Fuzzy Argumentation for Trust

Ruben Stranders, Mathijs de Weerdt, and Cees Witteveen

Delft University of Technology
rs06r@ecs.soton.ac.uk, {M.M.deWeerdt,C.Witteveen}@tudelft.nl

Abstract. In an open Multi-Agent System, the goals of agents acting on behalf of their owners often conflict with each other. Therefore, a personal agent protecting the interest of a single user cannot always rely on them. Consequently, such a personal agent needs to be able to reason about trusting (information or services provided by) other agents. Existing algorithms that perform such reasoning mainly focus on the immediate utility of a trusting decision, but do not provide an explanation of their actions to the user. This may hinder the acceptance of agent-based technologies in sensitive applications where users need to rely on their personal agents.

Against this background, we propose a new approach to trust based on argumentation that aims to expose the rationale behind such trusting decisions. Our solution features a separation of opponent modeling and decision making. It uses possibilistic logic to model behavior of opponents, and we propose an extension of the argumentation framework by Amgoud and Prade [1] to use the fuzzy rules within these models for well-supported decisions.

1 Introduction

An open Multi-Agent System (MAS) is characterized by an agent's freedom to enter and exit the system as it pleases, and the lack of central regulation and control of behavior. In such a MAS, agents are often not only dependent upon each other, as for example in Computer-Supported Cooperative Work (CSCW) [2], web services [3], e-Business [4,5], and Human-Computer interaction [6], but their goals may also be in conflict. As a consequence, agents in such a system are not reliable or trustworthy by default, and an agent needs to take into account the *trustworthiness* of other agents when planning how to satisfy its owner's demands.

Several algorithms have been devised to confront this problem of estimating trustworthiness by capturing past experiences in one or two values to estimate future behavior (e.g. see the survey by Dash et al. [7]). These algorithms, however, primarily focus on improving the immediate success of an agent. Less emphasis is laid on discovering patterns in the behavior of other agents, or—more challenging—their incentives. Moreover, the *rationale* of a decision often eludes the user: in most approaches it is 'hidden' in a large amount of numerical data, or simply incomprehensible. At any rate, these approaches do not provide human-readable information about these decisions, and were indeed not designed to do this.

F. Sadri and K. Satoh (Eds.): CLIMA VIII, LNAI 5056, pp. 214–230, 2008.

The following example illustrates the importance of the rationale behind the agent's decision. Suppose a user instructs a personal agent to buy a painting for his collection. When an interesting painting is offered, this agent estimates its value by requesting the opinion from a number of experts. To obtain a good estimate, it then assigns weights to the various received appraisals. When the user plans to buy a very valuable painting, he is not just interested in the final estimate of this agent, or in the retrieved estimates and their weights. When so much is at stake, he wants to know where these weights come from. Why, for example, is the weight for this famous expert so low? If the agent told him that this is because this expert is known to misrepresent his estimate in cases where he is interested in buying himself, and this may be such a case, would not this agent be much more useful than an agent that simply assigns a number to the trustworthiness of the expert?

The lack of such explanations can severely hamper the acceptance of agent-based technology, especially in areas where users rely on agents to perform sensitive tasks. Without the availability of these explanations, the user almost needs to have blind faith in his agent's ability to trust other agents. We believe that the state of the art in dealing with trust in Multi-Agent Systems has not sufficiently addressed this issue. Therefore, we are interested in an approach that lays more emphasis on the rationale of trusting decisions, and in this paper we work towards a proof-of-concept of such an approach.

Due to the uncertainty of information in Multi-Agent Systems, this setting gives rise to some specific requirements of the opponent model an agent should be able to build: (i) The model should be able to represent inherently uncertain, ambiguous, and incomplete knowledge about other agents, and (ii) it should support an argumentation framework capable of making decisions and explaining them. This implies that the opponent model should support logical rules.

We put forward such a model in Section 2, where the core idea of our approach is presented: a unique combination of a fuzzy rule opponent modeling technique and a solid argumentation framework applied to the process of making trust decisions. In this section we also explain how the argumentation framework by Amgoud and Prade [1] can be extended to deal with situations with not only possibilistic rules, but also where the rules themselves are not always fully applicable to a given situation. In Section 3 we show how this model can be applied within the context of an art appraisal domain, as described in the Agent Reputation and Trust (ART) testbed [8]. The final section summarizes the benefits of an argumentation-based approach to explaining trusting decisions, discusses related work, and gives some interesting ways of extending the ideas given in this paper.

2 An Architecture for Fuzzy Argumentation

The goal of the approach presented in this paper is to capture uncertain knowledge about other agents in logical rules, and to use this knowledge to derive not only good decisions, but also arguments to support these decisions. In this section

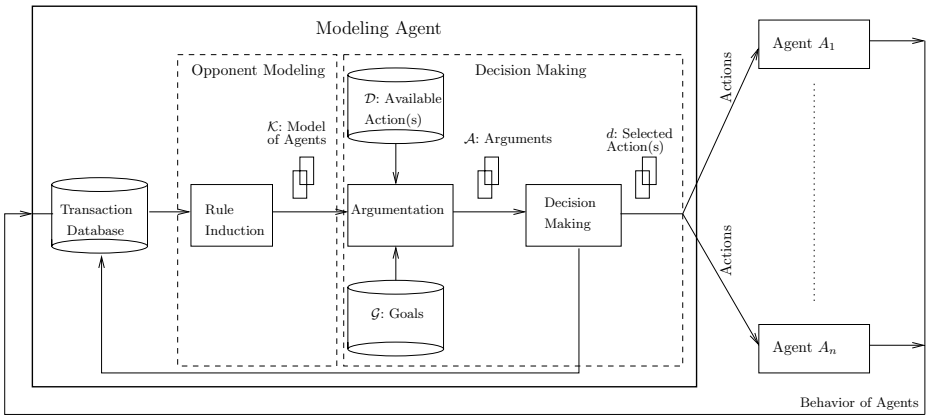

Fig. 1. The architecture for the Modeling Agent

we describe the global architecture of our approach, the formal argumentation framework for making the decisions, and the opponent-modeling algorithm we used in our proof of concept.

