Improving Trust and Reputation Modeling in E-Commerce Using Agent Lifetime and Transaction Count

Catherine Cormier and Thomas Tran

School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada K1N 6N5

Abstract. Effective and reliable trust and reputation modeling systems are central to the success of decentralized e-commerce systems where autonomous agents are relied upon to conduct commercial transactions. However, the subjective and social-based qualities that are inherent to trust and reputation introduce many complexities into the development of a reliable model. Existing research has successfully demonstrated how trust systems can be decentralized and has illustrated the importance of sharing trust information, or rather, modeling reputation. Still, few models have provided a solution for developing an initial set of advisors from whom to solicit reputation rankings, or have taken into account all of the social criteria used to determine trustworthiness. To meet these objectives, we propose the use of two new parameters in trust and reputation modeling: agent lifetime and total transaction count. We describe a model that employs these parameters to calculate an agent's seniority, then apply this information when selecting agents for soliciting and ranking reputation information. Experiments using this model are described. The results are then presented and discussed to evaluate the effect of using these parameters in reputation modeling. We also discuss the value of our particular model in contrast with related work and conclude with directions for future research.

Keywords: Trust, Reputation, E-commerce, Multi-Agent Systems, Agent Lifetime, and Transaction Count

1. Introduction

In any multi-agent system, autonomous agents must be able to determine which other agents it trusts for any given interaction. "Trust is central to all transactions" as stated by Dasgupta [4]. Since e-commerce systems exist to facilitate transactions, it follows that trust is key to all e-commerce environments.

In order to have a successful e-commerce system, it is imperative that reliable and effective trust models be in place. The critical challenge in developing a sound trust model is that trust is subjective [6]. As well, the open and distributed nature of multi-agent e-commerce systems where agents act autonomously, based on their own interests, values and beliefs makes designing a robust trust model difficult. Given the significance and complexity of the problem, a number of researchers have tried to tackle this problem and have proposed systems that assess trust using different methods and parameters [3, 7, 8, 10, 11, 12, 13].

Since "trust is based on reputation" [4] many of these approaches consider agent reputation a key factor in the trust model. Reputation is based on past behavior observed and reported by (e.g., via word-of-mouth) other agents and is typically communicated between agents using a reputation rating. Reputation modeling is the design of approaches to (i) generate, (ii) discover and (iii) aggregate rating information [13]. This paper aims to recommend new parameters that can be used to enhance reputation modeling.

Several successful online marketplaces such as eBay and the Apple App Store offer centralized systems for reporting rating information [1, 2]. At the eBay site, other users (buyers and sellers) are rated, while at the Apple App Store, software applications available for purchase are rated. In both of these systems, ratings are generated by users once they have engaged in a transaction. The ratings are published publicly at the site for discovery by other users. Rating aggregation is then left to the individual user; each may interpret the ratings and other information about the user or application and make a trust decision in their own manner.

In these online marketplaces, the fact that a user must complete a transaction prior to submitting a rating suggests that a user's opinion is valued only if it is based in experience. Moreover, the influence that a user's opinion may have increases with the number of transactions in which the user participates (because the user's opinion may appear up to once per transaction). From these observations, we assert that the value of a reputation rating is related to the number of transactions in which an agent has participated.

In addition, eBay presents a summary for each user, which includes a "member since" date as well as a total number of ratings received. As described above, the number of ratings must be less than or equal to the number of transactions in which the user has participated, and is therefore closely related to the number of transactions in which the subject has participated. Since both the membership date and number of ratings are readily available in the description of all sellers, as well as the users who have rated the seller we suggest that this information must influence the buyer's decision to trust the seller and its aggregation of ratings.

In our research, we have sought to investigate the effectiveness of applying these principles observed in successful centralized e-commerce systems in decentralized multi-agent e-commerce systems. We propose that reputation modeling in distributed e-commerce systems can be improved by considering the amount of time an agent has been part of a system, or *agent lifetime*, and the number of transactions in which an agent has participated, *or transaction count*, when discovering and aggregating reputation ratings.

We present our research and findings in this paper as follows: In section 2 we discuss related work. We then formally present the proposed parameters and approach in section 3. In section 4 we describe our experimental technique and results. A discussion is given in section 5 and we conclude with future work in section 6.

2. Related Work

Due to the importance of having reliable and effective trust and reputation systems in e-commerce marketplaces, trust and reputation modeling has become an active area of research. As a result, over the past several years a variety of proposed approaches have emerged.

