Fuzzy Argumentation for Trust

Ruben Stranders, Mathijs de Weerdt, and Cees Witteveen

Delft University of Technology
R.Stranders@gmail.com, (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, and therefore such agents can be unreliable or deceitful. Consequently, an agent representing a human owner needs to reason about trusting (information or services provided by) other agents. Existing algorithms to 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.

In this paper, we propose a new approach to trust in Multi-Agent Systems based on argumentation that aims to expose the rationale behind such trusting decisions. Our solution features a clear 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 translate the fuzzy rules within these models into 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 they please, and the lack of central regulation and control of behavior. In such a MAS, often agents are 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 also their goals may easily 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 their owner’s demands.

Several algorithms have already been devised to confront the problem of estimating trustworthiness by capturing past experiences with other agents in one or two values that can be used to estimate future behavior [7]. These algorithms, however, primarily focus on improving the immediate success of an agent. Less emphasis has been laid on discovering patterns in the behavior of other agents, or—more challenging—their motives and incentives (or goals). Moreover, the rationale of the 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.
As an example, consider the following situation. Suppose a user instructs a personal agent to buy a painting for his collection. When an interesting painting is offered, this agent starts by estimating the value of this painting before submitting any bids. The agent retrieves this information by searching some databases and asking a number of experts. To obtain a good estimate, it then assigns weights to the various received appraisals. A more reliable and trustworthy source gets a higher weight. 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 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 so much more useful?

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 needs to have almost 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, in our research, we explore the requirements for a new approach to trust in Multi-Agent Systems 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.

This goal gives rise to the following conditions for such an approach: (i) A personal agent should be able to explain why certain decisions were made, and why alternatives were discarded, (ii) it should formulate these explanations in terms of the perceived behavior of other agents, and (iii) it should present a logical (symbolic) reasoning supporting its decisions by this observed behavior.

Each (personal) agent has therefore a knowledge base, capturing the behavior of other agents in rules, a set of actions or decisions it can make, and some goals to attain, which are given by the human owner. Due to the uncertainty, ambiguity, and incompleteness of information regarding trust in Multi-Agent Systems, this setting gives rise to some specific requirements of the opponent model an agent should be able to build:

1. The model should be able to represent inherently uncertain, incomplete and ambiguous knowledge about other agents, and
2. it should support an argumentation framework capable of making decisions and explaining them. This implies that it should be composed of logical rules.

Moreover, we have a strong preference for a model that is commonly used, to ensure the existence of sufficiently tested and accepted induction algorithms. 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 framework for argumentation 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 the results of applying this model 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 in this paper is to represent the uncertain knowledge about other agents using logical rules, and use this knowledge to derive not only good decisions, but also an argument to support those decisions. In this section 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 Architecture

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. These past experiences are stored in a transaction database. Data from the transaction database is used to identify behavioral patterns. This is done by applying a data mining algorithm. Together, the behavioral patterns form an opponent model—a description of how an opponent reacts in different situations.

The decision making component is responsible for making the actual decisions. It uses the opponent models to predict the outcomes of each available action. Using these outcomes, and the knowledge acquired by the opponent model, arguments are constructed to support (or reject) the action. These arguments explicitly refer to the outcomes in terms of the agent’s goals. The more the predicted outcomes are favorable in terms of these goals, the greater the strength of the argument supporting the action. Based on this, the generated arguments can be paraphrased as: “when I take decision d to execute that action, the model that I have of the behavior of the other agent predicts a certain outcome, which conflicts with/attains some positive goals. Decision d is therefore desirable/undesirable.” Figure 1 shows the relation between the opponent modeling and the decision making component.

