Robust Exchange*

Yuichiro Kamada† and Yosuke Yasuda‡

September 17, 2025

Abstract

This paper offers the first rigorous economic analysis of the *dropout* problem—a practical challenge in designing exchange markets. We introduce a model incorporating situations where participants may drop out of the market before transactions occur, which results in the failure of all transactions within the same exchange cycle. We show that the celebrated Top Trading Cycles (TTC) mechanism is disadvantaged due to its potential reliance on large exchange cycles, which implies a high likelihood of transaction defaults. Furthermore, agents may not have an incentive to truthfully report their preferences under the TTC mechanism when dropouts can happen. To improve efficiency, we propose k-greedy mechanisms designed to manage the risks associated with exchange cycle sizes while maintaining reasonable efficiency, with a constraint on the maximum possible cycle size of k. We show, both theoretically and through simulations, that the k-greedy mechanisms (with small values of k) outperform the TTC mechanism.

Keywords: top trading cycles, cycle size, dropout, efficiency, k-greedy mechanism

*We thank Péter Biró, Haluk Ergin, Di Feng, Fuhito Kojima, Alex Teytelboym, Gerelt Tserenjigmid, Rakesh Vohra, and seminar participants at Doshisha University, UC Berkeley, CLIMB Workshop, ERATO Kojima Market Design Project, the Lisbon Meetings in Game Theory and Applications #13, Greater Bay Area Market Design Workshop 2024 at the University of Macau, the 15th East Asia Theory Conference at HKU Business School, Summer Workshop on Economic Theory 2025 in Otaru, and the 2025 World Congress of the Econometric Society in Seoul for their helpful comments. Nanami Aoi, Asuka Hirano, Jingzhuo Wu, and especially Jiarui Xie provided excellent research assistance. All remaining errors are ours.

†University of California, Berkeley, Haas School of Business, Berkeley, 2220 Piedmont Avenue, CA 94720, USA and The University of Tokyo, Faculty of Economics, 7-3-1 Hongo, Bunkyoku, Tokyo, 113-0033, Japan. E-mail: y.cam.24@gmail.com

‡The University of Osaka, Graduate School of Economics, 1-7 Machikaneyama, Toyonaka, Osaka 560-0043 Japan. E-mail: yosuke.yasuda@gmail.com

1 Introduction

1.1 Motivation

Allocation of scarce resources is the central theme of economics, and market design has found applications in various allocation problems in which monetary transfer is not allowed. The two gold standards in this literature are the deferred acceptance mechanism (Gale and Shapley, 1962) and the Top Trading Cycles (TTC) mechanism (proposed by Gale and formalized by Shapley and Scarf (1974)). The deferred acceptance mechanism is used for medical matching and school choice in many countries.[1] In contrast, the applications of the TTC mechanism are limited, with a notable exception being kidney exchange, where, oftentimes, the preference structure is simple and severe restrictions on cycle sizes are imposed.[2] Why are applications of the TTC mechanism rarely observed in practice?

We argue that this is because (i) the TTC mechanism results in large cycles,[3] and (ii) if agents drop out of the market after the assignment is determined but before the transactions occur, that causes a cascade of other agents in the same cycle to be affected: the dropping agent would leave the market with her initial endowment, and the agent who was supposed to receive that object can no longer receive it, which leads him to receive his initial endowment, and so on. Figure 1 demonstrates how large the cycles can be under the TTC mechanism. These histograms record the number of cycles and the number of agents involved in each cycle size, respectively, in exchange markets with 200 agents, where we generated the preferences of each agent uniformly at random for 100 simulation runs. The cycle sizes can be quite high, where 20% of agents are involved in the cycle size of 18 or greater.[4]

The large cycles have a significant effect on the agents' welfare when agents drop out. It is a modeling practice to assume that once a mechanism determines an as-

[1] For medical matching, see Roth (1984) and Roth and Peranson (1999) for seminal work. See Roth (2008) for an extensive review of applications. For school choice, see Balinski and Sönmez (1999) and Abdulkadiroğlu and Sönmez (2003) for seminal work. See Fack et al. (2019) for an extensive review of applications.

[2] See Roth et al. (2005a,b) and Ashlagi and Roth (2021) for TTC-like mechanisms in practical kidney exchange. Another exceptions are a few applications in school choice (Abdulkadiroglu et al., 2017; Morrill and Roth, 2024), which we discuss in Section 4.4.

[3] We formally define the TTC mechanism in Section 2.1.

[4] The probability that at least one agent out of 18 agents drops out is more than 30% even when the dropout probability is as low as 2%.

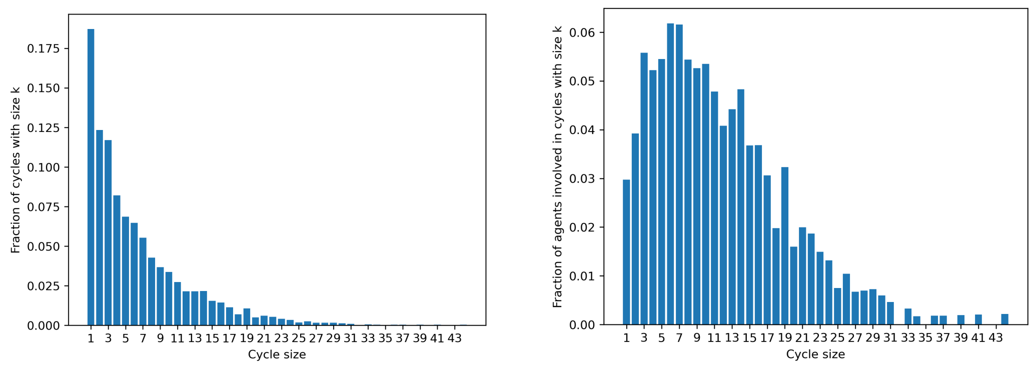


Figure 1: Fraction of cycle sizes (the left panel) and the fraction of agents involved in each cycle size (the right panel) under the TTC mechanism. We set $N = 200$ in both panels.

signment, it is finalized. However, in real-life markets, some agents drop out of the market after their assignment is determined but before transactions occur, and we take this reality seriously. We are agnostic about why agents drop out and treat a dropout as a probability-p event that can happen to each agent independently. If an agent drops out, then all agents in the same cycle will be *affected*, meaning that they will have to receive their initial endowment.[5] When there are 200 agents and their preferences are randomly generated, we varied p and recorded the average performance of the resulting mechanisms over 100 simulation runs. In Figure 2, the left panel records the number of agents who are assigned their l-th choice or better for each l (hence the mechanism is better if the corresponding curve is higher) and the right panel records the fraction of agents who receive someone else's object at a given p relative to the number with $p = 0$. The graphs show that the introduction of small p has a significant effect on the performance of the TTC mechanism.

1.2 Contribution

This paper offers the first rigorous economic analysis of the dropout problem—a practical challenge in designing exchange markets. In an attempt to prevent the problem caused by large cycle sizes and dropouts, we propose the approach of *robust*

[5] One could imagine a situation where the affected agents participate in a certain rewiring process among themselves. While we do not primarily consider such a scenario to focus on the first-order effect of dropouts, we will briefly discuss this issue as well (cf. Section 4.3).

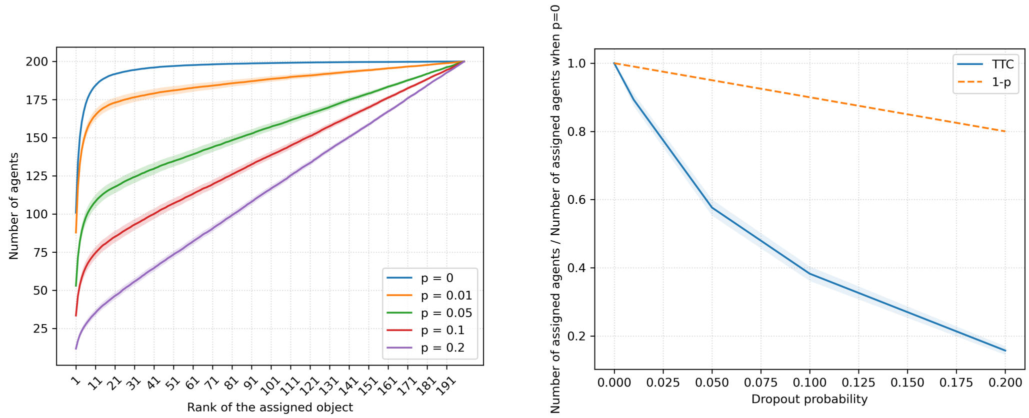


Figure 2: The performance of the TTC mechanism under various p (the left panel) and the fraction of agents who receive someone else's object at a given p relative to the number with $p = 0$. We set $N = 200$ in both panels.

exchange, where we impose a restriction on the cycle sizes. With such a restriction, one agent's dropout can only cause a small number of affected agents, and in that sense, the output of the mechanism is robust to dropouts.

We first consider a market without dropouts and examine the effect of a cycle size restriction. We show that such a restriction is incompatible with strategy-proofness and what we call k-unanimity—the condition requiring that every cycle that respects the cycle size restriction (the cycle size no greater than k) and gives each involved agent their first-choice object must be implemented. Given this result, we propose a new mechanism called the *k-greedy mechanism*, where the parameter k represents the maximum cycle size allowed. This mechanism respects the cycle size restriction, unanimity, and (constrained) efficiency. The mechanism is based on what we call the *k-greedy algorithm*, and we prove that it runs in polynomial time. We also show that, although the k-greedy mechanism is not strategy-proof (since it respects k-unanimity), it has certain good incentive properties where agents would not be able to find an "obvious way" in which they can improve their assignment.

We then turn to the model that explicitly incorporates the possibility of dropouts. We first show that, under the possibility of dropouts, the TTC mechanism no longer has a desirable incentive property: every agent truthfully reporting their preferences is not a Nash equilibrium under the TTC mechanism for some specification of cardinal utilities. This is because agents might misreport their preferences in order to be involved in a smaller cycle than in the case of truthtelling, lowering the probability of being affected by other agents' dropouts. This is a novel economic force revealed through our introduction of the dropout model.

To evaluate the welfare implications of dropouts, we next compare the k-greedy mechanisms with various values of k and the TTC mechanism by considering the limit as the number of agents tends to infinity, where preferences are generated uniformly at random in each market. We show that the fraction of agents who receive someone else's object tends to 0 under the TTC mechanism, while such a fraction tends to a positive number under the k-greedy mechanisms for any fixed k. This result analytically confirms the observation in Section 1.1 that the TTC mechanism is inefficient, while demonstrating the advantages of the k-greedy mechanisms.

Finally, we provide a wealth of simulation results that compare the performance of the TTC mechanism and the k-greedy mechanisms with various values of k. We find that the k-greedy mechanisms with small k mostly outperform the TTC mechanism when there are many agents.

Overall, the present paper sheds light on the reason why the TTC mechanism is not successfully used in practice despite its desirable properties, and proposes an alternative mechanism that addresses the weak point of the TTC mechanism. To circle back on the comparison in our opening paragraphs, we note that the deferred acceptance mechanism in the two-sided matching context, such as medical residency matching, does not suffer from the same problems: since doctors are not endowed with a hospital initially, even if a doctor drops out, that would not take away an assignment of some other doctor.[6] In contrast, in our exchange markets, agents who drop out can still exercise ownership over their endowed object. It is the combination of this ownership and the possibility of dropout that places the TTC mechanism at a disadvantage.

1.3 Related Literature

Notions similar to k-unanimity are shown to be incompatible with strategy-proofness in the literature in settings different from ours. Takagi and Serizawa (2010) consider a one-to-many two-sided matching model and show that there is no individually

[6]Moreover, even in a model that incorporates dropouts, it is straightforward to see that the deferred acceptance mechanism has the following desirable incentive property: If each doctor-hospital pair drops out with a positive probability that is independent of the hospital's identity, truthful preference submission by all doctors is a weakly dominant strategy equilibrium.

rational and strategy-proof mechanism under which two agents are matched if they regard each other as their respective first choice.[7] A special case of this result, when the market is one-to-one, can be considered to be a special case of our impossibility result with $k = 2$ where the preference domain is restricted in a "two-sided" manner.

Takamiya (2013) studies a coalitional formation model where there is a set of available coalitions and each agent submits a ranking over such coalitions. He shows that, under certain assumptions, there is no strategy-proof mechanism that satisfies the property that a given coalition must be formed if that coalition is ranked at the top of every agent in the coalition.[8] When the set of available coalitions have all coalitions with size k or less, his model might look similar to ours and hence one might think his result implies our Theorem 1 that shows that no k-robust mechanism is k-unanimous and strategy-proof. However, his model is different from ours whenever $k > 2$ because a coalition is merely a set of agents and does not specify who in the coalition gets which object.[9] Due to this difference, his result does not imply our Theorem 1 when $k > 2$.

Without constraints on the cycle size, the TTC mechanism selects the unique allocation in the core (Roth and Postlewaite, 1977) and is characterized by individual rationality, efficiency, and strategy-proofness (Ma, 1994). With the constraint on the cycle size, Gale and Shapley (1962) show that a stable matching may not exist in the context of a roommate problem.[10] Roth et al. (2005b) are motivated by kidney exchange problems and consider roommate problems (i.e., the cycle size must be 1 or 2) with dichotomous preferences. They propose a class of mechanisms that are individually rational, constrained-efficient and strategy-proof. Nicoló and Rodríguez-Álvarez (2012) is a seminal paper on general constraints on the cycle size, which shows the incompatibility between strategy-proofness and constrained efficiency in the model where agents cannot misreport their ranking over objects except that they can mis-

[7]The same notion is considered in Toda (2006) under a two-sided matching model and in Klaus (2011) in a roommate problem.

