Mechanism Design for Multi-type Housing Markets with Acceptable Bundles

Sujoy Sikdar and **Sibel Adalı** and **Lirong Xia**
Department of Computer Science
Rensselaer Polytechnic Institute
sikdas@rpi.edu, {sibel,xial}@cs.rpi.edu

Abstract

We extend the Top-Trading-Cycles (TTC) mechanism to select strict core allocations for housing markets with multiple types of items, where each agent may be endowed and allocated with *multiple items of each type*. In doing so, we advance the state of the art in mechanism design for housing markets along two dimensions: First, our setting is more general than multi-type housing markets (Moulin 1995; Sikdar, Adali, and Xia 2017) and the setting of Fujita et al. (2015). Further, we introduce *housing markets with acceptable bundles* (HMABs) as a more general setting where each agent may have arbitrary sets of *acceptable* bundles. Second, our extension of TTC is strict core selecting under the weaker restriction on preferences of CMI-trees, which we introduce as a new domain restriction on preferences that generalizes commonly-studied languages in previous works.

Introduction

Suppose there are two families: Family 1 owns two houses and a car, while Family 2 owns a house and two cars. How should they exchange their possessions when they each wish to have exactly as many houses and cars as they initially owned, and have preferences over combinations of houses and cars? Shapley and Scarf (1974)'s *housing markets* are an important model of such exchange economies. In a housing market, there are multiple agents, each initially endowed with some indivisible items and preferences over bundles of items. The goal is to design a mechanism without money to re-allocate the items.

When each agent is initially endowed with a single item and agents' preferences are linear orders over all items, Gale's *Top-Trading-Cycles (TTC)* mechanism (Abdulkadiroğlu and Sönmez 1999) satisfies many desirable properties. For example, it satisfies *core-selection* and *strategy-proofness*, and runs in polynomial time. Core-selection requires that the outcome of the mechanism must be in the *core*, which is the set of allocations where no group of agents has an incentive to deviate by reallocating their initial endowments. Strategy-proofness requires that no agent has incentive to misreport preferences to obtain a better outcome.

However, when some agents initially own multiple items, the problem becomes much more challenging as no mechanism satisfies both core-selection and strategy-proofness in such cases (Sönmez 1999). There have recently been two major positive developments on core-selecting mechanisms under natural assumptions on agents' preferences with certain constraints on the final allocation. Fujita et al. (2015) extend TTC under the assumption of lexicographic preferences, to the setting where agents are endowed with multiple items, and are allocated exactly as many items as their initial endowments. Sikdar, Adali, and Xia (2017) extend TTC to multi-type housing markets (Moulin 1995), where there are multiple types of items, and each agents' endowment and allocation consist of one item of each type, under the assumption that agents' preferences are represented by lexicographic extensions of CP-nets. However, (Fujita et al. 2015) do not consider multiple types of items, and lexicographic extensions of CP-nets considered in (Sikdar, Adali, and Xia 2017) cannot express preferences over bundles consisting multiple items of each type. This leaves the following open question: *How to redistribute items when there are multiple types of items and when agents are endowed with multiple items of each type?*

Our Contributions

We improve the state of the art in housing markets where agents may own multiple items along two dimensions:

Dimension 1: We introduce the setting where agents may be endowed and allocated with multiple items of each type. Our setting is more general than previous works (Moulin 1995; Fujita et al. 2015; Sikdar, Adali, and Xia 2017).

Dimension 2: We introduce a new domain restriction on preferences called *conditionally-most-important trees (CMI-trees)*. CMI-trees extends several popular languages studied in previous work (in particular, the preference representations assumed in (Fujita et al. 2015) and (Sikdar, Adali, and Xia 2017)), and represent preferences over bundles consisting of multiple items of each type.

We provide an extension of the TTC mechanism that is strict core selecting for housing markets where agents may be endowed and allocated with multiple items of each type under the assumption of CMI-tree preferences.

A housing market $\mathcal{M}$ is a tuple $(\mathcal{N}, \mathcal{I}, \mathcal{O})$, where $\mathcal{N}$ is a set of n agents, $\mathcal{I}$ is a set of indivisible items of p types, and $\mathcal{O}$ gives each agent's endowment of possibly multiple items of each type. The goal is to (re)allocate items to the agents based on their preferences over *bundles* of items, such that each agent is allocated exactly as many items of each type as their initial endowment.

In a CMI-tree V, each node is labeled with an item, and each directed edge points to the next most-important item, conditioned on whether items along the path are allocated to the agent. Given any partial allocation of items, an agent either (i) has a unique most important item such that every extension of the partial allocation which contains this item is preferred over every extension without this item, or (ii) is indifferent between all extensions of the partial allocation. A *strict* CMI-tree induces a linear order over all bundles. Strict CMI-trees generalize lexicographic extensions of CP-net preferences (Sikdar, Adali, and Xia 2017), GLPs (Monte and Tumennasan 2015), and LP-trees for housing markets with multiple types. We refer to the discussion on CMI-trees for more details and relationships with other languages. We characterize CMI-trees over any set of bundles as linear orders over equivalence classes. Importantly, our results hold for any preferences represented using any languages that can be represented as a CMI-tree.

At each round of the TTC algorithm, each remaining agent points at her (unique) most important item conditioned on the partial allocation computed so far, and every item points at its initial owner. The algorithm then implements all cycles formed in the current round by assigning to agents involved in a cycle the items they were pointing at.

We prove that **for strict CMI-profiles, TTC is strict core selecting (Theorem 1)** for housing markets. Moreover, cycles can be implemented one by one (instead of implementing all cycles in each step) in any order when they are available (**Theorem 2**), TTC is non-bossy (**Theorem 3**), and it is NP-complete to compute a beneficial manipulation (**Theorem 4**). Here, a strict core allocation means that no group has incentive to deviate and exchange their initial endowments to get a better allocation. Non-bossiness means that no agent can misreport her preferences to change the allocation of any other agent without changing her own allocation.