Many of these models propose using a *direct trust* rating in conjunction with *reputation ratings* obtained from a set of advisor agents [1, 3, 7, 8, 10, 11, 12, 13]. Generally, *direct trust* is the trust that one agent has in another based on its past experiences, for example, the trust that agent *a* has in agent *b*. *Reputation ratings* represent the reputation information supplied by an advisor agent, for example, the rating that agent *c* reports to agent *a* about agent *b*.

Some of the models also consider additional facets of trust. For example, REGRET combines the individual, social and ontological dimensions of trust [12]. In this case, not only direct trust and reputation ratings are considered, but also information about the agent's group and ratings of different aspects of an agent's performance are factored. In FIRE, direct trust ("interaction trust"), role-based trust (determined from pre-set rules about ratings for different relationships), reputation ratings ("witness reputation") and certified reputations (provided by the agent itself but signed by a recommender) are combined to determine an aggregated trust value [8].

A few of the models are experience or evidenced-based, in particular REGRET [12], FIRE [8], CertainTrust [11], as well as Hang et al.'s evidence-based model [7] and Reece et al.'s experienced-based model [10]. In these approaches, the number of experiences or pieces of evidence on which a rating is based is considered in the computation of the aggregated rating. This is important, as claimed by Hang et al., because when a rating estimated by a single value, such as a probability, it is impossible to know whether that rating is based on very few or many experiences [7]. And, as Ries points out, "the certainty of an opinion increases with the number of evidence on which that opinion is based" [11]. Thus, these systems recognize and support the principle that advice based on greater experience is of more value. However, these systems consider only the number of interactions between the agent rating and the agent being rated; they do not consider an agent's experience in the system overall.

Finally few systems specifically define the process for initializing the list of advisor agents. In fact, Abdul and Hailes indicate that their model is not suitable for bootstrapping the advisor list [1]. Others, such as [7], [8], [10] and [13] simply use a set of neighbors as their advisor list.

As with others who have developed experience and evidence-based models, we propose that an opinion that is based on a high level of experience is generally more valuable than an opinion based on less experience. In contrast to these approaches, which consider strictly the number of experiences directly between two agents, however, we suggest considering an agent's overall experience level. This experience level, based on the two metrics agent lifetime and transaction count, can be used as a determining factor for trusting the advice an agent provides. We also use these parameters to develop an initial advisor agent list, which is a problem generally left unaddressed by the above approaches.

3. Proposed Approach

For the purposes of improving the initial selection of advisor agents and to accurately weight the advice received by advisor agents, we propose the use of agent lifetime and transaction count. The use of these parameters is derived from the principle that advice obtained from the most experienced members of a group is generally the most valued. It is proposed that by using these factors to select initial advisors, novice agents can benefit from the experience of more senior agents, thereby reducing risk by achieving desirable results sooner.

3.1 Agent Lifetime

Agent lifetime is the amount of time that an agent has been a part of the multi-agent system, or, more simply, the agent's age within the system. A characteristic of open distributed systems is the agent's ability to enter and leave the system freely. As Ramchurn et al. point out, this characteristic of the system can be leveraged by malicious agents who leave and reenter the system in order to change their identities and escape their past behavior [9]. Conversely, agents who have established a positive reputation over time would be better suited to stay within the system. As a result, an agent's lifetime can be used as an indicator of trustworthiness.

Agent lifetime is calculated at any given time t using a *timestamp* assigned to the agent when it enters the multi-agent system. For any agent a, with assigned timestamp T_a, the agent lifetime is given by:

$$l_a(t) = 1 - \frac{T_a}{t} \tag{1}$$

Using this formula, agent lifetime is normalized so that $l_a(t)$ is always in the range [0,1]. Agents that have been in the system for a long time will have a lifetime approaching 1, while agents that have recently entered the system will have a lifetime close to 0.

3.2 Transaction Count

An agent's *transaction count* is the total number of transactions in which the agent has participated over the course of its lifetime. This is a measure of the agent's activity level within the system. It is presumed that an agent that has participated in a large number of transactions is more experienced than an agent that has participated in fewer transactions. And, by extension, the advice provided by the agent with a higher transaction count is more valuable. Exceptions may exist where, for example, malicious agents engage in a large number of low-value transactions in order to falsely inflate their transaction count and to use ballot-stuffing techniques to falsely inflate or deflate another agent's reputation. Agent lifetime and transaction count could be used to detect such malicious behavior, for example by determining that an agent with a short lifetime has engaged in a suspiciously high number of transactions. This application of agent lifetime and transaction count is beyond the scope of the research presented here; therefore, we defer it to our future work (Section 6).

