Bayesian Incentive Compatibility via Matchings

Jason D. Hartline*

Robert Kleinberg[†]

Azarakhsh Malekian[‡]

Abstract

Optimally allocating cellphone spectrum, advertisements on the Internet, and landing slots at airports is computationally intractable. When the participants may strategize, not only must the optimizer deal with complex feasibility constraints but also with complex incentive constraints. We give a very simple method for constructing a Bayesian incentive compatible mechanism from any, potentially non-optimal, algorithm that maps agents' reports to an allocation. The expected welfare of the mechanism is, approximately, at least that of the algorithm on the agents' true preferences.

The construction is based on a maximum weight matching in the type space of each agent that be calculated quickly when the type spaces are reasonably sized. Furthermore, the computation can be performed independently for each agent and, therefore, scales well with the number of agents. A similar construction was previously given by Hartline and Lucier (2010) for agents with single-dimensional types; ours allows multi-dimensional types.

JEL classification. C70, D44.

Keywords. Bayesian mechanism design, algorithmic mechanism design, incentive compatibility, multi-dimensional preferences.

1 Introduction

The automation of commerce, e.g., with a computer, has made it possible to contemplate the design of economic mechanisms in very complex environments, e.g., for allocating cell phone spectrum, landing slots at airports, and advertisements on the Internet. It also makes it clear that computational limitations are limitations for mechanism design. Unfortunately, this is one aspect in which the classical theory of mechanism design is incompatible with reality. From a computational point of view, i.e., even if there were no strategizing on the part of the participants, the optimization problems inherent in the above allocation problems are intractable. Classical economic mechanisms such as the Vickrey-Clarke-Groves (VCG) mechanism, which require solving the inherent optimization problem, must therefore be regarded as not generally practical.

---

*Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL. Email: hartline@eecs.northwestern.edu. Supported in part by NSF Grant CCF-0830773 and NSF Career Award CCF-0846113. Part of this work was done while the author was visiting Harvard University.

[†]Department of Computer Science, Cornell University, Ithaca, NY. Email: rdk@cs.cornell.edu. Supported by NSF grants CCF-0643934 and AF-0910940, a Google Research Grant, an Alfred P. Sloan Foundation Fellowship, and a Microsoft Research New Faculty Fellowship. Part of this work was done while the author was visiting Microsoft Research New England.

[‡]Rotman School of Management, University of Toronto, Toronto, ON. Email: azarakhsh.malekian@rotman.utoronto.ca. Part of this work was done while the author was at Northwestern University.

Fortunately, sophisticated techniques in combinatorial optimization and approximation algorithms, and even ad hoc approaches, can lead to very good (but not necessarily optimal) computationally efficient solutions to intractable optimization problems in both theory and practice. An important question to address in mechanism design is whether these non-optimal algorithms can be used as a basis for mechanisms, or whether novel mechanisms must be developed to consider both the incentive constraints and computational constraints together. Notice that the VCG mechanism can be viewed as a reduction from optimal mechanism design to optimal algorithm design, i.e., it constructs an optimal mechanism from an optimal algorithm. Our question is analogous: *Does there exist a reduction from ad hoc mechanism design to ad hoc algorithm design that preserves performance?*

We give a positive answer to this question for environments where the agents’ private preferences (i.e., the input) are independently distributed. For any algorithm that is good from a computational perspective (i.e., it runs quickly on random inputs from the distribution) and an economic perspective (i.e., its welfare is good on random inputs), we construct a mechanism with only better welfare (in expectation, modulo small losses due to sampling error) while preserving the algorithm’s computational efficiency. The preprocessing step in constructing this mechanism takes time proportional to the product of original running time of the algorithm, the sum of the sizes of the agents’ type spaces, and a term related to the number of samples we would need to estimate the value an agent would obtain when the algorithm is run on a random input. Such a running time is considered computationally tractable when the sizes of the agents’ type spaces are not too large.

The Vickrey-Clarke-Groves construction gives dominant strategy incentive compatible mechanisms, i.e., ones where truthful reporting is a dominant strategy for each agent. Our construction is weaker in the sense that it gives mechanisms that are only Bayesian incentive compatible (BIC), i.e., truthful revelation is a Bayes-Nash equilibrium. Moreover, part of our construction requires ascertaining the expected value an agent would have for a random outcome from the algorithm. This can be estimated by sampling, but insofar as the estimation is approximate, so may be the incentive compatibility. A mechanism is ε -BIC if no agent can gain more than ε in utility from a non-truthful report under the assumption that all other agents are truthtelling.

This work is a direct follow-up to work of Hartline and Lucier (2010) that gives such a Bayesian reduction for the special case where the selfish agents’ preferences are single-dimensional, e.g., a value for receiving service. While this prior work has elegant economic motivation, there is no clear generalization of its ironing-based approach to more general (and fundamental) multi-dimensional environments, e.g., where an agent might have distinct values for several services that are available. In contrast we consider multi-dimensional environments and give a simple matching-based reduction. Essentially: incentive constraints in BIC mechanism design are as simple as solving a maximum-weight-matching problem independently on the type space of each agent.

The template of our approach is the same as that of Hartline and Lucier (2010). Our reduction will accept reported types from agents. It will transform each of these types to a distribution over types with the properties: (a) the transformation applied to a random type from the distribution results in a transformed type with the same distribution, (b) the transformation weakly improves performance, and (c) the algorithm composed with the transformation is Bayesian incentive compatible. It then inputs the transformed types into the original algorithm and returns its output with the appropriate payments.

The transformation we apply to each agent’s type is based on computing the maximum weighted matching in a bipartite graph between a set of random types from the distribution (including the agent’s real type), termed “replicas”, and the set of outcomes of the algorithm on another equal-sized random set of types drawn from the distribution, termed “surrogates”. The transformation then

outputs the surrogate type to which the agent’s real type is matched. This basic *replica-surrogate matching* approach can be further refined and improved for the special cases of discrete explicitly-specified type spaces. (The latter reduction was independently and concurrently discovered by Bei and Huang, 2011.)

To solve the above maximum weighted matching problem we need to be able to evaluate the value of each replica for the outcome the algorithm obtains for each of the surrogates. If a replica is matched to a surrogate, the value it obtains is the expected value over outcomes the algorithm produces when run on the surrogate and random types of the other agents. We consider two computational models: an *ideal* model (Section 3) where the expected values an agent has for distributions over outcomes produced by the algorithm are given, and a *black-box* model (Section 4) where we can only estimate this value by repeatedly sampling the types of the other agents and running the algorithm. The ideal model exposes the main economic intuition of the approach while the black-box model is computationally realistic.

Results. In the ideal model the reduction is BIC and we give bounds on the number of replicas and surrogates needed to ensure that the surplus loss of the resulting mechanism is at most a small additive ε . These bounds degrade with a suitable notion of the complexity of the type space. For the black-box model we obtain the following three results. When the agents’ preferences are single-dimensional, the reduction can be made BIC with estimates of the replicas’ values of surrogate outcomes. When the agents’ types are multi-dimensional, discrete, and explicitly specified, we can correct errors inherent in sampling to give a BIC reduction in time polynomial in the sizes of the type and outcome spaces. For general continuous type spaces our reduction is ε -BIC.

Related Work. The Vickrey-Clarke-Groves (VCG) mechanism generalizes the Vickrey auction and reduces optimal mechanism design to optimal algorithm design for the objective of social welfare (Vickrey, 1961; Clarke, 1971; Groves, 1973). However, as was observed by Lehmann et al. (2002), (a) the optimal algorithm design problem may be computationally intractable and (b) the VCG approach is generally incompatible with non-optimal algorithms.

The theory for the design of computationally tractable algorithms is the following. An algorithm is considered tractable if its worst-case running time grows at most polynomially in the “size” of the input.¹ For instance, in mechanism design problems the size of the input depends on the number of agents, the number of outcomes, and the number of bits required to specify an agent’s valuation over outcomes or (in Bayesian settings) to specify an agent’s valuation distribution. An optimization problem is considered intractable if there is no tractable algorithm. It is notoriously difficult to prove lower bounds on the running times of all algorithms that solve a problem; therefore, NP-hardness — the property of being at least as hard as any problem in NP, a class that includes many problems that are believed to be intractable — is used as a proxy for intractability.

When algorithmic solutions to an optimization problem are intractable, e.g., via NP-hardness, an approach to obtain positive results is to try to approximate the optimum. An *approximation algorithm* is one that always finds a solution whose objective performance is within a specific ratio of the optimum. This worst-case ratio is known as the algorithm’s *approximation factor*. While optimally solving a given problem may be intractable, approximating it may be tractable. An approximation algorithm is said to be *optimal* if it obtains the best possible approximation factor of any tractable algorithm.

---

¹This definition of tractability is standard in computer science. For input size N , consider the *polynomial running time* of N^c for a constant c . Notice that a polynomial running time scales well: if the input gets a constant factor larger, i.e., from N to βN for constant β , then the running time gets a constant factor longer, i.e., from N^c to $(\beta N)^c = \beta^c N^c$ where, importantly, β^c is a constant.

Lehmann et al. (2002) studied the single-minded combinatorial auction problem, i.e., where each agent desires a specific subset of the items for sale and has a private value for receiving this bundle. They showed that a certain class of greedy algorithms is both computationally efficient and (dominant strategy) incentive compatible, i.e., there exist payments that will induce truthful behavior. For the single-minded combinatorial auction problem with m items, the resulting mechanism obtains the optimal approximation factor of $\Theta(\sqrt{m})$.

Since the work of Lehmann et al. (2002) there has been extensive study of *approximation mechanisms*, i.e., mechanisms that are (a) tractable and (b) obtain good worst-case approximation factors. These studies either focus on a particular paradigmatic problem and attempt to construct novel approximation mechanisms that match the performance of the optimal approximation algorithm, or they focus on general methods for converting specific kinds of algorithms into mechanisms while preserving their approximation factor. Our work is most related to this latter class of papers, surveyed below.

A standard approach in approximation algorithms is to formulate the problem as an *integer program*, solve the *linear program relaxation*, and then *round* the fractional solution from the linear program to obtain a feasible integral solution to the original program without much loss. The *integrality gap* is the worst-case ratio between the optimal solution to the linear program and the optimal solution to the integer program. (Obviously, since the linear program has only relaxed constraints, its solution is only better than that of the integer program.) Lavi and Swamy (2011) consider multi-parameter packing problems and give a technique for constructing a (randomized) β -approximation mechanism from any β -approximation algorithm that “verifies” that the integrality gap is β .

The best kind of approximation algorithm, from the standpoint of objective performance, is one that comes from a *fully polynomial time approximation scheme* (FPTAS). A fully polynomial time approximation scheme is an algorithm that is additionally parameterized by a desired loss ε . For any $\varepsilon > 0$ the algorithm will in polynomial time (in the size of the input and $1/\varepsilon$) find a solution that is a $(1 + \varepsilon)$ -approximation to the optimal solution. Briest et al. (2005) give an FPTAS for a single-minded multi-unit auction problem that is based on an FPTAS for the non-game-theoretic knapsack problem. Dughmi and Roughgarden (2014) generalize this approach in two respects. First, they generalize it to environments where agents’ preferences are multi-dimensional and additive and the feasibility constraint is downward-closed. Second, they show that any FPTAS algorithm can be converted into an FPTAS mechanism. Their result is based on a connection between FPTASes, smoothed analysis of algorithms (Spielman and Teng, 2004), and randomized mechanisms.

Up to this point, the results described have been for specific kinds of algorithms, the performance guarantees have been worst-case, and the mechanisms produced have been dominant strategy incentive compatible. In contrast to these works, Hartline and Lucier (2010) give a black-box reduction from mechanism design to algorithm design that is Bayesian both in performance guarantees and incentive compatibility. In comparison to the preceding discussion, this result (a) makes no assumptions on the structure of the algorithm, (b) does not require the algorithm to have good worst-case properties, and (c) gives a mechanism that is Bayesian incentive compatible (meaning: truth-telling is a Bayes-Nash equilibrium). While (a) and (b) place fewer requirements on the algorithm, (c) places more assumptions on the agents’ rationality. The Hartline and Lucier (2010) approach requires the agents’ preferences to be single-dimensional.

This paper builds on the construction provided by Hartline and Lucier (2010) and gives a reduction from mechanism design to algorithm design when the agents have general multi-dimensional preferences. The mechanisms we produce are Bayesian incentive compatible and they preserve the expected performance of the algorithm on which they are based, up to an additive loss which can be made arbitrarily small. In concurrent and independent work, Bei and Huang (2011) reduce ε -

BIC mechanism design to algorithm design in environments with discrete and explicitly given type spaces. In contrast, our work for discrete environments uses the same basic approach but produces a BIC mechanism.

2 Preliminaries

We consider mechanisms for self-interested agents with independent private values. An agent has a private *type* t from *type space* $\mathcal{T}$. There are n agents. When we wish to be specific about a particular agent $\kappa \in [n]$ we index as t^κ and $\mathcal{T}^\kappa$. The *type profile* of the n agents is denoted by $\mathbf{t} = (t^1, \dots, t^n) \in \mathcal{T}^1 \times \dots \times \mathcal{T}^n$. An algorithm $\mathcal{A}$ maps a type profile $\mathbf{t}$ to an *outcome* x from *outcome space* $\mathcal{X}$. Agent κ with type t^κ has *valuation* $v(t^\kappa, x)$ for outcome x . We assume that agent values are non-negative and come from a bounded range that, without loss of generality, is $[0, 1]$. A mechanism $\mathcal{M} = (\mathcal{A}, \mathbf{p})$ maps the type profile into an outcome via an algorithm $\mathcal{A}$ and into a *payment profile* $\mathbf{p} = (p^1, \dots, p^n)$. We overload notation and refer to p^κ as both the payment for κ and, when invoked on a type profile, as the mapping from type profile to a payment, e.g., as $p^\kappa(\mathbf{t})$. Agent κ with type t^κ has *utility* $v(t^\kappa, x) - p^\kappa$ for outcome x and payment p^κ . Agents are *risk-neutral quasi-linear* utility maximizers, i.e., they aim to maximize their expected utilities. For outcome x drawn from a distribution of outcomes X , the agent's value for the distribution is $v(t^\kappa, X) = \mathbf{E}[v(t^\kappa, x)]$.

A further distinction that is relevant to our results concerns the structure of the agents' type spaces. When we refer to *single-dimensional types*, we mean that $\mathcal{X} \subseteq [0, 1]^n$, $\mathcal{T}^\kappa = [0, 1]$ for all κ , and value $v(t^\kappa, x) = t^\kappa x^\kappa$ for outcome $x = (x^1, \dots, x^n) \in \mathcal{X}$. This is the standard model of single-dimensional mechanism design, normalized so that types and allocations take values on the interval $[0, 1]$. When we refer to the setting of (multi-dimensional) *discrete type spaces*, we mean that each type space $\mathcal{T}^\kappa$ has finite cardinality, and the algorithm is given a complete description of the type distribution of agent κ , in the form of a listing of all the points of $\mathcal{T}^\kappa$ and their associated probabilities. Finally, the setting of (multi-dimensional) *continuous types* is the fully general setting defined in the preceding paragraph.

We are looking for mechanisms for maximizing the *social surplus*, i.e., the sum of the values of the agents minus any cost to the designer. Denote by $c(x)$ the designer's cost for outcome x . The optimal social surplus is $\text{OPT}(\mathbf{t}) = \sup_{x \in \mathcal{X}} \sum_{\kappa} v(t^\kappa, x) - c(x)$. For algorithm $\mathcal{A}$, we will overload notation and use $\mathcal{A}(\mathbf{t})$ to refer to both the outcome and its surplus; i.e., algorithm $\mathcal{A}$ with outcome $x = \mathcal{A}(\mathbf{t})$ has social surplus $\mathcal{A}(\mathbf{t}) = \sum_{\kappa} v(t^\kappa, x) - c(x)$.

In Bayesian algorithm design and mechanism design the types are drawn from a distribution. F^κ denotes the distribution for $t^\kappa \in \mathcal{T}^\kappa$ and the joint distribution is the product $\mathbf{F} = F^1 \times \dots \times F^n$. When reporting on the expected surplus of the algorithm we will often take the distribution as implicit and denote the expected surplus of an algorithm on types from the distribution as $\mathcal{A} = \mathbf{E}_{\mathbf{t} \sim \mathbf{F}}[\mathcal{A}(\mathbf{t})]$ and the expected contribution to this surplus by agent κ as $\mathcal{A}^\kappa = \mathbf{E}_{\mathbf{t} \sim \mathbf{F}}[v(t^\kappa, \mathcal{A}(\mathbf{t}))]$.

From agent κ 's perspective, the algorithm and payment rule are composed with the other agent types which are drawn from the joint distribution $\mathbf{F}^{-\kappa}$. We will denote by $\mathcal{A}(t^\kappa)$ the distribution over outcomes when κ 's type is t^κ and the other agents' types are drawn from the distribution $\mathbf{F}^{-\kappa}$. Agent κ 's value for the outcome distribution $\mathcal{A}(t^\kappa)$ is thus $v(t^\kappa, \mathcal{A}(t^\kappa)) = \mathbf{E}_{\mathbf{t} \sim \mathbf{F}}[v(t^\kappa, \mathcal{A}(\mathbf{t})) | t^\kappa]$. Similarly, we will denote by $p^\kappa(t^\kappa) = \mathbf{E}_{\mathbf{t} \sim \mathbf{F}}[p^\kappa(\mathbf{t}) | t^\kappa]$ the expected payment of agent κ with type t^κ when the other agents' types are drawn from the distribution.

A mechanism $(\mathcal{A}, \mathbf{p})$ is *Bayesian incentive compatible* (BIC) if for all agents κ , and all types $t^\kappa, s \in \mathcal{T}^\kappa$, when all agents except κ truthfully report their types then κ 's expected utility is

maximized by truthful reporting as well, i.e.,

$$v(t^\kappa, \mathcal{A}(t^\kappa)) - p^\kappa(t^\kappa) \geq v(t^\kappa, \mathcal{A}(s)) - p^\kappa(s).$$

For instance, the well known Vickrey-Clarke-Groves (VCG) mechanism (Vickrey, 1961; Clarke, 1971; Groves, 1973) selects the outcome that maximizes the social surplus, i.e., $\text{OPT}(\mathbf{t})$, and assigns each agent κ a payment equal to the *externality that κ imposes on the other agents*, i.e., $p^\kappa = \text{OPT}(\mathbf{t}^{-\kappa}) - \text{OPT}^{-\kappa}(\mathbf{t})$ where the first term denotes the surplus of the optimal outcome that excludes agent κ and the second term denotes the surplus of the optimal outcome with agent κ but does not include κ 's value in the sum.

