A Probabilistic Trust Model for Handling Inaccurate Reputation Sources

Jigar Patel, W.T. Luke Teacy, Nicholas R. Jennings, and Michael Luck

Electronics & Computer Science, University of Southampton, Southampton SO17, 1BJ, UK
{jp03r, wtlt03r, nrj, mml}@ecs.soton.ac.uk

Abstract. This research aims to develop a model of trust and reputation that will ensure good interactions amongst software agents in large scale open systems in particular. The following are key drivers for our model: (1) agents may be self-interested and may provide false accounts of experiences with other agents if it is beneficial for them to do so; (2) agents will need to interact with other agents with which they have no past experience. Against this background, we have developed *TRAVOS* (Trust and Reputation model for Agent-based Virtual OrganisationS) which models an agent's trust in an interaction partner. Specifically, trust is calculated using probability theory taking account of past interactions between agents. When there is a lack of personal experience between agents, the model draws upon reputation information gathered from third parties. In this latter case, we pay particular attention to handling the possibility that reputation information may be inaccurate.

1 Introduction

Computational systems of all kinds are moving toward large-scale, open, dynamic and distributed architectures, which harbour numerous *self-interested* agents. The Grid is perhaps the most prominent example of such an environment and is the context of this paper. However, in all these environments the concept of self-interest, which is endemic in such systems, introduces the possibility of agents interacting in a way to maximise their own gain (perhaps at the cost of another). Therefore, in such contexts it is essential to ensure good interaction between agents so that no single agent can take advantage of the others in the system. In this sense, good interactions can be defined as those in which the expectations of the interacting agents are fulfilled; for example, if the expectation of one of the agents is recorded as a contract which is then fulfilled by its interaction partner, it is a good interaction.

We view the Grid as a multi-agent system (MAS) in which autonomous software agents, owned by various organisations, interact with each other. In particular, many of the interactions between agents are conducted in terms of Virtual Organisations (VOs), which are collections of agents (representing individuals or organisations), each of which has a range of problem-solving capabilities and resources at its disposal. A VO is formed when there is a need to solve a problem

P. Herrmann et al. (Eds.): iTrust 2005, LNCS 3477, pp. 193–209, 2005.

or provide a resource that a single agent cannot address. Here, the problem of assuring good interactions between individual agents is further complicated by the size of the Grid, and the large number of agents and interactions between them. Nevertheless, solutions to these problems are integral to the wide-scale acceptance of the Grid and agent-based VOs [4].

It is now well established that computational *trust* is important in such open systems [10]. Specifically, trust provides a form of social control in environments in which agents are likely to interact with others whose intentions are not known. It allows agents within such systems to reason about the reliability of others. More specifically, trust can be utilised to account for uncertainty about the willingness and capability of other agents to perform actions as agreed, rather than defecting when it proves to be more profitable. For the purpose of this work, we use an adaptation of the definition offered by Gambetta [5], and define trust to be *a particular level of subjective probability with which an agent assesses that another agent will perform a particular action, both before the assessing agent can monitor such an action and in a context in which it affects the assessing agent's own action.*

Trust is often built over time by accumulating personal experience with others, and using this experience to judge how they will perform in an as yet unobserved situation. However, when assessing our trust in someone with whom we have no direct personal experience, we often ask others about their experiences with this individual. This collective opinion of others regarding an individual is known as the individual's *reputation* and it is the reputation of a trustee that we use to assess its trustworthiness, if we have no personal experience.

Given the importance of trust and reputation in open systems and their use as a form of social control, several computational models of trust and reputation have been developed, each with requirements for the domain to which they apply (see [10] for a review of such models). In our case, the requirements can be summarised as follows. First, the model must provide a trust metric that represents a level of trust in an agent. Such a metric allows comparisons between agents so that one agent can be inferred as more trustworthy than another. The model must be able to provide a trust metric given the presence or absence of personal experience. Second, the model must reflect an individual's *confidence* in its level of trust for another agent. This is necessary so that an agent can determine the degree of influence the trust metric has on its decision about whether or not to interact with another individual. Generally speaking, higher confidence means a greater impact on the decision-making process, and lower confidence means less impact. Third, an agent must not assume that opinions of others are accurate or based on actual experience. Thus, the model must be able to discount the opinions of others in the calculation of reputation, based on past reliability and consistency of the opinion providers. However, existing models do not allow an agent to efficiently assess the reliability of reputation sources and use this assessment to discount the opinion provided by that source (see Section 5 for a detailed discussion). To meet the above requirements, therefore, we have developed TRAVOS, a trust and reputation model for agent-based VOs.

The remainder of this paper is organised as follows. Section 2 presents the basic TRAVOS model. Following from this, Section 3 provides a description of how the basic model has been expanded to include the functionality of handling inaccurate opinions from reputation sources. A scenario using these mechanisms is then presented in Section 4. Section 5 presents related work. Finally, Section 6 concludes the paper and provides an outline for future work.

2 The TRAVOS Model

TRAVOS equips an agent (the truster) with two methods for assessing the trustworthiness of another agent (the trustee). First, the truster can make the assessment based on the direct interactions it has had with the trustee. Second, the truster may assess the trustworthiness of another based on the reputation of the trustee.

