DESIGNING RANDOM ALLOCATION MECHANISMS: THEORY AND APPLICATIONS

ERIC BUDISH, YEON-KOO CHE, FUHITO KOJIMA, AND PAUL MILGROM

ABSTRACT. Lotteries loom large in resource allocation when the resources assigned are indivisible and monetary transfers are limited. We study random allocations by focusing on the marginal distribution of objects to agents – rather than the joint lottery for all objects and agents – an approach pioneered by Hylland and Zeckhauser (1979). We show how to broaden the approach to accommodate various features and constraints encountered in real-world markets, including group-specific quotas ("controlled choice") and endogenous capacities often present in school choice and housing allocation, and scheduling and curriculum constraints arising in course allocation. We then apply the method to find allocations that are ex ante fair and efficient in the presence of these constraints. The method can also be applied to certain two-sided matching problems to produce a fair matchup design in interleague games and speed dating.

KEYWORDS: Market Design, Random Assignment, Birkhoff-von Neumann Theorem, Probabilistic Serial, Pseudo-Market, Utility Guarantee, Assignment Messages.

1. INTRODUCTION

Lotteries are commonplace in everyday resource allocation. Lotteries are used to break ties among students applying for overdemanded public schools and for popular after-school programs, to ration offices, parking spaces, and tasks among employees, to allocate courses and university dormitories among college students, and to assign jury and military duties

Date: October 19, 2010.

Budish: University of Chicago Booth School of Business, 5807 S Woodlawn Ave, Chicago IL 60637, eric.budish@chicagobooth.edu. Che: Department of Economics, Columbia University, 420 West 118th Street, 1016 IAB New York, NY 10027, and YERI, Yonsei University, yeonkooche@gmail.com. Kojima: Department of Economics, Stanford University, CA 94305, fuhitokojima1979@gmail.com. Milgrom: Department of Economics, Stanford University, CA 94305, paul@milgrom.net. We are grateful to Emir Kamenica, Sebastien Lahaie, Mihai Manea, Herve Moulin, Tayfun Sönmez, Al Roth, Utku Ünver, and seminar participants at Boston College, Gakushuin, Johns Hopkins, Princeton, Tokyo, UC San Diego, Waseda, Yahoo! Research, and Yale for helpful comments. We especially thank Tomomi Matsui, Akihisa Tamura, Jay Sethuraman, and Rakesh Vohra. Yeon-Koo Che is grateful to Korea Research Foundation for World Class University Grant, R32-2008-000-10056-0. Paul Milgrom's research was supported by NSF grant SES-0648293.

among citizens.[1] The ubiquitous "first-come first-served" method, or "queuing," is often only a less apparent way to include random elements in setting a priority order. Lotteries are sensible in these examples and many others because the objects to be assigned are indivisible and monetary transfers are limited or unavailable.[2] In these circumstances, any assignment of resources is likely to be unfair ex post, and, absent monetary transfers, randomization promotes a perception of ex ante fairness that makes the ex post unfairness more acceptable to participants. Naturally, lotteries figure prominently in market design, for these two features are present in many markets.

To find a desirable random allocation, it is useful to view each agent as directly consuming a lottery of goods.[3] Since these **random assignments**—the lotteries of goods received by agents—are "divisible" in probability units, one can then apply the classical frameworks developed for divisible goods. Hylland and Zeckhauser (1979) ("HZ") were the first to apply this perspective to market design. Their pseudo-market mechanism endows agents with (the same) fixed budget in a fictitious currency, asks them to expend their budget to "buy" probability shares of alternative goods, and the mechanism finds a competitive equilibrium by solving for prices (per unit probability of obtaining each good) that clear the market. The resulting allocation is then ex ante efficient by the first welfare theorem. Bogomolnaia and Moulin (2001) ("BM") also adopted the random assignment approach to find an allocation that is fair and efficient in the ordinal sense.[4]

[1] Lotteries played historical roles in assigning public lands to homesteaders (Oklahoma Land Lottery of 1901), and radio spectra to broadcasting companies (FCC assignment of radio frequencies during 1981-1993). Lotteries are also used annually to select 50,000 winners of the US permanent residency visas ("green cards") from those qualified in the DoJ's immigration diversity program.

[2] The limitation of monetary transfers arises from moral objection to "commoditizing" objects such as human organs and from fairness consideration. Assignment of resources based on prices often favor those best endowed with money rather than those most deserving, and can be regarded as unfair for many goods and services. See Che and Gale (2008) for making this point based on utilitarian efficiency.

[3] The conventional perspective focuses on agents' ex post assignment of goods rather than the ex ante lotteries they receive. That is, a mechanism typically identifies ex post desirable allocations and, to ensure ex ante fairness, randomizes among such allocations. The random serial dictatorship (RSD), widely used in practice, is a case in point. In RSD, the agents are randomly ordered, and, following that order, each agent is assigned his/her most preferred object not yet assigned. This resulting assignment is ex post efficient and ex ante fair, but is known to entail ex ante efficiencies even in the ordinal sense (see Bogomolnaia and Moulin (2001)). Other well known mechanisms such as Gale and Shapley's deferred acceptance and top trading cycles follow this conventional approach and suffer similar ex ante inefficiencies, when ties arise and need to be broken randomly.

[4] BM's Probabilistic Serial mechanism takes agents' ordinal preference orders as input and runs an "eating" algorithm, whereby each agent consumes probability shares of alternative goods at unit speed

The random assignment method, such as HZ's and BM's mechanisms, has been developed so far in a simple setting in which n indivisible objects are assigned to n agents, one for each agent. The purpose of the current paper is to broaden the random assignment methodology (including HZ and BM) to enhance its practical applicability. For these mechanisms to be applicable to practical assignment problems, one should be able to extend them to accommodate a variety of constraints and features present in practice. First of all, the model itself need to be generalized. For instance, the mechanism must allow for agents or objects to be unassigned, a possibility in school choice. For course allocation problems, since a single course accepts multiple students, and a student takes multiple courses, the extensions should accommodate many-to-many matchings.

More challenging are a variety of other constraints that practical mechanisms often need to satisfy. A case in point is the so-called "controlled-choice" in school assignment. Schools often seek to balance their student bodies in terms of gender, ethnicity, race, test scores, and the geographic location of students' residence. For instance, public schools in Massachusetts are discouraged by the Racial Imbalance Law from having student enrollments that are more than 50% minority. Miami-Dade County Public Schools control for the socioeconomic status of students in order to diminish concentrations of low-income students at certain schools. In New York City, "Educational Option" (EdOpt) schools must balance their student bodies in terms of students' test scores.[5] Public schools in Seoul restrict the number of seats for those students residing in distant school districts, in order to alleviate morning commutes. In a course allocation problem, a student may wish to enroll in no more than a certain number of courses in a given subject (curriculum constraints) or in a given time slot (scheduling constraints). Finally, the objects may not exist in fixed supply but may be adjusted based on the demands. For instance, schools assign their students to different foreign language programs, the exact composition of which is adjustable to a degree determined by the available staff and resource.

The presence of these constraints raises a methodological issue for the random assignment method. Since the constraints apply to realized ex post assignments, it is not clear how they restrict the set of feasible random assignments. In one direction, it is obvious

in the order of their preferences, starting with the most preferred object. The total cumulative shares at time 1 then become random assignments. The resulting random assignment is ex ante efficient in the ordinal sense, namely, it is not dominated by any other random assignment in the sense of first-order stochastic dominance.

[5]In particular, 16 percent of students that attend an EdOpt school must score above grade level on the standardized English Language Arts test, 68 percent must score at grade level, and the remaining 16 percent must score below grade level (Abdulkadiroglu, Pathak, and Roth, 2005).

that any lottery over deterministic assignments that each satisfying a set of constraints must induce a random assignment which also satisfies the same constraints, but what about the reverse implication? Namely, if a random assignment satisfies a set of constraints, can it be implemented by a joint lottery over deterministic assignments each of which satisfies the same constraints? In the simple one-to-one matching environment of HZ and BM, this issue is resolved by the celebrated Birkhoff-von Neumann theorem (Birkhoff, 1946; von Neumann, 1953). To what extent this theorem can be generalized to handle more complicated constraints is unclear.

The first part of the paper addresses this methodological challenge by maximally generalizing the Birkhoff-von Neumann theorem. We consider a general model of indivisible goods that allows for many-to-one and many-to-many matchings, as well as unassignment. We then characterize an underlying structure of constraints—expressed in terms of sets whose assignment are subject to maximal or minimal quotas—that is sufficient for a random assignment to be implementable. We next provide a polynomial time algorithm that implements any random assignment in the general environment satisfying the sufficient condition. Last, we show that our sufficient condition is also necessary in typical bilateral matching environments.

The second part of the paper applies the random assignment methodology to specific market design contexts. Our first application is a generalization of BM's Probabilistic Serial mechanism, to accommodate new kinds of supply-side constraints that may be important in unit-demand applications such as school choice and dormitory assignment. We show how to modify BM's algorithm to accommodate constraints such as controlled choice and adjustable capacities and prove that the attractive properties of BM's algorithm extends to this more general environment.

Our second application is a generalization of HZ's pseudo-market mechanism, to accommodate new kinds of demand-side constraints that may express important aspects of participants' preferences in multi-unit demand applications such as course allocation and the assignment of supplier leads. We enrich the messages of the agents, via the "assignment messages" developed by Milgrom (2009), to allow them to express constraints such as scheduling and curricular constraints ("I want just one course in finance and at most one course before noon") arising in course allocation, and to express nonlinear preferences such as diminishing marginal utilities for an item or category ("the second finance course is worth less to me than the first"), and we imbed these enriched messages to develop a generalized multi-unit pseudo-market mechanism. We then establish existence of competitive equilibrium prices in the pseudo-market, and invoke our sufficiency result to ensure implementability of the random assignment that results from this competitive equilibrium. We finally show that this generalization inherits the attractive efficiency and fairness properties of the original, which had limited attention to the case of unit demand. This mechanism may be useful for practice, especially because several multi-unit assignment mechanisms currently used in practice have been shown to have important flaws with respect to both efficiency and fairness (Sonmez and Unver 2010; Budish and Cantillon, 2009).

Finally, our implementation result has an unexpected application for ensuring ex post fairness in multi-unit resource allocation. When agents demand multiple goods there can be many ways to implement a given random assignment, some of which may be fairer than others. For instance, suppose there are two agents, 1 and 2, dividing four objects, a, b, c, and d, which they prefer in the order listed. An ex ante fair allocation would be to assign each agent each good with probability 0.5. One way to implement this outcome is to assign a and b to 1 and c and d to 2 with probability one half and a and b to 2 and c and d to 1 with the remaining probability one half. While ex ante fair, this implementation will be ex post unfair since one agent always gets two best and the other gets the two worst. There is another implementation of the same random assignment that is ex post fair: whenever one agent gets one of the two best objects, he must also get one of the two worst objects. It turns out our implementation result can be utilized to ensure this latter implementation, more generally to ensure that every realized assignment chosen to implement a random assignment approximiates the original random assignment in terms of expected utilities. This procedure can be applied in the context of course allocation e.g., in conjunction with our generalization of HZ, or in other multi-unit demand environments such as task assignment and fair division of estates. The utility guarantee can also be adapted to a two-sided matching problem, in which both sides of the market are agents. Starting with any random matching, we can introduce ex post utility guarantees on both sides, ensuring ex post utility levels that are close to the promised ex ante levels. This method can be used, for example, to design a fair schedule of inter-league sports matchups or a fair speed-dating mechanism.

The rest of the paper is organized as follows. Section 2 presents the model. Section 3 presents the sufficiency and necessity results for implementing random assignments. Section 4 presents the generalization of Bogomolnaia and Moulin's (2001) Probabilistic Serial mechanism, for applications such as school choice. Section 5 presents the generalization of Hylland and Zeckhauser's (1979) Pseudo-market mechanism, for applications such as course allocation. Section 6 presents the utility guarantee results, including the application to two-sided matching. Section 7 collects some negative results for non-bilateral matching environments. Section 8 presents the algorithm for implementing random assignments. Section 9 concludes.

2. Setup

An **environment** is a tuple $\mathcal{E} = \langle N, O, \mathcal{H}, q\rangle$ where N and O are sets of agents and objects where $|N|, |O| \geq 2$; $\mathcal{H}$ is a set of subsets of $N \times O$ that includes all singletons (we call $\mathcal{H}$ a **constraint structure**); and $q = (\underline{q}_S, \overline{q}_S)_{S\in\mathcal{H}}$ is the set of quotas associated with each set in $\mathcal{H}$. We call $\underline{q}_S$ the **floor constraint** and $\overline{q}_S$ the **ceiling constraint** for S. For each $S \in \mathcal{H}$, we assume $\underline{q}_S \in \mathbb{Z} \cup \{-\infty\}$ and $\overline{q}_S \in \mathbb{Z} \cup \{\infty\}$, where $\mathbb{Z}$ is the set of integers.

A (generalized) **random assignment** is a $|N| \times |O|$ matrix $P = [P_{ia}]$ where $P_{ia} \in (-\infty, \infty)$ for all $i \in N, a \in O$. A **deterministic assignment** is a random assignment P each of whose entries is an integer. Note that we allow for assigning more than one unit of a good and even for assigning a negative amount of a good. One interpretation of receiving a negative amount of a good is supplying the good. Given environment $\mathcal{E} = \langle N, O, \mathcal{H}, q\rangle$, P is said to be **feasible in $\mathcal{E}$** if

$$\underline{q}_S \leq P_S \leq \overline{q}_S, \text{ for all } \quad S \in \mathcal{H},$$

where we define

$$P_S := \sum_{(i,a)\in S} P_{ia},$$

for any random assignment P and $S \in \mathcal{H}$.

Definition 1. *The constraint structure $\mathcal{H}$ is **decomposable** if, for any random assignment P and any quotas $(\underline{q}_S, \overline{q}_S)_{S\in\mathcal{H}}$, such that $\underline{q}_S \leq P_S \leq \overline{q}_S$ for all $S \in \mathcal{H}$, there exist $\lambda^1, \ldots, \lambda^K$ and $P^1, \ldots, P^K$ such that*

(1) $P = \sum_{k=1}^K \lambda^k P^k$,
(2) $\lambda^k > 0, k = 1, \ldots, K$, *and* $\sum_{k=1}^K \lambda^k = 1$,
(3) $\underline{q}_S \leq P^k_S \leq \overline{q}_S$ *for each* $k = 1, \ldots, K$ *and* $S \in \mathcal{H}$,
(4) P^k_{ia} *is an integer for each* (i, a).

If $\mathcal{H}$ is decomposable, then every P satisfying all the given constraints in $\mathcal{H}$ can be expressed as a convex combination of deterministic assignments satisfying the constraints. In other words, any random assignment satisfying the constraints in $\mathcal{H}$ can be implemented as a lottery over deterministic outcomes each of which respects constraints in $\mathcal{H}$.

Decomposability of a constraint structure has another, more convenient, formulation. Since $\mathcal{H}$ is decomposable only if conditions (1)-(4) hold for *any* matrix P and quotas $\{q_S\}$ satisfying the feasibility inequalities, the conditions must hold in particular for quotas $\underline{q}_S = \lfloor P \rfloor$ and $\overline{q}_S = \lfloor P \rceil$.[6] Conversely, for any given P, a decomposition that satisfies (1)-(4) with quotas $\underline{q}_S = \lfloor P \rfloor$ and $\overline{q}_S = \lfloor P \rceil$ also satisfies the conditions for every other vector of quotas such that P is feasible. Consequently, $\mathcal{H}$ is decomposable if and only if for every random assignment P there exist $\lambda^1, \ldots, \lambda^K$ and $P^1, \ldots, P^K$ such that

(1) $P = \sum_{k=1}^{K} \lambda^k P^k$,
(2) $\lambda^k > 0, k = 1, \ldots, K$, and $\sum_{k=1}^{K} \lambda^k = 1$,
(3) $P_S^k \in \{\lfloor P_S \rfloor, \lceil P_S \rceil\}$ for all $k \in \{1, \ldots, K\}$ and $S \in \mathcal{H}$.

This alternative formulation requires that each assignment in the decomposition rounds the random assignment either up or down to the nearest integer, with respect to every constraint set.

Note that our notion of decomposability is defined for a constraint structure without reference to specific quotas. This approach has at least two advantages. First, our formulation produces a sharp result: A necessary and sufficient condition for decomposability will be obtained in terms of constraint structures. Second, and more importantly, specific quotas may vary over time or with market conditions, so it is desirable to characterize a robust condition under which a random mechanism is guaranteed to work for any possible quota.

3. Theory of Implementing Random Assignments

A constraint structure $\mathcal{H}$ is a **hierarchy** if $S \subset S'$ or $S' \subset S$ or $S \cap S' = \emptyset$ for every $S, S' \in \mathcal{H}$.[7] The following concept plays a central role in the rest of this paper.