[8]Rodríguez-Álvarez (2009) studies the same requirement in a different context of analysis.

[9]For example, Takamiya (2013)'s model would not distinguish a cycle in which agent 1 receives object 2, agent 2 receives object 3, and agent 3 receives object 1 and another cycle in which agent 1 receives object 3, agent 2 receives object 1, and agent 3 receives object 2. This is fine for his purpose but can be an issue in our setting if, for example, agents 1 and 2 prefer the former cycle while agent 3 prefers the latter. In the special case of $k = 2$, our Theorem 1 is implied by Takamiya (2013) (his other assumptions are satisfied in our setting).

[10]Chung (2000) provides a general condition for the existence of a stable matching for roommate problems.

report the cutoff between acceptable and unacceptable objects.[11] Our result on the incompatibility between k-unanimity and strategy-proofness is not implied by their result. Indeed, the two concepts turn out to be compatible with each other in their environment.[12] Balbuzanov (2020) considers general preferences and proposes a random mechanism that respects constrained efficiency and strategy-proofness, among others. Our results differ from the ones in any of the above papers because our primary focus is on the k-greedy mechanisms and their performance in the model with dropouts. Moreover, we show that the k-greedy mechanism achieves not only k-efficiency but also k-unanimity, and that k-unanimity is incompatible with strategy-proofness.

We study a model where agents drop out of the market. Such a model is extensively analyzed in the computer science literature in the context of kidney exchange. See, e.g., Alvelos et al. (2015); Constantino et al. (2013); Dickerson et al. (2019); Feigenbaum and He (2024). The typical focus of those papers is to find algorithms that maximize specific objective functions, such as the number of transplants, where some papers further assume dichotomous preferences and/or pairwise exchanges. Recent economics papers, such as Ashlagi et al. (2018); Akbarpour et al. (2020), have also studied stochastic processes where agents may leave the market. However, the agents in their model would only leave before an assignment is made, while agents in our model would leave after an assignment is made. Overall, the results from the above literature on dropouts are not logically related to ours.

The paper proceeds as follows. Section 2 sets up a model without dropouts and introduces the concept of k-robustness. We provide impossibility results involving k-robustness and propose the k-greedy algorithm and k-greedy mechanism. Section 3 provides a model with dropouts and analyzes incentives and efficiency under the TTC mechanism and the k-greedy mechanism. Section 4 discusses simulation results, and Section 5 concludes. The Appendix provides a detailed description of the k-greedy algorithm and the proofs for all the results in this paper. The Online Appendix discusses an alternative version of the k-greedy algorithm and its relationship to the "k-robust version" of the core. It also provides additional simulation results and some more discussion on the k-greedy mechanism.

[11] See Nicolò and Rodríguez-Álvarez (2013), Nicolo and Rodriguez-Alvarez (2017) and Rodríguez-Álvarez (2023) for results on incompatibility between constrained efficiency and incentive compatibility under various settings.

[12] See footnote 21 for why this claim holds.

2 k-Robustness

2.1 Model

Let $I = \{1, 2, \dots, N\}$ be the set of agents. Each agent i possesses an indivisible object i, and thus I also denotes the set of objects. Each agent i has a strict preference relation $\succ_i$ over objects in I. We write $j \succeq_i j'$ if and only if $j \succ_i j'$ or $j = j'$. We say that object j is **acceptable** to agent i if $j \succeq_i i$. We sometimes let $\succ := (\succ_j)_{j \in I}$ to denote a preference profile.

An **exchange** is a mapping $\mu : I \to I$ that assigns to each agent an object, with a restriction that there is no pair of agents (i, j) with $i \neq j$ such that $\mu(i) = \mu(j)$. For notational simplicity, we denote $\mu(i)$ by μ_i. If $\mu_i = i$, we say that i is **unassigned** under μ. If $\mu_i \neq i$, we say that i **receives an object**. We say that an exchange μ is **individually rational** if $\mu_i \succeq_i i$ for every $i \in I$.

Given an exchange μ, a **cycle** is a sequence of agents, $(i_1, i_2, \dots, i_L)$ for some positive integer L, such that (i) $i_l \neq i_{l'}$ for any pair (l, l') with $l \neq l'$, and (ii) $\mu_{i_l} = i_{l+1}$ for each l where we set $i_{L+1} := i_1$. Given a sequence of agents $(i_1, \dots, i_L)$, we say we **implement** (or **form**) the cycle $(i_1, \dots, i_L)$ to mean that we assign object i_{l+1} to agent i_l where we let $i_{L+1} := i_1$. We say that a cycle in μ is a L**-cycle** if it involves exactly L agents.

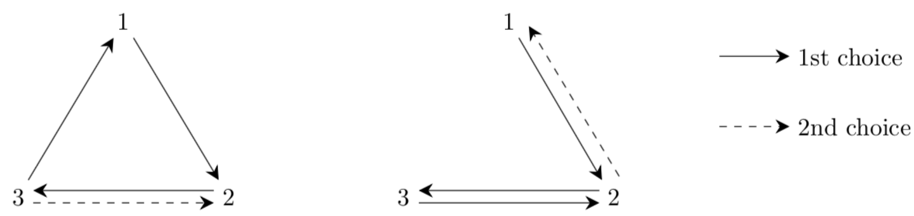
The following definition, which considers constraints on the maximum cycle size, provides a central concept of this paper.

Definition 1. An exchange μ is k**-robust** if there is no k'-cycle in μ such that $k' > k$.

That is, an exchange is said to be k-robust if all cycles have sizes k or fewer. Note that our model coincides with the standard house exchange problem if $k \geq N$, i.e., exchanges have no size restriction.

Definition 2. Given a preference profile $(\succ_j)_{j \in I}$, a **unanimous trading cycle** is a sequence $(i_1, \dots, i_L)$ such that (i) $i_l \neq i_{l'}$ for any pair (l, l') with $l \neq l'$, and (ii) $i_{l+1} \succeq_{i_l} i_{l'}$ for any pair (l, l') where we set $i_{L+1} := i_1$.

This is equivalent to a standard top trading cycle used in the well-known Top Trading Cycles (TTC) mechanism (which we formally define shortly) when all objects are in the market. We use a different terminology in this paper because we do not call a top trading cycle after some objects are eliminated a unanimous trading cycle,



(a) Preference profile $\succ$.

(b) Preference profile $\succ'$.

Figure 3: k-unanimity and k-efficiency. In panel (a), the exchange where everyone is unassigned is 2-unanimous but not 2-efficient. In panel (b), the exchange where agents 1 and 2 receive each other's object is not 2-unanimous but 2-efficient.

and this difference becomes important later.[13] Note that the notion of unanimous trading cycle is defined given a preference profile, while that of "cycle" is not.

An exchange μ is k**-unanimous** if, whenever $(i_1, \ldots, i_{k'})$ is a unanimous trading cycle with size $k' \leq k$, we have $\mu_{i_l} = i_{l+1}$ for each l where we set $i_{k'+1} := i_1$.[14] An exchange μ is called k**-efficient** if it is k-robust and there is no k-robust exchange μ' such that $\mu'_i \succeq \mu_i$ for every $i \in I$ and $\mu'_j \succ \mu_j$ for some $j \in I$. The following example illustrates that k-unanimity does not imply or is implied by k-efficiency.

Example 1 (k-unanimity and k-efficiency)**.** Suppose that $I = \{1, 2, 3\}$ and $k = 2$. Consider the following preferences.[15]

$$\succ_1: 2, 1; \qquad \succ_2: 3, 2; \qquad \succ_3: 1, 2, 3.$$

Panel (a) of Figure 3 provides a graphical representation of these preferences. Since there is no unanimous trading cycle with size 2 or fewer, the exchange where everyone is unassigned is 2-unanimous. However, such an exchange is not 2-efficient because an alternative exchange where agents 2 and 3 receive each other's object while agent 1 is unassigned is strictly better for agents 2 and 2 while making agent 1 indifferent.

Now, consider the following preferences.

$$\succ'_1: 2, 1; \qquad \succ'_2: 3, 1, 2; \qquad \succ'_3: 2, 3.$$

[13]See Remark 1 in Section 2.3.

[14]Note that no agent can be involved in multiple unanimous trading cycles under any preference profile.

[15]We list all acceptable objects (which include the endowed object) in order, while not listing those that are not acceptable.

Panel (b) of Figure 3 provides a graphical representation of these preferences. The exchange where agents 1 and 2 receive each other's object while agent 3 is unassigned is 2-efficient because any other 2-robust exchange would make either of agents 1 and 2 unassigned, making them worse off. However, such an exchange is not 2-unanimous because agents 2 and 3 constitute a unanimous trading cycle with size 2. □

The intuition for this example is simple: k-unanimity is not stronger because it only considers the first choices, while k-efficiency also pertains to worse choices. k-efficiency is not stronger because it can give priority to agents who are not anyone's first choice at the expense of others in a unanimous trading cycle with size k.

A mechanism is a function that assigns to each preference profile an exchange. Given a mechanism ψ and a preference profile $(\succ_j)_{j\in I}$, we denote the object that agent i receives by $\psi_i((\succ_j)_{j\in I})$. A mechanism is called **individually rational** if it always returns an individually rational exchange. We say that a mechanism ψ is **strategy-proof** if, for every preference profile $(\succ_j)_{j\in I}$ and every agent $i \in I$,

$$\psi_i((\succ_j)_{j\in I}) \succeq_i \psi_i(\succ'_i, (\succ_j)_{j\neq i})$$

holds for every $\succ'_i$.

Definition 3. A mechanism is

1. k**-robust** if it returns a k-robust exchange given any input;
2. k**-unanimous** if it returns a k-unanimous exchange given any input;
3. k**-efficient** if it returns a k-efficient exchange given any input.

Section 2.3 presents a mechanism that satisfies all the three properties in the above definition. Throughout this section, we take k-robustness as a hard requirement. k-unanimity is a normatively appealing property and can furthermore be seen as a minimal criterion that one would reasonably impose as a positive desideratum. In particular, k-unanimity is a weaker notion than being in the core.[16]

The **TTC mechanism** is a mechanism that outputs the exchange given by the following algorithm: Let $S_0 := I$. We start with Round 1. In Round l of the algorithm,

[16]Recall that, although in the context of two-sided matching, Roth (2002) provided justification for requiring stability as a positive desideratum, and that a matching is stable if and only if it is in the core.

each agent $i \in S_{l-1}$ points to the agent who owns the most desirable object for i among S_{l-1}. This generates a directed graph where each node has one outgoing arrow, and such a graph must have at least one cycle. Pick an arbitrary cycle and let the set of agents in the cycle be S''_{l-1}. Have each agent in the cycle receive the object that the agent points to and let $S_l = S_{l-1} \setminus S''_{l-1}$. The algorithm stops at the end of Round L when $S_L = \emptyset$ (such L exists because there must exist at least one cycle in each round).

We note that the TTC mechanism is individually rational, N-unanimous, N-efficient and strategy-proof, while it is not k-robust for any $k < N$.

2.2 Preliminary Results: Impossibility

We will propose in Section 2.3 a mechanism that is k-robust, k-unanimous, and k-efficient, but the mechanism is not strategy-proof. In fact, this section shows that strategy-proofness is incompatible with k-unanimity under k-robustness (Theorem 1). The example we use to explain our Theorem 1 will be useful in understanding some results that follow as well.

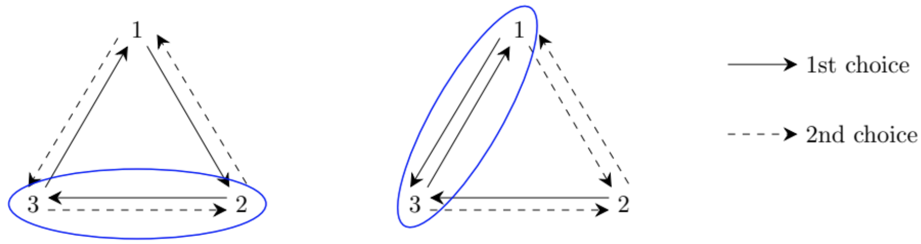
Theorem 1. *For any I and k such that $1 < k < |I|$, no k-robust mechanism is k-unanimous and strategy-proof.*

Note that the theorem rules out the cases of $k = 1$ and $k \geq |I|$. In those cases, a k-robust mechanism that is k-unanimous and strategy-proof trivially exists: if $k = 1$, then the mechanism that always returns the original exchange (i.e., everyone is unassigned) would do; If $k \geq |I|$, then the standard TTC mechanism would do.

To understand the idea of the proof for the (nontrivial) case of $1 < k < |I|$, consider an economy with three agents, $I = \{1, 2, 3\}$ and set $k = 2$. Consider the following preferences:

$$\succ_1: 2, 3, 1; \qquad \succ_2: 3, 1, 2; \qquad \succ_3: 1, 2, 3. \tag{1}$$

See Figure 4 for a graphical representation. Consider any 2-robust mechanism ψ that is 2-unanimous. Since ψ is 2-robust, it must make at least one agent unassigned. By symmetry, without loss of generality, let agent 1 be unassigned. Then, consider



(a) Exchange under $\succ$.