2.1 Basic Notation

In a MAS consisting of n agents, we denote the set of all agents as $\mathcal{A} = \{a_1, a_2, ..., a_n\}$. Over time, distinct pairs of agents $\{a_x, a_y\} \subseteq \mathcal{A}$ may interact with one another, governed by contracts that specify the obligations of each agent towards its interaction partner. An interaction between a_1 and a_2 is considered successful by a_1 if a_2 fulfils its obligations. From the perspective of a_1, the outcome of an interaction between a_1 and a_2 is summarised by a binary variable[1], O_{a_1,a_2}, where $O_{a_1,a_2} = 1$ indicates a successful (and $O_{a_1,a_2} = 0$ indicates an unsuccessful) interaction[2] for a_1 (Equation 1). Furthermore, we denote an outcome observed at time t as $O^t_{a_1,a_2}$, and the set of all outcomes observed from time t_0 to time t as $O^{n:t}_{a_1,a_2}$.

$$O_{a_1,a_2} = \begin{cases} 1 \text{ if contract fulfilled by } a_2 \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

At any point of time t, the history of interactions between agents a_1 and a_2 is recorded as a tuple, $\mathcal{R}^t_{a_1,a_2} = (m^t_{a_1,a_2}, n^t_{a_1,a_2})$ where the value of $m^t_{a_1,a_2}$ is the number of successful interactions of a_1 with a_2, while $n^t_{a_1,a_2}$ is the number of unsuccessful interactions of a_1 with a_2. The tendency of an agent a_2 to fulfil or default on its obligations to an agent a_1, is governed by its behaviour. We model the behaviour of a_2 towards a_1, denoted B_{a_1,a_2}, as the *intrinsic* probability with which $O_{a_1,a_2} = 1$. In other words, B_{a_1,a_2} is the *expected value* of O_{a_1,a_2} given

[1] Representing a contract outcome with a binary variable is a simplification made for the purpose of our model. We concede that in certain circumstances, a more expressive representation may be appropriate.

[2] The outcome of an interaction from the perspective of one agent is not necessarily the same as from the perspective of its interaction partner. Thus, it is possible that $O_{a_1,a_2} \neq O_{a_2,a_1}$.

complete information about a_2's decision processes and all environmental factors that effect its capabilities (Equation 2). For simplicity, we admit the subscripts for B when the identity of the interacting agents is irrelevant to the discussion.

$$B_{a_1,a_2} = E[O_{a_1,a_2}], \quad \text{where } B_{a_1,a_2} \in [0,1] \tag{2}$$

In TRAVOS, each agent maintains a *level of trust* in each of the other agents in the system. Specifically, the level of trust of an agent a_1 in an agent a_2, denoted as τ_{a_1,a_2}, represents a_1's assessment of the likelihood of a_2 fulfilling its obligations. The *confidence* of a_1 in its assessment of a_2 is denoted as γ_{a_1,a_2}. Confidence is a metric that represents the accuracy of the trust value calculated by an agent given the number of observations (the evidence) it uses in the trust value calculation. Intuitively more evidence would result in more confidence. The precise definitions and reasons behind these values are discussed in Sections 2.2 and 2.3 respectively.

2.2 Modelling Trust

The first basic requirement of a computational trust model is that it should provide a metric for comparing the relative trustworthiness of different agents. From our definition of trust, we consider an agent to be trustworthy if it has a high probability of performing a particular action which, in our context, is to fulfil its obligations during an interaction. This probability is unavoidably subjective, because it can only be assessed from the individual viewpoint of the truster, based on the truster's personal experiences.

In light of this, we have adopted a probabilistic approach to modelling trust, based on the individual experiences of any agent in the role of a truster. If a truster, agent a_1, has complete information about a trustee, agent a_2, then, according to a_1, the probability that a_2 fulfils its obligations is expressed by B_{a_1,a_2}. In general, however, complete information cannot be assumed; the best we can do is to use the expected value of B_{a_1,a_2} given the experience of a_1, which we consider to be the set of all interaction outcomes it has observed. Thus, we define the level of trust τ_{a_1,a_2} at time t, as the expected value of B_{a_1,a_2} given the set of outcomes $O^{1:t}_{a_1,a_2}$ (3).

$$\tau_{a_1,a_2} = E[B_{a_1,a_2}|O^{1:t}_{a_1,a_2}] \tag{3}$$

The expected value of a continous random variable is dependent on the *probability density function* (pdf) used to model the probability that the variable will have a certain value. Thus, we must choose such a function that is suitable to our domain. In Bayesian analysis, the beta family of pdfs is commonly used as a prior distribution for random variables that take on continuous values in the interval $[0,1]$. For example beta pdfs can be used to model the distribution of a random variable representing the unknown probability of a binary event [2]— B is an example of such a variable. For this reason, we use beta pdfs in our model. (Beta pdfs have also been previously applied to trust for similar reasons by [7]).

The general formula for beta distributions is given in Equation 4. It has two parameters, α and β, which define the shape of the density function when plotted. Examples, plotted for B with various parameter settings are shown in Figure 1; here, the horizontal axis represents the possible values of B, while the vertical axis gives the *relative* probability that each of these values is the true value for B. The most likely (expected value) of B is the curve maximum. The width of the curve represents the amount of uncertainty over the true value of B. If α and β both have values close to 1, a wide density plot results, thus representing a high level of uncertainty about B. In the extreme case of $\alpha = \beta = 1$, the distribution is uniform, with all values of B considered equally likely.

$$f(b|\alpha,\beta) = \frac{b^{\alpha-1}(1-b)^{\beta-1}}{\int U^{\alpha-1}(1-U)^{\beta-1}dU}, \quad \text{where } \alpha,\beta > 0 \tag{4}$$