We propose a general framework called *housing markets with acceptable bundles (HMAB)* as a further generalization of our housing market setting along Dimension 1. An HMAB is denoted by $(\mathcal{M}, \mathcal{D})$, where $\mathcal{M}$ is a housing market, and each agent $j \leq n$ has a set of acceptable bundles D_j, and an *acceptable allocation* is a member of $\mathcal{D} = D_1 \times \cdots \times D_n$.

Our main result is that Theorems 1–3 extend naturally to HMABs. For any HMAB $(\mathcal{M}, \mathcal{D})$, and any CMI-profile P, TTC(P) is in the weak core if TTC(P) is an acceptable allocation (every agent is allocated an acceptable bundle) (**Theorem 5**), TTC(P) is in the strict core if TTC(P) is an acceptable full allocation (acceptable allocation where every item is allocated to some agent) (**Theorem 6**), cycles can be implemented one by one in any order when they are available (**Theorem 7**), and TTC is non-bossy at P (**Theorem 8**).

Theorems 5–8 discussed above are quite general and positive. They can be applied to any housing market with any acceptable bundles under the relatively weak assumption of CMI-tree preferences.

− First, if we can show that the output of TTC is always an acceptable (full) allocation, then it immediately implies that TTC is weak (strict) core-selecting, insensitive to the order of implementing cycles one by one, and non-bossy. To give an example of such settings, our setting of housing markets where agents must be allocated as many items of each type as their initial endowment is a special case of HMABs, where agents only accept bundles with exactly as many items of each type as their initial endowment.
− Second, even for problems for which the output of TTC is not always an acceptable full allocation, the theorems still guarantee desirable properties of TTC for “good” instances.

Related Work and Discussions. We are not aware of previous work that explicitly formulates acceptable bundles as we do in this paper. Our model is more general than assuming that all bundles ranked below the agent’s initial endowments are unacceptable, because we allow an agent to deem her initial endowment unacceptable. The acceptable bundles can be seen as *soft* constraints on the allocation. We do not aim to design a mechanism that always outputs an acceptable full allocation. Instead, we prove that TTC has good theoretical guarantees *when the output is an acceptable full allocation*.

Our work is related to the literature on matching with constraints (see (Kojima 2015) for a recent survey), where there are two sets of agents, and the goal is to find a stable matching of agents from opposite sides where no pair of agents has an incentive to deviate. Constraints may be in the form of agents specifying acceptable matchings, or quotas on how many other agents a given agent can be matched with (Fragiadakis et al. 2016). In this paper, we are interested in the resource allocation problem where preferences are one sided, i.e. agents have preferences over items, and the solution concept is that of core stability. Our notion of strict core is a natural extension of the standard strict core property to housing markets with acceptable bundles.

A natural extension of the standard housing markets setting is that agents own and desire multiple items, possibly of different types (Moulin 1995; Pápai 2007; Todo, Sun, and Yokoo 2014; Sonoda et al. 2014; Fujita et al. 2015; Sun et al. 2015; Sikdar, Adali, and Xia 2017). In a seminar class, students may want to exchange papers and dates for presentation (Mackin and Xia 2016); in cloud computing, agents may want to exchange multiple types of resources such as CPU, memory, and storage (Ghodsi et al. 2011; 2012); or patients may want to exchange multiple types of medical resources including surgeons, nurses, rooms, and equipment (Huh, Liu, and Truong 2013).

Konishi, Quint, and Wako (2001) showed that the core may be empty when there are multiple types of items, even when agents’ preferences are additively separable. However, two lines of work provide core selecting mechanisms under certain restrictions:

(1) When there is a single type and agents may own multiple items, the ATTC mechanism by (Fujita et al. 2015) is strict-core selecting under lexicographic preferences and allocates exactly as many items as agents’ endowments.

(2) In multi-type housing markets (Moulin 1995), agents are endowed with, and must be allocated bundles containing

exactly one item per type. (Sun et al. 2015) assume separable lexicographic preferences with common importance orders. Under preferences that are represented by lexicographic extensions of CP-nets (Boutilier et al. 2004) with possibly different importance orders, the MTTC mechanism by (Sikdar, Adali, and Xia 2017) is strict core-selecting and non-bossy.

Our setting of housing markets combine and generalize these two settings. Therefore, (Theorems 1– 3) also apply to every instance of the problems in (Fujita et al. 2015) and (Sikdar, Adali, and Xia 2017) under CMI-profiles that are linear orders over acceptable bundles, which is a weaker assumption than in previous works as we will show later. Aside from the standard TTC, the MTTC mechanism (Sikdar, Adali, and Xia 2017), and ATTC mechanisms (Fujita et al. 2015) are special cases of our extension of TTC. Theorems 2 and 3 are new to ATTC to the best of our knowledge. The NP-hardness of computing a beneficial misreport provides a computational barrier against agents' strategic behavior under TTC, similar to the idea of using computational complexity to protect elections (Bartholdi, Tovey, and Trick 1989; Conitzer and Walsh 2016).

Our setting is related to the work on housing markets with indifferences (Quint and Wako 2004; Yilmaz 2009; Alcalde-Unzu and Molis 2011; Jaramillo and Manjunath 2012; Aziz and de Keijzer 2012; Plaxton 2013; Saban and Sethuraman 2013) which assume that agents own and accept a single item and their preferences can be any weak order over the items. Therefore, their results do not directly apply to our setting where agents can be allocated multiple items. On the other hand, our results do not directly apply to the setting of these works since not all weak orders can be represented by CMI-trees. Theorems 5-8 hold for any preferences with indifferences which can be represented by CMI-trees.

