Matching under multiple single-peaked domains

Abhigyan Bose*[1] and Souvik Roy†[2]

[1,2]Economic Research Unit, Indian Statistical Institute, Kolkata

September 12, 2023

Abstract

We model a housing market where each agent is endowed with a house, and a social planner uses a matching rule to allocate these houses amongst the agents. Agents have multiple single-peaked preferences over these houses . We introduce the C-TTC algorithm which encapsulates the idea of the Crawler algorithm from Bade (2019) and the famous Top-trading cycle (TTC) algorithm. We finally show that the C-TTC algorithm satisfies the three desirable properties of individual rationality, pareto optimality and strategy-proofness.

JEL Classification : C78, D47

Keywords : Shapley-Scarf housing market, multiple single-peaked preference, C-TTC algorithm

*Contact: boseabhigyan@gmail.com
†Corresponding Author, Contact: souvik.2004@gmail.com

1. Introduction

We look at the housing problem introduced in Shapley and Scarf (1974). There are n houses and n agents. Initially, each agent owns a unique house and different agents own different houses. Agents have strict preferences over the houses. A matching rule reallocates the houses amongst the agents based on the preferences of the agents over the houses. A social designer's problem is to find matching rules that satisfy certain desirable properties. The matching literature (Sönmez (1999), Sethuraman (2016), Ma (1994)) focuses predominantly on three desirable properties for such matching rules: Individual rationality (IR), Pareto optimality (PO), and strategy-proofness. IR requires that the house allocated to an agent is at least as good as his/her initial endowed house. PO guarantees that there is no "better" way to allocate the houses, that is, there is no other way to allocate the houses so that every agent will be weakly better off and some agents will be strictly better off. Strategy-proofness ensures that no agent can (unilaterally) misreport his/her preference and get a strictly better allocation.

When agents' preferences over the set of houses are allowed to be unrestricted, we know from Roth and Postlewaite (1977),Roth (1982) and Ma (1994) that Gale's Top Trading Cycle (TTC) algorithm is the only matching rule that satisfies these three conditions. However the uniqueness of the result does not hold when the domain of preferences is restricted. Bade (2019) provides a different algorithm called the Crawler, which satisfies IR, PO, and strategy-proofness on the domain of single-peaked preferences. This has further inspired studies on other restricted preference domains to explore the nature of matching rules that satisfy these three conditions(e.g., Tamura (2023)).

In this paper, we have considered the multiple single-peaked domain introduced in Reffgen (2015). While single-peaked preferences are defined with respect to an underlying ordering which is common for all agents, multiple single-peaked preference domain is an union of several single-peaked domains with different underlying orderings. Thus it captures scenarios where there is no single objective way to order the houses like house size or distance from the nearest public school etc. Instead each agent may have different subjective perception of how to order the houses , but given their personal understanding of the ordering, they still have single-peaked preferences.

Multiple single-peaked preferences capture a wide variety of preference domains. If all the agents' orderings are same, then we have the usual single-peaked domain. On the other extreme, if we allow for every possible agents' orderings , then we have the unrestricted domain. However in reality, agents' personal understanding of the orderings are often not too dissimilar from each other. Agents agree on the basic structure but differ in the exact positioning of some of the houses in their underlying orderings. More precisely, we assume that agents can partition the set of houses into H_L, H_M, H_R such that they agree on the exact positioning of the houses in H_L and H_R and the relative positioning of the houses in H_M with respect to H_L and H_R. However they disagree about the exact positioning of the houses in H_M. Since there is no common ordering over the houses in the set H_M, the Crawler algorithm fails to be defined in this domain. On the other hand, although the TTC algorithm still satisfies IR, PO and strategy-proofness, it is a complicated algorithm as it requires every agent, at every step to point to their best house in the remaining set and then compute and resolve all cycles formed. In this paper, we have introduced a new simpler matching rule called the C-TTC algorithm. The C-TTC algorithm combines the Crawler with the TTC algorithm. We subsequently show that the C-TTC algorithm satisfies the three desirable properties.

The rest of the paper is organized as follows. Section 2 introduces the basic model. In section 3, we define the C-TTC algorithm. In section 4 we show that the C-TTC algorithm satisfies individual rationality, pareto optimality and strategy-proofness.

2. Preliminaries

Let $N = \{1, \ldots, n\}$ denote the set of agents and $H = \{h_1, \ldots, h_n\}$ denote the set of houses.