Definition 2. *A set $\mathcal{H} \subset 2^{N \times O}$ is a* ***bihierarchy*** *if there exist $\mathcal{H}_N$ and $\mathcal{H}_O$ such that*

(1) *$\mathcal{H} = \mathcal{H}_N \cup \mathcal{H}_O$ and $\mathcal{H}_N \cap \mathcal{H}_O = \emptyset$, that is, $\mathcal{H}_N$ and $\mathcal{H}_O$ partition $\mathcal{H}$, and*
(2) *$\mathcal{H}_N$ and $\mathcal{H}_O$ are hierarchies.*

In many applications $\mathcal{H}_N$ and $\mathcal{H}_O$ include, respectively, sets of the form $\{i\} \times O$ and $N \times \{a\}$ where $i \in N$, $a \in O$. These sets represent constraints imposed on each agent and object, thus the mnemonic notation $\mathcal{H}_N$ and $\mathcal{H}_O$. However, at this point we do not impose such a restriction, and we will state the restriction whenever applicable. (Partitions of

[6] For any $x \in \mathbb{R}$, $\lfloor x \rfloor$ and $\lceil x \rceil$ are the largest integer no larger than x and the smallest integer no smaller than x, respectively.

[7] A hierarchy is called a laminar family in the combinatorial optimization literature.

the sets $\mathcal{H}$ into $\mathcal{H}_N$ and $\mathcal{H}_O$ need not be unique, either. For instance, singleton sets can be partitioned into the two families in any arbitrary fashion.)

Theorem 1. (ROUNDING THEOREM) *If $\mathcal{H}$ is a bihierarchy, then it is decomposable.*

Theorem 1 shows that any random assignment P can be decomposed into matrices where the sum of the entries within each element of the bihierarchy is rounded up or down to the nearest integer. It follows immediately from results of Hoffman and Kruskal (1956) and Edmonds (1970). For the proof see Appendix A.

3.1. **Corollary: the Birkhoff-von Neumann Theorem.** We denote the environment studied by Birkhoff and von Neumann by $\mathcal{E}^{BvN} \equiv \langle N, O, \mathcal{H}^{BvN}, q\rangle$ where

$$\begin{aligned}
&\mathcal{H}^{BvN} = \{\{(i,a)\}|(i,a) \in N \times O\} \cup \{\{i\} \times O | i \in N\} \cup \{N \times \{a\} | a \in O\} \\
&\underline{q}_{\{(i,a)\}} = 0, \overline{q}_{\{(i,a)\}} = 1, \quad \text{for all } (i,a) \in N \times O, \\
&\underline{q}_S = \overline{q}_S = 1, \quad \text{for all } S \in \mathcal{H} \setminus \{\{(i,a)\}|(i,a) \in N \times O\}.
\end{aligned}$$

This is an environment in which each agent receives exactly one object and each object is allocated to exactly one agent, and no other constraints are imposed. $\mathcal{H}^{BvN}$ is a bihierarchy since it can be partitioned, for example, into $\mathcal{H}^{BvN}_N$ and $\mathcal{H}^{BvN}_O$ where

$$\begin{aligned}
\mathcal{H}^{BvN}_N &:= \{\{(i,a)\}|(i,a) \in N \times O\} \cup \{\{i\} \times O | i \in N\}, \\
\mathcal{H}^{BvN}_O &:= \{N \times \{a\} | a \in O\},
\end{aligned}$$

and clearly $\mathcal{H}^{BvN}_N$ and $\mathcal{H}^{BvN}_O$ are hierarchies.

A random assignment feasible in $\mathcal{E}^{BvN}$ is called a **bistochastic matrix** or a doubly stochastic matrix. Equivalently, P is a bistochastic matrix if

(1) $P_{ia} \geq 0$ for all $i \in N$ and $a \in O$,
(2) $\sum_{a \in O} P_{ia} = 1$ for all $i \in N$, and
(3) $\sum_{i \in N} P_{ia} = 1$ for all $a \in O$.

An integer-valued bistochastic matrix is called a permutation matrix. Since $\mathcal{H}^{BvN}$ is a bihierarchy, the following Birkhoff-von Neumann Theorem is an immediate corollary of the Rounding Theorem.

Corollary 1. (BIRKHOFF, 1946; VON NEUMANN, 1953) *Any bistochastic matrix can be written as a convex combination of permutation matrices.*

3.2. **Necessity of a bihierarchical constraint structure.** Theorem 1 shows that bihierarchy is sufficient for decomposition. This section examines the sense in which it is necessary. Doing so also provides an intuition about the role bihierarchy plays for implementation of random assignments. We begin with an example of a non-bihierarchical constraint structure that is not decomposable.

Example 1. *Consider the following environment with 2 goods and 2 agents and the constraint structure*

$$\mathcal{H} = \{\{(1,a),(1,b)\},\{(1,a),(2,a)\},\{(1,b),(2,a)\}\}.$$

Clearly, $\mathcal{H}$ is not a bihierarchy. Suppose each set in $\mathcal{H}$ has a common floor and ceiling quota of one. The following random assignment

$$P = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

cannot be decomposed into feasible deterministic assignments. To see this first observe that, for any convex decomposition of P, there exists P^k that is part of the decomposition of P with $P^k_{1a} = 1$. Since the constraint set $\{(1,a),(1,b)\}$ has a quota of one, it follows that $P^k_{1b} = 0$. Since the quota of one binds for $\{(1,b),(2,a)\}$, it follows that $P^k_{2a} = 1$. This is a contradiction because $P^k_{\{(1,a),(2,a)\}} = P^k_{1a} + P^k_{2a} = 2$ violates the quota for $\{(1,a),(2,a)\}$, which is one.

Example 1 suggests that the failure of decomposability is caused by a "cycle" of an odd number formed by constraint sets. In the above example, for instance, a cycle formed by three constraint sets $\{(1,a),(1,b)\},\{(1,a),(2,a)\},\{(1,b),(2,a)\}$ leads to a situation where at least one of the constraints is violated. Generalizing this idea, we say that a sequence $(S_1,\dots,S_l) \in \mathcal{H}^l$ is an **odd cycle** if $S_i \neq S_j$ for all $i \neq j$, l is odd, and there exists a sequence $(x_1,\dots,x_l) \in (N \times O)^l$ such that for each $i = 1,\dots,l$, $x_i \in S_i \cap S_{i+1}$ and $x_i \notin S_j$ for any $j \neq i, i+1$, where subscript $l+1$ is understood to be 1. An argument generalizing the above example yields the following (a formal proof is in the Appendix).

Lemma 1. (Odd Cycles) *If $\mathcal{H}$ contains an odd cycle, then $\mathcal{H}$ is not decomposable.*

An important role of the bihierarchy is to rule out odd cycles. To see this, suppose that $\mathcal{H} = \mathcal{H}_N \cup \mathcal{H}_O$ is a bihierarchy that contains an odd cycle, $\{S_1,\dots,S_l\}$. Assume without loss $S_1 \in \mathcal{H}_N$. Then, S_2 must belong to $\mathcal{H}_O$, since $S_1 \cap S_2 \neq \emptyset$ and neither is a subset of the other (since $x_2 \in S_2 \setminus S_1$ and $x_l \in S_1 \setminus S_2$). Arguing in the same fashion, S_3 must be in $\mathcal{H}_N$, S_4 in $\mathcal{H}_O$, $\dots$, and S_l must be in $\mathcal{H}_N$ since l is an odd number. But $S_l \cap S_1 \neq \emptyset$

and neither is a subset of the other. So $\mathcal{H}_N$ cannot be a hierarchy, and $\mathcal{H}$ cannot be a bihierarchy, a contradiction.

Is bihierarchy necessary for decomposition? The next example suggests that this is not the case.[8]

Example 2. Consider an environment with 2 goods and 2 agents as before, but let

$$\mathcal{H} = \{\{(1,a),(1,b)\},\{(1,a),(2,a)\},\{(1,a),(2,b)\}\},$$

and the floor and ceiling quotas for each constraint set be one. Any feasible random assignment

$$P = \begin{pmatrix} s & t \\ t & t \end{pmatrix},$$

with $s+t=1$, can be decomposed by a convex combination of deterministic assignments as

$$P = \begin{pmatrix} s & t \\ t & t \end{pmatrix} = s\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + t\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Note that the constraint structure does not allow for an odd cycle although it is not a bihierarchy.

Notice, however, that Example 2 is somewhat non-standard in that some row and column constraints are not present. Decomposability would fail if all row and column constraints are added to the constraint structure of Example 2 as the new constraint structure has an odd cycle. This observation turns out to be true more generally. We show that *bihierarchy is in fact necessary for decomposability in an important sense* — namely, whenever all the "standard" constraint sets are present.

Theorem 2. (Necessity) *Suppose $\mathcal{H}^{BvN} \subset \mathcal{H}$. If $\mathcal{H}$ is not a bihierarchy, then it is not decomposable.*

Recall that the condition $\mathcal{H}^{BvN} \subset \mathcal{H}$ is natural in bilateral matching settings and is imposed in all applications in this paper.

The formal proof of Theorem 2 is in the Appendix. The basic strategy of the proof is to show that there exists an odd cycle whenever $\mathcal{H} \supset \mathcal{H}^{BvN}$ is not a bihierarchy.

[8] The argument presented in the example does not *show* that the constraint structure is decomposable since it limits attention to just a single set of quota constraints. It turns out that the dual of the constraint structure (defined in Section 7) forms a bihierarchy, a condition that Theorem 5 shows is sufficient for decomposability.

Remark 1. *In light of Examples 1 and 2, one might wonder whether an absence of odd cycles is sufficient for decomposability. This turns out to be false. Consider* $\mathcal{H} = \{\{(1,a),(1,b)\},\{(1,a),(2,a)\},\{(1,a),(2,b)\},\{(1,a),(1,b),(2,a),(2,b)\}\}$. *This structure does not contain an odd cycle (and it is not a bihierarchy). Assume the quota for each of the first three sets is one and the quota for the last set is two. One can check that the random assignment* P *above cannot be feasibly decomposed.*

In Section 7 we consider a more general environment than the bilateral setting, and present a further generalization of the Rounding Theorem to that environment. We also use Lemma 1 to show that the decomposition result cannot be extended to any multilateral matching of more than two kinds of agents or to roommate matching.

3.3. **Examples of Bihierarchy.** As discussed above, $\mathcal{H}^{BvN}$ is a bihierarchy. This section discusses more examples.

3.3.1. *Flexible Capacity.* Consider a school choice problem in which the school authority wishes to run several education programs within one building. Several capacity constraints can be represented using a hierarchy $\mathcal{H}_O$ containing sets of the form $S = N \times O'$. The ceiling $\overline{q}_S$ then describes the total capacity that can be allocated within O', which can apply to a program or a set of programs. Notice that the hierarchical structure $\mathcal{H}_O$ allows for the nested constraints on program sizes.

3.3.2. *Group-specific Quotas.* Affirmative action policies are sometimes implemented as quotas on students fitting specific gender, racial, or economic profiles.[9] A similar mathematical structure results from New York City's Educational Option programs, which achieve a mix of students by imposing quotas on students with test scores.(Abdulkadiroğlu, Pathak, and Roth, 2005). Quotas may be based on the residence of applicants as well: The school choice program set to begin in 2010 in Seoul, Korea, limits the percentage of seats allocated to the applicants from outside the district.[10] and a number of school choice programs in Japan have similar quotas based on residential areas as well.

Such quotas can be incorporated by $\mathcal{H}_O$ containing sets of the form $N' \times \{a\}$ for $a \in O$ and $N' \subsetneq N$. The ceiling $\overline{q}_{N' \times \{a\}}$ then determines the maximum number of agents school a can admit from group N'. Quotas on multiple groups can be imposed for each a without violating a hierarchical structure of $\mathcal{H}_O$ as long as they do not overlap with each other.

[9]Abdulkadiroğlu and Sönmez (2003) and Abdulkadiroğlu (2005) analyze assignment mechanisms under affirmative action constraints.

[10]See "Students' High School Choice in Seoul Outlined," Digital Chosun Ilbo, October 16, 2008 (http://english.chosun.com/w21data/html/news/200810/200810160016.html).

Moreover, a nested series of constraints can be accommodated. For instance, a school system can require that a school admit at most 50 students from district one, at most 50 students from district two, and at most 80 students from either district one or two.

It is also possible to accomodate both flexible-capacity constraints and group-specific quota constraints within the same hierarchy $\mathcal{H}_O$. Flexible-capacity constraints are defined on multiple columns of a random assignment matrix P, whereas group-specific quota constraints are defined on subsets of single columns of P. Any subset of a single column will be a subset of or disjoint from any set of multiple columns.

3.3.3. *Course Allocation.* The course allocation problem begins with a set of students and courses. Each student may enroll in multiple courses, but cannot receive more than one seat in any single course. Moreover, each student may have preference or feasibility constraints that limit the number of courses taken from certain sets. For example, scheduling constraints prohibit any student from taking two courses that meet during the same time slot. Or, a student might prefer to take at most two courses on finance, at most three on marketing, and at most four on finance or marketing in total.

Many such restrictions can be modeled using a bihierarchy such that $\mathcal{H}_N \supset \mathcal{H}_N^{BvN}$. Setting $\overline{q}_{\{(i,a)\}} = 1$ and $\overline{q}_{\{i\}\times O} > 1$ for each $i \in N$ and $a \in O$ ensures that each student i can enroll in multiple courses but be assigned to at most one seat in each course. Letting F and M be finance courses and marketing courses, if $\mathcal{H}_N$ contains $\{i\} \times F$,$\{i\} \times M$ and $\{i\} \times (F \cup M)$, then we can express the constraints "student i can take at most $\overline{q}_{\{i\}\times F}$ courses in finance, $\overline{q}_{\{i\}\times M}$ courses in marketing, and $\overline{q}_{\{i\}\times (F\cup M)}$ in finance and marketing combined." Scheduling constraints are handled similarly; for instance, F and M are sets of classes offered at different times (e.g., Friday morning and Monday morning). It may be impossible, however, to express both subject and scheduling constraints while still maintaining a bihierarchy constraint structure.

Note that the flexible production and group-specific quota constraints described in Sections 3.3.1-3.3.2 can also be incorporated into the course allocation problem without jeopardizing the bihierarchical structure. These constraints pertain to $\mathcal{H}_O$, while the preference and scheduling constraints described above pertain to $\mathcal{H}_N$. So long as $\mathcal{H}_N$ and $\mathcal{H}_O$ are each hierarchies, $\mathcal{H} = \mathcal{H}_N \cup \mathcal{H}_O$ is a bihierarchy.

3.3.4. *Interleague Play.* Some professional sports associations, including Major League Baseball (MLB) and the National Football League (NFL), have two separate leagues. In MLB, teams in the American League (AL) and National League (NL) had traditionally played against teams only within their own league during the regular season, but play across the AL and NL, called interleague play, was introduced in 1997.[11] Unlike the intraleague games, the number of interleague games is relatively small, and this can make the indivisibility problem particularly difficult to deal with in designing the matchups. For example, suppose there are two leagues, N and O, each with 9 teams. Suppose each team must play 15 games against teams in the other league. There are some matchup constraints: Each team in N has a geographic rival in O, and they must play twice. For fairness reasons, teams in each league must face opponents in the other league of similar difficulty. Specifically, one could require each team to play at least 4 games with the top 3 teams, 4 games with the middle 3 teams and 4 games with the bottom 3 teams of the other league. It is not difficult to see that the resulting constraint structure form a bihierarchy.

4. A Generalization of Bogomolnaia and Moulin's Probabilistic Serial Mechanism for Assignment with Single-unit Demand

In this section, we consider a problem of assigning indivisible objects to agents who can consume at most one object each. Examples include university housing allocation, public housing allocation, office assignment, and student placement in public schools.

Formally, consider an environment $\mathcal{E} = \langle N, O, \mathcal{H}, \{\underline{q}_S, \overline{q}_S\}_{S\in\mathcal{H}}\rangle$, where the bihierarchy $\mathcal{H} = \mathcal{H}_N \cup \mathcal{H}_O$ has the following structure:

$$\mathcal{H}_N = \{\{(i,a)\}|(i,a) \in N \times O\} \cup \{\{i\} \times O | i \in N\}$$

and

$$N \times \{a\} \in \mathcal{H}_O, \forall a \in O,$$

with constraints such that $\underline{q}_{\{i\}\times O} = \overline{q}_{\{i\}\times O} = 1$ for all $i \in N$ and $\underline{q}_S = 0$ for all $S \in \mathcal{H} \setminus \{\{i\} \times O | i \in N\}$. The maximum quota for each object a, $\overline{q}_{N\times\{a\}}$, can be arbitrary, unlike in $\mathcal{E}_{BvN}$. It is understood also that O contains a **null object** $\emptyset$ with unlimited supply, that is, $\overline{q}_S = +\infty$ for any $S \in \mathcal{H}_O$ with $(N \times \{\emptyset\}) \cap S \neq \emptyset$.

As mentioned earlier, the bihierarchy structure in this section accommodates a range of practical situations faced by a mechanism designer. First, the objects may be produced endogenously based on the reported preferences of the agents, as in the case of school choice with flexible capacity (Section 3.3.1). Second, a mechanism designer may need to treat different groups of agents differently in assignment, as in the case of school choice with group-specific quotas (Section 3.3.2).

[11]See "Interleague play", Wikipedia (http://en.wikipedia.org/wiki/Interleagueplay).

