HOUSING ALLOCATION: EXISTING TENANTS AND MULTIPLE-OCCUPANCY

MICHAEL WALLACE

ABSTRACT. Housing allocation problems deal with assigning indivisible objects (houses) to agents who have preferences over these objects. We examine the housing allocation problem with existing tenants, then with both new and existing tenants. In both cases, we present and evaluate an algorithm for assigning houses. We then consider a version of the housing allocation problem where agents are grouped into groups of size k and houses can be occupied by k agents. We present an algorithm to assign houses to agents in this case, and evaluate this algorithm.

1. INTRODUCTION

Housing allocation problems deal with assigning indivisible objects (houses) to agents who have preferences over these objects. In general, housing allocation problems consist of

(1) a set of agents A,
(2) a set of indivisible objects (houses) H,
(3) a preference profile $\succ = (\succ_a)_{a \in A}$, that is, a list of preference relations of agents over houses.

For simplicity, we restrict our attention to strict preference profiles where each agent defines a strict total order over houses. An important characteristic of housing allocation problems, is that it is assumed that housing comes with no externalities. That is, there is no medium of exchange for agents (for example, money) outside of houses.

These types of problems have a number of applications. In this paper, we contextualize variants of the housing allocation problem through the scope of college dormitory housing. Consider a college dormitory that must assign rooms to students. We begin by considering only single-occupancy rooms (at most one student per room). In Section 3 we consider the constrained case where all students to be allocated a room begin the allocation process with a room. In Section 4 we consider the case where there may be both new and existing tenants entering the allocation process. For both of these cases, we cite and evaluate algorithms based upon Gale's Top Trading Cycles algorithm [5].

Date: May 12, 2016.

Key words and phrases. matching, housing allocation, housing markets, Gale's Top Trading Cycles, existing tenants, multiple-occupancy.

In Section 5 we consider the case where rooms have a fixed occupancy k, and students enter the allocation process in groups of k. We allow for both new and existing students, as well as students who previously lived with different groups. We present and evaluate a novel algorithm for allocating rooms to students, that also resembles Gale's Top Trading Cycles algorithm.

2. Definitions

In this section we present some definitions that will allow us to evaluate the outcomes of housing allocation problems, and the algorithms that produce them.

2.1. Matchings.

The outcome of the housing allocation problem $(A, H, \succ)$ is a matching $\mu : A \mapsto H$. The interpretation of a matching μ is that each agent a is allocated the house $\mu(a)$. For a fixed housing problem, we will use $\mathcal{M}$ to denote the set of all possible matchings.

In addition having preference relations over houses, agents can also be thought to have preference relations over matchings. Suppose μ, ν are matchings in $\mathcal{M}$, then

$$\begin{aligned} \mu \succ_a \nu &\quad\longleftrightarrow\quad \mu(a) \succ_a \nu(a), \\ \mu \succeq_a \nu &\quad\longleftrightarrow\quad \nu \not\succ_a \mu \\ \mu \sim_a \nu &\quad\longleftrightarrow\quad \mu(a) = \nu(a). \end{aligned}$$

This preference relation defines a weak total order on $\mathcal{M}$.

It is natural to ask whether some matchings are more desirable than others. It seems obvious that this is the case. Ideally, we could always produce a matching such that

(1) there is no other matching that makes some agents strictly happier, and all other agents no less happy,
(2) all agents weakly prefer their newly allocated house to their initial house.

We provide Definitions 1, 2 and 3 to formalize these notions.

Definition 1 (Pareto domination)**.** Suppose μ, ν are matchings. Then μ *Pareto dominates* ν if and only if

(1) $\mu \succeq_a \nu$ for all $a \in A$,
(2) $\mu \succ_a \nu$ for some $a \in A$.

Definition 2 (Pareto efficiency)**.** Suppose μ is a matching. Then μ is *Pareto efficient* if and only if it is not Pareto dominated by any matching $\nu \in \mathcal{M}$.

Definition 3 (Individually rational)**.** Suppose μ is a matching resulting from the housing problem $(A, H, \succ, \mu_0)$. Then μ is *individually rational* if $\mu(a) \succeq_a \mu_0(a)$ for all $a \in A$.

