Bayesian Network Based Trust Management*

Yong Wang[1,2], Vinny Cahill[1], Elizabeth Gray[1],
Colin Harris[1], and Lejian Liao[3]

[1] Distributed Systems Group, Department of Computer Science,
Trinity College Dublin, Dublin 2, Ireland
wangyong@bit.edu.cn, {vinny.cahill, grayl, colin.harris}@cs.tcd.ie
http://www.dsg.cs.tcd.ie
[2] School of Software, Beijing Institute of Technology, Beijing 100081, China
[3] School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China
liaolj@bit.edu.cn

Abstract. Trust is an essential component for secure collaboration in uncertain environments. Trust management can be used to reason about future interactions between entities. In reputation-based trust management, an entity's reputation is usually built on ratings from those who have had direct interactions with the entity. In this paper, we propose a Bayesian network based trust management model. In order to infer trust in different aspects of an entity's behavior, we use multi-dimensional application specific trust values and each dimension is evaluated using a single Bayesian network. This makes it easy both to extend the model to involve more dimensions of trust and to combine Bayesian networks to form an opinion about the overall trustworthiness of an entity. Each entity can evaluate his peers according to his own criteria. The dynamic characteristics of criteria and of peer behavior can be captured by updating Bayesian networks. Risk is explicitly combined with trust to help users making decisions. In this paper, we show that our system can make accurate trust inferences and is robust against unfair raters.

1 Introduction

Ubiquitous computing foresees a massively networked infrastructure supporting a large population of diverse but cooperation entities. Entities will be both autonomous and mobile and will have to be able to capable of dealing with unforeseen circumstances ranging from unexpected interactions with other entities to disconnected operation [1]. Trust management is a suitable solution to provide self-protection for these entities by reasoning about another entity's trustworthiness in future interactions according to one's direct interactions with that entity and recommendations (ratings) from other entities.

Trust is a multi-faceted concept. It is subjective and non-symmetric [1]. Each entity makes its own decision to trust or not based on the evidence available for personal evaluation. Even if two entities are presented with the same evidence they may not necessarily interpret this information in the same way. Trust is also context-specific such that trust in one environment does not directly transfer to another environment. Trust usually changes according to the outcomes of the latest interactions, which means that the dynamic property of trust must be captured. Trust is inherently linked to risk and explicitly involving risk can help users to understand the semantics of trust. In order to provide finer-grained inference of trust in different aspects of an entity's behavior, trust should also be multi-dimensional.

Many existing computational trust models use intuitive and ad hoc hand-crafted formulae to calculate trust values. Some models using probabilistic approaches based on Bayesian theory have also appeared during last few years. However none of these models consider all the characteristics of trust described above. For example, ratings are weighted and then summed to get reputation values in [2][3][6], but the weights are selected in an ad hoc way and remain unchanged over time, which does not reflect the dynamic context. All ratings are treated equally in [2][3], a rater's confidence in his level of trust for a ratee is not considered. As to the unfair rating problem, [4][8][10] use the difference between the aggregated and individual ideas about an entity to determine the reliability of raters. The subjectivity of trust is not taken into consideration. If the proportion of unfair raters is large, say 40%, this kind of system can not determine which raters are malicious. However, the approach proposed in [5] is not practical in many distributed applications (e.g., E-commerce), since the actual outcomes of interactions between two agents can not be observed by other agents. [2][6] do not consider risk explicitly. [2][3][6][9][10] use a single value or a pair to represent trust in a user, making it difficult to evaluate different dimensions of trust.

The above Bayesian-based trust models simply sum rather than using statistical methods to combine direct observations and weighted ratings from different raters as beta distribution parameters. It is difficult to show whether the aggregated beta distribution is close to the real distribution. Bayesian networks can be used to tackle this issue. There are two trust models based on Bayesian networks. The authors of [6] use Bayesian networks to combine different dimensions of trust and estimate the reliability of raters. But the reputation values are still calculated using hand-crafted formula with fixed weights. Their system with Bayesian networks performs only slightly better than the one without Bayesian networks. In [7], the authors use a polytree (singly connected directed acrylic graph) to revise belief in some knowledge and to update the reputation of information sources. In fact, it uses the aggregated reputation as evidence to determine the reliability of information sources. In addition, the assumption of the probability of unreliable information sources giving correct information reduces the accuracy of inference. In this paper, we describe a probabilistic computational trust model which covers all important characteristics of trust. We use the e-commerce scenario, where buyers evaluate the trustworthiness of sellers, to illustrate the model in the following sections.