To formally study the problem of assignment, we need to introduce the preferences of agents. Each agent $i \in N$ has a **strict preference** $\succ_i$ over O. We write $a \succeq_i b$ if either $a \succ_i b$ or $a = b$ holds. When N and O are fixed, we write $\succ$ for $(\succ_i)_{i \in N}$, $\succ_{N'}$ for $(\succ_i)_{i \in N'}$ where $N' \subset N$. A quadruple $\Gamma = (N, O, \mathcal{H}, \{\underline{q}_S, \overline{q}_S\}_{S \in \mathcal{H}}, (\succ_i)_{i \in N})$ then defines a **random assignment problem**. Recall that a feasible random assignment is a matrix $P = [P_{ia}]_{i \in N, a \in O}$ satisfying $\underline{q}_S \leq \sum_{(i,a) \in S} P_{ia} \leq \bar{q}_S$ for each $S \in \mathcal{H}$. Recall that $\mathcal{P}_{\mathcal{E}}$ denotes the set of all random assignments feasible in $\mathcal{E} = \langle N, O, \mathcal{H}, q \rangle$. A special case of random assignment is a **deterministic assignment**, represented by a matrix $P \in \mathcal{P}_{\mathcal{E}}$ with $P_{ia} \in \{0, 1\}$ for each $(i, a) \in N \times O$.

A solution used in many applications is **random priority** (Bogomolnaia and Moulin, 2001), also called **random serial dictatorship** (Abdulkadiroğlu and Sönmez, 1999), originally studied under environment $\mathcal{E}^{BvN}$. In the current setup, we define random priority as follows: (i) randomly order agents with equal probability, and (ii) the first agent obtains her favorite object, the second agent obtains her favorite object among the remaining objects, and so on, *as long as* allocating the good to agents so far is consistent with all ceiling constraints. In the environment $\mathcal{E}^{BvN}$, it is a familiar mechanism that orders agents uniform randomly, and assigns the first agent her favorite object, the second agent her favorite remaining object, and so forth.

While the random priority is a popular mechanism, it may result in efficiency loss (Bogomolnaia and Moulin, 2001). To see how the loss of efficiency may occur in our context, consider the following example, adapted from Bogomolnaia and Moulin (2001).

Example 3 (Random priority may result in a suboptimal production plan)**.** Let $N = \{1, 2, 3, 4\}$, $O = \{a, b, c, \emptyset\}$, $\mathcal{H}_O = [\bigcup_{a' \in O} \{N \times \{a'\}\}] \cup \{N \times \{a, b, c\}\}$. Assume $\overline{q}_{N \times \{a\}} = \overline{q}_{N \times \{b\}} = \overline{q}_{N \times \{c\}} = 1, \overline{q}_{N \times \{a,b,c\}} = 2$. This is a situation in which each good has individual quota of one, and furthermore only two out of three goods can actually be produced.

Let

$\succ_1$: $a, b, \emptyset,$

$\succ_2$: $a, b, \emptyset,$

$\succ_3$: $c, b, \emptyset,$

$\succ_4$: $c, b, \emptyset,$

where the notation means that agent one prefers a to b to $\emptyset$, and so on. Under the random priority mechanism, the assignment

$$RP = \begin{pmatrix} 5/12 & 1/12 & 0 & 1/2 \\ 5/12 & 1/12 & 0 & 1/2 \\ 0 & 1/12 & 5/12 & 1/2 \\ 0 & 1/12 & 5/12 & 1/2 \end{pmatrix},$$

will be obtained.[12] The following random assignment is preferred by everyone.

$$P' = \begin{pmatrix} 1/2 & 0 & 0 & 1/2 \\ 1/2 & 0 & 0 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \end{pmatrix}.$$

One notable feature of the random priority mechanism is that, under this mechanism, good b is produced although everyone prefers some other good, a or c, to be produced. Good b is produced either when agent 1 and 2 get highest priorities or when agent 3 and 4 get highest priorities. All agents will be made better off if the social planner can first decide to produce a and c and then allocate the goods: It is easy to see that random assignment P' results if the random priority is conducted after the production plan is fixed to producing a and c. Of course, such a production plan is inefficient if agents prefer b to other goods. Thus a good mechanism may be one that simultaneously decides production of goods as well as the allocation of them, based on preference information reported by agents.

We begin by introducing the efficiency concept in our setup, called ordinal efficiency. A random assignment P **ordinally dominates** another random assignment $P' \in \mathcal{P}_{\mathcal{E}}$ **at** $\succ$ if for each agent i the lottery P_i first-order stochastically dominates the lottery P'_i, that is,

$$\sum_{b \succeq_i a} P_{ib} \geq \sum_{b \succeq_i a} P'_{ib} \; \forall i \in N, \forall a \in O,$$

[12]Given $N = \{1,2,3,4\}$, $O = \{a,b,c,\emptyset\}$ and random assignment P, we write

$$P = \begin{pmatrix} P_{1a} & P_{1b} & P_{1c} & P_{1\emptyset} \\ P_{2a} & P_{2b} & P_{2c} & P_{2\emptyset} \\ P_{3a} & P_{3b} & P_{3c} & P_{3\emptyset} \\ P_{4a} & P_{4b} & P_{4c} & P_{4\emptyset} \end{pmatrix}.$$

Similar notation will be used elsewhere in this paper as well.

with strict inequality for some i, a (we say that P_i weakly stochastically dominates P'_i when either P_i stochastically dominates P'_i or $P_i = P'_i$). If P ordinally dominates P' at $\succ$, then every agent i prefers P_i to P'_i according to any expected utility function with utility index consistent with $\succ_i$. The random assignment $P \in \mathcal{P}_{\mathcal{E}}$ is **ordinally efficient at** $\succ$ if it is not ordinally dominated at $\succ$ by any other random assignment in $\mathcal{P}_{\mathcal{E}}$. Note that our model allows for a variety of constraints, so the current notion has the flavor of "constrained efficiency" in that the efficiency is defined within the set of assignments satisfying the constraints.

Now we introduce the **generalized probabilistic serial** mechanism, which is a generalization of the mechanism proposed by Bogomolnaia and Moulin to our setting. The idea is to regard each object as a divisible object of "probability shares." Each agent "eats" the best available object with speed one at every time $t \in [0, 1]$. The resulting profile of shares of objects eaten by agents by time 1 obviously corresponds to a random assignment matrix, which we call the **generalized probabilistic serial random assignment**.

Before giving a formal definition, note that we will need to modify the definition of the algorithm from the version of Bogomolnaia and Moulin (2001). First, we will specify availability of goods with respect to both agents and objects in order to accommodate complex constraints such as affirmative action. Second, we need to keep track of multiple constraints for each pair of agent-good pair (i, a) during the algorithm, since there are potentially multiple constraints that would make the consumption of the good a by the agent i no longer feasible.

Formally, the generalized probabilistic serial mechanism is defined through the following **symmetric simultaneous eating algorithm**, or the eating algorithm for short.

Mechanism 1: Generalized Probabilistic Serial Mechanism. For any $(i, a) \in S \subseteq N \times O$, let

$$\chi(i, a, S) = \begin{cases} 1 & \text{if } (i, a) \in S \text{ and } a \succeq_i b \text{ for any } b \text{ with } (i, b) \in S, \\ 0 & \text{otherwise,} \end{cases}$$

be the indicator function that a is the most preferred object for i among objects b such that (i, b) is listed in S.

Given a preference profile $\succ$, the eating algorithm is defined by the following sequence of steps. Let $S^0 = N \times O, t^0 = 0$, and $P^0_{ia} = 0$ for every $i \in N$ and $a \in O$. Given

$S^0, t^0, [P^0_{ia}]_{i\in N, a\in O}, \dots, S^{v-1}, t^{v-1}, [P^{v-1}_{ia}]_{i\in N, a\in O}$, for any $(i,a)\in S^{v-1}$ define

$$t^v(i,a) = \min_{S\in\mathcal{H}_O:(i,a)\in S} \sup \left\{ t\in[0,1] \,|\, \sum_{(j,b)\in S} [P^{v-1}_{jb} + \chi(j,b,S^{v-1})(t-t^{v-1})] < \overline{q}_S \right\}, \tag{4.1}$$

$$t^v = \min_{(i,a)\in S^{v-1}} t^v(i,a), \tag{4.2}$$

$$S^v = S^{v-1} \setminus \{(i,a)\in S^{v-1} | t^v(i,a) = t^v\}, \tag{4.3}$$

$$P^v_{ia} = P^{v-1}_{ia} + \chi(i,a,S^{v-1})(t^v - t^{v-1}). \tag{4.4}$$

Since $N\times O$ is a finite set, there exists $\bar{v}$ such that $t^{\bar{v}} = 1$. We define $PS(\succ) := P^{\bar{v}}$ to be the generalized probabilistic serial random assignment for the preference profile $\succ$.

The random assignment resulting from the generalized probabilistic serial mechanism is implementable.

Proposition 1. *For any preference profile $\succ$, the generalized probabilistic serial assignment $PS(\succ)$ can be implemented.*

Proof. Follows immediately from Theorem 1, because the constraint structure $\mathcal{H}_N \cup \mathcal{H}_O$ forms a bihierarchy and $PS(\succ)$ satisfies all inequality constraints associated with $\mathcal{H}_N\cup\mathcal{H}_O$ by construction. □

In this sense, the mechanism is well-defined as a random assignment mechanism in the current setting.

4.1. **Properties of The Generalized Probabilistic Serial Mechanism.** Bogomolnaia and Moulin (2001) show that the probabilistic serial mechanism results in an ordinally efficient random assignment in their simplified setting $\mathcal{E}^{BvN}$. Their proof can be adapted to our setting using Proposition 5, although the proof is somewhat more involved because of the constraints that are not present in their setting.

Proposition 2. *For any preference profile $\succ$, the generalized probabilistic serial random assignment $PS(\succ)$ is ordinally efficient at $\succ$.*

Bogomolnaia and Moulin (2001) also show that the probabilistic serial mechanism is fair in a specific sense in their simple setting. Formally, a random assignment P is said to be **envy-free at** $\succ$ if P_i weakly first-order stochastically dominates P_j with respect to $\succ_i$ for every $j\in N$. It turns out that the generalized generalized probabilistic serial random assignment may not be envy-free in our environment. To see this point, consider a random assignment problem in which $N=\{1,2,3\}$, $O=\{a,\emptyset\}$, $\mathcal{H}_O = \{\{1,2\}\times\{a\}, N\times$

$\{a\}\}, \overline{q}_{\{1,2\}\times\{a\}} = 1, \overline{q}_{N\times\{a\}} = 2$, and $a \succ_i \emptyset$ for every $i \in N$. In this problem it is easy to see that

$$PS(\succ) = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \\ 1 & 0 \end{pmatrix}.$$

The generalized probabilistic serial random assignment $PS(\succ)$ is not envy-free since $PS_3(\succ)$ is not weakly stochastically dominated by $PS_1(\succ)$ with respect to $\succ_1$ (indeed, $PS_3(\succ)$ stochastically dominates $PS_1(\succ)$ in this example). However, existence of envy may not immediately imply that the allocation is unfair. To see this point, note that it is infeasible to assign a to agent 1 with higher probability simply by moving probability share of a from agent 3 to agent 1, because there is a constraint on $\{1, 2\} \times \{a\}$. In that sense the envy is based on a desire of agent 1 that cannot be feasibly accommodated. Motivated by this observation, we introduce the following concept. Random assignment $P \in \mathcal{P}_{\mathcal{E}}$ is **feasible envy-free at** $\succ$ if there is no i and j such that $P_j \neq P_i$, P_j is not first-order stochastically dominated by P_i at $\succ_i$ and an assignment Q defined by

$$Q_{ka} = \begin{cases} P_{ja} & \text{if} \quad k = i, \\ 0 & \text{if} \quad k = j, a \neq \emptyset \\ 1 & \text{if} \quad k = j, a = \emptyset \\ P_{ka} & \text{otherwise}, \end{cases} \tag{4.5}$$

is in $\mathcal{P}_{\mathcal{E}}$.[13] In the above example, $PS(\succ)$ is feasible envy-free. This property turns out to hold generally, as shown below.

Proposition 3. *For any preference profile $\succ$, the generalized probabilistic serial random assignment $PS(\succ)$ is feasible envy-free at $\succ$.*

Ordinal efficiency and feasible envy-freeness are not satisfied by random priority. Indeed, random priority violates both properties even in the simplest setting $\mathcal{E}^{BvN}$ (Bogomolnaia and Moulin, 2001).

Unfortunately the mechanism is not strategy-proof, that is, there are situations in which an agent is made better off misstating her preferences.[14] However, Bogomolnaia

[13] Alternatively, one could define P to allow for feasible envy if exchanging P_i and P_j between i and j is feasible and P_j is not weakly stochastically dominated by P_i at i. This alternative definition is weaker than our current definition. Thus, by Proposition 3, the generalized probabilistic serial mechanism is feasible envy-free with this alternative definition.

[14] By contrast, the random priority mechanism is strategy-proof.

and Moulin (2001) show that the probabilistic serial mechanism is **weakly strategy-proof**, that is, an agent cannot misstate his preferences and obtain a random assignment that stochastically dominates the one obtained under truth-telling. Formally, we claim that the generalized probabilistic serial mechanism is weakly strategy-proof, that is, there exist no $\succ$, $i \in N$ and $\succ'_i$ such that $PS_i(\succ'_i, \succ_{-i})$ stochastically dominates $PS_i(\succ)$ at $\succ$ in our more general environment.[15]

Proposition 4. *The generalized probabilistic serial mechanism is weakly strategy-proof.*

Proof. The proof is an adaptation of Proposition 1 of Bogomolnaia and Moulin (2001) and we omit the proof. □

One limitation of our generalization is that the algorithm is defined only for cases with *maximum* quotas: The minimum quota for each group must be zero. In the context of school choice, this precludes the administrator from requiring that at least a certain number of students from a group attend a particular school. Despite this limitation, administrative goals can often be sufficiently represented using maximum quotas alone. For instance, if there are two groups of students, "rich" and "poor", a requirement that at least a certain number of poor students attend some highly desirable school might be adequately replaced by a maximum quota on the number of rich students who attend.

5. A Generalization of Hylland and Zeckhauser's Pseudo-market Mechanism for Assignment with Multi-Unit Demand

In a seminal paper, Hylland and Zeckhauser (1979) propose an ex-ante efficient mechanism for the problem of assigning n objects amongst n agents with single-unit demand. It is based on the old idea of Competitive Equilibrium from Equal Incomes. Agents report their von Neumann-Morgenstern preferences over individual objects. Each agent is allocated an equal budget of an artificial currency. The mechanism then computes a competitive equilibrium, where the goods being priced and allocated are *probability shares* of objects. Each agent is allocated the vector of probability shares that maximizes her expected utility subject to her budget constraint at the competitive equilibrium prices. This random assignment is efficient by the first welfare theorem, and it is envy free because agents have equal incomes. It can be implemented by appeal to the Birkhoff-von Neumann theorem.

[15]Kojima and Manea (2008) show that truthtelling becomes a dominant strategy for a sufficiently large market under the probabilistic serial mechanism in a simpler environment than the current one. Showing such a claim in a more general environment is beyond the scope of this paper, but we conjecture that the argument readily extends.

We propose a generalization of HZ's mechanism for application to assignment problems with multi-unit demand, such as assigning course schedules to students.[16] Multi-unit assignment is well known to be a challenging market design problem: axiomatic analysis of the problem is mostly negative,[17] and the mechanisms used in practice are known to suffer from inefficiency and fairness problems.[18] Our mechanism is ex-ante efficient and interim envy free like HZ's, and it allows for agents to express realistic preferences with realistic constraints. As a caveat, we note that we are not able to allow for fully general vNM preferences over bundles. If we allowed for fully general preferences, then competitive equilibrium prices might not exist, and even when they do exist we might not be able to guarantee that the resulting random assignment is implementable. Some examples that illustrate the flexibility and limitations of our preference language are discussed below.[19]

In our generalization, agents report their von Neumann-Morgenstern preferences over bundles using Milgrom's (2009) class of *integer assignment messages*. Specifically, each agent i submits a finite collection of K_i "bids", with agent i's kth bid consisting of

- A vector of valuations, one for each object: $v_{ik} = (v_{ik1}, \ldots, v_{ik|O|})$
- A set $\mathcal{H}_{ik}$ of constraint sets, which form a hierarchy. We require that each $\mathcal{H}_{ik}$ includes the "row" constraint $\{(i,k)\} \times O$, where $\{(i,k)\}$ indexes the row associated with i's kth bid.
- Finite integral floor and ceiling bounds for each constraint set, satisfying $\underline{q}_{S_i} \leq 0 \leq \bar{q}_{S_i}$ for each $S_i \in \mathcal{H}_{ik}$.

The agent also submits a set $\mathcal{H}_{i0}$ of multi-bid constraints, again satisfying $\underline{q}_{S_i} \leq 0 \leq \bar{q}_{S_i}$ for each $S_i \in \mathcal{H}_{i0}$. We require that the set $\mathcal{H}_i = \cup_{k=0}^{K_i} \mathcal{H}_{ik}$ forms a hierarchy. Note that the set $\mathcal{H}_N = \cup_{i=1}^{|N|} \mathcal{H}_i$ itself forms a hierarchy.

---

[16]Similar problems include the assignment of tasks within an organization, the division of heirlooms and estates among heirs, and the allocation of access to jointly-owned scientific resources.

[17]Papai (2001) shows that sequential dictatorships are the only deterministic mechanisms that are nonbossy, strategy-proof, and Pareto optimal; dictatorships are unattractive for many applications because they are highly unfair *ex post*. Ehlers and Klaus (2003), Hatfield (2008), and Kojima (2008) provide similarly pessimistic results.

[18]See Sönmez and Ünver (2008) and Budish and Cantillon (2009).

