem. *SIAM J. Discret. Math.* *7*(2), 275–283.

Kostochka, A. V. (1984). Lower bound of the hadwiger number of graphs by their average degree. *Combinatorica*, *4*(4), 307–316.

Kreps, D. M. (1988). *Notes on the theory of choice*. Westview Press.

Kreps, D. M. & Porteus, E. L. (1978). Temporal resolution of uncertainty and dynamic choice theory. *Econometrica*, *46*(1), 185–200.

Mandler, M. (2015). Rational agents are the quickest. *Journal of Economic Theory*, *155*, 206–233.

Mandler, M., Manzini, P., & Mariotti, M. (2012). A million answers to twenty questions: choosing by checklist. *Journal of Economic Theory*, *147*(1), 71–92.

Manski, C. F. (2011, August). Actualist rationality. *Theory and Decision*, *71*(2), 195–210.

Martin, V. (2017). When to quit: narrow bracketing and reference dependence in taxi drivers. *Journal of Economic Behavior & Organization*, *144*, 166–187.

Moher, E. & Koehler, D. J. (2010, August). Bracketing effects on risk tolerance: Generalizability and underlying mechanisms. *Judgment and Decision Making*, *5*(5), 339–346.

Al-Najjar, N. I. & Weinstein, J. (2009). The ambiguity aversion literature: a critical assessment. *Economics and Philosophy*, *25*(3), 249–284.

Papadimitriou, C. H. & Yannakakis, M. (1994). On complexity as bounded rationality (extended abstract). In *Proceedings of the twenty-sixth annual acm symposium on theory of computing* (pp. 726–733). STOC ’94. Montreal, Quebec, Canada: Association for Computing Machinery.

Pennings, J. M., Isengildina-Massa, O., Irwin, S. H., Garcia, P., & Good, D. L. (2008). Producers’ complex risk management choices. *Agribusiness*, *24*(1), 31–54.

Rabin, M. & Weizsäcker, G. (2009, September). Narrow bracketing and dominated choices. *American Economic Review*, *99*(4), 1508–43.

Read, D., Loewenstein, G., & Rabin, M. (1999, December). Choice bracketing. *Journal of Risk and Uncertainty*, *19*(1), 171–197.

Robson, A. J. (1996). A biological basis for expected and non-expected utility. *Journal of Economic Theory*, *68*(2), 397–424.

Rubinstein, A. (1986). Finite automata play the repeated prisoner’s dilemma. *Journal of Economic Theory*, *39*(1), 83–96.

Samuelson, P. A. (1938). A note on the pure theory of consumer’s behaviour. *Economica*, *5*(17), 61–71.

Savage, L. J. (1954). *The Foundations of Statistics*. Wiley Publications in Statistics.

Simonsohn, U. & Gino, F. (2013). Daily horizons: evidence of narrow bracketing in judgment from 10 years of m.b.a. admissions interviews. *Psychological Science*, *24*(2), 219–224.

Stracke, R., Kerschbamer, R., & Sunde, U. (2017). Coping with complexity: experimental evidence for narrow bracketing in multi-stage contests. *European Economic Review*, *98*, 264–281.

Szekeres, G. & Wilf, H. S. (1968). An inequality for the chromatic number of a graph. *Journal of Combinatorial Theory*, *4*(1), 1–3.

von Neumann, J. & Morgenstern, O. (1944). *Theory of games and economic behavior*. Princeton University Press.

A Addendum to Section 3

A.1 Comments on Theorem 2

Remark 3. *As mentioned earlier, theorem 2 is a nearly-complete characterization. Here is a complete characterization, with additional assumptions. Let the choice correspondence ϕ be rich* (1, 2, 3)*, rational* (4)*, and monotone* (5)*. Furthermore, suppose one of the following:*

1. *The sequence (G_n, H_n) is efficiently computable and the ETH holds.*
2. *The sequence G_n is efficiently computable and the non-uniform ETH holds.*

Then the Hadwiger numbers of the violation graphs determine whether ϕ is tractable.

Remark 4. *When a researcher proposes a utility function u in an economic model, they may wonder whether SEU maximization is a credible assumption. Theorem 2 suggests the following procedure for evaluating credibility. First, the researcher should describe the violation graph G_n for some dimension n that is appropriate for their setting. Second, they should specify an efficiently computable function $f(n) = O(\log(\text{poly}(n)))$. Third, decide whether $\text{Had}(G_n) \leq f(n)$. This can be done in $\text{poly}(n)$ time. If the answer to the third step is "yes", there is evidence that a sophisticated agent could optimize exactly. The alternative, if the answer is "no", suggests the opposite.*

Suppose the sequence G_n is efficiently computable, but not H_n.If $\text{P} \neq \text{NP}$ but $\text{NP} \subset \text{P/poly}$, the tractability result (theorem 2.3) still applies and the intractability result (theorem 2.1) can be partially recovered (corollary 2.1). Likewise, if the ETH holds but the non-uniform ETH fails, the tractability result (theorem 2.3) still applies and the intractability result (theorem 2.2) can be partially recovered (corollary 2.2).

