Eliciting Truthful Feedback for Binary Reputation Mechanisms

Radu Jurca and Boi Faltings
Artificial Intelligence Laboratory (LIA),
Computer Science Department, Swiss Federal Institute of Technology (EPFL)
CH-1015 Ecublens, Switzerland
{radu.jurca, boi.faltings}@epfl.ch
`http://liawww.epfl.ch/`

Abstract

Reputation mechanisms offer an efficient way of building the necessary level of trust in electronic markets. Feedback about an agent's past behavior can be aggregated into a measure of reputation, and used by other agents for taking trust decisions. Unfortunately, true feedback cannot be automatically assumed. In the absence of Trusted Third Parties, the mechanism has to make it rational for agents to truthfully share reputation information. In this paper we describe two mechanisms that can be used in decentralized environments for eliciting true feedback. The mechanisms are accompanied by examples inspired by real scenarios.

1. Introduction

The availability of ubiquitous communication through the Internet is driving the migration of business transactions from direct contact between people to electronically mediated interactions. People interact electronically either through human-computer interfaces or even through programs representing humans, so-called agents. In either case, no physical interactions among entities occur and the systems are much more susceptible to fraud and deception.

Humans exhibit in social interactions behavioral characteristics that are favoring honest behavior, even if they are irrational from the game-theoretic viewpoint. Having purely electronic interactions lowers the social barriers for cheating for humans. From a game-theoretic perspective, many business transactions have the characteristic of the Prisoner's Dilemma game, where cooperation is the highly desirable outcome for the players, but the dominant strategy (the Nash equilibrium) is to cheat.

A standard approach in traditional business to avoid cheating is to use trusted third parties (TTP) that oversee the transactions and rule out or at least punish cheating. However, TTP's introduce a form of centralization that can hamper scalability, and increase transaction cost in unacceptable ways. Moreover, network communities often have a strong desire of being independent of any authorities (as illustrated by the successful P2P systems) and thus do not accept any outside authority.

Reputation mechanisms offer a novel and efficient way of ensuring the necessary level of trust which is essential to the functioning of any market. They are based on the observation that agent strategies change when we consider that interactions are repeated: the other party will remember past cheating, and changes its terms of business accordingly in the future. In this case, the expected future gains due to future transactions in which the agent has a higher reputation can offset the loss incurred by not cheating in the present transaction. This effect can be amplified considerably if such reputation information is shared among a large population and thus multiplies the expected future gains made accessible by honest behavior.

Existing reputation mechanisms enjoy huge success. Systems such as eBay[1] or Amazon[2] implement successful reputation mechanisms which are partly credited for the businesses' success. Studies show that buyers seriously take into account the reputation of the seller when placing their bids in online auctions [7] and that despite the incentive to free ride, feedback is provided in more than half of the transactions on eBay [15].

However, obtaining true feedback from the agents involved in the transaction is not a trivial problem. Reputation reports can be distorted in order to serve the selfish interests of the reporter, and therefore, truthful feedback is obtained only by making the reputation mechanism incentive-compatible, i.e. making it in the best interest of the agents to truthfully share reputation information. This is the problem we are addressing in this paper. Section 2 presents related work, Section 3 defines a binary reputation mecha-

[1] www.ebay.com
[2] www.amazon.com

nism and describes its components. In Section 4 we present two solutions for making binary reputation mechanisms incentive compatible, and Section 5 concludes our work.

2. Related Work

The notion of trust is used to refer to a subjective decision making process that takes into consideration a diversity of factors. The *Social Auditor Model* [10] is one of the existing models that explain how humans take trust decisions by using a set of rules. One input information that is often used in a trust decision making process is the *reputation* of the partner. Reputation can be regarded as a unitary appreciation of the personal attributes of the trustee: competence, benevolence, integrity and predictability. In [13] Mui et al. present an extensive classification of reputation by the means of collecting it.

Formally, reputation signals the preference of an agent to act according to a specific *type*[3]. For example, an agent that has a reputation for cooperating, signals the fact that she prefers to cooperate (i.e. play according to the *cooperative type*). Building reputation affects the outcome of a repeated game, generating the *reputation effect*. At some point, the partners of an agent A who builds reputation for a certain type T, will *trust* that A is always going to behave according to the type T, and will switch to a strategy that is a best response against A's type T. It might be possible that this new equilibrium makes A better off, making it rational for A to build the reputation.