[19]Budish (2009) proposes a mechanism that adapts CEEI to multi-unit assignment in a different way. Whereas here agents consume their most preferred affordable random assignment of objects, in Budish's mechanism agents consume their most preferred affordable sure bundle of objects. Randomness enters through the budgets, which are approximately equal rather than exactly equal. Budish's mechanism accommodates arbitrary ordinal preferences over schedules, but is only approximately ex-post efficient whereas our mechanism is exactly ex-ante efficient. Additional discussion on the tradeoffs between these two approaches can be found in Section 8.2 of Budish (2009).

Each agent's collection of bids defines her utility function. Specifically, i's utility from consumption bundle $x_i = (x_{i1}, \ldots, x_{i|O|})$ is the solution to the following linear program

$$u_i(x_i) = \max \sum_{k=1}^{K_i} \sum_{a \in O} v_{ika} x_{ika} \text{ subject to}$$

$$\begin{aligned} \sum_{k=1}^{K_i} \sum_{a \in O} x_{ika} &= x_i \text{ (adding up constraint)} \\ \underline{q}_{S_i} \leq \sum_{\{((i,k),a)\} \in S_i} x_{ika} &\leq \bar{q}_{S_i} \text{ for all } S_i \in \mathcal{H}_i \text{ (agent constraints)} \\ x_{ika} &\geq 0 \text{ for all } k, a \end{aligned}$$

If there is no way for the agent to consume x_i in a manner that is consistent with her constraints $\mathcal{H}_i$ then her utility from x_i is $-\infty$. For technical convenience we require that each agent has a unique bliss point.

In our generalization the object hierarchy $\mathcal{H}_O$ is simple, consisting just of standard column constraints. Write $q_a \in \mathbb{Z}_+$ for the capacity of $a \in O$.

Mechanism 2: The Generalized Hylland-Zeckhauser Mechanism.

(1) Each agent i reports an integer assignment message as described above.
(2) Each agent i is given an equal budget of an artificial currency, say $b^* > 0$
(3) The mechanism computes a vector of item prices, $\mathbf{p}^* = (p_1^*, \ldots, p_{|O|}^*)$, which clear the market as described below
(4) Each agent i is allocated a bundle x_i^* which maximizes her utility subject to her preference and budget constraints. Specifically, x_i^* solves the following linear program

$$x_i^* \in \arg\max \sum_{k=1}^{K_i} \sum_{a \in O} v_{ika} x_{ika} \text{ subject to}$$

$$\begin{aligned} \sum_{k=1}^{K_i} \sum_{a \in O} p_a^* x_{ika} &\leq b^* \text{ (budget constraint)} \\ \underline{q}_{S_i} \leq \sum_{\{((i,k),a)\} \in S_i} x_{ika} &\leq \bar{q}_{S_i} \text{ for all } S_i \in \mathcal{H}_i \text{(agent constraints)} \end{aligned}$$

(5) The probability-shares market clears in the sense of

$$\begin{aligned}\sum_{i=1}^{|N|}\sum_{k=1}^{K_i} x_{ika} &\leq q_a \text{ for all } a \in O \text{ (object constraints)}\\ &< q_a \text{ only if } p_a^* = 0 \text{ (complementary slackness)}\end{aligned}$$

(6) The random assignment $\mathbf{x}^* = (x_1^*, \ldots, x_{|N|}^*)$ is implemented.

To show that the mechanism is well defined we need the following two Lemmas:

Proposition 5. *There exist competitive equilibrium prices* $\mathbf{p}^*$ *which clear the market in the sense of Steps 3-5.*

Proof. See the Appendix. □

Standard competitive equilibrium existence results cannot be applied here due to the failure of local non satiation (e.g., each student requires at most one seat in any course and at most a certain number of courses overall).[20] Two features of our environment that we exploit in our existence proof are (i) demand does not explode at price zero, specifically because of satiation; and (ii) the price space can be bounded above, because we work with an economy in which agents are endowed directly with finite budgets, rather than an exchange economy with probability-share endowments. It is also important for our existence result that $\underline{q}_S \leq 0 \leq \bar{q}_S$ for each constraint set S. This assumption, together with the linearity of preferences, helps ensure that each agent's demand correspondence is convex and upper-hemicontinuous in price. Hylland and Zeckhauser (1979) assume strictly positive floor constraints – in HZ, each agent requires *exactly* one object – and for this reason their method of proof is more involved and does not readily generalize to our environment.

Proposition 6. *The random assignment* $\mathbf{x}^*$ *produced in Step 6 of the Mechanism can be implemented.*

Proof. Follows immediately from Theorem 1, because $\mathcal{H}_N \cup \mathcal{H}_O$ forms a bihierarchy. □

5.1. **Properties of the Generalized HZ Mechanism.** The original HZ mechanism is attractive for single-unit assignment because it is ex-ante efficient and interim envy free. We show that these properties carry over to this more general environment.

[20]Hylland and Zeckhauser (1979) discuss the failure of standard existence proof techniques in their footnote 14.

Proposition 7. *The random assignment* $\mathbf{x}^*$ *is ex-ante Pareto efficient, and every realization of the lottery is ex-post efficient.*

Proof. Suppose there exists a random assignment $\tilde{\mathbf{x}}$ that Pareto improves upon $\mathbf{x}^*$. If $u_i(\tilde{x}_i) > u_i(x_i^*)$ then revealed preference implies that $\mathbf{p}^* \cdot \tilde{x}_i > \mathbf{p}^* \cdot x_i^*$. Suppose $u_i(\tilde{x}_i) = u_i(x_i^*)$. We claim that $\mathbf{p}^* \cdot \tilde{x}_i \geq \mathbf{p}^* \cdot x_i^*$. Towards a contradiction, suppose that $\mathbf{p}^* \cdot \tilde{x}_i < \mathbf{p}^* \cdot x_i^*$. Let $\bar{x}_i$ denote i's bliss point. From our assumption that bliss points are unique it follows that $u_i(\bar{x}_i) > u_i(\tilde{x}_i) = u_i(x_i^*)$. By revealed preference, $\mathbf{p}^* \cdot \bar{x}_i > b^*$. However, since $\mathbf{p}^* \cdot \tilde{x}_i < \mathbf{p}^* \cdot x_i^* \leq b^*$ there exists $\lambda \in (0,1)$ such that $\mathbf{p}^* \cdot (\lambda \bar{x}_i + (1-\lambda)\tilde{x}_i) \leq b^*$. By strict convexity of preferences $u_i(\lambda \bar{x}_i + (1-\lambda)\tilde{x}_i) > u_i(\tilde{x}_i) = u_i(x_i^*)$, which contradicts x_i^* being a solution to (4) for i. Hence $\mathbf{p}^* \cdot \tilde{x}_i \geq \mathbf{p}^* \cdot x_i^*$.

From the above and from the assumption that $\tilde{\mathbf{x}}$ is a Pareto improvement on $\mathbf{x}^*$, we have established that $\mathbf{p}^* \cdot \tilde{x}_i \geq \mathbf{p}^* \cdot x_i^*$ for all i with at least one strict. Thus $\sum_i \mathbf{p}^* \cdot \tilde{x}_i > \sum_i \mathbf{p}^* \cdot x_i^*$. But this contradicts $\mathbf{x}^*$ being a competitive equilibrium, since in a competitive equilibrium (see (5.) above) any good that has a strictly positive price is at full capacity, hence no feasible allocation can cost strictly more at $\mathbf{p}^*$ than does $\mathbf{x}^*$. Hence $\mathbf{x}^*$ is (ex-ante) Pareto efficient.

Ex-ante efficiency immediately implies ex-post efficiency; if some realization of a lottery were ex-post inefficient, then by executing Pareto improvements for that realization we could generate an ex-ante Pareto improvement. □

Proposition 8. *The random assignment* $\mathbf{x}^*$ *is interim envy free. That is, for any agents* $i \neq j$, $u_i(x_i^*) \geq u_i(x_j^*)$.

Proof. Follows immediately from the definition of the mechanism given that all agents have the same budget. □

Propostion 8 concerns interim fairness. The main result of Section 6 can be used to enhance *ex-post* fairness in cases where agents' bids take a simple additive-separable form without additional constraints.

5.2. **Flexibility and Limitations of the Proposed Preference Language.** Our preference reporting language allows agents to express several kinds of constraints that may be useful for practice. To fix ideas we focus on constraints specific to the problem of course allocation.

5.2.1. *Scheduling Constraints.* Scheduling constraints can often be expressed by means of a hierarchy. One example is students at Harvard Business School, who require 10 courses per school year, of which 5 should be in each of the two semesters, and of which no more than one should meet at any given time.

More generally, if there is a set of time slots, constraints of the form "no more than one course at any time slot" form a simple kind of hierarchy.

5.2.2. *Curricular Constraints.* Students often seek variety in their schedules due to diminishing returns. Our language can accommodate constraints of the form "at most 2 courses in Finance", and it can also accommodate more elaborate constraints like "at most 2 courses in Finance, at most 2 courses in Marketing, and at most 3 courses in Finance or Marketing."

Our language is not able to accommodate most kinds of complementarity that arise from curricular considerations. For instance, a student cannot express that they wish to take Macroeconomics if and only if they can also take Microeconomics.

5.2.3. *Diminishing Marginal Returns.* An important feature of our language is that it allows agents to express certain kinds of diminishing returns. This can be achieved by submitting multiple bids, with constraints defined on each.

For instance, suppose that there are two "star professors", a and b, and that agent i has a high value for the first of these courses she gets to take, and a lower value for the second. Then she can submit two bids: in the first bid, she expresses high values for the first unit of a star-professor course, e.g., $v_{i1} = (10000, 5000)$ with a ceiling constraint of 1. In the second bid, she expresses a lower value for the second unit of a star-professor course, e.g.,$v_{i2} = (1000, 500)$ with a ceiling constraint of 1. If needed, she could also have a multi-row ceiling constraint defined on the sum of the amounts awarded across the two bids.

Notice that the linear program in Mechanism 5 Step 4 will fully fill the high-valued bid before filling any of the lower-valued bid.

5.2.4. *Limitations.* The primary limitation of our language is that these multiple uses cannot be readily combined. For instance, if there is a Finance course and a Marketing course that meets at time slot 1 $(F1, M1)$, and another Finance course and another Marketing course that meets at time slot 2 $(F2, M2)$, then the scheduling constraints on $\{F1, M1\}$ and $\{F2, M2\}$ and the curricular constraints on $\{F1, F2\}$ and $\{M1, M2\}$ cannot coexist in the same hierarchy.

6. The Utility Guarantee for Multi-Unit Assignment and Matching

We call our third application the "utility guarantee". In general, there can be many ways to implement a given random assignment, and the choice among them may be important. To fix ideas, suppose that two agents are to divide $2n$ objects, that the agents' preferences are additively separable, and that agents' ordinal rankings of the items are the same. Suppose the "fair" random assignment specifies that each agent receive half of each object. One way to implement this is to randomly choose n objects to assign to the first agent, and then give the remaining n objects to the other agent. This method, however, could entail a highly "unfair" outcome *ex post*, in which one agent gets the n best objects and the other gets the n worst ones.

Based on the Rounding Theorem, we provide a method to reduce the variation in utility outcomes resulting from randomization. Formally, consider an input $\langle N, O, P, \mathcal{H}\rangle$ where $\mathcal{H}$ is a bihierarchy partitioned into hierarchies $\mathcal{H}_O$ and $\mathcal{H}_N = \mathcal{H}_N^{BvN}$. Assume that P satisfies $\sum_a P_{ia} \in \mathbb{Z}$ for each $i \in N$. This assumption is without loss of generality because any random assignment with non-integral row sums is equivalent to a random assignment with an additional column representing a null object, the sole purpose of which is to ensure that rows sum to integer amounts.

Theorem 3. (Utility Guarantee) *Suppose that there is a set of values $(v_{ia})_{(i,a)\in N\times O}$ such that, for each i, agent i's expected utility from a random assignment P is $\sum\limits_{a\in O} P_{ia}v_{ia}$. Then, for any P, there exists a decomposition of P that satisfies all of the conditions of the Rounding Theorem, and also:*

$$\sum_a P'_{ia}v_{ia} - \sum_a P''_{ia}v_{ia} \in [-\Delta_i, \Delta_i], \tag{6.1}$$

$$\sum_a P'_{ia}v_{ia} \in \left[\sum_a P_{ia}v_{ia} - \Delta_i, \sum_a P_{ia}v_{ia} + \Delta_i\right], \tag{6.2}$$

for each i and each P' and P'' in the convex combination, where $\Delta_i := \max\{v_{ia} - v_{ib} | a, b \in O, P_{ia}, P_{ib} \notin \mathbb{Z}\}$.

A proof sketch can be given based on the Rounding Theorem. The idea is to supplement the actual constraints of the problem with a set of "utility proximity" constraints, as follows. If there are n objects, for each agent we create n additional constraints. The j^{th} constraint set of agent i, S_{ij}, consists of his $1^{\text{st}}, 2^{\text{nd}}, \ldots, j^{\text{th}}$ most preferred objects; its floor and ceiling constraints are $\left\lfloor \sum_{a\in S_{ij}} P_{ia}\right\rfloor$ and $\left\lceil \sum_{a\in S_{ij}} P_{ia}\right\rceil$, respectively. The resulting constraint structure is still a bihierarchy after this addition, so the Rounding

Theorem guarantees that the random assignment can be implemented with all of the constraints satisfied. Satisfying the constraints on all of the artificial "upper contour" sets means that in each realized assignment, each agent receives her j most preferred objects, for each j, with approximately the same probability as in the original random assignment. Our method thus ensures that each realized assignment inherits roughly the same fairness properties as the original random assignment; more precisely, the approximation error is at most the utility difference between the most valuable and the least valuable objects.

In principle Theorem 3 can be used to augment ex-post fairness in conjunction with any multi-unit random assignment algorithm, as long as agents' preferences are additive separable. One specific pairing that may be useful is with our multi-unit generalization of HZ developed in Section 5. Multi-unit HZ is *interim* envy free because all agents have the same budget and face the same prices; see Proposition 8. By utilizing the Utility Guarantee as well, we can also ensure that *ex-post* envy is bounded.

6.1. **Application: Two-Sided Matching.** With slight modification, the Utility Guarantee can also be applied to two-sided matching environments. Let N and O both be sets of agents, and consider many-to-many matching in which each agent in N can be matched with multiple agents in O, and vice versa.

Theorem 4. *Consider a problem where the constraint structure is $\mathcal{H}^{BvN}$. Suppose that there are sets of values $(v_{ia})_{(i,a)\in N\times O}$ and $(w_{ia})_{(i,a)\in N\times O}$ such that, for each agent $i \in N$ (respectively agent $a \in O$), her expected utility from any random assignment P is $\sum_{a\in O} P_{ia}v_{ia}$ (respectively $\sum_{i\in N} P_{ia}w_{ia}$). Then, for any P, there exists a decomposition of P that satisfies the conditions of the Rounding Theorem, and also:*

$$\sum_a P'_{ia}v_{ia} - \sum_a P''_{ia}v_{ia} \in [-\Delta_i, \Delta_i]$$

$$\sum_a P'_{ia}v_{ia} \in \left[\sum_a P_{ia}v_{ia} - \Delta_i, \sum_a P_{ia}v_{ia} + \Delta_i\right],$$

$$\sum_i P'_{ia}w_{ia} - \sum_i P''_{ia}w_{ia} \in [-\Delta_a, \Delta_a]$$

$$\sum_i P'_{ia}w_{ia} \in \left[\sum_i P_{ia}w_{ia} - \Delta_a, \sum_i P_{ia}w_{ia} + \Delta_a\right],$$

for each i, a and each P' and P'' being part of the convex decomposition, where $\Delta_i = \max\{v_{ia} - v_{ib} | a, b \in O, P_{ia}, P_{ib} \notin \mathbb{N}\}$ and $\Delta_a = \max\{w_{ia} - w_{ja} | i, j \in N, P_{ia}, P_{ja} \notin \mathbb{N}\}$.

Proof. The proof is a straightforward adaptation of the proof of Theorem 3 and hence is omitted. □

Let us suggest one possible application. There are two leagues of sports teams N and O, say the American League and National League in professional baseball, and the planner wants to schedule interleague play. The planner wants to ensure that the strength of opponents that teams in a league play against is as equalized as possible among teams in the same league. For that goal, the planner could first give a uniform probability for each match: That will give one specific random assignment in which any pair of teams in the same league is treated equally. Then, using Theorem 4, the planner finds a deterministic assignment, in which differences in schedule strength are bounded by the difference between one game with the strongest opponent and one with the weakest opponent in the other league.

We note that transforming this feasible match into a specific schedule — i.e., not only how often does Team A play Team B, but *when* — is considerably more complicated. For example, the problem involves scheduling both intraleague and interleague matches simultaneously, dealing with geographical constraints and so forth. See Nemhauser and Trick (1998) for further discussion of sport scheduling.

7. Beyond Bilateral Assignment

Throughout the paper we have focused on a random assignment of objects (or agents) to agents. However, some of our results can be extended beyond pairwise assignment, as described below.