Corollary 2 (to the proof of Theorem 2)**.** *Assume richness* (1, 2, 3)*, rationality* (4)*, monotonicity* (5)*, and tractability* (6)*. Then*

1. *If $\text{NP} \not\subseteq \text{P}/G_n$, then $\text{dgn}(G^n) = o(\text{poly}(n))$.*
2. *If $\text{3SAT} \notin \text{DTIME}\left(2^{o(n)}\right)/G_n$, then $\text{dgn}(G^n) = O(\log(\text{poly}(n)))$.*

Proof. This follows from the fact that a graph G_n of degeneracy k has a subgraph S (the k-core) with average degree no less than k. In particular, S must have a clique minor S' of order

$$\Omega\left(k/\sqrt{\log k}\right)$$

(Kostochka 1984). Furthermore, H' can be found in $O(\text{poly}(n))$ time (Alon et al. 2007). Finally, since S' is a minor of S and S is a subgraph of G_n, S' is also a minor of G_n. Therefore, we can substitute the violation minor S_n with S' in the proof of theorem 2. As long as k grows sufficiently quickly in n, the reduction from MAX2SAT is unaffected. □

Recall from proposition 1 that the degeneracy is bounded above (in order) by the Hadwiger number. On the other hand, there are graph sequences G_n where the Hadwiger number increases at a poly(n) rate but the degeneracy remains constant. Therefore, the conclusions of this corollary are weaker than the conclusions of theorem 2. Nonetheless, the corollary is strong enough to establish a rationalizability gap in section 5.2. It also implies theorem 1.1 and may be useful in deriving similar results, where label-invariance is replaced with a weaker assumption.

B Omitted Proofs in Sections 3 and 4

The following observation will be useful. If a rational, monotone choice correspondence ϕ is tractable, then the utility function u is efficiently computable (up to affine transformations).

Proposition 3. *Assume richness (1, 2, 3), rationality (4), monotonicity (5), and tractability (6). Then u is efficiently computable in the following sense. There exists a TM M that takes in a constant $\epsilon > 0$ and an n-tuple $(z_1, \ldots, z_n) \in [\underline{z}, \bar{z}]$, and then outputs $x \in \mathbb{R}$ where*

$$x - \epsilon \le \frac{u^n(z_1, \ldots, z_n) - u^n\left(\underline{z}, \ldots, \underline{z}\right)}{u^n\left(\bar{z}, \ldots, \bar{z}\right) - u^n\left(\underline{z}, \ldots, \underline{z}\right)} \le x + \epsilon \tag{5}$$

with runtime $O(\text{poly}(n, 1/\epsilon))$.

Proof. Let $z = (z_1, \ldots, z_n, e_{n+1}, e_{n+2}, \ldots)$. Define a constant n-dimensional act h^z where $h^z(\theta) = z$ for all $\theta \in [0, 1]$. Define an regular n-dimensional act h^x where $h^x(\theta) = \left(\underline{z}, \ldots, \underline{z}, e_{n+1}, e_{n+2}, \ldots\right)$ for $\theta \le x$ and $h^\alpha(\theta) = \left(\bar{z}, \ldots, \bar{z}, e_{n+1}, e_{n+2}, \ldots\right)$ for $\theta > x$. Let M_ϕ be the TM referred to in the tractability axiom (6). Define

$$I_x = \mathbf{1}\left(h^\alpha = M_\phi\left(\{h^z, h^x\}\right)\right)$$

Redefine $\epsilon := \min\{\epsilon, 1\}$, let $k = \lfloor 1/\epsilon \rfloor$, and construct a grid

$$X = \{\epsilon, 2\epsilon, \ldots, (k-1)\epsilon, k\epsilon\}$$

Iterate over all $x \in X$. Output the smallest x such that $I_x = 1$. □

B.1 Satisfiability

Let $x_1, \ldots, x_n$ be boolean variables. Let $c_1, \ldots, c_m$ be clauses of up to n literals (i.e. x_i or $\neg x_i$). An assignment gives a truth value to each x_i. A clause is satisfied if at least one literal is true. For example, if $n = 3$, $c_1 = (x_1 \vee x_2)$, and $c_2 = (\neg x_1 \vee x_3)$, then the solution $(\mathtt{true}, \mathtt{true}, \mathtt{true})$ satisfies both clauses while $(\mathtt{false}, \mathtt{false}, \mathtt{false})$ only satisfies clause c_2.

If each clause is restricted to two literals, MAX2SAT requires us to find an assignment that maximizes the number of satisfied clauses. This problem is NP-hard (Johnson 1974). Similarly, if each clause is restricted to two literals, MIN2SAT requires us to find an assignment that minimizes the number of satisfied clauses. This problem is also NP-hard (Kohli et al. 1994).

B.2 Proof of Theorem 1

Assume richness (1, 2, 3), rationality (4), monotonicity (5), tractability (6), and label-invariance (7). I want to show that if u is not additively separable, then $\phi \in$ NP-hard.

Since u is not additively separable, it must possess an (i, j)-pairwise violation of additive separability. By label-invariance and remark 1, there is an N-dimensional consequence z and a pair (a, b) such that

$$u^2(a, a) + u^2(b, b) \neq u^2(a, b) + u^2(b, a) \tag{6}$$

where we define $u^2 : \left[\underline{z}, \overline{z}\right] \to \mathbb{R}$ as follows:

$$u^2(x, y) := u\left(\ldots, z_{i-1}, x, z_{i+1}, \ldots, z_{j-1}, y, z_{j+1}, \ldots\right)$$

By label-invariance, u is not affected by rearranging the entries (x, y) and z. Therefore, this one (i, j)-pairwise violation easily leads to (k, l)-pairwise violation for dimensions k, l. This includes dimensions $k, l > N$ that exceed the dimensionality of the consequence z. Likewise, the function u^2 is not affected by the choice of (k, l). We can describe z, a, b in $O(N)$ time, evaluate $u^2(x, y)$ in $O(N)$ time by proposition 3, and rearrange dimensions in $O(n)$ time. Since N is a property of u and not affected by the dimension n of the instance H, these operations are all tractable.

The next two lemmas describe reductions from MAX2SAT and MIN2SAT to ϕ, respectively, depending on the direction of inequality (6).