2 Bayesian Network Based Trust Model Overview

Our model consists of three components: trust formation, reputation estimation and risk-related decision making.

A trust value is calculated as the expectation value of beta probability density functions as [2][3][9][10], but it is finer-grained in our model to represent more application-specific notions of trust and to allow the definition of trust to be extended flexibly. For example, a buyer can evaluate the trustworthiness of a seller in three dimensions: the probability of shipping goods as described, the probability of shipping lower quality goods and the probability of not shipping any goods.

Only ratings from reliable buyers (raters) are used to estimate the sellers' reputation. First, we compare the ratings of the decision maker and other raters. If the probability of having similar opinions to the decision maker is very small, the rater will be regarded as unfair. Both unfairly positive raters and unfairly negative raters can be identified.

After filtering out unreliable raters, we use a Bayesian Network to estimate reputation values from ratings given by reliable raters.

A reputation value for each dimension of trust is calculated. Since decision makers have different attitudes towards risk and their attitudes can change over time, we should combine reputation values and risk attitudes in order to make subjective and context-specific decisions. A natural way is to use utility functions to model attitudes towards risk and then use the estimated reputation values as parameters to calculate the expected utility, which can be used as the basis of the decision.

3 Bayesian Network Based Trust Management

3.1 Trust Formation

We use beta probability density functions to represent the distribution of trust values according to interaction history as in [2]. The beta-family of distributions is a continuous family of distribution functions indexed by the two parameters and . The beta PDF can be expressed using the gamma function as:

$$\mathrm{beta}(p\,|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{(\alpha-1)}(1-p)^{(\beta-1)}, where\ 0 \le p \le 1,\ \alpha,\beta > 0. \tag{1}$$

The probability expectation value of the beta distribution is given by:

$$\mathrm{E}(p) = \frac{\alpha}{\alpha+\beta} \tag{2}$$

Posteriori probabilities of binary events can be represented as beta distributions. Suppose a process has several possible outcomes and one of them is outcome x. Let r be the observed number of outcome x and let s be the number of outcomes other than x. Then the probability density function of observing outcome x in the future can be expressed as a function of past observations by setting:

$$\alpha = r + 1, \beta = s + 1, where\ \ r, s \ge 0. \tag{3}$$

The authors of [2] only consider two possible outcomes for each interaction. We make more detailed analysis of interactions and extend to multi-dimension application-specific outcomes. The granularity (the number of dimensions) is determined by the complexity of applications and the requirement of users.

In the e-commerce scenario, we use three tuples $(r_G, s_{\overline{G}})$,$(r_L, s_{\overline{L}})$,$(r_C, s_{\overline{C}})$ for three dimensions of trust of a seller. G means shipping goods as described and $\overline{G}$ means not shipping goods as described. L means shipping lower quality goods and $\overline{L}$ means not shipping lower quality goods. C means not shipping any goods and $\overline{C}$ means shipping goods. are the numbers of interactions with outcome i, and $r_i(i \in \{G, L, C\})$ is the number of interactions without outcome i. Then the parameters of the beta probability density functions are set as:

$$\alpha_{\mathrm{i}} = r_i \ + \ 1, \beta_{\bar{\mathrm{i}}} = s_{\bar{i}} \ + \ 1, \ \ (i \in \{G, L, C\}, \bar{i} \in \{\overline{G}, \overline{L}, \overline{C}\}). \tag{4}$$

$$\tau_{\mathrm{i}} = \frac{\alpha_i}{\alpha_i + \beta_{\bar{i}}}, \ \ (i \in \{G, L, C\}, \bar{i} \in \{\overline{G}, \overline{L}, \overline{C}\}). \tag{5}$$

$$\text{Since} \qquad \beta_{\overline{\mathrm{G}}} = \alpha_L + \alpha_C - 1, \tag{6}$$

$$\text{then} \qquad \tau_{\mathrm{i}} = \frac{\alpha_i}{\sum\limits_{j \in \{G,L,C\}} \alpha_j - 1} = \frac{r_i \ + \ 1}{\sum\limits_{j \in \{G,L,C\}} r_j \ + \ 2} \, (i \in \{G, L, C\}). \tag{7}$$

We then normalize these to get the trust values:

$$P_i = \frac{\tau_i}{\sum\limits_{j \in \{G,L,C\}} \tau_j} \, (i \in \{G, L, C\}). \tag{8}$$

Where, P_i can be interpreted as the probability of outcome i happening in the future.