(b) Exchange under $(\succ_1', \succ_2, \succ_3)$.

Figure 4: Three-agent example for Theorem 1. If agent 1 is unassigned (e.g., the exchange $(2, 3)$ realizes) under $\succ$ as in panel (a), agent 1 has an incentive to misreport her preferences to be assigned object 3 as in panel (b).

the following misreporting by agent 1:

$$\succ_1' : 3, 2, 1. \tag{2}$$

Given the preference profile $(\succ_1', \succ_2, \succ_3)$, ψ must assign object 3 to agent 1 because it is 2-unanimous and agents 1 and 3 deem each other as the first choice. But then, this ψ is not strategy-proof because agent 1 is better off by misreporting $\succ_1'$ when her preferences are $\succ_1$.[17]

The idea of this example is that if the outcome of a k-unanimous mechanism is not desirable for an agent (in our example, agent 1 is unassigned, which is not desirable for her), she can misreport her preferences to designate an alternative object to be the first choice, thereby creating a unanimous trading cycle which, by k-unanimity, the mechanism has to implement. The proof of Theorem 1 in the Appendix generalizes this idea to the case with any k and any number of agents.[18] We note that the same idea is also used in proving strategic problems with the TTC mechanism in the dropout model (Theorem 4).

We note that the statement of the theorem does not require individual rationality.

[17]The setup of this example is essentially the same as one proposed by Gale and Shapley (1962) in the context of a roommate problem, where they show that no stable matching exists. Here we use a similar argument to show that 2-unanimity is incompatible with strategy-proofness, and our Theorem 1 extends such an argument for any cycle size k.

[18]A notable feature of this example is the cyclical nature of preferences. This raises a question of whether the impossibility still holds when we restrict attention to a narrower class of preferences that excludes some cyclicality. In Kamada and Yasuda (2025), we consider single-peaked preferences and show that the impossibility still holds, using a more contrived construction of preferences.

The reason is that k-unanimity and strategy-proofness imply individual rationality.[19] This is because one can always claim her own object as her first choice, by which she can form a unanimous trading cycle of size 1.[20]

Theorem 1 implies that one has to make a choice between k-unanimity and strategy-proofness under k-robustness. Our k-robust mechanism in Section 2.3 is k-unanimous, and thus, it is not strategy-proof. The primary reason why we made this choice is that, as we discussed, k-unanimity is a property that has normative and positive appeals. Another reason is that strategy-proofness severely restricts the scope of possible mechanisms. In fact, a straightforward corollary of Nicoló and Rodríguez-Álvarez (2012) is that strategy-proofness is incompatible with k-efficiency as well under k-robustness and individual rationality, as we formally state below.[21]

Proposition 1 (Nicoló and Rodríguez-Álvarez (2012))**.** *For any I and k such that $1 < k < |I|$, no k-robust mechanism is individually rational, k-efficient and strategy-proof.*[22]

To summarize our preliminary analysis, we motivated the mechanism in the next section by showing that k-unanimity and strategy-proofness are incompatible with each other under k-robustness. Since k-unanimity seems to be a minimal desideratum and strategy-proofness imposes quite a bit of restriction (Proposition 1), we will drop strategy-proofness in search of a better mechanism. We will, however, discuss (good) incentive properties of our mechanism as well (Proposition 2).

2.3 k-Greedy Algorithm

We propose a class of mechanisms, *k-greedy mechanisms*, that satisfy k-unanimity and k-efficiency. By Theorem 1 (or Proposition 1), those mechanisms are not strategy-

[19] This result is formally stated in Lemma 1 in the Appendix.

[20] In the next result (Proposition 1), we switch k-unanimity with k-efficiency, and thus we will need to impose individual rationality separately.

[21] Nicoló and Rodríguez-Álvarez (2012) consider a setting where agents cannot misreport their ranking over objects except that they can misreport the cutoff between acceptable and unacceptable objects. We note that the conclusion of our Theorem 1 does not hold in such an environment. Indeed, if we consider a mechanism based on only running Round 0 of the k-greedy algorithm that we present later, then such a mechanism is k-robust, k-unanimous, and strategy-proof in their environment.

[22] As in the case of our Theorem 1, the proof in Nicoló and Rodríguez-Álvarez (2012) relies on a cyclical nature of preferences. In Kamada and Yasuda (2025), we consider single-peaked preferences and show that the impossibility still holds when $k \geq 3$ using a more contrived construction of preferences, but we also show that there is a 2-robust mechanism that is individually rational, 2-efficient and strategy-proof under single-peaked preferences.

proof. Each mechanism in this class is based on an algorithm called a *k-greedy algorithm*. This section defines this algorithm.

In order to define a k-greedy algorithm, we fix an ordering of agents, $\sigma : I \to I$, which is one-to-one and onto. The interpretation is that agent i's position in the ordering is $\sigma(i)$. In what follows, unless explicitly noted, we consider σ such that $\sigma(i) = i$ for every $i \in I$.

Given a set of agents S, their preferences $(\succ_j)_{j \in S}$ over S, and an agent $i \in S$, we first define the *(i,k)-serial dictatorship algorithm on S* to be an algorithm that, intuitively speaking, assigns to agent i as desirable an object as possible with a constraint that the exchange has to be done in a k-robust and individually rational manner within the set S. Call the object agent i receives in this case object i'. If there are multiple ways for agent i to receive object i', we consider giving as desirable an object as possible to agent i' under the constraint that agent i receives i' and we satisfy k-robustness and individual rationality. Note that agent i' is not necessarily agent $i+1$ as in the standard serial dictatorship.[23] Call the object agent i' receives here object i''. If there are still multiple ways to give object i'' to agent i', we assign as desirable an object as possible to agent i'' under the constraint that agent i receives object i', agent i' receives object i'', and we satisfy k-robustness and individual rationality, and so on. We continue this process, which forms a unique cycle. We provide a formal description of this algorithm in the Appendix.

We use the (i,k)-serial dictatorship algorithm on S to define the k-greedy algorithm. The latter algorithm starts with Round 0.

<u>k**-Greedy Algorithm**</u>:

<u>Round 0:</u> Define μ^0 as follows: If agent j is involved in a unanimous trading cycle, then μ^0_j is set to be agent j's first-choice object according to $\succ_j$. Otherwise, $\mu^0_j = j$. Define $S^0 := \{j \in I | \mu^0_j = j\}$. Go to Round 1.

<u>Round 1:</u> If $|I| = 1$, end the algorithm and output μ^0. Otherwise, if agent 1 is not in S^0, let $\mu^1 := \mu^0$ and $S^1 = S^0$, and go to Round 2. Otherwise, run the $(1,k)$-serial dictatorship algorithm on S^0. Let the outcome be $\tilde{\mu}^1$. Define μ^1 as follows: $\mu^1_j = \mu^0_j$ if $j \in I \setminus S^0$ and $\mu^1_j = \tilde{\mu}^1_j$ if $j \in S^0$. Define $S^1 := \{j \in S^0 | \tilde{\mu}^1_j = j\} \setminus \{1\}$. Go to Round 2.

<u>Round i ($i \geq 2$):</u> If $i = |I|$, end the algorithm and output $\mu^{|I|-1}$. Otherwise, if agent

[23]See footnote 24 for a more detailed comparison with the standard serial dictatorship.

i is not in S^{i-1}, let $\mu^i := \mu^{i-1}$ and $S^i = S^{i-1}$, and go to Round $i+1$. Otherwise, run the (i,k)-serial dictatorship algorithm on S^{i-1}. Let the outcome be $\tilde{\mu}^i$. Define μ^i as follows: $\mu^i_j = \mu^{i-1}_j$ if $j \in I \setminus S^{i-1}$ and $\mu^i_j = \tilde{\mu}^i_j$ if $j \in S^{i-1}$. Define $S^i := \{j \in S^{i-1} | \tilde{\mu}^i_j = j\} \setminus \{i\}$. Go to Round $i+1$.

This algorithm ends at Round $|I|$ and outputs a well-defined exchange.

The idea is simple: In Round 0, we implement all unanimous trading cycles with size k or fewer so that the resulting exchange satisfies k-unanimity. Then, in Round 1, we run the $(1,k)$-serial dictatorship, that is, agent 1 becomes the dictator first if she was not assigned in Round 0, and otherwise we immediately move to Round 2. Then, in Round 2, agent 2 becomes a dictator if he was not assigned in Round 0 or Round 1, and otherwise we immediately move to Round 3, where we consider agent 3, and so on.

The algorithm runs reasonably quickly, making it useful in practice.

Theorem 2. *The k-greedy algorithm terminates in polynomial time.*

The key step in the proof is to note that there are at most N^{k-1} different cycles that involve agent i and satisfy k-robustness. Since the algorithm essentially conducts a version of serial dictatorship in this set of cycles, the algorithm terminates in polynomial time.[24]

Remark 1. Round 0 of the k-greedy algorithm only runs unanimous trading cycles with size k or fewer. The Online Appendix analyzes an alternative algorithm, called the modified k-greedy algorithm, that replaces this round with iterative steps in which, after all unanimous trading cycles with size k or fewer are eliminated, the top trading cycles with size k or fewer are implemented until there are no such cycles. It turns out that the set of exchanges induced by the modified k-greedy algorithm may be disjoint with what we call the k-robust core, which is the set of exchanges that are robust to coalitions of size k or fewer, even when the k-robust core is nonempty. We

[24] Such a proof method is not applicable if we simply apply serial dictatorship under the constraint of k-robustness (that is, we first consider the set of exchanges where agent 1 receives as desirable an object as possible among all k-robust exchanges, then consider the set of exchanges where agent 2 receives as desirable an object as possible among the remaining exchanges, and so forth). This is because, for example, there may exist multiple cycles in which agent 1 receives the most desirable object among the set of k-robust exchanges, where agent 2 is involved in some of them but not others, and whether agent 2 should be in such a cycle may depend on how that affects agent 3's assignment.

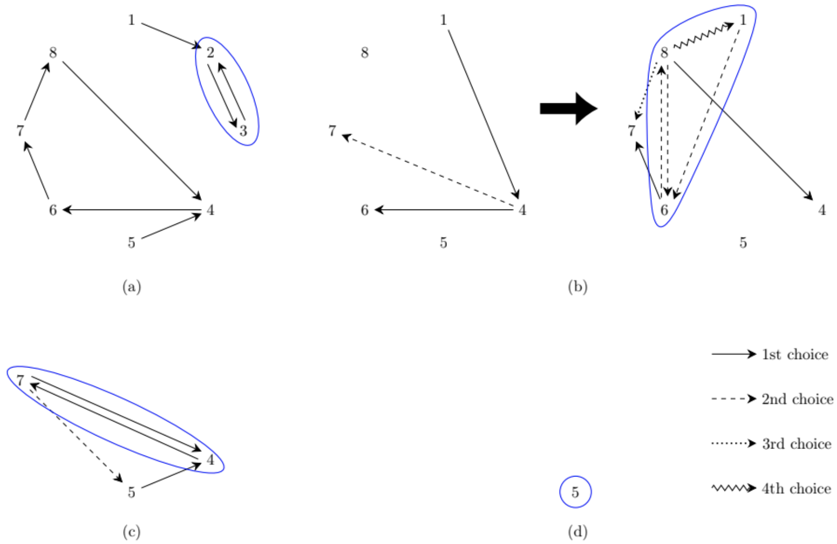


Figure 5: Example 2. Each panel only depicts relevant parts of the preference profile $\succ$. Panel (a): Round 0 where we consider all agents' first choices and find cycles with size k or fewer. Panel (b): Round 1 where we give as desirable an object as possible to agent 1. Panel (c): Round 4 where we give as desirable an object as possible to agent 4. Panel (d): Round 5 where we give as desirable an object as possible to agent 5. Rounds 2, 3, 6, and 7 immediately end because the corresponding agents are already assigned before the respective rounds.

also show that if the k-greedy algorithm induces an exchange in the k-robust core, then the modified k-greedy algorithm must induce the same exchange.

Let us illustrate the workings of the k-greedy algorithm using an example.

Example 2 (k-greedy algorithm)**.** Consider an economy with eight agents, $I = \{1, 2, \ldots, 8\}$ and set $k = 3$. Consider the following preferences:

$\succ_1$: 2, 4, 6, 7, 1;	$\succ_2$: 3, 2;	$\succ_3$: 2, 3;	$\succ_4$: 6, 7, 4;
$\succ_5$: 4, 5;	$\succ_6$: 7, 8, 6;	$\succ_7$: 8, 4, 5, 7;	$\succ_8$: 4, 6, 7, 1, 8.

See Figure 5 for a graphical representation.

<u>Round 0:</u> We first find unanimous trading cycles. By inspection, we can see that there are two unanimous trading cycles, which are $(2, 3)$ and $(4, 6, 7, 8)$. Since the cycle sizes are 2 and 4, respectively, only the cycle $(2, 3)$ is implemented because $k = 3$. This means that $\mu_2^0 = 3$, $\mu_3^0 = 2$, and $\mu_j^0 = j$ for all agents in $S^0 = I \setminus \{2, 3\} =$

$\{1,4,5,6,7,8\}$.