Lemma 1. *The choice correspondence ϕ is NP-hard if*

$$u^2(a, a) + u^2(b, b) > u^2(a, b) + u^2(b, a) \tag{7}$$

Proof. I show that if the choice correspondence ϕ^n were tractable, then we could solve MAX2SAT in polynomial time. First, recall that the objective for MAX2SAT is

$$\max_{x_1, \ldots, x_n} \sum_{j=1}^{m} \mathbf{1}(c_j = \mathrm{true}) \tag{8}$$

where

$$c_j = (x_{j1} \vee x_{j2}) = (x_{j1} \wedge x_{j2}) \vee (\neg x_{j1} \wedge x_{j2}) \vee (x_{j1} \wedge \neg x_{j2}) \tag{9}$$

Our goal is to construct a menu H such that, given the solution $\phi(H)$, we could solve program (8) in poly(n, m). For every clause c_j, create three states j_1, j_2, j_3 (i.e. create three isometric intervals used to define a regular act). For every variable x_i, create a dimension i. Let each submenu H_i consist of a two acts: h_i^T, which indicates $x_i = \mathrm{true}$, and h_i^F, which indicates $x_i = \mathrm{false}$. We defer the definition of these objects until later. For now, define a menu $H = H_1 \times \ldots \times H_n$ where an act h corresponds to an assignment x as described.

As written, the objective for BDT is

$$\max_{h \in H} \sum_{j=1}^{m} \sum_{k=1}^{3} u(h_1(j_k), \ldots, h_n(j_k))$$

Clearly, the following condition is sufficient for our purposes:

$$\sum_{k=1}^{3} u\left(h_1^{x_1}(j_k), \ldots, h_n^{x_n}(j_k)\right) \propto \underbrace{\mathbf{1}\left((x_{j1} \wedge x_{j2}) \vee (\neg x_{j1} \wedge x_{j2}) \vee (x_{j1} \wedge \neg x_{j2})\right)}_{\mathbf{1}(c_j = \text{true})} \tag{10}$$

assuming that the proportionality constants are the same for all clauses j, which will turn out to be true for my construction.

Let us simplify this condition. On the left-hand side, we refer to all n variables whereas on the right-hand side we refer only to two variables. Let z be the consequence used to define u^2. As discussed earlier, we can rearrange the consequence as needed so that the arguments (x, y) to $u^2(x, y)$ correspond to dimensions $j1, j2$ of u. If $x_i \notin c_j$ and $\neg x_i \notin c_j$, set $h_i(j_1) = h_i(j_2) = h_i(j_3) = z_i$. Then, we can rewrite the left-hand side as

$$\sum_{k=1}^{3} u^2\left(h_{j1}^{x_{j1}}(j_k), h_{j2}^{x_{j2}}(j_k)\right)$$

At this point, we can focus on the problem where $n = 2$, $m = 1$. Without loss of generality, let $c_j = (x_1 \vee x_2)$. This will simplify our notation. The left-hand side of condition (10) becomes

$$u^2\left(h_1^{x_1}(1), h_2^{x_2}(1)\right) + u^2\left(h_1^{x_1}(2), h_2^{x_2}(2)\right) + u^2\left(h_1^{x_1}(3), h_2^{x_2}(3)\right) \tag{11}$$

while the right-hand side becomes

$$\mathbf{1}\left((x_1 \wedge x_2) \vee (\neg x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)\right) \tag{12}$$

The reader may have gathered that I intend to somehow associate $u\left(h_1^{x_1}(1), h_2^{x_2}(1)\right)$ with $(x_1 \wedge x_2)$, $u\left(h_1^{x_1}(2), h_2^{x_2}(2)\right)$ with $(\neg x_1 \wedge x_2)$, and $u^2\left(h_1^{x_1}(3), h_2^{x_2}(3)\right)$ with $(x_1 \wedge \neg x_2)$. Indeed, this is true. This would be immediate if, for example, $u(z_1, z_2) = \min\{z_1, z_2\}$ since we could set

$$h_1^T(1) = 1; \quad h_1^T(2) = 0; \quad h_1^T(3) = 1; \quad h_2^T(1) = 1; \quad h_2^T(2) = 1; \quad h_2^T(3) = 0$$

and $h_i^F(k) = 1 - h_i^T(k)$ for all i, k. In that case, $\min\left(h_1^{x_1}(1), h_2^{x_2}(1)\right) = \mathbf{1}(x_1 \wedge x_2)$ and so forth. Since only one of the three expressions in the disjunctive normal form (DNF) (9) can be true at one time, the sum (11) would be one if and only if (9) were true, and zero otherwise. In other words, (11) would equal (12) and we would be done.

The case for more general utility functions u is not quite so straightforward. Roughly, we can think of u as a function of two literals (e.g. (x_1, x_2), $(\neg x_1, x_2)$, $(x_1, \neg x_2)$). Since u is increasing, it will assign high utility when both variables are true and low utility when both variables are false. However, it will also assign medium utility when one variable is true and the other false. Essentially, this implies a three-valued logic where "medium" corresponds to a statement that is not quite true but also not quite false. This makes it difficult to map our problem onto MAX2SAT, which deals with a traditional two-valued logic.