A preference P over H is a complete, transitive and anti-symmetric binary relation on H. We denote the weak part of a preference P with R i.e. aRb implies either aPb or $a = b$. A preference is called single-peaked with respect to a prior order $\prec$ if the farther away one goes from his/her top ranked house, the lower is their rank. Formally we define single-peaked preference as follows:

Definition 2.1. A preference P is said to satisfy single peakedness with respect to a prior order $\prec$ if for all $a, b \in H$, aPb whenever $[r_1(P) \prec a \prec b$ or $b \prec a \prec r_1(P)]$.[1]

For a prior ordering $\prec$, the single-peaked domain $\mathcal{S}(\prec)$ is defined as the collection of preferences on H such that they satisfy single-peakedness with respect to the prior order $\prec$.

The houses are partitioned into three parts: $H_L = \{h_1, \ldots, h_{\underline{\kappa}}\}$, $H_M = \{h_{\underline{\kappa}+1}, \ldots, h_{\bar{\kappa}-1}\}$, and $H_R = \{h_{\bar{\kappa}}, \ldots, h_n\}$. We denote by $\mathbb{L}$ the set of prior (linear) orders on the set H such that for each $\prec \in \mathbb{L}$, $h_t \prec h_{t+1}$ for all $t \in \{1, \ldots, \underline{\kappa} - 1\}$, $h_t \prec h_{t+1}$ for all $t \in \{\bar{\kappa}, \ldots, n-1\}$, and $h_x \prec h_y \prec h_z$ for all $x \in \{1, \ldots, \underline{\kappa}\}$, $y \in \{\underline{\kappa} + 1, \ldots, \bar{\kappa} - 1\}$, and $z \in \{\bar{\kappa}, \ldots, n\}$. Throughout the paper, we assume that $\underline{\kappa}$ and $\bar{\kappa}$ (and hence, H_L, H_M, H_R) are arbitrary but fixed. We denote by $\mathcal{S}$ the union of all single-peaked domains with respect to prior orders in $\mathbb{L}$, that is, $\mathcal{S}(\mathbb{L}) = \cup_{\prec \in \mathbb{L}} \mathcal{S}(\prec)$.

[1] $r_1(P)$ denotes the 1st ranked alternative in preference P.

Each agent $i \in N$ has a single-peaked preference over H with respect to some prior order in $\mathbb{L}$, that is, a preference in $\mathcal{S}$, denoted by P_i. An element of $\mathcal{S}^n$ is called a preference profile and is denoted by $P_N = (P_1, \ldots, P_n)$.

A matching $\mu : N \to H$ is a bijection from the set of agents to the set of houses. For convenience, we sometimes denote μ^{-1} by μ. For instance, we write $\mu(h) = i$ to mean that agent i is matched to the house h in the matching μ. Let $\mathcal{M}$ be the set of all matching. The houses are already matched with the agents according to a matching e, which we call the initial endowment. Similarly, given an endowment e, we denote by $o(h)$ the owner of the house h, that is, $o(h) = i$ if and only if $e(i) = h$. WLOG, we assume $e(i) = h_i$ for all $i \in N$.

A matching rule $\varphi : \mathcal{S}^n \to \mathcal{M}$ is a function that maps each preference profile to a matching.

A matching rule is said to be individually rational if for every preference profile, the house allocated to any agent is at least as good as his/her initial endowed house.

Definition 2.2. A matching $\mu \in \mathcal{M}$ is individually rational (IR) at the preference profile $P_N \in \mathcal{S}^n$ if for each $i \in N$, $\mu(i) R_i e(i)$. A matching rule $\varphi : \mathcal{S}^n \to \mathcal{M}$ is individually rational (IR) if $\varphi(P_N)$ is IR at every $P_N \in \mathcal{S}^n$.

A matching rule is pareto optimal if for every preference , no group of agents can exchange their allocated houses amongst themselves and be weakly better off.

Definition 2.3. A matching $\mu \in \mathcal{M}$ is pareto optimal (PO) at the preference profile $P_N \in \mathcal{S}^n$ if there exists no other matching $\mu' \in \mathcal{M}$ such that

(i) for each $i \in N$, $\mu(i)' R_i \mu(i)$, and

(ii) for some $j \in N$, $\mu(j)' P_j \mu(j)$.

A matching rule $\varphi : \mathcal{S}^n \to \mathcal{M}$ is pareto optimal (PO) if $\varphi(P_N)$ is PO at every $P_N \in \mathcal{S}^n$.

A matching rule is said to be strategy-proof if no agent can be better off by misreporting his/her true preference.

Definition 2.4. A matching rule $\varphi : \mathcal{S}^n \to \mathcal{M}$ is strategy-proof if for all $i \in N$, $P_i, P_i' \in \mathcal{S}$ and $P_{-i} \in \mathcal{S}^{n-1}$, $\varphi_i(P_i, P_{-i}) R_i \varphi_i(P_i', P_{-i})$.

