Stability and Pareto Optimality in Refugee Allocation Matchings

Haris Aziz
UNSW Sydney and Data61
Sydney, Australia
haris.aziz@unsw.edu.au

Jiayin Chen
UNSW Sydney and Data61
Sydney, Australia
jiayin.chen@student.unsw.edu.au

Serge Gaspers
UNSW Sydney and Data61
Sydney, Australia
sergeg@cse.unsw.edu.au

Zhaohong Sun
UNSW Sydney and Data61
Sydney, Australia
zhaohong.sun@student.unsw.edu.au

ABSTRACT

We focus on the refugee matching problem—a general "two-sided matching under preferences" model with multi-dimensional feasibility constraints. We propose a taxonomy of stability concepts for the problem; identify relations between them; and show that even for two natural weakenings of the standard stability concept, non-existence and NP-hardness results persist. We then identify several natural weaker stability concepts for which we present a polynomial-time and strategy-proof algorithm that returns a stable matching. We also examine the complexity of computing and testing Pareto optimal matchings.

KEYWORDS

Matching under preferences; stable matching; Pareto optimality; game theory

ACM Reference Format:
Haris Aziz, Jiayin Chen, Serge Gaspers, and Zhaohong Sun. 2018. Stability and Pareto Optimality in Refugee Allocation Matchings. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

Centralized matching markets based on the preferences of the concerned agents have been one of the successful stories of algorithmic economics. These approaches have been successfully deployed in school admissions, placement of hospital-resident, and centralized kidney markets (see e.g., [1, 16, 17, 23]).

In recent years, one of the most pressing issues is the safe and timely placement of refugees in places that can host them. Often, this placement is done in an ad hoc manner where neither the preferences of the refugees nor the hosts is taken into account. For example, the host locality may prefer people who speak the same language and a refugee family may prefer a country with which they have some affinity. This calls for a centralized matching market approach to the refugee allocation problem (see e.g., [13, 19]).

Delacrétaz et al. [7] formalized refugee allocation as a centralized matching market design problem. The problem is more general than the traditional school choice or hospital-resident setting [1] because unlike a school seat that accommodates a single student, a family can only be hosted by a locality, if it can satisfy a multi-dimensional requirement of the family that could involve services such as hospital beds, children's day care, special medical services, etc. Thus the refugee allocation problem is a generalisation of the traditional two-sided matching problem by considering multi-dimensional feasibility constraints. Delacrétaz et al. [7] pointed out that for the refugee allocation problem, the standard stability concept may lead to non-existence of a stable matching. Hence they focus on a weaker stability notion called quasi-stability for which they propose algorithms. In another recent work, Andersson and Ehlers [2] focused on a restricted version of the refugee allocation problem with unidimensional service demands and capacity vectors but with a feature that captures language compatibility of families and hosts. For this setting, they present an algorithm that finds a stable maximum matching.

Although refugee allocation as a matching problem has started receiving interest both in academia and in practice, several important aspects of it need further understanding. One fundamental research question is what is the right stability concept that is weak enough so as to guarantee existence of a stable computable matching but strong enough to lead to reasonable and meaningful outcomes? Secondly, what is the complexity of computing stable matchings for different notions of stability in this model? Similar questions also apply to other standard concepts such as Pareto optimality. We consider these questions in this paper.

Contributions. We first focus on stability in refugee allocation. We present a clear taxonomy of stability concepts for the refugee allocation problem. Two of the concepts (*stability* and *quasi-stability*) have been studied in prior work [7] whereas the others (*strong stability, weak stability, stability by demand, weak stability by demand*) are natural variants of the original two that we propose in this paper. Whereas stability is too stringent to guarantee the existence of a stable matching, quasi-stability is a very weak concept since an empty matching satisfies it. We prove the logical relations between the stability concepts to unify the discussion on stability in refugee allocation (see Figure 1).

We start from stability and weaken it in two orthogonal directions: (1) a deviating family can replace at most one other family, and (2) if a family replaces a set of families, then at most the same number of units of each service are used by the new family as the set of families that are replaced. For each of the weakening operations, the resulting stability notions weak stability and stability by demand still do not guarantee the existence of a stable matching.

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

We additionally show that the problems of checking whether such matchings exist are NP-complete. Based on these negative results, we focus on the stability notion weak stability by demand which is obtained from stability by applying both weakening operations (1) and (2). This notion seems to have some merit over quasi-stability. We show that a weakly stable matching by demand is guaranteed to exist. We also propose a polynomial-time, strategy-proof algorithm for computing a matching that is weakly stable by demand.

We also present two stability concepts that are based on the *master list* principle that defines a global priority over the families and could be based on factors such as the education level of the families or the urgency of their resettlement. Our main algorithm *(Hierarchical Family Proposing Deferred Acceptance (HFPDA))* also achieves stability based on the master list principle.

Since tailor-made algorithms for stability concepts cannot easily be extended to satisfy other feasibility constraints or objectives such as maximizing the number of refugees hosted, we take an integer/constraint programming approach to the problem. This type of approach has only recently gained traction for more restricted settings such as hospital-resident matching with couples [4]. We propose constraint programming formulations for finding stable matchings in this general setting. Our formulations provide a general framework where additional constraints can easily be placed. The constraints also provide simple algorithms for testing stability.

We also focus on Pareto optimal allocations from the families' perspective and show that testing weak Pareto optimality of a given matching as well as Pareto optimality is coNP-complete even for unidimensional constraints. Our results provide a formal justification to the comment by Delacrétaz et al. [7] that finding a Pareto improvement appears to be a challenging task.

Most of our computational results are summarized in Table 1. Before we proceed further, we note that although the model we examine and the terminology we use is inspired by refugee allocation, the model is a very general matching model with multi-dimensional constraints. Hence the results can have other interpretations depending on the application domain.

	Complexity of Testing	Complexity of Computing	Existence Guaranteed
Stability	in P [1]	NP-c [2]	No [1]
Weak Stability	in P	NP-c	No
Stability by demand	in P	NP-c	No
Weak stability by demand	in P	in P	Yes
Quasi-Stability	in P [1]	in P [1]	Yes [1]
Pareto optimality	coNP-c	NP-h	Yes
Weak Pareto optimality	coNP-c	NP-h	Yes

[1] Delacrétaz et al. [7]
[2] McDermid and Manlove [18]

Table 1: Summary of results.

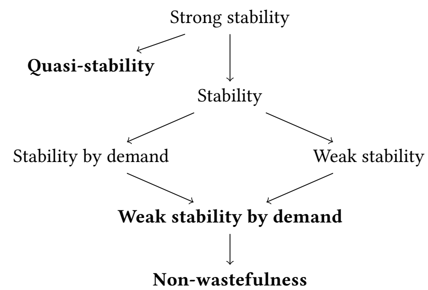

Figure 1: Logical relations between stability concepts. An arrow from (A) to (B) denotes that stability concept (A) implies stability concept (B). The solution concepts in bold guarantee the existence of a corresponding stable matching.