Domain Restriction on Preferences. CMI-trees are more general, and impose a weaker restriction on agents' preferences than the assumptions of previous works such as lexicographic extensions of CP-nets (Sikdar, Adali, and Xia 2017), LP-trees (Booth et al. 2010), and GLPs (Monte and Tumennasan 2015). We look at the relationship with other languages in more detail later.

Preliminaries

A housing market, denoted $\mathcal{M}$, is given by a tuple $(\mathcal{N}, \mathcal{I}, \mathcal{O})$, where $\mathcal{N} = \{1, \ldots, n\}$ is a set of *agents*, $\mathcal{I}$ is a set of *indivisible items* consisting of $p \geq 1$ types of items, and for each $j \leq n$, $\mathcal{O}(j) \subseteq \mathcal{I}$ denotes agents' *initial endowments* that are disjoint. For any $i \leq p$, the set of items of type i is denoted by T_i. We refer to any subset of items $I \subseteq \mathcal{I}$ as a *bundle*. Given any subset $I \subseteq \mathcal{I}$, we use $[I]_i$ to denote the subset of items of type i in I. Each agent desires to consume exactly as many items of each type as they were initially endowed. A bundle $I \subseteq \mathcal{I}$ is *acceptable* for agent j, if for every type $i \leq p$, $|[I]_i| = |[\mathcal{O}(j)]_i|$. A bundle $\hat{I}$ *extends* a bundle I if $\hat{I} \supseteq I$.

An allocation A is a mapping from $\mathcal{N}$ to $2^{\mathcal{I}}$, such that for each $j \leq n$, $A(j)$ is the bundle allocated to j, and for any $\hat{j} \neq j$, $A(\hat{j}) \cap A(j) = \emptyset$. When $\bigcup_{j \leq n} A(j) = \mathcal{I}$, A is called a *full* allocation. Otherwise, it is a *partial* allocation. A (partial) allocation $\hat{A}$ *extends* A, if for every $j \leq n$, $\hat{A}(j) \supseteq A(j)$. Every (partial) allocation A induces a mapping $F_A : \mathcal{N} \to 2^{\mathcal{I}}$, where $F_A(j)$ is the set of items that are *forbidden* to agent j by A. A partial allocation A is *acceptable* given F_A if there exists an extension $\hat{A}$ of A, such that for every agent $j \leq n$, $\hat{A}(j)$ is acceptable to j and $\hat{A}(j) \cap F_A(j) = \emptyset$. An item $o \in \mathcal{I}$ is *forbidden* to agent j if there is no acceptable extension $\hat{A}$ of A where $o \in \hat{A}(j)$. Otherwise, o is *allowable*.

Example 1. *Consider the housing market with* 2 *types* $H = \{1_H, 1'_H, 2_H\}$ *and* $C = \{1_C, 2_C, 2'_C\}$, *and* 2 *agents with initial endowments* $\mathcal{O}(1) = \{1_H, 1'_H, 1_C\}$ *and* $\mathcal{O}(2) = \{2_H, 2_C, 2'_C\}$. *Consider the partial allocation* A *where* $A(1) = \{2_H, 1_C\}$ *and* $A(2) = \{1_H\}$. *For agent* 2, *the house* $1'_H$ *is forbidden because* 2 *cannot accept another house, items* $2_H, 1_C$ *are forbidden because they have been assigned to agent* 1, *and the items* $2_C, 2'_C$ *are allowable.* $F_A(1) = \{1_H, 2_C, 2'_C\}$ *and* $F_A(2) = \{2_H, 1'_H, 1_C\}$. *A can be extended by any combination of adding the items* $1'_H$ *to* $A(1)$, *or adding* 2_C *and* $2'_C$ *to* $A(2)$. *Every other extension is unacceptable.* □

A *preference profile* $P = (R_j)_{j \leq n}$ is a collection of agents' preferences, where R_j represents agent j's preferences over $2^{\mathcal{I}}$. Given a housing market $\mathcal{M}$, a mechanism f is a function that maps any profile P to a (partial) allocation.

Desirable Properties. We are interested in the following desirable properties. A mechanism f is: (a) *strict core selecting,* if for every profile P, there is no coalition which *weakly blocks* $f(P)$. A coalition of agents $S \subseteq \mathcal{N}$ weakly blocks an allocation A, if there is an acceptable reallocation B of items initially endowed to agents in S, such that (i) every agent in S weakly prefers her allocation in B to her allocation in A, and (ii) some agent in S strictly prefers her allocation in B to her allocation in A. (b) *weak core selecting,* if for every profile P, there is no coalition which *strictly blocks* $f(P)$. A coalition $S \subseteq \mathcal{N}$ strictly blocks an allocation A, if there is an acceptable reallocation B of items initially endowed to agents in S, such that every agent in S strictly prefers her allocation in B to her allocation in A. (c) *non-bossy,* if for every profile P, no agent can change another agent's allocation by misreporting her preferences, without changing her own allocation.

Conditionally-Most-Important (CMI) Preferences

Conditionally-Most-Important (CMI) preferences are represented by a tree defined below.

Definition 1. *Given* $\mathcal{I}$, *a CMI-tree* V *is a directed rooted tree, where*

- *Every node* d *is labeled with an item label*$(d) \in \mathcal{I}$.
- *Every item appears at most once on every branch.*
- *Every non-leaf node has either one outgoing edge labeled* $\{0, 1\}$ *or two outgoing edges labeled* 0 *or* 1.

If each node in V *only has one outgoing edge, then* V *is said to be* unconditional. *For any node* d, *let Anc*(d) *denote* *the set of all ancestor nodes of* d. *Let Path*$_V(d)$ *denote the path from root to* d. *Let Absent*(d) *(respectively, Present*(d)*) denote the set of all items labeling Anc*(d) *with outgoing edges labeled* 0 *(respectively,* 1*) along Path*$_V(d)$.