In our model, we consider buyers' confidence $\gamma_i(i \in \{G, L, C\})$ in calculated trust values, which is a measure of the probability that the actual trust value lies within an acceptable level of error ϵ about the calculated trust value P_i. The confidence factor can be calculated as (9):

$$\gamma_i = \frac{\int_{P_i - \epsilon}^{P_i + \epsilon} \kappa^{\alpha_i - 1} (1 - \kappa)^{\beta_{\bar{i}} - 1} \mathrm{d}\kappa}{\int_0^1 \rho^{\alpha_i - 1} (1 - \rho)^{\beta_{\bar{i}} - 1} \mathrm{d}\rho} \, (i \in \{G, L, C\}). \tag{9}$$

Then the buyer can set a threshold θ_γ to determine if he has enough confidence. If the buyer's trust value about a seller fulfills formula (10), then he feels confident to predict the seller's future behavior.

$$\gamma_G \ > \ \theta_\gamma, \, \gamma_L \ > \ \theta_\gamma, \, \gamma_C \ > \ \theta_\gamma \tag{10}$$

Since the three tuples have relationships (see formula (6)), we can combine them to save storage space. The formats of trust values and confidence factors for sellers stored by buyers are (P_G, P_L, P_C) and $(\gamma_G, \gamma_L, \gamma_C)$ respectively.

Because sellers may change their behavior over time, we use a fading factor $\lambda_\gamma \in [0, 1]$ to forget old observations as (11).

$$r_{t+1} = \lambda_\gamma r_t + r \qquad s_{t+1} = \lambda_\gamma s_t + s. \tag{11}$$

In which, r_{t+1} and s_{t+1} are the trust parameters for t+1 interactions, and the outcome of the t+1th interaction is (r, s).

3.2 Reputation Estimation

If a buyer is not confident enough in a seller's trustworthiness according to formula (10) or wants to interact with an unknown seller, he will ask for other buyers' advice or recommendations (ratings). The ratings are the trust values given to sellers by other buyers. Then the buyer can estimate the seller's reputation according to the ratings for the seller as well as the other buyers' reliability as raters. A Bayesian network is constructed to perform the estimation for each dimension of trust (shipping described goods or shipping lower quality goods or not shipping goods in the E-commence scenario). All Bayesian networks have the same structure but different parameters. Figure 1 shows the structure of Bayesian networks.

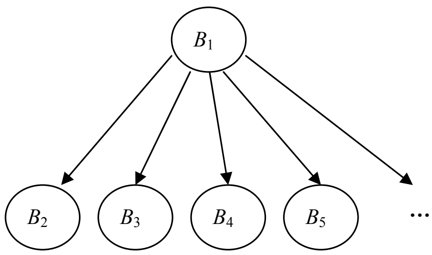

Fig. 1. Bayesian Network for Reputation Estimation

Node B_i represents one dimension of the trust value (P_G, P_L, P_C) given by buyer b_i. Since it is buyer b_1 that estimates a sellers' reputation, B_1 is the root node and others are leaf nodes. The trust value is a real number between 0 and 1. In order to express a trust value using the state of a node, we need to discretise it by dividing the interval into several subintervals according to the required accuracy denoted as the comparison threshold θ_P^1 set by b_1. The subintervals are $[k\theta_P^1, (k+1)\theta_P^1)$, $(k = 0, 1, \ldots, \frac{1}{\theta_P^1} - 1)$ and the number of subintervals is $\frac{1}{\theta_P^1}$, which determines the numbers of states for each node. For simplicity, we denote the subintervals with $I_k\,(k = 0, 1, \ldots, \frac{1}{\theta_P^1} - 1)$ and denote the states as $S_i\,(i = 0, 1, \ldots, \frac{1}{\theta_P^1} - 1)$. The conditional probability table (CPT) of node B_1 and node $B_i\,(i = 2, 3, 4, \ldots)$ are shown in table 1 and table 2 respectively.