2 MODEL

Let there be a set of refugee families F and a set of localities L. Each family f has a preference ordering $\succsim_f$ over the set of localities L and the option of being unmatched, denoted by $\emptyset$. $\ell \succsim_f \ell'$ means that f prefers ℓ to ℓ' or f is indifferent between ℓ and ℓ'. Each locality ℓ also has a priority ordering $\succsim_\ell$ over the set of families $F \cup \{\emptyset\}$. A locality ℓ is *acceptable* to f if $\ell \succsim_f \emptyset$ and a family f is acceptable to ℓ if $f \succsim_\ell \emptyset$. Let $\succsim$ denote the preference and priority profile of all families and all localities.

Unlike classical two-sided matching problems, different types of services or multidimensional constraints need to be taken into account, e.g., each family may require several units of house rooms, school seats and job vacancies. Let S denote a set of services and let the matrix d denote the service demands of all families. Each row vector d_f corresponds to the demand of family f and each element d_f^s specifies the demand for service s of family f. Let the matrix c denote the service capacities of all localities. Each row vector c_ℓ corresponds to the capacity of locality ℓ and each element c_ℓ^s specifies the locality ℓ's capacity of service s.

A refugee allocation instance consists of a tuple $\mu = (F, L, \succsim, S, d, c)$. A contract $x = (f, \ell)$ is a family-locality pair which implies f and ℓ are matched to each other. An outcome of a refugee allocation instance is a set of contracts $X \subseteq F \times (L \cup \{\emptyset\})$ in which every family gets a contract. Let $L_f(X)$ denote the assignment to family f which is a subset of $L \cup \{\emptyset\}$ and let $F_\ell(X)$ be the set of families matched to ℓ under X. Denote $F(X) = \cup_{\ell \in L} F_\ell(X)$ and $L(X) = \cup_{f \in F} L_f(X)$. Let $F_\ell^{\succ f}(X) = \{f' : f' \in F_\ell, f' \succ_\ell f\}$ be the set of families that are matched to ℓ with higher priority than f and $F_\ell^{\prec f}(X) = \{f' : f' \in F_\ell, f \succ_\ell f'\}$ be the set of families that are matched to ℓ with lower priority than f.

We assume demand or capacity vectors can be compared in this way: For any two vectors $\omega = (\omega_1, ..., \omega_k), \omega' = (\omega'_1, ..., \omega'_k)$, we write $\omega \leq \omega'$ if for each $i \in [1, k]$, $\omega_i \leq \omega'_i$. In other words, a vector ω is smaller or equal to ω' if each element of ω is smaller or equal to the counterpart of ω'. We also write $\omega < \omega'$ if and only if $\omega \leq \omega'$ and $\omega' \nleq \omega$.

An outcome X is *feasible* if (i) for each $f \in F(X)$, $|L_f(X)|=1$ and (ii) for each $\ell \in L(X)$, $\sum_{f \in F_\ell(X)} d_f \leq c_\ell$. In other words, an outcome is feasible if each family is matched with one locality or remains unmatched and each locality is matched with a set of families whose demands do not exceed its capacity. A feasible outcome X is *individually rational* if (i) for each $f \in F$, $L_f(X) \succsim_f \emptyset$, and (ii) for each $\ell \in L$ and for each $f' \in F_\ell(X)$, we have that $f' \succsim_\ell \emptyset$.

That is, no family is matched with an unacceptable locality and no locality is matched with any unacceptable family.

A feasible outcome X is *Pareto optimal* if there is no feasible outcome X' such that $\forall f \in F, L_f(X') \succsim_f L_f(X)$ and $\exists f \in F, L_f(X') \succ_f L_f(X)$. A feasible outcome X is *weakly Pareto optimal* if there is no feasible outcome X' such that $\forall f \in F, L_f(X') \succ_f L_f(X)$. Note that we have defined the Pareto optimality notions from the point of view of the families since they have real preferences whereas the localities can be viewed as having priorities. Our main results concerning these concepts are computational hardness results. Hence they even apply if the localities have preferences (in the form of complete indifference among the families).

A *mechanism* is a function that maps refugee allocation instances to outcomes. A mechanism is feasible if it always produces a feasible outcome for any instance and a mechanism is *strategy-proof* if no family can achieve a more preferred outcome if it misreports its preference. The reason why we consider strategy-proofness only from the view of one side is that it is well-known that there is no stable matching mechanism that is strategy-proof for both sides [1, 21].

3 TAXONOMY OF STABILITY

In this section, we describe a taxonomy of stability concepts and show which concepts can guarantee the existence of stable outcomes. The logical relation of all concepts is summarized in Figure 1.

3.1 Quasi-stability and Strong Stability

The first notion is quasi-stability proposed by Delacrétaz et al. [7] which is an extension of fairness studied in the literature of school choice to the setting of refugee allocation.

Definition 3.1 (Quasi-stability). A feasible outcome X is *quasi-stable* if for each locality $\ell \in L$, for each locality $\ell' \in L \setminus \{\ell\}$, and for each family $f' \in F_{\ell'}(X)$, either $\ell' \succ_{f'} \ell$ or $f \succ_\ell f'$ for all $f \in F_\ell(X)$.

This notion captures the idea that any family and locality pair cannot block an outcome if the family would have the lowest priority in the new locality, even though the new locality can provide sufficient services to accommodate it.

One drawback of this concept is that it allows a family f who cannot be matched to ℓ under any feasible outcome to actually block some outcome with ℓ. Consider one locality ℓ with capacity vector $c_\ell = (1)$ and two families f_1, f_2 with demand vectors $d_{f_1} = (2), d_{f_2} = (1)$. If both families consider ℓ acceptable and ℓ prefers f_1 to f_2, then the only non-empty feasible outcome, $X = (f_2, \ell)$, is not quasi-stable. In addition, quasi-stability can be wasteful since even an empty outcome satisfies it. Wastefulness might be intolerable in practice and it is desirable to accommodate more refugee families rather than fewer or none.

Definition 3.2 (Non-wastefulness). A feasible matching X is *non-wasteful* if there is no pair (f, ℓ) with $f \in F$ and $\ell \in L$ such that (i) $f \succ_\ell \emptyset, \ell \succ_f L_f(X)$ and (ii) $X \cup \{(f, \ell)\} \setminus \{(f, L_f(X))\}$ is feasible.