Round 1: We run the $(1,k)$-greedy algorithm on S^0. For this, we aim to find a way to give as desirable an object as possible to agent 1. Since her first-choice object, which is object 2, is already taken in Round 0, we consider her second choice, namely object 4. There is no way to assign this object to agent 1 by forming a cycle of size 3 or fewer in an individually rational manner (the cycles $(1,4,6,7,8)$ and $(1,4,6,8)$ would satisfy individual rationality, but their cycle sizes exceed k). Hence, we consider agent 1's third choice, which is object 6. There is exactly one way to assign this object to agent 1, which is to form a cycle $(1,6,8)$ (the cycle $(1,6,7,8)$ would satisfy individual rationality, but its cycle size exceeds k). Hence, we implement this cycle. This gives us:

$$(\tilde{\mu}_1^1, \tilde{\mu}_4^1, \tilde{\mu}_5^1, \tilde{\mu}_6^1, \tilde{\mu}_7^1, \tilde{\mu}_8^1) = (6,4,5,8,7,1),$$

and thus,

$$(\mu_1^1, \mu_2^1, \mu_3^1, \mu_4^1, \mu_5^1, \mu_6^1, \mu_7^1, \mu_8^1) = (6,3,2,4,5,8,7,1)$$

and $S^1 = S^0 \setminus \{1,6,8\} = \{4,5,7\}$. Note that there is one more cycle in which agent 1 receives an object while satisfying individual rationality and k-robustness, which is $(1,7,8)$. This cycle gives agent 1 her fourth choice while giving agent 7 her first choice object, 8. In contrast, the $(1,k)$-greedy algorithm chooses $(1,6,8)$, which gives agent 1 her third choice while eliminating the possibility for agent 7 to receive her first-choice object 8. Since the $(1,k)$-greedy algorithm gives priority to agent 1, the algorithm chooses the cycle $(1,6,8)$ over $(1,7,8)$.

Rounds 2 and 3: Since agent 2 is not in S^1, we let $\mu^2 = \mu^1$ and $S^2 = S^1 = \{4,5,7\}$, and this ends Round 2. Since agent 3 is not in S^2, we let $\mu^3 = \mu^2$ and $S^3 = S^2 = \{4,5,7\}$, and this ends Round 3.

Round 4: We run the $(4,k)$-greedy algorithm on S^3. For this, we aim to find a way to give as desirable an object as possible to agent 4. Since her first-choice object, which is object 6, is already taken in a previous round, we consider object 7, which is her second choice. There are two ways to assign this object to agent 4, which are to form a cycle $(4,7)$ or $(4,7,5)$. Note that both cycles give agent 4 object 7, and hence we look at agent 7's preferences. Since agent 7 prefers receiving object 4 to receiving object 5, we implement the cycle $(4,7)$. As will be clear, this will imply that in the subsequent steps, agent 5 will be unassigned. Thus, in a sense, this algorithm gives priority to agent 7 instead of agent 5, although agent 7 is ordered later than agent 5 according to σ (which specifies $\sigma(i) = i$ for all i).[25] This gives us:

$$(\tilde{\mu}_4^3, \tilde{\mu}_5^3, \tilde{\mu}_7^3) = (7, 5, 4),$$

and thus,

$$(\mu_1^3, \mu_2^3, \mu_3^3, \mu_4^3, \mu_5^3, \mu_6^3, \mu_7^3, \mu_8^3) = (6, 3, 2, 7, 5, 8, 4, 1)$$

and $S^4 = S^3 \setminus \{4, 7\} = \{5\}$.

Round 5: We run the $(5, k)$-greedy algorithm on S^4. For this, we aim to find a way to give as desirable an object as possible to agent 5. Since all choices but her own object are already taken in the previous rounds, we can only form a cycle of size 1, which involves agent 5 alone. Hence, we implement this cycle, which gives us $\tilde{\mu}_5^5 = 5$, and thus,

$$(\mu_1^5, \mu_2^5, \mu_3^5, \mu_4^5, \mu_5^5, \mu_6^5, \mu_7^5, \mu_8^5) = (6, 3, 2, 7, 5, 8, 4, 1)$$

and $S^6 = S^5 \setminus \{5\} = \emptyset$.

Rounds 6 and 7: Since agent 6 is not in S^5, we let $\mu^6 = \mu^5$ and $S^6 = S^5 = \emptyset$, and this ends Round 6. Similarly, since agent 7 is not in S^6, we let $\mu^7 = \mu^6$ and $S^7 = S^6 = \emptyset$, and this ends Round 7.

Round 8: Since $8 = |I|$, we end the algorithm and return μ^7.

Thus, the output of the k-greedy algorithm, denoted $\hat{\mu}$ is

$$(\hat{\mu}_1, \hat{\mu}_2, \hat{\mu}_3, \hat{\mu}_4, \hat{\mu}_5, \hat{\mu}_6, \hat{\mu}_7, \hat{\mu}_8) = (6, 3, 2, 7, 5, 8, 4, 1). \qquad \square$$

Define the **k-greedy mechanism** to be the mechanism that outputs the outcome of the k-greedy algorithm for any input. When we make the dependence of the associated k-greedy algorithm on the ordering σ explicit, we call the mechanism the **k-greedy mechanism with** σ.

This mechanism has desirable properties, as stated in the following theorem:

Theorem 3. *The k-greedy mechanism is a k-robust mechanism, and it is individually rational, k-unanimous, and k-efficient.*

Theorem 1 and Theorem 3 (or Proposition 1 and Theorem 3) together imply that a k-greedy mechanism with given σ is not strategy-proof. This may question the applicability of a k-greedy mechanism in practice. The next result, however, suggests

[25] Cf. footnote 24.

that there is no "obvious way" in which agents can improve their assignment under a k-greedy mechanism.

Say that preference relation $\succ_i'$ is **equivalent** to $\succ_i$ if, for any $j, j' \in I$, we have $j \succ_i' j' \succ_i' i$ if and only if $j \succ_i j' \succ_i i$. In other words, the two preferences agree with each other for the set of acceptable objects and the rankings among them.

Proposition 2. *Fix any I, k, and $\succ_i$.*

1. *There is no $\succ_i'$ such that for any ordering σ, there is $\succ_{-i}$ such that we have $\psi_i(\succ_i', \succ_{-i}) \succ_i \psi_i(\succ)$ where ψ is the k-greedy mechanism with σ.*

2. *There is no $\succ_i'$ such that for any $\succ_{-i}$, there is ordering σ such that we have $\psi_i(\succ_i', \succ_{-i}) \succ_i \psi_i(\succ)$ where ψ is the k-greedy mechanism with σ.*

3. *If $k \geq 3$, then there is no $\succ_i'$ that is not equivalent to $\succ_i$ such that for any $\succ_{-i}$, there is ordering σ such that we have $\psi_i(\succ_i', \succ_{-i}) \succeq_i \psi_i(\succ)$ where ψ is the k-greedy mechanism with σ.*

The first part is motivated by the flexibility in choosing σ. In practice, the designer may want to randomize over possible σ's. The result shows that there is no misreporting that "works" for all possible orderings: agent i cannot find a misreporting that, for any possible orderings, makes her strictly better off under some preference profile of other agents.

The second and the third parts show that there is no misreporting that "works" for all possible preference profiles of the other agents. More specifically, the second part shows that agent i cannot find a misreporting that, under any possible preference profile of others, makes her strictly better off for some ordering.

The third part is similar to the second part but shows a stronger result for the case when $k \geq 3$ by replacing strict preferences with weak preferences[26]: agent i cannot find a misreporting with a different ranking over acceptable objects that, under any possible preference profile of others, makes her weakly better off for some ordering. That is, for any misreporting that changes the ranking of acceptable objects, there must exist a preference profile of others that makes agent i strictly worse off for any

[26] An analogue of this strengthening for part 1 would not work: If $\succ_{-i}$ is such that all $j \neq i$ regards object j as the first choice, then $\succ_i'$ would always do as good as $\succ_i$ under any σ because agent i receives object i anyway. In the Online Appendix, we provide an example to show that even a weaker claim does not hold either, where the strict preferences are replaced with weak preferences with at least one strict relation.

ordering. Note that this conclusion does not generally hold for $k = 2$. This is because, if $\sigma(i) = |I|$, i.e., agent i is at the end of the ordering, then agent i's ranking over the objects whose owners are ordered before i according to σ does not affect the object that agent i receives, as long as they are not in the first place (so they are irrelevant in Round 0) and are deemed acceptable under $\succ'_i$.[27]

3 Dropouts

Now we consider a model in which, after a given mechanism Γ is implemented, each agent drops out of the market with probability p independently. We call such an agent a *dropped agent*, and the dropped agent receives her own object. If agent i is in a cycle in which there is at least one dropped agent, then agent i receives her own object i.[28] We call such agent i (who is not a dropped agent) an *affected agent*. Other agents receive the object specified by the outcome of mechanism Γ. If agent i receives object j, then her payoff is u_{ij}, where, for each $j, j' \in I$, we have $u_{ij} > u_{ij'}$ if and only if $j \succ_i j'$. As in the previous section, we assume strict preferences, and hence, we have $u_{ij} \neq u_{ij'}$ for all $j, j' \in I$ with $j \neq j'$. This market is characterized by $(N, \Gamma, p, (u_{ij})_{1 \leq i,j \leq N})$.

3.1 Preference Submission Game

We first view a market $(N, \Gamma, p, (u_{ij})_{1 \leq i,j \leq N})$ as a game in which agents simultaneously submit strict ordinal preferences to the mechanism and dropouts occur as specified above. Say that it is a **preference submission game** $(N, \Gamma, p, (u_{ij})_{1 \leq i,j \leq N})$.

Since the preference submission games involve probabilities of dropout, agents' best response depends, in general, not only on their ordinal preferences but also on cardinal preferences. The next theorem shows that there are cardinal preferences such that agents have incentives to misreport their ordinal preferences in the pref-

[27] More specifically, suppose that $\sigma(i) = |I|$. In such a case, any $\succ'_i$ that agrees with $\succ_i$ in (i) the first-choice object and (ii) the set of other acceptable objects would result in $\psi_i(\succ'_i, \succ_{-i}) = \psi_i(\succ)$ for any $\succ_{-i}$. If $\succ_i$ deems three or more objects as acceptable besides object i, then $\succ'_i$ can be different from $\succ_i$.

[28] One could imagine an alternative model in which the agents who are affected by a dropped agent would participate in a certain rewiring process among themselves. While we do not primarily consider such a scenario to focus on the first-order effect of dropouts, we briefly discuss this issue in Section 4.3.

erence submission game under the TTC mechanism. That is, the TTC mechanism, which is strategy-proof in the standard environment, may also suffer from strategic manipulation.

Theorem 4. *Suppose that* Γ *is the TTC mechanism. For any* (N, p) *such that* $N \geq 3$ *and* $p > 0$*, there exists* $(u_{ij})_{1 \leq i,j \leq N}$ *such that truthful preference submission by all agents is not a Nash equilibrium of the preference submission game* $(N, \Gamma, p, (u_{ij})_{1 \leq i,j \leq N})$.

To prove this result, we consider the preference profile (1) in Section 2.2 (depicted in Figure 4). This example features a unanimous trading cycle of size 3. But if an agent's second-choice object is almost as good as her first-choice object, then she could misreport that her true second choice was her first choice, by which she would form a unanimous trading cycle of size 2. By this misreporting, the agent could reduce the risk of someone in the cycle dropping while receiving an object that is almost as good as her first choice, so this misreporting would improve her payoff. This is a novel economic force revealed through our introduction of the dropout model.

Note that the intuition is similar to that of Theorem 1. There, forming a 3-cycle was prohibited, so the agent who would be unassigned had an incentive to misreport that her true second choice was her first choice. This enabled her to form a 2-cycle. In Theorem 4, forming a 3-cycle is not prohibited but is risky, and an agent was able to initiate the same misreporting to form a 2-cycle to improve her payoff.

The result can be generalized as follows, which reinforces the appeal of the k-greedy mechanisms with small values of k.

Proposition 3. *Fix* $N \geq 2$ *and* $l \in \{2, \ldots, N\}$*. For each* $k \leq N$*, fix an ordering* σ *and the* k*-greedy mechanism with* σ*, denoted* Γ^k*. For any* $p > 0$*, there exists* $(u_{ij})_{1 \leq i,j \leq N}$ *such that truthful preference submission by all agents is a Nash equilibrium of the preference submission game* $(N, \Gamma^k, p, (u_{ij})_{1 \leq i,j \leq N})$ *for any* $k \leq l$*, but it is not a Nash equilibrium of* $(N, \Gamma^k, p, (u_{ij})_{1 \leq i,j \leq N})$ *for any* $k > l$*.*

The result demonstrates a sense in which truthtelling becomes more and more difficult to incentivize as k increases. Under such an interpretation of the result, k-greedy mechanisms (especially with small k) can be said to have a good incentive property compared with the TTC mechanism because the largest k corresponds to the case of the TTC mechanism.

To prove this result for a given l, we consider a utility function such that the utilities from the first $N - l + 1$ choices are high and close to each other, while the rest of the utilities decrease exponentially fast. With such a utility function, attempting to form a cycle smaller than size k would not pay off if $k \leq l$ while it does if $k > l$ (forming a larger cycle is infeasible under the k-greedy mechanisms).