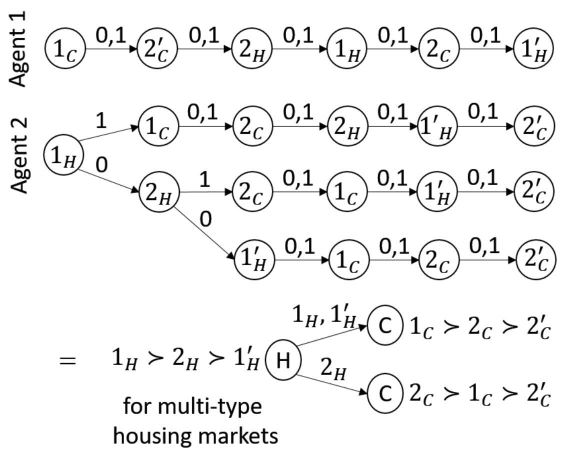

Figure 1: A strict CMI-profile with two agents and two types $H = \{1_H, 1'_H, 2_H\}$ and $C = \{1_C, 2_C, 2'_C\}$. Agent 2's preferences can be represented by an LP-tree with type labels for multi-type housing markets (Moulin 1995).

Semantics of CMI-trees. Given a CMI-tree, each node d has the following meaning: given that the agent is allocated all items in Present(d) and all of the items in Absent(d) are forbidden in a given partial allocation, any bundle which contains the item label(d) is more preferred than any bundle without label(d). We call item label(d) the agent's *most important item* conditioned on items Present(d) and Absent(d) being present and absent respectively in the partial allocation. Comparing a pair of bundles $B_1, B_2 \subseteq \mathcal{I}$ involves traversing the CMI-tree from the root, following outgoing edges labeled 0 or 1, depending on whether the item label(d) labeling the current node d is absent from or present in both B_1 and B_2, until a node d^* is encountered such that label(d^*) is present in only one of the bundles, which *decides* the preference relation in favor of the bundle that includes label(d^*). If no decision node is encountered, the agent is indifferent between B_1 and B_2.

Example 2. Consider the CMI-tree preferences of agent 2 in Figure 1. Comparing bundles $(1'_H 2_H 2_C)$ and $(1'_H 2_H 1_C)$ according to agent 2's preferences is performed by following edges corresponding to 1_H being absent, 2_H being present in both the bundles to reach the node labeled 2_C. Now, 2_C is the most important item and decides the pairwise relationship in favor of $(1'_H 2_H 1_C)$. □

A *CMI-profile* $P = (V_1, \ldots, V_n)$ is a collection of agents' preferences where each V_j is agent j's CMI-tree preferences over $2^{\mathcal{I}}$. A CMI-tree is said to be a *strict CMI-tree* over a set of bundles B if it induces a linear order over B. Given a market $\mathcal{M}$, a profile P is a *strict CMI-profile* if for every agent j, her preferences represented by V_j is a strict CMI-tree over $2^{\mathcal{I}}$.

Given a CMI-tree V representing the preferences of an agent, and sets I and F representing the sets of items that are assigned and forbidden to the agent respectively, we use $V|_{I,F}$ to denote the agent's most important item, conditioned on I being assigned and items in F being forbidden. Algorithm 1 computes $V|_{I,F}$ in polynomial time.

Example 3. Consider the CMI-profile in Figure 1, and the housing market with two agents and partial allocation A from Example 1. Agent 2's conditionally most important item 2_C is computed by traversing the CMI-tree representing agent 2's preferences as follows. Starting from the root node labeled 1_H, we follow the edge labeled 1 to the node labeled 1_C since 1_H is in $A(2)$. Since 1_C has been assigned to agent 1 and is therefore forbidden to agent 2, we follow the edge labeled 0 to node labeled 2_C which is allowable. □

Algorithm 1 Most important allowable item $V|_{I,F}$

Input: A CMI-tree V, assigned items I, and forbidden items F.
$o^* \leftarrow null$
Traverse V from the root following outgoing edges labeled 0 (resp. 1) from current node d, if the item label(d) is in F (resp. in I), until a node d^* is encountered where item label(d^*) is not forbidden. Set $o^* \leftarrow$ label(d^*). If no such node is found, do nothing.
return o^*

CMI-Trees Generalize Other Languages in Multi-Type Housing Markets

We start by showing that CMI-trees are a strict generalization of two important preference languages: (1) LP-trees with type labels (Booth et al. 2010) which are given by a tree V where each node v is labeled with a type $type(v)$, and a conditional preference table $CPT(v)$, and outgoing directed edges from each non-leaf node are labeled by the items of $type(v)$, such that each item in $type(v)$ labels exactly one outgoing edge of v. Each type appears once, and only once on every branch of V. $CPT(v)$ is composed of local preferences over items of $type(v)$, conditioned on the allocation of each of the items of the type corresponding to the parent node of v in V.
(2) Generalized lexicographic preferences (GLPs) (Monte and Tumennasan 2015) which are represented by a linear order η over $\mathcal{I}$. Given any pair $\vec{a}, \vec{b} \in 2^{\mathcal{I}}$, $\vec{a} \succ \vec{b}$ if and only if there is an item o in $\vec{a}$ but not in $\vec{b}$, and all items preferred to o in η are either in both $\vec{a}$ and $\vec{b}$, or in neither.

