# Strategyproof Matching of Roommates and Rooms

**Hadi Hosseini**[1], **Shivika Narang**[2], **Sanjukta Roy**[3]

[1]Pennsylvania State University
[2]University of New South Wales, Sydney
[3]Indian Statistical Institute, Kolkata
hadi@psu.edu, s.narang@unsw.edu.au, sanjukta@isical.ac.in

## Abstract

We initiate the study of matching roommates and rooms wherein the preferences of agents over other agents and rooms are complementary and represented by Leontief utilities. In this setting, $2n$ agents must be paired up and assigned to $n$ rooms. Each agent has cardinal valuations over the rooms as well as compatibility values over all other agents. Under Leontief preferences, an agent's utility for a matching is the minimum of the two values. We focus on the tradeoff between maximizing utilitarian social welfare and strategyproofness. Our main result shows that—in a stark contrast to the additive case—under binary Leontief utilities, there exist SP mechanisms that maximize the social welfare. We further devise a SP mechanism that implements such a welfare maximizing algorithm and is parameterized by the number of agents. Along the way, we highlight several possibility and impossibility results, and give upper bounds and lower bounds for welfare with or without strategyproofness.

**Extended version** — https://arxiv.org/abs/2412.13887

## 1 Introduction

The roommates matching problem is concerned with pairing up $2n$ agents (roommates) and assigning them to $n$ rooms. This problem originates from the seminal paper by Gale and Shapley—and discussed by Knuth—as a generalization of Stable Marriage (Gale and Shapley 1962). The goal is finding a stable disjoint pairing of agents such that no pair of agents would prefer each other to their current match. This problem has inspired a large body of work on roommate matching (Irving 1985), 3-dimensional matching (McKay and Manlove 2021), and coalition formation (Knuth 1976).

Even though the problem was mainly inspired by the college dormitory assignment, the majority of work on roommate matchings study stability where agents have preferences over other agents while being agnostic to their rooms. 3-Dimensional extensions of this problem primarily focus on finding stable coalitions of size three and do not consider preferences over different types of (often) complementary entities (e.g. roommates and rooms). This is particularly an important issue in common online marketplaces for short and long-term stays such as Airbnb, Roomi[1], and Roomsurf[2], where agents can specify their preferences over rooms as well as roommates.

Within three-dimensional matchings (as well as hedonic games), additivity of preferences is a common assumption. Similarly, some recent work on matching roommates and rooms considers additive utilities over rooms and roommates (Chan et al. 2016; Gan, Li, and Li 2023). Additive preferences model a more *substitutionary* approach to aggregate preferences. That is, receiving one undesirable match/item can be compensated by also receiving a desirable one. While this is plausible when partitioning agents into coalitions, it does not always provide a sufficiently meaningful way to compare choices, particularly when outcomes involve *complementary* alternatives. When matching roommates and rooms, for a person who values proximity to their classes, the utility derived from living with a close friend cannot necessarily *substitute* for being assigned a room in a remote corner of the campus.

**Complementary Preferences via Leontief.** A compelling alternative model for capturing complementarity is Leontief utilities that are common in production economics (Leontief 1965) as well as resource allocation and scheduling (Ghodsi et al. 2011; Dolev et al. 2012). Under Leontief utilities, different types of items or alternatives may be complements such as burgers and buns, pen and paper, someone's room and their roommate etc. This extends to many different settings. In group projects, students are only able to produce good quality work if they get along with their partners and have some interest in their topic.

In this paper, we consider the problem of matching roommates and rooms where the preferences of agents over other agents and rooms are given by cardinal values. The two fundamental notions in game theory and social choice are economic efficiency–measured by social welfare–and strategyproofness–no agent can obtain a more preferred outcome by manipulating its preferences. Prior works in this space largely explored the objective of maximizing social welfare–as a desirable notion for economic efficiency–under additive utilities without considering strategyproofness.

We depart from prior works by (i) assuming that pref-

[1] Roomi: https://roomiapp.com/

[2] Roomsurf: https://www.roomsurf.com/

| | **Leontief** | | **Additive** | |
|---|---|---|---|---|
| | **General** | **Binary** | **General** | **Binary** |
| **SW** | APX-h$^\dagger$ (Thm. 3.6) | APX-h$^\dagger$ (Thm. 3.6) | NP-h$^*$ | NP-h$^*$ |
| **SP Upper Bound** | ✗ (Thm. 4.1) | **1 (Thm. 4.6)** | ✗ (Thm. 4.1) | $2/3$ (Thm. 4.2) |
| **SP (polynomial time)** | ✗ (Thm. 4.1) | $1/3$ (Thm. 4.10) | ✗ (Thm. 4.1) | $1/7$ (Thm. 4.8) |

Table 1: Summary of the tradeoff between social welfare (SW) and strategyproofness (SP). Here ✗ denotes that no strategyproof mechanism can give any non-zero approximation of SW. APX-h and NP-h denote APX and NP-hardness, resp. $^*$ denotes a result by (Chan et al. 2016) and $^\dagger$ indicates bounds even for binary and symmetric valuations.

erences are represented by Leontief utilities, and (ii) primarily focusing on designing SP mechanisms that maximize the social welfare. Gibbard (1973) and Satterthwaite (1975)'s seminal works showed that every strategyproof social choice function is either dictatorial or imposing (with more than two alternatives). Similarly, the incompatibility between strategyproofness and economic efficiency in several object/resource allocation settings (Zhou 1990; Hylland and Zeckhauser 1979) persisted, which motivated a variety of approaches to bypass the negative results by, for example, restricting the domain (e.g. preferences or outcomes) or sacrificing/weakening the efficiency requirements.