The final step in decision making is determining the most appropriate action and executing it. The action that is supported by the argument with the highest strength is the one that is the most prudent. When this action has been executed, the actual outcomes are observed and recorded in the transaction database. These new results are subsequently used to refine the model of the agent once again, completing the circle.
The symbolic method of reasoning needed in our approach operates on a different level than simple numerical data that is observed from the environment. Moreover, we need to reason about the inherent vagueness and ambiguousness of information in a trust domain. Fuzzy (possibilistic) logic [9] provides us with a way to tackle this modeling problem, because it provides a natural way of translating back and forth between logic rules on the one hand, and uncertain data on the other hand.

Several different algorithms exist to infer fuzzy rules from numerical data. Together, these rules form a fuzzy rule base that approximates the data. Many learning algorithms also assign a measure of confidence to each rule in the rule base. Usually, this measure is (inversely) proportional to the error the rule makes with respect to the past transactions.

### 2.2 Argumentation

To carefully weigh the pros and cons of each decision under consideration, and to select the decision that is most likely to have acceptable consequences, we used a framework for argumentation. We consider the work by Amgoud and Prade [1,10] to be a good point of departure for such an argumentation framework. It supports reasoning under uncertainty with fuzzy logic. This framework uses the agent’s knowledge base $K$, a set of its goals $G$, and a set of possible decisions (or actions) $D$. An argument in favor of a decision is then defined as follows:
**Definition 1.** An argument \( A \) in favor of a decision \( d \) is a triple \( A = (S, C, d) \), where:

- \( S \) is the support of the argument. The support of the argument contains the knowledge from the agent’s knowledge base \( K \) used to predict the consequences of decision \( d \).
- \( C \) are the consequences of the argument. These consequences are goals reached by decision \( d \), and form a subset of the goal base \( \mathcal{G} \).
- \( d \) is the conclusion of the argument, and is a member of the set of all available decisions \( D \). Decision \( d \) is recommended by argument \( A \).

Moreover, \( S \cup \{d\} \) should be consistent, \( S \cup \{d\} \vdash C \), \( S \) should be minimal, and \( C \) maximal among the sets satisfying the above conditions.

The set \( A \) gathers all the arguments which can be constructed from \( (K, \mathcal{G}, D) \). The construction of these arguments is very straightforward: for each decision, the consequences are predicted using the knowledge base \( K \). Next, the consequences are evaluated in terms of the agent’s goals \( \mathcal{G} \). Finally the arguments are ordered by their strength, and the decision supported by the strongest argument is selected.

This leaves open only the concept of an argument’s strength. As in the original framework we make a distinction between the **Level** and the **Weight** of an argument. The former refers to the confidence in support \( S \) of the argument, the latter to the importance of the goals in \( C \). In the original framework the knowledge base \( K \) consisted of elements \( (k_i, \rho_i) \) where \( k_i \) is a propositional formula, and \( \rho_i \) can be thought of as the confidence the agent has in this rule or fact. In our framework \( k_i \) is a fuzzy rule. Consequently, given an environment state \( \omega \), the valuation \( v_\omega \) of a fuzzy rule or fact \( k_i \) is not just 0 or 1 as in the original framework, but \( 0 \leq v_\omega(k) \leq 1 \). This means that rules can be partially applicable to the current state of the environment. We call this the **match strength** of a rule.

We generalize the original framework to deal with this partial applicability of knowledge. The **Level** of an argument \( A \) depends on the strength of the weakest rule \( k_j \) used in the argument:

\[
Level(A) = \rho_j \cdot v_\omega(k_j)
\]  

where \( j \) (the index of the weakest rule) is obtained using the following equation:

\[
j = \arg \min_i \frac{\rho_i}{v_\omega(k_i)} \text{ for } \{(k_i, \rho_i) \mid (k_i, \rho_i) \in S, v_\omega(k_i) \neq 0\}
\]  

This redefinition ensures that:

1. For equal confidence levels \( \rho_i \), the knowledge with the highest match strength determines the Level of the argument. The higher the match strength, the more the knowledge is applicable, and the more reliable it is in this particular case.
2. For equal match strengths, the knowledge with the lowest level of confidence determines the Level of the argument. This is consistent with the argumentation framework presented in [1].