2.1 High-level Architecture

Figure 1 shows the architecture of our proposed approach, and introduces some of the terminology used throughout the rest of the paper. The two main components of our framework are *opponent modeling* and *decision making*. The opponent modeling component is responsible for modeling the behavior of other agents, based on past experiences with these agents. Data from these past experiences are stored in a *transaction database*. From this data a knowledge base of rules is induced that models the behavior of each agent. The details of opponent modeling are discussed in Section 2.2.

The decision making component (details in Section 2.3) is responsible for making the actual decisions. Decisions may be supported by *arguments*. An argument relates to a prediction of future behavior of opponents, and is obtained using the opponent models. The extent to which an argument supports a decision is expressed in terms of its *strength*. The strength of an argument is composed of the argument's *weight*, which defines the desirability of the predicted result of this decision, and of its *level*, which is the amount of confidence in the accuracy of the prediction. After having executed the decision that is supported by the strongest argument, the *actual* outcomes are observed and recorded in the transaction database. These new results are subsequently used to refine the model of the opposing agents once again, completing the circle.

2.2 Opponent Modeling

In order to explain an argument for trust to a user, the agent first needs to possess knowledge about other agents. The format in which the knowledge is

expressed should be capable of capturing the inherent vagueness and ambiguousness of information in a trust domain. Fuzzy (or possibilistic) logic [9] is an adequate tool to tackle this modeling problem, because it provides a natural way of translating back and forth between logical rules describing the expected (or learned) behavior of other agents, and uncertain numerical data.

For brevity and clarity, we omit the details of the specific variety of fuzzy logic used in the knowledge bases of our agent, and instead focus on the intuition and the ideas behind our approach. Therefore, it is sufficient to know that a fuzzy proposition is a statement of the form "property x is high", meaning that x is a member of the fuzzy set "high". A formula in our fuzzy language $\mathcal{L}$ can be composed of such elementary statements using the fuzzy logic operators that intuitively extend the semantics of standard propositional logic to the fuzzy domain.

Now, the knowledge base to model the other agents consists of such fuzzy formulas that each describe a specific aspect of another agent's behavior. However, since such knowledge is constructed based on past interactions with other agents, not all of these learned formulas will have the same status; the inherent unreliability and unpredictability of other agents might cause our agent to add imprecise or even incorrect rules to its knowledge base. Therefore, we also add a confidence value to each formula in the knowledge base to represent the certainty with which the formula has been learned.

Definition 1. *A* knowledge base $\mathcal{K}$ *is a set of tuples* (k_i, ρ_i) *where* $k_i \in \mathcal{L}$ *is a fuzzy formula, and* $\rho_i \in [0, 1]$ *is the confidence the agent has in* k_i*.*

The valuation of a fuzzy formula depends on a given state of the world w. Such a state w is a description of the current state of the environment by a set of propositions. In our application, a world state represent the actions of our agent towards other agents in the past, which might influence their behavior in future interactions. Given a world state, the extent to which a fuzzy formula is valid can be determined using a valuation function, which assigns a measure of applicability to each formula.

Definition 2. *Given a world state* w*, the* valuation function $v_w : \mathcal{L} \to [0, 1]$ *gives the applicability of a fuzzy formula in the world* w*.*

In most situations, the knowledge base consists of fuzzy *rules*, i.e. a material implication from an observation (condition) to an expected/learned effect (conclusion). Such a rule can be *partially* applicable in a particular world state, instead of just being fully applicable or not at all. If k_i is a fuzzy rule, we say that $v_w(k_i)$ is the *match strength* of k_i. Consider the following example of such a fuzzy rule.

Example 1. Suppose we own a (possibly) very valuable painting, and we would like to have it appraised by taking a weighted average over a set of appraisal agents. Each of these agents not only gives the appraisal itself, but also a claim on its certainty about this appraisal. To know which agents to trust, we look at

their behavior in the past. Such previous interactions have led us to believe that the following rule k_1 accurately describes one of the agent's (a) behavior: "**if** a says it is certain with level c_{high} (very certain) **then** agent a's appraisal error ae_{low} (low)". This rule should be interpreted as follows: if agent a's certainty is a member of the fuzzy set c_{high}, which contains all high values of certainty, its appraisal error will be a member of the fuzzy set ae_{low}, containing all low values of appraisal error. So, if this agent claims it is very certain about its appraisal, we conclude that its appraisal error is low, and we will base our own estimate strongly on this agent's appraisal.

As hinted above, *membership* of a fuzzy set is not just true or false, but can take on a range of values between 0 and 1. Suppose that in a certain world w agent a's certainty c is not exactly c_{high}, but slightly lower. In that case c's membership of c_{high} is less than 1. Rule k_1 still applies, but its match strength $v_w(k_1)$ will also be less than one. In this case, we say the rule fires *partially*, and consequently we cannot predict that the appraisal error will be exactly ae_{low}. To be more precise, the membership of the actual appraisal error in ae_{low} is less than 1. In plain English, this implies that we should expect an appraisal error that is not low, but slightly higher.

At this point, it is important to note the difference between the confidence ρ_i in a rule k_i, and its match strength $v_w(k_i)$ for a certain world state w. The former represents the *validity* of the rule in describing a *certain system or agent*, and the latter represents the *applicability* of a rule to the system or agent in a particular *state of the world*. In the previous example, rule k_1 might not be valid at all for describing a's behavior. Put differently, the rule could be wrong, in which case the confidence ρ_1 should be close to 0. On the other hand, given a certain scenario (for example, in which certainty equals c), rule k_1 could be used to predict the behavior of the agent, *provided that* c is a member of c_{high}. Otherwise, the preconditions of the rule are not met, and the rule does not apply to the world state w. As a result, the rule's match strength $v_w(k_i)$ is zero.