## 1.1 Our Contribution

We study the problem of matching roommates and rooms when agents have preferences over each other as well as the rooms. These preferences are expressed in the form of non-negative cardinal values. We also study the case when these are restricted to binary values $(0/1)$. From the conceptual perspective, binary valuations represent practical applications in object or resource allocation or voting theory. They continue to pose many technical challenges in achieving axiomatic results when dealing with strategyproofness. They are well-motivated in our setting as well. For instance, a student may approve/disapprove a potential roommate based on factors such as smoking habits; or she may have a value of zero for room options without dedicated bathrooms.

Table 1 summarizes our main results under Leontief and additive utilities with general/binary valuations. We denote an $\alpha$-approximation to the optimal social welfare as $\alpha$-SW.

**Social Welfare (SW).** We begin by analyzing the welfare bounds for various maximal roommate matchings and show that a greedy approach performs the best ensuring $1/3$-SW. We then show that under Leontief utilities, finding a social welfare maximizing matching is APX-hard even for binary and symmetric valuations[3] and when the preference graph of agents is a tree of maximum degree three (Theorem 3.6). We then design an algorithm to obtain $0.559$-SW for any type of utility function (Proposition 3.8).

**Strategyproof (SP) Mechanisms.** When requiring strategyproofness, we show that when agent valuations are unrestricted, no SP mechanism can satisfy any approximation of social welfare. This holds for both Leontief and additive utilities (Theorem 4.1). A natural next step is to see if this continues to hold for binary valuations. For binary additive utilities, we show that no SP mechanism can satisfy $\alpha$-SW for any $\alpha > 2/3$ for binary but not (necessarily) symmetric preferences or $\alpha > 3/4$ for binary and symmetric preferences (Theorem 4.2 and Corollary 4.3).

Our main result shows that—in stark contrast with its additive counterpart—under binary Leontief utilities, there exists an SP mechanism that maximizes social welfare (Theorem 4.4). Our APX-hardness result rules out the possibility of a polynomial time algorithm, unless P=NP. We develop a novel SP mechanism that repeatedly reduces the welfare set based on the agents' preferences. However, this mechanism runs in time $O^*(n^{2n})$.[4] We improve upon this by developing an FPT algorithm[5] that implements our SP mechanism while maximizing the social welfare, parameterized by the number of agents, and runs in time $O^*(c^n)$ where $c > 0$ is a constant.

Finally, we explore various SP mechanisms to produce approximate SW in polynomial time. In particular, we show that the greedy mechanism is SP for binary Leontief utilities (Theorem 4.10). We also give an SP mechanism for binary additive utilities which is $1/7$-SW in general (Theorem 4.8) and $1/6$-SW for symmetric preferences (Corollary 4.9). Due to space constraints, omitted proofs are deferred to the extended version (Hosseini, Narang, and Roy 2024).

## 1.2 Related Work

Matchings have been studied in a variety of models and with several different objectives. In the house allocation problem each agent is matched to a house and agents have preferences over the houses. Economic efficiency and strategyproofness has been studied separately(Green and Laffont 1979; Hylland and Zeckhauser 1979) and together (Krysta et al. 2019). Unlike our problem, a deterministic serial dictatorship mechanism easily gives better than a $1/2$-approximation for maximum size house allocation. Thus, randomized variations of serial dictatorship have been instrumental in achieving truthful behavior for pareto optimality and envy-freeness(Krysta et al. 2019; Filos-Ratsikas, Frederiksen, and Zhang 2014). When matching pairs of agents and each side has preference over the other, no SP mechanism exists that produces a stable solution (Roth

[3] Preferences are symmetric if Alice likes Bob if and only if Bob likes Alice. It is a natural assumption between the roommates.

[4] The $O^*$ notation hides factors polynomial in input size.

[5] FPT parameterized by $k$ implies it runs in time $g(k)n^{O(1)}$ for some computable function $g$.

1982), although finding one stable matching is easy (Gale and Shapley 1962).

In classical roommate matching, pair of agents are assigned to a room where agents have preference over other agents, but are indifferent between rooms (Knuth 1976; Teo and Sethuraman 1998; Aziz 2013; Irving 1985). Settings where triples of agents, instead of pairs, need to be matched have also been well studied (McKay and Manlove 2021; Lam and Plaxton 2019; Bredereck et al. 2020; Chen and Roy 2022). Maximum welfare and even achieving stability is NP-hard in these settings (Chen and Roy 2022) where each agent has preference over at least a subset of agents (if not all of them). In contrast, in our setting, rooms have no preferences/priorities over the agents.

Our setting where agents have to be matched in pairs to rooms has been studied only very recently. Amongst this work, Chan et al. (2016); Huzhang et al. (2017); Li et al. (2019) all consider maximizing social welfare under different settings. None of the work in this space considers any incentives to manipulate and further consider settings with only additive utilities. Leontief utilities have been widely studied by economists and game theorists, particularly in the context of markets and auctions. These utilities are useful in capturing complementary preferences. Beyond auctions, they have also been studied in settings like resource allocation (Parkes, Procaccia, and Shah 2015; Ghodsi et al. 2011; Brandt et al. 2024) and fair division (Brânzei, Hosseini, and Miltersen 2015; Bogomolnaia and Moulin 2023). We give an extended review of literature in the extended version (Hosseini, Narang, and Roy 2024).

## 2 Model

We study a roommate matching model where agents have preferences over both their roommate and the room. Our model involves a set $\mathcal{N}$ of $2n$ agents and a set $\mathcal{R}$ of $n$ rooms. Agent $i \in \mathcal{N}$ has *valuation functions* $v_i$ and $\widehat{v}_i$, where $v_i : \mathcal{N} \setminus \{i\} \to \mathbb{R}_{\geq 0}$ denotes the "compatibility values" for the other agents and $\widehat{v}_i : \mathcal{R} \to \mathbb{R}_{\geq 0}$ denotes the values for rooms. Under *binary valuations*, $v_i$ and $\widehat{v}_i$ values are either 0 or 1 for each agent $i$. *Symmetric valuations* are where $v_i(j) = v_j(i)$ for each pair of agents $i$ and $j$.