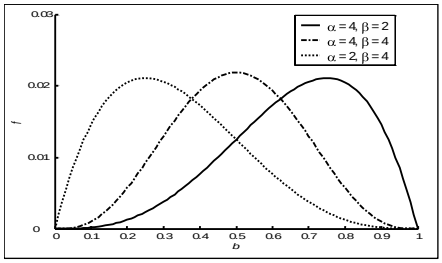


Fig. 1. Example beta plots, showing how the beta curve shape changes with the parameters α and β

Against this background, we now show how to calculate the value of τ_{a_1,a_2} based on the interaction outcomes observed by a_1. First, we must find values for α and β that represent the beliefs of a_1 about a_2. Assuming that prior to observing any interaction outcomes with a_2, a_1 believes that all possible values for B_{a_1,a_2} are equally likely, then a_1's initial settings for α and β are $\alpha = \beta = 1$. Based on standard techniques, the parameter settings in light of observations are achieved by adding the number of successful outcomes to the initial setting of α, and the number of unsuccessful outcomes to β. In our notation, this is given in Equation 5. Then the final value for τ_{a_1} is calculated by applying the standard equation for the expected value of a beta distribution (Equation 6) to these parameter settings.

$$\alpha = m^{1:t}_{a_1,a_2} + 1 \quad \text{and} \quad \beta = n^{1:t}_{a_1,a_2} + 1 \quad \text{where } t \text{ is the time of assessment} \tag{5}$$

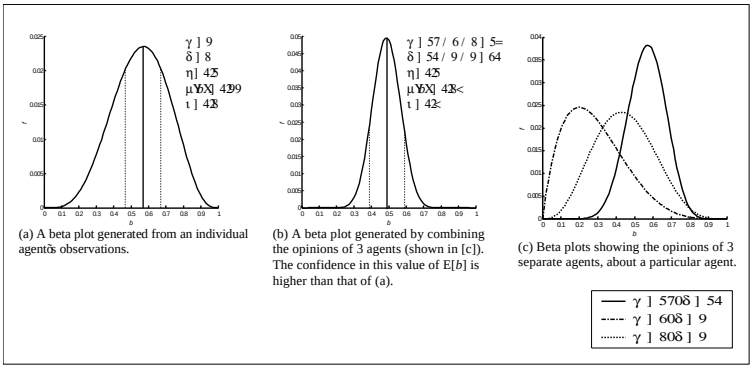
$$E[B|\alpha,\beta] = \frac{\alpha}{\alpha+\beta} \tag{6}$$

2.3 Modelling Confidence

In the previous section, we showed how an agent can establish trustworthiness so that it can be used to compare the trustworthiness of different agents. However, this method is susceptible to two problems created by the need for adequate evidence (observations) to calculate a meaningful value for trust. Firstly, an agent may not have interacted with another agent for which it is calculating a level of trust. This means that it has no personal experience and $m^t_{a_1,a_2} = n^t_{a_1,a_2} = 0$. Secondly, an agent may have had few interactions and observed outcomes with another. In both these cases, the calculated value of τ_{a_1,a_2} will be a poor estimate for the actual value of B_{a_1,a_2}. Intuitively, having observed many outcomes for an event will lead to a better estimate for the future probability for that event (assuming all other things are equal). These problems create the need for an agent to be able to measure its *confidence* in its value of trust. To this end, we incorporate a confidence metric in TRAVOS, based on standard methods of calculating confidence intervals taken from statistical analysis.

Specifically, the confidence metric γ_{a_1,a_2} is a measure of the probability that the actual value of B_{a_1,a_2} lies within an acceptable level of error ϵ about τ_{a_1,a_2}. It is calculated using Equation 7. The acceptable level of error ϵ influences how confident an agent is given the same number of observations. For example, if the number of observations remains constant, a larger value of ϵ causes an agent to be more confident in its calculation of trust than a lower value of ϵ.

$$\gamma_{a_1,a_2} = \frac{\int_{\tau_{a_1,a_2}-\epsilon}^{\tau_{a_1,a_2}+\epsilon} (B_{a_1,a_2})^{\alpha-1}(1-B_{a_1,a_2})^{\beta-1} dB_{a_1,a_2}}{\int_0^1 U^{\alpha-1}(1-U)^{\beta-1} dU} \tag{7}$$



(a) A beta plot generated from an individual agentõs observations.

(b) A beta plot generated by combining the opinions of 3 agents (shown in [c]). The confidence in this value of E[*b*] is higher than that of (a).

(c) Beta plots showing the opinions of 3 separate agents, about a particular agent.

Fig. 2. Example beta plots showing how 3 agents' opinions are aggregated to yield a more confident value of trust in a particular agent

2.4 Modelling Reputation

Until now, we have only considered how an agent uses its own direct observations to calculate a level of trust. However, by using confidence, we can specify a decision-making process in an agent to lead it to seek more evidence when required. In TRAVOS, an agent a_1 calculates τ_{a_1,a_2} based on its personal experiences with a_2. If this value of τ_{a_1,a_2} has a corresponding confidence γ_{a_1,a_2} below that of a predetermined *minimum confidence level*, denoted θ^γ, then a_1 will seek the opinions of other agents about a_2 to boost its confidence above θ^γ. These collective opinions form a_2's reputation and, by seeking it, a_1 can effectively obtain a larger set of observations.

