Approximate Random Allocation Mechanisms*

Mohammad Akbarpour† Afshin Nikzad‡

Abstract. We generalize the scope of random allocation mechanisms, in which the mechanism first identifies a feasible "expected allocation" and then implements it by randomizing over nearby feasible integer allocations. Previous literature had shown that the cases in which this is possible are sharply limited. We show that if some of the feasibility constraints can be treated as goals rather than hard constraints then, subject to weak conditions that we identify, any expected allocation that satisfies all the constraints and goals can be implemented by randomizing among nearby integer allocations that satisfy all the hard constraints exactly and the goals approximately. By defining ex post utilities as goals, we are able to improve the ex post properties of several classic assignment mechanisms, such as the random serial dictatorship. We use the same approach to prove the existence of ϵ-competitive equilibrium in large markets with indivisible items and feasibility constraints.

Keywords: Market Design, Matching, Random Allocation, Concentration Inequalities, Random Serial Dictatorship

JEL Classification Numbers: C78, D47, D82.

---

*Several conversations with Paul Milgrom, Alvin Roth, Eric Budish, and Roger Myerson have been essential. We are grateful to Ben Brooks, Gabriel Carroll, Yeon-Koo Che, Kareem Elnahal, Alex Frankel, Emir Kamenica, Fuhito Kojima, Matthew Jackson, Bobby Pakzad-Hurson, Bob Lucas, Ilya Segal, Bob Wilson, and Alex Wolitzky for their great suggestions, and several seminar participants for their comments and feedback. All errors are ours.

†Graduate School of Business, Stanford University. Email: mohamwad@stanford.edu

‡Department of Economics, Stanford University. Email: nikzad@stanford.edu

1 Introduction

When cash transfers are limited and goods are indivisible, it can sometimes be impossible to allocate goods in an efficient and envy-free ("fair") way. This challenge is faced, for example, when assigning students to courses, cadets to military bases, or setting a competitive sports schedule. Early economic studies of this problem by Hylland and Zeckhauser ("HZ") and Bogomolnaia and Moulin ("BM") assume that each agent must receive just a single good and show that it is then possible to allocate the probabilities of receiving each good in an efficient, envy-free manner [Hylland and Zeckhauser, 1979], [Bogomolnaia and Moulin, 2001]. Budish, Che, Kojima, and Milgrom ("BCKM") propose expanding this approach to a wider set of multi-item allocation problems in which the constraints may be more complex than merely a set of one-item-per-person constraints [Budish et al., 2013]. For example, in course allocation, a student may wish to have at least one class in science and one in humanities in a particular term. They show that for any expected allocation that satisfies all the constraints, if the constraints have a particular "bihierarchy" structure, then the expected allocation can always be achieved by randomizing among pure allocations in which each fractional expected allocation is rounded up or down to an adjacent integer and all the constraints are simultaneously satisfied. When the conditions are satisfied, this sometimes makes it possible to use mechanisms that select efficient, envy-free expected allocations and to implement those through randomization.

However, BCKM also found that the bihierarchy condition can be a necessary condition, and so even their expansion of the previous works can rule out some potential applications. For instance, the condition is violated in school choice if a school with limited capacity has at least two of the walk-zone, gender, or racial diversity constraints.

The goal of this paper is to generalize this approach to a much broader class of allocation problems by reconceptualizing the role of constraints. Our analysis shows that many more constraints can be managed if some of them are "soft", in the sense that they can bear small errors with relatively small costs. More precisely, we partition the full set of constraints into a set of *hard* constraints that must always be satisfied exactly, and a set of *soft* constraints that should be satisfied approximately. The main theorem of the paper identifies a rich constraint structure that is approximately implementable, meaning that if an expected allocation satisfies all the constraints, then it can be implemented by randomizing among pure allocations that satisfy all the hard constraints and satisfy the soft constraints with only small errors.

The importance of this result arises from the way it expands potential applications. In the school choice example, the requirement that each student must be assigned to exactly one school is (in our conception) a hard constraint that must be satisfied, but the requirement that 50% of students in a school live in the walk zone may be a soft constraint – if necessary, 48% will do. Allowing this flexibility is particularly important when the constraints are inconsistent, and in other cases it provides greater scope for accommodating individual student preferences.

1.1 Model and Contributions

In this paper, we analyze a general model of matching with indivisible objects. Section 2 introduces the building blocks of our model. In Section 3, we propose a new notion of approximate implementation. A constraint is approximately satisfied if the probability of violating that constraint is exponentially decreasing in the size of the constraint.[1] We partition the set of constraints into a set of hard constraints that are inflexible and a set of soft constraints that are flexible, and we call it a *hard-soft partitioned* constraint set. We say that a hard-soft partitioned constraint set is *approximately implementable* if for any feasible fractional assignment that satisfies both hard and soft constraints, there exists a lottery over pure assignments such that the following three properties hold: (*i*) the expected value of the lottery is equal to the fractional assignment, (*ii*) the outcome of the lottery satisfies hard constraints, and (*iii*) the outcome of the lottery satisfies soft constraints approximately. The question that we ask is: What kinds of hard-soft partitioned constraint structures are approximately implementable?

The first theoretical contribution of the paper is stated in Theorem 1. The theorem identifies a rich structure for soft constraints under which the whole structure is approximately implementable, given that the structure of hard constraints is the same maximal structure introduced in BCKM – the "bihierarchical" structure. We augment this theorem in Appendix H by showing that the proved bounds are tight.

We prove Theorem 1 in the Appendix A by constructing a matching algorithm which approximately implements any given feasible fractional assignment. We introduce a matrix operation that takes a fractional assignment as its input and (randomly) generates another assignment with fewer fractional elements as its output. By iterative applications of this operation, an integral assignment is generated.[2] The (random) assignment matrix satisfies the martingale property, i.e. the expected value of the assignment matrix after the next iteration remains the same as its current value. We apply probabilistic concentration bounds to our

[1]Intuitively, the size of a constraint is the value of the right-hand side (or left-hand size) of a constraint. If a school has a capacity for 1000 students, then the size of this capacity constraint is 1000.

[2]It is worth mentioning that the randomized mechanism stops in *polynomial* time, which is an important requirement for a practical matching algorithms in relatively large markets.

randomized mechanism in order to prove that soft constraints are satisfied with small errors. It is worth mentioning that the previous literature on the economic theory of implementation relies on the Birkhoff-von Neumann theorem [Birkhoff, 1946, Neumann, 1953] (in HZ and BM) or its generalizations, such as [Edmonds, 2003] (e.g. the implementation method of BCKM is based on a theorem of Edmonds on deterministic rounding of mathematical programs). Our paper, on the contrary, builds an implementation method by bringing and developing techniques from the randomized rounding literature into matching theory.[3]

The second theoretical contribution of the paper considers a setting with agent *types*, where two agents have the same type if the set of constraints imposed on them is the same. Theorem 2 shows that a modified version of our allocation mechanism can implement a fractional assignment in such a way that none of the soft constraint are violated with more than an additive error equal to the number of agent types.

The rest of the paper explores applications of our framework. In Section 5, we employ our framework to improve two classical allocation mechanisms by transforming some of their *ex ante* properties to *ex post* properties. For many properties, market-makers are concerned with ex post properties of allocation mechanisms. An ex ante "fair" allocation is not necessarily fair ex post. Our main result in this part guarantees that under our proposed randomized mechanism, ex post utilities of the agents (or objects) are approximately equal to their ex ante utilities, and ex post social welfare is approximately equal to the ex ante social welfare.

In Section 5.1, we employ our utility and welfare guarantees to improve the random serial dictatorship (RSD) mechanism[4]. It is well-known that RSD with multi-unit demand is ex ante fair and strategy-proof, but ex post (very) unfair. We fix the ex post unfairness of RSD by constructing the expected allocation associated with the RSD, and then approximately implementing it by employing our main theorem. In addition, our utility bounds guarantee that risk-averse agents ex ante prefer our implementation mechanism to the standard RSD.

Next, in Section 5.2, we employ the pseudo-market mechanism introduced in BCKM to construct a fractional assignment, and then employ our method to implement that fractional assignment, while (approximately) satisfying many more constraints, compared to the pseudo-market mechanism of BCKM. Our utility bounds show that ex post utilities are approximately equal to ex ante utilities, and therefore the mechanism is "approximately envy-free".

In another application in Section 6, we prove the existence of ϵ-competitive equilibrium

[3]We provide a brief review of this literature in the related work section.

[4]RSD works as follows: draw a fair random ordering of the agents and then let agents select their most favorite bundle of objects (among those remaining) according to the realized random ordering without violating the constraints.

(ϵ-CE)[5] in a market with indivisible objects and distributional constraints. Two features distinguish this result from the previous ϵ-CE existence results: First, one can impose hierarchical constraints to the final (indivisible) allocation and the existence result will continue to hold. This can be applied to, for instance, an online advertisement setting where multiple advertiser are buying impressions, and they prefer to diversify the set of their audience (For example, Facebook is considering a competitive equilibrium model of pricing where advertisers specify their target audience using distributional constraints [Hou et al., 2016]). Second, the method we provide to prove the existence would also find the ϵ-CE with high probability, if it is given an ϵ-CE for the corresponding market with divisible items.

Lastly, we discuss applications of our framework in implementing intersecting constraints in the school choice setting in Section 7. In particular, we introduce a new method to accommodate walk-zone priorities in the school choice. The way many school choice systems such as the Boston Public Schools handle walk-zone constraints is that schools are required to dedicate a specific fraction of their seats to students with "walk-zone" priority. This means that the lottery treats two students who are a few blocks away, but on the two sides of a walk-zone border, very differently. This "discontinuity" comes with considerable consequences: for example, a student could get lower priority in a school that is closer to her than a school further away. Our framework, however, allows the market maker to directly incorporate each student's *distance* from different schools into the constraints.

1.2 Related Work

Randomization is commonplace in everyday life and has been studied in various settings such as school choice, course allocation, and house allocation [Abdulkadiroglu et al., 2005, Abdulkadiroglu and Sonmez, 1998, Budish, 2011, Pathak and Sethuraman, 2011]. Perhaps the most practically popular random mechanism is to draw a fair random ordering of agents and then let the agents select their most favorite object (among those remaining) according to the realized random ordering without violating the constraints. This mechanism, which is known as *Random Serial Dictatorship* (RSD) is a desirable mechanism as it is strategy-proof and ex post Pareto efficient [Abdulkadiroglu and Sonmez, 1998, Chen and Sonmez, 2002]. Nevertheless, RSD is *ex ante* inefficient, ex post (highly) unfair, and cannot handle lower quotas [Bogomolnaia and Moulin, 2001, Kojima, 2009, Hatfield, 2009]. Several papers compare PS and RSD and analyze their connections in large markets [Manea, 2009, Kojima and Manea, 2010, Che and Kojima, 2010, Liu and Pycia, 2016].

[5] An ϵ-equilibrium in an indivisible objects setting is a vector of prices and a partition of objects in which all agents' utilities are at least $(1-\epsilon)$ of their utilities in the competitive equilibrium if objects were divisible, no agent's budget constraint is violated, and market clears.

The idea to construct a fractional assignment and then implementing it by a lottery over pure assignments was first introduced in [Hylland and Zeckhauser, 1979] for cardinal utilities. [Bogomolnaia and Moulin, 2001] construct a mechanism, the *Probabilistic Serial Mechanism* (PS), for ordinal utilities based on the same technique. Both papers model one-to-one matching markets with no other constraints. [Hashimoto, 2016] approximates implementation of HZ equilibrium, and its primary focus is on feasibility (no additional chairs to students) and strategy-proofness (ex post incentive compatibility). The approximated CEEI mechanism is exactly feasible and exactly strategy-proof, but efficiency and envy-freeness are achieved only in approximate senses. [Budish et al., 2013] build on those two papers by considering a richer constraint structure.[6] Our paper generalizes this literature by designing a randomized mechanism which can accommodate a much richer class of constraints.

The approximate satisfaction of constraints has been studied in a few recent papers. In [Budish, 2011], Budish studies the problem of combinatorial assignment by introducing a notion of approximate competitive equilibrium from equal incomes, which treats course capacities as flexible constraints. A "soft bound" approached is introduced in [Ehlers et al., 2014], where authors introduce a deferred acceptance algorithm with soft bounds in which they adjust group-specific lower and upper bounds to achieve a fair and non-wasteful mechanism. There are some key points that separate our paper from these works. First, we propose a framework which can handle "overlapping" constraints. For instance, in the school choice setting, we can accommodate racial, gender, and walk-zone priority constraints simultaneously. Second, we provide a rich language for the market-maker to declare a partitioned constraint set, which contains both flexible and inflexible constraints. Third, our mechanism runs in polynomial time, whereas the approach introduced in [Budish, 2011], as discussed in [Budish et al., 2016, Rubinstein, 2014], is computationally hard.[7]

Compared to BCKM, who build their implementation method based on a theorem of Edmonds on deterministic rounding of mathematical programs, we build our implementation method based on the literature on randomized rounding. Various rounding techniques have been developed in the computer science literature; [Ageev and Sviridenko, 2004, Gandhi et al., 2006, Chekuri et al., 2010] are among the closest to our work. [Ageev and Sviridenko, 2004] introduce a deterministic randomized rounding method, called *pipage rounding*, and [Gandhi et al., 2006, Chekuri et al., 2010] design

[6] In a recent work, Pycia and Ünver study a more general structure (the *Totally Unimodular* or TU structure) and show that they can accommodate constraints such as strategy-proofness and envy-freeness as linear constraints as long as they fit into the TU structure [Pycia and Ünver, 2015]. Our approach is conceptually different from theirs since we consider flexible constraints (i.e. goals) which may not fit into the TU structure. [Kesten et al., 2015] also works with fractional allocations and improves RSD.

[7] [Alon et al., 2015] also use an approximation approach and propose a polynomial time algorithm to solve the problem of couples in the Israeli Medical Match problem.

rounding methods following the same idea, although in a randomized fashion and for different applications. We remark that none of these methods could be used directly to handle our application, i.e. a bihierarchical constraint structure with upper and lower quotas. We design our implementation method by extending the approach of [Gandhi et al., 2006] to bihierarchical structures. The techniques used in [Gandhi et al., 2006], although inspired our design, are specifically designed for the job scheduling problem. As a result, their randomized algorithms accommodate neither non-local soft constraints, nor (bi)hierarchical hard constraints.

Other rounding methods have been used in the literature for (approximately) implementing fractional allocations. [Nguyen et al., 2014, Nguyen and Vohra, 2015] model matching markets with complementarities. They use *iterative rounding* methods [Lau et al., 2011] to design implementation schemes specific to their problem structure. The goal there is to handle complementarities (in frameworks with only capacity constraints), while our paper is concerned with implementing generalized constraint structures (and is not concerned with complementarities).[8]

The problem of reduced-form implementation in the auction literature is also related to our work [Matthews, 1984, Border, 1991, Che et al., 2013]. In this problem, an *interim* allocation, which describes the marginal probabilities of each bidder obtaining the good as a function of his type, is constructed. Then, same as our problem, the question that is asked here is: which interim allocations can be implemented by a lottery over feasible pure allocations? The approximate satisfaction of constraints, however, is not studied in that literature.

2 Setup

Consider an environment in which a finite set of **objects** O has to be allocated to a finite set of **agents** N. We denote the set of agent-object pairs, $N \times O$, by E, where each $(n, o) \in E$ is an **edge**. Sometimes we use 'e' to denote edges. A **pure assignment** is defined by a non-negative matrix $X = [X_{no}]$ where each $X_{no} \in \{0, 1\}$ denotes the amount of object o which is assigned to agent n for all $(n, o) \in E$. We require the matrix to be binary valued to capture the indivisibility of the objects.

A **block** $B \subseteq E$ is a subset of edges. A **constraint** S is a triple $(B, \underline{q}_B, \bar{q}_B)$, which is a block B associated with a vector of integer **quotas** $(\underline{q}_B, \bar{q}_B)$ as the floor and ceiling quotas

[8]The specific structure of [Nguyen et al., 2014] allows them to provide small additive bounds on the violation of capacity constraints by using techniques different than ours. We also show that under (reasonably) specific structures, our technique can provide small additive error bounds (Section 4.3)

on B. A **structure** is a subset $\mathcal{E} \subseteq 2^E$; i.e. a collection of blocks. A **constraint set** is a set of constraints. Let $\mathbf{q} = [(\underline{q}_B, \bar{q}_B)_{B\in\mathcal{E}}]$.

We say that X is **feasible with respect to** $(\mathcal{E}, \mathbf{q})$ (or simply, with respect to $\mathcal{E}$ when $\mathbf{q}$ is clearly known from the context) if

$$\underline{q}_B \leq \sum_{e\in B} X_e \leq \bar{q}_B \qquad \forall B \in \mathcal{E}. \tag{1}$$

We call a block $B \in \mathcal{E}$ **agent k's capacity block** when $B = \{X_{kj} | j \in O\}$. Similarly, we call a block $B \in \mathcal{E}$ an **object m's capacity block** when $B = \{X_{im} | i \in N\}$. A **capacity constraints** is a constraint $(B, \underline{q}_B, \bar{q}_B)$, where $B \in \mathcal{E}$ is a capacity block. We sometimes refer to capacity constraints of agents and objects as *row constraints* and *column constraints*, respectively.