Unlike (dominant strategy) IC mechanisms that, for all profiles of bids, must have a truthful reporting best response, for BIC mechanisms all that is relevant is the outcome behavior of the composition of the mechanism with the distribution of values of the other agents. The following theorem rephrases the well-known characterization of Bayesian implementable allocation rules via cyclic monotonicity (Rochet, 1987).

Theorem 2.1. *For an algorithm $\mathcal{A}$, there exists $\mathbf{p}$ such that mechanism $(\mathcal{A}, \mathbf{p})$ is BIC if and only if the following maximum weighted matching property is satisfied on an induced graph on any finite subset of the agent types. In the weighted bipartite graph given by*

- *left-hand-side vertices: any finite subset of the agent's types $\mathcal{T}' \subset \mathcal{T}$,*
- *right-hand-side vertices: the corresponding multiset of outcome distributions*

$$\mathcal{X}' = \{X = \mathcal{A}(t) \mid t \in \mathcal{T}'\}, \text{ and}$$

- *edge weights: $v(t, X)$ between every $t \in \mathcal{T}'$ and $X \in \mathcal{X}'$*

the maximum weighted matching is the one induced by $\mathcal{A}$, i.e., t matched to $X = \mathcal{A}(t)$.

From this perspective the various types of an agent are competing amongst themselves for potential outcomes of the algorithm. For the algorithm to be BIC, it must resolve this intra-agent competition to maximize expected social surplus. For finite type spaces, appropriate payments can be calculated from the maximum weighted matching on the full type space $\mathcal{T}' = \mathcal{T}$ as they would be by VCG.

We will be giving reductions from BIC mechanism design to algorithm design. In other words, we will take an algorithm $\mathcal{A}$ and convert it into a mechanism $\mathcal{M}$ that is BIC. Importantly, while the algorithm $\mathcal{A}$ may not satisfy the maximum weighted matching property, the mechanism $\mathcal{M}$ will. While our goal is to perform this transformation and preserve the expected performance of the algorithm, we will only achieve this goal approximately. We refer to ε as the *additive loss* of our reduction when $\mathcal{M}$'s performance satisfies $\mathcal{M} \geq \mathcal{A} - \varepsilon$. Our reductions will allow the designer to make the additive loss arbitrarily small at the expense of increasing the running time required to compute the outcome and payments of the mechanism.

We will evaluate running times of algorithms in terms of the input size, N , which is considered to be the number of discrete elements in the input. For a discrete n -agent type space, where valuation functions map an agent type and outcome to a valuation, the most direct way to represent the type space is by explicitly specifying the valuation function for each agent. The number of discrete elements in such an input is $N = |\mathcal{X}| \cdot \sum_{\kappa} |\mathcal{T}^\kappa|$. This definition of the size of the input is appropriate assuming that standard numerical operations can be performed in constant time.

We consider two computational models: an *ideal model* that makes a strong (but computationally unrealistic) assumptions about our ability to exactly compute expected values, and a *black-box*

model in which our only access to the algorithm $\mathcal{A}$ and distribution $\mathbf{F}$ is by sampling and black-box evaluation, i.e., we can draw a random type profile $\mathbf{t}$ from $\mathbf{F}$, sample outcome $x \sim \mathcal{A}(\mathbf{t})$, and evaluate the value $v(t, x)$ that an agent with a given type t derives from a specific outcome x . We present our reduction for the conceptually simpler ideal model in Section 3 and explain how to extend most of these results to the black-box model in Section 4.

3 The Ideal Model

In this section we make the simplifying, but computationally unrealistic (see Theorem 4.2), assumption that we have access to an oracle that gives the expected value that an agent derives from the outcome of the algorithm on a distribution of agent types. That is, for any two types $r^\kappa, t^\kappa \in \mathcal{T}^\kappa$, $v(r^\kappa, \mathcal{A}(t^\kappa)) = \mathbf{E}_{\mathbf{t}}[v(r^\kappa, \mathcal{A}(\mathbf{t})) \mid t^\kappa]$ is given.

3.1 Surrogate Selection Rules

At the heart of our reduction is a *surrogate selection rule* Γ that, for a given agent type, selects a distribution over types (*surrogates*) from which to draw a replacement for the agent’s real type when the algorithm is run. We will typically specify a surrogate selection rule as a randomized algorithm that for any type outputs a surrogate type. The surrogate selection rule is then employed in the following generic reduction that converts an algorithm into a mechanism (essentially from Hartline and Lucier, 2010).

Definition 3.1. *The Γ reduction is:*

1. *Apply the surrogate selection rule Γ independently to each agent’s type to get a surrogate type profile.*
2. *Call $\mathcal{A}$ on the profile of surrogate types.*
3. *Output the outcome of $\mathcal{A}$ and appropriate payments.*

As in Hartline and Lucier (2010), this generic reduction will result in a good mechanism if the surrogate selection rule satisfies the following three properties.

1. *Distribution preservation:* When applying the surrogate selection rule to a random type from the distribution, the resulting distribution of the surrogate type is identical to the original distribution of the type.
2. *(Approximate) surplus preservation:* The expected valuation of an agent in the algorithm approximately improves by the mapping.
3. *Bayesian maximum weighted matching property:* The composition of the algorithm (with other agent types drawn from the distribution) with the surrogate selection rule satisfies the maximum weighted matching property (from Theorem 2.1).

Notice that the distribution preservation condition will imply that we can perform the surrogate selection rule independently to each agent and no agent notices that the other agents’ types have been transformed. The Bayesian maximum weighted matching property, then, implies that the reduction is BIC.

It remains to describe a surrogate selection rule that satisfies the above conditions and discuss payment computation. These operations pertain to each agent independently; therefore, we proceed

by considering a single agent and assume the algorithm is hard-wired with the distribution of types from other agents. For convenience, since we consider only a single agent, we drop the agent index κ from all notations.

Definition 3.2. *The Replica-Surrogate Matching (RSM) surrogate selection rule is implemented by the following randomized algorithm, given agent's type t and an integer parameter m :*

1. *Pick i^* uniformly at random from $[m]$.²*
2. *Sample the replica type profile, $\mathbf{r}$, an m -tuple of types with $r_{i^*} = t$, the real agent's type, and with each remaining replica type in $\mathbf{r}_{-i^*}$ drawn independently and identically from type distribution F .*
3. *Sample the surrogate type profile, $\mathbf{s}$, an m -tuple of types with each surrogate type independent and identically drawn from type distribution F .*
4. *Define the surrogate outcome profile as $\mathbf{X} = (X_1, \dots, X_m)$ where $X_j = \mathcal{A}(s_j)$ is the distribution over outcomes obtained by running the algorithm on s_j .*
5. *Define an edge-weighted bipartite graph between replicas (left-hand side) and surrogate outcomes (right-hand side) with the weight of the edge between replica i and surrogate outcome j equal to $v(r_i, X_j)$.*
6. *Simulate the VCG mechanism to select a matching between replicas (as agents) and surrogate outcomes (as items), i.e., compute the maximum weighted matching and appropriate payments.*
7. *Select the surrogate to whose outcome the real agent's type, i^* , is matched.*

The appropriate payment for the replica-surrogate matching reduction is the VCG payment for the real agent's type calculated in Step 6.

We argue that this reduction is distribution preserving and Bayesian incentive compatible.

Lemma 3.1. *RSM is distribution preserving.*

Proof. Each surrogate is i.i.d. from F . Each surrogate is matched to some replica. Using the principle of deferred decisions, we now pick the replica i^* that corresponds to the real type. Since this choice of replica is uniform from $[m]$, then the selection of surrogate types is uniform from $[m]$, and therefore the distribution of the selected surrogate type is F . $\square$

Theorem 3.1. *In the ideal model, RSM is Bayesian incentive compatible.*

Proof. Consider an agent. VCG on the constructed bipartite matching problem is incentive compatible for any instantiation of $\mathbf{r}$ and $\mathbf{s}$ under the assumption that the expected value that any type r_i derives from being matched to surrogate outcome X_j is exactly $v(r_i, X_j)$. This property holds in the Bayesian sense, i.e., when the surrogate types of other agents are drawn from their respective distributions, as happens when the other agents bid truthfully. (Here we are applying the distribution-preservation property, Lemma 3.1.) Therefore, conditioning on the replica and surrogate type profiles, $\mathbf{r}$ and $\mathbf{s}$, the mechanism is Bayesian incentive compatible. The theorem now follows by averaging over the random choice of $\mathbf{r}$ and $\mathbf{s}$. $\square$

---

²We sample i^* uniformly to ensure uniform-random tie breaking in the matching algorithm and so that the proof that the transformation is distribution preserving follows more transparently from the principle of deferred decisions. If our matching algorithm already breaks ties uniformly at random, then any choice of i^* would be equivalent.

3.2 Social Surplus of the RSM Reduction

We now turn to evaluating the social surplus. We will compare three quantities which can be understood in terms of matchings between replicas, surrogates, and surrogate outcomes.

- The *maximum-surplus replica-to-surrogate-outcome* matching is the one that is calculated by the RSM surrogate selection rule. In this matching, the weight of the edge between a replica and a surrogate is given by the value that the replica has for the outcome the surrogate obtains from the algorithm. In other words, the weight between r and s is $v(r, \mathcal{A}(s))$. Denote the set of edges in this maximum matching by $\Pi(\mathbf{r}, \mathcal{A}(\mathbf{s}))$ and the surplus of the matching by $v(\mathbf{r}, \mathcal{A}(\mathbf{s})) = \sum_{(r,s) \in \Pi(\mathbf{r}, \mathcal{A}(\mathbf{s}))} v(r, \mathcal{A}(s))$. The surplus of the RSM reduction is equal to a $\frac{1}{m}$ fraction of the surplus of this matching.
- The *identity surrogate-to-surrogate-outcome* matching is the one given by assigning each surrogate to its own surrogate outcome. The surplus of this matching is denoted by $v(\mathbf{s}, \mathcal{A}(\mathbf{s})) = \sum_{s \in \mathbf{s}} v(s, \mathcal{A}(s))$. The surplus of the algorithm is equal to a $\frac{1}{m}$ fraction of the surplus of this matching.
- The *minimum-swap-disutility replica-to-surrogate* matching considers edge weights equal to how much the replica and surrogate dislike swapping outcomes from the algorithm. This swap disutility, i.e.,

$$d_{\mathcal{A}}(r, s) = v(r, \mathcal{A}(r)) - v(r, \mathcal{A}(s)) + v(s, \mathcal{A}(s)) - v(s, \mathcal{A}(r)), \quad (3.1)$$

measures how different the replica and surrogate are with respect to the outcome of the algorithm. Denote the edges in this minimum matching by $\Pi(\mathbf{r}, \mathbf{s})$ and the cumulative swap disutility (henceforth: cost) of the matching by $d_{\mathcal{A}}(\mathbf{r}, \mathbf{s}) = \sum_{(r,s) \in \Pi(\mathbf{r}, \mathbf{s})} d_{\mathcal{A}}(r, s)$.

We can swap the roles of the replicas and surrogates to define the *maximum-surplus surrogate-to-replica-outcome* matching, $\Pi(\mathbf{s}, \mathcal{A}(\mathbf{r}))$ with surplus $v(\mathbf{s}, \mathcal{A}(\mathbf{r}))$, and the *identity replica-to-replica-outcome* matching with surplus $v(\mathbf{r}, \mathcal{A}(\mathbf{r}))$; by symmetry $\mathbf{E}[v(\mathbf{r}, \mathcal{A}(\mathbf{s}))] = \mathbf{E}[v(\mathbf{s}, \mathcal{A}(\mathbf{r}))]$ and $\mathbf{E}[v(\mathbf{s}, \mathcal{A}(\mathbf{s}))] = \mathbf{E}[v(\mathbf{r}, \mathcal{A}(\mathbf{r}))]$. Of the next two lemmas, the first relates the surplus of the first two matchings to the cost of the last matching and the second relates the surplus of the first two matchings to the surplus of the algorithm and mechanism.

Lemma 3.2. *The difference between the expected surpluses of the maximum-surplus replica-to-surrogate-outcome matching and the identity surrogate-to-surrogate-outcome matching is at most half the expected cost of the minimum-swap-disutility replica-to-surrogate matching. That is, $\mathbf{E}[v(\mathbf{s}, \mathcal{A}(\mathbf{s})) - v(\mathbf{r}, \mathcal{A}(\mathbf{s}))] \leq \frac{1}{2} \mathbf{E}[d_{\mathcal{A}}(\mathbf{r}, \mathbf{s})]$.*

Proof. Consider the minimum-swap-disutility replica-to-surrogate matching and separate the terms in the definition of swap disutility as summed across the edges in the matching. The cost of this matching is:

$$\begin{aligned} d_{\mathcal{A}}(\mathbf{r}, \mathbf{s}) &= \sum_{(r,s) \in \Pi(\mathbf{r}, \mathbf{s})} d_{\mathcal{A}}(r, s) \\ &= \sum_{r \in \mathbf{r}} v(r, \mathcal{A}(r)) - \sum_{(r,s) \in \Pi(\mathbf{r}, \mathbf{s})} v(r, \mathcal{A}(s)) \\ &\quad + \sum_{s \in \mathbf{s}} v(s, \mathcal{A}(s)) - \sum_{(r,s) \in \Pi(\mathbf{r}, \mathbf{s})} v(s, \mathcal{A}(r)). \end{aligned} \quad (3.2)$$

The matching $\Pi(\mathbf{r}, \mathcal{A}(\mathbf{s}))$ maximizes the value of the second term over all matchings, including $\Pi(\mathbf{r}, \mathbf{s})$; the matching $\Pi(\mathbf{s}, \mathcal{A}(\mathbf{r}))$ maximizes the value of the fourth term over all matchings, including $\Pi(\mathbf{r}, \mathbf{s})$. Thus,

$$\begin{aligned} d_{\mathcal{A}}(\mathbf{r}, \mathbf{s}) &\geq \sum_{r \in \mathbf{r}} v(r, \mathcal{A}(r)) - \sum_{(r, s) \in \Pi(\mathbf{r}, \mathcal{A}(\mathbf{s}))} v(r, \mathcal{A}(s)) \\ &\quad + \sum_{s \in \mathbf{s}} v(s, \mathcal{A}(s)) - \sum_{(s, r) \in \Pi(\mathbf{s}, \mathcal{A}(\mathbf{r}))} v(s, \mathcal{A}(r)) \\ &= v(\mathbf{r}, \mathcal{A}(\mathbf{r})) - v(\mathbf{r}, \mathcal{A}(\mathbf{s})) + v(\mathbf{s}, \mathcal{A}(\mathbf{s})) - v(\mathbf{s}, \mathcal{A}(\mathbf{r})). \end{aligned} \tag{3.3}$$

By the symmetry between replicas and surrogates, the first and third terms have the same expectation and the second and fourth terms have the same expectation. Thus,

$$\mathbf{E}[d_{\mathcal{A}}(\mathbf{r}, \mathbf{s})] \geq 2\mathbf{E}[v(\mathbf{s}, \mathcal{A}(\mathbf{s}))] - 2\mathbf{E}[v(\mathbf{r}, \mathcal{A}(\mathbf{s}))]. \quad \square$$

Lemma 3.3. *The expected surplus of the algorithm is equal to a $\frac{1}{m}$ fraction of the expected surplus of the identity replica-to-replica-outcome matching (resp., the identity surrogate-to-surrogate-outcome matching); the expected surplus of the mechanism is equal to a $\frac{1}{m}$ fraction of the expected surplus of the maximum-surplus replica-to-surrogate-outcome matching.*

Proof. Both of these observation follow from considering the matching and allowing, as in the definition of the RSM reduction, the true type of the agent to be one of the replicas uniformly at random, i.e., with probability $\frac{1}{m}$. By symmetry the expected surpluses of the identity replica-to-replica-outcome matching and identity surrogate-to-surrogate-outcome matching are equal. $\square$

The analysis above of the minimum-swap-disutility matching of Lemma 3.2 is essentially tight. In equation (3.2) the first and third terms are fixed, so the terms being optimized are the second and fourth. The difference between equation (3.2) and (3.3) is that in (3.3) we find the optimal matchings for the two terms individually, and in (3.2) we require the same matching in both terms. The symmetry of the problem suggests that there should not be too much benefit in optimizing these individually.³

We conclude, from Lemma 3.2 and Lemma 3.3, that the loss in surplus from the RSM reduction is proportional to the minimum-swap-disutility replica-to-surrogate matching and turn our attention to bounding it. Our approach is to relate this quantity to *transportation cost*, also known as the Monge-Kantorovich metric or the Wasserstein W_1 distance (Monge, 1781; Kantorovich, 1942; Villani, 2003), a standard quantity in the stochastic optimization literature. Given a metric cost function d , which importantly satisfies the triangle inequality, and given two distributions over points, the transportation cost is the minimum expected value of $d(r, s)$ over all joint distributions of (r, s) whose two marginals are the two given distributions. In the special case when each of the two given distributions is uniform over an m -point support, the transportation cost can equivalently be defined by taking the minimum-cost matching of the two support sets—with $d(r, s)$ representing the cost of matching r with s —and scaling its cost by $\frac{1}{m}$. Minimum swap utility is not a metric; e.g., even for single-dimensional types it does not satisfy the triangle inequality. Therefore, to connect our analysis question to transportation cost, we bound the swap-disutility of two types by their maximum distinguishability over all outcomes and show that this upper bound is indeed a metric.

---

³The loss in the RSM reduction is the worst for algorithms that already satisfy the maximum weight matching property of Theorem 2.1. For single-dimensional agents, this property is equivalent to monotonicity and implies that the two matchings $\Pi(\mathbf{r}, \mathcal{A}(\mathbf{s}))$ and $\Pi(\mathbf{s}, \mathcal{A}(\mathbf{r}))$ are the same.

Definition 3.3. For valuation function $v(\cdot, \cdot)$, the swap-disutility of type pair (t, t') on a pair of outcome (x, x') is

$$d(t, t', x, x') = v(t, x) - v(t, x') + v(t', x') - v(t', x)$$

and the (maximum) distinguishability of type pair (t, t') over pairs of outcomes from outcome space $\mathcal{X}$ is

$$d(t, t') = \max\{d(t, t', x, x') : x, x' \in \mathcal{X}\}.$$

The distinguishing outcome pair for (t, t') is the pair (x, x') identified in the maximum.