3.2 Top Trading Cycle Algorithm vs. k-Greedy Algorithm

In practice, the constraint of k-robustness may be hard or soft. When it is a hard constraint and $k < |I|$, one cannot use the TTC mechanism because it is not k-robust. When the constraint is a soft one, one could still use the TTC mechanism instead of k-greedy mechanisms, but doing so has benefits and costs. On the one hand, in an ideal situation, no agents would drop out of the mechanism and the TTC mechanism achieves efficiency, while the k-greedy mechanisms typically lead to a welfare loss due to their limited cycle sizes. On the other hand, if more agents drop out, then the TTC mechanism can lead to a significant loss in welfare because the entire group of agents in a cycle would be affected by just one agent dropping out of the cycle.

We use theory and simulations to evaluate this trade-off. In this subsection, we show limit results as the number of agents goes to infinity: the fraction of assigned agents tends to 0 under the TTC mechanism, while it is bounded away from 0 under k-greedy mechanisms with $k \geq 2$. Then, in Section 4, we discuss our simulation results which demonstrate that k-greedy mechanisms outperform the TTC mechanism. We assume truthful reporting throughout these analyses.

Fix $p > 0$ and consider a sequence of random markets, parameterized by the number of agents, N. For each N, each agent's ranking over objects in I is drawn uniformly over all possible rankings, independently across agents. In such a market, let $F(N; \Gamma, p)$ be the expectation of the fraction of assigned agents under mechanism Γ and the dropout probability p, assuming the truthful reporting of the preferences.

Theorem 5. *Fix $p > 0$. The following two limit results hold.*

1. $\lim_{N\to\infty} F(N; \Gamma, p) = 0$ *if Γ is the TTC mechanism.*

2. $\liminf_{N\to\infty} F(N; \Gamma, p) > 0$ *if Γ is a k-greedy mechanism with $k \geq 2$.*

That is, when there are many agents, almost all agents receive their own object under the TTC mechanism, while a positive fraction of agents can participate in an exchange with other agents under the k-greedy mechanisms. The theorem illustrates the appeal of the k-greedy mechanisms over the TTC mechanism.

The intuition for the results is the following. Under the TTC mechanism, one can prove that almost all agents are involved in a large cycle when there are many agents, and it is likely that at least one agent in each of such cycles drops out.[29] This results in almost all agents ending up receiving their own goods. To prove that almost all agents are involved in a large cycle, we invoke the result of Frieze and Pittel (1995) that provides a bound on the number of cycles for large random exchange markets. For k-greedy mechanisms, if the vanishing fraction of agents are assigned, we show that it would contradict the k-efficiency of k-greedy mechanisms in Theorem 3. More specifically, if most agents become unassigned, it is unlikely that we cannot find any mutually acceptable pairs in the pool of unassigned agents, given that the preferences are uniformly random. However, k-efficiency of k-greedy mechanisms implies that, under the exchange produced by any k-greedy mechanisms, there are no pairs of agents among those who are unassigned such that they find each other acceptable, and this is a contradiction.

4 Simulations

4.1 Simulation Overview

To further compare the performances of the k-greedy mechanisms and the TTC mechanism, we ran simulations. Specifically, we varied the mechanism and the parameters (N, p), and each specification is iterated 100 times. In our baseline market, u_{ij} is independently drawn according to the normal distribution $N(0, 1)$.

Before going into the details, let us summarize the main takeaways from the simulations. A fuller account is provided in Section 4.2.

1. Even when (in fact, especially when) N is large, the k-greedy mechanism with a small k (such as $k = 2, 3, 4, 5$) would be sufficient to outperform the TTC mechanism.

2. Compared to the TTC mechanism, the outcome of the k-greedy mechanism tends to have a relatively smaller number of agents who are assigned their very

[29] We note that the use of the terminology "large cycle" is different from the one in the standard network theory, where it would mean a cycle that encompasses a positive fraction of all agents. Instead, our claim here is that, for any fixed (possibly very large) number K, the expected fraction of agents who are in a cycle of size at least K tends to 1 as N goes to infinity.

good and very bad choices, while relatively more agents are assigned moderate choices.

3. Increasing p or N makes the TTC mechanism perform relatively worse.

These conclusions are obtained in our rather special specification with independent and symmetric preferences. This setting is generalized in various directions in the Online Appendix, and we find that the main findings above are robust to those generalizations. Moreover, we uncover additional insights from each of such modified settings. We briefly review those insights in Section 4.3.

The overall message of this section is that the use of the TTC mechanism may be practically problematic. However, the TTC mechanism was used in some school-choice environments. In Section 4.4, we discuss why the TTC mechanism could be used in such a context.

4.2 Details of the Results

Let us now be more detailed about our simulation results. Figure 6 demonstrates the performance of each mechanism. In the graphs, each curve represents a mechanism. The horizontal axis represents the rank of the object received by agents, and the vertical axis shows the number of students who are assigned an object ranked at the k-th place or better. The left panel is for the case with $N = 1000$ and $p = 0.1$, and the right panel is for $N = 200$ and $p = 0.05$. Each curve represents the average of relevant values for 100 iterations with a 95% confidence interval.[30] For example, the left panel shows that, on average, 447 students are assigned the 51st or better objects under the 2-greedy mechanism. By definition, one can interpret a mechanism to be "better performing" than another if the former has a curve above the one for the latter. The graphs in Figure 6 show that the k-greedy mechanisms perform mostly well relative to the TTC mechanism. In what follows, we explain our findings in more detail.

1. When N is large, the possibility of large cycles under the TTC mechanism has two effects on the relative performance of k-greedy mechanisms. On the one hand, the gain from not restricting to cycles with size only k or fewer is large. On the other hand, the risk of dropouts becomes large. We know from

[30] Some confidence intervals are so narrow and hard to see. See item 5 below on this point.

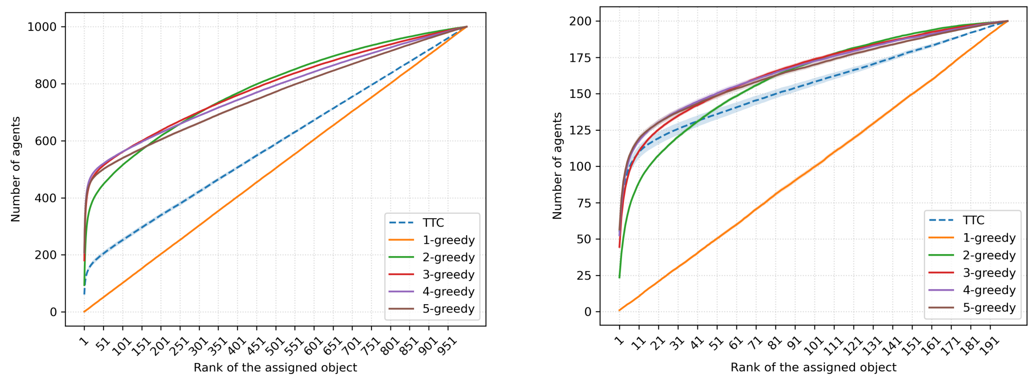


Figure 6: The performance of each mechanism under the TTC mechanism and the k-greedy mechanisms for $k = 2, 3, 4, 5$. We set $N = 1000$ and $p = 0.1$ in the left panel and $N = 200$ and $p = 0.05$ in the right panel.

Theorem 5 that the second effect will become larger and larger as N tends to infinity. Moreover, this effect is magnified due to the fact that larger cycles are typically formed at earlier steps of the algorithm for the TTC mechanism, where better-ranked matches occur. The first effect is especially large when k is small. In the left panel of Figure 6 where we set $N = 1000$ and $p = 0.1$, the second effect dominates even when k is as small as 2-5. We also find that, at larger values of k,the performance of the k-greedy mechanisms become worse.[31]

2. Although the k-greedy mechanisms first-order stochastically dominate the TTC mechanism in the left panel of Figure 6, it is not a general pattern. For smaller N and smaller p, the TTC mechanism tends to assign either very desirable objects (when there are no dropouts, which is not too unlikely given small N and p) or very undesirable objects (when there are dropouts) to the agents, as shown in the right panel of Figure 6. In contrast, under the k-greedy mechanisms, the agents do not receive very desirable objects due to the cycle size constraint, but the dropout probability is small. For this reason, agents are typically assigned moderately desirable objects. Similarly, there is no clear dominance relationship among k-greedy mechanisms with small values of k ($k = 2, 3, 4, 5$).

3. What is the effect of increasing the dropout probability? The left panel of Figure 7 illustrates the effect of varying p for a fixed N, where we plot the

[31] For example, when $N = 200$ and $p = 0.05$, the performance of the 10-greedy mechanism is similar to that of the TTC mechanism.

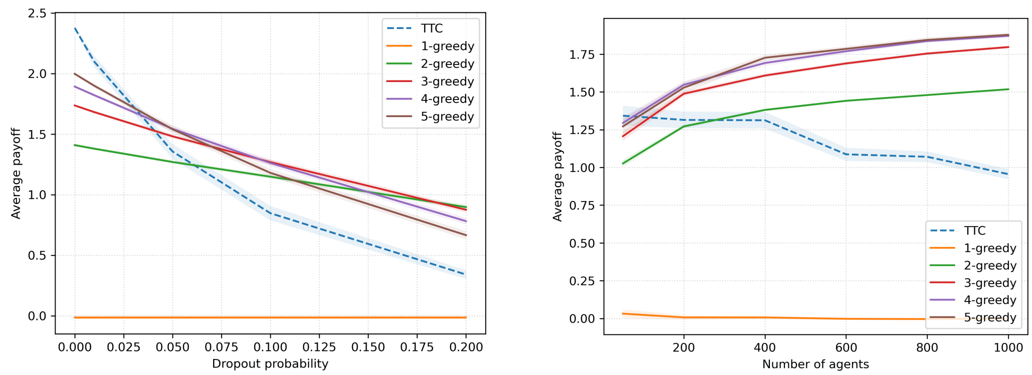


Figure 7: The average payoff under each mechanism under the TTC mechanism and the k-greedy mechanisms for $k = 2, 3, 4, 5$. The left panel varies the dropout probability p under $N = 200$, while the right panel varies the number of agents N under $p = 0.05$.

agents' average payoff. It shows that the performance of the TTC mechanism becomes relatively worse as p increases: the average payoff decreases in p for both mechanisms, while the reduction is drastic for the TTC mechanism and it is moderate for the k-greedy mechanisms. To see why this disparity occurs, note that the probability that an agent in a cycle with size l drops out or is affected is $1 - (1 - p)^l$. The derivative of this probability with respect to p is $-l(1 - p)^{l-1} \simeq -l(1 - (l - 1)p)$ for small p. For small p, the absolute value of this slope is large when l is large. Since there is a small bound on the cycle size under the k-greedy mechanism with small k while there is no such bound under the TTC mechanism, we obtain the relationship as in the graph. Note also that the simulation result shows that the effect of the difference in the cycle sizes is so significant that the k-greedy mechanisms mostly outperform the TTC mechanism even when p is as small as $p = 0.05$ (for small $k \geq 3$).

4. Next, consider the effect of increasing N. There are two effects associated with this change. First, the choice set expands, which implies that for any given agent and a given cycle size, it becomes easy to form a cycle with that size where the agent receives an object with a higher payoff. Second, for the TTC mechanism, the expected cycle size becomes larger, which can negatively affect its performance. This latter effect, however, barely exists for the k-greedy mechanisms due to the cycle size constraint. For the TTC mechanism, Theorem 5 implies that the second effect completely dominates the first effect

in the limit as $N \to \infty$. The right panel of Figure 7 shows that the dominance of the second effect holds even for relatively small N. The average payoff is increasing in N for the k-greedy mechanisms as expected (since the second effect barely matters), and it is decreasing in N for the TTC mechanism, resulting in the dominance of the second effect.

5. In Figures 6 and 7 (and in any graphs we present in the Online Appendix with the same axes), the confidence intervals of the k-greedy mechanisms tend to be narrower than those for the TTC mechanism. This suggests a yet new advantage of the k-greedy mechanisms in terms of the predictability of the welfare distribution. The differences in the widths of the confidence intervals arise because it is much less likely under the k-greedy mechanism to become an affected agent. This implies that the rank of the object that a given agent receives is close to deterministic when N is large.[32] In contrast, under the TTC mechanism, (i) agents receive highly desirable or highly undesirable objects depending on whether they are affected by other agents' dropout, and (ii) agents are likely to belong to large cycles and the probabilities of being affected (i.e., at least one agent in the cycle drops out) are high in large cycles.

4.3 Modifications

To test the robustness of our simulation results, the Online Appendix considers various modifications of our base model. More specifically, we consider the case when there are ex ante popular and unpopular objects, the case when the market is two-sided, and the cases when each agent has an ad-hoc utility/disutility for the object that they originally own. We also tested some mixtures of those models. In any of these specifications, we observe the robustness of the effect that the k-greedy mechanisms with small values of k perform relatively well even with small p. Additionally, we find that some of these modifications amplify this effect.