A **fractional assignment** is defined by a matrix $x = [x_{no}]$, where each $x_{no} \in [0, 1]$ is the quantity of object o assigned to agent n. To distinguish between pure and fractional assignments, we usually use X to denote a pure assignment and x for a fractional assignment. We sometimes use the term *expected assignment* to address fractional assignments.

Given a structure $\mathcal{E}$ and associated quotas $\mathbf{q}$, a fractional assignment matrix x is **implementable under quotas q** if there exist positive numbers $\lambda_1, \ldots, \lambda_K$, which sum up to one, and pure assignments $X_1, \ldots, X_K$, which are feasible under $\mathbf{q}$, such that

$$x = \sum_{i=1}^{K} \lambda_i X_i.$$

We also say that a structure $\mathcal{E}$ is **universally implementable** if, for any quotas $\mathbf{q} = (\underline{q}_B, \bar{q}_B)_{B\in\mathcal{E}}$, every fractional assignment matrix satisfying $\mathbf{q}$ is implementable under $\mathbf{q}$.

The main existing theoretical result on the implementability of a structure is introduced in the BCKM's paper [Budish et al., 2013], where they identify *bihierarchy* as a sufficient condition for universal implementability of a structure. More precisely, a structure $\mathcal{H}$ is a **hierarchy** if for every pair of blocks B and B' in $\mathcal{H}$, we have that $B' \subset B$ or $B \subset B'$ or $B \cap B' = \emptyset$. A simple hierarchy is depicted in Figure 2.1. A structure $\mathcal{H}$ is a **bihierarchy** if there exists two hierarchies $\mathcal{H}_1$ and $\mathcal{H}_2$ such that $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$. The following theorem identifies a sufficient, and almost necessary, condition under which a structure is universally implementable.

Theorem 0. *[BCKM, 2013] If a structure $\mathcal{H}$ is a bihierarchy, then it is universally implementable. In addition, if $\mathcal{H}$ contains all agents and objects capacity blocks, then it is universally implementable if and only if it is a bihierarchy.*

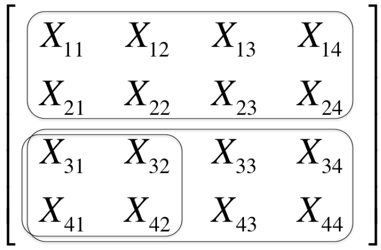

Figure 2.1: A hierarchy

3 Approximate Implementation

In many assignment problems, the involved constraints have multiple intersections and the bihierarchy assumption fails. The following example clarifies the bihierarchy limitations in the school choice setting.

Example 1. *In the Boston School Program (as of January 2016), fifty percent of each school seats were set aside for walk-zone priority students. Consider a school which also has a group-specific quota on low socioeconomic status (SES) students. Together with the requirement that each student should be assigned to one school, these blocks do not form a bihierarchy.*

In this paper, we show that by treating some constraints as *goals* rather than inflexible constraints, we can accommodate many more constraints.

More precisely, we ask the market-maker to partition the full set of constraints into a set of hard constraints that must be satisfied exactly and a set of soft constraints that must be satisfied approximately. Accordingly, the constraint structure will be partitioned into two sets: a set of **hard** blocks, $\mathcal{H}$, which are blocks of inflexible constraints, and a set of **soft** blocks, $\mathcal{S}$, which are blocks of flexible constraint. We refer to $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ as a **hard-soft partitioned structure**, or simply a partitioned structure.

Another way in which our framework generalizes BCKM is that in our model elements of soft constraints can have any arbitrary weights; that is, for a soft block B', we say X is feasible with respect to B' if:

$$\underline{q}_{B'} \leq \sum_{e \in B'} w_e X_e \leq \bar{q}_{B'}$$

where w_e can take any arbitrary non-negative value and $\underline{q}_{B'}$ and $\bar{q}_{B'}$ can be any non-negative real number. The weights associated with an edge need not be equal for all blocks. Recall that, similar to BCKM, for a hard block B, we require $w_e = 1$ for all $e \in B$ and $\underline{q}_B$

and $\bar{q}_B$ are restricted to be non-negative integers. This generalization expands the scope of practical applications of the model; this will be discussed in Section 7.

Our goal in this paper is to identify structural conditions imposed on $\mathcal{H}$ and $\mathcal{S}$ under which $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ is "approximately implementable". In the following, we rigorously define the notion of approximate implementation.

Definition 1. *Given a hard-soft partitioned structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$, we say $\mathcal{E}$ is* **Approximately Implementable** *if for any vector of quotas $\mathbf{q}$ and any expected assignment x which is feasible with respect to $(\mathcal{E}, \mathbf{q})$, there exists a lottery (probability distribution) over pure assignments $X_1, \ldots, X_K$ such that, if we denote the outcome of the lottery by the random variable X, the following properties hold:*

P1. **Assignment Preservation:** $\mathbb{E}[X] = x$.

P2. **Exact Satisfaction of Hard Constraints:** *All constraints in $\mathcal{H}$ are satisfied.*

P3. **Approximate Satisfaction of Soft Constraints:** *For any soft block $B \in \mathcal{S}$ with $\sum_{e \in B} x_e = \mu$ and for any $\epsilon > 0$, we have*

$$\Pr(\mathsf{dev}^+ \geq \epsilon\mu) \leq e^{-\mu\frac{\epsilon^2}{3}} \tag{2}$$

$$\Pr(\mathsf{dev}^- \geq \epsilon\mu) \leq e^{-\mu\frac{\epsilon^2}{2}} \tag{3}$$

where dev^+ and dev^- are defined as follows:[9]

$$\mathsf{dev}^+ = \max\big(0, \sum_{e \in B} X_e - \mu\big)$$

$$\mathsf{dev}^- = \max\big(0, \mu - \sum_{e \in B} X_e\big)$$

Property 1 simply states that there exists a lottery which implements x. Property 2 states that hard constraints are satisfied with no error. Property 3 defines our notion of approximation. By this property, the probability of violating a soft constraint by a factor greater than ϵ decays exponentially with the right-hand side (or the left-hand side) of the constraint. Figure 3.1 illustrates the strength of our bounds as μ grows. Property 3 also guarantees that the probability of violating soft constraints by a multiplicative factor ϵ exponentially decays with ϵ. For example, in a school with 2000 seats, the probability

[9] In the case of having weights, definitions would be adjusted similarly to $\mathsf{dev}^+ = \max\big(0, \sum_{e \in B} w_e X_e - \mu\big)$, where $\mu = \sum_{e \in B} w_e x_e$.

of admitting more than 2100 students is bounded above by 0.19, while the probability of admitting more than 2200 students is no more than 0.0013.

The probabilistic bounds of Definition 1 might not seem practical for small markets. We address this concern in Section 4.4 after stating our main results. One may also wonder why implementation in our setting is a non-trivial problem, and why simple implementation approaches fail. We will discuss this issue in Section 4.5.

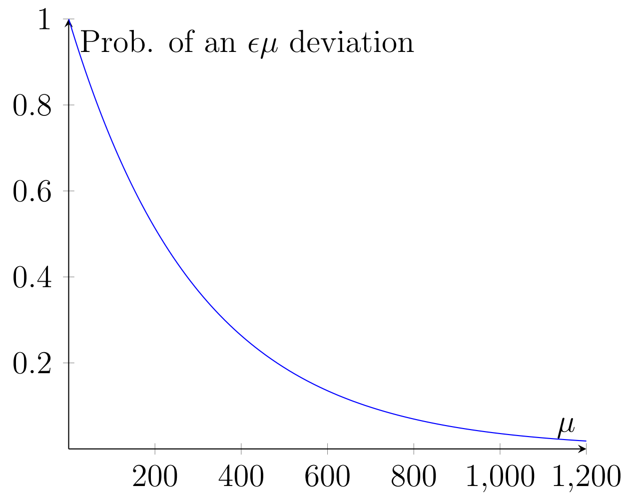

Figure 3.1: This graph corresponds to the theoretical upper bound that we provide on the probability of an $\epsilon\mu$ deviation in Equation 2. For a fixed $\epsilon = 0.1$, the probability of observing a deviation more than $\epsilon\mu$ goes to zero exponentially as μ increases.

4 The Main Theorems

Given a partitioned structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$, we first identify structures for $\mathcal{H}$ and $\mathcal{S}$ under which $\mathcal{E}$ is approximately implementable in the sense of Definition 1. We then state a generalized version of our main theorem and show that given a *bi*hierarchy of hard constraints, *any* soft constraint can be approximately satisfied, but with a weaker notion of approximate satisfaction. Finally, in Section 4.3, under more specific constraint structures we provide more powerful, additive bounds.

First of all, note that Theorem 0 shows that even if $\mathcal{S} = \emptyset$ (i.e., there are no soft constraints), in order for $\mathcal{E}$ to be implementable, bihierarchy is a sufficient and (almost) necessary condition[10] for $\mathcal{H}$. In other words, the bihierarchy is the weakest condition we can impose on hard constraints. We maintain this maximal structure and let hard blocks form

[10] We use the term "almost" because it is not a necessary condition in general, but it is necessary in two-sided matching markets with finite capacities.

a bihierarchy; i.e., we assume $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$, where $\mathcal{H}_1$ and $\mathcal{H}_2$ are two hierarchies. Then, given a bihierarchical hard structure, we aim to identify a structural condition, if any, for soft blocks $\mathcal{S}$ under which $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ is approximately implementable. It is worth pointing out that when $\mathcal{H}$ is a bihierarchy, a fully general set of soft constraints is not approximately implementable (this is shown in Appendix D).

4.1 The Structure of Soft Blocks

Now we show that if $\mathcal{H}$ forms a bihierarchy, there exists a rich structure for the soft blocks $\mathcal{S}$ under which $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ is approximately implementable. To do so, we need to define one new concept. For a block $B \in \mathcal{S}$, we say that B is in the **deepest level of $\mathcal{H}_1$** if for any block $C \in \mathcal{H}_1$, either $B \subseteq C$ or $B \cap C = \emptyset$. (See Figure 4.1 for an illustration) We also say that $B \in \mathcal{S}$ is in the **deepest level of a bihierarchy $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$** if it is in the deepest level of either of $\mathcal{H}_1$ or $\mathcal{H}_2$.

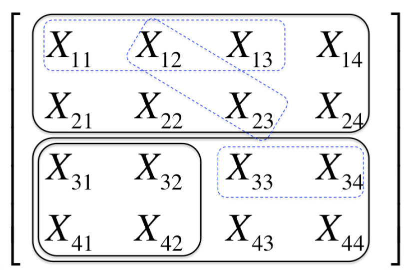

Figure 4.1: The solid blocks form a hierarchy $\mathcal{H}_1$. The dashed blocks are in the deepest level of $\mathcal{H}_1$. A block that, for example, contains X_{32} and X_{33} is not in the deepest level of $\mathcal{H}_1$.

Theorem 1. *[The Main Theorem] Let $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ be a hard-soft partitioned structure such that $\mathcal{H}$ is a bihierarchy and any block in $\mathcal{S}$ is in the deepest level of $\mathcal{H}$. Then, $\mathcal{E}$ is approximately implementable.*

Proof Overview. We present only an overview of the proof here. The full proof can be found in the Appendix A. The proof is constructive; that is, we propose a randomized mechanism that, given a partitioned structure with properties as described in Theorem 1, approximately implements a given feasible fractional assignment. To do so, let us define a constraint to be *tight* if it is binding, and to be *floating* otherwise. This definition applies to the implicit constraints $0 \leq x_e \leq 1$ for all $e \in E$.

The core of our randomized mechanism is a probabilistic operation that we design, called **Operation $\mathcal{X}$**. We iteratively apply Operation $\mathcal{X}$ to the initial fractional assignment until a pure assignment is generated. At each iteration t, the fractional assignment x_t is converted to x_{t+1} in a way such that: (1) the number of floating constraints decreases, (2) $\mathbb{E}(x_{t+1}|x_t) = x_t$, and (3) x_{t+1} is feasible with respect to $\mathcal{H}$. The first property guarantees that after a finite (and small) number of iterations[11], the obtained assignment is pure. The second property ensures that the resulting pure assignment is equal to the original fractional assignment *in expectation*. The third property guarantees that all hard constraints are satisfied throughout the whole process of the mechanism.

In the last step, we prove that after iterative applications of Operation $\mathcal{X}$, soft constraints are approximately satisfied. Roughly speaking, we design Operation $\mathcal{X}$ in such a way that it never increases (or decreases) two (or more) elements of a soft constraint at the same iteration. Consequently, elements of each soft block become "negatively correlated". Negative correlation then allows us to employ probabilistic concentration bounds to prove that soft constraints are approximately satisfied. In Appendix A, we design Operation $\mathcal{X}$ and prove its desired properties.[12] □

In Appendix H, we prove that the probabilistic bounds proved in this theorem are tight; that is, to improve them, one needs to impose more structure on constraints.

4.2 Two Corollaries of Theorem 1

An immediate corollary of the Theorem 1 asserts that if $\mathcal{H}$ forms a single hierarchy (rather than two hierarchies), then for *any* arbitrary set of soft constraints, $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ is approximately implementable.

Corollary 1 (of Theorem 1)**.** *Let $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ be a hard-soft partitioned constraint set, where $\mathcal{H}$ is a single hierarchy; i.e., $\mathcal{H}_1 = \emptyset$ or $\mathcal{H}_2 = \emptyset$. Then, for all $\mathcal{S} \subseteq 2^E$, $\mathcal{E}$ is approximately implementable.*

Proof. By assumption, at least one of the $\mathcal{H}_1$ or $\mathcal{H}_2$ is empty. Without loss of generality, suppose $\mathcal{H}_1 = \emptyset$. We add a "dummy" constraint to $\mathcal{H}_1$, which contains all the elements, i.e. the constraint $0 \le \sum_{e \in E} x_e < \infty$. Obviously, any soft constraint block is in the deepest level of $\mathcal{H}_1$. Hence, by Theorem 1, $\mathcal{E}$ is approximately implementable. ■

The second corollary of the Theorem 1 is about **local** structures. We say that a block is local if it involves one agent with possibly multiple objects or one object with possibly

[11] Our randomized mechanism stops after at most $|\mathcal{H}| + |E|$ iterations.

[12] We would like to remind that the Operation $\mathcal{X}$ is designed by bringing techniques from the randomized rounding literature; e.g. [Gandhi et al., 2006], as well as developing new techniques. We discussed this issue extensively in Section 1.2.

multiple agents, but not multiple agents and multiple objects at the same time. Equivalently, a block is local if it includes a subset of the elements of a single column or a single row.[13]

Corollary 2 (of Theorem 1)**.** *Let $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ be a structure such that $\mathcal{H}$ is the set of all capacity blocks and $\mathcal{S}$ only contains local blocks. Then $\mathcal{E}$ is approximately implementable.*

Proof. Since $\mathcal{H}$ is the set of all capacity blocks, any block in $\mathcal{S}$ is in the deepest level of $\mathcal{H}_1$ or $\mathcal{H}_2$. The corollary follows from Theorem 1. ∎

We can use Theorem 1 to provide probabilistic bounds for *any* soft constraint, even if it is not in the deepest level of $\mathcal{H}$. We say that a block $B \in \mathcal{S}$ is in **depth k of hierarchy** $\mathcal{H}_1$ if B can be partitioned into k subsets $B_1, B_2, \cdots, B_k$ such that all are in the deepest level of $\mathcal{H}_1$, and moreover, no partitioning of B into $k-1$ subsets satisfies this property We also say that $B \in \mathcal{S}$ is in the **depth k of bihierarchy** $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$ if it is in depth k of either of $\mathcal{H}_1$ or $\mathcal{H}_2$.

With these definitions, the exact same random mechanism that we designed implements a feasible fractional allocation in such a way that all hard constraints are exactly satisfied, and soft constraints that are in depth k of $\mathcal{H}$ are satisfied approximately, with the following notion of approximation: (instead of (2) and (3))

$$\Pr(\mathsf{dev}^+ \geq \epsilon\mu) \leq k \cdot e^{-\mu\frac{\epsilon^2}{3k}} \tag{4}$$

$$\Pr(\mathsf{dev}^- \geq \epsilon\mu) \leq k \cdot e^{-\mu\frac{\epsilon^2}{2k}} \tag{5}$$

Clearly, when $k = 1$ (a soft block is in the deepest level of $\mathcal{H}$), the above bounds coincide with the bounds of Theorem 1. Appendix E contains the proof for the general case, which is based on union bounds and the bounds provided by Theorem 1.

4.3 More Restricted Structures: Stronger Bounds

In this section, we show that it is possible to design Operation $\mathcal{X}$-based lotteries with stronger guarantees when the constraint structure is more specific. We use school choice as a motivating example to describe this setting, but readers can easily see that the framework can be extended to other two-sided markets. Consider a school-choice setting where students are to be assigned to schools. In the language of our model, suppose N represents the set of students, and O represents the set of schools. We are given a partitioned structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$, where $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$. We suppose $\mathcal{H}_1$ contains all of the row blocks (to ensure that every

[13] This model of 'local' structures, which is a special case of our model, has been studied in [Khuller et al., 2006] as well.

student will be assigned to a school), $\mathcal{H}_2$ is a hierarchical constraint structure, and $\mathcal{S}$ is a local constraint structure that is in the deepest level of $\mathcal{H}$.[14]

We first define a slightly different notion of approximate implementation.