Table 1. The CPT of Node B_1

$B_1 = S_0$	$B_1 = S_1$	...	$B_1 = S_{\frac{1}{\theta_P^1}-1}$
$P(B_1 \in I_0)$	$P(B_1 \in I_1)$	...	$P(B_1 \in I_{\frac{1}{\theta_P^1}-1})$

Table 2. The CPT of Node B_i

B_1	$B_i = S_0$	$B_i = S_1$	...	$B_i = S_{\frac{1}{\theta_P^1}-1}$
S_0	$P(B_i \in I_0 \mid B_1 \in I_0)$	$P(B_i \in I_1 \mid B_1 \in I_0)$	...	$P(B_i \in I_{\frac{1}{\theta_P^1}-1} \mid B_1 \in I_0)$
S_1	$P(B_i \in I_0 \mid B_1 \in I_1)$	$P(B_i \in I_1 \mid B_1 \in I_1)$	...	$P(B_i \in I_{\frac{1}{\theta_P^1}-1} \mid B_1 \in I_1)$
...	...	...	...	...
$S_{\frac{1}{\theta_P^1}-1}$	$P(B_i \in I_0 \mid B_1 \in I_{\frac{1}{\theta_P^1}-1})$	$P(B_i \in I_1 \mid B_1 \in I_{\frac{1}{\theta_P^1}-1})$	...	$P(B_i \in I_{\frac{1}{\theta_P^1}-1} \mid B_1 \in I_{\frac{1}{\theta_P^1}-1})$

Next we use the expectation maximization algorithm [11] to learn probabilities in CPTs from the ratings, which are called cases. Let the rating of shipping described goods for seller s_j given by b_i at time 0 be denoted as $P^j_{i(0)}$, then we can get the cases shown in table 3. In this table, the elements in column "B_1" are the trust values given by b_1 in which he has sufficient confidence according to (10).

Table 3. Cases for CPT Learning

B_1	B_2	...	B_i
$P^1_{1(0)}$	$P^1_{2(0)}$	...	$P^1_{i(0)}$
$P^2_{1(0)}$	$P^2_{2(0)}$	...	$P^2_{i(0)}$
...	...	...	...
$P^1_{1(1)}$	$P^1_{2(1)}$	...	$P^1_{i(1)}$

To reflect the dynamic characteristic of trust, a buyer should update his Bayesian networks regularly by fading old probabilities before taking new cases into consideration. We adopt the method from Netica [11] shown in formula (12) and (13) to update CPTs, in which, $\lambda_\gamma \in [0, 1]$ is the fading factor; P_i $(i \geq 0)$ are the probabilities at time i; t_i $(i \geq 0)$ is the normalization constant at time i.

$$t_0 = 1; \qquad t_{n+1} = \frac{1}{\sum_{j=0}^{\frac{1}{\theta_P^1}-1} (P_n(B_i \in I_j \mid B_1 \in I_k) t_n \lambda_\gamma + 1 - \lambda_\gamma)} \tag{12}$$

$$P_0(B_i \in I_j | B_1 \in I_k) = \frac{1}{\theta_P^1};$$
$$P_{n+1}(B_i \in I_j | B_1 \in I_k) = \frac{1}{t_{n+1}(P_n(B_i \in I_j | B_1 \in I_k) t_n \lambda_\gamma + 1 - \lambda_\gamma)} \tag{13}$$

The reputation of a seller can be estimated easily using Bayes rule. Suppose that the recommendation given to a seller by b_i lies in I_i, then the probability that b_i's rating lies in I_{i_1} can be calculated using formula (14). Since all leaf nodes B_i are conditionally independent given B_1 , we can deduce formula (15) to formula (16). The value of B_1 can be estimated as the expectation value of its states using formula (16).

$$P(B_1 \in I_{i_1} \mid B_2 \in I_{i_2}, \ldots, B_k \in I_{i_k}) = \frac{P(B_2 \in I_{i_2}, \ldots B_k \in I_{i_k} \mid B_1 \in I_{i_1}) P(B_1 \in I_{i_1})}{P(B_2 \in I_{i_2}, \ldots, B_k \in I_{i_k})} \tag{14}$$

$$P(B_1 \in I_{i_1} \mid B_2 \in I_{i_2}, \ldots, B_k \in I_{i_k}) = \frac{P(B_2 \in I_{i_2} \mid B_1 \in I_{i_1}) \ldots P(B_k \in I_{i_k} \mid B_1 \in I_{i_1}) P(B_1 \in I_{i_1})}{P(B_2 \in I_{i_2}) \ldots P(B_k \in I_{i_k})} \tag{15}$$

$$P = \sum_{i=0}^{i=\frac{1}{\theta_P^1}-1} \frac{(2i+1)\theta_P^1}{2} P(B_1 \in I_i) \tag{16}$$

In order to increase the accuracy of estimates and to reduce the computational overhead, we can first select the most reliable buyers as raters and then use only the recommendations from these reliable buyers to infer sellers' reputations. In other words, in the Bayesian network, only the nodes corresponding to reliable buyers are set to specific states according to the ratings, the other nodes are set to an unknown state.