As the first paper providing an economic analysis of the dropout problem, to focus on first-order issues, we did not primarily consider situations in which the affected agents take part in a rewiring process after dropouts happen. The Online Appendix, however, considers such situations as well. Specifically, we consider *centralized* and

[32] Indeed, it is evident in the right panel of Figure 7 that the confidence intervals become narrower and collapse as N increases.

decentralized rewiring processes. Under the centralized rewiring process, as an extreme scenario, we let all the affected agents participate in the given mechanism and none of them drop out of the market. In reality, however, some of the affected agents may not participate in the rewiring process and, even if they do, they may drop out after the process is run. Thus, we consider a setting where those reassigned agents again drop out with probability p and the agents in the same cycle are affected. We find that, although the TTC mechanism outperforms k-greedy mechanisms in the first unrealistic specification, our main insights from Section 4.2 carry over to the second specification. Under the decentralized rewiring process, we assume that the affected agents are randomly paired up, and the two agents in a pair exchange their objects if they find each other's object acceptable. We then consider a rather unrealistic situation where any agents who did not exchange their objects in their pair are randomly paired up again, and this procedure continues indefinitely until no pair can be formed to exchange objects in an "acceptable" manner. We find that, under both settings, the k-greedy mechanisms with small values of k outperform the outcome of the TTC mechanism. The details are in the Online Appendix.

4.4 School Choice

The simulation results so far demonstrate that the TTC mechanism performs poorly, providing an explanation for why the TTC mechanism is not widely used in practice. TTC, however, was used in some school-choice contexts, such as New York City and New Orleans, where students can be thought of as being endowed with a school (for example, the "walk zone school").[33] Why is this the case?

We view that the key is in how "soft" the capacity constraint is. In the house exchange problem, it would be practically infeasible for multiple agents to consume the same object. Thus, once an agent in a given cycle drops out, all agents in the cycle have to become unassigned. The school-choice situation is different in that it would not be fatal for schools to accommodate slightly more students than their capacity because they typically have many available seats. Hence, even if one student drops out of a cycle, one could imagine a system that lets the other agents in the cycle

[33]New York City has adopted a version of the TTC mechanism in a reassignment process of high school students for several years (Morrill and Roth, 2024), and the Recovery School District of New Orleans used the TTC mechanism for one year (Abdulkadiroglu et al., 2017). In addition, the school board of San Francisco approved the adoption of the TTC mechanism but it was never implemented (Abdulkadiroglu et al., 2017).

receive the school seat specified in the cycle and only changes the assignment of the dropped student to the school she was initially endowed with, such as the "walk-zone school." One concern, then, is that such a system may result in a significant excess of the number of students who are assigned a given school compared to that school's capacity, when many dropout students return to the school. In the Online Appendix, we provide simulation results to demonstrate that such a concern may not be too much of an issue: although the number of students who return to the endowed school can be large, the resulting excess (i.e., the number of returning students minus that of the students who leave) is typically much smaller.

5 Conclusion

We considered a problem of "robust exchanges." When cycle sizes are restricted to be k or fewer, there is no strategy-proof mechanism that always implements a cycle such that every agent receives their first-choice object, if any. Given the impossibility, we proposed the k-greedy mechanism, which always implements such a cycle and is (constrained) efficient. Although the mechanism necessarily violates strategy-proofness given these conditions, we have shown that it possesses various desirable strategic properties. We then considered the possibility of using the k-greedy mechanism in a model where there is no cycle-length constraint but each agent drops out with a fixed probability p. We showed, both theoretically and through simulations, that the k-greedy mechanisms (with small values of k) outperform the TTC mechanism.

Our paper opens a variety of avenues for future research. First, as far as we know, our paper is the first to consider the possibility of dropouts seriously, analyze its effect, and propose a remedy. Being the first paper, we started with the simplifying assumption that the dropout probability is the same p independently across agents and their assignments. In the future research, one could consider endogenizing p, that is, p depends on the object received. Such an assumption might be particularly plausible if the reason for the dropout is to receive an outside option whose value realizes only after the mechanism is run. In such a case, p would be higher for agents who receive less desirable objects from the given mechanism.

Second, we could develop a theory of optimal k. Our simulations show that k-greedy mechanisms with small values of k generally outperform TTC mechanisms, and large values of k do not work well. We did not delve into the question of which

k would be optimal, and such an analysis would need a more specific performance measure of mechanisms than what we have now.

Third, there may be other ways to guarantee k-robustness than using a k-greedy mechanism. One possibility that is frequently suggested to us is to partition the set of agents into groups of k agents (with some adjustment when N is not divisible by k), and then apply the TTC mechanism within each group. Such a mechanism would not be k-unanimous or k-efficient, and according to our simulations, its performance is considerably worse than the k-greedy mechanism when k is small (k = 2-5). Even though this particular mechanism may not be ideal, there may be other k-robust mechanisms that work well. Exploration of such mechanisms is left for future research.

Fourth, k-greedy mechanisms impose a hard constraint of cycle size k. As we have seen, the restriction entails a tradeoff between a lower likelihood of agents being affected by a dropout and reduced efficiency when there is no dropout. One could consider optimizing against this tradeoff by allowing for a certain number of larger cycles, provided that doing so results in highly desirable assignments. One way to generate occasional larger cycles is to again partition the set of agents and apply the TTC mechanism within each group as described above, where each group consists of m agents with $m > k$. Although such mechanisms again fail to be k-unanimous or k-efficient whenever $m < N$, we simulated various values of m and found that the larger values of m work better than k. One would then be interested in identifying the optimal m. Such an analysis would again need some specific performance measure, and thus, it falls outside the scope of the current paper.

There are many ways to develop the theory for the cycle-size restriction, even without the possibility of dropouts. For example, one could characterize the maximal domain of preferences such that the impossibility results as in Section 2.2 hold. Kamada and Yasuda (2025) consider single-peaked preferences, motivated by the cyclic nature of the preferences in the counterexamples in Section 2.2, but other ways of restricting the preference domain may be possible. Another possibility is to generalize the model of exchange to allow for agents with no endowment and objects that are not initially owned by any agents. In such a market, under the assignment generated by the TTC mechanism (with an appropriate adjustment of the pointing rule as in Abdulkadiroğlu and Sönmez (1999)), whether an agent's dropout affects another agent would depend on their relative positions in the cycle. For example, if only an agent without the initial endowment drops out in a given cycle, other agents in the cycle would not be affected. However, if an agent who received an object without an initial owner drops out, then all other agents in the same cycle would be affected. An analysis of this complex situation needs to wait for future research.

This paper offers the first rigorous economic analysis of the dropout problem—a practical challenge in designing exchange markets. We hope that our study contributes not only to the theoretical advancement of exchange mechanisms but also to their practical implementation.

References

Abdulkadiroglu, A., Y.-K. Che, P. A. Pathak, A. E. Roth, and O. Tercieux (2017): "Minimizing justified envy in school choice: the design of New Orleans' OneApp," Tech. rep., National Bureau of Economic Research.

Abdulkadiroğlu, A. and T. Sönmez (1999): "House allocation with existing tenants," *Journal of Economic Theory*, 88, 233–260.

——— (2003): "School choice: A mechanism design approach," *American Economic Review*, 93, 729–747.

Akbarpour, M., S. Li, and S. O. Gharan (2020): "Thickness and information in dynamic matching markets," *Journal of Political Economy*, 128, 783–815.

Alvelos, F., X. Klimentova, A. Rais, and A. Viana (2015): "A compact formulation for maximizing the expected number of transplants in kidney exchange programs," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 616.

Ashlagi, I., M. Burq, P. Jaillet, and A. Saberi (2018): "Maximizing efficiency in dynamic matching markets," *arXiv preprint arXiv:1803.01285*.

Ashlagi, I. and A. E. Roth (2021): "Kidney exchange: An operations perspective," *Management Science*, 67, 5455–5478.

Balbuzanov, I. (2020): "Short trading cycles: Paired kidney exchange with strict ordinal preferences," *Mathematical Social Sciences*, 104, 78–87.

Balinski, M. and T. Sönmez (1999): "A tale of two mechanisms: student placement," *Journal of Economic Theory*, 84, 73–94.

Chung, K.-S. (2000): "On the existence of stable roommate matchings," *Games and Economic Behavior*, 33, 206–230.

Constantino, M., X. Klimentova, A. Viana, and A. Rais (2013): "New insights on integer-programming models for the kidney exchange problem," *European Journal of Operational Research*, 231, 57–68.

Dickerson, J. P., A. D. Procaccia, and T. Sandholmb (2019): "Failure-aware kidney exchange," *Management Science*, 65, 1768–1791.

Fack, G., J. Grenet, and Y. He (2019): "Beyond truth-telling: Preference estimation with centralized school choice and college admissions," *American Economic Review*, 109, 1486–1529.

Feigenbaum, I. and C. He (2024): "Strategyproofness in Failure-Aware Kidney Exchange," *Available at SSRN 4796610*.

Frieze, A. and B. G. Pittel (1995): "Probabilistic analysis of an algorithm in the theory of markets in indivisible goods," *The Annals of Applied Probability*, 768–808.

Gale, D. and L. S. Shapley (1962): "College admissions and the stability of marriage," *The American Mathematical Monthly*, 69, 9–15.

Kamada, Y. and Y. Yasuda (2025): "Robust exchange under single-peaked preferences," *Available at SSRN 5124793*.

Klaus, B. (2011): "Competition and resource sensitivity in marriage and roommate markets," *Games and Economic Behavior*, 72, 172–186.

Ma, J. (1994): "Strategy-proofness and the strict core in a market with indivisibilities," *International Journal of Game Theory*, 23, 75–83.

Morrill, T. and A. E. Roth (2024): "Top trading cycles," *Journal of Mathematical Economics*, 112, 102984.

Nicoló, A. and C. Rodríguez-Álvarez (2012): "Transplant quality and patients' preferences in paired kidney exchange," *Games and Economic Behavior*, 74, 299–310.

Nicolò, A. and C. Rodríguez-Álvarez (2013): "Incentive compatibility and feasibility constraints in housing markets," *Social Choice and Welfare*, 41, 625–635.

Nicolo, A. and C. Rodriguez-Alvarez (2017): "Age-based preferences in paired kidney exchange," *Games and Economic Behavior*, 102, 508–524.

Rodríguez-Álvarez, C. (2009): "Strategy-proof coalition formation," *International Journal of Game Theory*, 38, 431–452.

——— (2023): "Maximal domains for strategy-proof pairwise exchange," *Mathematical Social Sciences*, 126, 107–118.

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

——— (2002): "The economist as engineer: Game theory, experimentation, and computation as tools for design economics," *Econometrica*, 70, 1341–1378.

——— (2008): "Deferred acceptance algorithms: History, theory, practice, and open questions," *International Journal of Game Theory*, 36, 537–569.

Roth, A. E. and E. Peranson (1999): "The redesign of the matching market for American physicians: Some engineering aspects of economic design," *American Economic Review*, 89, 748–780.

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

Roth, A. E., T. Sönmez, and M. U. Ünver (2005a): "A kidney exchange clearinghouse in New England," *American Economic Review*, 95, 376–380.

——— (2005b): "Pairwise kidney exchange," *Journal of Economic Theory*, 125, 151–188.

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

Takagi, S. and S. Serizawa (2010): "An impossibility theorem for matching problems," *Social Choice and Welfare*, 35, 245–266.

Takamiya, K. (2013): "Coalitional unanimity versus strategy-proofness in coalition formation problems," *International Journal of Game Theory*, 42, 115–130.

Toda, M. (2006): "Monotonicity and consistency in matching markets," *International Journal of Game Theory*, 34, 13–31.

APPENDIX

A (i, k)-Serial Dictatorship Algorithm on S

Given a set of agents $S \subseteq I$, their preferences $(\succ_j)_{j\in S}$ over S, an agent $i \in S$, and $k \in \mathbb{N}$, define the (i, k)-serial dictatorship algorithm on S as follows:

(i, k)-Serial Dictatorship Algorithm on S:

Consider a graph, denoted by G, with the set of nodes being S where the node corresponding to a given agent has an outgoing arrow to every node corresponding to an object that is acceptable for the agent. For any $j \in I$, let $b^l(j)$ be agent j's l-th choice in S according to $\succ_j$.

Step 1-1: Let $j^1 := i$. If $b^1(i) = i$, then let the algorithm output $\mu_j = j$ for all $j \in S$. If $b^1(i) \neq i$, check if there is a cycle on G with size k or fewer that involves the arrow from i to $b^1(i)$.

- If there is such a cycle, then let $\mu_i = b^1(i) =: j^2$ and go to Step 2-1.
- If there is no such cycle, then go to Step 1-2.

⋮

Step 1-l: If $b^l(i) = i$, then let the algorithm output $\mu_j = j$ for all $j \in S$. If $b^l(i) \neq i$, check if there is a cycle on G with size k or fewer that involves the arrow from i to $b^l(i)$.

- If there is such a cycle, then let $\mu_i = b^l(i) =: j^2$ and go to Step 2-1.
- If there is no such cycle, then go to Step 1-$(l + 1)$.

⋮

Step m-1: Check if there is a cycle on G with size k or fewer that involves the sequence of arrows from j^1 to … to j^m to $b^1(j^m)$.

- If there is such a cycle, then let $\mu_{j^m} = b^1(j^m) =: j^{m+1}$. If $j^{m+1} = i$, let $\mu_j = j$ for every $j \in S \setminus \{j^1, \dots, j^m\}$ (which specifies the object that each agent receives) and end the algorithm. Otherwise, go to Step $(m+1)$-1.
- If there is no such cycle, then go to Step m-2.

⋮

Step m-l: Check if there is a cycle on G with size k or fewer that involves the sequence of arrows from j^1 to … to j^m to $b^l(j^m)$.