We denote an instance of roommate matching by the tuple $\langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$ where $v$ is the vector of compatibility values $(v_1, v_2, \ldots, v_{2n})$ and $\widehat{v}$ is the vector of valuations for the rooms $(\widehat{v}_1, \widehat{v}_2, \ldots, \widehat{v}_{2n})$. We assume that each room is matched to exactly two agents. Formally, for any distinct $i, j \in \mathcal{N}$ and $r \in \mathcal{R}$, we call $(i, j, r)$ a *triple* and we define a roommate matching as follows.

**Definition 1** (Roommate Matching)**.** *A roommate matching $\mu \subseteq \mathcal{N} \times \mathcal{N} \times \mathcal{R}$ is such that each agent or room $a \in \mathcal{N} \cup \mathcal{R}$ is contained in exactly one element (triple) of $\mu$.*

**Leontief and Additive Utilities.** For the ease of presentation, we use $v_i(\mu)$ and $\widehat{v}_i(\mu)$, respectively, to denote the value of agent $i \in \mathcal{N}$ from their assigned roommate and room under $\mu$, respectively. Given a roommate matching $\mu$, we define the *utility* of an agent $i \in \mathcal{N}$ for $\mu$ to be $u_i(\mu) = f(v_i(\mu), \widehat{v}_i(\mu))$ for some computable function $f$ that we call a utility function. We shall use $u_i(j, r)$ for $i, j \in \mathcal{N}$ and $r \in \mathcal{R}$ to denote the utility of $i$ for being matched to agent $j$ and room $r$. In this paper, the utility function $f$ will be either Leontief or additive[6]. Of the two, the more well-studied one is **additive utilities** where $u_i(\mu) = \widehat{v}_i(\mu) + v_i(\mu)$. The other is the natural model of **Leontief utilities** where $u_i(\mu) = \min\{\widehat{v}_i(\mu), v_i(\mu)\}$ (for example, see (Parkes, Procaccia, and Shah 2015)).

**Mechanism.** Given an instance $\langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$, a mechanism $M$ uses the valuations $v$ and $\widehat{v}$ reported by the agents and implements a specific algorithm to output a matching $\mu$, written as $\mu = M(\langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle)$. We distinguish between a mechanism and the algorithm it deploys only when reasoning about agents' incentives to misreport their valuations. The aim of all the mechanisms we explore is to match agents and rooms in an efficient manner.[7]

**Social Welfare.** Given an instance $I = \langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$, the *social welfare* (SW) of a matching $\mu$ is defined as the sum of the utilities of all agents, that is $\text{SW}(\mu) = \sum_{i \in \mathcal{N}} u_i(\mu)$. A mechanism is social welfare maximizing if it produces a matching with maximum social welfare. Typically, we shall use $\mu^*$ to denote a maximum welfare matching. We say a matching has welfare $\alpha$-SW if its social welfare is at least $\alpha$ fraction of $\text{SW}(\mu^*)$ for some $0 \leq \alpha \leq 1$. We consider the problem of maximizing SW, both independently and in conjunction with strategyproofness.

**Incentives and Strategyproofness.** We say an agent $i \in \mathcal{N}$ *misreports* their valuation if their reported valuation $v'_i$ (resp. $\widehat{v}'_i$) is not their true valuation $v_i$ (resp. $\widehat{v}_i$). For a fixed $i \in \mathcal{N}$, we refer to the set of compatibility valuations of all other agents $\mathcal{N} \setminus \{i\}$ as $v_{-i}$ and the room valuations analogously as $\widehat{v}_{-i}$. Given a mechanism $M$, an agent has an *incentive to misreport* if their utility $u_i(\mu)$ is strictly less than $u_i(\mu')$ where $\mu$ and $\mu'$ are the output of $M$ for reported valuations $(v, \widehat{v})$ and $(v'_i, v_{-i}, \widehat{v}'_i, \widehat{v}_{-i})$, respectively. We say that a mechanism is *strategyproof* (SP) if no agent has an incentive to misreport their preferences.

**Definition 2** (Strategyproofness)**.** *A roommate matching mechanism $M$ is strategyproof, if for any instance $I = \langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$ and any agent $i \in \mathcal{N}$, it holds that $u_i(\mu) \geq \max_{\widehat{v}'_i, v'_i} u_i(\mu')$ for any choice of $v'_i$ and $\widehat{v}'_i$ where $\mu = M(\mathcal{N}, \mathcal{R}, v, \widehat{v})$ and $\mu' = M(\mathcal{N}, \mathcal{R}, \{v'_i, v_{-i}\}, \{\widehat{v}'_i, \widehat{v}_{-i}\})$.*

Our main aim is to develop roommate matching mechanisms that maximize social welfare and are strategyproof under Leontief utilities. We also contrast these results with the case of additive utilities. When we study SP mechanisms under binary (or symmetric) valuations, we will assume that the *mechanism is aware* that valuations are binary (resp. symmetric), so an agent can only misreport with other binary (resp. symmetric) valuation functions.

## 3 Social Welfare Guarantees

We first explore social welfare maximization alone, without considering incentives to manipulate. Chan et al. (2016) show that maximizing social welfare under additive utilities

[6] Some of our results hold for any polynomial-time computable function $f$, as will be stated in the remarks.

[7] We assume complete matchings with no outside options.

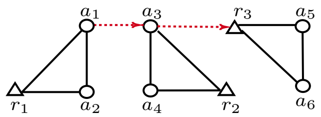

Figure 1: An instance showing that a naive maximal matching cannot guarantee $\alpha$-SW for any $\alpha > 0$ under binary Leontief utilities. Solid black edges show an optimal matching and dashed red edges show a wrong choice of triple.

is NP-hard and give a $^2/_3$-SW algorithm. This algorithm does not extend beyond additive utilities and can result in matchings that are 0-SW for Leontief utilities (see the example in (Hosseini, Narang, and Roy 2024).) We begin by analyzing the SW guarantees provided by maximal matchings.