If the trust value calculated by b_i according to his own observations lies in the same or adjacent subintervals as the trust value given by b_1, then b_i has a similar opinion to b_1. From the CPT's point of view, the probabilities on the three diagonals from top left corner to bottom right corner represent situations where b_i has similar opinions to b_1. Then we use the average value of the probabilities on the three diagonals to evaluate b_i's reliability in rating sellers. If no trust value from b_1 lies in I_j, then $P(B_1 \in I_j) = 0$ or close to 0 (different parameter learning algorithms give different estimation), which means b_1 has no idea about sellers whose trust values lie in I_j. So it is not reasonable to consider $P(B_i \in I_k \mid B_1 \in I_j), (k = 0, 1, \ldots, \frac{1}{\theta_P^1} - 1)$ when calculating b_i's reliability. Usually, uniform probabilities are given to unknown states, that is $P(B_i \in I_k \mid B_1 \in I_j) = \theta_P^1, (k = 0, 1, \ldots, \frac{1}{\theta_P^1} - 1)$. The number of subintervals covered by b_1's interaction experience is denoted as N_1, and then the reliability of b_i can be calculated using formula (17).

$$P(b_i \text{ is reliable}) = \frac{1}{N_1}\Big(\sum_{j=0, P(B_1 \in I_j) \neq 0}^{j=\frac{1}{\theta_P^1}-1} P(B_i \in I_j \mid B_1 \in I_j) + \sum_{j=1, P(B_1 \in I_j) \neq 0}^{j=\frac{1}{\theta_P^1}-1} P(B_i \in I_{j-1} \mid B_1 \in I_j)$$
$$+ \sum_{j=0, P(B_1 \in I_j) \neq 0}^{j=\frac{1}{\theta_P^1}-2} P(B_i \in I_{j+1} \mid B_1 \in I_j) \tag{17}$$

b_1 then sets a threshold θ_R^1 to determine b_i's reliability. If formula (18) is fulfilled, then b_i is reliable enough in providing ratings.

$$P(b_i\ is\ reliable) \geq \theta_R^I \tag{18}$$

The probabilities below the three diagonals represent b_i giving lower ratings than b_1. Then the average value of these probabilities can be used to estimate the probability that b_i gives unfairly negative ratings as (19). Similarly, the probabilities above the three lines represent b_i giving higher ratings than b_1. The average value of these probabilities can be used to estimate the probability that b_i gives unfairly positive ratings as (20).

$$P(b_i\ is\ unfairly\ negative) = \frac{1}{N_1} \sum_{j=2, P(B_1 \in I_j) \neq 0}^{j=\frac{1}{\theta_P^I}-1} \sum_{k=0}^{k=j-2} P(B_i \in I_k \mid B_1 \in I_j) \tag{19}$$

$$P(b_i\ is\ unfairly\ positive) = \frac{1}{N_1} \sum_{j=0, P(B_1 \in I_j) \neq 0}^{j=\frac{1}{\theta_P^I}-2} \sum_{k=2}^{k=j+2} P(B_i \in I_k \mid B_1 \in I_j) \tag{20}$$

3.3 Decision Making Based on Reputation and Risk

After obtaining reputation values (P_G, P_L, P_C) for a seller, b_1 normalizes them to get (P'_G, P'_L, P'_C) which are then used to calculate the utility of dealing with the seller. Buyers can select a utility function according to their attitude to risk. A buyer with risk-tolerant behavior can choose an exponential function for the utility, a risk neutral buyer can choose a linear one and a risk-averse buyer can choose a logarithmic function.