Example 4. For multi-type housing markets (Moulin 1995), agent 2's preferences in Figure 1 can be represented as an LP-tree as shown. Agent 1's preferences in Figure 1 realizes the GLP with $\eta = [1_C \succ 2'_C \succ 2_H \succ 1_H \succ 2_C \succ 1'_H]$. □

Proposition 1. *LP-trees with type labels and GLPs are strict subsets of strict CMI-trees.*

Proof. The proof is available in a full version. It is easy to check that we can model any GLP or LP-tree with type labels as a CMI-tree. We provide examples to prove that LP-trees with type labels and GLPs are a strict subset of CMI-trees. We then provide examples to show that not every GLP can be represented by an LP-tree with type labels and vice versa. Finally, we give an example of a CMI-tree which can neither be represented by an LP-tree with type labels nor a GLP. □

Relationship with LP-trees with item labels. An LP-tree with item labels is represented by a tree, where every node v is labeled by an item $item(v)$, each item is treated as a binary variable, and the local preferences specify whether it is preferred that the item labeling the node is present ($1 \succ 0$) or absent ($0 \succ 1$) from a bundle. Since each item appears once, and only once on every branch, this induces a linear order over $2^{\mathcal{I}}$, unlike LP-trees with type labels where preferences over bundles with multiple items of a type may not be well defined. We say that an LP-tree with item labels is *monotonic* if for every node v, $CPT(v) = 1 \succ 0$, (being assigned an item is always preferred). Otherwise, it is *non-monotonic*.

Proposition 2. *(1) Strict CMI-trees over $2^{\mathcal{I}}$ are equivalent to monotonic LP-trees with item labels, and (2) CMI-trees cannot represent the preferences of any non-monotonic LP-tree with item labels over $2^{\mathcal{I}}$.*

Figure 2 illustrates the relationship between CMI-trees, GLPs, LP-trees with type labels, and LP-trees with item labels which is discussed below.

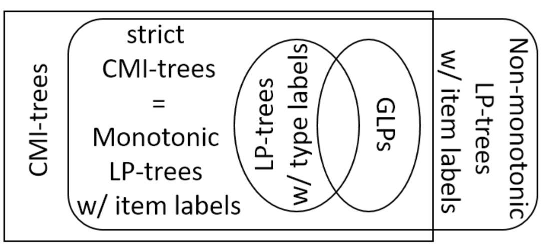

Figure 2: Relationship between popular preference languages (Propositions 1 and 2).

Properties of CMI-trees. CMI-trees are compact in the same way that LP-trees are compact. The size of the representation and time taken to compute most important items may depend on the amount of branching in the CMI-tree. Importantly, they can be compact for special cases such as when they represent the same preferences as an LP-tree or GLP. We note that computing the most important item and deciding pairwise comparisons can be performed with polynomial number of queries for the next conditionally most important item. This can be done efficiently for the special cases where CMI-trees are compact. Also, note that TTC only requires knowledge of agents' most important items at each round, instead of requiring agents to formulate and communicate full preferences a priori.

Our next proposition is that CMI-trees induce a weak order, which is a linear order over a set of equivalence classes, with no incomparabilities.

Proposition 3. *Given any CMI-tree V and any $D \subseteq 2^{\mathcal{I}}$, the restriction of V to D is a weak order over D.*

Not all linear orders or equivalence classes can be represented by CMI-trees. Consider the example with two houses 1_H and 2_H. For example, no CMI-tree can represent $1_H \succ 1_H 2_H \succ 2_H$ over $D = \{1_H, 2_H, 1_H 2_H\}$. The next proposition states that it is easy to check whether a given linear order over equivalence classes can be represented by a CMI-tree. An algorithm and proof are available in the full version.

Proposition 4. *Given a partition $D = E_1 \cup \cdots \cup E_k$ of $2^{\mathcal{I}}$, and a strict linear order $\succ$ over the E_i's, there is a polynomial time algorithm that decides whether $\succ$ can be represented by a CMI-tree.*

CMI-trees as defined in Definition 1 may not be a compact representation, just like LP-trees (Booth et al. 2010) and CP-nets (Boutilier et al. 2004) are not necessarily compact representations. Irrespective of the size of representation, it is possible that CMI-trees can be compactly represented by a computable function f that outputs the most important item given any partial allocation. Our results apply to all compact languages that are special cases of CMI-trees. Instead of further pursuing compact representations, we use CMI-trees as weak domain constraints in our theorems for TTC.

TTC is Strict Core Selecting Under Strict CMI-tree Preferences

Algorithm 2 TTC for CMI-profiles.

Input: $\mathcal{M} = (\mathcal{N}, \mathcal{I}, \mathcal{O})$ and a strict CMI-profile P.
$t \leftarrow 1$. For each $j \leq n$, $A^t(j) = \emptyset$, $F_{A^t}(j) = \emptyset$.
Initialize forbidden items. For each $j \leq n$, each $i \leq p$, add T_i to $F_{A^t}(j)$ if j has no endowment of type i.
while at least one agent remains **do**
  Identify most important item $o_j^* = V_j|_{A^t(j), F_{A^t}(j)}$ using Algorithm 1 for every agent $j \in \mathcal{N}$.
  Build the graph $G_t = (\mathcal{N} \cup \mathcal{I}, E)$**.** For every agent j in $\mathcal{N}$:
    6.1 For every item $o \in \mathcal{O}(j) \cap \mathcal{I}$, add (o, j) to E.
    6.2 If $o_j^* \neq null$, add edge (j, o_j^*) to E.
  Implement cycles. For each cycle C in G_t, for every (j, o_j^*) in C, add o_j^* to $A^t(j)$, remove o_j^* from $\mathcal{I}$, make o_j^* forbidden to every other agent, and update $F_{A^t}(j)$.
  For all agents j with no outgoing edge in G_t, remove j from $\mathcal{N}$, remove remaining items in $\mathcal{O}(j)$ from $\mathcal{I}$ and make them forbidden to every remaining agent.
  For every agent $j \in \mathcal{N}$, update $A^{t+1}(j) \leftarrow A^t(j)$, $F_{A^{t+1}}(j) \leftarrow F_{A^t}(j)$. Set $t \leftarrow t + 1$.
return The allocation $A = (A^t(j))_{j \leq n}$

At each round of Algorithm 2, we start by identifying the most important item for every remaining agent using Algorithm 1 by skipping items that are forbidden until a most important allowable item is found. Every remaining agent points to their most important allowable item, and each remaining item points to its initial owner. Every node in the