The only equilibrium outcome of a finitely repeated Prisoners' Dilemma game is when both agents defect ([14], p. 135). Kreps et al. [11] show, however, that in this game, agents could obtain the efficient outcome when they build a reputation for playing according to the *tit-for-tat* strategy [1]. This observation was the basis of theoretic research on reputation.

[2, 3] describe computational trust mechanisms based on direct interaction-derived reputation. Agents learn to trust their partners, which increases the global efficiency of the market. However, the time needed to build the reputation information prohibits the use of this kind of mechanisms in a large scale online market.

A number of reputation mechanisms also take into consideration indirect reputation information, i.e. information reported by peers. [16, 17] use social networks in order to obtain the reputation of an unknown agent. Agents ask their friends, who in turn can ask their friends about the trustworthiness of an unknown agent. Recommendations are afterwards aggregated into a single measure of the agent's reputation. This class of mechanisms, however intuitive, does not provide any rational participation incentives for the agents. Moreover, there is little protection against untruthful reporting, and no guarantee that the mechanism cannot be manipulated by a malicious provider in order to obtain higher payoffs.

Dellarocas [6] presents an efficient binary reputation mechanism that encourages a cooperative equilibrium in an environment of purely opportunistic buyers and sellers. The mechanism is centralized, it works for single-value transactions, and is robust (within certain limits) against mistakes made by reporters.

One major challenge associated with designing reputation mechanisms is to ensure that truthful reports are gathered about the actual outcome of the transaction. We start from the assumption that the outcome of the transaction (i.e. the agent has cooperated or not) is only known to the parties involved. Any reputation mechanism will therefore have information that is distorted by the strategic interests of the reporters.

Most real situations do not make it rational for an agent to report the truth. The private information of a buyer for example, about the trustworthiness of a seller is often regarded as an asset which should not be freely shared. Paying for the buyer's reputation report could overcome this inconvenient, however, no guarantee can be offered that the information provided is true. For example, a true positive report might create inconveniences for the reporting buyer because of decreased future availability of that particular seller. Moreover, in a competitive environment, a false negative report about a seller slightly increases the buyer's own reputation with regards to the other agents.

The problem of incentive compatibility can be addressed by paying for a reputation report, such that the payment is conditioned on the correlation with future reports (assumed to be true) about the same seller. [12] and [8] describe such schemes that make truth revelation a Nash equilibrium. These schemes, however, require certain constraints on the behavior of the sellers and on the beliefs of the reporting buyers: i.e. the signals observed by the buyers about the seller's behavior must be independently identically distributed.

In the same group of work that addresses the property of incentive compatibility, we mention [4] and [5]. [4] considers exchanges of goods for money and proves that a market in which agents are trusted to the degree they deserve to be trusted is equally efficient as a market with complete trustworthiness. By scaling the amount of the traded product, the authors prove that it is possible to make it rational for sellers to truthfully declare their trustworthiness. Truthful declaration of one's trustworthiness eliminates the need of reputation mechanisms and significantly reduces the cost of trust management.

For e-Bay-like auctions, the Goodwill Hunting mecha-

[3] a type designates a preference relation on the set of possible outcomes of a game and implicitly on the set of action profiles in a game [14]

nism [5] provides a way in which the sellers can be made indifferent between lying or truthfully declaring the quality of the good offered for sale. Momentary gains or losses obtained from misrepresenting the good's quality are later compensated by the mechanism which has the power to modify the announcement of the seller.

Finally, [9] takes a different approach and achieves in equilibrium truthful reporting by comparing the two reports coming from the buyer and the seller involved in the same transaction. The details of this mechanism will be explained in Section 4.2.

3. Binary Reputation Mechanisms

We look at Reputation Mechanisms (RM) in a Multi-agent System in which a large number of agents interact pair-wise doing some business (the typical example is the exchange of a service or good for payment). In the absence of TTP's most of the interactions have an inefficient equilibrium: a rational seller will not deliver the service after he has received the payment, and anticipating this, the buyer will not send the payment at all. Thus, the exchange does not happen. If, however, the agents trust each other, the inefficient equilibrium can be abandoned in favor of the cooperative one.