Keeping in mind the requirements identified in the introduction (the ability to model uncertainty and at the same time support an argumentation framework), we decided to use a simple theory revision algorithm called Fuzzy Rule Learner (FURL) [10] to construct such a knowledge base containing the observed behavior of the other agents. Taking observations from the environment as input, FURL is capable of creating a rule base of fuzzy rules. FURL's output consists of a multi-level rule base known as a Hierarchical Prioritized Structure [11] and, for each rule, the prediction error it causes on past observations (the training set). Rules in each level can be thought of as an exception to rules in the layer below it. For our application, however, we can think of the result just as a (flat) rule base with fuzzy rules and their prediction error where the confidence values are taken to be the inverse of the prediction error of the rule according to FURL.

2.3 Decision Making

In this section we introduce the argumentation framework used in the decision making component. The work by Amgoud and Prade [1,12] was considered to

be a good basis for such a framework, because it inherently supports reasoning under uncertainty with fuzzy logic. This framework uses the agent's knowledge base $\mathcal{K}$, a set of its goals $\mathcal{G}$, and a set of possible decisions (or actions) $\mathcal{D}$. An argument A in favor of a decision $d \in \mathcal{D}$ is then defined as follows [1,12].

Definition 3. *Given an agent* $\langle \mathcal{K}, \mathcal{G}, \mathcal{D} \rangle$*, an* argument A in favor *of a decision* $d \in \mathcal{D}$ *is a triple* $A = \langle S, C, d \rangle$*, where*

- $S \subseteq \mathcal{K}$ *is the support of the argument, containing the knowledge from the agent's knowledge base* $\mathcal{K}$ *used to predict the consequences of decision* d*,*
- $C \subseteq \mathcal{G}$ *are the consequences of the argument, i.e. the goals reached by decision* d*, and*
- $d \in \mathcal{D}$ *is the conclusion argument* A *recommends.*

Moreover, $S \cup \{d\}$ *should entail* C*,* S *should be minimal, and* C *maximal among the sets satisfying the above conditions.*

The original framework proposed in [1] requires that the support S should be *consistent* with d. That is, applying d should not result in a contradiction with previously acquired knowledge. However, the original framework is based on propositional logic, whereas our method uses fuzzy logic. In contrast to propositional logic, applying a decision d on a fuzzy knowledge base $\mathcal{K}$, will not result in a contradiction, regardless of the contents of $\mathcal{K}$. The consistency requirement is therefore no longer relevant. This is due to the fact that a fuzzy rulebase is inherently capable of resolving inconsistencies. More specifically, when multiple rules fire at the same time, with different outputs, these outputs are fused together and converted into a scalar using a process called *defuzzification* [13].

The set $\mathcal{A}$ gathers all arguments that can be constructed from $\mathcal{K}$, $\mathcal{G}$, and $\mathcal{D}$ as follows: for each decision $d \in \mathcal{D}$, the consequences $C \subseteq \mathcal{G}$ are predicted using a subset S of the knowledge base $\mathcal{K}$, resulting in an argument $\langle S, C, d \rangle$. Subsequently, a decision is made by selecting the argument(s) with the highest strength.

The process of reaching a decision can be determined in four steps:

1. The *level* of the argument is calculated based on the confidence in support S. Remember from the previous section that the confidence in a rule depends on how well it models another agent's behavior.
2. The *weight* (or desirability) of the outcomes C is evaluated in light of the goals of the agent (or those of its owner).
3. The *level* and the *weight* of each argument are combined in its *strength.* Strength can be considered as a summary of the argument's validity and the desirability of the predicted outcomes of the decision it supports.
4. The decision supported by the argument with the highest strength is selected.

We will now discuss each step in more detail.

In the original framework [1,12], the *level* of an argument solely referred to the amount of *confidence* in the rules and facts in the support of the argument: $\mathit{Level}\,(\langle S, C, d \rangle) = \min\{\rho_i \mid (k_i, \rho_i) \in S\}$. However, in our model, rules in the

knowledge base cannot be applied regardless of the state of the world. Often, their precondition matches only partially with the facts in this state (as in Example 1). Therefore, our definition of the level of an argument needs to take care of the balance between this match strength in an environment w and the confidence of the rules in the knowledge base:

1. For equal confidence levels ρ, the knowledge rule with the highest match strength should determine the *Level* of the argument. The higher the match strength, the more the knowledge applies to the current world state, and the more reliable it is in this particular case.
2. For equal match strengths v_w, the knowledge rule with the lowest level of confidence should determine the *Level* of the argument. This is consistent with the argumentation framework presented in [1].
3. In cases where a rule is fully matched, or not matched at all (e.g. $v_w(k) \in \{0, 1\}$), our definition should reduce to the definition of *Level* in the original framework.

Therefore we base the level of an argument on the weakest part of the argument. In our case, the weakest rule in a support set S given a world state w has not only a low confidence, but also a high match strength.

Definition 4. *Given a world state w, and a support set S, the weakest rule (k_j, ρ_j) is obtained by:*

$$(k_j, \rho_j) = \underset{(k_i,\rho_i)\in S}{\arg\min} \left\{ \frac{\rho_i}{v_w(k_i)} \;\middle|\; v_w(k_i) \neq 0 \right\}. \tag{1}$$

We define the *level* of an argument as the product of the confidence and the match strength of this weakest rule.

Definition 5. *Given a current world state w, the* level of an argument $A = \langle S, C, d\rangle$ *is defined by $Level_w(A) = \rho_j \cdot v_w(k_j)$, where (k_j, ρ_j) is the weakest rule.*

It is easy to check that this definition meets all three of the requirements stated above: the rule with the lowest confidence level and the highest match strength is selected, and the resulting level is the confidence level of this rule times multiplied by its match strength.