3. In boundary cases where a rule is fully matched, or not matched at all (e.g. \( v_\omega(k) = \{ 0, 1 \} \)), Equation 2 reduces to the definition of Level in the original framework.

The Weight of an argument \( A \) depends on the goals that can be reached. The goals are given as tuples \( (g_j, \lambda_j) \) in the set \( G \). 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. The original framework did not factor in the possibility of partially satisfied goals. To deal with this, we redefine Weight as follows:

\[
\text{Weight}(A) = \sum_{(g_j, \lambda_j) \in G} v_\omega(g_j) \cdot \lambda_j
\]

This definition ensures that the weight of the argument increases with the utility of the expected consequences of the decision. More specifically, if a goal \( g \) with preference \( \lambda \) 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.

Finally, we need to compare the Weight and Level of each argument to determine which argument is the most powerful. Put differently, a preference relation among arguments is required:

**Definition 2.** Let \( A \) and \( B \) be two arguments in \( A \). \( A \) is preferred to \( B \) iff 
\[
\text{Level}(A) \cdot \text{Weight}(A) \leq \text{Level}(B) \cdot \text{Weight}(B).
\]

### 2.3 Opponent Modeling

For our proof of concept, we need a fuzzy (possibilistic) rule learning algorithm to build a rule base. For this, we use a simple theory revision algorithm called Fuzzy Rule Learner (FURL) [11]. Taking observations from the environment as input, FURL is capable of creating a rule base of fuzzy rules. Rules can be more or less plausible, depending on the prediction error they cause on past observations. In fact, FURL uses a Hierarchical Prioritized Structure [12] consisting of layers of rules where each layer consists of rules that are exceptions to rules in the layer below it. However, for our application we can think of the result just as a (flat) rule base \( K \) with fuzzy rules.

Each rule in our case is an implication from an observation (condition) to an expected/learned effect (conclusion). For example, a fuzzy rule like “if certainty is \( c_1 \) (low) then appraisal-error is \( ae_5 \) (high)” should be interpreted as follows: if the value for certainty is a member of the fuzzy set \( c_1 \), which represents all low values for certainty, then we can expect that the value for the appraisal error (of this opponent) will be a member of the fuzzy set \( ae_5 \), which represents high values for appraisal error. Membership of a fuzzy set is not just true or false, but can be a value between 0 and 1. Consequently, a rule like this can be partially applicable, as discussed in the previous section.
The amount of confidence in a fuzzy rule in the knowledge base is related to the certainty with which this rule is believed to be true. In our framework, this measure of confidence is obtained by calculating the inverse of the error measure produced by FURL.

3 Evaluation

As we remarked in the introduction we are interested in the way trust in MAS can benefit from argumentation for decisions. To have a preliminary impression about the contribution of our approach to that end, we would like to investigate (i) how an agent based on our approach behaves in a simple art appraisal environment where other agents with fixed decision tactics operate, and whether it is capable of explaining its decisions, and (ii) how this agent performs with respect to these other agents in a competitive setting.

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 a number of other agents to estimate the value of a painting. Agents can ask each other for their opinion, and may expect an answer and a claimed certainty of this answer. The agents need to combine the opinions of others to arrive at a final appraisal of the painting. Each agent has its own area of expertise for which it can give good opinions to others. All agents compete with each other a number of rounds (appraising different paintings) in making the best estimate, so it may be worthwhile to try to feed the other agents the wrong information at some point(s). Knowing when and whom to trust is essential to be successful in this domain.

In the 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 always honestly tells how certain it is that it can accurately appraise a painting. If it has low expertise, it gives a low certainty, and vice versa. The behavior of Reciprocal is somewhat more complicated: the behavior of its opponent influences its behavior towards that opponent. If the opponent has been dishonest by misrepresenting its expertise, Reciprocal responds in kind: it becomes dishonest as well. However, if Reciprocal’s opponent is honest, Reciprocal behaves exactly the same as Honest.