Let X be a finite set and $\mathcal{H}$ be a collection of subsets of X. We call the pair $\mathcal{X} = (X, \mathcal{H})$ a **hypergraph**. A (generalized) random assignment is a vector $P = [P_x]$ where $P_x \in (-\infty, \infty)$ for all $x \in X$. For each $S \in \mathcal{H}$, $P_S = \sum_{x \in S} P_x$. A deterministic assignment is a random assignment each of whose entries is an integer. As before, the constraint structure $\mathcal{H}$ is decomposable if, for each $(\underline{q}_S, \overline{q}_S)_{S \in \mathcal{H}}$ and P with $\underline{q}_S \leq P_S \leq \overline{q}_S$ for all $S \in \mathcal{H}$, there exist $\lambda^1, \dots, \lambda^K$ and $P^1, \dots, P^K$ such that

(1) $P = \sum_{k=1}^{K} \lambda^k P^k$,
(2) $\lambda^k > 0, k = 1, \dots, K$, and $\sum_{k=1}^{K} \lambda^k = 1$,
(3) P_x^k is an integer for each x,
(4) $\underline{q}_S \leq P_S^k \leq \overline{q}_S$ for each $k = 1, \dots, K$ and $S \in \mathcal{H}$.

Equivalently, $\mathcal{H}$ is decomposable if and only if for every random assignment P there exist $\lambda^1, \dots, \lambda^K$ and $P^1, \dots, P^K$ such that

(1) $P = \sum_{k=1}^{K} \lambda^k P^k$,

(2) $\lambda^k > 0, k = 1, \ldots, K$, and $\sum_{k=1}^{K} \lambda^k = 1$,

(3) $P_S^k \in \{\lfloor P_S \rfloor, \lceil P_S \rceil\}$ for all $k \in \{1, \ldots, K\}$ and $S \in \mathcal{H}$.

We say that $\mathcal{X}$ forms a bihierarchy if there exist $\mathcal{H}_1$ and $\mathcal{H}_2$ such that $\mathcal{H}_1 \cup \mathcal{H}_2 = \mathcal{H}$, $\mathcal{H}_1 \cap \mathcal{H}_2 = \emptyset$, and $\mathcal{H}_i$ is a hierarchy for each $i = \{1, 2\}$: if $S, S' \in \mathcal{H}_i$, then $S \cap S' = \emptyset$ or $S \subset S'$ or $S' \subset S$.

It is useful to define the dual of a hypergraph. Given a hypergraph $\mathcal{X} = (X, \mathcal{H})$, its dual is $\mathcal{X}^T = (\mathcal{H}, X)$. A bihierarchy can be defined for its dual. To this end, for each $x \in X$, let $\mathcal{S}(x) := \{S \in \mathcal{H} | x \in S\}$ be the collection of sets in $\mathcal{H}$ each containing x. We say **the dual of $\mathcal{X}$ forms a bihierarchy** if there are X_1 and X_2 such that $X_1 \cup X_2 = X$, $X_1 \cap X_2 = \emptyset$ and X_i, $i = 1, 2$, is a **dual hierarchy**: if $x, x' \in X_i$, then $\mathcal{S}(x) \cap \mathcal{S}(x') = \emptyset$ or $\mathcal{S}(x) \subset \mathcal{S}(x')$ or $\mathcal{S}(x') \subset \mathcal{S}(x)$.

A hypergraph $\mathcal{X} = (X, \mathcal{H})$ can be represented by an incidence matrix $A = [a_{xS}]$ such that $a_{xS} = 1_{\{x \in S\}}$. The incidence matrix of the dual $\mathcal{X}^T$ is A^T, the transpose of A.

Theorem 5. *A hypergraph is decomposable if either it forms a bihiearchy or its dual forms a bihierarchy.*

The proof of Theorem 5 is in Appendix A. The key step in the proof, due to Edmonds (1970), is to show that the incidence matrix A of a bihierarchical constraint structure satisfies a condition called total unimodularity. By Hoffman and Kruskal (1956), this implies that the set of random assignments satisfying a bihierarchical constraint structure has integral extreme points, which enables decomposition because the set is convex.[21]

We present two examples of bihierarchies whose dual satisfies the bihierarchy condition.

Example 4. *Consider $X = \{a, b, c, d, e, f\}$, and*

$$\mathcal{H} = \{\{a, d\}, \{a, e\}, \{a, f\}, \{b, d\}, \{b, e\}, \{b, f\}, \{c, d\}, \{c, e\}, \{c, f\}\}.$$

The hypergraph $\mathcal{X} = (X, \mathcal{H})$ is in fact a bipartite graph in this case. Even though it does not form a bihierarchy, its dual forms a bihierarchy. Its dual is an assignment between three agents and three objects, with only row and column constraints.

Example 5. *Consider $X = \{a, b, c, d, e, f, \alpha, \beta, \delta, \epsilon\}$, and*

$$\mathcal{H} = \{\{a, d, \alpha, \delta\}, \{a, e, \alpha, \epsilon\}, \{a, f\}, \{b, d, \beta, \delta\}, \{b, e, \beta, \epsilon\}, \{b, f\}, \{c, d\}, \{c, e\}, \{c, f\}\}.$$

[21] We thank Tomomi Matsui and Akihisa Tamura for informing us of the connection with the mathematics literature on matroids and Edmonds (1970).

The hypergraph $\mathcal{X} = (X, \mathcal{H})$ again does not form a bihierarchy, but its dual forms a bihierarchy. Its dual is the 3 by 3 matching, with row and column constraints, and two subrow and two subcolumn constraints.

We note that Lemma 1 clearly holds in this general environment (with an identical proof), providing a necessary condition for decomposability. We can apply this lemma to show the difficulty one faces in implementing random assignments in multilateral matching and roommate matching.

7.1. **Multilateral Matching.** Thus far, we have focused on bilateral matching in which agents on one side are assigned to objects (or agents) on the other side. As noted, many important market design problems fall into the bilateral matching environment. Sometimes, however, matching involves more than two sides. For instance, students may be assigned to different schools and after-school programs, in which case the matching must be trilateral, consisting of student/school/after-school triples. Or, manufacturers may need to match with multiple suppliers, ensuring mutual compatibility of products or the right combination of capabilities.

Our main point is most easily made by starting with a trilateral matching problem in which we introduce another finite set M of say agents, in addition to N and O. A matching then consists of a triple $(i, a, m) \in N \times O \times M$, and a random assignment is defined by a profile $[P_{(i,a,m)}]_{(i,a,m)\in N\times O\times M}$ that assigns a real number to each triple (i, a, m). Constraints on the random assignment can be described as before via the constraint structure, i.e., the sets of (i, a, m)'s whose entries are subject to a ceiling or a floor. That is, the constraint structure $\mathcal{H} \subset 2^{N\times O\times M}$ is a collection of subsets of $N \times O \times M$. As in the classical setup, the basic constraints arise from the fact that each agent in N, each object in O and each agent in M must be assigned to some pair in the other two sides (which may include a null object or null agent, by including such entities to the sets). Hence, it is natural to assume that $\mathcal{H}$ contains the sets $\overline{\mathcal{H}}^{BvN} := \{\{i\} \times O \times M | i \in N\} \cup \{N \times \{a\} \times M | a \in O\} \cup \{N \times O \times \{m\} | m \in M\}$.

Notice that the problem reduces to that of bilateral matching if the cardinality of N or O or M is one. It turns out that, except for such cases, no analogue of the Birkhoff-von Neumann theorem holds with a trilateral matching.

Theorem 6. (Impossibility with Trilateral Matching) *In trilateral matching with $N \times O \times M$ where $|N|, |O|, |M| \geq 2$, any $\mathcal{H} \supset \overline{\mathcal{H}}^{BvN}$ is not decomposable.*

Proof. We prove the result by showing that any $\mathcal{H} \supset \overline{\mathcal{H}}^{BvN}$ contains an odd cycle. By Lemma 1, this is sufficient for failure of decomposability. (Even though the proof of

Lemma 1 formally deals with the bilateral matching setup, its proof does not depend on it.)

Fix $i \in N, a \in O, m \in M$ and consider three sets $S_i := \{i\} \times O \times M, S_a := N \times \{a\} \times M$, and $S_m := N \times O \times \{m\}$. Fix $i' \in N, a' \in O, m' \in M$ such that $i' \neq i$, $a' \neq a$, and $m' \neq m$ (such i', a', and m' exist since $|N|, |O|, |M| \geq 2$). Then $(i, a, m') \in S_i \cap S_a \setminus S_m$, $(i, a', m) \in S_i \cap S_m \setminus S_a$, and $(i', a, m) \in S_a \cap S_m \setminus S_i$. We thus conclude that S_i, S_a, and S_m form an odd cycle. □

It is clear from the proof that the same impossibility result holds for any multilateral matching of more than two kinds of agents.

Remark 2. (Matching with Contracts) *Firms sometimes hire workers for different positions with different terms of contract. For instance, hospitals hire medical residents for different kinds of positions (such as research and clinical positions), and different positions may entail different duties and compensations. To encompass such situations, Hatfield and Milgrom (2005) develop a model of "matching with contracts," in which a matching specifies not only which firm employs a given worker but also at what contract terms. At first glance, introducing contract terms may appear to transform the environment into a trilateral matching setting. This is in fact not the case. If we let M denote the set of possible contract terms, there is no sense in which the constraint structure contains sets of the form $N \times O \times \{m\}$. In words, there is no reason that each contract term should be chosen by some worker-firm pair. Rather, the matching with contracts can be subsumed into our bilateral matching setup by redefining the object set as $O' := O \times M$.*

7.2. **Roommate Matching.** The "roommate problem" describes another interesting matching problem, in which any agent can, in principle, be matched to any other. One example is "pairwise kidney exchange" (Roth, Sonmez, and Ünver, 2005), in which a kidney patient with a willing-but-incompatible donor is to be matched to another patient-donor pair. If two such pairs are successfully matched, then the donor in each pair donates her kidney to the patient of the other pair.

For our analysis, the important elements of a roommate matching problem include a (finite) set of agents, N, and a set $X_N := \{\{i, j\} | i, j \in N\}$ of possible (unordered) pairs of agents who can be matched as roommates. If the pair $\{i, i\}$ is formed, that means that i is unmatched. Let $\mathcal{H}_N$ be a collection of subsets of X_N. A hypergraph $\mathcal{X}_N = (X_N, \mathcal{H}_N)$ allows us to capture the basic environment arising in roommate matching, and we can describe a (generalized) random assignment as a vector $P = [P_x]_{x \in X_N}$ where $P_x \in [0, 1]$ for all $x \in X_N$. We assume that each i must be assigned to some agent (possibly himself), so $\mathcal{H}_N$ must contain set $S_i := \{\{i,j\}|j \in N\}$ for each $i \in N$. We call a hypergraph $\mathcal{X}_N$ satisfying this property a **canonical roommate matching problem with N agents**.

Notice that the problem reduces to that of bilateral matching if $|N| \leq 2$, implying that any canonical roommate matching problem with such N is decomposable. The next result shows that these are the only cases for which decomposability holds.

Theorem 7. (Impossibility with Roommate Matching) *A canonical roommate matching problem with N agents, $|N| \geq 3$, is not decomposable.*

Proof. We prove the result by showing that $\mathcal{H}_N$ in the canonical roommate matching contains an odd cycle. Consider $i, j, k \in N$, who are all distinct (such agents exist since $|N| \geq 3$). Then, $\{i,j\} \in (S_i \cap S_j) \setminus S_k$, $\{j,k\} \in (S_j \cap S_k) \setminus S_i$, and $\{i,k\} \in (S_i \cap S_k) \setminus S_j$. We thus conclude that S_i, S_j, and S_k form an odd cycle. □

8. Polynomial-Time Algorithm for Implementing Random Assignments

Theorem 5 demonstrates implementability of random assignments without revealing the method. For practical purposes, knowing simply that a random assignment is implementable is not sufficient; implementation must be computable or sufficiently fast. Here we provide an algorithm that implements random assignments in polynomial time.[22] At each step of the algorithm, which is described more fully in Appendix B, a given feasible random assignment P is decomposed into a convex combination $\gamma P' + (1-\gamma)P''$ of two feasible random assignments, each of which has at least one more integer-valued agent-object pair (or integer-valued constraint set). Then, a random number is generated and with probability γ the algorithm continues by similarly decomposing P', while with probability $1-\gamma$ the algorithm continues by decomposing P''. The algorithm stops when it reaches an integer assignment.

We explain the decomposition method in the context of the following example. Consider a hypergraph $\mathcal{X} = (X, \mathcal{H})$ where $X = \{x_1, x_2, x_3, x_4\}$, $\mathcal{H} = \{\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, S_1, S_2\}$ and $S_1 := \{x_2, x_3\}$ and $S_2 := \{x_3, x_4\}$. Observe that $\mathcal{H}$ is a bihierarchy consisting of two hierarchies, $\mathcal{H}_1 = \{\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, S_1\}$ and $\mathcal{H}_2 = \{S_2\}$, where $S_1 := \{x_2, x_3\}$ and $S_2 := \{x_3, x_4\}$. Suppose we wish to implement a random assignment P with $P_{\{x_1\}} = 0.3, P_{\{x_2\}} = 0.7, P_{\{x_3\}} = 0.3$ and $P_{\{x_4\}} = 0.7$. We represent the given random assignment P as a network flow. The particular way in which the flow network is constructed is

[22] We thank Tomomi Matsui and Akihisa Tamura for suggesting this algorithm. An earlier draft of this paper included an alternative algorithm generalizing the stepping-stones algorithm described by Hylland and Zeckhauser (1979). We have made our old algorithm available in a Web Appendix.

crucial for the algorithm, and we formally describe its construction in Appendix B. Here, we present an informal discussion based on our example. See Figure 1.

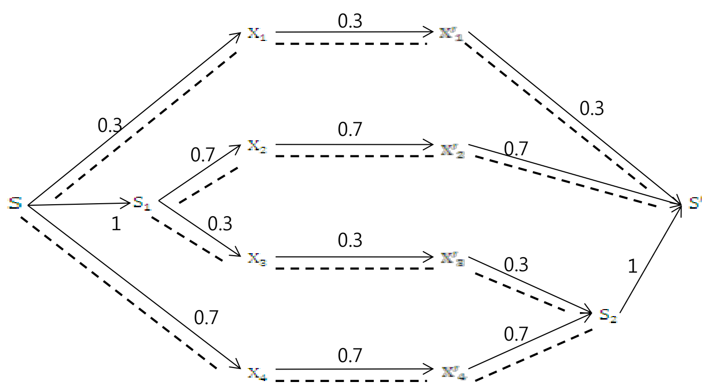

Figure 1: A network flow representation of the example P.

Intuitively, we view the total assignment as flows that travel from source s to sink s' of a network (s and s' can be interpreted as corresponding to the entire set X). First, the flows travel through the sets in one hierarchy $\mathcal{H}_1$, arranged in "descending" order of set-inclusion; the flows move from bigger to smaller sets along the directed edges representing the set-inclusion tree, reaching at last the singleton sets. This accounts for the left side of the flow network in Figure 1, where the numbers on the edges depict the flows. From then on, the flows travel through the sets in the other hierarchy $\mathcal{H}_1$ which is augmented, without loss, to include the singleton sets and the entire set X, with primes attached for notational clarity. These sets are now arranged in "ascending" order of set-inclusion; the flows travel from smaller to bigger sets along the directed edges representing the reverse set-inclusion tree, reaching at the end the total set s', or the sink.

Notice that the flow associated with each edge reflects the random assignment for the corresponding set. For instance, the flow from x_2 to x_2' is the random assignment $P_{\{x_2\}} = 0.7$ for set x_2, and likewise the flow from x_3 to x_3' is $P_{\{x_3\}} = 0.3$. The flow from s to S_1 represents the random assignment $P_{S_1} = 1$ for set S_1. Naturally, the latter flow must be the sum of the two former flows. More generally, the additive structure of the random assignment is translated into the "law of conservation": *the flow reaching each (intermediate) vertex must equal the flow leaving that vertex.*

Given the flow network, the algorithm identifies a cycle of agent-object pairs with fractional assignments. Starting with any edge with fractional flow, say (x_2, x_2'), we find another edge with a fractional flow that is adjacent to x_2'. Such an edge, (x_2', s'), exists due to the law of conservation: if all neighboring flows were integer we would have a contradiction. We keep adding new edges with fractional flows in this fashion, the ability to do so ensured by the law of conservation, until we create a cycle. In this case, the cycle of vertices is $x_2 - x_2' - s' - x_1' - x_1 - s - x_4 - x_4' - S_2 - x_3' - x_3 - S_1 - x_2$. This cycle is denoted by the dotted lines in Figure 1.

We next modify the flows of the edges in the cycle. First, we raise the flow of each forward edge and reduce the flow of each backward edge at the same rate until at least one flow reaches an integer value. In our example, the flows along all the forward edges rise from 0.7 to 1 and the flows along all the backward edges fall from 0.3 to 0. Importantly, this process preserves the law of conservation, meaning that the operation maintains the feasibility of the new random assignment. The resulting network flow then gives rise to a random assignment P' where $P'_{\{x_1\}} = 0, P'_{\{x_2\}} = 1, P'_{\{x_3\}} = 0$, and $P'_{\{x_4\}} = 1$. Next, we readjust the flows of the edges in the cycle in the reverse direction, raising those with backward edges and reducing those with forward edges in an analogous manner, which gives rises to another random assignment P'' where $P''_{\{x_1\}} = 1, P''_{\{x_2\}} = 0, P''_{\{x_3\}} = 1$, and $P''_{\{x_4\}} = 0$. We can now decompose P into these two matrices, i.e., $P = 0.7P' + 0.3P''$.