Trust is a symmetric relation. An efficient transaction between agents A and B requires both that A trusts B and that B trusts A. In many practical situations, however, the symmetry can be broken by choosing an interaction protocol that makes one of the parties completely trustworthy. In existing e-commerce systems, the buyer is required to pay first, and then wait for the seller to ship the good. The buyer thus becomes completely trustworthy, and an efficient transaction requires only that the buyer trusts the seller.

Because of the above explained asymmetry, agents will assume roles. The *trusting agent* (or the *trustor*, usually referred as "she", the buyer, in our case) is the completely trustworthy agent who needs to trust the *trusted agent* (or the *trustee*, usually referred as "he"). One physical agent can assume both roles in different interactions, however, for trust management, we are only interested in the agent's behavior when playing the role of the *trusted agent*.

Trust can easily be based on the reputation of agents (i.e. information about the agent's past behavior). We use a model in which after every interaction between a trusting and a trusted agents, a RM records a feedback report about the behavior of the trusted agent in that transaction. The RM aggregates feedback into meaningful reputation information that is made available to the agents. Even if we use the expression "one" RM, please note that the RM can very well be decentralized and implemented in a distributed way by the same agents that use it.

The goal of the reputation mechanism is (1) to protect the trusting agent against cheating from the trusted agent, and (2) to inflict an efficient equilibrium in the environment. The two goals are interrelated. A RM that effectively protects trustors against trustees will drive the defective trustees out of the market and hence achieve an efficient equilibrium. Vice-versa, a RM that achieves efficiency also guarantees to the trustors that they will benefit from cooperative interactions.

Agents communicate with the RM through an interface that allows them to submit feedback, and retrieve reputation information about trustees. Intuitively, only trustors are required to communicate with the RM. However, since one physical agent can act in both roles, it is not reasonable to assume that the trustee does not have full access to the RM. Besides, as we will later show in Section 4.2 we can use the fact that the trustee can also submit feedback to design a better RM.

A RM has to clearly specify the type and form of information that should be collected about the behavior of trusted agents. In an e-commerce transaction, we are mainly interested whether or not the seller delivered the promised product; however, we could also consider the promptness of the response, the quality of the customer support, etc. Feedback can be binary, (the seller delivered or not the good), but also discrete (Amazon's star rating) or even continuous (the quality of a product given as a real number between 0 and 1). We will restrict our attention to binary feedback consisting of a positive ($R+$) or a negative ($R-$) report indicating whether the trustee has respected or not his promise. Binary feedback is the simplest feedback available, is very intuitive for reporting agents, and can efficiently model many real situations.

Figure 1 presents the reputation mechanism as an equilibrium answer to the following three interrelated questions:

1. How do trusting agents use the reputation information to make trust decisions regarding trusted agents?
2. What is the value rational trustees associate with reputation information?
3. How can a designer build an efficient reputation mechanism?

Trusting Decisions. In electronic environments, trustors use precise rules for taking trusting decisions based on reputation information. With rational trustors, the goal of the trusting decision is to maximize the agent's payoff given the description of the environment and the available data (i.e. the reputation of the trustee). Typical trusting decisions allow a trustor to:

- scale the value of a transaction with a specific trustee; e.g. in an exchange of a good of value p, when the reputation R of the seller has the semantics of *probabil-*

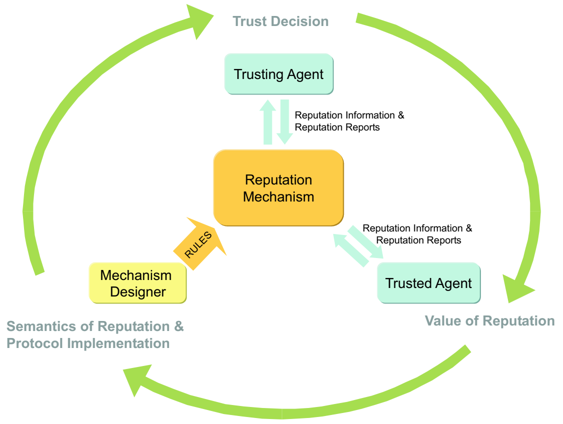

Figure 1. The reputation mechanism as an equilibrium solution to interrelated and conflicting aspects.

ity of delivering the good, a rational buyer decides to pay $R \cdot p$ for that good;

- decide whether or not it is beneficial to trade with a specific trustee; e.g. A business transaction brings the trustor $p_c > 0$ if the trustee cooperates and $p_d < 0$ if the trustee defects. When the reputation R of the trustee indicates the probability with which the trustee will cooperate, it is rational for the trustor to participate in the transaction only if $R \cdot p_c + (1 - R) \cdot p_d > 0$.