### 3.1 Maximal Matchings

Maximal matchings provide non-trivial approximations to SW in two-dimensional settings under binary/dichotomous preferences. Prior work in the roommate matching setting has not considered maximal matchings at all. We now discuss different ways of building maximal roommate matchings and their guarantees on SW when agents have binary values. First, observe that a maximal matching constructed by arbitrarily choosing *any* triple could lead to a matching with 0-SW approximation. We shall consider two ways of generating maximal matchings: i) From the preference graph, and ii) from a given set of triples (usually having some specific property, e.g. non-zero SW).

**Naive Maximal Matchings.** Given an instance $I = \langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$ with *binary valuations*, the preference graph $G_I = (X, E)$ is a directed graph with an agent vertex $x_i$ for each agent $i \in \mathcal{N}$ and a room vertex $x_r$ for each room $r \in \mathcal{R}$, i.e., $X = \mathcal{N} \cup \mathcal{R}$. There is a directed edge from an agent vertex $x_i$ to a vertex $x \in X$ if agent $i$ has a non-zero value for the agent/room corresponding to vertex $x$ in $I$.

We can generate a **naive maximal matching** from the preference graph $G_I$ by arbitrarily adding edges from $G_I$ to create matching triples. If both endpoints of the edge are agents, remove both agents' outgoing and incoming edges to/from other agents. Else, if exactly one endpoint is an agent $i$ (the other a room), remove all outgoing edges of $i$ to any room. If selected edge is incident to an edge selected previously, match the agents and room and remove all edges to/from the triple. Once no edges remain in $G_I$, complete the remaining matched edges to triples arbitrarily.

**Leontief Utilities.** We first study the naive maximal matching approach for Leontief and additive utility. This can be arbitrarily bad for non-symmetric valuations under Leontief utility (see the instance depicted in Figure 1).

**Proposition 3.1.** *A naive maximal matching cannot achieve better than* $\alpha$-SW *for any* $\alpha > 0$ *under Leontief utilities.*

The naive approach can be tweaked for binary symmetric Leontief utilities to give a constant approximation guarantee.

**Proposition 3.2.** *There is a maximal matching algorithm that is* $^1/_6$-SW *for binary symmetric Leontief utilities* .

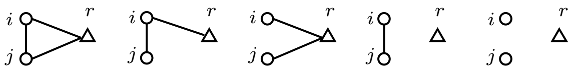

Figure 2: Possible triple structures in a roommate matching problem. The first structure on the left is a triangle (T), the next one is L-shaped (L) and gives non-zero utility to agent $i$. The last three show the cases where both $i$ and $j$ get utility 0 under Leontief.

**Algorithm 1:** Triangle-then-L Maximal Matching

```
Input: Roommate Matching instance $\langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$
Output: A matching $\mu$
Initialize matching $\mu \leftarrow \varnothing$ ;
while there exists an unmatched triple in $\mathcal{N} \cup \mathcal{R}$ do
    Let the set of unmatched triples
      $P = \{(i, j, r) | i, j \in \mathcal{N}, r \in \mathcal{R}\}$;
    $(i, j, r) \leftarrow \arg\max_{(i,j,r) \in P} (u_i(j, r) + u_j(i, r))$
                ▷ Tie-breaking is arbitrary;
    Update $\mu \leftarrow \mu \cup (i, j, r)$;
    Update $\mathcal{N} \leftarrow \mathcal{N} \setminus \{i, j\}$; $\mathcal{R} \leftarrow \mathcal{R} \setminus \{r\}$;
```

**Additive utilities.** We find that the naive maximal matching performs much better for binary additive utilities.

**Proposition 3.3.** *A naive maximal matching is* $^1/_4$-SW *under binary additive utilities.*

**Matching with Triangles and L-Shapes.** Proposition 3.1 illustrates the challenges of adopting graph-matching algorithms to roommate matching instances, especially under Leontief. To develop a polynomial-time algorithm that achieves a constant approximation of welfare, we introduce triple structures according to their contribution to welfare.

**Triple Structures.** Given an instance of the roommate matching problem, a triple $(i, j, r)$ can be represented as a graphical structure in the preference graph. These structures are illustrated graphically in Figure 2. A triple may form a triangle match (T) if (i) both agents like each other and (ii) both like the matched room. If only one edge between an agent and a room is missing in a triangle match, that is both agents like each other, but only one likes the room, then we say the match is an L-shaped (L) match.

**Definition 3.** *A matching is L/T-matching if it only consists of L-shapes and triangles. An L/T-matching $M$ is maximal if it admits no additional L/T triples.*

**Theorem 3.4.** *An L/T-maximal matching guarantees* $^1/_6$-SW *under binary Leontief utilities.*

**Triangle-then-L Maximal Matching** Our previous analysis shows that it is important to not eliminate triples with more social welfare too early. Thus, we now propose an algorithm based on repeatedly choosing the highest utility triple (triangles first then Ls) produces a maximal matching. This approach guarantees $^1/_3$-SW, for unrestricted valuations and all utility types.

**Theorem 3.5.** *A Triangle-then-L maximal matching is* $^1/_3$-SW.

**Remark 3.1.** *The approximation guarantee of the Triangle-then-L mechanism (Algorithm 1) does not depend on how the agents combine the values they receive from their room and partner. Consequently, this result is applicable to any computable utility function (e.g., submodular or monotonic).*

## 3.2 Computational Complexity

We now show that maximizing social welfare is APX-hard, even under binary symmetric Leontief utilities. We present a factor preserving reduction from 3SAT. The construction will be similar to the standard reduction from 3SAT to 3-DIMENSIONAL MATCHING (Garey and Johnson 1979). Interestingly, we show that it is hard even if there are only Ls.

