A decision-theoretic approach to resourcing plans in a policy-constrained environment

Chukwuemeka David Emele[1], Timothy J. Norman[1], Frank Guerin[1], and Simon Parsons[2]

[1] University of Aberdeen, Aberdeen, AB24 3UE, UK
{c.emele, t.j.norman, f.guerin}@abdn.ac.uk
[2] Brooklyn College, City University of New York, 11210 NY, USA
parsons@sci.brooklyn.cuny.edu

Abstract. In multi-agent societies, agents often inherit the social characteristics of the individuals or organisations they represent. These social characteristics (e.g policies) often inform the behaviour of the agents. We present an efficient combination of argumentation, machine learning and decision theory for identifying, learning and modeling the policies of others, using argumentation-derived evidence. This is useful for choosing who to talk to in a collaborative setting. In a set of experiments, we demonstrate that this combination of techniques can lead to a statistically significant increase in performance (utilities), improvement in argumentation strategies and reduction in communication overhead.

1 Introduction

Developing and coordinating coherent and efficient plans for joint action is an important area of research interest in multi-agent systems [4, 5, 11]. An important aspect of the problem is how these plans can be resourced. Resourcing plans is a complex task on its own, and becomes even more challenging when agents in the domain are guided by policies. Policies dictate under what circumstances agents may provide some resource to other agents, and conversely when they are prohibited from doing so. Such environments are said to be policy-constrained. We focus on individual policies, which are often private to that individual member or subset of the team, and are not necessarily shared with others.

Consider the following dialogue that may occur between a visiting lecturer x (*seeker*) and a system administrator y (*provider*). Suppose agents are able to exchange arguments then x can, potentially, gather more information regarding the policies guiding y. Agents can ask for and receive explanations (Scenario 1, *line* 9 *to* 10) [1, 2] or suggest alternatives (Scenario 2, *line* 8) [2]. Also, agents can ask for more information about the attributes of a request (*lines* 2 *to* 7) [3].

In this paper, we present a framework that allows agents to exchange arguments about their policies and resource constraints during dialogue. We hypothesise that such arguments can be exploited (by combining appropriate machine learning and decision-theoretic model) to yield a higher utility and a significant reduction in communication overhead than when there is no such combination.

#	Scenario 1	Scenario 2
1	x: Can I have a *laptop*?	x: Can I have a *laptop*?
2	y: What do you want to use it for?	y: What do you want to use it for?
3	x: To present a lecture.	x: To present a lecture.
4	y: Where is the lecture holding?	y: Where is the lecture holding?
5	x: In Lecture Room 1.	x: In Lecture Room 1.
6	y: Which day is the lecture?	y: Which day is the lecture?
7	x: On Saturday 24/4/2010.	x: On Saturday 24/4/2010.
8	y: No, I can't provide you with a *laptop*.	y: I can provide you with an account to log into the *desktop* in Lecture Room 1 instead.
9	x: Why?	x: I accept *desktop* account.
10	y: I'm not permitted to release a *laptop* in these circumstances.	

The remainder of this paper is organised as follows: Section 2 describes the negotiation dialogue and Section 3 presents the reasoning framework of the agent. Our simulation environment is described in Section 4 and Section 5 presents experimental results. Section 6 discusses related work and Section 7 concludes.

2 Negotiation Dialogue

The negotiation for resources takes place in a turn-taking fashion. The *seeker* sends a request for a resource to a *provider*. If the *provider* has the requested resource in its resource pool and it is in a usable state then it checks whether there is any policy that forbids it from providing the resource to the *seeker*. If the *provider* needs more information in order to make a decision, the *provider* would ask questions regarding the use of that resource. There is a cost attached to the revelation of private information to other agents. An agent might refuse to reveal a piece of information if doing so is costly [8]. The decision-theoretic model provides a metric for determining whether to reveal more information or not. The negotiation protocol is shown in Figure 1.

The *provider* releases the resource to the *seeker* if there is no policy that prohibits the *provider* from doing so. Otherwise, the *provider* offers an alternative resource (if there are no policies that forbid that line of action and the alternative resource is available). When an alternative resource is suggested by the *provider*, the *seeker* evaluates it. If it is acceptable, the *seeker* accepts it. Otherwise, the *seeker* refuses the alternative. In principle, this cycle may be repeated until an alternative is accepted or all alternatives are exhausted but for simplicity, we allow only one alternative to be suggested per request.

