Knowledge-Based Stable Roommates Problems

Müge Fidan

Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey

Abstract

The Stable Roommates problems are characterized by the preferences of agents over other agents as roommates. A solution is a partition of the agents into pairs that are acceptable to each other (i.e., they are in the preference lists of each other), and the matching is stable (i.e., there do not exist any two agents who prefer each other to their roommates, and thus block the matching). This study focuses on a human-centered and computationally-challenging interdisciplinary problem of the Stable Roommates problem and its variations. Motivated by real-world applications, and considering that stable roommates problems do not always have solutions, the goal is to develop novel computational methods to solve these problems, that are not only computationally efficient but also applicable in real-world to benefit humans.

Keywords

stable roommates problem, answer set programming, declarative problem solving

1. Introduction

The Stable Roommates problem [1] (SR) is a matching problem (well-studied in Economics and Game Theory) characterized by the preferences of an even number n of agents over other agents as roommates: each agent ranks all others in strict order of preference. A solution to SR is then a partition of the agents into pairs that are *acceptable* to each other (i.e., they are in the preference lists of each other), and the matching is *stable* (i.e., there exist no two agents who prefer each other to their roommates, and thus *block* the matching).

Table 1
Summary of the complexities of SR problems

Problem	Complexity
SR	P [2]
SRI	P [3]
SRTI (super)	P [4]
SRTI (strong)	P [5] [6]
SRT (weak) (and thus SRTI (weak))	NP-complete [7, Thm 1.1, Prop 2.2]
SRTI (weak)	NP-complete* [8, Thm 5]
Egalitarian SR	NP-hard [9, Thm 8.3]
Egalitarian SRI	NP-hard* [10, Cor 4]
Rank Maximal SR (and thus Rank Maximal SRTI)	NP-hard [11]
Almost SR (and thus SRT (weak))	NP-hard [12, Thm 1]
Almost SRI	NP-hard* [13, Thm 1]

* for short lists of size ≤ 3

SR is an interesting computational problem, not only due to its applications (e.g., for pairing in large-scale chess competitions [14], for campus house allocation [15], pairwise kidney exchange [16], creating partnerships in P2P networks [17]) but also due to its computational properties (Table 1). Therefore, variations of SR have been investigated in the literature.

SR is studied with incomplete preference lists (SRI) [3], with preference lists including ties (SRT) [7], and with incomplete preference lists including ties (SRTI) [4]. While SR and SRI are tractable [2, 3],

ICLP DC 2025: 21st Doctoral Consortium on Logic Programming, September 2025, Rende, Italy.
mugefidan@sabanciuniv.edu (M. Fidan)
0000-0002-7646-6531 (M. Fidan)

SRT and SRTI are intractable under weak stability [7, 8].

Optimization variants of SR are also studied to find more fair stable solutions. For instance, Egalitarian SR aims to maximize the total satisfaction of preferences of all agents; it is NP-hard [9]. Rank Maximal SRI aims to maximize the number of agents matched with their first preference, and then, subject to this condition, to maximize the number of agents matched with their second preference, and so on; it is also NP-hard [11].

As first noted by Gale and Shapley [1], unlike the Stable Marriage problem (SM), there is no guarantee to find a solution to every SR problem instance (i.e., there might be no stable matching). When an SR instance does not have a stable solution, variations of SR have been studied to find a good-enough solution. For instance, Almost SR aims to minimize the total number of blocking pairs (i.e., pairs of agents who prefer each other to their roommates); it is NP-hard [12].

Motivated by the real-world applications and the computational challenges, our study aims to develop methods for finding personalized, fair, relaxed, explainable solutions for SRTI taking into account further knowledge. To find such solutions, we introduce novel knowledge-based methods, and utilize Answer Set Programming (ASP) [18] based on answer set semantics [19, 20]. We use the ASP solver Clingo [21] for declarative problem solving.

2. Stable Roommates problem with Ties and Incomplete Lists (SRTI)

We define SRTI as in our earlier work [22]. Let A be a finite set of agents. For every agent $x \in A$, let $A_x \subseteq A \backslash \{x\}$ be a set of agents that are *acceptable* to x as roommates. For every y in A_x, we assume that x prefers y as a roommate compared to being single.