Taking an exponential utility function $U_R(x) = 1 - e^{\frac{-x}{R}}$ as an example, the parameter R, called the risk tolerance, determines how risk-averse the function is. As R becomes smaller, the function becomes more risk-averse. Suppose that the price of an item is q and the intrinsic value of it is v. If the seller ships the item as described, the gain of b_1 is $v - q$. If the seller ships a lower quality item, whose intrinsic value is $v' < v$, the gain of b_1 is $v' - q$. In case the seller does not ship any item, the gain of b_1 is $-q$. The utility function is $U_R(x)$, thus the expected utility can be calculated as (21). $EU > 0$ means that dealing with the seller is worth the risk, otherwise it is too risky.

$$EU = P'_G \times U_R(v - q) + P'_L \times U_R(v' - q) + +P\prime_C \times U_R(-q) \tag{21}$$

4 Simulation

To evaluate the effectiveness of our approach in estimating reputation values in different scenarios and filtering unfair ratings, we used simulations. We revised the Trade Network Game (TNG)[12] to simulate an e-commerce environment. TNG is open source software that implements an agent-based computational model for studying the formation and evolution of trade networks in decentralized market economies. We modified the trader behavior sequence generation and partner selection part as described below and added a rating function for buyers.

For the sake of simplicity, each agent in our system plays only one role at a time, either seller or buyer. The behaviors of sellers are determined by behavioral probabilities (the probabilities of shipping described goods, shipping lower quality goods and not shipping goods). Buyers rate sellers differently. They can be honest, unfairly positive or unfairly negative or combined according to some rating behavior patterns. We use the Netica API [11]to implement the Bayesian Networks.

4.1 Simulation Setup

The market consists of 5 sellers and 10 buyers. The simulation is divided into 800 sessions and each session has 10 rounds. In each round, each buyer can buy one item with fixed price from a selected seller who has the maximal expectation utility according to (20), while each seller can interact with as many buyers as its productive capacity allows. We use similar behavior patterns for sellers as [3]. Sellers' original behavioral probabilities B_G,B_L and B_C are 0.85, 0.1 and 0.05 respectively. After each session, sellers change their behavioral probabilities randomly as follows.

$$\begin{aligned}
&B_G = B_G + \delta, B_L = B_L - \frac{\delta}{2}, B_C = B_C - \frac{\delta}{2}\,(with\,probability\,0.33)\\
&B_G = B_G - \delta, B_L = B_L + \frac{\delta}{2}, B_C = B_C + \frac{\delta}{2}\,(with\,probability\,0.33)\\
&B_G = B_G, B_L = B_L, B_C = B_C\,(with\,probability\,0.34)\\
&\delta = 0.01
\end{aligned} \tag{22}$$

Buyers exchange ratings for sellers after each session. The exchanged ratings of the first 20 sessions are used to estimate the reliabilities of other buyers. They update their Bayesian networks by fading all the CPTs first, then learning again using the latest case. $\lambda_v = 0.98$; $\lambda_\gamma = 0.99$; $\theta_p^I = 0.05$; $\theta_R^I = 0.8$; $q = 100$, $v = 200$, v'=60; the utility function is $U_{50}(x) = 1 - e^{\frac{-x}{50}}$.

4.2 Simulation Results

The Effect of Unfair Raters In order to evaluate the accuracy of the estimation, we compare the trust values calculated by a buyer using his own observations and the reputation values estimated using his Bayesian networks.

As buyers only select reliable buyers as raters, it would be expected that the proportion of unfair raters has little effect on the accuracy of estimation. Figure 2 and figure 3 show the reputation values and trust values ($P_G and P_L$) of seller s_4 for 780 sessions when there are no unfair buyers. It can be seen that reputation values follow trust values closely with only little difference. In figure 2, the average difference between reputation values and trust values is 0.024 and the maximal difference is 0.114 (in sessions 530-540). During sessions 530-540, the trust value decreases suddenly to a very low level that never happened before (never appeared in the cases), so the estimate given by the Bayesian network does not concentrate on one or two most possible subintervals but on several neighboring subintervals according to previous case learning. In figure 3, the average difference and maximal difference are 0.016 and 0.086 respectively. Figure 4 and figure 5 show the comparison results when the proportion of unfair raters is 50% (5 fair buyers) and 80% (only one fair buyer other than b_1) respectively. The average difference and maximal difference are 0.023 and 0.103 in figure 4. We can see that the accuracy of estimation in the situation where more than half of the raters are unfair is very similar to that in the situation without unfair raters. In figure 5, the average difference and maximal difference are 0.033 and 0.185. Although the estimation is not as accurate as that of figure 4 because of the presence of fewer reliable raters, the system still gives a reasonable reference for decision makers.