The transaction count for any agent a is denoted by n_a and is automatically incremented by the system whenever a transaction occurs between agent a and any other agent.

3.3 Seniority

In the proposed approach, an agent who has been an active participant of the multi-agent system for a relatively long time, as compared with the other agents in the system, is considered a senior agent within the system. By identifying agents who are the most senior in the system, agents entering the system will be able to establish an initial set of advisors whose recommendations are based on as much experience as possible. Further, agents' seniority can be used to weight the advice received from advisor agents so that advice based on more experience more heavily impacts the overall reputation calculated.

The *seniority* of agent a at any time t is given by the product of its lifetime and transaction count:

$$s_a(t) = n_a \cdot l_a(t) \tag{2}$$

An agent a is said to be more senior than agent b if $s_a(t) > s_b(t)$. Any agent a that is new to the system or has never participated in a transaction is considered a novice agent and has $s_a(t) = 0$.

3.4 Building an Initial Advisor List

By adding agent lifetime and transaction count to a reputation model, novice agents are able to build an informed initial advisor list and therefore make good selections of agents with whom to engage in transactions, even with no or limited experience of their own. In order to find N advisors upon entering the system, the agent collects a list C of all candidate advisors that it can discover—this could be a list of neighbors, referred advisors or agents discovered using some other technique. It then calculates the seniority $s_c(t)$ for each candidate advisor c in the set C. Finally, the agent selects the N candidate advisors with the highest $s_c(t)$ values as its advisor list, denoted as A.

As time progresses, agent a may wish to refresh its list of advisor agents by replacing one or more of the advisors in its list. At that time, the agent may use the same or a similar technique to select a new set of advisors by using seniority values calculated at that time.

3.5 Weighting Advice

Once an agent has established its list of advisors and is faced with the decision to participate in a transaction with another agent, it will solicit advice in the form of a reputation ranking from each of its advisors. Since advice provided by advisors who are more experienced is deemed more valuable than advice given by those who are less experienced, the seniority of the advisor agent can be used to weight the reputation rating received from each advisor.

In order to decide whether or not to trust agent b, agent a solicits advice about agent b from each advisor agent adv_i in its advisor list A. When each advisor agent receives the request for advice, it can respond by sending its reputation rating r^b_{adv} to agent a. Agent a then computes a total reputation rating for agent b, $r_b(t)$, using each advisor's current seniority to weight the advice received, as follows:

$$r_b(t) = \frac{\sum_{adv_i \in A} r^b_{adv_i} \cdot s_{adv_i}(t)}{\sum_{adv_i \in A} s_{adv_i}(t)} \tag{3}$$

This reputation value can then be combined with the direct trust rating that agent a has for b (based on its own previous experiences with b, if any). This can be accomplished using a simple technique such as computing the average of the reputation value and direct trust rating, or more elaborately following the techniques such as those presented by other researchers [1, 3, 7, 8, 10, 11, 12, 13]. However, in order to specifically examine the effect of using this approach, this aggregated reputation value $r_b(t)$ is used alone for selecting agents in the experiments described in Section 4.

4. Experimental Results

We implemented a software simulation to examine whether the use of agent lifetime and transaction count as described in Section 3 would: (a) enable agents who are new to the system to make effective decisions immediately; (b) enable agents to make more effective decisions overall. To determine the effects of these parameters independently of other factors, we employed a very simple reputation modeling system as presented in Section 4.1.

Furthermore, for analysis purposes, both a base case model and a test case model were implemented, as described in detail below. The differences between the two models are kept as minimal as possible, thereby further isolating the effect of the agent lifetime and transaction count parameters.

4.1 Description

The test software simulates an e-commerce marketplace where buyer agents may purchase from any seller agents. To simplify the experiment, it is assumed that all selling and buying occurs in the same context. That is, all sellers are offering competitive products and all buyers are in the market to purchase similar products. Furthermore, all buying agents can act as advisor agents to other buying agents.