Definition 2. *We say that a partitioned constraint structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$ is* **Approximately Implementable with Additive Error k**, *if all conditions of Definition 1 are satisfied with the difference that for the Property 3 (the approximate satisfaction of soft constraints), the new requirement is that for any soft block $B \in \mathcal{S}$ with $\sum_{e\in B} x_e = \mu$, we have*

$$|\sum_{e\in B} X_e - \mu| \leq k.$$

We say two students have the same **type** if whenever one of them is in a constraint in $\mathcal{H}_2 \cup \mathcal{S}$, the other one is also in the same constraint. Denote the set of all types by $\mathcal{T} = \{1, \ldots, T\}$. Our main result in this section states that that any feasible fractional assignment is approximately implementable with additive error T, and the proof, as in the main theorem, is based on Operation $\mathcal{X}$ (see Appendix B for the proof).

Theorem 2. *If $\mathcal{H}$ is the set of all capacity blocks and there are T student types, then any feasible fractional assignment x is approximately implementable with additive error T.*

For example, consider a school choice setting where schools have quotas for low (versus high) socioeconomic status, inside (versus outside) walk-zone , and female (versus male) students. Then, there are 8 different types of students.[15] Now, if a school has a minimum quota of 500 for students who live inside the walk-zone, our result guarantees that at least $500 - 8 = 492$ such students would be admitted.

4.4 On Probabilistic Bounds

It is straightforward to check that the bounds in Property 3 are practically appealing only if μ is not very small. For example, to guarantee a less than 0.05 probability for the violation of a constraint, we need to have $\mu\epsilon^2 \simeq 9$, and so for $\epsilon = 10\%$, we need $\mu \simeq 900$. This "large market" requirement is clearly the case for many real-world markets. For instance, in the allocation of impressions to advertisers in Facebook [Hou et al., 2016], the numbers are in the order of thousands or millions.[16] Or in the allocation of sports tournament tickets

[14] The latter assumption is in the spirit of Theorem 1, and can be relaxed in exchange for weaker guarantees on violation of the soft constraints).

[15] For instance, "low socioeconomic, outside walk-zone, male" is one type, and there are 8 different such combinations.

[16] With $\mu = 10^6$, the chance of a 1 percent violation would be at most 2^{-20}.

through lottery, constraint limits are often very large. That said, this is not the case for many other applications of interest, and below we argue that our implementability result is still useful even when μ is small.

First, and from a theoretical standpoint, in Appendix H, we prove that the probabilistic bounds are tight: Unless the structure of constraints is more limited than what we identify, it is impossible to find a better implementation method than what we design.

Second, as we proved in Theorem 2, our matching algorithm could perform much better when there is more structure imposed on constraints.

Moreover, by simulating our matching algorithm on a school-choice example with several intersecting constraints, we can show that the *average* performance of our matching algorithm is better than what we can theoretically prove as a worst-case bound. (See Appendix C) For a goal to admit 500 students from a specific walk-zone, for instance, our theoretical worst-case bounds guarantee that the probability of a 10% violation is no more than 0.19. Nevertheless, our simulations show that the empirical probability of a 10% error is less than 0.012.

Why the average performance of our matching algorithm is better than the worst-case? One intuition comes from the proof of the main theorem: We first prove that the random variables of each constraint block are *negatively correlated*. Then, since negative correlation is stronger than *independence*, we apply standard concentration bounds of independent random variables to prove the bounds. Although in general negative correlation does not provide stronger theoretical bounds (recall our tightness result), we expect our algorithm to perform better in practice because of it.

Our small, additive theoretical bounds for a simplified constraint structure, as well as our computational experiments, clarify that for specific applications, one can potentially provide stronger bounds. However, this paper does not aim to solve a specific problem instance with the tightest possible bound. Rather, our intention is to break the theoretical barrier of implementing intersecting constraints, which we do by designing a general framework that models them as "goals"; this, in our conception, is a more realistic way to think about many of such constraints.[17]

[17]As a note related to the literature, the relatively new "approximation paradigm" of the theory of market design typically consists of "limit results", without explicit error bounds. (see, for instance, [Immorlica and Mahdian."Marriage, 2005, Kojima and Parag, 2009]) Our theorem, which guarantees that the probability of violating constraints is *exponentially* decreasing in μ, implies a new limit result with explicit convergence rates that are interesting in their own rights.

4.5 Why Simple Implementation Approaches Fail?

We can now go back and ask a simple but important question: Why simple lotteries will fail to satisfy our desired properties? For instance, what is wrong with a simple lottery like this: Set $X_e = 1$ with probability x_e and $X_e = 0$ with probability $1 - x_e$, independently for all edges $e \in E$. This simple lottery will satisfy the soft constraints in the sense of Property 3 in Definition 1, but it is easy to see that it does not satisfy the hard constraints (Property 2 in Definition 1) because we are essentially ignoring the constraints.

We can add one level of sophistication to this naïve lottery: Order the fractional elements in some arbitrary order, namely $e_1, \ldots, e_m$ where $m = |N \times O|$, and visit them one by one. When visiting e_i: If there exists $b \in \{0, 1\}$ such that $X_e = b$ contradicts with the satisfaction of hard constraints, then let $X_e = 1 - b$. Otherwise, set $X_e = 1$ with probability x_e and $X_e = 0$ with probability $1 - x_e$, independently.

This lottery, however, has a fundamental issue: It does not satisfy the Assignment Preservation (Property 1 in Definition 1). Suppose $|N| = |O| = n$. Let $x_e = 1/n$ for all $e \in E$, and let the set of hard constraints be the row and column constraints which ensure that each of the rows and columns of X sum up to 1. Suppose that the first n edges that the lottery visits are $e_1, \ldots, e_n$, with $e_i = (1, i)$. The chance that the lottery sets $X_{e_n} = 1$ is $(1 - 1/n)^{n-1}$, which approaches $1/e$ as $n \to \infty$. But $1/e$ is much larger than $x_{e_n} = 1/n$, so this lottery is not implementing the original fractional assignment.

One can make the above lottery more sophisticated: If the ordering of the edges is randomly chosen, then at least in the above example, Assignment Preservation will be satisfied. This, however, is only a consequence of the symmetry in the fractional solution that we started with. In the Appendix I, we show that even when the edges are visited in random order, Assignment Preservation will not always be satisfied.

For a fixed problem structure, it may be possible to carve out a lottery specific to that structure to satisfy all desired properties. But this is exactly why naïve approaches often fail, and designing a lottery that works for generic constraint structures is a nontrivial problem.

5 Application I: Improving Two Classic Mechanisms

In practice, the indivisibility of the objects and (possibly) the lack of monetary transfers make the allocation of resources to be asymmetric and unfair. One of the main motivations for randomization is to restore fairness by constructing an ex ante fair allocation. Nevertheless, given a fair fractional allocation, there could be very large discrepancies in realized utilities.

Our next family of applications concerns ex post properties of randomized mechanisms. We first employ our proposed framework and guarantee that the mechanism approximately maintains the fairness and efficiency properties of the original fractional assignment. Then, we exploit those guarantees to refine the random serial dictatorship (RSD) mechanism, and the pseudo-market mechanism (introduced in HZ and BCKM). Our expansions help to manage a richer set of constraints, and to provide ex post guarantees for the utilities.

The following definition is the notion we use for our approximate guarantees on utilities and welfare.

Definition 3. *A random variable x is* approximately lower-bounded *by a constant μ (denoted by $\mu \lesssim x$) if the following two conditions hold:*

1. $\mathbb{E}(x) = \mu$
2. $\Pr\Big(x \leq \mu(1-\epsilon)\Big) \leq e^{-\mu\epsilon^2/2}$

In words, a random variable x is approximately lower-bounded by a constant μ if x is equal to μ in expectation and the probability of x being less than $\mu(1-\epsilon)$ is very small, for any $\epsilon > 0$.[18]

Consider an environment where the set of hard blocks $\mathcal{H}$ forms a single hierarchy[19]. We impose no restriction on the soft structure $\mathcal{S}$. We assume that utilities are Von Neumann-Morgenstern utilities and are additive; that is, there exist values $(u_{ik})_{k\in O}$ such that an agent i's utility from an allocation x, with i'th row $x_i = (x_{i1}, x_{i2}, \cdots, x_{i|O|})$, is $v_{ix} = \sum_{k=1}^{|O|} x_{ik}u_{ik}$, where, without loss of generality, $u_{ik} \in [0,1]$ for all i, k. Also, let $W(x) = \sum_{i=1}^{|N|} v_{ix}$ be the *social welfare* associated with allocation x. In the following theorem, we guarantee that the ex post utility of any agent i and the ex post social welfare are approximately lower-bounded by v_{ix} and $W(x)$, respectively.

Theorem 3 (Utility and Welfare Bounds)**.** *Any feasible fractional assignment x is approximately implementable in such a way that for each i, if X is the outcome of the lottery, then $v_{ix} \lesssim v_{iX}$ and $W(x) \lesssim W(X)$.*

Proof. The idea of the proof is to add the following artificial constraints for the social welfare and for the utility of agents to the soft constraint set:

$$\sum_{k=1}^{|O|} X_{ik}u_{ik} \geq v_{ix} \quad \forall i \in N,$$

[18]It is clear that if $\mu = 0$, then any random variable for which $\mathbb{E}(x) = 0$ is approximately lower-bounded by 0. As will be clear soon, this definition is particularly interesting for larger values of μ.

[19]This assumption is for expositional clarity. In fact, it is enough if every all-row blocks are in the deepest level of $\mathcal{H}$.

$$\sum_{i=1}^{|N|}\sum_{k=1}^{|O|} x_{ik}u_{ik} \geq W(x).$$

Since hard blocks form a single hierarchy, the blocks associated with these new constraints are in the deepest level of the empty hierarchy of the hard structure. The proof follows immediately from Theorem 1. ∎

Remark 1. Theorem 3 provides lower bounds that are interesting when v_{ix} is relatively large, which is the case when each agent is (in expectation) allocated to several objects (since u_{ik}'s are normalized to be in $[0, 1]$). Therefore, in settings such as school choice, our bounds are not practically interesting for providing fairness among students. In fact, it is clear that because each student is assigned to a single school, guaranteeing an envy-free ex post allocation is nearly impossible. Our bounds give strong ex post guarantees for when agents (objects) are assigned to a large number of objects (agents). For example, in the mentioned school-choice setting, our bounds provide ex post guarantees for schools or for social welfare. Note that we can define the "utility of the schools" similar to that of students; that is, $v_{xj} = \sum_{k=1}^{|N|} x_{kj}u_{kj}$ is the utility of object j from assignment x, where $(u_{kj})_{k\in N}$ is the value of agent k for object j. In addition, since $W(x)$ is the sum of all utilities of the agents and thus often has a large value relative to individual agents' utilities, the bound provided in Theorem 3 for $W(X)$ is a strong probabilistic bound.

5.1 Fixing Random Serial Dictatorship

In this section we use the results from the previous section to fix the random serial dictatorship (with multi-unit demand) mechanism by making it approximately ex post fair.

Random serial dictatorship is one of the most popular mechanisms for the allocation of indivisible objects. RSD works as follows: The planner draws an ordering of agents uniformly at random and then lets the agents select their favorite bundle of objects (among those remaining without violating the constraints) one by one according to the realized random ordering. In Section 1.2, we discussed that although this mechanism is strategy-proof and ex ante fair[20], it is ex ante inefficient and ex post (very) unfair. Che and Kojima (2010) [Che and Kojima, 2010] shows that the ex ante inefficiency disappears in large markets. ex post unfairness, however, remains a serious issue for the RSD mechanism since agents with best priorities can choose most favorite items. The following example clarifies this problem.

[20] An allocation mechanism respects *equal treatment of equals* if agents with the same utilities over bundles of objects have the same allocations. The RSD mechanism satisfies the 'equal treatment of equals' and the 'SD envy-freeness' criteria.

Example 2. *Consider a course allocation setting, where there are two students, s_1 and s_2 each planning to take two coerces. Suppose there are four different courses, c_1, c_2, c_3 and c_4, each with capacity of* 1*. Let us assume that both students prefer c_1 and c_2 the most.*

Now if we run the RSD mechanism and choose one of the two random orderings with equal probability. This mechanism is obviously ex ante fair, in the sense that it is treating students in a symmetric fashion. Yet, the student with the best priority will take c_1 and c_2 and the other student has no choice but to take c_3 and c_4, which is ex post very unfair.

Consider the same model as in Section 5 in which agents have additive utilities over all subsets of objects and suppose all constraints' lower quotas are set to zero. For any $k \in \{1, 2, \cdots, N!\}$, let π_k be a priority ordering of agents. We introduce a new mechanism, the *Approximate Random Serial Dictatorship* (ARSD) mechanism, and prove that this mechanism is strategy-proof, ex ante fair, and ex post approximately fair. The idea is simple: the RSD mechanism induces an ex ante assignment, which is potentially fractional. We ask for agents' preferences, construct the expected assignment induced by the RSD, and then employ our randomized mechanism based on the Operation $\mathcal{X}$ to implement it. We formally define ARSD below.

The Approximate Random Serial Dictatorship Mechanism (ARSD)

1. Agents report their ordinal preferences over individual objects.
2. Construct the *expected random serial dictatorship assignment* x_{rsd} in the following way: run the serial dictatorship mechanism, with prioritizing agents according to π_k, and without violating any of the (hard and soft) constraints. Denote the resulting pure assignment by X_k. Let $x_{rsd} = \frac{1}{N!}\sum_1^{N!} X_k$.
3. The mechanism approximately implements x_{rsd}.

The following theorem shows that ARSD is strategy-proof, and in contrast to the standard RSD, the realized utilities of the agents is approximately equal to their expected utilities.

Theorem 4. *The ARSD mechanism is strategy-proof and respects equal treatment of equals. Moreover, ex post utilities of the agents are approximately lower-bounded by their ex ante utilities.*

Proof. Since x_{rsd} is the fractional assignment induced by the RSD and this mechanism is strategy-proof, a straightforward argument based on the revelation principle clarifies that ARSD is also strategy-proof since it implements x_{rsd}. Note that, in expectation, we are *exactly* implementing x_{rsd} and there are no approximations in this step. The second part of the theorem, that ex post utilities of the agents are approximately lower-bounded by their ex ante utilities, follows immediately from Theorem 3. ∎

5.2 Approximate Pseudo-Market Mechanism

A remarkable idea for designing ex ante efficient fractional assignments was introduced in Hylland and Zeckhauser (1979) [Hylland and Zeckhauser, 1979]. Their idea is to allocate all agents with an equal amount of an artificial currency, then ask them to report their von Neumann-Morgenstern preferences and solve for the competitive equilibrium from equal incomes (CEEI) of this 'pseudo-market'. The resulting fractional assignment is ex ante efficient and envy-free by the properties of the competitive equilibrium allocation. BCKM generalized that framework to a many-to-many matching markets, where agents may have constraints that fit into a hierarchy and objects may only have capacity constraints. We generalize the pseudo-market idea to a richer class of constraints. In our framework, one-side of the market can have intersecting soft constraints, as well as a hierarchy of hard constraints.

Suppose that the structure of hard constraints $\mathcal{H}$ is partitioned into a family $\mathcal{H}_1$ of sets that includes all column blocks, and a family $\mathcal{H}_2$ of sets that includes all-row blocks, as well as sub-rows which form a hierarchy. In addition, we enrich the structure by letting soft constraints $\mathcal{S}$ include any sub-rows such that $\mathcal{S}$ is in the deepest level of $\mathcal{H}_2$. We also assume that all lower quotas are set to zero.

We assume that agents' utilities are additive, that is, there exists values $(u_{ik})_{k\in O}$ such that an agent i's utility from any bundle (fractional or pure) $x_i = [x_{ik}]_{k\in O}$, is $v_i(x_i) = \sum\limits_{k\in O} x_{ik}u_{ik}$. We assume, without loss of generality, that $u_{ik} \in [0, 1]$ for all i, k. We let $\mathcal{F}_i$ be the set of feasible bundles of agent i, i.e. $\mathcal{F}_i = \{x_i | x_i = [x_{ik}] \in \mathbb{R}^{|O|}, 0 \leq \sum\limits_{(i,k)\in S_i} x_{ik} \leq \bar{q}_{S_i}\ ,\ \forall S_i \in \mathcal{S}\cup\mathcal{H}\}$.

The approximate pseudo-market mechanism works as follows:

The Approximate Pseudo-Market Mechanism (ACEEI)

1. Agents report their cardinal object values, as well as their hard and soft constraints.

2. The mechanism computes a vector of nonnegative prices $p = [p_k]_{k\in O}$ and a fractional assignment $x = [x_i]_{i\in N}$ in a way such that if B is the artificial budget assigned to each agent, the following conditions hold:

   (a) $x_i \in \text{argmax}_{x_i\in\mathcal{F}_i}\{v_i(x_i) s.t. \sum\limits_{k\in O} p_k x_{ik} \leq B\}$, for all $i \in N$,

   (b) $\sum\limits_{i\in N} x_{ik} \leq q_k$, for all $k \in O$, and $\sum\limits_{i\in N} x_{ik} < q_k$ only if $p_k = 0$.

3. The mechanism approximately implements the fractional assignment x.

From the results of the BCKM, it is easy to show that a there exists an envy-free assignment x and a vector p such that (p, x) satisfies the conditions of the step 2 of the approximate pseudo-market mechanism. Our results also guarantee that x is approximately implementable.