Value of Reputation. When the trustor uses reputation information about the trustee to take trusting decisions, the present value of the reputation has a direct impact on the future business of the trustee. Reputation is assigned a value reflecting the influence of the reputation on future revenues. The value of the reputation R can be computed by determining how much more can a trustee gain by having the reputation R rather than the minimum reputation.

In practical situations it is useful to attribute a value to a feedback report, or more precisely, to attribute a value to the fact that a positive reputation report has been filed instead of a negative one. We can therefore talk about the values $V(R+)$ and $V(R-)$ of a positive, respectively a negative reputation report, however, only the difference $V(R+) - V(R-)$ is well defined.

Mechanism Designer. The mechanism designer has: (1) to define the semantics of reputation, and (2) to design the interaction protocol of the mechanism, such that the RM meets certain requirements as perfect as possible.

The first requirement is that the RM be efficient, i.e. impose the cooperative outcome in the market. Efficiency can be achieved by designing feedback aggregation rules such that the loss caused by receiving a negative reputation report instead of a positive one outweighs the gain obtained from cheating in the present transaction.

Even the perfect feedback aggregation rules cannot make a RM efficient unless correct feedback is available to the mechanism. In the absence of verification authorities, true feedback can be obtained only from the agents themselves, making incentive-compatibility a second requirement.

Equally important properties are also (1) the resistance to collusion (i.e. by colluding, two or more agents should not be able to manipulate the RM to the detriment of other agents), (2) scalability, (3) robustness of information, and (4) the reliability of the reputation information itself.

4. Incentive Compatibility

In a decentralized environment in which verification authorities are not available, truthful reputation information can be obtained only if we make it in the best interest of the agents to report the truth. Generating feedback is quite an easy task with rational agents: any positive payment for a feedback report will determine agents to provide feedback. Generating true feedback, however, is a more complicated issue.

Incentive-compatibility can be treated differently for two distinct cases, depending on how we model the behavior of the trusted agent. In the first category, we treat agent behavior that can be modeled by a Markov chain of length n. For this case, we explain in Section 4.1 how it is possible to make truthful reporting a Nash equilibrium by conditioning the side payment for a report on the correlation of that report with future reports submitted about the same agent.

In the second category we treat completely opportunistic agent behavior. In this case, we assume that a trustee can use whatever means necessary in order to maximize his payoff, including cheating, lying, bribing or blackmailing other agents. The side payments used in the previous case do not work for this situation. More subtle means have to be devised, such that truthful reporting emerges as an equilibrium of the entire game.

4.1. Behavior that can be Modeled by a Markov chain

We will use the example of a Web Service (WS) that provides a weather forecast report in exchange for a fee. The client of such a Web Service (the trusting agent) is first required to pay and then wait for WS (the trusted agent) to deliver the report. WS can deliver or not the information, and the client perfectly perceives the action of the WS. We assume that the quality of the information is not an issue; the only risk for the client is that WS might not send the information. At the end of every interaction, the client is asked to submit a binary report about WS.

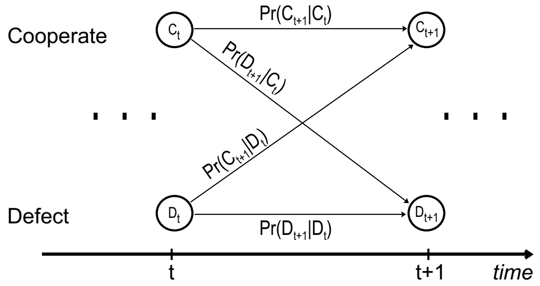

Figure 2. The behavior of the Web Service, modeled as a Markov chain of length 1.

The behavior of the Web Service in one particular transaction is not directly determined by the strategic will of the owner of that service. We can safely assume that the WS was honestly designed to cooperate in every interaction (i.e. send the weather forecast to the client). However, due to various reasons (network congestion, security attacks, software and hardware failures), it might happen that WS fails from time to time. Moreover, there is a strong correlation between WS's present behavior and his behavior in the immediate future: e.g. if the present invocation failed due to some hardware problems, it is likely that the few next invocations will also fail. The web service behaves in a Markovian manner, and Figure 2 presents WS's behavior modeled by a Markov chain of length 1, such that the action taken by WS at time $t + 1$ is given by a probability distribution conditioned on the action taken at time t.