Let $\prec_x$ be a partial ordering of x's preferences over A_x where incomparability is transitive. We refer to $\prec_x$ as agent x's preference list. For two agents y and z in A_x, we denote by $y \prec_x z$ that x prefers y to z. In this context, ties correspond to indifference in the preference lists: an agent x is *indifferent* between the agents y and z, denoted by $y \sim_x z$, if $y \not\prec_x z$ and $z \not\prec_x y$. We denote by $\prec$ the collection of all preference lists.

A *matching* for a given SRTI instance is a function $M : A \mapsto A$ such that, for all $\{x, y\} \subseteq A \times A$ such that $x \in A_y$ and $y \in A_x$, $M(x) = y$ if and only if $M(y) = x$. If agent x is mapped to itself, we then say he/she is *single*.

A matching M is *blocked* by a pair $\{x, y\} \subseteq A \times A$ $(x \neq y)$ if

- B1 both agents x and y are acceptable to each other,
- B2 x is single with respect to M, or $y \prec_x M(x)$, and
- B3 y is single with respect to M, or $x \prec_y M(y)$.

A matching for SRTI is called *stable* if it is not blocked by any pair of agents.

Student x	$\prec_x$
a	$\langle b,\ d,\ c\rangle$
b	$\langle c, \{a,\ d\}\rangle$
c	$\langle a,\ b,\ d\rangle$
d	$\langle a,\ c\rangle$

Table 2
A SRTI instance defined over an agent set A = {a, b, c, d}. M_1=$\{\{a,b\},\{c,d\}\}$ is not stable matching, M_2=$\{\{a,d\},\{b,c\}\}$ is stable matching.

Example 1. *Suppose that a set $\{a,b,c,d\}$ of students applies for an accommodation at a dormitory. For that, each student provide their preferences over others. Table 2 shows preferences $\prec_x$ of these students. For the given SRTI instance, M_1=$\{\{a,b\},\{c,d\}\}$ is not stable matching since $\{c,d\}$ blocks this matching, while M_2=$\{\{a,d\},\{b,c\}\}$ is stable matching since it is not blocked by any pair of students.*

3. Solving SRTI and its hard variants

In our earlier work [22], we have developed a formal framework, called SRTI-ASP, that is flexible enough to provide solutions to all variations of SR mentioned above, including the intractable decision/optimization versions: SRT, SRTI, Egalitarian SRTI, Rank Maximal SRTI, Almost SRTI. For each variation of SR, given a problem instance, SRTI-ASP returns a solution (or all solutions) if one exists; otherwise, it returns that the problem does not have a solution. We have proved that SRTI-ASP is sound and complete [22, Thm 1].

We have performed experiments to evaluate SRTI-ASP over different sizes of randomly generated SRTI instances in term of scalability of its variants (i.e., SRI, SRTI, Almost SRTI, optimization variants of SRTI). We have also compared SRTI-ASP with two related approaches over different sizes of randomly generated SRI instances: SRI-CP [23] solves SRI using constraint programming, and SRI-AF (extending from SR-AF [24]) solves SRI using an argumentation framework. Comparisons with SRI-CP and SRI-AF has helped us to better observe the flexibility of SRTI-ASP due to elaboration tolerant ASP representations. It is easier to extend SRTI-ASP to address different variations of SR, while SRI-CP and SRI-AF require further studies in modeling as well as implementation.

While the optimization variants of SRTI are based on domain-independent measures (e.g., Egalitarian), in real-world applications (e.g., assigning students as roommates at a dormitory), there are also domain-dependent criteria based on further knowledge, such as the habits and desires of the students. With this motivation, in our earlier work [25], we have developed two knowledge-based methods for SRTI: Personalized-SRTI and Most-SRTI.

4. Knowledge-based SRTI

In real-world applications, we have observed that the preference lists are often too short or empty (e.g., for freshmen students). Thus, for SRI and SRTI instances, it gets harder to find a stable matching: students who are not in the preference lists of each other are considered unacceptable to each other, and thus they cannot be matched. In such cases, we still need to find a matching that is "good-enough."

To find good-enough matchings, in addition to Almost SRTI, we have proposed a novel approach, called Knowledge-Based SRTI [25], that extends the preference lists by identifying "suitable" candidates by considering domain-specific knowledge, e.g., the habits and desires of the students. We have introduced two knowledge-based methods for SRTI: Personalized-SRTI and Most-SRTI.