The random algorithm then selects P' with probability 0.7 and P'' with probability 0.3. Since in this particular example both P' and P'' are integer valued, there is no need to re-iterate the decomposition process. In general, each step in the algorithm reduces the number of fractional flows in the network, converting at least one to an integer. The total number of steps in the random algorithm is therefore limited to the number of fractional flows. Also, each step visits each remaining fractional flow at most once, so the total number of visits grows at most as the square of the number of fractional flows. Thus, the run time of the algorithm is polynomial in $|\mathcal{H}|$.

9. Conclusion

We generalize the Birkhoff-von Neumann theorem by applying it to general real matrices, rather than just bistochastic matrices, and allowing a much larger class of constraints, rather than just row and column constraints. We show that if the constraints specify integer floors and ceilings for sums over sets of elements in the matrix, and if the constraint sets form a bihierarchy, then the matrix is a convex combination of integer matrices each of which also satisfies the constraints. This convex combination is a "decomposition" of the expected allocation described by the originally given matrix. We also establish a converse: if the constraint sets described above include row and column constraints but do not form a bihierarchy, then there are matrices and constraint bounds such that the matrix is not a convex combination of feasible integer matrices. We show that these results are usefully applicable to a range of matching problems, including (i) single-unit assignment, (ii) multi-unit assignment, (iii) fair division, (iv) job scheduling, and (v) two-sided matching, and that the results enable useful generalizations of Bogomolnaia and Moulin's (2001) probabilistic serial mechanism and Hylland and Zeckhauser's (1979) pseudo-market mechanism. We also introduce a polynomial algorithm that implements the random allocation, selecting each pure outcome with the appropriate probability.

We investigate other kinds of constraints as well, mostly with negative results. There is no similar decomposability of expected allocations for matching with three sides or more, nor for roommate problems.

For the future, one may expect to find closer connections to similar decomposability problems arising in optimal mechanism theory. But the goal of research in market design is to facilitate applications, and we are most hopeful that the examples of implementable random assignments described here herald still further applications to come.

Appendix A. Proofs of Theorems 1 and 5

Since Theorem 1 is a special case of Theorem 5, we prove the latter.

A matrix is **totally unimodular** if the determinant of every square submatrix is 0, -1 or $+1$. We make use of the following result.

Lemma 2. *(Hoffman and Kruskal (1956)) If a matrix A is totally unimodular, then the vertices of the polyhedron defined by linear integral constraints are integer valued.*

The proof strategy for Theorem 5 proceeds in two steps. First we show that if either a hypergraph or its dual forms a bihierarchy, then the incidence matrix of the hypergraph is totally unimodular. Second we apply Lemma 2 to show that the hypergraph is decomposable.

After an earlier draft was circulated, we were informed that Edmonds (1970) has previously shown that the incidence matrix of a bihierarchical constraint structure is totally unimodular. We include our own proof for completeness below. The case in which the dual of a hypergraph forms a bihierarchy is not in Edmonds (1970).

We utilize the following result for our proof.

Lemma 3. *(Ghouila-Houri (1962)) A $\{0,1\}$ incidence matrix is totally unimodular if and only if each subcollection of its columns can be partitioned into red and blue columns such* *that for every row of that collection, the sum of entries in the red columns differs by at most one from the sum of the entries in the blue columns.*

Proof of Theorem 5. Suppose first $\mathcal{X}$ forms a bihierarchy, with $\mathcal{H}_1$ and $\mathcal{H}_2$ such that $\mathcal{H}_1 \cup \mathcal{H}_2 = \mathcal{H}$, $\mathcal{H}_1 \cap \mathcal{H}_2 = \emptyset$ and both $\mathcal{H}_1$ and $\mathcal{H}_2$ are hierarchies. Let A be the associated incidence matrix. Take any collection of columns of A, corresponding to a subcollection E of $\mathcal{H}$. We shall partition E into two sets, B and R. First, for each $i = 1, 2$, we partition $E \cap \mathcal{H}_i$ into nonempty sets $E_i^1, E_i^2, \ldots, E_i^{k_i}$ defined recursively as follows: Set $E_i^0 \equiv \emptyset$ and, for each $j = 1, \ldots,$ we let

$$E_i^j := \{S \in (E \cap \mathcal{H}_i) \setminus (\bigcup_{j'=1}^{j-1} E_i^{j'}) \,|\, \nexists S' \in (E \cap \mathcal{H}_i) \setminus (\bigcup_{j'=1}^{j-1} E_i^{j'} \cup \{S\}) \text{ such that } S' \supset S\}.$$

(The non-emptiness requirment means that once all sets in $E \cap \mathcal{H}_i$ are accounted for, the recursive definition stops, which it does at a finite $j = k_i$.) Since $\mathcal{H}_i$ is a hierarchy, any two sets in E_i^j must be disjoint, for each $j = 1, \ldots, k_i$. Hence, any element of X can belong to at most one set in each E_i^j. Observe next for $j < l$, $\bigcup_{S \in E_i^l} S \subset \bigcup_{S \in E_i^j} S$. In other words, if an element of X belongs to a set in E_i^l, it must also belong to a set in E_i^j for each $j < l$.

We now define sets B and R that partition E:

$$B := \{S \in E | S \in E_i^j, i + j \text{ is an even number }\},$$

and

$$R := \{S \in E | S \in E_i^j, i + j \text{ is an odd number }\}.$$

We call the elements of B "blue" sets, and call the elements of R "red" sets.

Fix any $x \in X$. If x belongs to any set in $E \cap \mathcal{H}_1$, then it must belong to exactly one set $S_1^j \in E_1^j$, for each $j = 1, \ldots, l$ for some $l \leq k_1$. These sets alternate in colors in $j = 1, 2, \ldots,$ starting with blue: S_1^1 is blue, S_1^2 is red, S_1^3 is blue, and so forth. Hence, the number of blue sets in $E \cap \mathcal{H}_1$ containing x either equals or exceeds by one the number of red sets in $E \cap \mathcal{H}_1$ containing x. By the same reasoning, if x belongs to any set in $E \cap \mathcal{H}_2$, then it must belong to one set $S_2^j \in E_2^j$, for each $j = 1, \ldots, m$ for some $m \leq k_2$. These sets alternate in colors in $j = 1, 2, \ldots,$ starting with red: S_2^1 is red, S_2^2 is blue, S_2^3 is red, and so forth. Hence, the number of blue sets in $E \cap \mathcal{H}_2$ containing x is less by one than or equal to the number of red sets in $E \cap \mathcal{H}_2$ containing x. In sum, the number of blue sets in E containing x differs at most by one from the number of red sets in E containing x. Thus A is totally unimodular by Lemma 3.

Choose an arbitrary random assignment P and consider the set

$$\{P' | \lfloor P_S \rfloor \leq P'_S \leq \lceil P_S \rceil, \forall S \in \mathcal{H}\}. \tag{A.1}$$

By Lemma 2, every vertex of the set (A.1) is integer valued. Since (A.1) is a convex polyhedron, any point of it (including P) can be written as a convex combination of its vertices. Since we chose P arbitrarily, the hypergraph $\mathcal{X}$ is decomposable.

We next consider the case where the dual of $\mathcal{X}$ forms a bihierarchy. To this end, consider a hypergraph $\mathcal{X}^* = (X^*, \mathcal{H}^*)$ such that $X^* = \mathcal{H}$ and $\mathcal{H}^* = X$. That is, X^* is a finite ground set whose elements share the same labels as the elements in $\mathcal{H}$, and $\mathcal{H}^*$ is a collection of subsets of $\mathcal{H}^*$ that have the same labels as X. Assume that $S \in X^*$ is an element of $x \in \mathcal{H}^*$ in $\mathcal{X}^*$ if and only if x is an element of S in $\mathcal{X}$. The fact that the dual of $\mathcal{X}$ forms a bihierarchy means that (the primal of) $\mathcal{X}^*$ forms a bihierarchy. The argument made above then implies that the incidence matrix A^* associated with $\mathcal{X}^*$ is totally unimodular. Since this matrix coincides with the incidence matrix of the dual of $\mathcal{X}$, $A^* = A^T$. Since a transpose of a totally unimodular matrix is totally unimodular in general by definition, it follows that the incidence matrix A of $\mathcal{X}$ must be also totally unimodular. Hence, by an analogous argument to that above, the hypergraph $\mathcal{X}$ is decomposable. □

Appendix B. Algorithm for Implementing Random Assignments

This section provides a computable algorithm, which also serves as a constructive proof for Theorem 5 for the bihierarchy case (and hence Theorem 1).

Let $(X, \mathcal{H})$ be a hypergraph and assume that $\mathcal{H}$ is a bihierarchy, where $\mathcal{H}_1$ and $\mathcal{H}_2$ are hierarchies such that $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$. Let $P = [P_x]$ be a random assignment whose entries sum up to an integer (the generalization to the case with a fractional sum is straightforward). We construct a flow network as follows. The set of vertices is composed of the source s and the sink s', two vertices v_x and v'_x for each element $x \in X$, and v_S for each $S \in \mathcal{H} \setminus [(\bigcup_{x \in X}\{x\}) \cup (N \times O)]$. We place (directed) edges according to the following rule.[23]

(1) For each $x \in X$, an edge $e = (v_x, v'_x)$ is placed from v_x to v'_x.
(2) An edge $e = (v_S, v_{S'})$ is placed from S to $S' \neq S$ where $S, S' \in \mathcal{H}_1$, if $S' \subset S$ and there is no $S'' \in \mathcal{H}_1$ where $S' \subset S'' \subset S$.[24]

[23] An edge is defined as an ordered pair of verticies. All edges in this paper are directed, so we omit the adjective "directed."

[24] For the purpose of placing edges, we regard v_x as a vertex corresonding to a singleton set $\{x\} \in H_1$, and v'_x as a vertex corresonding to a singleton set $\{x\} \in H_2$.

(3) An edge $e = (v_S, v_{S'})$ is placed from S to $S' \neq S$ where $S, S' \in \mathcal{H}_2$, if $S \subset S'$ and there is no $S'' \in \mathcal{H}_2$ where $S \subset S'' \subset S'$.
(4) An edge $e = (s, v_S)$ is placed from the source s to v_S if $S \in \mathcal{H}_1$ and there is no $S' \in \mathcal{H}_1$ where $S \subset S'$.
(5) An edge $e = (v_S, s')$ is placed from v_S to the sink s' if $S \in \mathcal{H}_2$ and there is no $S' \in \mathcal{H}_2$ where $S \subset S'$.

We associate flow with each edge as follows. For each $e = (v_x, v'_x)$, we associate flow $P_e = P_x$. For each e that is not of the form (v_x, v'_x) for some $x \in X$, the flow P_e is (uniquely) set to satisfy the flow conservation, that is, for each vertex v different from s and s', the sum of flows into v is equal to the sum of flows from v. Observe that the construction of the network (specifically items (2)-(5) above) utilizes the fact that $\mathcal{H}$ is a bihierarchy.

We define the **degree of integrality** of P with respect to $\mathcal{H}$:

$$\deg[P(\mathcal{H})] := \#\{S \in \mathcal{H} | P_S \in \mathbb{Z}\}.$$

Lemma 4. *(Decomposition) Suppose a hypergraph $\mathcal{X} = (X, \mathcal{H})$ forms a bihierarchy. Then, for any P such that $\deg[P(\mathcal{H})] < |\mathcal{H}|$, there exist P^1 and P^2 and $\gamma \in (0,1)$ such that*

(i) $P = \gamma P^1 + (1-\gamma)P^2$:
(ii) $P^1_S, P^2_S \in [\lfloor P_S \rfloor, \lceil P_S \rceil]$, $\forall S \in \mathcal{H}$.
(iii) $\deg[P^i(\mathcal{H})] > \deg[P(\mathcal{H})]$ for $i = 1, 2$.

The following algorithm gives a constructive proof of Lemma 4 and hence the Theorem. Let P be a random assignment on a bihierarchy $\mathcal{H}$ with $\deg[P(\mathcal{H})] < |\mathcal{H}|$.

☐ **Decomposition Algorithm**

(1) **Cycle-Finding Procedure**
   (a) **Step 0:** Since $\deg[P(\mathcal{H})] < |\mathcal{H}|$ by assumption, there exists an edge $e_1 = (v_1, v'_1)$ such that its associated flow P_{e_1} is fractional. Define an edge $f_1 = (v_1, v'_1)$ from v_1 to v'_1.
   (b) **Step t=1,...:** Consider the vertex v'_t that is the destination of edge f_t.
      (i) If v'_t is the origin of some edge $f_{t'} \in \{f_1, \dots, f_{t-1}\}$, then stop.[25] The procedure has formed a cycle $(f_{t'}, f_{t'+1}, \dots, f_t)$ composed of edges in $\{f_1, \dots, f_t\}$. Proceed to **Termination - Cycle**.

[25]Since there are a finite number of vertices, this procedure terminates in a finite number of steps.

(ii) Otherwise, since the flow associated with f_t is fractional by construction and the flow conservation holds at v'_t, there exists an edge $e_{t+1} = (u_{t+1}, u'_{t+1}) \neq e_t$ with fractional flow such that v'_t is either its origin or destination. Draw an edge f_{t+1} by $f_{t+1} = e_{t+1}$ if v'_t is the origin of e_{t+1} and $f_{t+1} = (u'_{t+1}, u_{t+1})$ otherwise. Denote $f_{t+1} = (v_{t+1}, v'_{t+1})$.

(2) **Termination - Cycle**

(a) Construct a set of flows associated with edges (P^1_e) which is the same as (P_e), except for flows $(P_{e_\tau})_{t' \leq \tau \leq t}$, that is, flows associated with edges that are involved in the cycle from the last step. For each edge e_τ such that $f_\tau = e_\tau$, set $P^1_{e_\tau} = P_{e_\tau} + \alpha$, and each edge e_τ such that $f_\tau \neq e_\tau$, set $P^1_{e_\tau} = P_{e_\tau} - \alpha$, where $\alpha > 0$ is the largest number such that the induced random assignment $P^1 = (P^1_x)_{x \in X}$ still satisfies all constraints in $\mathcal{H}$. By construction, $P^1_S = P_S$ if P_S is an integer, and there is at least one constraint set $S \in \mathcal{H}$ such that P^1_S is an integer while P_S is not. Thus $\deg[P^1(\mathcal{H})] > \deg[P(\mathcal{H})]$.

(b) Construct a set of flows associated with edges (P^2_e) which is the same as (P_e), except for flows $(P_{e_\tau})_{t' \leq \tau \leq t}$, that is, flows associated with edges that are involved in the cycle from the last step. For each edge e_τ such that $f_\tau = e_\tau$, set $P^1_{e_\tau} = P_{e_\tau} - \beta$, and each edge e_τ such that $f_\tau \neq e_\tau$, set $P^1_{e_\tau} = P_{e_\tau} + \beta$, where $\beta > 0$ is the largest number such that the induced random assignment $P^2 = (P^2_x)_{x \in X}$ still satisfies all constraints in $\mathcal{H}$. By construction, $P^2_S = P_S$ if P_S is an integer, and there is at least one constraint set $S \in \mathcal{H}$ such that P^2_S is an integer while P_S is not. Thus $\deg[P^2(\mathcal{H})] > \deg[P(\mathcal{H})]$.

(c) Set γ by $\gamma\alpha + (1-\gamma)(-\beta) = 0$, i.e., $\gamma = \frac{\beta}{\alpha+\beta}$.

(d) The decomposition of P into $P = \gamma P^1 + (1-\gamma)P^2$ satisfies the requirements of the Lemma by construction.

Appendix C. Proof of Propositions2 and 3

As with Bogomolnaia and Moulin (2001), a different characterization of ordinal efficiency proves useful. To this end, we first define the **minimal constraint set containing** (i, a):

$$\nu(i,a) := \bigcap_{S \in \mathcal{H}(i,a)} S,$$

if the set $\mathcal{H}(i,a) := \{S \in \mathcal{H}_O : (i,a) \in S, \sum_{(j,b) \in S} P_{jb} = \overline{q}_S\}$ is nonempty. If $\mathcal{H}(i,a) = \emptyset$ (or equivalently $\sum_{(j,b) \in S} P_{jb} < \overline{q}_S$ for all $S \in \mathcal{H}_O$ containing (i,a)), then we let $\nu(i,a) = N \times O$.

We next define the following binary relations on $N \times O$ given (Γ, P) as follows:[26]

$$(j,b) \rhd_1 (i,a) \iff i = j, b \succ_i a, \text{ and } P_{ia} > 0,$$
$$(j,b) \rhd_2 (i,a) \iff \nu(j,b) \subseteq \nu(i,a). \tag{C.1}$$

We then say

$$(j,b) \rhd (i,a) \iff (j,b) \rhd_1 (i,a) \text{ or } (j,b) \rhd_2 (i,a).$$

We say a binary relation $\rhd$ is **strongly cyclic** if there exists a finite cycle $(i_0, a_0) \rhd (i_1, a_1) \rhd \cdots \rhd (i_k, a_k) \rhd (i_0, a_0)$ such that $\rhd = \rhd_1$ for at least one relation. We next provide a characterization of ordinal efficiency.