A feasible matching is non-wasteful if there does not exist any family f and locality ℓ such that f prefers ℓ to its assignment $L_f(X)$ and ℓ still has enough services to accommodate f without removing any matched family at ℓ.

In contrast to quasi-stability, we consider non-wastefulness as an important part of stability and integrate it into the definition of blocking pairs. The following notion of strong blocking pairs is derived by combining the idea of quasi-stability and non-wastefulness.

Definition 3.3 (Strong Stability). For a feasible outcome X, a family $f \in F$ and a locality $\ell \in L$, the pair (f, ℓ) is called a *weakly blocking pair* if $(f, \ell) \notin X$, $\ell \succ_f L_f(X)$ and either i) $\exists f' \in F_\ell(X), f \succ_\ell f'$ or ii) $X \cup \{(f, \ell)\} \setminus \{(f, L_f(X))\}$ is feasible. A feasible outcome X is *strongly stable* if it is individually rational and admits no weakly blocking pair.

We do not advocate strong stability since it inherits the drawback from quasi-stability where a family that cannot be accommodated can block other families, and it is not hard to show that the set of strongly stable outcomes can be empty. This concept serves as the connection between quasi-stability and other stability concepts.

3.2 Stability and Weak Stability

Next we extend the traditional stability concept in the two-sided matching problem to the model with multi-dimensional constraints.

Definition 3.4 (Stability). For a feasible outcome X, a family $f \in F$ and a locality $\ell \in L$, the pair (f, ℓ) is a *blocking pair* if there is a feasible outcome X' with $(f, \ell) \in X' \setminus X$ such that $\ell \succ_f L_f(X)$ and for each $f' \in F_\ell(X) \setminus F_\ell(X'), f \succ_\ell f'$. A feasible outcome X is *stable* if it is individually rational and admits no blocking pair.[1]

In words, f and ℓ form a blocking pair if f prefers ℓ to its current assigned locality and ℓ can accommodate f by (optionally) removing families matched to ℓ with lower priority than f. When there are only unidimensional demands / capacities and each family only consumes one unit, the definition coincides with the classical stability concept.

If we impose a restriction that ℓ can accommodate f by removing exactly one family f' matched to ℓ with lower priority than f, then we derive the following weak stability concept.

Definition 3.5 (Weak Stability). For a feasible outcome X, a family $f \in F$ and a locality $\ell \in L$, the pair (f, ℓ) is a *strongly blocking pair* if there is a feasible outcome X' with $(f, \ell) \in X' \setminus X$ and $|F_\ell(X) \setminus F_\ell(X')| \leq 1$ such that $\ell \succ_f L_f(X)$ and for each $f' \in F_\ell(X) \setminus F_\ell(X'), f \succ_\ell f'$. A feasible outcome X is *weakly stable* if it is individually rational and admits no strongly blocking pair.[2]

Next we show that the set of weakly stable outcomes can be empty. We can also derive the same conclusion for strong stability and stability due to their logical relations.

Proposition 3.6. *The set of weakly stable outcomes can be empty even if preferences and priorities are strict and there are only unidimensional demands and capacities.*

Proof. Consider the following instance.[3] We adhere to the convention throughout the paper that only acceptable agents are ranked in the preference/priority profile.

[1] We define stability in the same way as Delacrétaz et al. [7], except we also consider individual rationality.
[2] Strong stability and weak stability are different from their namesakes in [17].
[3] This counterexample was also considered by McDermid and Manlove [2010] and Delacrétaz et al. [2016].

$$
\begin{array}{ll}
F = \{f_1, f_2, f_3\} & L = \{\ell_1, \ell_2\} \\
d_{f_1} = d_{f_2} = c_{\ell_2} = (1) & d_{f_3} = c_{\ell_1} = (2) \\
\ell_2 \succ_{f_1} \ell_1 & f_1 \succ_{\ell_1} f_3 \succ_{\ell_1} f_2 \\
\ell_1 \succ_{f_2} \ell_2 & f_2 \succ_{\ell_2} f_1 \succ_{\ell_2} f_3 \\
\ell_1 \succ_{f_3} \ell_2 &
\end{array}
$$

Suppose that there exists a weakly stable outcome X. If $(f_3, \ell_1) \in X$, then f_1 must be matched to ℓ_2 or else f_1 and ℓ_1 will form a blocking pair. However, this outcome will be blocked by (f_2, ℓ_2), which leads to a contradiction. Hence $(f_3, \ell_1) \notin X$. Then $X = \{(f_2, \ell_1), (f_1, \ell_2)\}$, otherwise (f_2, ℓ_1) or (f_1, ℓ_2) will form a blocking pair. However, X is blocked by (f_3, ℓ_1), which is a contradiction. □

3.3 Stability by Demand

In an attempt to overcome the negative results in the previous section, we weaken stability in an orthogonal manner. A natural idea is that a family f, which prefers to be matched with a better locality ℓ but requires more resources than a set of families matched to ℓ with lower priority than f, cannot form a blocking pair with ℓ.

Definition 3.7 (Stability by demand). For a feasible outcome X, a family $f \in F$ and a locality $\ell \in L$, the pair (f, ℓ) is a *blocking pair by demand* if there is a feasible outcome X' with $(f, \ell) \in X' \setminus X$ such that $\ell \succ_f L_f(X)$, for each $f' \in F_\ell(X) \setminus F_\ell(X')$ we have that $f \succ_\ell f'$, and $d_f \leq \sum_{f' \in F_\ell(X) \setminus F_\ell(X')} d_{f'}$. A feasible outcome X is *stable by demand* if it is individually rational, non-wasteful and admits no blocking pair by demand.

The family f and locality ℓ form a blocking pair by demand if f prefers ℓ to its current assigned locality and ℓ can accommodate f without removing any family matched to ℓ with higher priority than f and f's demand vector is smaller or equal to the sum of the demand vectors of the removed families. Thus, f cannot replace a subset of families that have lower priority if f requires more services than them. Note that we are not overriding or ignoring the actual priorities of localities. We are using a natural requirement that if a family f has higher priority and requires less resources than a set of families, then it has some justification over the set. However, it turns out that the set of stable outcomes by demand can be empty.