To clear this hurdle, we rely on our supermodularity condition (7) and asymmetry across the states $k = 1, \ldots, 3$. While it will not be true that $u\left(h_1^{x_1}(1), h_2^{x_2}(1)\right) \propto (x_1 \wedge x_2)$, our sufficient condition (10) will hold because it sums across these three states. Formally, define

$$h_1^T(1) = c; \quad h_1^T(2) = a; \quad h_1^T(3) = b; \quad h_2^T(1) = c; \quad h_2^T(2) = b; \quad h_2^T(3) = a$$

$$h_1^F(1) = a; \quad h_1^F(2) = b; \quad h_1^F(3) = a; \quad h_2^F(1) = a; \quad h_2^F(2) = a; \quad h_2^F(3) = b$$

By setting $b > c > a$, we devalue the satisfaction of the first expression in the DNF (9) relative to the latter two. Now, condition (10) is true if and only if

$$u\left(h_1^T(1), h_2^T(1)\right) + u\left(h_1^T(2), h_2^T(2)\right) + u\left(h_1^T(3), h_2^T(3)\right) = u(c, c) + u(a, b) + u(b, a) = B \quad (13)$$
$$u\left(h_1^T(1), h_2^F(1)\right) + u\left(h_1^T(2), h_2^F(2)\right) + u\left(h_1^T(3), h_2^F(3)\right) = u(c, a) + u(a, a) + u(b, b) = B \quad (14)$$
$$u\left(h_1^F(1), h_2^T(1)\right) + u\left(h_1^F(2), h_2^T(2)\right) + u\left(h_1^F(3), h_2^T(3)\right) = u(a, c) + u(b, b) + u(a, a) = B \quad (15)$$
$$u\left(h_1^F(1), h_2^F(1)\right) + u\left(h_1^F(2), h_2^F(2)\right) + u\left(h_1^F(3), h_2^F(3)\right) = u(a, a) + u(b, a) + u(a, b) = A \quad (16)$$

for some $B > A$. This is because the first three choices make the DNF (9) true, which demands a high value B, and the last one makes it false, which demands a low value A.

Conditions (14) and (15) are equivalent since u is symmetric across dimensions. Conditions (13) and (14) hold if and only if $\psi(c) = 0$ where

$$\psi(z) = u(z, z) - u(z, a) - u(a, a) - u(b, b) + 2u(a, b)$$

Note that $\psi(a) < 0$ and $\psi(b) \geq 0$ by assumption (7). Since u is continuous, it follows from the intermediate value theorem that there exists $c \in (a, b]$ such that $\psi(c) = 0$. Finally, the fact that $A < B$ follows from

$$u(a, a) + u(b, a) + u(a, b) < u(c, c) + u(a, b) + u(b, a) \iff u(a, a) < u(c, c)$$

which is true since $c > a$ and u is strictly increasing along the diagonal. This completes the reduction, since the consequences $h_i(j)$ can be defined similarly for other variables and states. Moreover, defining the menu H only requires us to define $O(nm)$ such consequences.

Notice that c depends on u^2 but not on any other aspect of the problem, including n, m, or the clauses $c_1, \ldots, c_m$. This proof is non-constructive in the sense that I only prove the existence of a polynomial-time reduction (parameterized by c) from MAX2SAT to BDT, but do not provide an algorithm to find c itself. But whether c is easy or hard to compute is irrelevant for our purposes, so long as it does not need to be re-computed for different inputs to MAX2SAT. $\square$

Lemma 2. *The choice correspondence ϕ is NP-hard if*

$$u^2(a,a) + u^2(b,b) < u^2(a,b) + u^2(b,a) \tag{17}$$

Proof. I show that if the choice correspondence ϕ^n were tractable, then we could solve MIN2SAT in polynomial time. First, recall that the objective for MIN2SAT is

$$\max_{x_1,\ldots,x_n} \sum_{j=1}^{m} \mathbf{1}(c_j = \text{false}) \tag{18}$$

Our goal is to construct a menu H such that, given the solution $\phi(H)$, we could solve program (18) in poly(n, m) time. My approach will be almost identical to the proof of lemma 1, so I skip ahead to the point of divergence.

Here, I define acts similarly to the previous lemma. However, because our objective is to minimize rather than maximize the number of satisfied clauses, we give false literals a high utility and true literals a low utility. Formally, define

$$h_1^T(1) = a; \quad h_1^T(2) = b; \quad h_1^T(3) = a; \quad h_2^T(1) = a; \quad h_2^T(2) = a; \quad h_2^T(3) = b$$

$$h_1^F(1) = c; \quad h_1^F(2) = a; \quad h_1^F(3) = b; \quad h_2^F(1) = c; \quad h_2^F(2) = b; \quad h_2^F(3) = a$$

Now, our (negatively proportional) analog to condition (10) is true if and only if

$$u\left(h_1^T(1), h_2^T(1)\right) + u\left(h_1^T(2), h_2^T(2)\right) + u\left(h_1^T(3), h_2^T(3)\right) = u(a,a) + u(b,a) + u(a,b) = A \tag{19}$$
$$u\left(h_1^T(1), h_2^F(1)\right) + u\left(h_1^T(2), h_2^F(2)\right) + u\left(h_1^T(3), h_2^F(3)\right) = u(a,c) + u(b,b) + u(a,a) = A \tag{20}$$
$$u\left(h_1^F(1), h_2^T(1)\right) + u\left(h_1^F(2), h_2^T(2)\right) + u\left(h_1^F(3), h_2^T(3)\right) = u(c,a) + u(a,a) + u(b,b) = A \tag{21}$$
$$u\left(h_1^F(1), h_2^F(1)\right) + u\left(h_1^F(2), h_2^F(2)\right) + u\left(h_1^F(3), h_2^F(3)\right) = u(c,c) + u(a,b) + u(b,a) = B \tag{22}$$

for some $B > A$. This is because the first three choices make the DNF (9) true, which demands a low value A, and the last one makes it false, which demands a high value B.