Personalized-SRTI infers a new type of preference ordering by considering (i) the importance of each domain-specific criterion for each agent (e.g., one student may give more importance to sleeping habits whereas another student may give more importance to smoking habits), and (ii) the agents' preferred choices for each domain-specific criterion (e.g., whether a student prefers a roommate who does not smoke). Then, for each agent, it appends this new type of criteria-based personalized preference list inferred from the habitual preferences of the agent, to the end of the preference list given by the agent.

Table 3
A Personalized-SRTI instance defined over an agent set A = {a, b, c, d, e}, a criteria list B = ⟨"smoking", "room environment", "sleep habits", "study habits"⟩, and the following choice lists for each criterion, C_1 = ⟨"Smoker","Non-smoker"⟩, C_2 = ⟨"Quiet","Social","Social and quiet"⟩, C_3 = ⟨"Goes to bed early","Goes to bed before midnight","Goes to bed after midnight⟩, C_4 = ⟨"Studies in the room","Studies out of the room","Studies in and out of the room"⟩.

Student x	$\prec_x$ (given)	Profile P_x	Weight list W_x	$\prec'_x$ (inferred)
a	$\langle e\rangle$	⟨2, 1, 1, 2⟩	⟨0,0,5,0⟩	$\langle b\rangle$
b	$\langle e\rangle$	⟨1, 2, 1, 1 ⟩	⟨1,3,0,4⟩	
c	$\langle b\rangle$	⟨2, 1, 3, 2 ⟩	⟨5,0,4,4⟩	$\langle\{a,\ e\}\rangle$
d	$\langle b\rangle$			
e	$\langle d\rangle$	⟨ 2, 3, 1, 2 ⟩	⟨0,3,0,0⟩	

Example 2. *Consider the example shown in Table 3. The preference profile P_c for student c is ⟨2, 1, 3, 2⟩ where $B = \langle$"smoking", "environment", "sleep habits", "study habits"⟩. According to P_c, c prefers a roommate that is a "Non-smoker", "Quiet","Goes to bed after midnight","Studies out of the room". The weight list W_c for c is ⟨5,0,4,4⟩: the most important criterion for c is "smoking", and the "room environment" criterion is not important. We define a sorted profile $P_x^{'}$ for each agent $x \in A$, with respect to P_x and W_x. According to our definition, the sorted profile for c $P_c^{'} = \langle\{(2,$"smoking"$)\},\{(3,$"sleep habits"$),\ (2,$"study habits"$)\}\rangle$. Hence, a and e are choice-acceptable for c. With respect to $P_c^{'}$, c is indifferent between a and e since their choices are the same. Then, the inferred preference list $\prec_c^{'}$ is $\langle\{a,\ e\}\rangle$.*

Most-SRTI introduces a new incremental definition of a stable matching, by considering (i) the ordering of the most preferred criteria (e.g., identified by large surveys) and (ii) the agents' preferred choices for each domain-specific criterion, with the motivation that the agents with close choices are matched. Most-SRTI aims to compute such most preferred criteria based stable matchings.

Utilizing these methods, we have extended SRTI-ASP to consider domain-specific knowledge about each individual's preferences about a set of criteria, and about the diversity preferences of dormitories and schools. We have experimentally evaluated our methods to understand their scalability, usefulness and applicability. We have observed the usefulness of this approach in computing not only good-enough matchings that are stable with respect to extended lists, but also more personalized, more diverse and/or more inclusive matchings.

Details of the definitions, the experimental evaluations and the real-world application of our knowledge-based methods are presented in our journal paper [25].

5. Relaxed SRTI

Although our knowledge-based methods that extend the preference lists of agents with suitable candidates by considering domain-specific knowledge are shown to be useful, it might not always be sufficient to find suitable candidates because of relevant but inaccessible knowledge about the students, due to ethical and fairness concerns. For instance, although people with similar political opinions tend to get along better, it would not be appropriate to ask questions, in a roommate search questionnaire, about students' political opinions. To address this limitation, we have been studying two different extensions to relax the problem, by further extending preference lists (k-Personalized-SRTI) and by considering weaker notions of stability (Sticky-SRTI).

k-Personalized-SRTI. Students often meet other students through mutual friends, e.g., the friends of their friends, and sometimes get along with each other. Based on this observation, we have proposed a new method [26] to further extend preference lists to be able to find a matching when there is no stable matching. In particular, for every agent x, our method extends the preference list of x by including every agent y who is not already in x's preference list but "k-connected" to x (i.e., x and y are connected a chain of k PFOAF—preferred friend of a friend—relations). Also, we have observed that some existing students may request not to be matched with some students (e.g., previous roommates). Based on this observation, we have extended our method to consider "forbidden pairs".