Lemma 3.4. Distinguishability is a metric.

Proof. We verify the metric properties:

1. Distinguishability is non-negative, i.e., $d(t, t') \geq 0$. For any pair of types (t, t') and outcomes (x, x') , either the swap disutility is non-negative, or swapping the roles of x and x' gives non-negative swap disutility. Thus, the largest swap disutility for any pair of outcomes is non-negative.
2. Distinguishability satisfies the identity property, i.e., $d(t, t) = 0$.⁴ The disutility from swapping outcomes of any type with itself is zero.
3. Distinguishability is symmetric, i.e., $d(t, t') = d(t', t)$. For the distinguishing outcome pair (x, x') for type pair (t, t') , $d(t, t', x, x') = d(t', t, x', x)$.
4. Distinguishability satisfies the triangle inequality, i.e., $d(t, t') + d(t', t'') \geq d(t, t'')$. Consider types t, t' , and t'' and outcomes x and x'' where (x, x'') is the distinguishing outcome pair for (t, t'') . We have,

$$\begin{aligned} d(t, t') + d(t', t'') &\geq d(t, t', x, x'') + d(t', t'', x, x'') \\ &= v(t, x) - v(t, x'') + v(t', x'') - v(t', x) \\ &\quad + v(t', x) - v(t', x'') + v(t'', x'') - v(t'', x) \\ &= d(t, t'', x, x'') + d(t', t', x, x'') \\ &= d(t, t''). \end{aligned}$$

□

The transportation cost is the extension of the distinguishability of pairs of types to pairs of type profiles via its minimum cost matching, normalized by the factor $\frac{1}{m}$. The subsequent proposition formally relates the transportation cost to the loss in surplus of the RSM reduction.

Definition 3.4. For two m -tuples $\mathbf{r} = (r_1, \dots, r_m)$ and $\mathbf{s} = (s_1, \dots, s_m)$, the transportation cost is

$$d(\mathbf{r}, \mathbf{s}) = \min \left\{ \frac{1}{m} \sum_{(r,s) \in \Pi} d(r, s) \mid \Pi \text{ a perfect matching} \right\}.$$

---

⁴Technically, distinguishability is a *pseudometric* as distinct types, when measured relative to an outcome space $\mathcal{X}$, may look identical, i.e., with $d(t, t') = 0$. The distinction between metrics and pseudometrics plays no role in our analysis.

Proposition 3.1. *For the mechanism $\mathcal{M}$ from the RSM reduction on algorithm $\mathcal{A}$, the contribution to the surplus from agent κ satisfies,*

$$\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - \frac{1}{2} \mathbf{E}[d^\kappa(\mathbf{r}^\kappa, \mathbf{s}^\kappa)],$$

and the total surplus satisfies,

$$\mathcal{M} \geq \mathcal{A} - \frac{1}{2} \sum_{\kappa=1}^n \mathbf{E}[d^\kappa(\mathbf{r}^\kappa, \mathbf{s}^\kappa)],$$

where $d^\kappa(\mathbf{r}^\kappa, \mathbf{s}^\kappa)$ denotes the transportation cost on the replicas and surrogates in the RSM surrogate selection rule for agent κ .

Proof. The surplus of the algorithm is equal to the sum of the surplus from each agent less the cost of the outcome produced by the algorithm; likewise for the mechanism. For each agent Lemma 3.2 and Lemma 3.3 relate the surplus loss of the RSM reduction to the minimum-swap-disutility replica-to-surrogate matching. Both the mechanism and the algorithm produce the same distribution over outcomes and thus the expected costs are the same, i.e., $\mathbf{E}[c(\mathcal{A}(\mathbf{t}))] = \mathbf{E}[c(\mathcal{M}(\mathbf{t}))]$. $\square$

Proposition 3.1 reduces the problem of bounding the surplus of RSM reductions to the problem of bounding the expected transportation cost between random m -tuples of points in a metric space. This problem has seen extensive study in the literature especially for standard spaces, such as the multi-dimensional Euclidean space, and standard distributions, such as the uniform distribution. Our metric is neither Euclidean nor is our distribution uniform; therefore, our problem requires novel analysis. We defer the specialized analysis of transportation cost to Section 5 and summarize here three consequences of that analysis for the performance of the RSM reduction. The first applies to single-dimensional settings, the second applies to arbitrary type spaces, and the third applies when the distinguishability metric on type space $\mathcal{T}$ has bounded *doubling dimension*. (This last case generalizes multi-dimensional Euclidean space, see Definition 3.5 below.) These transportation cost results are of independent economic interest: they provide a specific type of quantitative answer to the question, “How many random samples from a continuous type distribution are necessary to construct a discrete approximation that is accurate from the standpoint of Bayesian mechanism design?”

The loss in the RSM reduction improves with m , the number of replicas and surrogates drawn. Therefore, we will phrase our bounds in terms of how big m needs to be to guarantee that the loss from this agent κ is at most $O(\varepsilon)$, i.e., m so that $\mathcal{M}$ from the RSM reduction on $\mathcal{A}$ has expected surplus satisfying $\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - c \cdot \varepsilon$, for some constant c which is specified explicitly in each theorem statement. We start with single-dimensional type spaces.

Theorem 3.2. *For any single-dimensional agent and*

$$m \geq \frac{1}{4\varepsilon^2}$$

replicas and surrogates in the RSM reduction, the loss from this agent is at most ε .

To assess the loss for general type spaces, imagine that the type space can be partitioned into $\eta(\varepsilon)$ pieces of diameter ε . Then the loss from matching replicas and surrogates in the same piece is at most ε . If there is imbalance in this piece, i.e., more replicas than surrogates or vice versa, then these replicas or surrogates will have to be matched to ones outside the piece with distance at most 1. The next theorem answers the question of how large m needs to be to guarantee that this additional loss is also at most ε on average. The total average loss would then be 2ε .

Theorem 3.3. *For any agent with type space $\mathcal{T}$ that can be partitioned into $\eta(\varepsilon)$ pieces, each of diameter at most ε , and*

$$m \geq \frac{\eta(\varepsilon)}{4\varepsilon^2}$$

replicas and surrogates in the RSM reduction, the loss from this agent is at most 2ε .

To state our third result, we need to define the doubling dimension of a metric space. This notion of dimension matches other standard notions of dimension (up to a constant factor) for spaces with non-pathological geometry. For example, any subset of $\mathbb{R}^\Delta$ under the L^p -norm, for $1 \leq p \leq \infty$, has doubling dimension $O(\Delta)$.

Definition 3.5 (Assouad, 1983). *The doubling dimension of a metric is the smallest constant Δ such that every bounded subset S can be partitioned into at most 2^Δ subsets, each having diameter at most half of the diameter of S .*

We now proceeded similarly to the setup for Theorem 3.3 with partitions of type space into pieces with diameter ε . Because the type space has doubling dimension Δ , these pieces can be clustered in groups of size 2^Δ with the diameter of the cluster being at most 2ε , etc. Imbalance at any resolution can then be resolved at the next higher resolution at twice the cost. The total cost of the matching then is at most ε for the bottom level, and we pick m so that the average contribution to the matching from all other levels is at most ε as well.

Theorem 3.4. *For any agent with type space $\mathcal{T}$ that has doubling dimension $\Delta \geq 2$ and*

$$m \geq \frac{1}{2\varepsilon^{\Delta+1}}$$

replicas and surrogates in the RSM reduction, the loss from this agent is at most 2ε (with ε a power of two).⁵

Since the techniques used in the proofs of our transportation cost results are orthogonal to the issues of incentives in the RSM reduction, we consolidate the discussion of transportation cost and give full proofs of the theorems above in Section 5.

3.3 Discrete Type Spaces

In the ideal model, settings with discrete type spaces can be solved with no loss by a reduction whose running time is polynomial in the combined cardinality of the type spaces. The reduction is a variant on the RSM reduction, though we will not make the connection precise. It is based on solving a fractional assignment problem and, for convenience, we will adopt the terminology of replica and surrogate even though neither the replicas nor the surrogates are random draws from the distribution.

Without loss of generality the type space is $\mathcal{T} = [m]$ and type $t \sim F$ occurs with probability F_t .

Definition 3.6. *The Replica-Surrogate Fractional Assignment (RSF) surrogate selection rule for type space with cardinality m is:*

1. *For each distinct type $t \in [m]$, let replica $r_t = t$, surrogate $s_t = t$, and surrogate outcome $X_t = \mathcal{A}(s_t)$.*

---

⁵~~2. Solve the following fractional assignment problem (P) and its dual (D).⁶~~
Theorem 3.4 can be extended to $\Delta \in (1, 2)$ requiring $m \geq \frac{1}{2(\Delta-1)\varepsilon^{\Delta+1}}$ replicas and surrogates.

⁶The primal is for selecting a surrogate; the dual gives the appropriate payments.

$$\begin{array}{ll}
\max & \sum_{i,j} F_i v(r_i, X_j) q_{ij} \\
s.t. & \sum_j q_{ij} = 1 \quad \forall i \\
& \sum_i F_i q_{ij} = F_j \quad \forall j \\
& q_{ij} \geq 0 \quad \forall i, j
\end{array}
\qquad
\begin{array}{ll}
\min & \sum_i \mu_i + \sum_j F_j \pi_j \\
s.t. & \mu_i + F_i \pi_j \geq F_i v(r_i, X_j) \quad \forall i, j
\end{array}$$

Let $\{q_{ij}\}$ and $\{\mu_i\}, \{\pi_j\}$ denote optimal solutions to the primal and dual problems, respectively.

3. Elicit the agent's declared type, t , and replace it with a randomly sampled surrogate, selecting surrogate type s_j with probability q_{tj} .
4. The payment charged to the agent is $\sum_j \pi_j q_{tj} + \min_i \{\frac{\mu_i}{F_i}\}$.

The RSF reduction closely resembles the RSM reduction, with the crucial difference that in RSM the true type is on the replica side of the matching, and if the agent deviates then the matching problem changes. On the other hand, for RSF, the fractional assignment problem can be solved before the types are considered, and if the agent deviates the assignment problem is unchanged. After the assignment is determined, it must be that the true type would prefer its corresponding outcome rather than the outcome of another replica; in other words, the outcome produced must be envy free (see Gul and Stacchetti, 1999, or MuAlem, 2009). The payment rule defined in Step 4 of the reduction is designed to guarantee an envy-free outcome.

Theorem 3.5. *In the ideal model, RSF is Bayesian incentive compatible.*

Proof. The constraints of the fractional assignment problem (P) ensure that the surrogate selection rule is stationary, so if all other agents are reporting types truthfully then $v(r_i, X_j)$ accurately reflects the expected valuation of type r_i for the allocation chosen by the mechanism conditional on s_j being selected as its surrogate.

Therefore, to verify Bayesian incentive compatibility, it suffices to prove that for any pair of types (t, s) , the following inequality holds, in which the left side represents the utility of an agent of type t when reporting its type truthfully, and the right side represents the utility when misreporting its type as s . We also define $v_{ij} = v(r_i, X_j)$, $\varrho_i = \frac{\mu_i}{F_i}$, and $\varrho_{\min} = \min_{i \in [m]} \{\varrho_i\}$.

$$\sum_j (v_{tj} - \pi_j) q_{tj} - \varrho_{\min} \geq \sum_j (v_{tj} - \pi_j) q_{sj} - \varrho_{\min}. \quad (3.4)$$

Dual feasibility implies that $v_{tj} - \pi_j \leq \varrho_t$ for all t, j , and complementary slackness implies that $v_{tj} - \pi_j = \varrho_t$ whenever $q_{tj} > 0$. Consequently we have

$$\sum_j (v_{tj} - \pi_j) q_{sj} - \varrho_{\min} \leq \sum_j \varrho_t q_{sj} - \varrho_{\min} = \varrho_t - \varrho_{\min} \quad (3.5)$$

$$\sum_j (v_{tj} - \pi_j) q_{tj} - \varrho_{\min} = \sum_j \varrho_t q_{tj} - \varrho_{\min} = \varrho_t - \varrho_{\min} \quad (3.6)$$

The equalities in equations (3.5) and (3.6) follow from $\sum_i q_{tj} = 1$ for all t . The Bayesian incentive compatibility constraint (3.4) follows immediately by combining (3.5) and (3.6). Note that interim individual rationality also follows from (3.6) since the left side represents the agent's utility when bidding truthfully, and the right side is non-negative. $\square$

The fractional assignment problem (P) and its dual (D) formulated in Step 2 can be solved in polynomial time, for example by reducing to a minimum-cost flow problem (see the textbook by Ahuja et al., 1993). Furthermore, the expected surplus of the resulting mechanism is equal to the value of the optimum of (P), which is at least the value of the “identity assignment” defined by $q_{ij} = 1$ if $i = j$, and 0 otherwise. The value of this identity assignment is the expected surplus of the original algorithm. Summarizing these observations, we have the following theorem.

Theorem 3.6. *In the ideal model in discrete settings with finite type spaces, the mechanism $\mathcal{M}$ from the RSF reduction on algorithm $\mathcal{A}$ has expected surplus satisfying $\mathcal{M} \geq \mathcal{A}$ and runs in polynomial time in m , the number of distinct types, and n , the number of agents.*

4 The Black-box Model

In this section we explain how to modify the RSM and RSF reductions to achieve incentive compatibility in the black-box model, in which we can only access the algorithm $\mathcal{A}$ by evaluating it at a given type profile. In particular, we cannot directly evaluate the expectation of $v(t^\kappa, \mathcal{A}(t^\kappa, \mathbf{t}^{-\kappa}))$ over a random type profile $\mathbf{t}^{-\kappa}$, as is allowed in the ideal model. Instead, such quantities must be estimated by sampling. The main difficulty, then, lies in designing a mechanism that satisfies Bayesian incentive compatibility despite the inevitability of small sampling error.

As before, we focus on a single agent κ and omit its index; $\mathcal{A}(t)$ denotes the distribution of outcomes obtained by evaluating the algorithm $\mathcal{A}$ on the agent with type t assuming all other agent types are random from the distribution.

Consider the following definitions.

Definition 4.1. *For a given positive integer L , the empirical estimation procedure Est for outcome distribution X is the following. Draw L outcomes $x_1, \dots, x_L$ independently and identically from X . The outcome estimate $\tilde{X} = \text{Est}(X)$ is the uniform distribution over these L sampled outcomes. The value estimate for type t and outcome estimate $\tilde{X}$ is $v(t, \tilde{X}) = \frac{1}{L} \sum_{j=1}^L v(t, x_j)$.*

Definition 4.2. *The Estimated Replica-Surrogate Matching (ERSM) surrogate selection rule is identical to RSM except with outcome distribution $X_j = \mathcal{A}(s_j)$ replaced with outcome estimate $\tilde{X}_j = \text{Est}(\mathcal{A}(s_j))$ for all $j \in [m]$. Similarly, for discrete settings define ERSF for RSF using outcome estimates in place of outcome distributions.*

Unfortunately the ERSM and ERSF reductions do not generally give BIC mechanisms. We first show that the ideal model is computationally intractable given only black-box access to the algorithm. Next we show that in single-dimensional settings a variant of ERSM with (to be defined) *binomial estimates* is BIC. We then turn to discrete settings and show that a modification of the ERSF mechanism, in combination with a (to be defined) strictly incentive compatible mechanism, is BIC. Finally, for general continuous settings, as we know of no general BIC reduction, we show that ERSM itself is in fact ε -BIC for a suitable choice of L .

In our analysis we will make use of the following Chernoff-Hoeffding tail bound for sums of independent random variables.

Theorem 4.1 (Hoeffding, 1963). *For distribution over outcomes X , estimate $\tilde{X} = \text{Est}(X)$ from L independent random draws from X , type t , small $\zeta > 0$, and valuation function $v(\cdot, \cdot)$ normalized to $[0, 1]$,*

$$\Pr[|v(t, \tilde{X}) - v(t, X)| \geq \zeta] \leq 2e^{-2L\zeta^2}.$$

4.1 Intractability of the Ideal Model

To substantiate the claim that the ideal model makes unrealistic computational assumptions, this section examines the problem of evaluating the expectation of $v(t^\kappa, \mathcal{A}(t^\kappa, \mathbf{t}^{-\kappa}))$ over a random type profile $\mathbf{t}^{-\kappa}$. We shall relate this problem to the computational complexity class #P (pronounced “sharp P”), a class of combinatorial counting problems that contains problems that are NP-hard. In fact, every problem that is #P-hard (meaning: at least as hard as any problem in the class #P) is also NP-hard, and it is widely believed that #P-hard problems are strictly harder than any problem in the class NP. The proof of the following theorem can be found in Appendix A.

Theorem 4.2. *The problem of computing the expected valuation of an agent with a given type in an algorithm, when the types of other agents are drawn from a product distribution, is #P-hard.*

4.2 Single-dimensional Types

Consider single-dimensional settings where, without loss of generality, $\mathcal{X} \subseteq [0, 1]^n$, $\mathcal{T}^\kappa = [0, 1]$, and $v(t^\kappa, x) = t^\kappa x^\kappa$ for $x = (x^1, \dots, x^n)$. We view x^κ as the service level of agent κ .

We first show that ERSM does not generally result in a BIC mechanism.

Lemma 4.1. *In single-dimensional settings, there exists an algorithm $\mathcal{A}$ and parameters m and L such that the ERSM reduction does not result in a BIC mechanism.*

Proof. Consider ERSM with m replicas and surrogates and $L = 1$ outcome estimates for each surrogate.⁷

The algorithm $\mathcal{A}$ that satisfies the lemma has the following structure. An agent with a high type $t > 1/2$ gets a high outcome: With probability 1, $\mathcal{A}(t) = 4/m$. An agent with a low type $t \leq 1/2$ gets a low outcome: With probability $2/m$, $\mathcal{A}(t) = 1$ and otherwise $\mathcal{A}(t) = 0$ (the expected service level of a low outcome is $2/m$). Notice that $\mathcal{A}$ is monotone, i.e., higher types are assigned outcomes with higher expected service levels. This monotonicity is necessary for incentive compatibility.