We now define the notion of an approximately envy-free assignment. Recall that an assignment x is **envy-free** if $v_i(x_i) \geq v_i(x_j)$ for all $i, j \in N$.We say that an assignment X is **approximately envy-free** if there exists an envy-free assignment x such that $v_i(x_i) \lesssim v_i(X_i)$ and $v_i(x_j) \gtrsim v_i(X_j)$. Now we state the result of this section.

Theorem 5. *Let X be the (random) pure allocation of the approximate pseudo-market mechanism. Then, $W(x) \lesssim W(X)$ and X is approximately envy-free.*

The proof of the theorem follows from Theorem 3.

6 Application II: Existence of ϵ-Competitive Equilibrium with Indivisible Objects

In this section, we apply our framework to an allocation environment with indivisible objects and prove the existence of an ϵ-CE in large markets. Our proof of existence takes a new approach, which is based on Operations $\mathcal{X}$, and can potentially be employed in deriving other existence results. We prove the existence of ϵ-CE in an environment where each agent has a hierarchical set of constraints as part of her preferences. This can be applied in settings such as the online advertisement, where CE is the solution concept of interest. For instance, advertisers are allowed to target specific groups of Facebook users by specifying certain distributional constraints. Solving for the CE and using the CE prices for pricing ads (which is being considered by Facebook [Hou et al., 2016]) is an application of this result.[21]

Consider an environment with a group of n agents (buyers) and a set of objects, which are denoted by N, O respectively. A bundle is a subset of objects. Agents have preferences over bundles of objects. Objects are in unit supply.[22] For now, we assume that agents' utilities are fully additive, that is, there exists values $(u_{ik})_{k \in O}$ such that an agent i's utility from any bundle (fractional or pure) $x_i = [x_{ik}]_{k \in O}$, is $v_i(x_i) = \sum_{k \in O} x_{ik} u_{ik}$, where, w.l.o.g, $u_{ij} \in [0, 1]$ for

[21] The existence of equilibrium in large markets has been studied in several studies; e.g. see [?, ?] for Walrasian equilibrium in large markets, and [?] for equilibrium in large matching markets.

[22] This can easily be extended to a multi-unit supply by considering each "copy" of an object as an object.

all i, j. (Later, we will show how the same approach can incorporate hierarchical constraints as part of agents' preferences.) Each agent is endowed with an initial budget of $w_a \in \mathbb{R}^+$.

Definition 4. *For a price vector* $\mathbf{p}$*, the budget-set of an agent* a *is defined by*

$$B_a(\mathbf{p}) = \Big\{S : \sum_{o \in S} p_o \leq w_a\Big\}.$$

Definition 5. *For a price vector* $\mathbf{p}$*, the indirect utility function of agent* $a \in N$ *is defined by*

$$v_a(\mathbf{p}) = \max_{S \in O} \Big\{u_a(S) : S \in B_a(\mathbf{p})\Big\}.$$

Definition 6. *For a price vector* $\mathbf{p}$ *and an assignment* x *of objects to agents,* $(\mathbf{p}, x)$ *is called an* ϵ*-Competitive Equilibrium if:*

1. x *is a partition of items.*

2. *For all* $a \in N$, $x_a \in B_a(\mathbf{p})$.

3. $u_a(x) \geq v_a(\mathbf{p}) \cdot (1 - \epsilon)$.

We prove the existence of ϵ-CE in a large market regime that is formally defined below. Intuitively, we require each agents' utility from the set of *all* objects to grow as the market size grows.

The Large Market Assumption. Consider a sequence of markets $\mathcal{M}_1, \dots, \mathcal{M}_q, \dots$ where market $\mathcal{M}_q$ has O_q as the set of objects and $N_q = N_1$ as the set of agents.[23] We say that we are in the *large market regime* if, as $q \to \infty$, for all agents $a \in N$, $u_a(O_q) \to \infty$.

Note that the above large market regime is satisfied if, as $q \to \infty$, the set of agents A and their budgets remains constant (i.e. $A_q = A$ for all q), while $u_a(O) \to \infty$ for all $a \in A$. We now state the main theorem of this section.

Theorem 6. *For any fixed* $\epsilon > 0$*, there exists* q_0 *such that for all* $q > q_0$*, an* ϵ*-CE exists in* $\mathcal{M}_q$.

Proof Overview: Here we only present the idea behind the proof. The full proof can be found in Appendix F. To prove the existence of an ϵ-CE, we first prove the existence of a *fractional* ϵ-CE in which all objects' prices are "small"; that is, as market size grows, no agent spends a constant fraction of his endowments for a single object. Then, we implement

[23] One could allow for the set of agents to grow as well (with a rate slower than the set of objects). We analyze the constant set of agents for expositional simplicity.

the *fractional* ϵ-CE by our Operation $\mathcal{X}$-based lottery. By defining utilities of the agents as soft constraints, and because the probability of violating these constraints vanishes as the market size grows, we can employ the union bound to show that the probability that *all* "utility constraints" are violated by a factor more than ϵ is strictly less than 1 for sufficiently large markets. This guarantees that there is at least one realization in which none of the constraints are violated by a factor more than ϵ. ∎

Existence of ϵ-CE Under a Hierarchical Constraint Structure

Suppose each agent $a \in N$ has a hierarchical set of constraints $\mathcal{H}_a$, such that the corresponding lower quotas are equal to zero for all constraints in $\mathcal{H}_a$.[24] Let

$$u_a(S) = \max\{\sum_{o \in S'} u_{ao} : S' \subseteq S \text{ and } S' \text{ is feasible with respect to } \mathcal{H}_a\}.$$

In addition, we assume that $|\mathcal{H}_i| \leq \Delta$ for a fixed constant[25] Δ. Our large market assumption remains the same as before.

Theorem 7. *Suppose the constraint structure of each agent is a hierarchy with size no larger than Δ. Then, for any fixed $\epsilon > 0$, there exists q_0 such that for all $q > q_0$, an ϵ-CE exists in $\mathcal{M}_q$.*

The theorem is proved at the end of Appendix F.

7 Application III: Overlapping Constraints in School Choice

Consider a school choice setting, where n students are to be assigned to k schools. Several types of constraints naturally arise in this market. A few examples are: Capacity constraints of schools and students, walk-zone priorities, affirmative action policies[26], and grade-based

[24]This is the setting for the pseudo-market mechanism discussed in BCKM.

[25]We make this assumption for expositional simplicity; our proof could allow Δ to be a function of the market size that grows with a sufficiently slow rate

[26]Affirmative action is defined as *"positive steps taken to increase the representation of women and minorities in areas of employment, education, and culture from which they have been historically excluded."*[SEP, 2013] One goal of such policies is to increase diversity, and balancing out the social effects that weaken specific groups. Another argument in favor of affirmative action policies is that they increase structural integration, which *"serves the ideal of equal opportunity."*[Jacobs, 2004] Affirmative action policies are usually implemented as minimum quotas on students within a minority group. See [Hafalir et al., 2013, Abdulkadiroglu and Sonmez, 2003, Kominers and Sonmez, 2013] for detailed theoretical analysis of affirmative action policies.

quotas.[27]. The bihierarchy assumption often fails when multiple constraints such as these exist. For example, if a school has minimum quotas on both low socioeconomic status students and walk-zone priority students, the blocks associated with these two constraints overlap. Therefore, the bihierarchy assumption fails and it would not be possible to model all of these constraints as hard constraints. In the other hand, considering some of them as soft constraints makes our results are applicable.

In the next example, we discuss an alternative approach for handling walk-zone priorities in school-choice. A typical way to implement walk-zone priorities is partitioning the city into artificial zones and imposing lower quotas on the number of students who live in the same zone as schools. This method treats students who live just inside and outside of each zone's border very differently, as it assigns a weight 1 to students who live inside and a weight 0 to students who live outside.

A more natural way to include distance-based priorities in the school choice problem is to assign weights to each student-school pair (s, h) based on student s's distance to school h. More formally, one can impose $\sum_{(s,h)\in B} d_{(s,h)} x_{(s,h)} \leq \bar{q}_B$ as a soft constraint, where $d_{(s,h)}$ is either the distance of student s from school h, or any other "penalty function".[28] It is very straightforward to see that this can be accommodated into our framework, as soft constraints can have real-valued coefficients.

8 Conclusion

We study the mechanism design problem of allocating indivisible objects to agents in a setting where cash transfers are precluded and the final allocation needs to satisfy some constraints. One efficient and *ex ante* fair solution to this problem is the "expected assignment" method, in which the mechanism first finds a feasible fractional assignment, and then implements that fractional assignment by running a lottery over feasible pure assignment. Previous literature have characterized a maximal 'constraint structure' that can be accommodated into the expected assignment method. Such structure rules out many real-world applications. We show that by reconceptualizing the role of constraints and treating some of them as *goals* rather than hard constraints, one can accommodate many more constraints.

The main theorem of the paper identifies a rich constraint structure that is approximately implementable, meaning that any expected assignment that satisfies both hard constraints and soft constraints (i.e. goals) can be implemented by a lottery over nearby pure assignments

[27]Schools may have grade-based diversity policies. For instance, New York City's Educational Option program has quotas based on test scores; see [Abdulkadiroglu et al., 2005]

[28]For example, $d_{(s,h)}$ can incorporate considerations such as accessibility by public transportation.

in a way such that hard constraints can be exactly satisfied and goals can be satisfied with only very small errors. We also show that our bounds on approximate satisfaction of goals are tight, but if we restrict the structure of constraints more, then we can prove smaller, additive bounds.

The main technical contribution of this study is the randomized mechanism that we design in order to implement a fractional assignment. We quantify the violations in soft constraints by applying probabilistic concentration bounds. This framework helps us to preserve some desirable properties of the expected allocation in the *ex post* allocation. For instance, an envy-free or efficient expected allocation remains approximately envy-free and efficient *ex post*. We exploit the same technique to modify the random serial dictatorship mechanism (by making it *ex post* approximately fair) and the pseudo-market mechanism (by expanding the structure of constraints that can be accommodated into it). Our mathematical technology also helps us to prove the existence of an ϵ-equilibrium in an indivisible items setting with constraints.

We are hopeful that the proposed framework for partitioning constraints into hard constraints and soft goals, and the randomized mechanism we designed will pave the way for designing improved allocation mechanisms in practice.

References

[SEP, 2013] (2013). http://plato.stanford.edu/entries/affirmative-action/.

[wik, 2016] (July 2016). https://en.wikipedia.org/wiki/Binomial_coefficient#Bounds_and_asymptotic_formulas.

[Abdulkadiroglu et al., 2005] Abdulkadiroglu, A., Pathak, P. A., and Roth, A. E. (2005). The new york city high school match. *American Economic Review :*, pages 364–367.

[Abdulkadiroglu and Sonmez, 1998] Abdulkadiroglu, A. and Sonmez, T. (1998). Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66.3:689–701.

[Abdulkadiroglu and Sonmez, 2003] Abdulkadiroglu, A. and Sonmez, T. (2003). School choice: A mechanism design approach. *American economic review :*, pages 729–747.

[Ageev and Sviridenko, 2004] Ageev, A. A. and Sviridenko, M. I. (2004). Pipage rounding: a new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328.

[Alon et al., 2015] Alon, N., Bronfman, S., Hassidim, A., and Romm, A. (2015). Redesigning the israeli medical internship match.

[Auger and Doerr, 2011] Auger, A. and Doerr, B. (2011). Theory of randomized search heuristics foundations and recent developments.

[Birkhoff, 1946] Birkhoff, G. (1946). Three observations on linear algebra. *Revi. Univ. Nac. Tucuman, ser A*, 5:147–151.

[Bogomolnaia and Moulin, 2001] Bogomolnaia, A. and Moulin, H. (2001). A new solution to the random assignment problem. *Journal of Economic Theory*, 100.2:295–328.

[Border, 1991] Border, K. C. (1991). Implementation of reduced form auctions: A geometric approach. *Econometrica*, 59.4:1175–1187.

[Budish, 2011] Budish, E. (2011). The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119.6:1061–1103.

[Budish et al., 2016] Budish, E., Cachon, G. P., Kessler, J. B., and Othman, A. (2016). Course match: A large-scale implementation of approximate competitive equilibrium from equal incomes for combinatorial allocation. *Operations Research*.

[Budish et al., 2013] Budish, E., Che, Y.-K., Kojima, F., and Milgrom, P. (2013). Designing Random Allocation Mechanisms: Theory and Applications. American economic review. 103(2):585–623.

[Che et al., 2013] Che, Y.-K., Kim, J., and Mierendorff, K. (2013). Generalized reduced-form auctions: A network-flow approach. *Econometrica*, 81.6:2487–2520.

[Che and Kojima, 2010] Che, Y.-K. and Kojima, F. (2010). Asymptotic equivalence of probabilistic serial and random priority mechanisms. *Econometrica*, 78.5:1625–1672.

[Chekuri et al., 2010] Chekuri, C., Vondrak, J., and Zenklusen, R. (2010). Dependent randomized rounding via exchange properties of combinatorial structures. *Foundations of Computer Science (FOCS)*, 2010:51.

[Chen and Sonmez, 2002] Chen, Y. and Sonmez, T. (2002). Improving efficiency of on-campus housing: an experimental study. *The American Economic Review*, 92.5:1669–1686.

[Edmonds, 2003] Edmonds, J. (2003). *Submodular Functions, Matroids, and Certain Polyhedra*, pages 11–26. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Ehlers et al., 2014] Ehlers, L., Hafalir, I. E., Yenmez, M. B., and Yildirimy., M. A. (2014). School choice with controlled choice constraints: Hard bounds versus soft bounds. *Journal of Economic Theory*, 153:648–683.

[Gandhi et al., 2006] Gandhi, R., Khuller, S., Parthasarathy, S., and Srinivasan, A. (2006). Dependent rounding and its applications to approximation algorithms. *Journal of the ACM (JACM)*, 53(3):324–360.

[Hafalir et al., 2013] Hafalir, I. E., Yenmez, M. B., and Yildirim, M. A. (2013). Effective affirmative action in school choice. *Theoretical Economics*, 8.2:325–363.

[Hashimoto, 2016] Hashimoto, T. (2016). The generalized random priority mechanism with budgets.

[Hatfield, 2009] Hatfield, J. W. (2009). Strategy-proof, efficient, and nonbossy quota allocations. *Social Choice and Welfare*, 33.3:505–515.

[Hou et al., 2016] Hou, M., Sodomka, E., and Stier-Moses, N. (2016). Game abstractions for counterfactual prediction in online markets. *Mimeo*.

[Hylland and Zeckhauser, 1979] Hylland, A. and Zeckhauser, R. (1979). The efficient allocation of individuals to positions. *The Journal of Political Economy :*, pages 293–314.

[Immorlica and Mahdian."Marriage, 2005] Immorlica, N. and Mahdian."Marriage, M. (2005). *honesty, and stability." Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics.

[Jacobs, 2004] Jacobs, L. A. (2004). *Pursuing equal opportunities: the theory and practice of egalitarian justice*. Cambridge University Press.

[Kesten et al., 2015] Kesten, O., Kurino, M., and Nesterov, A. S. (2015). Efficient lottery design. *Social Choice and Welfare*, pages 1–27.

[Khuller et al., 2006] Khuller, S., Parthasarathy, S., and Srinivasan, A. (2006). Dependent rounding and its applications to approximation algorithms. *Journal of the ACM*, 53:324–360.

[Kojima, 2009] Kojima, F. (2009). Random assignment of multiple indivisible objects. *Mathematical Social Sciences*, 57.1:134–142.

[Kojima and Manea, 2010] Kojima, F. and Manea, M. (2010). Incentives in the probabilistic serial mechanism. *Journal of Economic Theory*, 145.1:106–123.

[Kojima and Parag, 2009] Kojima, F. and Parag, A. (2009). *Pathak."Incentives and stability in large two-sided matching markets." The American Economic Review :*, pages 608–627.

[Kominers and Sonmez, 2013] Kominers, S. D. and Sonmez, T. (2013). *Designing for diversity in matching*. EC.

[Lau et al., 2011] Lau, L. C., Ravi, R., and Singh, M. (2011). *Iterative methods in combinatorial optimization*, volume 46. Cambridge University Press.

[Liu and Pycia, 2016] Liu, Q. and Pycia, M. (2016). Ordinal efficiency, fairness, and incentives in large markets.

[Manea, 2009] Manea, M. (2009). Asymptotic ordinal inefficiency of random serial dictatorship. *Theoretical Economics*, 4.2:165–197.

[Matthews, 1984] Matthews, S. (1984). *A."On the implementability of reduced form auctions." Econometrica: Journal of the Econometric Society :*, pages 1519–1522.

[Neumann, 1953] Neumann, V. (1953). *John. "A certain zero-sum two-person game equivalent to the optimal assignment problem." Contributions to the Theory of Games*, 2:5–12.

[Nguyen et al., 2014] Nguyen, T., Peivandi, A., and Vohra, R. (2014). One-sided matching with limited complementarities.

[Nguyen and Vohra, 2015] Nguyen, T. and Vohra, R. (2015). Near feasible stable matchings with complementarities.

[Pathak and Sethuraman, 2011] Pathak, P. A. and Sethuraman, J. (2011). Lotteries in student assignment: An equivalence result. *Theoretical Economics*, 6(1):1–17.

[Pycia and Ünver, 2015] Pycia, M. and Ünver, M. U. (2015). Decomposing random mechanisms. *Journal of Mathematical Economics*, 61:21–33.