Lemma 5. *Random assignment $P \in \mathcal{P}_{\mathcal{E}}$ is ordinally efficient if and only if $\rhd$ is not strongly cyclic given (Γ, P).*[27]

A remark is in order. In the environment $\mathcal{E}^{BvN}$, Bogomolnaia and Moulin (2001) define the binary relation $\rhd$ over the set of objects where $b \rhd a$ if there is an agent i such that $b \succ_i b$ and $P_{ia} > 0$. Bogomolnaia and Moulin show that, in $\mathcal{E}^{BvN}$, a random assignment is ordinally efficient if and only if $\rhd$ is acyclic. Our contribution over their characterization is that we expand the domain over which the binary relation is defined to the set of agent-good pairs, in order to capture the complexity that results from a more general environment than $\mathcal{E}^{BvN}$.

Proof of Lemma 5. **"Only if" part.** First note that the following property holds.

Claim 1. *$\rhd_1$ and $\rhd_2$ are transitive, that is,*

$$(k,c) \rhd_1 (j,b), (j,b) \rhd_1 (i,a) \Rightarrow (k,c) \rhd_1 (i,a),$$
$$(k,c) \rhd_2 (j,b), (j,b) \rhd_2 (i,a) \Rightarrow (k,c) \rhd_2 (i,a).$$

Proof. Suppose $(k,c) \rhd_1 (j,b)$ and $(j,b) \rhd_1 (i,a)$. Then, by definition of $\rhd_1$, we have $i = j = k$ and (i) $c \succ_i b$ since $(k,c) \rhd_1 (i,b)$ and (ii) $b \succ_i a$ since $(j,b) \rhd_1 (i,a)$. Thus $c \succ_i a$. Since $(j,b) \rhd_1 (i,a)$, we have $P_{ia} > 0$. Therefore $(k,c) \rhd_1 (i,a)$ by definition of $\rhd_1$.

[26] Given that $\mathcal{H}_O$ has a hierarchical structure,

$$(j,b) \rhd_2 (i,a) \iff (j,b) \in S \text{ for any } S \in \mathcal{H}_O \text{ such that } (i,a) \in S, P_S = \overline{q}_S.$$

[27] In Kojima and Manea (2008), ordinal efficiency is characterized by two conditions, acyclicity and non-wastefulness. We do not need non-wastefulness as a separate axiom in our current formulation since a "wasteful" random assignment (in their sense) contains a strong cycle as defined here.

Suppose $(k,c) \rhd_2 (j,b)$ and $(j,b) \rhd_2 (i,a)$. Then $\nu(k,c) \subseteq \nu(j,b)$ and $\nu(j,b) \subseteq \nu(i,a)$ by property (C.1). Hence $\nu(k,c) \subseteq \nu(i,a)$ which is equivalent to $(k,c) \rhd_2 (i,a)$, completing the proof by property (C.1). □

To show the "only if" part of the Proposition, suppose $\rhd$ is strongly cyclic. By Claim 1, there exists a cycle of the form

$$(i_0,b_0) \rhd_1 (i_0,a_0) \rhd_2 (i_1,b_1) \rhd_1 (i_1,a_1) \rhd_2 (i_2,b_2) \rhd_1 (i_2,a_2) \rhd_2 \cdots \rhd_1 (i_k,a_k) \rhd_2 (i_0,b_0),$$

in which every pair (i,a) in the cycle appears exactly once except for (i_0,b_0) which appears exactly twice, namely in the beginning and in the end of the cycle. Then there exists $\delta > 0$ such that a matrix Q defined by

$$Q_{ia} = \begin{cases} P_{ia} + \delta & \text{if} \quad (i,a) \in \{(i_0,b_0),(i_1,b_1),\ldots,(i_k,b_k)\}, \\ P_{ia} - \delta & \text{if} \quad (i,a) \in \{(i_0,a_0),(i_1,a_1),\ldots,(i_k,a_k)\}, \\ P_{ia} & \text{otherwise}, \end{cases}$$

is in $\mathcal{P}_{\mathcal{E}}$. Since $\delta > 0$ and $b_l \succ_{i_l} a_l$ for every $l \in \{0,1,\ldots,k\}$, Q ordinally dominates P. Therefore P is not ordinally efficient.

"If" part. Suppose P is ordinally inefficient. Then, there exists $Q \in \mathcal{P}_{\mathcal{E}}$ which ordinally dominates P. We then prove that $\rhd$, given (Γ, P), must be strongly cyclic.

(1) **Step 1: Initiate a cycle.**

  (a)

   Claim 2. *There exist* $(i_0,a_0),(i_1,a_1) \in N \times O$ *such that* $i_0 = i_1$, $P_{i_1a_1} < Q_{i_1a_1}$ *and* $(i_1,a_1) \rhd_1 (i_0,a_0)$ *given* (Γ, P).

   Proof. Since Q ordinally dominates P, there exists $(i_1,a_1) \in N \times O$ such that $Q_{i_1a_1} > P_{i_1a_1}$ and $Q_{i_1a} = P_{i_1a}$ for all $a \succ_{i_1} a_1$. So there exists $a_0 \prec_{i_1} a_1$ with $P_{i_1a_0} > Q_{i_1a_0} \geq 0$ since $P_{\{i_1\}\times N} = Q_{\{i_1\}\times N}$ by assumption. Hence, we have $(i_1,a_1) \rhd_1 (i_1,a_0) = (i_0,a_0)$ given (Γ, P). □

  (b) If $(i_0,a_0) \in \nu(i_1,a_1)$, then $(i_0,a_0) \rhd_2 (i_1,a_1) \rhd_1 (i_0,a_0)$, so we have a strong cycle and we are done.

  (c) Else, circle (i_1,a_1) and go to Step 2.

(2) **Step $t+1$ ($t \in \{1,2\ldots\}$): Consider the following cases.**

  (a) Suppose (i_t,a_t) is circled.

   (i)

Claim 3. *There exists* $(i_{t+1}, a_{t+1}) \in \nu(i_t, a_t)$ *such that* $P_{i_{t+1}a_{t+1}} > Q_{i_{t+1}a_{t+1}}$. *Hence,* $(i_{t+1}, a_{t+1}) \rhd_2 \nu(i_t, a_t)$.

Proof. Note that $\nu(i_t, a_t) \subsetneqq N \times O$ since if $\nu(i_t, a_t) = N \times O$, then there exists $(i_{t'}, a_{t'})$ with $t' < t$ and $(i_{t'}, a_{t'}) \in \nu(i_t, a_t)$, so we have terminated the algorithm. Thus we have $\sum_{(i,a)\in\nu(i_t,a_t)} P_{ia} = \overline{q}_{\nu(i_t,a_t)}$. Since $P_{i_t a_t} < Q_{i_t a_t}$, there exists $(i_{t+1}, a_{t+1}) \in \nu(i_t, a_t)$ such that $P_{i_{t+1}a_{t+1}} > Q_{i_{t+1}a_{t+1}}$. □

(ii) If $(i_{t'}, a_{t'}) \in \nu(i_{t+1}, a_{t+1})$ for $t' < t$, then we have a strong cycle, $(i_{t'}, a_{t'}) \rhd (i_{t+1}, a_{t+1}) \rhd ... \rhd (i_{t'}, a_{t'})$, and at least one $\rhd$ is $\rhd_1$, so we are done.

(iii) Else, square (i_{t+1}, a_{t+1}) and move to the next step.

(b) Case 2: Suppose (i_t, a_t) is squared.

(i)

Claim 4. *There exists* $(i_{t+1}, a_{t+1}) \in \nu(i_t, a_t)$ *such that* $i_{t+1} = i_t$, $P_{i_{t+1}a_{t+1}} < Q_{i_{t+1}a_{t+1}}$, *and* $(i_{t+1}, a_{t+1}) \rhd_1 \nu(i_t, a_t)$.

Proof. Since (i_t, a_t) is squared, by Claim 3, $P_{i_t a_t} > Q_{i_t a_t}$. Since Q ordinally dominates P, there must be $(i_{t+1}, a_{t+1}) \in \nu(i_t, a_t)$ with $i_{t+1} = i_t$ such that $P_{i_{t+1}a_{t+1}} < Q_{i_{t+1}a_{t+1}}$, and $a_{t+1} \succ_{i_t} a_t$. Since $P_{i_t a_t} > Q_{i_t a_t} \geq 0$, we thus have $(i_{t+1}, a_{t+1}) \rhd_1 \nu(i_t, a_t)$. □

(ii) If $(i_{t'}, a_{t'}) \in \nu(i_{t+1}, a_{t+1})$ for $t' \leq t$, then we have a strong cycle, $(i_{t'}, a_{t'}) \rhd (i_{t+1}, a_{t+1}) \rhd ... \rhd (i_{t'}, a_{t'})$, and at least one $\rhd$ is $\rhd_1$, so we are done.

(iii) Else, circle (i_{t+1}, a_{t+1}) and move to the next step.

The process must end in finite steps and, at the end we must have a strong cycle. □

Given the above lemma, we are ready to proceed to the proofs of Propositions 2 and 3.

Proof of Proposition 2. Although the proof is a relatively simple modification of Theorem 1 of Bogomolnaia and Moulin (2001), we present the proof for completeness. We prove the claim by contradiction. Suppose that $PS(\succ)$ is ordinally inefficient for some $\succ$. Then, by Proposition 5 and Claim 1 there exists a strong cycle

$$(i_0, b_0) \rhd_1 (i_0, a_0) \rhd_2 (i_1, b_1) \rhd_1 (i_1, a_1) \rhd_2 (i_2, b_2) \rhd_1 (i_2, a_2) \rhd_2 \cdots \rhd_1 (i_k, a_k) \rhd_2 (i_0, b_0),$$

in which every pair (i, a) appears exactly once except for (i_0, b_0) which appears exactly twice, namely in the beginning and the end of the cycle. Let v^l and w^l be the steps of the symmetric simultaneous eating algorithm at which (i_l, a_l) and (i_l, b_l) become unavailable, respectively (that is, $(i_l, a_l) \in S^{v_l - 1} \setminus S^{v_l}$ and $(i_l, a_l) \in S^{w_l - 1} \setminus S^{w_l}$.) Since $(i_l, b_l) \rhd_1 (i_l, a_l)$, by the definition of the algorithm we have $w^l < v^l$ for each $l \in \{0, 1, \dots, k\}$. Also, by $(i_l, a_l) \rhd_2 (i_{l+1}, b_{l+1})$, we have $v^l \leq w^{l+1}$ for any $l = \{0, 1, \dots, k\}$ (with notational convention $(i_{k+1}, a_{k+1}) = (i_0, a_0)$.) Combining these inequalities we obtain $w^0 < v^0 \leq w^1 < v^1 \leq \cdots \leq w^k < v^k \leq w^{k+1} = w^0$, a contradiction. □

Proof of Proposition 3. Let $P = PS(\succ)$. Fix $i \in N$ and let O be ordered in the decreasing order of $\succ_i$, that is, $a_1 \succ_i a_2 \succ_i \cdots \succ_i a_{|N|}$. Let v_1 be the step in which i stops receiving probability share of a_1. In that step we have $P_{ia_1} = P^{v_1}_{ia_1} = t^{v_1}$ and there is $S_1 \in \mathcal{H}_O$ such that $(i, a_1) \in S_1$ and $P^{v_1}_{S_1} = \overline{q}_{S_1}$. Suppose $P_{ja_1} > P_{ia_1}$ for some $j \in N$. Then we have $(j, a_1) \notin S_1$ since $P_{ja_1} \leq t^{v_1} = P_{ia_1}$ if $(j, a_1) \in S_1$ by definition of the algorithm. Also $S_1 = N_1 \times \{a_1\}$ for some $N_1 \subseteq N$ with $i \in N_1$ and $j \notin N_1$ since $(i, a_1) \in S_1$ and $(j, a_1) \notin S_1$. Let Q be defined as in (4.5). Then, since $i \in N_1$ and $j \notin N_1$,

$$\begin{aligned} Q_{S_1} &\geq \sum_{k \in N_1} P_{ka_1} - P_{ia_1} + P_{ja_1} \\ &> \sum_{k \in N_1} P_{ka_1} \\ &\geq \sum_{k \in N_1} P^{v_1}_{ka_1} \\ &= P^{v_1}_{S_1} = \overline{q}_{S_1}, \end{aligned}$$

which implies that $Q \notin \mathcal{P}_{\mathcal{E}}$.

Let $l \geq 2$ and v_l be the step in which i stops receiving probability share of a_l. In that step we have $\sum_{m=1}^{l} P_{ia_m} = \sum_{m=1}^{l} P^{v_l}_{ia_m} = t^{v_l}$ and there is $S_l \in \mathcal{H}_O$ such that $(i, a_l) \in S_l$ and $P^{v_l}_{S_l} = \overline{q}_{S_l}$. Suppose $\sum_{m=1}^{m'} P_{ja_m} \leq \sum_{m=1}^{m'} P_{ia_m}$ for all $m' \leq l-1$ and $\sum_{m=1}^{l} P_{ja_m} > \sum_{m=1}^{l} P_{ia_m}$ for some $j \in N$. Then we have $(j, a_l) \notin S_l$ since $\sum_{m=1}^{l} P_{ja_m} \leq t^{v_l} = \sum_{m=1}^{l} P_{ia_m}$ if $(j, a_l) \in S_l$ by definition of the algorithm. Also $S_l = N_l \times \{a_l\}$ for some $N_l \subseteq N$ with $i \in N_l$ and $j \notin N_l$ since $(i, a_l) \in S_l$ and $(j, a_l) \notin S_l$. Let Q be defined as in (4.5). Then, since $i \in N_l$ and $j \notin N_l$,

$$
\begin{aligned}
Q_{S_l} &\geq \sum_{k\in N_l} P_{ka_l} - P_{ia_l} + P_{ja_l} \\
&> \sum_{k\in N_l} P_{ka_l} \\
&\geq \sum_{k\in N_l} P^{v_l}_{ka_l} \\
&= P^{v_l}_{S_l} = \overline{q}_{S_l},
\end{aligned}
$$

which implies that $Q \notin \mathcal{P}_{\mathcal{E}}$. By induction, we complete the proof. □

Appendix D. Proofs of Lemma 1 and Theorem 2

Proof of Lemma 1. Suppose for contradiction that $\mathcal{H}$ is decomposable and contains the odd cycle $S^1, \ldots, S^l$, with $x_i \in S_i \cap S_{i+1}$, $i = 1, \ldots, l-1$ and $x_l \in S_l \cap S_1$. Consider a random assignment P specified by

$$
P_x = \begin{cases} \frac{1}{2} & \text{if } x \in \{x_1, \ldots, x_l\}, \\ 0 & \text{otherwise,} \end{cases}
$$

where P_x is the entry corresponding to $x \in N \times O$. By definition of an odd cycle, $P_{S_i} = 1$ for all $i \in \{1, \ldots, k\}$. Since $\mathcal{H}$ is decomposable, there exist $P^1, P^2, \ldots, P^K$ and $\lambda^1, \lambda^2, \ldots, \lambda^K$ such that

(1) $P = \sum_{k=1}^{K} \lambda^k P^k$,
(2) $\lambda^k \in (0, 1]$ for all k and $\sum_{k=1}^{K} \lambda^k = 1$,
(3) $P^k_S \in \{\lfloor P_S \rfloor, \lceil P_S \rceil\}$ for all $k \in \{1, \ldots, K\}$ and $S \in \mathcal{H}$.

In particular, it follows that $P^k_{S_i} = 1$ for each i and k. Thus there exists k such that $P^k_{x_1} = 1$. Since $P^k_{S_2} = 1$, it follows that $P^k_{x_2} = 0$. The latter equality and the assumption that $P^k_{S_3} = 1$ imply $P^k_{x_3} = 1$. Arguing inductively, it follows that $P^k_{x_i} = 0$ if i is even and $P^k_{x_i} = 1$ if i is odd. In particular, we obtain $P^k_{x_l} = 1$ since l is odd by assumption. Thus $P^k_{S_l} = P^k_{x_l} + P^k_{x_1} >= 2$, contradicting $P^k_{S_l} = 1$. □

Proof of Theorem 2. In order to prove the Theorem, we study several cases.

- Assume there is $S \in \mathcal{H}$ such that $S = N' \times O'$ where $2 \leq |N'| < |N|$ and $2 \leq |O'| < |O|$. Let $\{i, j\} \times \{a, b\} \subseteq S$, $k \notin N'$ and $c \notin O'$ (observe that such $i, j, k \in N$ and $a, b, c \in O$ exist by the assumption of this case). Then the sequence of constraint sets

$$S_1 = S, S_2 = \{i\} \times O, S_3 = N \times \{c\}, S_4 = \{k\} \times O, S_5 = N \times \{b\},$$

is an odd cycle together with

$$x_1 = (i,a), x_2 = (i,c), x_3 = (k,c), x_4 = (k,b), x_5 = (j,b).$$

Therefore, by Lemma 1, $\mathcal{H}$ is not decomposable.

- Assume there is $S \in \mathcal{H}$ such that, for some $i, j \in N$ and $a, b \in O$, we have $(i,a), (j,b) \in S$ with $i \neq j$ and $a \neq b$, and $(i,b) \notin S$. Then the sequence of constraint sets

$$S_1 = S, S_2 = \{i\} \times O, S_3 = N \times \{b\},$$

is an odd cycle together with

$$x_1 = (i,a), x_2 = (i,b), x_3 = (j,b).$$

Thus, by Lemma 1, $\mathcal{H}$ is not decomposable.