**Theorem 3.6.** *Finding a maximum* SW *roommate matching is APX-hard under binary Leontief utilities.*

In particular, the instance creates a setting where all triples have utility at most 1. That is a maximum welfare matching would essentially be a maximal matching of Ls with maximum size. This is an important difference from the reduction in (Chan et al. 2016) which proves NP-hardness with mixtures of Ls and Ts. In fact, our reduction shows that the problem is NP-hard even when each agent has degree three in the preference graph, i.e., the problem in paraNP-hard[8] with respect to degree of agents.

**Corollary 3.7.** *Maximizing* SW *is NP-hard under Leontief utilities in the marriage setting even when agents' valuations are binary, symmetric, and each agent likes at most one room and at most two other agents.*

In light of this result, we now show an approximation algorithm via the SET PACKING problem irrespective of agents' underlying utility function.

**Welfare Approximation via 3-SET PACKING** In WEIGHTED 3-SET PACKING we are given a universe $\mathcal{U}$ and a collection $\mathcal{C}$ of weighted 3-sized sets over $\mathcal{U}$, and the goal is to obtain a subset of $\mathcal{C}$ of pairwise disjoint sets with maximum total weight. It is well-known that 3-SET PACKING generalises 3-dimensional matching. We show that our problem reduces to weighted 3-SET PACKING and can utilize the prior works on its approximation (Thiery and Ward 2023) and exact (Zehavi 2023) algorithms to obtain the following results.

**Proposition 3.8.** *There exists an approximation preserving reduction from SW maximization in roommate matchings to* WEIGHTED 3-SET PACKING. *This gives a polynomial time algorithm that is* 0.559-SW *under Leontief utilities.*

**Corollary 3.9.** *There exists an algorithm to maximize the* SW *under Leontief utilities for arbitrary valuations that runs in time* $O^*(8.097^n)$.

# 4 Strategyproof Mechanisms

The algorithms given in Section 3.2 are not SP. We now show upper and lower bounds on the approximation guarantees possible for SP mechanisms under both Leontief and additive utilities.

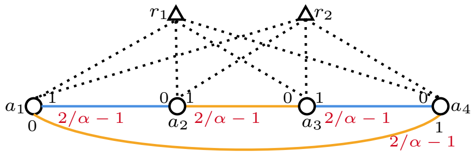

Figure 3: Agent compatibility values and misreports (in red) for the instance described in Theorem 4.1. The two colors on the edges correspond to the two types of maximum SW matchings for this instance.

## 4.1 Impossibility Results

We start by finding that strategyproofness is often incompatible with welfare maximization under some natural utility models (e.g. general valuations, additive utilities). These impossibilities motivate the study of SP mechanisms for restricted valuations (e.g. binary valuations).

**Theorem 4.1.** *There exists no strategyproof mechanism that can guarantee* $\alpha$-SW *for any* $0 < \alpha \leq 1$, *for general (unrestricted) valuations under either Leontief or additive utilities.*

In contrast to Theorem 4.1, under binary valuations, approximate SW mechanisms can be strategyproof. We now provide an upper bound to these approximations for binary additive utilities.

**Theorem 4.2.** *There exists no strategyproof mechanism that can guarantee* $\alpha$-SW *for any* $\alpha > {}^2\!/_3$ , *for binary valuations under additive utilities.*

**Corollary 4.3.** *There exists no strategyproof mechanism that can guarantee* $\alpha$-SW *for any* $\alpha > {}^3\!/_4$, *for binary and symmetric valuations under additive utilities.*

A typical approach to finding SP mechanisms is to use serial dictatorship where agents pick a roommate-room pair one by one from the pool of unassigned items according to a fixed priority ordering. In (Hosseini, Narang, and Roy 2024) we discuss the standard serial dictatorship mechanisms and its various extensions and show how they either fail to give a guarantee on the SW or fail to be strategyproof.

**Remark 4.1.** *The naive serial dictatorship mechanism is strategyproof but cannot guarantee* $\alpha$-SW *for any* $\alpha > 0$ *under additive or Leontief utilities.*

## 4.2 Maximum Welfare SP Mechanisms

We now consider strategyproofness under binary Leontief utilities. In this setting, agents are constrained in how they can change the apparent social welfare of a matching by misreporting their values. Surprisingly, we show that unlike the additive setting, for binary Leontief utilities, there exist SP mechanisms that guarantee maximum SW. We shall design two such mechanisms. First, in Algorithm 2 we find all maximum welfare matchings and then select a matching from them in a manner that ensures strategyproofness. We then refine this approach to Algorithm 3 which is FPT parameterized by the number of agents. Both these mechanism rely

[8] Unless P=NP, we cannot expect to get an algorithm that runs in $n^d$ where $d$ is the degree of the agents.

**Algorithm 2:** Welfare Set Reduction Mechanism

```
Input: An instance $\langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$ with binary
       valuations
Output: A matching $\mu$ with maximum SW
Initialize $S_0$ to be the set of all roommate matchings
  with maximum SW ;
Pick an ordering on agents $\sigma$;
for $t = 1$ to $2n$ do
    Select $j$ to be the $i^{\text{th}}$ agent on $\sigma$;
    $S_i \leftarrow \arg\max_{\mu \in S_{i-1}} u_j(\mu)$   ▷ Reduce based on
      $j$;
Select $\mu$ arbitrarily from $S_{2n}$;
```

on the structure provided by binary Leontief utilities. We begin with a simple observation that constrains the way agents can manipulate under binary Leontief utilities.

**Observation 4.2.** *Under binary Leontief utilities, consider agent $i \in \mathcal{N}$ and matchings $\mu_1$ and $\mu_0$ where utilities of agent $i$ are 1 and 0, respectively. Here, when all other agents $N \setminus \{i\}$ are honest, by misreporting $v_i$ or $\widehat{v}_i$:*

1. *agent $i$ cannot increase the apparent* SW *of $\mu_1$ and*
2. *agent $i$ cannot decrease the apparent* SW *of $\mu_0$ .*

While this does not guarantee that there are never any incentives to lie, it proves to be useful in developing SP mechanisms that maximize social welfare.