We consider an environment like the one in Figure 3 in which many client agents and many web services might interact pair-wise. In [8] we describe an incentive compatible RM that is based on side payments organized through a set of broker agents that we have called *R-agents*. R-agents buy feedback and sell reputation information (resulted from aggregating the feedback) and act as local pieces of the global RM. There isn't however, any synchronicity requirement on the information stored by two different R-agents (i.e. they could have conflicting reputation information about the same web service), which makes the global mechanism distributed, robust and easily scalable.

The incentive compatible property is based on a very simple payment rule that makes it rational for the clients to submit true feedback: the R-agent pays the reputation report coming from client A about the web service WS$_B$ only if it matches the next report submitted by another client C about the same web service WS$_B$.

To understand how the mechanism works, let us consider the parameters of the web service behavior model as in Figure 4. It is easy to verify that a client maximizes her payment for the feedback report when reporting the truth.

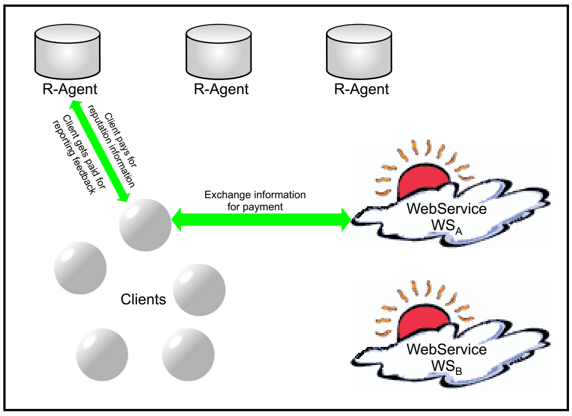
R-Agent
R-Agent
R-Agent
Client pays for reputation information
Client gets paid for reporting feedback
Exchange information for payment
WebService
WS_A
Clients
WebService
WS_B

Figure 3. Clients interact with Web Services and use Reputation Agents.

When used in real situations, the mechanism itself has to be trustworthy enough. Two security issues are of great concern:

- Identity theft, i.e. agents impersonate other agents in order to steal their reputation;
- Integrity of information, i.e. the reputation of one agent cannot be modified without the agent's consent.

We have addressed these problems by using a cryptographic mechanism based on public key infrastructure. Reputation information about an agent WS$_A$ is stored by R-agents encrypted and signed with WS$_A$'s secret key. Therefore no agent will be able to impersonate agent WS$_A$ and remain undetected. In order to update an agent's reputation we have introduced a contracting phase in every interaction. As part of the contract, the Web Service provides the client with two cryptographic tokens, representing the encrypted signed values of his updated reputation with a positive respectively a negative report. After the transaction, instead of submitting a simple binary feedback report, the client submits directly the updated reputation of the web service (i.e. the reputation of WS$_A$, updated with a positive or a negative report depending on the outcome of the transaction, signed and encrypted with WS$_A$'s secret key). This protocol guarantees that WS$_A$'s reputation can be modified only if WS$_A$ has been involved in some interaction.

4.2. Opportunistic Behavior

When agents are purely opportunistic, the above described mechanism cannot be used to elicit truthful reputation reports. To exemplify this, let us consider the simple example of one agent (an online seller) who has the freedom to rationally decide for every transaction if he is going to cooperate or not. A reasonable strategy for such a seller

	C_t	D_t
C_{t+1}	$Pr(C_{t+1}\|C_t) = 0.95$	$Pr(C_{t+1}\|D_t) = 0.3$
D_{t+1}	$Pr(D_{t+1}\|C_t) = 0.05$	$Pr(D_{t+1}\|D_t) = 0.7$

Figure 4. Possible parameters of the behavior of the Web Service

is to “work” very hard in the first transactions and establish an excellent reputation for cooperative behavior. When the good reputation spreads in the environment, the seller can start cheating from time to time. Using the mechanism described above, a customer agent that gets cheated by a reputable seller does not have any incentive to report the truth. In the next interaction the seller will probably cooperate, so it is more profitable for the client agent to report false positive feedback.