Advisor agents report their reputation rating, a value in the continuous range [-1, +1], for a given seller at the request of a buying agent. As an advisor agent, the agent may be either: honest, dishonest_high, dishonest_low or dishonest_erratic. An honest agent truthfully reports the average of it internal ratings for that seller, where each internal rating is simply the average utility it has gained from a transaction with the seller, normalized to be in the [-1, +1] range. Dishonest agents constantly report either a high, low or erratic value, depending on their advisor type. Specifically, the reputation rating that each advisor type provides is as follows:

Table 1. Reputation ratings returned by different advisor agent types.

Advisor Type	Reputation Rating Returned
Honest	Average of internal ratings for seller in question
Dishonest_high	Random value in the range [0, +1.0]
Dishonest_low	Random value in the range [-1.0, 0]
Dishonest_erratic	Random value in the range [-1.0, +1.0]

When a new buying agent is created and enters the marketplace, it initializes its list of advisor agents using either the technique described in the base case model or the test case model, as described in sections 4.2 and 4.3, respectively.

On each time step, every buying agent in the marketplace is given the opportunity to buy. To simulate agents with a variety of transaction counts, each buying agent decides whether or not to buy based on its buying activity level, which can be constant, high, medium, low or very low. Buyers with a constant activity level must buy on every time step, while agents with a very low activity only by on every 11^{th} time step. Other buyers determine how many time steps to wait between purchases by randomly selecting a wait period over a given range, as specified in Table 2.

Table 2. Time steps between purchases for different buyer activity levels.

Buyer Activity Level	Time steps between purchases
Constant	0
High	Random value in the range [0, 2]
Medium	Random value in the range [3, 6]
Low	Random value in the range [7, 9]
Very Low	10

Once a buying agent decides to buy, for each candidate seller s, the buying agent asks all of its advisors for their advice about s. If the buying agent does not receive any advice about s, it adds s to its list of *unrated sellers*. If it does receive reputation ratings for s, the buying aggregates all of the ratings received, calculating r_s by following either the base case model (Section 4.2) or the test case model (Section 4.3). It then compares this aggregated reputation rating against its personal *trust threshold*, which is the minimum reputation rating the selling agent must have to be selected. This value represents the buyer's preference and could therefore vary from one agent to another in practice. However, for this purpose of this simulation, every buying agent has the same trust threshold, as given in Table 4. If the aggregated reputation rating for the most highly rated seller is greater than the buyer's trust threshold, then the buyer proceeds with that seller. Otherwise, the agent randomly selects from any unrated sellers or, if no unrated sellers are available, selects the highest rated seller (even though its rating was below the *trust threshold*).

When a buyer chooses to buy from a particular seller, the buyer receives a utility value which is a discrete value in the range [0, 10], where 0 denotes a very bad outcome and 10 denotes a very good outcome. Sellers may be good, average, bad or erratic, which means that they will randomly return a value in the corresponding range, as specified in Table 3.

Table 3. Utility range for different types of selling agents.

Seller Type	Utility Range
Good	[7, 10]
Average	[4, 6]
Bad	[0, 3]
Erratic	[0, 10]

After the transaction is complete, the buying agent converts the utility received to a trust rating in the continuous range [-1,1] and stores it in its internal rating table for the corresponding seller.

For each run, the simulated marketplace is initialized with a set number of buyers, *N_BUYERS* and sellers *N_SELLERS*. In order to simulate agents with different lifetimes and transaction counts, a new group of buyers is added to the marketplace after each interval I of time steps has passed. The simulator continues to add groups of buyers to the marketplace every I time steps. These groups are each assigned a number so that their behavior may be analyzed as a group. As well, each buying agent refreshes its advisor list by creating a new list every Jth time step.

4.2 Base Case Model

The base case model is a simple approach that is provided to compare and evaluate the test case model (i.e., the proposed model that employs the use of the agent lifetime and transaction count parameters). The base case model and test case model differ in two respects: (i) how buying agents select their list of advisor agents; (ii) how buying agents aggregate the advice received from advisor agents.

Selection of Advisor Agents: In the base case model, buying agents generate their list of advisor agents by randomly selecting *N_ADVISORS* advisor agents from all of the possible advisor agents (i.e., all other buying agents) in the marketplace.

Calculating Seller Reputation: For the base case model, buying agents compute the average of all of the ratings received from advisor agents. Therefore, if buying agent b is evaluating selling agent s, it solicits advice from all agents in its advisor agent list then computes:

$$r_s = \frac{\sum_{r_i \in R} r_i}{|R|} \tag{4}$$

where r_s is the consolidated reputation rating and R is the set of reputation ratings obtained from the buying agent's advisors.

