D2.4.9 Reputation Mechanisms

Radu Jurca (EPFL), Boi Faltings (EPFL),
Walter Binder (EPFL), Swalp Rastogi (EPFL),
David Portabella Clotet (EPFL)

Abstract.

EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB
Deliverable D2.4.9 v2 (WP2.4)

This report summarizes the state-of-the-art regarding reputation mechanisms, investigates a reputation model suitable in a semantic service-oriented environment as well as methods for stimulating honest reporting. Finally we provide the design of a framework for allowing agents to query and submit reputation information and a simple approach to integrate reputation mechanisms into the process of service selection. A prototype implementation will be available as part of D2.4.6.2.

Keyword list: Reputation Mechanisms, Web Services, Semantic Web Services, Semantic Web

Document Identifier	KWEB/2005/D2.4.9/v2
Project	KWEB EU-IST-2004-507482
Version	v2
Date	November 30, 2005
State	draft
Distribution	public

Knowledge Web Consortium

This document is part of a research project funded by the IST Programme of the Commission of the European Communities as project number IST-2004-507482.

University of Innsbruck (UIBK) - Coordinator
Institute of Computer Science
Technikerstrasse 13
A-6020 Innsbruck
Austria
Contact person: Dieter Fensel
E-mail address: dieter.fensel@uibk.ac.at

École Polytechnique Fédérale de Lausanne (EPFL)
Computer Science Department
Swiss Federal Institute of Technology
IN (Ecublens), CH-1015 Lausanne
Switzerland
Contact person: Boi Faltings
E-mail address: boi.faltings@epfl.ch

France Telecom (FT)
4 Rue du Clos Courtel
35512 Cesson Sévigné
France. PO Box 91226
Contact person : Alain Leger
E-mail address: alain.leger@rd.francetelecom.com

Freie Universität Berlin (FU Berlin)
Takustrasse 9
14195 Berlin
Germany
Contact person: Robert Tolksdorf
E-mail address: tolk@inf.fu-berlin.de

Free University of Bozen-Bolzano (FUB)
Piazza Domenicani 3
39100 Bolzano
Italy
Contact person: Enrico Franconi
E-mail address: franconi@inf.unibz.it

Institut National de Recherche en
Informatique et en Automatique (INRIA)
ZIRST - 655 avenue de l'Europe -
Montbonnot Saint Martin
38334 Saint-Ismier
France
Contact person: Jérôme Euzenat
E-mail address: Jerome.Euzenat@inrialpes.fr

Centre for Research and Technology Hellas /
Informatics and Telematics Institute (ITI-CERTH)
1st km Thermi - Panorama road
57001 Thermi-Thessaloniki
Greece. Po Box 361
Contact person: Michael G. Strintzis
E-mail address: strintzi@iti.gr

Learning Lab Lower Saxony (L3S)
Expo Plaza 1
30539 Hannover
Germany
Contact person: Wolfgang Nejdl
E-mail address: nejdl@learninglab.de

National University of Ireland Galway (NUIG)
National University of Ireland
Science and Technology Building
University Road
Galway
Ireland
Contact person: Christoph Bussler
E-mail address: chris.bussler@deri.ie

The Open University (OU)
Knowledge Media Institute
The Open University
Milton Keynes, MK7 6AA
United Kingdom
Contact person: Enrico Motta
E-mail address: e.motta@open.ac.uk

Universidad Politécnica de Madrid (UPM)
Campus de Montegancedo sn
28660 Boadilla del Monte
Spain
Contact person: Asunción Gómez Pérez
E-mail address: asun@fi.upm.es

University of Karlsruhe (UKARL)
Institut für Angewandte Informatik und Formale
Beschreibungsverfahren - AIFB
Universität Karlsruhe
D-76128 Karlsruhe
Germany
Contact person: Rudi Studer
E-mail address: studer@aifb.uni-karlsruhe.de

University of Liverpool (UniLiv)
Chadwick Building, Peach Street
L697ZF Liverpool
United Kingdom
Contact person: Michael Wooldridge
E-mail address: M.J.Wooldridge@csc.liv.ac.uk

University of Manchester (UoM)
Room 2.32. Kilburn Building, Department of Computer
Science, University of Manchester, Oxford Road
Manchester, M13 9PL
United Kingdom
Contact person: Carole Goble
E-mail address: carole@cs.man.ac.uk

University of Sheffield (USFD)
Regent Court, 211 Portobello street
S14DP Sheffield
United Kingdom
Contact person: Hamish Cunningham
E-mail address: hamish@dcs.shef.ac.uk

University of Trento (UniTn)
Via Sommarive 14
38050 Trento
Italy
Contact person: Fausto Giunchiglia
E-mail address: fausto@dit.unitn.it

Vrije Universiteit Amsterdam (VUA)
De Boelelaan 1081a
1081HV. Amsterdam
The Netherlands
Contact person: Frank van Harmelen
E-mail address: Frank.van.Harmelen@cs.vu.nl

Vrije Universiteit Brussel (VUB)
Pleinlaan 2, Building G10
1050 Brussels
Belgium
Contact person: Robert Meersman
E-mail address: robert.meersman@vub.ac.be

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document:

Ecole Polytechnique Fédérale de Lausanne

Changes

Version	Date	Author	Changes
0.5	01.03.05	Walter Binder	creation
0.6	15.03.05	Walter Binder	state-of-the-art added
0.7	12.04.05	Walter Binder	state-of-the-art extended
0.8	20.05.05	Walter Binder	model overview and scenarios added
1.0	21.06.05	Walter Binder	finalization of first version
1.1	31.07.05	Walter Binder	addressed review comments
1.2	03.08.05	Radu Jurca	addressed further review comments
1.3	23.09.05	David Portabella Clotet	outline of D2.4.9 v2
1.4	07.10.05	David Portabella Clotet	updated chapter 3, 5
1.5	07.10.05	Swalp Rastogi & David Portabella Clotet	chapter 9
1.6	21.10.05	Radu Jurca & David Portabella Clotet	example scenarios
1.7	27.10.05	Radu Jurca & David Portabella Clotet	chapter 6
1.8	04.11.05	Radu Jurca & David Portabella Clotet	IC and Evening Planner
1.9	11.11.05	David Portabella Clotet	addressed review comments
1.95	30.11.05	David Portabella Clotet	addressed review comments
2.0	11.01.06	Radu Jurca & David Portabella Clotet	addressed review comments

Executive Summary

In an open environment where malicious parties may advertise false service capabilities the use of reputation services is a promising approach to mitigate such attacks. Misbehaving services receive a bad reputation and will be avoided by other clients. Reputation mechanisms help to improve the global efficiency of the overall system because they reduce the incentive to cheat.

This report summarizes the state-of-the-art regarding reputation mechanisms, investigates a reputation mechanism suitable in a semantic service-oriented environment and methods for stimulating honest reporting.

We consider ways of modeling trust, computational models of trust, as well as incentive-compatible reputation mechanisms. Concerning computational models of trust, we distinguish social trust networks, probabilistic estimation techniques, and game-theoretic models.

We design a reputation mechanism that is suitable for semantic service-oriented applications where providers may provide services with different production quality levels and clients can consciously accept a lower service quality level for a lower price. This reputation mechanisms requires that agents report honestly and so we also provide some ways for incentive compatibility.

Finally we provide a framework for allowing agents to query and submit reputation information and a simple approach to integrate reputation mechanisms into the process of service selection. A prototype implementation will be available as part of D2.4.6.2.

Contents

Chapter 1

Introduction

The availability of ubiquitous communication through the Internet is driving the migration of commerce and business from direct interactions between people to electronically mediated interactions. It is enabling a transition to peer-to-peer commerce without intermediaries and central institutions.

Most business transactions have the form of prisoners' dilemma games where dishonest behaviour is the optimal strategy. For instance, in a service level agreement, there is no incentive for the service provider to deliver the promised quality of service once he has received the client's payment (assuming the absence of a public security infrastructure that mediates every transaction). In conventional commerce, personal relations create psychological barriers against such behaviour. In electronically mediated peer-to-peer commerce, there is no physical contact and even identities can be easily faked. Fraud and deception are a major obstacle to realizing the huge economic benefits of peer-to-peer commerce.

A standard approach in traditional business to avoid cheating (i.e., to avoid deviation from a promise / from partners' expectations) is to use trusted third parties (TTP) that oversee the transactions and rule out or at least punish cheating. In electronic interactions this approach is not always possible, as they pose problems of verification, scalability, cost and legality when several countries are involved in transactions.

In this report, we pursue a fundamentally different approach to electronically mediated business that can do without enforcement by third parties. We consider reporting, sharing and using reputation information in a network of agents as part of a mechanism that makes cooperation the dominant strategy in business transactions. Such an approach aims at re-establishing a social framework that supports trusted interactions needed in a semantic web service environment where services are discovered and composed on the fly. It is based on the observation that agent strategies change when we consider that interactions are repeated: the other party will remember past cheating, and change its terms of business accordingly in the future. In this case, the expected future gains due to future transactions can offset the loss incurred by not cheating in the present transaction [43].

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 behaviour. Game theorists have studied the reputation effect for many years and established results that show its feasibility in a wide variety of scenarios. What is missing now are robust and scalable computational mechanisms for implementing them in electronic peer-to-peer commerce.

Most of the existing reputation mechanisms tend to act by social exclusion, i.e. separating trustworthy and untrustworthy agents. While this can work well in some systems (eBay, Slashdot, Amazon), it is not suitable in a service-oriented environment where the perfect service is impossible to provide (or prohibitively expensive), and various providers might prefer different production quality levels [31]. Such settings require a more flexible mechanism in which quality of service can be traded, for example, with the price: i.e. clients consciously accept a lower service quality level for a lower price. We show a reputation mechanism where repeated failures do not automatically exclude a provider from the market, but rather influence the price the provider can charge for a future service. However this mechanism requires that agents report honestly for the system to work, which it is not likely to happen naturally in an environment with self-interested agents. Mechanisms based on side-payments can be conceived such that honest reporting becomes rational (i.e. Nash equilibrium). Unfortunately, for every incentive-compatible Nash equilibrium there seems to also be a dishonest Nash equilibrium strategy that sometimes is more attractive [39], [25]), [28] and so we need to find ways to relax the assumption of truth-telling.

Finally we provide a framework for allowing agents to query and submit reputation information and a simple approach to integrate reputation mechanisms into the process of service selection. A prototype implementation will be available as part of D2.4.6.2.

This report is structured as follows: In Chapter 2 we discuss the current state-of-the-art concerning reputation mechanisms. We address issues regarding the modeling of trust as perceived by human beings, computational models of trust, and incentive-compatible reputation mechanisms. In Chapter 3 we describe four example scenarios in which a reputation mechanism can significantly improve the efficiency of the system. In Chapter 4 we focus on the requirements for a reputation mechanism and we define the methodology for modeling and implementing these mechanisms. A reputation-based pricing of web services is presented in Chapter 5 with theoretical guaranties that the proposed mechanism works for the requirements given in the previous chapter, provided that agents reports truthfully. To relax this latter assumption, Chapter 6 describes three incentive-compatible strategies for honest reporting. In Chapter 7 we describe a simplified reputation mechanism that filters out malicious reports rather than providing rational incentives for truthful reporting. In Chapter 8 we design the reputation framework that will be implemented in D2.4.6.2. Finally, Chapter 9 concludes this report.

Chapter 2

State of the Art

We see trust related research as going into three directions:

1. Work that models the notion of "real world" trust (as used in sociology and psychology primarily) and propose definitions of trust that are appropriate for use in online settings. The definition of trust and its corresponding meaning is a much disputed issue among the computer science community. Since the human understanding of the notion of trust is much too complex to be modelled within an artificial system, authors usually consider just facets of the notion of trust, and define it corresponding to their needs. Trust modeling is addressed in Section 2.1.

2. Computational models of trust, proposing concrete models for trust evaluation. We characterize these models along the following two dimensions:

   - How precisely they boost trust in the community in which they are deployed.
   - How efficiently they can be implemented in a decentralized network of agents.

   Computational models of trust are discussed in Section 2.2.

3. Incentive compatibility related works. Incentive compatibility is one of the most desirable properties of protocols involving communication among autonomous, self-interested agents. Its existence assures that specific behaviour (truth telling, in particular) is equilibrium of the game constructed from the protocol. Applied to the trust models, incentive compatibility would imply truthful reporting of reputation information. Incentive-compatible reputation mechanisms are covered by Section 2.3.

2.1 Modeling Human Trust

Generally, the notion of trust is used to refer to a subjective decision making process that takes into consideration a lot of factors. [33] explains how human beings deal with the trust decision making process by using a set of rules. The model proposed is the Social Auditor Model. The author also studies the efficiency of different rules and strategies that can be used within an artificial society.

In [5, 21, 14] the authors look at the dynamics associated with the notion of trust. Trust and distrust responsiveness (trust from the trustee increases the probability of cooperative behaviour from the trustor, while distrust from the trustee increases the probability of defective behaviour) are presented as facts of human behaviour. Also the dialectic link between trust and degree of control is addressed.

In [36] a multi-disciplinary literature survey on the notion of trust and distrust is presented. The paper develops a conceptual topology of the factors that contribute towards trust and distrust decisions and defines as subsets of the high level concepts measurable constructs for empirical research.

One of the 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 trustor: competence, benevolence, integrity and predictability. [40] presents an extensive classification of reputation by the means of collecting it. Experiments for finding out which component contributes the most towards correct trust decisions are also conducted.

