Model Based Critique of Justifications for Action

Adam WYNER, Katie ATKINSON, Trevor BENCH-CAPON
Department of Computer Science, The University of Liverpool, U.K.

Abstract. Justifications for actions can typically be critiqued on a number of grounds. Critiques of such justifications can be structured using an argumentation scheme for practical reasoning and associated critical questions. The form that the critiques can take in a computational system, however, depends crucially on the semantic underpinnings of the scheme. We describe a semantic model which, we argue, can support a rich and systematic critique of justifications for actions presented using a argumentation scheme specifically designed to support practical reasoning. The approach has been realised as a Prolog program, and the paper is illustrated by a running example and fragments of the code for the application.

1. Introduction

Argumentation schemes have become popular as a way of capturing different styles of argument. Argument schemes are considered to license presumptive conclusions, which are then subject to critiques based on so-called *critical questions* associated with the schemes. In this paper we will first consider standard approaches to representing schemes in structured arguments, where they are incorporated into the rules of knowledge bases. We will then present an approach based on a richer semantic model which, we argue permits a richer critique to be offered, and more fully exploits the diversity of possible criticisms encapsulated in the critical questions. The generation of the critiques has been realised as a Prolog program, and the paper is illustrated by a running example and fragments of the code for the application.

Protocols for persuasion dialogues typically envisage the participants (often termed *Proponent* and *Opponent*) as equipped with differing knowledge bases. The persuasion dialogue then consists of an exchange of facts and rules intended by the Proponent to enable the Opponent to derive the claim at issue from its own knowledge base, extended by the information acquired in the dialogue. See [7] for an excellent discussion of systems of this type. Justifying an action has many similarities with persuasion, but also some important differences, which mean that a different sort of dialogue is required. As well as concerning actions rather than beliefs, motivations play a key role in action justification: justifications often rely on the reasons an agent had for performing the action in question, and such reasons, and the priority given to competing reasons, will legitimately differ from agent to agent, according to the their individual tastes, aspirations, cultural background and the like.

Standard persuasion dialogues such as those of [7] can be seen as based on an argumentation scheme such as the following, which we will call *Defeasible Modus Ponens*(DMP). We will adopt the style of presenting argumentation schemes found in [10], in which schemes are described in terms of a set of premises, a conclusion and a set of *critical questions*, characteristic of the scheme, which can be used to challenge the presumptive conclusion.

> *DMPP1 Premise:* The Conjunction A is True
> *DMPP2 Premise:* If A is true then some conclusion C is presumptively true
> *DMPC Conclusion:* C is presumptively true.
> *DMPCQ1 Critical Question:* Is every conjunct in A true?
> *DMPCQ2 Critical Question:* Is the rule of Premise 2 applicable?
> *DMPCQ3 Critical Question:* Is C consistent with the rest of the knowledge base?

Now suppose we are using a formalism such as ASPIC+ [8]. There the arguments are all grounded in a knowledge base comprising facts and rules (both strict and defeasible). Arguments are thus based on instantiating rules. Where, as is usually the case, the argument requires a chain of rules, each link represents a subargument. The base case is where the argument for a fact f is simply that f is in the knowledge base. Arguments can be attacked by showing the contrary of a term in the antecedent of a rule used in the argument (DMPCQ1), or showing the contrary of the consequent of a rule used in the argument (DMPCQ3), or by showing that the rule is not applicable (DMPCQ2). In ASPIC+ [8], argumentation schemes are expressed as defeasible rules in the knowledge base, and undercutters for these schemes take the form of rules expressing when these rules are not applicable. An example of this representation of argumentation schemes can be found in [9]. Thus although the specific critical questions associated with particular argumentation schemes can be used as sources for rules expressing undercutters, the effect is always to offer grounds to pose DMPCQ2, thus homogenising these questions.

If we wish to use this very rule oriented style of reasoning to justify actions we can do so by constructing what is known in game theory as a *strategy profile*. For every situation which can arise in the game, we provide a rule prescribing the action that should be taken. Thus, for noughts and crosses[1], we might have rules such as:

1. Can make a line: *make the line*.
2. Opponent can make line next on next turn: *block line*.
3. Centre is unoccupied: *occupy centre*
4. And so on ...

with the rules listed in order of priority. Now we can justify occupying the centre by saying that it was not possible to make a line, our opponent could not make a line and the centre was unoccupied. To challenge the justification it would be necessary to show that one of the premises was untrue: we have no rule to argue

[1]Called *tic-tac-toe* in the US.

against occupying the centre, and no way of showing Rule 3 to be inapplicable. In other words the justification of any action is simply *I am applying my strategy profile*, and any challenge is only whether the profile is being applied properly, not whether it expresses a desirable way to act. There is no scope to challenge the strategy itself.