4.3 Test Case Model

The test case model employs the proposed parameters agent lifetime and transaction count, following the proposed approach presented in Section 3. It differs from the base case only in how buying agents select advisor agents and in how it uses ratings provided by advisor agents to calculate seller reputation.

Selection of Advisor Agents: In the test case model, when a buying agent needs to select advisor agents, it first computes the seniority of all candidate advisor agents (i.e., all other buying agents in the marketplace) following equation (2). It then selects the *N_ADVISORS* agents that have the highest seniority values as its list of advisors.

Calculating Seller Reputation: To aggregate the reputation ratings received from its advisor agents, each buying agent in the test case model uses the seniority weighting formula given in equation (3).

4.4 Results

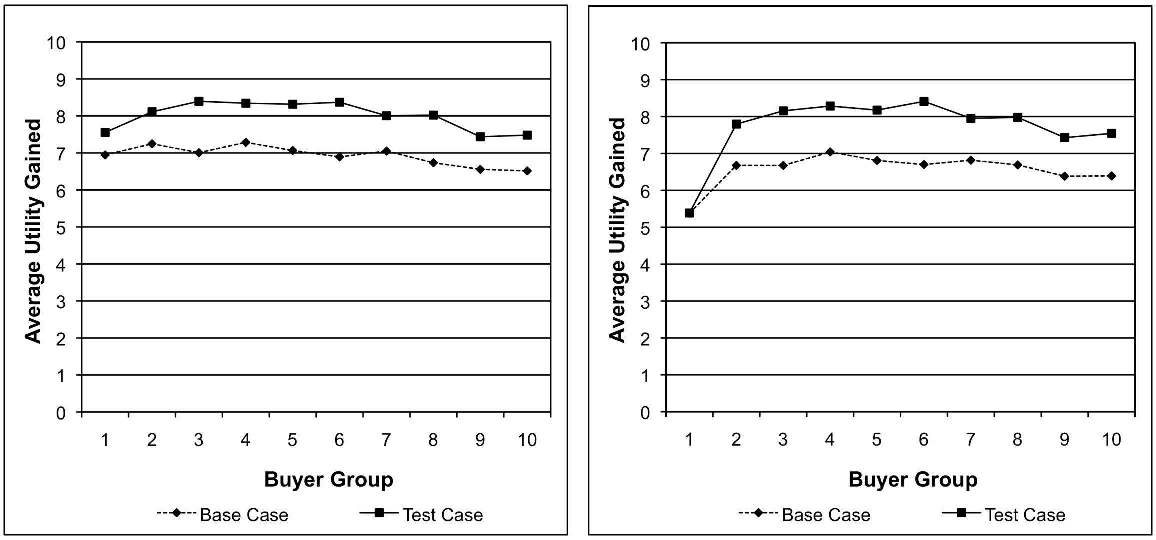
The simulation marketplace was run ten times using each the base case model and the test case model. For each run of the simulation, the marketplace parameters were set as follows:

Table 4. Experimental parameters used for simulation marketplace.

Parameter	Value	Parameter	Value
Buyers (initial)	50	New buyer group size	25
Sellers	100	Interval to add buyers (*I*)	100
Time steps	1000		
		Buyer Activity Constant (%)	5
Bad Sellers (%)	15	Buyer Activity High (%)	10
Good Sellers (%)	15	Buyer Activity Medium (%)	70
Average Sellers (%)	60	Buyer Activity Low (%)	10
Erratic Sellers (%)	10	Buyer Activity Very Low (%)	5
Honest Advisors (%)	70	Trust Threshold	0.75
Dishonest_high Advisors (%)	10	Advisor List Size	10
Dishonest_low Advisors (%)	10	Advisor Refresh Cycle (*J*)	100
Dishonest_erratic Advisors (%)	10		