In each of the following scenarios, our agent has interacted with both agents in 200 transactions. From the observations made during 200 transactions, we used FURL to build an opponent model. The confidence in the truth of each rule is calculated using the error measures FURL associates with each of the rules in the knowledge base. The models our agent has learned after 200 transactions are presented in Tables 1 and 2. These models contain multiple fuzzy if-then rules describing the opponent’s behavior.

Using the opponent model, the agent needs to make a decision about trusting its opponents in the next transaction. More specifically, it needs to determine the weights given to each of the opponent’s appraisals. Of course, it is preferred
to assign more weight to the opponent that is the more skilled in appraising the painting.

3.1 Scenario 1: Requester Role

In this scenario our agent consults Honest and Reciprocal to appraise its own painting. For each agent, it searches for arguments to support the decision to get an opinion from both Honest and Reciprocal. The strengths of these arguments are used to determine the delegation weight, i.e. the extent to which other agents’ appraisals are used.

**Goals** Because it is in our agent’s interest to appraise the painting as accurately as possible, it has a single goal $g_1 = \text{(appraisal-error is acceptable, 1)}$. In this case, acceptable is a fuzzy set that assesses the acceptability of the expected appraisal error from an opponent. Its membership function is defined as:

### Table 1. Model of Honest’s behavior after 200 interactions

<table>
<thead>
<tr>
<th>Rule</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 if certainty is $e_0$ then appraisal-error is $a_0$</td>
<td>0.00082</td>
</tr>
<tr>
<td>2 if certainty is $e_1$ then appraisal-error is $a_1$</td>
<td>0.00084</td>
</tr>
<tr>
<td>3 if certainty is $e_2$ then appraisal-error is $a_2$</td>
<td>0.00086</td>
</tr>
<tr>
<td>4 if certainty is $e_3$ then appraisal-error is $a_4$</td>
<td>0.00088</td>
</tr>
<tr>
<td>5 if certainty is $e_5$ then appraisal-error is $a_5$</td>
<td>0.00090</td>
</tr>
<tr>
<td>6 if certainty is $e_6$ then appraisal-error is $a_6$</td>
<td>0.00092</td>
</tr>
</tbody>
</table>

### Table 2. Model of Reciprocal’s behavior after 200 interactions

<table>
<thead>
<tr>
<th>Rule</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 if certainty is $e_0$ then appraisal-error is $a_0$</td>
<td>0.00084</td>
</tr>
<tr>
<td>2 if certainty is $e_1$ then appraisal-error is $a_2$</td>
<td>0.00082</td>
</tr>
<tr>
<td>3 if certainty is $e_2$ then appraisal-error is $a_4$</td>
<td>0.00086</td>
</tr>
<tr>
<td>4 if certainty is $e_3$ then appraisal-error is $a_6$</td>
<td>0.00088</td>
</tr>
<tr>
<td>5 if certainty is $e_5$ then appraisal-error is $a_8$</td>
<td>0.00090</td>
</tr>
<tr>
<td>6 if certainty is $e_6$ then appraisal-error is $a_10$</td>
<td>0.00092</td>
</tr>
</tbody>
</table>
\[
acceptable(x) = 1 - x
\]  

(4)

Put differently, goal \( g_1 \) states that our agent favors accurate appraisals from its opponents. So, in this particular transaction, our agent tries to find out from whom it can get the most accurate appraisal.

Observations Before deciding which opponent to trust, each opponent tells how certain it is of its own expertise. HONEST asserts a certainty of \( c_1 \), while RECIPROCAL replies that it can appraise the painting with a certainty between \( c_4 \) and \( c_5 \). Also, in the previous round, our agent has been somewhat dishonest towards RECIPROCAL (the dishonesty was a member of the fuzzy set \( d_3 \)).