This is, of course, a perfectly good way to play, or to write a program to play, a game such as noughts and crosses. It is, however, less attractive if the game is too complicated to permit such a strategy to be developed, or if the strategy is unclear, or if the strategy is not objective. In a game, where the goal is to win, all agents may be presumed to have the same aspirations and so an objective strategy is possible. But in most situations calling for the justification of actions, whether government policies or some personal action, the diversity of aspirations and values of different agents means that justification has to be rather more particular and personalised to the agents concerned: it cannot be assumed that there will be an acceptable, unquestionable strategy that requires no justification for following it.

In order to represent these more sophisticated justifications, argumentation schemes for practical reasoning have been proposed. Instead of simply stating the appropriate response to a list of possible situations, we need to consider both what is true before the action and what will become true as a result of the action. Moreover we need to consider which of the consequences of the action are desirable and which are undesirable, the reasons why these consequences are thought desirable and undesirable, and the relative merits of different options for the agents involved.

An argumentation scheme for practical reasoning with these elements is proposed in [1]. We will refer to this scheme as *PR*:

> *PRP1 Premise:* In the current circumstances R
> *PRPC Conclusion:* We should perform action A
> *PRP2 Premise:* Which will result in new circumstances S
> *PRP3 Premise:* Which will realise goal G
> *PRP4 Premise:* Which will promote some value V.

In this scheme, R is what is true before the action, S is what is true after the action, G is a desirable consequence and V the reason why achieving G from R is desirable. In [1] we can also find seventeen critical questions which can be used to attack justifications using *PR*. Difficulties can arise when interpreting the critical questions: although they may make sense informally, they are not always entirely precise. For example, one critical question against PR is whether the current circumstances hold. As we will see in section 3 this may be asked either because the questioner believes the action not to be currently possible, or because the questioner believes that the action will have different consequences given the circumstances that are actually currently true. To resolve these problems of vagueness and ambiguity, it is necessary to anchor the questions in a well defined semantical structure. This was done in [1], which used Action-based Alternating Transition Systems (AATS) [11] for this purpose, and gave the conditions under which each of the critical questions could be posed in terms of this structure. Using this richer semantical structure enables a wider range of critical questions to be posed than is possible when the representation is only a set of rules.

In this paper we will describe how such a structure can be used to automate the provision of a systematic critique of an argument made using PR based on the critical questions associated with PR. We will illustrate our approach with a running example, relating to the use of speed cameras, based on that in [2], and some fragments of a Prolog program which we have produced to make our definitions operational.

Section 2 will define an AATS, give an instantiation of the AATS for our example, and provide the Prolog predicates required to realise it. Section 3 will step through the critique based on the critical questions of [1]. Section 4 will give a brief discussion and some concluding remarks.

2. Model

As defined in [11], the AATS made no reference to values. In order to adapt it for use with PR, therefore [1] extended the structure to include labels on the transitions indicating which values were promoted and which demoted by following a transition. The resulting structure (an *AATS+V*) can then be defined as follows.

An *Action-based Alternating Transition System* (AATS+V) is an $(n + 9)$-tuple $S = \langle Q, q_0, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Phi, \pi, V, \delta\rangle$, where:

- Q is a finite, non-empty set of *states*;
- $q_0 \in Q$ is the *initial state*;
- $Ag = \{1,\ldots,n\}$ is a finite, non-empty set of *agents*;
- Ac_i is a finite, non-empty set of actions, for each $i \in Ag$ where $Ac_i \cap Ac_j = \emptyset$ for all $i \neq j \in Ag$;
- $\rho : Ac_{Ag} \rightarrow 2^Q$ is an *action pre-condition function*, which for each action $\alpha \in Ac_{Ag}$ defines the set of states $\rho(\alpha)$ from which α may be executed;
- $\tau : Q \times J_{Ag} \rightarrow Q$ is a partial *system transition function*, which defines the state $\tau(q, j)$ that would result by the performance of j from state q – note that, as this function is partial, not all joint actions are possible in all states (cf. the pre-condition function above);
- Φ is a finite, non-empty set of *atomic propositions*; and
- $\pi : Q \rightarrow 2^\Phi$ is an interpretation function, which gives the set of primitive propositions satisfied in each state: if $p \in \pi(q)$, then this means that the propositional variable p is satisfied (equivalently, true) in state q.
- V is a finite, non-empty set of values.
- $\delta : Q \times Q \times V \rightarrow \{+, -, =\}$ is a *valuation function* which defines the status (promoted (+), demoted (−) or neutral (=)) of a value $v_u \in V$ ascribed to the transition between two states: $\delta(q_x, q_y, v_u)$ labels the transition between q_x and q_y with one of $\{+, -, =\}$ with respect to the value $v_u \in V$.

AATSs are particularly concerned with the joint actions of the set of agents. j_{Ag} is the joint action of the set of k agents Ag, and is a tuple $\langle\alpha_1,\ldots,\alpha_k\rangle$, where for each α_j (where $j \leq k$) there is some $i \in Ag$ such that $\alpha_j \in Ac_i$. Moreover, there are no two different actions α_j and $\alpha_{j'}$ in j_{Ag} that belong to the same Ac_i. The set of all joint actions for the set of agents Ag is denoted by J_{Ag}, so $J_{Ag} = \prod_{i\in Ag} Ac_i$. Given an element j of J_{Ag} and an agent $i \in Ag$, i's action in j is denoted by j_i.

2.1. *Instantiating the AATS*

We will use the same example and AATS+V as [2]. To describe a model using the AATS+V we need to specify the various components of the structure. We need the set of propositions Φ with which we can identify the possible member states of Q. Given Φ, we can constrain the size of Q by identifying logical relationships between members of Φ, such that for p_1, $p_2 \in \Phi$, $\neg(p_1 \wedge p_2)$. We need to give the set of agents, Ag and the actions they can perform, so identifying the set of joint actions, J. We need the set of values that may be promoted and demoted by the movement from one state to another. Finally, we need a transition matrix expressing ρ, τ and δ. This matrix comprises a row for each state in Q and a column for each joint action in J. Where there is an entry in a cell the preconditions for the joint action are satisfied (ρ). Entries comprise triples: the state reached if that joint action is executed (τ); the set of values promoted; and the set of values demoted (δ). These transitions are a reflection of a causal theory which explains the effects of various actions, and an evaluative theory which tells us when values are promoted and demoted.

Our example is a justification of an action to be taken by a Government in response to a policy problem, as might be found in an e-participation application. Specifically we consider an issue in UK Road Traffic policy, modelled in [2] and previously used in [6] and [3]. The number of fatal road accidents is an obvious cause for concern, and in the UK there are speed restrictions on various types of road, in the belief that excessive speed causes accidents. The particular issue which we will consider is how to reduce road deaths. One suggestion would be to deter motorists from speeding by introducing speed cameras, which would greatly increase detection and punishment of speeding offences. Points that might be contested are whether fines are sufficient to deter, and whether speeding is an important factor in road accidents. Additionally there are civil liberties issues associated with the loss of privacy resulting from the increased surveillance. A more expensive alternative to speed cameras would be to have a programme of education for motorists which could make them more aware of the dangers of speeding, better able to control their vehicles at speed, or both.

This gives the set of propositions as:

> $\Phi = \{R, S, P\}$, where R is that there are excessive road accidents, S is that there is excessive speeding and P that the intrusions on privacy are unacceptable.

These three propositions give rise to, potentially, eight states. We may, if we wish, exclude one or more of these as impossible. For example if we believe that it is impossible that there should be a reduction in road deaths without a reduction in speeding, the states with $\neg R$ and S would not be possible and would not appear in Q. We also need to identify the current state, q_0, which we take to be $\{R \wedge S \wedge \neg P\}$. The main agents involved are the *Government*, and *Motorists*, considered as a body. In some cases the consequences of action are indeterminate (or at least cannot be determined using the elements we are modelling). To account for this we introduce a third agent, termed *Nature*. The action ascribed to Nature determines the outcomes of the actions of the other agents, where these outcomes are uncertain or probabilistic. The Government has three actions: introducing speed cameras, educating motorists, or doing nothing. Motorists may reduce their speed or do nothing. Nature has two actions according to which fatal accidents are or are not reduced as a result of the Government and motorist actions. For values we consider the cost in terms of human life (l), compliance with the law (c), the financial cost to the Government (b for budget) and the impact of civil liberties (f for freedom). Figure 1 shows the transitions from the current state for the six possible joint actions:

- j_0 Government does nothing, motorists do nothing and nature does nothing.
- j_1 Government introduces cameras, motorists do nothing and nature does nothing.
- j_2 Government introduces cameras, motorists reduce speed and nature reduces accidents.
- j_3 Government introduces cameras, motorists reduce speed and nature does nothing.
- j_4 Government educates motorists, motorists reduce speed and nature reduces accidents.
- j_5 Government educates motorists, motorists do nothing and nature reduces accidents.

Accidents are always reduced when motorists are educated since either they do not speed or can control their vehicles better.

2.2. Prolog Representation

To realise the AATS+V of Figure 1 in a Prolog program we represent the literals corresponding to the propositions in Φ and their negations (to give more flexibility in providing natural language forms), the states in Q, the joint actions J and the transitions defined by τ and δ. Here, as in Figure 1, we show only the fragment representing the transitions from the initial state.

```
%literal(id,positive or negative, english text, other text)
literal(1,1,[there,is,excessive,speeding],[]).
literal(2,0,[speed,limits,are, generally,obeyed],[]).
literal(3,1,[there,are,too,many, road,deaths],[]).
literal(4,0,[road,deaths,are,acceptable],[]).
literal(5,1,[there,are,unacceptable,intrusions,on, privacy],[]).
literal(6,0,[privacy,is,respected],[]).

%state(id, R, S, P). R S and P where the positive or negative
% literals hold in the state. 0 is the current state
state(0,1,3,6).   state(2,2,4,6).      state(3,1,4,6).
state(4,1,3,5).   state(5,2,4,5).      state(6,2,3,5).

%jointAction(id,government,motorist,nature).
jointAction(j0,[do,nothing],[do,nothing],[there,is,no,effect]).
jointAction(j1,[introduce,speed,cameras],[do,nothing],
                [there,is,no,effect]).
```

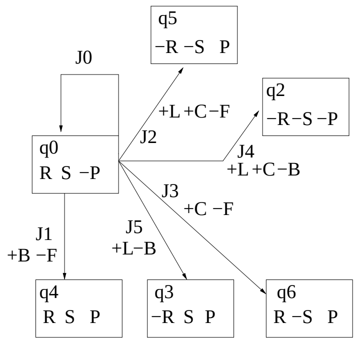

Figure 1. Transitions from q_0

```
jointAction(j2,[introduce,speed,cameras],[reduce,speed],
                [there,are,fewer,accidents]).
jointAction(j3,[introduce,speed,cameras],[reduce,speed],
                [there,is,no,effect]).
jointAction(j4,[educate,motorists],[reduce,speed],
                 [there,are,fewer,accidents]).
jointAction(j5,[educate,motorists],[do,nothing],
                 [there,are,fewer,accidents]).

%transition(id, source state, target state,
%joint action, values promoted, values demoted).
transition(1,0,5,j2,[l,c],[f]). transition(2,0,0,j0,[],[]).
transition(3,0,4,j1,[b],[f]). transition(4,0,2,j4,[l,c],[b]).
transition(5,0,6,j3,[c],[f]). transition(6,0,3,j5,[l],[b]).
```

Note that the Prolog program represents *the model*, not a set of statements about the world. As such questions that can be posed in terms of the model can be generated from the program. We cannot, of course, question (or justify) the model itself: but the scope for generating questions is considerably greater than is the case where we have a only set of rules representing responses to situations.

The generic use case which we have in mind is an interaction between two agents each of which is equipped with a knowledge base representing a model as described above. To allow discussion there will be some agreement, especially as to the elements used to model the situation. In particular we assume that the agents have in common Φ, Ag, Ac_i for each agent, and V. To require discussion the agents can also disagree: they may differ as to ρ, τ and δ. Thus they may differ as to the causal relations, and the evaluation of changes of state, and so may have different `transitions` and may label them differently. Additionally agents may disagree as to the current state, and may, of course, have different preference orderings on V. The agents will interact using a suitable protocol for the exchange of arguments based on the argument scheme and its critical questions.