2.2. **Matching Mechanisms.** Given a housing allocation problem, agents may announce any strict preference relation over houses. That is, we do not assume that agents are truthful when announcing their preferences to some mechanism that allocates houses. Fix some housing allocation problem. We use $\mathcal{P}$ to denote the set of all preference profiles of all agents over houses. Like above, we use $\mathcal{M}$ to denote the set of all matchings of agents to houses.

A matching mechanism is a procedure for determining a matching given a housing allocation problem. Formally, a mechanism is a function

$$\varphi : \mathcal{P} \mapsto \mathcal{M}.$$

A mechanism is Pareto efficient if it always produces a matching that is Pareto efficient on the announced preference profile. Similarly, a mechanism is individually rational if it always selects a matching that is individually rational on the announced preference profile.

Another desirable property of a matching mechanism is that it induces agents to be truthful in reporting their preferences. We call such a mechanism *strategy proof*, and formalize this notion in Definition 4.

Definition 4 (Strategy proof)**.** Suppose φ is a matching mechanism that induces agents to announce the preference profile $\rho \in \mathcal{P}$. Then φ is *strategy proof* if and only if every agent $a \in A$ has weakly prefers their allocation (under $\succ$) when they choose ρ_a over their allocation when they choose some other preference relation, regardless of the preference relations of all other agents in A.

3. Housing Markets

We now turn our attention to housing allocation problems where houses are single-occupancy, and there is an initial endowment of houses to agents. These problems are known as housing market problems. In the context of assigning college students to dormitory rooms, this is the situation of reorganizing the existing students of a dormitory (with single-occupancy rooms). Formally, a housing market problem consists of a tuple $(A, H, \succ, \mu_0)$, where A is a set of agents, H is a set of (indivisible) houses, $\succ$ is a list of preferences over houses, and μ_0 is an initial endowment. Specifically, each agent $a \in A$ has a strict total order on H, where $h_i \succ_a h_j$ means that agent a strictly prefers h_i over h_j. Here we assume that $|A| = |H|$. The initial endowment $\mu_0 : A \mapsto H$ is a bijection from agents to houses such that each agent a is endowed with house $\mu_0(a)$ at the beginning of the allocation problem.

The outcome of a housing allocation problem is a matching $\mu : A \mapsto H$, where μ defines a bijection from A to H. The interpretation of a matching μ is that each agent a is assigned to the house $\mu(a)$. We define $\mathcal{M}$ be the set of all possible matchings.

We now present an abstract object known as the "core" of a housing market.

Definition 5 (Housing market core)**.** The *core* of a housing market problem $(A, H, \succ, \mu_0)$ is a set of matchings $\mathcal{C}$. Some matching $\mu \in \mathcal{M}$ is in $\mathcal{C}$ if and only if there in not a "coalition" $A' \subseteq A$ and some other matching $\nu \in \mathcal{M}$ such that

(1) $\nu(a) \in \{\mu_0(b) \mid b \in A'\} \quad \forall\, a \in A'$
(2) $\nu(a) \succeq_a \mu(a) \quad \forall\, a \in A'$
(3) $\exists\, a \in A'$ such that $\nu(a) \succ_a \mu(a)$

3.1. **Gale's Top Trading Cycles (TTC) Algorithm.** Gale's Top Trading Cycles (TTC) algorithm is one example of a matching mechanism, and works as follows [5].

Step 1: Each agent points to the owner of their most preferred house. If a cycle of agents exists, then match all agents in the cycle with the house of the agent he points to. Remove the matched agents and houses from the problem. If any agents or houses remain unmatched, continue to the next step.

Step t**:** Each agent points to the owner of their most preferred remaining house. If a cycle of agents exists, then match all agents in the cycle with the house of the agent he points to. Remove the matched agents and houses from the problem. If any agents or houses remain unmatched, continue to the next step.

Example. Suppose $A = \{a_1, a_2, a_3\}$ and $H = \{h_1, h_2, h_3\}$, with $\mu_0(a_i) = h_i$ for $i \in \{1, 2, 3\}$, and the agents have the following preferences over houses:

$$a_1 : h_1 \succ_{a_1} h_2 \succ_{a_1} h_3$$
$$a_2 : h_3 \succ_{a_2} h_1 \succ_{a_2} h_2$$
$$a_3 : h_1 \succ_{a_3} h_2 \succ_{a_3} h_3$$

The progression of the algorithm is shown graphically in Figure 1.