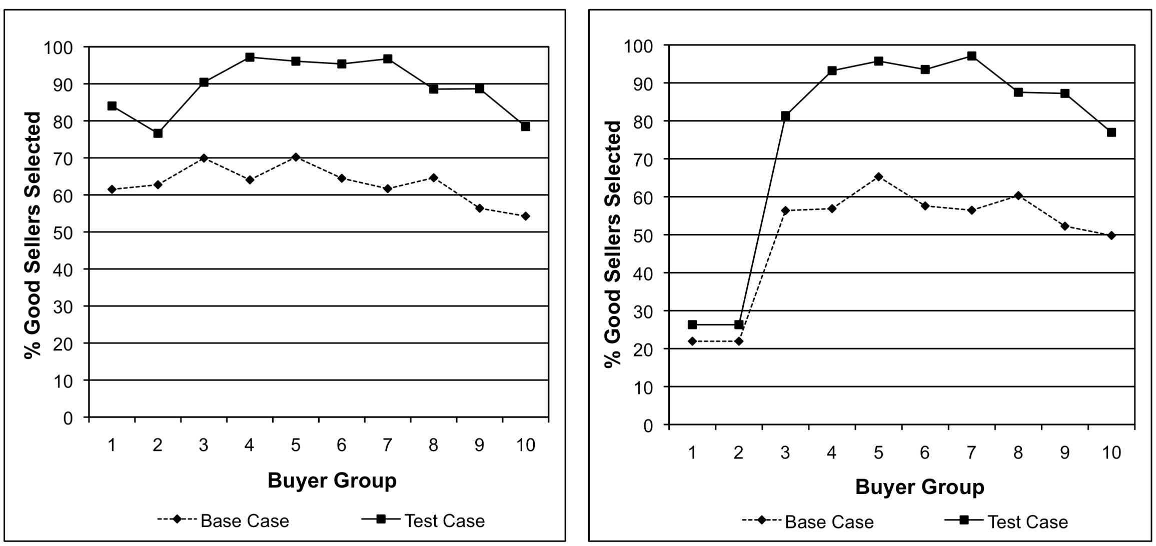
By analyzing the average utility gained over the entire simulation run for each group of buyers (where group 1 is the initial buyer set and groups 2 through 10 are the sets of added buyers), we see that the test case model produces a higher average utility for all groups (Figure 1(a)). Furthermore, if we consider only the first ten time steps of the simulation, we see that the buying agents in groups 2 through 10 gain significantly more utility in the test case model than in the base case model (Figure 1(b)). This indicates that agents that are new to the system (novice agents) are immediately more effective when following the test case model over the base case model.

In the first ten time steps, however, group one performs almost identically in both models. This is due to the fact that all agents in the marketplace at that time are novice, and therefore there aren't any senior agents from whom to solicit advice. As a result, the agents in both models behave in the same manner for the first few times steps.



(a) *(b)*

Fig. 1. Average utility gained by each buying group: (a) for all time steps of the simulation; (b) for the first 10 time steps that the buying agent group exists in the marketplace.



(a) *(b)*

Fig. 2. Average percent of sellers selected by each buying group that were good sellers after: (a) all time steps of the simulation; (b) first ten time steps for which the buyer group exists in the marketplace.

By examining the choice of sellers by the buying agents, we see that the agents in the test case model were consistently significantly better at choosing good sellers than the buying agents in the base case model (Figure 2(a)). As with the average utility gained, by examining the behavior in the first ten time steps of each group's existence, we see that buying agents are able to make considerably better choices in their first few purchases by using the test case model over the base case model (Figure 2(b)). Again, the exceptions are with the first groups which yield similar results in the first ten time steps regardless of the model used since the system does not at that time contain any experienced agents to use as advisors.

5. Discussion

The experimental results that we have presented show that the proposed parameters, agent lifetime and transaction count, can be used as part of a reputation model to improve results obtained in an agent's first few transactions, as well as over the lifetime of the agent. Overall, these parameters employed in the proposed manner led to better selection of seller agents, which in turn yielded more successful outcomes than in the base (random) case.

However, closer examination of the results reveals that while in most simulation runs, the test case model produced nearly perfect results for the selection of good sellers, in a small number of simulation runs, the results were poor. The latter runs were characterized by having an unusually high percentage of advisors who were both dishonest and had high seniority (i.e., had a long lifetime and a high transaction count) highly active. Therefore, the experimental results show that the test case model is vulnerable to scenarios where agents who have been in the system for a relatively long time conspire together to bloat their transaction counts (e.g., by engaging in a large number of transactions amongst themselves) then provide dishonest reputation ratings. The impact of this vulnerability can be reduced by adopting some of the techniques presented by others, such as maintaining a direct trust value for each advisor agent using a technique such as one of those described in [1, 3, 7].

We have shown that agent lifetime and transaction count can be used to improve reputation modeling, but that their use alone is not sufficient to cover all scenarios.