The *Weight* of an argument A depends on the goals that can be reached. The goals are given as tuples (g_j, λ_j) in the set $\mathcal{G}$. Like an element from the knowledge base, a goal g_j is a fuzzy rule or fact. The attached value $0 \leq \lambda_j \leq 1$ denotes the preference of the goal. In the original definition [1], weight is inversely proportional to the preference of the most important goal that is not satisfied. However, when using fuzzy logic, the predicate *satisfied* becomes fuzzy as well, making this definition very difficult to apply. We therefore chose to re-establish a similar relation between weight on the one hand, and preference and goal satisfaction on the other hand. One of the key properties of such a relation is that the more important a goal is, the more detrimental the reduction in weight when the goal becomes less satisfied. This is realized by the following definition.

Definition 6. *Given a current world state w, the* weight of an argument $A = \langle S, C, d\rangle$ *is defined by*

$$Weight_w(A) = \sum_{(g_j,\lambda_j)\in C} v_{w\cup S}\left(g_j\right)\cdot\lambda_j. \tag{2}$$

This definition ensures that the weight of the argument is proportional to the utility of the expected consequences of the decision. More specifically, if a goal g with preference λ is 50% true, we expect the utility to increase with $\lambda/2$. We sum over all goals of the agent to obtain the weight of the argument. As Section 3.2 shows, this also brings about a more intuitive trade-off between (possibly conflicting) goals.

Finally, the *Weight* and *Level* of an argument are combined into its the strength. Here we just follow the original definition [1].

Definition 7. *Given a current world state w, the* strength of an argument *is defined by* $Strength_w(A) = Level_w(A)\cdot Weight_w(A)$.

Such a value of *Strength* can then be used to determine which argument is more preferred.

Definition 8. *Let A and B be two arguments in $\mathcal{A}$. Argument A* is preferred to *B iff $Strength(A) \leq Strength(B)$.*

The upcoming section illustrates how an agent built according to this architecture operates in a simple problem domain.

3 Examples

In this section, we would like to investigate how an agent based on our approach behaves in a simple art appraisal environment. We assume the environment is inhabited with other agents with fixed strategies, and show that it is capable of explaining its decisions in terms of aggregated observations (rules).

The Agent Reputation and Trust (ART) testbed provides a simple environment to do our experiments [8]. ART is becoming the *de facto* standard for experimenting with trust algorithms and evaluating their performance. In this environment our personal agent is put in competition with N other agents to estimate the true value v of a painting. Each agent has its own area of expertise for which it can give good opinions to others. Consequently, it is often wise to consult other agents for an estimate of the value of the painting. Each other agent i can send a tuple (c_i, e_i) to our agent upon request where e_i is the estimate, and c_i is the *claimed* certainty of this estimate (c_1 being a low certainty and c_6 a high certainty). Our agent then should combine these estimates in its own appraisal by submitting a weight vector $\mathbf{w} = \{w_1, w_2, \ldots, w_N\}$ to the testbed, where $w_i \geq 0$, and $\sum_i w_i = 1$. The weight for an agent i should not only depend on its claimed certainty c_i, but also on its trustworthiness. Our slightly

modified version of the testbed[1] subsequently calculates the weighted average of the estimates to obtain the final appraisal $a = \sum_{i=1}^{N} w_i e_i$.

As agents are rewarded based on the accuracy of their appraisals, they should aim to find the weight vector $\mathbf{w}$ that minimizes the appraisal error for a painting with true value v:

$$\mathbf{w} = \arg\min_{\mathbf{w} \in \mathbb{R}^N} \left| v - \sum_{i=1}^{N} w_i e_i \right| \tag{3}$$

To determine a suitable $\mathbf{w}$, our agent attempts to find a relation between the claimed certainty c_i, and the accuracy $|v - e_i|$ for each agent i. Now, since agents compete with each other for a number of rounds (appraising different paintings), it may be worthwhile to deceive other agents misrepresenting the claimed certainty at some point. Needless to say, this creates an issue of trust. Knowing when and whom to trust is therefore a prerequisite for success in this domain.

In the two scenarios that follow, we study the decision making process of our agent while in competition with two other agents: HONEST and RECIPROCAL. HONEST is an agent that honestly asserts a certainty c proportional to its expected accuracy, i.e. $c_{\text{HONEST}} \propto |v - e_{\text{HONEST}}|$.

RECIPROCAL's behavior is somewhat more complicated. When an opponent has misrepresented its expertise by overstating its certainty, RECIPROCAL responds in kind by being dishonest as well. However, if RECIPROCAL's opponent is honest, RECIPROCAL behavior towards that opponent is identically to that of HONEST.

In each of the following scenarios, our agent has interacted with both agents in 200 transactions. From the observations made during these 200 transactions, we used FURL to build an opponent model which constitutes the knowledge base $\mathcal{K}$. The knowledge bases use three different fuzzy domains: c_0 to c_5 denote very low to very high certainty, ae_0 to ae_7 denote eight different levels of the appraisal error from low to high, and d_0 to d_5 denote the levels of dishonesty of our agent in the previous round, also from low to high.

To give an example, Figure 2 shows the accuracy of the predictions made by the model learned by FURL from observing RECIPROCAL's behavior. During this run, our agent 'tests' RECIPROCAL by alternating between honest and dishonest behavior towards it. As can be observed from Figure 2, FURL is reasonably capable of learning the effect of dishonesty on RECIPROCAL's behavior. At the end of the run, the learned model contains multiple fuzzy if-then rules describing the behavior, together with a confidence measure. To show what an opponent model looks like, Tables 1 and 2 contain a selection of rules from the opponent models of RECIPROCAL and HONEST after 200 transactions.