Alternatively the "agent" giving the justification may be a human. A human user will also be required to express the argument using the elements of the model, but now the responses will be in the form of natural language, generated from fragments of stored text.

3. Critique

In this section we describe the elements of a critique based on our model. To provide a target for our critique, we must first instantiate PR. We can generate justifications of actions from the Prolog progam using:

```
argumentPro(A,S,R,V):-
 transition(ID,S,R,J,X,_),member(V,X),jointAction(J,A,_,_),
 pp([government, should,A,in,S,to,reach,R,and,promote,V]).
```

where `pp` is simply a pretty print function: in an agent-agent setting a performative in the protocol will be used to supply the argument to the other agent. Thus the query `argumentPro(A,S,R,V)` with none of the variables instatiated will produce every possible justification from our model. For example, one answer (Ans1) will be:

```
government should introduce speed cameras in 0 to reach 4 and promote b
A = [introduce, speed, cameras], S = 0, R = 4, V = b ;
```

The number of responses can be restricted by instantiating one or more of the variables. Thus if we believe the current state to be q_3 so that A is `[introduce, speed, cameras]` and S is 3, we will get only the arguments for introducing speed cameras when $\neg R \wedge S \wedge \neg P$ including Ans2:

```
government should introduce speed cameras in 3 to reach 4 and promote c
R = 5, V = c ;
```

The justification, with the literals substituted for the current state and the consequences to accommodate epistemic disagreements, is given to the critiquing agent. The critiquing agent now identifies the assumed start and finish states by matching the literals supplied with its own `state` predicate, and can now supply a critique of n questions using the main predicate:

```
critique(A,S,R,V):- question(1,A,S,R,V), ..., question(n,A,S,R,V).
```

Each question will have its own procedure of the form:

```
question(Q,A,S,R,V):- test(Q,A,S,R,V), respond(Q,A,S,R,V).
question(Q,A,S,R,V).
```

In the first clause for `question`, `test` is used to operationalise the conditions for posing the various critical questions given in [1], and the particular tests for each question will be discussed below. The second clause for `question` will succeed if the first fails, so that `critique` does not fail and the next question can be considered. The `respond` predicates will output whatever is appropriate for the use case, either an utterance in the current protocol, or, where the user is human, some stored text message. We now move through the particular points that make up a comprehensive critique of the justification. The critical question numbers refer to the critical questions presented in [1]: here we use appropriate natural language forms and suggest text messages for the responses.

Is the Action Possible? (CQ1, CQ13). The first thing to check is whether the initial state is agreed. If the state is not q_0, the argument needs to be critiqued as it is not about the current state, although it would hypothetically justify the action from that other state. This gives the simple test:

```
test(1,A,R,_,_):- not(R == 0).
```

This, however, may or may not affect the justification. Consider Ans2. Since `argumentPro([introduce, speed, cameras],0,5,c)` succeeds, the advocated action can still be performed, and so the disagreement is not material, since the action precondition is satisfied in both q_0 and q_3. The critique here is thus a warning rather than a potentially fatal attack: *I believe the current facts to be* R*,* S *and* $\neg P$*, but the action you propose is still possible.* If, however, perhaps because the argument had been generated from a model with a different transition function, the justification is of an action *act* and `argumentPro([act,0,R,V)` fails, *act* cannot be performed in the current situation, and so the critique becomes *I believe the current facts to be* R*,* S *and* $\neg P$*, and so the action you propose is not possible.* This attack is not a warning, but a fatal objection. Thus we define `respond` as:

```
respond(1,A,S,_,_):- argumentPro(act,0,_,_),state(0,F1,F2,F3),
  pp([i,believe,the,current,facts,to,be,F1,F2,F3,but,the,action,
  you,propose,is,still,possible]).
respond(1,A,S,R,V):-pp([i,believe,the,current,facts,to,be,F1,F2,F3,
  and,so,the,action,you,propose,is,not,possible)],fail.
```

The `fail` causes `question` to fail and so terminates the critique if the objection is fatal.