Available Decisions As said before, in this transaction, our agent can request an appraisal from two opponents. Consequently, it must consider two possible decisions: \( d_{\text{Honest}} \), i.e. accept the appraisal from HONEST, or \( d_{\text{Reciprocal}} \), i.e. accept the appraisal from RECIPROCAL. Of course, 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.

On the one hand, we expect a poor appraisal from HONEST, because its certainty is quite low. On the other hand, RECIPROCAL’s certainty is very high, but our agent has to take its own dishonesty towards RECIPROCAL into account. The opponent model has to decide what the effect of this will be on RECIPROCAL’s appraisal. Using these goals, observations, and decisions, our agent generates two arguments. The first argument \( A_{\text{Honest}} \) supports decision \( d_{\text{Honest}} \), the second argument \( A_{\text{Reciprocal}} \) supports decision \( d_{\text{Reciprocal}} \).

Decision for HONEST Remember from Definition 1 that an argument consists of three parts: support, consequences and conclusion. The support of the argument is a subset of the knowledge base of the agent, and consists of knowledge used to predict the consequences of the decision under consideration. The support of \( A_{\text{Honest}} \) consists of parts of the opponent model of HONEST relevant to this particular transaction. This is summarized in Table 3.

The consequences of \( A_{\text{Honest}} \) relate to the desirability of the consequences of decision \( d_{\text{Honest}} \) 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 \( ae_1 \). Given this prediction, we can determine the utility in terms of goal \( g_1 \) (see Table 4(a)). When we defuzzify \( ae_1 \)\(^1\), we obtain a numerical value of 0.75. Using the membership function of acceptable from Equation 4, 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 and Weight of argument \( A_{\text{Honest}} \) (see Equations 1 on page 5, and 3

\(^1\) 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 3. The support for Argument $A_{Honest}$

<table>
<thead>
<tr>
<th>Knowledge</th>
<th>Match</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>certainty is $c_1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>if certainty is $c_1$ then appraisal-error is $ae_4$</td>
<td>100%</td>
<td></td>
</tr>
<tr>
<td>appraisal-error is $ae_4$</td>
<td>100%</td>
<td>0.00832</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Goal</th>
<th>Match</th>
<th>Preference</th>
<th>Property</th>
<th>Calculation</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g_1$</td>
<td>0.25</td>
<td>1</td>
<td>$Level(A_{Honest})$</td>
<td>$1 \times 0.00832$</td>
<td>0.00832</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$Weight(A_{Honest})$</td>
<td>$1 \times 0.25$</td>
<td>0.25</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(b) The Level and Weight calculations of Argument $A_{Honest}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

on page 6). Table 4(b) lists the steps for this calculation. Our agent can now determine the strength of the argument for $Honest$: $0.00832 \times 0.25 = 0.00208$ (see Definition 2).

Decision for Reciprocal. Next, our agent performs the same steps for Reciprocal. For determining the support and consequences of argument $A_{Reciprocal}$, we follow the same procedure as above. They are summarized in Tables 5 and 6(a), respectively. This time, four rules fire based on the information Reciprocal provided. We can see that the appraisal error is expected to be somewhere between $ae_6$ and $ae_3$. After defuzzifying the output of the opponent model and using the membership function of the acceptable set, we find that goal $g_1$ is satisfied for 75%. 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: $0.00438 \times 0.75 = 0.00329$.

Concluding. In the final step, our agent compares the strengths of both arguments. This is done in Table 7. When normalized, the strengths of the arguments provide the appraisal weights towards both agents. As we can see, Reciprocal determines 61% of the appraisal. Apparently, our agents favors a low appraisal

