Adding Preferences to Answer Set Planning

Tran Cao Son Enrico Pontelli

Knowledge Representation, Logic, and Advanced Programming Laboratory
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003, USA
{tson|epontell}@cs.nmsu.edu

Planning—in its classical sense—is the problem of finding a sequence of actions that achieves a predefined goal. As such, much of the research in AI planning has been focused on methodologies and issues related to the development of efficient planners. To date, several efficient planning systems have been developed (e.g., see [3] for a summary of planners that competed in the International Conference on Artificial Intelligent Planning and Scheduling). These developments can be attributed to the discovery of good domain-independent heuristics, the use of domain-specific knowledge, and the development of efficient data structures used in the implementation of the planning algorithms. Logic programming has played a significant role in this line of research, providing a declarative framework for the encoding of different forms of knowledge and its effective use during the planning process [5].

However, relatively limited effort has been placed on addressing several important aspects in real-world planning domains, such as *plan quality* and *preferences about plans*. In many real world situations, the space of feasible plans to achieve the goal is dense, but many of such plans, even if executable, may present undesirable behavior. In these situations, it may not be difficult to find a solution; rather, the challenge is to produce a solution that is considered satisfactory w.r.t. the needs and preferences of the user. Thus, feasible plans may have a measure of quality and only a subset may be considered acceptable. These issues can be illustrated with the following example:

Example 1. It is 7 am and Bob*, a Ph.D. student, is at home. He needs to be at school at 8am to take his qualification exam. His car is broken and he cannot drive to school. He can take a bus, a train, or a taxi to go to school, which will take him 55, 45, or 15 minutes respectively. Taking the bus or the train will require* Bob *to walk to the nearby station, which may take 20 minutes. However, a taxi can arrive in only 5 minutes. When in need of a taxi,* Bob *can call either the* MakeIt50 *or the* PayByMeter *taxi company.* MakeIt50 *will charge a flat rate of $50 for any trip, while* PayByMeter *has a fee schedule of $20 for the trip to school. If he takes the bus or the train, then* Bob *will spend only $2. Furthermore,* Bob*, being a student, prefers to pay less whenever possible.*

It is easy to see that there are only two feasible *plans for* Bob *to arrive at the school on time for his exam: calling one of the two taxi companies. However, a* PayByMeter *taxi would be preferable, as* Bob *wants to save money.* In this case, both plans are feasible but *Bob*'s preference is the deciding factor to select which plan he will follow.

The example demonstrates that users' preferences play a deciding role in the choice of a plan. Thus, we need to be able to evaluate plan components at a finer granularity than simply as consistent or violated. In [4], it is argued that users' preferences are likely to be more important in selecting a plan for execution, when a planning problem has too many solutions. It is worth noticing that, with a few exceptions, like the system SIPE-2 with metatheoretic biases [4], most planning systems *do not* allow users to specify their preferences and to use them in finding the plans. As such, the responsibility in selecting the most appropriate plan for their purpose rests solely on the users. It is also important to observe that *preferences* are different from *goals* in a planning problem; they might or might not be satisfied by a plan. The distinction is similar to the separation between *hard* and *soft* constraints [1]. For instance, if *Bob*'s goal is to spend at most $2 to go to school, then he does not have any feasible plans to arrive at school on time.

In this paper, we will investigate the problem of integrating users' preferences into a logic programming-based planner. We develop a language, called $\mathcal{PP}$ [6], for the specification of user preferences. We divide the preferences that a user might have into different categories:

- *Preference about a state:* the user prefers to be in a state s that satisfies a property ϕ rather than a state s' that does not satisfy it, even though both satisfy his/her goal;
- *Preference about an action:* the user prefers to perform the action a, whenever it is feasible and it allows the goal to be achieved;
- *Preference about a trajectory:* the user prefers a trajectory that satisfies a certain property ψ over those that do not satisfy this property;
- *Multi-dimensional Preferences:* the user has a *set* of preferences about the trajectory, with an ordering among them. A trajectory satisfying a higher priority preference is preferred over those that satisfy lower priority preferences.

It is important to observe the difference between ϕ and ψ in the above definitions. ϕ is a *state* property, whereas ψ is a formula over the whole *trajectory* (from the initial state to the state that satisfies the given goal).

We also provide a logic programming implementation of the language, based on answer set programming [2]. As demonstrated in this work, normal logic programs with answer set semantics provide a natural and elegant framework to effectively handle planning with preferences

References

1. S. Bistarelli et al. Labeling and Partial Local Consistency for Soft Constraint Programming. In *Practical Aspects of Declarative Languages*, Springer Verlag, 2000.
2. V. Lifschitz. Answer set planning. In *International Conference on Logic Programming*, pages 23–37, 1999.
3. D. Long, M. Fox, D.E. Smith, D. McDermott, F. Bacchus, and H. Geffner. International Planning Competition.
4. K.L. Myers and T.J. Lee. Generating Qualitatively Different Plans through Metatheoretic Biases. In *AAAI*, 1999.
5. T.C. Son, C. Baral, and S. McIlraith. Domain dependent knowledge in planning - an answer set planning approach. In *LPNMR*, Springer, pages 226–239, 2001.
6. T.C. Son and E. Pontelli. Planning with Preferences using Logic Programming. Technical Report CS-2003-007, New Mexico State University, 2003. `http://www.cs.nmsu.edu/TechReports`.