3. C-TTC ALGORITHM

In this section we introduce the C-TTC algorithm. The C-TTC algorithm is a sort of combination of the Crawler and TTC algorithm. Roughly at each stage, it applies a restricted TTC on the agents in H_M, a *left* Crawler on the agents in H_L and a *right* Crawler on the agents in H_R. [2]

We now formally define the C-TTC algorithm. Consider a preference profile P_N. The C-TTC algorithm at P_N is defined as follows:

Initialize : Set $H_L^1 = H_L$, $H_M^1 = H_M$ and $H_R^1 = H_R$. We denote the endowment at stage s for any agent $i \in N$ by $e^s(i)$. Set $e^1(i) = e(i)$. Let $o^s(h)$ be the owner of the house h at stage s.

Stage s. Let the remaining houses in H_L, H_M, H_R at stage s , denoted by H_L^s, H_M^s, H_R^s, be indexed such that $H_L^s = \{h_1^s, h_2^s, \ldots, h_{\underline{\kappa}(s)}^s\}$, $H_M^s = \{h_{\underline{\kappa}(s)+1}^s, h_{\underline{\kappa}(s)+2}^s, \ldots, h_{\bar{\kappa}(s)-1}^s\}$, $H_R^s = \{h_{\bar{\kappa}(s)}^s, h_{\bar{\kappa}(s)+1}^s, \ldots, h_{n(s)}^s\}$ where $n(s)$ denotes the number of houses still remaining at stage s. Here one or more sets in H_L^s, H_M^s, H_R^s may be empty. . Define $\bar{H}_M^s = \{h_{\underline{\kappa}(s)}^s, h_{\underline{\kappa}(s)+1}^s, \ldots, h_{\bar{\kappa}(s)-1}^s, h_{\bar{\kappa}(s)}^s\}$.

DIRECTED GRAPH: Construct the directed edges of the graph on $H_L^s \cup H_M^s \cup H_R^s$ as follows :

If $h \in H_L^s$, then $o^s(h)$ points to his/her best house in the set $H_L^s \cup \bar{H}_M^s$. If $h \in H_M^s$, then $o^s(h)$ points to his/her best house in the set $\bar{H}_M^s$. If $h \in H_R^s$, then $o^s(h)$ points to his/her best house in the set $H_R^s \cup \bar{H}_M^s$.

OUTCOME RESOLUTION: Let h_j^s be the first house from the left (i.e. low-

[2] A left Crawler is the one defined in Bade (2019). A right Crawler is it's 'dual' version by proceeding in the reverse order. Tamura and Hosseini (2022) provides a precise description.

est indexed) such that its owner $o^s(h_j^s)$ does not point to his/her right. Suppose $o^s(h_j^s)$ points to $h_{\hat{j}}^s$. Match $o^s(h_j^s)$ with $h_{\hat{j}}^s$ and get them out of the algorithm. Define the modified owners of the remaining houses $H_L^{s+1} := \{h_1^s, \dots, h_{\hat{j}-1}^s, h_{\hat{j}+1}^s, \dots, h_{\underline{\kappa}(s)}^s\}$ as follows: $o^{s+1}(h_k^s) = o^s(h_k^s)$ for all $k < \hat{j}$, $o^{s+1}(h_k^s) = o^s(h_{k-1}^s)$ for all $\hat{j} < k \leq j$, and $o^{s+1}(h_k^s) = o^s(h_k^s)$ for all $j < k < \underline{\kappa}(s)$. Reindex the remaining houses in H_L^{s+1} as $\{h_1^{s+1}, \dots, h_{\underline{\kappa}(s+1)}^{s+1}\}$ where $h_k^{s+1} = h_k^s$ for all $k < \hat{j}$, and $h_k^{s+1} = h_{k+1}^s$ for all $\hat{j} \leq k \leq \underline{\kappa}(s+1)$ and $\underline{\kappa}(s+1) = \underline{\kappa}(s) - 1$.