The *true* opinion of a_3 at time t, about the trustee a_2, is the tuple, $\mathcal{R}^t_{a_3,a_2} = (m^t_{a_3,a_2}, n^t_{a_3,a_2})$, defined in Section 2.1. In addition, we denote the *reported* opinion of a_3 about a_2 at time t as $\hat{\mathcal{R}}^t_{a_3,a_2} = (\hat{m}^t_{a_3,a_2}, \hat{n}^t_{a_3,a_2})$. This distinction is important because a_3 may not reveal $\mathcal{R}^t_{a_3,a_2}$ truthfully. The truster, a_1, must form a single trust value from all such opinions that it receives. An elegant and efficient solution to this problem is to enumerate all the successful and unsuccessful interactions from the reports that it recieves (see Equation 8). The resulting values, denoted N_{a_1,a_2} and M_{a_1,a_2} respectively, represent the reputation of a_2 from the perspective of a_1. These values can then be used to calculate shape parameters (see Equation 9) for a beta distribution, to give a trust value determined by opinions provided from others. In addition, the truster takes on board any direct experience it has with the trustee, by adding its own values for n_{a_1,a_2} and m_{a_1,a_2} with the same equation. The confidence value γ_{a_1,a_2} for this combined distribution will be higher than for any of the component opinions, because more observations will have been taken into account (see Figure 2).

$$N_{a_1,a_2} = \sum_{k=0}^{p} \hat{n}_{a_k,a_2}, \quad M_{a_1,a_2} = \sum_{k=0}^{p} \hat{m}_{a_k,a_2}, \quad \text{where } p = \text{number of reports} \tag{8}$$

$$\alpha = M_{a_1,a_2} + 1 \quad \text{and} \quad \beta = N_{a_1,a_2} + 1 \tag{9}$$

The desirable feature of this approach is that, provided Conditions 1 & 2 hold, the resulting trust value and confidence level is the same as it would be if all the observations had been observed directly by the truster itself.

Condition 1. *The behaviour of the trustee must be independent of the identity of the truster it is interacting with. Specifically, the following should be true:* $\forall a_2 \quad \forall a_3, \quad B_{a_2,a_1} = B_{a_3,a_1}$.

Condition 2. *The reputation provider must report its observations accurately and truthfully. In other words, it must be true that:* $\forall a_2 \quad \forall a_3, \quad \mathcal{R}^t_{a_3,a_2} = \hat{\mathcal{R}}^t_{a_3,a_2}$.

Unfortunately, we cannot expect these conditions to hold in a broad range of situations. For instance, a trustee may value interactions with one agent over another, it might therefore commit more resources to the valued agent to increase its success rate, thus introducing a bias in its perceived behaviour. Similarly, in

the case of a rater's opinion of a trustee, it is possible that the rater has an incentive to misrepresent its true view of the trustee. Such an incentive could have a positive or negative effect on a trustee's reputation; if a strong co-operative relationship exists between trustee and rater, the rater may choose to overestimate its likelihood of success, whereas a competitive relationship may lead the rater to underestimate the trustee. Due to these possibilities, we consider the methods of dealing with inaccurate reputation sources an important requirement for a computational trust model. In the next section, we introduce our solution to this requirement, building upon the basic model introduced thus far.

3 Filtering Inaccurate Reputation Reports in TRAVOS

Inaccurate reputation reports can be due to the reputation report provider being malevolent or having incomplete information. In both cases, an agent must be able to assess the reliability of the report passed to it. The general solution to coping with inaccurate reputation reports is to adjust or ignore opinions judged to be unreliable (in order to reduce their effect on the trustee's reputation). There are two basic approaches to achieving this that have been proposed in the literature; these can be referred to as *endogenous* and *exogenous* methods. The former techniques attempt to identify unreliable reputation information by considering the statistical properties of the reported opinions alone (e.g. [12, 3]). The latter techniques rely on other information to make such judgements, such as the reputation of the source, or the relationship with the trustee (e.g. [1]).

Many proposals for endogenous techniques assume that inaccurate or unfair raters will generally be in a minority among reputation sources. Based on this assumption, they consider reputation providers whose opinions deviate in some way from mainstream opinion to be those most likely to be inaccurate. Our solution is also based on an endogenous approach, but we make our judgements on an individual basis — we judge a reputation provider on the perceived accuracy of its past opinions, rather than its deviation from mainstream opinion. More specifically, we calculate the probability that an agent will provide an accurate opinion given its past opinions, and later observed interactions with the trustees, for which those opinions were given. Using this value, we reduce the distance between a rater's opinion and the prior belief that all possible values for an agent's behaviour are equally probable. Once all the opinions collected about a trustee have been adjusted in this way, the opinions are aggregated using the technique described in Section 2.4.

In the following subsections we describe this technique in more detail: Section 3.1 describes how the probability of accuracy is calculated and then Section 3.2 shows how opinions are adjusted and the combined reputation obtained.

3.1 Calculating the Probability of Accuracy

The first stage in our solution is to estimate the probability that a rater's stated opinion of a trustee is an accurate predictor of the trustee's behaviour towards the truster. More specifically, let $\hat{\tau}_{a_3,a_2}$ be the trust value calculated using $\hat{\mathcal{R}}^t_{a_3,a_2}$.