Proposition 3.8. *The set of stable outcomes by demand can be empty even if preferences and priorities are strict and all families and localities are acceptable to each other.*

Proof. To prove the impossibility result, it is sufficient to show that for a particular instance, every feasible, individually rational and non-wasteful outcome is blocked by some family-locality pair, since all other feasible outcomes are either individually irrational or wasteful, which cannot be stable by demand. Consider the following instance.

$$
\begin{array}{ll}
L = \{\ell_1, \ell_2\} & F = \{f_1, f_2, f_3, f_4\} \\
\ell_1 \succ_{f_1} \ell_2 & d_{f_3} = c_{\ell_1} = (2) \\
\ell_1 \succ_{f_2} \ell_2 & d_{f_1} = d_{f_2} = d_{f_4} = c_{\ell_2} = (1) \\
\ell_1 \succ_{f_3} \ell_2 & f_4 \succ_{\ell_1} f_3 \succ_{\ell_1} f_1 \succ_{\ell_1} f_2 \\
\ell_2 \succ_{f_4} \ell_1 & f_1 \succ_{\ell_2} f_4 \succ_{\ell_2} f_2 \succ_{\ell_2} f_3
\end{array}
$$

First we find all feasible, individually rational and non-wasteful outcomes. Suppose f_3 is unmatched, then we can match any two families from $\{f_1, f_2, f_4\}$ with ℓ_1 and the remaining one with ℓ_2. The corresponding outcomes are X_1, X_2, X_3. Suppose f_3 is matched, then f_3 needs to be matched with ℓ_1, since only ℓ_1 can accommodate it. Then ℓ_2 can be matched with any family from $\{f_1, f_2, f_4\}$. The corresponding outcomes are X_4, X_5, X_6.

$$
\begin{array}{ll}
X_1 = \{(f_1, \ell_1), (f_2, \ell_1), (f_4, \ell_2)\} & X_4 = \{(f_3, \ell_1), (f_4, \ell_2)\} \\
X_2 = \{(f_1, \ell_1), (f_4, \ell_1), (f_2, \ell_2)\} & X_5 = \{(f_3, \ell_1), (f_1, \ell_2)\} \\
X_3 = \{(f_2, \ell_1), (f_4, \ell_1), (f_1, \ell_2)\} & X_6 = \{(f_3, \ell_1), (f_2, \ell_2)\}
\end{array}
$$

However, for each of these outcomes, we can exhibit a blocking pair by demand: X_1 is blocked by (f_3, l_1) via X_4; X_2 is blocked by (f_4, l_2) via X_1; X_3 is blocked by (f_1, l_1) via X_2; X_4 is blocked by (f_1, l_2) via X_5; X_5 is blocked by (f_4, l_1) via X_3; X_6 is blocked by (f_1, l_2) via X_5. □

Next, we weaken stability by demand to obtain weak stability by demand. Weak stability by demand is also implied by weak stability.

Definition 3.9 (Weak stability by demand). For a feasible outcome X, a family $f \in F$, and a locality $\ell \in L$, the pair (f, ℓ) is a *strongly blocking pair by demand* if there is a feasible outcome X' with $(f, \ell) \in X' \setminus X$ and $|F_\ell(X) \setminus F_\ell(X')| \leq 1$ such that $\ell \succ_f L_f(X)$, for each $f' \in F_\ell(X) \setminus F_\ell(X')$ we have that $f \succ_\ell f'$, and $d_f \leq \sum_{f' \in F_\ell(X) \setminus F_\ell(X')} d_{f'}$. A feasible outcome X is *weakly stable by demand* if it is individually rational, non-wasteful and admits no weakly blocking pair by demand.

The distinction from stability by demand is that a strongly blocking pair by demand requires that the family f can replace a less preferred family f' at locality ℓ without using more services.

Contrary to the non-existence result for stability by demand, the weaker variant can guarantee the existence of desirable outcomes.

Proposition 3.10. *The set of weakly stable outcomes by demand is non-empty.*

To prove this proposition we will present an algorithm that always yields a weakly stable outcome by demand, that will be shown in next section.

4 HIERARCHICAL FAMILY-PROPOSING DEFERRED ACCEPTANCE ALGORITHM

In this section, we present the hierarchical family-proposing deferred acceptance (HFPDA) algorithm that is strategy-proof for families, polynomial-time and returns a matching that is weakly stable by demand. Before we proceed to the HFPDA algorithm, we first describe the family-proposing deferred acceptance (FPDA) algorithm that only applies to the case where all families have the same demand, which is formally specified as Algorithm 1.

When indifferences arise in the preference / priority profile, the stable matching will still exist [22]. In our FPDA algorithm, indifferences are allowed. When they appear, we break all ties lexicographically.

Proposition 4.1. *The HFPDA algorithm returns a weakly stable outcome by demand.*

Proof. To prove the outcome is weakly stable by demand, first we need to show it satisfies individual rationality. Since each family only proposes to acceptable localities and localities only temporarily

Input: A refugee allocation instance $\mu = (F, L, \succsim, S, d, c)$ where all families in F have the same demand vector.
Output: A stable outcome X.

For each $\ell \in L$, let $q(\ell)$ denote the maximum number of families in F that ℓ can host.
Consider the hospital-resident instance $(F, L, \succsim, q)$ where F corresponds to the set of residents, L corresponds to the set of hospitals, the size of each family is considered to be 1 and the quota of each locality ℓ is $q(\ell)$.
If members of F or L have ties in their preferences, break ties lexicographically and update $\succsim$ accordingly.
Run the classical DA (Deferred Acceptance) algorithm on the hospital-resident instance $(F, L, \succsim, q)$. Let the outcome of DA be the matching X.
return X.

Algorithm 1: FPDA Algorithm

Input: A refugee allocation instance $\mu = (F, L, \succsim, S, d, c)$
Output: A weakly stable outcome X by demand

$c' \longleftarrow c$ % capacities will be modified
Find an ordered partition $\mathcal{H} = (H_1, \ldots, H_{|\mathcal{H}|})$ of the families in F such that
- Any two families $f, f' \in F$ are in the same set if and only if $d_f = d_{f'}$
- If $d_f < d_{f'}$, then $f \in H_i$ and $f' \in H_j$ for some $i < j$.

% The order can be found as follows. Construct a partial order G over $\mathcal{H}$ in which H_i points to H_j if $d_f < d_{f'}$ for $f \in H_i$ and $f' \in H_j$. Use topological sort to order the elements in $\mathcal{H}$.
for $k = 1, 2, \ldots, |\mathcal{H}|$ **do**
    Run the FPDA algorithm on families in H_k and all localities in L while considering the current capacities c'. Let the outcome of FPDA for problem $(H_k, L, \succsim, S, d, c')$ be X_k.
    Update the corresponding capacities of each locality ℓ as follows

$$c'_\ell \longleftarrow c'_\ell - \sum_{f \in F_\ell(X_k)} d_f$$

return $X = \bigcup_{k=1}^{|\mathcal{H}|} X_k$

Algorithm 2: Hierarchical Family Proposing Deferred Acceptance (HFPDA)

accommodate acceptable families, the outcome must be individual rational.

