Knowledge-Based Stable Roommates Problem: A Real-World Application (Extended Abstract)

MÜGE FIDAN and ESRA ERDEM
Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey
{mugefidan,esra.erdem}@sabanciuniv.edu

The Stable Roommates (SR) problem (Gale and Shapley, 1962) is a matching problem characterized by the preferences 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), such that the matching is *stable* (i.e., there exist no two agents who prefer each other to their roommates, and thus *block* the matching).

Motivated by real-world applications (c.f. placing students to dormitories) where it is hard to provide complete and strict orders of preferences, variations of SR have been investigated with incomplete preference lists (Gusfield and Irving, 1989) possibly with ties (SRTI) (Irving and Manlove, 2002). Furthermore, different fairness criteria have been investigated based on domain-independent measures, e.g., to maximize the total satisfaction of preferences of all agents, or to maximize the number of agents matched with their first preferences.

In our earlier studies (Erdem et al., 2020; Fidan and Erdem, 2021), we have introduced a formal framework to solve all variations of SRTI based on such domain-independent fairness criteria, and extended SRTI further towards computing more personalized solutions by including domain-specific criteria as well, such as the habits of students (e.g., smoking, studying), or the policies of universities (e.g., to increase diversity and inclusion). These studies have led to a flexible knowledge-based framework, called SRTI-ASP,

- that is general enough to combine different fairness criteria,
- that is specialized enough to compute personalized solutions, and
- that is adaptable enough to allow exceptions.

Accordingly, our method employs a knowledge representation and reasoning paradigm, called Answer Set Programming (ASP) (Niemelä, 1999; Marek and Truszczyński, 1999; Lifschitz, 2002), based on answer set semantics (Gelfond and Lifschitz, 1988, 1991). This extended abstract provides a summary of these contributions, and discusses our plans for future work.

Personalized-SRTI. We have introduced a new type of preference ordering considering (i) the importance of each 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). We have defined an extended preference list $\prec''_x$ for agent x, that combines two types of preference lists: a preference list $\prec_x$ of the agent over other agents (as in SRTI) and this new type of criteria-based personalized preference list $\prec'_x$ of the agent. Personalized-SRTI considers these extended preference lists to compute personalized stable matchings.

For instance, consider the example (Fidan and Erdem, 2021, Table 1) shown in Table 1. There are four students: Cem, Buse, Duru, Ayse. The preference list $\prec_{Cem}$ of Cem is $\langle$Ayse, Buse$\rangle$, and expresses that Cem prefers Ayse to Buse as a roommate.

Table 1: A personalized-SRTI instance with agents A={Ayse, Buse, Cem, Duru}, a common criteria based on habits, B=⟨"smoking", "cleanliness", "room environment", "sleep habits", "study habits"⟩. Each criterion $B[i]$ has a list C_i of possible choices:
C_1=⟨"Smoker","Non-smoker"⟩, C_2=⟨"Clean","Messy"⟩, C_3=⟨"Quiet","Social","Social and quiet"⟩,
C_4=⟨"Goes to bed early","Goes to bed before midnight","Goes to bed after midnight"⟩,
C_5=⟨"Studies in the room","Studies out of the room","Studies in and out of the room"⟩.

Agent x	Preference list $\prec_x$	Profile P_x	Weight list W_x	$\prec_x^{'}$	$\prec_x^{''}$
Ayse (non-smoker)	⟨Duru⟩	⟨2, 1, 1, 1, 1⟩	⟨5,4,3,2,1⟩	⟨Cem⟩	⟨Duru, Cem⟩
Buse (smoker)	⟨⟩	⟨1, 2, 3, 3, 3⟩	⟨1,0,3,4,5⟩	⟨Duru,Cem⟩	⟨Duru, Cem⟩
Cem (non-smoker)	⟨Ayse, Buse⟩	⟨2, 1, 3, 2, 3⟩	⟨5,5,4,3,2⟩	⟨Duru⟩	⟨Ayse, Buse, Duru⟩
Duru (non-smoker)	⟨Cem⟩	⟨2, 1, 3, 3, 3⟩	⟨3,3,3,3,3⟩	⟨Buse,Ayse⟩	⟨Cem, Buse, Ayse⟩

Suppose that the students consider a list B of criteria while specifying their preferences of the habits of a roommate. In this example, B = ⟨"smoking", "cleanliness", "environment", "sleep habits", "study habits"⟩. Every student x has a (preference) profile P_x, represented as a tuple, to show x's choices for each criterion in B respectively. The preference profile P_{Buse} of Buse is ⟨1, 2, 3, 3, 3⟩, and expresses that Buse prefers a roommate that is a "Smoker", "Messy", "Social and quiet", "Goes to bed after midnight", "Studies in and out of the room."

Meanwhile, every criterion in B may have a different importance for each agent. Every student x has a weight list W_x, represented as a tuple, to specify the respective weights of criteria in B for x. According to Table 1, W_{Buse} = ⟨1,0,3,4,5⟩: the most important criterion for Buse is "study habits", and the "cleanliness" criterion is not important.