Agents could ask for justification (with respect to a request or the response to a request), which indicates what constraints others may be operating within. For instance, let us assume that a *provider* has a policy that forbids it from providing a *laptop* to any *seeker* that intends to use it for *playing online games*. Then, whenever a *laptop* is requested the *provider* seeks to ascertain that the *seeker* does not intend to use it for *playing online games*. This pattern of behaviour could serve as evidence for the kind of policies under which the *provider*

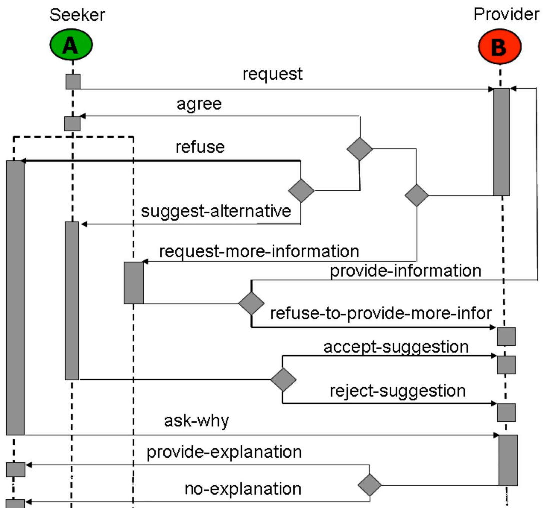


Fig. 1. The negotiation protocol.

operates. This evidence could be exploited by the *seeker* in preempting what extra information the provider might want to ask for. To this end, the seeker may provide the purpose for which the *laptop* is requested along with the request. In the foregoing example, if the purpose for requesting the *laptop* was to present a lecture then the location of the lecture may become an important information that the provider will need to make a decision. Furthermore, whenever a *seeker*'s request is refused then the *seeker* will ask for explanations/justifications for the refusal. This additional evidence is beneficial, and could be used to improve the model and, hence, the quality of the decisions made in future encounters.

3 Reasoning

In a domain where there are underlying constraints that could yield similar behaviours, learning from dialogue could improve the accuracy of the information learned about the policies of others. We claim that significant improvements in utility can be achieved if evidence derived from argumentation [3] is modeled and utilised in advising agents with regards to the policies of others.

Learning Policies

Policies can be conceived as properties of an agent and can be teased out using dialogue, and then learned. In this work, agent policies are implemented as rules and these rules are inferred from attributes (e.g. resource, purpose, location, etc.) that are communicated back and forth during negotiation. Since policies guide the way agents act by providing rules for their behaviour it makes sense to use a rule-based learning algorithm. A policy, expressed as a rule, has the form $\pi_C = \langle C, action \rangle$ and states that if the constituents of a *seeker's* request satisfies the context C, then the *provider* should perform *action* [10]. Let the set of attributes that characterise agents' policies be denoted as follows: the set of affiliations (O), resources (R), purposes (P), locations (L) and days (D). The context C is an ordered list comprising of the affiliation of the agent $o \in O$, the resource requested $r \in R$, the purpose $p \in P$, the location $l \in L$ and the day $d \in D$. In other words, context C consist of the attributes of the request, and *action* is the decision as to whether the request should be granted (in which case the resource is provided) or denied. That is,

$$\pi_C = \langle \langle o, r, p, l, d \rangle, \{grant, deny\} \rangle$$

Examples of policies that a *provider* may be operating under may include:

π_{C1}**:** *You are* ***permitted*** *to release a laptop to a user if the user is a staff of a university in the UK and the laptop is to be used at LectureRoom1 for presenting a lecture.*

$$\pi_{C1} = \langle \langle uk, laptop, present\text{-}lecture, LectureRoom1, any \rangle, grant \rangle$$

π_{C2}**:** *You are* ***prohibited*** *from releasing a laptop to a user if the laptop is to be used for playing online games.*

Similarly, $\pi_{C2} = \langle \langle any, laptop, play\text{-}online\text{-}game, any, any \rangle, deny \rangle$

The term *any* is used to denote that the attribute in question is permitted to take on any value. For example, policy π_{C1} will be invoked regardless of the day.