[1] In our preliminary experiments, estimates are generated deterministically, instead of being drawn from a probability distribution. That way, we could significantly reduce the length of each competition run and still obtain useful results on the explanatory power of the arguments generated by our agent.

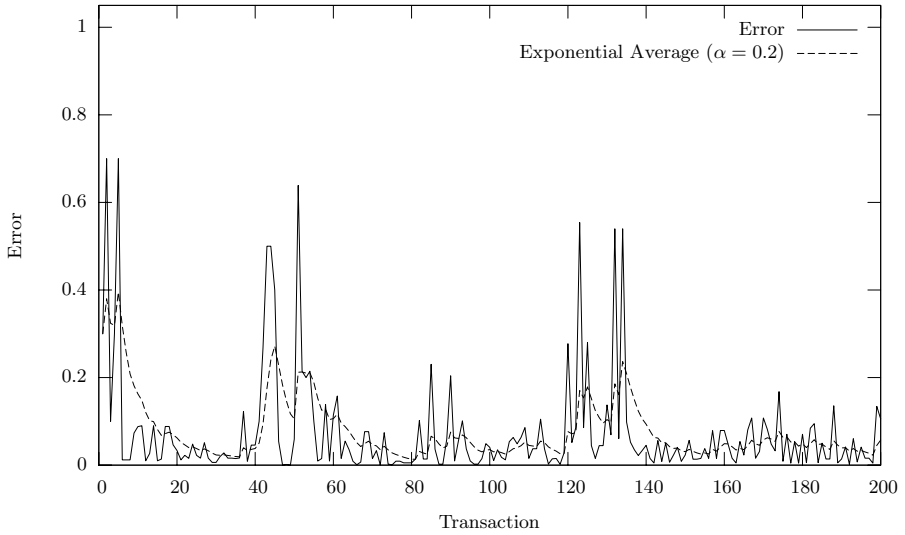

Fig. 2. Prediction errors of the learned opponent model for RECIPROCAL

Table 1. Model of HONEST's behavior after 200 interactions. These learned rules describe the fact that if HONEST claims that its certainty is low (c_0), its error is usually quite high (ae_5), and vice-versa.

#	Rule	Confidence
1	**if** certainty is c_0 **then** appraisal-error is ae_5	0.00381
	...	
6	**if** certainty is c_5 **then** appraisal-error is ae_0	0.00520

Table 2. Model of RECIPROCAL's behavior after 200 interactions. These learned rules describe how RECIPROCAL works approximately. For example, rule 13 describes that if the claimed certainty is moderate, and our agent was honest itself in the previous round (d_0), then the appraisal error is quite low (ae_0).

#	Rule	Confidence
1	**if** certainty is c_0 **then** appraisal-error is ae_7	0.09824
7	**if** certainty is c_6 **then** appraisal-error is ae_2	0.01403
12	**if** certainty is c_1 **and** dishonesty is d_6 **then** appraisal-error is ae_6	0.05282
13	**if** certainty is c_2 **and** dishonesty is d_0 **then** appraisal-error is ae_0	0.03136
26	**if** certainty is c_3 **and** dishonesty is d_6 **then** appraisal-error is ae_5	0.04653

Using the opponent model, the agent is able to make a decision about trusting its opponents in the next transaction. More specifically, it chooses a weight vector $\mathbf{w}$ to determine how the opponents appraisals are combined. Obviously, the optimal decision is to assign all the weight to the agent that is most skilled at appraising the painting (in which case $w_{i^*} = 1 \Leftrightarrow i^* = \arg\min_i |v - e_i|$, and $w_i = 0 \Leftrightarrow i \neq i^*$). However, because of possible noise in the environment or

suboptimal learning capabilities, determining i^* is not a trivial task. Using the argumentation framework, our agent is able to find a balance between utility of the outcome of a decision, and the confidence in the knowledge used to predict these outcomes.

3.1 Scenario 1: Requester Role

In this example scenario our agent consults HONEST and RECIPROCAL to appraise one of its paintings. For each agent, it constructs arguments to support the desirability of obtaining an estimate from both HONEST and RECIPROCAL. The strengths of these arguments are used to determine the delegation weights $\mathbf{w} = \{w_{\text{HONEST}}, w_{\text{RECIPROCAL}}\}$. In what follows, we focus on our agent's goals, and its available decisions. These, combined with the actual observations during a transaction determine the strength of the arguments supporting the decisions and subsequently the delegation weights.

Because it is in our agent's interest to appraise the painting as accurately as possible, its goal set $\mathcal{G}$ contains a single goal $g_1 =$ (appraisal-error is *acceptable*, 1), where *acceptable* is a fuzzy set with a membership function that is inversely proportional to the relative appraisal error $ae_i = \frac{|v - e_i|}{v}$. Put differently, goal g_1 states that our agent favors accurate appraisals from its opponents. Since g_1 is the only goal, it has maximal relative priority.

As mentioned before, the claimed certainty c is a key indicator of the expertise of agent i. In this example, suppose that HONEST returns a numerical value that is 100% member of fuzzy set c_1, meaning that it is quite uncertain (see Figure 3 for the fuzzy partitioning of the certainty domain), while RECIPROCAL replies that it can appraise the painting with a certainty between c_4 and c_5. Also, we know that in the previous round, our agent has somewhat misrepresented its certainty towards RECIPROCAL (dishonesty was a member of the fuzzy set d_3).