Assume that the agent’s type distribution is uniform on $[0, 1]$. Since $L = 1$, each $\tilde{X}_j$ is obtained by simply sampling a random type $s_j \in [0, 1]$ and drawing one sample from the distribution $\mathcal{A}(s_j)$. Thus, surrogates are high or low with equal probability $1/2$. The possible estimated surrogate outcomes are from $\{0, 4/m, 1\}$. Outcome estimates 0 and 1 are from low surrogates and $4/m$ outcomes are from high surrogates. In ERSM we assign replicas to estimated surrogate outcomes in order. Therefore, if there is a 1 outcome then the agent with the highest type is assigned to it. Of course, this outcome comes from a low surrogate, thus the highest agent receives a low outcome and monotonicity is contradicted.

The precise calculation works out as follows. The probability of a 1 outcome estimate for any surrogate is the probability the surrogate is low times the probability we draw the 1 outcome, i.e., $(1/2) \cdot (2/m) = 1/m$. Therefore, the probability that there is no 1 outcome among the m surrogates is $(1 - 1/m)^m \approx 1/e$; and the probability there is a 1 outcome is approximately $1 - 1/e$. Consider an agent with type $t = 1$. This agent is matched with the 1-outcome, and thus a low surrogate, if there is one, i.e., with probability approximately $1 - 1/e$, and otherwise a high surrogate. Consider an agent with type $t = 3/4$. With high probability this agent is matched to a $4/m$ outcome, and thus a high surrogate. For these two types the allocation rule of the ERSM reduction is not monotone. $\square$

⁷Similar counterexamples can be constructed for any value of L , by modifying the parameters in this counterexample to have the appropriate dependence on L .

The problem with ERSM is that the head-to-head comparison between random values from different distributions may not be consistent with the expected values of these distributions. To address this we explicitly turn each estimate distribution into the binomial distribution by redrawing each sample from the Bernoulli distribution with expectation equal to the sample value. This yields, in place of the original sample, a Bernoulli random variable with expectation equal to the true expectation.

Definition 4.3. *The binomial outcome estimate $\tilde{X} = \text{BinomEst}(X)$ for outcome distribution X is the average of L Bernoulli random variables with mean $\mathbf{E}[X]$, i.e., it is a binomial distribution with parameters L and $\mathbf{E}[X]$ scaled by $1/L$. This distribution can be sampled by the following binomial estimation procedure: Draw L outcomes $x_1, \dots, x_L \in [0, 1]$ independently and identically from X , draw L Bernoulli trials $y_1, \dots, y_L$ independently by setting $y_j = 1$ with probability x_j and 0 otherwise, and output the average of these trials $\frac{1}{L} \sum_j y_j$.*

To see that the binomial sampling procedure for X is indeed sampling according to the binomial outcome estimate distribution $\tilde{X} = \text{BinomEst}(X)$, notice that without conditioning on the x_j 's, the y_j 's are i.i.d. Bernoulli trials with success probability equal to $\mathbf{E}[X]$.

Definition 4.4. *The Binomial Estimated Replica-Surrogate Matching (BERSM) surrogate selection rule is identical to RSM except with outcome distribution $X_j = \mathcal{A}(s_j)$ replaced with outcome estimate $\tilde{X}_j = \text{BinomEst}(\mathcal{A}(s_j))$ for all $j \in [m]$.*

The surplus of the black-box model BERSM reduction is close to that of the ideal model RSM reduction.

Lemma 4.2. *The expected contribution to the surplus of agent κ in the (black-box model) BERSM mechanism $\tilde{\mathcal{M}}$ relative to the (ideal model) RSM mechanism $\mathcal{M}$ satisfies $\tilde{\mathcal{M}}^\kappa \geq \mathcal{M}^\kappa - 3\varepsilon$ when surrogate outcomes are estimated with $L \geq \frac{1}{2}\varepsilon^{-2}\ln(2m/\varepsilon)$ samples.*

Proof. For any $\varepsilon > 0$, the Chernoff-Hoeffding bound (Theorem 4.1) implies that when $L > \frac{1}{2}\varepsilon^{-2}\ln(2m/\varepsilon)$, the probability that one estimate $\tilde{X}_j$ differs from its expectation by more than ε is at most ε/m . The union bound implies that the probability of any of the m estimates deviating from its expectation by more than ε is at most ε . The surplus is at most one, so the expected surplus lost from this event is at most ε . When this event does not occur, the maximum matching on the estimates has total surplus (with respect to the true values) within $2\varepsilon m$ of the surplus of the maximum matching on the true values. The performance of a mechanism is a $\frac{1}{m}$ fraction of the surplus of the maximum matching (by Lemma 3.3). The total loss in expectation is $\mathcal{M}^\kappa - \tilde{\mathcal{M}}^\kappa \leq 3\varepsilon$. $\square$

To prove that the reduction is BIC, we use the following property of binomial distributions that we call *posterior monotonicity*.

Lemma 4.3 (Posterior Monotonicity). *Suppose p, q are real numbers with $0 \leq p \leq q \leq 1$, Z_p, Z_q are random variables with binomial distributions $B(n, p)$ and $B(n, q)$, respectively, and a, b are integers with $0 \leq a \leq b$, then*

$$\Pr(Z_p = a)\Pr(Z_q = b) \geq \Pr(Z_p = b)\Pr(Z_q = a).$$

Proof. The lemma is trivial if either p or q is equal to 0 or 1. Consequently, let us assume henceforth that $p(1-p)q(1-q) \neq 0$. Then it is easy to see that

$$\begin{aligned} \frac{\Pr(Z_p = a)\Pr(Z_q = b)}{\Pr(Z_p = b)\Pr(Z_q = a)} &= \frac{p^a(1-p)^{n-a}q^b(1-q)^{n-b}}{p^b(1-p)^{n-b}q^a(1-q)^{n-a}} \\ &= \left(\frac{q(1-p)}{p(1-q)}\right)^{b-a} = \left(1 + \frac{q-p}{p(1-q)}\right)^{b-a}, \end{aligned}$$

which is greater than or equal to 1 by our hypotheses that $p \leq q, a \leq b$. $\square$

Theorem 4.3. *In the black-box model and single-dimensional settings, BERSM is BIC.*

Proof. BERSM draws $m-1$ replicas, m surrogates, and binomial random variables for each surrogate outcome. Denote the sorted vector of binomial estimates as $\tilde{\mathbf{x}} = (\tilde{x}_1, \dots, \tilde{x}_m)$ with $\tilde{x}_j \geq \tilde{x}_{j+1}$. Consider two possible types $t > t'$ of the agent and suppose that in the sorted order of the replicas one of these types is in position j and the other is in position $j' > j$. Thus, BERSM assigns t to the surrogate that generated binomial estimate $\tilde{x}_j$ and t' to the surrogate that generated $\tilde{x}_{j'}$. Conditioning on the replicas $\mathbf{r}$, surrogates $\mathbf{s}$, and the sorted binomial estimates $\tilde{\mathbf{x}}$, the identity of the surrogate that generated each binomial estimate is still a random variable.⁸

Let $\mathcal{E}$ denote the conditioning event of these replicas, surrogates, and binomial estimates, and additionally condition on the (unordered) pair of surrogates that generated estimate pair $\{\tilde{x}_j, \tilde{x}_{j'}\}$, but not which surrogate generated which estimate. Without loss of generality, denote the pair of surrogates $(s_j, s_{j'})$ with $x = \mathcal{A}(s_j)$ and $x' = \mathcal{A}(s_{j'})$. Lemma 4.3 implies that the probability that $\tilde{x}_j$ came from s_j and $\tilde{x}_{j'}$ came from $s_{j'}$ exceeds the probability that $\tilde{x}_j$ came from $s_{j'}$ and $\tilde{x}_{j'}$ came from s_j ; thus, the former probability is at least $1/2$ and the latter probability is at most $1/2$. In the first case $\mathcal{M}(t) = x \geq x' = \mathcal{M}(t')$, in the latter case $\mathcal{M}(t) = x' \leq x = \mathcal{M}(t')$. Taking expectation of this inequality we have $\mathbf{E}[\mathcal{M}(t) | \mathcal{E}] \geq \frac{x+x'}{2} \geq \mathbf{E}[\mathcal{M}(t') | \mathcal{E}]$.

Removing the conditioning we get $\mathbf{E}[\mathcal{M}(t)] \geq \mathbf{E}[\mathcal{M}(t')]$ in general, i.e., the BERSF reduction is monotone. $\square$

Finally, we describe how to compute payments so as to satisfy BIC when using the RSM reduction combined with the binomial estimator. For a single-dimensional agent κ the outcome distribution can be summarized by the probability of service. Overloading notation for the remainder of this section, denote this probability of service for agent κ with type t in mechanism $\mathcal{M}$ by $\mathcal{M}^\kappa(t)$. According to the well-known characterization of Bayesian incentive compatibility for single-dimensional settings (Myerson, 1981), a mechanism is Bayesian incentive compatible as long as each agent κ 's probability of service $\mathcal{M}^\kappa(t)$ is monotone in t , and her expected payment when bidding t satisfies

$$\begin{aligned} \mathbf{E}[p^\kappa(t)] &= t \cdot \mathcal{M}^\kappa(t) - \int_0^t \mathcal{M}^\kappa(z) dz \\ &= t \cdot \int_0^1 [\mathcal{M}^\kappa(t) - \mathcal{M}^\kappa(wt)] dw. \end{aligned} \tag{4.1}$$

Thus, to compute an unbiased estimate of the payment for an agent, the mechanism can sample a uniformly random $w \in [0, 1]$, run the RSM reduction again replacing t with wt , and charge agent κ a price of $t \cdot [\mathcal{M}(t) - \mathcal{M}(wt)]$. The expectation of this price is equal to the right side of (4.1),

⁸In fact, assuming that no surrogate outcome is deterministically zero or one, there is a non-zero probability of any surrogate generating any of the binomial estimates.

thus ensuring Bayesian incentive compatibility. The following example demonstrates that a non-standard payment rule like the one above is necessary as the VCG payments from BERSM are not generally incentive compatible.

Example 4.1. *Consider an algorithm $\mathcal{A}$ that assigns the same probability of service of $\frac{1}{2}$ to all outcomes. Consider BERSM with m replicas and surrogates and $L = 1$ outcome estimates. Notice that $\mathcal{A}$ is weakly monotone, i.e., higher types are assigned outcomes with weakly higher expected service levels. However, as all outcomes are equal, in any BIC mechanism the expected payments of all types must be equal. In VCG on the maximum weight matching between replicas and surrogate outcome estimates, however, the expected payments of the highest and lowest valued replica are not equal whenever there are at least two distinct surrogate outcome estimates and two distinct replicas.*

4.3 Discrete Type Spaces

In this section we give a BIC reduction for discrete, finite type spaces. The main idea behind this reduction is that ERSF, with a large number of samples L to estimate the value that each replica has for each surrogate outcome, is nearly BIC. Because it is only nearly BIC, though, an agent may have a small incentive to misreport. To compensate for this, with some small probability we will run a *strictly IC mechanism* for which an agent strictly prefers truthtelling to any other report. In this strictly IC mechanism, the disincentive for an agent of type t to misreport her type as t' will be proportional to the *distinguishability* of t and t' , which quantifies the maximum amount by which type t 's preference for one outcome over another exceeds that of type t' (Definition 3.3). Unfortunately, if one merely runs ERSF using the empirical estimates of each replica's value for each surrogate, then the incentive to misreport type t as t' , and thus the incentive to misreport, is proportional to the magnitude of the types' values and not their distinguishability; the latter can be much smaller than the former. To enable the strictly IC mechanism to counterbalance the potential misreporting incentive in ERSF, it is necessary for us to modify ERSF so that the potential incentive for misreporting type t as t' is at most proportional to the distinguishability of the two types. We accomplish this by “smoothing” the ERSF allocation rule so that similar types are guaranteed to receive similar outcomes. The smoothing is achieved using a *regularizer*, a convex penalty term in the objective function. Regularizers are commonly used in machine learning algorithms for the same purpose.

The reduction of this section applies to general feasibility settings, i.e., where the designer's cost function is $c(\cdot) \in \{0, \infty\}$.⁹ Without loss of generality the type space is $\mathcal{T} = [m]$ and type $t \sim F$ occurs with probability F_t .

4.3.1 Strictly IC Mechanism

We assume oracle access to the agent's valuation function, i.e., for any type t and deterministic outcome x we can evaluate $v(t, x)$. A key step in deriving a mechanism where agents strictly prefer truthtelling is, for all pairs of types, identifying a pair of outcomes where the two types' difference in value for the pair of outcomes are maximally distinct (Recall Definition 3.3 in Section 3.2).

---

⁹Reductions based on RSM or RSF alone can be applied to settings with general costs. As the outcome of the mechanism is identically distributed to the outcome of the algorithm on a random input, the expected cost is unchanged by the reduction. In this section, the outcome may be selected by the strictly IC mechanism which does not generally preserve the distribution over outcomes of the algorithm. Restricting to $c(\cdot) \in \{0, \infty\}$, thus, simplifies the analysis. In Hartline and Lucier (2010), a similar approach is taken and the extra analysis necessary to bound general costs is given, albeit for the simpler setting of single-dimensional agents.

Each pair of types (t, t') is either distinguishable or indistinguishable relative to a set of outcomes $\mathcal{X}$. Denote by M the number of distinguishable pairs of types. Recall, there are $m = |\mathcal{T}|$ types, so $M \leq \binom{m}{2}$. The following mechanism is strictly (dominant strategy) incentive compatible. Similar mechanisms have been employed in the literature for the same purpose; this mechanism is novel in that it does not require that there is some outcome x_0 for which all types $t \in \mathcal{T}$ have value $v(t, x_0) = 0$.¹⁰

Definition 4.5. *For a pair of types (t, t') and distinguishing outcome pair (x, x') (as by Definition 3.3), the distinguishing mechanism for an agent κ with type t^κ (not necessarily equal to t or t') is:*

1. *Let $\nu = v(t, x) - v(t, x')$ and $\nu' = v(t', x) - v(t', x')$ be the amount by which t and t' prefer outcome x to x' , respectively. Note, the type pair's distinguishability is $d(t, t') = \nu - \nu' > 0$.*
2. *Define price $p = (\nu + \nu')/2$. Assume that $p \geq 0$. This can be ensured, if necessary, by interchanging the names of t with t' and x with x' .*
3. *Select the outcome that agent κ with type t^κ prefers, between x with payment p and x' with no payment; all other agents pay nothing.*

Parameterized by a set of M pairs of distinguishable types, each with a corresponding pair of distinguishing outcomes, the strictly IC mechanism for an agent κ is denoted by SIC^κ and draws one of the M pairs of distinguishable types uniformly at random, then runs the distinguishing mechanism for this pair of types.

Lemma 4.4. *The strictly IC mechanism SIC^κ for agent κ with M pairs of distinguishable types and their pairs of distinguishing outcomes is IC. Moreover, for distinguishable types (t, t') the agent with type t has utility at least $\frac{d(t, t')}{2M}$ more for reporting t than t' (and vice versa for type t'), where $d(t, t')$ is the distinguishability of (t, t') .*

Proof. All of the M distinguishing mechanisms in the support of the strictly IC mechanism are IC: we have expressed each as an offer of two outcomes (with payments) for which we select the type's favorite.

Given a pair of distinguished types (t, t') and their distinguishing outcomes (x, x') , as in the definition of the distinguishing mechanism, let the difference in values for these outcomes be $\nu = v(t, x) - v(t, x')$ and $\nu' = v(t', x) - v(t', x')$ and average difference in values be $p = (\nu + \nu')/2$. Assume $\nu > p > \nu'$. Between the two alternatives of the mechanism's outcome, type t prefers outcome x with payment p , while type t' prefers outcome x' with no payment. Each preference has a magnitude of $(\nu - \nu')/2$ which, by the distinguishability of (t, t') via distinguishing outcome pair (x, x') , is $d(t, t')/2$.

The strictly IC mechanism picks this distinguishing mechanism with probability $1/M$; thus, the type t prefers to report t over t' by at least $\frac{d(t, t')}{2M}$ (and vice versa). $\square$

4.3.2 Regularized RSF Mechanism

We begin by defining and analyzing a new surrogate selection rule for the ideal model. It is an adaptation of the RSF surrogate selection rule (Definition 3.6) where a convex regularizer is employed to make the solution of the fractional assignment problem less sensitive to small variations in the replicas' valuations.

¹⁰For example, if the outcome space is to build one of two public projects, then some agents may prefer the first, some the second, and there is no "zero" outcome.

Definition 4.6. Let ϕ denote a differentiable convex function defined on $\mathbb{R}_+^m$. The regularized replica-surrogate fractional assignment (RRSF) surrogate selection rule with regularizer ϕ is the following procedure.

1. For each distinct type $t \in [m]$, let replica $r_t = t$, surrogate $s_t = t$, and surrogate outcome, $X_t = \mathcal{A}(s_t)$.
2. Solve the following concave maximization problem, in which $\{q_{ij}\}$ is an m -by- m matrix of variables, and the notation $\mathbf{q}_i$ denotes the vector $(q_{i1}, \dots, q_{im})$.

$$\begin{aligned}
\max \quad & \sum_{i,j} F_i v(r_i, X_j) q_{ij} - \sum_i F_i \phi(\mathbf{q}_i) \\
\text{s.t.} \quad & \sum_j q_{ij} = 1 & \forall i \\
& \sum_i F_i q_{ij} = F_j & \forall j \\
& q_{ij} \geq 0 & \forall i, j.
\end{aligned} \tag{4.2}$$

Let $\{q_{ij}\}$ denote the optimum solution.

3. Let $\{\lambda_{ij}\}$, $\{\mu_i\}$, and $\{\pi_j\}$ denote Lagrange multipliers satisfying the KKT conditions

$$F_i (v(r_i, X_j) - \partial_j \phi(\mathbf{q}_i)) = \lambda_{ij} + \mu_i + F_i \pi_j \quad \forall i, j \tag{4.3}$$

$$\lambda_{ij} \leq 0 \quad \forall i, j \tag{4.4}$$

$$\lambda_{ij} q_{ij} = 0 \quad \forall i, j \tag{4.5}$$

4. Elicit the agent's declared type, t , and replace it with a randomly sampled surrogate, selecting surrogate type s_j with probability q_{tj} .
5. The payment charged to the type t agent is $p_t = \sum_j \pi_j q_{tj} + \phi(\mathbf{q}_t) - \phi(\mathbf{0}) + \min_i \frac{\mu_i}{F_i}$.