Conditions (20) and (21) are equivalent since u is symmetric across dimensions. Conditions (19) and (20) hold if and only if $\psi(c) = 0$ where

$$\psi(z) = 2u(b,a) - u(b,b) - u(a,a) + u(a,a) - u(a,z)$$

Note that $\psi(a) = 2u(b,a) - u(b,b) - u(a,a) > 0$ and $\psi(b) = u(b,a) - u(b,b) \leq 0$ by assumption (17). Since u is continuous, it follows from the intermediate value theorem that there exists $c \in (a,b]$ such that $\psi(c) = 0$. Finally, the fact that $B > A$ follows from

$$u(c,c) + u(a,b) + u(b,a) > u(a,a) + u(b,a) + u(a,b) \iff u(c,c) > u(a,a)$$

which is true since $c > a$ and u is strictly increasing along the diagonal. This completes the reduction, for the same reasons as in the previous lemma. □

B.3 Proof of Theorem 2

B.3.1 Hardness

Assume richness (1, 2, 3), rationality (4), and monotonicity (5). Given the violation minor S_n of order k as advice, I want to show that an efficient algorithm for ϕ can be used to solve MAX2SAT with k variables in $O(\text{poly}(n))$ time.

1. If $k = \Omega(\text{poly}(n))$, this is a polynomial-time algorithm for MAX2SAT.
2. If $k = \omega(\log(\text{poly}(n)))$, this is a subexponential-time algorithm for MAX2SAT. There is a well-known polynomial-time reduction from 3SAT with n' variables and m' clauses to MAX2SAT with $n' + m'$ variables and $10m'$ clauses. The description length of the 3SAT instance is $\Theta(n') + \Theta(m')$, which is the same order as the description length of the MAX2SAT instance. So, a subexponential-time algorithm for MAX2SAT implies the same for 3SAT.

Consider a boolean formula with k variables $x_1, \dots, x_k$, i.e. an instance of MAX2SAT. I will refer to this as the original formula. This terminology is meant to contrast with an auxilliary formula with variables $y_1, \dots, y_n$ that will describe an instance of weighted MAX2SAT. As we will see, the auxilliary problem is constructed in a way that its solution corresponds to the solution of the original problem. Then, we will reduce the auxilliary problem to SEU maximization.

Suppose dimensions $d_1, \dots, d_l$, corresponding to nodes of G_n, be combined via edge contractions in the contracted node d of S_n. In the auxilliary problem, we wish to impose the constraint $y_{d_i} = y_{d_j}$ for all $i, j = 1, \dots, l$. This allows us to treat the contracted node d as a single variable in the original formula, where any dimension d_i can represent the variable d. This is useful because, for a given dimension j, some dimensions in d may share a pairwise violation (i.e. an edge in G_n) with j, but not others.

Fortunately, it is possible to impose the constraint $x = y$ by adding clauses $x \vee \neg y$ and $\neg x \vee y$ to the auxilliary instance, with weight exceeding twice the total weight of all other clauses where x, y are represented (excepting other clauses representing equality constraints). The reason is that setting $x = y$, regardless of whether the value is true or false, will make both of these clauses true. In contrast, setting $x \neq y$ will make one of these clauses false. Given the weights, no benefit from setting $x \neq y$ that comes from other clauses can outweigh this advantage.

Since S_n is a complete graph, every contracted node d is adjacent to every other contracted node d'. To represent a clause $x \vee y$ in the original formula, it suffices to choose some node in d representing x and some node in d' representing y. These nodes, corresponding to dimensions of u, must have a pairwise violation in order for the reduction to work. Equivalently, they must share an edge in G_n. Of course, it is always possible to find such a pair. If no such pair existed, there would be no sequence of edge contractions in G_n that would make d adjacent to d'.

Having described the auxilliary problem, it remains to construct a menu such that SEU maximization corresponds to solving weighted MAXSAT. Essentially, we want to recreate the argument that we used in lemmas 1 and 2. There, we observed that the following condition (10)

$$\sum_{l=1}^{3} u\left(h_1^{y_1}(j_l), \ldots, h_n^{y_n}(j_l)\right) \propto \underbrace{\mathbf{1}\left((y_{j1} \wedge y_{j2}) \vee (\neg y_{j1} \wedge y_{j2}) \vee (y_{j1} \wedge \neg y_{j2})\right)}_{\mathbf{1}(c_j = \text{true})}$$

is nearly sufficient for SEU with submenus $H_i = \{h_i^T, h_i^F\}$ to correspond to an optimal assignment in MAX2SAT. The qualifier "nearly" reflects the fact that we also need the proportionality constants to be the same across all clauses $y_{j1} \vee y_{j2}$. In the case of weighted MAXSAT, these constants should reflect the weight of that clause.

Consider a clause $y_i \vee y_j$. As in the proof of theorem 1, let (z, a_0, b_0, a_1, b_1) constitute an (i, j)-pairwise violation of additive separability, and define

$$u^2(a, b) := u\left(\ldots, z_{i-1}, a, z_{i+1}, \ldots, z_{j-1}, b, z_{j+1}, \ldots\right)$$

However, without label invariance, u^2 is not the same across all dimensions i, j. For this reason, we need each (i, j)-pairwise violation to be included in the description of S_n, which we take as advice. This description takes $O(n^3)$ space. In any case, we are focusing on a single clause, so I will suppress the dependence on i, j. As before (see lemma 1), we construct acts so that condition (10) can be rewritten in terms of only two variables, i.e.

$$\sum_{l=1}^{3} u^2\left(h_1^{y_l}(l), h_2^{y_l}(l)\right) \propto \mathbf{1}\left((y_1 \wedge y_2) \vee (\neg y_1 \wedge y_2) \vee (y_1 \wedge \neg y_2)\right) \tag{23}$$

As before, there are two cases to consider, based on the direction of the (i, j)-pairwise violation. We consider each in turn. The next two observations will imply that we can satisfy the above condition, and then we complete the definition of subacts h_i^T, h_i^F in the same way as theorem 1.