These policy rules are learned using sequential covering [6], SC for short. SC is a rule-based learning technique, which constructs rules by sequentially covering the examples. The sequential covering algorithm induces one rule at a time (by selecting attribute-value pairs that satisfy the rule), and removes the data covered by the rule and then iterates the process. SC generates rules for each class by looking at the training data and adding rules that completely describe all tuples in that class. For each class value, rule antecedents are initially empty sets, augmented gradually for covering as many examples as possible.

Although in the experimental study discussed here we use sequential covering, other machine learning techniques may be adopted. In earlier experiments [3], however, sequential covering proved to perform very well in building a good model of others' policies rapidly.

Decision-Theoretic Approach

Based on the model of others' policies that has been learned over time, the *seeker* agent employs decision theory in choosing who among the potential providers to talk to with a view to resourcing a plan. Here, we formalise the decision-theoretic model that is used in this work. Let X be the set of *seeker* agents such that $x \in X$, and Y be the set of *provider* agents such that $y \in Y$. Let R be the set of resources such that $r \in R$, and $Tasks$ be the set of tasks such that $t \in Tasks$. The set of information is denoted as I such that $i \in I$.

Definition 1. ***(Benefit)*** *The benefit gained in resourcing a task* $t \in Tasks$ *with resource* $r \in R$*, denoted as* $benefit(r, t)$ *is given as* $benefit : R \times Tasks \rightarrow \mathbb{R}$

Generally, *seeker* agents receive benefits for obtaining a resource to be used in fulfilling a task, and incur costs in providing information to *provider* agents. In other words, benefit is the value added or satisfaction derived from resourcing t using r. This satisfaction is the *seeker*'s valuation of r for the task instance considered and this value varies from task to task.

Definition 2. ***(Unit Cost)*** *The cost of revealing a specific piece of information, $i \in I$, to a specific agent, $y \in Y$, denoted as $cost(i, y)$ is given as:*

$$cost : I \times Y \rightarrow \mathbb{R}$$

Definition 3. ***(Total Cost)*** *The cost of revealing all pieces of information in the set I to agent y, denoted as $Cost(I, y)$ is the summation of the cost of revealing each $i \in I$ to agent y. That is,*

$$Cost(I, y) = \sum_{i \in I} cost(i, y) \tag{1}$$

This cost depends on y, but not on the task/resource. In other words, the cost of revealing a piece of information to agent y (by agent x) is constant irrespective of the task being resourced or the resource under consideration.

Definition 4. ***(Information Required)*** *The set of information required for y to make available resource r according to our prediction of their policy is denoted as $info\text{-}required(r, y)$.*

Note that this depends upon resource r because this represents the minimum information needed to convince an agent to provide the resource according to x's model of y. If there is no way to convince y to provide the resource then $info\text{-}required(r, y)$ is the empty set. From our earlier example, if we predict that agent y needs to know where and when the *laptop* will be used then:
$info\text{-}required(laptop, y) = \{laptop, LectureRoom1, friday\}$.

Let $Pr(YesPolicy|y, r, I_x)$ be the probability of a Yes response given we ask agent y for resource r and the information required is I_x (that is, $info\text{-}required(r, y) = I_x$). Let $Pr(YesAvailable|y, r, I_x)$ be the probability of the resource being available given we ask agent y for resource r and $info\text{-}required(r, y) = I_x$.

Definition 5. ***(Joint Probability)*** *The probability that the policy is Yes and the resource is available given we ask agent y for resource r and $info\text{-}required(r, y) = I_x$, denoted as $Pr(Both)$ is given as:*

$$Pr(Both) = Pr(YesPolicy|y, r, I_x) * Pr(YesAvailable|y, r, I_x) \tag{2}$$

Definition 6. ***(Utility Function)*** *Every seeker agent y has a utility function $u_y^x : 2^Y \rightarrow \mathbb{R}$.*

The expected utility, $E(u_y^x)$ that agent x derives from acquiring a resource r for a task t from agent y requiring the revelation of I_x to y is computed as:

$$E(u_y^x) = benefit(r, t) \times Pr(Both) - Cost(I_x, y) \tag{3}$$

Definition 7. ***(Selection Strategy)*** *A selection strategy for an agent is a function $s : Y \rightarrow y_{opt}$, where y_{opt} is computed as follows:*

$$y_{opt} = \underset{y \in Y \, s.t. \, info\text{-}required(r,y) \neq \emptyset}{\text{argmax}} E(u^x_y) \tag{4}$$

Now, we can make a judgement about whether or not to even bother asking anyone at all. For instance, if $\mathrm{E}(u^x_{y_{opt}}) \leq 0$ then one could argue that a rational *seeker* should not attempt to acquire r as it could yield a negative or zero utility.