[Rubinstein, 2014] Rubinstein, A. (2014). Inapproximability of Nash equilibrium. preprint, arXiv.

Appendices

A Implementation: A Random Mechanism

In this section, we present the complete proof of Theorem 1. As discussed in the proof overview of the theorem, the proof is constructive. We will propose an implementation mechanism (or, equivalently, a lottery) which approximately implements a partitioned structure that satisfies the properties described in Theorem 1.

To describe the main idea of our mechanism, we need to introduce the notion of *tight* and *floating* constraints: A constraint is tight if it is binding. This notion is precisely defined in the following definition. First, for any block B, let $x(B) = \sum_{e \in B} x_e$.

Definition. *A constraint $S = (B, \underline{q}_B, \bar{q}_B)$ is* tight *if, either $x(B) = \underline{q}_B$ or $x(B) = \bar{q}_B$; otherwise, S is* floating*. Similarly, we say that a block B is tight when the constraint corresponding to it (S in here) is tight.*

Note that this definition naturally applies on the (implicit) constraints that for all $e \in E$, we must have that $0 \leq x_e \leq 1$.

In the core of our randomized mechanism is a stochastic operation that we call **Operation $\boldsymbol{\mathcal{X}}$**. We iteratively apply Operation $\mathcal{X}$ to the initial fractional assignment. In each iteration t, the fractional assignment x_t is converted to x_{t+1} in a way such that:(1) the number of floating constraints decreases, (2) $\mathbb{E}(x_{t+1}|x_t) = x_t$, and (3) x_{t+1} is feasible with respect to $\mathcal{H}$. The first property guarantees that after a finite (and small) number of iterations[29], the obtained assignment is pure. The second property makes sure that the resulting pure assignment is equal to the original fractional assignment *in expectation*. The third property guarantees that all hard constraints are satisfied throughout the whole process of the mechanism. As the last step, we need to show that by iteratively applying of Operation $\mathcal{X}$, soft constraints are approximately satisfied. This is a more technical property of Operation $\mathcal{X}$, which we discuss in Appendix A.4. Roughly speaking, we design Operation $\mathcal{X}$ in a way such that it never increases (or decreases) two (or more) elements of a soft constraint at the same iteration. Consequently, elements of each soft block become "negatively correlated". It then allows us to employ Chernoff concentration bounds to prove that soft constraints are approximately satisfied.

In the rest of this section, we design Operation $\mathcal{X}$ and prove that it satisfies the above-mentioned properties.

[29] Our randomized mechanism stops after at most $|\mathcal{H}| + |E|$ iterations.

A.1 Definitions

In this section, we introduce the required notions for defining Operation $\mathcal{X}$. Given a feasible fractional assignment x, we define the following notions:

1. For any two links e, e', a block B is *separating* e, e' if B contains exactly one of them.

2. A block is *tight* if $\sum_{e \in B} x_e$ is equal to either the upper or the lower quota of the constraint corresponding to that block.

3. Given a hierarchy $\mathcal{H}$, a (hard) block $B \in \mathcal{H}$ is *supporting* a pair of links (e, e') if it is the smallest block (in the number of involved edges) that contains both e, e', and moreover, no tight block in $\mathcal{H}$ separates e, e'.

4. We say that a hierarchy $\mathcal{H}$ is supporting the pair (e, e') if there exists a block in $\mathcal{H}$ which supports (e, e'). In particular, if the subset $\{e, e'\}$ is in the deepest level of $\mathcal{H}$, then (e, e') is supported by $\mathcal{H}$.

5. A *floating cycle* is a sequence $e_1, \ldots, e_l$ of distinct edges such that:

   - x_{e_i} is non-integral for all integers i.
   - (e_i, e_{i+1}) is supported by $\mathcal{H}_1$ for even integers i.
   - (e_i, e_{i+1}) is supported by $\mathcal{H}_2$ for odd integers i.

   where *length* the of cycle, l, is an even number and $i + 1 = 1$ for $i = l$. Figure A.1 represents a floating cycle of length 6. A floating cycle is said to be *minimal* if it does not contain a smaller floating cycle as a subset. We often drop the minimal phrase and whenever we say a *floating cycle*, we refer to a minimal floating cycle, unless otherwise specified.

Next, we define the notion of *floating paths*; loosely speaking, their structure is very similar to floating cycles, except in their endpoints. Floating paths start from a hierarchy and end in the same hierarchy if their length is even, otherwise, they end in the other hierarchy.

5. A *floating path* is a sequence $e_1, e_1, \ldots, e_l$ of distinct edges such that:

   - x_{e_i} is non-integral for all integers i.
   - There exists $a \in \{1, 2\}$ such that if we define $\overline{a} = \{1, 2\} \backslash \{a\}$, then:
     - (e_i, e_{i+1}) is supported by $\mathcal{H}_a$ for even integers $i < l$.

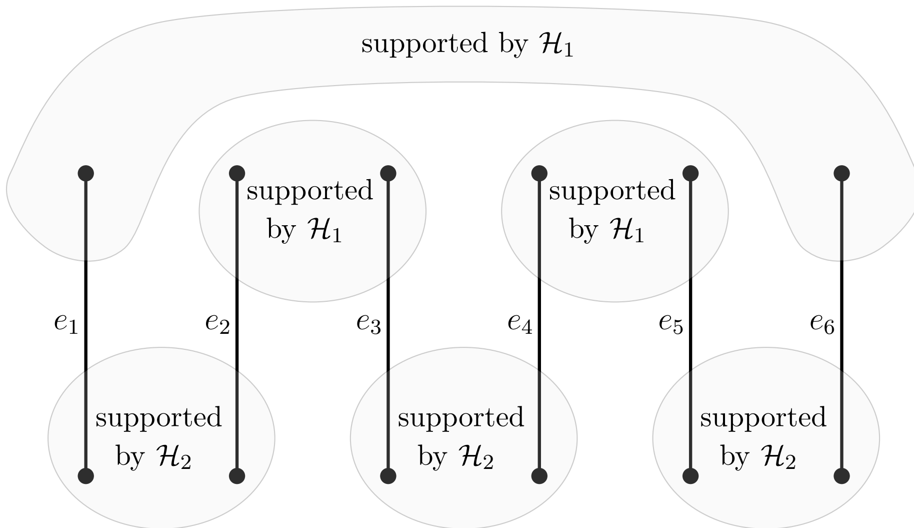

Figure A.1: A floating cycle of length 6

  - (e_i, e_{i+1}) is supported by $\mathcal{H}_{\overline{a}}$ for odd integers $i < l$.

- No tight block in $\mathcal{H}_a$ contains e_1, and no tight block in $\mathcal{H}_b$ contains e_l where $b = a$ if l is even and $b = \overline{a}$ if l is odd.

Figure A.2 contains a visual example of a floating path. A floating path is said to be *minimal* if it does not contain a smaller floating path as a subset. Whenever we say a *floating path*, we refer to a minimal floating path, unless otherwise specified.

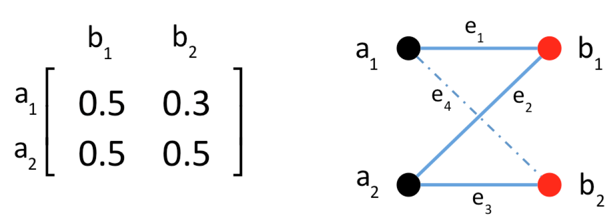

Figure A.2: Example of a floating path: Suppose that in the above fractional assignment $\mathcal{H}_1$ is the set of row blocks and $\mathcal{H}_2$ is the set of column blocks. Also, suppose the lower quotas and upper quotas are set to 0 and 1, respectively. Then, e_1, e_2, e_3 is a (minimal) floating path. However, e_1, e_4, e_3 is *not* a floating path.

Finally, we introduce the following crucial concept.

Definition. *Assume we are given a fractional assignment x. For any block B and any $\epsilon > 0$, let $x \uparrow_\epsilon B$ denote a new (fractional) assignment in which the element of the matrix corresponding to edge e is increased by ϵ if $e \in B$ (i.e. it changes to $x_e + \epsilon$), and it remains* *unchanged otherwise. Similarly, let $x \downarrow_\epsilon B$ denote the fractional assignment in which the element of the matrix corresponding to edge e is decreased by ϵ if $e \in B$ (i.e. it changes to $x_e - \epsilon$), and it remains unchanged otherwise.*

Example 3. *$(x \uparrow_\epsilon B) \downarrow_\epsilon B'$ denotes the fractional assignment in which the value of any edge $e \in B - B'$ becomes $x_e + \epsilon$, the value of any edge $e \in B' - B$ becomes $x_e - \epsilon$, and the value of the rest of the edges does not change.*

A.2 Operation $\mathcal{X}$

Operation $\mathcal{X}$ can be applied on a given floating cycle or a floating path of a fractional assignment x (if none of them exist, then the assignment must be pure by Lemma 4). We first define this operation for a given floating cycle. Let $F = \langle e_1, \ldots, e_l \rangle$ be a floating cycle in x. Define

$$F_o = \{e_i : i \text{ is odd}\},$$
$$F_e = \{e_i : i \text{ is even}\}.$$

We call the pair (F_o, F_e) the *odd-even decomposition* of F. Given two non-negative reals ϵ, ϵ' (which we describe how to set soon), Operation $\mathcal{X}$ generates an assignment $x' \in \mathbb{R}^{N \times O}$ in one of the following ways:

- $x' = (x \uparrow_\epsilon F_o) \downarrow_\epsilon F_e$ with probability $\frac{\epsilon'}{\epsilon + \epsilon'}$
- $x' = (x \downarrow_{\epsilon'} F_o) \uparrow_{\epsilon'} F_e$ with probability $\frac{\epsilon}{\epsilon + \epsilon'}$.

Both ϵ and ϵ' are chosen to be the largest possible numbers such that both of the assignments $(x \uparrow_\epsilon F_o) \downarrow_\epsilon F_e$ and $(x \downarrow_{\epsilon'} F_o) \uparrow_{\epsilon'} F_e$ remain feasible, in the sense that they satisfy all hard constraints.

The definition of Operation $\mathcal{X}$ on a floating path is the same as its definition on a floating cycle. To summarize, we give a formal definition of the Operation $\mathcal{X}$ below.

Definition 7. *Consider a fractional assignment x and a floating path or a floating cycle, namely F, given as the inputs to Operation $\mathcal{X}$. Then Operation $\mathcal{X}$ generates a new assignment x', where $x' = (x \uparrow_\epsilon F_o) \downarrow_\epsilon F_e$ with probability $\frac{\epsilon'}{\epsilon + \epsilon'}$ and $x' = (x \downarrow_{\epsilon'} F_o) \uparrow_{\epsilon'} F_e$ with probability $\frac{\epsilon}{\epsilon + \epsilon'}$, where ϵ, ϵ' are positive numbers chosen to be the largest possible numbers such that both $(x \uparrow_\epsilon F_o) \downarrow_\epsilon F_e$ and $(x \downarrow_{\epsilon'} F_o) \uparrow_{\epsilon'} F_e$ are feasible assignments.*

We also denote x' (which is a random variable) by $x \updownarrow F$.

A.3 The Implementation Mechanism

Our implementation mechanism which is based on Operation $\mathcal{X}$ is formally defined below.

The Implementation Mechanism Based on the Operation $\mathcal{X}$:

1. A fractional assignment x is reported to the mechanism.
2. Set i to 1 and let $x_i = x$.
3. Repeat the following as long as x_i contains a floating cycle or a floating path:
   (a) If x_i contains a floating cycle, let F be an arbitrary floating cycle, otherwise, let F be an arbitrary floating path.
   (b) Define x_{i+1} to be $x_i \updownarrow F$.
   (c) Increase i by one.
4. Report x_i as the outcome of the mechanism.

In the rest of this section, we show that the above mechanism approximately implements x in the sense of Definition 1.

The first step of the proof is verifying that if the assignment has no floating cycles or paths, then it is necessarily pure. We prove this claim in Claim 2. The next step of the proof is to show that Operation $\mathcal{X}$ is well-defined in the sense that both ϵ, ϵ' cannot be zero at the same time. We will state and prove this fact in Lemma 4. Next, we prove the following three important properties of Operation $\mathcal{X}$:

i. The outcome of Operation $\mathcal{X}$ satisfies the hard constraints.

ii. Operatoin $\mathcal{X}$ satisfies the martingale property, i.e.

$$\mathbb{E}\left[x \updownarrow F \,\middle|\, x\right] = x$$

iii. The outcome of the Operation $\mathcal{X}$ has more tight constraints (compared to x).

These properties are proved separately in three Lemmas below.

Lemma 1. *The outcome of Operation $\mathcal{X}$ satisfies the hard constraints.*

Proof. By definition, Operation $\mathcal{X}$ chooses ϵ, ϵ' such that both of its two possible outcomes are feasible with respect to $\mathcal{H}$. ■

Lemma 2. *Operatoin $\mathcal{X}$ satisfies the martingale property, i.e.*

$$\mathbb{E}\left[x \updownarrow F \;\middle|\; x\right] = x$$

Proof. We prove the lemma by verifying that this property holds for any entry (i, j) of the assignment matrix, i.e. if $(x \updownarrow F)_{(i,j)}$ denotes the (i, j)-th element of $x \updownarrow F$, then we have

$$\mathbb{E}\left[(x \updownarrow F)_{(i,j)} \;\middle|\; x\right] = x_{(i,j)}.$$

In simple words, we prove that operation $\mathcal{X}$ does not change the value of entry (i, j) of the assignment matrix in expectation.

Observe that by the definition of Operation $\mathcal{X}$

$$\mathbb{E}\left[x \updownarrow F \mid x\right] = \frac{\epsilon'}{\epsilon + \epsilon'} \cdot ((x \uparrow_\epsilon F_o) \downarrow_\epsilon F_e) + \frac{\epsilon}{\epsilon + \epsilon'} \cdot ((x \downarrow_{\epsilon'} F_o) \uparrow_{\epsilon'} F_e).$$

The claim is trivial if $(i, j) \notin F$. So, assume $(i, j) \in F$. Then, we either have $(i, j) \in F_o$ or $(i, j) \in F_e$:

1. If $(i, j) \in F_o$, then Operation $\mathcal{X}$ increases $x_{(i,j)}$ by ϵ with probability $\frac{\epsilon'}{\epsilon+\epsilon'}$ and decreases it by ϵ' with probability $\frac{\epsilon}{\epsilon+\epsilon'}$. In this case, the expected amount by which $x_{(i,j)}$ changes is equal to $\epsilon \cdot \frac{\epsilon'}{\epsilon+\epsilon'} - \epsilon' \cdot \frac{\epsilon}{\epsilon+\epsilon'} = 0$.

2. If $(i, j) \in F_e$, then Operation $\mathcal{X}$ decreases $x_{(i,j)}$ by ϵ with probability $\frac{\epsilon'}{\epsilon+\epsilon'}$, and increases it by ϵ' with probability $\frac{\epsilon}{\epsilon+\epsilon'}$. In this case, the expected amount by which $x_{(i,j)}$ changes is equal to $-\epsilon \cdot \frac{\epsilon'}{\epsilon+\epsilon'} + \epsilon' \cdot \frac{\epsilon}{\epsilon+\epsilon'} = 0$.

This proves the lemma. ■

Lemma 3. *The outcome of operation $\mathcal{X}$ has more tight constraints (compared to x).*

Proof. Suppose F is a floating cycle in x. The proof for the path case is almost identical. We show that $x \updownarrow F$ has more tight constraints than x. To do so, we first show that a tight constraint remains tight after Operations $\mathcal{X}$. Second, we show that at least one of the floating constraints in x becomes tight in $x \updownarrow F$.

To prove the first step, we show that for any tight constraint S, its corresponding block, B, contains an equal number of elements (edges) from the sets F_o and F_e. This fact is formally proved below.

Claim 1. *Suppose we are given a floating cycle F in the fractional assignment x, and let (F_o, F_e) be the odd-even decomposition of F. Then, any tight block (in x) contains an equal number of elements from F_o and F_e.*

Proof. Let $S = (B, \underline{q}_B, \bar{q}_B)$ be a tight constraint and w.l.o.g. assume $B \in \mathcal{H}_1$. Then, it must be that for any element $e_i \in B \cap F_e$, the element that comes right after e_i in F, i.e. e_{i+1}, belongs to B. This holds because by the definition of floating cycles, (e_i, e_{i+1}) is supported by $\mathcal{H}_1$, which means no tight block in $\mathcal{H}_1$ separates e_i, e_{i+1}. Consequently, both e_i and e_{i+1} belong to B, or else B itself would separate e_i, e_{i+1}.

Therefore, for any element $e_i \in B \cap F_e$, there exists a distinct element $e_{i+1} \in B \cap F_o$ which corresponds to e_i. Similarly, any element in $B \cap F_o$ corresponds to a distinct element in $B \cap F_e$. This proves the claim. ■

Now recall that whenever Operation $\mathcal{X}$ increases (decreases) the elements in F_o, it decreases (increases) the elements in F_e. This fact and Claim 1 together imply that $x(B) = (x \updownarrow F)(B)$ (regardless of the choice of ϵ, ϵ'). This ensures that any tight constraint remains tight after operation $\mathcal{X}$.