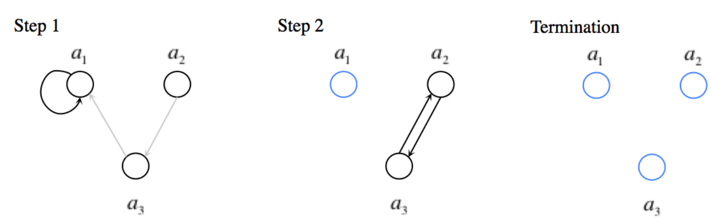

FIGURE 1. Gale's Top Trading Cycles Algorithm Progression

The algorithm terminates with the matching μ such that

$$\mu(a_1) = h_1, \quad \mu(a_2) = h_3, \quad \mu(a_3) = h_2.$$

Proposition 1. Gale's Top Trading Cycles algorithm terminates with a matching.

Proof. At each iteration, there are a finite number of agents. Hence, there exists a cycle of agents. Thus, at each step, the number of remaining houses strictly decreases, so the algorithm terminates.

A house is only assigned to one agent before it is removed from the list of unassigned houses. Thus, all houses are assigned to distinct agents, and we get a bijection $\mu : A \mapsto H$ and hence the algorithm finds a matching. □

Theorem 1 (Theorem 2 in [4]). The outcome of Gale's TTC algorithm is the unique matching in the core of each housing market.

Theorem 2 (Theorem 1 in [3]). A mechanism that provides the matching in the core is the only mechanism that is Pareto efficient, individually rational, and strategy-proof.

Together, Theorems 1 and 2 show that any mechanism that is Pareto efficient, individually rational, and strategy proof provides the outcome of Gale's TTC algorithm. Hence, Gale's TTC algorithm provides a method to realize a desirable (as discussed above) allocation of houses in this contained case of the housing allocation problem.

4. Housing Allocation with Existing Tenants

We now generalize the housing allocation problem to include both new and existing tenants. Formally, a *housing allocation problem with existing tenants* consists of a tuple $(I_E, I_N, H_O, H_V, \succ, \mu_0)$ where

(1) A_E is the set of existing agents (agents who begin with a house),
(2) A_N is the set of new agents (agents who begin without a house),
(3) H_O is the set of occupied houses, with $|H_O| = |I_E|$,
(4) H_V is the set of vacant houses,
(5) $\succ = (\succ_a)_{a \in A_E \cup A_N}$ is a preference profile: a list of strict preference relations of agents over all houses,
(6) $\mu_0 : A_E \mapsto H_O$ is a bijection from existing agents to occupied houses, which defines the initial house allocation for each existing tenant.

For simplicity, we will henceforth use

$$A = A_E \cup A_N$$
$$H = H_O \cup H_V \cup \{h_0\},$$

where h_0 denotes the "null house". In context, being assigned the null house is equivalent to not being allocated any real house. For simplicity, we assume that h_0 is the least preferred house for each agent. The outcome of such a problem is a matching $\mu : A \mapsto H$ such that μ is a total function such that there is no $h \in H_O \cup H_V$ with $|\mu^{-1}(h)| > 1$. In other words, μ satisfies

(1) every agent in A is assigned exactly one house,
(2) only h_0 may be assigned more than one house.

The definitions of a matching being Pareto efficient and individually rational extend naturally from Definitions 2 and 3.

Once again, agents may report any preference relation they desire. Let $\mathcal{P}$ denote the set of all preference profiles for all agents. A mechanism is some function that for a fixed housing allocation problem, takes a reported preference profile for all agents and produces a matching. Formally, a mechanism is a mapping $\varphi : \mathcal{P} \mapsto \mathcal{M}$ where $\mathcal{M}$ is the set of all possible matchings for the housing allocation problem.

Similar to before, a mechanism is Pareto efficient and individually rational if it always results in Pareto efficient and individually rational matchings, respectively. A mechanism is strategy proof if it induces agents to be truthful in reporting their preferences over houses (Definition 4).

4.1. **Agent Priority.** In many situations, a natural strict priority exists over the agents. For example, in the context of college dormitory room assignment, priority may be a function of students' seniority and contribution to the dormitory. We formalize agent priority as some bijection $f : \{1, 2, \ldots, |A|\} \mapsto A$ that assigns a ranking to each agents, where agent $f(1)$ has the highest priority, and so on.