4 Simulation Environment

We implemented a simulation environment, and integrated our argumentation, machine learning and decision-theoretic model into the framework. The architecture is sketched in Figure 2. Each agent has three main modules: the dialogue module, the learning module and the strategy module. The dialogue module embodies the dialogue controller, which handles all communication with other agents in the domain. The learning module reasons over the dialogue and attempts to identify and learn models of other agents' policies and resource availabilities based on arguments exchanged during encounters. The arguments include the features that an agent requires in order to make a decision about providing a resource. The strategy module looks up policy and resource availability constraints from the knowledge-base and selects which potential provider yields the highest utility. The private datastore acts as a repository where an agent stores its private information and constraints.

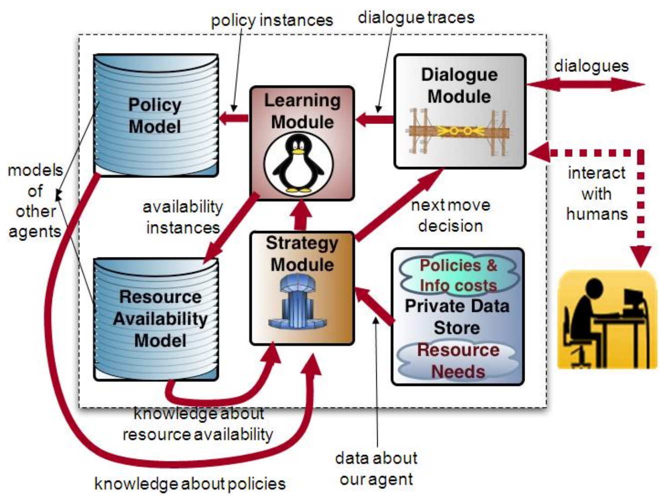


Fig. 2. Architecture of the framework to support resourcing a plan.

The simulation environment allows us to generate multiple *providers* with randomised policies. *Seekers* are also generated with randomised initial models of the policies of *providers* in the simulation and randomised problems for the *seeker* to solve. The *seeker* predicts whether the *provider* has a policy that forbids/permits the provision of such resource in that context.

5 Experiments and Results

In a series of experiments, we show how decision theory, argumentation and machine learning techniques can support agents engaging in collaborative activities. This novel combination of techniques can increase their predictive accuracy, reduce communication overhead, and increase utility gained. In this simulation, agents can exchange between 2 to 10 messages in order to complete a task. The number of messages exchanged depends on the amount of information required by the *provider* in order to make a policy-compliant decision. The experiments show that by effectively preempting the questions and information requirements of the provider, the seeker can reduce the number of messages required to complete a task. The results show that combining the decision-theoretic model with a more accurate and stable model of the agents' policies led to an increase in the utility gained from completing tasks.

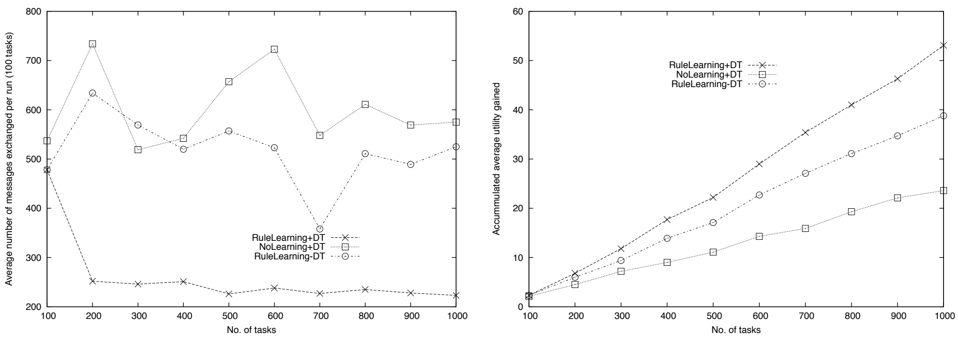
The scenario adopted in this research involves a team of five software agents (one *seeker* and four *providers*) collaborating to complete a joint activity in a region over a period of three simulated days. The region is divided into five locations. There are five resource types, and five purposes that a resource could be used to fulfill. This results in 375 possible configurations. A task involves the *seeker* identifying resource needs for a plan and collaborating with the *providers* to see how that plan can be resourced. Every *provider* agent could, potentially, ask for additional information in order to make a decision on the provision of resources based on its policies. A simple lookup table was used as a control condition and it serves as a structure for simple memorisation of outcomes.