With this in mind, our goal is to calculate the probability that $\hat{\tau}_{a_3,a_2} = B_{a_1,a_2}$. We denote this probability as ρ_{a_1,a_3} — the accuracy of a_3 according to a_1. If a_1 had observed sufficient direct interactions with a_2, then it would already have the means to calculate this probability: it is given using the beta pdf (Equation 4) with parameters set using a_1's direct experience with a_2. Unfortunately, the very reason that a_1 seeks reputation information about a_2 is because its direct experience with a_2 is not enough to make such a judgement accurately. However, we can avoid this problem by taking advantage of the fact that O_{a_1,a_2} is conditionally independent of the identity of a_2 given $B_{a_1,a2}$. In other words, if we had a set of agents $\mathcal{C} \subseteq \mathcal{A}$, and $\forall a_k \in \mathcal{C}, \forall a_l \in \mathcal{C}, B_{a_1,a_k} = B_{a_1,a_l}$, then regardless of which member of $\mathcal{C}$ we interacted with, the probability of that member fulfilling its obligations would be the same. This means that we could calculate $E[B_{a_1,a_k}]$ using not only the outcomes of interactions with a_k, but the outcomes of all interactions with any member of $\mathcal{C}$. In light of this, we can derive a beta probability function based on the outcomes of all interactions for which a rater gives the same value for $\hat{\tau}$ — regardless of which agents these opinions were given for. Using the parameter settings of this distribution, we can use Equation 4 to calculate ρ_{a_1,a_3} for a given value of $\hat{\tau}_{a_3,a_2}$.

Two problems must be solved before this solution can work in practice. First, the number of possible values of $\hat{\tau}_{a_3,a_2}$ is infinite — we cannot in general expect to see the same value of $\hat{\tau}_{a_3,a_2}$ more than once, so will never observe enough contracts to estimate ρ_{a_1,a_3} confidently. Second, for the same reason, ρ_{a_1,a_3} will always be vanishingly small. We solve both of these problems by approximation. All possible values for $\hat{\mathcal{R}}_{a_3,a_2}$ are split into bins according to the expected value and standard deviation of the resulting beta distribution (Equation 10); a single beta distribution is generated from all observations for which $\hat{\mathcal{R}}_{a_3,a_2}$ belongs to a given bin. For each bin, the probability that the true value of B_{a_1,a_2} lies within the range of expected values belonging to that bin is calculated — it is this value that we use for ρ_{a_1,a_3}. Calculation of this value is done by integrating Equation 4 over the expected value range of the bin.

$$\sigma = \sqrt{\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}} \text{where } \sigma \text{ is the standard deviation} \qquad (10)$$

3.2 Adjusting the Reputation Ratings

Our goal in adjusting a reputation source's opinion is to reduce the *effect* of unreliable opinions on a trustee's overall reputation. To achieve this, we consider the properties of a beta distribution, based on a single rater's opinion, that determine its effect on the final reputation value. Specifically, we consider the expected value of the distribution and its standard deviation. Effectively, by adding a rating to a trustee's reputation (Equation 8) we move the expected value of the combined distribution in the direction of the rater's opinion. The standard deviation of the opinion distribution contributes to the confidence value for the combined reputation value but, more importantly, its value relative to prior standard deviation determines how far towards the rater's opinion the

expected value will move. However, the relationship between the change in the expected value, and the standard deviation is non-linear. Consider as an example three distributions d_1, d_2 and d_3, with shape parameters, expected value and standard deviation (denoted σ) as shown in Table 1; the results of combining d_1 with each of the other two distributions are shown in the last two rows.

Table 1. Example beta distributions and the results of combining them

Distribution	α	β	$E[B]$	σ
d_1	540	280	0.6585	0.0165
d_2	200	200	0.5000	0.0250
d_3	5000	5000	0.5000	0.0050
$d_1 + d_2$	740	480	0.6066	0.0140
$d_1 + d_3$	5540	5280	0.5120	0.0048

As can be seen, distributions d_2 and d_3 have identical expected values with standard deviations of 0.025 and 0.005 respectively. Although the difference between these values is small (0.02), the result of combining d_1 with d_2 is quite different from combining d_1 and d_3. Whereas the expected value in the first case falls approximately between the expected values for d_2 and d_1, in the latter case, the relatively small parameter values of d_1 compared to d_3 mean that d_1 has virtually no impact on the combined result. Obviously, the reason for this is due to our method of reputation combination (Equation 8), in which the parameter values are summed. This is an important observation because it shows how, if left unchecked, an unfair rater could purposely increase the weight an agent puts in its opinion by providing very large values for m and n, which in turn determine α and β.

In light of this, we adopt an approach that significantly reduces very high parameter values unless the probability of the rater's opinion being accurate is very close to 1. Specifically, we reduce the distance between the expected value and standard deviation of the rater distribution, and the uniform distribution, $\alpha = \beta = 1$, which represents a state of no information; we denote the standard deviation of this uniform distribution as $\sigma_{uniform}$ and its expected value as $E_{uniform}$. Returning to our example scenario of a rater agent a_3 providing an opinion to agent a_1 about agent a_2, this is performed according to Equations 11 and 12 respectively. In these equations, we use the over-bar, for example $\bar{\alpha}$, to indicate that we are referring to the adjusted distribution. Adjusting the expected value as well as the standard deviation in this way results in a more conservative estimate. This is important in cases in which few more reliable ratings are available to mediate the expected value of the combined reputation.

$$\bar{E} = E_{uniform} + \rho_{a_1,a_3} \cdot (E - E_{uniform}) \tag{11}$$

$$\bar{\sigma} = \sigma_{uniform} + \rho_{a_1,a_3} \cdot (\sigma - \sigma_{uniform}) \tag{12}$$

Once all reputation opinions have been adjusted in this way, we sum the ratings as normal according to Equation 8. To do this, we must calculate the

adjusted values for $\hat{m}_{a_3,a_2}$ and $\hat{n}_{a_3,a_2}$. It can be shown that the adjusted parameter values $\bar{\alpha}$ and $\bar{\beta}$, can be calculated according to Equation 13 and Equation 14. The new values for $\hat{m}_{a_3,a_2}$ and $\hat{n}_{a_3,a_2}$ are then given by subtracting the prior parameter settings from the adjusted distribution parameters (Eqn. 15).