Table 5. The support for Argument $A_{Reciprocal}$

<table>
<thead>
<tr>
<th>Knowledge</th>
<th>Match</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>certainty is $c_4$</td>
<td>50%</td>
<td></td>
</tr>
<tr>
<td>certainty is $c_5$</td>
<td>50%</td>
<td></td>
</tr>
<tr>
<td>dishonesty is $d_3$</td>
<td>40%</td>
<td></td>
</tr>
<tr>
<td>if certainty is $c_4$ then appraisal-error is $ae_3$</td>
<td>50%</td>
<td>0.00876</td>
</tr>
<tr>
<td>if certainty is $c_5$ then appraisal-error is $ae_2$</td>
<td>50%</td>
<td>0.01042</td>
</tr>
<tr>
<td>if certainty is $c_4$ and dishonesty is $d_3$ then appraisal-error is $ae_0$</td>
<td>40%</td>
<td>0.01640</td>
</tr>
<tr>
<td>if certainty is $c_4$ and dishonesty is $d_3$ then appraisal-error is $ae_1$</td>
<td>40%</td>
<td>0.01640</td>
</tr>
</tbody>
</table>
Table 6. The consequences, and the Level and Weight calculations for argument \( A_{Reciprocal} \)

<table>
<thead>
<tr>
<th>Goal</th>
<th>Match</th>
<th>Preference</th>
<th>Property</th>
<th>Calculation</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>( g_1 )</td>
<td>0.75</td>
<td>1</td>
<td>Level( (A_{Reciprocal}) )</td>
<td>0.5 ( \times ) 0.00876</td>
<td>0.00438</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weight( (A_{Reciprocal}) )</td>
<td>1 ( \times ) 0.75</td>
<td>0.75</td>
</tr>
</tbody>
</table>

(a) The consequences of Argument \( A_{Reciprocal} \)

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

Table 7. The delegation weights for Honest and Reciprocal in scenario 1.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Level</th>
<th>Weight</th>
<th>Strength</th>
<th>Delegation weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honest</td>
<td>0.00832</td>
<td>0.25</td>
<td>0.00208</td>
<td>0.39</td>
</tr>
<tr>
<td>Reciprocal</td>
<td>0.00438</td>
<td>0.75</td>
<td>0.00329</td>
<td>0.61</td>
</tr>
</tbody>
</table>

error, and more or less takes the reduced confidence of the knowledge of Reciprocal’s behavior for granted.

In this scenario, we have seen that our agent had 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 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. Consequently, our agent chose to depend most on Reciprocal for appraising its painting.

3.2 Scenario 2: Provider Role

In the previous scenario, we focused on the appraisals received from our agent’s opponents. Now, we reverse the roles: our agent provides its opponents with advice. To this end, we add a new goal, and apply the decision making procedure to the appraisals generated by our agent, instead of its opponents. The new goal, called \( g_2 \), essentially encourages our agent to be as deceptive as possible towards other agents (by overstating its certainty of correctly appraising a painting). This will, however, influence the quality of returned appraisals by Reciprocal. So, we must find a balance between achieving goal \( g_1 \) and goal \( g_2 \). In other words, deceiving other agents must not negatively influence the accuracy of appraisals received from those agents too much.

Deciding the extent of the deception towards an agent is different from deciding delegation weights in scenario 1. For one, the value of the decision variable is now not only a result from the decision making procedure, but also influences a part of the opponent model. In scenario 1, the decision variable was the delegation weight towards each agent. Now, the decision variable is dishonesty, which is part of the opponent model. Second, the decision pertains to transactions in the near future, instead of the current transaction. Our agent needs to predict the effect of its deception on future transactions.
This introduces a problem because information about a transaction in the future is not yet available. In particular, the certainty asserted by an opponent in a future transaction is important for predicting the appraisal error, but is not known beforehand. Using the opponent model without a value for certainty would cause none of the rules in the rule base to fire. In this case, the opponent model does not produce a prediction for the appraisal error, rendering it essentially useless.

Our solution to this problem is to generate a set of arguments for each decision for a number of hypothetical values of certainty. This way, we effectively removed the certainty variable from the opponent model, leaving the relation between dishonesty and appraisal error. Next, the Level and Weight of each of these arguments is averaged and compared 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. 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.