Table 4
A k-Personalized-SRTI instance, characterized by a set of students x, sets A_x^- of unwanted students as roommates, and the k-extended lists $\prec_x^{k''}$ of preferred students as roommates for $k = 1$ and $k = 2$.

	Unwanted sets and preference lists						
Student x	A_x^-	$\prec_x$	$\prec_x^{'}$	$\prec_x^1$	$\prec_x^{1''}$	$\prec_x^2$	$\prec_x^{2''}$
a		$\langle e\rangle$	$\langle b\rangle$		$\langle e,\ b\rangle$	$\langle d\rangle$	$\langle e,\ b,\ d\rangle$
b	$\{d\}$	$\langle e\rangle$		$\langle c\rangle$	$\langle e,\ c\rangle$	$\langle c,\ a\rangle$	$\langle e,\ c,\ a\rangle$
c		$\langle b\rangle$	$\langle e,\ a\rangle$		$\langle b,\ e,\ a\rangle$		$\langle b,\ e,\ a\rangle$
d		$\langle b\rangle$		$\langle e\rangle$	$\langle b,\ e\rangle$	$\langle e,\ a\rangle$	$\langle b,\ e,\ a\rangle$
e		$\langle d\rangle$		$\langle\{a,b\}\rangle$	$\langle d,\{a,b\}\rangle$	$\langle\{a,b\},\ c\rangle$	$\langle d,\{a,b\},\ c\rangle$

Example 3. *Consider the Personalized-SRTI instance, in Table 3, does not have a stable matching. Let us denote a pair $\{x, y\}$ of students in A by xy. Suppose that b states that d is unwanted $A_b^- = \{d\}$. Then, $A^- = \{bd\}$. The k-acceptability graph $G_A = (V, E)$ is defined as follows: $V = \{a, b, c, d, e\}$ and $E = \{ae, bc, be, de\}$, where every edge $\{x, y\} \in E$ denotes pairs of agents that know each other and none of them is unwanted by the other. We say that agents x and y form a k-*connected pairs *if there exists a path of length k that connects x and y in G_A, and P^k denotes the set of all k-connected pairs in G_A. According to the k-acceptability graph $G_A = (V, E)$, k-connected pairs for $k = 1, 2, 3$ are defined as follows: $P^1 = \{ae, bc, be, de\}$, $P^2 = \{ad, ab, bd, ce\}$, and $P^3 = \{ac, cd\}$.*

According to $\prec'_c$, student c is indifferent between student a and e. We can break this tie by considering k-connected pairs: $e \prec'_{Carlos} a$ since c is more close to e rather than a, i.e., there exist a value $k = 2$ such that $\in P^2$ and $ac \notin P^2$, but $ac \in P^3$.

Table 4 shows k-extended preference lists for each student, for $k = 1$ and $k = 2$. Students a and c may be acceptable to b. Since $bc \in P^1$, $ab \notin P^1$ and $ab \in P^2$, $c \in \prec_b^1$ and $c \prec_b^2 a$.

For the running example, we can find a 1-stable matching $M_1 = \{a, bc, de\}$. This solution is also 2-stable.

We have conducted experiments with both objective and subjective measures to evaluate the usefulness of the additional knowledge about the networks of the agents' preferred friends. We have obtained promising results from the computational perspectives and from the users' perspectives. For further information about this study, we refer the reader to our paper [26].

Sticky-SRTI. Afacan et al. [27] have introduced the notion of "sticky stability" to accommodate appeal costs in school placements, and allows priority violations when the rank difference between the claimed and the received objects are less than a certain threshold. We plan to the use of sticky-stability in the context of SRTI to aid the computation of "good-enough" solutions.