$$\bar{\alpha} = \frac{\bar{E}^2 - \bar{E}^3}{\bar{\sigma}^2} - \bar{E} \tag{13}$$

$$\bar{\beta} = \frac{(1-\bar{E})^2 - (1-\bar{E})^3}{\bar{\sigma}^2} - (1-\bar{E}) \tag{14}$$

$$\bar{m}_{a_3,a_2} = \bar{\alpha} - 1 \quad , \quad \bar{n}_{a_3,a_2} = \bar{\beta} - 1 \tag{15}$$

4 TRAVOS in Action

This section provides an agent-based VO scenario in which we demonstrate the use of TRAVOS. We begin by stating that there is a need to create a VO to meet a specific requirement to provide a composite multimedia communication service to an end user. The composite consists of the following basic services: text messaging, HTML content provision and phone calls (this example is taken from [9]). Now, assume agent a_1 has identified this need and wishes to capitalise on the market niche. However, a_1 only has the capability to provide a text messaging service. It can only achieve its goal by forming a VO with an agent that can supply a service for phone calls and one for HTML content. For simplicity, we assume that each agent in the system has the ability to provide only one service. Agent a_1 is aware of three agents that can provide a phone call service, and its interaction history with these is shown in Table 2. Similarly, it is aware of three agents that are capable of providing HTML content, and its past interactions with these entities are given in Table 3.

Table 2. Agent a_1's interaction history with phone call service provider agents

Agent	**Past interactions**	
	Successful	**Unsuccessful**
a_2	17	5
a_3	2	15
a_4	18	5

Table 3. Agent a_1's interaction history with HTML content service provider agents

Agent	**Past interactions**	
	Successful	**Unsuccessful**
a_5	9	14
a_6	3	0
a_7	18	11

Agent a_1 would like to choose the most trustworthy phone call and HTML content service provider from the selection. The following describes how this is achieved using TRAVOS. Before we calculate which of the possible candidates are the most trustworthy, we must specify certain parameters that a_1 requires. First, we specify the level of error that a_1 is willing to accept when determining the confidence in a calculated trust value as $\epsilon = 0.2$. Second, we specify that $\theta^{\gamma} = 0.95$, below which a_1 will seek other's opinions about the trustee.

4.1 Calculating Trust and Confidence

Using the information from Tables 2 and 3, a_1 can determine the number of successful interactions n, and the number of unsuccessful interactions m, for each agent it has interacted with. Feeding these into Equation 5, a_1 can obtain shape parameters for a beta distribution function that represents the behaviour of each service provider agent. For example, the shape parameters α and β, for a_2, are calculated as follows:

Using Table 2: $n_{a_1,a_2} = 17$, $m_{a_1,a_2} = 5$.

Using Equation 5: $\alpha = 17 + 1 = 18$ *and* $\beta = 5 + 1 = 6$.

The shape parameters for each agent are then used in Equation 6 to calculate a trust value for each agent that a_1 is assessing. For example, the trust value τ_{a_1,a_2} for a_2 is calculated as follows:

Using Equation 6: $\tau_{a_1,a_2} = \frac{\alpha}{\alpha+\beta} = \frac{18}{18+6} = 0.75$.

For a_1 to be able to use the trust values it obtains for each agent, it must also determine the confidence it has in the calculated trust value. This is achieved by using Equation 7 and the variable ϵ (which in this scenario has been set to 0.2). For example, the confidence γ_{a_1,a_2} that a_1 has in the trust value τ_{a_1,a_2} is calculated as shown below:

Using Equation 7:

$$\gamma_{a_1,a_2} = \frac{\int_{\tau_{a_1,a_2}+\epsilon}^{\tau_{a_1,a_2}-\epsilon} B^{\alpha-1}(1-B)^{\beta-1}dB}{\int_0^1 U^{\alpha-1}(1-U)^{\beta-1}dU} = \frac{\int_{0.95}^{0.55} B^{\alpha-1}(1-B)^{\beta-1}dB}{\int_0^1 U^{\alpha-1}(1-U)^{\beta-1}dU} = 0.98$$

Table 4. Agent a_1's calculated trust and associated confidence level for HTML content and phone call service provider agents

Agent	α	β	τ_{a_1,a_x}	γ_{a_1,a_x}
a_2	18	6	0.75	0.98
a_3	3	16	0.16	0.98
a_4	19	6	0.76	0.98
a_5	10	15	0.40	0.97
a_6	4	1	0.8	0.87
a_7	19	12	0.61	0.98

The shape parameters, trust values and associated confidence for each agent, a_2 to a_7, which a_1 computes using TRAVOS, are shown in Table 4. From this,

it is clear that the trust values for agents a_2, a_3 and a_4, all have a confidence above θ^γ (=0.95). This means that a_1 does not need to consider the opinions of others for these three agents. Agent a_1 is able to decide that a_4 is the most trustworthy out of the three phone call service provider agents and chooses it to provide the phone call service for the VO.

4.2 Calculating Reputation

The process of selecting the most trustworthy HTML content service provider is not as straightforward. Agent a_1 has calculated that out of the possible HTML service providers, a_6 has the highest trust value. However, it has determined that the confidence it is willing to place in this value is 0.87, which is below that of θ^γ and means that a_1 has not yet interacted with a_6 enough times to calculate a sufficiently confident trust value. In this case, a_1 has to use the opinions from other agents that have interacted with a_6, and form a reputation value for a_6 that it can compare to the trust values it has calculated for other HTML providers (a_5 and a_7).