For every student x, combining these two sorts of preferences (specified by P_x and W_x), a criteria-based personalized preference list $\prec_x^{'}$ is constructed. For example, in Table 1, for Buse, Ayse is not choice-acceptable for Buse since Ayse has no common choice with Buse. On the other hand, Cem and Duru are choice-acceptable for Buse. Then, first consider their choices for the most important criterion for Buse: "study habits". All of them prefers a roommate that "studies in and out of the room." Then, we consider the next important criterion for Buse: "sleep habits". While Cem prefers a roommate that "Goes to bed before midnight", Buse and Duru prefer a roommate that "Goes to bed after midnight". Hence, Buse prefers Duru to Cem according to their personalized preferences: $\prec_{Buse}^{'}$ is ⟨Duru, Cem⟩.

We define $\prec_x^{''}$ as an extended preference list by concatenating $\prec_x$ and $\prec_x^{'}$ depending on the importance given to these two types of lists. In this example, suppose that the preference lists $\prec_x$ are more important. Thus, the preference list $\prec_x^{'}$ is appended to the end of $\prec_x$. Then the extended preference list of Buse is $\prec_{Buse}^{''}$= ⟨Duru, Cem⟩. The extended preference lists for other agents are as shown in Table 1.

Most-SRTI. We have introduced a new incremental definition of a stable matching considering (i) a common ordering of the most preferred criteria (e.g., identified by large surveys) and (ii) the agents' preferred choices for each domain-specific criterion according to this order. The motivation is to try to match the agents with close choices.

For instance, assume that "sleep habits" is a more preferred criteria than "study habits". To maximize the number of roommates who are close to each other, first in terms of their sleep habits and then in terms of their study environments, we extend SRTI-ASP by adding the following weak constraints in the ASP formulation of SRTI:

$$\stackrel{\sim}{\leftarrow} \{room(x,y), bedTime(x,r1), bedTime(y,r2)\}.[|r1-r2|@p_1,x,y]$$

$$\overset{\sim}{\leftarrow} \{room(x,y), studyHabit(x,r1), studyHabit(y,r2)\}.[|r1-r2|@p_2,x,y]$$

Here, the priority p_1 is assigned a higher value compared to the priority p_2, i.e., $p_1 > p_2$, since "sleep habits" is preferred more than "study habits".

Diversity and inclusion. In addition to domain-independent fairness criteria and the students domain-specific preferences based on habits, the schools may prefer roommate matchings (e.g., from different departments, classes, or countries) that promote inclusion and diversity. Thanks to the expressive languages of ASP, these preferences can be expressed utilizing weak constraints as shown above.

Details of the definitions, notations, and ASP formulations mentioned above are presented in our paper (Fidan and Erdem, 2021).

A real-world application. We have illustrated an application of our method for Personalized-SRTI, by interacting with at least 200 students at Sabanci University. First, we have conducted a survey to select the most important five criteria that should be included in a dormitory application. Next, we have conducted a survey to get the preferences of each student for each criterion; in this way we have also collected real data for our experiments. Next, we have conducted two surveys to evaluate the usefulness of Personalized-SRTI from the perspective of students. We have observed that many participants have chosen the solutions computed by our method for Personalized-SRTI, and have given more importance to the additional information about habits and room environments. In that sense, extending SRTI to include additional domain-specific knowledge is useful.

Discussions. As humans play an important role in matching problems, our research objectives have evolved towards building robust yet provably beneficial matching methods and tools. Such AI methods require general flexibility to combine different domain-independent or domain-specific criteria, yet allow special personalization with respect to users' preferences/constraints, and adapt tolerably to allow exceptions. Thus, our research on Stable Roommates problems so far has been along these lines utilizing a logic-based knowledge representation and reasoning paradigm. Our ongoing and future studies aim to further extend these capabilities with explainability, to benefit humans.

References

Esra Erdem, Müge Fidan, David Manlove, and Patrick Prosser. A general framework for stable roommates problems using answer set programming. *Theory and Practice of Logic Programming*, 20(6): 911–925, 2020. doi: 10.1017/S1471068420000277.

Müge Fidan and Esra Erdem. Knowledge-based stable roommates problem: A real-world application. *Theory and Practice of Logic Programming*, 21(6):852–869, 2021. doi: 10.1017/S1471068421000302.

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

M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proc. of ICLP*, pages 1070–1080. MIT Press, 1988.

Michael Gelfond and Vladimir Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.

Dan Gusfield and Robert W. Irving. *The Stable Marriage Problem: Structure and Algorithms*. MIT Press, Cambridge, MA, USA, 1989.

Robert W Irving and David F Manlove. The stable roommates problem with ties. *Journal of Algorithms*, 43(1):85–105, 2002.

Vladimir Lifschitz. Answer set programming and plan generation. *Artificial Intelligence*, 138:39–54, 2002.

Victor Marek and Mirosław Truszczyński. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm: a 25-Year Perspective*, pages 375–398. Springer Verlag, 1999.

Ilkka Niemelä. Logic programs with stable model semantics as a constraint programming paradigm. *Annals of Mathematics and Artificial Intelligence*, 25:241–273, 1999.