Goals In addition to goal $g_1$ from scenario 1, goal $g_2 = (\text{dishonesty is deceptive, 0.5})$ is included in the goal base of our agent. Deceptive is a fuzzy set in the domain of dishonesty. The higher dishonesty, the more our agent misrepresents its expertise by overstating its certainty. Note that goal $g_1$ has a lower priority than goal $g_2$.

Observations There are no relevant observations in this particular decision making process, because it pertains to transactions in the future.

Available Decisions 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 expected, because it punishes dishonesty by increasing its own. Consequently, when increasing dishonesty while keeping the certainty equal, 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.

We also determined the influence of the importance of goal $g_2$ on the preferred decision. Figure 2 shows the weights of the arguments supporting decisions $d_A$

---

2 More specifically, we generated an argument for 100 equally spaced values of ‘certainty’ between 0 and 1.

3 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.

<table>
<thead>
<tr>
<th>Dishonesty</th>
<th>Appraisal Error</th>
<th>Goal Satisfaction</th>
<th>Level</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>0.00</td>
<td>0.63</td>
<td>0.37</td>
<td>0.49</td>
</tr>
<tr>
<td>$d_2$</td>
<td>0.25</td>
<td>0.75</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0.50</td>
<td>0.85</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0.75</td>
<td>0.87</td>
<td>0.13</td>
<td>0.75</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1.00</td>
<td>0.85</td>
<td>0.15</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Fig. 2. The priority of goal $g_2$ determines whether Reciprocal is treated dishonestly by our agent. The lower the priority of this goal, the less weight is assigned to arguments supporting high levels of dishonesty. Table 8 shows detailed results when the priority of $g_2$ is 0.5.

to $d_E$ for different priorities of goal $g_2$. Of course, as this priority decreases, goal $g_1$ becomes relatively more important. When the priority of $g_2$ drops below 0.2, the scale suddenly tips in favor of decision $d_4$. Apparently, then our agent favors accurate appraisals from Reciprocal instead of deceiving it.

3.3 Competition in the ART Testbed

In the previous two scenario’s, our agent made decisions in the scope of a single transaction. Of course we are also interested in the behavior of our agent during multiple transactions and show that our approach does not only satisfy our primary requirement (i.e. being capable of explaining trusting decisions), but that it is also capable of competing against other agents.

An important performance criterion in ART is the market-share of agents participating in the simulation. In ART, agents do not appraise paintings for themselves, but on behalf of their clients. If an agent appraises its paintings more accurately than its competitors, it will receive more clientele in the future, thereby increasing its market-share.
Fig. 3. Market-shares during ART Testbed simulation with three agents

To calculate these market-shares, we made an assumption about how Honest and Reciprocal calculate delegation weights towards their opponents. We decided that both Honest and Reciprocal use the asserted certainty as an indicator for expected result. This means that they do not expect their opponents to lie. The certainties received from their opponents are therefore used to weigh their influence on the final appraisal.

Figure 3 shows the results of this simulation. Our agent performs best, followed by Reciprocal and Honest. Reciprocal beats Honest, because Honest is deceived by our agent, whereas Reciprocal persuades our agent to cooperate. In this particular simulation our agent beats both agents, because it does not blindly trust the asserted certainties from its opponents. Instead, has built an opponent model that predicts the actual appraisal error based on a number of variables. For example, it can predict the appraisal error of Reciprocal based on the deception towards it in the previous round. That way, it is more capable of deciding whom to trust, giving it a strategic edge over its competitors.

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 [13] the \textit{quality} and the \textit{reliability} of past transaction-results are derived and used for future decision making. An application of the Dempster-Shafer theory collects evidence of trustworthiness [14], and another approach using probabilistic reciprocity captures utility provided to and received from an agent [15], or the probability that task delegation towards an agent is successful [16]. 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. [6,17], 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 \textit{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 set 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 [18].

In contrast to the decision model of Castelfranchi et al., the modified doxastic logic for Belief, Information acquisition, and Trust (BIT) [19] 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 \textit{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.
References