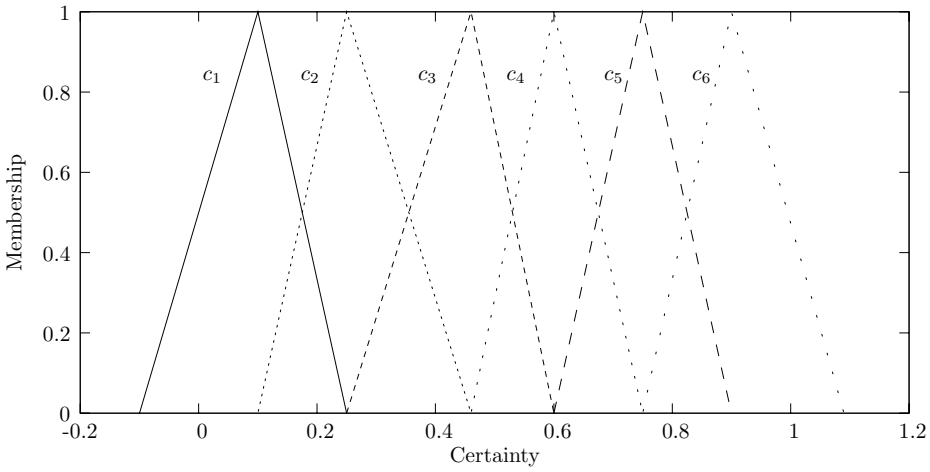

Fig. 3. Example fuzzy partitioning of the *certainty* domain

Table 3. The support for Argument A_{HONEST}

Knowledge	**Match**	**Confidence**
certainty is c_1	100%	
if certainty is c_1 **then** appraisal-error is ae_4	100%	0.00832
appraisal-error is ae_4	100%	

Table 4. The consequences, and the *Level* and *Weight* calculations of argument A_{HONEST}

Goal	**Match**	**Preference**
g_1	0.25	1

(a) The consequences of Argument A_{HONEST}

Property	**Calculation**	**Result**
$Level(A_{\text{HONEST}})$	1×0.00832	0.00832
$Weight(A_{\text{HONEST}})$	1×0.25	0.25

(b) The *Level* and *Weight* calculations of Argument A_{HONEST}

Our agent then has to consider two possible decisions in the set of decisions $\mathcal{D} = \{d_{\text{HONEST}}, d_{\text{RECIPROCAL}}\}$: to accept the estimate from HONEST, and to accept the estimate from RECIPROCAL. Because our agent can weigh the estimates received from both agents, these decisions are not mutually exclusive. For example, our agent can decide to weigh the appraisals from both agents equally, resulting in a final appraisal that is the average of both agent's appraisals.

Based on the claimed certainties from both agents, and the models in Tables 1 and 2, we see that our agent expects a poor appraisal from HONEST. On the other hand, RECIPROCAL's certainty is very high, but our agent has to take into account its own dishonesty towards RECIPROCAL. Using the goals $\mathcal{G}$, the knowledge base $\mathcal{K}$ (containing the model and the observations), and decisions $\mathcal{D}$, our agent generates two arguments. The first argument A_{HONEST} supports decision d_{HONEST}, the second argument $A_{\text{RECIPROCAL}}$ supports decision $d_{\text{RECIPROCAL}}$.

The support of A_{HONEST} consists of parts of the opponent model of HONEST relevant to this particular transaction. This is summarized in Table 3. The *consequences* of A_{HONEST} relate to the desirability of the consequences of decision d_{HONEST} in terms of the agent's goals. For a certainty of c_1, a single rule in the opponent model fires, and predicts an appraisal error of ae_4 (last row in Table 3). Given this prediction, we can determine the utility in terms of goal g_1 (see Table 4(a)). When we defuzzify ae_4, we obtain a numerical value of 0.75.[2] Using the membership function of *acceptable*, we determine that goal g_1 is only 25% satisfied. From the information in Tables 3 and 4(a), we can now calculate the *Level,* relating to the desirability of the consequences, and *Weight,* relating to the confidence of argument A_{HONEST} (see Equation 2): Table 4(b) lists the steps for this calculation. From this, our agent can now determine the strength of the argument for HONEST: $0.00832 \times 0.25 = 0.00208$ (see Definition 8).

[2] Defuzzification is a mapping from membership of one or more fuzzy sets to the original domain. There are a couple of ways to do this, but often the center of gravity of the membership functions is taken [9].

Table 5. The support for Argument $A_{\text{RECIPROCAL}}$

Knowledge	**Match**	**Confidence**
certainty is c_4	50%	
certainty is c_5	50%	
dishonesty is d_3	40%	
if certainty is c_4 **then** appraisal-error is ae_3	50%	0.00876
if certainty is c_5 **then** appraisal-error is ae_2	50%	0.01042
if certainty is c_4 **and** dishonesty is d_3 **then** appraisal-error is ae_0	40%	0.01640
if certainty is c_4 **and** dishonesty is d_3 **then** appraisal-error is ae_1	40%	0.01640

Table 6. The consequences, and the *Level* and *Weight* calculations for argument $A_{\text{RECIPROCAL}}$

Goal	**Match**	**Preference**
g_1	0.75	1

(a) The consequences of Argument $A_{\text{RECIPROCAL}}$

Property	**Calculation**	**Result**
$Level(A_{\text{RECIPROCAL}})$	0.5×0.00876	0.00438
$Weight(A_{\text{RECIPROCAL}})$	1×0.75	0.75

(b) The *Level* and *Weight* calculations of Argument A_2

Next, our agent performs the same steps for RECIPROCAL. For determining the support and consequences of argument $A_{\text{RECIPROCAL}}$, we follow the same procedure as above. These are summarized in Tables 5 and 6(a), respectively. This time, four rules fire based on the certainty received from RECIPROCAL and our agent's dishonesty towards it in the previous round. The resulting appraisal error is expected to be somewhere between ae_0 and ae_3. This corresponds with a 75% satisfaction of goal g_1. Table 6(b) shows the calculation of the *Level* and *Weight* of this argument. Based on these measures, we now calculate the strength of the argument: $0.00438 \times 0.75 = 0.00329$.

