Goal Directed Policy Conflict Detection and Prioritisation: An Empirical Evaluation

(Extended Abstract)

Mukta S. Aphale*
University of Aberdeen, UK
m.aphale@abdn.ac.uk

Timothy J. Norman
University of Aberdeen, UK
t.j.norman@abdn.ac.uk

Murat Şensoy
Özyeğin University, Turkey
murat.sensoy@ozyegin.edu.tr

ABSTRACT

We address the problem of developing effective automated reasoning support for the detection and resolution of conflicts between plans and policies (or norms). How automated reasoning mechanisms can effectively support human decision makers in this process is little understood. In this research, we have conducted experiments with human subjects to assess how effective these reasoning mechanisms are. We found that providing guidance to users regarding what problems to prioritise and highlighting related conflicts led to higher quality outcomes, and problems were successfully solved more rapidly.

Keywords

Policies, Norms, Conflict Resolution

1. INTRODUCTION

The use of policies (or norms) to guide and regulate behaviour of various entities in a system is commonplace in human, agent and mixed societies. Policies, however, operate in conjunction with individual and organisational goals to protect important information or resources, and promote ideal action. Policies may, however, impede the achievement of goals. There is often an important trade-off between policy compliance and goal achievement. The question addressed here is how may we provide effective automated reasoning support for users responsible for specifying appropriate policies/norms, while ensuring the achievement of individual/organisational goals?

Automated support for humans making agreements on what actions to take in complex, norm-governed scenarios has been explored by Sycara *et al.* [5]. The problem of supporting humans in authoring policies has also been addressed [3], but there is very limited automated support for conflict detection and resolution. In Uszok *et al.* [6] some reasoning support for conflict detection has been explored, but this is confined to the detection and resolution of conflicts between policies. There is little research to date that adequately addresses how to support humans in detecting and resolving conflicts between policies and plans/goals.

Conflicts that occur between policies and plans may vary in significance. In reasoning about conflicts it would make sense to prioritise those that are more likely to impact on system performance. We explore the effectiveness of support for two activities: (i) automated detection of policy/plan conflicts; and (ii) providing guidance to users so that they may prioritise the conflicts that they consider. We present the results obtained from experiments with human users.

2. DETECTION AND PRIORITISATION

We build upon research on policy/norm representation and reasoning: OWL-POLAR [2] is a language for expressing, and a set of mechanisms for reasoning about policies. OWL-POLAR provides means to identify conflicts between policies (logical conflicts). We have extended this to include the detection of conflicts between policies and plans expressed in OWL-S [4] (functional conflicts), and for prioritising plan/policy conflict reasoning.

Functional conflicts may occur for a number of reasons, including because a side-effect of an action is prohibited, because a pre-condition of an action requires a prohibited state, or because an action requires access to a resource and the actor/role is not permitted access. Conflicts between policies (logical conflicts) may occur only in specific conditions, but these conditions may be highly unlikely to occur. Similarly, it may be unlikely (but not impossible) that access to a prohibited resource is necessary for some user. At the same time, compliance with some policies may be very costly in terms of plan efficiency.

Driven by these pragmatic concerns, we have developed an efficient activity prioritisation algorithm. This algorithm rapidly identifies critical paths in sets of possible plans to achieve specific goals, ranking these activities. This ranking can be used to prioritise policy conflict reasoning in a way that balances goal achievement (based on the costs of actions and their likelihood of success) and plan choice [1]. In this way the, relatively costly, reasoning about plan/policy and policy/policy conflicts may be focussed on more important problems. This is particularly important when automated reasoning is used in support of human decision-making.

In the experiments conducted, we were interested in how these mechanisms support reasoning. The metrics we are concerned about are: (i) the ability of the user to identify a valid (conflict-free) plan; (ii) the utility of the best plan identified; (iii) the time taken to identify a plan; and (iv) the number of conflicts resolved during the experiment.

*Supported by the award made by the RCUK Digital Economy programme, reference EP/G066051/1

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

3. EXPERIMENT DESIGN

We developed a user interface where plans are presented as directed acyclic graphs. Nodes represent activities with associated costs and likelihood of success, and edges represent a precedence relation over activities. A travel agency domain was chosen to ensure that the problems given to participants were accessible without significant prior training. We used two problems (α and β) from the same domain, although with different goals, policies and action choices. In α there were 640 possible plans, the average plan utility was 126 with a variance of 1240. In β there were 480 possible plans with average utility 132, variance 1730.