Can the Action have the Stated Effects? (CQ1, CQ2). When there is disagreement as to the current state it is possible that, even though the action can be performed, the consequences will be different. Suppose the justification to be critiqued is Ans3:

```
government should introduce speed cameras in 3 to reach 4 and promote b
A = [introduce, speed, cameras], S = 3, R = 2, V = b ;
```

Now while `argumentPro([introduce, speed, cameras],0,R,V)` succeeds, R is bound to 5 not 2 and V can bind to l or c but not b. Thus the action can be performed, even though there is disagreement as to the current state, and so question 1 is not fatal, but there is now also disagreement as to the state that will be reached and the value that will be promoted. So whereas if the justification had been Ans2, the discrepancy in starting state would make no difference to what was achieved and the value promoted, Ans3 does make this difference and we must offer the critique *Performing A will not result in* $\neg R$*,* $\neg S$ *and* $\neg P$. Whether this is a fatal objection or a warning, however, will depend on our next question.

If the desired value is still promoted, it is not crucially important that we agree on the consequences of the action. Thus if Ans3 had cited l rather than b as the value justifying the action, we can tolerate disagreement about the consequences.

Does the Action Promote the Value? (CQ4). Whether or not the action has the consequences claimed, it may still not promote the desired value: as when Ans3 claims that b is promoted. Since `argumentPro([introduce, speed, cameras],0,_,b)` fails, there is a significant disagreement, and the reason why the action is performed does not apply. Thus we should offer the critique, intended as a fatal objection, that *Performing A will not promote the value b.* Assuming, however, that `argumentPro` does succeed with the instantiations claimed, the justification will have crossed the first hurdle and represent a *prima facie* acceptable instantiation of PR. The critique must now turn to whether the argument is acceptable when set against other objections.

Are There Negative Side Affects? (CQ8, CQ9). Next the other consequences of performing the action are considered. Firstly it is possible that the action will demote some values as well as promoting the identified value. Any such demotion would give us an argument against performing the action. Such arguments can be discovered from the program using

```
argumentCon(A,0,R,W):-
 transition(ID,S,R,J,_,X),member(W,X),jointAction(J,A,_,_),
 pp([government,should,not,A,in,S,to,avoid,R,which,would,demote,W]).
```

If this succeeds with one or more values binding W we will have one or more grounds for objecting to the original justification. For example the demotion of f is an argument against reaching q_5 by introducing cameras, and so would be the basis of an objection to Ans2. How serious the objection is depends on the value demoted. The associated `test` is thus whether `argumentCon` succeeds, but the `response` will depend on the value bound to W.

Most serious is if $W = V$, that is, the value demoted is the value cited as promoted in the justification, since this would mean the action represents a self defeating way of advancing this value. In this case the critique is that *You should not introduce speed cameras since this will demote the value you are trying to promote.* This is intended to be sufficient to defeat the offered justification.

If, on the other hand, $W \neq V$, the objection is only a warning that the side effects should be considered. If the audience considers that original value is still worth promoting despite these problems, the action remains justified for that audience. In this case the response is something like *Performing introduce speed cameras will demote value f. Are you sure it is still worth promoting c?* In this way, an advocate of speed cameras can continue to hold that view, but now does so on the understanding that freedom will be reduced. In this case the agent proposing the justification chooses whether to continue or terminate the critique.

Are there Other Ways to Promote the Value? (CQ7). If there is a $B \neq A$ for which *argumentPro(B,0,nextState,promotedValue)* succeeds with *nextState* and $promotedValue$ as identified in the justification, then it is possible to promote the desired value with a different action. This different action may be preferable to the chosen action, and so a warning should be issued. For example, if the proposal was for education as a way of promoting l, the critique could draw attention to the possibility of promoting the value by introducing cameras with the response *Performing introduce speed cameras will also promote promotedValue. Are you sure you still wish to perform educate motorists?*. This might cause the advocates of eduction to change their mind, particularly if they have already been alerted to the fact that education will take them over budget and so demote b. Again it is the proposer who chooses whether to stop or carry on.

Could Other Values be Promoted? (CQ11). If there is a $B \neq A$ for which *argumentPro(B,0,nextState,otherValue)* succeeds but $otherValue$ is not the value supplied in the justification, then performing the proposed action precludes promoting this other value. Moreover there may be several such values. For our current purposes, we will not object if performing the action promotes the stated value and other values as well (but see CQ10 of [1]), so the values we are interested in for this objection are those which can be promoted in q_0, but are not promoted by an action taking us to *nextState*. Again this is only a warning: it may not matter that the other value is not promoted depending on the way the values are ranked. But a warning should be issued: e.g. for Ans1, since education will promote l: *If you perform introduce speed cameras you will miss an opportunity to promote l. Do you still want to perform introduce speed cameras?*. Now the proposing agent must choose whether to carry on, or act to promote $otherValue$ instead.