From Theorem 3.6, any maximum SW mechanism cannot run in poly time even for binary valuations. Consequently, we first consider a brute force approach where we look at all possible matchings and choose those that maximize SW. Tie-breaking rules are crucial to achieve strategyproofness. We show in (Hosseini, Narang, and Roy 2024) that there exist methods of tie-breaking which give some agent an incentive to lie about their preferences. A careful consideration of tie-breaking through a priority ordering over agents recovers strategyproofness.

**Welfare Set Reduction Mechanism.** Algorithm 2 proceeds by first computing the set of all roommate matchings with maximum SW $S_0$. It then fixes an *arbitrary* ordering of agents $\sigma$. We use $i = \sigma(j)$ to denote the position of the agent $j$ in the ordering. We sequentially define the set $S_i = \arg\max_{\mu \in S_{i-1}} u_j(\mu)$ to be the subset of $S_{i-1}$ containing only those matchings that maximize agent $j$'s utility where $i = \sigma(j)$. Finally, the mechanism returns a matching from the set $S_{2n}$. We first give an example of the execution.

**Example.** *We use the instance in Theorem 4.1 (depicted in Figure 3) to show an execution of Algorithm 2. Let the ordering be $\sigma = (a_1, a_2, a_3, a_4)$. Recall that the max* SW *matchings are of type $\mu = \{(a_1, a_2, \cdot), (a_3, a_4, \cdot)\}$ or as $\mu' = \{(a_1, a_4, \cdot), (a_2, a_3, \cdot)\}$. Therefore, in $S_0$ agents are matched either as in $\mu$ or as in $\mu'$ with all possible room assignments. Agent $a_1$ gets utility 1 only when matched to $a_2$. Thus, $a_1$ deletes the matchings where agents are matched as in $\mu'$ from $S_0$ to construct $S_1$.*

*Agent $a_2$ gets utility 0 when matched to $a_1$ and consequently in both the matchings in $S_1$ utility of $a_2$ is 0. Thus, $S_2 = S_1$. Analogously, $a_3$ gets utility 1 from both matchings in $S_2$, hence, $S_3 = S_2$. Finally, by similar arguments, $S_4$ will contain the two matchings from $S_1$. Consequently, the algorithm arbitrarily picks between them. This is illustrated in Figure 4.*

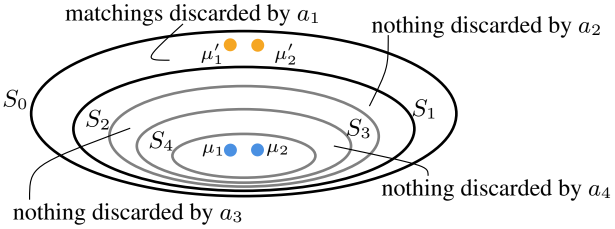
matchings discarded by $a_1$
nothing discarded by $a_2$
$\mu'_1$
$\mu'_2$
$S_0$
$S_1$
$S_2$
$S_3$
$S_4$
$\mu_1$
$\mu_2$
nothing discarded by $a_4$
nothing discarded by $a_3$

Figure 4: Algorithm 2 on the instance depicted in Figure 3.

**Theorem 4.4.** *The Welfare Set Reduction mechanism (Algorithm 2) returns a matching with maximum* SW *and is strategyproof under binary Leontief utilities.*

*Proof Sketch.* Clearly, Algorithm 2 always returns a maximum welfare roommate matching. To show that the mechanism is strategyproof for any agent $j \in \mathcal{N}$ s.t. $\sigma(j) = i$, we do a careful case analysis based on whether $S_{i-1}$ or $S_0$ contain a matching where $j$ gets utility 1. We show that:
(1) if $S_{i-1}$ does contain one such matching then so would $S_i$ and as a result so would $S_{2n}$.
(2) if $S_0$ contains matching where $j$ gets utility 1, we can invoke Observation 4.2.2 to show that $j$ cannot benefit by misreporting their values. Finally,
(3) if $S_0$ contains at least one matching where $j$ gets utility 1 but $S_{i-1}$ does not, there must exist a set of agents with higher precedence to $j$ who eliminate each such matching. We now can invoke Observation 4.2 to show that $j$ (i) cannot decrease the perceived SW of a matching in $S_0$ and (ii) cannot add matchings to $S_0$ and ensure that $j$ gets utility 1 from the matchings in $S_{2n}$. □

**Remark 4.3.** *Observe that to show Algorithm 2 is strategyproof, we need to use Observation 4.2 which holds only for binary Leontief utilities. Thus, we can not hope to find a strategyproof algorithm for more general valuations.*

Both Observation 4.2 and the proof of Theorem 4.4 rely on the fact that agents utilities can only be 0 or 1 and are not contingent on the utilities being specifically Leontief. Consequently, the following holds.

**Corollary 4.5.** *Let $f$ be a utility function such that for each agent $i \in \mathcal{N}$, the utility $u_i(\mu) \in \{0, 1\}$. Then there exists a welfare maximizing matching $\mu$ that is strategyproof under utility function $f$.*

Although the Welfare Set Reduction mechanism (Algorithm 2) shows the existence of a SP mechanism, it is far from being efficient. It parses the set of all maximum welfare matchings which can have size up to $n^{2n}$. We now build on the ideas in this mechanism to produce a strategyproof mechanism that is parameterized by the number of agents.