Then we show the outcome admits no weakly blocking pair by demand. For the sake of contradiction, assume there exists a weakly blocking pair by demand $(f, \ell) \in X' \setminus X$ and another family $f' \in F_\ell(X)$ such that $f \succ_\ell f'$ and $d_f \leq d_{f'}$. Since f has a weakly smaller demand vector than f', it must propose to l no later than f'. Locality ℓ would be matched with f if ℓ had enough capacity, which contradicts the fact that f' can be matched to ℓ with a weakly larger demand vector. □

We note that HFPDA runs in polynomial time since the DA algorithm runs in polynomial time.

Proposition 4.2. *The HFPDA algorithm has running time* $O(|F|^2 \cdot |S| + |F| \cdot |L|)$.

The algorithm takes time $O(|F|^2 \cdot |S|)$ to construct the ordered permutation and $O(|F| \times |L|)$ for the part where FPDA is run on the families. The overall running time is $O(|F|^2 \cdot |S| + |F| \cdot |L|)$.

Proposition 4.3. *The HFPDA algorithm is strategy-proof with respect to the families.*

Proof. The proof is by induction on the groups of families $H_1, \ldots, H_{|\mathcal{H}|}$. We first establish that the algorithm is strategy-proof for the families in H_1. Note that the matches of families in H_1 are not changed in subsequent rounds of the for loop in HFPDA. In group H_1, if we modify the demand vectors to a unidimensional vector with one unit of demand and set the capacity of each locality to the maximum number of families that the locality can accommodate, then we have a one-to-one mapping from matching of families in H_1 to a hospital-resident problem (HR) [20] in which the families in H_1 correspond to the residents and localities correspond to hospitals with preferences / priorities being unchanged.

The families in H_1 have no incentive to misreport their preferences since the DA algorithm is strategy-proof for residents in the corresponding HR problem under strict preferences [21]. It follows that when all the ties in the preferences are broken lexicographically, and DA is run on the resultant preferences, the mechanism remains strategy-proof for the families in H_1. Suppose for contradiction that some family in H_1 can misreport its preferences and get a better locality. This implies that a corresponding resident can misreport by reporting some other strict preference and get a better hospital which contradicts the fact that resident proposing DA is strategy-proof for the residents.

Since the matches of families in H_1 are irrevocable, the same argument can be applied inductively to the subsequent groups of families. So none of the families have an incentive to misreport. □

We note here that there cannot exist a weakly stable by demand matching algorithm that is strategyproof for the localities. This follows from well-known results in school choice / hospital-resident matching that there is no stable matching algorithm that is strategyproof for the hospitals [1, 21].

5 COMPLEXITY OF STABILITY

We have discussed how to find a weakly stable outcome by demand in polynomial-time. In this section we present a complete picture of the complexity of testing whether an outcome is stable with respect to different definitions as well as finding such a stable outcome.

5.1 Deciding whether a stable matching exists

McDermid and Manlove [18] proved that, even with unidimensional demands and capacities consisting of 1's and 2's and preference lists of length at most 3, it is NP-complete to decide whether a stable matching exists (Theorem 3.7). Hence it follows that checking whether there exists a stable matching for the refugee problem is NP-complete. For unidimensional demand, refugee allocation is similar to stable matching problem with sizes [5, 6] and stable matching problem with budget constraints [10, 15]. However, the authors do not consider complexity issues concerning weak stability and stability by demand. Next, we present complexity results on these two stability concepts and for several of our computational hardness results we reduce from the following NP-complete problem.

3-Partition	
Input:	A finite set $E = \{e_1, \ldots, e_{3n}\}$ of $3n$ elements, a bound W and integer weight $w(e_j)$ for each $e_j \in E$ such that $\frac{W}{4} < w(e_j) < \frac{W}{2}$ and $w(E) = \sum_{j=1}^{3n} w(e_j) = nW$.
Question:	Can E can be partitioned into n disjoint sets $E_1, \ldots, E_n$ with weight $w(E_i) = W$ for all $i \in [n]$?

Proposition 5.1. *Checking whether a weakly stable matching exists is NP-complete if indifferences are allowed, even when there are only unidimensional demands and capacities.*

Proof. To show that deciding whether a weakly stable matching exists is in NP, we can guess an outcome X as a certificate and then check whether X is weakly stable in polynomial time as shown briefly in the next section.

To show it is NP-hard, we reduce from 3-Partition and construct a refugee allocation instance. For each element e_i, create a corresponding gadget consisting of three families and two localities such that each family / locality prefers the localities / families from the same gadget to localities / families from other gadgets. The gadget is based on the proof of Proposition 3.6 where we showed that a weakly stable matching may not exist.

The preference/priority profiles for each gadget are as follows:

$$d_{f_1} = d_{f_2} = c_{l_2} = w(e_i) \qquad d_{f_3} = c_{l_1} = 2w(e_i)$$
$$l_2 \succ_{f_1} l_1 \qquad l_1 \succ_{f_2} l_2 \qquad l_1 \succ_{f_3} \emptyset$$
$$f_2 \succ_{l_2} f_1 \qquad f_1 \succ_{l_1} f_3 \succ_{l_1} f_2$$

There are n new localities $k_1, ..., k_n$, each of capacity W. For each gadget, f_1 prefers these new localities to l_1 and the other families have the least preference for the new localities. Each new locality prefers the families of type f_1 to the families of other types and they are completely indifferent among all f_1 families. The construction can be done in polynomial time.

We can then show that there exists a stable matching if and only if the 3-Partition instance is a yes-instance. □

Next we prove a similar a result for stability by demand.

Proposition 5.2. *Checking whether a stable matching by demand exists is NP-complete if indifferences are allowed, even when there are only unidimensional demands and capacities.*

Proof. We can prove this proposition by a analogous reduction algorithm from 3-Partition, but with a different a gadget for each element as followed:

$$d_{f_1} = d_{f_2} = d_{f_4} = c_{l_2} = w(e_i) \quad d_{f_3} = c_{l_1} = 2w(e_i)$$
$$l_1 \succ_{f_1} l_2 \quad l_1 \succ_{f_2} l_2 \quad l_1 \succ_{f_3} l_2 \quad l_2 \succ_{f_4} l_1$$
$$f_4 \succ_{l_1} f_3 \succ_{l_1} f_1 \succ_{l_1} f_2 \quad f_1 \succ_{l_2} f_4 \succ_{l_2} f_2 \succ_{l_2} f_3$$

There are n new localities $k_1, ..., k_n$, each of capacity W. For each gadget, f_1 prefers these new localities to l_1 and the other families have the least preference for the new localities. Each new locality prefers the families of type f_1 to the families of other types and they are completely indifferent among all f_1 families. The gadget is based on the proof of Proposition 3.8 where we showed that a weakly stable matching may not exist.