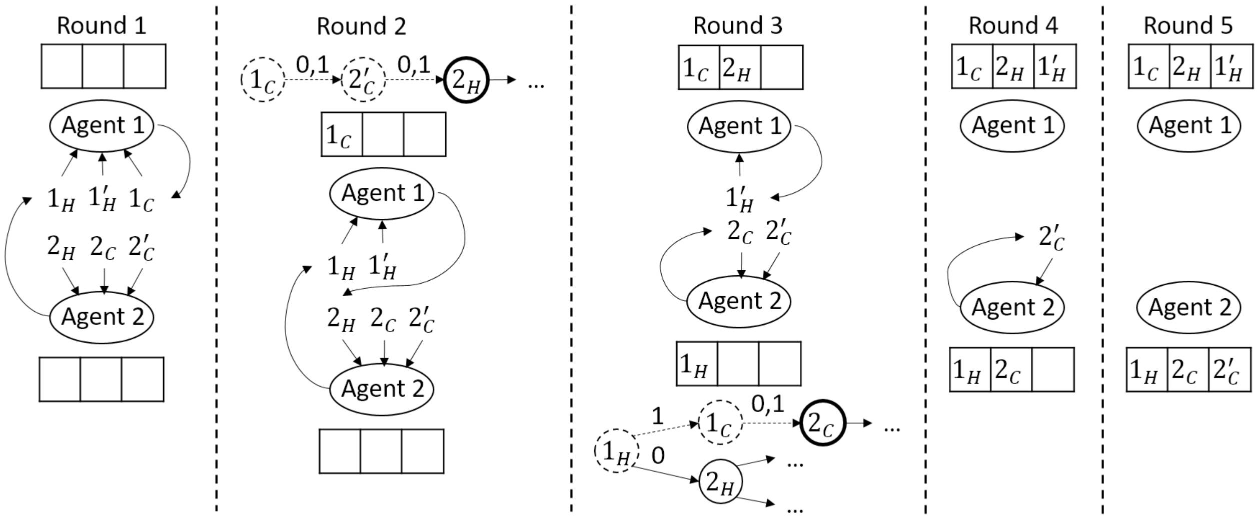

Figure 3: A run of TTC for the housing market from Example 1 and preferences in Figure 1.

corresponding graph has an out-degree of exactly 1, so a cycle is guaranteed to exist. TTC proceeds by implementing all available cycles at each round, by assigning to agents involved in each cycle, the item they were pointing at, updating the partial allocations, removing the assigned items, and forbidding them to all other remaining agents. At any round t of the algorithm, the partial allocation is A^t, and for any remaining agent j, $V_j|_{A^t(j),F_{A^t}(j)}$ represents j's most important item under consideration. If no most important allowable item can be identified for an agent, she is removed along with the remaining items from her endowment, which are made forbidden to every remaining agent.

Example 5. Figure 3 is an example of a run with the CMI-profile in Figure 1 as input for the housing market from Example 1. The partial allocations A^t at the beginning of the round are shown. Computation of most important items (darker circle), by following the dashed nodes and edges, are shown for agent 1 at round 2 and agent 2 at round 3. □

Theorem 1. *For any housing market $\mathcal{M}$ and any strict CMI-profile P, TTC(P) is a strict core allocation, an acceptable full allocation, and computed in polynomial time.*

Proof sketch. The complete proof is available in the full version.

Claim 1. *For any strict CMI-profile P and housing market $\mathcal{M}$, TTC(P) is an acceptable full allocation, and computed in polynomial time.*

A proof by induction shows that at the end of every round t, the (partial) allocation A^t is acceptable, and the number of remaining items of each type is exactly the number of items to fulfill the demand for the type.

In the interest of space, we provide a proof sketch for only the base case. At round 1, by the assumption of strict CMI-tree preferences, every agent has a most important allowable item, and at Step 6 of round 1, agents only point at an item of type i if they have demand for items of type i. At least one cycle must exist, and cycles are disjoint. Then, at the end of round 1: items assigned in Step 7 are allowable to agents assigned the items, and that for any type i, the number of remaining items, and the demand for items of that type reduce by exactly the number of items of type i assigned to agents in Step 7. It follows that the number of remaining items of any type i is exactly the number of items demanded by the agents, and that A^1 is acceptable.

It follows from the structure of housing markets that if TTC(P) is acceptable, it is a full allocation, and it is easy to see that TTC(P) is computed in polynomial time.

Claim 2. *For any strict CMI-profile P and any housing market $\mathcal{M}$, TTC(P) is a strict core allocation.*

Suppose for sake of contradiction, that the allocation $A =$ TTC(P) does not belong to the strict core. Then, there exists a coalition S and an acceptable allocation B on S that blocks A, i.e. that every agent in S weakly prefers B to A, and at least one agent in S strictly prefers B to A.

Let t^* be the first round in Algorithm 2 when the partial assignments at the end of the round for agents in S are not compatible with B, by which we mean that there exists an agent $j \in S$ such that either (i) $A^{t^*}(j) \not\subseteq B(j)$ (j gets an item in $A^{t^*}(j)$ which she does not get in $B(j)$), or (ii) $F_{A^{t^*}}(j) \cap B(j) \neq \emptyset$ (an item in $B(j)$ gets forbidden).

We note that A^{t^*} and $F_{A^{t^*}}$ are updated only at Steps 7 and 8 of Algorithm 2. We can ignore the case where A is a partial allocation because an agent was removed in Step 8 due to Claim 1 that Algorithm 2 always outputs acceptable full allocation for housing markets.

(Case 1) **Conflict in Step 7.** An item o is assigned to j in A, but is not assigned to j in B. We will show that this implies that $A(j) \succ_j B(j)$. By definition of Algorithm 2, o is j's most important item given A^{t^*} immediately before the Step 7. By our assumption on t^*, Step 7 is the first time A^{t^*} is incompatible with $B(j)$. Therefore, right before o is assigned to j, $B(j)$ is an acceptable extension of $A^{t^*}(j)$. Therefore, every item in $B(j) \setminus A^{t^*}(j)$ is allowable. By definition of CMI-trees, $A(j) \succ B(j)$, a contradiction to our assumption that B blocks A.