Lets assume that a_1 is aware of three agents that have interacted with a_6, denoted by a_8, a_9 and a_{10}, whose opinions about a_6 are $(15, 46)$, $(4, 1)$ and $(3, 0)$ respectively. Agent a_1 can obtain beta shape parameters based solely on the opinions provided, by using Equations 9 and 8, as shown below:

Opinions from providers: $a_8 = (15, 46), a_9 = (4, 1)$ and $a_3 = (3, 0)$
Using Equation 8: $N = 15 + 4 + 3 = 22, M = 46 + 1 + 0 = 47$
Using Equation 9: $\alpha = 22 + 1 = 23, \beta = 47 + 1 = 48$

Having obtained the shape parameters, a_1 can obtain a trust value for a_6 using Equation 6, as follows:

Using Equation 6: $\tau_{a_1,a_6} = \frac{\alpha}{\alpha+\beta} = \frac{23}{23+48} = 0.32$

Now a_1 is able to compare the trust in agents a_5, a_6 and a_7. Before calculating the trustworthiness of a_6, agent a_1 considered a_6 to be the most trustworthy (see Table 4). Having calculated a new trust value for agent a_6 (which is lower than the first assessment), agent a_1 now regards a_7 as the most trustworthy. Therefore a_1 chooses a_7 as the service provider for the HTML content service.

4.3 Handling Inaccurate Opinions

The method a_1 uses to assess the trustworthiness of a_6, as described in Section 4.2, is susceptible to errors caused by reputation providers giving inaccurate information. In our scenario, suppose a_8 provides the HTML content service too, and is in direct competition with a_6. Agent a_1 is not aware of this fact, which makes a_1 unaware that a_8 may provide inaccurate information about a_6 to influence its decision on which HTML content provider agent to incorporate into the VO. If we look at the opinions provided by agents a_8, a_9 and a_{10}, which are $(20, 46)$, $(4, 1)$ and $(3, 0)$ respectively, we can see that the opinion provided by a_8 does not correlate with the other two. Agents a_9 and a_{10} provide

a positive opinion of a_6, whereas agent a_8 provides a very negative opinion. Suppose that a_8 is providing an inaccurate account of its experiences with a_6. We can use the mechanism discussed in Section 3 to allow a_1 to cope with this inaccurate information, and arrive at a better decision that is not influenced by self-interested reputation providing agents (such as a_8).

Before we show how TRAVOS can be used to handle inaccurate information, we must assume the following. Agent a_1 obtained reputation information from a_8, a_9 and a_{10} on several occasions, and each time a_1 recorded the opinion provided by a reputation provider and the actual observed outcome (from the interaction with an agent to which the opinion is applied). Each time an opinion is provided, the outcome observed is recorded in the relevant bin. Agent a_1 keeps information of like opinions in bins as shown in Table 6. For example, if a_8 provides an opinion that is used to obtain a trust value of 0.3, then the actual observed outcome (successful or unsuccessful) is stored in the $0.2 < E[B] \leq 0.4$ bin.

Using the information shown in Table 6, agent a_1 can calculate the weighting to be applied to the opinions from the three reputation sources by applying the technique described in Section 3.1. In so doing, agent a_1 uses the information from the bin, which contains the opinion provided, and integrates the beta distribution between the limits defined by the bin's boundary. For example, a_8's opinion falls under the $0.2 < E[B] \leq 0.4$ bin. In this bin, agent a_1 has recorded that $n = 15$ and $m = 3$. These n and m values are used to obtain a beta distribution, using Equations 4 and 5, which is then integrated between 0.2 and 0.4 to give a weighting of 0.0039 for a_6's opinion. Then, by using Equations 11 and 12, agent a_1 can calculate the adjusted mean and standard deviation of the opinion, which in turn gives the adjusted α and β parameters for that opinion. The results from these calculations are shown in Table 5.

Table 5. Agent a_1's adjusted values for opinions provided by a_8, a_9 and a_{10}

Agent	**Weighting**	**Adjusted Values**			
		μ	σ	α	β
a_8	0.0039	0.5	0.29	1.0091	1.0054
a_9	0.78	0.65	0.15	5.8166	3.1839
a_{10}	0.74	0.62	0.17	4.3348	2.6194

Summing the adjusted values for α and β from Table 5, a_1 can obtain a more reliable value for the trustworthiness of a_6. Using Equation 4, a_1 calculates a trust value = 0.62 for a_6. This means that from the possible HTML content providers, a_1 now sees a_6 as the most trustworthy and selects it to be a partner in the VO. Unlike a_1's decision in Section 4.2 (when a_7 was chosen as the VO partner), here we have shown how a reputation provider cannot influence the decision made by a_1 by providing inaccurate information.

Table 6. Observations made by a_1 given opinion from a reputation source. n represents that the interaction (to which the opinion applied) was successful, and likewise m means unsuccessful

	[0, 0.2]		[0.2, 0.4]		[0.4, 0.6]		[0.6, 0.8]		[0.8, 1]		**Total**
	n	**m**	**n**	**m**	**n**	**m**	**n**	**m**	**n**	**m**	
a_8	2	0	11	4	0	0	0	0	2	3	25
a_9	0	2	1	3	0	0	22	10	6	4	30
a_{10}	1	3	0	2	0	0	18	8	5	3	25

5 Related Work

There are many computational models of trust, a review of which can be found in [10]. Generally speaking, however, models not based on probability theory (e.g. [6, 11, 8]) consist of calculating trust from hand-crafted formulae that yield the desired results. For the purpose of our work, we only consider models that are similar to TRAVOS in the manner of calculating trust.