In the payment formula in Step 5, the last two terms are required for individual rationality, while the first two terms ensure incentive compatibility. At first glance, incentive compatibility may appear automatic since the outcome selected in Step 2 is *maximal in range* (cf. Nisan and Ronen, 2007). However, as in Section 3.3 incentive compatibility of the resulting mechanism corresponds to envy freedom of the outcome and payments of Steps 2 and 5. Importantly, in the optimization problem in Step 2, the summands in the second term (the regularizer term) are weighted by the type distribution. Any other weighting of those terms would yield a choice of outcome that is maximal in range, but only the weighting by type distribution allows us to prove envy freedom. The following paragraph sketches this proof.

The KKT conditions can be interpreted as local incentive constraints for the RRSF reduction. With the notation $\varrho_i = \mu_i/F_i$, notice that the KKT conditions (4.3)–(4.5) can be summarized by observing that for all i and j ,

$$v(r_i, X_j) - \pi_j - \partial_j \phi(\mathbf{q}_i) \leq \varrho_i, \tag{4.6}$$

with equality whenever $q_{ij} > 0$. Now imagine that the agent is free to choose from a menu of lotteries, where the price of any lottery $\mathbf{q}$ in the probability simplex is equal to $\sum_j \pi_j q_j + \phi(\mathbf{q}) - \phi(\mathbf{0}) + \min_{i \in [m]} \varrho_i$. An agent of type t , who assigns value $v(r_t, X_j)$ to outcome j , has utility

$$\sum_j v(r_t, X_j) q_{tj} - \sum_j \pi_j q_{tj} - \phi(\mathbf{q}_t) + \phi(\mathbf{0}) - \min_i \varrho_i$$

for receiving lottery $\mathbf{q}_t$ and paying the corresponding amount. The partial derivative of this quantity with respect to q_{tj} is equal to the left side of (4.6). An infinitesimal deviation from selecting $\mathbf{q}_t$ involves increasing some components of the probability vector $\mathbf{q}$ and decreasing some other components by an equal amount. The KKT condition (4.6) ensures that increasing components of the probability vector increases utility at a rate of at most ϱ_i , while decreasing other components of the probability vector decreases utility at exactly rate ϱ_i since those components must satisfy $q_{tj} > 0$, and hence (4.6) holds with equality. Thus, an infinitesimal deviation from selecting $\mathbf{q}_t$ cannot improve utility, i.e., utility is locally maximized by selecting lottery $\mathbf{q}_t$. Since utility is a concave function of the lottery chosen, a locally optimal choice must be globally optimal, which implies Bayesian incentive compatibility. In Appendix A.3 we formalize this argument and use it to prove the following theorem.

Theorem 4.4. *In the ideal model, RRSF is Bayesian incentive compatible.*

The usefulness of RRSF in the present context stems from a smoothness property that we now turn to proving: If two types are similar, i.e., their distinguishability is small (Definition 3.3), then they must receive similar outcomes. This property does not hold for every instance of RRSF (for example when $\phi = 0$, RRSF is identical to RSF and the distribution over surrogate outcomes for similar replicas need not be similar) but it holds when ϕ satisfies *strong convexity*. In this section we consider only the surrogate outcome distributions $\{X_1, \dots, X_m\}$ when defining the distinguishability $d(t, s)$ of a type pair (t, s) .

Definition 4.7. *For any $\gamma > 0$, a convex function ϕ is γ -strongly convex if the function $\phi(\mathbf{q}) - \frac{1}{2}\gamma\|\mathbf{q}\|^2$ is twice differentiable and convex.*

The following lemma expresses this similarity of outcomes for a pair of types in terms of their distinguishability with respect to the surrogate outcome distributions $\{X_1, \dots, X_m\}$. Its proof is given in Appendix A.3.

Lemma 4.5. *For any types $t, s \in [m]$, the corresponding surrogate distributions $\mathbf{q}_t, \mathbf{q}_s$ computed by the RRSF surrogate selection rule with a γ -strongly-convex regularizer ϕ satisfy*

$$\|\mathbf{q}_t - \mathbf{q}_s\|_1 \leq \frac{m}{\gamma} d(t, s),$$

where $d(t, s)$ is the distinguishability with respect to surrogate outcome distributions $\{X_1, \dots, X_m\}$.

We now turn to analyzing the surplus of the regularized RSF reduction. Notice that with the identically-zero regularizer $\phi(\cdot) = 0$, the RRSF and RSF reductions are identical. The analysis of the surplus of RRSF then is to show that with a general regularizer, which will distort the outcome relative to RSF, the surplus lost is minimal. The next lemma shows that the RRSF reduction approximately preserves social surplus; its proof is given in Appendix A.3.

Lemma 4.6. *For convex regularizer ϕ with maximum $\phi_{\max}$ and minimum $\phi_{\min}$ on the probability simplex $D = \{\mathbf{q} \in [0, 1]^m \mid \sum_j q_j = 1\}$, the expected contribution to the surplus of agent κ in the regularized RSF mechanism $\mathcal{M}$ relative to the algorithm $\mathcal{A}$ satisfies $\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - (\phi_{\max} - \phi_{\min})$.*

Up to this point, we have been treating ϕ as an arbitrary strongly convex function. In light of Lemma 4.6 we should choose ϕ to minimize $\phi_{\max} - \phi_{\min}$, subject to γ -strong-convexity. An appropriate choice, which also turns out to be convenient from the standpoint of efficient computation, is given by the quadratic regularizer, defined below. For the quadratic regularizer ϕ , the subsequent proposition summarizes the properties of the RRSF reduction.

Definition 4.8. The quadratic regularizer with parameter γ is the function $\phi(\mathbf{q}) = \frac{1}{2}\gamma\|\mathbf{q}\|^2$.

Proposition 4.1. When the RRSF reduction is applied using the quadratic regularizer with parameter γ , it yields a mechanism $\mathcal{M}$ with the following properties.

1. $\mathcal{M}$ is Bayesian incentive compatible.
2. The expected contribution to the surplus of any agent κ in $\mathcal{M}$ satisfies

$$\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - \frac{\gamma}{2}.$$

3. For any types t and s the corresponding surrogate distributions $\mathbf{q}_t, \mathbf{q}_s$ computed by $\mathcal{M}$ satisfy

$$\|\mathbf{q}_t - \mathbf{q}_s\|_1 \leq \frac{m}{\gamma} d(t, s).$$

4. The outcome and payments of $\mathcal{M}$ can be computed in polynomial time in the ideal model.

Proof. Property 1 is a restatement of Theorem 4.4. Property 2 follows from Lemma 4.6 if we can verify that $\phi_{\max} - \phi_{\min} \leq \frac{\gamma}{2}$ when ϕ is the quadratic regularizer with parameter γ . In fact, as $\mathbf{q}$ varies over the probability simplex, the function $\phi(\mathbf{q}) = \frac{1}{2}\gamma\|\mathbf{q}\|^2$ achieves its maximum value of $\frac{\gamma}{2}$ at the corners of the simplex, and it achieves its minimum value of $\frac{\gamma}{2m}$ at the centroid the simplex (i.e., the point corresponding to the uniform distribution over m alternatives), so

$$\phi_{\max} - \phi_{\min} = \frac{\gamma}{2} \left(1 - \frac{1}{m}\right) < \frac{\gamma}{2},$$

as desired. Property 3 follows from Lemma 4.5 if we can verify that ϕ is γ -strongly convex (Definition 4.7), a property that follows immediately from the definition of the quadratic regularizer (Definition 4.8).

Finally, to prove Property 4, let us discuss the implementation of the RRSF surrogate selection rule by computationally efficient algorithms. Referring to the description of the RRSF reduction in Definition 4.6, the payments in Step 5 can trivially be computed in linear time, so the only computationally non-trivial steps are Step 2, which requires solving the concave program (4.2), and Step 3, which computes the Lagrange multipliers by solving the dual linear program (4.3)–(4.5). It is known from the work of Kozlov et al. (1979) that an exact solution of a concave quadratic program, such as (4.2), can be computed in polynomial time using the ellipsoid method. Given the vectors $\mathbf{q}_i$ from the solution of the concave program, we can then solve the linear program (4.3)–(4.5) in polynomial time using the ellipsoid method (Khachiyan, 1979) or interior point methods (Karmarkar, 1984). $\square$

4.3.3 Regularized ERSF Mechanism

We combine the regularized RSF (Definition 4.6) with empirical value estimates (Definition 4.1) and analyze an agent's incentive to misreport.

Definition 4.9. The Regularized Estimated Replica-Surrogate Fractional Assignment (RERSF) surrogate selection rule, parameterized by L , uses the empirical estimation procedure to estimate the value of each replica for the algorithm's outcome for each surrogate from L samples from the algorithm. The surrogate selection is given by the Regularized RSF selection rule according to the estimated values.

Our definition of distinguishability of type pairs depends on the pairs of outcomes that can be used to distinguish the types. The value estimates in RERSF are given by L draws from each of m surrogate outcomes. Contrast the following quantifications of distinguishability for type pair (t, s) :

- The *estimated surrogate outcome distinguishability*, denoted $\tilde{d}(t, s)$, is distinguishability with respect to all pairs of estimated surrogate outcomes $\{\tilde{X}_1, \dots, \tilde{X}_m\}$.
- The *sampled outcome distinguishability*, denoted $d(t, s)$, is the distinguishability with respect to all pairs of the mL outcomes sampled in the calculation of the surrogate outcome estimates.

Conditioning on the value of either of these notions of distinguishability leaves the sample outcomes, which are unconditionally independent, conditionally correlated. The following notion of distinguishability lies between these two notions in magnitude, but conditioning the sampled outcomes with respect to it preserves their independence and, thus, simplifies analysis. When there are L rounds of sampling, define a pair of sampled outcomes to be *aligned* if they were sampled in the same round, and take the maximum distinguishability over all aligned pairs.

- The *maximum aligned distinguishability*, denoted $d_\star(t, s)$, is the distinguishability with respect to all aligned pairs of outcomes sampled in the L rounds, that is, the maximum over rounds of the distinguishability with respect to pairs of outcomes that were sampled in that round.

It is easy to observe that the estimated surrogate outcome distinguishability is at most the maximum aligned distinguishability which is at most the sampled outcome distinguishability.

The following lemma shows that the incentive to misreport from t to s in RERSF can be made an arbitrarily small fraction of the distinguishability $d(t, s)$. We adopt the notation of Lemma 4.5. For the true distribution of outcomes of the algorithm, $v_{tj} = v(r_t, X_j)$ and $\mathbf{v}_t = (v_{t1}, \dots, v_{tm})$; for the estimated outcome distributions, $\tilde{v}_{tj} = v(r_t, \tilde{X}_j)$ and $\tilde{\mathbf{v}}_t = (\tilde{v}_{t1}, \dots, \tilde{v}_{tm})$; the probability that t is assigned to surrogate j by RERSF is $\tilde{q}_{tj}$ and the vector of these probabilities is $\tilde{\mathbf{q}}_t$; the payment of t in RERSF is $\tilde{p}_t$.

Lemma 4.7. *Fix two types $t, s \in \mathcal{T}$, a real number $\zeta \in (0, 1/2)$, and a γ -strongly-convex regularizer. If $L > \frac{1}{2}\zeta^{-2} \ln(2m/\zeta)$ in RERSF then type t prefers the outcome (including payments) of type s to its own outcome by at most $\frac{2\zeta m}{\gamma} \mathbf{E}[d(t, s)]$ where $d(\cdot, \cdot)$ is the sampled outcome distinguishability.*

Proof. For fixed draws of L outcomes for each surrogate, write the benefit of t deviating to s in RERSF as

$$\text{Gain} = \mathbf{v}_t \cdot \tilde{\mathbf{q}}_s - \tilde{p}_s - \mathbf{v}_t \cdot \tilde{\mathbf{q}}_t + \tilde{p}_t. \quad (4.7)$$

The incentive compatibility of the RERSF reduction implies that, for type pair (t, s) , type t does not want to report s , i.e., $\tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_t - \tilde{p}_t \geq \tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_s - \tilde{p}_s$, or rearranging,

$$\tilde{p}_t - \tilde{p}_s \leq \tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_t - \tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_s.$$

Substituting this upper bound on the difference in payments into equation (4.7) we bound the deviation utility by

$$\begin{aligned} \text{Gain} &\leq \mathbf{v}_t \cdot \tilde{\mathbf{q}}_s - \mathbf{v}_t \cdot \tilde{\mathbf{q}}_t + \tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_t - \tilde{\mathbf{v}}_t \cdot \tilde{\mathbf{q}}_s \\ &= (\mathbf{v}_t - \tilde{\mathbf{v}}_t) \cdot (\tilde{\mathbf{q}}_s - \tilde{\mathbf{q}}_t) \\ &\leq \|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty \|\tilde{\mathbf{q}}_s - \tilde{\mathbf{q}}_t\|_1. \end{aligned}$$

Lemma 4.5 shows that $\|\tilde{\mathbf{q}}_s - \tilde{\mathbf{q}}_t\|_1 \leq \frac{m}{\gamma} \tilde{d}(t, s)$ where $\tilde{d}(t, s)$ is the estimated surrogate outcome distinguishability, and the maximum aligned distinguishability $d_\star(t, s)$ upper bounds the estimated surrogate outcome distinguishability $\tilde{d}(t, s)$. Thus,

$$\text{Gain} \leq \frac{m}{\gamma} \|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty \tilde{d}(t, s) \leq \frac{m}{\gamma} \|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty d_\star(t, s). \quad (4.8)$$

Now take the expectation of equation (4.8) over the L sampled outcomes for each surrogate and apply Lemma 4.9 (given below) which shows that $\mathbf{E}[\|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty d_\star(t, s)] \leq 2\zeta \mathbf{E}[d_\star(t, s)]$.

$$\begin{aligned} \mathbf{E}[\text{Gain}] &\leq \frac{2\zeta m}{\gamma} \mathbf{E}[d_\star(t, s)] \\ &\leq \frac{2\zeta m}{\gamma} \mathbf{E}[d(t, s)]. \end{aligned}$$

The final inequality follows as the sample outcome distinguishability $d(t, s)$ is an upper bound on the maximum aligned distinguishability $d_\star(t, s)$. $\square$

To prove that $\mathbf{E}[\|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty d_\star(t, s)] \leq 2\zeta \mathbf{E}[d_\star(t, s)]$, the following lemmas show that the product of the average and the maximum of i.i.d. random variables can be bounded by product of the standard tail bound on the expected average and the expectation of the maximum. Intuitively, the maximum is given by one of the variables, while any single variable has an insignificant influence on the average in the tail bound. Hence, these random variables are nearly independent and their expected values essentially factor into the product of their expectations.

Lemma 4.8. *For $\zeta \in (0, 1)$ and $L \geq \frac{1}{2}\zeta^{-2} \ln(2m/\zeta)$ i.i.d. samples of vectors with support $[0, 1]^m$, with $\mathbf{v}$ denoting the coordinate-wise expected value vector and $\tilde{\mathbf{v}}$ denoting the coordinate-wise sample average vector,*

$$\Pr[\|\mathbf{v} - \tilde{\mathbf{v}}\|_\infty > \zeta] \leq \zeta.$$

Proof. The Chernoff-Hoffding bound (Theorem 4.1) and assumption on L implies that, for any j ,

$$\Pr[|v_j - \tilde{v}_j| > \zeta] \leq 2 \exp(-2L\zeta^2) \leq \zeta/m.$$

The union bound and $\max_j |v_j - \tilde{v}_j| = \|\mathbf{v} - \tilde{\mathbf{v}}\|_\infty$ imply that

$$\Pr[\|\mathbf{v} - \tilde{\mathbf{v}}\|_\infty > \zeta] \leq \zeta. \quad \square$$

Lemma 4.9. *If $\zeta \in (0, 1/2)$ and $L \geq \frac{1}{2}\zeta^{-2} \ln(2m/\zeta)$, then $\mathbf{E}[\|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty d_\star(t, s)] \leq 2\zeta \mathbf{E}[d_\star(t, s)]$.*

Proof. Let $A = \|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty$ and $B = d_\star(t, s) = \max_\ell d_\ell(t, s)$ where $d_\ell(t, s)$ denotes the distinguishability of (t, s) on the ℓ^{th} round draws of surrogate outcomes. The objective of this lemma is to bound $\mathbf{E}[AB]$; we analyze $\mathbf{E}[AB | B = b]$ and then take expectation over B .

One way to sample A from the conditional distribution with $B = b$ is to draw the first round surrogate outcomes conditioned on distinguishability $d_1(t, s) = b$ and the ℓ^{th} round surrogate outcome, for $\ell \in \{2, \dots, L\}$, i.i.d. and conditioned on $d_\ell(t, s) \leq b$. Consider also the random variable A' that differs from A in that the first round outcomes are drawn identically to the later rounds, i.e., conditioned on $d_1(t, s) \leq b$ not $d_1(t, s) = b$; denote the resulting value estimates by $\tilde{\mathbf{v}}'_t$. The following two observations drive the proof of the lemma.

- Minimal individual contribution: Changing one round of estimates of $\tilde{\mathbf{v}}_t$ to $\tilde{\mathbf{v}}'_t$ in $\|\mathbf{v}_t - \tilde{\mathbf{v}}_t\|_\infty$ affects its value by at most an additive $1/L$, i.e., $A \leq A' + 1/L$.

- Tail bound: $A' = \|\mathbf{v}_t - \tilde{\mathbf{v}}'_t\|_\infty$ has outcomes in each round drawn i.i.d. and satisfies the assumptions of Lemma 4.8, i.e., $\Pr[A' > \zeta \mid B = b] < \zeta$.

Combine these two observations to obtain:

$$\Pr[A > \frac{1}{L} + \zeta \mid B = b] < \zeta. \quad (4.9)$$

We can bound the conditional expectation of A given $B = b$ as follows.

$$\mathbf{E}[A \mid B = b] \leq [\frac{1}{L} + \zeta] \cdot \Pr[A \leq \frac{1}{L} + \zeta \mid B = b] + 1 \cdot \Pr[A > \frac{1}{L} + \zeta \mid B = b].$$

Below, the first inequality follows from equation (4.9) and the second inequality follows from $L \geq 1/\zeta^2$.

$$\mathbf{E}[A \mid B = b] \leq [\frac{1}{L} + \zeta] \cdot [1 - \zeta] + 1 \cdot \zeta \leq 2\zeta.$$

To complete the proof, write $\mathbf{E}[AB \mid B = b] \leq 2\zeta b$ and integrate the right-hand side over the density of B . We have $\mathbf{E}[AB] \leq 2\zeta \mathbf{E}[B]$. $\square$

4.3.4 A BIC Mechanism

In this section we combine the strictly IC mechanism (Definition 4.5 of Section 4.3.1) with the regularized estimated RSF mechanism (Definition 4.9 of Section 4.3.3) to obtain a BIC mechanism from any algorithm. The time it takes to execute this mechanism is polynomial in the number of distinct types of an agent m , the number of agents n , and the inverse of the additive loss ε that is permitted with respect to the algorithm.