We can talk about mechanisms that produce matchings for some fixed priority, and use φ_f to denote the mechanism under priority f.

4.2. **Top Trading Cycles Mechanism.** Abdulkadiroğlu and Sönmez [1] introduce a mechanism for assigning matchings given a housing allocation problem with existing tenants $(A_E, A_N, H_O, H_V, \succ, \mu_0)$, an announced preference profile $\rho = (\rho_a)_{a \in A}$, and a fixed agent priority f. We will refer to this mechanism as ψ_f and an algorithm to find the matching given by ψ_f is given below. Note this this mechanism and algorithm resembles Gales TTC algorithm, presented in section 3.1.

Step 1: Each agent $a \in A$ points to their favorite house under their announced preference relation ρ_a. Each house $h \in H_O$ points to $\mu_0^{-1}(h)$, that is, each house that is occupied points to its occupant. Each available (vacant) house points to agent with the highest priority, that is, agent $f(1)$. If a cycle (of alternating agents and houses) exists, then assign each agent the house that they point to. Remove all assigned houses and the agents they are assigned to for the purpose of future steps. If there are remaining agents and houses, then continue to the next step.

Step t: Each agent $a \in A$ points to their favorite remaining house under their announced preference relation ρ_a. Each remaining occupied house points to its occupant. Each remaining vacant house points to the remaining agent with the highest priority. If a cycle (of alternating agents and houses) exists, then assign each agent the house that they point to. Remove all assigned houses and the agents they are assigned to for the purpose of future steps. If there are remaining agents and houses, then continue to the next step.

Finally: Assign the null house to any remaining agents.

Note that if $A_E = H_V = \emptyset$, then this algorithm is equivalent to Gale's TTC. We give an example with new and existing tenants below.

Example. Suppose $A_E = \{a_1, a_2\}$, $A_N = \{a_3, a_4, a_5\}$, $H_O = \{h_1, h_2\}$, $H_V = \{h_3, h_4\}$, and $\mu_0(a_i) = h_i$ for $i \in \{1, 2\}$. Suppose that agents announce the preference profile ρ over houses as follows

$$a_1 : h_1\ h_2\ h_3\ h_4$$
$$a_2 : h_1\ h_3\ h_2\ h_4$$
$$a_3 : h_1\ h_2\ h_3\ h_4$$
$$a_4 : h_1\ h_3\ h_2\ h_4$$
$$a_5 : h_4\ h_1\ h_2\ h_3$$

and suppose that f defines the following priority over agents

$$f : a_3\ a_1\ a_2\ a_4\ a_5$$

where a_3 has the highest priority and a_5 has the lowest priority. The progression of the algorithm is shown graphically in Figure 2.

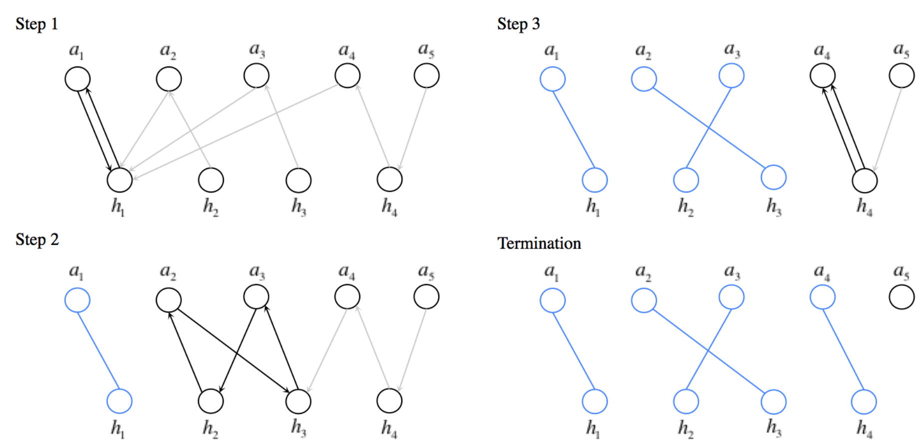

FIGURE 2. Top Trading Cycles With Existing Tenants Example

The algorithm terminates with the matching μ such that

$$\mu(a_1) = h_1, \quad \mu(a_2) = h_3, \quad \mu(a_3) = h_2, \quad \mu(a_4) = h_4, \quad \mu(a_5) = h_0.$$

Proposition 2. This top trading cycles mechanism always terminates with a matching.