There were three conditions considered in the experiments: (i) no agent support (N); (ii) support for the detection of conflicts (D); and (iii) support for both prioritisation of reasoning and conflict detection (PD). The detection support was in the form of a box that pops up when the user selects an activity on the interface, advising whether there is a conflict and what the conflict is. Prioritisation highlights which activity the user should explore for conflicts next.

The participants were given a small training task before the experiment. They were then each given a problem to solve under one of the conditions (N, D or PD), and then a second problem to solve under another condition. Participants were given a written list of policies, including obligations and prohibitions, and were asked to identify the best conflict-free plan they could. No time limits were set for the tasks, but the time taken for each problem was recorded.

We conducted two experiments with 20 participants in each. In experiment 1: 5 participants were given α under condition N (no support) then β under D; 5 had β under N then α under D; 5 had α under D then β under N; and the final 5 had β under D then α under N. Similar combinations were used in experiment 2 with problems α and β and conditions D and PD. The rationale for this design was to control for small variations in the difficulty of the problems, and the effect of prior training on performance.

4. RESULTS

To assess the significance of our results we used a one-tailed Welch Two Sample t-test (robust to sample size differences); $p < 0.05$ is our standard for claiming significance. We are interested in the *quality* of the plan with respect to the distribution of the utilities of all possible plans for a given problem (α or β); i.e. the percentile in which the resulting plan resides (see Tables 1 and 2). We also recorded the time taken to identify a plan and the number of conflicts resolved by the user.

An initial analysis of the results for experiment 1 indicates that the quality of plans identified with detection support (D) is significantly greater than the quality of those with no support (N): $p < 0.02$. Closer inspection of the results, however, revealed that in 6 out of the 10 cases with no support, no conflict-free plan was identified. Removing these cases, and hence considering the difference between N and D given that a valid plan was identified, revealed that there was no significant difference: $p = 0.702$ (Percentile′ in Table 1).

In experiment 2 we again checked the validity of the plans, and in one of the detection support cases (D) no valid plan was identified. We removed this case from our analysis, and the results without this case are reported in Table 2. In comparing condition D with PD, the results show that there is a significant difference in both plan quality and time taken to identify a valid plan. Plan quality is on average within a percentile 12.9 points higher in the PD condition than in the D condition ($p < 0.0003$). Time taken to identify a valid plan was on average 176 seconds faster in the PD condition than in condition D ($p < 0.002$). There were no significant differences in the number of conflicts solved in any of the cases, and in either experiment.

Variable	N	D	t	df	p-value
Percentile	59.2	81.4	2.17	26.1	0.0195
Percentile′	84.5	81.4	-0.535	30.6	0.702
Time	546	514	0.389	37.6	0.350
Conflicts	2.07	2	-0.373	24.7	0.356

Table 1: Condition N versus condition D.

Variable	D	PD	t	df	p-value
Percentile	86.2	99.1	4.18	18.8	0.000256
Time	482	306	-3.24	36.9	0.00127
Conflicts	2.05	2.4	2.16	23.5	0.979

Table 2: Condition D versus condition PD.

5. CONCLUSION

Given our analysis, we draw the following conclusions: (i) support for the detection of conflicts significantly improves the likelihood that a valid plan will be identified by a human decision-maker; and (ii) support for the prioritisation of reasoning about conflicts in conjunction with support for conflict detection is significantly better than conflict detection support alone, both in terms of plan quality and the time taken to identify a valid plan.

6. REFERENCES

[1] M. S. Aphale, T. J. Norman, and M. Şensoy. Goal-directed policy conflict detection and prioritisation. In J. S. Sichman and H. Aldewereld, editors, *COIN 2012*, volume 7756 of *LNAI*, pages 87–104. Springer, 2013.

[2] M. Şensoy, T. J. Norman, W. W. Vasconcelos, and K. Sycara. OWL-POLAR: A framework for semantic policy representation and reasoning. *Journal of Web Semantics*, 12–13:148–160, 2012.

[3] M. Johnson, J. Karat, C.-M. Karat, and K. Grueneberg. Usable policy template authoring for iterative policy refinement. In *Proc. IEEE Policy*, pages 18–21, 2010.

[4] D. Martin *et al.* Bringing semantics to web services: The OWL-S approach. In J. Cardoso and A. Sheth, editors, *SWSWPC 2004*, volume 3387 of *LNCS*, pages 26–42. Springer, 2005.

[5] K. Sycara *et al.* Agent support for policy-driven collaborative mission planning. *The Computer Journal*, 53(5):528–540, 2010.

[6] A. Uszok *et al.* New developments in ontology-based policy management: Increasing the practicality and comprehensiveness of KAoS. In *Proc. IEEE Policy*, pages 145–152, 2008.