Definition 4.10. *The blackbox RSF mechanism is parameterized by sampling parameter L , probability δ , and regularizer parameter γ , and is implemented as follows:*

1. *For each agent, run the RERSF surrogate selection rule with L samples and the γ -quadratic regularizer.*
2. *With probability $1 - \delta$, output the outcome of the algorithm on surrogates selected for the type profile with payments from RERSF.*
3. *Otherwise (with probability δ), pick an agent κ uniformly at random and run her strictly IC mechanism on outcomes sampled by the simulation.*

Theorem 4.5. *For n agents, m distinct types per agent, any algorithm $\mathcal{A}$, and parameters L , γ , and δ set according to equation (4.12) below, the blackbox RSF mechanism $\mathcal{M}$ is Bayesian incentive compatible and its surplus satisfies $\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - \varepsilon$.*

Proof. We first argue Bayesian incentive compatibility. The strictly IC mechanism for agent κ (which is executed with probability δ/n) compensates the non-incentive-compatibility of RERSF (which is executed with probability at most 1) if:

$$\frac{2\zeta m}{\gamma} \leq \frac{\delta}{2Mn},$$

where γ is the convexity parameter of the regularizer, $M \leq m^2/2$ is the number of distinguishing outcome pairs, and ζ is the accuracy corresponding to estimation from L sample outcomes for each surrogate. This inequality follows because the expected loss when deviating from t to s in the strictly IC mechanism is at least $\frac{1}{2M} \mathbf{E}[d(t, s)]$ (Lemma 4.4) and the expected gain when deviating from t to s in the RERSF mechanism is at most $\frac{2\zeta m}{\gamma} \mathbf{E}[d(t, s)]$ (Lemma 4.7). For the blackbox RSF mechanism to be BIC, it is sufficient to choose

$$\zeta = \frac{\delta\gamma}{2m^3n}, \text{ and} \quad (4.10)$$

$$L = \left\lceil \frac{1}{2\zeta^2} \ln\left(\frac{2m}{\zeta}\right) \right\rceil = \left\lceil \frac{2m^6n^2}{\delta^2\gamma^2} \ln\left(\frac{4m^4n}{\gamma\delta}\right) \right\rceil, \quad (4.11)$$

with L following from ζ so as to satisfy the assumption of Lemma 4.7.

We now analyze the surplus lost in $\mathcal{M}$ relative to $\mathcal{A}$ for agent κ . The surplus lost when running the strictly IC mechanism is trivially bounded above by 1 since valuations are in $[0, 1]$; we run the strictly IC mechanism with probability δ , thus contributing δ to the expected surplus lost in $\mathcal{M}$. With the remaining probability we run the RERSF reduction with γ -quadratic regularizer; according to Proposition 4.1 this would entail $\gamma/2$ surplus loss if the estimated valuations were equal to the true ones. To ensure that both of these contributions to the surplus loss are sufficiently small, we will set

$$\gamma = \varepsilon, \quad \delta = \varepsilon/4, \quad L = \left\lceil \frac{32m^6n^2}{\varepsilon^4} \ln\left(\frac{16m^4n}{\varepsilon^2}\right) \right\rceil; \quad (4.12)$$

with L following from γ and δ according to (4.11). If the estimated valuations were equal to the true ones, then the surplus loss would be bounded above by $\delta + \frac{\gamma}{2} = \frac{3\varepsilon}{4}$. However, the estimated valuations differ from the true ones by the sampling error, and it remains to show that this sampling error contributes at most $\frac{\varepsilon}{4}$, in expectation, to the surplus loss.

Recall, from Lemma 4.8 above, that the probability that any of the m estimates of the value of a type t for a surrogate outcome differs from the true value by more than ζ is at most ζ . Taking the union bound over the m types, the probability that any of the m^2 estimates of the values of replicas for surrogate outcomes differs from the corresponding true value by more than ζ is at most $m\zeta$. When all of the estimates have error at most ζ , the surplus of RERSF has loss at most 2ζ relative to the optimal surplus. Otherwise, the surplus loss is at most 1. Thus, the expected loss is at most $2\zeta + m\zeta \leq 2m\zeta$ (as $m \geq 2$). Applying equation (4.10) for ζ , the expected loss from sampling errors is at most $\frac{\varepsilon^2}{4m^2n} \leq \frac{\varepsilon}{4}$. $\square$

4.4 Multi-dimensional Continuous Types

When agents have multi-dimensional continuous types and we only have black-box access to the algorithm $\mathcal{A}$, we do not know of any way to use an RSM-style reduction to transform the algorithm into a perfectly BIC mechanism. As in the single-dimensional case, there are examples to illustrate that one cannot simply estimate the value of replica r for the surrogate outcome $\mathcal{A}(s)$ by sampling. Unlike in the single-dimensional case, this difficulty cannot be overcome by using a binomial estimator. Binomial estimators fail even for two-dimensional type spaces; see Lemma A.1 in Appendix A.2. On the other hand, it is easy to see that even the ERSF reduction results in an ε -BIC mechanism.

Theorem 4.6. *In the black-box model, when agents have multi-dimensional continuous types and replica values are estimated from $L \geq \frac{1}{2} \left(\frac{3}{\varepsilon}\right)^2 \ln\left(\frac{6m}{\varepsilon}\right)$ samples of each surrogate outcome, ERSF is ε -BIC.*

Proof. Let $\tilde{X}_j$ be the average of L samples from the outcome distribution $X_j = \mathcal{A}(s_j)$ in the definition of ERSM. The Chernoff-Hoeffding inequality (Theorem 4.1) implies that

$$\Pr[|v(t, X_j) - v(t, \tilde{X}_j)| > \frac{\varepsilon}{3}] < 2 \exp(-2L(\frac{\varepsilon}{3})^2) = \frac{\varepsilon}{3m}.$$

The union bound then implies that type t 's value estimates for all surrogate outcomes are correct to an additive $\varepsilon/3$ error with probability at least $1 - \varepsilon/3$.

By the taxation principle, the estimated values of the other replicas for the surrogate outcomes induce a payment for type t to be matched to each surrogate outcome. RSM matches t to the surrogate outcome (with payment) that, according to t 's value estimates, give it the highest utility.

When type t 's estimates are all correct to an additive $\varepsilon/3$, the maximum loss in utility to t by misestimation is bounded by $2\varepsilon/3$. Otherwise, and with probability at most $\varepsilon/3$, type t 's maximum loss is at worst 1. Combining these bounds we get an expected loss of at most ε . $\square$

5 Transportation Cost

A useful intuition for the effectiveness of the RSM reduction is the following: the uniform distribution over the surrogates of agent κ is a discrete approximation to agent κ 's type distribution, and the additive loss suffered by the reduction is completely determined by the error in this approximation. Proposition 3.1 makes this intuition precise by pinpointing the expected transportation cost (Definition 3.4) between two random samples as the proper measure of approximation error in this context. Thus, analyzing the transportation cost between random subsets of a type space is a question of independent economic interest: it gives a precise way of framing the question, “*How well does a large collection of random samples from a continuous type distribution approximate the distribution itself?*”

Recall that the transportation cost for two m -tuples of points in a metric space $\mathcal{T}$ is $\frac{1}{m}$ times the minimum-cost matching between the two tuples, where the cost of a matching is the sum of the distances between matched pairs with respect to the metric (Definition 3.4). In this section we give general methods for bounding the expected transportation cost when the points are drawn i.i.d. from any distribution. The special case of this problem in which $\mathcal{T} = [0, 1]^\Delta$, the metric is Euclidean, and the distribution over $\mathcal{T}$ is uniform, has a rich history in mathematics and computer science (Ajtai et al., 1984; Karp et al., 1984; Talagrand, 1992, 1994). For the Euclidean space the expected transportation cost is $\Theta(m^{-1/\Delta})$ when $\Delta > 2$ and $\Theta(\sqrt{\log(m)/m})$ when $\Delta = 2$; the two-dimensional case requires much deeper techniques than the higher-dimensional cases. In contrast, our analysis is for general metric spaces with arbitrary distributions. Our treatment of this topic resembles the analysis of random matching problems on hierarchically separated trees by Abrahamson et al. (2008). In particular, our analysis hinges on understanding the expected transportation cost between two random subsets of a finite tree, a topic that we address in Section 5.2.

5.1 Single-dimensional Settings

Before tackling the case of general metric spaces, we present a simple bound on the expected transportation cost for general distributions over the single-dimensional metric space $\mathcal{T} = [0, 1]$ with $d(t_1, t_2) = |t_1 - t_2|$.

Theorem 5.1. *For two m -tuples $\mathbf{r}$ and $\mathbf{s}$ of i.i.d. samples from any distribution on $[0, 1]$,*

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] \leq \sqrt{\frac{1}{2m}}.$$

Proof. Sort the elements of $\mathbf{r}$ and $\mathbf{s}$ in ascending order as $r_1 \leq \dots \leq r_m$ and $s_1, \dots, s_m$. The minimum-cost matching of $\mathbf{r}$ to $\mathbf{s}$ is the one that matches r_i to s_i for each i . The cost of this matching is equal to the sum of the lengths of the intervals I_i defined as $[r_i, s_i]$ or $[s_i, r_i]$, depending whether $r_i \leq s_i$ or $s_i < r_i$. Figure 1 illustrates this set of intervals. To show that the expected sum of their lengths is bounded by $\sqrt{m/2}$, we will show that the average height of the “stack of intervals” sitting above y , for every $y \in [0, 1]$, is no greater than $\sqrt{m/2}$; the integral over y then sums the lengths of all intervals. Intuitively, the number of intervals containing y is equal to the (absolute value of the) difference between the number of $\mathbf{r}$ -samples in $[0, y]$ and the number of $\mathbf{s}$ -samples; this difference has mean zero and standard deviation $O(\sqrt{m})$ because the $\mathbf{r}$ -samples and $\mathbf{s}$ -samples are i.i.d., and there are m of each.

Figure 1: Transportation cost of two subsets of an interval.

To proceed with the formal proof, let $\chi_i(y)$ denote the indicator function of the interval I_i containing y . We have

$$m \cdot d(\mathbf{r}, \mathbf{s}) = \sum_{i=1}^m |r_i - s_i| = \sum_{i=1}^m \int_0^1 \chi_i(y) dy = \int_0^1 \left[\sum_{i=1}^m \chi_i(y) \right] dy.$$

Letting $N_{\mathbf{r}}(y), N_{\mathbf{s}}(y)$ denote the number of elements of $\mathbf{r}, \mathbf{s}$, respectively, that are greater than or equal to y , we find that $\sum_{i=1}^m \chi_i(y) = |N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y)|$ and therefore

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] = \frac{1}{m} \mathbf{E} \left[\int_0^1 |N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y)| dy \right] \leq \frac{1}{m} \sqrt{\mathbf{E} \left[\int_0^1 (N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y))^2 dy \right]},$$

so to complete the proof, it suffices to show that for every $y \in [0, 1]$, $\mathbf{E}[(N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y))^2] \leq m/2$. Note that $\mathbf{E}[(N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y))^2]$ is the variance of the random variable $N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y)$, since that random variable has expectation 0. Now, $N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y)$ is a sum of $2m$ independent random variables; the first m of these take values in $\{0, 1\}$ and the remaining m take values in $\{0, -1\}$. Using the fact that a Bernoulli random variable has variance at most $1/4$, and that the variance of a sum of independent random variables is the sum of the variances, we now see that $\text{Var}(N_{\mathbf{r}}(y) - N_{\mathbf{s}}(y)) \leq 2m \cdot (1/4) = m/2$, as desired. $\square$

5.2 Trees

As mentioned earlier, our analysis for general metric spaces hinges on understanding the special case in which the metric space is a finite tree. In this section we show how to relate the expected

transportation cost in trees to two other parameters, the *size* and (weighted) *height* of the tree.

Definition 5.1. A rooted tree is a connected undirected acyclic graph with a designated root. If $\Upsilon = (V, E, \psi)$ is a tree with root ψ and $d_\Upsilon : E \rightarrow \mathbb{R}_+$ is a function that assigns a positive length to every edge of Υ then the tree metric $d_\Upsilon(u, v)$ is defined to be the cumulative length of the edges on the unique path joining u to v in Υ . We define the size and the height of Υ by

$$\text{size}(\Upsilon) = \sum_{e \in E} d_\Upsilon(e), \text{ and } \text{height}(\Upsilon) = \max_{v \in V} d_\Upsilon(\psi, v).$$

The following lemma yields a simple upper bound on the expected transportation cost between two random m -tuples of tree nodes.

Lemma 5.1. For every tree Υ with positive edge lengths, for every distribution over the nodes of Υ , the expected transportation cost between two random m -tuples of i.i.d. samples from the distribution is bounded above by $\sqrt{2} \text{size}(\Upsilon) \text{height}(\Upsilon)/m$.

The proof is, in some sense, a generalization of the proof of Theorem 5.1.

1. For each edge we define its “imbalance” to be the minimum number of matched pairs that must lie on opposite sides of the edge, in any matching of the m -tuples.
2. We prove that the transportation cost between the two m -tuples is equal to the sum, over all tree edges, of the length times the imbalance of the edge.
3. We relate this sum to the size and height of Υ using the Cauchy-Schwartz inequality and a simple upper bound on the variance of a sum of independent Bernoulli random variables.

Before proving Lemma 5.1, we characterize the transportation cost between any two m -tuples of nodes $\mathbf{r}$ and $\mathbf{s}$ in a tree Υ . To do so, we must establish some notation. Deleting any edge e from Υ divides it into exactly two connected components; we will use $\Upsilon(e)$ to denote the component that does not contain the root ψ . Let $N_{\mathbf{r}}(e)$ and $N_{\mathbf{s}}(e)$ denote the number of elements of $\mathbf{r}$ and $\mathbf{s}$, respectively, that belong to $\Upsilon(e)$. Define the *imbalance* at e to be $\delta(e) = |N_{\mathbf{r}}(e) - N_{\mathbf{s}}(e)|$.

Lemma 5.2. If $\Upsilon = (V, E, \psi)$ is a tree with positive edge lengths $\{d_\Upsilon(e)\}_{e \in E}$, and $\mathbf{r}, \mathbf{s}$ are any two m -tuples of nodes of Υ , then

$$d_\Upsilon(\mathbf{r}, \mathbf{s}) = \frac{1}{m} \sum_{e \in E} d_\Upsilon(e) \delta(e). \quad (5.1)$$

*Proof.*¹¹ We show that in any minimum-cost matching, the number of matched pairs that cross any edge e is exactly $\delta(e)$, which implies the lemma. For a contradiction, suppose that there exist matched pairs on opposite sides of e in both orientations, i.e., with an element from $\mathbf{r}$ in $\Upsilon(e)$ matched to an element from $\mathbf{s}$ not in $\Upsilon(e)$, and vice versa. The paths for these matched pairs both cross over e . Therefore, we can reduce the cost of the matching by swapping these two matched pairs and the new matching is cheaper by at least $2 \cdot d_\Upsilon(e)$. This strict improvement contradicts our original choice of a minimum cost matching. $\square$

Proof of Lemma 5.1. Denote the two random m -tuples by $\mathbf{r}, \mathbf{s}$. Lemma 5.2 ensures that

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] = \frac{1}{m} \sum_{e \in E} d_\Upsilon(e) \mathbf{E}[\delta(e)] \leq \frac{1}{m} \sqrt{\sum_{e \in E} d_\Upsilon(e)} \cdot \sqrt{\sum_{e \in E} d_\Upsilon(e) \mathbf{E}[\delta(e)^2]}. \quad (5.2)$$

---

¹¹We are grateful to an anonymous referee for suggesting the following simplification of our original proof.

For edge $e = (u, v)$, we have

$$\delta(e)^2 = [N_{\mathbf{r}}(e) - N_{\mathbf{s}}(e)]^2.$$

The random variable $Y_e = N_{\mathbf{r}}(e) - N_{\mathbf{s}}(e)$ has expectation 0, and it is a sum of $2m$ independent random variables each of which is (up to sign) an indicator random variable of the event $r_i \in \Upsilon(e)$ or $s_i \in \Upsilon(e)$ for some index i . Letting q_e denote the expected value of each such random variable, i.e. the probability of sampling a node in $\Upsilon(e)$ under the distribution on Υ , we find that

$$\mathbf{E}[\delta(e)^2] = \mathbf{E}[Y_e^2] = \text{Var}(Y_e) = 2m \cdot q_e (1 - q_e) \leq 2m \cdot q_e.$$

Summing over edges of Υ ,

$$\sum_{e \in E} d_{\Upsilon}(e) \mathbf{E}[\delta(e)^2] \leq 2m \sum_{e \in E} d_{\Upsilon}(e) q_e.$$

Since q_e is equal to the probability that e lies on the path from the root node ψ to a random node v sampled from the distribution over Υ , we see that the right side is simply $2m$ times the expected length of the path from ψ to v , and is therefore bounded above by $2m \cdot \text{height}(\Upsilon)$. Substituting this bound into (5.2) completes the proof of the lemma. $\square$

5.3 Multi-dimensional Settings

We can now derive upper bounds on the expected transportation cost in arbitrary metric spaces by mapping them into tree metrics and applying the results of the preceding section. Intuitively, this mapping will partition the metric space into pieces of diameter ε and connect these pieces in a hierarchical tree structure so that the distance between two points is bounded above by ε plus the distance in the tree between the vertices to which the two points are mapped.