Proof. At each iteration, there are a finite number of agents and houses. Hence, there exists a cycle. Thus, at each step, the number of remaining agents and houses strictly decreases, so the algorithm terminates. Any house may only be in one cycle per step, so any house that is not the null house is assigned to at most one agent, and the null house is assigned to all agents not assigned a house in $H_O \cup H_V$. Thus, the result of this algorithm is a matching. □

For any agent priority f, the corresponding top trading cycles mechanism ψ_f has some very desirable properties. In particular, ψ_f is

(1) Pareto efficient [1, Proposition 1],
(2) individually rational [1, Proposition 2], and
(3) strategy proof [1, Theorem 1].

This mechanism also has the property that it respects seniority [1, Theorem 2]. A mechanism φ_f that respects seniority meets the following criteria:

(1) as far as agent $f(1)$ is concerned, φ_f assigns them a house that is weakly preferred to the house assigns by any other mechanism that is Pareto efficient, individually rational and strategy proof,
(2) out of all mechanisms that perform equally well for agent $f(1)$, φ_f assigns $f(2)$ a house that is weakly preferred to the house assigns by any other mechanism that is Pareto efficient, individually rational and strategy proof,
(3) and so on, for all agents $f(3), f(4), \ldots$.

Hence, the top trading cycles mechanism for housing allocation problems with existing tenants meets the criteria that we outlined in Section 2.

5. Housing Allocation with Multiple-Occupancy Houses

In many practical settings, houses can be occupied by more than one agent simultaneously. Consider the case where all houses have fixed occupancy k, and agents are exogenously grouped together, in groups of size k. Groups may comprise of new and existing students, including existing students who previously lived in separate houses. Formally, we define a *housing allocation problem with multiple-occupancy houses* as

$$(A_E, A_N, G, H_O, H_V, \succ, \mu_0, r)$$

where

(1) A_E is the set of existing agents (agents who begin with a house),
(2) A_N is the set of new agents (agents who begin without a house),
(3) G is a list of groups of agents,
(4) H_O is the set of occupied houses (houses with at least one agent in A_E living there previously)
(5) H_V is the set of vacant houses,
(6) $\succ = (\succ_a)_{a \in A_E \cup A_N}$ is a preference profile: a list of strict preference relations of agents over all houses,
(7) $\mu_0 : A_E \mapsto H_O$ is a bijection from existing agents to occupied houses, which defines the initial house allocation for each existing tenant,
(8) $r : A_E \cup A_N \mapsto R$ is a function that maps agents to their corresponding housing group.

Furthermore, we place the following restrictions on these inputs

(1) all groups in G must have size k,
(2) all agents in the same group must have the same preference relation over houses.

For some group $g \in G$, we may use $\succ_g$ to refer to the preference relation of any agent in group g. As before, we assume some natural priority f on the agents.

Below we present a novel algorithm that provides a mechanism similar to the top trading cycles mechanism shown above. The algorithm proceeds as follows.

Step t: Each agents $a \in A$ points to group $r(a)$. Each group points to their favorite remaining house. For every remaining occupied house $h \in H_O$, if $\mu^{-1}(h)$ contains remaining agents, h points to the remaining agent with the highest priority; otherwise h points to the remaining agent in A with the highest priority. Each vacant house points to the remaining agent in A with the highest priority. If a cycle exists, then for every group g in the cycle that points to h, assign all agents in g the house h. Remove all assigned agents, groups and houses. If there are remaining agents and houses, then continue to the next step.

Finally: Assign the null house to any remaining agents.

Proposition 3. This mechanism terminates with a matching.

Proof. The proof is almost identical to the proof of Proposition 2 and is omitted here. □

Theorem 3. This mechanism is Pareto efficient.

Proof. Consider the algorithm given. Any agent who leaves at step 1 is assigned their top choice and cannot be strictly happier. Any agent who leaves at step 2 is assigned their top choice among the remaining houses, and cannot be made strictly happier without making an agent assigned in step 1 strictly less happy. Proceeding in a similar way, no agent can be made strictly happier without making another agent who was assigned at an earlier step strictly less happy. Therefore, the mechanism always produces a Pareto efficient matching. □

Theorem 4. No mechanism exists for this problem that is individually rational.