(Case 2) **Removed item in Step 7.** An item o is assigned to j in B, but is assigned to another agent, say $\hat{j}$, in A^{t^*}, in Step 7. Let C be the cycle that is implemented to assign o to $\hat{j}$. Suppose $\hat{j} \in S$, $o \in B(j)$ implies that $o \notin B(\hat{j})$. This is equivalent to (Case 1). Suppose $\hat{j} \notin S$. Since $o \in B(j)$, there exists an agent $j' \in S$ such that $o \in \mathcal{O}(j')$, and by definition of Algorithm 2, $j' \in C$. Now, consider the agents in C. By definition of Algorithm 2, there exists a pair of agents j'' and j''', and an item $o''' \in \mathcal{O}(j''')$ such that $j'' \in S$, $j''' \notin S$, and j'' points to o''' in C. This implies that $o''' \in A(j'')$. Since $j''' \notin S$, this implies that $o''' \notin B(j'')$. Again, this reduces to (Case 1).

(Case 3) **Denied item in Step 7.** An item o is assigned to j in B but immediately after Step 7, o is remaining but forbidden. We will show that $B(j)$ is unacceptable. By our assumption on the minimality of t^*, B is an acceptable extension of A^{t^*}, immediately before Step 7. Suppose an item $\hat{o}$ was added to $A^{t^*}(j)$ in Step 7. Suppose $\hat{o} \notin B(j)$, this reduces to (Case 1). Suppose $\hat{o} \in B(j)$, and $o \in B(j)$ gets forbidden, we must have that $B(j)$ is unacceptable, a contradiction to our assumption that B is acceptable.

□

Our next theorem states that the order of implementing cycles in the execution of TTC does not matter. We begin by defining a class of algorithms called TTC* for CMI-profiles similar to (Sikdar, Adali, and Xia 2017). The proof of Theorem 3 that TTC is non-bossy relies on Theorem 2. Missing proofs are available in a full version online.

Definition 2. *Given a housing market $\mathcal{M}$ and a CMI-profile P, let TTC$^*_P$ denote the set of algorithms, each of which is a modification of TTC (Algorithm 2), where instead of implementing all cycles in each round, the algorithm implements exactly one available cycle in each round.*

Theorem 2. *For any strict CMI-profile P and any housing market $\mathcal{M}$, the output of every TTC$^*_P$ algorithm is the same and equals TTC(P).*

Theorem 3. *For any strict CMI-profile P and any housing market $\mathcal{M}$, TTC is non-bossy at P.*

Manipulation of TTC. TTC is not necessarily strategy-proof under strict CMI-tree preferences, as shown through an example in the full version. However, as we show subsequently (Theorem 4), computing a beneficial misreport is NP-complete, for GLPs, LP-trees, and CMI-trees using a reduction from ATTC-BENEFICIAL-MISREPORT for ATTC (Fujita et al. 2015) which is NP-complete. The proof is available in the full version.

Definition 3. *(*TTC-BENEFICIAL-MISREPORT*) Given a housing market $\mathcal{M}$, a profile P, and an agent j^*, we are asked if agent j^* has a misreport $\hat{V}_{j^*}$ so that $TTC(\hat{P} = (\hat{V}_{j^*}, V_{-j^*}))(j^*) \succ_{V_{j^*}} TTC(P)(j^*)$.*

Theorem 4. TTC-BENEFICIAL-MISREPORT *is NP-complete under GLP, LP-tree, or CMI preferences.*

Housing Markets with Acceptable Bundles

A housing market with acceptable bundles (HMAB), is given by a tuple $(\mathcal{M}, \mathcal{D})$, where $\mathcal{M}$ is a housing market, and $\mathcal{D} = D_1 \times \cdots \times D_n$, where for each $j \leq n$, let $D_j \subseteq 2^{\mathcal{I}}$ denote agent j's *acceptable bundles*. Given an HMAB $(\mathcal{M}, \mathcal{D})$, a mechanism f is a function that maps agents' profile P to a (partial) allocation. We extend TTC (Algorithm 2) naturally to HMABs and general CMI-profiles, and prove that several desirable properties are retained.

Algorithm 2 is not guaranteed to run in polynomial time for HMABs in general, since updating the set of forbidden items in Step 7 can potentially take exponential time for arbitrary acceptable bundles. However, given oracle access to a function g which outputs the set of forbidden items, TTC(P) can be computed in polynomial time. We note that for housing markets, the set of forbidden items can always be computed in polynomial time.

Theorem 5. *For any HMAB $(\mathcal{M}, \mathcal{D})$ and any CMI-profile P, if TTC(P) is an acceptable allocation, then it is in the weak core.*

Theorem 6 states that TTC(P) is a strict core allocation whenever TTC(P) is an acceptable full allocation. Theorems 7 and 8 are extensions of Theorems 2 and 3 respectively to HMABs. The proofs of Theorems 6–8 are similar to the proofs of Theorems 1–3. TTC$^*_P$ for HMABs are defined similarly to housing markets.

Theorem 6. *For any HMAB $(\mathcal{M}, \mathcal{D})$ and any CMI-profile P, if TTC(P) is an acceptable full allocation then it is in the strict core.*

Theorem 7. *For any HMAB $(\mathcal{M}, \mathcal{D})$ and any CMI-profile P, the output of every TTC$^*_P$ algorithm is the same and equals TTC(P).*

Theorem 8. *For any HMAB $(\mathcal{M}, \mathcal{D})$, any CMI-profile P, TTC is non-bossy at P.*

Applications and Future Directions

We extended TTC to HMABs when agents' preferences are represented by CMI-trees, and proved that TTC satisfies desirable properties when the output is an acceptable (full) allocation. We also showed that TTC is strict core selecting under strict CMI profiles for housing markets where agents may be endowed and allocated with multiple items of each type, which combine the settings of previous works. Although TTC may not be strategy-proof, computing a beneficial manipulation is NP-hard. Open questions include characterizations of other properties of the extended TTC under other types of preferences, and how to choose an acceptable allocation when the output of TTC is unacceptable.