Lemma 5.3. *Suppose that $\mathcal{T}$ is a metric space, Υ is a tree with positive edge lengths, $\varepsilon \geq 0$ is a real number, and $h : \mathcal{T} \rightarrow \Upsilon$ is a function from $\mathcal{T}$ to the nodes of Υ satisfying*

$$\forall t_1, t_2 \in \mathcal{T} \quad d(t_1, t_2) \leq \varepsilon + d_{\Upsilon}(h(t_1), h(t_2)); \quad (5.3)$$

then for any distribution over $\mathcal{T}$, if $\mathbf{r}, \mathbf{s}$ denote two m -tuples of independent random samples from this distribution,

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] \leq \varepsilon + \sqrt{2 \text{size}(\Upsilon) \text{height}(\Upsilon)/m}.$$

Proof. Letting π denote a minimum-cost perfect matching between the m -tuples $h(\mathbf{r}), h(\mathbf{s})$ in Υ , we have

$$d(\mathbf{r}, \mathbf{s}) \leq \frac{1}{m} \sum_{i=1}^m d(r_i, s_{\pi(i)}) \leq \varepsilon + \frac{1}{m} \sum_{i=1}^m d_{\Upsilon}(h(r_i), h(s_{\pi(i)})) = \varepsilon + d_{\Upsilon}(h(\mathbf{r}), h(\mathbf{s})), \quad (5.4)$$

where the first inequality follows from the definition of transportation cost, the second follows from (5.3), and the third follows from our choice of π . The lemma now follows by taking the expectation of both sides of (5.4) and applying Lemma 5.1. $\square$

As a first illustration of this technique, we rederive a weaker form of Theorem 5.1, the upper bound for the expected transportation cost in single-dimensional settings. Let k be an even number greater than $\sqrt{m}$, and let Υ be a path of length k rooted at its midpoint, with each edge having length $1/k$. Map the interval $[0, 1]$ to the nodes of Υ by partitioning it into subintervals of length $\frac{1}{k+1}$ and mapping the i^{th} subinterval to the i^{th} node of Υ . This mapping satisfies (5.3) with $\varepsilon = 1/k \leq \sqrt{1/m}$. Since $\text{size}(\Upsilon) = 1$ and $\text{height}(\Upsilon) \leq 1/2$, Lemma 5.3 implies that $\mathbf{E}[d(\mathbf{r}, \mathbf{s})] \leq$

$\sqrt{1/m} + \sqrt{1/m}$. This is worse than the upper bound $\sqrt{1/(2m)}$ proven in Theorem 5.1, but only by a factor of $2\sqrt{2}$.

Below we apply Lemma 5.3 to general metric spaces. The first bound we give concerns metric spaces that can be partitioned into a small number of pieces, each piece with diameter at most ε . For example, the Euclidean space $[0, 1]^\Delta$ can be broken into $\varepsilon^{-\Theta(\Delta)}$ such pieces. The second bound we give concerns metric spaces that have doubling dimension Δ (Definition 3.5). For example, the Euclidean space $[0, 1]^\Delta$ has doubling dimension $O(\Delta)$.

Theorem 5.2. *If metric space $\mathcal{T}$ with diameter one can be partitioned into $\eta(\varepsilon)$ pieces, each of diameter at most ε , then*

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] \leq \varepsilon + \sqrt{\frac{\eta(\varepsilon)}{2m}}.$$

Proof. Let Υ be a tree consisting of a root node with $\eta(\varepsilon)$ leaves, each joined to the root by an edge of length $1/2$. Define $h : \mathcal{T} \rightarrow \Upsilon$ to be a mapping that sends each piece of the partition to a distinct leaf of Υ . To prove that this mapping h satisfies (5.3) we consider two cases. If t_1 and t_2 belong to the same piece of the partition then $d(t_1, t_2) \leq \varepsilon$ and the right side of (5.3) equals ε . If t_1 and t_2 belong to different pieces of the partition then $d(t_1, t_2) \leq 1$ and the right side of (5.3) is at least 1.

Noting that $\text{size}(\Upsilon) = \frac{\eta(\varepsilon)}{2}$ and $\text{height}(\Upsilon) = \frac{1}{2}$, we apply Lemma 5.3 to obtain the theorem. $\square$

Theorem 5.3. *If metric space $\mathcal{T}$ has diameter one and doubling dimension $\Delta > 1$, and if $\varepsilon \leq 1$ is a power of two,*

$$\mathbf{E}[d(\mathbf{r}, \mathbf{s})] \leq \varepsilon + \sqrt{\frac{\varepsilon^{1-\Delta}}{m \min\{1, \Delta - 1\}}}.$$

Proof. Construct the complete 2^Δ -ary tree Υ and label its nodes with subsets of $\mathcal{T}$ corresponding to a hierarchical partitioning of $\mathcal{T}$ into sets with progressively smaller diameter until level $k = \log_2(1/\varepsilon)$ where the parts have diameter at most ε . In this fashion the root node is labeled with $\mathcal{T}$ itself. The remaining tree nodes and their labels are recursively defined so that an internal node with label S has 2^Δ children whose labels constitute a partition of S into subsets of diameter at most half of the diameter of S . Each edge e of Υ between level i and level $i + 1$ has length $d_\Upsilon(e) = 2^{-i-2}$, which is $1/4$ of the upper bound on the diameter of the sets that label tree nodes at level i .

For every $t \in \mathcal{T}$, there is a unique leaf of Υ whose label is a set containing t , and we define $h(t)$ to be this leaf. The following analysis shows that $h(\cdot)$ and $d_\Upsilon(\cdot)$ satisfy (5.3). Suppose the deepest common ancestor of $h(t_1)$ and $h(t_2)$ is on level i . Then $d(t_1, t_2)$ is bounded above by 2^{-i} , the maximum possible diameter of the set labeling this ancestor. We have (recall $\varepsilon = 2^{-k}$):

$$d_\Upsilon(h(t_1), h(t_2)) = 2 \sum_{j=i}^{k-1} 2^{-j-2} = 2(2^{-i-1} - 2^{-k-1}) = 2^{-i} - 2^{-k} = 2^{-i} - \varepsilon.$$

Therefore, (5.3) holds.

We now turn to the task of estimating the height and size of Υ . Every node is joined to the root by a path whose edge lengths are bounded above by the terms of the geometric series $\frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \dots$, and consequently $\text{height}(\Upsilon) < 1/2$. To calculate the size of Υ , recall that a complete 2^Δ -ary tree has $2^{\Delta i}$ nodes at level i . Each of these nodes is connected to the previous level by an edge with length 2^{-i-1} . Summing over all levels we have:

$$\text{size}(\Upsilon) \leq \sum_{i=1}^k 2^{\Delta i} 2^{-i-1} < 2^{(\Delta-1)k-1} \sum_{j=0}^{\infty} 2^{-(\Delta-1)j} = \frac{\varepsilon^{1-\Delta}}{2} \cdot \left(1 - 2^{-(\Delta-1)}\right)^{-1} \leq \frac{\varepsilon^{1-\Delta}}{\min\{1, \Delta - 1\}}.$$

where the last inequality follows from the fact that $(1 - 2^{-z})^{-1} \leq 2/\min\{1, z\}$ for any $z \geq 0$. Lemma 5.3, then, implies the theorem. $\square$

In Section 3, Theorems 3.2, 3.3, and 3.4 are immediate consequences of Proposition 3.1 and Theorems 5.1, 5.2, and 5.3.

6 Conclusions

We have shown in this paper that any non-optimal algorithm for welfare maximization can be converted into a Bayesian incentive compatible mechanism by solving a collection of single-agent problems. These single-agent problems consider the distribution over outcomes of the algorithm (with truthful reporting by all agents) and the distribution over the agent’s types, and they match types to outcomes via the surplus-optimal matching. Importantly, while the original mechanism design problem might correspond to linear optimization subject to an arbitrary combinatorial constraint — a computationally difficult problem in general — the problem we must solve to incentivize truthful reporting is simply a maximum weighted matching problem which is computationally easy. These reoptimizations can be performed independently for each agent and, therefore, the complexity of the reduction scales well as the number of agents grows.

As is evident by both our transportation cost analysis (for continuous type spaces) and the replica-surrogate flow construction (for discrete type spaces), the complexity of the single-agent problems grow with the complexity of the type space. Essentially, these approaches are brute-force in the size of the type space in that every type must be inspected. For agents with multi-dimensional preferences, this size should not necessarily be assumed to be small. For instance, if there are m items for sale and the agent has values for each item drawn independently from distributions with support size at least two, then the number of types of the agent is exponential in m . For such an agent even a modest dimension m renders intractable the direct approach to solving the matching or flow problems. A main direction for future work, on understanding how to align incentives of agents in non-optimal mechanisms, is in the development of methods that are not brute-force in the size of the agent’s type space. The goal of such methods would be to scale well as the dimension of the agent’s type space increases. This direction is especially important when the agents’ type spaces are large but their natural descriptions are succinct. Unit-demand or additive preferences drawn from a product distribution provide a good starting place for this exploration, see e.g., Armstrong (1996, 1999), Chawla et al. (2007, 2010a,b), Cai and Daskalakis (2011), Hart and Nisan (2013), Haghpanah and Hartline (2014), and Daskalakis et al. (2014).

References

- Abrahamson, J., Csaba, B., and Shokoufandeh, A. (2008). Optimal random matchings on trees and applications. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, volume 5171 of *Lecture Notes in Computer Science*, pages 254–265.
- Ahuja, R., Magnanti, T., and Orlin, J. (1993). *Network flows: Theory, algorithms, and applications*. Prentice Hall.
- Ajtai, M., Komlós, J., and Tusnády, G. (1984). On optimal matchings. *Combinatorica*, 4(4):259–264.
- Armstrong, M. (1996). Multiproduct Nonlinear Pricing. *Econometrica*, 64(1):51–75.

- Armstrong, M. (1999). Price discrimination by a many-product firm. *Review of Economic Studies*, 66(1):151–168.
- Assouad, P. (1983). Plongements Lipschitziens dans $\mathbb{R}^n$. *Bull. Soc. Math. France*, 111:429–448.
- Bei, X. and Huang, Z. (2011). Bayesian incentive compatibility via fractional assignments. In *Proc. 22nd ACM Symp. on Discrete Algorithms*, pages 720–733.
- Briest, P., Krysta, P., and Vöcking, B. (2005). Approximation techniques for utilitarian mechanism design. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 39–48.
- Cai, Y. and Daskalakis, C. (2011). Extreme-value theorems for optimal multidimensional pricing. In *Proceedings of the 52nd Annual Symposium on Foundations of Computer Science*, pages 522–531.
- Chawla, S., Hartline, J., and Kleinberg, R. (2007). Algorithmic pricing via virtual valuations. In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 243–251.
- Chawla, S., Hartline, J., Malec, D., and Sivan, B. (2010a). Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 311–320.
- Chawla, S., Malec, D., and Sivan, B. (2010b). The power of randomness in Bayesian optimal mechanism design. In *Proc. 11th ACM Conf. on Electronic Commerce*, pages 149–158.
- Clarke, E. H. (1971). Multipart pricing of public goods. *Public Choice*, 11:17–33.
- Daskalakis, C., Deckelbaum, A., and Tzamos, C. (2014). Strong duality for a multiple-good monopolist. *arXiv preprint arXiv:1409.4150*.
- Dughmi, S. and Roughgarden, T. (2014). Black-box randomized reductions in algorithmic mechanism design. *SIAM Journal on Computing*, 43(1):312–336.
- Groves, T. (1973). Incentives in teams. *Econometrica*, 41:617–631.
- Gul, F. and Stacchetti, E. (1999). Walrasian equilibrium with gross substitutes. *Journal of Economic Theory*, 87(1):95–124.
- Haghpanah, N. and Hartline, J. (2014). Reverse mechanism design. *arXiv preprint arXiv:1404.1341*.
- Hart, S. and Nisan, N. (2013). The menu-size complexity of auctions. In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce*, pages 565–566.
- Hartline, J. and Lucier, B. (2010). Bayesian algorithmic mechanism design. In *Proc. 41th ACM Symp. on Theory of Computing*, pages 301–310.
- Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30.
- Kantorovich, L. (1942). On the translocation of masses. *C. R. (Doklady) Acad. Sci. URSS (N.S.)*, 37:199–201.
- Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395.

- Karp, R., Luby, M., and Marchetti-Spaccamela, A. (1984). A probabilistic analysis of multidimensional bin packing problems. In *Proc. 16th ACM Symp. on Theory of Computing*, pages 289–298.
- Khachiyan, L. G. (1979). A polynomial algorithm in linear programming. *Soviet Math. Doklady*, 20:191–194.
- Kozlov, M., Tarasov, S. P., and Khachiyan, L. G. (1979). Polynomial solvability of convex quadratic programming. *Soviet Math. Doklady*, 20:1108–1111.
- Lavi, R. and Swamy, C. (2011). Truthful and near-optimal mechanism design via linear programming. *Journal of the ACM*, 58(6):25.
- Lehmann, D., O’Callaghan, L., and Shoham, Y. (2002). Truth revelation in rapid, approximately efficient combinatorial auctions. *Journal of the ACM*, 49:577–602.
- Monge, G. (1781). Mémoire sur la théorie des déblais et des remblais. In *Histoire de l’Académie Royale des Sciences de Paris, avec les Mémoires de Mathématique et de Physique pour la même année*, pages 666–704.
- MuAlem, A. (2009). On multi-dimensional envy-free mechanisms. In *Algorithmic Decision Theory*, pages 120–131. Springer.
- Myerson, R. (1981). Optimal auction design. *Mathematics of Operations Research*, 6:58–73.
- Nisan, N. and Ronen, A. (2007). Computationally feasible vcg mechanisms. *J. Artificial Intelligence Research (JAIR)*, 29:19–47.
- Rochet, J. (1987). A necessary and sufficient condition for rationalizability in a quasi-linear context. *Journal of Mathematical Economics*, 16(2):191–200.
- Spielman, D. and Teng, S.-H. (2004). Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. ACM*, 51(3):385–463.
- Talagrand, M. (1992). Matching random samples in many dimensions. *Annals of Applied Probability*, 2(4):846–856.
- Talagrand, M. (1994). Matching theorems and empirical discrepancy computations using majorizing measures. *J. Amer. Math. Soc.*, 7:455–537.
- Valiant, L. (1979). The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201.
- Vickrey, W. (1961). Counterspeculation, auctions, and competitive sealed tenders. *J. of Finance*, 16:8–37.
- Villani, C. (2003). *Topics in optimal transportation*, volume 58 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI.

A Omitted Proofs

A.1 Proof of Theorem 4.2

Recall that a perfect matching in a bipartite graph is an edge set such that each vertex in the graph is incident to exactly one edge in the set. Valiant (1979) proved that the problem of counting the number of perfect matchings in a bipartite graph is #P-hard. The following theorem, restated from Section 4.1, reduces the problem of counting the number of perfect matchings in a bipartite graph to the problem of evaluating $v(t^\kappa, \mathcal{A}(t^\kappa, \mathbf{t}^{-\kappa}))$ over a random type profile $\mathbf{t}^{-\kappa}$. Thus, the existence of a tractable algorithm for the latter problem would imply the existence of a tractable algorithm for the former problem. Contrapositively, assuming that #P-hard problems are intractable (as is widely believed), this confirms that evaluating $v(t^\kappa, \mathcal{A}(t^\kappa, \mathbf{t}^{-\kappa}))$ over a random type profile $\mathbf{t}^{-\kappa}$ is intractable.

Theorem 4.2. *The problem of computing the expected valuation of an agent with a given type in an algorithm, when the types of other agents are drawn from a product distribution, is #P-hard.*

Proof. We reduce the problem of counting perfect matchings in a bipartite graph to that of calculating the expected valuation of an agent with a given type and types of other agents drawn randomly from a distribution. To do so, we start with a bipartite graph $G = (A, B, E)$ with parts $A = \{a_1, \dots, a_n\}$ and $B = \{b_1, \dots, b_n\}$ and an edge set $E = A \times B$. We construct based on this graph a mechanism design problem for $n + 1$ agents and n items. There are n agents where agent $\kappa \in \{1, \dots, n\}$ is associated with vertex $a_\kappa \in A$ (the additional agent $\kappa = 0$ will be defined subsequently). There are n items $j \in \{1, \dots, n\}$ where item j is associated with vertex $b_j \in B$. An agent κ may desire item j if and only if there is an edge between the corresponding vertices, i.e., $(a_\kappa, b_j) \in E$. The type space of each of these agents is $\mathcal{T} = n \times [0, 1]$ where $t = (j, v)$ has value v if x assigns the agent item j , zero otherwise. The distribution over types F^κ for agent κ selects an item j^κ uniformly from the set of adjacent items in the bipartite graph, i.e., $\{j : (a_\kappa, b_j) \in E\}$ and a value v^κ uniformly from $[0, 1]$. Denoting the degree in G of vertex a_κ by d_κ , this uniform distribution assigns probability $1/d_\kappa$ to each item j adjacent to κ .

There is one more agent 0 with type space $\mathcal{T}^0 = [0, 1]$. This agent is single-dimensional and has type t^0 that assigns value t^0 to any outcome in which she obtains any item. Consider the algorithm $\mathcal{A}$ that maximizes the social welfare, i.e., for type profile $\mathbf{t}$, it selects the matching of items to agents with maximum total valuation. We now show that, from the valuation of agent 0 with type t^0 for the distribution over outcomes $\mathcal{A}^0(t^0)$, we can compute the number of perfect matchings in the bipartite graph G .

Agent 0 receives an item and thus obtains value t^0 unless all other agents κ have value $v^\kappa \geq t^0$ and the desired items of the other agents form a perfect matching, i.e., in the bipartite graph $G' = (A, B, E')$ where $E' = \{(\kappa, j^\kappa) : \kappa \in [n]\}$. Therefore,

$$\mathbf{E}_{\mathbf{t}}[v(t^0, \mathcal{A}(\mathbf{t})) \mid t^0] = t^0 \cdot [1 - (1 - t^0)^n \cdot \mathbf{Pr}[E' \text{ is a perfect matching}]].$$

Observe that the distribution over edge sets E' induced by the agents' type distribution is uniform and the probability of any of these edge sets is $\prod_{\kappa=1}^n \frac{1}{d_\kappa}$. The probability that E' is a perfect matching is equal to the number of perfect matchings divided by the number of sets in the support of the distribution. This latter number is $\prod_{\kappa=1}^n d_\kappa$. Thus, the number of perfect matchings is equal to

$$[\prod_{\kappa=1}^n d_\kappa] \cdot [1 - \frac{1}{t^0} \mathbf{E}_{\mathbf{t}}[v(t^0, \mathcal{A}(\mathbf{t})) \mid t^0]] \cdot [1 - t^0]^{-n}$$

which can be easily calculated from agent 0's expected valuation as a function of her type. $\square$

Figure 2: In (a) the replica-to-surrogate-outcome matching problem is depicted. The realization in (b) happens with probability $0.49 \cdot 0.25 \cdot 0.25$, and the straight-across matching, where r_i is matched to x_i for $i \in \{1, 2\}$, has the maximum weight.

A.2 Inadequacy of Binomial Estimation in Multi-Dimensional Settings

This section is devoted to proving the following lemma.