Assume without loss of generality that $b_1 > a_1$ and $b_0 > a_0$. Suppose that

$$u^2(b_0, a_1) + u^2(a_0, b_1) < u^2(b_0, b_1) + u^2(a_0, a_1) \tag{24}$$

I claim that there exist constants $B > A$ and weights $\alpha \in \Delta\{1, 2, 3\}$ such that

$$\alpha_1 u^2(b_0, b_1) + \alpha_2 u^2(b_0, a_1) + \alpha_3 u^2(a_0, b_1) = B \tag{25}$$
$$\alpha_1 u^2(b_0, a_1) + \alpha_2 u^2(b_0, b_1) + \alpha_3 u^2(a_0, a_1) = B \tag{26}$$
$$\alpha_1 u^2(a_0, b_1) + \alpha_2 u^2(a_0, a_1) + \alpha_3 u^2(b_0, b_1) = B \tag{27}$$
$$\alpha_1 u^2(a_0, a_1) + \alpha_2 u^2(a_0, b_1) + \alpha_3 u^2(b_0, a_1) = A \tag{28}$$

To see this, note that (25) exceeds (28) by the fact that u^2 is strictly increasing (inherited from monotonicity of u). Therefore, we only need to satisfy the first three equations. The fact that a strictly positive solution exists follows from (24).

These weights α represent the likelihood of the underlying states. Recall that the effective state corresponds to a subinterval of $\Theta = [0, 1]$ on which the acts $h \in H$ are constant. Consider the subintervals corresponding to the three states associated with clause $y_1 \vee y_2$. These three subinterval should have length proportional to

$$\frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3}$$

Next, consider the case

$$u^2(b_0, a_1) + u^2(a_0, b_1) > u^2(b_0, b_1) + u^2(a_0, a_1) \tag{29}$$

The previous observation no longer holds, since a solution α is not guaranteed to exist. On the other hand, there do exist constants $B > A$ and weights $\beta \in \Delta\{1, 2, 3\}$ such that

$$\beta_1 u^2(b_0, b_1) + \beta_2 u^2(b_0, a_1) + \beta_3 u^2(a_0, b_1) = A \tag{30}$$
$$\beta_1 u^2(b_0, a_1) + \beta_2 u^2(b_0, b_1) + \beta_3 u^2(a_0, a_1) = A \tag{31}$$
$$\beta_1 u^2(a_0, b_1) + \beta_2 u^2(a_0, a_1) + \beta_3 u^2(b_0, b_1) = A \tag{32}$$
$$\beta_1 u^2(a_0, a_1) + \beta_2 u^2(a_0, b_1) + \beta_3 u^2(b_0, a_1) = B \tag{33}$$

We used a similar observation in lemma 2 to obtain a reduction from MIN2SAT. Unfortunately, without label-invariance, u^2 depends on the dimension pair i, j; condition (29) may apply to some pairs while condition (24) applies to other pairs. So, we need to stick with MAX2SAT.

A straightforward way to handle this case is to modify the auxilliary formula. Replace the original clause $y_1 \vee y_2$ with three new clauses: $\neg y_1 \vee \neg y_2$, $\neg y_1 \vee y_2$, and $y_1 \vee \neg y_2$. The original clause is satisfied iff only two of the new clauses are satisfied. It is falsified iff all three of the new clauses are satisfied. So, instead of trying to satisfy the original clause, we can aim to falsify as many of the new clauses as possible. The way to interpret equations (30) through (33) is that, with effective states that are proportional to β, SEU conditioned on those states is proportional to the falsification of a clause. This is exactly what we need.

At this point, our construction satisfies (23). The only remaining issue is that, unlike in the proof of theorem 1, the proportionality constants will differ from clause to clause. Furthermore, some of the clauses are weighted. This can be handled in a straightforward way by manipulating the intervals used to define the effective state space. Suppose a clause receives weight w and the difference between its satisfied value and its unsatisfied value is Δ. Then the three effective states associated with said clause should have interval length proportional to w/Δ. Integrating over states in the sense of SEU gives precisely the objective function of weighted MAX2SAT, so ϕ gives a solution to the auxilliary problem.

B.3.2 Tractability

Definition 15 (DP Algorithm). *Specify a utility function u and a graph $G = (V, E)$ of order n and contraction degeneracy k. Given an n-dimensional product menu H, the* dynamic programming *(DP) algorithm proceeds as follows:*

1. *Construct a directed acyclic graph $\tilde{G} = (V, \tilde{E})$ where each node has out-degree at most k.*

2. *Perform a topological sort of $\tilde{G}$. In particular, nodes are numbered, i.e. $V = \{1, \dots, n\}$, and if $i < j$ then there is no path in $\tilde{G}$ from node j to node i.*

3. *Initialize the act $\tilde{h}$ by setting $\tilde{h}_i(\theta) = \underline{z}$ for all dimensions i and states θ.*

4. *Let i be the smallest node in $\tilde{G}$. Initialize the frontier $F \subseteq V$ as $\emptyset$.*

   (a) *Let S_i consist of i's successor nodes j, i.e. where $(i, j) \in \tilde{E}$. Let F_i be the set of nodes $j \in F$ for which, in the undirected graph G, there exists a path from i to j that does not pass through F. I will refer to the pair (i, j) as $(G \setminus F)$-connected.*

      i. *If $|F_i| > k$, return to step (4a) with $i := i + 1$.*

      ii. *Else let $L_i = \prod_{j \in F_i \cup S_i} H_i$ be a set of partially-specified acts $h_{F_i \cup S_i}$.*

   (b) *Initialize the act $\tilde{h}$ by setting $\tilde{h}_i(\theta) = \underline{z}$ for all dimensions i and states θ.*
   item In this step, we construct a function $f_i : L_i \to H_i$ where

$$f_i\left(h_{F_i \cup S_i}\right) \in \arg\max_{h_i \in H_i} \int_0^1 u\left(h_i(x), \hat{h}_{-i}(x)\right) dx$$

   To define it, iterate over all $h_{F_i \cup S_i} \in L_i$.