We now prove the second step, which is to show that at least one of the floating constraints in x becomes tight in $x \updownarrow F$. Observe that any floating constraint $S = (B, \underline{q}_B, \bar{q}_B)$ provides a positive *slack* for setting the values of ϵ, ϵ'. In simple words, since S is a floating constraint, we have that $\underline{q}_B < x(B) < \bar{q}_B$. By this fact, we can compute the positive upper bounds that S imposes on ϵ, ϵ'. Finally, taking the minimum of these upper bounds (over all floating constraints S) determines the values for ϵ, ϵ'. We formalize this argument below. Let

$$\begin{aligned}
\overline{s} =& \bar{q}_B - x(B)\,, \\
\underline{s} =& x(B) - \underline{q}_B\,, \\
k =& |F_o \cup B| - |F_e \cup B|\,.
\end{aligned}$$

Then, in order to guarantee that $x \updownarrow F$ satisfies constraint S, the following inequalities (that can be translated into upper bounds) are imposed on ϵ, ϵ' by Operation $\mathcal{X}$:

$$\begin{cases} \epsilon \cdot k \leq \overline{s} & \text{if } k \geq 0 \\ \epsilon \cdot |k| \leq \underline{s} & \text{if } k < 0 \end{cases} \tag{6}$$

$$\begin{cases} \epsilon' \cdot k \leq \underline{s} & \text{if } k \geq 0 \\ \epsilon' \cdot |k| \leq \overline{s} & \text{if } k < 0 \end{cases} \tag{7}$$

Now, let $u(S), u'(S)$ respectively denote the (positive) upper bounds imposed by Inequalities (6),(7) on ϵ, ϵ'. By definition of ϵ, ϵ', we have that $\epsilon = \min_S u(S)$ and $\epsilon' = \min_S u'(S)$ where the minimum is over all the floating constraints S. This argument implies that:

Claim 2. *Operation $\mathcal{X}$ chooses ϵ, ϵ' such that $\epsilon, \epsilon' > 0$.*

Proof. It is enough to show that $u(S), u'(S) > 0$ for all S. This is implied by noting that, given a floating constraint S, we have $\overline{s}, \underline{s} > 0$. ■

The above argument also implies the existence of a floating constraint S_1 for which one of the corresponding inequalities in (6) is tight. Similarly, there exists a floating constraint S_2 for which one of the corresponding inequalities in (7) is tight. These two facts imply that after operation $\mathcal{X}$, either S_1 or S_2 becomes a tight constraint.

To summarize, we first showed that if a constraint is tight, then it remains tight after operation $\mathcal{X}$. Moreover, we showed that there always exists at least one floating constraint which becomes tight after operation $\mathcal{X}$. Therefore, the number of tight constraints decreases, which proves the lemma. ■

Next, we show that if a fractional assignment contains neither a floating cycle nor a floating path, then it must be a pure assignment. This guarantees that the assignment generated by our implementation mechanism is always pure.

Lemma 4. *An assignment is pure if and only if it does not contain floating cycles and floating paths.*

Proof. One direction is trivial: if the assignment is pure then it has no floating cycles or floating paths. We prove the other direction by showing that any assignment x which is not pure contains a floating path or a floating cycle. Since x is not pure, it must contain a floating edge e, i.e. an edge e with $0 < x_e < 1$. We say that a floating edge e is $\mathcal{H}_1$-*loose* ($\mathcal{H}_2$-*loose*) if no tight block in $\mathcal{H}_1$ ($\mathcal{H}_2$) contains e. We say that e is *loose* if it is either $\mathcal{H}_1$-loose or $\mathcal{H}_2$-loose.

We need another definition before presenting the proof. Suppose $S = (B, \underline{q}_B, \bar{q}_B)$ is a tight hard constraint and e is a floating edge in B. Since S is tight, and since the quotas $\underline{q}_B, \overline{q}_B$ are integral, then B must also contain another floating edge e'. We denote this edge by $p(e, B)$. If there is more than one such edge, then let $p(e, B)$ denote one of them arbitrarily.

The proof has two cases, either there is a floating edge which is loose, or there is no such edge.

Case 1: There exists a loose edge. As the first step of the proof, note that we are done if there exists a floating edge which is both $\mathcal{H}_1$-loose and $\mathcal{H}_2$-loose: the edge would form a floating path of length 1. So, w.l.o.g. suppose there is a floating edge e which is not

$\mathcal{H}_2$-loose. In this case, we iteratively construct a floating path that starts from edge e, i.e. a path $F = \langle e_1, \ldots, e_l \rangle$ such that $e_1 = e$. At the end, our iterative construction will either find such a path, or we will find a floating cycle.

Since e_1 is not $\mathcal{H}_2$-loose, then there must be a minimal tight block $B^1 \in \mathcal{H}_2$ that contains e_1. Since B^1 is tight, and since the quotas are integral, then B^1 must also contain another floating edge $p(e_1, B^1)$. We extend our (under construction) floating path by setting $e_2 = p(e_1, B^1)$. Now, if e_2 is $\mathcal{H}_1$-loose, then $\langle e_1, e_2 \rangle$ is a floating path and the proof is complete. So, suppose e_2 is not $\mathcal{H}_1$-loose. Consequently, there must be a minimal tight block $B^2 \in \mathcal{H}_1$ that contains e_2. Similar to before, B^2 must contain another floating edge $p(e_2, B^2)$; we extend F by setting $e_3 = p(e_2, B^2)$.

By repeating this argument, we can extend F iteratively until the new floating edge that is added to F, namely e_k, either (i) is loose, or (ii) is contained in one of the previous tight blocks $B^1, \ldots, B^{k-1}$. If case (i) happens, then F is a floating path and we are done. If case (ii) happens, then we have found a floating cycle: suppose $e_k \in B_j$ with $j < k$. Then, it is straight-forward to verify that $\langle e_{j+1}, \ldots, e_k \rangle$ is a floating cycle.

Case 2: There is no loose edge. Similar to Case 1, we iteratively construct a floating cycle $F = \langle e_1, \ldots, e_l \rangle$. The cycle starts from a floating edge e; initially, we have $e_1 = e$. Since e_1 is not loose, there must be minimal tight blocks $B^0 \in \mathcal{H}_1$ and $B^1 \in \mathcal{H}_2$ such that $e_1 \in B^0$ and $e_1 \in B^1$. Then, let $e_2 = p(e_1, B^1)$. Similarly, since e_2 is not loose, there must be a tight block $B^2 \in \mathcal{H}_1$ such that $e_2 \in B^2$. Let $e_3 = p(e_2, B^2)$. By applying this argument repeatedly, we can extend F until the new floating edge that is added to F, namely e_k, satisfies $e_k \in B_j$ for some j with $0 \leq j < k$. Then, it is straight-forward to verify that $\langle e_{j+1}, \ldots, e_k \rangle$ is a floating cycle. ∎

A.4 Approximate Satisfaction of Soft Constraints

Here we prove that soft constraints are approximately satisfied in the sense of Definition 1. Loosely speaking, Operation $\mathcal{X}$ is designed in a way such that it never increases (or decreases) two (or more) elements of a soft constraint at the same iteration. Consequently, elements of each soft constraint become "negatively correlated". This allows us to employ Chernoff concentration bounds to prove that soft constraints are approximately satisfied.

We show the approximate satisfaction of soft constraints by proving two lemmas below. In the first lemma, we formally (define and) prove that elements of each soft constraint are "negatively correlated"; the proof uses a negative correlation proof technique from [Khuller et al., 2006]. Then, in the second lemma, we prove the approximate satisfaction of soft constraints by applying Chernoff concentration bounds. Before stating the lemmas, we recall the definition of negative correlation.

Definition 8. *For an index set B, a set of binary random variables $\{X_e\}_{e\in B}$ are negatively correlated if for any subset $T \subseteq B$ we have*

$$\Pr\left[\prod_{e\in T} X_e = 1\right] \leq \prod_{e\in T} \Pr\left[X_e = 1\right], \tag{8}$$

$$\Pr\left[\prod_{e\in T}(1 - X_e) = 1\right] \leq \prod_{e\in T} \Pr\left[X_e = 0\right]. \tag{9}$$

Lemma 5. *Let $\{X_e\}_{e\in E}$ denote the set of random variables which represent the outcome of the implementation mechanism (i.e. the integral assignment); also, let B be a block corresponding to an arbitrary soft constraint. Then, the set of random variables $\{X_e\}_{e\in B}$ are negatively correlated.*

Proof. We need to show that (8) and (9) hold for any subset $T \subseteq B$. We fix an arbitrary subset T and prove (8) for it; the proof for (9) is identical and follows by replacing the role of zeros and ones. Since the random variables are binary, we can prove (8) by showing that

$$\mathbb{E}\left[\prod_{e\in T} X_e\right] \leq \prod_{e\in T} \mathbb{E}\left[X_e\right] = \prod_{e\in T} x_e. \tag{10}$$

To prove (10), we introduce a set of random variables $\{X_{e,i}\}$ where $X_{e,i}$ denotes the value of entry e of the matrix after the i-th execution of operation $\mathcal{X}$. So we would have $X_{e,0} = x_e$ for all e. Inductively, we show that for all i:

$$\mathbb{E}\left[\prod_{e\in T} X_{e,i+1}\right] \leq \mathbb{E}\left[\prod_{e\in T} X_{e,i}\right]. \tag{11}$$

The lemma is proved if (11) holds: Assuming that operation $\mathcal{X}$ is executed j times, using (11) we can write

$$\mathbb{E}\left[\prod_{e\in T} X_e\right] = \mathbb{E}\left[\prod_{e\in T} X_{e,j}\right] \leq \mathbb{E}\left[\prod_{e\in T} X_{e,0}\right] = \prod_{e\in T} x_e$$

which shows (10) holds and proves the lemma.

To prove (11), we can alternatively show that

$$\mathbb{E}\left[\prod_{e\in T} X_{e,i+1} \;\middle|\; \{X_{e,i}\}_{e\in T}\right] \leq \prod_{e\in T} X_{e,i}. \tag{12}$$

We consider three cases to prove (12): since B is in the deepest level of a hierarchy, then operation $\mathcal{X}$ changes either 0, 1, or 2 elements of T. We prove this fact in a separate claim below.

Claim 3. *Suppose T is a block in the deepest level of a hierarchy, then, Operation $\mathcal{X}$ changes either* 0*,* 1*, or* 2 *elements of T.*

Proof. W.L.O.G. assume that T is in the deepest level of $\mathcal{H}_1$. We prove a stronger claim. Let T' be the largest subset of links that contains T and is in the deepest level of $\mathcal{H}_1$. We prove that Operation $\mathcal{X}$ changes at most 2 elements of T'. To this end, let F be the floating cycle or path used in Operation $\mathcal{X}$. We need to show that F contains at most 2 elements of T'; this proves the claim.

For contradiction, suppose F contains at least 3 elements of T'. Let the elements of F be denoted by the sequence $e_1, \ldots, e_l$, and let e_i, e_j, e_k be the first three elements of T' which appear in F, where $i < j < k$.

First, note that by the definitions of floating cycle and floating path, we must have that $j = i+1$. We will prove that $\langle e_j, e_{j+1} \ldots, e_{k-1}, e_k\rangle$ makes a floating cycle, which contradicts with the minimality of F (recall that by definition, operation $\mathcal{X}$ always chooses minimal floating paths and cycles). To this end, first note that (e_j, e_{j+1}) is supported by $\mathcal{H}_2$: this holds because $e_{j-1}, e_j \in T'$, which means (e_{j-1}, e_j) is supported by $\mathcal{H}_1$. Consequently, (e_j, e_{j+1}) must be supported by $\mathcal{H}_2$ since F is a floating path or cycle. Similarly, (e_{j+1}, e_{j+2}) is supported by $\mathcal{H}_1$, (e_{j+2}, e_{j+3}) is supported by $\mathcal{H}_2$, and so on and so forth. Finally, note that (e_k, e_j) is supported by $\mathcal{H}_1$, since $e_k, e_j \in T'$. This proves that $\langle e_j, e_{j+1} \ldots, e_{k-1}, e_k\rangle$ is a floating cycle, which concludes the claim. ■

We continue the proof of lemma by considering each of the three cases separately. The proof is trivial if Operation $\mathcal{X}$ changes 0 elements of T: (12) holds with equality. So, it remains to consider the two other cases.

First, assume that Operation $\mathcal{X}$ changes exactly one element of T, namely $e' \in T$. Let

$T' = T \backslash \{e'\}$. Then we have

$$\begin{aligned}
&\mathbb{E}\left[\prod_{e\in T} X_{e,i+1} \,\middle|\, \{X_{e,i}\}_{e\in T}\right] \\
=&\frac{\epsilon'}{\epsilon+\epsilon'}\cdot(X_{e',i}+\epsilon)\cdot\prod_{e\in T'}X_{e,i}+\frac{\epsilon}{\epsilon+\epsilon'}\cdot(X_{e',i}-\epsilon')\cdot\prod_{e\in T'}X_{e,i}=\prod_{e\in T}X_{e,i}
\end{aligned}$$

which proves (12) with equality in this case. It remains to prove (12) for the case when Operation $\mathcal{X}$ changes exactly 2 elements of T, namely $e', e'' \in T$. Let $T'' = T\backslash\{e', e''\}$. Then, w.l.o.g. we can write:

$$\begin{aligned}
&\mathbb{E}\left[\prod_{e\in T} X_{e,i+1} \,\middle|\, \{X_{e,i}\}_{e\in T}\right] \\
=&\frac{\epsilon'}{\epsilon+\epsilon'}\cdot(X_{e',i}+\epsilon)(X_{e'',i}-\epsilon)\cdot\prod_{e\in T''}X_{e,i}+\frac{\epsilon}{\epsilon+\epsilon'}\cdot(X_{e',i}-\epsilon')(X_{e'',i}+\epsilon')\cdot\prod_{e\in T''}X_{e,i} \\
=&\prod_{e\in T}X_{e,i}-\epsilon\epsilon'\cdot\prod_{e\in T''}X_{e,i} \\
\leq&\prod_{e\in T}X_{e,i}
\end{aligned}$$

which proves (12) in the third case as well. This finishes the proof of lemma. ■

Lemma 6. *The randomized mechanism based on Operation $\mathcal{X}$ satisfies the soft constraints approximately in the sense of Definition 1.*

Proof. Based on Definition 1, we need to prove that for any soft constraint defined on a block B of the links with $\sum_{e\in B} w_e x_e = \mu$, and for any $\epsilon > 0$, we have

$$\begin{aligned}
\Pr\left(\sum_{e\in B} w_e X_e - \mu < -\epsilon\mu\right) &\leq e^{-\mu\frac{\epsilon^2}{2}}, \\
\Pr\left(\sum_{e\in B} w_e X_e - \mu > \epsilon\mu\right) &\leq e^{-\mu\frac{\epsilon^2}{3}}.
\end{aligned}$$

These probabilistic bounds, as we mentioned before, are known as Chernoff concentration bounds (see Section G for more details). These bounds hold on any set of binary random variables which are negatively correlated [Auger and Doerr, 2011]. Lemma 5 just says that the set of random variables $\{X_e\}_{e\in B}$ are negatively correlated, which means Chernoff concentration bounds hold for $\{X_e\}_{e\in B}$. ■

B Small, Additive Bounds for Specific Structures

We use school choice as a motivating example to describe this setting, but readers can easily see that the framework can be extended to other two-sided market. Consider a school-choice setting where students are to be assigned to schools. In the language of our model, suppose N represents the set of students, and O represents the set of schools. We are given a partitioned structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$, where $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$. We suppose $\mathcal{H}_1$ contains all of the row blocks (to ensure that every student will be assigned to a school), $\mathcal{H}_2$ is a hierarchical constraint structure, and $\mathcal{S}$ is a local constraint structure that is in the deepest level of $\mathcal{H}$ (This assumption is in the spirit of Theorem 1, and can be relaxed in exchange of weaker guarantees on violation of the soft constraints). For expositional simplicity, in the rest of this section we will assume that $\mathcal{H}_2$ is the set of all capacity constraints of the schools. Our result still holds when this assumption is dismissed.

We say two students are of the same *type* if whenever one of them is in a constraint in $\mathcal{H}_2 \cup \mathcal{S}$, the other one is also in the same constraint. Denote the set of all types by $\mathcal{T} = \{1, \ldots, T\}$. In the following, we define the notion of approximate implementation with additive errors.

Definition 9. *Given a hard-soft partitioned structure $\mathcal{E} = \mathcal{H} \cup \mathcal{S}$, we say $\mathcal{E}$ is* **Approximately Implementable with Additive Error *k*** *if for any vector of quotas $\mathbf{q}$ and any expected assignment x which is feasible with respect to $(\mathcal{E}, \mathbf{q})$, there exists a lottery (probability distribution) over pure assignments $X_1, \ldots, X_K$ such that, if we denote the outcome of the lottery by the random variable X, the following properties hold:*

P1. **Assignment Preservation:** $\mathbb{E}[X] = x$.

P2. **Exact Satisfaction of Hard Constraints:** *All constraints in $\mathcal{H}$ are satisfied.*

P3. **Approximate Satisfaction of Soft Constraints:** *For any soft block $B \in \mathcal{S}$ with $\sum_{e \in B} x_e = \mu$, we have*

$$|\sum_{e \in B} X_e - \mu| \leq k$$

Our main theorem states that that any feasible fractional assignment is approximately implementable with additive error T, and the proof, as in the main theorem, based on Operation $\mathcal{X}$.