- If there is such a cycle, then let $\mu_{j^m} = b^l(j^m) =: j^{m+1}$. If $j^{m+1} = i$, let $\mu_j = j$ for every $j \in S \setminus \{j^1, \dots, j^m\}$ (which specifies the object that each agent receives) and end the algorithm. Otherwise, go to Step $(m+1)$-1.
- If there is no such cycle, then go to Step m-$(l+1)$.

⋮

This algorithm ends in a finite number of steps and outputs a well-defined exchange μ.

B Proofs

B.1 Proof of Theorem 1

We first provide a lemma that is useful for proving Theorem 1.

Lemma 1. *For any I and k such that $1 \leq k \leq |I|$, if a mechanism is k-unanimous and strategy-proof, then it is individually rational.*

Proof. Take a mechanism ψ that is k-unanimous and is not individually rational. Then, there are a preference profile $(\succ_j)_{j\in I}$ and an agent i such that $i \succ_i \psi_i((\succ_j)_{j\in I})$. But consider any $\succ'_i$ whose first choice is object i. Then, k-unanimity and $k \geq 1$ imply that $\psi_i(\succ'_i, (\succ_j)_{j\neq i}) = i$. This implies $\psi_i(\succ'_i, (\succ_j)_{j\neq i}) \succ_i \psi_i((\succ_j)_{j\in I})$. This shows that ψ is not strategy-proof, completing the proof. □

Proof of Theorem 1. Since the core idea of the proof is explained via the 3-agent example in the main text after the statement of Theorem 1, we relegate the formal extension of that agrument to the Online Appendix. □

B.2 Proof of Theorem 2

Proof. First, we show that Round 0 terminates in polynomial time. To see this, consider a procedure where we start with an arbitrary agent and follow outgoing arrows where agent i's arrow points to agent j if object j is agent i's first-choice object. At some point in the sequence of agents connected by arrows, there must be an arrow that points to an agent who was already pointed to along the sequence. This procedure defines a cycle. If the cycle size is k or fewer, we implement the cycle and remove all the agents who were involved in the sequence of agents. Otherwise, we remove all the agents who were involved in the sequence. Once we remove agents, we again start with an arbitrary agent and follow the same procedure, and we continue having this procedure until no more agents are left. Since (i) each procedure takes at most N steps because there are only N agents and (ii) at least one agent is involved in a cycle that is implemented and/or is removed for each procedure, Round 0 takes at most N^2 steps.

Second, we show that each round i terminates in polynomial time for each $i \in I$. If agent i is not in S^{i-1} (i.e., agent i was assigned in a previous round), then this round takes only 1 step. Otherwise, we consider the following two procedures.

In the first procedure, we note that there are at most N^{k-1} different cycles that agent i can be involved with in any k-robust exchange.[34] For those cycles, we first remove all cycles such that some agent receives an object that she deems unacceptable. Since we check at most k agents for each cycle, this procedure takes at most $k \cdot N^{k-1}$ steps. Let C^1 be the set of remaining cycles.

In the second procedure, we form a cycle. We begin with identifying the object that agent i receives. In at most N^{k-1} steps, we can identify the best object that agent i receives among the cycles in C^1, which has at most N^{k-1} possible cycles.

[34] This can be shown by mathematical induction. First, the claim obviously holds for $k = 1$. Suppose that the claim holds for $k = l$. That is, we suppose that the number of different cycles that agent i can be involved with in any k-robust exchange is at most N^{l-1}. Then, the number of different cycles that agent i can be involved with in any k-robust exchange is at most $N^{l-1} + \prod_{l'=1}^{l}(N - l')$, which is no greater than $N^{l-1} + N^{l-1}(N-1)$, which is equal to N^l. Thus, the claim holds for $k = l + 1$.

This object is denoted j^2. Let C^2 be the set of cycles in C^1 such that agent i receives object j^2. If $j^2 = i$, we terminate the procedure, and in this case, the number of steps for this procedure is N^{k-1}. Otherwise, for each of the cycles in C^2 (which has at most N^{k-1} cycles), we can identify the best object that agent j^2 receives in C^2. This object is denoted j^3. Let C^3 be the set of cycles in C^2 such that agent j^2 receives object j^3. In general, if $j^l = i$, we terminate the procedure, and in this case, the number of steps for this procedure is $(l-1) \cdot N^{k-1}$. Otherwise, for each of the exchanges in C^l (which has at most N^{k-1} cycles), we can identify the best object that agent j^l receives in C^l. This object is denoted j^{l+1}. Let C^{l+1} be the set of cycles in C^l such that agent j^l receives object j^{l+1}. Note that, by the definition of the procedure, if $j^l \neq i$ holds for all $l \in \{2, \dots, k\}$ then j^k must deem agent i as acceptable. In that case, the number of steps for this procedure is $k \cdot N^{k-1}$.

This implies that Round i takes at most $2k \cdot N^{k-1}$ steps.

Overall, since Round 0 takes at most N^2 steps and each Round i takes at most $2k \cdot N^{k-1}$ steps, the k-greedy algorithm terminates in polynomial time. □

B.3 Proof of Theorem 3

Proof. Fix a k-greedy mechanism Γ. It is k-robust because of the description of the k-greedy algorithm. It is individually rational because agents only point to acceptable objects in any Round of the k-greedy algorithm. It is also k-unanimous due to Round 0 of the k-greedy algorithm. In what follows, we show that Γ is k-efficient.

To show the k-efficiency, let μ be the outcome of Γ and suppose, for a contradiction, that it is not k-efficient. Then, there must exist another k-robust μ' such that $\mu'_i \succeq_i \mu_i$ for all i.

Consider an ordering $\tilde{\sigma}$ such that $\tilde{\sigma}(i) = l$ if agent i is the l-th agent whose assignment is determined in the k-greedy algorithm, where we let assignments for agents be determined one by one in Round 0 as well.[35] Let $i^* \in I$ be such that (i) $\mu'_i = \mu_i$ for all i such that $\tilde{\sigma}(i) < \tilde{\sigma}(i^*)$ and (ii) $\mu'_{i^*} \succ_{i^*} \mu_{i^*}$. That is, i^* is the "first agent" whose assignment differs between μ and μ'. But then, μ_{i^*} can never be the most desirable object that agent i^* receives under a k-robust exchange such that each i with $\tilde{\sigma}(i) < \tilde{\sigma}(i^*)$ receives μ_i. This contradicts our assumption that μ is the outcome

[35] This ordering $\tilde{\sigma}$ may be different from the ordering σ that is used in the k-greedy algorithm. For example, if no one is assigned in Round 0 and agent 1 receives object 7 under μ, then we have $\tilde{\sigma}(7) = 2$.

of Γ. This completes the proof. $\square$

B.4 Proof of Proposition 2

Proof. **Part 1:** Fix any $\succ_i'$ and consider σ such that $\sigma(i) = 1$. We show that for any $\succ_{-i}$, we have $\psi_i(\succ) \succeq_i \psi_i(\succ_i', \succ_{-i})$, which proves the desired result. To simplify notation, let $\hat{j} := \psi_i(\succ_i', \succ_{-i})$ and $\tilde{j} := \psi(\succ_i, \succ_{-i})$. Given the definition of the k-greedy algorithm, agent i must have been assigned object $\tilde{j}$ under the input $(\succ_i, \succ_{-i})$ either in Round 0 or in Round 1. If it is Round 0, then that means $\tilde{j}$ is agent i's first choice according to $\succ_i$, which implies $\tilde{j} \succeq_i \hat{j}$ as desired. Thus, in what follows, we consider the case when agent i is assigned object $\tilde{j}$ under the input $(\succ_i, \succ_{-i})$ in Round 1. Now, similarly to the case of $\tilde{j}$, given the definition of the k-greedy algorithm, agent i must have been assigned object $\hat{j}$ under the input $(\succ_i', \succ_{-i})$ either in Round 0 or in Round 1.

Suppose it was Round 0. This means that there was a unanimous trading cycle with size k or less in which agent i receives $\hat{j}$ under the input $(\succ_i', \succ_{-i})$. Let F be the set of agents involved in this unanimous trading cycle. Note that, since agent i does not receive an object in Round 0, we must have $F \subseteq S^0$ where S^0 is the set of agents who received their own objects in Round 0 of the k-greedy algorithm under the input $(\succ_i, \succ_{-i})$. Now, in Round 1 under the input $(\succ_i, \succ_{-i})$, the k-greedy algorithm finds all the cycles with size k or less on the graph G defined in the definition of (i, k)-serial dictatorship algorithm on S (in this proof, whenever we refer to cycles, we mean the cycles defined there) and chooses one that gives the best object to agent i among them. Since the cycle in which agents in F except agent i receive their respective first-choice objects is available, agent i must receive an object as good as $\hat{j}$. This implies $\tilde{j} \succeq_i \hat{j}$, as desired.

Suppose now that, in Round 1, agent i was assigned object $\hat{j}$ under the input $(\succ_i', \succ_{-i})$. Then, S^0 under the input $(\succ_i, \succ_{-i})$ and S^0 under the input $(\succ_i', \succ_{-i})$ are the same as each other. The fact that agent i is assigned object $\tilde{j}$ in Round 1 under the input $(\succ_i, \succ_{-i})$ implies that for any j' such that $j' \succ_i \tilde{j}$, there is no cycle of size k or less in which agent i receives object j'. This and the fact that agent i receives $\hat{j}$ under the input $(\succ_i', \succ_{-i})$ (so there exists a cycle with size k or less in S^0 in which agent i receives object $\hat{j}$) imply that $\tilde{j} \succeq_i \hat{j}$, as desired.

Overall, the proof is now complete.

<u>Part 2</u>: Let object j be agent i's first choice.

Fix any $\succ'_i$ and consider $\succ_{-i}$ such that agent j's first choice is i. We show that for any σ, we have $\psi_i(\succ) \succeq_i \psi_i(\succ'_i, \succ_{-i})$, which proves the desired result. For any σ, given $\succ_{-i}$ that is chosen, we have $\psi_i(\succ_i, \succ_{-i}) = j$, so $\psi_i(\succ_i, \succ_{-i}) \succeq_i \psi_i(\succ'_i, \succ_{-i})$ holds because object j is agent i's first choice. This completes the proof.

<u>Part 3</u>: Fix any $\succ'_i$ that is not equivalent to $\succ_i$ and consider $\succ_{-i}$ that depends on "the first place" at which $\succ_i$ and $\succ'_i$ differ from each other in the manner we specify shortly. We show that for any σ, we have $\psi_i(\succ) \succ_i \psi_i(\succ'_i, \succ_{-i})$, which proves the desired result.

To specify $\succ_{-i}$, for given preferences $\bar{\succ}_i$, let $f_m(\bar{\succ}_i)$ be agent i's m-th choice under $\bar{\succ}_i$. Define l and l' such that $l \neq l'$ as follows: There exists m such that $f_m(\succ_i) = l$ and $f_m(\succ'_i) = l'$, and $f_n(\succ_i) = f_n(\succ'_i)$ for all $n < m$. That is, l and l' are "the first place" at which $\succ_i$ and $\succ'_i$ are different from each other. Since $\succ'_i$ is not equivalent to $\succ_i$, the pair satisfying the above condition uniquely exists, and so l and l' are well defined. Moreover, l is acceptable to agent i under $\succ_i$ and l' is acceptable under $\succ'_i$.

If $l = i$, then consider $\succ_{-i}$ such that every agent $j \in I \setminus \{i, l'\}$ regards object j as her first choice and agent l' regards agent i as the first choice. Then, irrespective of σ, we have $\psi_i(\succ_i, \succ_{-i}) = i$ and $\psi_i(\succ'_i, \succ_{-i}) = l'$. Since $l' \neq l = i$ and the definition of (l, l') imply $i \succ_i l'$, we have $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$.

If $l' = i$, then consider $\succ_{-i}$ such that every agent $j \in I \setminus \{i, l\}$ regards object j as her first choice and agent l regards agent i as the first choice. Then, irrespective of σ, we have $\psi_i(\succ_i, \succ_{-i}) = l$ and $\psi_i(\succ'_i, \succ_{-i}) = i$. Since $i = l' \neq l$ and the definition of (l, l') imply $l \succ_i i$, we have $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$.

Finally, if $l \neq i \neq l'$, consider $\succ_{-i}$ such that every agent $j \in I \setminus \{i, l, l'\}$ regards object j as her first choice, $i \succ_l l' \succ_l l$, and $i \succ_{l'} l \succ_{l'} l'$. Note that the definition of (l, l') imply $l \succ_i l'$. Fix any σ, and consider the following three cases.

1. If $\sigma(i) < \min\{\sigma(l), \sigma(l')\}$, then $\psi(\succ_i, \succ_{-i})$ would be such that a cycle (i, l) is formed in either Round 0 or Round $\sigma(i)$ (and agent l' would be unassigned), so agent i receives object l. On the other hand, $\psi(\succ'_i, \succ_{-i})$ would be such that a cycle (i, l') is formed in either Round 0 or Round $\sigma(i)$ (and agent l would be unassigned), so agent i receives object l'. Hence, we have $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$.

2. If $\sigma(l) < \min\{\sigma(i), \sigma(l')\}$, then $\psi(\succ_i, \succ_{-i})$ would be such that a cycle (l, i) is

formed in either Round 0 or Round $\sigma(l)$ (and agent l' would be unassigned), so agent i receives object l. On the other hand, $\psi(\succ'_i, \succ_{-i})$ would be such that a cycle (i, l') is formed (in Round 0, if l' is i's first choice under $\succ'_i$) or cycle (l, i, l') is formed (in Round $\sigma(l)$, if l' is not i's first choice under $\succ'_i$), so agent i receives object l'. Hence, we have $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$.