As belonging to this group can be regarded works that investigate some inherent characteristics of the online world that any trust management model must be aware of. [22] discusses risks associated with the ease at which members of online communities can change their identities. Through a game theoretic modelling, they come to a conclusion that newcomers must start with the lowest possible reputation value in order to be discouraged to misbehave and change their identity afterwards. [15] identifies a number of possible attacks on reputation reporting systems (“ballot stuffing”, “bad mouthing”, etc.) and proposes an appropriate solution to reduce effects of those attacks. [44] identifies main patterns of human behaviour with respect to trust. It argues that, despite clear incentives to free ride (not leave feedback) and leave only positive feedback, trust among eBay traders emerges due to its reputation system.

While humans address trust issues in a complex way, considering the direct experience with a provider, the experience of others with the provider, as well as social aspects (e.g., nationality, group membership, etc.), only one aspect – the reputation of the provider – is modeled in computer systems, as discussed in the following section.

2.2 Computational Models of Trust

In the categorization of the computational models of trust we will adopt here a division based on how they perform the goal of bootstrapping trust. We see the following three broad classes of approaches: 1) social (trust) networks formation (Section 2.2.1), 2) probabilistic estimation techniques (Section 2.2.2), and 3) game-theoretic models (Section 2.2.3). For all these approaches, different feedback aggregation strategies are possible, as discussed in Section 2.2.4.

2.2.1 Social Trust Networks

The underlying assumption of the class of social networks formation is that the agents engage in bilateral interactions whose outcomes are evaluated and aggregated, which results in forming a trust graph in which each branch (a,b) is assigned a weight representing the trust of agent a towards agent b aggregated across all interactions between them in which agent a happened to have relied on agent b. Having the local interactions among the agents encoded this way, the challenge is how to merge these local beliefs to enable the agents to compute the trustworthiness of non-neighbouring agents, whom they never met before. The main distinguishing points among the numerous works belonging to this class are: 1) the strategy to aggregate individual experiences to give the mentioned weights, 2) the strategy to aggregate the weights along a path of an arbitrary length to give a path wide external opinion and 3) the strategy to aggregate this external opinion across multiple paths between two given agents.

[7] presents an early example in which a clear distinction between direct experiences and recommendations has been made, which is reflected in the strategy for path wide external opinion aggregation. However, this separation of the two contexts led to an exponential complexity of the trust derivation algorithm. Clearly, this is unacceptable for large scale networks.

[54] does not treat recommendations and direct service provisions separately. It uses a variation of the delta learning method to aggregate "positive" and "negative" experiences of the agents into the weights assigned to the corresponding branches and simple multiplication as the strategy to compute the path wide external opinions. As for the strategy to aggregate the external opinion of different paths the authors use a variation of the simple maximum function. All this results in a polynomial time algorithm for the overall trust aggregation.

In [45] Richardson et al. offer important theoretical insights on how the computational complexity of the trust derivation algorithms relates to the mentioned aggregation strategies by characterizing the combinations of path and across-path aggregation strategies that may lead to a non-exponential trust computation algorithm (we note that many other works use such combinations: e.g., [42] and [32]). The authors also offer such an algorithm which is, however, based on a synchronous participation of all agents in the network. As such it is not quite appropriate for usage in P2P networks due to their inherent high dynamicity. With respect to this problem [53] offers a considerable improvement in terms of an appropriate caching scheme that enables asynchronous computation while retaining good performance.

A common denominator of all these works is that the computed values have unclear semantics and are hard to interpret on an absolute scale, without ranking them. In many applications this imposes certain problems. On the other hand, as shown by many simulations, they are very robust to a wide range of misbehaviours.

2.2.2 Probabilistic Estimation Techniques

Probabilistic estimation techniques present certain improvement with respect to the meaningfulness of the computed values. Namely, they output probability distributions (or at least the most likely outcome) over the set of possible behaviours of the trusted agents enabling thus the trusting agents to evaluate explicitly their utilities from the decision to trust or not. [41] presents the well-known method of Bayesian estimation as the right probabilistic tool for assessing the future trusting performance based on past interactions. Only direct interactions were studied - the question of including recommendations was not considered. [13] goes a step further by taking into account the "second-hand" opinions also. However, the strategy for merging own experiences with those of other witnesses is intuitive (giving more weight to own experiences, though plausible, is still intuitive) rather than theoretically founded.

[3] presents a decentralized trust management model that analyzes past interactions among agents to make a probabilistic assessment of whether any given agent cheated in his past interactions. The emphasis is put not only on assessing trust but also on providing a scalable data management solution particularly suitable for decentralized networks. The problem in decentralized networks is that the reputation data is aggregated along wrong dimension in the sense that each agent has information about his own past interactions with others but cannot easily obtain opinions of others about any other particular agent in the network. To achieve the needed reaggregation of reputation data, the authors use P-Grid, a scalable data access structure for P2P networks [2]. For any particular agent, they designate a set of replicas to store the feedbacks, ratings of trusting behaviour of that agent (complaints filed by him about others and complaints filed by others about him) so that the reputation data can be accessed and collected efficiently, in logarithmic time. As replicas may provide false data, an appropriate replication factor along with a proper voting scheme to choose the most likely reputation data set are chosen in order to achieve accurate predictions. Trust assessments themselves are made based on an analysis of agent interactions modelled as Poisson processes. As was shown by simulations, cheating behaviour of the agents can be identified with a very high probability. The model is simplistic in the sense that, for any agent, it outputs whether the agent cheated in the past or not, but it can be easily extended to give predictions of the agents' trusting behaviour,

as done e.g. in [53].

[20] is a step further towards analyzing how trust predictions can be used in the context of making business decisions. Safe exchange represents an approach to gradual exchanges of goods and money in which both payments and goods are chunked with their deliveries scheduled in such a way that both exchange partners are better off by continuing the exchange till its end than by breaking it at any step before. The authors provide a trust aware extension of the original approach [46] by modelling trust explicitly.

2.2.3 Game-Theoretic Models

Game-theoretic reputation models make a further clarification in the interpretation of the agents' trustworthiness in the sense that, if the reputation system is designed properly, trust is encoded in the equilibria of the repeated game the agents are playing. Thus, for rational players trustworthy behaviour is enforced. The real challenge here is how to define the feedback aggregation strategies that will lead to socially desirable outcomes carrying trust.

Theoretic research on reputation mechanisms started with the seminal papers of Kreps, Milgrom, Wilson and Roberts [34, 35, 37] who explained how a small amount of incomplete information is enough to generate the reputation effect, (i.e., the preference of agents to develop a reputation for a certain type) in the finitely repeated Prisoners' Dilemma game and Selten's Chain-Store game [49].

Fudenberg and Levine [23] and Schmidt [48] continue on the same idea by deriving lower bounds on the equilibrium payoff received by the reputable agent in two classes of games in which the reputation effect can occur.

[18] focuses on a specific game and derives its equilibria. Apart from this the author also raises questions concerning the overall game-theoretic reputation systems design, such as incentivizing players to leave feedback, dealing with incomplete feedback etc. However, an underlying assumption of this work is that a central trusted authority does the feedback aggregation. We see this as a major obstacle to transferring game-theoretic models to decentralized environments.

2.2.4 Feedback Aggregation Strategies

For the previously mentioned approaches, there are different strategies to aggregate the external opinion. [15, 16, 57, 1] use collaborative filtering techniques to calculate personalized reputation estimates of as weighted averages of past ratings in which weights are proportional to the similarity between the agent who computes the estimate and the raters.

[8, 9, 10] 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.

[52] uses machine learning techniques and heuristic methods to increase the global performance of the system by recognizing and isolating defective agents. Common to these works is that they consider only direct reputation. Similar techniques, extended to take into account indirect reputation, are used in [6, 47, 26, 55, 50].

A number of reputation mechanisms also take into consideration indirect reputation information, i.e., information reported by peers. [47, 56] 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.

In [26] the authors present an example of a reputation sharing mechanism that is also incentive-compatible (see Section 2.3 for details). The mechanism is based on side payments that are organized through a set of broker agents called R-agents, which buy and sell reputation information. A simple payment rule (incoming reputation reports are paid only if they mach the next reputation report filed about the same agent) makes it rational for agents to truthfully share reputation information. The mechanism is decentralized and robust (up to certain limits) to irrational untruthful reporting.

2.3 Incentive Compatibility

The vast majority of the previously discussed models are not fully incentive-compatible in the sense that it is in the best interest of all agents to report their feedbacks truthfully (and leave feedback whatsoever, without an incentive to free ride). The closest to the idea of incentive-compatibility are [11, 12, 17, 38, 25, 29].

[11] considers exchanges of goods for money and proves that markets in which agents are trusted to the degree they deserve to be trusted is equally efficient as a market with complete trustworthiness. It then presents an exchange scenario in which buyers announce their trustworthiness and sellers compute their estimates of the same. It was shown that, with an appropriately chosen advance payment, buyers cannot benefit from announcing false levels of trustworthiness. We must note that the generalness of the results is somewhat limited by the assumption that the contract price is chosen according to a particular bargaining solution (Nash's bargaining solution in this case). For auctions which are not completely enforceable, the same authors describe in [12] a mechanism based on discriminatory bidding rules that separate trustworthy from untrustworthy bidders. Again, this result has the same limitations as mentioned above.

For e-Bay-like auctions, the Goodwill Hunting mechanism [17] provides a way in which the sellers can be made indifferent between lying or truthfully declaring the quality of the product offered for sale. Momentary gains or losses obtained from misrepresenting the product's quality are later compensated by the mechanism which has the power to modify the announcement of the seller.

A significant contribution towards eliciting honest reporting behaviour is made in [38]. The authors propose scoring rules as payment functions which induce rational honest reporting. The scoring rules however, cannot be implemented without accurately knowing the parameters of the agents' behaviour model, which can be a problem in real-world systems. Moreover, this mechanism works only when the set of possible seller types is countable and contains at least 2 elements, and when the signals received by the buyers about the seller's behaviour are independently identically distributed from one interaction to the other.

Using the same principle, [25] overcomes the need to know the parameters of the agents' behaviour model at the expense of further reducing the acceptable provider behaviour types. [29] describes a novel protocol to elicit truthful reputation information in electronic markets lacking independent verification authorities by correlating the reports of the seller and buyer involved in the same transaction.

[25] further strengthens this result by theoretically delimiting the minimum set of conditions necessary for the mechanism to be incentive-compatible. Moreover, using digital signatures and a contracting protocol, the authors show how the mechanism can be made secure against identity theft (agents stealing the identity of other agents in order to benefit from an undeserved reputation) and manipulation by any single agent. A concrete implementation of this mechanism is deployed on the Agentcities platform [4].

As opposed to side-payment schemes that correlate a present report with future reports submitted about the same agent, [29] presents a mechanism that discovers (in equilibrium) the true outcome of a transaction by analyzing the two reports coming from the agents involved in the exchange. For two long-run rational agents, the authors show that it is possible to design such a mechanism that makes cooperation a stable equilibrium. The mechanism involves no independent verification authority, and is easily distributable as the decision about the true outcome of a transaction does not depend on any past or future interactions.

Chapter 3

Example Scenarios

Due to the complexity of the interactions in distributed environments, we do not expect to be able to evaluate all properties analytically. Therefore, we need a decentralized software platform and several example scenarios in which reputation mechanisms can be empirically evaluated and compared through simulation. We describe here some scenarios that illustrate the different kinds of environments that are being constructed in the Internet in which a reputation mechanism can significantly improve the efficiency of the system.

3.1 Web services

A client is interested in using a specific web service. As a first step, the client discovers potential service providers by matching her needs against the service descriptions advertised by the providers. We assume that the semantic reasoning needed for this matching is available to the clients, and that a directory facilitates the discovery of web services.

The client selects the preferred web service provider by taking into account the service description and the experience of previous clients who have interacted with that provider (i.e. reputation information). She then initiates the exchange with the chosen provider by negotiating the terms of the interaction (i.e. quality of service parameters, price, etc).

If the provider accepts the interaction, the exchange takes place, and the client is allowed to submit feedback about the provider. The interaction between the participants is depicted graphically in Figure 3.1.

3.1.1 Problem

The problem we are trying to solve in this context is that of moral hazard for service providers. In the absence of trusted third parties or verification authorities, service providers can promise high service standards, but exert very low effort in assuring such

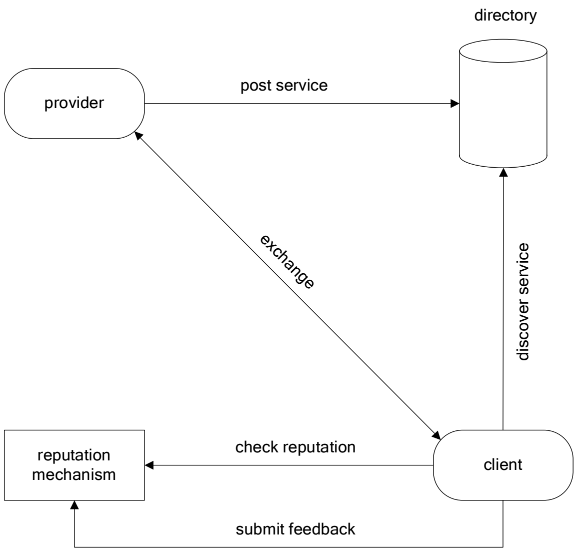

Figure 3.1: Web services: The interaction between participants.

standards. Rational clients anticipate this opportunistic behavior and will be willing to pay only the small prices corresponding to low quality services (delivered when providers do not exert effort). Low prices drive honest providers away from the market, leading to a market of "lemons".

This inefficient equilibrium can be eliminated by a reputation mechanism. Reputation reflects the previous effort of service providers, and clients learn to trust reputable providers. The role of a reputation mechanism in such a setting is twofold:

1. *signaling function*: i.e. it must allow clients to differentiate between providers having different capabilities.