Will the Other Agents Do What they are Supposed To Do? (CQ17). Finally, suppose that `argumentPro(action,0,nextState,promotedValue)` succeeds with *action*, *nextState* and $promotedValue$ all as claimed in the justification with `jointAction(joint1, action, motorist1, nature1)` but there is another transition from q_0 such that `jointAction(joint2, A, motorist2, nature2)` where $motorist1 \neq motorist2$ or $nature1 \neq nature2$ or both, which would move us from q_0 to some new state without promoting $promotedValue$. That is, our chosen action may fail to promote the desired value because one or more other agents do something other than what we had anticipated or hoped for. For example, if motorists do not modify their behaviour when cameras are introduced we will reach q_4 instead of the desired q_5. Thus in such cases the final warning *Are you confident that motorists reduce speed and there are fewer accidents will result?* should be proffered. Again this is only a warning: the alternative behaviours may be considered unlikely, or at least worth risking.

4. Discussion

We have shown how a thorough critique of a position intended to justify an action can be generated from a model of actions and their consequences expressed as an AATS+V. The above critique covers nine of the seventeen critical questions of [1]. Of the eight not covered some relate to elements of the AATS+V common to the two agents, in particular propositions, agents and actions. The other missing critical questions concern goals, which are present in PR, but have no correspondence in an AATS. Such goals can be thought of as defined in terms of the propositions in Φ. For example one might consider a society to be civilised if road deaths are not excessive and privacy is respected, i.e $\neg R \wedge \neg P$. Thus the goal of a civilised society could be realised in q_0 by moving to q_2 or q_3. The ability to define complex aspirations in terms of members of Φ might be useful in a more complex domain.

Justifications for action are based on a wide variety of different kinds of knowledge including at least: knowledge as to what is currently the case, knowledge of actions and their effects, awareness of the effect on values, knowledge of what other agents are likely to do and knowledge of preferences between competing values. This diversity is reflected in the range of perspectives from which an action justification can be critiqued. The use of an argumentation scheme such as PR enables the critique to systematically explore possible weaknesses with respect to all these aspects. What we have described above enables the systematic critique to be delivered, but while it raises questions, these are not argued for or resolved: the criticisms are accepted or ignored but not debated. Any deeper exploration would require further argument and nested dialogues relating to different argumentation schemes All the different kinds of knowledge raised by the critique will require their own dialogues. Disagreement about facts [5] and preferences [4] has been investigated, but much remains to be done for discussion of preferences and for dialogues disputing causal theories, agent behaviour and evaluative assessments.

Acknowledgments

This work was partially supported by the FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement Number 247228. The views expressed are those of the authors and are not necessarily representative of the project.

References

[1] K. Atkinson and T. Bench-Capon. Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artif. Intell.*, 171(10-15):855–874, 2007.
[2] K. Atkinson, T. Bench-Capon, D. Cartwright, and A. Wyner. Semantic models for policy deliberation. In *Proceedings of the 13th ICAIL*, pages 81–90, 2011.
[3] T. Bench-Capon and H. Prakken. A lightweight formal model of two-phase democratic deliberation. In *Proceedings of JURIX 2010*, pages 27–36. IOS Press, 2010.
[4] E. Black and K. Atkinson. Choosing persuasive arguments for action. In *Proceedings of AAMAS 2011*, pages 905–912, 2011.
[5] E. Black and A. Hunter. An inquiry dialogue system. *Autonomous Agents and Multi-Agent Systems*, 19(2):173–209, 2009.
[6] D. Cartwright and K. Atkinson. Using computational argumentation to support e-participation. *IEEE Intelligent Systems*, 24(5):42–52, 2009.
[7] H. Prakken. Formal systems for persuasion dialogue. *KER*, 21, 2006.
[8] H. Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.
[9] H. Prakken. Reconstructing popov v. hayashi in a framework for argumentation with structured arguments and dungean semantics. *AI and Law*, 20(1):57–82, 2012.
[10] D. Walton. *Argumentation Schemes for Presumptive Reasoning*. LEA, 1996.
[11] M. Wooldridge and W. van der Hoek. On obligations and normative ability. *J. Applied Logic*, 3(3-4):396–420, 2005.