In fact, this is not just a drawback of the payment scheme described in Section 4.1. Any existing payment scheme [12, 5] based on some form of correlation between the present report and future, unknown, reports submitted about the same agent, relies on the assumption that the observable signals emitted by different sellers types are independently, identically distributed. This assumption does not hold for purely opportunistic agents. No matter how we will chose the payment scheme, there will always be an agent behavior for which the payment scheme is not incentive compatible.

Fortunately there are ways in which a reputation mechanism can be made incentive compatible even when trusted agents are purely opportunistic, if we assume that trusting agents (i.e. client agents) also have a persistent presence in the environment. The intuition behind this class of mechanisms is quite simple. While for the sellers it is profitable to build a reputation for cooperative behavior, for the clients we can make it profitable to build a reputation for truthful reporting. A rational seller will not cheat on a client who has a good reputation for always reporting the truth: the resulting negative report will be believed by the community, resulting in a decrease of the seller’s reputation which offsets the momentary gain obtained from cheating. Therefore, given that a client interacts with a particular seller a sufficient number of times, it is rational for the client to build a reputation for truthful reporting in order to determine the seller to cooperate in the future transactions when the buyer’s reputation as truthful reporter becomes credible.

Let us consider the following example inspired by the realistic scenario of the online hotel booking industry: One hotel has N rooms which offer exactly the same accommodation conditions. The quality of the hotel is judged by taking into consideration a number of criteria, e.g. the level of noise, cleanness, available facilities, the professionalism of the staff, etc. We make the simplifying assumption that the values of all these attributes can be combined into one measure of the quality of the service offered by the hotel.

Let us use a normalized value for the quality of service of the hotel, such that a quality of 1, denotes the best possible service offered by any hotel. Similarly, a quality of 0 denotes the worst possible service. It is common knowledge in the environment that any customer is willing to pay w dollars for a night spent in a hotel offering the best possible service (i.e. quality of service 1). We define the real quality of service, α, of a particular hotel, such that a customer who knows the service of that hotel is willing to pay αw dollars for one room. We also assume that all customers, given enough information, agree on the same number for the quality of service of one hotel.

Each hotel has an a priori maximum quality level α, determined by the available space, the quality of interior decorations, the training of the personnel, etc. However, in order to actually provide the quality α, the hotel has to spend every night for every room w_r dollars representing running costs like wages for enough employees, maintaining working elevators and phones, cleaning, etc.

The hotel can decide every night, for every room, whether or not to spend the running costs, w_r. If w_r is spent for a particular room, that customer will experience the maximum quality level α. On the contrary, if w_r is not spent for one room, the respective customers will experience a quality level much lower than α.

After every night, for every client, a reputation mechanism records a binary report about the hotel. (i.e. indicating whether or not the client experienced the promised quality of service that night). All the reports are aggregated into one measure $R \in [0, 1]$, of the reputation of the hotel. R can be interpreted as the probability that the hotel will fulfill its promise, and directly affects its occupancy rate.

The reputation mechanism we have designed [9] requires both the hotel and the client to submit feedback. First the hotel is required to report its behavior towards that client, and then, if the hotel claims having cooperated, the client is also asked to submit a report. (Figure 5)

Three cases are possible:

1. The hotel admits having cheated. For a hotel, falsely acknowledging defection implies a double loss (i.e. the future loss due to a negative reputation report, and the momentary loss coming from not taking the opportunity to defect) and therefore no rational hotel will report *defection* without actually defecting. Regardless of the clients’s report, we can conclude in this case that the hotel indeed defected.

2. Both agents report *cooperation*. In this case a positive report can be recorded for the hotel.

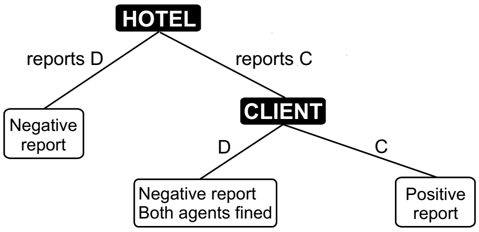

Figure 5. Both agents submit feedback.

3. The hotel claims *cooperation* while the client reports *defection*. In this case, we can only be sure that one of the agents is lying. Since untruthful reporting is what we seek to avoid, both agents will be punished in this case: a negative report is being recorded for the hotel, and both the client and the hotel are fined for lying with ε_C and ε_H respectively.