2. *sanctioning function*: i.e. it must give incentive to providers to exert the efficient effort.

3.1.2 Assumptions about the Environment

- Generally, one physical agent will assume only one of the client-provider roles.[1]

[1] An exception are the agents which act as intermediaries, and compose simpler services into a more complex one. Such agents act both as clients (for the different providers offering the constituent services) and as providers (for the end-users). However, the behavior of the agent in the client role is irrelevant for the behavior of the same agent in the provider role, and therefore, different online identities could be attributed to the different roles of the same agent. We do not elaborate further on the composite web services.

- After any interaction both the provider and the client will continue to use the system with high probability. Generally, providers will return to the market with higher probability than clients.
- We assume that any service description can be decomposed into two basic blocks: the service *category* and service *attributes*. The service category describes the general function of the service, while service attributes characterize *what* exactly the service does. For example, one service category can be *transportation service*; the attributes of a transportation service ca be: package delivered in less than one week, it is insured up to that amount, the probability of loss is less than x percent, etc.
- We assume that providers accurately describe the service category. However, they can lie regarding the service attributes.
- A tuple of values describing each of the service attributes constitute a Service Level Agreement (SLA). Within a given service category, different providers advertise different SLA's.
- We assume that SLAs are non-negotiable. i.e. when clients chose to acquire the service of a provider, they implicitly accept the SLA advertised by the provider. Likewise, when a provider accepts to interact with a client, he is bounded to the advertised SLA.
- The identity of service providers is easily verifiable (e.g. certification authorities). We assume that any identity change of a service provider involves a non-negligible cost.
- The identity of clients is harder to verify (however possible, due to the payments made to providers). We assume that any identity change of a client also involves a cost.

3.2 P2P file sharing

We assume there is a client interested in downloading a file with a certain description. The client uses the index (that is made available through the underlying P2P overlay network) to discover the set of file providers offering the desired file. Having discovered the set of potential providers, the client uses the reputation mechanism to select a subset of most reputable providers, from which she will download the file. The client then immediately initiates the download from the selected sources (i.e. the exchange phase).

The providers can choose how fast and how well to satisfy the client's request (in the extreme case they can also refuse the download) by taking into consideration the reputation of the client. When the download is completed or a timeout has expired, the client can submit feedback about the provider(s) it interacted with. The entire scenario is pictured in Figure 3.2.

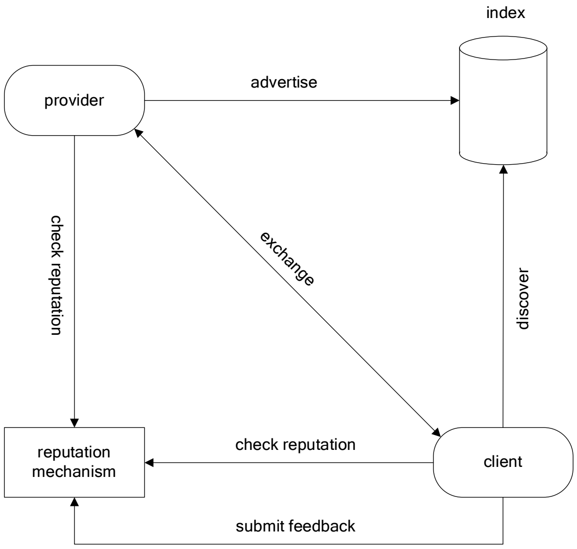

Figure 3.2: P2P File Sharing: The interaction between participants.

3.2.1 Problem

The problem we are trying to solve in this scenario is that of *free-riding*: Peers consume resources (i.e. download files) but are not motivated to contribute by also sharing files. This problem comes from the fact that agents obtain utility only when downloading files. Sharing is costly and does not bring any benefits.

Since payments are not possible in such a scenario, the role of the reputation mechanism is to make sharing rational. A cooperative equilibrium can be sustained by reciprocal actions: good reputation as a provider is rewarded by improved services when acting as a client.

3.2.2 Assumptions about the Environment

- The number of peers is big. Thus we need a scalable reputation mechanism.
- It is difficult to have side payments.
- The roles of a client and a provider are interchangeable in the sense that any client can be a provider in another file exchange and vice versa.
- After any interaction both the provider and the client will continue to use the system (in both roles) with a high probability.
- All files are equally valuable to any specific client.

- Each client is interested in only a small subset of the document space.
- The document space, as well as the interests of the peers change over time (i.e. new files are added, older files loose their interest)
- We can estimate the probability that each file in the document space is requested by the clients.
- IDENTITY: One agent can have multiple online identities. There is a small cost for creating a new identity:
  - an online identity requires a valid e-mail address.
  - the createion of an online identity requires some manual work (identifying some picture, or art text).
  - an identity can be restricted to some hardware (i.e. only one agent per machine can access the network).

3.3 Information provisioning

Information providers post information items (e.g. the review of a product or a book, financial predictions, etc) on a content distribution system (CDS). Information items are categorized and labeled such that potential clients can easily discover information of interest.

Clients pay to access information through the CDS. A charge is perceived for every accessed unit of information. The charge can be fixed (one price for all units) or variable (each unit has a different price, depending on the domain, content, reputation, author, etc).

Having accessed information, clients are entitled to submit feedback to a reputation system. Previous feedback is aggregated by the reputation mechanism into reputation information which can be accessed by later clients in order to decide which information providers / information items to access. The interaction between the participants is presented in Figure 3.3.

3.3.1 Problem

The problem we have to solve in this scenario is that of moral hazard for providers coupled with acute conflicts of interests. Since acquiring qualitative information is expensive, providers can advertise high quality information but provide only cheap information. Anticipating this behavior, clients will not buy information, and as a consequence the market collapses.

The difference from the webservices scenario is that in the present case, information providers can be affected by conflicts of interests. Biased information is also useless for

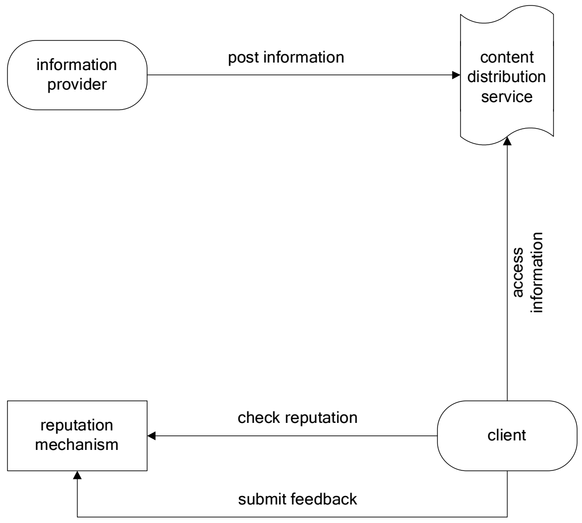

Figure 3.3: Information provisioning: The interaction between participants.

clients. A reputation mechanism needs to make the difference between real information and ”spam“. The primary role of the mechanism is therefore to *signal* qualitative information. Ideally, the mechanism should also function as a *sanctioning* tool which gives incentives to information providers to 1) exert effort in acquiring information and 2) keep neutral as well as possible from conflicts of interest.

3.3.2 Assumptions about the Environment

- Clients use information items to take decisions. The more accurate the information a client has, the better are the business decisions that client is going to make, and therefore the higher is the payoff she can expect. It is therefore important for the clients to obtain accurate information.

- Accurate information is costly for the providers.

- Submitted information items cannot be changed. Therefore the same unit of information is available to many clients.

- Conflicts of interest are present in the environment. Let us consider for example, that information items are reviews about a product. The reviews are later used by clients to decide whether or not to purchase the product in question.

- information providers have high discount factors (i.e. they return to the market with high probability).

- clients have low discount factors (i.e. the frequency with which a clients accesses information is low when compared to that of information providers).
- clients do not act as information providers.
- clients do not forward information. i.e. the amount of information circulating outside the content distribution system is negligible.
- information items become outdated after some time (i.e. have a fixed TTL)
- Both clients and providers can easily change their online identities. Obtaining a new identity involves a small cost (i.e. see the remarks about identity in Section 3.2.2).

3.4 Online Auctions

Direct Auctions.A seller is offering a product or service to a set of potential buyers through an online market. Each buyer places a bid according to her private preferences, by taking into account the description provided by the seller, and the reputation of the seller. Sellers can choose to ignore bids coming from buyers with bad reputation (i.e. buyers that do not pay once they have won the auction)

The winning bidder and the price of the product is set according to one auctioning protocol (e.g. Vickrey auction). The winner is expected to pay the seller, and then wait to receive the product. At the end of the transaction, both the buyer and the seller can submit feedback about each other. The interaction protocol is graphically depicted in Figure 3.4.

A particular case of online auctions is that of **Reversed Auctions.** In such an auction, the buyer is auctioning a job to a set o potential providers through an online market. In this case, the bids signal the commitment of the provider to offer the required service for the specified amount. The winning bidder is trusted to provide the service in exchange for payment. One common characteristic of reversed auctions is that computing the bids is expensive. For public service procurements (e.g. building a bridge or a tunnel) computing the bid (with the detailed documentation) requires an enormous work. Moreover, this particular example is governed by strict legislation regarding the commitment of the auctioneer (i.e. the buyer) once the auction has ended.

3.4.1 Problem

In this setting we identify the following problems: First, direct auctions pose a moral hazard problem for the sellers. The buyer first pays, and then waits for the product/service. The seller can very well cash the payment and ”forget“ to deliver the expected product/service. A reputation mechanism can eliminate this problem by differentiating between honest and dishonest sellers. In the same time, such a reputation mechanism can

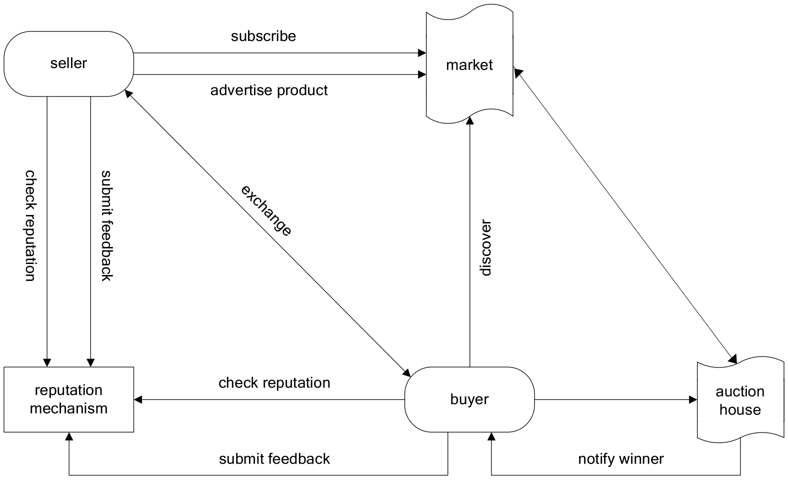

Figure 3.4: Online Auctions: The interaction between participants.

create incentives for the sellers to make their best in fulfilling the promises to the buyer. The reputation mechanism would work by associating good reputation with higher future revenues.

Second, the presence of irresponsible bidders (bidders that bid without intending to buy) in direct auctions provokes efficiency losses in the market. The goal of a reputation mechanism is to isolate (or discourage) irresponsible bidders.

Third, the buyers from reversed auctions risk being deceived by the provider that won the auction. Due to technical incompetence or bad will, the provider can miss important deadlines, or might provide a low quality service. This risk can create huge losses for the buyer (e.g. a tunnel that is not delivered on time, or that is not well constructed might create serious losses). In this case, the main role of the reputation mechanism is to screen out incompetent or malicious providers.

3.4.2 Assumptions about the Environment

- After any interaction both the provider and the client will continue to use the system with high probability. Generally, providers will return to the market with higher probability than clients.

- The auction participants are not expected to behave entirely rational.

- The probability of semantic misunderstanding with respect to the advertised product descriptions is not negligible.

- Reputation information can be used to filter out bids. Two solutions are possible:

  1. *ex-ante* filtering: bids cannot be submitted by participants with bad reputation.
  2. *ex-post* filtering: anyone can submit bids, but only the bids coming from reputable agents will be considered in winner determination.

- For reversed auctions, we assume that the buyer is legally bounded to pay for the service.

Chapter 4

Overview of Reputation Model

In this chapter we focus on the requirements concerning the selected reputation model and we define the methodology for modeling and implementing these mechanisms.

4.1 Requirements

We consider the setting shown in Figure 4.1, where trust between trusting and trusted agents is mediated by a reputation mechanism designed by a mechanism designer. The reputation mechanism appears as an equilibrium solution to the following three interrelated problems:

1. Trusting agents use the reputation information in order to make trust decisions regarding trusted agents.

2. The value associated by rational agents to reputation information depends on the trusting decisions of the trusting agents (e.g. trusting decisions affect future gains, hence the value of reputation)

3. The mechanism designer should build the reputation mechanism such that the value associated by trusted agents with "good" reputation outweighs the monetary gains obtained from cheating.

None of these questions can be answered separately. They need to be treated together such that a stable solution is reached (i.e., the answer to one question does not trigger a change in the solution of the next).