Proof. Consider the following counterexample.

$$\begin{aligned} A_E &= \{a_1, a_3\} \\ A_N &= \{a_2, a_4\} \\ G &= \{g_1, g_2\} \\ H_O &= \{h_1\} \\ H_V &= \emptyset \\ \mu_0(a_1) &= h_1 \\ \mu_0(a_3) &= h_1 \\ r(a) &= \begin{cases} g_1 & \text{if } a = a_1 \\ g_1 & \text{if } a = a_2 \\ g_2 & \text{if } a = a_3 \\ g_2 & \text{if } a = a_4 \end{cases} \end{aligned}$$

with the preference profile where h_1 is the only house in each agents' preference relation.

There are exactly two possible matchings,

$$\mu_1(a) = \begin{cases} h_1 & \text{if } a = a_1 \\ h_1 & \text{if } a = a_2 \\ h_0 & \text{if } a = a_3 \\ h_0 & \text{if } a = a_4 \end{cases}, \text{ and}$$

$$\mu_2(a) = \begin{cases} h_0 & \text{if } a = a_1 \\ h_0 & \text{if } a = a_2 \\ h_1 & \text{if } a = a_3 \\ h_1 & \text{if } a = a_4 \end{cases}.$$

In the case of μ_1, we have $\mu_0(a_3) \succ_{a_3} \mu_1(a_3)$, and in the case of μ_2, we have $\mu_0(a_1) \succ_{a_1} \mu_2(a_1)$. Thus, there is no possible individually rational matching.

Hence, no individually rational mechanism can exist for this problem.

□

However, we conjecture that a weaker form of the individual rationality constraint is satisfied by this mechanism, formalized in Conjecture 1.

Conjecture 1. Suppose μ is the matching produced by this mechanism. For every $a \in A$, if $\mu(a) \prec_a \mu_0(a)$, then for any other matching ν with $\nu(a) \succ_a \mu(a)$, there exists some $b \in A$ such that

(1) $\mu(b) \succ_b \nu(b)$ (b is strictly happier under μ)
(2) $f(b) < f(a)$ (b has higher priority than a)
(3) $\mu_0(a) = \mu_0(b)$ (a and b used to live together)

This conjecture essentially states that the only way that the individual rationality constraint can be broken for some a, is if it allows some previous roommate with higher priority than a to be assigned a more preferable room. Furthermore, we conjecture that it is individually rational for all agents a who have the highest priority amounts agents who previously lived in $\mu_(a)$.

Conjecture 2. This mechanism is strategy proof.

Conjecture 3. We define a priority h on the groups in G as

$$h(g_i) < h(g_j) \quad \longleftrightarrow \quad \exists\, a \in g_i \text{ such that } f(a) < f(b) \text{ for all } b \text{ in group } g_j.$$

Then the mechanism respects the seniority of groups under h.

6. Final Remarks

We provided algorithms for a number of variants of the housing allocation problem. The novel algorithm presented in Section 5 has a number of unproved conjectures about the mechanism that it provides, which we hope to prove at a later stage.

7. Acknowledgements

We thank Professor Alexander Postnikov of the MIT Mathematics Department for his suggestions about this topic and help in reviewing this paper.

References

[1] Abdulkadiroğlu, Atila and Sönmez, Tayfun, *House Allocation with Existing Tenants*, Journal of Economic Theory, 88, pp. 233–260, 1999.

[2] Hylland, Aanund and Zeckhauser, Richard, *The Efficient Alloclatonof Individuals to Positions*, Journal of Political Economy, 87, pp. 293–314, 1979.

[3] Ma, Jinpeng, *Strategy-Proofness and the Strict Core in a Market with Indivisibilities*, International Journal of Game Theory, 23, pp. 75–83.

[4] Roth, Alvin and Postlewaite, Andrew, *Weak Versus Strong Domination in a Market with Indivisible Goods*, Journal of Mathematical Economics, 4, pp. 131–137.

[5] Shapley, Lloyd and Scarf, Herbert, *On Cores and Indivisibility*, Journal of Mathematical Economics, 1, pp. 23–37, 1974.