Theorem 8. *If $\mathcal{H}$ is the set of all capacity constraints and there are at most T student types, then any feasible fractional assignment x is approximately implementable with additive error*

T.

Proof. The implementation is done in two steps. In the first step, we determine how many students from each type are assigned to each school. In the second step, we determine the assignment of students to schools within each type.

To describe the first step, we define the *super-assignment* corresponding to x, which we denote by $\hat{x}$, as follows: $\hat{x}$ is a matrix that has a row for each type and a column for each school, and $\hat{x}_{t,c}$ represent the total fraction of students of type t who are assigned to school c in the given fractional assignment x, i.e. $\hat{x}_{t,c} = \sum_{i \in N(t)} x_{i,t}$, where $N(t)$ denotes the set of students of type t.

We use our main theorem (Theorem 1) to implement $\hat{x}$ as a pure super-assignment that we denote by $\hat{x}^*$ (as we mentioned earlier, we will use this pure super-assignment in the second step to determine the final pure assignment). To use our main theorem, we need to impose a set of hard and soft constraints on $\hat{x}^*$. Let $\hat{\mathcal{H}} = \hat{\mathcal{H}}_1 \cup \hat{\mathcal{H}}_2$ and $\hat{S}$ respectively denote the hard and soft constraint structure that we impose on $\hat{x}^*$. Define

$$\hat{\mathcal{H}}_1 = \Big\{ \sum_{c \in O} \hat{x}^*_{t,c} = \sum_{c \in O} \hat{x}_{t,c} : \forall t \in \mathcal{T} \Big\} \bigcup \Big\{ \lfloor \hat{x}_{t,c} \rfloor \leq \hat{x}^*_{t,c} \leq \lceil \hat{x}_{t,c} \rceil : \forall t \in \mathcal{T}, c \in O \Big\},$$
$$\hat{\mathcal{H}}_2 = \Big\{ \sum_{t \in \mathcal{T}} \hat{x}^*_{t,c} \leq \lceil \sum_{t \in \mathcal{T}} \hat{x}_{t,c} \rceil : \forall c \in O \Big\},$$

and $\hat{\mathcal{S}} = \emptyset$. Applying Theorem 1 then would imply that the outcome of the lottery, $\hat{x}^*$, will satisfy all of the constraints in $\hat{\mathcal{H}}$. This finishes the first step.

In the second step, we construct a pure assignment x^* from the pure super-assignment $\hat{x}^*$. We will see that the assignment x^* implements the original assignment x, satisfies $\mathcal{H}$, and satisfies any constraint in $\mathcal{S}$ with error at most T. The assignment x^* is constructed by running a separate lottery for each type which would determine the final assignment for students of that type.

Fix a type t. Construct an assignment matrix y^t (that corresponds to type t) as follows: y^t has a row for each student of type t and a column for each school. For any $i \in N_t, c \in O$, let $y^t_{i,c} = x_{i,c}$ (in words, y^t represents a submartrix of x that includes all students of type t). To determine the final assignment for students of type t, we implement y^t by applying Theorem 1, i.e. y^t will be implemented by a lottery over pure assignments. Let y^{t*} denote the outcome of the lottery. We impose a bihierarchical hard constraint structure $\mathcal{H}^y = \mathcal{H}^y_1 \cup \mathcal{H}^y_2$ on y^{t*} (and thus the final outcome of the lottery, y^{t*}, will be feasible with respect to $\mathcal{H}^y$, as it is guaranteed by Theorem 1). Define this hard constraint structure as follows:

$$\mathcal{H}_1^y = \Big\{ \sum_{c \in O} y_{i,c}^{t*} = 1 : \forall i \in N_t \Big\},$$
$$\mathcal{H}_2^y = \Big\{ \sum_{i \in N_t} y_{i,c}^{t*} = \hat{x}_{t,c}^* : \forall c \in O \Big\}.$$

The final assignment of students, x^*, is determined by the set $\{y^{t*} : t \in \mathcal{T}\}$, i.e. $x^*(i) = y^{t*}(i)$ for all $i \in N_t$. This finishes the second step.

It is straight-forward to see that, by construction, x^* satisfies $\mathcal{H}$ and that $\mathbb{E}[x^*] = x$. In the rest of the proof, we will show that for any soft constraint $C \in \mathcal{S}$, x^* violates C with error at most T. Recall that C is a local constraint, and so it should involve a single school c and a subset of students whose type belongs to a subset $\mathcal{T}' \subseteq \mathcal{T}$. ($C$ could also involve a single student and multiple schools, but in that case, it is clear that C will not be violated with error more than 1). Constraint C then could be represented as

$$\underline{q}_C \leq \sum_{t \in \mathcal{T}'} \sum_{i \in N(t)} x_{i,c} \leq \overline{q}_C.$$

First, note that by definition, $\hat{x}$ satisfies this constraint in the sense that

$$\underline{q}_C \leq \sum_{t \in \mathcal{T}'} \hat{x}_{i,c} \leq \overline{q}_C. \tag{13}$$

Also, recall that by the definition of $\hat{\mathcal{H}}$, we have $\lfloor \hat{x}_{t,c} \rfloor \leq \hat{x}_{t,c}^* \leq \lceil \hat{x}_{t,c} \rceil$ for all types t and schools c. This fact and (13) together imply that

$$\Big| \sum_{t \in \mathcal{T}'} \hat{x}_{t,c} - \sum_{t \in \mathcal{T}'} \hat{x}_{t,c}^* \Big| \leq |\mathcal{T}'| \leq T. \tag{14}$$

Also, by the construction of x^* in the second step, we have that

$$\sum_{t \in \mathcal{T}'} \hat{x}_{t,c}^* = \sum_{t \in \mathcal{T}'} \sum_{i \in N(t)} x_{i,c}^*.$$

The above equation and (14) together imply that

$$\Big| \sum_{t \in \mathcal{T}'} \hat{x}_{t,c} - \sum_{t \in \mathcal{T}'} \sum_{i \in N(t)} x_{i,c}^* \Big| \leq T,$$

which, due to (13), implies that x^* violates C with error at most T. ∎

C Computational Experiments of the Matching Algorithm

In this section, we implement our matching algorithm on an example. The goal of this example is to show that the average performance of our matching algorithm is *much* better than the worst-case bounds that one can theoretically prove. For the sake of clarity, we use a simple example with multiple intersecting constraints.

Setup of example: Consider a school choice setting, with 10 schools and 10000 students. Suppose each school has a capacity for 1000 students. Also, suppose half of the students are from the walk-zone of schools 1, 2, 3, 4, and 5 and the other half are from the walk-zone of schools 6, 7, 8, 9, and 10. Also, half of the students are categorized as low-socioeconomic status (LSES) students, and half of the students are male. Suppose all students have the same utility function (or rank-order list) over schools.

Hard and soft constraints: The only hard constraints imposed on this problem are "all-row" constraints: All student should be assigned to exactly one school. All schools have three diversity goals that we model them as soft constraints: Their goal is to admit 500 students (i.e. 50% of their capacity) from the students of their own walk-zone, 500 student from LSES students, and 500 female students.

Fractional assignment: Let x be a fractional assignment, where $x_{(i,j)} = \frac{1}{10}$ for all pairs (i, j). One can easily show that x satisfies all hard and soft constraints exactly.[30]

Simulation: We implement this fractional assignment by our matching algorithm based on Operation $\mathcal{X}$ for 1000 times. We then calculate the "empirical probability" of violating each one of the diversity constraints by a factor of $\epsilon = 1\%, 2\%, \cdots, 10\%$.

Figure C.1 illustrates the empirical probability of admitting less than $500(1-\epsilon)$ students of a specific diversity type. As can be seen, the average performance of our matching algorithm is *much* better than the worst-case bound that we can theoretically prove. Figure C.2 illustrates the empirical probability of admitting more than $500(1+\epsilon)$ students of a specific diversity type. Again, the average performance of our matching algorithm is *much* better than the theoretical worst-case bound.

[30] It is also clear that because of the symmetry of the problem, this assignment is fair and Pareto efficient.

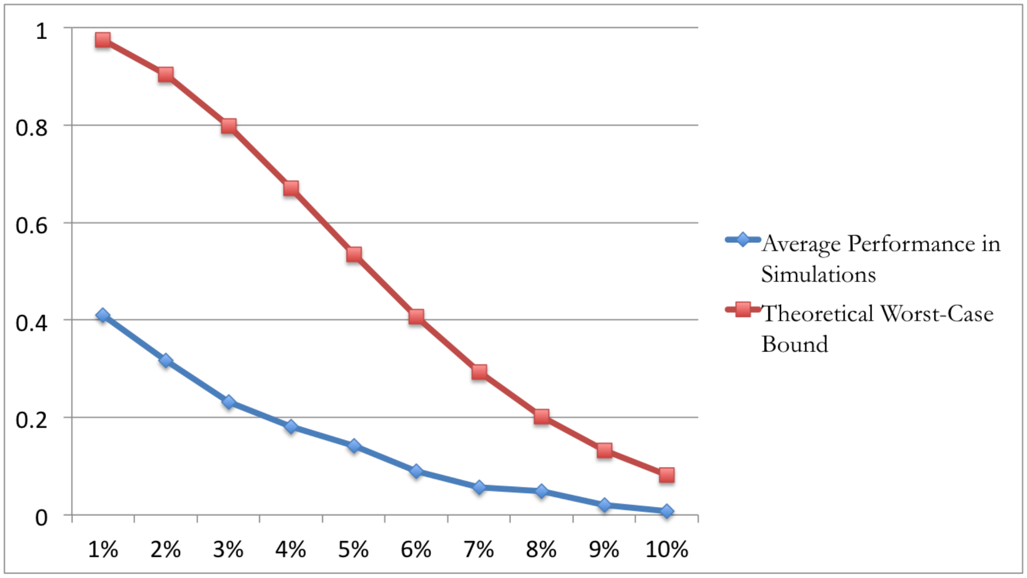

Figure C.1: The empirical probability of violating a constraint from below by $\epsilon\%$ (equivalently, $\Pr(\mathsf{dev}^- \geq \epsilon\mu)$), where $\mu = 500$. The probability is calculated by running the matching algorithm for $T = 1000$ times and then computing the probability of admitting less than $500(1-\epsilon)$ students.

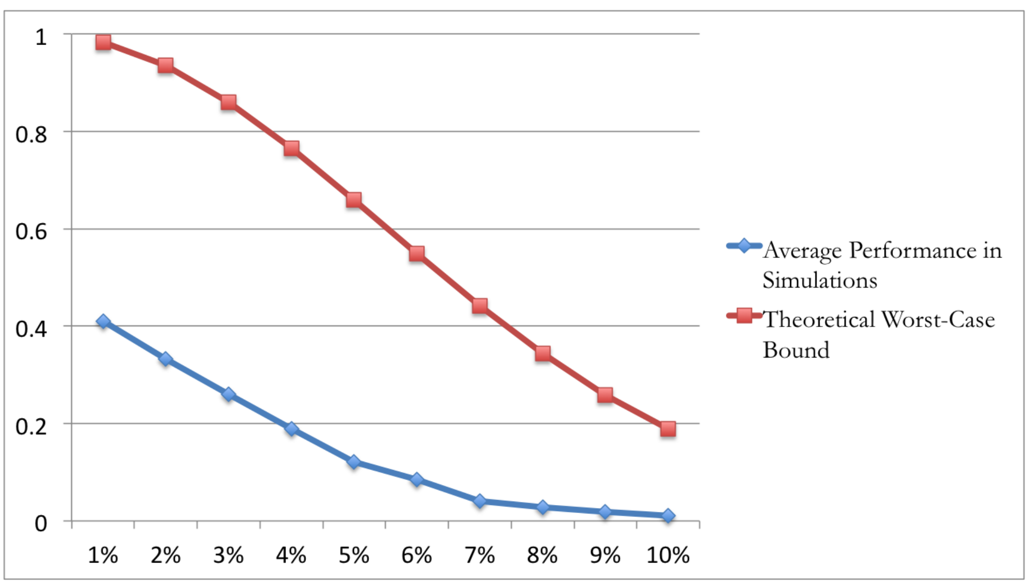

Figure C.2: The empirical probability of violating a constraint from above by $\epsilon\%$ (equivalently, $\Pr(\mathsf{dev}^+ \geq \epsilon\mu)$), where $\mu = 500$. The probability is calculated by running the matching algorithm for $T = 1000$ times and then computing the probability of admitting more than $500(1+\epsilon)$ students.

D Impossibility of Fully General Structures: An Example

The following example shows that without any structure on soft constraints, guaranteeing small errors is impossible.

Let $N = \{1, \ldots, n\}$ and $O = \{1, \ldots, n\}$. Consider the following constraints: agent i wants to have exactly one of the objects $i, i+1$ (where $n+1 = 1$), and each object cannot be assigned to more than one agent, i.e. there is only one copy of each object. These constraint can be modeled by a set of hard bihierarchical constraints, $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$, as follows:

$$\mathcal{H}_1 = \{x_{(i,i)} + x_{(i,i+1)} \leq 1\}_{i=1,\ldots,n},$$
$$\mathcal{H}_2 = \{x_{(i,i)} + x_{(i-1,i)} \leq 1\}_{i=1,\ldots,n}.$$

where again for simplicity in notation we have assumed $0 = n$. Also, we define a single soft constraint

$$\mathcal{S} = \left\{ \sum_{i=1}^{\lfloor n/2 \rfloor} x_{(2i,2i)} + \sum_{i=1}^{\lfloor n/2 \rfloor} x_{(2i-1,2i)} = 2\lfloor n/2 \rfloor \right\}.$$

Observe that the fractional assignment defined by

$$\bar{x}_{(i,i)} = \bar{x}_{(i,i+1)} = \frac{1}{2}, \qquad \forall i = 1, \ldots, n$$

satisfies all of the hard and soft constraints exactly. Now, we show that any lottery which implements $\bar{x}$ and satisfies the hard constraints exactly, must severely violate the soft constraint by an additive factor of at least $\lfloor n/2 \rfloor$.

First, observe that there exists a unique convex combination of pure assignments which is equal to $\bar{x}$ and it is defined by $\bar{x} = 0.5y + 0.5z$ where y, z are defined as follows:

$$\bar{y}_{(i,i)} = 1, \bar{y}_{(i,i+1)} = 0, \qquad \forall i = 1, \ldots, n$$
$$\bar{z}_{(i,i)} = 0, \bar{z}_{(i,i+1)} = 1, \qquad \forall i = 1, \ldots, n.$$

So, the outcome of the unique lottery that implements $\bar{x}$ must be y with probability 0.5 and z otherwise. In both of these cases, the left-hand side of the soft constraint is exactly equal to $\lfloor n/2 \rfloor$, which leads to a severe ex post violation of this constraint.

E Probabilistic guarantees for general soft constraints

For simplicity we only give the proof for upper deviation, i.e. for the probabilistic bound (4). The proof for (5) is similar. Since B has depth k, it can be partitioned into k blocks $B_1, \ldots, B_k$ all of which are in the deepest level of $\mathcal{H}$. In order to provide a guarantee on the satisfaction of the soft constraint corresponding to B, we add k constraints, one for each of $B_1, \ldots, B_k$, to our soft constraint set. The (soft) constraint corresponding to block B_i, denoted by C_i, would be

$$\sum_{e \in B_i} X_e \leq \mu_i,$$

where $\mu_i = \sum_{e \in B_i} x_e$. Since C_i is in the deepest level of $\mathcal{H}$, the following guarantee would hold on X, the outcome of our mechanism: (by Theorem 1)

$$\Pr(\mathsf{dev}_i^+ \geq \epsilon_i \mu_i) \leq e^{-\mu_i \frac{\epsilon_i^2}{3}},$$

where ϵ_i can be any positive number and

$$\mathsf{dev}_i^+ = \max\big(0, \sum_{e \in B_i} X_e - \mu_i\big).$$

The key is to define ϵ_i's such that

$$e^{-\mu_i \frac{\epsilon_i^2}{3}} = e^{-\mu \frac{\epsilon^2}{3k}}, \tag{15}$$

$$\sum_{i=1}^{k} \epsilon_i \mu_i \leq \epsilon \mu. \tag{16}$$

If these two properties hold, then a union bound on the constraints $C_1, \ldots, C_k$ would prove the claim: By (15), the probability that (at least) one of the constraints C_i is is violated with (additive) error more than $\epsilon_i \mu_i$ is at most $k e^{-\mu \frac{\epsilon^2}{3k}}$. On the other hand, if all constraints C_i are satisfied with (additive) error not more than $\epsilon_i \mu_i$, then using (16) we get:

$$\mathsf{dev}^+ \leq \sum_{i=1}^{k} \mathsf{dev}_i^+ \leq \sum_{i=1}^{k} \epsilon_i \mu_i \leq \epsilon \mu. \tag{17}$$

This would prove the claim. So, to finish the proof, it only remains to define ϵ_i's such that (15) and (16) would hold. To this end, define $\alpha_i = k\mu_i/\mu$ and let $\epsilon_i = \epsilon/\sqrt{\alpha_i}$. It is straightforward to verify that this definition satisfies (15). To see that (16) also holds, we rewrite its left-hand side as follows:

$$\sum_{i=1}^{k} \epsilon_i \cdot \mu_i = \sum_{i=1}^{k} \frac{\epsilon}{\sqrt{\alpha_i}} \cdot \frac{\alpha_i \mu}{k} = \frac{\epsilon\mu}{k} \cdot \sum_{i=1}^{k} \sqrt{\alpha_i} \leq \epsilon\mu,$$

where in the last inequality uses the fact that $\sum_{i=1}^{k} \alpha_i = k$, which implies $\sum_{i=1}^{k} \sqrt{\alpha_i} \leq k$. The above inequality shows that (16) holds; this completes the proof.