Probabilistic approaches are not commonly used in the field of computational trust, but there are a couple of such models in the literature. In particular, the Beta Reputation System (BRS) [7] is a probabilistic trust model like TRAVOS, which is based on the beta distribution. The system is specifically designed for online communities and is centralised. It works by users giving ratings to the performance of other users in the community. Here, ratings consist of a single value that is used to obtain positive and negative feedback values. The feedback values are then used to calculate shape parameters that determine the reputation of the user the rating applies to. However, BRS does not show how it is able to cope with misleading information.

Whitby et al. [12] also build on the BRS and show how it can be used to filter unfair ratings, either unfairly positive or negative, towards a certain agent. In their model, the ratings for an individual are stored in a vector, which is then used to obtain an aggregated reputation value for that individual (represented as a beta distribution). However, this method is only effective when there are sufficient ratings to allow successful identification of unfair ratings. Filtering in this manner disregards the possibility of ratings that may seem unfair, but which truthfully represent the perspective of the rater (as the rater may have an inaccurate view of the world). In TRAVOS, no opinion is disregarded, and, instead, the consistency between the reputation provider's opinion and the actual behaviour of an individual to which the opinion refers forms the weight that is to be placed in opinions provided by that particular reputation source.

6 Conclusions and Future Work

This paper has presented a novel model of trust, TRAVOS, for use in open agent systems. The main benefits in using TRAVOS are that it provides a mechanism

for assessing the trustworthiness of others in situations both in which the agents have interacted before and share past experiences, and in which there is little or no past experience between the interacting agents. Establishing the trustworthiness of others, and then selecting the most trustworthy, gives an agent the ability to maximise the probability that there will be no harmful repercussions from the interaction. In particular, through the example scenario, we have demonstrated how TRAVOS can be used to handle inaccurate opinions expressed by reputation providers. Here, reputation providers are requested to provide opinions about a certain individual when an agent is not confident in the amount of evidence it has in order to assess the trustworthiness of that individual. In this situation, it is particularly important that the opinions that are given are accurate and based on past experiences. By using TRAVOS, an agent ensures that opinions from inaccurate reputation sources are not given equal weighting to those from accurate sources in the aggregation of reputation. This gives the agent the ability to handle inaccurate information effectively.

In the short term, we will be carrying out empirical analysis on TRAVOS, and evaluating it against similar approaches. As it stands, TRAVOS assumes that the behaviour of agents does not change over time, but in many cases this is an unsafe assumption. In particular we believe that agents may well change their behaviour over time, and that some will have time-based behavioural strategies. Future work will therefore include the removal of this assumption and the use of functions that allow an agent to take into account the fact that very old experiences may not be relevant in predicting the behaviour of an individual. In addition we will extend the model to represent a continuous outcome space, instead of the current binary outcome space. Further extensions to TRAVOS will include using the rich social metadata that exists within a VO environment in the calculation of a trust value. Thus, as described in Section 1, VOs are social structures, and we can draw out social data such as roles and relationships that exist both between VOs and VO members. The incorporation of such data into the trust metric should allow for more accurate trust assessments to be formed.

References

1. S. Buchegger and J. Y. L. Boudec. A robust reputation system for mobile ad-hoc networks ic/2003/50. Technical report, EPFL-IC-LCA, 2003.
2. M. DeGroot and M. Schervish. *Probability & Statistics*. Addison-Wesley, 2002.
3. C. Dellarocas. Mechanisms for coping with unfair ratings and discriminatory behavior in online reputation reporting systems. In *ICIS*, pages 520–525, 2000.
4. I. Foster, N. R. Jennings, and C. Kesselman. Brain meets brawn: Why grid and agents need each other. In *Proceedings of the 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems*, pages 8–15, 2004.
5. D. Gambetta. Can we trust trust? In D. Gambetta, editor, *Trust: Making and Breaking Cooperative Relations*, chapter 13, pages 213–237. Basil Blackwell, 1988.
6. T. D. Huynh, N. R. Jennings, and N. Shadbolt. Developing an integrated trust and reputation model for open multi-agent systems. In *Proceedings of the 7th Int Workshop on Trust in Agent Societies*, pages 62–77, 2004.

7. R. Ismail and A. Jøsang. The beta reputation system. In *Proceedings of the 15th Bled Conference on Electronic Commerce*, Bled, Slovenia, 2002.
8. A. Moukas, G. Zacharia, and P. Maes. Amalthaea and histos: Multi-agent systems for www sites and reputation recommendations. In M. Klusch, editor, *Intelligent Information Agents*, chapter 13. Springer-Verlag, 1999.
9. T. J. Norman, A. Preece, S. Chalmers, N. R. Jennings, M. Luck, V. Dang, T. D. Nguyen, V. Deora, , J. Shao, A. Gray, and N. J. Fiddian. Agent-based formation of virtual organisations. *Knowledge-Based Systems*, 17(2–4):103–111, May 2004.
10. S. D. Ramchurn, D. Hunyh, and N. R. Jennings. Trust in multi-agent systems. *Knowledge Engineering Review*, 19(1), 2004.
11. J. Sabater and C. Sierra. Regret: A reputation model for gregarious societies. In *4th Workshop on Deception Fraud & Trust in Agent Societies*, pages 61–70, 2001.
12. A. Whitby, A. Jøsang, and J. Indulska. Filtering out unfair ratings in bayesian reputation systems. In *Proceedings of the Workshop on Trust in Agent Societies, at the 3rd Int. Conf. on Autonomous Agents & Multi Agent Systems*, 2004.