Let h_i^s be the first house from the right (i.e. highest indexed) such that its owner $o^s(h_i^s)$ does not point to his/her left. Suppose $o^s(h_i^s)$ points to $h_{\hat{i}}^s$. Match $o^s(h_i^s)$ with $h_{\hat{i}}^s$ and get them out of the algorithm. Define the modified owners of the remaining houses $H_R^{s+1} := \{h_{\bar{\kappa}(s)}^s, \dots, h_{\hat{i}-1}^s, h_{\hat{i}+1}^s, \dots, h_{n(s)}^s\}$ as follows: $o^{s+1}(h_k^s) = o^s(h_k^s)$ for all $k < i$, $o^{s+1}(h_k^s) = o^s(h_{k+1}^s)$ for all $i \leq k < \hat{i}$, and $o^{s+1}(h_k^s) = o^s(h_k^s)$ for all $\hat{i} < k < n(s)$. Reindex the remaining houses in H_R^{s+1} as $\{h_{\bar{\kappa}(s+1)}^{s+1}, \dots, h_{n(s+1)}^{s+1}\}$ where $h_k^{s+1} = h_k^s$ for all $\bar{\kappa}(s+1) \leq k < \hat{i}$, and $h_k^{s+1} = h_{k+1}^s$ for all $\hat{i} \leq k \leq n(s+1)$. where $\bar{\kappa}(s+1) - n(s+1) = \bar{\kappa}(s) - n(s) - 1$

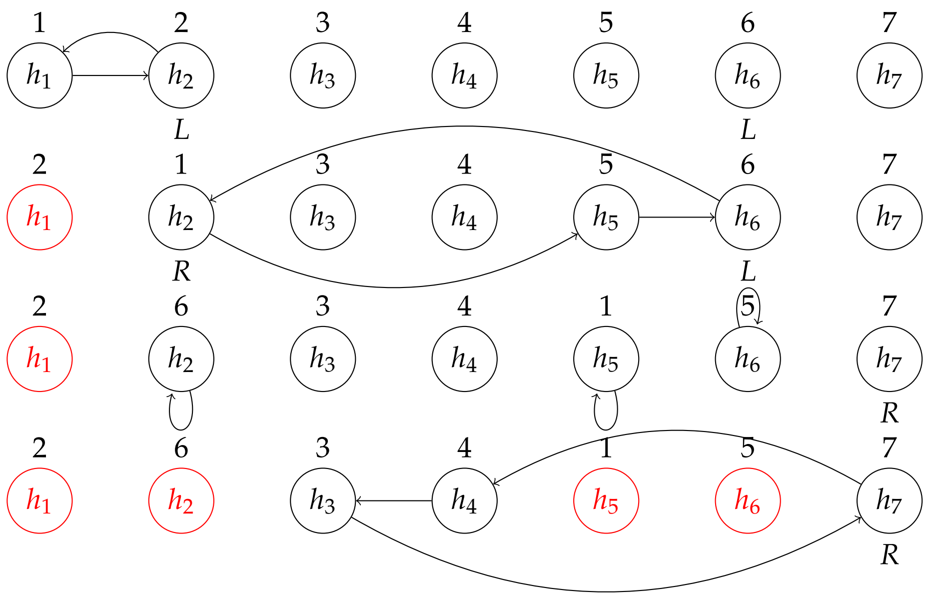
For each cycle $\{h_{j_1}^s, \dots, h_{j_k}^s\}$ involving the houses in H_M^s, then match $o^s(h_{j_l}^s)$ with $h_{j_{l+1}}^s$ for all $l = 1, \dots, k$ where $k+1 = 1$ and get them out of the algorithm. Index the remaining houses H_M^{s+1} as $\{h_{\underline{\kappa}(s+1)+1}^{s+1}, \dots, h_{\bar{\kappa}(s+1)-1}^{s+1}\}$ such that $h_{\underline{\kappa}(s+1)+1}^{s+1} \prec \dots \prec h_{\bar{\kappa}(s+1)-1}^{s+1}$ (Here we take any arbitrary but fixed $\prec$ from $\mathbb{L}$ for representation purpose). For each cycle $\{h_{j_1}^s, \dots, h_{j_k}^s\}$ in $\bar{H}_M^s$ involving $h_{\underline{\kappa}(s)}^s$ or $h_{\bar{\kappa}(s)}^s$, define the modified owner for stage $s+1$ as : $o^{s+1}(h_{j_{l+1}}^s) = o^s(h_{j_l}^s)$ for all $l = 1, \dots, k$ where $k+1 = 1$. Keep the owners of all the other houses unchanged.

We continue this procedure until all the houses are out of the algorithm.

Example 3.1. Consider $N = \{1, \dots, 7\}$ and $H_L = \{h_1, h_2\}$, $H_M = \{h_3, h_4, h_5\}$, $H_R = \{h_6, h_7\}$ and a preference profile given as follows:

P_1	P_2	P_3	P_4	P_5	P_6	P_7
h_5	h_1	h_7	h_5	h_6	h_1	h_2
h_2	h_2	h_6	h_3	h_7	h_2	h_5
h_3	h_5	h_3	h_4	h_5	h_3	h_1
h_4	h_3	h_4	h_2	h_4	h_4	h_4
h_6	h_4	h_5	h_6	h_3	h_5	h_3
h_7	h_6	h_2	h_1	h_2	h_6	h_6
h_1	h_7	h_1	h_7	h_1	h_7	h_7

Table 1



The above diagram shows how the C-TTC algorithm functions. The circles denote the houses while the agents occupying a house is labelled at the top of that house. Houses marked red are those that are no longer in the market. For ease of presentation we have only presented the arrows from the agents whose endowments are changing in a particular stage or if they are leaving the algorithm.