In [9] we formally prove that if the hotel believes with non-zero probability that the client might always report the truth, than there is un upper bound on the number of times the hotel will defect on the client, given that the client has always reported the truth. This upper bound on the number of defections, directly translates into an upper bound on the probability with which the reputation mechanism records false probabilities.

When the hotel has 20 rooms, a room costs 140 dollars offering a quality level of 0.7, the running costs are 10 dollars per room per night, the client returns with probability 0.7 to the same hotel and the lying fees are $\varepsilon_H = 20$, $\varepsilon_C = 1$, than the reputation mechanism presented above accepts false reports with vanishing probability. Moreover, the reputation mechanism thus obtained can be easily decentralized and scales well in large environments.

5. Conclusions and Future Work

In this paper we address the essential problem of eliciting truthful feedback from the rational agents interacting with a binary reputation mechanism. We present the basic assumptions behind binary reputation mechanisms used in decentralized environments, and describe the main components and requirements of a binary RM. We show how a RM can be made incentive compatible depending on how we model the behavior of trusted agents. The mechanisms are accompanied by examples inspired by real internet applications.

As future work we plan to study the influence of mistakes and irrational behavior on the property of incentive compatible. We will also look at ways to adapt our mechanisms to fully decentralized, peer-to-peer environments.

References

[1] R. Axelrod. *The Evolution of Cooperation*. Basic Books, New York, 1984.

[2] A. Birk. Learning to Trust. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 133–144. Springer-Verlag, Berlin Heidelberg, 2001.

[3] A. Biswas, S. Sen, and S. Debnath. Limiting Deception in a Group of Social Agents. *Applied Artificial Intelligence*, 14:785–797, 2000.

[4] S. Braynov and T. Sandholm. Incentive Compatible Mechanism for Trust Revelation. In *Proceedings of the AAMAS*, Bologna, Italy, 2002.

[5] C. Dellarocas. Goodwill Hunting: An Economically Efficient Online Feedback. In J. Padget and et al., editors, *Agent-Mediated Electronic Commerce IV. Designing Mechanisms and Systems*, volume LNCS 2531, pages 238–252. Springer Verlag, 2002.

[6] C. Dellarocas. Efficiency and Robustness of Binary Feedback Mechanisms in Trading Environments with Moral Hazard. MIT Sloan Working Paper #4297-03, 2003.

[7] D. Houser and J. Wooders. Reputation in Internet Auctions: Theory and Evidence from eBay. University of Arizona Working Paper #00-01, 2001.

[8] R. Jurca and B. Faltings. An Incentive-Compatible Reputation Mechanism. In *Proceedings of the IEEE Conference on E-Commerce*, Newport Beach, CA, USA, 2003.

[9] R. Jurca and B. Faltings. "CONFESS". An Incentive Compatible Reputation Mechanism for the Online Hotel Booking Industry. In *Proceedings of the IEEE Conference on E-Commerce*, San Diego, CA, USA, 2004.

[10] R. Kramer. Trust Rules for Trust Dilemmas: How Decision Makers Think and Act in the Shadow of Doubt. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 9–26. Springer-Verlag, Berlin Heidelberg, 2001.

[11] D. M. Kreps, P. Milgrom, J. Roberts, and R. Wilson. Rational Cooperation in the Finitely Repeated Pisoner's Dilemma. *Journal of Economic Theory*, 27:245–252, 1982.

[12] N. Miller, P. Resnick, and R. Zeckhauser. Eliciting Honest Feedback in Electronic Markets. Working Paper, 2003.

[13] L. Mui, A. Halberstadt, and M. Mohtashemi. Notions of Reputation in Multi-Agents Systems:A Review. In *Proceedings of the AAMAS*, Bologna, Italy, 2002.

[14] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1997.

[15] P. Resnick and R. Zeckhouser. Trust Among Strangers in Electronic Transactions: Empirical Analysis of eBay's Reputation System. In M. Baye, editor, *The Economics of the Internet and E-Commerce*, volume 11 of Advances in Applied Microeconomics. Elsevier Science, Amsterdam, 2002.

[16] M. Schillo, P. Funk, and M. Rovatsos. Using Trust for Detecting Deceitful Agents in Artificial Societies. *Applied Artificial Intelligence*, 14:825–848, 2000.

[17] B. Yu and M. Singh. Detecting Deception in Reputation Management. In *Proceedings of the AAMAS*, Melbourne, Australia, 2003.