Lemma A.1. *The binomial estimated replica-surrogate matching reduction does not generally result in a BIC mechanism with agents with multi-dimensional preferences.*

Proof. We prove the lemma by exhibiting a counterexample. Consider the uniform distribution F over the two-type type space $\mathcal{T} = \{t_1, t_2\}$ and an algorithm $\mathcal{A}$ that produces outcome from $\mathcal{X} = \{x_1, x_2\}$ as $\mathcal{A}(t_i) = x_i$. The agent's valuation function $v(t, x)$ is defined in the following table:

	x_1	x_2
t_1	0.51	0.75
t_2	0.75	1

We will show that the BERSM reduction with $m = 2$ and $L = 1$ applied to the scenario above does not result a BIC mechanism.

First consider just the stochastic matching problem when both the replicas and surrogates are drawn from the distribution. In the case that either the replicas are identical or the surrogates are identical, any matching obtains the same welfare. By symmetry, the matching from BERSM is uniform over the two matchings. Thus, the interesting case for analysis is that where the replica and surrogate types are both distinct. Without loss of generality assume that $r_1 = s_1 = t_1$ and $r_2 = s_2 = t_2$. Since our matching is replacing the values on the edges with Bernoulli draws, the optimal matching that results is stochastic. The straight-across matching has welfare 1.51 and the criss-crossed matching has welfare 1.50. The stochastic outcome can be summarized by Q the probability of the straight-across matching. If $Q < 1/2$ then the BERSM outcome is not incentive compatible (by Theorem 2.1). Misreporting effectively swaps the role of the straight-across and criss-crossed matchings; and for $Q < 1/2$ this is a strict improvement to welfare.¹²

¹²In fact, $Q < 1/2$ implies that the Binomial estimates variant of both RSM and RSF are not incentive compatible. The difference between these reductions is only the precise formula for utility in terms of Q ; in both $Q < 1/2$ implies non-welfare-optimality (with respect to swapping straight-across and criss-crossed matchings) and, thus, non-incentive-compatibility.

We calculate Q as follows. For $m = 2$ we have eight possible realizations over the edges (note that $v(r_2, x_2)$ is deterministically 1). See Figure 2.

We will denote each realization with a matrix $W = \begin{pmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & 1 \end{pmatrix}$ where $w_{i,j}$ denotes the Bernoulli realization assigned to the edge (r_i, s_j) . We will next present all possible realizations, their probability, and the outcome produced. The outcome is either the straight-across matching, the criss-crossed matching, or a uniform mix of the two matchings.

realization	probability	outcome
$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	$0.49 \cdot 0.25 \cdot 0.25$	straight across
$\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$	$0.49 \cdot 0.75 \cdot 0.25$	uniform mix
$\begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$	$0.49 \cdot 0.75 \cdot 0.25$	uniform mix
$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$	$0.49 \cdot 0.75 \cdot 0.75$	criss-crossed
$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$0.51 \cdot 0.25 \cdot 0.25$	straight across
$\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$	$0.51 \cdot 0.75 \cdot 0.25$	straight across
$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$0.51 \cdot 0.25 \cdot 0.75$	straight across
$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$	$0.51 \cdot 0.75 \cdot 0.75$	uniform mix

Summing over all scenarios, we have $Q < 1/2$ as:

$$Q = 0.49 (0.25 \cdot 0.25 + 2 \times 1/2 \cdot 0.25 \cdot 0.75) + 0.51 (0.25 \cdot 0.25 + 2 \times 0.25 \cdot 0.75 + 1/2 \cdot 0.75 \cdot 0.75) = 0.4890625. \quad \square$$

A.3 Proofs from Section 4.3.2

In this section we restate and prove results relating to the regularized RSF surrogate selection rule.

Theorem 4.4. *In the ideal model, RRSF is Bayesian incentive compatible.*

Proof. The constraints of the optimization problem in Step 2 ensure that the surrogate selection rule is stationary; in fact, the first constraint specifies that the matrix $\{q_{ij}\}_{i,j=1}^m$ is a Markov transition matrix, and the second constraint specifies that F is its stationary distribution. Thus, if all other agents are reporting types truthfully then $v(r_i, X_j)$ accurately reflects the expected valuation of type r_i for the allocation chosen by the mechanism conditional on s_j being selected as its surrogate.

The KKT condition (4.6) can be rewritten as

$$v(r_i, X_j) - \pi_j \leq \varrho_i + \partial_j \phi(\mathbf{q}_i), \quad (\text{A.1})$$

with equality whenever $q_{ij} > 0$. Combining this with the observations in the preceding paragraph, we can calculate the utility of an agent of type t for the outcome obtained when reporting type s . Let $\varrho_{\min} = \min_i \frac{\mu_i}{F_i}$.

$$\begin{aligned} \sum_j (v(r_t, X_j) - \pi_j) q_{sj} - \phi(\mathbf{q}_s) + \phi(\mathbf{0}) - \varrho_{\min} &\leq \sum_j (\varrho_t + \partial_j \phi(\mathbf{q}_t)) q_{sj} - \phi(\mathbf{q}_s) + \phi(\mathbf{0}) - \varrho_{\min} \\ &= \varrho_t + \nabla \phi(\mathbf{q}_t) \cdot \mathbf{q}_s - \phi(\mathbf{q}_s) + \phi(\mathbf{0}) - \varrho_{\min} \\ &= \varrho_t - \varrho_{\min} + \phi(\mathbf{0}) - \phi(\mathbf{q}_s) + \nabla \phi(\mathbf{q}_t) \cdot \mathbf{q}_s. \end{aligned} \quad (\text{A.2})$$

Furthermore, the first inequality in (A.2) is tight (i.e., holds with equality) when $t = s$ because (A.1) is tight in that case. Thus, the utility of an agent of type t who bids truthfully is

$$\sum_j (v(r_t, X_j) - \pi_j) q_{tj} - \phi(\mathbf{q}_t) + \phi(\mathbf{0}) - \varrho_{\min} = \varrho_t - \varrho_{\min} + \phi(\mathbf{0}) - \phi(\mathbf{q}_t) + \nabla \phi(\mathbf{q}_t) \cdot \mathbf{q}_t. \quad (\text{A.3})$$

Individual rationality is immediate from inspecting the right side of (A.3) and recalling that ϕ is convex, which implies $\phi(\mathbf{0}) \geq \phi(\mathbf{q}_t) - \nabla \phi(\mathbf{q}_t) \cdot \mathbf{q}_t$. Bayesian incentive compatibility follows by subtracting (A.3) from (A.2) to deduce that the benefit of misreporting s rather than truthfully reporting t is bounded above by

$$\phi(\mathbf{q}_t) - \phi(\mathbf{q}_s) - \nabla \phi(\mathbf{q}_t) \cdot (\mathbf{q}_t - \mathbf{q}_s),$$

which is non-positive because ϕ is convex. □

Lemma 4.5, restated below, relates the similarity of surrogate distributions of two replicas to their distinguishability and the convexity parameter of the regularizer. Regularized RSF was introduced specifically to have this property.

Lemma 4.5. *For any types $t, s \in [m]$, the corresponding surrogate distributions $\mathbf{q}_t, \mathbf{q}_s$ computed by the RRSF surrogate selection rule with a γ -strongly-convex regularizer ϕ satisfy*

$$\|\mathbf{q}_t - \mathbf{q}_s\|_1 \leq \frac{m}{\gamma} d(t, s),$$

where $d(t, s)$ is the distinguishability with respect to surrogate outcome distributions $\{X_1, \dots, X_m\}$.

The proof of Lemma 4.5 will follow shortly from two lemmas which we now state and prove.

Lemma A.2. *If ϕ is γ -strongly convex on a domain D and $\mathbf{q}, \mathbf{q}'$ are any two points of D , then $(\nabla \phi(\mathbf{q}) - \nabla \phi(\mathbf{q}')) \cdot (\mathbf{q} - \mathbf{q}') \geq \gamma \|\mathbf{q} - \mathbf{q}'\|^2$.*

Proof. Let $\psi(\mathbf{q}) = \phi(\mathbf{q}) - \frac{1}{2}\gamma\|\mathbf{q}\|^2$, and note that $\nabla \psi(\mathbf{q}) = \nabla \phi(\mathbf{q}) - \gamma\mathbf{q}$. Our assumption on ϕ implies that ψ is convex, from which we derive

$$\begin{aligned} \psi(\mathbf{q}) - \psi(\mathbf{q}') &\leq \nabla \psi(\mathbf{q}) \cdot (\mathbf{q} - \mathbf{q}'), \text{ and} \\ \psi(\mathbf{q}') - \psi(\mathbf{q}) &\leq \nabla \psi(\mathbf{q}') \cdot (\mathbf{q}' - \mathbf{q}). \end{aligned}$$

Summing these two inequalities we obtain

$$0 \leq (\nabla \psi(\mathbf{q}) - \nabla \psi(\mathbf{q}')) \cdot (\mathbf{q} - \mathbf{q}') = (\nabla \phi(\mathbf{q}) - \nabla \phi(\mathbf{q}') - \gamma\mathbf{q} + \gamma\mathbf{q}') \cdot (\mathbf{q} - \mathbf{q}'),$$

and the lemma follows by rearranging terms. □

By combining Lemma A.2 with the KKT conditions, we obtain the following lemma which expresses the similarity between surrogate distributions of RRSF for a pair of types in terms of their Euclidean distance. For a type $t \in [m]$, denote r_t and $X_t = \mathcal{A}(s_t)$ as in the definition of the RRSF surrogate selection rule (Definition 4.6) and introduce the notations $v_{tj} = v(r_t, X_j)$ and $\mathbf{v}_t = (v_{t1}, \dots, v_{tm})$.

Lemma A.3. *For any types $t, s \in [m]$, the corresponding surrogate distributions $\mathbf{q}_t, \mathbf{q}_s$ computed by the RRSF reduction with a γ -strongly-convex regularizer ϕ satisfy*

$$\|\mathbf{q}_t - \mathbf{q}_s\| \leq \frac{1}{\gamma} \|\mathbf{v}_t - \mathbf{v}_s\|.$$

Proof. The KKT condition (4.3) can be rewritten as

$$v_{tj} - \partial_j \phi(\mathbf{q}_t) = \frac{\lambda_{tj} + \mu_t}{F_t} + \pi_j. \quad (\text{A.4})$$

Introducing the notations $\mathbf{1} = (1, \dots, 1)$, $\boldsymbol{\lambda}_t = (\lambda_{t1}, \dots, \lambda_{tm})$, $\boldsymbol{\pi} = (\pi_1, \dots, \pi_m)$, we may combine the equations (A.4) over $j = 1, \dots, m$ into a single equation

$$\mathbf{v}_t - \nabla \phi(\mathbf{q}_t) = \frac{\boldsymbol{\lambda}_t + \mu_t \mathbf{1}}{F_t} + \boldsymbol{\pi}.$$

The same equation holds with s in place of t , and when we subtract the two equations and rearrange terms we obtain

$$\nabla \phi(\mathbf{q}_t) - \nabla \phi(\mathbf{q}_s) = (\mathbf{v}_t - \mathbf{v}_s) - \left(\frac{\boldsymbol{\lambda}_t}{F_t} - \frac{\boldsymbol{\lambda}_s}{F_s} \right) - \left(\frac{\mu_t}{F_t} - \frac{\mu_s}{F_s} \right) \mathbf{1}. \quad (\text{A.5})$$

Take the dot product of both sides of (A.5) with $\mathbf{q}_t - \mathbf{q}_s$, while noting that $\mathbf{1} \cdot \mathbf{q}_t = \mathbf{1} \cdot \mathbf{q}_s = 1$.

$$(\nabla \phi(\mathbf{q}_t) - \nabla \phi(\mathbf{q}_s)) \cdot (\mathbf{q}_t - \mathbf{q}_s) = (\mathbf{v}_t - \mathbf{v}_s) \cdot (\mathbf{q}_t - \mathbf{q}_s) - \left(\frac{\boldsymbol{\lambda}_t}{F_t} - \frac{\boldsymbol{\lambda}_s}{F_s} \right) \cdot (\mathbf{q}_t - \mathbf{q}_s).$$

The KKT conditions imply that $\boldsymbol{\lambda}_t \cdot \mathbf{q}_t = \boldsymbol{\lambda}_s \cdot \mathbf{q}_s = 0$ and that $\boldsymbol{\lambda}_t \cdot \mathbf{q}_s \leq 0$ and $\boldsymbol{\lambda}_s \cdot \mathbf{q}_t \leq 0$. Hence,

$$(\nabla \phi(\mathbf{q}_t) - \nabla \phi(\mathbf{q}_s)) \cdot (\mathbf{q}_t - \mathbf{q}_s) \leq (\mathbf{v}_t - \mathbf{v}_s) \cdot (\mathbf{q}_t - \mathbf{q}_s).$$

Lemma A.2 implies that the left side is bounded below by $\gamma \|\mathbf{q}_t - \mathbf{q}_s\|^2$, while Cauchy-Schwartz implies that the right side is bounded above by $\|\mathbf{v}_t - \mathbf{v}_s\| \|\mathbf{q}_t - \mathbf{q}_s\|$. Hence,

$$\gamma \|\mathbf{q}_t - \mathbf{q}_s\|^2 \leq \|\mathbf{v}_t - \mathbf{v}_s\| \|\mathbf{q}_t - \mathbf{q}_s\|$$

and the proposition follows upon dividing both sides by $\gamma \|\mathbf{q}_t - \mathbf{q}_s\|$. $\square$

Lemma 4.5 now extends the bound of Lemma A.3, by expressing the smoothness of RRSF in terms of the distinguishability of two types with respect to the surrogate outcome distributions $\{X_1, \dots, X_m\}$ rather than in terms of the Euclidean distance between the vectors $\mathbf{v}_t, \mathbf{v}_s$.

Proof of Lemma 4.5. The outcome produced by the regularized RSF reduction is invariant to uniform additive shifts of a type's values, i.e., $\mathbf{v}_t$ versus $\mathbf{v}_t + c\mathbf{1}$ for any constant c . The Euclidean distance between two types' values $\mathbf{v}_t$ and $\mathbf{v}_s$ is then minimized when the value vectors are in the same hyperplane orthogonal to $\mathbf{1}$. Thus, to bound $\|\mathbf{q}_t - \mathbf{q}_s\|_2$ (an outcome of the algorithm) in terms of $\|\mathbf{v}_t - \mathbf{v}_s\|_2$ we may as well assume that for all types t , the value vector $\mathbf{v}_t$ is normalized by subtracting $(\frac{\mathbf{v}_t \cdot \mathbf{1}}{n}) \mathbf{1}$, equivalently, that $\mathbf{v}_t \cdot \mathbf{1} = 0$. Adopting this assumption for the remainder of the proof, we have,

$$\begin{aligned} \|\mathbf{q}_t - \mathbf{q}_s\|_1 &\leq \sqrt{m} \|\mathbf{q}_t - \mathbf{q}_s\|_2 \\ &\leq \frac{\sqrt{m}}{\gamma} \|\mathbf{v}_t - \mathbf{v}_s\|_2 \\ &\leq \frac{m}{\gamma} \|\mathbf{v}_t - \mathbf{v}_s\|_\infty \\ &\leq \frac{m}{\gamma} d(t, s). \end{aligned}$$

The first and third lines follow from standard properties of the norms $\|\cdot\|_1$, $\|\cdot\|_2$, and $\|\cdot\|_\infty$. The second line follows from Lemma A.3. The following argument justifies the last line and concludes the proof.

The distinguishability $d(t, s)$ with respect to surrogate outcome distributions is

$$\begin{aligned} d(t, s) &= \max_{j, j'} [v_{tj} - v_{sj} - v_{tj'} + v_{sj'}] \\ &= \max_j [v_{tj} - v_{sj}] - \min_{j'} [v_{tj'} - v_{sj'}], \end{aligned} \tag{A.6}$$

i.e., the maximum difference of two coordinates of the vector $\mathbf{v}_t - \mathbf{v}_s$. Our assumption that $\mathbf{v}_t \cdot \mathbf{1} = \mathbf{v}_s \cdot \mathbf{1} = 0$ implies that the maximum coordinate of $\mathbf{v}_t - \mathbf{v}_s$ is positive while the minimum coordinate is negative. Thus,¹³

$$d(t, s) \geq \|\mathbf{v}_t - \mathbf{v}_s\|_\infty. \quad \square$$

We conclude the section of omitted lemmas with the proof that the RRSF reduction approximately preserves (or improves) social surplus.

Lemma 4.6. *For convex regularizer ϕ with maximum $\phi_{\max}$ and minimum $\phi_{\min}$ on the probability simplex $D = \{\mathbf{q} \in [0, 1]^m \mid \sum_j q_j = 1\}$, the expected contribution to the surplus of agent κ in the regularized RSF mechanism $\mathcal{M}$ relative to the algorithm $\mathcal{A}$ satisfies $\mathcal{M}^\kappa \geq \mathcal{A}^\kappa - (\phi_{\max} - \phi_{\min})$.*

Proof. It is feasible (but generally suboptimal) for the regularized RSF to output the identity, i.e., to assign replica r_i to surrogate outcome $\mathcal{A}(s_j)$ with $i = j$ and $r_i = s_j$. This outcome is the same as given by the original algorithm $\mathcal{A}$. Denote this non-optimal solution to the program (4.2) by $\{q_{ij}^A\}$ as with $q_{ij}^A = 1$ with $i = j$ and otherwise $q_{ij}^A = 0$. The objective function of (4.2) at $\{q_{ij}^A\}$ evaluates to

$$\sum_i F_i v(r_i, X_i) - \sum_i F_i \phi(\mathbf{q}_i^A) \geq \mathcal{A}^\kappa - \phi_{\max}. \tag{A.7}$$

Since the RRSF reduction chooses the optimal solution to (4.2), denoted $\{q_{ij}\}$, it must be that the value of the objective function at $\{q_{ij}\}$ is greater than or equal to the right side of (A.7). Thus,

$$\mathcal{A}^\kappa - \phi_{\max} \leq \sum_{i,j} F_i v(r_i, X_j) q_{ij} - \sum_i F_i \phi(\mathbf{q}_i) = \mathcal{M}^\kappa - \sum_i F_i \phi(\mathbf{q}_i) \leq \mathcal{M}^\kappa - \phi_{\min}.$$

The lemma follows by rearranging terms. $\square$

---

¹³A more careful analysis can show $d(t, s) \geq 2 \|\mathbf{v}_t - \mathbf{v}_s\|_\infty$.