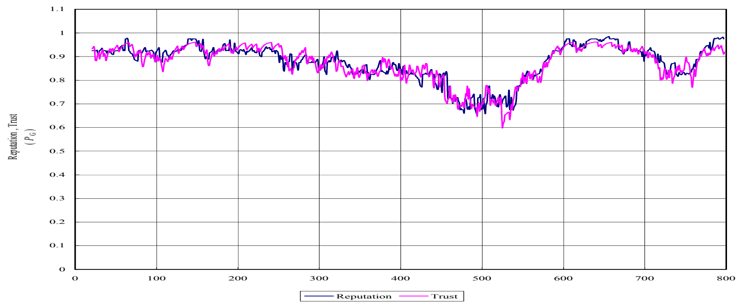

Fig. 2. Reputation Value and Trust Value (P_G) for Seller s_4 by Buyer b_1 without Unfair Buyers

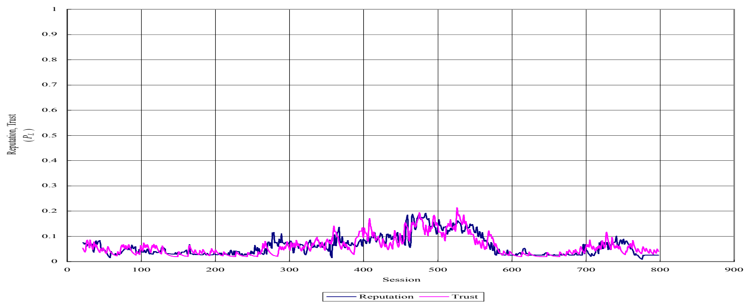

Fig. 3. Reputation Value and Trust Value (P_L) for Seller s_4 by Buyer b_1 without Unfair Buyers

The Effect of Different Rating Criteria As mentioned in section 1, trust is subjective. Each buyer can give different estimate to the same sellers according to their different criteria. Suppose there is a kind of unstable buyer, whose behavior changes over time. Sometimes they give neutral ratings as ordinary buyers, sometimes they give harsher ratings because they require that items should be exactly the same as described by sellers, and they can also be tolerant to giving better ratings. Assume they change their

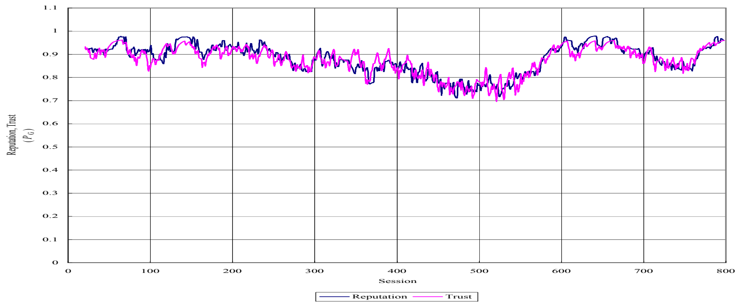

Fig. 4. Reputation Value and Trust Value (P_G) for Seller s_4 by Buyer b_2 with 50% Unfair Buyers

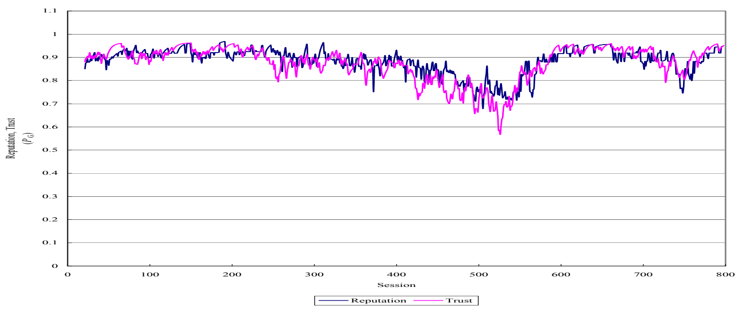

Fig. 5. Reputation Value and Trust Value (P_G) for Seller s_4 by Buyer b_1 with 80% Unfair Buyers

behavior according to the state sequence shown in figure 6. Figure 7 shows his estimation based on similar ratings as his own for seller s4. The estimate is still very accurate although it is obviously different from the real behavioral probability.