We can then show that there exists a stable matching if and only if the 3-Partition instance is a yes-instance. □

The complexity of checking whether a weakly stable matching or stable matching by demand exists under strict preferences is still open.

5.2 Capturing Stability by Constraints

We present constraints capturing feasibility, individual rationality, and different stability concepts. The constraints not only help us obtain polynomial-time algorithm to test stability but also lead to compact integer or constraint programs that help find a stable matching whenever it exists. Capturing stability requirements via constraints is useful for several reasons including (1) Providing simple algorithms for testing stability of an outcome; (2) Allowing to use ready-made and optimized integer programming and constraint programming tools to our advantage, and (3) Ease of adding more constraints and objectives while requiring stability. The constraints also help in our mathematical understanding of stability concepts and provide alternative formulations of stability requirements.

Let $x(f, \ell)$ denote the function such that $x(f, \ell) = 1$ if f is matched to ℓ and otherwise $x(f, \ell) = 0$. Formula 1 is the constraint for feasible and individual rational outcomes.

$$\begin{aligned} &\sum_{f \in F} d_f^s x(f, \ell) \leq c_\ell^s && \forall s \in S,\ \forall \ell \in L \\ &\sum_{\ell \in L} x(f, \ell) \leq 1 && \forall f \in F \\ &x(f, \ell) \in \{0, 1\} && \forall f \in F,\ \forall \ell \in L \\ &x(f, \ell) = 0 && \text{if } \emptyset \succ_f \ell \text{ or } \emptyset \succ_\ell f \end{aligned} \tag{1}$$

Below we formulate inequalities capturing non-wastefulness and different forms of blocking pair with respect to stability concepts.

Non-wastefulness For each (f, ℓ), the following constraint is satisfied for at least one service s:

$$\sum_{\ell' \succsim_f \ell} x(f, \ell') \times c_\ell^s + \sum_{f' \in F} x(f', \ell) \times d_{f'}^s + d_f^s > c_\ell^s. \tag{2}$$

Stability For each (f, ℓ), the following constraint is satisfied for at least one service s:

$$\sum_{\ell' \succsim_f \ell} x(f, \ell') \times c_\ell^s + \sum_{f' \succ_\ell f} x(f', \ell) \times d_{f'}^s + d_f^s > c_\ell^s. \tag{3}$$

If f is not matched with a weakly better locality than ℓ, then f cannot coexist with all families that are matched to ℓ with higher priority than f.

Weak stability For each (f, ℓ) and any $f'' \in F$ such that $f \succ_l f''$, the following constraint is satisfied for at least one service s:

$$\sum_{\ell' \succsim_f \ell} x(f, \ell') \times c_\ell^s + \sum_{f' \in F} x(f', \ell) \times d_{f'}^s - x(f'', \ell) \times d_{f''}^s + d_f^s > c_\ell^s. \tag{4}$$

If f is not matched with a weakly better locality than ℓ, then ℓ cannot accommodate f by removing any matched family f'' that has lower priority than f.

Stability by demand requires that both non-wastefulness and the following constraints are satisfied: For each (f, ℓ), the following constraint is satisfied for at least one service s:

$$\sum_{\ell' \succsim_f \ell} x(f, \ell') \times c^s_{\ell} + d^s_f - \sum_{f \succ_{\ell} f'} x(f', \ell) \times d^s_{f'} > 0. \tag{5}$$

If f is not matched with a weakly better locality than ℓ, then the demand vector of f is not weakly smaller than the sum of demand vectors of all families that are matched to ℓ with lower priority than f.

Weak stability by demand requires that both non-wastefulness and the following constraints are satisfied: For each (f, ℓ) and each family f' such that $f \succ_{\ell} f'$, the following constraint is satisfied for at east one service s:

$$\sum_{\ell' \succsim_f \ell} x(f, \ell') \times c^s_{\ell} + d^s_f - d^s_{f'} > 0. \tag{6}$$

If f is not matched with a weakly better locality than ℓ, then the demand vector of f is not weakly smaller than the demand vector of any family f' that is matched to ℓ with lower priority than f.

One can establish that the linear constraints presented above capture the corresponding concepts. We present the argument for stability. The other arguments work similarly. We need to prove that an outcome X is stable if and only if it can satisfy formula 3. First, assume X is stable, then there is no blocking pair. In other words, for any f and any ℓ, either f is matched with a weakly better locality than ℓ or ℓ cannot accommodate f with $F^{\succ f}_{\ell}(X)$. For both cases, formula 3 holds. Second, assume formula 3 holds. Then there will be two cases, if $x(f, \ell') = 1$, then f is not interested in forming a blocking pair with ℓ. Otherwise, $x(f, \ell') = 0$ and f is matched with a locality worse than ℓ. However, there exists at least one service s such that ℓ cannot accommodate f with $F^{\succ f}_{\ell}(X)$.

If there is exactly one service, the stability constraints can be achieved by a polynomial number of IP constraints. If there are more than one service, one can use a constraint program that requires at least one constraint from each family of constraints to be satisfied. If we want to write an integer program for any number of services, then each disjunction of constraints can be modeled by a logarithmic number of binary variables [25].

6 MASTER LISTS

In addition to the individual priority relations of localities, there may be some global priority ordering that may also be needed to be considered while making an allocation. In this section, we discuss these global priorities over families that have been referred to as the master list in the literature [12]. The master list can be based on any given global criterion such as the position in the queue, educational level of the family, or critical need for health care.

Master lists have been employed in the mechanism design for matching problems with couples [3] and distributional constraints [8, 9, 14, 24]. Take hospital-resident problem for example, the idea is to impose a master list over all residents for all hospitals and both original priority ordering of each hospital and master list need to be considered in defining stability. Similar master lists have been used in practice, for example in the Scottish entry-labor market for medical school graduates [11].

Input: A refugee allocation instance $\mu = (F, L, \succsim, S, d, c)$ and master list $\mathcal{ML}$ over F such that $f \sim_{\mathcal{ML}} f' \Rightarrow d_f = d_{f'}$.
Output: A stable by $\mathcal{ML}$ outcome X.