      i. *Initialize the act $\hat{h} = \tilde{h}$ but set $\hat{h}_{F_i \cup S_i}(x) = h_{F_i \cup S_i}(x)$.*

      ii. *Iterate through $j \notin F_i \cup S_i \cup \{i\}$.*

         A. *If $j > i$ then $\hat{h}_j(\theta) = \underline{z}$ for all states θ.*

         B. *Else if i, j are not $(G \setminus F)$-connected, set $\hat{h}_j(\theta) = \underline{z}$ for all θ.*

         C. *Else proceed as follows. Because $i \leq j$ and $j \notin F_i$, we have already constructed f_j. Assume that f_j is up-to-date, in that it only depends on subacts h_l for dimensions $l \in F_i \cup \{i\}$. Define $\hat{h}_j = f_j\left(h_{F_i}\right)$.*

   (c) *If $F_i \cup S_i = \emptyset$, set $\tilde{h}_i = f_i(\emptyset)$.*

   (d) *Iterate through $j \notin F_i \cup S_i \cup \{i\}$.*

      i. *If $j > i$, do nothing.*

      ii. *Else if i, j are not $(G \setminus F)$-connected, do nothing.*

      iii. *Else update f_j as follows. Currently, f_j is a function of h'_i and h'_{F_i}. Replace the argument h'_i with $f_i\left(h'_{S_i}, h'_{F_i}\right)$. Now, reconstruct f_j as a function of $h'_{S_i \cup F_i}$. Moreover, if $F_i \cup S_i = \emptyset$, set $\tilde{h}_j = f_j(\emptyset)$.*

(e) *Redefine F with node i deleted and nodes S_i added. Redefine $\tilde{G}$ with node i deleted. If any nodes remain, repeat step (3).*

5. *Return act $\tilde{h}$, which belongs to the menu H by construction.*

The following is a fixed-parameter tractability result that combines usefully with the log-polynomial rates established in theorem 2.

Proposition 4. *Given an n-dimensional product menu with submenus of size l and an effective state space of size m, the DP algorithm parameterized by graph G will halt in $O(\text{poly}(n, m, l^k))$ time, where k is the contraction degeneracy of G.*

Proof. Most of these steps are obviously polynomial in n, k, l or involve iterating over sets of size l^k. Here, I will justify the less obvious steps and assumptions. The key observation is the third one.

1. Step 1 takes $O(\text{poly}(n))$ time since G has degeneracy k. The algorithm is simple: find a node i with degree $\leq k$, orient all the edges outwards. In the subgraph without i, repeat.

2. Step 4 will halt in $O(n^2)$ steps. Whenever step 4.e is reached, the number of nodes in $\tilde{G}$ decreases by one, so this can be reached at most n times. The only other possibility is that step 4.a.i is reached, but this can happen at most n consecutive times before it fails.

3. Step 4.a.i will never fail. This is due to the following observation. Consider the minor S of graph G with all edges contracted except for those connecting two nodes $i, j \in F$. Then there is an edge between i and j if and only if they are adjacent in G or if there is a path from i to j that does not go through F. Note that S has a node of degree less than or equal to k, by the definition of contraction degeneracy. This node contains some $j \in F$. Step 4.a.i will eventually reach this j and move on to step 4.a.ii.

4. Step 4.b.ii.C assumes that the function f_j is up-to-date. This is guaranteed by step 4.d, which updates any function f_j that may have been affected by changes in iteration i of step 4.

5. Step 5 assumes that $\tilde{h}$ belongs to the menu H. This follows from the fact that $F = \emptyset$ by the time step 4 terminates. Since step 4.d ensures that functions are up-to-date, every function f_j must have reached a point where its argument is vacuous. At that point, $\tilde{h}_j$ would have been defined in steps 4.c or 4.d.iii.

□

B.4 Proof of Proposition 1

These rankings are well-known except for (possibly) the last one, which I now prove.

1. $\mathrm{cdgn}(G) \preceq \mathrm{Had}(G)$. There exists a minor H with minimal degree $k = \mathrm{cdgn}(G)$. Furthermore, $\mathrm{avg}(H) \geq \delta(H) = k$. Kostochka (1984) proved that G has a clique minor of order

$$\Omega\left(\mathrm{avg}(H)/\sqrt{\log \mathrm{avg}(H)}\right)$$

Therefore, $\mathrm{Had}(G)$ is at least proportional to k, up to log factors.

2. $\mathrm{cdgn}(G) \succeq \mathrm{Had}(G)$. There is a minor clique H of order $k = \mathrm{Had}(G)$. It is complete, so $\delta(H) = k$. Therefore, $\mathrm{cdgn}(G) \geq \delta(H) = k$.

C Omitted Proofs in Section 5.2

C.1 Proof of Theorem 3

Let $\phi \in \Phi_R$ have revealed utility function u and violation graph G_n. Let $k = 1 + \mathrm{dgn}(G_n)$. I will argue that if $\bar{u}^n(\bar{z}, \dots, \bar{z}) = o(n/k)$, then ϕ's approximation ratio is $o(1)$, i.e. it is bounded above by a term that is approaching zero. In contrast, the greedy algorithm obtains a ½-approximation by proposition 2. Therefore, the ratio of the approximation ratios is $\omega(1)$, i.e. it is bounded below by a term that is approaching infinity. The three assertions of theorem 3 follow immediately, since:

1. By corollary 2.1, if NP $\not\subset$ P/poly, then $k = o(\mathrm{poly}(n))$.

2. By corollary 2.2, if the non-uniform ETH holds, then $k = O(\log(\mathrm{poly}(n)))$.

3. By theorem 1, if P $\neq$ NP and ϕ satisfies label-invariance (7), then $k = 1$.

To prove this, I construct a menu H where ϕ obtains a vanishingly small fraction of the optimal SEU. In doing so, I use one additional concept from graph theory.