We concentrate on the two main aspects of a reputation mechanism: (1) the semantics of reputation and (2) the protocol implementation. A clear semantics of reputation is given by deciding on:

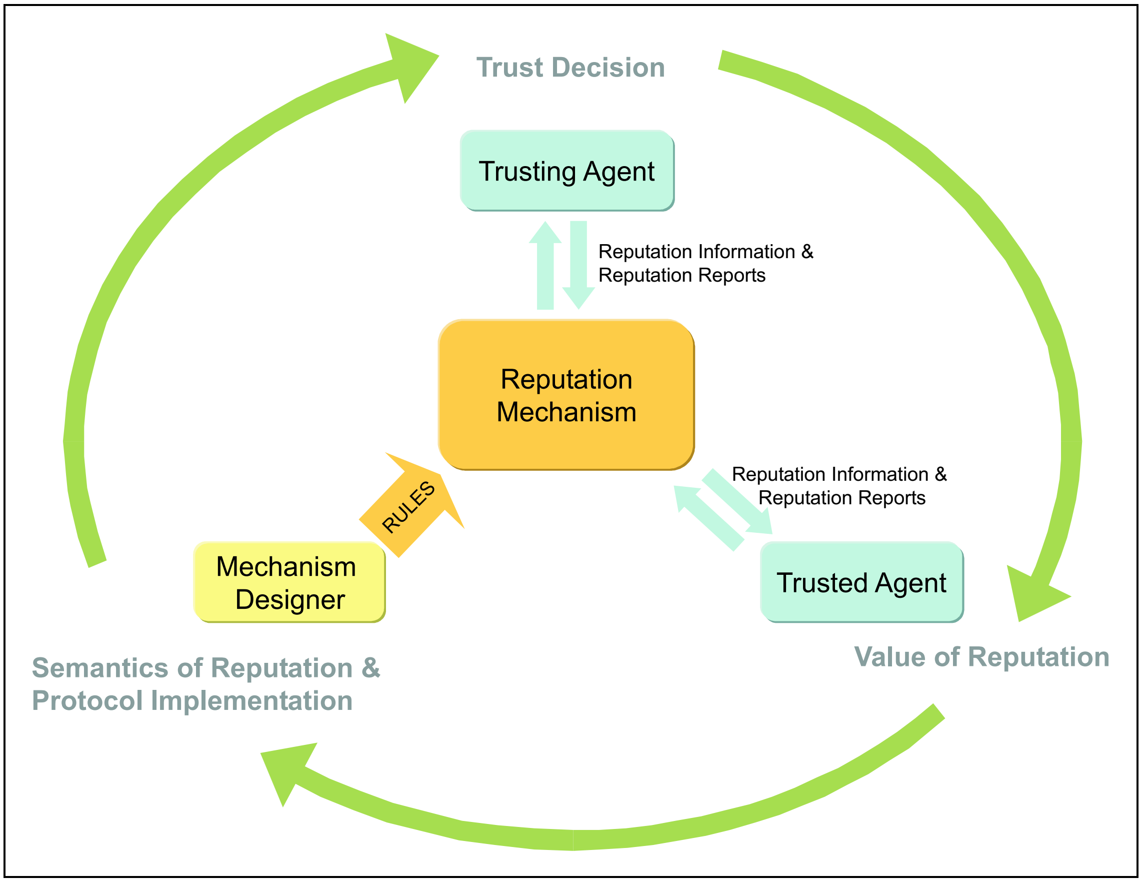

Figure 4.1: This figure illustrates the reputation mechanism as an equilibrium solution to interrelated and conflicting aspects. Note that the reputation mechanism could be physically implemented by the same agent using the mechanism.

- The type of feedback collected by the mechanism: i.e., what kind of information about an agent's past behaviour is relevant for a trusting agent in making trust decisions. The type of feedback is context dependent.
- Feedback aggregation rules: i.e., how can feedback be aggregated into meaningful reputation information.

Well defined reputation information results in clear guidelines for taking trust decisions based on reputation and for evaluating the value of reputation.

Regarding protocol implementation, a number of functional properties have to be taken into account, that determine the quality and the reliability of the mechanism proposed:

1. Incentive-compatibility: agents should have the incentive to provide truthful feedback to the mechanism.

2. Security against non-collusion: an important aspect of multi-agent systems is that agents are independent and do not collude. Certain protocols are robust against collusion, while for others collusion has to be ruled out for example by randomisation or cryptographic mechanisms.

3. Scalability: implementing the mechanism in a large network of independent, self-organizing agents leads to non-trivial scalability problems in terms of communication and information management cost.

4. Robustness: in real-world settings robustness of the mechanism against failure, faulty information and malicious behaviour cannot be ignored.

5. Bounded rationality: the limits of knowledge and computation time influence agent strategies and hence the entire mechanism.

The main research methodology is to develop and implement methods that address in particular issues of incentive-compatibility and security against collusion. For experimental evaluation of the methods we propose to develop working prototypes and validate them. An iterative development process could be employed in which the experimental results would be used to refine the methods and start a new iteration until a stable point is reached in which no further refinement is needed. Our goal is not only to gain an understanding of the possibilities and issues regarding reputation mechanisms in distributed environments, but also to propose algorithms and evaluate them on implementations and with human participants.

4.2 Modeling and Implementation Techniques

Simple, binary feedback mechanisms can be deployed in many environments. They are intuitive, easy to use, and require the least effort from the reporter. We will develop theoretical models for aggregating binary feedback into semantically well defined reputation information.

We will proceed by decomposing the problem in two distinct steps. First, we will study possible models of binary reputation mechanisms when reporting agents are cooperative (i.e. they do not explicitly try to manipulate the reputation mechanism), however not entirely reliable (i.e. they can make unintentional mistakes when submitting feedback). For this step, the research methodology is the following:

1. Assume a setting in which reporters always tell the truth and undistorted information is available to the reputation mechanism.

2. Derive feedback aggregation and trust decision rules that assign a value to a feedback report such that the momentary gain obtained from cheating is offset by the loss due to negative feedback.

3. Study the effect of mistakes and imperfect information on the value of a reputation report.
4. Validate the theoretical results in 2 and 3 against a simulated environment in which reporters make unintentional mistakes.

Second, we will investigate the resulting models when agents strategically try to manipulate the mechanism. The objective of this second step is to enhance reputation mechanisms with interaction protocols that make them incentive-compatible (i.e. rational agents have the incentive to report the truth) and secure against manipulation by other agents.

Due to computational limitations or to the inherent uncertainty of the environment, the behavior of many service providers (trusted agents) can be modelled by Markov Chains [51]. For such behavior models we will develop interaction protocols based on side-payments and cryptographic methods that (1) make it in the best interest of the reporting agents to file true feedback and (2) make it impossible (i.e., sufficiently expensive) for agents to manipulate the reputation of any single provider. The methodology used for achieving this goal involves iteratively:

- Designing interaction protocols achieving the desired properties.
- Validating them in simulated distributed environments in which mistakes and failures can occur.

The extension to finer grained feedback will be addressed in future work. However, we do not expect major issues emerging from this added freedom in reporting feedback. There are two reasons for this [19]:

1. Any finer grained feedback value can be expressed as a set of binary feedback reports (e.g. feedback values from 1 to 8 can be expressed as sets of 3 binary reports)
2. Fundamental research in reputation mechanisms proves that binary reputation mechanisms are as efficient as any other finer-grained mechanisms

Chapter 5

Reputation-based Pricing of Web Services

Most of the existing reputation mechanisms (eBay, Slashdot, Amazon) use the average of past feedback reports to assess the reputation of one agent. Clear semantics is not attributed to reputation information and clients use threshold rules (i.e. buy if reputation is above a certain threshold) to take trusting decisions. Such systems tend to separate agents in two categories (trustworthy or untrustworthy) and do not leave much room in between.

Black and white mechanisms cannot work in a service-oriented environment. The perfect service is impossible to provide (or prohibitively expensive), and various providers might prefer different production quality levels. Such settings require a more flexible mechanism in which quality of service can be traded, for example, with the price: i.e. clients consciously accept a lower service quality level for a lower price [31].

In this chapter we show how a very simple mechanism (based on averaging feedback) can also work in a service-oriented environment. Repeated failures do not automatically exclude a provider from the market (as they would on eBay), but rather influence the price the provider can charge for a future service. Service providers can thus rationally plan how to invest available resources in order to maximize their (and the social) revenue. Such a reputation mechanism does not act by social exclusion but rather through incentive-compatible service level agreements.

Before we go into the details of the mechanism, let us consider a practical example. Consider a service which provides computation power: clients submit processing jobs and wait for the answer. A client is satisfied given that the correct[1] answer is provided before a certain fixed timeout. We assume that an incorrect answer (or a late answer) does not have any value for the clients.

[1] we assume that the client can easily verify if the answer is correct or not. e.g. the solution of a constraint satisfaction problem can be easily verified for correctness.

The probability with which request are satisfied directly depends on the capacity of the provider, and on the number of service requests in a given "day".[2] Given that environmental perturbations are small, the provider can fine-tune in every day the expected success rate of the provided service, by deciding what computing resources to allocate, or how many client requests to accept.

We start from the assumption that the perfect service is impossible to provide (i.e. has infinite cost). Unless the provider has unlimited resources, there always is a small chance of service failure due to a conflict of resources. On the other hand, clients can be happy with less than perfect service if the tradeoff between price and quality is right. When the provider has an idea about the price curve of the clients (i.e. how much clients are willing to pay for various rates of successful service) it can find the optimal success rate that should be guaranteed: i.e. the success rate q^* which maximizes the total reward: $price(q) - cost(q)$. q^* is the socially efficient "quality" level of the provider.

Different providers have different characteristics (i.e. "types") and therefore, different cost functions, and different efficient quality levels. It is usually the case that the set of efficient quality levels of all types spans a continuous interval between q_{min} and q_{max}, the generally accepted limits in which the success rate can vary.

The role of the reputation mechanism is to drive every provider (regardless of its type) to provision the optimal success rate, despite the temptation to cheat. We show how a simple mechanism based on averaging past feedback and reputation-based service level agreements can push the market towards an efficient equilibrium point.

Section 5.1 formally presents the setting in which we situate our work. A detailed description of the reputation mechanism, as well as proofs about its properties is presented in Section 5.2. Section 5.3 comments on the validity of the assumptions that we make, and discusses practical issues regarding the implementation of our mechanism in real settings. We conclude by presenting related and future work.

5.1 The Model

We consider a distributed system in which rational service providers (sellers) repeatedly offer the same service to the interested clients (buyers), in exchange for payment. We assume that service requests are uniformly distributed over a (possibly infinite) number of "days", with the understanding that our "day" can have any pre-established length of physical time.

Clients are risk-neutral[3] and can observe only two quality levels: "satisfactory" quality

[2] We consider the "day" to be the atomic decisional time unit of the service provider. Our "day" can in fact represent a second, an hour or a week of physical time.

[3] A risk-neutral measure is a probability measure in which today's fair (i.e. arbitrage-free) price of a derivative is equal to the discounted expected value (under the measure) of the future payoff of the derivative.

for a successful invocation or "unsatisfactory" quality for a failure. A satisfactory service has value v to the clients, while unsatisfactory service is worthless.

The observation of the clients depends on the effort exerted by the provider, and on external factors characterizing the environment (e.g. network or hardware failures). External factors are assumed to be "small enough", and time-independent. Effort is expensive, however, it positively impacts the probability that clients experience a satisfactory service.

Providers have different characteristics (cost functions, capabilities, etc) which define their *type*. We denote the set of possible types as Θ; members of this set are denoted as θ. Excepting environmental factors, the type of the provider, and the effort invested in providing the service, completely determine the probability that the clients will be satisfied.

Let $c : [q_{min}, q_{max}] \times \Theta \to \mathbb{R}$ describe the cost function of service providers, such that $c(q, \theta)$ is the cost of the effort that needs to be exerted by a type θ seller in order to provide a satisfactory service with probability q. q can be viewed as a measure of the "production quality"; thus, the experience of the clients is an imperfect discrete observation of the continuous production quality q. As it is the case for most computational services, higher probability of success is increasingly more expensive to guarantee: i.e. $c_1(q, \theta) > 0, c_{11}(q, \theta) > 0$, for all $\theta \in \Theta$.[4]

We assume that the cost function can be piecewise linearized on the set of intervals $\{[q(\theta)_i, q(\theta)_{i+1}]\}$, $i = \{0, \ldots, N(\theta)\}$, such that: $q(\theta)_0 = q_{min}$, $q(\theta)_{N(\theta)} = q_{max}$ and $q(\theta)_i < q(\theta)_{i+1}$ for all i and all θ. The cost function can thus be approximated by:

$$c(q, \theta) \simeq c_0(\theta) + \sum_{i=1}^{t-1} M(\theta)_i \big(q(\theta)_{i+1} - q(\theta)_i\big) + M(\theta)_t \big(q - q(\theta)_t\big)$$

when $q(\theta)_t < q < q(\theta)_{t+1}$. The marginal costs $M(\theta)_i$ become increasingly bigger for all seller types, and we assume that each provider can choose each day the effort it wants to exert the next day. However, all requests coming during one day are satisfied with the same "quality" (i.e. probability of success) q.

When clients have perfect information about the service provider (i.e. know its type and the exerted effort level in one particular day), they can deduce the probability q with which they are going to experience a satisfactory service, and therefore will be willing to pay $q \cdot v$ for that service. Perfect information also drives providers to exert the socially efficient (i.e. which maximizes social welfare) effort level. For each type θ, the socially efficient effort level characterizes the quality maximizing the revenue: $q \cdot v - c(q, \theta)$. We call $q^*(\theta) = argmax(q \cdot v - c(q, \theta))$ the *efficient production quality level*. Figure 5.1 shows the price function of clients, an example of a cost function of a certain type θ, and the corresponding efficient production quality level of the type θ.

[4] $c_1(\cdot, \cdot), c_2(\cdot, \cdot)$ denote the first order partial derivatives of $c(\cdot, \cdot)$ with respect to the first, and respectively the second parameter; $c_{11}(\cdot, \cdot), c_{12}(\cdot, \cdot), c_{22}(\cdot, \cdot)$ denote second order partial derivatives.