6. Future Work

Beyond computing solutions, we aim to make SRTI-ASP transparent and interactive enough to provide evidence-based explanations. In our interactions with the dormitory administration or the student resources, we have observed their interests to know more about the recommended matching computed by our system SRTI-ASP: Why does (or does not) student *a* match with student *b*? What if student *a* matches with student *b*? etc. To increase the trust of the users, it is therefore desirable for SRTI-ASP not only to compute solutions but also to generate explanations for the recommended matching, and to support interactive engagement with the user.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

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

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

[3] D. Gusfield, R. W. Irving, The Stable Marriage Problem: Structure and Algorithms, MIT Press, Cambridge, MA, USA, 1989.

[4] R. W. Irving, D. F. Manlove, The stable roommates problem with ties, Journal of Algorithms 43 (2002) 85–105.

[5] A. Kunysz, The Strongly Stable Roommates Problem, in: Proc. of ESA, volume 57, 2016, pp. 60:1–60:15.

[6] S. Scott, A study of stable marriage problems with ties, University of Glasgow (United Kingdom), 2005.
[7] E. Ronn, Np-complete stable matching problems, Journal of Algorithms 11 (1990) 285–304.
[8] R. W. Irving, D. F. Manlove, G. O'Malley, Stable marriage with ties and bounded length preference lists, Journal of Discrete Algorithms 7 (2009) 213–219.
[9] T. Feder, A new fixed point approach for stable networks and stable marriages, Journal of Computer and System Sciences 45 (1992) 233 – 284.
[10] Á. Cseh, R. W. Irving, D. F. Manlove, The stable roommates problem with short lists, Theory of Computing Systems 63 (2019) 128–149.
[11] F. Cooper, Fair and large stable matchings in the Stable Marriage and Student-Project Allocation problems, Ph.D. thesis, University of Glasgow, 2020.
[12] D. J. Abraham, P. Biró, D. F. Manlove, "Almost stable" matchings in the roommates problem, in: International Workshop on Approximation and Online Algorithms, Springer, 2005, pp. 1–14.
[13] P. Biró, D. F. Manlove, E. J. McDermid, "almost stable" matchings in the roommates problem with bounded preference lists, Theoretical Computer Science 432 (2012) 10–20.
[14] E. Kujansuu, T. Lindberg, E. Makinen, The stable roomates problem and chess tournament pairings, Divulgaciones Matematicas 7 (1999) 19–28.
[15] E. M. Arkin, S. W. Bae, A. Efrat, K. Okamoto, J. S. B. Mitchell, V. Polishchuk, Geometric stable roommates, Inf. Process. Lett. 109 (2009) 219–224.
[16] A. E. Roth, T. Sönmez, M. U. Ünver, Pairwise kidney exchange, Journal of Economic theory 125 (2005) 151–188.
[17] A.-T. Gai, F. Mathieu, F. de Montgolfier, J. Reynier, Stratification in P2P networks: Application to BitTorrent, in: Proc. of ICDCS, 2007, p. 30.
[18] G. Brewka, T. Eiter, M. Truszczynski, Answer set programming: An introduction to the special issue, AI Magazine 37 (2016) 5–6.
[19] M. Gelfond, V. Lifschitz, The stable model semantics for logic programming, in: Proc. of ICLP, MIT Press, 1988, pp. 1070–1080.
[20] M. Gelfond, V. Lifschitz, Classical negation in logic programs and disjunctive databases, New Generation Computing 9 (1991) 365–385.
[21] M. Gebser, B. Kaufmann, R. Kaminski, M. Ostrowski, T. Schaub, M. T. Schneider, Potassco: The Potsdam Answer Set Solving Collection, AI Communications 24 (2011) 107–124.
[22] E. Erdem, M. Fidan, D. Manlove, P. Prosser, A general framework for stable roommates problems using answer set programming, Theory and Practice of Logic Programming 20 (2020) 911–925.
[23] P. Prosser, Stable roommates and constraint programming, in: Proc. of CPAIOR, 2014, pp. 15–28.
[24] G. Amendola, Solving the stable roommates problem using incoherent answer set programs (2018).
[25] M. Fidan, E. Erdem, Knowledge-based stable roommates problem: A real-world application, Theory and Practice of Logic Programming 21 (2021) 852–869.
[26] M. Fidan, E. Erdem, Finding personalized good-enough solutions to unsatisfiable stable roommates problems, Theory and Practice of Logic Programming (2025). Accepted for publication.
[27] M. O. Afacan, Z. H. Aliogullari, M. Barlo, Sticky matching in school choice, Economic Theory 64 (2016) 509–538.