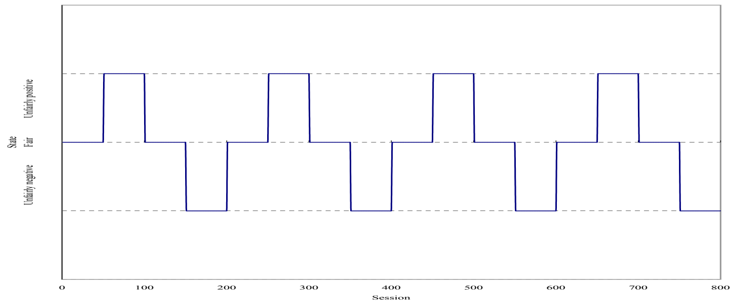

Fig. 6. Sequence of Rating Behavior States of Unstable Buyers

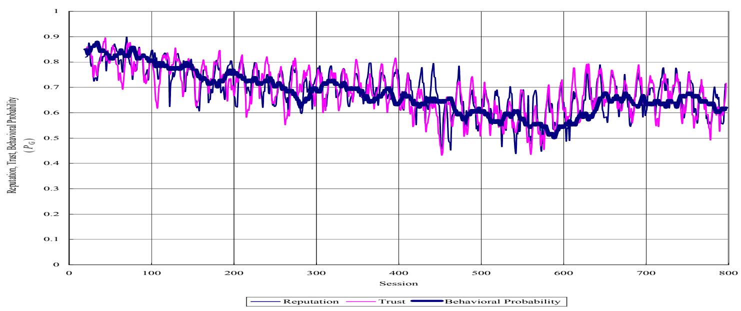

Fig. 7. Reputation Value and Trust Value (P_G) for Seller s_4 by Buyer b_3

5 Conclusions

In our trust management model, we use Bayesian networks for both estimating reputation values of other entities and filtering out unfair raters. Simulations show that the estimation follows the decision makers' subjective, dynamic criteria very well. Our model can be easily extended in two directions. On one hand, we can add a Bayesian network for a new dimension of trust or for a new context. On the other hand, we can combine different dimensions of trust or different contexts to get a general opinion using a Bayesian network with some nodes for representing trust dimensions and some nodes for different contexts. Users can select different components according to specific situations. In fact, different context can be represented by different cases. We plan to add this trust management model to the SECURE framework [1] to provide more flexible security mechanisms for collaboration in uncertain environments.

References

1. V. Cahill, E. Gray, J.-M. Seigneur, C. Jensen, Y. Chen, B. Shand, N. Dimmock, A. Twigg, J. Bacon, C. English, W. Wagealla, S. Terzis, P. Nixon, G. Serugendo, C. Bryce, M. Carbone, K. Krukow, and M. Nielsen. Using Trust for Secure Collaboration in Uncertain Environments. In Pervasive Computing Magazine, volume 2, IEEE Computer Society Press, 2003.
2. L. Mui, M. Mohtashemi, C. Ang, P. Szolovits, and A. Halberdtadt. Ratings in Distributed Systems: A Bayesian Approach. 2002.
3. R. Ismail and A. Josang. The beta reputation systems. In Proceedings of the 15th Bled Conference on Electronic Commerce,Slovenia, 2002.
4. A.Whitby, A. Josang, 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 and Multi Agent Systems, 2004.
5. J. Patel, W.T. Luke Teacy, N. R. Jennings, and M. Luck. A Probabilistic Trust Model for Handling Inaccurate Reputation Sources. In Proceedings of the Third International Conference, iTrust 2005, Paris, France, May 2005.
6. Y. Wang, J. Vassileva. Trust and Reputation Model in Peer-to-Peer Networks. In Proceedings of the Third International Conference on Peer-to-Peer Computing 2003.
7. K. Suzanne Barber, Joonoo Kim. Belief Revision Process Based on Trust: Agents Evaluating Reputation of Information Sources. Trust in Cyber-societies. Lecture Notes in Artificial Intelligence, 2001.
8. C. Dellarocas. Mechanisms for coping with unfair ratings and discriminatory behavior in online reputation reporting systems. In ICIS,2000.
9. S. Ganeriwal, M. B. Srivastava. Reputation-based framework for high integrity sensor networks,?ACM Security for Ad-hoc and Sensor Networks, 2004.
10. Buchegger, S., Boudec, J.Y.L. A robust reputation system for p2p and mobile ad-hoc networks. In: Proceedings of the Second Workshop on the Economics of Peer-to-Peer Systems, 2004.
11. http://www.norsys.com/.
12. http://www.econ.iastate.edu/tesfatsi/tnghome.htm.