In the final step, our agent compares the strengths of both arguments. This is done in Table 7. After normalizing these strengths, we obtain the weight vector $\mathbf{w}$. From Table 7 it can be seen that RECIPROCAL determines 61% of the appraisal. Evidently, our agent favors a low appraisal error, while taking the reduced confidence of the knowledge of RECIPROCAL's behavior for granted.

In this scenario, it has been demonstrated that our agent is able to make a trade-off between an agent whose behavior can be reliably predicted (HONEST) and an agent for which a less reliable opponent model is available, but which probably provides a more accurate appraisal (RECIPROCAL). The strengths of the arguments supporting both decision reflect this trade-off. In the end, the lower predicted appraisal error for RECIPROCAL proved to be decisive.

Table 7. The delegation weights for HONEST and RECIPROCAL in scenario 1

Agent	**Level**	**Weight**	**Strength**	**Delegation weight**
HONEST	0.00832	0.25	0.00208	0.39
RECIPROCAL	0.00438	0.75	0.00329	0.61

3.2 Scenario 2: Provider Role

In the previous scenario, we focused on the appraisals received from the opponents. Now, we also include another type of decision. Other agents may ask our agent for an appraisal. In such a situation, our agent needs to decide how truthfully it should report its estimate of the value of their paintings. To this end, we add a new goal, and apply the decision making procedure to the appraisals generated by *our agent*. The new goal, called g_2, essentially encourages our agent to be as deceptive as possible towards other agents (by overstating the accuracy of the provided estimate). However, when other agents discover this behavior, they may give our agent worse estimates as well. Deceiving other agents must not negatively influence the accuracy of appraisals received from those agents *too much*. Consequently, we must find a balance between the members of the new goal base $\mathcal{G} = \{g_1, g_2\}$.

Deciding the extent of the deception towards an opponent differs from deciding delegation weights in scenario 1 in that the certainty variable is not relevant. In the previous scenario, the agent wanted to predict the appraisal error based on the claimed certainty of its opponents. Now, the agent attempts to predict the effect of its own dishonesty on the appraisal error in the *next* round. Therefore, the opponent models in Tables 1 and 2 need first to be made independent of the certainty variable.

This is done by generating a set of arguments for each available decision for a number of *certainty* values.[3] This way, we effectively factored out the *certainty* variable from the opponent model, while the relation between *dishonesty* and *appraisal error* remains intact. The *Level* and *Weight* of each of these arguments are averaged to obtain an aggregated *Level* and *Weight*. The recommended decision is then calculated in the normal fashion. Of course, deciding on the amount of deception towards HONEST is trivial, because HONEST does not respond to the behavior of its opponents.[4] Because of this, our agent is capable of being totally dishonest with this agent, without surrendering accuracy. In what follows, we therefore illustrate this process by calculating the best level of deception towards RECIPROCAL.

In addition to goal g_1 from scenario 1, goal g_2=(dishonesty is *deceptive*, 0.5) is included in the goal base $\mathcal{G}$ of our agent. The membership of the fuzzy set *deceptive* is proportional to the extent to which the agent misrepresents its expertise by overstating its certainty. Note that goal g_2 has a lower priority than goal g_1.

We consider five different decisions: d_A, i.e. dishonesty is 0.0, d_B, i.e. dishonesty is 0.25, ..., and d_E, i.e. dishonesty is 1.0. Table 8 shows the arguments generated for each decision. We see that the extent of our agent's dishonesty towards RECIPROCAL influences the average appraisal error. Of course, due to the nature of RECIPROCAL, this is to be expected, because it punishes dishon-

[3] More specifically, we generated an argument for 100 equally spaced values of 'certainty' between 0 and 1.

[4] This is reflected in Table 1, which shows only a relation between certainty and the appraisal error.

Table 8. Properties of the set of arguments supporting different values of dishonesty towards RECIPROCAL

	Dishonesty	**Appraisal Error**	**Goal Satisfaction**		**Level**	**Weight**
			g_1	g_2		
d_A	0.00	0.63	0.37	0.00	1.49	0.37
d_B	0.25	0.75	0.25	0.25	1.49	0.38
d_C	0.50	0.85	0.15	0.50	1.49	0.40
d_D	0.75	0.87	0.13	0.75	1.49	0.51
d_E	1.00	0.85	0.15	1.00	1.49	0.65

esty by responding in kind. Consequently, increasing dishonesty while keeping the certainty constant, the appraisal error increases.

The interesting aspect of this scenario is the trade-off between goals g_1 and g_2. Our agent has to decide what it values most: an accurate appraisal *from*, or its deception *towards* RECIPROCAL. With this particular goal base and its associated priorities, we conclude from Table 8 that our agent favors the latter. Decision d_E is preferred based on the fact that it has the highest weight.

4 Discussion

In this paper we showed how arguments can be based on fuzzy rules. This generalization of Amgoud and Prade's argumentation framework [1] is able to come up with a reasoning for each of the possible decisions. We showed how the confidence and match strength of the underlying rules, and the priority of the decisions influence the decisions of our agent. Combined with a fuzzy rule learner this argumentation framework forms a unique method for agents to reason about trust, and provide a logical explanation for the actions (to be) taken.

Existing work on opponent modeling in the context of trust uses scalars or small vectors to represent trust. For example, in FIRE [14] the *quality* and the *reliability* of past transaction-results are derived and used for future decision making. An application of the Dempster-Shafer theory collects evidence of trustworthiness [15], and another approach using probabilistic reciprocity captures utility provided to and received from an agent [16], or the probability that task delegation towards an agent is successful [17]. Because of the limited amount of information present in these models, much of the information gathered during interacting with an opponent is lost. Consequently, the decision models they support are quite limited.