Acknowledgments

We thank all anonymous reviewers for helpful comments and suggestions. This work is supported by NSF #1453542 and ONR #N00014-17-1-2621.

References

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

Alcalde-Unzu, J., and Molis, E. 2011. Exchange of indivisible goods and indifferences: The Top Trading Absorbing Sets mechanisms. *Games and Economic Behavior* 73:1–16.

Aziz, H., and de Keijzer, B. 2012. Housing Markets with Indifferences: A Tale of Two Mechanisms. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 1249–1255.

Bartholdi, III, J.; Tovey, C.; and Trick, M. 1989. The computational difficulty of manipulating an election. *Social Choice and Welfare* 6(3):227–241.

Booth, R.; Chevaleyre, Y.; Lang, J.; Mengin, J.; and Sombattheera, C. 2010. Learning conditionally lexicographic preference relations. In *Proceeding of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, 269–274.

Boutilier, C.; Brafman, R.; Domshlak, C.; Hoos, H.; and Poole, D. 2004. CP-nets: A tool for representing and reasoning with conditional ceteris paribus statements. *Journal of Artificial Intelligence Research* 21:135–191.

Bouveret, S.; Endriss, U.; and Lang, J. 2009. Conditional importance networks: A graphical language for representing ordinal, monotonic preferences over sets of goods. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI'09, 67–72.

Conitzer, V., and Walsh, T. 2016. Barriers to Manipulation in Voting. In Brandt, F.; Conitzer, V.; Endriss, U.; Lang, J.; and Procaccia, A., eds., *Handbook of Computational Social Choice*. Cambridge University Press. chapter 6.

Fragiadakis, D.; Iwasaki, A.; Troyan, P.; Ueda, S.; and Yokoo, M. 2016. Strategyproof matching with minimum quotas. *ACM Transactions on Economics and Computation* 4(1):6.

Fujita, E.; Lesca, J.; Sonoda, A.; Todo, T.; and Yokoo, M. 2015. A Complexity Approach for Core-Selecting Exchange with Multiple Indivisible Goods under Lexicographic Preferences. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 907–913.

Ghodsi, A.; Zaharia, M.; Hindman, B.; Konwinski, A.; Shenker, S.; and Stoica, I. 2011. Dominant Resource Fairness: Fair Allocation of Multiple Resource Types. In *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, 323–336.

Ghodsi, A.; Sekar, V.; Zaharia, M.; and Stoica, I. 2012. Multi-resource Fair Queueing for Packet Processing. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, volume 42, 1–12.

Huh, W. T.; Liu, N.; and Truong, V.-A. 2013. Multiresource Allocation Scheduling in Dynamic Environments. *Manufacturing and Service Operations Management* 15(2):280–291.

Jaramillo, P., and Manjunath, V. 2012. The difference indifference makes in strategy-proof allocation of objects. *Journal of Economic Theory* 147(5):1913–1946.

Kojima, F. 2015. Recent Developments in Matching Theory and its Practical Applications. Working paper.

Konishi, H.; Quint, T.; and Wako, J. 2001. On the Shapley–Scarf economy: the case of multiple types of indivisible goods. *Journal of Mathematical Economics* 35(1):1—15.

Mackin, E., and Xia, L. 2016. Allocating Indivisible Items in Categorized Domains. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, 359–365.

Monte, D., and Tumennasan, N. 2015. Centralized allocation in multiple markets. *Journal of Mathematical Economics* 61:74—85.

Moulin, H. 1995. *Cooperative Microeconomics: A Game-Theoretic Introduction*. Prentice Hall.

Pápai, S. 2007. Exchange in a general market with indivisible goods. *Journal of Economic Theory* 132(1):208—235.

Plaxton, C. G. 2013. A Simple Family of Top Trading Cycles Mechanisms for Housing Markets with Indifferences. In *Proceedings of the 24th International Conference on Game Theory*.

Quint, T., and Wako, J. 2004. On Houseswapping, the Strict Core, Segmentation, and Linear Programming. *Mathematics of Operations Research* 29(4):861–877.

Roth, A. E.; Sonmez, T.; and Unver, M. U. 2004. Kidney exchange. *Quarterly Journal of Economics* 119(2):457–488.

Saban, D., and Sethuraman, J. 2013. House allocation with indifferences: a generalization and a unified view. In *Proceedings of the fourteenth ACM conference on Electronic Commerce*, 803–820.

Shapley, L., and Scarf, H. 1974. On cores and indivisibility. *Journal of Mathematical Economics* 1(1):23–37.

Sikdar, S.; Adali, S.; and Xia, L. 2017. Mechanism Design for Multi-Type Housing Markets. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*.

Sönmez, T. 1999. Strategy-proofness and essentially single-valued cores. *Econometrica* 67(3):677–689.

Sonoda, A.; Fujita, E.; Todo, T.; and Yokoo, M. 2014. Two Case Studies for Trading Multiple Indivisible Goods with Indifferences. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 791–797.

Sun, Z.; Hata, H.; Todo, T.; and Yokoo, M. 2015. Exchange of Indivisible Objects with Asymmetry. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, 97–103.

Todo, T.; Sun, H.; and Yokoo, M. 2014. Strategyproof Exchange with Multiple Private Endowments. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 805–811.

Yilmaz, Ö. 2009. Random assignment under weak preferences. *Games and Economic Behavior* 66(1):546–558.