However, perfect information is not available and therefore the expected utility of a service invocation can be estimated only from the previous behavior of the provider. For that purpose, a reputation mechanism collects feedback from the clients about the quality of service of different providers. Feedback is binary (1 for satisfactory service, 0 otherwise) and aggregated on a daily basis.

Let R_t be the reputation of a service provider at the beginning of day t, and let r_t be the set of feedback reports submitted about the same provider during the day t. The reputation mechanism computes the new reputation of the provider for day $t+1$ as $R_{t+1} = f(R_t, t, r_t)$ where f is some time-independent function.

The clients use a decision rule which allows them to compute the price they should pay in day t for a service provided by a seller whose reputation is R_t. If $p(R_t)$ defines this rule, a rational provider whose type is θ will optimally chose the efforts exerted in each day, and therefore obtains a lifetime revenue:

$$V(R_0, \theta) = \max_{(q_t)} \sum_{t=0}^{\infty} \delta^t (p(R_t) - c(q_t, \theta));$$

where δ is the daily discount factor[5] of the provider.

The remaining question is (1) how to chose the reputation updating function f, and (2) what price recommendation to give to the clients (the payment rule $p(R_t)$ such that:

- providers have the incentive to exert the effort level which maximizes social revenue;
- clients do not have the incentive to deviate from the recommendations of the reputation mechanism.

5.2 Reputation Mechanism

The reputation mechanism we propose is very simple:

1. the reputation of a service provider in day t is the probability of a successful service invocation in day $t - 1$.
2. The service level agreement in day t should specify a price corresponding to a quality level equal to the reputation in day t.

[5] the daily discount factor models the fact that future revenues are less valuable to an agent in the present. In other words, any user would choose to have 1 dollar today rather than tomorrow. The rate of depreciation (or conversely the rate of interest yielded by presently available funds) is reflected by δ. (e.g. 1 dollar received by an agent tomorrow values only $\delta \leq 1$ dollars today)

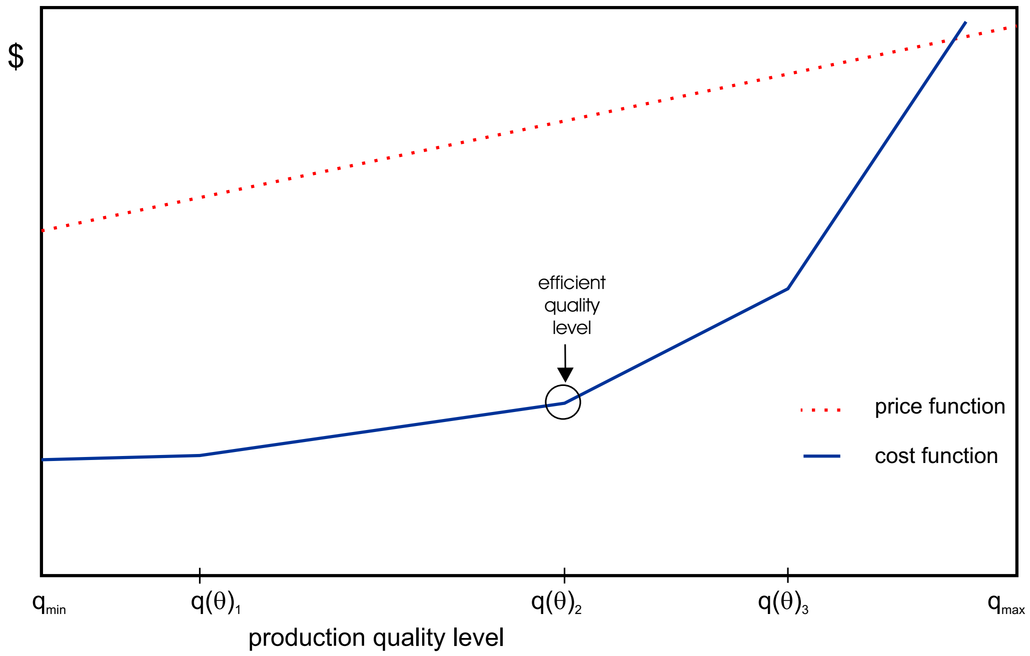

Figure 5.1: Example of price function and cost function for a service provider.

Formally,

$$\begin{aligned} r_t &= \frac{\textit{nr of positive reports in day t}}{\textit{total nr of reports from day t}}; \\ R_{t+1} &= f(R_t, t, r_t) = r_t; \\ p(R_t) &= R_t \cdot v; \end{aligned}$$

Given that clients report the truth, and that "enough" clients submit feedback in day t, by the law of large numbers, the value of r_t is equal to q_t, the production quality chosen by the provider for day t. However, mistakes in reporting, as well as environmental factors influencing the experience of clients introduce a noise in the value of r_t. To account for this influence , we assume that the value of r_t is normally distributed around q_t with the variance σ. σ can be approximated by the system designer from previous experience, and as long as it is small enough it doesn't influence the properties of the mechanism.

The goal of this reputation mechanism is to determine providers to invest the effort level which makes the market socially efficient. Proposition 1 establishes that when the daily discount factors are greater than a given threshold, the market is socially efficient. We use the notion of Nash Equilibrium, which captures a steady state of the play of a strategic game in which each player holds the correct expectation about the others' behavior and acts rationally. It does not attempt to examine the process by which a steady state is reached.

Proposition 1 *If* $\delta > \frac{M^*(\theta)}{v}$, *(where* δ *is the daily discount factor, and* $M^*(\theta)$ *is the* *marginal cost of a provider of type θ when exerting an effort smaller than the efficient one) the above reputation mechanism has a Nash equilibrium in which the clients pay $p(R_t)$ and the provider exerts the socially efficient effort level in every day.*

PROOF. Let us consider a provider of type θ. The efficient production quality level $q^*(\theta) = argmax(p(q) - c(q,\theta))$ is the point $q(\theta)_{\hat{i}}$ such that $M(\theta)_{\hat{i}} \leq v$ and $M(\theta)_{\hat{i}+1} > v$. This point is unique if the inequality is strict.

The marginal cost of exerting an effort smaller than the efficient one is $M^*(\theta) = M(\theta)_{\hat{i}}$.

Assuming that clients follow the recommendation of the reputation mechanism (i.e. pay the price corresponding to the probability of success of the previous day), the lifetime revenue of the service provider is:

$$\begin{aligned} V(R_0,\theta) &= \max_{(q_t)} \sum_{t=0}^{\infty} \delta^t \big(p(R_t) - c(q_t,\theta)\big); \\ &= \max_{q_0} \big(p(R_0) - c(q_0,\theta)\big) + \delta E\big[V(R_1,\theta)\big]; \end{aligned}$$

where the expectation is with respect to possible values of R_1. By generalizing the above relation for the continuation payoff expected by a type θ provider from day t onward (and including day t) if its reputation is R_t:

$$\begin{aligned} V(R_t,\theta) &= \max_{q_t}\big(p(R_t) - c(q_t,\theta)\big) + \delta E\big[V(R_1,\theta)\big]; \\ &= \max_{q_t}\big(p(R_t) - c(q_t,\theta)\big) + \delta E_{r_t}\Big[V\big(f(R_t,t,r_t),\theta\big)\Big]; \\ &= \max_{q_t}\big(p(R_t) - c(q_t,\theta)\big) + \delta E_{r_t}\big[V(r_t,\theta)\big]; \end{aligned}$$

Given that r_t is normally distributed around q_t, we have $r_t = q_t + z$ where z is the noise, normally distributed around 0 with variance σ. Therefore:

$$V(R_t,\theta) = \max_{q_t}\big(p(R_t) - c(q_t,\theta)\big) + \delta \int_Z V(q_t + z,\theta)\cdot \pi(z)dz; \tag{5.1}$$

where $\pi : Z \rightarrow \mathbb{R}$ is the probability distribution function of the noise z (i.e. the normal pdf).

Let q^* the optimal control satisfying equation (5.1). We than have:

$$\begin{aligned} \big(p(R_t) - c(q^*,\theta)\big) + \delta \int_Z V(q^* + z,\theta)\cdot\pi(z)dz \leq \\ \big(p(R_t) - c(q^* + \delta q,\theta)\big) + \delta \int_Z V(q^* + \delta q + z,\theta)\cdot\pi(z)dz; \end{aligned} \tag{5.2}$$

for all possible deviations δq. Since the price and the cost functions are linear, equation (5.2) can be rewritten as:

$$-c_1(q^*,\theta)\cdot\delta q+\delta\int_Z V_1(q^*+z,\theta)\delta q\cdot\pi(z)dz\leq 0; \tag{5.3}$$

Since $V_1(R_t,\theta)=\frac{\partial(p(R_t)-c(q_t,\theta)}{\partial R_t}=v$ (see [51], page 85), for all R_t, and since $\int_Z\pi(z)dz=1$, equation (5.3) becomes:

$$-c_1(q^*,\theta)\cdot\delta q+\delta\cdot v\cdot\delta q\leq 0; \tag{5.4}$$

When $\delta>\frac{M^*(\theta)}{v}$, relation (5.4) can only be satisfied in $q^*=q(\theta)_{\hat{i}}$ since:

$$c_1\big(q(\theta)_{\hat{i}},\theta\big)-\delta\cdot v=M(\theta)_{\hat{i}}-\delta\cdot v\leq 0;$$

when $\delta q<0$, and

$$-c_1\big(q(\theta)_{\hat{i}},\theta\big)+\delta\cdot v=-M(\theta)_{\hat{i}+1}+\delta\cdot v\leq 0;$$

when $\delta q>0$.

Given that the provider always produces at the same quality level (i.e. the efficient one) an estimate of the production quality of day t is also a valid estimate for the production quality of day $t+1$. When the number of reports collected in one day is big enough, the sum of positive reports is a "sufficient" statistics for the real production quality. Therefore, in the absence of private information, it is rational for the clients to follow the recommendation of the reputation mechanism.

As a consequence, the reputation mechanism described in Section 5.2 has a Nash equilibrium in which providers exert the socially efficient effort, and clients follow the recommendation of the mechanism. Q.E.D. □

In other words, all service providers types will exert the socially efficient effort given that (a) they are patient enough (i.e. value enough future revenues), and that (b) rational clients will consider the reputation when estimating the price they are willing to pay for the service. Moreover, the resulting service level agreement is incentive-compatible: neither the provider, nor the client have the incentive to deviate from the agreement.

5.3 Discussion and Future Work

The mechanism presented above describes a pricing rule based on reputation which can be used by rational clients in order to incentivize service providers of all types to exert the socially efficient effort level. Possible applications of such a mechanism are numerous, therefore this section discusses practical issues which might occur in the implementation of such a mechanism.

The mechanism has very little memory requirements. The reputation of a provider "survives" only one day, and therefore does not need to be protected in a usually highly volatile distributed system. The reputation mechanism is also resistent to failures. Given the simple aggregation rule (percentage of positive reports) a certain fraction of the votes can be lost without greatly impacting the performance of the mechanism (i.e. if failures are sufficiently small and uncorrelated with the value of the report, there is no perturbation in the functioning of the mechanism).

Reporting mistakes and interferences due to environmental factors are accounted for through a normally distributed noise variable which affects the way reputation is updated from one day to the next. While our main result is independent of these perturbations, we acknowledge that real noise can have a significant impact on the equilibrium strategies. This problem is more stringent for service providers whose efficient production quality level is close to the maximum one, q_{max}. For such types, the noise can only bias downwards the reputation, and therefore, the optimal strategy is to exert less effort than the efficient one. Moreover, the noise does impact the actual revenues of service providers (but not their strategies).

From our presentation, the reputation mechanism acts as a central entity collecting feedback and disseminating reputation information. However, the implementation of the mechanism does not have to be centralized. Semi-centralized (in which replicated, specialized agents implement parallel reputation mechanisms) or fully decentralized (in which feedback is broadcasted and privately processed by each peer) systems are perfectly viable. The essence of the mechanism is that it allows rational agents (clients and providers) to act only based on local information (the feedback from the previous day). This model is perfectly compatible with a web services setting.

One important assumption that we make when proving Proposition 1 is that clients report the truth. In real applications feedback can be distorted by the strategic interests of the reporters. Providing feedback may be expensive, and conflicts of interest could make the report unreliable. Fortunately, incentive-compatible schemes (i.e. they make it rational for clients to report the truth) exist when the report of one agent can be statistically correlated with reports coming from other agents [39, 25]. These schemes are not usually applicable when the provider behaves strategically. However, the equilibrium strategy (providers always exert the same effort level) of our particular reputation mechanism allows the use of such schemes.

Another problem is the resistance to collusion. Returning agents could collude and submit negative feedback in order to pay smaller prices in the following day. While the solution to this problem is far from clear, we argue that such coalitions are not stable when services are scarce: colluding agents will afford to pay more than the price dictated by the reputation mechanism in order to obtain the service. This question needs however a deeper investigation in our future work.

We leave it up to the mechanism designer to estimate the physical length of one “day”. Longer periods of time might violate the assumption that the provider cannot change its behavior during one day, shorter periods on the other hand, might not alow to gather enough binary reports in order to have a reliable estimation of the production quality level. In the same time, for shorter days, the impendency assumption on failures can be violated. At a macroscopic level, failures are correlated: the failure of the present invocation most likely attracts the failure of the next invocation. Only when averaged over longer periods of time failures can be regarded as random.

Real users might not be risk-neutral. Risk-seeking or risk-averse behavior implies non-linear price function. The analysis of equilibrium behavior becomes much more complicated for non-linear functions. As future work, we’ll investigate how piecewise linearized price function affect equilibrium strategies.