Results

Experiments were conducted with *seekers* initialised with random models of the policies of *providers*. 100 runs were conducted for each case, and tasks were randomly created during each run from 375 possible configurations.

Figure 3(i) illustrates the effectiveness of combining argumentation, machine learning and decision-theoretic model in resourcing plans. It shows the average number of messages exchanged and the standard deviations for each of the approaches, namely: Decision-theoretic model combined with rule learning in reasoning over argumentation-derived evidence (RL+DT), Decision-theoretic model combined with lookup table in reasoning over argumentation-derived evidence (NL+DT), and reasoning over argumentation-derived evidence with the aid of rule learning only (RL-DT). In each case, the model of others' policies is recomputed after each set of 100 tasks. The number of messages exchanged in RL+DT was consistently and significantly lower than those in the other two cases. The reason for this, after detailed analysis of the data, is because the *seeker* is (1) able to make an informed decision concerning which provider to approach for a given resource, and (2) able to preempt the information requirements of the provider and thereby present it without having to be asked.

In the *NL+DT* case, the seeker has the opportunity to use the decision-theoretic model to select which provider is more likely to provide the resource. The results, however, show that the number of messages per task varies between around 7 and 5. The reason for this poor performance is because the *seeker* in this setup is unable to build an accurate model of the *provider*'s policies and resource availabilities. Therefore, the input into the decision-theoretic model is defective and possibly deficient. Similarly, in the *RL-DT* approach, the number of messages exchanged per task rose to as high as 6, which confirms that adopting machine learning and argumentation alone might achieve greater utility but may not reduce the number of arguments exchanged. The regression analysis of these results shows that as the number of tasks increases, the number of messages exchanged in the *RL+DT* scenario consistently converges with a 95% confidence interval. On the other hand, with significance $p > 0.05$, there is no statistical significance as to whether *NL+DT* and *RL-DT* converge or not respectively. These results show that combining argumentation, machine learning and decision-theoretic model can help reduce the communication overhead.



(i) *(ii)*

Fig. 3. The number of messages exchanged (i) and cumulative average utility (ii).

Figure 3(ii) compares the cumulative average utility of the *seeker* in the three configurations. The results show that the three configurations recorded increase in utility. However, *RL+DT* and *RL-DT* significantly and consistently outperforms *NL+DT*. This is because the rule learning algorithm with the aid of argumentation-derived evidence built more accurate models of others' policies and resource availabilities than the control condition in *NL+DT*. Similarly, *RL+DT* consistently outperforms *RL-DT*. The reason for this is the fact that *seekers* in the *RL-DT* are not equipped with the decision-theoretic heuristics which helps to make better choices in terms of cost of information revelation.

6 Related Work

The research presented in this paper represents the first model for combining argumentation, machine learning and decision theory to learn underlying social characteristics (e.g. policies) of others and exploit the models learned to reduce communication overhead and improve strategic outcomes. There is, however, some prior research in combining machine learning and argumentation, and in using argument structures for machine learning. In that research, Možina et al. [7] propose a novel induction-based machine learning mechanism using argumentation. The work implemented an argument-based extension of CN2 rule learning (ABCN2) and showed that ABCN2 out-performed CN2 in most tasks. However, the framework developed in that research will struggle to learn and build an accurate model of policies from argumentatiom-derived evidence, which is the main issue we are addressing in our work. Also, the authors assume that the agent knows and has access to the arguments required to improve the prediction accuracy, but we argue that it is not always the case. As a result, we employ information-seeking dialogue to tease out evidence that could be used to improve performance (in terms of utility gained and communication overhead).