F Proof of Theorem 6 and Theorem 7

The proof has multiple steps. We first show that as the market size grows a *fractional* ϵ-equilibrium, in which all objects' prices are "small" (relative to agents' budgets) exists. Then we show that by implementing the *fractional* ϵ-equilibrium by our lottery based on Operation $\mathcal{X}$, one can prove the existence of an equilibrium in sufficiently large markets.

First, we prove the following lemma. To state the theorem, without loss of generality, let $w_1 = \min_{i \in N} w_i$.

Lemma 7. *For any fixed $\delta > 0$ and any $\epsilon > 0$, there exists q_0 such that for all $q > q_0$, in M_q there exists a* fractional *ϵ-equilibrium in which for all agents $a \in A$ and for all objects $o \in O_q$ that are in the demand correspondence of agent a we have $p_o/w_1 < \delta$.*

Proof. To prove the lemma, we first prove the following claim. This claim shows that as the market size grows, agents' utilities from their most favorite bundle grow as well.

Claim 4. *For each agent, we should have $v_a(\mathbf{p}) \geq \frac{w_a}{\sum_{i \in N} w_i} \cdot u_a(O)$.*

Proof. Let $v_a(\mathbf{p}, x)$ denote $v_a(\mathbf{p})$ when $w_a = x$. It is straight-forward to show that $v_a(\mathbf{p}, x)$ is concave in x. This, and the fact that $v_a(\mathbf{p}, \sum_{i \in N} w_i) = u_a(O)$, together prove the claim. ■

If in a competitive equilibrium (CE) $(\mathbf{p}, x)$, the conditions of the lemma are satisfied (*i.e.*, prices are "small"), then we are done. So, suppose this is not the case and in CE, the condition is not satisfied. So there is a set of objects S such that for all $o \in S$, $p_o \geq w_1\delta$. Let $W = \sum_{i \in N} w_i$ and $\beta = W/w_1$. In any CE, we must have $|S| \leq \beta/\delta$ (even if all agents' budgets are consumed for objects in S).

Fix δ and ϵ. Now we construct an ϵ-equilibrium from the above CE $(\mathbf{p}, x)$ in the following way: We decrease the prices of all objects in S to $w_1\delta$ and denote the new price vector by $\mathbf{p}'$. We claim that there exists some q_0 such that for $q > q_0$, $(\mathbf{p}', x)$ is a fractional ϵ-equilibrium. That is because in $\mathbf{p}'$ (compared to $\mathbf{p}$), only the demand for objects in S may increase (as their prices have decreased), but even if an agent consumes all of the objects whose prices are decreased, we have that: $v_a(\mathbf{p}') \leq v_a(\mathbf{p}) + \beta/\delta = u_a(x) + \beta/\delta$. Now, by the large market assumption, $u_a(x)$ approaches infinity, but β/δ is constant. Hence, there exists some q_0 such that for $q > q_0$, $v_a(\mathbf{p}') \geq u_a(x) \geq v_a(\mathbf{p}')(1-\epsilon)$.

■

The rest of the proof of the Theorem 6 is based on the following steps. We first define an (ϵ, ϵ')-equilibrium which is same as the notion of ϵ-equilibrium with the difference that we allow for budgets to be violated by a factor of at most $1+\epsilon'$. Then, by applying Lemma 7, we prove that an (ϵ, ϵ')-equilibrium exists in large markets. More precisely, we prove:

Lemma 8. *For any fixed $\epsilon, \epsilon' > 0$, there exists q_0 such that for all $q > q_0$, an (ϵ, ϵ')-equilibrium exists in $\mathcal{M}_q$.*

Proof. For a given ϵ, pick a δ such that $(1-\delta)^2 \geq 1-\epsilon$. By Lemma 7, for sufficiently large markets, there exists a fractional assignment x such that $(\mathbf{p}, x)$ is a fractional δ-equilibrium and prices are small relative to agents' budgets. Recall that at $\mathbf{p}$ and for x, the budget constraint is exactly satisfied. Let $u_a(x)$ be agents' utilities in this equilibrium. Now suppose we implement x by the iterative application of the Operation $\mathcal{X}$. Let x^* be the outcome of this lottery based on Operation $\mathcal{X}$.

Let S_a be the set of items assigned to agent a. Then for any agent a,

$$\Pr\Big(u_a(x^*) \leq v_a(\mathbf{p})(1-\delta)^2\Big) \leq e^{-v_a(\mathbf{p})(1-\delta)\delta^2/2},$$

which holds since the utility of agent a in the fractional δ-equilibrium is at least $v_a(\mathbf{p})(1-\delta)$.

In addition, if we think of each agent's budget constraint as a soft constraint, then the same bounds will go through for budgets as well; that is, if we define $p_m = \max_{o \in O} p_o$, then we have that for any $\epsilon' > 0$ and for all agents,

$$\Pr\Big(\sum_{o \in S_a} p_o \geq w_a(1+\epsilon')\Big) \leq e^{-w_a \epsilon'^2/3p_m}.$$

Therefore,

$$\sum_{a \in N}\Big(\Pr\Big(u_a(x) \leq v_a(\mathbf{p})(1-\delta)^2\Big) + \Pr\Big(\sum_{o \in S_a} p_o \geq w_a(1+\epsilon')\Big)\Big) \\ \leq \sum_{a \in N}\Big(e^{-v_a(\mathbf{p})(1-\delta)\delta^2/2} + e^{-w_a \epsilon'^2/3p_m}\Big)$$

Now by Claim 4, agents' utilities approach infinity as the market size grows, and by

Lemma 7, w_a/p_m approaches infinity as the market size grows. Hence, there exists some q_0 such that for all $q > q_0$, in $\mathcal{M}_q$, we have:

$$\sum_{a \in N} \left(e^{-v_a(\mathbf{p})(1-\delta)\delta^2/2} + e^{-w_a \epsilon'^2/3p_m} \right) < 1$$

Note that the above sum is strictly less than 1 in $\mathcal{M}_q$; that is, the probability that *all* utility and budget constraints are violated in $\mathcal{M}_q$ at the same time is less than 1. This means that, there exists *some* pure assignment in which for all agents $a \in N$, $u_a(x) \geq v_a(\mathbf{p})(1-\delta)^2$ and $\sum_{o \in S_a} p_o \leq w_a(1+\epsilon')$. Now, since by the choice of δ we have $(1-\delta)^2 \geq 1-\epsilon$, then, $u_a(x) \geq v_a(\mathbf{p})(1-\epsilon)$, and so (ϵ, ϵ')-equilibrium exists.

■

In the next step, we define a new instance of the problem by scaling all agents' budgets to $w_a/(1+\epsilon)$. For this instance and for any $\epsilon > 0$, an (ϵ^2, ϵ)-equilibrium exists by Lemma 8. This means that in this equilibrium, an agent's payment is at most $(1+\epsilon)w_a/(1+\epsilon) = w_a$ and an agent's utility is at least $(1-\epsilon^2)v_a(\mathbf{p})/(1+\epsilon) = (1-\epsilon)v_a(\mathbf{p})$. Therefore, this equilibrium is an ϵ-equilibrium. □

F.1 Proof of Theorem 7

We use Theorem 6 of BCKM, which implies the existence of a (fractional) competitive equilibrium assuming that the items are divisible. Given this fractional equilibrium, the rest of the proof is almost identical to the proof of Theorem 6. We use identically the same argument that was used in Theorem 6 to prove the existence of a fractional ϵ-CE in which all objects' prices are "small"; that is, as market size grows, no agent spends a constant fraction of his endowments for a single object. Then, we implement the fractional ϵ-CE by our Operation $\mathcal{X}$-based lottery.

We remark that in the proof, we have to use the generalized bounds (4) and (5), rather than the bounds that were used in Theorem 6 which only hold for the deepest level constraints.[31]

[31] The proof uses the generalized bounds because the agents' soft utility constraints are not necessarily in the deepest level of the bihierarchy, rather, they have depth at most Δ.

G Chernoff Bounds

Let $X_1, \ldots, X_n$ be a sequence of n independent random binary variables such that $X_i = 1$ with probability p_i and $X_i = 0$ with probability $1 - p_i$. Also, let $\mu = \sum_{i=1}^{n} \mathbb{E}[X_i]$. Then for any ϵ with $0 \le \epsilon \le 1$ we have:

$$\Pr\left[\sum_{i=1}^{n} X_i > (1+\epsilon)\mu\right] \le e^{-\epsilon^2\mu/3} \tag{18}$$

$$\Pr\left[\sum_{i=1}^{n} X_i < (1-\epsilon)\mu\right] \le e^{-\epsilon^2\mu/2}. \tag{19}$$

Moreover, the above inequalities still hold if the variables $X_1, \ldots, X_n$ are negatively correlated. (We refer the reader to Definition 8 for the formal definition of negative correlation)

H Tightness of the probabilistic bounds

We will show that our bounds for approximate satisfaction of soft constraints are optimal up to multiplicative constants in the exponents. More precisely, we can show that there exists a constant $c > 0$ such that any lottery that satisfies the hard constraints can approximately satisfy soft constraints (in the sense of Definition 1) with a probabilistic guarantee no better than $e^{-\frac{\epsilon^2\mu}{c}}$. Formally, consider a lottery that guarantees to satisfy the hard constraints and gives the following guarantees for the satisfaction of soft constraints: for any constant $\epsilon > 0$, and for any soft constraint in $\mathcal{S}$ defined on the block S with $\sum_{e \in S} x_e = \mu$, we have

$$\Pr\left[\sum_{e \in S} X_e \le \mu(1-\epsilon)\right] \le f(\mu, \epsilon),$$

$$\Pr\left[\sum_{e \in S} X_e \ge \mu(1+\epsilon)\right] \le f(\mu, \epsilon).$$

Then, there exists a constant $c > 0$ such that $\lim_{\mu \to \infty} \frac{e^{-\frac{\epsilon^2\mu}{c}}}{f(\mu,\epsilon)} = 0$ holds for all $\epsilon > 0$.

To keep the proof simple, here we prove a weaker version of this result which allows the constant c to be a function of ϵ.

Proposition 1. *Consider a lottery that, given any hard-soft partitioned constraint structure, guarantees to satisfy the hard constraints and gives the following guarantees for the satisfaction of soft constraints: for any constant $\epsilon > 0$, and for any soft constraint defined on the* *block S with $\sum_{e \in S} x_e = \mu$, the lottery guarantees that*

$$\Pr\left[\sum_{e\in S} X_e \leq \mu(1-\epsilon)\right] \leq f(\mu,\epsilon),$$
$$\Pr\left[\sum_{e\in S} X_e \geq \mu(1+\epsilon)\right] \leq f(\mu,\epsilon).$$

Then, for any constant $\epsilon > 0$, there exists a constant $c_\epsilon > 0$ such that $\lim_{\mu\to\infty} \frac{e^{-\frac{\epsilon^2 \mu}{c_\epsilon}}}{f(\mu,\epsilon)} = 0$ holds.

Note that in Proposition 1, we allow c_ϵ to be a function of ϵ. As we mentioned earlier, it is possible to prove a stronger version of this result by choosing c_ϵ so that $\sup_{\epsilon>0}\{c_\epsilon\} < \infty$. We only prove the weaker version here, for the sake of simplicity. We remark that if the range of ϵ that we are interested in is bounded away from 0, i.e. $\epsilon > \underline{\epsilon} > 0$, then $\sup_{\epsilon > \underline{\epsilon}}\{c_\epsilon\} < \infty$, and therefore, in this case, the weak version of the result implies the strong version as well.

The following fact will be useful in the proof.

Fact 1 ([wik, 2016]). *If $\kappa \to \infty$ and $j/\kappa \to x$ for a real number x, then*

$$\frac{\binom{z+\kappa}{j}}{\binom{\kappa}{j}} \to \left(1 - \frac{j}{\kappa}\right)^{-z}.$$

Proof of Proposition 1: The proof goes by constructing a family of instances, $\mathcal{F}$. For any integer n, any constant $\epsilon > 0$, and any function $\mu : \mathbb{Z}_+ \to \mathbb{R}_+$ for which $\mu(n) \leq n/2$ and $\lim_{n\to\infty} \frac{\mu(n)}{n} \to x$ for a real number x, $\mathcal{F}$ contains one instance. To define this instance, fix n, and denote $\mu(n)$ by μ, for notational simplicity. This instance contains a set of n agents, $N = \{1, \ldots, n\}$, and one object.[32] We use the variables $x_1, \ldots, x_n$ to denote the assignment of agent i to the object. We define the hard-soft partitioned constraint structure as

$$\mathcal{H} = \left\{\lfloor \mu \rfloor \leq \sum_{i\in N} x_i \leq \lceil \mu \rceil\right\},$$
$$\mathcal{S} = \left\{\sum_{i\in S} x_i \geq \mu/2 : \forall S \subseteq [n], |S| = \lfloor n/2 \rfloor\right\}.$$

Consider the fractional assignment that assigns μ/n to all variables, i.e. $x_i = \mu/n$ for all $i \in N$. We denote this assignment by x. Our goal is showing that any integer assignment

[32]The assumption $\mu < n/2$ is made for technical simplicity. We can replace it with $\mu < n\delta$ for any positive constant $\delta < 1$, given that $\epsilon > 0$ is a sufficiently small constant.

that satisfies the hard constraints violates at least $|\mathcal{S}| \cdot e^{-\frac{\epsilon^2 \mu}{\Theta(1)}}$ of the soft constraints. This will prove the proposition.

Let $\mu = n/k$ for some scalar k (therefore, $k > 2$ must hold). Also, let x^* denote the outcome of the lottery. Therefore, we should have $x_i^* = 0$ for at least $n - n/k - 1$ different elements $i \in N$; let S' denote the set of all such elements. Note that any $S'' \subseteq S'$ with $|S''| = n/2$ is a soft constraint, which is obviously not approximately satisfied. (For notational simplicity, we avoid carrying floors and ceilings in the notation.) Therefore, at least a fraction

$$\frac{\binom{n-\frac{n}{k}-1}{n/2}}{\binom{n}{n/2}}$$

of the soft constraints will not be satisfied, so long as the hard constraint is satisfied. Note that as $n \to \infty$, this expression approaches

$$\left(1 - \frac{1}{2(1 - 1/k - 1/n)}\right)^{-\frac{n}{k}-1}, \tag{20}$$

by Fact 1. Also, since $e^{-x} \leq 1 - x/2$ for all positive $x \leq 1$, then (20) can be bounded from below by

$$e^{-n\cdot\left(\left(\frac{1}{k}+\frac{1}{n}\right)-\left(\frac{1}{k}+\frac{1}{n}\right)^2\right)}.$$

Therefore,

$$\frac{\binom{n-\frac{n}{k}-1}{n/2}}{\binom{n}{n/2}} \geq e^{-n\cdot\left(\left(\frac{1}{k}+\frac{1}{n}\right)-\left(\frac{1}{k}+\frac{1}{n}\right)^2\right)} \geq e^{-\frac{n}{k}-1}, \tag{21}$$

which means that a fraction $e^{-\frac{n}{k}-1}$ of the soft constraints will not be satisfied. This is just saying $f(\mu, \epsilon) \geq e^{-\mu-1}$. Setting c_ϵ to any constant strictly greater than ϵ^{-2} proves the claim.

■

I Why Simpler Lotteries Fail

Here, we show that in the simple lottery designed in Section 4.5, even when the edges are visited in random order, Assignment Preservation will not always be satisfied. Suppose $N = \{1\}$, and $O = \{1, \ldots, m\}$. Let $E = \{e_1, \ldots, e_m\}$, where $e_i = (1, i)$. Let $x_1 = p$, and $x_i = q$ for all $i > 1$, where $p + q(m-1) = 1$. Assume that there is only one hard constraint: $\sum_{e \in E} X_e = 1$.

Recall that the lottery picks a permutation (over the set of all edges) uniformly at random, and visits the edges in that order. The chance that $X_1 = 1$ is then precisely equal to

$$\begin{aligned}&\left(\sum_{i=0}^{n-2} \frac{1}{n} \cdot (1-q)^i p\right) + \frac{1}{n}(1-q)^{n-1} \\ =&\frac{p}{n} \cdot \frac{1-(1-q)^{n-1}}{q} + \frac{1}{n} \cdot (1-q)^{n-1} \end{aligned} \tag{22}$$

We will show that (22) is not always equal to p, which would imply that Assignment Preservation will not always be satisfied. Noting that $p = 1 - q(n-1)$, and multiplying (22) by n/p gives

$$\frac{1-(1-q)^{n-1}}{q} + \frac{(1-q)^{n-1}}{1-q^{n-1}},$$

which we denote by $f(q)$. Therefore, it suffices to show that $f(q) = n$ is not satisfied for all values of $q \leq 1/n$. A straight-forward calculation shows that

i. for all $q \in (0, 1/n]$, $f(q)$ is a strictly convex function of q,

ii. $f(q) = n$ at $q = 1/n$,

iii. and $f(q) \to n$ as $q \to 0^+$.

Therefore, for any positive $q < 1/n$, $f(q) \neq n$, and Assignment Preservation is not satisfied.