Real clients also have different valuations for the service. To a certain extent, our mechanism tolerates such settings. When clients have different valuations, the value v used to specify the service level agreement actually represents an average over the valuations of the possible clients. This value can be computed by the provider itself based on market statistics. When the estimation of the provider is accurate enough, we conjecture that the mechanism provides the same guarantees. However, a formal characterization of this setting requires a more complex model: one in which clients decide to buy or not a service at a certain price, based on their private preferences and on the reputation of the provider. We plan to analyze this model as future work.

Since there is no restriction on the number of intervals on which the cost function can be piecewise linearized, we do not regard this assumption to be restrictive. However, as the number of linear segments increases, the difference between two successive marginal cost constants ($M(\theta)_i$, $M(\theta)_{i+1}$) decreases, and therefore small perturbations in the price function (see the argument in the previous paragraph) can trigger significant changes in the exerted effort level.

The equilibrium strategy enounced by Proposition 1 is probably not unique. Other Nash equilibria might exist which do not have the desired properties. As future work, we’ll search and try to eliminate these “undesired” equilibrium points.

One final remark raises the question of human rationality. The desired equilibrium behavior can emerge only if clients and providers try to maximize their revenues. This is not always true with human users. However, in the future online market driven by agents and web services, the human intervention will be minimized, and rational strategies will predominate. This is the kind of environment for which we target our work.

5.4 Conclusion

We have described how a simple reputation mechanism allows a web service-oriented market to function efficiently. Reputation information is computed using an average-based aggregation rule, and drives service providers of different types to exert the socially efficient effort levels. The basic principle behind the mechanism is not social exclusion (separate and exclude "untrustworthy" agents) but rather incentive-compatible service level agreements which trade quality for price.

We have also discussed practical issues arising from the implementation of our mechanism in real applications. While acknowledging the limitations of our model, we believe that such mechanisms can bring significant improvements in decentralized service-oriented computing systems.

Chapter 6

Honest Reporting Incentive

The reputation mechanisms described in the previous section requires that agents report honestly for the system to work. However obtaining honest feedback from self-interested agents is not a trivial problem. Mechanisms based on side-payments can be conceived such that honest reporting becomes rational (i.e. Nash equilibrium). Unfortunately, for every incentive-compatible Nash equilibrium there seems to also be a dishonest Nash equilibrium strategy that sometimes is more attractive. In this section we analyze three incentive-compatible reputation mechanisms and investigate how what can be done to encourage rational agents to report truthfully.

The first two are based on side-payments schemes (The MRZ mechanisms, described by Miller, Resnick and Zeckhouser in [39] and the JF mechanism described by Jurca and Faltings in [25], [30]) that correlate a present report with future reports submitted about the same agent. The last mechanisms [28] discovers (in equilibrium) the true outcome of a transaction by analyzing the two reports coming from the agents involved in the exchange.

6.1 The MRZ Incentive Compatible Reputation Mechanism

In [39], Miller et al. consider that a number of agents sequentially experience the same service whose *type*[1] is drawn from a set of possible types T.[2]

[1] The type of a service defines the totality of relevant characteristics of that service. For example, quality, and possibly other attributes define the type of a product.

[2] The set of possible types is the combination of all values of the attributes which define the type. While this definition generates an infinite-size set of types, in most practical situations, approximations make the set of possible types countable. For example, the set of possible types could have only two elements: *good* and *bad*. This implies that there is common understanding among the agents in the environment that the service can be exhaustively classified as good or bad: i.e. no other information about the service is relevant for the decision taken by the agents in that environment.

The real type of the service does not change during the experiment, and is not known by the agents. However, after every interaction, the agent receives one signal s (from a possible set of signals S of cardinality M) about the type of the service. For a certain product type $t \in T$, the signals perceived by the agents are independently identically distributed such that the signal s_i is observed with probability $f(s_i|t)$ for all $s_i \in S$. $\sum_{s_i \in S} f(s_i|t) = 1$ for all $t \in T$.

After every interaction, the participating agent is asked to submit feedback about the signal she has observed. The reputation mechanism collects the reports, and updates the reputation of the product. Reputation information consists of the probability distribution over the possible types of the service. Let p be the current belief of the reputation mechanism (and therefore of all agents that can access the reputation information) about the probability distribution over types of the service. $p(t)$ is the probability assigned by the current belief to the fact that the service is of type t, and $\sum_{t \in T} p(t) = 1$. When the reputation mechanisms receives a report $r \in S$, the belief p is updated using Bayes' Law:

$$p(t|r) = \frac{f(r|t) \cdot p(t)}{Pr[r]}$$

where $Pr[r] = \sum_{t \in T} f(r|t) \cdot p(t)$ is the probability of observing the signal r.

Each feedback report is compared against another future report. Let $r \in S$ be the report submitted by agent a and let r_r be the future report submitted by agent a_r which serves to assess the honesty of r. The agent a_r is called the *rater* of the agent a since the report r_r is used to "rate" the report r. Typically, the next report is used to evaluate the present one. Miller et al. show that if agent a is paid according to the scoring rule $R(r_r, r)$, she will honestly report her observation given that a_r also honestly reports his observation. The three best known scoring rules are:

1. Quadratic Scoring Rule: $R(r_r, r) = 2Pr[r_r|r] - \sum_{s_h \in S} Pr[s_h|r]^2$
2. Spherical Scoring Rule: $R(r_r, r) = \frac{Pr[r_r|r]}{\left(\sum_{s_h \in S} Pr[s_h|r]^2\right)^{1/2}}$
3. Logarithmic Scoring Rule: $R(r_r, r) = \ln Pr[r_r|r]$

where $Pr[s_h|r] = \sum_{t \in T} f(s_h|t) \cdot p(t|r)$ is the posterior probability that the signal s_h will be observed, as known by the reputation mechanism immediately after r has been reported.

To illustrate how this mechanism works, let us consider that a service can have two possible types, i.e., *good* (G) or *bad* (B). Buyers can observe two signals, $+$ or $-$ such that the distribution of signals conditional of the type of the service is: $f(+|G) = 0.9$, $f(-|G) = 0.1$, $f(+|B) = 0.15$, $f(-|B) = 0.85$. Let us assume that the current belief about the type of the service assigns probability 0.4 to the service being *good* and 0.6 to the service being *bad*. i.e. $p(G) = 0.4$ and $p(B) = 0.6$.

Let us assume that the agent a has the next interaction with the service, and that she has observed a $+$. Figure 6.1 shows how the side-payments are computed, and how beliefs are updated if a reports $+$ or $-$.

<table>
<tr><th>a <b>reports</b> $+$</th><th>a <b>reports</b> $-$</th></tr>
<tr><td colspan="2">Beliefs of a regarding the posterior distribution over types</td></tr>
<tr><td colspan="2">$p(G|+) = \frac{f(+|G) \cdot p(G)}{f(+|G) \cdot p(G) + f(+|B) \cdot p(B)} = 0.8;$
$p(B|+) = 1 - p(G|+) = 0.2$</td></tr>
<tr><td colspan="2">Beliefs of the Reputation Mechanism regarding the posterior distribution over types</td></tr>
<tr><td>$p(G|+) = \frac{f(+|G) \cdot p(G)}{f(+|G) \cdot p(G) + f(+|B) \cdot p(B)} = 0.8;$
$p(B|+) = 1 - p(G|+) = 0.2$</td><td>$p(G|-) = \frac{f(-|G) \cdot p(G)}{f(-|G) \cdot p(G) + f(-|B) \cdot p(B)} = 0.07;$
$p(B|-) = 1 - p(G|-) = 0.93$</td></tr>
<tr><td colspan="2">Beliefs of a regarding the distribution of signals received by a_r</td></tr>
<tr><td colspan="2">$Pr[+|+] = f(+|G) \cdot p(G|+) + f(+|B) \cdot p(B|+) = 0.75$
$Pr[-|+] = 1 - Pr[+|+] = 0.25$</td></tr>
<tr><td colspan="2">Beliefs of the Reputation Mechanism regarding the distribution of signals received by a_r</td></tr>
<tr><td>$Pr[+|+] = f(+|G) \cdot p(G|+) + f(+|B) \cdot p(B|+) = 0.75$
$Pr[-|+] = 1 - Pr[+|+] = 0.25$</td><td>$Pr[+|-] = f(+|G) \cdot p(G|-) + f(+|B) \cdot p(B|-) = 0.2$
$Pr[-|-] = 1 - Pr[+|-] = 0.8$</td></tr>
<tr><td colspan="2">Payment made to a (Using spherical scoring rule)</td></tr>
<tr><td>$R(+,+) = \frac{Pr[+|+]}{\sqrt{Pr[+|+]^2 + Pr[-|+]^2}} = 0.95 \text{ if } r_r = + \; R(-,+) = \frac{Pr[-|+]}{\sqrt{Pr[+|+]^2 + Pr[-|+]^2}} = 0.32 \text{ if } r_r = -.$</td><td>$R(+,-) = \frac{Pr[+|-]}{\sqrt{Pr[+|-]^2 + Pr[-|-]^2}} = 0.24 \text{ if } r_r = +$
$R(-,-) = \frac{Pr[-|-]}{\sqrt{Pr[+|-]^2 + Pr[-|-]^2}} = 0.97 \text{ if } r_r = -.$</td></tr>
<tr><td colspan="2">Expected payment to a</td></tr>
<tr><td>$E_{s_j \in \{+,-\}}[R(s_j|+)] = Pr[+|+] \cdot R(+,+) + Pr[-|+] \cdot R(-,+) = 0.79$</td><td>$E_{s_j \in \{+,-\}}[R(s_j|-)] = Pr[+|+] \cdot R(+,-) + Pr[-|+] \cdot R(-,-) = 0.42$</td></tr>
</table>

Figure 6.1: Updating of beliefs, and computation of side payments according to the MRZ mechanism, given that a has observed a $+$.

Given that a_r reports the truth, a maximizes her expected payoff by also reporting the truth. The same would be true if a had observed a $-$ rather than a $+$. Miller et al. show that in every situation (for every signal observed, for every belief about the service, and for all generic distributions of signals conditional on types) it is better for a to report the truth, given that a_r also reports the truth. Honest reporting is therefore a Nash equilibrium.

6.2 The JF Incentive-Compatible Reputation Mechanism

The MRZ mechanism can be easily adapted to a variety of contexts. However, it assumes (1) common knowledge about the distribution of signals conditional on types and (2) lack of private information.

Since the MRZ mechanism is based on side-payments which are computed using scoring rules, the agents always have the incentive to approximate as good as possible the signal that will be observed by the rater. When the two assumption above are satisfied, the incentive to provide the best approximation for the signal received by the rater coincides with honestly reporting the observed signal. However, when the reporting agent and the reputation mechanism have different views of the world (i.e. different beliefs about the service), the agent can manipulate her report depending on her private beliefs (about the service and about the beliefs of the reputation mechanism).

The JF mechanism eliminates this drawback at the expense of limiting the contexts in which the incentive-compatible property holds. The model used by Jurca and Faltings in [25] is that of a service having a "dynamic type". The signals perceived by the agents do not only depend on the type of the service, but also on temporary information. The model adopted for the probability distribution of signals is that of a Markov chain of variable length, and the possible set of signals consist of only two values $+$ and $-$.

The side-payment for reports follows a very simple rule, and does not depend on the beliefs of the agent or of the reputation mechanism. A report is paid only if the next report submitted about the same service has the same value. The amount of the payment is dynamically scaled such that the whole mechanism is budget-balanced.

The Markov model for the observable signals is very appropriate for services offered by software agents. Let us recall the service used in section 6.1, and let us also consider that the service is provided by a software agent (i.e. a webservice). One possibility is to consider that the webservice is always providing the same service (*good* or *bad*) and that the agents perceive the signals $+$ and $-$ with the probabilities: $f(+|G), f(-|G), f(+|B)$ and $f(-|B)$. However, if the *good* and *bad* types are interpreted as successful, respectively defective service (and the $+$ and $-$ signals are interpreted as satisfactory, respectively unsatisfactory answers, perceived with some inherent noise) it is more realistic to assume that failures are correlated. Intuitively, the failure of the present invocation is an indication of exceptional conditions (hardware failure, blocks in the software, overload, etc) and therefore is likely to influence the result of the next invocation. For example, a failure of the present invocation due to hardware problems indicates a big probability of failure for the next invocation as well. On the contrary, present failure due to an overload might indicate a bigger probability of success for the next invocation.

While the MRZ mechanism can easily be adapted for Markov models of behavior, it requires that the model be common knowledge among the agents: i.e. all agents must agree on the length of the model and on the fact that there is a unique set of parameters characterizing that model. By having side-payments that do not depend on the beliefs of the agents, the JF mechanism allows the agents to have any private beliefs about the model of the webservice, as long as these beliefs satisfy some general constraints. Of course, the freedom of having private beliefs is paid by the constraints which must be satisfied, that limit the contexts in which incentive-compatibility is guaranteed.

6.3 "CONFESS" Reporting Scheme

As opposed to side-payment schemes that correlate a present report with future reports submitted about the same agent, we present a mechanism that discovers (in equilibrium) the true outcome of a transaction by analyzing the two reports coming from the agents involved in the exchange. For two long-run rational agents, we show that it is possible to design such a mechanism that makes cooperation a stable equilibrium.