3. If $\sigma(l') < \min\{\sigma(i), \sigma(l)\}$, then $\psi(\succ_i, \succ_{-i})$ would be such that a cycle (i, l) is formed (in Round 0, if l is i's first choice under $\succ_i$) or cycle (l', i, l) is formed (in Round $\sigma(l')$, if l is not i's first choice under $\succ_i$), so agent i receives object l. On the other hand, $\psi(\succ'_i, \succ_{-i})$ would be such that a cycle (l', i) is formed in either Round 0 or Round $\sigma(l')$ (and agent l would be unassigned), so agent i receives object l'. Hence, we have $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$.

Overall, in all the cases analyzed above, we obtained $\psi_i(\succ_i, \succ_{-i}) \succ_i \psi_i(\succ'_i, \succ_{-i})$. This completes the proof. □

B.5 Proof of Theorem 4

Proof. Consider agents 1, 2, and 3, and their preference profile as in (1) in Section 2.2 (in particular, these agents regard objects other than 1, 2, and 3 as unacceptable). The preferences of other agents are set to be arbitrary. In what follows, we only consider agents 1, 2, and 3. Consider the following cardinal preferences:

$$u_{11} = 0, \quad u_{12} = 1, \quad u_{13} = 1 - \varepsilon.$$

If all agents submit their preferences truthfully, then a 3-cycle is formed and every agent receives their first-choice object under the TTC mechanism. Hence, agent 1 indeed receives her first-choice object if none of the agents in the 3-cycle drops, which happens with probability $(1-p)^3$. Thus, agent 1's expected payoff is $(1-p)^3 \cdot 1 = (1-p)^3$.

Fixing the other agents' strategies, if agent 1 has deviated to reporting the preferences (2) in Section 2.2, then she would be in a 2-cycle together with agent 3, and thus her expected payoff would be $(1-p)^2(1-\varepsilon)$. Since $p > 0$, there is $\varepsilon > 0$ such that

$$(1-p)^3 < (1-p)^2(1-\varepsilon),$$

showing that there exists $(u_{1j})_{1\leq j\leq N}$ such that agent 1 has an incentive to deviate. This implies that truthful preference submission by all agents is not a Nash equilibrium. □

B.6 Proof of Proposition 3

Proof. Consider the following preferences:

$$\begin{aligned} \succ_i &: i+1, i \qquad \text{for all } i \neq N, \\ \succ_N &: 1, 2, \dots, N-1, N. \end{aligned}$$

Fix $\varepsilon \in (0, \frac{1}{2})$. Consider agent N's utility function $(u_{Ni})_{i=1,\dots,N}$ such that $1 > u_{Ni} > 1-\varepsilon$ for $i \in \{1, \dots, N-l+1\}$, $u_{Ni} = \varepsilon^i$ for $i \in \{N-l+2, \dots, N-1\}$, and $u_{NN} = 0$.

Under any k-greedy mechanism, if all agents submit their preferences truthfully, then a k-cycle is formed and agent N receives her $(N-k+1)$th-choice object. Hence, agent N indeed receives her $(N-k+1)$th-choice object if none of the agents in the k-cycle drops out, which happens with probability $(1-p)^k$. Thus, agent N's expected payoff is

$$(1-p)^k \cdot u_{N(N-k+1)} + (1-(1-p)^k) \cdot 0 = (1-p)^k u_{N(N-k+1)}. \tag{3}$$

First, consider any k-greedy mechanism with $k \leq l$. By our specification of u, the payoff in (3) is greater than $(1-p)^k(1-\varepsilon)$ if $k = l$ and is equal to $(1-p)^k \varepsilon^{N-k+1}$ if $k < l$.

Fixing the other agents' strategies, if agent N has deviated to reporting some other preferences, then the only possibilities for which a different exchange is realized are when she is in a k'-cycle and receives $(N-k'+1)$th-choice object for $k' \in \{1, \dots, k-1\}$. For each such k', her expected payoff would be

$$(1-p)^{k'} u_{N(N-k'+1)} + (1-(1-p)^{k'}) \cdot 0, \tag{4}$$

which is no greater than $(1-p)^{k'}\varepsilon^{N-k'+1}$. Since $p < 1$ and $k' < k$, there is $\bar{\varepsilon} > 0$ such that for all $\varepsilon \in (0, \bar{\varepsilon})$,

$$(1-p)^{k'}\varepsilon^{N-k'+1} \leq (1-p)^k(1-\varepsilon) \text{ and } (1-p)^{k'}\varepsilon^{N-k'+1} \leq (1-p)^k\varepsilon^{N-k+1}.$$

Hence, submitting the true preferences is a best response for agent N. Since submitting the true preferences is obviously a best response for all other agents, this implies that truthful preference submission by all agents is a Nash equilibrium of the preference submission game $(N, \Gamma^k, p, (u_{ij})_{1\leq i,j\leq N})$ when $\varepsilon \in (0, \bar{\varepsilon})$.

Next, consider any k-greedy mechanism with $k > l$. Fixing the other agents' strategies, if agent N has deviated to reporting that her first-choice object is object $N - l + 1$ (note that this object is not object 1 or object N because $l < k \leq N$ and $l \geq 2$), she would be in a l-cycle and receives the $(N - l + 1)$th-choice object, and thus her expected payoff would be

$$(1-p)^l u_{N(N-l+1)} + (1 - (1-p)^l) \cdot 0 = (1-p)^l u_{N(N-l+1)}.$$

Note that for any ε, we have

$$\frac{u_{N(N-l+1)}}{u_{N(N-k+1)}} > \frac{1-\varepsilon}{1} = 1 - \varepsilon$$

by our specification of u. Moreover, since $p > 0$, there is $\bar{\bar{\varepsilon}} > 0$ such that for all $\varepsilon \in (0, \bar{\bar{\varepsilon}})$,

$$(1-p)^k < (1-p)^l (1-\varepsilon).$$

Hence, for such ε, we have

$$(1-p)^k < (1-p)^l (1-\varepsilon) < (1-p)^l \frac{u_{N(N-l+1)}}{u_{N(N-k+1)}},$$

and thus, we obtain:

$$(1-p)^k u_{N(N-k+1)} < (1-p)^l u_{N(N-l+1)}.$$

Hence, submitting the true preferences is not a best response for agent N. This implies that truthful preference submission by all agents is not a Nash equilibrium of the preference submission game $(N, \Gamma^k, p, (u_{ij})_{1\leq i,j\leq N})$ for any $k > l$ when $\varepsilon \in (0, \bar{\bar{\varepsilon}})$.

Overall, we have shown that there exists $(u_{ij})_{1\leq i,j\leq N}$ such that truthful preference submission by all agents is a Nash equilibrium of the preference submission game $(N, \Gamma^k, p, (u_{ij})_{1\leq i,j\leq N})$ for any $k \leq l$ but it is not a Nash equilibrium of the preference submission game $(N, \Gamma^k, p, (u_{ij})_{1\leq i,j\leq N})$ for any $k > l$. □

B.7 Proof of Theorem 5

<u>Part 1:</u> Given $K \in \mathbb{N}$ and $N \in \mathbb{N}$, let $Z(K, N)$ be the expected number of agents involved in a cycle of size no greater than K in the outcome of the TTC mechanism when there are N agents.

Claim 1. $\lim_{N\to\infty} \frac{Z(K,N)}{N} = 0$.

Proof of Claim 1. Suppose that the claim does not hold. That is, suppose that there exists $\alpha > 0$ such that

$$\limsup_{N\to\infty} \frac{Z(K,N)}{N} = \alpha.$$

Then, we can take a subsequence of N, $(N_l)_{l\in\mathbb{N}}$, such that

$$\lim_{l\to\infty} \frac{Z(K,N_l)}{N_l} = \alpha. \quad (5)$$

Now, let $Y(N)$ be the expected number of cycles when there are N agents.

When there are N_l agents, $Z(K, N_l)$ agents are involved in a cycle of size no greater than K on average, and thus there are at least $Z(K, N_l)/K$ cycles. Hence, for each l, we must have

$$\frac{Y(N_l)}{N_l} \geq \frac{Z(K,N_l)/K}{N_l},$$

where, by (5), the right-hand side tends to $\frac{\alpha}{K}$ as l goes to infinity.

However, Theorem 2 of Frieze and Pittel (1995) implies that

$$\lim_{N\to\infty} \frac{Y(N)}{N} = \lim_{N\to\infty} \frac{\sqrt{2\pi N} + O(\log(N))}{N} = 0.$$

Since $\frac{\alpha}{K}$ is strictly positive, this is a contradiction. The proof is complete. □

By the symmetry across agents, for each agent i, the ex-ante probability that i is unassigned is at least

$$\sum_{m\in\mathbb{N}} \left[\frac{Z(m,N)}{N} - \frac{Z(m-1,N)}{N}\right] [1-(1-p)^m],$$

where we let $Z(0,N) := 0$.[36] For any fixed K, this is no less than:

$$\begin{aligned}
&\sum_{m\geq K+1}\left[\frac{Z(m,N)}{N}-\frac{Z(m-1,N)}{N}\right][1-(1-p)^m] \\
&\geq \sum_{m\geq K+1}\left[\frac{Z(m,N)}{N}-\frac{Z(m-1,N)}{N}\right][1-(1-p)^{K+1}] \\
&= \left[\frac{Z(N,N)}{N}-\frac{Z(K,N)}{N}\right][1-(1-p)^{K+1}] \\
&= \left[1-\frac{Z(K,N)}{N}\right][1-(1-p)^{K+1}] \\
&=: H(K,N).
\end{aligned}$$

Notice that

$$\lim_{K\to\infty}\lim_{N\to\infty} H(K,N) = \lim_{K\to\infty}(1-0)[1-(1-p)^{K+1}] = 1,$$

where we used Claim 1 for the first equality. This implies that each agent i will be unassigned with ex-ante probability approaching 1 as $N \to \infty$. Hence, the proof is complete.

Part 2: Fix a k-greedy mechanism. Let $A(N)$ be the expected number of agents who receive an object under the k-greedy mechanism with $k \geq 2$ when there are N agents.

Claim 2. *There are $\beta > 0$ and $\bar{N} < \infty$ such that $\frac{A(N)}{N} > \beta$ for all $N > \bar{N}$.*

Proof of Claim 2. Suppose, toward a contradiction, that there are no $\beta > 0$ and $\bar{N} < \infty$ such that $\frac{A(N)}{N} > \beta$ for all $N > \bar{N}$. That is, there is a subsequence of N, $(N_l)_{l\in\mathbb{N}}$ with $N_l > 1$ for each l, such that

$$\lim_{l\to\infty}\frac{A(N_l)}{N_l} = 0. \tag{6}$$

Fix any l and consider the market with N_l agents. Since the exchange produced by the k-greedy mechanism is k-efficient by Theorem 3, for each realization of preferences, there must be no pair of agents from the set of unassigned agents such that the two agents (i,j) find each other acceptable. This is because, otherwise, letting agent

[36] The probability is "at least" this much because when $m = 1$, i.e., if an agent is in a cycle by herself, she receives her own object irrespective of whether she drops out or not. Thus, the corresponding probability is $\frac{Z(1,N)}{N}$ instead of $\frac{Z(1,N)}{N}(1-(1-p))$.

i receive object j and agent j receive object i, with everything else equal, would be a Pareto improvement, contradicting the k-efficiency of the k-greedy mechanism. This implies that for each l, the expected number of non-overlapping pairs such that the two agents find each other acceptable is at most $A(N_l)$ when there are N_l agents, where we say that two pairs are non-overlapping if no agent from each pair is common.

Consider a pair of agents, $(m, m+1)$. The ex-ante probability that they find each other's object acceptable is $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$ given that the agents' ranking distribution is independent and uniform. Hence, for any N, the expected number of non-overlapping pairs such that the two agents find each other's object acceptable is at least $\frac{(N-1)/2}{4}$, which is equal to $\frac{N-1}{8}$.

Hence, we must have $\frac{N_l-1}{8} \leq A(N_l)$ for each l, which implies

$$\frac{A(N_l)}{(N_l-1)/8} \geq 1$$

for each l.[37]

But we have

$$\lim_{l\to\infty} \frac{A(N_l)}{(N_l-1)/8} = \lim_{l\to\infty} \left(\frac{A(N_l)}{N_l} \cdot 8 \cdot \frac{N_l}{N_l-1} \right) = 0,$$

where the last equality follows from (6). This is a contradiction, which completes the proof. □

The expected number of agents who receive an object under the k-greedy mechanism with $k \geq 2$ is at least $(1-p)^k \cdot A(N)$. Hence, we must have

$$F(N;\Gamma,p) \geq \frac{(1-p)^k \cdot A(N)}{N} = (1-p)^k \frac{A(N)}{N},$$

where Γ is the fixed k-greedy mechanism. By Claim 2, any subsequence of N, $(N_l)_{l\in\mathbb{N}}$, must satisfy

$$\lim_{l\to\infty} F(N_l;\Gamma,p) \geq (1-p)^k \frac{A(N_l)}{N_l} > (1-p)^k \beta.$$

Since the far right-hand side is a constant independent of N, the proof is complete.

[37] Note that the denominator is nonzero because $N_l > 1$.