$c' \longleftarrow c$ % capacities will be modified
Use $\mathcal{ML}$ to divide the families into groups $H_1, \ldots, H_{|\mathcal{H}|}$ in the order of $\mathcal{ML}$ where each group H_i forms an indifference equivalence class with respect to $\mathcal{ML}$. % In each group, the families have the same demand vector.
for $k = 1, 2, \ldots, |\mathcal{H}|$ **do**
    Run the FPDA algorithm on families in H_k and all localities in L while considering the current capacities c'. Let the outcome of FPDA for problem $(H_k, L, \succsim, S, d, c')$ be X_k.
    Update the corresponding capacities of each locality ℓ as follows
$$c'_{\ell} \longleftarrow c'_{\ell} - \sum_{f \in F_{\ell}(X_k)} d_f$$
return $X = \bigcup_{k=1}^{|\mathcal{H}|} X_k$

Algorithm 3: Hierarchical Family Proposing Deferred Acceptance (HFPDA) for Master Lists

In refugee allocation, one possible reason for master list could be respective the amount of time the families have been in the matching market. Let $f \succsim_{\mathcal{ML}} f'$ denote that f has higher $\mathcal{ML}$ priority than f' or they have the same $\mathcal{ML}$ priority. The difference from previous work is that our new concepts take multi-dimensional constraints into account.

Definition 6.1 (Stability by $\mathcal{ML}$). Given a feasible matching X and a master list $\mathcal{ML}$, a pair $(f, \ell) \in X' \setminus X$ is called a *blocking pair by $\mathcal{ML}$* if $\ell \succ_f L_f(X)$ and $\forall f' \in F_{\ell}(X) \setminus F_{\ell}(X'), f \succ_{\ell} f'$ and $f \succsim_{\mathcal{ML}} f'$. A feasible outcome X is *stable by $\mathcal{ML}$* if it is individually rational, non-wasteful and admits no blocking pair by $\mathcal{ML}$.

The pair (f, ℓ) can form a blocking pair by $\mathcal{ML}$ if ℓ can remove a subset of matched families with lower preference and lower master list priority than f to accommodate it. Similarly, we can define weak stability as follows.

Definition 6.2 (Weak stability by $\mathcal{ML}$). Given a feasible matching X and a master list $\mathcal{ML}$, a pair $(f, \ell) \in X' \setminus X$ is called a *weak blocking pair by $\mathcal{ML}$* if $\ell \succ_f L_f(X)$ and $\forall f' \in F_{\ell}(X) \setminus F_{\ell}(X'), f \succ_{\ell} f'$ and $f \succsim_{\mathcal{ML}} f'$ with $|F_{\ell}(X) \setminus F_{\ell}(X')| \leq 1$. A feasible outcome X is *weakly stable by $\mathcal{ML}$* if it is individually rational, non-wasteful and admits no weak blocking pair by $\mathcal{ML}$.

If all families have the same priority in the master list $\mathcal{ML}$, then (weak) stability by $\mathcal{ML}$ is the same as (weak) stability and a (weakly) stable by $\mathcal{ML}$ outcome is not guaranteed to exist. Therefore, we make the following assumption on the master lists throughout the paper: every two families in the same $\mathcal{ML}$-equivalence class have the same demand vectors ($f \sim_{\mathcal{ML}} f' \Rightarrow d_f = d_{f'}$). A special case is when the master list gives a strict priority over the families.

Note that HFPDA is an algorithm that is designed to find a weak stable by demand outcome. We can view HFPDA more generally if we use some exogenous master list ordering over the families to partition them into equivalence classes. In this case, a modification (Algorithm 3) of HFPDA returns a stable by $\mathcal{ML}$ matching and the algorithm is strategyproof. The arguments are almost identical to those for weak stability by demand. Hence HFPDA can be viewed more generally as finding an ordered partitioning families with the earlier sets having priority over the later ones because they have a higher rank according to $\mathcal{ML}$.

7 COMPLEXITY OF PARETO OPTIMALITY

Under strict preferences, computing a Pareto optimal allocation is easy via serial dictatorship: first set a strict priority ordering over the families and then let each family take a place in the best possible locality that can accommodate it. On the other hand, computing a Pareto optimal allocation is NP-hard if there are indifferences in the preferences. Since checking whether there exists a matching that accommodates all the families is NP-hard, it follows that finding a Pareto optimal matching is NP-hard even if there is complete indifference in the preferences and priorities.

Proposition 7.1. *Computing a Pareto optimal outcome is NP-hard even if each family is indifferent among localities, each family is acceptable, and there are unidimensional demands and capacities.*

Moreover, finding Pareto improvements over an existing allocation seems challenging. Delacrétaz et al. [7] presumed that finding a Pareto improvement was computationally intractable and they described an exponential time algorithm for the problem. We formally prove intractability by reduction from 3-Partition.

Proposition 7.2. *Testing weak Pareto optimality is strongly coNP-complete even if families have strict preferences over localities, each family is acceptable, and there are unidimensional demands and capacities.*

Proof. To show that the complement of testing weak Pareto optimality (TWPO) is in NP, for a given outcome X, we can guess another outcome X' as a certificate and check whether all families strictly prefer X' to X in polynomial time.

We now prove that 3-Partition reduces to TWPO. The reduction algorithm begins with an instance of 3-Partition. Assume all families have strict preferences over localities and each family is acceptable and there is only one type of service. We construct a refugee allocation instance as follows. Let $F = \{f^*, f_1, \dots, f_{3n}\}$ be the set of families and $L = \{\ell^*, \ell', l_1, \dots, l_n\}$ be the set of localities. The family f^* requires nW units of service while each f_i requires $w(e_i)$ units. The capacity of both ℓ^* and ℓ' is nW and it is W for each other locality l_i. The family f^* has ℓ^* as the second most preferred locality and ℓ' as the most preferred locality. Each f_i has ℓ' as the second least preferred locality and ℓ^* as the least preferred locality. The allocation X is the one in which f^* is matched to ℓ^* and all the other families are matched to ℓ'. The construction can be done in polynomial time.

We can then show that E can be partitioned into n disjoint sets $E_1, \dots, E_n$ and weight $w(E_i) = W$ for all $i \in [n]$ if and only if X is not weakly Pareto optimal. ◻

Proposition 7.3. *Testing Pareto optimality is strongly coNP-complete when each family is indifferent among all localities, each family is acceptable and there are unidimensional demands and capacities.*

Proof. To show that testing Pareto optimality is in coNP, for a given outcome X, we can guess another outcome X' and check whether all families weakly prefer X' to X and at least one family strictly prefers $X'$to X in polynomial time.

To prove NP-hardness, we reduce from 3-Partition and construct a refugee allocation instance as follows.

$$F = \{f^*, f_1, \dots, f_{3n}\} \quad L = \{\ell^*, \ell', l_1, \dots, l_n\}$$
$$d_{f^*} = c_{\ell^*} = c_{\ell'} = (nW)\ d_{f_i} = w(e_i)\ c_{l_j} = (W)$$
$$\ell' \succ_{f^*} \ell^* \succ_{f^*} \ell_i \quad \ell_j \sim_{f_i} \ell_{j'} \sim_{f_i} \ell' \succ_{f_i} \ell^*$$
$$X = \{(f^*, \ell^*), (f_1, \ell'), ..., (f_{3n}, \ell')\}$$

Note that each f_i has ℓ' as the second least preferred locality, ℓ^* as the least preferred locality and are indifferent among all other localities ℓ_j We can then show that the 3-Partition instance is a yes-instance if and only if X is not Pareto optimal. ◻

We note that our central results concerning Pareto optimality are computational hardness results. As a corollary we obtain the same results if we also take into account the preferences of the localities by assuming that the localities have complete indifference among the families.