6.3.1 Assumptions

We consider an environment in which the following assumptions hold:

- A rational seller interacts repeatedly with several rational buyers by trading one product of value v_i in each round i. The values $v_i \in (\underline{v}, \overline{v})$ are randomly distributed according to the probability distribution function ϕ [3];

- All transactions have a fixed profit margin equal to $(\rho_B + \rho_S)v_i$, where $\rho_S v_i$ is the profit of the seller and $\rho_B v_i$ is the profit of the corresponding buyer;

- All buyers are completely trustworthy: i.e. Each buyer first pays the seller and then waits for the seller to ship the product. The seller may defect by not shipping the promised product, and the buyer perfectly perceives the action of the seller;

- There is no independent verification authority in the market, i.e. the behavior of the seller in round i is known only to the seller himself and the buyer with which he traded in that round;

- The seller cannot refuse the interaction with a specific buyer, and can trade with several buyers in parallel. A buyer can however end the interaction with the seller and choose to buy the goods from a completely trusted seller (e.g. a brick and mortar shop) for an extra cost representing a percentage (θ) of the value of the item bought. Once a buyer decides to terminate a business relationship with the seller,

[3] Following the same argumentation proposed in [17], this model is valid for settings where the act of accumulating inventory is independent from that of (re)selling it: e.g. a highly dynamic used car dealership.

she will never trade again in this market. The seller, however, can always find other buyers to trade with.

- The buyer and the seller discount future revenues by δ_B and δ_S respectively. The discount factors also reflect the probability with which the agents are going to participate to the next transaction. $0 < \delta_S, \delta_B < 1$, and $\delta_S >> \delta_B$ modeling the fact that the seller is likely to have a longer presence in the market than the buyer.

- The buyer and seller interact in a market (possibly a different one for each transaction) capable of charging listing fees and participation taxes.

- At the end of every transaction, both the seller and the buyer are asked to submit a binary report about the seller's behavior: a positive report, $R+$, signals cooperation while a negative report, $R-$, signals defection;

We also assume that in our environment there is a *semantically well defined, efficient* Reputation Mechanism (RM). Reputation is semantically well defined when buyers have exact rules for aggregating feedback into reputation information and for making trust decisions based on that reputation information. These rules determine sellers to assign a value to a reputation report ($R+$ or $R-$), reflecting the influence of that report on future revenues. RM is efficient if the values associated by sellers to reputation reports are such that in any transaction the seller prefers to cooperate rather than defect. If $V(R+, v)$ and $V(R-, v)$ are the values associated by the seller to the positive respectively the negative reputation report generated after a transaction of value v, we have: *V($R+, v$) + Payoff(cooperate,v)* > *V($R-, v$) + Payoff(defect,v)*[4]. A simple escrow service or Dellarocas' Goodwill Hunting Mechanism [17] satisfy these properties.

As the influence of reputation on the seller's future revenues can be isolated into a concrete value for each reputation report, every interaction between a seller and a particular buyer can be strategically isolated and considered independently. A rational seller will maximize his revenues in each such isolated interaction.

When perfect feedback (i.e. true and accurate) is available, a *well-defined, efficient* RM is enough to make rational sellers cooperate. Unfortunately, perfect feedback cannot be assumed. In the absence of independent verification means, we can only rely on the subjective reports submitted by the agents involved in the transaction; reports which are obviously biased by the strategic interests of the agents.

In the rest of this section we describe a mechanism that in equilibrium obtains true feedback about the outcome of the transaction by correlating the seller's and buyer's reports about that transaction.

[4] as an abuse of notation, we will sometimes use $V(R+, v) = V(R+)$ and ignore the fact that the value of a reputation report also depends on the value of the product.

6.3.2 The Mechanism

Every round i, a seller offers for sale a product of value v_i. The market charges the seller a listing fee ε_S, and advertises the product to the buyer. The buyer pays a participation tax ε_B, to the market, and the price v_i to the seller. If the seller cooperates, he ships the product directly to the buyer; otherwise the seller keeps the payment for himself and does not ship the product. After a certain deadline, the transaction is considered as over, and the market starts collecting information about the behavior of the seller. The seller is first required to submit a report. If the seller admits having defected, a negative report ($R-$) is submitted to the RM, the listing fees ε_S and ε_B are returned to the rightful owners, and the protocol is terminated. If, however, the seller claims to have cooperated, the buyer is also asked to provide a report. At this moment, the buyer can report cooperation, report defection, or she can report defection and terminate the interaction with the seller.

If the buyer reports cooperation, a positive reputation report ($R+$) is submitted to the RM, and the listing fees ε_S and ε_B are returned. If the buyer reports defection, both players will be punished as one of them is surely lying: a negative report ($R-$) is submitted to RM, and the listing fees ε_S and ε_B are confiscated. Finally, if the buyer decides to terminate the interaction, a negative report ($R-$) is submitted to RM, and the fees ε_S and ε_B are confiscated. Figure 6.2 provides a schematic description of the trading protocol of each round, i.

1. The seller offers for sale a product of value v_i.
2. The market charges the seller a listing fee ε_S and posts the product for sale. ε_S is the lying fine imposed by the market to the seller if contradictory reports are submitted.
3. The buyer pays v_i to the seller and the tax ε_B to the market. ε_B is the lying fine imposed by the market to the buyer if contradictory reports are submitted.
4. The seller decides whether or not to ship the product (i.e. whether to cooperate or defect). If the seller cooperates, he ships the product directly to the buyer.
5. The market requests the seller to submit a binary report (c_S for cooperation or d_S for defection) about his own behavior in the current round.
6. If the seller reports d_S, a negative report $R-$ is sent to the RM, and the market pays ε_B to the buyer and ε_S to the seller. The transaction is completed.
7. If the seller reports c_S, the market asks the buyer to submit a report. The buyer can report cooperation (c_B), defection (d_B) or she can quit the game (out).
8. If the buyer reports c_B, a positive report $R+$ is sent to the RM, and the market pays ε_S to the seller and ε_B to the buyer. The transaction is completed.
9. If the buyer reports d_B, a negative report $R-$ is sent to the RM, and the market pays nothing to either the seller or the buyer.
10. If the buyer decides to quit the game, a negative report $R-$ is sent to the RM, and the market pays nothing to either the seller or the buyer.

Figure 6.2: Description of the transaction protocol.

Chapter 7

Statistical Filtering of False Reports

In this chapter we introduce a statistical filtering of false reports system, a simplified reputation mechanism that filters out malicious reports rather than providing rational incentives for truthful reporting.

7.1 Properties of the system

The key properties of the system are:

1. Correctness: the system always gives a correct reputation in spite of the presence of some noise. The reputation always reflects the real service quality level of a service provider. To filter noise the algorithm might temporarily ignore the wrong feedback reports and give reputations relying on its own past experience. However it does not signify the system will depend on the past values constantly; once it reaches the threshold of the number of the ignored reports, the algorithm should perform some measures to match the reality.

2. Resistance: the reputation system should be resistant to some malicious manipulations. The algorithm ought to have certain ways to recognize the malicious feedbacks and to try to ignore them. Therefore the reputations will not be affected easily by these exceptional feedbacks.

3. Fast convergence: once the correct value of reputation is established this algorithm is supposed to converge fast to this value and all the feedback reports having the same evaluation value will be considered. This fast convergence will also give the system a reference level to learn about the truthfulness of the following feedbacks.

4. Rapid reaction: by analyzing feedback reports the system should be aware of the change of service quality level of some agents and it ought to react as soon as

possible to this change. This kind of reaction should be done automatically by the algorithm without interaction with the agents.

7.2 Reputation Definition

The consumer feedback should reflect his satisfaction level of the service he experienced. If the quality of the service was too low then the consumer will certainly be unhappy and he should give a poor feedback to the provider but if the service quality was outstanding then the consumer should reflect his satisfaction through his feedback. The user feedback is an integer between 1 and 5. If the user is very disappointed with the service he experienced he will give a feedback of 1 but if the service was perfect then he will give a feedback of 5.

7.3 Interaction Protocol

A client request a service with some functionality using a directory service. The directory service filters and ranks matching service adverisement according to their reputation using a reputation service. After the client interacts with a service, it reports his feedback to the reputation service.

A service consumer can submit a feedback freely through our system application according to his satisfaction about the services provided to him. A feedback is regarded as some kind of service performance report and it reflects the quality level of a service provided by a service provider.

By collecting and synthesizing all these feedback reports, this algorithm establishes a reputation record for each service providing agent about his historical service performance. This means that a reputation allows characterizing the service quality level according to an agent's past behavior and it can be referenced in order to help a consumer make optimal decisions for service integration in future applications.

Once feedback reports are submitted, the system will take these reports and forward them to the reputation mechanism where it will apply the reputation algorithm to calculate the evaluation of these feedback reports and synthesize them as a reputation for each service provider. After reputations are updated, they are stored into a reputation database and the updated reputation will serve as a reference about the services levels of the agents. Thereafter when the system has to make a choice to satisfy a client's request for the following applications, it will consult these reputations to make an optimal choice.

7.4 Explanation

In order to implement this concept, the algorithm fits the submitted feedback to a gaussian model. The basic idea is to compute the interval (i.e. the trust interval) where honest feedback lies with high probability. Future reports are accepted only if they are in this trust interval.

Trust intervals are dynamic. By modifying the variance and the amount of confidence we have on the system, we can increase or decrease the trust interval. A way to make the trust interval adaptive is to modify the amount of confidence we have on the system. For an agent whose reputation was very low but has drastically increased the system will not reflect this change immediately because of resistance. The system will first think that these feedbacks are malicious values or noise and will ignore them but after some time its reliability will decrease so it will decrease its confidence level and the impact will be the increase of the trust interval. By increasing the trust interval the system ensures that after some time it will consider the points that were ignored previously and so learn the new mean. The reliability will then again increase as most of the feedbacks will be inside the trust interval so the confidence level of the system will increase and the impact will be that the trust interval will decrease. The trust interval converges to its new mean. Storing the ignored feedbacks and considering them when the trust interval increases is something very important to ensure fast convergence.

This algorithm divides the system into three phases: an execution phase, a training phase and a learning phase. The phases are very important as they allow the system to have at every step a global view of the situation. The impact is that it accelerates transition and convergence speed. Intuitively, phases provide to the system at each step a picture of the situation with different resolutions. For a local phase, the resolution is high so the system gets only a very precise view of the current system's state but for a global phase, the resolution is low so the system gets a more global but imprecise picture of the current state. At each step all the phases are computed concurrently but there is only one phase that can be called as the current phase. The current phase is the phase in which the system is standing. The current phase can be different at any step depending on the current situation. The goal of having phases is to get different level of understanding of the current context at every step.

The system has to make at every step the decision on whether it should switch to a neighbour phase or it should stay in its current phase. This decision is not always easy to make as the system cannot predict the future feedbacks. The goal would be to always ensure a high reliability but in the same time to minimize the number of ping-pong. Ping-pong is the phenomenon of having phase shifting happening too often. Too frequent ping-pong will reduce the quality of the result as the system will not be able to decide which phase it should opt as the current phase.

The proposed system implements an efficient algorithm which has four properties: correctness, resistance, fast convergence and rapid reaction. The model of the algorithm is based on a Gaussian curve. It uses the trust interval of the Gaussian curve to test the confidence of the feedback reports and to decide whether a feedback report should be taken into account or should be ignored. The algorithm ensures also adaptiveness and dynamicity. These properties ensure that this algorithm gets the opportunity to adapt itself to the current context of the environment in order to learn and train its parameters according to the change of the context. To achieve this objective a conception of phases and a phase shifting mechanism have been introduced.

Chapter 8

Framework for Reputation Mechanisms for Web Services

In this chapter we design a framework for allowing agents to query and submit reputation information and a simple approach to integrate reputation mechanisms into the process of service selection. A prototype implementation will be available as part of D2.4.6.2.

8.1 A Web Service Reputation Framework

In this section we design a reputation service, implemented as a web service, that allows agents to query the reputation of other services and to submit reports.

The reputation framework has an extensible architecture, allowing to plugin and deploy different concrete reputation mechanisms for different example scenarios. The framework provides functionality for the normal end-user agents and also for the framework administration.

8.1.1 Funcionality for the end-user agents

- SubmitReport. The user submits a reputation report. His report will be processed with the current algorithm. This service takes as input the user name of the trustor and trustee agents, a transaction id if necessary and the reputation information. The user name is required also so that the system can debit or credit the user's account (if she is a new user, the system will create her profile automatically for the current algorithm). The service returns the user updated balance.

- RequestReputationInformation. The user requests the reputation information about a trustee agent using the current algorithm. This service takes as input the user name of the trustor and trustee agents. The user name of the trustor is required so that the system can debit or credit her account (if she is a new user, the system will create her profile automatically for the current algorithm). The service returns the requested reputation information.

8.1.2 Funcionality for framework administrator

- SetCurrentAlgorithm. The administrator selects which algorithm should be the current algorithm. This service takes as input parameter the algorithm that will be set as the current algorithm. This service will load the new algorithm exactly how it was when previously loaded. The algorithm will have exactly the same state as when it was last used. The implemented algorithms will be the Statistical Filtering of False Reports, the Incentive Compatible Reputation, CONFESS and the Peer Prediction Method.
- GetAlgorithmParameters. The administrator reads the whole information about the current algorithm. This service takes no input parameter. It allows the administrator to know which algorithm is set as the current algorithm. This service also outputs the values of the parameters that are used to customize the algorithm.
- SetAlgorithmParameters. The administrator customizes or adapts the current algorithm. The input parameters depend on which algorithm is set as the current algorithm. They are listed below for the 4 previously selected algorithms.
- ShowAnalysis. The administrator triggers the state of the current algorithm. This service takes no input parameter. This service will plot the historical reputation information and the historical behaviour of the current algorithm. This service should help the administrator to make decisions on whether or not the current algorithm needs to be adapted and how to adapt it.