An example where the model of trust is more elaborate can be found in the work by Castelfranchi et al. [6,18], where trust is decomposed in distinct beliefs. Such a more complex model would open up the possibility of implementing different intervention strategies, depending on the precise composition of trust, instead of just having a binary choice: delegation or non-delegation. However in their approach the reasons *why* an agent is trusted are still not very clear. An owner of an agent that uses a so-called fuzzy cognitive map is confronted with a list of specific beliefs on parts of the model of the other agent, such as the other's

competence, intentions, and reliability. It is not clear where these beliefs come from, and no method is given to learn such beliefs from past interactions. For this, we need to trace back the process that established a certain decomposition of trust for a specific agent. We believe that our approach forms a good basis to include such a more elaborate model of trust, but this may require a more advanced fuzzy rule learning algorithm.

Improving the opponent modeling algorithm is one of our goals for future work. The FURL algorithm we used in our approach has a number of limitations. Most importantly, FURL is incapable of detecting relatively complex behavior. It is not able to accurately model data sets with a large number of input variables as can be seen from the extensive experiments in our technical report [19].

In contrast to the decision model of Castelfranchi et al., the modified doxastic logic for Belief, Information acquisition, and Trust (BIT) [20] is more capable of explaining why certain facts are believed. For example, using BIT, an agent could be able to present the rationale of the decision to trust another. In terms of our aim, this is very appealing. However, due to the inherent uncertain, vague and continuous nature of observations in a Multi-Agent System it is not trivial to translate these to BIT. In this paper we showed how to make such a translation to fuzzy logic. Modal logic has no 'native' support for *directly* representing such observations, but possibly the ideas of our architecture can be reproduced in the context of modal logic.

As a final note, in the current work we have only used arguments in favor of a decision. The framework, however, also allows for contra-arguments, allowing for much more complex argumentation. Maybe even more interesting would be to add support for reputation in our approach. This would involve broadening our model, designing new algorithms to select agents from which reputation information is requested, and developing an algorithm to aggregate these reputations.

Acknowledgements

This work is partially supported by the Technology Foundation STW, applied science division of NWO, and the Ministry of Economic Affairs of the Netherlands.

References

1. Amgoud, L., Prade, H.: Using arguments for making decisions: a possibilistic logic approach. In: AUAI 2004: Proceedings of the 20th conference on Uncertainty in artificial intelligence, pp. 10–17. AUAI Press (2004)
2. Barthés, J.A., Tacla, C.: Agent-supported portals and knowledge management in complex R&D projects. In: Sixth International Conference on CSCW in Design (CSCWD 2001), pp. 287–292 (2001)
3. Maximilien, E., Singh, M.: Reputation and endorsement for web services. ACM SIGEcom Exchanges, 24–31 (2002)

4. Resnick, P., Zeckhauser, R.: Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system. In: Working Paper for the NBER Workshop on Empirical Studies of Electronic Commerce (2000)
5. Guttman, R., Moukas, A., Maes, P.: Agent-mediated electronic commerce: a survey. The Knowledge Engineering Review, 147–159 (1998)
6. Castelfranchi, C., Falcone, R.: Social trust: A cognitive approach. In: Castelfranchi, C., Tan, Y. (eds.) Trust and Deception in Virtual Societies, pp. 55–90. Kluwer Academic Publishers, Dordrecht (2001)
7. Dash, R.K., Ramchurn, S.D., Jennings, N.R.: Trust-based mechanism design. In: AAMAS 2004, pp. 748–755 (2004)
8. Fullam, K., Sabater, J., Barber, K.S.: Toward a testbed for trust and reputation models. Trusting Agents for Trusting Electronic Societies, 95–109 (2005)
9. Zadeh, L.: Fuzzy sets. In: Zadeh, L.A. (ed.) Fuzzy sets, fuzzy logic, and fuzzy systems, pp. 19–34. World Scientific Publishing Co., Inc., River Edge (1996)
10. Rozich, R., Ioerger, T., Yager, R.: FURL - a theory revision approach to learning fuzzy rules. In: Proceedings of the IEEE International Conference on Fuzzy Systems, pp. 791–796 (2002)
11. Yager, R.R.: On the hierarchical structure for fuzzy modeling and control. IEEE Transactions on Fuzzy Systems 23, 1189–1197 (1993)
12. Amgoud, L., Prade, H.: Explaining qualitative decision under uncertainty by argumentation. In: Proceedings of the AAAI. AAAI Press, Menlo Park (2006)
13. Harris, C., Moore, C., Brown, M.: Intelligent Control: Aspects of Fuzzy Logic and Neural Networks. Robotics and Automated Systems. World Scientific Press, Singapore (1993)
14. Huynh, D., Jennings, N.R., Nigel, R., Shadbolt.: Developing an integrated trust and reputation model for open multi-agent systems. In: Proceedings of 7th International Workshop on Trust in Agent Societies, pp. 62–77 (2004)
15. Yu, B., Singh, M.P.: An evidential model of distributed reputation management. In: Proceedings of the first international joint conference on Autonomous agents and multiagent systems, pp. 294–301. ACM Press, New York (2002)
16. Sen, S., Dutta, P.S.: The evolution and stability of cooperative traits. In: Proceedings of the first international joint conference on Autonomous agents and multiagent systems, pp. 1114–1120. ACM Press, New York (2002)
17. Mui, L., Mohtashemi, M., Halberstadt, A.: Notions of reputation in multi-agents systems: A review. In: First International Conference on Autonomous Agents and MAS, Bologna, Italy, pp. 280–287 (July 2002)
18. Falcone, R., Pezzulo, G., Castelfranchi, C.: A fuzzy approach to a belief-based trust computation. In: Falcone, R., Singhr, M., Tan, Y.H. (eds.) Trust, reputation, and security: theories and practice, pp. 73–86. Springer, Heidelberg (2003)
19. Stranders, R.: Argumentation based decision making for trust in multi-agent systems. Master's thesis, Delft University of Technology (2006)
20. Liau, C.J.: Belief, information acquisition, and trust in multi-agent systems: a modal logic formulation. Artificial Intelligence 149(1), 31–60 (2003)