At stage 1, agents in H_L^1 point to their best house in $H_L^1 \cup \bar{H}_M^1 = \{h_1, \ldots, h_6\}$. Thus agents 1 and 2 point to h_5 and h_1 respectively. Agents in H_R^1 point to their best house in $H_R^1 \cup \bar{H}_M^1 = \{h_2, \ldots, h_7\}$. Thus agents 6 and 7 point to houses h_2 and h_2 respectively. Agents in H_M^1 point to their best house in $\bar{H}_M^1 = \{h_2, \ldots, h_6\}$. Thus agent 3, 4 and 5 point to houses h_6, h_5 and h_6 respectively.

Agent 2 is the first agent from the left in H_L^1 who does not point to the right and so he/she is allowed to choose his/her best house i.e. h_1 and agent 1 who is currently occupying house h_1 is shifted one house to the right i.e. h_2. All agents in H_R^1 point to their left and thus there is no agent who is the first from the right in H_R^1 who does not point to the left. We further note that no cycle is formed involving owners of the houses in the set $\bar{H}_M^1$. We end the stage 1 with agent 2 leaving with house h_1.

At stage 2, we again construct the directed graph as before. Observe that there are no agents in H_L^2 who do not point to their right and no agents in H_R^2 who do not point to their left. A cycle is formed involving agents 1, 5 and 6. Since this cycle is not restricted to H_M^2 (as it involves the boundary houses), we simply exchange the houses and no one leaves.

At stage 3, agent 6 who currently occupies h_2 is the first agent in H_L^3 not pointing to the right. So he/she leaves the market by pointing to his/her best i.e. h_2. Agent 5 who occupies h_6 is the first agent from the right in H_R^3 not pointing to the left. So he/she leaves the market by pointing to his/her best i.e. h_6. The only cycle formed in H_M^3 is with agent 1 who points to his/her current house h_5. Since this cycle involves only agents in H_M^3, agent 1 leaves with house h_5.

Finally at stage 4, we have the remaining agents 3, 4 and 7 who form a cycle. Since this cycle involves houses in $\bar{H}_M^4$, we exchange the houses and do not let them leave at this stage. However they will leave in the following stage, when all of them will point towards themselves in the similar way agents $6, 1, 5$ left the market at stage 3.

OBSERVATION 3.1. Each agent moves in an unidirectional fashion from the starting endowment till they reach their outcome in the C-TTC algorithm i.e. for any two stage s, s' and agent i, $e^s(i) \prec e(i) \prec e^{s'}(i)$ or $e^{s'}(i) \prec e(i) \prec e^s(i)$ is not possible.

OBSERVATION 3.2. At any preference profile P_N, if agent $i \in N$ with $e(i) \notin H_M$, is matched with a house $h \in H_M$ at some stage s, then at some stage $s^* < s$ he/she must have reached a boundary house (i.e. either $h^{s^*}_{\underline{\kappa}(s^*)}$ or $h^{s^*}_{\bar{\kappa}(s^*)}$). He/she must have been unable to form any cycle till stage $s-2$ and formed a cycle at stage $s-1$ by pointing to his/her matched house h. Since this cycle does not only involve houses in H^{s-1}_M, he/she does not immediately leave the algorithm with h at this stage $s-1$. He/she is moved to the house h but still remains in the algorithm. Finally at stage s, he/she forms a cycle by pointing to his/her own house $h \in H^s_M$ and leaves the algorithm with h.

4. RESULTS

Theorem 4.1. *The C-TTC algorithm satisfies individual rationality, pareto optimality and strategy-proofness.*

Before we prove Theorem 4.1, we state and prove two Lemmas which we will need for the proof of Theorem 4.1.

Lemma 4.1. *For any profile $P_N \in \mathcal{S}^n$ and any agent $i \in N$, if i is matched with a house $h \in H$ at stage s of the C-TTC algorithm, then h is the most preferred house among the houses remaining at stage s according to P_i.*

Proof. Suppose agent i gets matched at stage s. We distinguish the following cases.