The set of parameters to customize the algorithm are as follows:

- Statistical Filtering of False Reports: The finite set of possible system configurations, where each configuration is defined by the reliability and the adaptiveness parameters of the system.
- Incentive Compatible Reputation: The number of last reports to consider and the cost to buy a reputation information
- CONFESS: The number of last reports to consider, the listing fee for the client and for the seller
- Peer Prediction Method: The maximum number of signals, the maximum number of types, the scoring rule (quadratic, spherical or logarithmic) and the conditional distribution of signals.

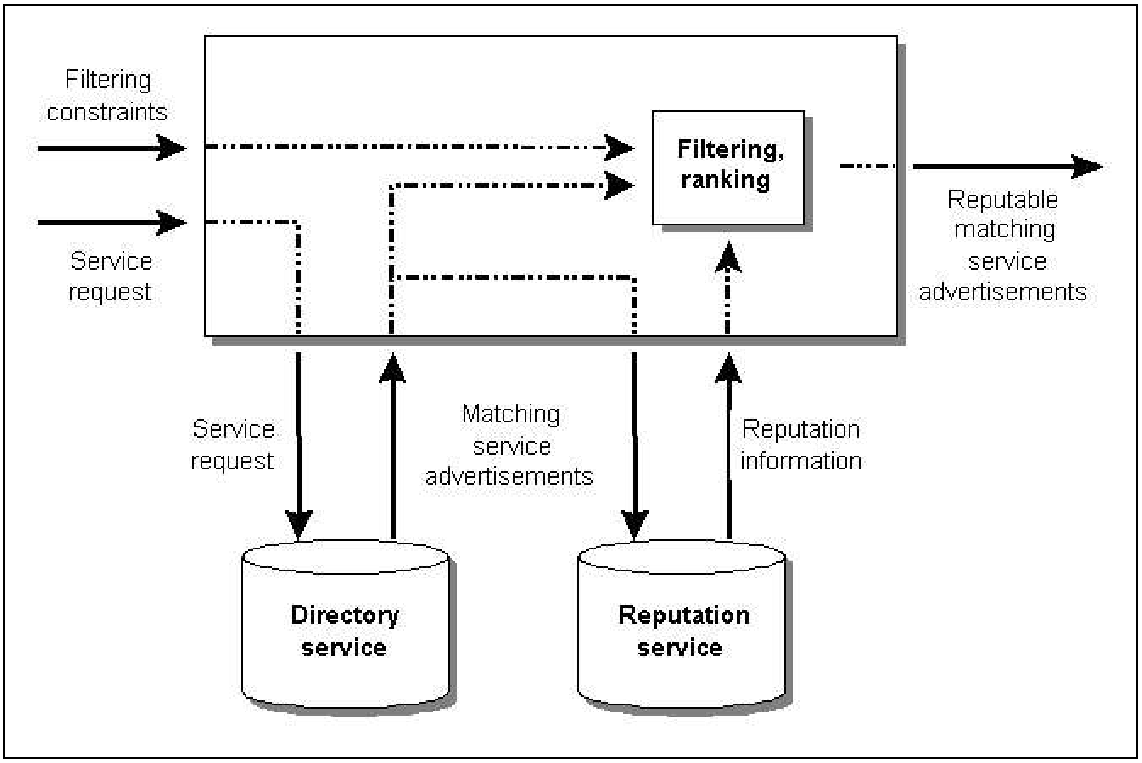

Figure 8.1: A Wrapper for service discovery filters and ranks matching service advertisements according to their reputation.

8.2 Integrating reputation mechanisms in the discovery process

In this section we discuss an extension of the integrated discovery and composition process which takes into account reputation aspects in the selection of web services to be composed. We outline a simple approach to integrate reputation mechanisms into the process of service selection.

Integrating reputation mechanisms into the discovery process allows to filter out services that have a bad reputation and to rank matching services according to their reputation. Hence, we will provide a wrapper to the discovery component that first forwards a query to the standard discovery component and afterwards obtains the reputation of the discovered services by accessing the reputation web service. Based on the reputation of the discovered services, certain services may be removed from the result (if the reputation is below a given threshold) or the order of the discovered services in the result may be changed according to reputation (services with higher reputation come first). Figure 8.1 illustrates our approach.

This approach has the advantage that it does not require any changes to the classical integrated discovery and composition architecture. The reputation mechanisms are well encapsulated within the reputation web service. The discovery and composition components are fully functional without the reputation web service, which can be integrated by simply installing the aforementioned wrapper for the discovery component.

Chapter 9

Conclusion

This report gave an overview of the state-of-the-art concerning reputation mechanisms, investigated a reputation mechanism suitable in a semantic service-oriented environment as well as methods for stimulating honest reporting.

In an open environment where semantic web services are discovered and composed on the fly, malicious parties may advertise false service capabilities. The use of reputation services is a promising approach to mitigate such attacks. Misbehaving services receive a bad reputation (reported by disappointed clients) and will be avoided by other clients. Reputation mechanisms help to improve the global efficiency of the overall system because they reduce the incentive to cheat [9]. Studies show that buyers seriously take into account the reputation of the seller when placing their bids in online auctions [24]. Moreover, it has been proven that in certain cases reputation mechanisms can be designed in such a way that it is in every party's interest to report correct reputation information (incentive compatible reputation services) [27]. Besides, reputation mechanisms can be implemented in a secure way [25].

In a service-oriented environment providers may prefer to provider a service with different production quality levels. Most of the existing reputation mechanisms (eBay, Slashdot, Amazon) tend to act by social exclusion, i.e. separating trustworthy and untrustworthy agents and so they are not suitable for this environment. We have shown a very simple reputation mechanism [31] where repeated failures do not automatically exclude a provider from the market, but rather influence the price the provider can charge for a future service. However in order the system to work, this mechanism requires that agents report honestly, which it is not likely to happen naturally in an environment with self-interested agents. We have analyzed three incentive-compatible reputation mechanisms [39], [25]), [28] and shown what can be done in order to encourage rational agents to report truthfully. We have presented also an alternative where instead of giving incentives for truthful reporting, false reports are filtered out.

Finally, considering the importance of reputation services in open environments, it is essential that service discovery and composition algorithms intended to operate in such environments those exploit these reputation services in order to favor services with a high reputation. We define the interfaces of our reputation mechanism, provided as a web service, and an implementation will be part of the prototype developed in D2.4.6.2.

Bibliography

[1] A. Abdul-Rahman and S. Hailes. Supporting Trust in Virtual Communities. In *Proceedings Hawaii International Conference on System Sciences*, Maui, Hawaii, 2000.

[2] K. Aberer. P-Grid: A self-organizing access structure for P2P information systems. *Lecture Notes in Computer Science*, 2172, 2001.

[3] K. Aberer and Z. Despotovic. Managing Trust in a Peer-2-Peer Information System. In *Proceedings of the Ninth International Conference on Information and Knowledge Management (CIKM)*, 2001.

[4] Agentcities. http://www.agentcities.net.

[5] M. Bacharach. How Human Trusters Assess Trustworthiness in Quasi-Virtual Contexts. In *Proceedings of the AAMAS Workshop on Trust Deception and Fraud*, Bologna, Italy, 2002.

[6] S. Barber and J. Kim. Belief Revision Process Based on Trust: Agents Evaluating Reputation of Information Sources. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 73–82. Springer-Verlag, Berlin Heidelberg, 2001.

[7] T. Beth, M. Borcherding, and B. Klein. Valuation of Trust in Open Networks. In *Proceedings of the European Symposium on Research in Computer Security (ESORICS)*, pages 3–18, Brighton, UK, 1994. Sprinter-Verlag.

[8] A. Birk. Boosting Cooperation by Evolving Trust. *Applied Artificial Intelligence*, 14:769–784, 2000.

[9] 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.

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

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

[12] S. Braynov and T. Sandholm. Auctions with Untrustworthy Bidders. In *Proceedings of the IEEE Conference on E-Commerce*, Newport Beach, CA, USA, 2003.

[13] S. Buchegger and J.-I. L. Boudec. The effect of rumour spreading in reputation systems for mobile ad-hoc networks. In *Proceedings of WiOpt '03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Sophia-Antipolis, France, Mar. 2003.

[14] C. Castelfranchi and R. Falcone. Trust and Control: A Dialectic Link. *Applied Artificial Intelligence*, 14:799–823, 2000.

[15] C. Dellarocas. Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behaviour. In *Proceedings of the 2nd ACM conference on Electronic Commerce*, Minneapolis, MN, 2000.

[16] C. Dellarocas. The Design of Reliable Trust Management Systems for Electronic Trading Communities. Working Paper, MIT, 2001.

[17] 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.

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

[19] C. Dellarocas. Reputation Mechanism Design in Online Trading Environments with Pure Moral Hazard. *Information Systems Research*, 16(2), 2005.

[20] Z. Despotovic and K. Aberer. Trust-aware delivery of composite goods. In *AP2PC*, pages 57–68, 2002.

[21] R. Falcone and C. Castelfranchi. The Socio-cognitive Dynamics of Trust: Does Trust create Trust. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 55–72. Springer-Verlag, Berlin Heidelberg, 2001.

[22] E. Friedman and P. Resnick. The Social Cost of Cheap Pseudonyms. *Journal of Economics and Management Strategy*, 10(2):173–199, 2001.

[23] D. Fudenberg and D. Levine. Reputation and Equilibrium Selection in Games with a Patient Player. *Econometrica*, 57:759–778, 1989.

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

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

[26] R. Jurca and B. Faltings. Towards Incentive-Compatible Reputation Management. In R. Falcone, R. Barber, L. Korba, and M. Singh, editors, *Trust, Reputation and Security: Theories and Practice*, volume LNAI 2631, pages 138 – 147. Springer-Verlag, Berlin Heidelberg, 2003.

[27] 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.

[28] R. Jurca and B. Faltings. "CONFESS". Eliciting Honest Feedback without Independent Verification Authorities. In *Sixth International Workshop on Agent Mediated Electronic Commerce (AMEC VI 2004)*, New York, USA, July 19 2004.

[29] R. Jurca and B. Faltings. Truthful reputation information in electronic markets without independent verification. Technical Report ID: IC/2004/08, EPFL, http://ic2.epfl.ch/publications, 2004.

[30] R. Jurca and B. Faltings. Eliminating Undesired Equilibrium Points from Incentive Compatible Reputation Mechanisms. Technical Report ID: IC/2005/024, EPFL, 2005.

[31] R. Jurca and B. Faltings. Reputation-based Pricing of P2P Services. In *Proceedings of the Wokshop on Economics of P2P Systems*, Philadelphia, USA, 2005.

[32] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. EigenRep: Reputation management in P2P networks. In *Proceedings of the 12th International World Wide Web Conference*, Budapest, Hungary, May 2003.

[33] 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.

[34] 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.

[35] D. M. Kreps and R. Wilson. Reputation and Imperfect Information. *Journal of Economic Theory*, 27:253–279, 1982.

[36] H. McKnight and N. Chervany. Trust and Distrust: One Bite at a Time. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 27–54. Springer-Verlag, Berlin Heidelberg, 2001.

[37] P. Milgrom and J. Roberts. Predation, Reputation and Entry Deterrence. *J. Econ. Theory*, 27:280–312, 1982.

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

[39] N. Miller, P. Resnick, and R. Zeckhauser. Eliciting Informative Feedback: The Peer-Prediction Method. Forthcoming in Management Science, 2005.

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

[41] L. Mui, M. Mohtashemi, and A. Halberstadt. A Computational Model of Trust and Reputation. In *Proceedings of the 35th Hawaii International Conference on System Sciences (HICSS)*, 2002.

[42] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA, Nov. 1998.

[43] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. Reputation systems. *Communications of the ACM*, 43(12):45–48, Dec. 2000.

[44] P. Resnick and R. Zeckhauser. 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.

[45] M. Richardson, P. Domingos, and R. Agrawal. Trust management for the semantic web. In *Proceedings of the Second International Semantic Web Conference*, pages 351–368, Sanibel Island, FL, USA, Sept. 2003.

[46] T. W. Sandholm. *Negotiation among self-interested computationally limited agents*. PhD thesis, University of Massachusetts at Amherst, 1996.

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

[48] K. M. Schmidt. Reputation and Equilibrium Characterization in Repeated Games with Conflicting Interests. *Econometrica*, 61:325–351, 1993.

[49] R. Selten. The Chain-Store Paradox. *Theory and Decision*, 9:127–159, 1978.

[50] S. Sen and N. Sajja. Robustness of Reputation-based Trust: Boolean Case. In *Proceedings of the AAMAS*, Bologna, Italy, 2002.

[51] N. Sokey and R. Lucas. *Recursive Methods in Economic Dynamics*. Harvard University Press, 1989.

[52] M. Witkowski, A. Artikis, and J. Pitt. Experiments in building Experiential Trust in a Society of Objective-Trust Based Agents. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume LNAI 2246, pages 111–132. Springer-Verlag, Berlin Heidelberg, 2001.

[53] L. Xiong and L. Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans. Knowl. Data Eng.*, 16(7):843–857, 2004.

[54] B. Yu and M. Singh. A Social Mechanism of Reputation Management in Electronic Communities. In *Proceedings of the Forth International Workshop on Cooperative Information Agents*, pages 154–165, 2000.

[55] B. Yu and M. Singh. An Evidential Model of Distributed Reputation Management. In *Proceedings of the AAMAS*, Bologna, Italy, 2002.

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

[57] G. Zacharia, A. Moukas, and P. Maes. Collaborative Reputation Mechanisms in Electronic Marketplaces. In *Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS)*, 1999.