Definition 16 (Chromatic Number)**.** *Let C be a set of colors. Let $G = (V, E)$ be a graph. A C-*coloring *of G is an assignment $f : V \to C$ of nodes $i \in V$ to colors $c \in C$ where adjacent nodes have different colors, i.e. $(i, j) \in E$ implies $f(i) \neq f(j)$. The* chromatic number *of G is the size of the smallest set C such that a C-coloring exists.*

The chromatic number of G_n is at most k. This follows from the fact that a graph $G = (V, E)$ of degeneracy d can be colored with $d + 1$ colors (Szekeres and Wilf 1968). Let $C = \{1, \dots, k\}$ and suppose we color G_n with C. Choose the color $c \in C$ with the most nodes, a number that must be at least n/k. Let V_c denote the set of c-colored nodes.

Next, I construct a product menu H. For all dimensions $i \notin V_c$, set $H_i = \{h_i\}$ where $h_i(\theta) = e_i$ for all states θ. For all dimensions $i \in V_c$, set $H_i = \{h_i^T, h_i^F\}$. Consider an evenly-spaced grid $X = \{0, x_1, \dots, x_{k+1}, 1\}$ on $[0, 1]$. Let $h^T(\theta) = \bar{z}$ if $\theta \geq x_k$ and $h^T(\theta) = \underline{z}$ otherwise. Let $h^T(\theta) = \bar{z}$ if $\theta \in [x_{i-1}, x_i]$ and $h^T(\theta) = \underline{z}$ otherwise.

Note that the consequences z_i for $i \notin V_c$ are fixed. Moreover, as a function of z_i for $i \in V_c$, u is additively separable. This follows from the fact that two nodes i, j with the same color c cannot be adjacent in the violation graph. As a consequence, the agent will always choose h_i^T over h_i^F. After all, the former obtains the high value $\bar{z}$ over two effective states (i.e. an interval of length $2/(k+1)$) and $\underline{z}$ otherwise. Whereas the latter obtains the high value $\bar{z}$ only in one effective state (i.e. an interval of length $1/(k+1)$). Essentially, we have constructed a menu in which the agent will narrowly choice bracket, myopically seeking the highest value in each dimension.

Unfortunately, from the perspective of $\bar{u}$, always choosing h_i^T can be a poor strategy, roughly due to satiation effects. This strategy will obtain $\bar{z}$ in every dimension but only for two effective states, for a SEU of

$$\frac{|V_c|-2}{|V_c|}\cdot \bar{u}\left(\underline{z},\dots,\underline{z}\right)+\frac{2}{|V_c|}\bar{u}\left(\bar{z},\dots,\bar{z}\right)$$

In contrast, always choosing h_i^F guarantees $\bar{z}$ in one dimension for every state, for a SEU of

$$|V_c|\cdot \bar{u}\left(\bar{z},\underline{z},\dots,\underline{z}\right)$$

Recall that $|V_c| \geq n/k$. The latter SEU is increasing linearly in $|V_c|$, while the former SEU is increasing sublinearly, assuming $\bar{u}\left(\bar{z},\dots,\bar{z}\right)=o(n/k)$. This is what I sought to show.

C.2 Proof of Proposition 2

Let ϕ_G be the choice correspondence associated with the greedy algorithm. Given a product menu $H \in \mathcal{H}$, let $\hat{h}=\phi_G(H)$. For each dimension i and state x, define

$$\begin{aligned}\Delta_i(x) &= \bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_i(x),\underline{z},\dots,\underline{z}\right)-\bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),\underline{z},\dots,\underline{z}\right)\\ &\geq \bar{u}^n\left(\hat{h}_1(x),\dots,h_i(x),\underline{z},\dots,\underline{z}\right)-\bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),\underline{z},\dots,\underline{z}\right)\\ &\geq \bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),h_i(x),\dots,h_n(x)\right)-\bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),\underline{z},h_{i+1}(x),\dots,h_n(x)\right)\\ &\geq \bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),h_i(x),\dots,h_n(x)\right)-\bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_i(x),h_{i+1}(x),\dots,h_n(x)\right)\end{aligned} \tag{34}$$

where the first inequality follows from the the greedy hypothesis, the second follows from the SCC with

$$z''=(0,\dots,0,\hat{h}_{i+1}(x)-\underline{z},\dots,\hat{h}_n(x)-\underline{z})$$

and the third follows from monotonicity. Define

$$\Delta_i=\int_0^1\Delta_i(x)dx$$

This can be interpreted as the added value from choosing $\hat{h}_i$, as evaluated by the greedy algorithm at step i. We want to compare this to the added value from choosing $\hat{h}_i$, having commited to the greedy choices in dimensions $1,\dots,i-1$, but otherwise choosing optimally. Define

$$\mathrm{OPT}_i=\max_{h_i,\dots,h_n}\int_0^1\bar{u}^n\left(\hat{h}_1(x),\dots,\hat{h}_{i-1}(x),h_i(x),\dots,h_n(x)\right)dx$$

It follows from inequality (34) that

$$\mathrm{OPT}_i \leq \mathrm{OPT}_{i+1} + \Delta_i \quad \text{and} \quad \mathrm{OPT}_1 \leq \mathrm{OPT}_{n+1} + \sum_{i=1}^{n} \Delta_i$$

Observe that OPT_1 is the true optimum, obtained by $\bar{\phi}$. Whereas OPT_{n+1} is the payoff of the greedy algorithm. It is easy to see that the greedy algorithm obtains SEU $\sum_{i=1}^{n} \Delta_i$. Therefore, ϕ_G is a $1/2$-approximation to $\bar{\phi}$.