6. Conclusions and Future Work

In this paper, we have presented two new parameters that can be used to improve reputation modeling systems: agent lifetime and transaction count. Furthermore, through experimental results we have demonstrated that the use of these parameters in a simple reputation modeling system can (i) enable novice agents to construct an effective initial advisor list, thus attaining better results sooner, and (ii) enable agents to make improved trust decisions overall. We believe that the results presented here indicate that these parameters can be used to develop improved reputation models based on approaches presented in other research or entirely new approaches.

In future work, we intend to introduce these parameters into a more complex reputation modeling system to verify that they can be used to improve other models and to overcome some of the challenges revealed by the results analysis. This more elaborate model will be validated through testing in simulation against models presented by other researchers.

In addition, we intend to investigate how agent lifetime and transaction count can be used in computing the direct trust that one agent has in another. This approach would have two facets: first, the agent lifetime and transaction count of the agent being rated can be factored in the computation of the trust rating; secondly, we could introduce lifetime and transaction count as attributes that describe the relationship between the agent being rated and the agent performing the rating. In this second scenario, the transaction count for the relationship would be similar to the values used in the experience-based approaches described in [7, 8, 10, 11, 12]; however, the introduction of the relationship lifetime would be entirely new.

Finally, we would like to investigate how agent lifetime and transaction count can be utilized to detect and avoid malicious behavior. For example, given the agent lifetime and transaction count for any agent, it should be possible to detect agents who leave the system to escape a bad reputation, then reenter and engage in a large number of transactions in a short period of time to falsely inflate their reputation.

7. References

1. Abdul-Rahman, A., Hailes, S.: Supporting Trust in Virtual Communities. In Proceedings of the 33rd Hawaii international Conference on System Sciences-Volume 6 - Volume 6 (January 04 - 07, 2000). HICSS. IEEE Computer Society, Washington, DC, 6007. (2000)

2. Apple iPhone App Store, http://www.apple.com/iphone/appstore/

3. Cohen R, Regan K., Tran T: Sharing Models of Sellers amongst Buying Agents in Electronic Marketplaces. In Proceedings of the 10th International Conference on User Modeling—Workshop on Decentralized, Agent Based and Social Approaches to User Modeling. (2005).

4. Dasgupta, P. : 'Trust as a Commodity', in Gambetta, Diego (ed.) Trust: Making and Breaking Cooperative Relations, electronic edition, Department of Sociology, University of Oxford, chapter 4, pp. 49-72. (2000)

5. Ebay, http://www.ebay.com/

6. Gambetta, D.: 'Can We Trust Trust?', in Gambetta, Diego (ed.) Trust: Making and Breaking Cooperative Relations, electronic edition, Department of Sociology, University of Oxford, chapter 13, pp. 213-237. (2000)

7. Hang, C., Wang, Y., Singh, M. P.: An adaptive probabilistic trust model and its evaluation. In Proceedings of the 7th international Joint Conference on Autonomous Agents and Multiagent Systems - Volume 3 (Estoril, Portugal, May 12 - 16, 2008). International Conference on Autonomous Agents. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1485-1488. (2008)

8. Huynh T. D., Jennings N. R., Shadbolt N.: Developing an integrated trust and reputation model for open multi-agent systems. Proceedings of the 7th International Workshop on Trust in Agent Societies, New York, USA, pages 65-74. (2004)

9. Ramchurn, S. D., Huynh, D., Jennings, N. R.: Trust in multi-agent systems. Knowl. Eng. Rev. 19, 1 (Mar. 2004), 1-25. (2004)

10. Reece S., Rogers A., Roberts S., Jennings N. R.: Rumors and Reputation: Evaluating Multi-Dimensional Trust within a Decentralized Reputation System. In Proceedings of the Sixth Intl. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS-07), pages 1063-1070. (2007)

11. Ries, S.: Certain trust: a trust model for users and agents. Proceedings of the 2007 ACM Symposium on Applied Computing (Seoul, Korea, March 11 - 15, 2007). SAC '07. ACM, New York, NY, pp. 1599-1604. (2007)

12. Sabater J., Sierra C. REGRET: Reputation in gregarious societies. Proceedings of the Fifth International Conference on Autonomous Agents, Montreal, Canada, pages 194-195, ACM Press, 2001.

13. Yu, B., Sycara, K., Singh, M.:. Developing Trust in Large-Scale Peer-to-Peer Systems. In Proceedings of First IEEE Symposium on Multi-Agent Security and Survivability (MASS-04). (2004)