**Precedence-Based Search Mechanism.** In Algorithm 3, we begin by fixing an ordering $\sigma$ of the agents. We then iterate

**Algorithm 3:** Precedence-Based Search Mechanism

```
Input: $I = \langle \mathcal{N}, \mathcal{R}, v, \widehat{v} \rangle$ with binary valuations
Output: A matching $\mu$ with maximum SW
Let $\sigma$ be a fixed arbitrary ordering of $\mathcal{N}$, and $k \leftarrow 2n$;
while $k \geq 1$ do
    Initialize $T_1 \cdots T_{\binom{2n}{k}}$ s.t. $T_t$ is the $t^{\text{th}}$ highest
      precedence subset of $\mathcal{N}$ of size $k$ under $\sigma$;
    for $t = 1$ to $\binom{2n}{k}$ do
        $\tau \leftarrow \{(i, j, r) | i \neq j \in \mathcal{N}, r \in \mathcal{R}$ s.t.
          $u_i(\{j, r\}) = 1 \Leftrightarrow i \in T_t$, $u_j(\{i, r\}) = 1 \Leftrightarrow$
          $j \in T_t\}$ ;
        $\mu \leftarrow$ 3-SET PACKING$(\mathcal{N} \cup \mathcal{R}, \tau)$;
        If $|\mu| = n$ then Return $\mu$;
    $k \leftarrow k - 1$;
Return an arbitrary matching;
```

over the (possible) value of the maximum SW starting from $2n$ to 1. For each value $k$ of SW, we consider the various subsets of agents $\mathcal{N}$ of size $k$ in the order implied by $\sigma$.[9] For each subset, we check if there exists a matching that gives utility one to each agent in this subset and gives utility zero to each agent not in the subset.

To this end, we construct an instance of (unweighted) 3-set packing $(\mathcal{U}, \tau)$ for a subset $T$ as follows. We define $\mathcal{U}$ to be $\mathcal{N} \cup \mathcal{R}$. We choose $\tau$ to be the set of triples $(i, j, r)$ s.t. either agent gets utility 1 from the triple if and only if they are in the subset $T$. If we find a 3-set packing of size $n$, we return its triples as a roommate matching. We show that there exists a 3-set packing of size $n$ in $(\mathcal{U}, \tau)$ if and only if a roommate matching exists which gives the agents in $T$ utility 1 and those not in $T$ utility 0. If no such packing is found for T, we continue to the next iteration. If no set packing exists for any choice of $k$ and $T$, we return an arbitrary matching, inferring that all roommate matching have SW of 0.

**Theorem 4.6.** *There exists a strategyproof algorithm to find a maximum* SW *roommate matching that runs in time* $O^*(c^n)$ *where* $c > 0$ *is a constant.*

*Proof Sketch.* Observe that if the algorithm returns a matching in the iteration $k = w$, then the SW of the matching is $w$. Using the fact that we started from the largest SW as value of $k$ and using the construction of $\tau$, we can argue that Algorithm 3 always finds a maximum welfare matching. Arguing for strategyproofness requires an intricate analysis.

We establish strategyproofness by showing that for the same precedence order $\sigma$, Algorithm 2 and Algorithm 3 return the same matching. As the mechanism described in Algorithm 2 is SP, this would imply that the mechanism in Algorithm 3 is also SP. It suffices to show that the matching $\mu$ returned by Algorithm 3 is in $S_{2n}$. We induct on the agents based on $\sigma$. For each $S_i$ we show that the matching $\mu$ is returned by Algorithm 3 if and only if $\mu \in S_i$.

Let $\mu \in S_{i-1}$ and the $i$th agent in $\sigma$ is $j$, i.e., $\sigma(j) = i$. If agent $j$ gets utility 1 from $\mu$, then clearly $\mu \in S_i$. If $j$ gets utility 0 from $\mu$, we show that any matching $\mu_j$ where $j$ gets utility 1 is not in $S_{i-1}$. That is $\mu'$ either has lower SW or if it is of equal welfare, $\mu'$ gives utility 0 to a higher precedence agent. Consequently, $\mu'$ would be eliminated in Algorithm 2 before $S_i$. Thus, the matching returned by Algorithm 3, $\mu$ would be in $S_i$. Thus, by induction we can show $\mu \in S_{2n}$. Based on the 3-set packing algorithm use, the running time for our algorithm would be $O^*(13.36^n)$ (randomized) or $O^*(27^n)$ (deterministic). □

### 4.3 Polynomial Time Mechanisms

We now demonstrate different polynomial time mechanisms that are strategyproof and give non-trivial guarantees on SW. We first attempt this via a serial dictatorship approach. While serial dictatorship is a well used approach to achieve strategyproofness, in its typical form it proves to be ineffective in providing any meaningful guarantee on SW. We discuss this at length in (Hosseini, Narang, and Roy 2024). However, we can adapt the L/T maximal matching approach we discussed in a serial dictatorship fashion to get a SP mechanism, described in Algorithm 4 in (Hosseini, Narang, and Roy 2024), that is $1/6$-SW for binary Leontief utilities.

**Theorem 4.7.** *There exists a strategyproof mechanism that is* $1/6$-SW *under binary Leontief utilities.*

In fact, an analogous mechanism (Algorithm 5 (Hosseini, Narang, and Roy 2024)) also gives a similar guarantee for additive utilities.

**Theorem 4.8.** *The Welfare Prioritized Serial Dictatorship mechanism (Algorithm 5) is strategyproof under binary additive utilities. Further, it is* $1/7$-SW.

For symmetric preferences, this gives a better guarantee.

**Corollary 4.9.** *When agent preferences are symmetric Algorithm 5 is* $1/6$-SW *for instances with binary additive utilities.*

**Triangle-then-L Mechanism.** We find that the Triangle-then-L mechanism provides us a SP mechanism with a better approximation guarantee for binary Leontief utilities. Recall that this mechanism is $1/3$-SW.

**Theorem 4.10.** *The Triangle-then-L mechanism (Algorithm 1) is strategyproof under binary Leontief utilities.*

## 5 Concluding Remarks

This paper initiates the study of strategyproofness and Leontief utilities in the setting of roommate matchings. Our work paves the way for further work on this topic. Future directions include finding better approximation algorithms for Leontief utilities and/or polynomial time strategyproof mechanisms with better SW guarantees for binary valuations. Another interesting direction is characterizing the set of strategyproof mechanisms or weakening the strateyproofness requirement (e.g. restricting manipulation) to improve the welfare.