8 DISCUSSION

Delacrétaz et al. [7] presented an Integer Programming (IP) formulation to accommodate the maximum number of refugees and tailor-made algorithms to find stable / quasi-stable outcomes. The largest number of refugees can be accommodated by applying the following objective to the integer program in feasibility constraints (1): max $\sum_{f \in F} \sum_{\ell \in L} n_f \times x(f, \ell)$ where n_f denotes the number of family members of f.

We can combine two orthogonal approaches of accommodating maximum number of refugees and finding stable outcomes as follows. We make a case that it is desirable to capture stability constraints and incorporate them into an integer or constraint program. By doing so, one can maximize other objectives such as accommodating a maximum number of refugees while maintaining some form of stability, especially if the stability constraint does not affect or significantly affect the number of refugees accommodated. If a stable matching exists and does not lead to a significant enough decrease in the number of people hosted, then we can select that matching. Otherwise, we can gradually replace the constraints for stronger stability notion by constraints for a weaker stability notion until we are satisfied with the number of people who are matched. Such an approach also makes it possible to additionally impose other feasibility constraints not considered in the paper.

ACKNOWLEDGMENTS

Aziz is supported by a Julius Career Award. Serge Gaspers is the recipient of an Australian Research Council (ARC) Future Fellowship (project number FT140100048) and he also acknowledges support under the ARC's Discovery Projects funding scheme (DP150101134).

REFERENCES

[1] A. Abdulkadiroğlu and T. Sönmez. 2003. School Choice: A Mechanism Design Approach. *American Economic Review* 93, 3 (2003), 729–747.

[2] T. Andersson and L. Ehlers. 2016. *Assigning refugees to landlords in Sweden: Stable maximum matchings.* Technical Report 2016-08. Université de Montréal Papyrus Institutional Repository.

[3] P. Biró, R. W. Irving, and I. Schlotter. 2011. Stable matching with couples: an empirical study. *Journal of Experimental Algorithms (JEA)* 16 (2011), 1–2.

[4] P. Biró, D. F. Manlove, and I. McBride. 2014. The hospitals/residents problem with couples: Complexity and integer programming models. In *International Symposium on Experimental Algorithms*. Springer, 10–21.

[5] P. Biró and E. McDermid. 2014. Matching with sizes (or scheduling with processing set restrictions). *Discrete Applied Mathematics* 164 (2014), 61–67.

[6] B. Dean, M. Goemans, and N. Immorlica. 2006. The unsplittable stable marriage problem. In *Fourth IFIP International Conference on Theoretical Computer Science-TCS 2006*. Springer, 65–75.

[7] D. Delacrétaz, S. D. Kominers, and A. Teytelboym. 2016. Refugee Resettlement. (2016).

[8] M. Goto, N. Hashimoto, A. Iwasaki, Y. Kawasaki, S. Ueda, Y. Yasuda, and M. Yokoo. 2014. Strategy-proof matching with regional minimum quotas. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. International Foundation for Autonomous Agents and Multiagent Systems, 1225–1232.

[9] M. Goto, F. Kojima, R. Kurata, A. Tamura, and M. Yokoo. 2017. Designing matching mechanisms under general distributional constraints. *American Economic Journal* 9, 2 (2017), 226–262.

[10] N. Hamada, A. Ismaili, T. Suzuki, and M. Yokoo. 2017. Weighted Matching Markets with Budget Constraints. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 317–325.

[11] R. Irving. 2011. Matching practices for entry-labor markets – Scotland. *MiP Country Profile 3* (2011), 11–14.

[12] R. W. Irving, D. F. Manlove, and S. Scott. 2008. The stable marriage problem with master preference lists. *Discrete Applied Mathematics* 156, 15 (2008), 2959–2977.

[13] W. Jones and A. Teytelboym. 2016. Choices, preferences and priorities in a matching system for refugees. *Forced Migration Review* 51 (2016), 81–82.

[14] Y. Kamada and F. Kojima. 2015. Efficient matching under distributional constraints: Theory and applications. *The American Economic Review* 105, 1 (2015), 67–99.

[15] Y. Kawase and A. Iwasaki. 2017. Near-Feasible Stable Matchings with Budget Constraints. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 242–248. https://doi.org/10.24963/ijcai.2017/35

[16] B. Klaus, D. F. Manlove, and F. Rossi. 2016. Matching under Preferences. In *Handbook of Computational Social Choice*, F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia (Eds.). Cambridge University Press, Chapter 14.

[17] D. F. Manlove. 2013. *Algorithmics of Matching Under Preferences*. World Scientific.

[18] E. J. McDermid and D. F. Manlove. 2010. Keeping partners together: algorithmic results for the hospitals/residents problem with couples. *Journal of Combinatorial Optimization* 19, 3 (2010), 279–303.

[19] J. F-H. Moraga and H. Rapoport. 2014. Tradable immigration quotas. *Journal of Public Economics* 115, C (2014), 94–108.

[20] A. E. Roth. 1984. The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of Political Economy* 92, 6 (1984), 991–1016.

[21] A. E. Roth. 1985. The college admissions problem is not equivalent to the marriage problem. *Journal of Economic Theory* 36, 2 (1985), 277–288.

[22] A. E. Roth. 2008. Deferred acceptance algorithms: history, theory, practice, and open questions. *International Journal of Game Theory* 36 (2008), 537—569.

[23] A. E. Roth, T. Sönmez, and M. U. Ünver. 2007. Efficient Kidney Exchange: Coincidence of Wants in Markets with Compatibility-Based Preferences. *American Economic Review* 97, 3 (2007), 828–851.

[24] S. Ueda, D. Fragiadakis, A. Iwasaki, P. Troyan, and M. Yokoo. 2012. Strategy-proof mechanisms for two-sided matching with minimum and maximum quotas. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. International Foundation for Autonomous Agents and Multiagent Systems, 1327–1328.

[25] J. P. Vielma and G. L. Nemhauser. 2008. Modeling Disjunctive Constraints with a Logarithmic Number of Binary Variables and Constraints. In *Proceedings of the 13th International Conference on Integer Programming and Combinatorial Optimization (IPCO'08)*. Springer-Verlag, Berlin, Heidelberg, 199–213.