Case 1: Suppose $i \in H_L^s$. By the definition of the C-TTC algorithm, i must have pointed to himself or to his/her left at stage s. By the domain assumption $r_1(P_i, H_L^s \cup \bar{H}_M^s) \in H_L^s$ implies that $r_1(P_i, H^s) \in H_L^s$ (since if $r_1(P_i, H^s) \in H_R^s$ then $H_L \prec H_M \prec H_R$ and single-peakedness of the domain imply that $r_1(P_i, H_L^s \cup \bar{H}_M^s)$ must be $h^s_{\bar{\kappa}(s)}$ i.e. the right boundary of $\bar{H}_M^s$). Since i is matched with $r_1(P_i, H_L^s \cup \bar{H}_M^s) \in H_L^s$ at stage s, this implies that i gets his/her most preferred house in the remaining houses. A similar logic applies for the case when $i \in H_R^s$.

Case 2: Suppose $i \in H_M^s$. By the definition of the C-TTC algorithm, he/she must have pointed to his/her best house in $\bar{H}_M^s$. Since i gets matched at stage s, by the definition of the C-TTC algorithm, i's most preferred house in $\bar{H}_M^s$ is in H_M^s. By the domain assumption, $r_1(P_i, \bar{H}_M^s) \in H_M^s$ implies that $r_1(P_i, H^s) \in H_M^s$ (if $r_1(P_i, H^s) \in H^s_{L(R)}$, then $H_L \prec H_M \prec H_R$ and single-peakedness of the domain imply that $r_1(P_i, H^s) = h^s_{\underline{\kappa}(s)(\bar{\kappa}(s))}$ i.e. the left (right) boundary of $\bar{H}_M^s$).Since i is matched with $r_1(P_i, \bar{H}_M^s) \in H_M^s$ at stage s, this implies that i gets his/her most preferred house in the remaining houses.

This completes the proof of Lemma 4.1. ∎

Lemma 4.2. *Suppose an agent $i \in N$ has a preference P_i such that $r_1(P_i) \prec e(i)$ ($e(i) \prec r_1(P_i)$). Then agent i cannot be better-off by misreporting to a preference P_i' such that $r_1(P_i) \prec e(i) \prec r_1(P_i')$ ($r_1(P_i') \prec e(i) \prec r_1(P_i)$).*

Proof. This follows directly from the single-peaked nature of the domain of preferences and Observation 3.1. ∎

Now let us begin with the proof of Theorem 4.1.

Proof. Strategy-proofness : Consider an agent $i \in N$ at a preference profile $P_N \in \mathcal{S}^n$ getting matched with $\phi_i(P_N)$ at stage s. We need to show that $\phi_i(P_i, P_{-i}) R_i \phi_i(P_i', P_{-i})$ for all $P_i' \in \mathcal{S}$. We distinguish the following cases.

Case 1: Consider the case where $e(i) \in H_L$ and $\phi_i(P_N) \in H_L$. If $\phi_i(P_N) \prec e(i)$, then agent i must have pointed to his/her left at every stage $t < s$. This implies that at each of these t stages, agent i was not the first agent who pointed to his/her left. Now when i misreports to P_i', at each of these stages $t < s$, agent i is still not the first agent who pointed to his/her left and thus the agents getting matched at each of these stages remains unchanged. Thus agent i cannot be matched at an earlier stage than s under P_i'. This along with Lemma 4.1 ensures that i cannot be better off by misreporting P_i'. Now suppose $e(i) \prec \phi_i(P_N)$. At each of the stages $t < s$, then first agent from the left who is pointing to his/her left remains unchanged since agent i under misreported pref P_i' still points to his/her right (by Lemma 4.2). Thus the agents getting matched as well as the outcome modification remains unchanged at each of these stages $t < s$.Thus agent i cannot be matched at an earlier stage than s under P_i'. This along with Lemma 4.1 ensures that i cannot be better off by misreporting P_i'. This completes the proof for case 1.