By the above arguments, it suffices to consider cases where all constraint sets in $\mathcal{H}$ have one of the following forms.

(1) $\{i\} \times O'$ where $i \in N$ and $O' \subseteq O$,
(2) $N' \times O$ where $N' \subseteq N$,
(3) $N' \times \{a\}$ where $a \in O$ and $N' \subseteq N$,
(4) $N \times O'$ where $O' \subseteq O$.

Therefore it suffices to consider the following cases.

(1) Assume that there are $S', S'' \in \mathcal{H}$ such that $S' = \{i\} \times O'$ and $S'' = \{i\} \times O''$ for some $i \in N$ and some $O', O'' \subset O$, $S' \cap S'' \neq \emptyset$ and S' is neither a subset nor a superset of S''. Then we can find $a, b, c \in O$ such that $a \in O' \setminus O''$, $b \in O' \cap O''$ and $c \in O'' \setminus O'$. Fix $j \neq i$, who exists by assumption $|N| \geq 2$. Then the sequence of constraint sets

$$S_1 = S', S_2 = S'', S_3 = N \times \{c\}, S_4 = \{j\} \times O, S_5 = N \times \{a\},$$

is an odd cycle together with

$$x_1 = (i,a), x_2 = (i,b), x_3 = (i,c), x_4 = (j,c), x_5 = (j,a).$$

Therefore, by Lemma 1, $\mathcal{H}$ is not decomposable.

(2) Assume that there are $S', S'' \in \mathcal{H}$ such that $S' = N' \times O$ and $S'' = N'' \times O$ for some $N', N'' \subset N$, $S' \cap S'' \neq \emptyset$ and S' is neither a subset nor a superset of S''. In such a case, we can find $i, j, k \in N$ such that $i \in N' \setminus N''$, $j \in N' \cap N''$ and $k \in N'' \setminus N'$. Fix $a, b \in O$. The sequence of constraint sets

$$S_1 = S', S_2 = S'', S_3 = N \times \{b\},$$

is an odd cycle together with

$$x_1 = (j, a), x_2 = (k, b), x_3 = (i, b).$$

Hence, by Lemma 1, $\mathcal{H}$ is not decomposable.

(3) Assume that there are $S', S'' \in \mathcal{H}$ such that $S' = N' \times \{a\}$ and $S'' = N'' \times \{a\}$ for some $a \in O$ and some $N', N'' \subset N$, $S' \cap S'' \neq \emptyset$ and S' is neither a subset nor a superset of S''. This is a symmetric situation with Case 1, so an analogous argument as before goes through.

(4) Assume that there are $S', S'' \in \mathcal{H}$ such that $S' = N \times O'$ and $S'' = N \times O''$ for some $O', O'' \subset O$, $S' \cap S'' \neq \emptyset$ and S' is neither a subset nor a superset of S''. This is a symmetric situation with Case 2, so an analogous argument as before goes through.

□

Appendix E. Proof of Proposition 5

(1) Define a price space, $\mathcal{P} = [0, |N|b^*]^{|O|}$.

(2) Define for each agent i his demand correspondence

$$d_i^*(p) := \left\{ (y_{ia})_{a \in O} \in \mathbb{R}_+^{|O|} \middle| \exists (x_{ika})_{k \in K_i, a \in O} \in X_i(p) \text{ s.t. } \sum_{k \in K_i} x_{ika} = y_{ia}, \forall a \in O \right\}.$$

We show that $d_i^*(p)$ is nonempty and convex for all $p \in \mathbb{R}_+^{|O|}$, and upper hemicontinuous in p. Proof. The demand correspondence is nonempty since the feasible set of the linear program is nonempty (since zero demand for each $(k, a) \in K_i \times O$ is feasible), since it is bounded (since $0 \leq x_{ika} \leq \overline{q}_{ika}$) and closed, so it is compact, and since the objective function is continuous. The convexity of $d_i^*(p)$ is shown as follows. Suppose $y, y' \in d_i^*(p)$. Then, there are $(x_{ika})_{k \in K_i, a \in O}, (x'_{ika})_{k \in K_i, a \in O} \in X_i(p)$ such that $x_{ika} = y_{ia}$ and $x'_{ika} = y'_{ia}$ for each $a \in O$. Now fix any $s \in (0, 1)$. Since the feasible set of the linear program is convex and since its objective function is quasi-concave (since it is linear), $X_i(p)$ is convex. Hence, $s(x_{ika})_{k \in K_i, a \in O} + (1 - s)(x'_{ika})_{k \in K_i, a \in O} \in X_i(p)$. It follows that $sy + (1 - s)y' =$

$s(x_{ika})_{k\in K_i,a\in O} + (1-s)(x'_{ika})_{k\in K_i,a\in O} \in d_i^*(p)$. We thus conclude that $d_i^*(p)$ is convex for any $p \in \mathbb{R}_+^{|O|}$. The upper-hemicontinuity of $d_i^*(\cdot)$ is shown as follows. We first note that the correspondence $X_i(p)$ is upper-hemicontinuous. This follows from Berge's theorem of the maximum, since the objective function is continuous and since the correspondence mapping from price vectors to the set of feasible $(x_{ika})_{k\in K_i,a\in O}$'s is continuous (ie., upper- and lower-hemicontinuous). Since $d_i^*(\cdot)$ is a continuous transformation of $X_i(\cdot)$, $d_i^*(\cdot)$ is upper-hemicontinuous. Q.E.D.

(3) Define the excess demand correspondence $z(p)$ in the usual way: $z(p) = \sum_i d_i^*(p) - \mathbf{q}$. Note that it too is upper hemicontinuous and convex (as the linear sum of upper hemicontinuous and convex functions)

(4) To handle some boundary issues that arise because we allow goods to be in excess supply at price zero (and preferences are satiable, so prices of zero may actually arise), do the following:
   (a) Let $\bar{q} = \max_{a\in O} q_a$. It is wlog to restrict that each agent can demand at most $\bar{q}$ copies of any object.[28]
   (b) Define an auxiliary enlargement of the price space, $\tilde{\mathcal{P}} = [-\bar{q}, |N|b^* + |N|\bar{q}]^{|O|}$
   (c) Define a truncation function $t : \tilde{\mathcal{P}} \to \mathcal{P}$ by

$$t(p) = \left(\max\left(0, \min(p_1, |N|b^*)\right), \ldots, \max\left(0, \min(p_{|O|}, |N|b^*)\right)\right)$$

(5) Define a correspondence $f : \tilde{\mathcal{P}} \to \tilde{\mathcal{P}}$ by $f(p) = t(p) + z(t(p))$. To show that we can apply Kakutani: First, note that $z(t(p))$ is uhc and convex on $\tilde{\mathcal{P}}$; all the truncation does is make $z(t(p))$ "flat" for $p \in \tilde{\mathcal{P}}\backslash\mathcal{P}$. This implies that $f(p)$ is uhc and convex as well. Second, note that the range of $t(p) + z(t(p))$ for $p \in \tilde{\mathcal{P}}$ lies in $\tilde{\mathcal{P}}$ as required, because the excess demand function is bounded below by $-\bar{q}$ and above by $|N|\bar{q}$. Thus $f(p)$ is a uhc and convex correspondence defined on the compact and convex set $\tilde{\mathcal{P}}$. Thus we can apply Kakutani: there exists $p^* \in f(p^*)$.

(6) The last step is to show that this fixed point yields a competitive equilibrium; specifically, $t(p^*)$ is a competitive equilibrium price vector. First, for arbitrary $a \in O$, suppose that $p_a^* \in [0, |N|b^*]$, i.e., that the truncation doesn't bite for good a. Then p^* being a fixed point implies that $z_a(t(p^*)) = 0$, i.e., the market for good a exactly clears at $t(p^*)$. Second, suppose that $p_a^* < 0$. Then p^* being a fixed point implies that $z_a(t(p^*)) = p_a^* < 0$, i.e., good a is in excess supply at $t(p^*)$. This excess supply is fine because $t_a(p^*) = 0$. Last, suppose that $p_a^* > |N|b^*$. Then p^* being a fixed point implies that $z_a(t(p^*)) = p_a^* - |N|b^* > 0$, i.e., good a is in

[28] In course allocation, we assume that each agent can demand at most a single copy of any object.

excess demand at $t(p^*)$. But this is impossible because $t_a(p^*) = |N|b^*$, so even if all agents spend their entire budget on good a at price vector $t(p^*)$, total demand is less than or equal to one (which is weakly less than supply by assumption).

Appendix F. Proof of Theorem 3

Proof. For each $i \in N$, let $(a_i^1, a_i^2, \dots, a_i^{|O|})$ be a sequence of objects in decreasing order of i's preferences so that $v_{ia_i^1} \geq v_{ia_i^2} \geq \dots, v_{ia_i^{|O|}}$. Define the class of sets $\mathcal{H}' = \mathcal{H}'_N \cup \mathcal{H}'_O$ by

$$\mathcal{H}'_N = \mathcal{H}_N \cup \left(\bigcup_{\substack{i \in N, \\ k \in \{1, \dots, |O|\}}} \{i\} \times \{a_i^1, \dots, a_i^k\} \right),$$
$$\mathcal{H}'_O = \mathcal{H}_O.$$

By inspection, $\mathcal{H}'$ is a bihierarchy. Therefore, by Theorem 1, there exists a convex decomposition such that

$$\sum_{(i,a) \in S} P'_{ia}, \sum_{(i,a) \in S} P''_{ia} \in \left\{ \left\lfloor \sum_{(i,a) \in S} P_{ia} \right\rfloor, \left\lceil \sum_{(i,a) \in S} P_{ia} \right\rceil \right\} \quad \text{for all } S \in \mathcal{H}', \tag{F.1}$$

for any integer-valued matrices P' and P'' that are part of the decomposition. In particular, property (F.1) holds for each $\{(i,a)\} \in \mathcal{H}'_N$ and $\{i\} \times \{a_i^1, \dots, a_i^k\} \in \mathcal{H}'_N$. This means that

- **Observation 1:** For any i and k, $P'_{ia_i^k} - P''_{ia_i^k} \in \{-1, 0, 1\}$. This follows from the fact that $|P'_{ia_i^k} - P''_{ia_i^k}| \leq \lceil P_{ia_i^k} \rceil - \lfloor P_{ia_i^k} \rfloor \leq 1$ and that $P'_{ia_i^k}$ and $P''_{ia_i^k}$ are integer valued.
- **Observation 2:** By the same logic as for Observation 1, it follows that $\sum_{j=1}^k (P'_{ia_i^j} - P''_{ia_i^j}) \in \{-1, 0, 1\}$ for any i and k.
- **Observation 3:** Let $(a_i^{k_l})_{l=1}^{\bar{l}}$ be the (largest) subsequence of $(a_i^1, \dots, a_i^k)$ such that $P'_{ia_i^{k_l}} \neq P''_{ia_i^{k_l}}$ for all l. Then, (i) $P_{ia_i^{k_l}} \notin \mathbb{Z}$ for all l, and (ii) $P'_{ia_i^{k_{2l'}}} - P''_{ia_i^{k_{2l'}}} = -(P'_{ia_i^{k_{2l'-1}}} - P''_{ia_i^{k_{2l'-1}}})$ for any $l' = 1, \dots, \bar{l}/2$.

  Observation 3 (ii) can be shown as follows. First, the result must hold for $l' = 1$, or else $\sum_{j=1}^{k_2} (P'_{ia_i^j} - P''_{ia_i^j}) = P'_{ia_i^{k_1}} - P''_{ia_i^{k_1}} + P'_{ia_i^{k_2}} - P''_{ia_i^{k_2}} \in \{-2, 2\}$, which violates Observation 2. Now, working inductively, suppose the statement holds for all

$l' = 1, \ldots, m-1$ for $m \le \bar{l}/2$. Then the statement must hold for $l' = m$, or else

$$\begin{aligned}
&\sum_{j=1}^{k_{2m}} (P'_{ia_i^j} - P''_{ia_i^j}) \\
= \quad & \sum_{l'=1}^{m-1} \left(P'_{ia_i^{k_{2l'-1}}} - P''_{ia_i^{k_{2l'-1}}} + P'_{ia_i^{k_{2l'}}} - P''_{ia_i^{k_{2l'}}} \right) + P'_{ia_i^{k_{2m-1}}} - P''_{ia_i^{k_{2m-1}}} + P'_{ia_i^{k_{2m}}} - P''_{ia_i^{k_{2m}}} \\
= \quad & P'_{ia_i^{k_{2m-1}}} - P''_{ia_i^{k_{2m-1}}} + P'_{ia_i^{k_{2m}}} - P''_{ia_i^{k_{2m}}}
\end{aligned}$$

must be either -2 or 2, which again violates Observation 2.

These observations imply that

$$\begin{aligned}
\sum_{a \in O} P'_{ia} v_{ia} - \sum_{a \in O} P''_{ia} v_{ia} &= \sum_{k=1}^{|O|} (P'_{ia_i^k} - P''_{ia_i^k}) v_{ia_i^k} \\
&= \sum_{l=1}^{\bar{l}} (P'_{ia_i^{k_l}} - P''_{ia_i^{k_l}}) v_{ia_i^{k_l}} \\
&\le \sum_{l'=1}^{\bar{l}/2} v_{ia_i^{k_{2l'-1}}} - v_{ia_i^{k_{2l'}}} \\
&\le v_{ia_i^{k_1}} - v_{ia_i^{k_{\bar{l}}}} \\
&\le \Delta_i,
\end{aligned}$$

where the first inequality follows from $v_{ia_i^k} \ge v_{ia_i^{k'}}$ for $k < k'$ and Observations 1 and 3-(ii), the second inequality follows from $v_{ia_i^k} \ge v_{ia_i^{k'}}$ for $k < k'$, and the last inequality follows from the definition of Δ_i and Observation 3-(i). Therefore, we obtain property (6.1). Property (6.2) follows immediately from property (6.1). □

References

Abdulkadiroğlu, A. (2005): "College Admission with Affirmative Action," *International Journal of Game Theory*, 33, 535–549.

Abdulkadiroğlu, A., P. A. Pathak, and A. E. Roth (2005): "The New York City High School Match," *American Economic Review Papers and Proceedings*, 95, 364–367.

Abdulkadiroğlu, A., and T. Sönmez (1999): "House Allocation with Existing Tenants," *Journal of Economic Theory*, 88, 233–260.

——— (2003): "School Choice: A Mechanism Design Approach," *American Economic Review*, 93, 729–747.

Birkhoff, G. (1946): "Three Observations on Linear Algebra," *Revi. Univ. Nac. Tucuman, ser A*, 5, 147–151.

Bogomolnaia, A., and H. Moulin (2001): "A New Solution to the Random Assignment Problem," *Journal of Economic Theory*, 100, 295–328.

Budish, E. (2009): "The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes," Harvard University, Unpublished mimeo.

Budish, E., and E. Cantillon (2009): "The Multi-Unit Assignment Problem: Theory and Evidence from Course Allocation at Harvard," Harvard University, Unpublished mimeo.

Edmonds, J. (1970): "Submodular functions, matroids, and certain polyhedra," *Combinatorial Structures and Their Applications, R. Guy, H. Hanani, N. Sauer, and J. Schonheim, eds, Gordon and Breach, New York*, pp. 69–87.

Ehlers, L., and B. Klaus (2003): "Coalitional strategy-proof and resource-monotonic solutions for multiple assignment problems," *Social Choice and Welfare*, 21, 265–280.

Ghouila-Houri, A. (1962): "Caractérisation des matrices totalment unimodulaires," *Comptes Rendus Hebdomadaires des Séances de l' Académie des Sciences (Paris)*, 254, 1192–1194.

Hatfield, J. W. (2008): "Strategy-Proof, Efficient, and Nonbossy Quota Allocations," forthcoming, *Social Choice and Welfare.*

Hatfield, J. W., and P. Milgrom (2005): "Matching with Contracts," *American Economic Review*, 95, 913–935.

Hoffman, A., and J. Kruskal (1956): "Integral boundary points of convex polyhedra," *in "Linear Inequalities and Related Systems" (H. Kuhn and A. Tucker, Eds.) Annals of Mathematics Studies*, 38, 223–246.

Hylland, A., and R. Zeckhauser (1979): "The Efficient Allocation of Individuals to Positions," *Journal of Political Economy*, 87, 293–314.

Kojima, F. (2008): "Random Assignment of Multiple Indivisible Objects," forthcoming, *Mathematical Social Sciences.*

Kojima, F., and M. Manea (2008): "Incentives in the Probabilistic Serial Mechanism," forthcoming, *Journal of Economic Theory.*

Nemhauser, G., and M. Trick (1998): "Scheduling a Major College Basketball Conference," *Operations Research*, 46, 1–8.

Papai, S. (2001): "Strategyproof and nonbossy multiple assignments," *Journal of Public Economic Theory*, 3, 257–271.

Roth, A. E., T. Sonmez, and U. Ünver (2005): “Pairwise Kidney Exchange,” *Journal of Economic Theory*, 125, 151–188.

Sönmez, T., and U. Ünver (2008): “Course Bidding at Business Schools,” forthcoming, *International Economic Review.*

von Neumann, J. (1953): “A certain zero-sum two-person game equivalent to the optimal assignment problem,” in *Contributions to the theory of games, Vol. 2*, ed. by H. W. Kuhn, and A. W. Tucker. Princeton University Press, Princeton, New Jersey.