[9] For two subsets $T$ and $T'$ of the same size, we say that $T$ has higher precedence than $T'$, if there is an agent $i \in T \setminus T'$ which has higher precedence than all agents in $T' \setminus T$.

## Acknowledgments

This research was supported in part by NSF Awards IIS-2144413 and IIS-2107173 and the NSF-CSIRO grant on "Fair Sequential Collective Decision-Making." We thank the anonymous reviewers for their helpful comments.

## References

Aziz, H. 2013. Stable marriage and roommate problems with individual-based stability. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, 287–294.

Bogomolnaia, A.; and Moulin, H. 2023. Guarantees in fair division: general or monotone preferences. *Mathematics of Operations Research*, 48(1): 160–176.

Brandt, F.; Greger, M.; Segal-Halevi, E.; and Suksompong, W. 2024. Coordinating Charitable Donations.

Brânzei, S.; Hosseini, H.; and Miltersen, P. B. 2015. Characterization and computation of equilibria for indivisible goods. In *Algorithmic Game Theory: 8th International Symposium, SAGT 2015*, 244–255. Springer.

Bredereck, R.; Chen, J.; Finnendahl, U. P.; and Niedermeier, R. 2020. Stable roommates with narcissistic, single-peaked, and single-crossing preferences. *Autonomous agents and multi-agent systems*, 34: 1–29.

Chan, P.; Huang, X.; Liu, Z.; Zhang, C.; and Zhang, S. 2016. Assignment and pricing in roommate market. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 446–452.

Chen, J.; and Roy, S. 2022. Multi-Dimensional Stable Roommates in 2-Dimensional Euclidean Space. In *30th Annual European Symposium on Algorithms, ESA*, volume 244, 36:1–36:16.

Dolev, D.; Feitelson, D. G.; Halpern, J. Y.; Kupferman, R.; and Linial, N. 2012. No justified complaints: On fair sharing of multiple resources. In *proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 68–75.

Filos-Ratsikas, A.; Frederiksen, S. K. S.; and Zhang, J. 2014. Social welfare in one-sided matchings: Random priority and beyond. In *International Symposium on Algorithmic Game Theory*, 1–12. Springer.

Gale, D.; and Shapley, L. S. 1962. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1): 9–15.

Gan, J.; Li, B.; and Li, Y. 2023. Your College Dorm and Dormmates: Fair Resource Sharing with Externalities. *Journal of Artificial Intelligence Research*, 77: 793–820.

Garey, M. R.; and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.

Ghodsi, A.; Zaharia, M.; Hindman, B.; Konwinski, A.; Shenker, S.; and Stoica, I. 2011. Dominant resource fairness: fair allocation of multiple resource types. In *Proceedings of the 8th USENIX conference on Networked Systems Design and Implementation*, 323–336.

Gibbard, A. 1973. Manipulation of voting schemes: a general result. *Econometrica: Journal of the Econometric Society*, 587–601.

Green, J.; and Laffont, J.-J. 1979. *Incentives in public decision-making*. Elsevier North-Holland.

Hosseini, H.; Narang, S.; and Roy, S. 2024. Strategyproof Matching of Roommates to Rooms. *ArXiv preprint*.

Huzhang, G.; Huang, X.; Zhang, S.; and Bei, X. 2017. Online Roommate Allocation Problem. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 235–241.

Hylland, A.; and Zeckhauser, R. 1979. The efficient allocation of individuals to positions. *Journal of Political economy*, 87(2): 293–314.

Irving, R. W. 1985. An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms*, 6(4): 577–595.

Knuth, D. E. 1976. *Mariages stables et leurs relations avec d'autres problemes combinatoires: introduction a l'analysis mathematique des algorithmes*. Les Presses de l'Université de Montréal.

Krysta, P.; Manlove, D.; Rastegari, B.; and Zhang, J. 2019. Size Versus Truthfulness in the House Allocation Problem. *Algorithmica*, 81: 3422–3463.

Lam, C.-K.; and Plaxton, C. G. 2019. On the existence of three-dimensional stable matchings with cyclic preferences. In *International Symposium on Algorithmic Game Theory*, 329–342. Springer.

Leontief, W. W. 1965. The structure of the US economy. *Scientific American*, 212(4): 25–35.

Li, Y.; Jiang, Y.; Wu, W.; Jiang, J.; and Fan, H. 2019. Room allocation with capacity diversity and budget constraints. *IEEE Access*, 7: 42968–42986.

McKay, M.; and Manlove, D. 2021. The Three-Dimensional Stable Roommates Problem with Additively Separable Preferences. In *International Symposium on Algorithmic Game Theory*, 266–280.

Parkes, D. C.; Procaccia, A. D.; and Shah, N. 2015. Beyond dominant resource fairness: Extensions, limitations, and indivisibilities. *ACM Transactions on Economics and Computation (TEAC)*, 3(1): 1–22.

Roth, A. E. 1982. The economics of matching: Stability and incentives. *Mathematics of operations research*, 7(4): 617–628.

Satterthwaite, M. A. 1975. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2): 187–217.

Teo, C.-P.; and Sethuraman, J. 1998. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23(4): 874–891.

Thiery, T.; and Ward, J. 2023. An Improved Approximation for Maximum Weighted k-Set Packing. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1138–1162. SIAM.

Zehavi, M. 2023. Forgetfulness Can Make You Faster: An O*(8.097 k)-time Algorithm for Weighted 3-set k-packing. *ACM Transactions on Computation Theory*, 15(3-4): 1–13.

Zhou, L. 1990. On a conjecture by Gale about one-sided matching problems. *Journal of Economic Theory*, 52(1): 123–135.