Case 2: Consider the case where $e(i) \in H_L$ and $\phi_i(P_N) \in H_M$. This implies that at some stage $s^* < s$ agent i must have moved to the boundary house of $H_L^{s^*}$. At each stage $s^* < t < s-1$ agent i pointed to his/her best house in the set $\bar{H}_M^t$ but was unable to form any cycle (from Observation 3.2). Suppose i misreports to P_i'. By similar logic as the $e(i) \prec \phi_i(P_N)$ sub-case of Case 1, we argue that at any stage $t < s^*$, agent i cannot move to the boundary house of H_L^t. If agent i misreports such that he/she leaves earlier then $\phi(P_i', P_{-i}) \in H_L$ which implies that $\phi(P_i, P{-}i) R_i \phi(P_i', P_{-i})$ due to the domain assumption. Now if P_i' be such that he/she reaches the boundary house of $H_L^{s^*}$ at stage s^* and he/she is able to form a cycle $\{h_{j_1}^v, \ldots, h_{j_l}^v\}$ at some stage $s^* < v < s$ where $o(h_{j_1}^v) = i$. We rule out the case where he/she points to the boundary house of H_R^v i.e. $h_{j_2}^v \in H_R^v$ since that would imply that he/she would move into the boundary house of H_R^v and by Observation 3.1, $\phi(P_i', P_{-i}) \in H_R$ which is worse than $\phi(P_i, P_{-i}) \in H_L$. Thus i points to some house $h_{j_2}^v \in H_M^v$ and moves into that house at the end of the stage v and subsequently leaves the market with $h_{j_2}^v$ by forming a cycle at stage $v + 1$. We now argue that this cycle must have been available at stage v when agent i was truthfully reporting P_i. Consider the agent $o(h_{j_l}^v)$ who is pointing to agent i's house to form the cycle. As long as i remains in the set $\bar{H}_M^v$, $o(h_{j_l}^v)$ will continue pointing to agent i. Again as long as $o(h_{j_l}^v)$ remains in the set $\bar{H}_M^v$, $o(h_{j_{l-1}}^v)$ will continue pointing to agent $o(h_{j_l}^v)$. Going in this manner it is evident that all the agents of the cycle will be in the algorithm till agent i leaves the set $\bar{H}_M^t$ or completes the cycle at some stage $t > v$. But under P_i, agent i had pointed to his/her best house in $\bar{H}_M^t$ for each stage $< s^* < t < s$. This implies that $\phi(P_i, P_{-i}) R_i \phi(P_i', P_{-i}) = h_{j_2}^v$. This completes the proof for case 2.

Case 3: Consider the case where $e(i) \in H_L$ and $\phi_i(P_N) \in H_R$. This implies that at some stage $s^* < s$ agent i must have moved to the boundary house of $H_L^{s^*}$ and at stage $s^* < \bar{s} < s$ agent i must have moved to the boundary house of $H_R^{\bar{s}}$. Thereafter at each stage $\bar{s} < t < s$,agent i was not the first agent from the right in H_R^t who did not point to the left. Finally at stage s, agent i leaves the algorithm by pointing his/her best house in H_R^s. Suppose agent i misreports to P_i'. By similar arguments as the $e(i) \prec \phi(P_N)$ sub-case of Case 1, we argue that i cannot do any better than reach the boundary house of H_L at a stage s^*. At the boundary house of H_L, if i is able to form a cycle by pointing to some house $h \in H_M^v$ at stage v, then i leaves the algorithm with h at stage $v + 1$. By similar logic as in Case 2, we observe that this cycle must have been available to i under truthful reporting at stage $\bar{s}$. This implies that i prefers the boundary house of H_R at stage $\bar{s}$ than h. Hence i cannot do any better than reach the boundary house of H_R at a stage $\bar{s}$. Once i reaches the boundary house of H_R at a stage $\bar{s}$, no matter what he/she reports, i cannot change his/her outcome until all agents to his right who do not point to their left leave. This completes the proof for this case.

Case 4: Consider the case where $e(i) \in H_M$ and $\phi_i(P_N) \in H_L$. This implies that at some stage $s^* < s$ agent i must have moved to the boundary house of $H_L^{s^*}$, i.e. $h^{s^*}_{\underline{\kappa}(s^*-1)}$. Thereafter, i stayed at the boundary house of $H_L^{s^*}$ till stage $s-1$ and finally left the algorithm at stage s by being the first agent from the left in H_L^s who did not point to the right. Suppose agent i misreports to P_i'. By Lemma 4.2, $r_1(P_i') \notin H_R$. Now suppose agent i can form a cycle $\{h^v_{j_1}, \ldots, h^v_{j_l}\}$ such that i points to some house $h \in \bar{H}^v_M$ at some stage v. If $hR_ih^{s^*}_{\underline{\kappa}(s^*-1)}$, then i must have pointed to h at some stage before s^*. Moreover using similar arguments as in Case 2, this cycle must have been available to i at that stage since all agents of this cycle will remain in the algorithm. Since i pointed to the boundary house $h^{s^*}_{\underline{\kappa}(s^*-1)}$, it must be the case that $h^{s^*}_{\underline{\kappa}(s^*-1)}R_ih$. Since i choose $\phi(P_N)$ at stage s when $h^{s^*}_{\underline{\kappa}(s^*-1)}$ was available to him/her, $\phi(P_N)R_ih^{s^*}_{\underline{\kappa}(s^*-1)}$. This implies that $\phi(P_N)R_ih$. Hence i cannot manipulate by forming a cycle in H_M. Now by misreporting, agent i cannot get any better house in H_M which implies that he/she will move into the boundary house $h^{s^*}_{\underline{\kappa}(s^*-1)}$ at some stage under P_i'. Once i reaches the boundary house, no matter what he/she reports, he/she cannot change his outcome until all agents to his left who do not point to their right leave. This completes the proof for this case.