One of the six primary types of dialogues that Walton and Krabbe identified is information-seeking dialogues [13]. Their work has been particularly influential in argumentation-based dialogue research. In a related work, Doutre et al. [1] present a semantics for communications protocols involving permissions (or policies) and arguments using information-seeking dialogues. However, this work does not attempt to learn those permissions, nor build a model of those policies.

In recent research, Sycara et al. [12] investigate agent support for human teams in which software agents aid the decision making of team members during collaborative planning. One area of support that was identified as important in this context is guidance in making policy-compliant decisions. This prior research focuses on giving guidance to humans regarding their own policies. An important and open question, however, is how can agents support human decision makers in developing models of others' policies and using these in guiding the decision maker? Our work employ a novel combination of techniques in identifying, learning and building accurate models of others' policies, and exploit these in supporting decision making. The decision maker can make informed judgement regarding who to ask for the provision of resources and which information needs to be revealed in order to convince the provider.

7 Conclusions

In this paper, we have presented a technique that combines argumentation, machine learning and decision theory for supporting agents engaging in collaborative activities. The results of our empirical investigations show that combining argumentation, machine learning and decision theory can have a statistically significant positive impact on resourcing plans in particular, and decision making in general. The results also demonstrate that accurate policy models can help in developing more robust and adaptive strategies for advising human decision makers on how a plan may be resourced and who to talk to in that regard [12]. This work can aid in the development of more effective strategies for agents [8].

Future Directions

In our future work, we plan to develop further strategies for advising human decision makers on how a plan may be resourced in critical situations. Parsons et al. [9] investigated the properties of dialogues and examined how different classes of protocols can have different outcomes. Furthermore, we plan to explore ideas from this work to see which class of protocol will yield the "best" result in this kind of task. We are hoping that some of these ideas will provide helpful feedback for future research on developing strategies for team-based activities.

References

1. S. Doutre, P. McBurney, M. Wooldridge, and W. Barden. Information-seeking agent dialogs with permissions and arguments. In *Proc. of the 5th Int. Joint Conference on Autonomous Agents and Multiagent Systems*, Hakodate, Japan, 2006.
2. C. D. Emele, T. J. Norman, F. Guerin, and S. Parsons. Learning policy constraints through dialogue. In *Proc. of the AAAI Fall Symposium on The Uses of Computational Argumentation*, pages 20–26, Virginia, USA, 2009.
3. C. D. Emele, T. J. Norman, F. Guerin, and S. Parsons. On the benefit of argumentation-derived evidence in learning policies. In *Proc. of the 3rd Int. Workshop on Argumentation in Multi-Agent Systems*, Toronto, Canada, 2010.
4. E. Ephrati and J. S. Rosenschein. Multi-agent planning as a dynamic search for social consensus. In *Proc. of the 13th Int. Joint Conf. on Artifical intelligence*, pages 423–429, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
5. B. Grosz and S. Kraus. Collaborative plans for group activities. In *Proc. of the 13th International Joint Conference on Artifical Intelligence*, pages 367–373, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
6. T. M. Mitchell. *Machine Learning.* McGraw Hill, 1997.
7. M. Možina, J. Žabkar, and I. Bratko. Argument based machine learning. *Artificial Intelligence*, 171(10-15):922–937, 2007.
8. N. Oren, T. J. Norman, and A. Preece. Loose lips sink ships: A heuristic for argumentation. In *Proc. of the 3rd Int'l Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2006)*, pages 121–134, 2006.
9. S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexities of some formal inter-agent dialogues. *Journal of Logic and Comp.*, 13(3):347–376, 2003.
10. J. Pearl. *Causality: Modeling, Reasoning, and Inference.* Cambridge University Press, Cambridge, 2000.
11. W. Shen, L. Wang, and Q. Hao. Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Trans. on*, 36(4):563 –577, July 2006.
12. K. Sycara, T. J. Norman, J. A. Giampapa, M. J. Kollingbaum, C. Burnett, D. Masato, M. McCallum, and M. H. Strub. Agent support for policy-driven collaborative mission planning. *The Computer Journal*, page bxp061, 2009.
13. D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning.* SUNY Press, Albany, NY, USA, 1995.