Case 5 : Consider the case where $e(i) \in H_M$ and $\phi_i(P_N) \in H_M$. Suppose agent i misreports to P_i' and forms a cycle at some stage t by pointing to a house $h \in \bar{H}^t_M$. Since this cycle must be available to i when he/she points to $\phi(P_N)$ under truthful reporting, it must be the case that $\phi(P_N)R_ih$. This completes the proof for this case.

Case 6 : Consider the case where $e(i) \in H_M$ and $\phi_i(P_N) \in H_R$. This is the symmetric case of Case 4. Similar arguments follow.

Case 7 : Consider the case where $e(i) \in H_R$ and $\phi_i(P_N) \in H_L$. This is the symmetric case of Case 3. Similar arguments follow.

Case 8 : Consider the case where $e(i) \in H_R$ and $\phi_i(P_N) \in H_M$. This is the symmetric case of Case 2. Similar arguments follow.

Case 9 : Consider the case where $e(i) \in H_R$ and $\phi_i(P_N) \in H_R$. This is the symmetric case of Case 1. Similar arguments follow.

This completes all possible cases and thus completes the strategy-proof part of the theorem.

Pareto optimality : At each stage s, the agents who can leave the market are the first agent from the left in H_L^s who does not point to his/her right or the first agent from the right in H_R^s who does not point to his/her left and any set of agents in H_M^s who form a cycle among themselves. Now from Lemma 4.1, we know that each of these agents are leaving with their most preferred house in the remaining set of houses. Thus starting from the first stage, at each stage the agents leaving the C-TTC algorithm are leaving with their best house in the remaining set of houses. This proves that the C-TTC algorithm satisfies Pareto optimality.

Individual rationality : Fix any agent $i \in N$. Let P_i^* denote the preference of i such that $r_1(P_i^*) = e(i)$. We now argue that for any P_{-i} , $\phi(P_i^*, P_{-i}) = e(i)$. Consider any agent with $e(i) \in H_L$. He/she must point to his/her own house at each stage. At every stage s whenever he/she is not matched, it must be the case that there is some other agent j such that $e^s(j) \prec e^s(i)$. This agent will leave with his/her best house h such that $h \prec e^s(j) \prec e^s(i)$ by the definition of the C-TTC algorithm. Moreover the outcome modification under such cases will be such that $e^s(i) = e^{s+1}(i)$. Finally there will be a stage s^* such that agent i will be the first agent from the left who doesn't point to the right. At this stage, agent i will leave the market with $e^{s^*}(i) = e(i)$. One can argue using similar logic for the case of an agent with $e(i) \in H_R$. Finally when $e(i) \in H_M$, agent i will point towards himself at the first stage and form an own cycle and gets out of the market with $e(i)$. This completes all the cases and proves that whenever an agent's own endowment is his/her most preferred house, then he/she will always be matched that house irrespective of the preference of other agents. Now using the fact that $\phi()$ is strategy-proof, we have $\phi(P_i, P_{-i}) R_i \phi(P_i^*, P_{-i}) = e(i)$ for every P_i, P_{-i}. This completes the proof that C-TTC algorithm is individually rational. ■

References

Bade, S. (2019). with single-peaked preferences. *Journal of Economic Theory*, 180:81–99.

Ma, J. (1994). Strategy-Proofness and the Strict Core in a Market with Indivisibilities. *International Journal of Game Theory*, 23(1):75–83.

Reffgen, A. (2015). Strategy-proof social choice on multiple and multi-dimensional single-peaked domains. *Journal of Economic Theory*, 157:349–383.

Roth, A. E. (1982). Incentive compatibility in a market with indivisible goods. *Economics Letters*, 9(2):127–132.

Roth, A. E. and Postlewaite, A. (1977). Weak versus strong domination in a market with indivisible goods. *Journal of Mathematical Economics*, 4(2):131–137.

Sethuraman, J. (2016). An alternative proof of a characterization of the ttc mechanism. *Operations Research Letters*, 44(1):107–108.

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

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

Tamura, Y. (2023). Object reallocation problems with single-dipped preferences. *Games and Economic Behavior*, 140:181–196.

Tamura, Y. and Hosseini, H. (2022). The crawler: Three equivalence results for object (re)allocation problems when preferences are single-peaked. *Journal of Economic Theory*, 203:105466.